SIEMENS

SIMATIC
S7
S7-1200 otomasyon sistemi

Sistem kitabı

Ürüne genel bakış 1

Yeni fonksiyonlar 2

STEP 7 programlama yazılımı 3

Montaj 4

PLC ile ilgili temel esaslar 5

Cihaz konfigürasyonu 6

Programlama konseptleri 7

Yordamlar 8

Gelişmiş yordamlar 9

Teknoloji yordamları 10

İletişim 11

Web sunucusu 12

İletişim işlemcisi ve Modbus TCP 13

TeleService iletişimi (SMTP ile e-posta iletişimi) 14

Çevrimiçi araçlar ve teşhis araçları 15

Devamı bir sonraki sayfada

V4.6 11/2022
A5E52544994-AP

S7
S7-1200 otomasyon sistemi

Sistem kitabı

Devam

Teknik veriler A

Güç bilançosunun hesaplanması B

Sipariş bilgileri C

Cihaz değişimi ve yedek parça uyumluluğu D

Yasal bilgi ve uyarılar

İkaz bilgisi konsepti

Bu kullanma kılavuzu, kendi güvenliğiniz ve mal kaybı veya zarar oluşmasını önlemek için dikkat etmeniz gereken bilgi ve uyarılar içermektedir. Kişisel güvenliğiniz ile ilgili bilgi ve uyarılar bir ikaz üçgeni ile belirtilmiştir, genel mal hasarı ile ilgili bilgi ve uyarılar için ise ikaz üçgeni kullanılmamıştır. Tehlike kademesine bağlı olarak, ikaz bilgi ve uyarıları, en önemliden daha az önemlilere göre, aşağıdaki şekilde sıralanmıştır.

⚠ TEHLİKE
eğer bildirilen dikkat etme önlemlerine uyulmazsa, ölüm veya ağır yaralanma **olacağı** anlamına gelir.

⚠ İKAZ
eğer bildirilen dikkat etme önlemlerine uyulmazsa, ölüm veya ağır yaralanma **olabileceği** anlamına gelir.

⚠ DİKKAT
eğer bildirilen dikkat etme önlemlerine uyulmazsa, hafif yaralanma olabileceği anlamına gelir.

DİKKAT
eğer bildirilen dikkat etme önlemlerine uyulmazsa, mal hasarı olabileceği anlamına gelir.

Birden fazla tehlike derecesinin aynı anda ortaya çıkması halinde, en yüksek tehlike derecesine ait uyarı bilgisi kullanılır. Eğer ikaz üçgenli bir uyarı bilgisinde insanlara zarar gelebileceği hususuna dikkat çekiliyorsa, aynı ikaz bilgisine ayrıca bir mal hasarı ile ilgili uyarı da eklenmiş olabilir.

Yetkili personel

Bu dokümantasyon içinde açıklanan ürünü/sistemi sadece ilgili görev için uygun nitelikte olan **kalifiye personel** kullanabilir. Ürünün/Sistemin kullanımı esnasında ilgili göreve ilişkin dokümantasyona ve özellikle bu dokümantasyon içinde belirtilen güvenlik ve uyarı bilgilerine dikkat edilecektir. Kalifiye personel, gerekli eğitime ve deneyime sahip olduğundan bu ürünleri/sistemleri kullanırken riskleri fark edebilecek ve olası tehlikeleri önleyebilecek bilgiye sahiptir.

Siemens ürünlerinin amaca uygun kullanımı

Lütfen şunlara dikkat ediniz:

⚠ İKAZ
Siemens ürünleri sadece katalogda ve ilgili teknik dokümantasyonda öngörülmüş kullanım durumları için kullanılmalıdır. Eğer yabancı ürünler ve yabancı bileşenler kullanılırsa, bu ürün ve bileşenler Siemens tarafından tavsiye edilmiş ya da kullanımına izin verilmiş olmalıdır. Ürünlerin kusursuz ve güvenli kullanımı için, gerektiği şekilde taşınması, gerektiği şekilde depolanması, yerleştirilmesi, montajı, kurulması, devreye sokulması, kullanılması ve muhafaza edilmesi ya da onarılması şarttır. İzin verilen çevre koşullarına uyulmalıdır. İlgili dokümantasyonlarda verilen bilgi ve uyarılara dikkat edilmelidir.

Markalar

Tescil ibaresi ® ile işaretlenmiş tüm isim ve tanımlar, tescil edilmiş Siemens AG markalarıdır. Bu yazıdaki diğer isim ve tanımlar, üçüncü kişiler tarafından kendi amaçları için kullanılmaları halinde sahiplerinin haklarına tecavüz edilmiş olması söz konusu olabilecek markalar olabilir.

Sorumluluk üstlenmeme mesuliyeti

Bu yazının içeriğini, tarif edilen donanım ve yazılıma uygunluğu açısından kıyasladık. Yine de farklılıklar ve sapmalar olabilir ve bu nedenle tamamen uyumluluk hususunda herhangi bir sorumluluk üstlenmiyoruz ve garanti vermiyoruz. Bu yazıda verilen bilgiler muntazam aralıklar ile kontrol edilmektedir ve gerekli düzeltmeler yazının müteakip baskılarına işlenmektedir.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
ALMANYA

A5E52544994-AP
Ⓟ 05/2023 Değişiklikler yapma hakkımız saklıdır

Copyright © Siemens AG 2022.
Tüm Hakları Saklıdır

Önsöz

İşletme kitabının amacı

S7-1200 ailesi, geniş bir yelpazedeki otomasyon görevlerini çözmek için kullanabileceğiniz çeşitli programlanabilir kontrol sistemlerini (PLC) kapsamaktadır. S71200; kompakt tasarımı, genişletme olanağı, uygun fiyatı ve güçlü komut seti sayesinde çok sayıda kumanda uygulaması için mükemmel derecede uygundur. S7-1200'ün modelleri ve Windows'taki çeşitli STEP 7 programlama olanakları (Sayfa 34), otomasyon çözümlerinizi uygularken son derece yüksek bir esneklik sağlar.

Bu işletme kitabı, S7-1200 kumanda ünitelerinin kurulumu ve programlanması ile ilgili bilgiler sunar ve otomasyon sistemleri hakkında genel bilgiye sahip mühendislere, programcılara ve bakım personeline yöneliktir.

Gerekli temel bilgiler

Bu işletme kitabıyla çalışabilmeniz için, otomasyon ve programlanabilir kontrol sistemleri alanında genel bilgilere ihtiyacınız bulunmaktadır.

İşletme kitabının kapsamı

Bu işletme kitabında aşağıdaki ürünler açıklanmaktadır:

- STEP 7 Basic ve Professional (Sayfa 34)
- S7-1200 CPU Firmware Sürüm V4.6

S71200 ürünlerinin eksiksiz bir listesini, Teknik veriler (Sayfa 1160) kısmında bulabilirsiniz.

Sertifikalandırmalar ve ruhsatlar

Ayrıntılı bilgileri, Teknik veriler (Sayfa 1160) kısmında bulabilirsiniz.

Servis ve destek

Dokümantasyon desteğimize ek olarak, Siemens, internette Müşteri desteği web sitesinde (http://support.industry.siemens.com) teknik bilgiler sunar.

Teknik sorularınız olursa, bir eğitime ihtiyaç duyarsanız veya S7 ürünleri sipariş etmek isterseniz lütfen Siemens temsilcinizle iletişime geçin. Teknik eğitimli satış personeli, kullanım olanakları ve proseslerin yanı sıra çeşitli Siemens ürünleri hakkında çok özel bilgilere sahiptir ve bu nedenle sorunlarla karşılaştığınızda size en hızlı ve en iyi şekilde yardımcı olabilir.

Dokümantasyon ve bilgi

S7-1200 ve STEP 7 teknik bilgi içeren çok sayıda dokümantasyon ve başka kaynaklar sunmaktadır.

- S7-1200 otomasyon sistemi adlı sistem kitabında, tüm S7-1200 ürün ailesinin çalışma şekli, programlanması ve teknik verileriyle ilgili özel bilgiler sunulmaktadır.
  Sistem kitabı, elektronik olarak (PDF formatında) mevcuttur. Bunu ve diğer elektronik işletme kitaplarını, Siemens Industry Online Support web sitesi (http://support.industry.siemens.com) üzerinde indirebilir veya görüntüleyebilirsiniz. Sistem kitabı ayrıca, her S7-1200 CPU'nun teslim kapsamına ait doküman CD'sinde de mevcuttur.
- STEP 7'nin çevrimiçi bilgi sistemi, programlama paketinin çalışma şekli ve fonksiyonları ile SIMATIC CPU'nun temel operasyon şekline ait konsept bilgileri ve özel notlara doğrudan erişim olanağı sunar.
- Siemens Industry Online Support web sitesi (http://support.industry.siemens.com), STEP 7'nin sistem kitabını ve bilgi sistemini kapsayan SIMATIC dokümantasyon setinin elektronik sürümlerine (PDF) erişim olanağı sunar. Mevcut dokümanları, Ürün desteği linkinin altında bulabilirsiniz. Çevrimiçi dokümantasyona sağlanan bu erişim aracılığıyla, sürükle ve bırak yöntemini kullanarak farklı dokümanlardaki konuları da kendiniz düzenleyebilir ve bu sayede kullanıcıya özel işletme kitapları oluşturabilirsiniz. Daha önce yayınlanan sistem kitaplarının güncellemelerine de Siemens Industry Online Support'ta ulaşılabilir.
  Çevrimiçi dokümantasyonu, bu web sitesinin sol tarafındaki "mySupport" linki aracılığıyla çağırabilirsiniz. Ardından "Dokümantasyon" seçeneğine tıklayın. mySupport kısmındaki dokümantasyon fonksiyonunu kullanabilmek için kayıtlı kullanıcı olarak oturum açmanız gerekir.
- Siemens Industry Online Support web sitesinde ayrıca S7-1200 ve STEP 7 ile ilgili SSS ve başka faydalı dokümanlar sunulmaktadır.
- Servis ve destek bölümündeki teknik forumda (https://support.industry.siemens.com/tf/ww/en/?Language=en&siteid=csius&treeLang=en&groupid=4000002&extranet=standard&viewreg=WW&nodeid0=34612486), ayrıca ürünlerle ilgili sohbetleri takip edebilir veya bu sohbetlere katılabilirsiniz. Aşağıdaki forumlar, size çeşitli ürün uzmanlarıyla iletişime geçme fırsatı sunmaktadır.
  - S7-1200 (https://support.industry.siemens.com/tf/ww/en/threads/237?title=simatic-s7-1200&skip=0&take=10&orderBy=LastPostDate+desc) ile ilgili forum
  - STEP 7 Basic (https://support.industry.siemens.com/tf/ww/en/threads/243?title=step-7-tia-portal&skip=0&take=10&orderBy=LastPostDate+desc) ile ilgili forum

Güvenlik uyarıları

Siemens; tesislerin, sistemlerin, makinelerin ve bilişim ağlarının güvenli bir şekilde çalışmasını destekleyen Industrial Security fonksiyonlarına sahip ürünler ve çözümler sunar.

Tesisleri, sistemleri, makineleri ve bilişim ağlarını siber tehditlere karşı korumak için bütüncül, en son teknoloji ile donatılmış bir Industrial Security konseptinin uygulanması (ve sürdürülmesi) gereklidir. Siemens ürünleri ve çözümleri bu tür bir konseptin bileşenleri olarak görev yapar.

Müşteriler; kendi tesislerine, sistemlerine, makinelerine ve bilişim ağlarına yetkisiz erişimi önlemekle yükümlüdür. Bu sistemler, makineler ve bileşenlerin, yalnızca gerekli olduğunda, gerekli olduğu kadar ve yalnızca yeterli koruma önlemlerinin (örn. güvenlik duvarları ve/veya ağ segmentasyonu) alınmış olması durumunda, kurum içi ağa veya İnternete bağlanmasına izin verilmelidir.

Industrial Security alanındaki olası koruma önlemlerine ilişkin daha fazla bilgiyi burada (https://www.siemens.com/industrialsecurity) bulabilirsiniz.

Daha da güvenli hale getirmek için, Siemens ürünleri ve çözümleri sürekli olarak geliştirilmektedir. Siemens, ürün güncellemelerinin sunuldukları anda hemen uygulanmasını ve yalnızca güncel ürün versiyonlarının kullanılmasını özellikle önerir. Güncel olmayan veya artık desteklenmeyen versiyonların kullanımı, siber tehdit riskini artırabilir.

Ürün güncellemeleri hakkında her zaman bilgilendirilmek için, buradan (https://www.siemens.com/cert) Siemens Industrial Security RSS Feed'e abone olun.

Tesisinizin çalışma güvenliğinin korunması

Tesisinizin çalışma güvenliğinin korunması

NOT

Tesisinizin çalışma güvenliğinin korunmasıyla ilgili önemli not

Güvenlikle ilgili fonksiyonlara sahip tesisler, kullanıcı tarafından sağlanan çalışma güvenliği açısından özel gerekliliklere tabidir. Tedarikçiler bile ürün denetimi sırasında özel önlemlere uymak zorundadır. Bu nedenle, güvenlik perspektifinden bakıldığında sistemlerin işletimiyle ilgili olan (veya olabilecek) ürün geliştirmeleri ve performans özellikleri hakkında sizi kişisel bildirimlerle bilgilendiriyoruz.

İlgili bildirimlere abone olarak, bilgilerinizin her zaman güncel kaldığından ve gerektiğinde sisteminizde değişiklikler yapabileceğinizden emin olabilirsiniz.

Industry Online Support'ta oturum açın. Aşağıdaki linkleri takip edin ve sayfada farenin sağ tuşuyla "Güncelleme durumunda e-posta" ifadesine tıklayın:

- SIMATIC S7-300/S7-300F (https://support.industry.siemens.com/cs/ww/en/ps/13751)
- SIMATIC S7-400/S7-400H/S7-400F/FH (https://support.industry.siemens.com/cs/ww/en/ps/13828)
- SIMATIC S7-1500 kontrolörü CPU yazılımı (https://support.industry.siemens.com/cs/ww/en/view/109811764)
- SIMATIC S7-1500/SIMATIC S7-1500F (https://support.industry.siemens.com/cs/ww/en/ps/13716)
- SIMATIC S7-1200/SIMATIC S7-1200F (https://support.industry.siemens.com/cs/ww/en/ps/13683)
- Desantral çevrebirim (https://support.industry.siemens.com/cs/ww/en/ps/14029)
- STEP 7 TIA Portal (https://support.industry.siemens.com/cs/ww/en/ps/14667)

İçindekiler

1 Ürüne genel bakış 27

1.1 S7-1200 PLC'ye giriş 27

1.2 CPU'nun genişleme özelliği 31

1.3 HMI panelleri 32

2 Yeni fonksiyonlar 33

3 STEP 7 programlama yazılımı 34

3.1 Farklı görünümlerle kolay çalışma 35

3.2 STEP 7 ile S7-1200 arasında uyumluluk 36

4 Montaj 38

4.1 S71200 cihazlarının montajıyla ilgili direktifler 38

4.2 Güç bilançosu 40

4.3 Takma ve sökme sırasında izlenecek prosedür 42
4.3.1 S7-1200 cihazları için sabitleme ölçüleri 42
4.3.2 CPU'nun takılması ve sökülmesi 45
4.3.3 Bir SB'nin, CB'nin veya BB'nin takılması ve sökülmesi 47
4.3.4 Bir SM'nin takılması ve sökülmesi 48
4.3.5 Bir CM veya CP'nin takılması ve sökülmesi 50
4.3.6 S7-1200 klemens bloğunun sökülmesi ve takılması 51
4.3.7 Uzatma kablosunun takılması ve sökülmesi 52

4.4 Kablaj direktifleri 54

5 PLC ile ilgili temel esaslar 60

5.1 Kullanıcı programının yürütülmesi 60
5.1.1 CPU'nun işletim durumları 64
5.1.2 RUN işletim durumunda çevrimin işlenmesi 68
5.1.3 Organizasyon blokları (OB'ler) 68
5.1.3.1 Program çevrimi OB'si 69
5.1.3.2 Yol alma OB 69
5.1.3.3 Gecikme alarmı OB'si 69
5.1.3.4 Uyandırma alarmı OB'si 70
5.1.3.5 Proses alarmı OB'si 71
5.1.3.6 Zaman hatası OB'si 72
5.1.3.7 Teşhis hatası OB'si 73
5.1.3.8 "Modüllerin çıkartılması veya takılması" OB'si 75
5.1.3.9 Modül taşıyıcısı veya istasyon hatası OB'si ("Rack or station failure OB") 76
5.1.3.10 Saat OB'si 76
5.1.3.11 Durum OB'si 77
5.1.3.12 Güncelleme OB'si 77
5.1.3.13 Profil OB'si 78

5.1.3.14 OB MC Servo ve MC-Interpolator ... 78
5.1.3.15 MC-PreServo ... 79
5.1.3.16 MC-PostServo ... 80
5.1.3.17 Olayların yürütülmesiyle ilgili öncelikler ve bekleme kuyruğu ... 81
5.1.4 Çevrim süresinin denetlenmesi ve yapılandırılması ... 84
5.1.5 CPU belleği ... 86
5.1.5.1 Sistem ve zamanlama markörü ... 88
5.1.6 Teşhis tamponu ... 91
5.1.7 Gerçek zaman saati ... 92
5.1.8 Çıkışların RUN işletim durumundan STOP işletim durumuna geçiş için yapılandırılması ... 92

5.2 Veri belleği, bellek alanları, G/Ç ve adresleme ... 93
5.2.1 S7-1200'ün verilerine erişim ... 93

5.3 Analog değerlerin işlenmesi ... 98

5.4 Veri tipleri ... 99
5.4.1 Bool, Bayt, Word ve DWord veri tipleri ... 100
5.4.2 Tamsayı veri tipleri ... 101
5.4.3 Kayar noktalı sayı/gerçek sayı veri tipleri ... 101
5.4.4 Saat ve tarih veri tipleri ... 102
5.4.5 Karakter ve karakter dizisi veri tipleri ... 104
5.4.6 ARRAY veri tipi ... 106
5.4.7 Struct veri tipi ... 106
5.4.8 PLC veri tipi ... 107
5.4.9 "Variant" işaretçi veri tipi ... 107
5.4.10 Bir değişken veri tipine ait bir "Slice"a erişim ... 108
5.4.11 Bir AT bindirmesi ile bir değişkene erişim ... 110

5.5 Memory card'ın kullanılması ... 112
5.5.1 Memory card'ın CPU'ya takılması ... 113
5.5.2 Projenin memory card'a kopyalanmadan önce, CPU'nun yol alma parametrelerinin yapılandırılması ... 115
5.5.3 Aktarım kartı ... 116
5.5.4 Program kartı ... 119
5.5.5 Memory card'ın, gizli PLC konfigürasyon verilerini korumak için kullanılması ... 122
5.5.6 Firmware güncellemesi ... 124

5.6 Şifre kaybedildiğinde izlenecek prosedür ... 127

6 Cihaz konfigürasyonu ... 128

6.1 Bir CPU ekleme ... 129

6.2 Konfigürasyonun bağlı bir CPU'dan yüklenmesi ... 131

6.3 Konfigürasyona modüller ekleme ... 133

6.4 Konfigürasyon kumandası ... 134
6.4.1 Konfigürasyon kumandasının avantajları ve kullanımı ... 134
6.4.2 Merkezi yapının ve opsiyonel modüllerin yapılandırılması ... 134
6.4.3 Konfigürasyon kumandasıyla ilgili örnek ... 141

6.5 Bir cihazın değiştirilmesi ... 144

6.6 CPU işletiminin yapılandırılması ... 145

6.6.1 CPU özellikleri ... 145
6.6.2 Entegre G/Ç'nin yapılandırılması ... 147
6.6.3 Dijital girişler için filtre sürelerinin ayarlanması ... 148
6.6.4 Darbe toplama ... 150
6.7 Çok dilli desteğin yapılandırılması ... 151
6.8 Koruma ve dijital güvenlik ... 153
6.8.1 Dijital güvenlik asistanında PLC dijital güvenlik ayarlarının belirlenmesi ... 153
6.8.2 Gizli PLC konfigürasyon verilerinin korunması ... 154
6.8.3 CPU için erişim koruması ... 156
6.8.4 Bağlantı mekanizmalarının yapılandırılması ... 158
6.8.4.1 Uzak partnerler için erişim mekanizmasının belirlenmesi ... 158
6.8.4.2 Güvenli PG/PC iletişimi ve HMI iletişimini etkinleştirme ve sertifikalar oluşturma ... 158
6.8.5 Harici yükleme belleği ... 159
6.8.6 Teknik bilgi koruması ... 160
6.8.7 Yazma koruması ... 161
6.8.8 Kopyalama koruması ... 161
6.9 Modül parametrelerinin yapılandırılması ... 162
6.10 CPU'nun iletişim için yapılandırılması ... 164
6.11 Saat senkronizasyonu ... 167

7 Programlama konseptleri ... 169

7.1 PLC cihazı ile bir otomasyon çözümü tasarlamak için kullanılan direktifler ... 169
7.2 Kullanıcı programınızın yapılandırılması ... 170
7.3 Programınızın yapılandırılmasına yönelik blokların kullanımı ... 172
7.3.1 Organizasyon bloğu (OB) ... 173
7.3.2 Fonksiyon (FC) ... 174
7.3.3 Fonksiyon bloğu (FB) ... 175
7.3.4 Veri bloğu (DB) ... 176
7.3.5 Yeniden kullanılabilen kod bloklarının oluşturulması ... 178
7.3.6 Parametrelerin bloklara aktarılması ... 179
7.4 Veri tutarlılığı ... 181
7.5 Programlama dili ... 182
7.5.1 Kontak planı (LAD) ... 183
7.5.2 Fonksiyon planı (FBD) ... 184
7.5.3 SCL ... 184
7.5.3.1 SCL programlama editörü ... 185
7.5.3.2 SCL ifadeleri ve işlemleri ... 186
7.5.3.3 PEEK ve POKE yordamlarıyla indekslenmiş adresleme ... 190
7.5.4 LAD, FBD ve SCL'deki EN ve ENO ... 192
7.6 Program elemanlarının yüklenmesi ... 194
7.7 Çevrimiçi CPU'nun ve çevrimdışı projenin senkronize edilmesi ... 197
7.8 Çevrimiçi CPU'dan yükleme ... 199
7.8.1 Çevrimiçi CPU'nun çevrimdışı CPU ile karşılaştırması ... 199

7.9 Programın hatalarının ayıklanması ve test edilmesi ... 200
7.9.1 CPU'daki verilerin izlenmesi ve kumanda edilmesi ... 200
7.9.2 İzleme tabloları ve Force tabloları ... 201
7.9.3 Kullanımı görüntülemek için çapraz referans ... 202
7.9.4 Çağrı hiyerarşisinin kontrolü için kullanılan çağrı yapısı ... 203

8 Yordamlar ... 204

8.1 Bit bağlantıları ... 204
8.1.1 Bit bağlantısı yordamları ... 204
8.1.2 Ayarlama ve sıfırlama işlemleri ... 207
8.1.3 Yükselen kenar ve alçalan kenar işlemleri ... 209
8.2 Zamanların çalışma şekli ... 211
8.3 Sayaçların çalışma şekli ... 218
8.4 Karşılaştırıcıların çalışma şekli ... 224
8.4.1 Karşılaştırma işlemleri ... 224
8.4.2 IN_Range (değer, aralık dahilinde) ve OUT_Range (değer, aralık dışında) ... 225
8.4.3 OK (geçerliliği kontrol et) ve NOT_OK (geçersizliği kontrol et) ... 225
8.4.4 Varyant ve Array karşılaştırma işlemleri ... 226
8.4.4.1 Eşittir ve Eşit değildir karşılaştırma işlemleri ... 226
8.4.4.2 Sıfır karşılaştırma işlemleri ... 227
8.4.4.3 IS_ARRAY (ARRAY açısından kontrol et) ... 227
8.5 Aritmetik fonksiyonlar ... 228
8.5.1 CALCULATE (hesapla) ... 228
8.5.2 Topla, Çıkar, Çarp ve Böl yordamları ... 229
8.5.3 MOD (bir bölme işleminin kalanı) ... 230
8.5.4 NEG (ikili tümleyen oluştur) ... 230
8.5.5 INC (arttır) ve DEC (azalt) ... 231
8.5.6 ABS (mutlak değer oluştur) ... 232
8.5.7 MIN (minimumu çağır) ve MAX (maksimumu çağır) ... 232
8.5.8 LIMIT (limiti ayarla) ... 233
8.5.9 Üs, logaritma ve trigonometri yordamları ... 234
8.6 Verilerin aktarılması için kullanılan yordamlar ... 236
8.6.1 MOVE (Değeri kopyala), MOVE_BLK (Aralığı kopyala), UMOVE_BLK (Aralığı kesintisiz olarak kopyala) ve MOVE_BLK_VARIANT (Aralığı kopyala) ... 236
8.6.2 Deserialize ... 239
8.6.3 Serialize ... 241
8.6.4 FILL_BLK (Belleği Bit örüntüsüyle doldur) ve UFILL_BLK (Belleği kesintisiz şekilde Bit örüntüsüyle doldur) ... 244
8.6.5 SCATTER ... 245
8.6.6 SCATTER_BLK ... 248
8.6.7 GATHER ... 252
8.6.8 GATHER_BLK ... 255
8.6.9 SWAP (sırayı değiştir) ... 259
8.6.10 Belleği oku / belleğe yaz yordamları ... 260
8.6.10.1 PEEK ve POKE (sadece SCL) ... 260
8.6.10.2 Big ve Little Endian formatını oku ve yaz (SCL) yordamları ... 262
8.6.11 Variant yordamları ... 263
8.6.11.1 VariantGet (VARIANT değişken değerini oku) ... 263
8.6.11.2 VariantPut (VARIANT değişken değerini yaz) ... 264
8.6.11.3 CountOfElements (ARRAY elemanlarının sayısını sorgula) ... 265

8.6.12 Array yordamları ... 266
8.6.12.1 LOWER_BOUND: (Alt ARRAY sınırını oku) ... 266
8.6.12.2 UPPER_BOUND: (Üst ARRAY sınırını oku) ... 268
8.6.13 Daha eski sistemlerdeki yordamlar ... 270
8.6.13.1 FieldRead (Alanı oku) ve FieldWrite (Alana yaz) yordamları ... 270

8.7 Dönüştürme işlemleri ... 272
8.7.1 CONV (değeri dönüştür) ... 272
8.7.2 SCL'deki dönüştürme yordamları ... 273
8.7.3 ROUND (sayıyı yuvarla) ve TRUNC (tamsayı oluştur) ... 275
8.7.4 CEIL ve FLOOR (kayar noktalı sayıdan bir sonraki daha yüksek veya bir sonraki daha düşük tamsayıyı oluştur) ... 276
8.7.5 SCALE_X (ölçekle) ve NORM_X (standartlaştır) ... 277
8.7.6 Varyant dönüştürme yordamları ... 280
8.7.6.1 VARIANT_TO_DB_ANY (VARIANT'ı DB_ANY'ye dönüştür) ... 280
8.7.6.2 DB_ANY_TO_VARIANT (DB_ANY'yi VARIANT'a dönüştür) ... 281

8.8 Program kumandası işlemleri ... 282
8.8.1 JMP (VKE = 1'de atla), JMPN (VKE = 0'da atla) ve Label (atlama işareti) yordamları ... 282
8.8.2 JMP_LIST (atlama listesini tanımla) ... 283
8.8.3 SWITCH (atlama dağıtma) ... 284
8.8.4 RET (geri dönüş değeri) ... 285
8.8.5 ENDIS_PW (CPU şifresini etkinleştir/devre dışı bırak) ... 286
8.8.6 RE_TRIGR (çevrim denetleme süresini yeniden başlat) ... 289
8.8.7 STP (Programı sonlandır) ... 290
8.8.8 GET_ERROR ve GET_ERROR_ID (Hatayı yerel olarak çağır ve Hata ID'sini yerel olarak çağır) yordamları ... 291
8.8.9 RUNTIME (program çalışma zamanını ölç) ... 294
8.8.10 SCL'deki program kumanda yordamları ... 296
8.8.10.1 IF-THEN ... 297
8.8.10.2 CASE ... 298
8.8.10.3 FOR ... 299
8.8.10.4 WHILE-DO ... 300
8.8.10.5 REPEAT-UNTIL ... 301
8.8.10.6 CONTINUE ... 301
8.8.10.7 EXIT ... 302
8.8.10.8 GOTO ... 303
8.8.10.9 RETURN ... 303

8.9 Sözcük bağlantısı ... 304
8.9.1 AND (VE), OR (VEYA) ve XOR (ÖZEL VEYA) bağlantı işlemleri ... 304
8.9.2 INV (tekli tümleyen oluştur) ... 304
8.9.3 DECO (şifreyi çöz) ve ENCO (şifrele) yordamları ... 305
8.9.4 SEL (seç), MUX (çokla) ve DEMUX (çoklamayı çöz) yordamları ... 306

8.10 Kaydır ve döndür ... 309
8.10.1 SHR (sağa kaydır) ve SHL (sola kaydır) yordamları ... 309
8.10.2 ROR (sağa döndür) ve ROL (sola döndür) yordamları ... 310

9 Gelişmiş yordamlar ... 311

9.1 Tarih, saat ve saat fonksiyonları ... 311
9.1.1 Tarih ve saat yordamları ... 311
9.1.2 Saat fonksiyonları ... 314
9.1.3 SET_TIMEZONE (zaman dilimini ayarla) ... 316

9.1.4 RTM (işletim saati sayacı) 318

9.2 Karakter dizileri ve karakter yordamları 320
9.2.1 String veri tipi 320
9.2.2 S_MOV (karakter dizisini kaydır) 320
9.2.3 Karakter dizilerinin dönüştürülmesi için kullanılan yordamlar 321
9.2.3.1 S_CONV, STRG_VAL ve VAL_STRG (Karakter dizisine/karakter dizisinden ve sayı değerine/sayı değerinden dönüştür) yordamları 321
9.2.3.2 Strg_TO_Chars ve Chars_TO_Strg (Karakter dizisine/karakter dizisinden ve CHAR elemanlarından oluşan Array'e/Array'den dönüştür) yordamları 329
9.2.3.3 ATH ve HTA (ASCII karakter dizisine/ASCII karakter dizisinden ve onaltılık sayıya/onaltılık sayıdan dönüştür) 331
9.2.4 Karakter dizisi yordamları 333
9.2.4.1 MAX_LEN (bir karakter dizisinin maksimum uzunluğu) 333
9.2.4.2 LEN (bir karakter dizisinin uzunluğunu belirle) 334
9.2.4.3 CONCAT (karakter dizilerini birleştir) 334
9.2.4.4 LEFT, RIGHT ve MID yordamları (bir karakter dizisindeki kısmi karakter dizilerini oku) 335
9.2.4.5 DELETE (bir karakter dizisindeki karakteri sil) 336
9.2.4.6 INSERT (bir karakter dizisine karakter ekle) 337
9.2.4.7 REPLACE (bir karakter dizisindeki karakteri değiştir) 338
9.2.4.8 FIND (Bir karakter dizisindeki karakterleri bul) 339
9.2.5 Çalışma zamanıyla ilgili bilgiler 340
9.2.5.1 GetSymbolName (giriş parametresindeki bir değişkenin adını oku) 340
9.2.5.2 GetSymbolPath (giriş parametresi kaynağının birleşik global adını sorgula) 343
9.2.5.3 GetInstanceName (blok örneğinin adını oku) 346
9.2.5.4 GetInstancePath (blok örneğinin birleşik global adını sorgula) 348
9.2.5.5 GetBlockName (bloğun adını oku) 350

9.3 Desantral G/Ç (PROFINET, PROFIBUS veya AS-i) 352
9.3.1 RDREC ve WRREC (veri setlerini oku/yaz) 352
9.3.2 GETIO (proses görünümünü oku) 354
9.3.3 SETIO (proses görünümünü aktar) 355
9.3.4 GETIO_PART (proses görünümü aralığını oku) 356
9.3.5 SETIO_PART (proses görünümü aralığını aktar) 358
9.3.6 RALRM (alarmı al) 359
9.3.7 D_ACT_DP (DP-Slave'leri devre dışı bırak/etkinleştir) 362
9.3.8 RDREC, WRREC ve RALRM için kullanılan STATUS parametreleri 367
9.3.9 Diğer konular 371
9.3.9.1 DPRD_DAT ve DPWR_DAT (tutarlı verileri oku/yaz) 371
9.3.9.2 RCVREC (veri setini al) 373
9.3.9.3 PRVREC (veri setini hazırla) 376
9.3.9.4 DPNRM_DG (bir DP-Slave'in teşhis verilerini oku) 378

9.4 PROFIenergy 380

9.5 Alarmlar 381
9.5.1 ATTACH ve DETACH (OB ve alarm olayını birbirlerine ata/OB ve alarm olayı atamasını kaldır) yordamları 381
9.5.2 Uyandırma alarmları 384
9.5.2.1 SET_CINT (uyandırma alarmını parametrelendir) 384
9.5.2.2 QRY_CINT (uyandırma alarmı parametrelerini sorgula) 386
9.5.3 Saat alarmları 387
9.5.3.1 SET_TINTL (saat alarmını belirle) 387
9.5.3.2 CAN_TINT (saat alarmını sil) 389
9.5.3.3 ACT_TINT (saat alarmını etkinleştir) 389

9.5.3.4 QRY_TINT (saat alarmının durumunu sorgula) ... 390
9.5.4 Gecikme alarmları ... 391
9.5.5 DIS_AIRT ve EN_AIRT yordamları (daha yüksek öncelikli alarmların ve asenkron hata olaylarının yürütülmesini geciktir/etkinleştir) ... 393

9.6 Alarmlar ... 394
9.6.1 Gen_UsrMsg (kullanıcı teşhis mesajları oluştur) ... 394

9.7 Teşhis (PROFINET veya PROFIBUS) ... 397
9.7.1 RD_SINFO (güncel OB'nin başlangıç bilgisini oku) ... 397
9.7.2 LED (LED durumunu okuma) ... 406
9.7.3 Get_IM_Data (tanımlama ve bakım verilerini oku) ... 407
9.7.4 Get_Name (Bir PROFINET IO cihazının adını oku) ... 408
9.7.5 GetStationInfo (bir PROFINET IO cihazının IP veya MAC adresini oku) ... 414
9.7.6 DeviceStates yordamı ... 420
9.7.6.1 DeviceStates konfigürasyonuyla ilgili örnekler ... 422
9.7.7 ModuleStates yordamı ... 425
9.7.7.1 ModuleStates konfigürasyonuyla ilgili örnekler ... 426
9.7.8 GET_DIAG (teşhis bilgilerini oku) ... 430
9.7.9 GetSMCInfo (memory card ile ilgili bilgileri oku) ... 435
9.7.10 Desantral çevrebirim için kullanılan teşhis olayları ... 438

9.8 Darbe ... 439
9.8.1 CTRL_PWM (darbe süresi modülasyonu) ... 439
9.8.2 CTRL_PTO (darbe dizisi) ... 441
9.8.3 Darbe çıkışlarının çalışma şekli ... 444
9.8.4 PWM veya PTO için bir darbe kanalının yapılandırılması ... 446

9.9 Reçeteler ve veri protokolleri ... 450
9.9.1 Reçeteler ... 450
9.9.1.1 Reçetelere genel bakış ... 450
9.9.1.2 Bir reçeteyle ilgili örnek ... 451
9.9.1.3 Reçete verilerinin aktarılması için kullanılan program yordamları ... 454
9.9.1.4 Reçetelerle ilgili örnek program ... 458
9.9.2 Veri protokolleri ... 460
9.9.2.1 Veri protokollerinin veri seti yapısı ... 461
9.9.2.2 Veri protokollerini kontrol etmeye yönelik program yordamları ... 462
9.9.2.3 Veri protokolleriyle çalışma ... 477
9.9.2.4 Veri protokolü dosyalarının boyutu için limit ... 479
9.9.2.5 Veri protokolleriyle ilgili örnek program ... 481

9.10 Veri bloğu kumandası ... 487
9.10.1 CREATE_DB (veri bloğu oluştur) ... 487
9.10.2 READ_DBL ve WRIT_DBL (yükleme belleğindeki veri bloğunu oku/veri bloğunu yükleme belleğine yaz) yordamları ... 490
9.10.3 ATTR_DB (bir veri bloğunun özniteliğini oku) ... 493
9.10.4 DELETE_DB (veri bloğunu sil) ... 495

9.11 Adres işleme ... 496
9.11.1 GEO2LOG (donanım kodunu yuva aracılığıyla belirle) ... 496
9.11.2 LOG2GEO (yuvayı donanım kodu aracılığıyla belirle) ... 498
9.11.3 IO2MOD (donanım kodunu bir G/Ç adresi aracılığıyla belirle) ... 499
9.11.4 RD_ADDR (G/Ç adreslerini donanım kodu aracılığıyla belirle) ... 500
9.11.5 GEOADDR sistem veri tipi ... 501

9.12 Gelişmiş yordamlar için ortak hata kodları ... 502

9.13 Dosyaların kullanımı ... 502
9.13.1 FileReadC: Dosyanın memory card'dan okunması ... 502
9.13.2 FileWriteC: Dosyayı memory card'a yaz ... 504
9.13.3 FileDelete: Memory card'daki dosyayı sil ... 507

10 Teknoloji yordamları ... 509

10.1 Sayma (hızlı sayaç) ... 509
10.1.1 CTRL_HSC_EXT (hızlı sayaca kumanda et, geliştirilmiş) ... 510
10.1.1.1 CTRL_HSC_EXT Örnek ... 511
10.1.1.2 CTRL_HSC_EXT yordamının sistem veri tipleri (SDT) ... 514
10.1.2 Hızlı sayacın işletimi ... 519
10.1.2.1 Senkronizasyon fonksiyonu ... 519
10.1.2.2 Geçit (gate) fonksiyonu ... 520
10.1.2.3 Kaydetme fonksiyonu ... 522
10.1.2.4 Karşılaştırma fonksiyonu ... 523
10.1.2.5 Uygulama ... 524
10.1.3 Bir hızlı sayacın yapılandırılması ... 525
10.1.3.1 Sayma türleri ... 527
10.1.3.2 İşletim aşaması ... 528
10.1.3.3 Başlangıç değerleri ... 532
10.1.3.4 Giriş fonksiyonları ... 532
10.1.3.5 Çıkış fonksiyonu ... 533
10.1.3.6 Alarm olayları ... 533
10.1.3.7 Donanım girişinin bağlantı yerleşimi ... 534
10.1.3.8 Donanım çıkışının bağlantı yerleşimi ... 536
10.1.3.9 HSC giriş adresleri ... 536
10.1.3.10 Donanım kodu ... 536
10.1.4 Eski CTRL_HSC yordamı (hızlı sayaca kumanda et) ... 537
10.1.4.1 CTRL_HSC (hızlı sayaca kumanda et) ... 537
10.1.4.2 CTRL_HSC'nin kullanımı ... 538
10.1.4.3 HSC'nin güncel sayma değeri ... 539

10.2 Hareket kumandası ... 539
10.2.1 Hareket kumandasına genel bakış ... 539
10.2.2 Hareket kumandası için kullanılan donanım bileşenleri ... 540
10.2.3 Hareket kumandası yordamları ... 541
10.2.3.1 MC yordamlarına genel bakış ... 541
10.2.3.2 MC_Power (ekseni serbest bırak/bloke et) ... 542
10.2.3.3 MC_Reset (hatayı onayla) ... 543
10.2.3.4 MC_Home (eksenin referans nokta hareketini gerçekleştir) ... 543
10.2.3.5 MC_Halt (ekseni duraklat) ... 544
10.2.3.6 MC_MoveAbsolute (ekseni mutlak şekilde konumla) ... 544
10.2.3.7 MC_MoveRelative (ekseni bağıl şekilde konumla) ... 545
10.2.3.8 MC_MoveVelocity (ekseni önceden tanımlanmış hızla hareket ettir) ... 545
10.2.3.9 MC_MoveJog (Ekseni JOG modunda hareket ettir) ... 546
10.2.3.10 MC_CommandTable (eksen kumanda komutunu hareket dizisi olarak yürüt) ... 546
10.2.3.11 MC_WriteParam (bir teknoloji nesnesinin parametrelerini yaz) ... 547
10.2.3.12 MC_ReadParam (teknoloji nesnesinin parametrelerini oku) yordamı ... 547
10.2.3.13 MC_ChangeDynamic (eksenin dinamik ayarlarını değiştir) ... 548

10.3 PID ayarlaması ... 549
10.3.1 PID fonksiyonları ... 549
10.3.2 PID yordamları ... 550
10.3.2.1 PID_Compact yordamı ... 550
10.3.2.2 PID_3Step yordamı ... 551
10.3.2.3 PID_Temp yordamı ... 552

11 İletişim ... 553

11.1 Genel bakış ... 553
11.2 Ethernet iletişimi için kullanılan iletişim protokolleri ve portlar ... 555
11.3 Asenkron iletişim bağlantıları ... 556
11.4 Desteklenen sertifikalar ... 560
11.5 PROFINET ... 561
11.5.1 Ağ bağlantısı oluşturma ... 563
11.5.2 Yerel ile partner CPU arasındaki bağlantı yolunun yapılandırılması ... 564
11.5.3 IP adreslerinin atanması ... 567
11.5.3.1 Programlama ve ağ cihazları için kullanılan IP adreslerinin atanması ... 567
11.5.3.2 Programlama cihazınızın IP adresinin belirlenmesi ... 568
11.5.3.3 Bir IP adresinin bir CPU'ya çevrimiçi olarak atanması ... 570
11.5.3.4 Projenizdeki bir CPU için IP adresinin yapılandırılması ... 571
11.5.4 PROFINET ağının test edilmesi ... 575
11.5.5 CPU'nun Ethernet adresinin (MAC adresi) belirlenmesi ... 577
11.5.6 NTP senkronizasyonunun (Network Time Protocol, NTP) yapılandırılması ... 579
11.5.7 Yol alma süresi, PROFINET cihazlarının adlandırılması ve adres ataması ... 581
11.5.8 Açık kullanıcı iletişimi ... 582
11.5.8.1 Protokoller ... 582
11.5.8.2 TCP ve ISO on TCP ... 583
11.5.8.3 Geçici mod ... 584
11.5.8.4 İletişim hizmetleri ve kullanılan port numaraları ... 585
11.5.8.5 Açık kullanıcı iletişimi için kullanılan yordamların bağlantı ID'leri ... 586
11.5.8.6 PROFINET bağlantısı için kullanılan parametreler ... 589
11.5.8.7 Desteklenen TLS sürümleri ... 595
11.5.8.8 Bir DNS'nin yapılandırılması ... 596
11.5.8.9 TIA Portal'da OUC bağlantısının yapılandırılması ... 597
11.5.8.10 TSEND_C ve TRCV_C yordamları ... 600
11.5.8.11 Daha eski sistemlerdeki TSEND_C ve TRCV_C yordamları ... 610
11.5.8.12 TCON, TDISCON, TSEND ve TRCV yordamları ... 616
11.5.8.13 TCONSettings ... 625
11.5.8.14 Daha eski TCON, TDISCON, TSEND ve TRCV yordamları ... 631
11.5.8.15 T_RESET (mevcut bağlantıyı sonlandır ve yeni bağlantı kur) yordamı ... 639
11.5.8.16 T_DIAG (bir bağlantının durumunu kontrol et ve bilgileri oku) yordamı ... 641
11.5.8.17 TMAIL_C (e-postayı CPU'nun Ethernet arayüzü üzerinden gönder) yordamı ... 645
11.5.8.18 UDP ... 664
11.5.8.19 TUSEND ve TURCV ... 665
11.5.8.20 T_CONFIG ... 670
11.5.8.21 Yordamlar için ortak parametreler ... 680
11.5.9 Bir programlama cihazı ile iletişim ... 681
11.5.9.1 Donanım iletişim bağlantısının oluşturulması ... 682
11.5.9.2 Cihazların konfigürasyonu ... 682
11.5.9.3 IP adreslerinin atanması ... 683

11.5.9.4 PROFINET ağınızın test edilmesi ... 683
11.5.10 HMI/PLC iletişimi ... 684
11.5.10.1 İki cihaz arasında mantıksal ağ bağlantılarının yapılandırılması ... 685
11.5.11 PLC/PLC iletişimi ... 685
11.5.11.1 İki cihaz arasında mantıksal ağ bağlantılarının yapılandırılması ... 686
11.5.11.2 Yerel ile partner CPU arasındaki bağlantı yolunun yapılandırılması ... 686
11.5.11.3 Gönderim ve alım parametrelerinin yapılandırılması ... 686
11.5.12 CPU ve PROFINET IO cihazının yapılandırılması ... 689
11.5.12.1 PROFINET IO cihazının eklenmesi ... 689
11.5.12.2 CPU'lar ve cihaz adlarının atanması ... 690
11.5.12.3 IP adreslerinin atanması ... 690
11.5.12.4 IO çevrim süresinin yapılandırılması ... 691
11.5.13 CPU ve PROFINET I-Device'ın yapılandırılması ... 692
11.5.13.1 I-Device fonksiyonu ... 692
11.5.13.2 I-Device'ın özellikleri ve avantajları ... 693
11.5.13.3 Bir I-Device'ın özellikleri ... 694
11.5.13.4 Daha yüksek ve daha düşük seviye G/Ç sistemi arasında veri alışverişi ... 697
11.5.13.5 I-Device yapılandırması ... 699
11.5.14 Shared Device'lar ... 702
11.5.14.1 Shared Device fonksiyonu ... 702
11.5.14.2 Örnek: Bir Shared Device'ın yapılandırılması (GSD konfigürasyonu) ... 705
11.5.14.3 Örnek: Bir I-Device'ın bir Shared Device olarak yapılandırılması ... 711
11.5.15 Medya yedeklilik protokolü (MRP) ... 719
11.5.15.1 Halka topolojilerinde medya yedekliliği ... 721
11.5.15.2 Medya yedeklilik protokolünün (MRP) kullanımı ... 723
11.5.15.3 Medya yedekliliğinin yapılandırılması ... 726
11.5.16 S7-Routing ... 729
11.5.16.1 CPU ve CP arayüzleri arasında S7-Routing ... 729
11.5.16.2 İki CP arayüzü arasında S7-Routing ... 730
11.5.17 SNMP ... 730
11.5.18 Teşhis ... 734
11.5.19 Desantral çevrebirimler için yordamlar ve teşhis yordamları ... 734

11.6 PROFIBUS ... 734
11.6.1 PROFIBUS CM'ler üzerinden iletişim hizmetleri ... 736
11.6.2 PROFIBUS CM'ler için kullanıcı kılavuzlarına ilişkin not ... 736
11.6.3 DP-Master ve DP-Slave'in yapılandırılması ... 737
11.6.3.1 CM 1243-5 (DP-Master) ve DP-Slave eklenmesi ... 737
11.6.3.2 İki PROFIBUS cihazı arasındaki mantıksal ağ bağlantılarının yapılandırılması ... 737
11.6.3.3 CM 1243-5 ve DP-Slave'e PROFIBUS adreslerinin atanması ... 738
11.6.4 Desantral çevrebirimler için yordamlar ve teşhis yordamları ... 739

11.7 AS
i ... 739
11.7.1 AS-i Master ve Slave cihazların yapılandırılması ... 740
11.7.1.1 AS-i Master modülü CM 1243-2 ve AS-i Slave'in eklenmesi ... 741
11.7.1.2 İki AS-i cihazı arasındaki mantıksal ağ bağlantılarını yapılandırılması ... 741
11.7.1.3 AS-i Master CM1243-2'nin özelliklerinin yapılandırılması ... 742
11.7.1.4 Bir AS-i Slave'e bir AS-i adresinin atanması ... 743
11.7.2 Kullanıcı programı ile AS-i Slave'ler arasında veri alışverişi ... 746
11.7.2.1 STEP 7 konfigürasyonu ... 746
11.7.2.2 AS-i Slave'lerin STEP 7 ile yapılandırılması ... 747
11.7.3 Desantral çevrebirimler için yordamlar ... 749

11.7.4 AS-i çevrimiçi araçlarıyla çalışma ... 749

11.8 S7 iletişimi ... 750
11.8.1 GET ve PUT (uzak CPU'dan okuma ve yazma) ... 750
11.8.2 S7 bağlantısının oluşturulması ... 753
11.8.3 Yerel ile partner CPU arasındaki bağlantı yolunun yapılandırılması ... 754
11.8.4 GET/PUT için bağlantı parametrelerinin atanması ... 755
11.8.4.1 Bağlantı parametreleri ... 756
11.8.4.2 CPU'dan CPU'ya S7 bağlantısını yapılandırılması ... 758

11.9 IP adresi üzerinden CPU'ya erişilemiyorsa ne yapmalısınız? ... 763

11.10 OPC UA sunucusu ... 764
11.10.1 OPC UA sunucusunun konfigürasyonu ... 764
11.10.1.1 OPC UA sunucusunun etkinleştirilmesi ... 765
11.10.1.2 İşletim sırasında OPC UA sunucusunun davranışı ... 766
11.10.1.3 OPC UA sunucusu için ayarlar ... 767
11.10.1.4 OPC UA sunucusunun sınırlamaları ... 769
11.10.2 OPC UA sunucusunun güvenliği ... 770
11.10.2.1 Desteklenen güvenlik direktifleri ... 771
11.10.2.2 Güvenilir istemciler ... 773
11.10.2.3 Kullanıcı kimlik doğrulaması ... 774
11.10.3 OPC UA sunucusunun arayüzü ... 776
11.10.3.1 Desteklenen veri tipleri ... 776
11.10.3.2 PLC gösterimi ... 777
11.10.3.3 Yüklenebilir sunucu arayüzleri ... 778
11.10.4 OPC UA teşhis tamponu ... 780
11.10.4.1 OPC UA limitine ulaşıldı ... 781
11.10.4.2 OPC UA güvenlik olayları ... 783
11.10.4.3 Yanlış OPC UA kullanımı ... 790
11.10.4.4 OPC UA için özet mesajlar ... 791
11.10.5 OPC UA yöntem çağrıları ... 791
11.10.5.1 Sunucu yöntemlerini kullanmak için sınır koşulları ... 795

12 Web sunucusu ... 796

12.1 Web sunucusunu etkinleştirilmesi ... 799

12.2 Web sunucusu kullanıcılarının yapılandırılması ... 801

12.3 PC üzerinden web sitelerine erişim ... 803

12.4 Web sitelerine bir mobil cihaz üzerinden erişilmesi ... 805

12.5 Web sitelerine erişim için bir CP modülünün kullanımı ... 806

12.6 Bir güvenlik sertifikasının indirilmesi ve kurulması ... 807

12.7 Standart web siteleri ... 808
12.7.1 Standart web sitelerinin yapısı ... 808
12.7.2 Temel siteler ... 809
12.7.3 Oturum açma ve kullanıcı yetkileri ... 810
12.7.4 Giriş ... 813
12.7.5 Start ... 814
12.7.6 Teşhis ... 815
12.7.7 Teşhis tamponu ... 818
12.7.8 Modül bilgileri ... 819

12.7.9 İletişim ... 824
12.7.10 Değişken durumu ... 828
12.7.11 İzleme tabloları ... 829
12.7.12 Çevrimiçi yedekleme ... 831
12.7.13 Veri protokolleri ... 833
12.7.14 Kullanıcı dosyaları ... 836
12.7.15 Veri protokolleri ve kullanıcı dosyaları için API ... 838
12.7.16 Dosya tarayıcısı ... 839

12.8 Kullanıcı tanımlı web siteleri ... 840
12.8.1 HTML sayfalarının oluşturulması ... 841
12.8.2 S7-1200 Web sunucusu tarafından desteklenen AWP komutları ... 843
12.8.2.1 Değişkenleri oku ... 844
12.8.2.2 Değişkenleri yaz ... 845
12.8.2.3 Özel değişkenlerin okunması ... 848
12.8.2.4 Özel değişkenlerin yazılması ... 849
12.8.2.5 Değişken referanslama için takma ad kullanımı ... 850
12.8.2.6 Enum tiplerinin tanımlanması ... 851
12.8.2.7 Bir Enum tipi ile CPU değişkenlerine referanslama ... 852
12.8.2.8 Fragmentlerin oluşturulması ... 854
12.8.2.9 Fragmentlerin içe aktarımı ... 855
12.8.2.10 Tanımların bağlanması ... 855
12.8.2.11 Özel karakterli değişken adlarının işlenmesi ... 856
12.8.3 Kullanıcı tanımlı web sitelerinin kullanımının yapılandırılması ... 858
12.8.4 Ana sayfanın yapılandırılması ... 859
12.8.5 Kullanıcı tanımlı web siteleri için WWW yordamının programlanması ... 860
12.8.6 Program bloklarının CPU'ya yüklenmesi ... 861
12.8.7 Kullanıcı tanımlı web sitelerine erişim ... 862
12.8.8 Kullanıcı tanımlı web sitelerindeki sınırlamalar ... 863
12.8.9 Kullanıcı tanımlı web siteleri için örnek ... 864
12.8.9.1 Bir rüzgâr türbinini izlemek ve denetlemek için web sitesi ... 864
12.8.9.2 Kumanda verilerinin okunması ve görüntülenmesi ... 866
12.8.9.3 Enum tipinin kullanılması ... 867
12.8.9.4 Kullanıcı veri girişlerinin kumanda ünitesine yazılması ... 868
12.8.9.5 Özel değişkenlerin yazılması ... 869
12.8.9.6 Referans: "Uzak rüzgâr türbini monitörü" web sitesinin HTML kodu ... 869
12.8.9.7 STEP 7'de örnek web sitesinin konfigürasyonu ... 875
12.8.10 Kullanıcı tanımlı, çok dilli web sitelerinin oluşturulması ... 876
12.8.10.1 Klasör yapısı oluşturulması ... 876
12.8.10.2 Dil değiştirmenin programlanması ... 877
12.8.10.3 STEP 7'nin çok dilli bir sayfa yapısı kullanacak şekilde yapılandırılması ... 879
12.8.11 Kullanıcı tanımlı web sitelerin genişletilmiş kumandası ... 880
12.8.12 Web API'si ... 884
12.8.12.1 Desteklenen Web API yöntemleri ... 885

12.9 Sınırlamalar ... 885
12.9.1 JavaScript kullanımı ... 886
12.9.2 İnternet opsiyonlarında çerezlere izin verilmezse fonksiyonlar kısıtlıdır ... 886
12.9.3 Değişken adlarını ve değerlerini girme kuralları ... 887
12.9.4 Veri protokollerinin CSV formatında Microsoft Excel'in ABD/İngilizce dışındaki sürümlerine içe aktarılması ... 887

13 İletişim işlemcisi ve Modbus TCP ... 889

13.1 Seri iletişim arayüzleriyle çalışma ... 889
13.2 Bir RS485 veriyolu bağlantı fişinin sonlandırılması ... 890
13.3 Noktadan noktaya iletişim (PtP) ... 891
13.3.1 PtP, serbestçe programlanabilir iletişim ... 892
13.3.2 3964(R) iletişimi ... 893
13.3.3 Serbestçe programlanabilen PtP iletişiminin konfigürasyonu ... 895
13.3.3.1 Akış kontrolü kumandası ... 897
13.3.3.2 Gönderim parametrelerinin yapılandırılması ... 898
13.3.3.3 Alım parametrelerinin yapılandırılması ... 899
13.3.4 3964(R) iletişiminin yapılandırılması ... 906
13.3.4.1 3964(R) iletişim portlarının yapılandırılması ... 906
13.3.4.2 3964(R) önceliğinin ve protokol parametrelerinin yapılandırılması ... 907
13.3.5 Noktadan noktaya yordamlar ... 908
13.3.5.1 Noktadan noktaya yordamlar için ortak parametreler ... 908
13.3.5.2 Port_Config (iletişim parametrelerini dinamik olarak yapılandırılması) ... 910
13.3.5.3 Send_Config (seri iletişim için parametrelerin dinamik olarak yapılandırılması) ... 913
13.3.5.4 Receive_Config (seri alım parametrelerinin dinamik olarak yapılandırılması) ... 915
13.3.5.5 P3964_Config (3964(R) protokolü yapılandırması) ... 919
13.3.5.6 Send_P2P (gönderim tamponu verilerinin aktarılması) ... 921
13.3.5.7 Receive_P2P (mesaj alımı etkinleştirme) ... 924
13.3.5.8 Receive_Reset (alım tamponunun silinmesi) ... 926
13.3.5.9 Signal_Get (RS-232 sinyallerinin sorgulanması) ... 927
13.3.5.10 Signal_Set (RS-232 sinyallerinin belirlenmesi) ... 928
13.3.5.11 Get_Features ... 929
13.3.5.12 Set_Features ... 930
13.3.6 PtP iletişiminin programlanması ... 931
13.3.6.1 Sorgulama mimarisi ... 932
13.3.7 Örnek: Noktadan noktaya iletişim ... 933
13.3.7.1 İletişim modülünün yapılandırılması ... 934
13.3.7.2 RS422 ve RS485 işletim türleri ... 937
13.3.7.3 STEP 7 programının programlanması ... 940
13.3.7.4 Terminal emülatörünün yapılandırılması ... 941
13.3.7.5 Örnek programın çalıştırılması ... 942
13.4 Üniversal seri arayüz (USS) üzerinden iletişim ... 942
13.4.1 USS yordamları ... 944
13.4.1.1 USS_Port_Scan (USS ağı üzerinden iletişim düzenlenmesi) ... 944
13.4.1.2 USS_Drive_Control (sürücü ile veri alışverişi) ... 945
13.4.1.3 USS_Read-Param (sürücüden parametre okuma) ... 948
13.4.1.4 USS_Write_Param (sürücüdeki parametreyi değiştirme) ... 949
13.4.2 USS durum kodları ... 950
13.4.3 USS sürücü sistemi için genel gereklilikler ... 952
13.4.4 Örnek: Genel USS sürücü bağlantısı ve kurulumu ... 952
13.5 Modbus iletişimi ... 955
13.5.1 Modbus RTU ve Modbus TCP üzerinden iletişime genel bakış ... 955
13.5.2 MODBUS TCP ... 957
13.5.2.1 Genel bakış ... 957
13.5.2.2 Modbus TCP yordamlar ... 958
13.5.2.3 MODBUS TCP ile ilgili örnekler ... 1015
13.5.3 Modbus RTU ... 1020
13.5.3.1 Genel bakış ... 1020

13.5.3.2 Desteklenen maksimum Modbus Slave sayısı ... 1022
13.5.3.3 Modbus RTU yordamları ... 1023
13.5.3.4 Modbus RTU örnekleri ... 1042

13.6 Eski sistemlerdeki PtP iletişimi yordamları (yalnızca CM/CB 1241) ... 1045
13.6.1 Eski noktadan noktaya yordamlar ... 1046
13.6.1.1 Port_Config (iletişim parametrelerinin dinamik olarak yapılandırılması) ... 1046
13.6.1.2 SEND_CFG (gönderim konfigürasyonu) ... 1048
13.6.1.3 RCV_CFG (alım konfigürasyonu) ... 1049
13.6.1.4 SEND_PTP (gönderim tamponu aktarımı) ... 1053
13.6.1.5 RCV_PTP (alım mesajlarını etkinleştirme) ... 1055
13.6.1.6 RCV_RST (alım tamponunun silinmesi) ... 1056
13.6.1.7 SGN_GET (RS232 sinyallerinin sorgulanması) ... 1057
13.6.1.8 SGN_SET (RS-232 sinyallerinin ayarlanması) ... 1058

13.7 Daha eski sistemlerdeki USS iletişim yordamları (yalnızca CM/CB 1241) ... 1060
13.7.1 Eski USS yordamları ... 1061
13.7.1.1 USS_PORT yordamı (USS ağı üzerinden iletişim işleme) ... 1061
13.7.1.2 USS_DRV yordamı (sürücü ile veri alışverişi) ... 1062
13.7.1.3 USS_Read_Param yordamı (sürücüden parametre okunması) ... 1065
13.7.1.4 USS_WPM yordamı (sürücüdeki parametreyi değiştirme) ... 1066
13.7.2 Eski USS durum kodları ... 1067
13.7.3 Sürücü ünitesi için olan eski USS yordamı ile ilgili genel gereklilikler ... 1069

13.8 Daha eski sistemlerdeki Modbus TCP iletişimi ... 1069
13.8.1 Genel bakış ... 1069
13.8.2 Eski Modbus TCP yordamları ... 1070
13.8.2.1 MB_CLIENT yordamları (PROFINET üzerinden Modbus TCP istemcisi olarak iletişim) ... 1070
13.8.2.2 MB_SERVER (Modbus TCP sunucusu olarak PROFINET üzerinden iletişim) ... 1076
13.8.3 Daha eski Modbus TCP örnekleri ... 1081
13.8.3.1 Örnek: Eski sistemlerdeki çoklu TCP-bağlantıları için MB_SERVER ... 1081
13.8.3.2 Örnek: Eski sistemlerdeki MB_CLIENT 1: Ortak TCP bağlantısında çoklu talepler ... 1082
13.8.3.3 Örnek: Eski sistemlerdeki MB_CLIENT 2: Farklı TCP bağlantılarıyla çoklu talepler ... 1083
13.8.3.4 Örnek: Eski sistemlerdeki MB_CLIENT 3: Çıkışların proses görünümü için yazma talebi ... 1084
13.8.3.5 Örnek: Eski sistemlerdeki MB_CLIENT 4: Birden fazla talebin koordine edilmesi ... 1084

13.9 Eski sistemlerdeki Modbus RTU iletişimi (yalnızca CM/CB 1241) ... 1085
13.9.1 Genel bakış ... 1085
13.9.2 Eski Modbus RTU yordamları ... 1086
13.9.2.1 MB_COMM_LOAD (Modbus RTU için PtP modülü üzerindeki portun yapılandırılması) ... 1086
13.9.2.2 MB_MASTER (Modbus RTU Master olarak PtP portu üzerinden iletişim) ... 1088
13.9.2.3 MB_SLAVE (Modbus RTU Slave olarak PtP portu üzerinden iletişim) ... 1094
13.9.3 Eski sistemlerdeki Modbus RTU ile ilgili örnek ... 1100
13.9.3.1 Örnek: Daha eski sistemlerdeki bir Modbus RTU-Master ile ilgili örnek program ... 1100
13.9.3.2 Örnek: Daha eski sistemlerdeki bir Modbus RTU Slave için örnek program ... 1102

13.10 Industrial Remote Communication (IRC) ... 1103
13.10.1 Telecontrol CP'lerine genel bakış ... 1103
13.10.2 Daha fazla bilgi ... 1105

14 TeleService iletişimi (SMTP ile e-posta iletişimi) ... 1106

14.1 TM_Mail (e-posta gönder) yordamı ... 1106

15 Çevrimiçi araçlar ve teşhis araçları 1113

15.1 Durum LED'leri 1113
15.2 Bir CPU ile çevrimiçi bağlantı kurma 1117
15.3 Bir PROFINET IO cihazına çevrimiçi olarak bir ad atanması 1119
15.4 IP adresi ve saatin ayarlanması 1120
15.5 Firmware güncellemesi 1120
15.6 Gizli PLC konfigürasyon verilerini koruma şifresinin belirlenmesi veya silinmesi 1122
15.7 Fabrika ayarlarına sıfırlama 1123
15.8 Bir modülün arızalar açısından kontrol edilmesi (servis verilerinin kaydedilmesi) 1124
15.9 Bir SIMATIC memory card'ın STEP 7 aracılığıyla biçimlendirilmesi 1126
15.10 Çevrimiçi CPU'nun kontrol paneli 1127
15.11 Çevrim süresinin ve bellek kullanımının denetlenmesi 1127
15.12 Teşhis olaylarının CPU'da görüntülenmesi 1128
15.13 Çevrimdışı ve çevrimiçi CPU'ların karşılaştırılması 1129
15.14 Bir çevrimiçi/çevrimdışı topoloji karşılaştırmasının gerçekleştirilmesi 1130
15.15 CPU'daki değerlerin izlenmesi ve bu değerlere kumanda edilmesi 1131
15.15.1 CPU'nun değerlerini izlemek için çevrimiçi olma 1132
15.15.2 Durumu programlama editöründe görüntüleme 1133
15.15.3 Değerleri geri yüklemek için bir DB'nin çevrimiçi değerlerinin bir anlık kaydının alınması 11-33
15.15.4 İzleme tablosu aracılığıyla CPU'daki değerlerin izlenmesi ve bu değerlere kumanda edilmesi 11-35
15.15.4.1 Değişkenleri tetikle izleme veya değişkenlere tetikle kumanda etme 1136
15.15.4.2 Çıkışların STOP işletim durumunda etkinleştirilmesi 1137
15.15.5 CPU'daki değerlerin zorlanması 1138
15.15.5.1 Force tablosuyla çalışma 1138
15.15.5.2 Force fonksiyonunun çalışma şekli 1139
15.16 RUN işletim durumunda yükleme 1140
15.16.1 "RUN işletim durumunda yükleme" fonksiyonuyla ilgili ön koşullar 1141
15.16.2 Programın RUN işletim durumunda değiştirilmesi 1142
15.16.3 Seçilen blokların yüklenmesi 1143
15.16.4 Bir derleme hatası içeren seçilmiş tek bir bloğun başka bir blokta yüklenmesi 1144
15.16.5 Mevcut blokların RUN işletim durumunda değiştirilmesi ve hedef sisteme yüklenmesi 1145
15.16.6 Yükleme işlemi başarısız olduğunda sistem tepkisi 1147
15.16.7 RUN işletim durumunda yükleme sırasındaki güvenlikle ilgili hususlar 1148
15.17 Tetikleme koşullarında CPU verilerinin takip edilmesi ve kaydedilmesi 1150
15.18 Bir tel kopmasının bir SM 1231 modülü aracılığıyla belirlenmesi 1152
15.19 Bir CPU'nun yedeklenmesi ve geri yüklenmesi 1154
15.19.1 Yedekleme ve geri yüklemeyle ilgili opsiyonlar 1154
15.19.2 Bir çevrimiçi CPU'nun yedeklemesi 1156
15.19.3 Bir CPU'nun geri yüklenmesi 1158

A Teknik veriler 1160

A.1 Çevrimiçi destek için Siemens web sitesi 1160

A.2 Genel teknik veriler ... 1160
A.3 Bağlantı yerleşimi - PROFINET arayüz portu X1 ... 1171
A.4 CPU 1211C ... 1173
A.4.1 Genel teknik veriler ve performans özellikleri ... 1173
A.4.2 CPU 1211C tarafından desteklenen süreler, sayaçlar ve kod blokları ... 1174
A.4.3 Dijital girişler ve çıkışlar ... 1179
A.4.4 Analog girişler ... 1180
A.4.4.1 CPU'nun entegre analog girişlerinin adım yanıtı ... 1181
A.4.4.2 CPU'nun entegre analog portlarının tarama aralığı ... 1181
A.4.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar) ... 1181
A.4.5 CPU 1211C'nin devre şemaları ... 1182
A.5 CPU 1212C ... 1185
A.5.1 Genel teknik veriler ve performans özellikleri ... 1185
A.5.2 CPU 1212C tarafından desteklenen süreler, sayaçlar ve kod blokları ... 1186
A.5.3 Dijital girişler ve çıkışlar ... 1191
A.5.4 Analog girişler ... 1193
A.5.4.1 CPU'nun entegre analog girişlerinin adım yanıtı ... 1193
A.5.4.2 CPU'nun entegre analog portlarının tarama aralığı ... 1194
A.5.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar) ... 1194
A.5.5 CPU 1212C'nin devre şemaları ... 1195
A.6 CPU 1214C ... 1199
A.6.1 Genel teknik veriler ve performans özellikleri ... 1199
A.6.2 CPU 1214C tarafından desteklenen süreler, sayaçlar ve kod blokları ... 1200
A.6.3 Dijital girişler ve çıkışlar ... 1205
A.6.4 Analog girişler ... 1207
A.6.4.1 CPU'nun entegre analog girişlerinin adım yanıtı ... 1207
A.6.4.2 CPU'nun entegre analog portlarının tarama aralığı ... 1208
A.6.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar) ... 1208
A.6.5 CPU 1214C'nin devre şemaları ... 1209
A.7 CPU 1215C ... 1213
A.7.1 Genel teknik veriler ve performans özellikleri ... 1213
A.7.2 CPU 1215C tarafından desteklenen süreler, sayaçlar ve kod blokları ... 1214
A.7.3 Dijital girişler ve çıkışlar ... 1219
A.7.4 Analog girişler ve çıkışlar ... 1221
A.7.4.1 CPU'nun entegre analog girişlerinin adım yanıtı ... 1221
A.7.4.2 CPU'nun entegre analog portlarının tarama aralığı ... 1222
A.7.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar) ... 1222
A.7.4.4 Analog çıkışların teknik verileri ... 1222
A.7.5 CPU 1215C'nin devre şemaları ... 1224
A.8 CPU 1217C ... 1228
A.8.1 Genel teknik veriler ve performans özellikleri ... 1228
A.8.2 CPU 1217C tarafından desteklenen süreler, sayaçlar ve kod blokları ... 1230
A.8.3 Dijital girişler ve çıkışlar ... 1234
A.8.4 Analog girişler ve çıkışlar ... 1237
A.8.4.1 Analog girişlerin teknik verileri ... 1237
A.8.4.2 CPU'nun entegre analog girişlerinin adım yanıtı ... 1238
A.8.4.3 CPU'nun entegre analog portlarının tarama aralığı ... 1238
A.8.4.4 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar) ... 1238
A.8.4.5 Analog çıkışların teknik verileri ... 1239

A.8.5 CPU 1217C'nin devre şemaları ... 1240
A.8.6 CPU 1217C diferansiyel giriş (DI) ayrıntıları ve uygulama örneği ... 1242
A.8.7 CPU 1217C diferansiyel çıkış (DO) ayrıntıları ve uygulama örneği ... 1243

A.9 Dijital sinyal modülleri (SM'ler) ... 1244
A.9.1 SM 1221 dijital giriş modülü için teknik veriler ... 1244
A.9.2 8 çıkışlı SM 1222 dijital çıkış modülü için teknik veriler ... 1246
A.9.3 16 çıkışlı SM 1222 dijital çıkış modülü için teknik veriler ... 1247
A.9.4 SM 1223 (V DC) dijital giriş/çıkış modülünün teknik verileri ... 1253
A.9.5 SM 1223 (V AC) dijital giriş/çıkış modülünün teknik verileri ... 1259

A.10 Analog sinyal modülleri (SM'ler) ... 1262
A.10.1 SM 1231 analog giriş modülünün teknik verileri ... 1262
A.10.2 SM 1232 analog çıkış modülünün teknik verileri ... 1266
A.10.3 SM 1234 analog giriş/analog çıkış modülünün teknik verileri ... 1268
A.10.4 Analog girişlerin adım yanıtı ... 1271
A.10.5 Analog girişlerin tarama süresi ve güncelleme süreleri ... 1271
A.10.6 Gerilim ve akım için analog girişlerin ölçüm aralıkları (SB ve SM) ... 1272
A.10.7 Gerilim ve akım için analog çıkışların ölçüm aralıkları (SB ve SM) ... 1273

A.11 Termo eleman ve RTD sinyal modülleri (SM'ler) ... 1274
A.11.1 SM 1231 termo eleman ... 1274
A.11.1.1 Bir termo elemanın temel çalışma şekli ... 1276
A.11.1.2 SM 1231 termo eleman için seçenekler tablosu ... 1277
A.11.2 SM 1231 RTD ... 1280
A.11.2.1 SM 1231 RTD için seçenekler tablosu ... 1283

A.12 Teknoloji modülleri ... 1286
A.12.1 SM 1278 4xIO-Link-Master SM ... 1286
A.12.1.1 SM 1278 4xIO-Link-Master - genel bakış ... 1289
A.12.1.2 Bağlantı ... 1292
A.12.1.3 Parametre/adres aralığı ... 1294
A.12.1.4 Alarm, hata ve sistem bildirimleri ... 1297

A.13 Dijital sinyal kartları (SB'ler) ... 1300
A.13.1 SB 1221 200 kHz dijital girişin teknik verileri ... 1301
A.13.2 SB 1222 200 kHz dijital çıkışın teknik verileri ... 1303
A.13.3 SB 1223 200 kHz dijital giriş/çıkışın teknik verileri ... 1306
A.13.4 2 x 24 V DC girişli / 2 x 24 V DC çıkışlı SB 1223 için teknik veriler ... 1309

A.14 Analog sinyal kartları (SB'ler) ... 1312
A.14.1 SB 1231 1 analog girişin teknik verileri ... 1312
A.14.2 SB 1232 1 analog çıkışın teknik verileri ... 1314
A.14.3 Analog girişler ve çıkışların ölçüm aralıkları ... 1316
A.14.3.1 Analog girişlerin adım yanıtı ... 1316
A.14.3.2 Analog girişlerin tarama süresi ve güncelleme süreleri ... 1316
A.14.3.3 Gerilim ve akım için analog girişlerin ölçüm aralıkları (SB ve SM) ... 1316
A.14.3.4 Gerilim ve akım için analog çıkışların ölçüm aralıkları (SB ve SM) ... 1318
A.14.4 Termo eleman sinyal kartı (SB'ler) ... 1319
A.14.4.1 SB 1231 1 termo eleman analog girişin teknik verileri ... 1319
A.14.4.2 Bir termo elemanın temel çalışma şekli ... 1320
A.14.5 RTD sinyal kartları (SB'ler) ... 1323
A.14.5.1 SB 1231 1 analog girişin teknik verileri nelerdir? ... 1323
A.14.5.2 SB 1231 RTD için seçenekler tabloları ... 1326

A.15 BB 1297 batarya kartı ... 1328

A.16 İletişim arayüzleri ... 1330
A.16.1 PROFIBUS ... 1330
A.16.1.1 CM 1242-5 PROFIBUS DP-SLAVE ... 1330
A.16.1.2 CM 1242-5'in Sub-D soketinin bağlantı yerleşimi ... 1331
A.16.1.3 CM 1243-5 PROFIBUS DP-Master ... 1331
A.16.1.4 CM 1243-5'in Sub-D soketinin bağlantı yerleşimi ... 1333
A.16.2 CP 1242-7 ... 1334
A.16.2.1 CP 1242-7 GPRS ... 1334
A.16.2.2 GSM/GPRS anteni ANT794-4MR ... 1335
A.16.2.3 Yassı anten ANT794-3M ... 1336
A.16.3 CM 1243-2 AS-i Master ... 1337
A.16.3.1 AS-i Master CM 1243-2'nin teknik verileri ... 1337
A.16.3.2 AS-i Master'ın elektrik bağlantısı ... 1338
A.16.4 RS232, RS422 ve RS485 ... 1339
A.16.4.1 CB 1241 RS485'in teknik verileri ... 1339
A.16.4.2 CM 1241 RS232'nin teknik verileri ... 1341
A.16.4.3 CM 1241 RS422/485'in teknik verileri ... 1342
A.17 TeleService (TS adaptörü ve TS adaptör modülü) ... 1343
A.18 SIMATIC memory card'lar ... 1344
A.19 Giriş simülatörleri ... 1344
A.20 S7-1200 potansiyometre modülü ... 1346
A.21 Genişletme modülleri için bağlantı kablosu ... 1347
A.22 İlişkili ürünler ... 1348
A.22.1 PM 1207 akım besleme modülü ... 1348
A.22.2 CSM 1277 Compact Switch modül ... 1348
A.22.3 CM CANopen modülü ... 1348
A.22.4 İletişim modülü RF120C ... 1349
A.22.5 SM 1238 enerji ölçer ... 1349
A.22.6 SIWAREX elektronik tartım sistemi ... 1349

B Güç bilançosunun hesaplanması ... 1350

C Sipariş bilgileri ... 1354

C.1 CPU modülleri ... 1354
C.2 Sinyal modülleri (SM'ler), sinyal kartları (SB'ler) ve batarya kartları (BB'ler) ... 1354
C.3 İletişim ... 1356
C.4 Hata güvenli CPU'lar ve sinyal modülleri ... 1357
C.5 Diğer modüller ... 1357
C.6 Memory card'lar ... 1357
C.7 Yedek parçalar ve diğer donanımlar ... 1358
C.8 Programlama yazılımı ... 1364
C.9 OPC UA lisansları ... 1364

D Cihaz değişimi ve yedek parça uyumluluğu 1365

D.1 Gizli PLC konfigürasyon verilerini koruyan bir CPU'nun değişimi 1365

D.2 Bir CPU V3.0'ın CPU V4.x ile değişimi 1366

D.3 S7-1200 ila V3.0 - yedek klemens blokları 1371

Endeks 1373

Ürüne genel bakış

1

1.1 S7-1200 PLC'ye giriş

S7-1200 kumanda ünitesi, otomasyon çözümleriniz için çok geniş bir cihaz yelpazesini kontrol etmeniz için gereken esnekliği ve gücü sağlar. S7-1200; kompakt yapı, esnek konfigürasyon ve güçlü komut seti sayesinde çok sayıda kumanda uygulaması için mükemmel derecede uygundur.

CPU, aşağıdaki ve diğer elemanları güçlü kompakt bir muhafaza içinde güçlü bir kumanda ünitesine bağlar:

- Bir mikroişlemci
- Entegre bir güç kaynağı
- Giriş ve çıkış akım devreleri
- Entegre PROFINET
- Hareket kumandası için yüksek hızlı girişler/çıkışlar

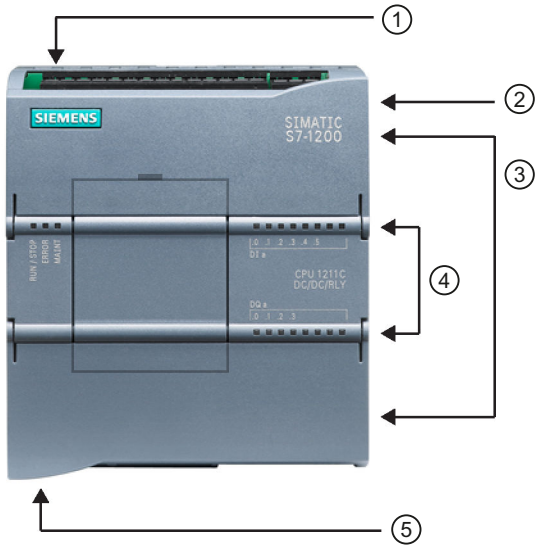
Programınızı yükledikten sonra, CPU, uygulamanızdaki cihazlarınızı gözlemleyebilmeniz ve bu cihazlara kumanda edebilmeniz için gereken lojik sistemi içerir. CPU; Boole ilişkilendirmeleri, sayma ve zaman fonksiyonları, karmaşık aritmetik işlemler, hareket kumandası ve diğer akıllı cihazlarla iletişimi kapsayabilen kullanıcı programınızın komutlarına göre girişleri gözlemler ve çıkışları değiştirir.

CPU'lar, bir PROFINET ağı üzerinden iletişim kurmak için kullanılan PROFINET portlarına sahiptir. Ağlar ve farklı protokoller üzerinden iletişim için başka modüller de mevcuttur:

- PROFIBUS
- Mobil iletişim (LTE, UMTS, GPRS)
- Ethernet
  - Güvenlik
  - Güvenlik duvarı
  - VPN
  - IEC 60870
  - SNP3
- RS485
- RS232
- RS422
- USS
- MODBUS

① Akım bağlantısı

② Üst koruyucu kapağın altında bulunan ve bir memory card için yuva



③ Kullanıcı kablajı için geçmeli klemens bloğu (koruyucu kapakların arkasında)

④ Entegre G/Ç'ler için kullanılan durum LED'leri

⑤ Bir veya iki PROFINET bağlantısı (CPU'nun alt tarafında)

Çeşitli güvenlik fonksiyonları, CPU'ya ve kumanda programına erişimi korur:

- CPU fonksiyonlarına erişimin ihtiyaca göre ayarlanabilmesini sağlayan şifre koruması (Sayfa 156).
- Belirli bir bloktaki kodun korunması için teknik bilgi koruması (Sayfa 159).
- Programınızı belirli bir memory card'a veya CPU'ya bağlamak için kopyalama koruması (Sayfa 161).
- Gizli PLC konfigürasyon verilerinin korunması (Sayfa 154)
- PG/PC ve HMI için güvenli iletişim (Sayfa 158)

Tablo 1-1 CPU varyantlarının karşılaştırması

Özellik		**CPU 1211C**	**CPU 1212C**	**CPU 1214C**	**CPU 1215C**	**CPU 1217C**
Boyutlar (mm)		90 × 100 × 75		110 × 100 × 75	130 × 100 × 75	150 × 100 × 75
Kullanıcı belleği	Çalışma belleği	75 kB	100 kB	150 kB	200 kB	250 kB
	Yükleme belleği	1 MB	2 MB	4 MB		
	Kalıcı	14 kB				
Yerel entegre G/Ç	Dijital	6 giriş/ 4 çıkış	8 giriş/ 6 çıkış	14 giriş/ 10 çıkış		
	Analog	2 giriş			2 giriş/2 çıkış	
Proses görünümünün boyutu	Girişler (E)	1024 Bayt				
	Çıkış (A)	1024 Bayt				
Markör (M)		4096 Bayt		8192 Bayt		
Genişletme: Sinyal modülü (SM)		Yok	2	8		
Sinyal kartı (SB), batarya kartı (BB) veya iletişim kartı (CB)		1				

1 HSC, A/B sayaç modu için yapılandırıldıysa daha düşük olan hız geçerlidir.

2 Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

3 200 kHz'e kadar, SB 1221 DI × 24 V DC 200 kHz ve SB 1221 DI 4 × 5 V DC ile 200 kHz ile kullanılabilir.

Özellik		**CPU 1211C**	**CPU 1212C**	**CPU 1214C**	**CPU 1215C**	**CPU 1217C**
İletişim modülleri (CM) (sol ek ünite)		3				
Hızlı sayaçlar	Toplam	Entegre girişlerin veya SB girişlerinin kullanımı için 6 adede kadar yapılandırılır				
	1 MHz	-				Eb.2 - Eb.5
	100/[1]80 kHz	Ea.0 - Ea.5				
	30/[1]20 kHz	--	Ea.6 - Ea.7	Ea.6 - Eb.5		Ea.6 - Eb.1
	200 kHz[3]					
Darbe çıkışları [2]	Toplam	Entegre girişlerin veya SB çıkışlarının kullanımı için 4 adede kadar yapılandırılır				
	1 MHz	--				Aa.0 - Aa.3
	100 kHz	Aa.0 - Aa.3				Aa.4 - Ab.1
	20 kHz	--	Aa.4 - Aa.5	Aa.4 - Ab.1		--
Memory card		SIMATIC memory card (opsiyonel)				
Veri protokolleri	Adet	Aynı anda maksimum 8 adet açık				
	Boyut	Veri protokolü başına 500 MB veya kullanılabilen maksimum yükleme belleği ile sınırlı olarak				
Gerçek zaman saati tamponlama		20 gün, tip./min. 12 gün, 40 derece C'de (bakım gerektirmeyen yüksek performanslı kondensatör)				
PROFINET Ethernet iletişim portu		1			2	
Yürütme süresi, aritmetik işlemler		2,3 µs/işlem				
Yürütme süresi, Boole işlemleri		0,08 µs/işlem				

[1] HSC, A/B sayaç modu için yapılandırıldıysa daha düşük olan hız geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] 200 kHz'e kadar, SB 1221 DI × 24 V DC 200 kHz ve SB 1221 DI 4 × 5 V DC ile 200 kHz ile kullanılabilir.

Çeşitli CPU modelleri, çok farklı uygulamalar için etkin çözümler oluşturabilmeniz amacıyla çok sayıda performans özelliği ve fonksiyon sunar. Belirli CPU'larla ilgili ayrıntılı bilgileri, Teknik veriler (Sayfa 1160) kısmında bulabilirsiniz.

Tablo 1-2 S7-1200 tarafından desteklenen bloklar, süreler ve sayaçlar

Eleman			**Açıklama**				
Bloklar	Tip		OB, FB, FC, DB				
	Boyut	**CPU varyantı**	**CPU 1211C**	**CPU 1212C**	**CPU 1214C**	**CPU 1215C**	**CPU 1217C**
		Kod blokları	50 KB	64 KB	64 KB	64 KB	64 KB
		Yükleme belleğindeki ve çalışma belleğindeki veri blokları[1]	50 KB	75 KB	100 KB	125 KB	150 KB
		Yükleme belleğindeki veri blokları	256 KB	256 KB	256 KB	256 KB	256 KB
	Adet		Toplam 1024 bloğa kadar (OB'ler + FB'ler + FC'ler + DB'ler)				

[1] Çalışma belleğindeki ve yükleme belleğindeki veri blokları, boş çalışma belleğinin veya yükleme belleğinin boyutunu aşamaz.

[2] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

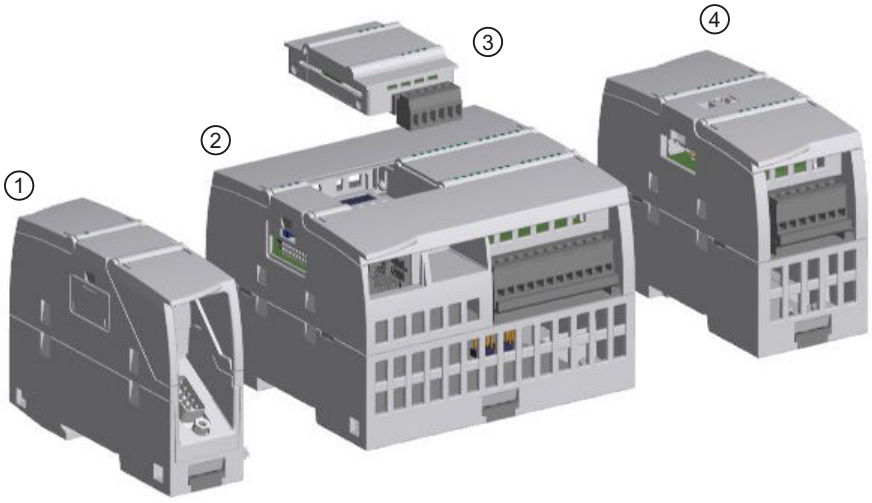
Eleman		Açıklama
Bloklar	Gömme derinliği	Çevrim veya yol alma OB'lerinden 16 adet Rastgele alarm olayı OB'lerinden 6 adet[2]
	Denetim	2 kod bloğunun durumu eş zamanlı olarak denetlenebilir.
OB'ler	Program çevrimi	Birden çok
	Yol alma	Birden çok
	Gecikme alarmları	4 (olay başına 1)
	Uyandırma alarmları	4 (olay başına 1)
	Proses alarmları	50 (olay başına 1)
	Zaman hatası alarmları	1
	Teşhis hatası alarmları	1
	Modüllerin çıkartılması veya takılması	1
	Modül taşıyıcısında veya istasyonda hata	1
	Saat alarmı	Birden çok
	Durum	1
	Güncelleme	1
	Profil	1
Zamanlar	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, süre başına 16 Bayt
Sayaç	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, boyut, sayma türüne bağlıdır • SInt, USInt: 3 Bayt • Int, UInt: 6 Bayt • DInt, UDInt: 12 Bayt

1 Çalışma belleğindeki ve yükleme belleğindeki veri blokları, boş çalışma belleğinin veya yükleme belleğinin boyutunu aşamaz.

2 Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

1.2 CPU'nun genişleme özelliği

S7-1200 ürün ailesi, CPU'nun ek G/Ç'lerle veya başka iletişim protokolleriyle genişletilmesi için çok sayıda modül ve geçmeli kart sunar. Belirli modüllerle ilgili ayrıntılı bilgileri, Teknik veriler (Sayfa 1160) kısmında bulabilirsiniz.



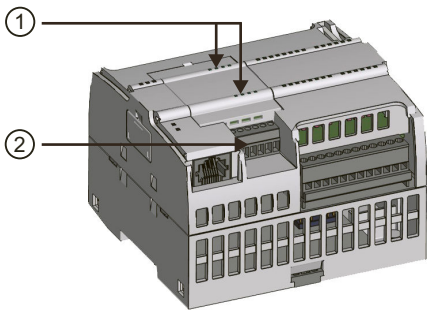
① İletişim modülü (CM) veya iletişim işlemcisi (CP) (Sayfa 1329)

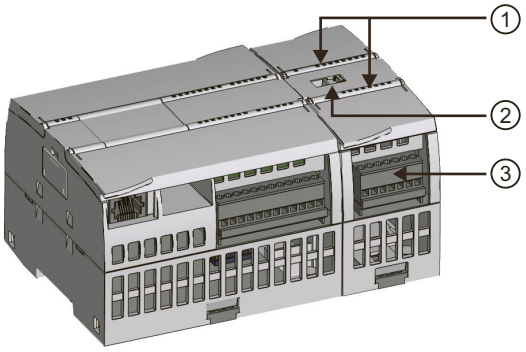
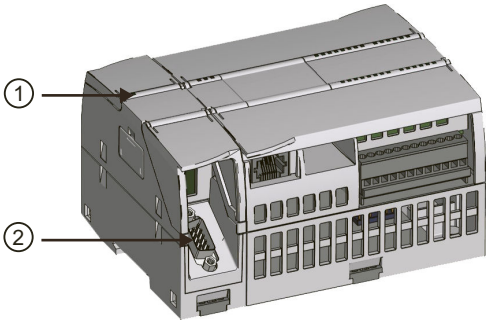
② CPU (CPU 1211C (Sayfa 1172), CPU 1212C (Sayfa 1184), CPU 1214C (Sayfa 1198), CPU 1215C (Sayfa 1212), CPU 1217C (Sayfa 1228))

③ Sinyal kartı (SB) (dijital SB (Sayfa 1299), analog SB (Sayfa 1311)), iletişim kartı (CB) (Sayfa 1339) veya batarya kartı (BB) CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C) (Sayfa 1327)

④ Sinyal modülü (SM) (dijital SM (Sayfa 1243), analog SM (Sayfa 1261), Termo eleman SM (Sayfa 1274), RTD-SM (Sayfa 1279), Teknoloji SM) (Sayfa 1286)

Tablo 1-3 S7-1200 genişletme modülleri

Modülün türü	Açıklama
CPU, geçmeli bir genişletme kartını destekler: • Bir sinyal kartı (SB), CPU'nuz için ek G/Ç sağlar. SB, CPU'nun ön tarafına bağlanır. • Bir iletişim kartı la (CB) kullanarak CPU'nuzu bir iletişim bağlantısıyla genişletebilirsiniz. • Bir batarya kartı (BB), gerçek zaman saatinin uzun süreyle tamponlanmasını sağlar.	 ① SB'deki durum LED'leri ② Kullanıcı kablajı için geçmeli klemens bloğu

Modülün türü	Açıklama
Sinyal modülleri (SM'ler), CPU'yu ek fonksiyonlarla genişletir. SM'ler, CPU'nun sağ tarafına bağlanır. • Dijital G/Ç • Analog G/Ç • RTD ve termo eleman • SM 1278 IO-Link-Master • SM 1238 enerji ölçer (https://support.industry.siemens.com/cs/ww/en/view/109483435)	
	① Durum LED'leri
	② Veriyolu konektörü itme kulağı
	③ Kullanıcı kablajı için geçmeli klemens bloğu
İletişim modülleri (CM'ler) ve iletişim işlemcileri (CP'ler), CPU'yu örn. PROFIBUS veya RS232/RS485 (PtP, Modbus veya USS için) veya AS-i-Master gibi çeşitli iletişim olanaklarıyla genişletir. Bir CP, başka iletişim türleri için olanaklar sunar, örn. CPU'nun bir GPRS, LTE IEC, DNP3 veya WDC ağı üzerinden bağlanması için. • CPU, en fazla üç CM veya CP'yi destekler. • Her CM veya her CP, CPU'nun sol tarafına (veya başka bir CM'nin veya CP'nin sol tarafına) bağlanır.	
	① Durum LED'leri
	② İletişim fişi

1.3 HMI panelleri

SIMATIC HMI panelleri, kontrol ve gözetleme alanlarındaki temel görevler için dokunmatik ekranlı cihazlar sunar. Tüm paneller, IP65 koruma sınıfına sahiptir ve CE, UL, cULus ve NEMA 4x uyarınca sertifikalıdır. Bir HMI cihazını, bir CPU eklediğiniz (Sayfa 128) şekilde projenize ekleyebilirsiniz.

HMI panellerinin siparişi için yetkili satıcınıza başvurun.

Yeni fonksiyonlar

2

Aşağıdaki fonksiyonlar, 4.6 sürümüne yeni eklenmiştir:

- S7-1200 CPU'ları için daha büyük çalışma belleği (Sayfa 27):
  - CPU 1211C, artık 75 kB'a sahip.
  - CPU 1212C, artık 100 kB'a sahip.
  - CPU 1214C, artık 150 kB'a sahip.
  - CPU 1215C, artık 200 kB'a sahip.
  - CPU 1217C, artık 250 kB'a sahip.
  - CPU 1212FC, artık 150 kB'a sahip.
  - CPU 1214FC, artık 200 kB'a sahip.
  - CPU 1215FC, artık 250 kB'a sahip.
- CPU, SNMP (Sayfa 730)'yi standart olarak devre dışı bırakır.

STEP 7 programlama yazılımı

3

STEP 7, TIA Portal'ın programlama ve projelendirme yazılımıdır. TIA Portal, STEP 7'ye ek olarak çalışma zamanına yönelik proses görselleştirmenin planlanması ve yürütülmesi için kullanılan WinCC'yi ve WinCC ve STEP 7 için kapsamlı bir bilgi sistemini de (online yardım) içerir.

STEP 7; uygulamanıza kumanda edilmesi için lojik sisteminin geliştirilmesi, düzenlenmesi ve gözlemlenmesine yönelik kullanıcı dostu bir ortam sunar. PLC ve HMI cihazları gibi, projenizdeki tüm cihazları yapılandırmaya yönelik araçları da sunar. STEP 7, ihtiyaç duyduğunuz bilgileri bulabilmeniz için kapsamlı bir online yardım özelliğine sahiptir.

STEP 7, uygulamanız için kumanda programını rahat ve verimli bir şekilde geliştirebileceğiniz standart programlama dilleri sunar.

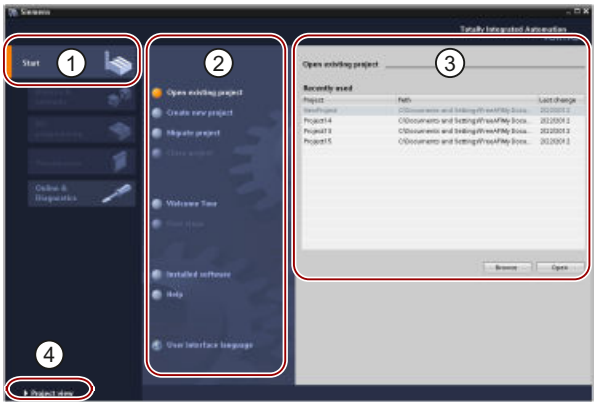
- LAD (kontak planı) (Sayfa 182), bir grafiksel programlama dilidir. Gösterim, devre şemalarını temel alır.
- FBD (fonksiyon planı) (Sayfa 183), Boole cebrinin grafiksel lojik sembollerini temel alan bir programlama dilidir.
- SCL (Structured Control Language) (Sayfa 184); metin tabanlı, daha yüksek seviyede bir programlama dilidir.

Bir kod bloğu oluşturursanız, bu blok için programlama dilini seçmeniz gerekir. Kullanıcı programınız, bu programlama dillerinden birinde oluşturulmuş olan kod bloklarıyla çalışabilir.

STEP 7'nin kurulumuna yönelik minimum sistem gereksinimlerini, TIA Portal bilgi sisteminde bulabilirsiniz.

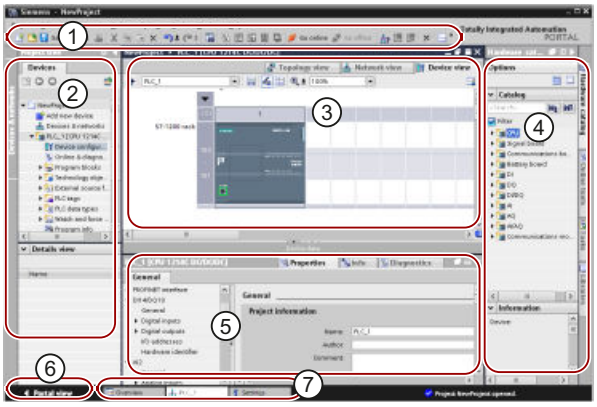
3.1 Farklı görünümlerle kolay çalışma

STEP 7; kumanda lojiğini geliştirebileceğiniz, HMI görselleştirmesini yapılandırabileceğiniz ve ağ iletişimini ayarlayabileceğiniz kullanıcı dostu bir ortam sağlar. STEP 7, verimliliğinizi arttırmak için projenin iki farklı görünümünü sunar: Münferit fonksiyonlar için çalışma odaklı bir görünüm (portal görünümü) ve projedeki elemanların proje odaklı bir görünümü (proje görünümü). En verimli şekilde hangi görünümde çalışabileceğinize kendiniz karar verebilirsiniz. Fareyle tıklayarak portal görünümü ile proje görünümü arasında geçiş yapabilirsiniz.



Portal görünümü

① Çeşitli görevler için portallar
② Seçilen portal için görevler
③ Seçilen işlem için seçme paneli
④ Proje görünümündeki değişiklikler



Proje görünümü

① Menüler ve fonksiyon çubuğu
② Proje gezgini
③ Çalışma alanı
④ Görev kartları
⑤ Denetçi penceresi
⑥ Portal görünümündeki değişiklikler
⑦ Editör çubuğu

Tüm bileşenler tek bir konumda bulunduğu için, projenizin her alanına hızlı bir şekilde erişebilirsiniz. Çalışma alanı, sekmelerde yer alan üç görünümden oluşur:

- Cihaz görünümü: Eklediğiniz veya seçtiğiniz cihazı ve bu cihazın ilgili modüllerini gösterir
- Ağ görünümü: Ağınızdaki CPU'ları ve ağ bağlantılarını gösterir
- Topoloji görünümü: Cihazlar, pasif bileşenler, portlar, ara bağlantılar ve port teşhisi de dahil olmak üzere ağın PROFINET topolojisini gösterir

Her görünümde, konfigürasyon görevlerini de gerçekleştirebilirsiniz. Denetçi penceresinde, çalışma alanında seçilen nesnenin özellikleri ve diğer bilgileri görüntülenir. Denetçi penceresinde, seçtiğiniz çeşitli nesneler için yapılandırılabilen özellikler görüntülenir. Denetçi penceresi ayrıca, teşhis bilgisi ve diğer mesajların görüntülendiği sekmeler içerir.

Editör çubuğunda, mevcut durumda açılmış olan tüm editörler görüntülenir. Bu sayede, editör çubuğuyla daha hızlı ve daha verimli bir şekilde çalışabilirsiniz. Açık editörler arasında geçiş yapmak için istediğiniz editöre tıklamanız yeterlidir. İki editörü aynı anda da

görüntüleyebilir ve bu editörleri dikey veya yatay olarak yerleştirebilirsiniz. Bu sayede, editörler arasında "Sürükle ve bırak" işlemleri gerçekleştirmek mümkün olur.

TIA Portal bilgi sistemi; STEP 7'nin tüm konfigürasyon, programlama ve denetim araçlarıyla ilgili kapsamlı bir online yardım sunar. Bu yardımda, bu işletme kitabında yer almayan ayrıntılı açıklamalar bulabilirsiniz.

3.2 STEP 7 ile S7-1200 arasında uyumluluk

STEP 7, S7-1200 CPU'nun konfigürasyonunu ve programlanmasını destekler.

S7-1200 CPU V4.x'in önceki sürümleri için oluşturulan projeleri STEP 7'den bir S7-1200 CPU V4.x'e yükleyebilirsiniz. Konfigürasyon ve program, S7-1200 CPU'nun önceki sürümü ve STEP 7 sürümünüz tarafından desteklenen fonksiyon ve yordamların kapsamı ile sınırlıdır.

Bu geriye dönük uyumluluk sayesinde, daha önce önceki sürümler için geliştirdiğiniz ve programladığınız programları S7-1200 CPU V4.x varyantlarında çalıştırmaya devam edebilirsiniz.

Projenizi yükseltmeden, V3.0'a kadar olan bir S7-1200 CPU Firmware sürümünden bir S7-1200 CPU V4.x'e hiçbir proje yükleyemezsiniz.

⚠ İKAZ

STEP 7'nin eski sürümlerinden program lojiğini düzenlerken ve yürütürken karşılaşılan riskler

Program lojiğini kopyalayarak sürümü V13'e kadar olan STEP 7'deki bir projeyi güncel sürüme yükseltemezsiniz. STEP 7 projenizin yükseltme işlemini, Bir CPU V3.0'ın CPU V4.x ile değişimi (Sayfa 1365) kısmında açıklandığı şekilde gerçekleştirmeniz gerekir. Daha eski bir sürümden daha yeni bir sürüme kopyaladığınız STEP 7 program lojiğini yürütmek, öngörülemeyen program davranışlarına neden olabilir ve ölümcül veya ciddi yaralanmalara yol açabilir.

NOT

S7-1200 CPU V1.x'in varyantlarıyla hazırlanan projeler

S7-1200 CPU'lar V1.x'in içerdiği bir STEP 7 projesini STEP 7 V14.0'da veya daha yeni bir sürümde açamazsınız. Mevcut projenizi kullanabilmek için, projenizi STEP 7 V13'te (her türlü güncellemeyi içeren) açmanız ve S7-1200 CPU'larını V1.x'i V2.0'a veya daha yeni bir sürüme dönüştürmeniz gerekir. Daha sonra, kaydedilen projeyi dönüştürülen CPU'larla birlikte STEP 7 V14.0'da veya daha yeni bir sürümde açabilirsiniz.

Dijital güvenli iletişim açısından uyumluluk

STEP 7 ve güvenli iletişimle ilgili bilgileri aşağıdaki konularda bulabilirsiniz:

- Dijital güvenlik asistanında PLC dijital güvenlik ayarlarının belirlenmesi (Sayfa 153)
- Güvenli PG/PC iletişimi ve HMI iletişimini etkinleştirme ve sertifikalar oluşturma (Sayfa 158)

Ayrıca bakınız

Cihaz değişimi ve yedek parça uyumluluğu (Sayfa 1365)

Montaj

4

4.1 S71200 cihazlarının montajıyla ilgili direktifler

S7-1200 cihazları kolayca monte edilecek şekilde tasarlanmıştır. Bir S7-1200'ü bir panoya veya standart bir montaj rayına takabilirsiniz; S71200 yatay veya dikey olarak monte edilebilir. S7-1200'ün kompakt boyutu, yeri verimli bir şekilde kullanmaya olanak sağlar.

SIMATIC S7-1200 sistemi, elektrikli cihazlar için hazırlanan standartlarla açık bir donanım olarak sınıflandırılmıştır. S7-1200'ü bir muhafaza, pano veya bir anahtarlama merkezinin içine monte etmeniz gerekir. Muhafazaya, panoya veya anahtarlama merkezine sadece yetkili personelin erişim hakkı olmalıdır.

Kurulum sırasında S7-1200 için kuru bir ortam öngörülmelidir. SELV/PELV elektrik devreleri kuru ortamlarda elektrik çarpmasına karşı koruma sağlamalıdır.

Kurulum, geçerli elektrik tekniği ve bina yönetmeliklerine uygun olarak özel kurulum yeri kategorinizdeki açık donanım için gereken mekanik sağlamlığı, yangın sınıfını ve sağlamlığı sağlamalıdır.

İletken parçaların toz, nem ve hava kirliliği nedeniyle kirlenmesi, PLC'de işletim hatalarına ve elektrik arızalarına yol açabilir.

PLC iletken parçaların kirlenebileceği bir alanda bulunuyorsa, PLC uygun koruma sınıfına sahip bir muhafaza ile korunmalıdır. IP54, genellikle çok kirli ortamlardaki elektronik sistemler için kullanılan bir koruma sınıfıdır ve muhtemelen uygulamanız için uygundur.

⚠ İKAZ
S7-1200'ün usulüne uygun olmayan şekilde kurulumu, elektrik arızalarına veya makinenin beklenmedik şekilde çalışmasına neden olabilir. Elektrik arızaları veya makinenin beklenmedik şekilde çalışması, ciddi veya hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir. Cihazların güvenli bir şekilde çalışmasını sağlamak için düzgün bir çalışma ortamının kurulması ve bakımı ile ilgili tüm talimatlara uyulmalıdır.

S7-1200 cihazlarını ısıdan, yüksek gerilimden ve elektriksel parazitlerden uzak tutun

Sisteminizdeki cihazların yerleşimi için genel bir kural olarak, yüksek gerilim veya yüksek elektriksel parazit üreten cihazları S7-1200 gibi elektronik alçak gerilim cihazlarından uzak tutun.

S7-1200'ün panonuzdaki yerleşim düzenini planlarken, ısı üreten cihazları göz önünde bulundurun ve elektronik cihazları panonuzun daha serin alanlarına yerleştirin. Bir elektronik cihazı yüksek sıcaklıktaki ortamlara ne kadar az maruz bırakırsanız, cihazın kullanım ömrü o kadar uzun olur.

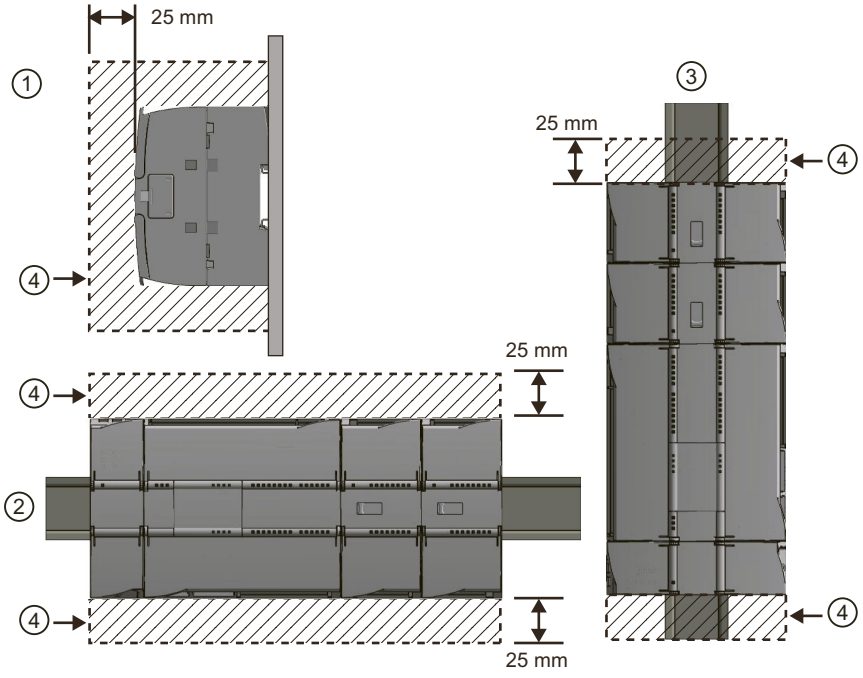
Ayrıca, cihazların kablajını panoda nasıl yapacağınızı da göz önünde bulundurun. Alçak gerilim sinyal hatlarını ve iletişim kablolarını AC besleme hatları ve hızlı anahtarlamalı DC yüksek akım hatları ile aynı kablo tavasına yerleştirmeyin.

Soğutma ve kablaj için yeterli mesafe bırakın

S71200 cihazları, konveksiyon yoluyla doğal ısı tahliyesine uygun şekilde tasarlanmıştır. Bu nedenle, ısı tahliyesinin gerçekleşmesini sağlamak için cihazların üstünde ve altında en az 25'er mm boşluk bırakın. Ayrıca modülün ön tarafı ile muhafazanın iç tarafı arasında en az 25 mm'lik bir derinlik kalmasına dikkat edin.

⚠ DİKKAT
Dikey montajda, izin verilen maksimum ortam sıcaklığı 10 °C daha düşüktür. Dikey olarak monte edilen bir S71200 sistemini aşağıdaki şekilde gösterildiği gibi hizalayın. S7-1200 sisteminin doğru şekilde monte edildiğinden emin olun.

S71200 sisteminiz için yerleşim düzenini planlarken, kablaj ve iletişim kablolarının bağlantıları için yeterli mesafe bırakın.



① Yandan görünüm
② Yatay montaj
③ Dikey montaj
④ Boş alan

4.2 Güç bilançosu

Tüm S7-1200 CPU'lar, CPU'nun kendisinin yanı sıra genişletme modüllerini ve 24 V DC sensör güç çıkışını besleyen dahili bir gerilim beslemesine sahiptir.

Genişletme modülleri: Sinyal modülleri, iletişim modülleri, sinyal kartları, iletişim kartları ve batarya kartları.

Teknik veriler (Sayfa 1160) kısmında, CPU'nuzun 5 V DC güç bilançosu ve çeşitli genişletme modüllerinin 5 V DC güç ihtiyacıyla ilgili bilgileri bulabilirsiniz. 5 V DC güç bilançosu değerini aşarsanız, muhtemelen CPU'nuza maksimum sayıda genişletme modülünü bağlayamazsınız. Güç bilançosu değerine uyulana kadar genişletme modüllerini çıkartmanız gerekir.

NOT

Güç bilançosu değerinin aşılması, sistemin devre dışı kalmasına yol açabilir.

"Güç bilançosunun hesaplanması" (Sayfa 1350) kısmındaki bilgilerin yardımıyla, CPU'nun konfigürasyonunuz için ne kadar bir güç (veya akım) verebileceğini hesaplayabilirsiniz.

Her CPU, entegre girişleri ve genişletme modüllerinin röle bobinlerini veya diğer gerekli bileşenleri 24 V DC gerilimle besleyen bir 24 V DC enkoder beslemesine sahiptir. 24 V DC güç ihtiyacı, enkoder beslemesinin gücünü aşarsa, sisteminize harici bir 24 V DC gerilim beslemesi bağlayabilirsiniz. Harici 24 V DC beslemeyi manuel olarak girişlere ve genişletme modüllerindeki röle bobinlerine bağlamanız gerekir. Teknik veriler (Sayfa 1160) kısmında, ilgili CPU'nuz için kullanılan 24 V DC enkoder beslemesinin güç bilançosunu bulabilirsiniz.

Harici bir 24 V DC gerilim beslemesine ihtiyaç duyarsanız, gerilim beslemesinin, CPU'nun enkoder beslemesine paralel bağlanmamasına dikkat etmeniz gerekir. En iyi parazite dayanıklılığı, ilgili gerilim beslemelerinin hatlarını (M) birbirlerine bağlayarak elde edebilirsiniz.

⚠ İKAZ

Paralel devrelerde ortaya çıkan riskler

24 V DC enkoder beslemesine harici bir 24 V DC gerilim kaynağını paralel olarak bağlarsanız, uygun çıkış gerilimi oluşturulurken iki gerilim kaynağı birbiriyle etkileşime girebilir.

Bunun sonucunda, kullanım ömrü kısalabilir veya iki gerilim kaynağından biri anında devre dışı kalabilir. Öngörülemeyen işletim, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilir.

DC enkoder beslemesi ve harici bir gerilim kaynağı, gerilimi farklı noktalara iletmek zorundadır.

S7-1200 sisteminin 24 V DC giriş portlarından bazıları birbirine bağlıdır; bu sırada, mantıksal bir referans iletken çok sayıda M klemensini bağlar. Örneğin aşağıdaki elektrik devreleri, bilgi formlarında "potansiyeli ayrı değil" olarak belirtilmişlerse birbirlerine bağlıdır: CPU'nun 24 V DC beslemesi, bir SM'nin röle bobini için kullanılan güç girişi veya potansiyeli ayrı olmayan bir analog girişin beslemesi. Potansiyeli ayrı olmayan tüm M klemensleri aynı harici referans potansiyeline bağlanmalıdır.

⚠ İKAZ

Potansiyeli ayrı olmayan M klemenslerinin farklı referans potansiyellerine bağlanması durumunda ortaya çıkan riskler

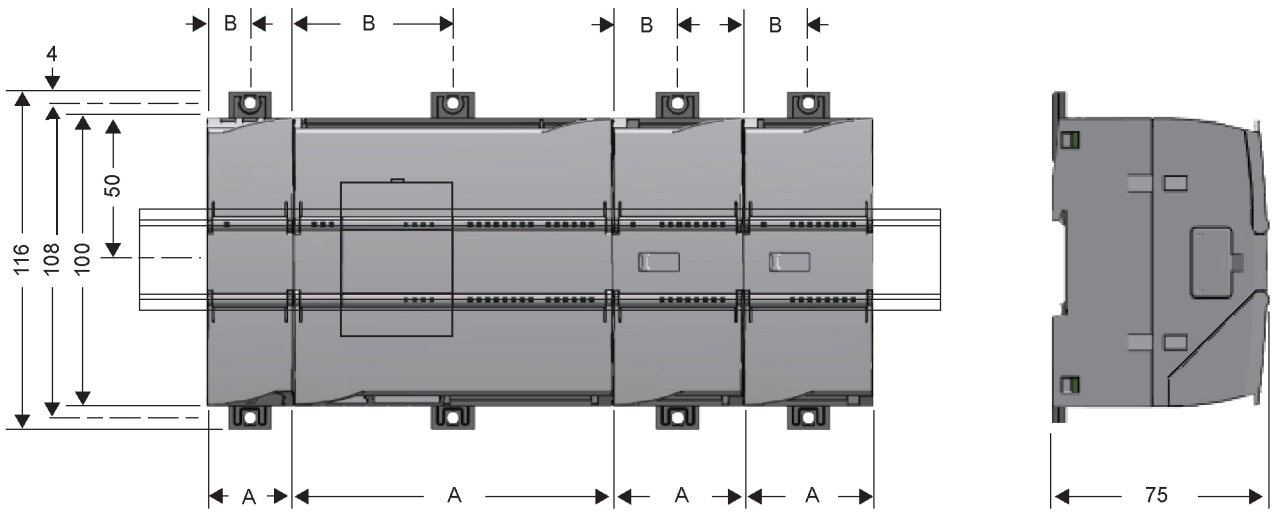
Potansiyeli ayrı olmayan M klemenslerini farklı referans potansiyellerine bağlarsanız, bu durum öngörülemeyen bir akım akışına neden olur. Öngörülemeyen akım akışı, hasarlara veya PLC veya cihazın hatalı şekilde çalışmasına ve bu da hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Bir S7-1200 sisteminde, potansiyeli ayrı olmayan tüm M klemenslerini her zaman aynı referans potansiyeline bağlayın.

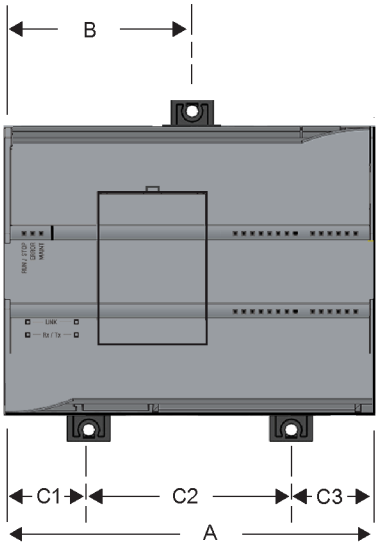
4.3 Takma ve sökme sırasında izlenecek prosedür

4.3.1 S7-1200 cihazları için sabitleme ölçüleri

CPU 1211C, CPU 1212C, CPU 1214C
(measurements in mm)



CPU 1215C, CPU 1217C
(measurements in mm)



Tablo 4-1 Montaj boyutları (mm)

S71200 cihazları		Genişlik A (mm)	Genişlik B (mm)	Genişlik C (mm)
CPU	CPU 1211C ve CPU 1212C	90	45	--
	CPU 1214C	110	55	--

[1] IE Advanced veya IE Basic TS (TeleService) adaptörünü monte etmeden önce TS adaptörünü ve bir TS modülünü bağlamanız gerekir. Toplam genişlik ("A genişliği") 60 mm'dir.

S71200 cihazları		**Genişlik A (mm)**	**Genişlik B (mm)**	**Genişlik C (mm)**
CPU	CPU 1215C	130	65 (üst)	Alt: C1: 32,5 C2: 65 C3: 32,5
	CPU 1217C	150	75	Alt: C1: 37,5 C2: 75 C3: 37,5
Sinyal modülleri	8 veya 16 dijital G/Ç 2, 4 veya 8 analog G/Ç Termo eleman, 4 veya 8 G/Ç RTD, 4 G/Ç SM 1278 IO Link-Master	45	22,5	--
	8 dijital çıkış × röle (geçiş)	70	35	--
	16 analog G/Ç RTD, 8 G/Ç	70	35	--
	SM 1238 enerji ölçer	45	22,5	--
İletişim arayüzleri	CM 1241 RS232 ve CM 1241 RS422/485 CM 1243-5 PROFIBUS-Master ve CM 1242-5 PROFIBUS-Slave CM 1242-2 AS-i-Master CP 1242-7 V2 CP 1243-7 LTE-US CP 1243-7 LTE-EU CP 1243-1 CP 1243-8 IRC RF120C	30	15	--
	IE Advanced TS (TeleService) adaptörü [1] IE Basic TS (TeleService) adaptörü [1] TS adaptörü TS modülü	 30 30	 15 15	 -- --

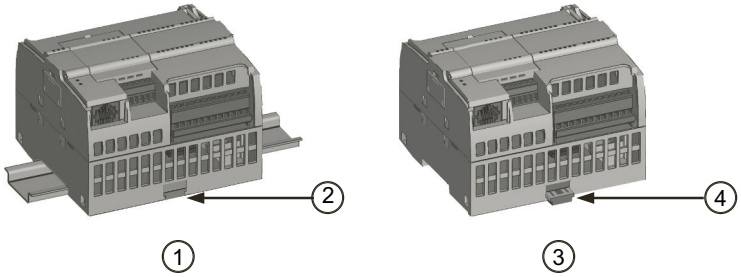
[1] IE Advanced veya IE Basic TS (TeleService) adaptörünü monte etmeden önce TS adaptörünü ve bir TS modülünü bağlamanız gerekir. Toplam genişlik ("A genişliği") 60 mm'dir.

Tüm CPU'lar, SM'ler, CM'ler ve CP'ler montaj rayına veya panoya monte edilebilir. Cihazı montaj rayına sabitlemek için montaj rayı klemenslerini kullanın. Bu klemensler, cihazın bir panoya montajına olanak sağlamak için dışarı çekilmiş pozisyonda da oturur. Cihazdaki montaj rayı klemensleri için açılan delik 4,3 mm'dir.

Cihazın üst ve alt kısmında, aşırı ısınmaya karşı koruma önlemi olarak hava sirkülasyonu için 25 mm'lik bir boş alan bırakılmalıdır.

S71200 cihazlarının takılması ve sökülmesi

CPU, kolay bir şekilde bir standart montaj rayının üzerine veya bir panoya takılabilir. Cihazı montaj rayına sabitlemek için uygun montaj rayı klemensleri birlikte teslim edilir. Bu klemensler, dışarı çekilmiş pozisyonda da yerine oturabilir ve ardından cihazın bir panonun içine vidalanmasına olanak sağlar.



① Montaj rayına montaj
② Standart montaj rayı klemensi kilitlenmiş pozisyonda
③ Pano montajı
④ Klemens, pano montajı için dışarı çekilmiş pozisyonda

Bir elektrikli cihazı takmadan veya sökmeden önce, cihazın gerilim beslemesinin kapalı olduğundan emin olmanız gerekir. Ayrıca bağlı tüm cihazların da kapalı olmasına dikkat edin.

⚠ İKAZ

S7-1200'ü veya ona bağlı cihazları açık durumdayken takar veya sökerseniz, sizi elektrik çarpabilir veya cihazlar beklenmedik şekilde çalışabilir.

Cihazlar takılırken veya sökülürken S7-1200'ün ve ona bağlı tüm cihazların gerilim beslemesi kapalı değilse, elektrik çarpması veya cihazların beklenmedik şekilde çalışması nedeniyle hayati tehlikeye yol açabilecek yaralanmalar ve/veya maddi hasarlar oluşabilir.

Gereken tüm güvenlik önlemlerini alın ve bir cihaz takılmadan veya sökülmeden önce S7-1200 CPU'ların gerilim beslemesinin kapalı olduğundan emin olun.

Bir S7-1200 cihazını takarken veya sökerken her zaman doğru modülü veya doğru cihazı kullanmaya dikkat edin.

⚠ İKAZ

Bir S7-1200 modülünün yanlış takılması, S7-1200'ün programının öngörülemeyen şekilde çalışmasına neden olabilir.

Bir S7-1200 cihazı, doğru şekilde hizalanmamış veya yanlış sırayla takılmış olan başka bir varyantla değiştirilirse, bu durum, cihazların beklenmeyen şekilde çalışması nedeniyle hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilir.

Bir S7-1200 cihazını her zaman aynı model cihazla değiştirin, cihazı doğru şekilde hizalayın ve doğru konuma takın.

⚠ İKAZ

Tutuşucu veya yanıcı bir atmosfer mevcutsa hiçbir cihazın bağlantısını kesmeyin.

Tutuşucu veya yanıcı bir atmosferde cihazların bağlantısının kesilmesi, yangına veya patlamaya ve bu durum da ölüme, ağır yaralanmalara ve/veya maddi hasarlara yol açabilir.

Tutuşucu veya yanıcı bir atmosfer mevcutsa her zaman uygun güvenlik önlemlerinin alınmasına dikkat edin.

NOT

Elektrostatik deşarjlar, cihazların veya CPU'da bu cihazlar için öngörülen yuvanın hasar görmesine neden olabilir.

Cihazla çalışırken iletken, topraklı bir zeminin üzerinde durmanız ve topraklı bir bileklik takmanız gerekir.

4.3.2 CPU'nun takılması ve sökülmesi

CPU'yu bir panonun içine veya bir standart montaj rayının üzerine takabilirsiniz.

NOT

İletişim modüllerini CPU'ya bağlayın ve tüm modülleri birlikte takın. Sinyal modüllerini ancak CPU monte edildikten sonra takın.

Cihazları montaj rayının üzerine veya bir panonun içine takarken dikkat etmeniz gerekenler:

- Bir montaj rayına takma sırasında, CPU'nun üst montaj rayı klemensinin ve bağlı CM'lerin yerine oturmuş (iç) ve alt montaj rayı klemensinin dışarı çekilmiş konumda olmasına dikkat edin.
- Cihazlar montaj rayının üzerine takıldıktan sonra, cihazları montaj rayının üzerinde sabitlemek için alt montaj rayı klemenslerini yerine oturmuş konuma getirin.
- Pano montajında, montaj rayı klemenslerinin dışarı çekilmiş konumda olmasına dikkat edin.

CPU'yu bir panoya takmak için aşağıdaki yöntemi izleyin:

1. Montaj boyutları (mm) (Sayfa 41-42) tablosundaki boyutlarda gösterilen spesifikasyonlarla delikleri açın (M4).
2. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun.
3. Montaj klemenslerini modülden dışarıya çekin. CPU'da üst ve alt montaj rayı klemenslerinin dışarı çekilmiş pozisyonda olmasına dikkat edin.
4. Modülü bir M4 silindir başlı cıvata, bir yay halka ve yassı bir düz rondelayla vidalayın. Havşa başlı vida kullanmayın.

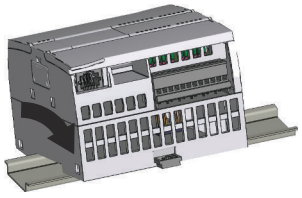
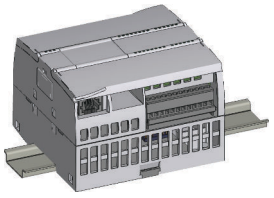
   NOT

   Hangi vida tipinin gerekli olduğu, modülün üzerine takılacağı malzemenin türüne bağlıdır. Yay halka düzleşene kadar vidayı uygun torkla sıkın. Vidaları aşırı bir torkla sıkmayın. Havşa başlı vidalar kullanmayın.

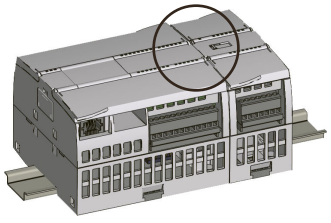
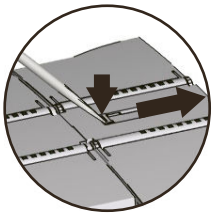
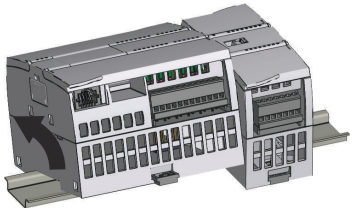
NOT

Güçlü titreşimlerin oluştuğu ortamlarda veya CPU dikey olarak monte edilirken, CPU'nun montaj rayının üzerinde montaj rayı klemensleriyle emniyete alınması faydalı olabilir. Modüllerin birbirine bağlı şekilde kalmasını sağlamak için bir montaj rayının üzerine uç tutucu (8WA1808 veya 8WA1805) takın. Güçlü titreşimlerin oluştuğu ortamlarda CPU'nun panoya takılması, titreşimlere karşı daha iyi bir koruma sağlar.

Tablo 4-2 CPU'nun bir montaj rayının üzerine takılması

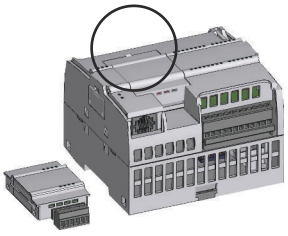
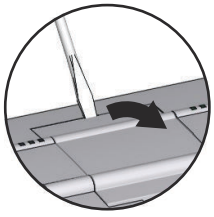
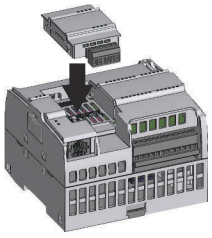
Görevler	İzlenecek prosedür
 	1. Montaj rayını takın. Montaj rayını, panoyla arada 75'er mm mesafeler olacak şekilde vidalayın. 2. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 3. CPU'yu montaj rayının üst kısmına asın. 4. CPU'nun rayın üzerine oturması için CPU'nun alt tarafındaki montaj rayı klemensini dışarıya çekin. 5. CPU'yu aşağıya doğru, rayın üzerindeki montaj pozisyonuna döndürün. 6. Klemensleri, CPU, rayın üzerine sıkıca oturacak şekilde aşağıya doğru bastırın.

Tablo 4-3 Bir montaj rayındaki CPU'nun sökülmesi

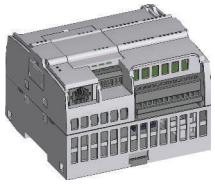
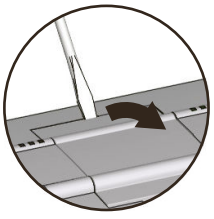
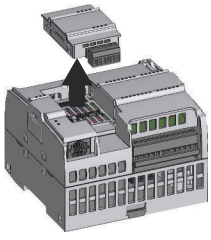
Görevler	İzlenecek prosedür
  	1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Giriş/çıkış soket bağlayıcıyı, kablajı ve kabloları CPU (Sayfa 50)'dan ayırın. 3. CPU'yu ve ilgili iletişim modüllerini birlikte sökün. Tüm sinyal modülleri takılı durumda kalmalıdır. 4. CPU'ya bir SM bağlıysa veriyolu konektörünü çekin: – Sinyal modülünün üst tarafındaki kulağın üzerine bir tornavida yerleştirin. – Klemens bloğunu CPU'dan çözmek için aşağı doğru bastırın. – Kulağı en sağa itin. 5. CPU'yu çıkartın: – CPU'nun raydan çözülmesi için montaj rayı klemensini dışarıya çekin. – CPU'yu raydan aşağıya doğru yukarı yönde döndürün ve CPU'yu sistemden çıkartın.

4.3.3 Bir SB'nin, CB'nin veya BB'nin takılması ve sökülmesi

Tablo 4-4 Bir SB'nin, CB'nin veya BB 1297'nin takılması

Görevler	İzlenecek prosedür
  	1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Klemens bloğunun üst ve alt kapağını CPU'dan çıkartın. 3. Kapağın arka tarafında CPU'nun üstündeki açıklığa bir tornavida sokun. 4. Kapağı, tutucusundan dikkatli bir şekilde yukarıya doğru düz kaldırın ve CPU'dan çıkartın. 5. Modülü yukarıdan CPU'daki montaj pozisyonuna düz şekilde yerleştirin. 6. Modülü, yerine oturana kadar aşağı doğru sıkıca bastırın. 7. Klemens kapaklarını yerleştirin.

Tablo 4-5 Bir SB'nin, CB'nin veya BB 1297'nin sökülmesi

Görevler	İzlenecek prosedür
  	1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Klemens bloğunun üst ve alt kapağını CPU'dan çıkartın. 3. Sinyal kartının soket bağlayıcısını, bir tornavidayla hafifçe dışarı doğru bastırarak çıkartın (mevcutsa). 4. Modülün üst tarafındaki açıklığa bir tornavida sokun. 5. Modülü, dikkatli bir şekilde tutucusundan kaldırın ve CPU'dan çıkartın. 6. Modülü, tornavida kullanmadan, montaj pozisyonunda üstten düz bir şekilde CPU'dan çıkartın. 7. CPU'nun kapağını yerine takın. 8. Klemens kapaklarını yerleştirin.

BB 1297 batarya kartının bataryasının takılması veya değiştirilmesi

BB 1297, CR1025 tipinde bir bataryaya ihtiyaç duyar. Batarya, BB 1297'nin teslim kapsamına dahil değildir ve ayrıca satın alınmalıdır. Bataryayı takmak veya değiştirmek için aşağıdaki yöntemi izleyin:

1. Yeni bir bataryayı, bataryanın pozitif tarafı yukarıya bakacak ve negatif tarafı üzeri baskılı devre elemanı plakasının yanında olacak şekilde BB 1297'ye yerleştirin.
2. BB 1297 şimdi CPU'ya yerleştirilebilir. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun ve BB 1297'yi takmak için yukarıda açıklanan yöntemi izleyin.

BB 1297'nin bataryasını aşağıdaki şekilde değiştirebilirsiniz:

1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. BB 1297'yi yukarıda açıklandığı şekilde CPU'dan çıkartın.
2. Eski bataryayı küçük bir tornavida yardımıyla dikkatli bir şekilde çıkartın. Bataryayı klemensin altından dışarı doğru itin.
3. Yeni bir CR1025 bataryayı, bataryanın pozitif tarafı yukarıya bakacak ve negatif tarafı üzeri baskılı devre elemanı kartının yanında olacak şekilde yerleştirin.
4. BB 1297 batarya kartını yukarıda açıklandığı şekilde takın.

⚠ İKAZ

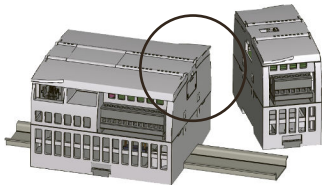
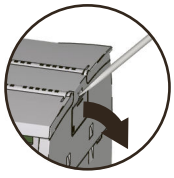
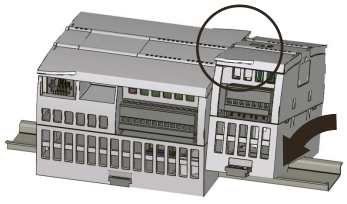
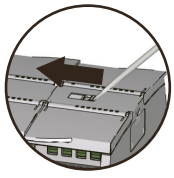
Belirtilmeyen bir bataryanın BB 1297'ye takılması veya belirtilmeyen bir bataryanın anahtarlama devresine başka bir şekilde bağlanması, yangına veya bileşenlerin hasar görmesine ve makinelerin öngörülemeyen şekilde çalışmasına neden olabilir.

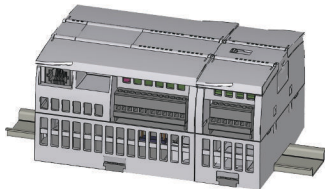
Yangın veya makinelerin öngörülemeyen bir şekilde çalışması, hayati tehlikeye yol açabilecek yaralanmalara veya maddi hasarlara neden olabilir.

Gerçek zaman saatini tamponlamak için sadece belirtilen CR1025 bataryaları kullanın.

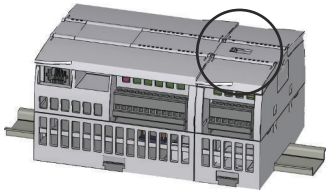
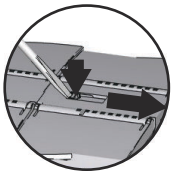
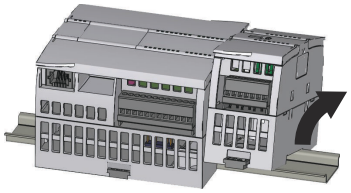
4.3.4 Bir SM'nin takılması ve sökülmesi

Tablo 4-6 Bir SM'nin takılması

Görevler		İzlenecek prosedür
		SM'nizi, CPU'nun montajından sonra takın. 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. CPU'nun sağ tarafındaki bağlantı için kullanılan kapağı çıkartın: – Kapağın üzerindeki açıklığa bir tornavida sokun. – Kapağı dikkatli bir şekilde tutucusundan yukarıya doğru kaldırın ve çıkartın. 3. Daha sonra kullanmak üzere kapağı saklayın.
		SM'nin CPU'ya bağlanması: 1. SM'yi CPU'nun yanına yerleştirin. 2. SM'yi yukarıya, montaj rayının üzerine asın. 3. SM'nin rayın üzerine oturması için alt montaj rayı klemensini dışarıya çekin. 4. SM'nin montaj rayının üzerine sıkıca oturması için, SM'yi, CPU'nun yanındaki pozisyona doğru aşağı yönde döndürün ve alt klemensi aşağı doğru bastırın.

Görevler	İzlenecek prosedür
	Bu sayede, veriyolu konektörü, SM için mekanik ve elektriksel bağlantıyı sağlar. 1. SM'nin üst tarafındaki kulağın üzerine bir tornavida yerleştirin. 2. Veriyolu konektörünü CPU'ya doğru itmek için kulağı en sola itin. Bir sinyal modülünün üzerine bir sinyal modülü takmak için aynı yöntemi izleyin.

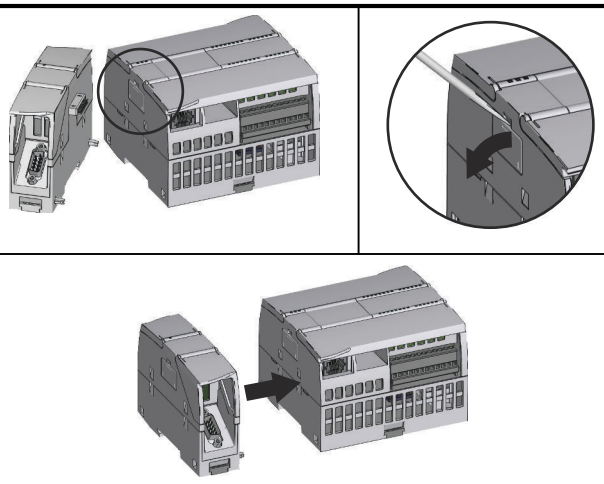
Tablo 4-7 Bir SM'nin sökülmesi

Görevler	İzlenecek prosedür
 	Her bir SM'yi, CPU'yu veya diğer SM'leri sökmek zorunda kalmadan sökebilirsiniz. 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. G/Ç soket bağlayıcıyı ve kablajı SM (Sayfa 50)'den çekerek çıkartın. 3. Veriyolu konektörünü geriye çekin. – SM'nin üst tarafındaki kulağın üzerine bir tornavida yerleştirin. – Klemens bloğunu CPU'dan çözmek için aşağı doğru bastırın. – Kulağı en sağa itin. Sağ tarafa başka bir SM takıldıysa, bu yöntemi SM için tekrarlayın.
	SM'yi çıkartın: 1. SM'yi raydan çıkartmak için alt montaj rayı klemensini dışarı doğru çekin. 2. SM'yi yukarı ve raydan aşağıya doğru döndürün. SM'yi sistemden çıkartın. 3. Gerekirse, kirlenmeyi önlemek için CPU'nun veriyolu konektörüne bir kapak takın. Bir sinyal modülünü başka bir sinyal modülünden çıkartmak için aynı yöntemi izleyin.

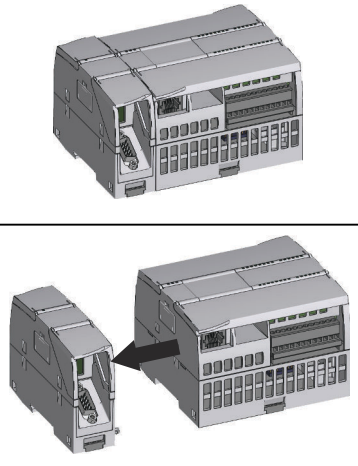
4.3.5 Bir CM veya CP'nin takılması ve sökülmesi

İletişim modüllerini CPU'ya bağlayın ve tüm modülleri birlikte takın. Bu, CPU'nun takılması ve sökülmesi (Sayfa 45) bölümünde gösterilmektedir.

Tablo 4-8 Bir CM veya CP'nin takılması

Görevler	İzlenecek prosedür
	1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. CM'yi CPU'ya bağlayın ve tüm modülleri birlikte montaj rayının üzerine veya panoya takın. 3. CPU'nun sol tarafındaki veriyolu kapağını çıkartın: – Veriyolu kapağının üzerindeki açıklığa bir tornavida sokun. – Kapağı dikkatli bir şekilde tutucusundan yukarıya doğru kanırtın. 4. Veriyolu kapağını çıkartın. Daha sonra kullanmak üzere kapağı saklayın. 5. CM veya CP'nin CPU'ya bağlanması: – CM'nin veriyolu konektörünü ve pimlerini CPU'daki deliklerle hizalayın. – Parçaları, pimler yerine oturana kadar sıkıca birbirlerine doğru bastırın. 6. CPU'yu ve CP'yi bir montaj rayının üzerine veya bir panoya takın.

Tablo 4-9 Bir CM veya CP'nin sökülmesi

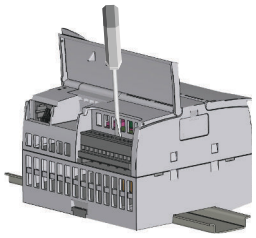
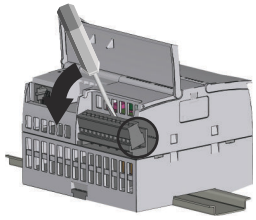
Görevler	İzlenecek prosedür
	CPU'yu ve CM'yi birlikte standart montaj rayından veya panodan çıkartın. 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. G/Ç soket bağlayıcıyı, kablajı ve kabloları CPU'dan ve CM'lerden çekerek çıkartın. 3. Montaj rayına montajda, CPU'daki ve CM'lerdeki alt montaj rayı klemenslerini dışarı çekilmiş konuma getirin. 4. CPU'yu ve CM'leri standart montaj rayından veya panodan çıkartın. 5. CPU'yu ve CM'leri sıkıca tutun ve birbirlerinden dışarıya doğru çekerek ayırın.

DİKKAT
Modüllerin ayrılması Modülleri ayırmak için bir alet kullanmayın, çünkü bu nedenle modüller hasar görebilir.

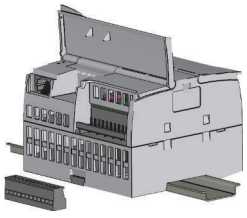
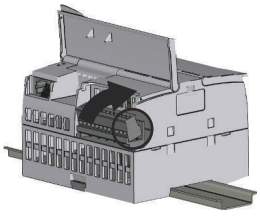
4.3.6 S7-1200 klemens bloğunun sökülmesi ve takılması

CPU, SB ve SM, kablajı kolaylaştırmak için çıkartılabilir soket bağlayıcılar içerir.

Tablo 4-10 Klemens bloğunun sökülmesi

Görevler	İzlenecek prosedür
 	CPU'nun gerilimini kapatarak ve klemens bloğunun üst kısmındaki kapağı açarak, klemens bloğunu sökmek için sistemi hazırlayın. 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Klemens bloğunun üst tarafında, tornavidanın sivri kısmı için kullanılan açıklığı arayın. 3. Açıklığa bir tornavida sokun. 4. Klemens bloğunun üst tarafını dikkatli şekilde CPU'daki tutucudan kaldırın. Klemens bloğu gevşediğinde bir klik sesi duyulur. 5. Klemens bloğunu elinizle tutun ve CPU'dan dışarıya çekin.

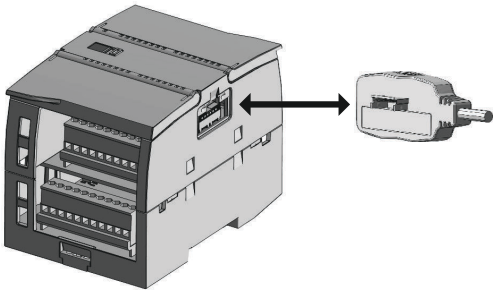
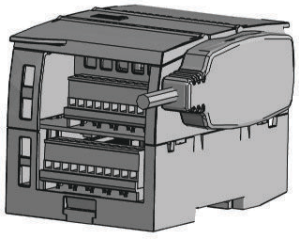
Tablo 4-11 Klemens bloğunun takılması

Görevler	İzlenecek prosedür
 	CPU'nun gerilimini kapatarak ve soket bağlayıcının kapağını açarak, klemens bloğunun takmak için kullanılan bileşenleri hazırlayın. 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Klemens bloğunu cihazdaki pimlerle hizalayın. 3. Klemens bloğunun kablaj tarafını soket modülde doğru şekilde hizalayın. 4. Klemens bloğunu, yerine oturana kadar bir döndürme hareketiyle sıkıca aşağıya doğru bastırın. Klemens bloğunun doğru hizalanıp hizalanmadığını ve sıkı oturup oturmadığını kontrol edin.

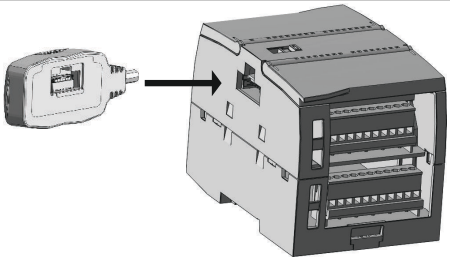
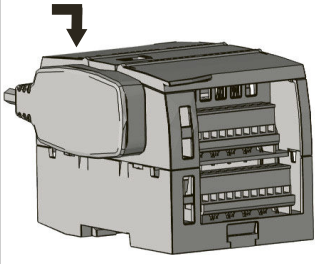
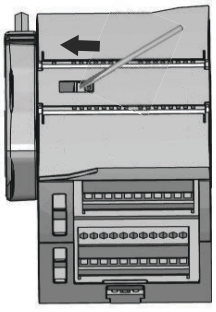
4.3.7 Uzatma kablosunun takılması ve sökülmesi

S7-1200 uzatma kablosu, S7-1200 sisteminizin yerleşim düzeni yapılandırılırken ek esneklik sağlar. CPU sistemi başına bir uzatma kablosu kullanabilirsiniz. Uzatma kablosunu CPU ile ilk SM'nin arasına ya da ikinci SM'nin arasına takabilirsiniz.

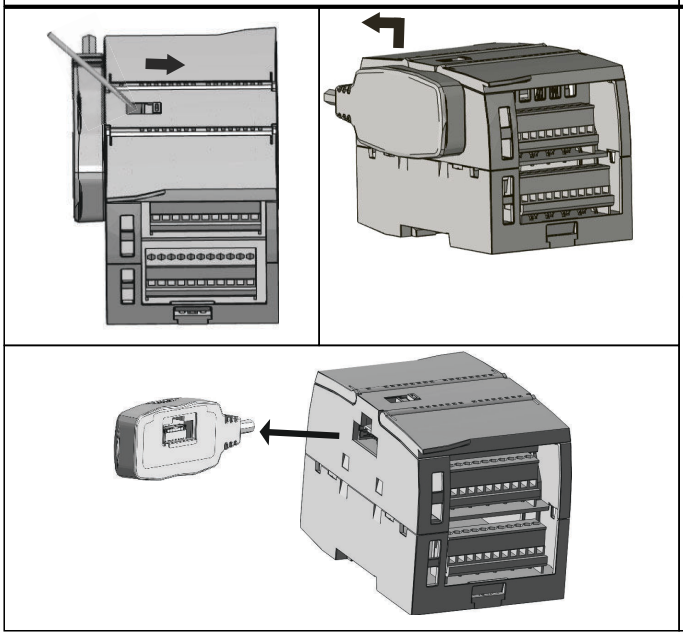
Tablo 4-12 Uzatma kablosu konnektörünün takılması ve sökülmesi

Görevler	İzlenecek prosedür
	Konnektörün takılması: 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Soket bağlayıcıyı, sinyal modülünün veya CPU'nun sağ tarafındaki veriyolu bağlantısına doğru bastırın. Konnektörün sökülmesi: 1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Konnektörü sinyal modülünden veya CPU'dan çıkartın.
	

Tablo 4-13 Uzatma kablosu soketinin takılması

Görevler	İzlenecek prosedür
	1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Soketi, sinyal modülünün sol tarafındaki veriyolu bağlantısına takın. 3. Soketin kanca uzantısını veriyolu bağlantısındaki muhafazanın içine doğru itin ve kancanın oturması için dikkatli bir şekilde bastırın. 4. Fiş-Priz bağlantısını sabitleyin: – Sinyal modülünün üst tarafındaki kulağın üzerine bir tornavida yerleştirin. – Kulağı en sola itin. Soket bağlayıcının oturması için, soket bağlayıcının kulağını en sola itmeniz gerekir. Soket bağlayıcı kulağı tamamen oturmalıdır.
 	

Tablo 4-14 Uzatma kablosu soketinin sökülmesi

Görevler	İzlenecek prosedür
	1. CPU'nun ve tüm S7-1200 cihazlarının elektrik gücü bağlantısının kesildiğinden emin olun. 2. Soket bağlayıcıyı gevşetin: – Sinyal modülünün üst tarafındaki kulağın üzerine bir tornavida yerleştirin. – Tornavidayı hafifçe aşağıya doğru bastırın ve kulağı en sağa itin. 3. Kancanın gevşemesi için soket bağlayıcıyı hafifçe kaldırın. 4. Soketi çıkartın.

NOT

Uzatma kablosunun titreşim yükü bulunan ortamlarda takılması

Uzatma kablosu hareket eden veya sıkıca takılmamış modüllere bağlanırsa, kablonun konnektöründeki kilit bağlantısı zaman içinde gevşeyebilir.

Bu durumda, ek çekme koruması sağlamak için konnektör tarafındaki kablo sonunu bir kablo bağıyla montaj rayının (veya başka bir taşıyıcının) üzerinde sabitleyin.

Takma işlemi sırasında kablonun çekilmesi gerekirse aşırı kuvvet uygulamayın. Takma işlemi tamamlandıktan sonra, kablo ile modül arasındaki bağlantının doğru pozisyonda olduğundan emin olun.

4.4 Kablaj direktifleri

Tüm elektrikli cihazların topraklama ve kablaj işlemlerin usulüne uygun şekilde yapılması, sisteminizin optimum şekilde çalışması ve uygulamanız ve S7-1200'ün ek parazite dayanıklılığı için önemlidir. S7-1200'ün devre şemalarını teknik verilerde (Sayfa 1160) bulabilirsiniz.

Ön koşul

Bir elektrikli cihazı topraklamadan veya cihazın kablajını yapmadan önce, cihazların gerilim beslemesinin kapalı olduğundan emin olmanız gerekir. Ayrıca bağlı tüm cihazların da kapalı olmasına dikkat edin.

S7-1200'ün ve bağlı tüm cihazların kablajı sırasında yürürlükteki ve bağlayıcı tüm standartlara uyduğunuzdan emin olun. Cihazların takılması ve işletimi sırasında ilgili ulusal ve bölgesel yönetmeliklere dikkat edin. Özel durumunuzda uyulması gereken standartları ve yönetmelikleri yerel makamlardan öğrenin.

⚠ İKAZ

S7-1200'ü veya ona bağlı cihazları açık durumda takar veya kablolarsanız, elektrik çarpabilir veya cihazlar beklenmedik şekilde çalışabilir.

Cihazlar takılırken veya sökülürken S7-1200'ün ve ona bağlı tüm cihazların gerilim beslemesi kapalı değilse, elektrik çarpması veya cihazların beklenmedik şekilde çalışması nedeniyle hayati tehlikeye yol açabilecek yaralanmalar ve/veya maddi hasarlar oluşabilir.

Gereken tüm güvenlik önlemlerini alın ve bir cihaz takılmadan veya sökülmeden önce S7-1200'ün gerilim beslemesinin kapalı olduğundan emin olun.

S7-1200 sisteminizin topraklama ve kablaj planlaması sırasında her zaman güvenliği düşünün. S7-1200 gibi elektronik kumanda cihazları arızalanabilir ve bunun sonucunda, kumanda edilen veya gözlemlenen cihazların beklenmedik şekilde çalışmasına neden olabilir. Bu nedenle, S7-1200'den bağımsız olan ve insanları olası zarar görme durumlarına ve/veya olası maddi hasarlara karşı koruyan güvenlik tertibatları kullanmanız gerekir.

⚠ İKAZ

Kumanda üniteleri, güvenli olmayan işletim durumlarında arızalanabilir ve bunun sonucunda, kumanda edilen cihazların kontrolsüz şekilde çalışmasına neden olabilir.

Öngörülemeyen işletim, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilir.

Bu nedenle, S7-1200'den bağımsız bir ACİL KAPAT fonksiyonu, elektro mekanik veya başka yedekli güvenlik tertibatları sağlayın.

Potansiyel ayırmayla ilgili direktifler

AC gerilim beslemesinin limitleri ve S7-1200'ün AC elektrik devrelerinin G/Ç sınırları, AC şebeke gerilimleri ile alçak gerilim devrelerinin arasında güvenli bir elektriksel ayırma olanağı sunmak için tasarlanmış ve onaylanmıştır. Standarda bağlı olarak bu sınırlar, çift veya güçlendirilmiş izolasyonu veya temel artı ek izolasyonu kapsamaktadır. Örn. optokuplörler, kondensatörler, trafolar ve röleler gibi bu sınırları aşan bileşenler, güvenli elektriksel ayırma olanağı sunmak için ruhsatlıdır. Sadece AC şebeke gerilimi için tasarlanan elektrik devreleri, diğer elektrik devreleriyle güvenlik temelli potansiyel ayırmayı kapsar. 24 V DC elektrik devrelerinin arasındaki potansiyel ayırma sınırları salt fonksiyoneldir ve güvenlik açısından bu sınırlara güvenmemeniz gerekir.

Entegre AC gerilim beslemesine sahip bir S7-1200'ün enkoder beslemesi çıkışı, iletişim elektrik devreleri ve dahili lojik elektrik devreleri, EN 61131-2 uyarınca güvenlikli çok düşük gerilim (SELV) olarak beslenir.

S7-1200'ün alçak gerilim devrelerinin güvenli özelliğini korumak için; iletişim portlarındaki harici bağlantılar, analog elektrik devreleri ve tüm 24 V DC nominal gerilim beslemesi ve tüm G/Ç elektrik devreleri, SELV, PELV, Sınıf 2, gerilim sınırlamalı veya güç sınırlamalı kaynaklar için tanımlanan çeşitli standartlar kapsamındaki gereklilikleri karşılayan onaylanmış kaynaklardan beslenmelidir.

⚠ İKAZ

AC hatlarından gelen alçak gerilim devreleri için potansiyeli ayrı olmayan veya tek yalıtımlı beslemelerin kullanılması, örn. iletişim devreleri veya alçak gerilim enkoder kabloları gibi dokunmaya karşı korumalı olduğu kabul edilen devrelerde tehlikeli gerilimlere neden olabilir.

Beklenmedik derecede yüksek gerilimler, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilecek bir elektrik çarpmasına neden olabilir.

Sadece, sınırlanmış gerilime sahip dokunmaya karşı korumalı devrelerin kaynakları olarak onaylanmış olan yüksek gerilim/alçak gerilim trafoları kullanın.

S7-1200'ün topraklanmasıyla ilgili direktifler

Uygulamanızı en iyi şekilde topraklamak için, S7-1200'ünüzün ve bağlı tüm cihazların tüm partner bağlantılarının ve tüm toprak bağlantılarının tek bir noktada topraklanmasına dikkat edin. Bu nokta, doğrudan sistem topraklamasına bağlanmalıdır. Blendajlı kablolar kullanırken, alçak ve yüksek frekanslı parazitleri azaltmak için kablo blendajlarını her zaman her iki uçta da toprağa bağlayın.

Topraklama hatları mümkün olduğunca kısa ve büyük bir tel kesitine sahip olmalıdır, örn. 2 mm^2 (14 AWG).

Topraklama noktalarını seçerken ilgili güvenlik talimatlarına dikkat edin ve elektrik devresini kesen koruma donanımlarının sorunsuz bir şekilde çalıştığından emin olun.

S71200'ün kablajıyla ilgili direktifler

S7-1200'ünüzün kablajını planlarken, aynı anda S7-1200 CPU'sunun gerilim beslemesini, tüm giriş devrelerinin gerilimini ve tüm çıkış devrelerinin gerilimini ayıran bir münferit ayırma şalteri kurun. Besleme kablajında kaçak akımı sınırlamak için örn. bir sigorta veya bir koruma şalteri aracılığıyla aşırı akım koruması sağlayın. Münferit çıkış devrelerine sigortalar veya başka akım sınırlama araçları aracılığıyla ek koruma uygulayabilirsiniz.

Yıldırım çarpma riski olan hatları uygun bir aşırı gerilim korumasıyla donatın. Daha fazla bilgiyi, teknik veriler kısmının "Genel" adlı alt bölümündeki Darbe gerilimlerine karşı parazite dayanıklılık (Sayfa 1160) başlığı altında bulabilirsiniz.

Alçak gerilim sinyal hatlarını ve iletişim kablolarını AC besleme hatları ve hızlı anahtarlamalı DC yüksek akım hatları ile aynı kablo tavasına yerleştirmeyin. Hatları çift olarak döşemeniz gerekir: Nötr iletkeni veya sıfır iletkenini faz iletkeniyle veya sinyal hattıyla birlikte.

Hatları mümkün olduğunca kısa döşeyin ve hat kesitinin ihtiyaç duyulan akım değerine uygun olmasına dikkat edin.

Hatlar ve kablolar, S7-1200'ün ortam sıcaklığından 30 °C daha yüksek bir sıcaklığa göre tasarlanmış olmalıdır (örnek: 55 °C'lik bir ortam sıcaklığında en az 85 °C'ye uygun şekilde tasarlanmış hatlar). Elektrik devrelerinin özel ölçüm değerleri ve kurulum ortamınız yardımıyla diğer kablaj türlerini ve malzeme gerekliliklerini belirleyin.

Parazitlere karşı optimum koruma için blendajlı hatlar kullanın. Blendajı her iki kablo sonunda da toprağa bağlayın İletişim kablolarının blendajını S7-1200 iletişim fişi muhafazalarıyla, kablo blendajına iletken olarak bağlı soket bağlayıcılar aracılığıyla veya iletişim kablosunun blendajını ayrı bir toprağa bağlayarak topraklamanız gerekir. Büyük bir yüzeyi topraklama noktasına bağlamak için diğer kablo blendajlarını klemensler veya blendajın çevresindeki bakır şerit aracılığıyla topraklayın.

Harici bir gerilim beslemesi üzerinden beslenen bir giriş devresinin kablajını yaparken giriş devresinde bir aşırı akım koruma cihazı kullanın. S7-1200'ün 24 V DC enkoder beslemesi aracılığıyla beslenen devrelerde, enkoder beslemesi zaten akımla sınırlı olduğu için harici bir koruma gerekmez.

Tüm S7-1200 modülleri, kullanıcı kablajı için geçmeli klemens bloklarına sahiptir. Bağlantıların gevşememesi için klemens bloğunun sıkı oturmasına ve hatların klemens bloğunda sabit şekilde kablolanmasına dikkat edin.

S7-1200, potansiyel ayırma sınırlarıyla çalışır, bu sayede tesisinizde istenmeyen akımların oluşması engellenir. Sisteminiz için kablaj planlaması yaparken bu potansiyel ayırma sınırlarını dikkate alın. Teknik veriler (Sayfa 1228) bölümünde, kullanıma sunulan potansiyel ayırmayı ve potansiyel ayırma sınırlarının düzenlenmesi için kullanılan değerleri bulabilirsiniz. AC şebeke gerilimi için tasarlanan elektrik devreleri diğer elektrik devreleriyle güvenlik temelli potansiyel ayırmayı kapsar. 24 V DC elektrik devrelerinin arasındaki potansiyel ayırma sınırları salt fonksiyoneldir ve güvenlik açısından bu sınırlara güvenmemeniz gerekir.

S7-1200 CPU'ları, SM'leri ve SB'leri için uygulanan kablaj kurallarının bir özetini aşağıda bulabilirsiniz:

Tablo 4-15 S7-1200 CPU'ları, SM'leri ve SB'leri için kablaj kuralları

Aşağıdakiler için kablaj kuralları:	**CPU ve SM bağlantısı**		**SB bağlantısı**
Bağlantı tekniği	**Push-in klemensi**	**Cıvata**	**Cıvata**
Standart teller için bağlanabilir iletken kesitleri	2 mm^2 ila 0,3 mm^2 (14 AWG ila 22 AWG)		1,3 mm^2 ila 0,3 mm^2 (16 AWG ila 22 AWG)
Bağlantı başına tel sayısı	1 veya 2 mm^2'ye kadar bir çift kaplamalı kabloda (toplam) 2 telden oluşan kombinasyon		1 veya 1,3 mm^2'ye kadar (toplam) 2 telden oluşan kombinasyon

Aşağıdakiler için kablaj kuralları:	CPU ve SM bağlantısı		SB bağlantısı
Bağlantı tekniği	**Push-in klemensi**	**Cıvata**	**Cıvata**
İzolasyon sıyırma uzunluğu	Güvenli bir elektrik bağlantısı için kablo yüksüklerinin kullanımı	6,4 mm	6,3 ila 7 mm
Sıkma torku (maksimum)	-/-	0,56 Nm	0,33 Nm
Alet	2,5 ila 3,0 mm düz tornavida		

Bağlantıya zarar vermemek için cıvataları aşırı döndürmemeye dikkat edin. Aksesuar olarak temin edilebilen Push-in klemens bloğunu da kullanabilirsiniz.

NOT

Örgülü hat üzerinde kablo yüksükleri kullanarak, çıkıntı yapan teller nedeniyle kısa devre oluşma tehlikesini azaltabilirsiniz. İletkenlerin yanal hareketleri nedeniyle kısa devre oluşmasını önlemek için, önerilen izolasyon sıyırma uzunluğundan daha uzun kablo yüksükleri bir izolasyon kovanına sahip olmalıdır. Çıplak iletkenlerin kesit alanıyla ilgili sınırlamalar kablo yüksükleri için de geçerlidir.

Ayrıca bakınız

Teknik veriler (Sayfa 1160)

Lamba yükleriyle ilgili direktifler

LED lambalar da dahil olmak üzere lamba yükleri, yüksek kısa süreli açma akımı nedeniyle röle kontaklarına zarar verir. Bu kısa süreli akım, bir tungsten lambanın sabit akımına oranla nominal olarak 10 ila 15 kat daha yüksektir. Uygulamanın kullanım ömrü içinde çok sık anahtarlanan lamba yükleri için değiştirilebilir bir kaplin röle veya bir kısa süreli akım sınırlayıcı kullanılması önerilir.

Endüktif yüklerle ilgili direktifler

Bir kumanda çıkışı kapatılırken gerilim artışını sınırlamak için endüktif yüklerde koruma devreleri kullanın. Koruma devreleri, akım akışı endüktif bir yük nedeniyle kesildiğinde ortaya çıkan yüksek gerilim geçici olayları nedeniyle çıkışları zamanından önce devre dışı kalmaya karşı korur.

Ayrıca koruma devreleri, endüktif yükler anahtarlanırken oluşan elektriksel parazitleri sınırlar. Yeterli koruma devresi yoksa, endüktif yüklerden kaynaklanan yüksek frekanslı parazitler PLC'nin çalışmasını durdurabilir. Elektriksel parazitleri azaltmanın en etkili yolu, yüke paralel ve fiziksel olarak yüke yakın bir yerde bir harici koruma devresi kullanmaktır.

S7-1200'ün DC çıkışları, çoğu uygulamadaki endüktif yükler için yeterli olan dahili koruma devrelerini kapsar. S7-1200'ün röle çıkış kontakları bir DC yükünü veya bir AC yükünü anahtarlamak için kullanılabildiğinden, dahili koruma mevcut değildir.

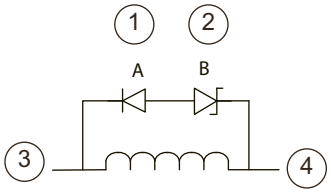
Üretici tarafından yük cihazına entegre edilen veya opsiyonel aksesuar olarak temin edilebilen koruma devrelerinin sunulduğu kontaktörleri ve diğer endüktif yükleri kullanmak iyi bir çözümdür. Ancak üretici tarafından teslim edilen bazı koruma devreleri uygulamanız için uygun olmayabilir. Kontaklarda optimum gürültü azaltma ve optimum kullanım ömrü elde etmek için ek bir koruma devresi gerekebilir.

AC yüklerinde, bir metal oksit varistör (MOV) veya paralel bir RC devre elemanına sahip başka bir gerilim sınırlayıcı kullanılabilir, ancak bu tertibatlar tek başına çok etkili değildir. Paralel RC devre elemanı olmayan bir MOV, genelde klemens gerilimi seviyesine kadar ciddi ölçüde yüksek frekanslı parazitlere yol açar.
İyi kontrol edilen bir kapatma geçici gerilimi, maksimum 10 kHz'lik bir halka frekansına sahiptir (1 kHz'den daha düşük bir değer tercih edilir). AC hatları için pik gerilimi, +/- 1200 V şasi değeri aralığında olmalıdır. Dahili PLC koruma devresine sahip DC yükler için negatif pik gerilimi, 24 V DC besleme geriliminin ~40 V altındadır. Harici bir koruma devresi, dahili koruma devresinin yükünü azaltmak için, geçici olayı beslemenin 36 V'u dahilinde sınırlamalıdır.

NOT

Bir koruma devresinin etkinliği, ilgili uygulamaya bağlıdır ve her zaman ilgili münferit durum için kontrol edilmelidir. Tüm bileşenlerin doğru şekilde boyutlandırılmasına dikkat edin ve kapatma geçici olaylarını bir osiloskop aracılığıyla gözlemleyin.

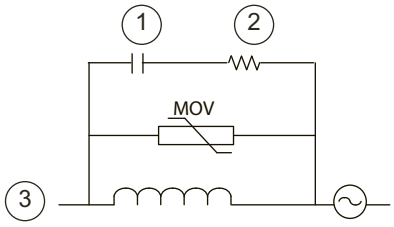
DC veya röle çıkışları için kullanılan ve endüktif DC yüklerini anahtarlayan tipik koruma devreleri



① 1N4001 diyot veya eşdeğeri
② Zener diyodu 8,2 V (DC çıkışları), Zener diyodu 36 V (röle çıkışları)
③ Çıkış
④ M, 24 V referans

Çoğu uygulamada, endüktif DC yükünün karşısında ek olarak bir diyot (A) kullanımı uygundur, ancak uygulamanız için daha hızlı kapatma süreleri gerekirse bir Zener diyodu (B) kullanılması önerilir. Zener diyodunu, çıkış devresindeki akım değerine göre boyutlandırın.

Endüktif AC yüklerini anahtarlayan röle çıkışları için tipik koruma devreleri



① C için kullanılan değeri tabloda bulabilirsiniz
② R için kullanılan değeri tabloda bulabilirsiniz
③ Çıkış

Metal oksit varistörün (MOV) çalışma geriliminin nominal gerilimden en az % 20 daha yüksek olduğuna dikkat edin.
Darbe uygulamaları için önerilen darbe boyutlandırmalı, endüktif olmayan dirençler ve kondensatörler (genellikle metal kaplamalı) kullanın. Bileşenlerin, ortalama güç, pik gücü ve pik gerilimi değerlerine yönelik gereklilikleri karşılamasına dikkat edin.

Kendi koruma devrenizi tasarlıyorsanız, çeşitli AC yükleri için direnç ve kondensatör değerlerini aşağıdaki tabloda bulabilirsiniz. Bu değerler, ideal bileşen parametreleriyle yapılan hesaplamaları temel alır. Tablodaki I efektif bilgisi, tamamen açık durumdayken yükün kararlı hâl akımıyla ilgilidir.

Tablo 4-16 AC koruma devreleri için kullanılan direnç ve kondensatör değerleri

Endüktif yük			**Koruma devresi değerleri**		
I efektif	230 V AC	120 V AC	Direnç		Kondensatör
A	VA	VA	Ω	W (nominal güç)	nF
0,02	4,6	2,4	15000	0,1	15
0,05	11,5	6	5600	0,25	47
0,1	23	12	2700	0,5	100
0,2	46	24	1500	1	150
0,5	115	60	560	2,5	470
1	230	120	270	5	1000
2	460	240	150	10	1500

[22] Maksimum kapatma adımı < 500 V

Diferansiyel girişler ve çıkışlarla ilgili direktifler

Diferansiyel girişler ve çıkışlar, standart girişler ve çıkışlara göre farklı davranır. Her diferansiyel giriş ve çıkış, iki bağlantı pimine sahiptir. Bir diferansiyel girişin veya çıkışın açık mı yoksa kapalı mı olduğunu belirlemek için, bu iki bağlantı pimi arasındaki gerilim farkının ölçülmesi gerekir.
Ek A'daki CPU 1217C (Sayfa 1228) ile ilgili ayrıntılı teknik verilere dikkat edin.

PLC ile ilgili temel esaslar

5

5.1 Kullanıcı programının yürütülmesi

CPU, kullanıcı programınızın uygun bir yapısı için aşağıdaki modül türlerini destekler:

- Organizasyon blokları (OB'ler) (Sayfa 172), programın yapısını belirler. Bazı OB'ler için ön tanımlı bir tutum ve yol verme olayları mevcuttur, ancak OB'leri kendi yol verme olaylarınızla da oluşturabilirsiniz.
- Fonksiyonlar (FC'ler) (Sayfa 174) ve fonksiyon blokları (FB'ler) (Sayfa 174) belirli görevlere karşılık gelen program kodunu içerir. Her FC veya FB, çağıran modülle verilerin ortak kullanımı için bir dizi giriş ve çıkış parametresi sağlar. Bir FB ayrıca, FB çağrısının ilgili örneğine ait veri değerlerini kaydetmek için başka bir veri bloğu (örnek DB) kullanır. Bir FB'yi her seferinde benzersiz bir örnek DB ile birkaç kez çağırabilirsiniz. Aynı FB'nin farklı örnek DB'lerle çağrılması, diğer örnek DB'lerden birindeki veri değerlerini etkilemez.
- Veri blokları (DB'ler) (Sayfa 176), program blokları tarafından kullanılabilen verileri kaydeder.

Kullanıcı programının yürütülmesi, periyodik olarak işlenen bir veya daha fazla çevrim OB'sinin izlediği, RUN işletim durumuna geçildikten sonra bir kez işlenen bir veya daha fazla opsiyonel yol alma OB ile başlar. Bir OB'yi, bir standart olayın veya bir hata olayının söz konusu olabileceği bir alarm olayına da atayabilirsiniz. Bu OB'ler, ilgili standart olay veya hata olayı ortaya çıktığında yürütülür.

Bir fonksiyon (FC) veya bir fonksiyon bloğu (FB), bir OB'den veya başka bir FC'den veya bir FB'den çağrılabilen program kodlu bir bloktur. Bu sırada aşağıdaki gömme derinlikleri mümkündür:

- Çevrim veya yol alma OB'lerinden 16 adet
- Herhangi bir alarm olayı OB'sinden 6
  Not: Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

FC'ler hiçbir belirli veri bloğuna (DB) atanmamıştır. FB'ler doğrudan bir DB'ye bağlanmıştır ve parametreleri aktarmak ve ara değerleri ve sonuçları kaydetmek için DB'yi kullanır.

Kullanıcı programının, verilerin ve konfigürasyonun boyutu, mevcut yükleme belleği ve CPU'daki çalışma belleği ile sınırlıdır. Münferit OB'ler, FC'ler, FB'ler ve DB'lerin sayısı sınırlı değildir.

Her çevrim; çıkışların yazılmasını, girişlerin okunmasını, kullanıcı programının yordamlarının düzenlenmesini ve arka planda işlemenin uygulanmasını kapsar. Çevrim, tarama çevrimi veya tarama olarak da tanımlanır.

S7-1200 otomasyon çözümünüz, S7-1200 CPU ile birlikte merkezi bir modül taşıyıcısı ve başka modüllerden oluşabilir. "Merkezi modül taşıyıcısı" terimi, CPU'nun ve ilgili modüllerin montaj rayı montajı veya pano montajını temel alır.

- Merkezi modül taşıyıcısında bir modülün takılması veya çıkartılması, cihaz açık durumdayken mümkün değildir. Merkezi modül taşıyıcısındaki bir modülü asla CPU açıkken takmayın ve çıkartmayın.

  ⚠ İKAZ

  Modüllerin takılması veya çıkartılmasına yönelik güvenlik gereklilikleri

  CPU açıkken merkezi modül taşıyıcısında bir modülün (SM, SB, BB, CD, CM veya CP) takılması veya çıkartılması, hasara veya öngörülemeyen davranışlara neden olabilir ve bu da ölüme, ciddi yaralanmalara ve/veya maddi hasara yol açabilir.

  Merkezi modül taşıyıcısında bir modülü takmadan veya çıkartmadan önce gerilim beslemesini CPU'dan ve merkezi modül taşıyıcısından ayırın ve ilgili güvenlik önlemlerini uygulayın.

- CPU açıkken bir SIMATIC memory card'ı takabilir veya çıkartabilirsiniz. Ancak CPU'nun RUN işletim durumunda bir memory card'ın takılması veya çıkartılması, CPU'nun STOP işletim durumuna geçmesine yol açar.

  DİKKAT

  CPU RUN işletim durumundayken memory card'ın çıkartılması durumunda ortaya çıkan riskler

  Bir memory card'ın, CPU'nun RUN işletim durumunda takılması veya çıkartılması ile CPU, STOP işletim durumuna geçer, bu da kumanda edilen cihazlarda veya kumanda edilen proseste maddi hasarlara yol açabilir.

  Bir memory card'ı takmadan veya çıkarmadan önce, CPU'nun bir makineyi veya prosesi aktif olarak kontrol etmediğinden emin olun. Uygulamanız veya prosesiniz için her zaman bir ACİL KAPAT anahtarlama devresi takın.

- Desantral bir G/Ç modül taşıyıcısında (AS-i, PROFINET veya PROFIBUS) CPU'nun RUN işletim durumundayken bir modülü takar veya çıkartırsanız, CPU, teşhis tamponunda bir giriş oluşturur, "Modüllerin çıkartılması veya takılması" OB'yi yürütür (varsa) ve standart olarak RUN işletim durumunda kalır.

Proses görünümünün ve kısmi proses görünümlerinin güncellenmesi

CPU, yerel dijital ve analog girişleri ve çıkışları, proses görünümü adı verilen dahili bir bellek alanı ile çevrimle senkronize olarak günceller. Proses görünümü, fiziksel giriş ve çıkışların (CPU, sinyal kartı ve sinyal modüllerinin fiziksel G/Ç'si) bir anlık kaydını içerir.

G/Ç'yi, her çevrimde proses görünümünde veya her zaman özel bir alarm olayı ortaya çıktığında güncellenecek şekilde ayarlayabilirsiniz. G/Ç'yi, proses görünümünün güncellenmesinden hariç tutulacak şekilde de ayarlayabilirsiniz. Örnek: Prosesiniz, belirli veri değerlerine muhtemelen her zaman proses alarmı gibi bir olay ortaya çıktığında ihtiyaç duyar. Bu G/Ç için proses görünümü güncellemesini bir kısmi proses görünümüne ve bunu da bir proses alarmı OB'ye atadığınızda, prosesiniz sürekli güncelleme gerektirmediği için, CPU bu veri değerlerini her çevrimde gereksiz yere güncellemez.

CPU, her çevrimde güncellenen G/Ç için her çevrimde aşağıdaki görevleri gerçekleştirir:

- CPU, çıkışların proses görünümündeki çıkışları fiziksel çıkışlara yazar.
- CPU, fiziksel girişleri, kullanıcı programının yürütülmesinden hemen önce okur ve giriş değerlerini, girişlerin proses görünümüne kaydeder. Bununla birlikte, kullanıcı yordamları yürütülürken bu değerler tutarlı kalır.
- CPU, kullanıcı yordamlarının lojiğini gerçekleştirir ve gerçek fiziksel çıkışlara yazmak yerine çıkışların proses görünümündeki çıkış değerlerini günceller.

  Bu işlem, ilgili çevrimdeki kullanıcı yordamlarının tüm yürütme süreci sırasında tutarlı bir lojik sağlar ve fiziksel çıkışların, çıkışların proses görünümündeki birden fazla durum değişimi ile salınım yapmasını önler.

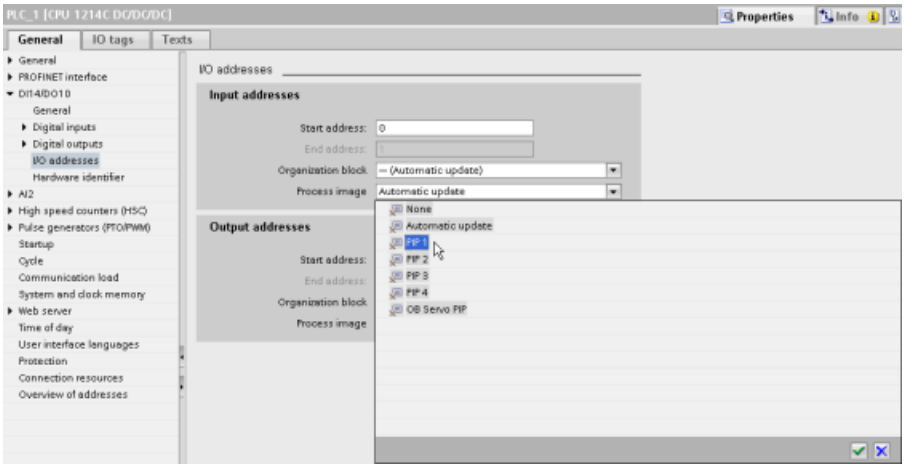
S7-1200, prosesinizin G/Ç'yi her çevrimde otomatik olarak mı yoksa olaylar tetiklendiğinde mi güncelleyeceğini kontrol etmek için beş kısmi proses görünümü sağlar. İlk kısmi proses görünümü olan TPA0, her çevrimde otomatik olarak güncellenen G/Ç için belirlenmiştir. Burada standart atama söz konusudur. Geri kalan dört kısmi proses görünümü olan TPA1, TPA2, TPA3 ve TPA4'e, çeşitli alarm olaylarında proses görünümünün güncellenmesini atayabilirsiniz. Alarm OB'leri oluşturur (Sayfa 172) veya OB özelliklerini düzenlerseniz (Sayfa 172), G/Ç'yi cihaz konfigürasyonundaki kısmi proses görünümlerine ve kısmi proses görünümlerini de da alarm olaylarına atayabilirsiniz.

STEP 7, standart olarak, cihaz görünümüne bir modül eklenirken proses görünümünün güncellenmesi için "Otomatik güncelleme" seçeneğini ayarlar. "Otomatik güncelleme" seçeneğinin kullanıldığı G/Ç'de, CPU, modül ile proses görünümü arasındaki veri alışverişini her çevrimde otomatik olarak düzenler.

Bir kısmi proses görünümüne dijital veya analog G/Ç atamak veya G/Ç'yi proses görünümünün güncellenmesi dışında bırakmak için aşağıdaki yöntemi izleyin:

1. İlgili cihaz için cihaz konfigürasyonundaki "Özellikler" sekmesini açın.
2. İstediğiniz G/Ç'yi görüntülemek için gerekirse "Genel" öğesinin altındaki seçimi genişletin.
3. "G/Ç adresleri" öğesini seçin.
4. Opsiyonel olarak, "Organizasyon bloğu" açılır listesinden belirli bir OB'yi seçin.

5. "Proses görünümü" açılır listesi aracılığıyla "TPA1", "TPA2", "TPA3", "TPA4"te "Otomatik güncelleme" ayarını veya "Yok" öğesini seçin. "Yok" öğesinin seçilmesi, bu G/Ç'yi sadece doğrudan yordamlar aracılığıyla okuyabileceğiniz ve yazabileceğiniz anlamına gelir. Giriş ve çıkışları tekrar proses görünümünün otomatik güncellemesine dahil etmek için, bu seçenek için tekrar "Otomatik güncelleme" ayarını yapın.



Bir yordam yürütülürken fiziksel girişlerin değerlerini doğrudan okuyabilir ve ayrıca fiziksel çıkışlara doğrudan değerler de yazabilirsiniz. Doğrudan okuma sırasında, fiziksel girişlerin güncel durumuna erişilir. Girişlerin proses görünümü, kaydetme işlemi için kullanılan girişin proses görünümünde yapılandırılmış olmasından bağımsız olarak, güncellenmez. Fiziksel çıkışa doğrudan yazma işlemi sırasında, hem çıkışların proses görünümü (çıkış, proses görünümüne kaydetme işlemi için yapılandırıldıysa) hem de fiziksel çıkış güncellenir. Programın G/Ç verilerine proses görünümü yerine doğrudan fiziksel giriş veya çıkış üzerinden erişmesini istiyorsanız, G/Ç adresine ":P" son ekini ekleyin.

NOT

Kısmi proses görünümlerinin kullanılması

TPA1'den TPA4'e kadar olan kısmi proses görünümlerinden birine bazı G/Ç'leri atar, ancak kısmi proses görünümüne hiçbir OB atamazsanız, CPU, bu G/Ç'leri proses görünümünde asla güncellemez. Uygun bir OB ataması olmadan bir TPA'ya G/Ç atamak, proses görünümünü "Yok" güncellemesi için ayarlamaya karşılık gelir. Direkt bir okuma yordamıyla G/Ç'yi doğrudan fiziksel G/Ç'den okuyabilir veya direkt bir yazma yordamıyla doğrudan fiziksel G/Ç'ye yazabilirsiniz. CPU, proses görünümünü güncellemez.

CPU; PROFINET, PROFIBUS ve AS-i ağlarında (Sayfa 553) desantral çevrebirimi destekler.

5.1.1 CPU'nun işletim durumları

CPU, üç işletim durumuna sahiptir: STOP işletim durumu, STARTUP işletim durumu ve RUN işletim durumu. CPU'nun ön tarafındaki durum LED'leri, güncel işletim durumunu gösterir.

- CPU, STOP işletim durumunda programı yürütmez. CPU'ya bir proje yükleyebilirsiniz.
- STARTUP işletim durumunda yol alma OB'leri (mevcutsa) bir kez işlenir. CPU, bu işletim durumunda hiçbir proses alarmı olayını işlemez.
- RUN işletim durumunda, program çevrimi OB'leri tekrar yürütülür. RUN işletim durumunda her an, ilgili alarm olayı OB'lerinin yürütülmesine neden olan alarm olayları ortaya çıkabilir. Bir projenin parçalarını RUN işletim durumunda yükleyebilirsiniz (Sayfa 1139).

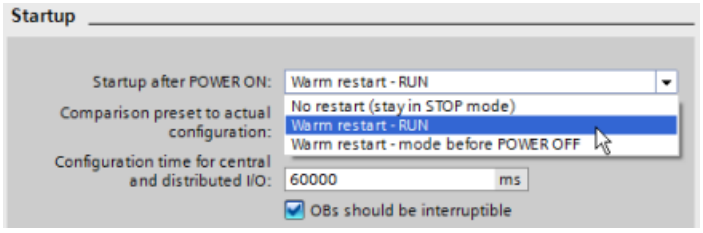
CPU, RUN işletim durumuna geçmek için sıcak başlatmayı destekler. Bir sıcak başlatma işlemi sırasında tam silme işlemi gerçekleştirilmez. CPU, bir sıcak başlatma işleminde kalıcı olmayan tüm sistem ve kullanıcı verilerini başlangıç ayarlarına getirir ve tüm kullanıcı verilerinin değerlerini ara belleğe alır.

Tam silme işleminde, çalışma belleği ile kalıcı olan ve kalıcı olmayan tüm bellek alanları silinir, yükleme belleği çalışma belleğine kopyalanır ve çıkışlar, yapılandırılmış "CPU STOP'a Tepki" olarak ayarlanır. Teşhis tamponu ve IP adresinin kalıcı olarak kaydedilen değerleri tam silme işleminde silinmez.

CPU'nun "GÜÇ AÇIK'tan sonra yol alma" ayarını yapılandırabilirsiniz. Bu ayarları, CPU'nun cihaz konfigürasyonundaki "Yol alma" kısmında bulabilirsiniz. CPU, açma işlemi sırasında bir dizi teşhis kontrolü ve ardından sistemi başlatma işlemini gerçekleştirir. CPU, sistemi başlatma işlemi sırasında markörün kalıcı olmayan tüm bellek alanını siler ve kalıcı olmayan tüm DB içeriklerini yükleme belleğinden başlangıç değerlerine döndürür. CPU, markörün kalıcı bellek alanını ve kalıcı DB içeriklerini tampon belleğe alır ve ardından ilgili işletim durumuna geçer. Belirli hatalar, CPU'nun RUN işletim durumuna geçmesini engeller. CPU, aşağıdaki konfigürasyon seçeneklerini destekler:

- Yeniden başlatma yok (STOP'ta kalınır)
- Sıcak başlatma - RUN

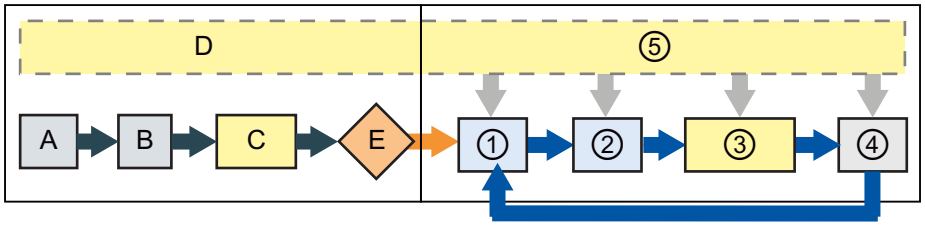
- Sıcak başlatma - ŞEBEKE KAPALI'dan önceki işletim türü



DİKKAT
Giderilebilir hatalar, CPU'nun STOP işletim durumuna geçmesine neden olabilir. CPU, giderilebilir hatalar nedeniyle örn. şu durumlarda STOP işletim durumuna geçebilir: – Değiştirilebilir bir sinyal modülünün devre dışı kalması – Şebeke hattındaki arızalar veya öngörülemeyen bir yol alma olayı gibi geçici hatalar. Bu tür durumlar maddi hasarlara neden olabilir. CPU için "Sıcak başlatma - ŞEBEKE KAPALI'dan önceki işletim türü" ayarı yapılandırıldıysa, CPU, şebeke kesintisinden veya hatadan önceki işletim durumuna döner. CPU, şebeke kesintisi veya hata anında STOP işletim durumunda idiyse, yol alma sırasında STOP işletim durumuna döner. CPU, RUN işletim durumuna geçmek için bir komut alana kadar STOP işletim durumunda kalır. CPU, şebeke kesintisi veya hata anında RUN işletim durumunda idiyse, yol alma sırasında RUN işletim durumuna döner. CPU, RUN işletim durumuna dönmeyi engelleyen bir hata algılamazsa RUN işletim durumuna döner. Bir STEP 7 bağlantısından bağımsız olarak çalışması gereken CPU'lar, "Sıcak başlatma - RUN" modu için yapılandırılmalıdır. Böylece CPU, bir sonraki yol alma işlemi sırasında RUN işletim durumuna dönebilir.

Güncel işletim durumunu, programlama yazılımının çevrimiçi araçlarında "STOP" ve "RUN" komutları (Sayfa 1126) ile değiştirebilirsiniz. Ayrıca, CPU'yu STOP işletim durumuna ayarlamak için programınıza bir STP yordamı (Sayfa 290) ekleyebilirsiniz. Bu şekilde, programınızın yürütülmesini program lojiğine bağlı olarak durdurabilirsiniz.

- CPU, STOP işletim durumunda iletişim gerekliliklerini düzenler (ihtiyaca bağlı olarak) ve kendi kendine teşhis işlemleri gerçekleştirir. CPU, kullanıcı programını yürütmez. Proses görünümü otomatik olarak güncellenmez.
- CPU, STARTUP ve RUN işletim durumlarında aşağıdaki ekranda görüntülenen görevleri gerçekleştirir.



STARTUP		RUN	
A	Fiziksel girişlerin durumu, E bellek alanına kopyalanır	①	A bellek alanı, fiziksel çıkışlara yazılır
B	A bellek alanı (Resim) konfigürasyona bağlı olarak sıfır değeriyle, son değerle veya yapılandırılan yedek değerle başlatılır. PB, PN ve AS-i çıkışları sıfıra ayarlanır.	②	Fiziksel girişlerin durumu, E bellek alanına kopyalanır
C	Markörün kalıcı olmayan bellek alanları ve veri blokları, başlangıç değerleriyle başlatılır ve yapılandırılan uyandırma alarmı ve saat olayları etkinleştirilir. Yol alma OB'leri yürütülür.	③	Program çevrimi OB'leri yürütülür
D	Alarm olayları, RUN işletim durumunda işlenmek için bekleme kuyruğuna kaydedilir	④	Kendi kendine teşhis işlemini gerçekleştirir
E	A bellek alanından fiziksel çıkışlara yazma işlemi etkinleştirilir	⑤	Alarmlar ve iletişim, çevrimin tüm bölümlerinde düzenlenir

NOT

HMI iletişimi de dahil olmak üzere iletişimi sadece program çevrimi OB'leri durdurabilir.

Yol almayı işleme

CPU, STOP işletim durumundan RUN işletim durumuna her geçildiğinde, girişlerin proses görünümünü temizler, çıkışların proses görünümünü başlatır ve yol alma OB'leri işler. Yol alma OB'lerindeki yordamların, girişlerin proses görünümüne yapılan tüm okuma erişimleri, fiziksel girişin güncel değeri yerine sıfır değeriyle sonuçlanır. Bu nedenle, STARTUP işletim durumunda bir fiziksel girişin güncel durumunu okumak için girişi doğrudan okumanız gerekir. Ardından CPU, yol alma OB'leri ve ilgili tüm FC'leri ve FB'leri yürütür. Çok sayıda yol alma OB mevcutsa, CPU bunları en düşük numaradan başlayarak OB numarasına göre sırayla işler.

Her yol alma OB'si, kalıcı verilerin ve gerçek zaman saatinin geçerliliğini belirleyebilmeniz için yol alma gerçek zaman saati bilgilerini içerir. Bu yol alma değerlerini incelemek ve uygun önlemleri almak için yol alma OB'lerinde yordamlar programlayabilirsiniz. Aşağıdaki yol alma adresleri, yol alma OB'leri tarafından desteklenir:

Tablo 5-1 Yol alma OB tarafından desteklenen yol alma adresleri

Giriş	Veri tipi	Açıklama
LostRetentive	Bool	Bu Bit, kalıcı verilerin bellek alanları kaybolduysa doğrudur.
LostRTC	Bool	Bu Bit, gerçek zaman saati kaybolduysa doğrudur.

CPU, yol almayı işleme sırasında aşağıdaki görevleri de gerçekleştirir.

- Alarmlar, yol alma aşamasında bir bekleme kuyruğuna alınır ve düzenlenmez
- Yol alma aşamasında çevrim süresi denetlemesi yapılmaz
- Yol alma sırasında, hızlı sayaçların (HSC), darbe katarının (PTO) ve noktadan noktaya iletişim modüllerinin (PtP) konfigürasyonu değiştirilebilir
- HSC, PTO ve PtP modüllerinin gerçek fonksiyon akışları sadece RUN işletim durumunda gerçekleşir

CPU, yol alma OB'leri yürütüldükten sonra, RUN işletim durumuna geçer ve kumanda görevlerini sürekli bir çevrim içinde düzenler.

5.1.2 RUN işletim durumunda çevrimin işlenmesi

CPU, her çevrimde çıkışlara yazar, girişleri okur, kullanıcı programını yürütür, iletişim modüllerini günceller ve kullanıcı alarm iletişim olaylarına ve iletişim taleplerine yanıt verir. İletişim talepleri, çevrim sırasında düzenli olarak işlenir.

Bu eylemler (kullanıcı alarm olayları hariç) sürekli ve periyodik olarak işlenir. Etkinleştirilen kullanıcı alarm olayları, önceliğe göre ortaya çıkma sıralarına uygun olarak işlenir. CPU, alarm olaylarında girişleri okur, OB'yi yürütür ve gerekirse ilgili kısmi proses görünümü (PIP) ile birlikte çıkışlara yazar.

Sistem, çevrimin maksimum çevrim süresi içinde işlenmesini sağlar, aksi takdirde bir zaman hatası oluşur.

- Her çevrim, proses görünümündeki dijital ve analog çıkışların güncel değerlerinin sorgusuyla ve bu değerlerin otomatik G/Ç güncellemesi için yapılandırılmış CPU, SB ve SM'lerin fiziksel çıkışlarına yazılmasıyla başlar (standart konfigürasyon). Bir yordam fiziksel bir çıkışa erişirse, proses görünümündeki çıkış ve fiziksel çıkış güncellenir.
- Çevrim devam ederken, dijital ve analog girişlerin güncel değerleri CPU'dan, SB'den ve otomatik G/Ç güncellemesi için yapılandırılmış olan SM'lerden (standart konfigürasyon) okunur ve bu değerler proses görünümüne yazılır. Bir yordam fiziksel bir girişe erişirse, fiziksel girişin değeri değiştirilir, ancak proses görünümündeki giriş güncellenmez.
- Girişler okunduktan sonra, kullanıcı programı ilk yordamdan son yordama kadar yürütülür. Tüm program çevrimi OB'leri ve ilgili tüm FC'ler ve FB'ler buna dahildir. Program çevrimi OB'leri, en düşük OB numarasından başlayarak OB numarasına göre sırayla işlenir.

İletişim işleme, çevrim sırasında düzenli olarak gerçekleşir ve muhtemelen kullanıcı programının yürütülmesini durdurur.

Kendi kendine teşhis işlemlerine, sistemin düzenli kontrolleri ve G/Ç modüllerinin durumunun sorgulanması dahildir.

Alarmlar, çevrimin her bölümünde ortaya çıkabilir, olay kontrollüdürler. Bir olay ortaya çıkarsa, CPU çevrimi durdurur ve olayın işlenmesi için OB'yi çağırır. OB olayı işlediyse, CPU, kullanıcı programının yürütülmesine daha önce kesintiye uğradığı noktadan devam eder.

5.1.3 Organizasyon blokları (OB'ler)

OB'ler, kullanıcı programının yürütülmesine kumanda eder. Bir organizasyon bloğunun yürütülmesi, CPU'daki belirli olaylar tarafından başlatılır. OB'ler birbirlerini çağıramaz. OB'ler bir FC'den veya bir FB'den de çağrılamaz. Sadece bir teşhis alarmı veya zaman gecikmesi gibi bir olay CPU'nun bir OB'yi yürütmesine neden olur. CPU, OB'leri ilgili öncelik sınıflarına göre ele alır ve daha yüksek öncelikli OB'ler daha düşük öncelikli OB'lerden önce yürütülür.

Ayrıca bakınız

Olayların yürütülmesiyle ilgili öncelikler ve bekleme kuyruğu (Sayfa 80)

5.1.3.1 Program çevrimi OB'si

CPU RUN işletim durumunda ise, program çevrimi OB'ler periyodik olarak yürütülür. Programın ana bloğu, program çevrimi OB'sidir. Programınızın kumandası için gerekli yordamları içerir ve diğer kullanıcı modülleri buradan çağrılır. CPU'nun numara sırasıyla yürüttüğü birden fazla Program çevrimi OB'si oluşturabilirsiniz. Main (OB 1), standart bloktur.

Program çevrimi olayları

Program çevrimi olayı, her program çevriminde bir kez ortaya çıkar. CPU, program çevrimi sırasında çıkışlara yazar, girişleri okur ve program çevrimi OB'leri yürütür. Program çevrimi olayı gereklidir ve her zaman etkinleştirilmiş durumdadır. Hiçbir program çevrimi OB'niz olmayabilir veya birden çok program çevrimi OB'niz olabilir. CPU, program çevrimi olayı ortaya çıktıktan sonra, en düşük numaralı program çevrimi OB'sini (normalde OB1) yürütür. Diğer program çevrimi OB'ler CPU tarafından program çevrimi içinde sırayla (numara sırasıyla) yürütülür. Program, program çevrimi olayı aşağıdaki zamanlarda gerçekleşecek şekilde periyodik olarak yürütülür:

- Son yol alma OB işlenmişse
- Son Program çevrimi OB'si işlenmişse

Tablo 5-2 Bir program çevrimi OB'sinin yol alma bilgileri

Giriş	Veri tipi	Açıklama
Initial_Call	Bool	OB'nin ilk çağrısında doğrudur
Remanence	Bool	Kalıcı veriler mevcutsa doğrudur

5.1.3.2 Yol alma OB

Yol alma OB'ler, CPU STOP işletim durumundan RUN işletim durumuna geçtiğinde, yol alma sırasında RUN işletim durumuna geçişte ve STOP işletim durumundan RUN işletim durumuna önceden belirtilen bir geçişte bir kez yürütülür. Ardından, program çevriminin yürütülmesi işlemi başlar. Bkz. "Yol almayı işleme", CPU'nun işletim durumları (Sayfa 63)

5.1.3.3 Gecikme alarmı OB'si

Gecikme alarmı OB'leri, yapılandırdığınız bir zaman gecikmesinden sonra yürütülür.

Gecikme alarmı olayları

Gecikme alarmı olaylarını, belirtilen bir gecikme süresinden sonra ortaya çıkacak şekilde yapılandırabilirsiniz. Gecikme süresini SRT_DINT yordamıyla atayabilirsiniz. Gecikme olayları, ilgili gecikme alarmı OB'sinin yürütülmesi için program çevrimini durdurur. Bir gecikme olayına sadece bir gecikme alarmı OB'si atanabilir. CPU dört gecikme olayını destekler.

Tablo 5-3 Bir gecikme alarmı OB'sinin başlangıç bilgileri

Giriş	Veri tipi	Açıklama
Sign	Word	SRT_DINT'den tetikleyici çağrıya aktarılan kod

5.1.3.4 Uyandırma alarmı OB'si

Uyandırma alarmı OB'leri belirli aralıklarla yürütülür. Her olaya bir OB karşılık gelecek şekilde dört adede kadar uyandırma alarmı olayı yapılandırabilirsiniz.

Uyandırma alarmı olayları

Uyandırma alarmı olayları, bir alarm OB'sinin yapılandırılmış bir çevrim süresinde yürütülmesini ayarlamanıza olanak tanır. Başlangıç çevrim süresi, uyandırma alarmı OB'si oluşturulurken yapılandırılır. Bir uyandırma alarmı olayı, program çevrimini durdurur ve ilgili uyandırma alarmı OB'yi yürütür. Bir uyandırma alarmı olayının, program çevrimi olayından daha yüksek bir öncelik sınıfına sahip olduğuna dikkat edilmelidir.

Bir uyandırma alarmı olayına sadece bir uyandırma alarmı OB'si atanabilir.

Her uyandırma alarmına, uyandırma alarmlarının yürütülmesi, faz kaymasının değeri kadar birbirlerine doğru kaydırılabilecek şekilde bir faz kayması atayabilirsiniz. Örneğin bir uyandırma alarmı olayı 5 msn için ve diğeri 10 msn için öngörüldüyse, her iki olay da 10 msn'de bir eşzamanlı olarak ortaya çıkar. 5 msn'lik olayın fazı 1 ila 4 msn kaydırılırsa ve 10 msn'lik olayın fazı 0 msn kaydırılırsa, iki olay aynı anda ortaya çıkmaz.

Faz kayması, standart olarak 0'a ayarlanmıştır. Bir periyodik olayın başlangıçtaki faz kaymasını veya çevrim süresini değiştirmek için aşağıdaki yöntemi izleyin:

1. Proje gezgininde farenin sağ tuşuyla uyandırma alarmı OB'ye tıklayın.
2. Bağlam menüsünde "Özellikler" komutunu seçin.
3. "Uyandırma alarmı [OB 30]" iletişim kutusundaki "Uyandırma alarmı" öğesine tıklayın ve yeni başlangıç değerlerini girin.

Maksimum faz kayması 6000 msn (6 saniye) veya maksimum çevrim süresidir; hangisi daha kısa ise.

Ayrıca programınızdan çevrim süresini ve faz kaymasını sorgulayabilir ve değiştirebilirsiniz. Bunun için, Uyandırma alarmı parametrelerini sorgula (QRY_CINT) ve Uyandırma alarmı parametrelerini ayarla (SET_CINT) yordamlarını kullanın. Çevrim süresi ve faz kayması için SET_CINT yordamı aracılığıyla girilen değerler, kapatma işlemi veya STOP işletim durumuna geçiş sırasında kaydedilmez; tekrar çalıştırma sırasında veya RUN işletim durumuna dönüldüğünde başlangıçtaki değerler tekrar kullanılır. CPU, toplam dört uyandırma alarmı olayını destekler.

5.1.3.5 Proses alarmı OB'si

Proses alarmı OB'leri, ilgili proses olayı ortaya çıktığında yürütülür. Bir proses alarmı OB'si, bir proses olayının bir sinyaline yanıt olarak normal program akışını durdurur.

Proses alarmı olayları

Proses alarm olayları, örneğin bir girişteki yükselen veya alçalan kenar veya bir HSC'deki (hızlı sayaç) bir olay gibi donanımdaki bir değişiklik tarafından tetiklenir. S7-1200, her proses alarmı olayı için bir alarm OB'sini destekler. Cihaz konfigürasyonundaki proses olaylarını etkinleştirebilir ve cihaz konfigürasyonundaki bir olaya veya kullanıcı programındaki bir ATTACH yordamı ile bir OB atayabilirsiniz. CPU, birçok proses alarmı olayını destekler. Hangi olayların kullanılabildiği, CPU modeline veya girişlerin sayısına bağlıdır.

Proses alarmı olayları için aşağıdaki limitler geçerlidir:

Kenarlar:

- Yükselen kenar olayları: maksimum 16
- Alçalan kenar olayları: maksimum 16

HSC olayları:

- CV=PV: maksimum 6
- Yön değişimi: maksimum 6
- Harici sıfırlama: maksimum 6

Tablo 5-4 Bir uyandırma alarmı OB'si için kullanılan yol alma bilgileri

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Uyandırma alarmını tetikleyen modülün donanım kodu.
USI	WORD	Kullanıcı yapısı kodu (16#0001 - 16#FFFF), daha sonra kullanmak için rezerve edilmiştir
IChannel	USINT	Alarmı tetikleyen kanalın numarası
EventType	BYTE	Uyandırma alarmını tetikleyen olayın modüle özel olay tipinin kodu, örneğin düşen veya yükselen bir kenar

EventType'daki Bit'ler tetikleyici modüle bağlıdır. Bu, aşağıda gösterilmiştir:

Modül/alt modül	Değer	Proses olayı
CPU veya SB'nin entegre G/Ç'si	16#0	Yükselen kenar
	16#1	Alçalan kenar
HSC	16#0	HSC CV=RV1
	16#1	HSC yön değişimi
	16#2	HSC'nin sıfırlanması
	16#3	HSC CV=RV2

5.1.3.6 Zaman hatası OB'si

Yapılandırıldıysa, maksimum çevrim süresi aşıldığında veya bir zaman hatası olayı ortaya çıktığında zaman hatası OB'si (OB 80) yürütülür. Bu OB tetiklenirse, normal program akışını veya başka bir olay OB'sini de durduran yürütme işlemi başlar.

Bu olaylardan birinin ortaya çıkması, teşhis tamponunda olayın açıklamasını içeren bir giriş oluşturur. Teşhis tamponundaki giriş, zaman hatası OB'nin mevcut olup olmadığından bağımsız olarak oluşturulur.

Zaman hatası alarm olayları

Herhangi bir zaman hatasının ortaya çıkması, bir zaman hatası olayına neden olur:

- Maksimum çevrim süresinin aşılması
  Program çevrimi belirtilen maksimum çevrim süresi içinde tamamlanmazsa "Maksimum çevrim süresi aşıldı" hatası oluşur. Bu konuyla ilgili daha fazla bilgi için, bkz. Çevrim süresinin denetlenmesi ve yapılandırılması (Sayfa 84).
- CPU talep edilen OB'yi başlatamıyor, çünkü CPU birinci alarm OB'sini yürütmeyi bitirmeden önce ikinci bir alarm (uyandırma veya gecikme alarmı) başlatıldı
- Bekleme kuyruğu taşması
  Alarmlar CPU tarafından işlenebileceklerinden daha hızlı ortaya çıkarsa "Bekleme kuyruğu taşması" hatası oluşur. Bekleyen olayların sayısı, her olay tipine kendine ait bir bekleme kuyruğu atanarak CPU tarafından sınırlandırılır. İlgili bekleme kuyruğu dolduğunda bir olay meydana gelirse, CPU tarafından bir zaman hatası olayı oluşturulur.

Tüm zaman hatası olayları, zaman hatası OB'nin (varsa) yürütülmesini tetikler. Bir zaman hatası OB'si mevcut değilse, CPU STOP işletim durumuna geçer.

Çevrim süresi denetlemesini yeniden başlatmak için RE_TRIGR yordamı (Sayfa 289) çalıştırılarak, kullanıcı programı, program çevriminin yürütülmesini, yapılandırılmış maksimum çevrim süresinin on katına kadar uzatabilir. Ancak "Maksimum çevrim süresi aşıldı" hatası, çevrim süresi sıfırlanmadan aynı program çevriminde iki kez meydana gelirse, CPU, zaman hatası OB'nin mevcut olup olmadığından bağımsız olarak STOP işletim durumuna geçer. Bkz. Çevrim süresinin denetlenmesi ve yapılandırılması (Sayfa 84).

Zaman hatası OB'si, zaman hatasını hangi olayın ve hangi OB'nin oluşturduğunu belirlemek için kullanabileceğiniz yol alma bilgilerini içerir. Bu yol alma değerlerini incelemek ve uygun önlemleri almak için OB'de yordamlar programlayabilirsiniz.

Tablo 5-5 Zaman hatası OB'si (OB 80) için kullanılan yol alma bilgileri

Giriş	Veri tipi	Açıklama
Fault_ID	BYTE	16#01 - Maksimum çevrim süresi aşıldı 16#02 - İstenen OB başlatılamıyor 16#07 ve 16#09 - Bekleme kuyruğu taşması
Csg_OBnr	OB_ANY	Hata oluştuğunda yürütülen OB'nin numarası
Csg_Prio	UINT	Hataya neden olan OB'nin önceliği

Organizasyon blokları (OB'ler) (Sayfa 68) altında, OB'lerin projenize eklenmesiyle ilgili talimatları bulabilirsiniz.

Yeni bir CPU V4.0'ın önceliği 22'dir. Bir CPU V3.0'ı bir CPU V4.0 (Sayfa 1365) ile değiştirirseniz, öncelik 26'dır, yani V3.0 için geçerli olan önceliktir. Her iki durumda da öncelik alanı değiştirilebilir ve öncelik için 22 ila 26 aralığında herhangi bir değer belirleyebilirsiniz.

5.1.3.7 Teşhis hatası OB'si

Teşhis hatası OB'si, CPU bir teşhis hatası tespit ettiğinde veya teşhis edilebilir bir modül bir hata tespit ettiğinde ve modül için teşhis hatası alarmı etkinleştirildiğinde yürütülür. Bir teşhis hatası OB'si, normal program akışını durdurur. CPU'nun STOP işletim durumuna geçmesi gerekirse, bu hata tipi tespit edildiğinde, teşhis hatası OB'ye bir STP yordamı ekleyebilirsiniz.

Programınıza hiçbir teşhis hatası OB'si eklemezseniz, CPU hatayı göz ardı eder ve RUN işletim durumunda kalır.

Teşhis hatası olayları

PROFINET, PROFIBUS, yerel analog ve bazı yerel dijital cihazlar, teşhis hatalarını algılayabilir ve bildirebilir. Çeşitli teşhis hatalarından birinin ortaya çıkması veya kaybolması bir teşhis hatası olayına neden olur. Aşağıdaki teşhis hataları desteklenir:

- Kullanıcı gerilimi yok
- Üst limit aşıldı
- Alt limit aşıldı
- Tel kopması
- Kısa devre

Teşhis hatası olayları, mevcutsa, teşhis hatası OB'nin (OB 82) yürütülmesini tetikler. Mevcut değilse, CPU hatayı göz ardı eder.

Organizasyon blokları (OB'ler) (Sayfa 68) altında, bir OB'nin projenize eklenmesiyle ilgili talimatları bulabilirsiniz.

NOT

Çok kanallı yerel analog cihazlarda (G/Ç, RTD ve termo eleman) teşhis hatası

Teşhis hatası OB'si, her seferinde sadece bir kanalın teşhis hatalarını işleyebilir.

Çok kanallı bir cihazın iki kanalında hata oluşursa, ikinci hata sadece aşağıdaki koşullarda teşhis hatası OB'sini tetikler: İlk kanalın hatası giderildi, ilk hata tarafından tetiklenen teşhis hatası OB'nin yürütülmesi işlemi tamamlandı ve ikinci hata hâlâ mevcut.

Teşhis hatası OB'si, olayın bir hatanın oluşması veya ortadan kalkması nedeniyle tetiklenip tetiklenmediğini ve hatayı hangi cihaz ve kanalın bildirdiğini belirlemek için kullanabileceğiniz yol alma bilgilerini içerir. Bu yol alma değerlerini incelemek ve ilgili önlemleri almak için teşhis hatası OB'sinde yordamları programlayabilirsiniz.

NOT

Teşhis hatası OB'nin yol alma bilgileri, hiçbir bir teşhis olayı mevcut değilse, bütün olarak alt modülle ilgilidir

Yol alma bilgileri, olayın kaynağında belirli bir kanalın söz konusu olması durumunda, giden olay nedeniyle alt modülde artık hiçbir teşhis mevcut olmadığı sürece, bütün olarak alt modülle (16#8000) ilgilidir.

Tablo 5-6 Teşhis hatası OB'si için kullanılan yol alma bilgileri

Giriş	Veri tipi	Açıklama
IOstate	WORD	Cihazın G/Ç durumu: • Konfigürasyon doğruysa Bit 0 = 1 ve konfigürasyon artık doğru değilse Bit 0 = 0. • Bir hata mevcutsa Bit 4 = 1, (örnek: tel kopması). (Hiçbir hata yoksa Bit 4 = 0.) • Konfigürasyon doğru **değilse** Bit 5 = 1 ve konfigürasyon tekrar doğru ise Bit 5 = 0. • Bir G/Ç erişim hatası oluştuysa Bit 7 = 1. Erişim hatasıyla birlikte G/Ç donanım kodunu LADDR bölümünde bulabilirsiniz. (Hiçbir hata yoksa Bit 6 = 0.)
LADDR	HW_ANY	Hatayı bildiren cihazın veya fonksiyonel birimin donanım kodu[1]
Channel	UINT	Kanal numarası
MultiError	BOOL	Birden fazla hata mevcutsa DOĞRUDUR

[1] LADDR girişi, hatayı bildiren cihazın veya fonksiyonel birimin donanım kodunu içerir. Donanım kodu, bileşenler cihaz veya ağ görünümüne eklendiğinde otomatik olarak atanır. PLC değişkenlerinin "Sistem sabitleri" sekmesinde görüntülenir. Donanım koduna ayrıca otomatik olarak bir ad atanır. PLC değişkenlerinin "Sistem sabitleri" sekmesindeki bu girişler değiştirilemez.

5.1.3.8 "Modüllerin çıkartılması veya takılması" OB'si

"Modüllerin çıkartılması veya takılması" OB'si, yapılandırılmış ve devre dışı bırakılmamış desantral bir G/Ç modülünün veya alt modülün (PROFIBUS, PROFINET, AS-i), bir modülün çıkartılması veya takılmasıyla bağlantılı bir olay oluşturması durumunda uygulanır.

"Modüllerin çıkartılması veya takılması" olayı

Aşağıdaki koşullarda, bir "Modüllerin çıkartılması veya takılması" olayı meydana gelir:

- Bir kullanıcı, yapılandırılmış bir modülü çıkartır veya takar
- Yapılandırılmış bir modül fiziksel olarak genişletme modül taşıyıcısında mevcut değil
- Genişletme modül taşıyıcısında, yapılandırılmış bir modüle karşılık gelmeyen uyumsuz bir modül bulunuyor
- Bir genişletme modül taşıyıcısında, yapılandırılmış modülle uyumlu bir modül bulunuyor, ancak konfigürasyon yedek cihazlara izin vermiyor
- Bir modül veya alt modül parametrelendirme hatası içeriyor

Bu OB'nin programlanmamış olması halinde, yapılandırılmış ve devre dışı bırakılmamış desantral bir G/Ç modülü için bu koşullardan biri gerçekleşirse CPU RUN işletim durumunda kalır.

Bu OB'nin programlanıp programlanmamasından bağımsız olarak, merkezi modül taşıyıcısındaki bir modül için bu koşullardan biri gerçekleşirse CPU STOP işletim durumuna geçer.

Tablo 5-7 "Modüllerin çıkartılması veya takılması" OB ile ilgili yol alma bilgileri

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım kodu
Event_Class	Byte	16#38: Modül takıldı 16#29: Modül çekildi
Fault_ID	Byte	Hata ID'si

5.1.3.9 Modül taşıyıcısı veya istasyon hatası OB'si ("Rack or station failure OB")

Modül taşıyıcısı veya istasyon hatası OB'si ("Rack or station failure OB"), CPU bir desantral modül taşıyıcısında/bir desantral istasyonda bir hata veya iletişim arızası algıladığında yürütülür.

Modül taşıyıcısı veya istasyon hatası olayı

CPU, aşağıdaki durumlardan biri algılandığı takdirde bir modül taşıyıcısı veya bir istasyon hatası olayı oluşturur:

- Bir DP-Master sisteminin veya bir PROFINET IO sisteminin devre dışı kalması (gelen veya giden bir olayda)
- Bir DP-Slave'in veya bir IO cihazının devre dışı kalması (gelen veya giden bir olayda)
- Bir PROFINET I cihazına ait bir alt modülün devre dışı kalması

Bu OB programlanmadıysa, bu koşullardan biri gerçekleştiğinde CPU RUN işletim durumunda kalır.

Tablo 5-8 Modül taşıyıcısı veya istasyon hatası OB'si ile ilgili yol alma bilgileri

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım kodu
Event_Class	Byte	16#38: giden olay 16#39: gelen olay
Fault_ID	Byte	Hata tanımlama

5.1.3.10 Saat OB'si

Saat OB'leri, yapılandırılmış saat koşulları aracılığıyla yürütülür. CPU iki saat OB'sini destekler.

Saat olayları

Bir saat alarmı olayını, belirli bir saatte veya periyodik olarak ortaya çıkacak şekilde yapılandırabilirsiniz; bu sırada aşağıdaki çevrimlerden birini ayarlayabilirsiniz:

- Dakikalık Alarm, dakikada bir kez ortaya çıkar.
- Saatlik: Alarm, saatte bir kez ortaya çıkar.
- Günlük: Alarm, her gün belirli bir zamanda (saat ve dakika) ortaya çıkar.
- Haftalık: Alarm; her hafta, haftanın belirli bir gününde, belirli bir zamanda ortaya çıkar (örneğin her Salı günü saat 16:30'da).
- Aylık: Alarm; her ay, ayın belirli bir günde, belirli bir zamanda ortaya çıkar. Gün için 1 ile 28 arasında bir sayının belirlenmesi gerekir.
- Ay sonunda: Alarm, her ayın sonu günü belirli bir zamanda ortaya çıkar.
- Yıllık: Alarm, yılda bir kez belirli bir tarihte (ay ve gün) ortaya çıkar. Tarih olarak 29 Şubat girilemez.

Tablo 5-9 Saat OB ile ilgili yol alma bilgileri

Giriş	Veri tipi	Açıklama
CaughtUp	Bool	Zaman önceye ayarlandığı için OB çağrısı durduruldu
SecondTime	Bool	Zaman ertelendiği için OB çağrısı yeniden başlatıldı

5.1.3.11 Durum OB'si

Durum OB'si, bir DPV1 Slave veya PNIO Slave bir durum alarmını tetiklediğinde yürütülür. Bu durum, bir DPV1 Slave'in veya PNIO Slave'in bir bileşeni (modül veya modül taşıyıcısı) işletim durumunu örneğin RUN iken STOP olarak değiştirdiğinde görülebilir.

Durum olayları

Bir durum alarmını tetikleyen olaylarla ilgili ayrıntılı bilgileri, DPV1 Slave'in veya PNIO Slave'in üretici dokümantasyonunda bulabilirsiniz.

Tablo 5-10 Bir durum OB ile ilgili yol alma bilgileri

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım kodu
Slot	UInt	Yuva numarası
Specifier	Word	Alarm belirteci

5.1.3.12 Güncelleme OB'si

Güncelleme OB'leri, bir DPV1 Slave veya PNIO Slave bir güncelleme alarmını tetiklediğinde yürütülür.

Güncelleme olayları

Bir güncelleme alarmını tetikleyen olaylarla ilgili ayrıntılı bilgileri, DPV1 Slave'in veya PNIO Slave'in üretici dokümantasyonunda bulabilirsiniz.

Tablo 5-11 Bir güncelleme OB ile ilgili yol alma bilgileri

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım kodu
Slot	UInt	Yuva numarası
Specifier	Word	Alarm belirteci

5.1.3.13 Profil OB'si

Profil OB'leri, bir DPV1 Slave veya PNIO Slave profile özel bir alarmı tetiklediğinde yürütülür.

Profil olayları

Bir profil alarmını tetikleyen olaylarla ilgili ayrıntılı bilgileri, DPV1 Slave'in veya PNIO Slave'in üretici dokümantasyonunda bulabilirsiniz.

Tablo 5-12 Bir profil OB ile ilgili yol alma bilgileri

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım kodu
Slot	UInt	Yuva numarası
Specifier	Word	Alarm belirteci

5.1.3.14 OB MC Servo ve MC-Interpolator

Hareket kumandası için bir teknoloji nesnesi oluşturur ve tahrik arayüzü için "Analog tahrik bağlantısı" veya "PROFIdrive"ı ayarlarsanız, STEP 7, otomatik olarak MC-Servo ve MC-Interpolator yazma korumalı OB'lerini oluşturur. Hiçbir OB özelliğini değiştirmeniz ve bu OB'yi de doğrudan oluşturmanız gerekmez. CPU bu OB'leri ayarlama için kullanır. Daha fazla bilgiyi, STEP 7 bilgi sisteminde bulabilirsiniz.

5.1.3.15 MC-PreServo

MC-PreServo OB'sini, STEP 7 programı için kullanılan program lojiğini içerecek şekilde programlayabilirsiniz. Daha sonra bu program lojiği, MC-Servo OB'nin işlenmesinden hemen önce yürütülür.

MC-PreServo'nun olayları

MC-PreServo OB'si, uygulama çevrimiyle ilgili yapılandırılmış bilgileri mikro saniyeler içinde okumanıza olanak sağlar.

Tablo 5-13 OB MC-PreServo ile ilgili başlangıç bilgileri

Giriş	Veri tipi	Açıklama
Initial_Call	BOOL	DOĞRU, bu OB'nin ilk çağrısının, STOP işletim durumundan RUN işletim durumuna geçerken gerçekleşeceğini belirtir
PIP_Input	BOOL	DOĞRU, girişlerin proses görünümünün en güncel durumda olduğunu belirtir.
PIP_Output	BOOL	DOĞRU, CPU'nun, ilgili çıkışların proses görünümünü son çevrimden sonra uygun bir zamanda çıkışlara aktardığını gösterir.
IO_System	USINT	Alarmı tetikleyen desantral çevrebirim sisteminin numarası
Event_Count	INT	n: Kayıp çevrimlerin sayısı -1: Kayıp çevrimlerin sayısı bilinmeyen sayısı (örn. çevrim değiştiği için)
Synchronous	BOOL	Rezerve
CycleTime	UDINT	Milisaniye cinsinden MC-Servo OB için yapılandırılan uygulama çevrimi göstergesi

5.1.3.16 MC-PostServo

MC-PreServo OB'sini, STEP 7 programı için kullanılan program lojiğini içerecek şekilde programlayabilirsiniz. Daha sonra bu program lojiği, MC-Servo OB'nin işlenmesinden hemen sonra yürütülür.

MC-PostServo olayları

MC-PreServo OB'si, uygulama çevrimiyle ilgili yapılandırılmış bilgileri mikro saniyeler içinde okumanıza olanak sağlar.

Tablo 5-14 OB MC-PostServo ile ilgili başlangıç bilgileri

Giriş	Veri tipi	Açıklama
Initial_Call	BOOL	DOĞRU, bu OB'nin ilk çağrısının, STOP işletim durumundan RUN işletim durumuna geçerken gerçekleşeceğini belirtir
PIP_Input	BOOL	DOĞRU, girişlerin proses görünümünün en güncel durumda olduğunu belirtir.
PIP_Output	BOOL	DOĞRU, CPU'nun, ilgili çıkışların proses görünümünü son çevrimden sonra uygun bir zamanda çıkışlara aktardığını gösterir.
IO_System	USINT	Alarmı tetikleyen desantral çevrebirim sisteminin numarası
Event_Count	INT	n: Kayıp çevrimlerin sayısı -1: Kayıp çevrimlerin sayısı bilinmeyen sayısı (örn. çevrim değiştiği için)
Synchronous	BOOL	Rezerve
CycleTime	UDINT	Milisaniye cinsinden MC-Servo OB için yapılandırılan uygulama çevrimi göstergesi

5.1.3.17 Olayların yürütülmesiyle ilgili öncelikler ve bekleme kuyruğu

Olaylar CPU işlemeye kumanda eder. Bir olay, bir alarm OB'sinin yürütülmesini tetikler. Blok oluşturulurken, cihaz konfigürasyonu sırasında veya bir ATTACH veya DETACH yordamı aracılığıyla bir olay için alarm OB'sini belirtebilirsiniz. Program çevrimi olayı gibi bazı olaylar veya periyodik olaylar düzenli olarak ortaya çıkar. Yol alma olayı veya zaman gecikmesi olayları gibi diğer olaylar bir kereliğine ortaya çıkar. Bazı olaylar, donanım bir olayı tetiklediğinde ortaya çıkar, örn. bir girişteki bir kenar olayı veya hızlı bir sayacın bir onayı. Teşhis hatası olayı veya zaman hatası olayı gibi olaylar sadece hata durumunda ortaya çıkar. Olay öncelikleri ve bekleme kuyrukları, alarm OB'lerinin işlenme sırasını belirlemek için kullanılır.

CPU, olayları önceliklerinin sırasına göre işler, bu sırada 1 en düşük öncelik ve 26 en yüksek önceliktir. Bir OB'ye, OB özelliklerinin özniteliklerinde bir öncelik sınıfı atayabilirsiniz.

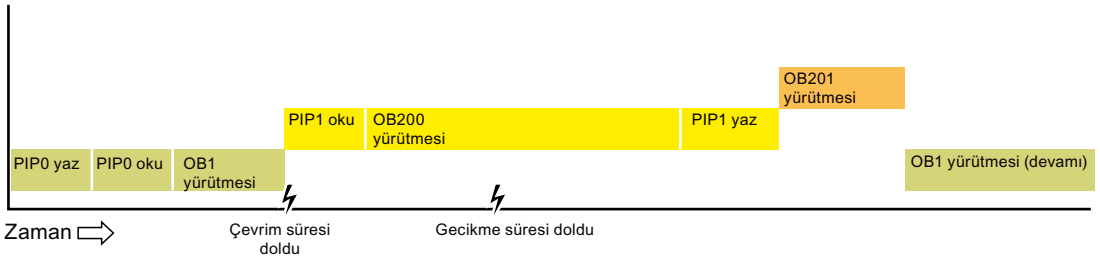
Durdurulabilir ve durdurulamaz yürütme türleri

OB'ler (Sayfa 68), OB'leri tetikleyen olayların öncelik sırasıyla yürütülür. CPU cihaz konfigürasyonunun yol alma özelliklerinde (Sayfa 144-145), OB yürütme işleminin durdurulabilir mi yoksa durdurulamaz mı olması gerektiğini yapılandırabilirsiniz. Program çevrimi OB'lerinin her zaman durdurulamaz özellikte olduğuna dikkat edin. Fakat diğer tüm OB'leri durdurulabilir veya durdurulamaz olarak yapılandırabilirsiniz.

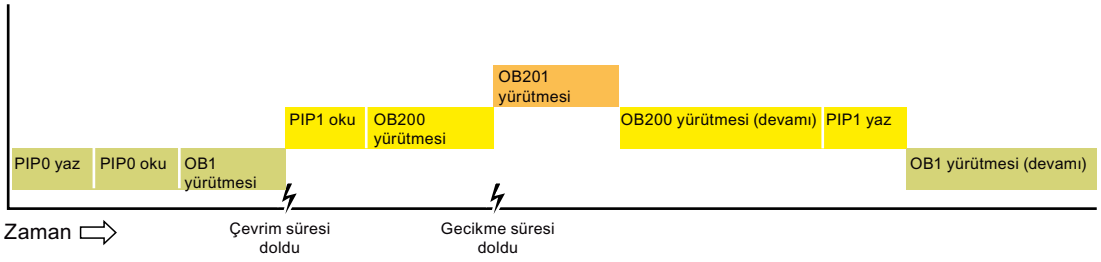
Yürütme türünü durdurulabilir olarak ayarlarsanız, OB'nin yürütülmesi sona ermeden önce daha yüksek öncelikli bir olay ortaya çıkarsa, daha yüksek öncelikli OB'nin yürütülmesine izin vermek için ilgili OB'nin yürütülmesi işlemi durdurulur. Bunun üzerine daha yüksek öncelikli olay yürütülür ve ardından, durdurulan OB devam ettirilir. Durdurulabilir bir OB'nin yürütülmesi sırasında birden fazla olay ortaya çıkarsa, CPU, bu olayları öncelik sırasına göre işler.

Yürütme türünü durdurulabilir olarak ayarlamazsanız, bir OB'nin yürütülmesi işlemi, bu zaman dilimi zarfında tetiklenen diğer olaylardan bağımsız olarak tamamlanır.

Alarm olaylarının bir çevrim OB'sini ve bir zaman gecikmesi OB'sini tetiklediği aşağıdaki iki duruma dikkat edin. Her iki durumda da zaman gecikmesi OB (OB 201) hiçbir kısmi proses görünümüne atanmamıştır (Sayfa 60) ve öncelik değeri 4 ile yürütülür. Çevrim OB'si (OB 200), TPA1 kısmi proses görünümüne atanmıştır ve öncelik değeri 2 ile yürütülür. Aşağıdaki şekillerde, yürütmedeki durdurulabilir yürütme türü ile durdurulamaz yürütme türü arasındaki farklar gösterilmektedir:



Resim 5-1 Durum 1: Durdurulamaz OB yürütme işlemi



Resim 5-2 Durum 2: Durdurulabilir OB yürütme işlemi

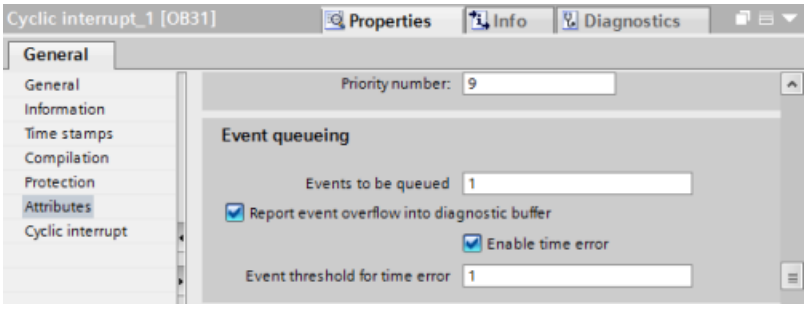
NOT

OB yürütme türünü durdurulamaz olarak ayarlarsanız, bir zaman hatası OB'si sadece program çevrimi OB'lerini durdurabilir. S7-1200 CPU'ların V4.0 sürümünden önce, bir zaman hatası OB'si yürütülen her OB'yi durdurabiliyordu. Bir OB'nin bir zaman hatası OB'si veya daha yüksek öncelikli başka bir OB tarafından durdurulmaması için, OB yürütme türünü durdurabilir olarak yapılandırmanız gerekir.

Olayların yürütülmesiyle ilgili öncelikler ve bekleme kuyruğu hakkında bilinmesi gerekenler

CPU, her olay tipine kendine ait bir bekleme kuyruğu atanarak, tek bir kaynaktaki bekleyen olayların sayısını sınırlar. Belirli bir tipteki bekleyen olayların maksimum sayısına ulaşıldığında bir sonraki olay artık işlenmez ve kaybolur. Bir zaman hatası OB'si (Sayfa 71) ile bekleme kuyruğunda oluşan bir taşmaya müdahale edebilirsiniz.

STEP 7'de uyandırma alarmı OB'si ve saat OB'si için, olay bekleme kuyruğu adına bazı özel parametreleri yapılandırabileceğinize dikkat edin.



CPU'nun aşırı yük davranışı ve olay bekleme kuyrukları ile ilgili daha fazla bilgiyi STEP 7 bilgi sisteminde bulabilirsiniz.

Bir CPU'nun her olayı bir önceliğe sahiptir. Olaylar, genelde CPU tarafından öncelik sırasına göre (en yüksek öncelikli ilk) işlenir. Aynı öncelik derecesine sahip olaylar, CPU tarafından First-In-First-Out (İlk Giren İlk Çıkar) prensibine göre işlenir.

Tablo 5-15 OB olayları

Olay	İzin verilen sayı	OB standart önceliği
Program çevrimi	1 program çevrimi olayı Birden fazla OB'ye izin verilir	1[1]
Yol alma	1 Yol alma olayı [1] Birden fazla OB'ye izin verilir	1[1]
Gecikme	Maksimum 4 zaman olayı Olay başına 1 OB	OB 20: 3 OB 21: 4 OB 22: 5 OB 23: 6 OB 123 ila OB 32767: 3
Uyandırma alarmı	Maksimum 4 olay Olay başına 1 OB	OB 30: 8 OB 31: 9 OB 32: 10 OB 33: 11 OB 34: 12 OB 35: 13 OB 36: 14 OB 37: 16 OB 38: 17 OB 123 ila OB 32767: 7
Proses alarmı	Maksimum 50 proses alarmı olayı[2] Olay başına 1 OB, ancak birden fazla olay için aynı OB'yi kullanabilirsiniz	18 18
Zaman hatası	1 olay (sadece yapılandırıldığında)[3]	22 veya 26[4]
Teşhis hatası	1 olay (sadece yapılandırıldığında)	5
Modüllerin çıkartılması veya takılması	1 olay	6
Modül taşıyıcısında veya istasyonda hata	1 olay	6
Saat alarmı	Maksimum 2 olay	2
Durum	1 olay	4
Güncelleme	1 olay	4
Profil	1 olay	4
MC-Servo	1 olay	25
MC-Interpolator	1 olay	24

[1] Program çevrimi başlatılmadan önce yol almanın tamamlanmış olması gerektiği için yol alma ve program çevrimi olayı asla aynı anda ortaya çıkmaz.

[2] DETACH ve ATTACH yordamları kullanıldığında 50'den fazla proses alarmı OB'sinin kullanılması mümkündür.

[3] Maksimum çevrim süresi aşıldığında, CPU'yu, RUN işletim durumunda kalacak şekilde yapılandırabilir veya RE_TRIGR yordamıyla çevrim süresini sıfırlayabilirsiniz. Ancak maksimum çevrim süresi bir çevrimde ikinci kez aşılırsa, CPU, STOP işletim durumuna geçer.

[4] Öncelik 22'dir, ancak 22 ile 26 aralığında herhangi bir değerle değiştirilebilir.

Bu konuyla ilgili daha fazla bilgi için, bkz. "Bir CPU V3.0'ın CPU V4.x ile değişimi (Sayfa 1365)".
Ayrıca CPU, ilişkili OB'lere sahip olmayan başka olayları algılar. Aşağıdaki tabloda, bu olaylar ve CPU'nun ilgili eylemleri gösterilmektedir:

Tablo 5-16 Ek olaylar

Olay	Açıklama	CPU eylemi
Çevrebirim erişim hatası	Giriş/çıkış doğrudan okunurken/yazılırken oluşan hata	CPU, ilk ortaya çıkma durumunu teşhis tamponuna yazar ve RUN işletim durumunda kalır. Hata sebebini GET_ERROR_ID (Sayfa 290) yordamıyla sorgulayabilirsiniz.
Maksimum çevrim süresiyle ilgili hata	CPU, yapılandırılan çevrim süresini ikinci kez aşar	CPU, hatayı teşhis tamponuna kaydeder ve STOP işletim durumuna geçer.
Çevrebirim erişim hatası	Proses görünümü güncellenirken oluşan G/Ç hatası	CPU, ilk ortaya çıkma durumunu teşhis tamponuna yazar ve RUN işletim durumunda kalır.
Programlama hatası	Program yürütülürken oluşan hata	• Modüle özel hata işleme etkinleştirildiyse, sistem, hata yapısına bir hata sebebi girer. Hata sebebini GET_ERROR_ID (Sayfa 290) yordamıyla sorgulayabilirsiniz. • Global hata işleme etkinleştirildiyse, sistem, teşhis tamponuna bir erişim hatası yol verme olayı girer ve RUN işletim durumunda kalır.

Gecikme süresi

Olay gecikme süresi (yani, CPU'nun bir olayın ortaya çıktığını bildirmesi ile olay işleme için kullanılan OB'deki ilk yordamın yürütülmeye başlanması arasındaki süre), alarm olayı anında işleme rutini olarak sadece bir Program çevrimi OB'si aktif ise yaklaşık 175 µsn'dir.

5.1.4 Çevrim süresinin denetlenmesi ve yapılandırılması

Çevrim süresi, CPU işletim sisteminin, RUN işletim durumunun periyodik fazını yürütmek için ihtiyaç duyduğu süredir. CPU, çevrim süresini denetlemek için iki yöntem sunar:

- Maksimum çevrim süresi
- Minimum çevrim süresi

Çevrim denetimi, tamamlanan yol alma olayından sonra başlar. Bu fonksiyon, CPU'da "Cihaz konfigürasyonu > Çevrim" öğelerinin altında projelendirilebilir.
CPU, çevrimi denetler ve çevrim süresi, yapılandırılan maksimum çevrim süresini aşarsa tepki verir. CPU, çevrim süresi, yapılandırılan maksimum çevrim süresini aşarsa bir hata üretir ve aşağıdaki şekilde tepki verir:

- Kullanıcı programı bir zaman hatası alarmı OB'si (Sayfa 71) içeriyorsa, CPU bunu yürütür.
- Kullanıcı programı bir zaman hatası alarmı OB'si içermiyorsa, zaman hatası olayı bir teşhis tamponu girişi üretir. Ardından CPU STOP işletim durumuna geçer.

RE_TRIGR (Sayfa 289) yordamı (çevrim süresi denetlemesini yeniden başlat), çevrim süresinin ölçümü için zamanlayıcının sıfırlanmasına olanak sağlar. Güncel program çevriminin yürütülmesi için geçen süre yapılandırılmış maksimum çevrim süresinin on katından daha azsa, RE_TRIGR yordamı, çevrim süresi denetlemesini tekrar tetikler ve ENO = DOĞRU değerini

döndürür. Aksi taktirde, RE_TRIGR yordamı çevrim süresi denetlemesini tekrar tetiklemez. ENO = YANLIŞ ifadesini döndürür.

Çevrim, tipik olarak mümkün olduğunca hızlı bir şekilde uygulanır ve önceki çevrim tamamlanınca bir sonraki çevrim başlar. Bununla birlikte çevrim süresi, kullanıcı programına ve iletişim görevlerine bağlı olarak farklılık gösterebilir. CPU, bu tür dalgalanmaları önlemek için opsiyonel bir minimum çevrim süresini destekler. Bu opsiyonel fonksiyonu etkinleştirir ve msn cinsinden bir minimum çevrim süresi belirtirseniz, CPU, program çevrimi OB'leri yürütüldükten sonra program çevriminin tekrarlanmasını minimum çevrim süresi dolana kadar geciktirir.

CPU bir çevrimi minimum çevrim süresi ile belirtilenden daha hızlı yürütürse, CPU, kalan süreyi çalışma zamanı teşhisleri için ve/veya iletişim taleplerinin işlenmesi için kullanır.

Çevrim, belirtilen minimum çevrim süresi içinde tamamlanmazsa, sonuna kadar normal şekilde yürütülür (iletişim işleme dahil) ve minimum sürenin aşılması, bir sistem tepkisi oluşturmaz. Aşağıdaki tabloda, çevrim süresi denetlemesiyle ilgili aralıklar ve ön ayarlar gösterilmektedir.

Tablo 5-17 Çevrim süresi aralığı

Çevrim süresi	Aralık (msn)	Ön ayar
Maksimum çevrim süresi[1]	1 ila 6000	150 msn
Minimum çevrim süresi[2]	1 ila maksimum çevrim süresi	Devre dışı bırakıldı

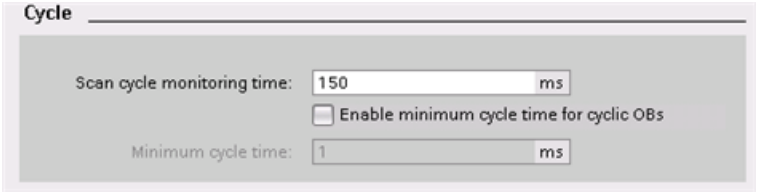
[1] Maksimum çevrim süresi her zaman etkinleştirilmiş durumdadır. 1 msn ile 6000 msn arasında bir çevrim süresi ayarlayın. Ön ayar 150 msn'dir.

[2] Minimum çevrim süresi opsiyoneldir ve standart olarak devre dışı bırakılmış durumdadır. Gerekirse 1 msn ile maksimum çevrim süresi arasında bir çevrim süresi ayarlayın.

Çevrim süresi ve iletişim yükünün yapılandırılması

Cihaz konfigürasyonunun CPU özelliklerinde şu parametreleri ayarlayabilirsiniz:

- Çevrim: Burada, bir maksimum çevrim denetleme süresi değeri girebilirsiniz. Bir minimum çevrim süresi de girebilir veya etkinleştirebilirsiniz.



- İletişim yükü: İletişim görevleri için zamanın bir yüzdesel oranını belirleyebilirsiniz.



NOT

İletişim önceliği

İletişim görevlerinin öncelik değeri 1'dir. 1 en düşük öncelik olduğu için diğer CPU olayları, iletişimin işlenmesini durdurabilir. Başka olaylarda oluşan kesintiler, çevrim sırasında iletişimin işlenmesini olumsuz etkileyebilir. İletişimin işlenmesi için kullanıma sunulan çevrim oranını artırmak için "İletişim nedeniyle oluşan çevrim yükü" için ayrılan yüzde değerini uyarlayabilirsiniz.

Çevrimlerle ilgili daha fazla bilgi için, bkz. "Çevrim süresinin denetlenmesi ve yapılandırılması (Sayfa 84)" bölümü.

5.1.5 CPU belleği

Bellek yönetimi

CPU; kullanıcı programı, veriler ve konfigürasyon için şu bellek alanlarını sağlar:

- Yükleme belleği; kullanıcı programı, veriler ve konfigürasyon için kullanılan geçici olmayan bir bellektir. CPU, bir proje CPU'ya yüklenirken programı önce yükleme belleğine kaydeder. Bu bellek, bir SIMATIC memory card'da (mevcutsa) ya da CPU'da bulunur. CPU bu geçici olmayan bellek alanını gerilim kesintisi durumunda da kaydeder. Memory card, CPU'ya entegre edilenden daha büyük bir bellek alanını destekler.
- Çalışma belleği, kullanıcı programı işlenirken kullanıcı projesinin bazı elemanları için kullanılan bir geçici bellektir. CPU, projenin bazı elemanlarını yükleme belleğinden çalışma belleğine kopyalar. Bu geçici bellek alanı, gerilim kesintisi durumunda kaybolur ve elektrik geldiğinde CPU tarafından geri yüklenir.
- Kalıcı bellek, çalışma belleği değerlerinin sınırlanmış bir miktarı için kullanılan bir geçici olmayan bellektir. CPU, gerilim kesintisi durumunda kullanıcı belleğinin seçilen adreslerinin değerlerini kaydetmek için kalıcı bellek alanını kullanır. Bir kapatma veya gerilim kesintisi durumu ortaya çıktığında, CPU bu kalıcı değerleri yol alma sırasında geri yükler.

Derlenmiş bir program bloğunun bellek kullanımını görüntülemek için, STEP 7 proje gezgininin "Program blokları" klasöründeki bloğa çift tıklayın ve bağlam menüsündeki "Özellikler" seçeneğini etkinleştirin. Derleme özelliklerinde, derlenmiş blok için kullanılan yükleme belleği ve çalışma belleği görüntülenir.
Çevrimiçi CPU'nun bellek kullanımını görüntülemek için STEP 7'de "Çevrimiçi ve teşhis"e tıklayın, "Teşhis" öğesini genişletin ve "Bellek" öğesini seçin.

Kalıcı bellek

Belirli verileri kalıcı olarak tanımlayarak gerilim kesintisinden sonra veri kaybını önleyebilirsiniz. Aşağıdaki verileri CPU'da kalıcı olarak tanımlayabilirsiniz:

- Markör (M): Markörler için kullanılan kalıcı belleğin boyutunu PLC değişken tablosunda veya atama listesinde tanımlayabilirsiniz. Kalıcı markör belleği, her zaman MB0'dan başlar ve belirtilen Bayt sayısı boyunca kesintisiz olarak devam eder. Fonksiyon çubuğunda "Kalıcı" sembolüne tıklayarak bu değeri PLC değişken tablosunda veya atama listesinde belirtin. MB0'dan itibaren kalıcı olması gereken markör Bayt'ı sayısını belirtin.
  Not: Atama listesi, her blok için program blokları klasöründe bir blok ve **Araçlar > Atama planı** menü komutu seçilerek görüntülenebilir.
- Bir fonksiyon bloğunun (FB) değişkenleri: Bir FB "Optimize edilmiş blok erişimi" tipinde ise, arayüz editörü bu FB için "Kalıcı" adında bir sütun içerir. Bu sütunda, her değişken için münferit olarak "Kalıcı", "Kalıcı değil" veya "IDB'de ayarla" öğelerinden birini seçebilirsiniz. Programa bu tür bir FB yerleştirirseniz, FB'ye karşılık gelen örnek DB aynı zamanda " Kalıcı" sütununu da içerir. Örnek DB'nin arayüz editöründe bir değişkenin kalıcı durumunu ancak optimize edilmiş FB'deki değişken için kullanılan "Kalıcı" alanında "IDB'de ayarla" (örnek veri bloğunda ayarla) öğesini seçtiyseniz değiştirebilirsiniz.
  Bir FB "Optimize edilmiş blok erişimi" tipinde **değilse**, arayüz editörü bu FB için "Kalıcı" adında bir sütun içermez. Ancak programa böyle bir FB yerleştirirseniz, FB'ye karşılık gelen örnek DB, düzenlenebilen bir "Kalıcı" sütunu içerir. Bu durumda, bir değişken için "Kalıcı" öğesinin seçilmesi, **tüm** değişkenlerin seçilmesine neden olur. Benzer şekilde, bir değişken için "Kalıcı" seçeneğinin devre dışı bırakılması, **tüm** değişkenlerin devre dışı bırakılmasına neden olur.
  Bir FB'nin optimize edilmiş blok erişimine sahip olup olmadığını göstermek için FB'nin özelliklerini açın ve öznitelikleri seçin.
- Global bir veri bloğunun değişkenleri: Veri bloğunun özelliklerindeki öznitelikler için "Optimize edilmiş blok erişimini" etkinleştirirseniz, her değişken için bu değişkenin kalıcı olup olmayacağını ayarlayabilirsiniz. "Optimize edilmiş blok erişimini" etkinleştirmezseniz, tüm veri bloğu değişkenleri aynı duruma sahip olur. Değişkenlerin tümü kalıcıdır ya da tümü geçicidir.

CPU, V4.5'ten daha yüksek bir Firmware sürümüne sahip bir PLC'ye yüklenen ve sürümü V4.5 veya daha yeni olan bir proje için toplam 14336 Bayt kalıcı veriyi destekler. V4.x projelerinde kalıcı veriler 10240 Bayt'tır. Kullanılabilir durumdaki bellek miktarını belirlemek için PLC değişken tablosundaki veya atama listesindeki fonksiyon çubuğundaki "Kalıcı" sembolüne tıklayın. Burada, markör belleği için kullanılacak kalıcı alanı belirtebilirsiniz; ikinci satır, M ve DB için birlikte kalan toplam belleği gösterir. Bu değerin doğru olması için, kalıcı değişkenlere sahip tüm veri bloklarının derlenmesi gerektiğine dikkat edin.

NOT

Bir programın yüklenmesi nedeniyle kalıcı bellekteki değerler silinmez ve değiştirilmez. Yükleme işleminden önce kalıcı belleğin silinmesi gerekirse, program yüklenmeden önce CPU sıfırlanarak fabrika ayarlarına döndürülmelidir.

5.1.5.1 Sistem ve zamanlama markörü

CPU özelliklerinde "Sistem belleği" ve "Zamanlama markörü" için Bayt'ları etkinleştirin. Bu fonksiyonların münferit Bit'leri, değişken adları aracılığıyla program lojiğinizde referanslanabilir.

- Markör alanında bir Bayt'ı sistem belleği olarak atayabilirsiniz. Sistem markörünün Bayt'ı, kullanıcı programınız tarafından aşağıdaki değişken adları üzerinden referanslanabilen şu dört Bit'i sunar:
  - İlk çevrim: Bit ("FirstScan" değişken adı), ilk çevrim için yol alma OB sona erdikten sonra 1'e ayarlanır. (İlk çevrim sona erdikten sonra "İlk çevrim" Bit'i 0'a ayarlanır.)
  - Teşhis durumu değiştirildi: (Değişken adı: "DiagStatusUpdate") CPU tampona bir teşhis olayını kaydettikten sonra bir çevrim için 1'e ayarlandı. CPU, "DiagStatusUpdate" Bit'ini ancak program çevrimi OB'sinin ilk yürütülüşü tamamlandıktan sonra ayarladığı için, kullanıcı programınız, bir teşhis değişikliğinin yol alma OB'sinin yürütülmesi sırasında mı, yoksa program çevrimi OB'sinin ilk yürütülüşü sırasında mı ortaya çıktığını algılayamaz.
  - Her zaman 1 (high): Bit ("AlwaysTRUE" değişken adı), her zaman 1'e ayarlanır.
  - Her zaman 0 (low): Bit ("AlwaysFALSE" değişken adı), her zaman 0'a ayarlanır.
- Markör alanında bir Bayt'ı zamanlama markörü olarak atayabilirsiniz. Zamanlama markörü olarak yapılandırılan bu Bayt'ın her Bit'i bir kare darbe üretir. Zamanlama markörünün Bayt'ı, 0,5 Hz (yavaş) ile 10 Hz (hızlı) arasında 8 farklı frekans sunar. Bu Bit'leri, özellikle kenar yordamlarıyla bağlantılı olarak, kullanıcı programında eylemlerin periyodik tetiklenmesi için kontrol Bit'leri olarak kullanabilirsiniz.

CPU, bu Bayt'ları STOP işletim durumundan STARTUP işletim durumuna geçilirken başlatır. Zamanlama markörünün Bit'leri, STARTUP ve RUN işletim türlerinde CPU adımıyla senkron olarak değişir.

⚠ DİKKAT

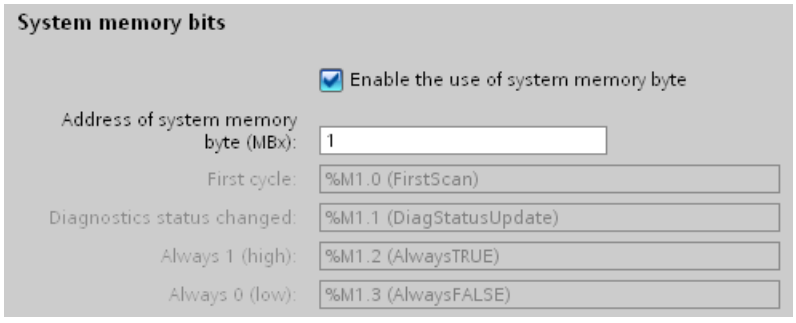
Sistem belleği veya zamanlama markörü tarafından Bit'lerin üzerine yazılırken oluşan tehlike

Sistem belleği veya zamanlama markörü tarafından Bit'lerin üzerine yazılması, bu fonksiyonlardaki verilerin zarar görmesine ve kullanıcı programınızın hatalı şekilde çalışmasına neden olabilir, bu da maddi hasara ve personelin yaralanmasına yol açabilir.

Zamanlama markörü ve sistem belleği markör alanındaki rezerve edilmemiş bellekler olduğu için, yordamlar ve iletişim öğeleri bu adreslere yazılabilir ve verilere zarar verebilir.

Bu fonksiyonların kusursuz şekilde çalıştığından emin olmak için, bu adreslere veri yazmayın ve prosesiniz veya makineniz için her zaman Acil Kapatma devresi uygulayın.

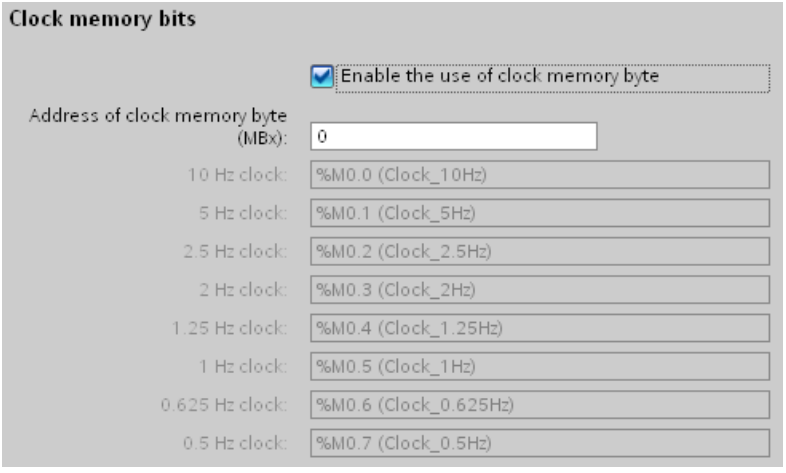
Sistem markörü Bit'leri, aşağıdaki tabloda gösterildiği gibi belirli anlamlara sahiptir:



Tablo 5-18 Sistem belleği

7	6	5	4	3	2	1	0
Rezerve Değer 0				Her zaman kapalı Değer 0	Her zaman açık Değer 1	Teşhis durumu göstergesi • 1: Değişiklik • 0: Değişiklik yok	İlk çevrim göstergesi • 1: Yol almadan sonraki ilk çevrim • 0: İlk çevrim değil

Zamanlama markörü, belirli mesafelerdeki münferit Bit'leri açan ve kapatan bir Bayt'ı yapılandırır. Her zamanlama markörü, ilgili M markör belleğinde bir kare darbe üretir. Bu Bit'ler, öncelikle kenar yordamlarıyla bağlantılı olarak, kullanıcı kodundaki eylemlerin periyodik olarak tetiklenmesi için kontrol Bit'leri olarak kullanılabilir.



Tablo 5-19 Zamanlama markörü

Bit numarası	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Değişken adı								
Periyot (periyotlar)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frekans (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Zamanlama markörü CPU çevrimiyle asenkron çalıştığı için, zamanlama markörünün durumu uzun bir çevrim sırasında birkaç kez değişebilir.

5.1.6 Teşhis tamponu

CPU, her teşhis olayı için bir giriş içeren bir teşhis tamponunu destekler. Her giriş; olayın ortaya çıktığı tarihi ve saati, bir olay kategorisini ve bir olay açıklamasını kapsar. Girişler, en yeni olay en üst konumda olacak şekilde kronolojik sırayla görüntülenir. Bu tampona maksimum 50 güncel olay kaydedilir. Tampon doluysa, her zaman yeni bir olay en eski olayın üzerine yazılır. CPU'nun gerilimi kesilirse teşhis tamponu korunur.

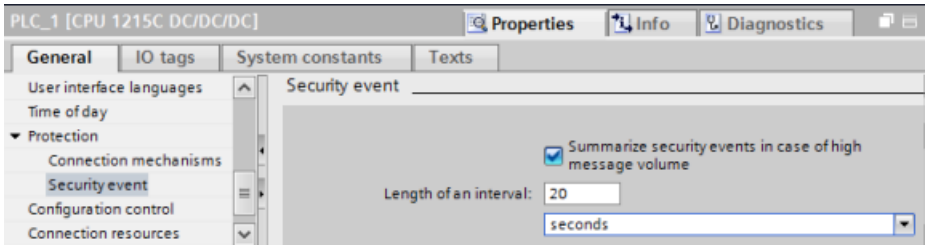
Teşhis tamponuna aşağıdaki olay tipleri kaydedilir:

- Sistemin her teşhis olayı, CPU hataları ve modül hataları
- CPU'nun her durum değişimi (yol alma, STOP veya RUN işletim durumuna geçiş)

Teşhis tamponuna (Sayfa 1128) erişim için çevrimiçi olmanız gerekir. Tamponu, "Çevrimiçi ve teşhis > Teşhis > Teşhis tamponu" öğelerini seçerek bulabilirsiniz.

Güvenlik teşhis olaylarının sayısının azaltılması

Bazı güvenlik olayları, teşhis tamponunda mükerrer girişler üretir. Bu mesajlar teşhis tamponunu doldurabilir ve diğer mesajları gizleyebilir. PLC'yi, güvenlik olayları nedeniyle tetiklenen teşhis mesajları sınırlanacak şekilde yapılandırabilirsiniz. Seçimlerinizi, CPU'nun cihaz konfigürasyonunda, sürekli ortaya çıkan mesajları engellemek istediğiniz zaman aralığına bağlı olarak yapabilirsiniz:



Güvenlik olaylarının bir zaman aralığında bir araya getirilmesi gerektiğini seçerseniz; saniye, dakika veya saat cinsinden bir zaman aralığı ve 1 ila 255 aralığında bir sayısal değer girebilirsiniz.

Güvenlik olaylarının kısıtlanması gerektiğini seçerseniz, şu olay tipleri kısıtlanır:

- Doğru veya yanlış şifreyle çevrimiçi olma
- Değişen iletişim verileri algılandı
- Memory card'da değişen veriler algılandı
- Değişen Firmware güncelleme dosyası algılandı
- CPU'ya yüklenen koruma kademesi (erişim koruması) değiştirildi
- Şifre yetkisi kısıtlandı veya etkinleştirildi (yordam veya CPU göstergesi aracılığıyla)
- Çevrimiçi erişim, mümkün olan aynı anda erişim denemesi sayısının aşılması nedeniyle reddedildi
- Zaman aşımı, mevcut bir çevrimiçi bağlantı devre dışıysa
- Web sunucusunda doğru veya yanlış şifreyle oturum açma
- CPU'nun bir yedek kopyasının oluşturulması
- CPU konfigürasyonunun geri yüklenmesi

5.1.7 Gerçek zaman saati

CPU bir gerçek zaman saatine sahiptir. Saat, CPU kapalıysa bir yüksek performanslı kondensatör aracılığıyla gerilimle beslenir. Yüksek performanslı kondensatör, CPU açıksa şarj edilir. CPU en az 24 saat boyunca açık kaldıysa, yüksek performanslı kondensatörün şarjı genellikle saati 20 gün süreyle çalıştırma için yeterli olur.

STEP 7, fabrikada ön ayarlı bir standart değere sahip olan veya standart değeri, sıfırlayarak fabrika ayarlarına döndürme yoluyla alan gerçek zaman saati için sistem saatini ayarlar. Gerçek zaman saatini kullanmak için bu saati ayarlamanız gerekir. Zaman notları, örneğin teşhis tamponu girişleri, veri protokolü dosyaları ve veri protokolü girişleri için sistem saatini temel alır. Gerçek zaman saatini, çevrimiçi CPU'nun "Çevrimiçi ve teşhis" görünümündeki "Saati ayarla" fonksiyonu (Sayfa 1119) aracılığıyla ayarlayabilirsiniz. Bu durumda STEP 7, ayarladığınız zaman artı veya eksi Windows işletim sisteminin UTC (Coordinated Universal Time) ile arasındaki saat farkı formülünün yardımıyla sistem saatini hesaplar. Gerçek zaman saati için yerel saat ayarlanırsa, Windows işletim sisteminizin saat dilimi ve yaz/kış saati uygulaması ayarlarının bölgesel saatinize karşılık gelmesi koşuluyla, UTC'ye uyan bir sistem saati oluşturulur.

STEP 7, sistem saatini okumak ve yazmak (RD_SYS_T ve WR_SYS_T), yerel saati okumak (RD_LOC_T) ve zaman dilimini ayarlamak (SET_TIMEZONE) için kullanılan yordamlar (Sayfa 313) içermektedir. RD_LOC_T yordamı, CPU'nun genel özelliklerindeki (Sayfa 144-145) gerçek zaman saatinin konfigürasyonunda girdiğiniz zaman dilimi ve yaz/kış saati için kullanılan zaman ofseti aracılığıyla yerel saati hesaplar. Bu ayarlar aracılığıyla yerel saatiniz için zaman dilimini ayarlayabilir, opsiyonel olarak yaz saati uygulamasını etkinleştirebilir ve yaz/kış saati uygulaması için başlangıç ve bitiş zamanlarını girebilirsiniz. Bu ayarları SET_TIMEZONE yordamlarıyla da yapılandırabilirsiniz.

5.1.8 Çıkışların RUN işletim durumundan STOP işletim durumuna geçiş için yapılandırılması

CPU'nun STOP işletim durumu için dijital ve analog çıkışların davranışını yapılandırabilirsiniz. Bir CPU'nun, bir SB'nin veya SM'nin her çıkışı için çıkışların değerlerini dondurabilir veya bir yedek değeri anahtarlayabilirsiniz:

- Önceden belirtilen bir çıkış değeriyle değiştirme (ön ayar): CPU'nun, SB'nin veya SM'nin her çıkışı (kanal) için bir yedek değer girebilirsiniz.
  Dijital çıkışlar için ö ayarlı yedek değer KAPALI, analog çıkışlar için 0'dır.
- Çıkışların son durumda dondurulması: Çıkışlar, RUN işletim durumundan STOP işletim durumuna geçiş gerçekleştiği anda sahip oldukları güncel değerlerini korur. Yol alma işleminden sonra çıkışlar standart yedek değere ayarlanır.

Çıkışların davranışını cihaz konfigürasyonunda yapılandırabilirsiniz. İlgili cihazın çıkışlarını yapılandırmak için münferit cihazları seçin ve "Özellikler" sekmesini açın.

NOT

Bazı desantral G/Ç modülleri, CPU'nun STOP işletim türüne verilecek yanıt için ek ayarlar sunar. Bu modüller için cihaz konfigürasyonundaki listeden istediğiniz opsiyonu seçin.

CPU RUN işletim durumundan STOP işletim durumuna geçerse, CPU proses görünümünü kaydeder ve dijital ve analog çıkışlar için ilgili değeri konfigürasyona uygun şekilde yazar.

5.2 Veri belleği, bellek alanları, G/Ç ve adresleme

5.2.1 S7-1200'ün verilerine erişim

STEP 7, sembolik programlamayı kolaylaştırır. Bu amaçla, verilerin adresleri için bellek adresleri ve G/Ç için PLC değişkenleri şeklinde veya bir kod bloğu içinde yerel değişkenler şeklinde ortaya çıkan sembolik adlar veya "değişkenler" oluşturun. Bu değişkenleri kullanıcı programınıza eklemek üzere, istediğiniz yordam parametreleri için değişken adlarını girebilirsiniz.

CPU'nun bellek alanlarını nasıl yapılandırdığını ve adreslediğini netleştirmek için, aşağıda PLC değişkenlerinin verilerin "mutlak" adreslemesine nasıl referans verdiğini gösterilmektedir. CPU, kullanıcı programının yürütülmesi sırasında verileri kaydetmek için birçok olanak sunar:

- Global bellek: CPU; girişler (G), çıkışlar (Ç) ve markörler (M) de dahil olmak üzere çok sayıda özel bellek alanı sunar. Bu belleğe, tüm kod blokları için kısıtlama olmadan erişilebilir.
- PLC değişken tablosu: STEP 7-PLC değişken tablosunda, bellekteki belirli adresler için sembolik adlar girebilirsiniz. Bu değişkenler, global olarak tüm STEP 7 programında geçerlidir ve anlamlı adlarla programlamaya izin verir.
- Veri bloğu (DB): Kullanıcı programınıza, kod blokları için verilerin kaydedilmesi amacıyla DB'ler ekleyebilirsiniz. Kaydedilen veriler, ilgili kod bloğu yürütüldükten sonra korunur. Bir "global" DB'ye, tüm kod blokları tarafından kullanılabilen veriler kaydedilir, ancak bir örnek DB'de sadece belirli bir FB için kullanılan veriler depolanır ve bu DB, FB'nin parametrelerine uygun şekilde yapılandırılır.
- Geçici bellek: Bir kod bloğu her çağrıldığında, CPU'nun işletim sistemi blok yürütülürken kullanılabilecek geçici veya yerel bellek alanını (L) serbest bırakır. Kod bloğunun yürütülmesi işlemi tamamlandıysa, CPU, başka kod bloklarının yürütülmesi için yerel belleği atar.

Her bellek alanı benzersiz bir adrese sahiptir. Kullanıcı programınız bu adres aracılığıyla bu bellek alanına erişebilir. Girişler (E) veya çıkışlar (A) için bellek alanlarına referans vererek, (örneğin E0.3 veya A1.7) proses görünümüne erişim gerçekleşir. Fiziksel girişe veya çıkışa doğrudan erişim için ":P" referansını (örn. E0.3:P, A1.7:P veya "Stop:P") ek olarak ekleyin.

Tablo 5-20 Bellek alanları

Bellek alanı	Açıklama	Force fonksiyonu	Kalıcı
E Girişlerin proses görünümü E_:P [1] (fiziksel giriş)	Çevrimin başlangıcında fiziksel girişlerden kopyalanır	Hayır	Hayır
	CPU, SB veya SM'den fiziksel girişlerin doğrudan okunması	Evet	Hayır
A Çıkışların proses görünümü A_:P [1] (fiziksel çıkış)	Çevrimin başlangıcında fiziksel çıkışlara kopyalanır	Hayır	Hayır
	CPU, SB veya SM'den fiziksel çıkışlara doğrudan yazma	Evet	Hayır

[1] Fiziksel girişlere ve çıkışlara doğrudan erişim (okuma veya yazma) için adresi veya değişkeni ":P" (örn. E0.3:P, A1.7:P veya "Stop:P") koduyla tamamlayabilirsiniz.

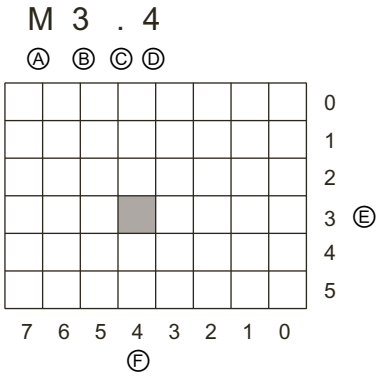
Bellek alanı	Açıklama	Force fonksiyonu	Kalıcı
M Markör	Kumanda ünitesi ve veri belleği	Hayır	Evet (opsiyonel)
L geçici bellek	Bir blok için kullanılan geçici, yerel veriler	Hayır	Hayır
DB Veri bloğu	FB'ler için kullanılan veri belleği ve parametre belleği	Hayır	Evet (opsiyonel)

[1] Fiziksel girişlere ve çıkışlara doğrudan erişim (okuma veya yazma) için adresi veya değişkeni ":P" (örn. E0.3:P, A1.7:P veya "Stop:P") koduyla tamamlayabilirsiniz.

Her bellek alanı benzersiz bir adrese sahiptir. Kullanıcı programınız bu adres aracılığıyla bu bellek alanına erişebilir. Mutlak adres, aşağıdaki elemanlardan oluşur:

- Bellek alanı kodu (E, A veya M gibi)
- Erişilmesi gereken verilerin boyutu (Byte için "B", Word için "W" veya DWord için "D" gibi)
- Verilerin (Bayt 3 veya Sözcük 3 gibi) başlangıç adresi

Bir Boole değeri için kullanılan adresteki bir Bit'e erişilirken, bellek alanını, Bayt adresini ve verilerin Bit adresini girin (E0.0, A0.1 veya M3.4 gibi).



A	Bellek alanı kodu	E	Bellek alanının Bayt'ları
B	Bayt'ın adresi: Bayt 3	F	Seçilen Bayt'ın Bit'leri
C	Ayırma işareti ("Bayt.Bit")		
D	Bayt'taki Bit adresi (8 / 4. Bit)		

Örnekte, Bit'in adresini (Bit 4) ayırmak için bellek alanı ve Bayt'ın adresinin (M = markörlerin alanı ve 3 = Byte 3) arkasından bir nokta (".") gelmektedir.

CPU'nun bellek alanlarındaki verilere erişim

STEP 7, sembolik programlamayı kolaylaştırır. Tipik olarak değişkenler, PLC değişken tablosunda, bir veri bloğunda veya bir OB, FC veya FB'nin arayüzünde oluşturulur. Bu değişkenler bir ad, veri tipi, ofset ve yorumu kapsar. Ayrıca bir veri bloğunda bir başlangıç değeri belirtilebilir. Değişken adlarını yordam için parametre olarak girerek programlama sırasında bu değişkenlerden faydalanabilirsiniz. Opsiyonel olarak mutlak işlemciyi (bellek, alan, boyut ve ofset) yordam parametreleri olarak girebilirsiniz. Aşağıdaki alt bölümlerde yer alan örneklerde mutlak işlemcileri nasıl girebileceğiniz gösterilmektedir. % karakteri, programlama editörü tarafından otomatik olarak mutlak işlemcinin önüne eklenir. Programlama editöründe şu görünümler arasından seçim yapabilirsiniz: sembolik, sembolik ve mutlak veya mutlak.

E (girişlerin proses görünümü): CPU, her çevrimde bir çevrim OB'si yürütülmeden hemen önce çevrebirim girişlerini (fiziksel girişler) sorgular ve bu değerleri girişlerin proses görünümüne yazar. Girişlerin proses görünümüne Bit, Bayt, sözcük ve çift sözcük formatında erişebilirsiniz. Hem okuma hem de yazma erişimine izin verilir, ancak proses görünümünün girişleri tipik olarak sadece okunur.

Tablo 5-21 E bellek alanı için mutlak adresleme

Bit	E[Bayt adresi].[Bit adresi]	E0.1
Bayt, sözcük veya çift sözcük	E[Boyut][Başlangıç Bayt'ının adresi]	EB4, EW5 veya ED12

Adrese ":P" referansını eklerseniz, CPU, SB, SM veya desantral modülün dijital ve analog girişlerini doğrudan okuyabilirsiniz. E üzerinden erişim ve E yerine E_:P üzerinden erişim arasındaki fark, verilerin, girişlerin proses görünümü yerine doğrudan adreslenen girişlerden gelmesidir. E_:P üzerinden bu erişim, veriler, girişlerin proses görünümünün en son güncellenmesi sırasında oluşturulan bir kopya yerine doğrudan kaynaktan okunduğu için doğrudan okuma erişimi olarak da tanımlanır.

Fiziksel girişler değerlerini doğrudan kendilerine bağlı alan cihazlarından aldıkları için, bu girişlere yazamazsınız. E_:P üzerinden erişimler, okunabilen veya yazılabilen girişlere erişimlerin tersine salt okuma erişimleridir.

E_:P üzerinden erişimler, ayrıca CPU, SB veya SM tarafından desteklenen girişlerin boyutu ile sınırlıdır (en yakın Bayt'a yuvarlanır). Örneğin, 2 DE/2 DA içeren bir SB'nin girişleri E4.0'dan başlayacak şekilde yapılandırıldıysa, bunlara E4.0:P ve E4.1:P veya EB4:P ile erişilebilir. E4.2:P ila E4.7:P'ye erişim hata olarak algılanmaz, ancak bu adresler dolu olmadığı için anlamsızdır. EW4:P ve ED4:P'ye erişim denemeleri, bu SB'nin Bayt ofsetini aştığı için yasaktır.

E_:P üzerinden erişimlerin, girişlerin proses görünümüne kaydedilen değer üzerinde hiçbir etkisi yoktur.

Tablo 5-22 E bellek alanı için mutlak adresleme (doğrudan)

Bit	E[Bayt adresi].[Bit adresi]:P	E0.1:P
Bayt, sözcük veya çift sözcük	E[Boyut][Bayt'ın başlangıç adresi]:P	EB4:P, EW5:P veya ED12:P

A (Çıkışların proses görünümü): CPU, çıkışların proses görünümüne kaydedilen değerleri fiziksel çıkışlara kopyalar. Çıkışların proses görünümüne Bit, Bayt, sözcük ve çift sözcük formatında erişebilirsiniz. Proses görünümünün çıkışlarına hem okuma hem de yazma erişimine izin verilir.

Tablo 5-23 A bellek alanı için mutlak adresleme

Bit	A[Bayt adresi].[Bit adresi]	A1.1
Bayt, sözcük veya çift sözcük	A[Boyut][Başlangıç Bayt'ının adresi]	AB5, AW10, AD40

Adrese ":P" referansını eklerseniz, doğrudan CPU, SB, SM veya desantral modülün dijital ve analog çıkışlarına yazabilirsiniz. A üzerinden erişim ve A yerine A_:P üzerinden erişim arasındaki fark, verilerin doğrudan adreslenen çıkışlara ve ek olarak çıkışların proses görünümüne gitmesidir (her iki alana da yazılır). A_:P üzerinden bu erişim, veriler doğrudan hedef adrese gönderildiği ve çıkışların proses görünümünden gelecek bir sonraki güncellemeyi beklemek zorunda olmadığı için doğrudan yazma erişimi olarak da adlandırılır.

Fiziksel çıkışlar kendilerine bağlı alan cihazlarını doğrudan kontrol ettiği için, bu çıkışlar okunmamalıdır. Bu nedenle A_:P üzerinden erişimler, okunabilen veya yazılabilen çıkışlara erişimlerin tersine, salt yazma erişimleridir.

A_:P üzerinden erişimler, ayrıca CPU, SB veya SM tarafından desteklenen çıkışların boyutu ile sınırlıdır (en yakın Bayt'a yuvarlanır). Örneğin, 2 DE/2 DA içeren bir SB'nin çıkışları A4.0'dan başlayacak şekilde yapılandırıldıysa, bunlara A4.0:P ve A4.1:P veya AB4:P ile erişilebilir. A4.2:P ila A4.7:P'ye erişim hata olarak algılanmaz, ancak bu adresler dolu olmadığı için anlamsızdır. AW4:P ve AD4:P'ye erişim denemeleri, bu SB'nin Bayt ofsetini aştığı için yasaktır.

A_:P üzerinden erişimler, fiziksel çıkışı ve ayrıca çıkışların proses görünümüne kaydedilen değeri de etkiler.

Tablo 5-24 A bellek alanı için mutlak adresleme (doğrudan)

Bit	A[Bayt adresi].[Bit adresi]:P	A1.1:P
Bayt, sözcük veya çift sözcük	A[Boyut][Bayt'ın başlangıç adresi]:P	AB5:P, AW10:P veya AD40:P

M (markörün bellek alanı): Yordamların ara sonuçlarını veya diğer kumanda bilgilerini kaydetmek üzere kontrol röleleri ve veriler için markörlerin (M) bellek alanını kullanabilirsiniz. Markörlerin bellek alanına Bit, Bayt, sözcük ve çift sözcük formatında erişebilirsiniz. Markör belleğine hem okuma hem de yazma erişimine izin verilir.

Tablo 5-25 M bellek alanı için mutlak adresleme

Bit	M[Bayt adresi].[Bit adresi]	M26.7
Bayt, sözcük veya çift sözcük	M[Boyut][Başlangıç Bayt'ının adresi]	MB20, MW30, MD50

Temp (geçici bellek): CPU, geçici belleği ihtiyaca bağlı olarak atar. CPU, geçici belleği kod bloğuna atar ve kod bloğu başlatıldığında (bir OB'de) veya çağrıldığında (bir FC veya FB'de) bellek adreslerini 0 olarak başlatır.

Geçici bellek, markör belleğine (M) benzer. Ancak aralarında önemli bir fark mevcuttur: Markör belleği "global" geçerlilik alanına sahipken, geçici bellek "yerel" geçerlilik alanına sahiptir:

- Markörün bellek alanı: Her OB, FB ve her FC, markörün bellek alanındaki verilere erişebilir, yani veriler kullanıcı programının tüm elemanlarının kullanımına global olarak sunulur.
- Geçici bellek: CPU, geçici belleğin verilerine erişimi, geçici bellekteki adresi oluşturan veya bildiren OB, FB veya FC ile sınırlandırır. Geçici bellekteki adresler yerel olarak kalır ve kod bloğu başka bir kod bloğunu çağırsa bile farklı kod blokları tarafından birlikte kullanılmaz. Örnek: Bir OB bir FC'yi çağırdığında, FC, çağrıldığı OB'nin geçici belleğine erişemez.

CPU, her OB öncelik sınıfı için geçici (yerel) bellek sunar:

- İlgili FB'ler ve FC'ler de dahil olmak üzere yol alma ve program çevrimi için 16 KB
- İlgili FB'ler ve FC'ler de dahil olmak üzere her ek alarm olayı iş parçacığı için 6 KB

Geçici belleğe sadece sembolik adresleme üzerinden erişebilirsiniz.

Programınızdaki blokların kapladığı geçici belleğin (yerel veriler) miktarını STEP 7'deki çağrı yapısı aracılığıyla belirleyebilirsiniz. Proje gezgininde "Program bilgilerini" ve ardından "Çağrı yapısı" sekmesini seçin. Burada, programınızdaki tüm OB'ler görüntülenir ve her seferinde, OB'ler tarafından çağrılan blokların görüntülenmesini sağlayabilirsiniz. Her blok için, atanmış

yerel veriler görüntülenir. Çağrı yapısını STEP 7'de de **Ek özellikler > Çağrı yapısı** menü komutu aracılığıyla çağırabilirsiniz.

DB (veri bloğu): Bir yordamın ara sonuçları veya FB'ler için diğer kontrol parametreleri ve zaman ve sayaçlar gibi birçok yordam için kullanılan veri yapıları da dahil olmak üzere farklı veri türlerini kaydetmek için veri bloklarını kullanın. Veri bloklarına Bit, Bayt, sözcük ve çift sözcük formatında erişebilirsiniz. Yazma korumalı olmayan veri blokları için hem okuma hem de yazma erişimine izin verilir. Yazma korumalı veri blokları için sadece okuma erişimine izin verilir.

Tablo 5-26 DB bellek alanı için mutlak adresleme

Bit	DB[Veri bloğu numarası].DBX[Bayt adresi].[Bit adresi]	DB1.DBX2.3
Bayt, sözcük veya çift sözcük	DB[Veri bloğu numarası].DB[Boyut].[Başlangıç Bayt'ı adresi]	DB1.DBB4, DB10.DBW2, DB20.DBD8

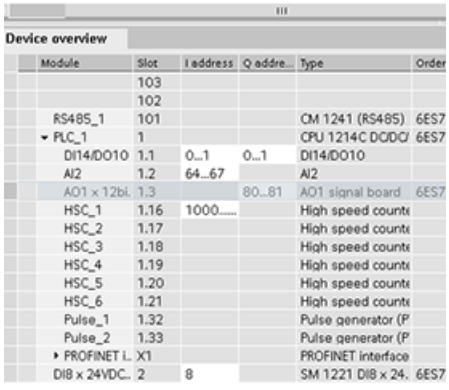
NOT

LAD veya FBD'de bir mutlak adres belirtirseniz, STEP 7, burada bir mutlak adresin söz konusu olduğunu belirtmek için bu adresin önüne "%" karakterini ekler. Programlama yaparken, bir mutlak adresi "%" karakteriyle ya da "%" karakteri olmadan girebilirsiniz (örnek: %E0.0 veya E.0). "%" karakterini eklemezseniz, STEP 7 bu karakteri ekler.

SCL'de, burada bir mutlak adresin söz konusu olduğunu belirtmek için adresin önüne "%" karakterinin girilmesi gerekir. "%" karakteri eksikse, STEP 7, derleme sırasında tanımsız bir değişken hatası üretir.

CPU'da ve G/Ç modüllerinde G/Ç'nin yapılandırılması





Cihaz konfigürasyonunuza bir CPU ve G/Ç modülleri eklediğinizde, STEP 7, G ve Ç adreslerini otomatik olarak atar. Cihaz konfigürasyonunda adres alanını seçip yeni sayılar girerek ön ayarlı adreslemeyi değiştirebilirsiniz.

- Dijital girişler ve çıkışlar, modülün tüm girişleri veya çıkışları (noktaları) kullanıp kullanmadığından bağımsız olarak STEP 7'de 8 noktalı (1 Bayt) gruplar halinde atanır.
- Analog girişler ve çıkışlar STEP 7'de 2'li gruplar halinde atanır ve burada her analog nokta 2 Bayt (16 Bit) yer kaplar.

Şekilde, iki SM ve bir SB'ye sahip bir CPU 1214C örneği gösterilmektedir. Bu örnekte, DI8 modülünün adresi 8 yerine 2 olarak değiştirilebilir. Araç, yanlış boyuta sahip olan veya diğer adreslerle çakışan adres aralıklarının değiştirilmesi yoluyla kullanıcıyı destekler.

5.3 Analog değerlerin işlenmesi

Analog sinyal modülleri, giriş sinyalleri iletir veya bir gerilim aralığını veya bir akım aralığını temsil eden çıkış değerleri bekler. Bu aralıklar ±10 V, ±5 V, ±2,5 V veya 0 ila 20 mA şeklindedir. Modüller tarafından sağlanan değerler tamsayı değerleridir. Bu sırada 0 ila 27648, akım ve -27648 ila 27648 ise gerilim için kullanılan nominal aralığı gösterir. Bu aralıkların dışındaki değerler, bir taşma ya da bir aşağı taşma olduğunu gösterir. Aralık dışında bulunan değer tipleriyle ilgili ayrıntıları, Analog girişlerin gösterimi (Sayfa 1271) ve Analog çıkışların gösterimi (Sayfa 1272) tablolarında bulabilirsiniz.

Kumanda programınızda, örneğin hacim, sıcaklık, ağırlık veya başka bir nicel değeri göstermek üzere muhtemelen bu değerleri fiziksel birimlerle kullanmanız gerekir. Bunun için analog girişi önce 0,0 ile 1,0 arasında bir gerçek sayı (kayan nokta değeri) olarak standartlaştırmanız gerekir. Ardından, görüntülenecek fiziksel birimin minimum ve maksimum değeri için değeri ölçeklemeniz gerekir. Bir analog çıkış değerine dönüştürmeniz gereken fiziksel birimlerdeki değerlerde, önce fiziksel birimlerdeki değeri 0,0 ile 1,0 arasında bir değer olarak standartlaştırın. Ardından analog modülün aralıklarına bağlı olarak değeri 0 ile 27648 veya -27648 ile 27648 arasında ölçekleyin. STEP 7, bunun için NORM_X ve SCALE_X yordamlarını (Sayfa 276) sunar. Analog değerleri (Sayfa 228) ölçeklemek için CALCULATE yordamını da kullanabilirsiniz.

Örnek: Analog değer işleme

Örneğin, akım aralığı 0 - 20 mA olan bir analog girişi ele alalım. Analog giriş modülü, 0 ila 27648 aralığında ölçüm değerleri döndürür. Bu analog giriş değerinin 50 °C ila 100 °C'lik bir sıcaklık aralığını ölçmek için kullanıldığını varsayalım. Bu durumda aşağıdaki örnek değerler, belirtilen değerlere karşılık gelir:

Analog giriş değeri	Teknik birimler
0	50 ℃
6192	62,5 ℃
12384	75 ℃
18576	87,5 ℃
27648	100 ℃

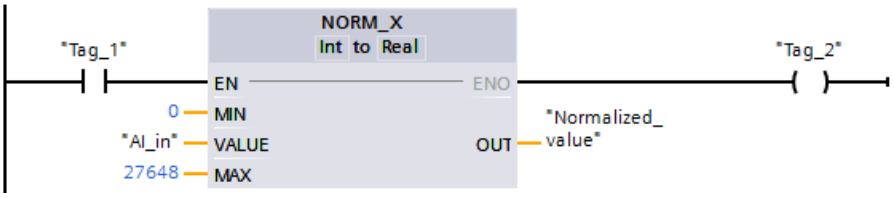
Bu örnekteki fiziksel birimler, analog giriş değeri aracılığıyla aşağıdaki şekilde hesaplanır:
Fiziksel birimlerdeki değer = 50 + (Analog giriş değeri) * (100 - 50) / (27648 - 0)
Genel olarak denklem şu şekilde görünür:

Fiziksel birimlerdeki değer = (fiziksel birimlerin alt aralığı) +
(Analog giriş değeri) *
(fiziksel birimlerin üst aralığı - fiziksel birimlerin alt aralığı) /
(maksimum analog giriş aralığı - minimum analog giriş aralığı)

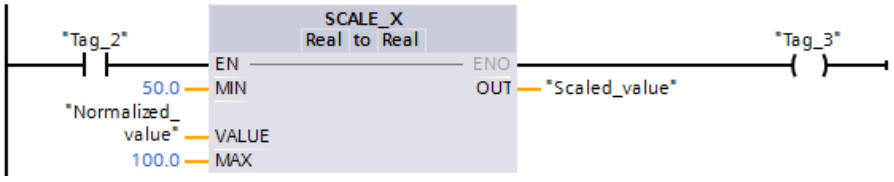
PLC uygulamalarında genelde analog giriş değeri 0,0 ile 1,0 arasında bir kayan nokta değeri olarak standartlaştırılır. Sonuç, daha sonra fiziksel birimlerin aralığında bir kayan nokta

değerine dönüştürülür. Kolaylık sağlama amacıyla, aşağıdaki LAD yordamlarında sabit aralık değerleri kullanılır; ancak değişkenler de kullanılabilir.

Ağ 1



Ağ 2



5.4 Veri tipleri

Veri tipleri, bir veri elemanının boyutunu ve verilerin değerlendirme türünü belirtir. Her yordam parametresi en az bir veri tipini, bazı parametreler birden fazla veri tipini destekler. İlgili parametre için hangi veri tiplerinin desteklendiğini görüntülemek için imleci bir yordamın parametre alanı üzerinde tutun.

Bir formel parametre, bir yordamda bulunan ve yordam tarafından kullanılacak verilerin adresini belirten koddur (örnek: Bir ADD yordamının IN1 girişi). Bir güncel parametre, yordam tarafından kullanılacak verileri içeren adres (öne eklenen "%" karakteriyle birlikte) veya sabittir (örnek: %MD400 "Sayı_Widgets"). Belirttiğiniz güncel parametrenin veri tipi, yordam tarafından belirtilen desteklenen formel parametre veri tiplerinden birine karşılık gelmelidir.

Bir güncel parametre belirtirseniz, bir değişken (sembol) veya bir mutlak (doğrudan) bir adres belirtmeniz gerekir. Değişkenler bir sembolik ada (değişken adı) bir veri tipi, bellek alanı, bellek ofseti ve yorum atar ve PLC değişken editöründe veya bir bloğun (OB, FC, FB ve DB) arayüz editöründe belirtilebilir. Atanmış değişkeni olmayan bir mutlak adres girerseniz, desteklenen bir veri türüne karşılık gelen bir boyut kullanmanız gerekir, giriş sırasında standart bir değişken oluşturulur.

String, Struct, Array ve DTL veri tipleri hariç tüm veri tipleri PLC değişken editöründe ve blok arayüzlerinde mevcuttur. String, Struct, Array ve DTL ise sadece blok arayüzlerinde mevcuttur. Giriş parametrelerinin birçoğu için sabit bir değer de girebilirsiniz.

- Bit ve Bit dizileri (Sayfa 100): Bool (Boole değeri veya Bit değeri), Byte (8 Bit Bayt değeri), Word (16 Bit sözcük değeri), DWord (32 Bit çift sözcük değeri)
- Tamsayı (Sayfa 101)
  - USInt (ön işaretsiz 8 Bit tamsayı), SInt (ön işaretli 8 Bit tamsayı),
  - UInt (ön işaretsiz 16 Bit tamsayı), Int (ön işaretli 16 Bit tamsayı),
  - UDInt (ön işaretsiz 32 Bit tamsayı), DInt (ön işaretli 32 Bit tamsayı),
- Kayar noktalı sayı/gerçek sayı (Sayfa 101): Real (32 Bit gerçek değer veya kayan nokta değeri), LReal (64 Bit gerçek değer veya kayan nokta değeri)
- Saat ve tarih (Sayfa 102): Time (32 Bit IEC zaman değeri), Date (16 Bit tarih değeri), TOD (32 Bit saat değeri), DTL (12 Bayt tarih ve saat yapısı)

- Karakter ve karakter dizisi (Sayfa 103): Char (8 Bit tek karakter), String (maksimum 254 karakter içeren değişken uzunlukta karakter dizisi)
- Array (Sayfa 105)
- Veri yapısı (Sayfa 106): Struct
- PLC veri tipi (Sayfa 106)
- Varyant veri tipi (Sayfa 107)

Ayrıca aşağıdaki sayısal BCD formatı, veri tipi olarak mevcut olmamasına rağmen dönüştürme yordamları tarafından desteklenir.

Tablo 5-27 BCD formatının boyutu ve aralığı

Format	Boyut (Bit)	Aralık	Sabit giriş örneği
BCD16	16	-999 ila 999	123, -123
BCD32	32	-9999999 ila 9999999	1234567, -1234567

5.4.1 Bool, Bayt, Word ve DWord veri tipleri

Tablo 5-28 Bit ve Bit dizisi veri tipleri

Veri tipi	Bit boyutu	Sayı tipi	Sayı aralığı	Sabitlerle ilgili örnekler	Adreslerle ilgili örnekler
Bool	1	Boole	YANLIŞ veya DOĞRU	DOĞRU	I1.0 Q0.1 M50.7 DB1.DBX2.3 Değişken adı
		İkili	2#0 veya 2#1	2#0	
		Ön işaretsiz tamsayı	0 veya 1	1	
		Sekizli	8#0 veya 8#1	8#1	
		Onaltılı	16#0 veya 16#1	16#1	
Byte	8	İkili	2#0 ila 2#1111_1111	2#1000_1001	IB2 MB10 DB1.DBB4 Değişken adı
		Ön işaretsiz tamsayı	0 ila 255	15	
		Ön işaretli tamsayı	-128 ila 127	-63	
		Sekizli	8#0 ila 8#377	8#17	
		Onaltılı	B#16#0 ila B#16#FF, 16#0 ila 16#FF	B#16#F, 16#F	
Word	16	İkili	2#0 ila 2#1111_1111_1111_1111	2#1101_0010_1001_0110	MW10 DB1.DBW2 Değişken adı
		Ön işaretsiz tamsayı	0 ila 65535	61680	
		Ön işaretli tamsayı	-32768 ila 32767	72	
		Sekizli	8#0 ila 8#177_777	8#170_362	
		Onaltılı	W#16#0 ila W#16#FFFF, 16#0 ila 16#FFFF	W#16#F1C0, 16#A67B	
DWord	32	İkili	2#0 ila 2#1111_1111_1111_1111_1111_1111_1111_1111	2#1101_0100_1111_1110_1000_1100	MD10 DB1.DBD8 Değişken adı
		Ön işaretsiz tamsayı*	0 ila 4_294_967_295	15_793_935	

* Alt çizgi "_", sekizden fazla rakam içeren sayıların okunmasını kolaylaştırmak için kullanılan bir binler basamağı ayırma karakteridir.

Veri tipi	Bit boyutu	Sayı tipi	Sayı aralığı	Sabitlerle ilgili örnekler	Adreslerle ilgili örnekler
DWord	32	Ön işaretli tamsayı*	-2_147_483_648 ila 2_147_483_647	-400000	MD10 DB1.DBD8 Değişken adı
		Sekizli	8#0 ila 8#37_777_777_777	8#74_177_417	
		Onaltılı	DW#16#0000_0000 ila DW#16#FFFF_FFFF, 16#0000_0000 ila 16#FFFF_FFFF	DW#16#20_F30A, 16#B_01F6	

* Alt çizgi "_", sekizden fazla rakam içeren sayıların okunmasını kolaylaştırmak için kullanılan bir binler basamağı ayırma karakteridir.

5.4.2 Tamsayı veri tipleri

Tablo 5-29 Tamsayı veri tipleri (U = ön işaretsiz, S = kısa, D = çift)

Veri tipi	Bit boyutu	Sayı aralığı	Sabitlerle ilgili örnekler	Örnek adresler
USInt	8	0 ila 255	78, 2#01001110	MB0, DB1.DBB4, Değişken adı
SInt	8	-128 ila 127	+50, 16#50	
UInt	16	0 ila 65.535	65295, 0	MW2, DB1.DBW2, Değişken adı
Int	16	-32.768 ila 32.767	30000, +30000	
UDInt	32	0 ila 4.294.967.295	4042322160	MD6, DB1.DBD8, Değişken adı
DInt	32	-2.147.483.648 ila 2.147.483.647	-2131754992	

5.4.3 Kayar noktalı sayı/gerçek sayı veri tipleri

Gerçek sayılar (veya kayar noktalı sayılar), ANSI/IEEE7541985 standardında açıklandığı gibi tek hassasiyetli 32 Bit sayılar (Real) veya çift hassasiyetli 64 Bit sayılar (LReal) olarak gösterilir. Tek hassasiyetli kayar noktalı sayılar 6 anlamlı basamağa kadar ve çift hassasiyetli kayar noktalı sayılar ise 15 anlamlı basamağa kadar doğrudur. Bir kayar noktalı sabit girerseniz maksimum 6 (Real) veya 15 (LReal) anlamlı basamak belirtebilirsiniz.

Tablo 5-30 Kayar noktalı sayı/gerçek sayı veri tipleri (L = uzun)

Veri tipi	Bit boyutu	Sayı aralığı	Sabitlerle ilgili örnekler	Adreslerle ilgili örnekler
Real	32	-3.402823e+38 ila -1.175 495e-38, ±0, +1.175 495e-38 ila +3.402823e+38	123.456, -3.4, 1,0e-5	MD100, DB1.DBD8, Değişken adı
LReal	64	-1.7976931348623158e+308 ila -2.2250738585072014e-308, ±0, +2.2250738585072014e-308 ila +1.7976931348623158e+308	12345,123456789e40, 1,2E+40	DB-Name.Var-Name Kurallar: • Direkt adresleme desteklenmez • Bir OB'nin, FB'nin veya FC'nin blok arayüzünde atanabilir

Çok büyük ve çok küçük sayılar da dahil olmak üzere uzun bir dizi değer gerektiren hesaplamalarda hatalı sonuçlarla karşılaşılabilir. Bu durum, sayılar 10 üzeri x kadar farklı olduğunda ortaya çıkabilir, bu sırada x > 6 (Real) veya 15'tir (LReal). Örnek (Real): 100 000 000 + 1 = 100 000 000.

5.4.4 Saat ve tarih veri tipleri

Tablo 5-31 Saat ve tarih veri tipleri

Veri tipi	Boyut	Aralık	Sabit giriş örnekleri
Time	32 Bit	T#-24d_20h_31m_23s_648ms ila T#24d_20h_31m_23s_647ms Şu şekilde kaydedilir: -2.147.483.648 msn ila +2.147.483.647 msn	T#5m_30s T#1d_2h_15m_30s_45ms TIME#10d20h30m20s630ms 500h10000ms 10d20h30m20s630ms
Date	16 Bit	D#1990-1-1 ila D#2168-12-31	D#2009-12-31 DATE#2009-12-31 2009-12-31
Time_of_Day	32 Bit	TOD#0:0:0.0 ila TOD#23:59:59.999	TOD#10:20:30.400 TIME_OF_DAY#10:20:30.400 23:10:1
DTL (Date and Time Long)	12 Bayt	Min.: DTL#1970-01-01-00:00:00.0 Maks.: DTL#2262-04-11:23:47:16.854 775 807	DTL#2008-12-16-20:30:20.250

Zaman

TIME verileri, milisaniye olarak değerlendirilen ön işaretli çift tamsayılar olarak kaydedilir. Editör formatı gün (g), saat (sa), dakika (dak), saniye (sn) ve milisaniye (msn) bilgilerini işleyebilir.

Tüm zaman birimlerini belirtmek gerekmez. Dolayısıyla, örn. T#5h10s ve 500h geçerlidir.

Münferit birimlerle belirtilen tüm değerlerin toplam değeri, zaman veri tipi için milisaniye cinsinden belirtilen üst veya alt limiti aşmamalı ve bu değerin altına inmemelidir (-2.147.483.648 msn ila +2.147.483.647 msn).

Tarih

DATE verileri, belirtilen tarihi elde etmek için 01.01.1990 şeklindeki temel tarihe eklenecek gün sayısı olarak değerlendirilen ön işaretsiz bir tamsayı değeri olarak kaydedilir. Editör formatı bir yıl, bir ay ve bir gün formatı belirtmek zorundadır.

TOD

TOD verileri (TIME_OF_DAY), belirtilen saat için gece yarısından itibaren geçen milisaniye sayısı olarak değerlendirilen ön işaretsiz çift tamsayı olarak kaydedilir (gece yarısı = 0 msn). Saat (24 s/gün), dakika ve saniye değerleri belirtilmelidir. Saniyenin ondalık basamak bilgisi opsiyoneldir.

DTL

DTL veri tipi (Date and Time Long), tarih ve saatle ilgili bilgileri kaydetmek için 12 Bayt'lık bir yapıdan faydalanır. DTL veri tipini bir bloğun geçici belleğinde veya bir DB'de tanımlayabilirsiniz. DB editöründeki başlangıç değeri için kullanılan sütuna tüm bileşenler için bir değer girilmelidir.

Tablo 5-32 DTL'nin boyutu ve aralığı

Uzunluk (Bayt)	Format	Değer aralığı	Değer girişi örneği
12	Saat ve takvim Yıl-Ay-Gün:Saat:Dakika: Saniye.Nanosaniye	Min.: DTL#1970-01-01-00:00:00.0 Maks.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20: 30:20.250

DTL veri tipinin her bileşeni farklı bir veri tipi ve değer aralığı içerir. Belirtilen bir değerin veri tipi, ilgili bileşenlerinde veri tipine karşılık gelmelidir.

Tablo 5-33 DTL yapısının elemanları

Bayt	Bileşen	Veri tipi	Değer aralığı
0	Yıl	UINT	1970 ila 2554
1			
2	Ay	USINT	1 ila 12
3	Gün	USINT	1 ila 31
4	Haftanın günü [1]	USINT	1 (Pazar) ila 7 (Cumartesi)[1]
5	Saat	USINT	0 ila 23
6	Dakika	USINT	0 ila 59
7	Saniye	USINT	0 ila 59
8	Nanosaniye	UDINT	0 ila 999.999.999
9			
10			
11			

[1] Yıl-Ay-Gün:Saat:Dakika:

5.4.5 Karakter ve karakter dizisi veri tipleri

Tablo 5-34 Karakter ve karakter dizisi veri tipleri

Veri tipi	Boyut	Aralık	Sabit giriş örneği
Char	8 Bit	16#00 ila 16#FF	'A', 't', '@', 'ä', '∑'
WChar	16 Bit	16#0000 ila 16#FFFF	'A', 't', '@', 'ä', '∑', Asya karakterleri, Kiril karakterleri ve diğerleri
String	n+ 2 Bayt	n = (0 ila 254 Bayt)	"ABC"
WString	n+ 2 sözcük	n = (0 ila 65534 sözcük)	"ä123@XYZ.COM"

Char ve WChar

Bir Char (karakter), bellekte bir Bayt yer kaplar ve genişletilmiş ASCII karakter kodları da dahil olmak üzere ASCII formatında tek bir karakter kaydeder. Bir WChar, bellekte bir sözcük yer kaplar ve herhangi bir çift Bayt karakter gösterimini içerebilir.

Editör sözdiziminde, karakterden önce ve sonra tek bir tırnak işareti girilir. Görünür karakterleri ve kontrol karakterlerini kullanabilirsiniz.

String ve WString

CPU, tek Bayt karakterlerden oluşan bir diziyi kaydetmek için String veri tipini destekler. String veri tipi, toplam karakter sayısını (karakter dizisindeki karakterlerin sayısı) ve gerçek karakter sayısını içerir. String veri tipi, maksimum toplam karakter sayısını (1 Bayt), gerçek karakter sayısını (1 Bayt) kaydetmek için 256 Bayt ve karakter dizisindeki maksimum 254 Bayt içerir. Bir String veri tipindeki her Bayt, 16#00 ile 16#FF arasında herhangi bir değeri alabilir.

WString veri tipi, bir sözcük (çift Bayt) içeren değerler için daha uzun karakter dizileri sağlar. İlk sözcük maksimum karakter sayısını; bir sonraki sözcük gerçek toplam karakter sayısını içerir, sonraki karakter dizisi maksimum 65334 sözcük içerebilir. Bir WString veri tipindeki her sözcük, 16#0000 ile 16#FFFF arasında herhangi bir değer olabilir.

IN tipi yordam parametreleri için münferit tırnak işaretleri içinde harflerden oluşan karakter dizileri (sabitler) belirtebilirsiniz. Örneğin 'ABC', S_CONV yordamının IN parametresi için giriş olarak kullanılabilen üç karakterden oluşan bir karakter dizisidir. OB, FC, FB ve DB için kullanılan blok arayüzü editörlerinde "String" veya "WString" veri tipini seçerek karakter dizisi değişkenleri de oluşturabilirsiniz. PLC değişken editöründe karakter dizisi oluşturamazsınız.

Veri tipi seçimi için kullanılan açılır listeden ilgili veri tipini seçtikten sonra, "String" veya "WString" anahtar sözcüğünün ardından köşeli parantezlerin yardımıyla maksimum karakter dizisi boyutunu Bayt (String) veya sözcük (WString) cinsinden belirtebilirsiniz. Örnek: "MyString String[10]" ifadesinde, MyString karakter dizisi için 10 Bayt'lık bir boyut belirtilmektedir. Maksimum boyut bilgisini içeren köşeli parantezleri kullanmazsanız, String veri tipi için 254 ve WString veri tipi için 65534 boyutu kabul edilir. Bununla birlikte "MyWString WString[1000]" ifadesi, 1000 sözcüklü bir WString veri tipi anlamına gelir.

Aşağıdaki örnekte, maksimum karakter sayısı 10 ve gerçek karakter sayısı 3 olan bir karakter dizisi gösterilmektedir. Bununla birlikte karakter dizisi, birer Bayt ile birlikte 3 karakter içerir, ancak her biri bir Bayt olan 10 karaktere kadar genişletilebilir.

Tablo 5-35 Bir String veri tipiyle ilgili örnek

Toplam Karakter sayısı	Akım Karakter sayısı	Karakter 1	Karakter 2	Karakter 3	...	Karakter 10
10	3	'C' (16#43)	'A' (16#41)	'T' (16#54)	...	-
Bayt 0	Bayt 1	Bayt 2	Bayt 3	Bayt 4	...	Bayt 11

Aşağıdaki örnekte, maksimum karakter sayısı 500 ve gerçek karakter sayısı 300 olan bir WString veri tipi gösterilmektedir. Bununla birlikte karakter dizisi, birer sözcükle birlikte 300 karakter içerir, ancak her biri birer sözcük olan 500 karaktere kadar genişletilebilir.

Tablo 5-36 Bir WString veri tipiyle ilgili örnek

Toplam Karakter sayısı	Akım Karakter sayısı	Karakter 1	Karakter 2 ila 299	Karakter 300	...	Karakter 500
500	300	'ä' (16#0084)	ASCII karakterli sözcükler	'M' (16#004D)	...	-
Sözcük 0	Sözcük 1	Sözcük 2	Sözcük 3 ila 300	Sözcük 301	...	Sözcük 501

ASCII kontrol karakterleri Char, WChar, String ve WString verilerinde kullanılabilir. Aşağıdaki tabloda, kontrol karakterlerinin sözdizimiyle ilgili örnekler gösterilmektedir.

Tablo 5-37 Geçerli ASCII kontrol karakterleri

Kontrol karakteri	ASCII onaltılık değeri (Char)	ASCII onaltılık değeri (WChar)	Kontrol fonksiyonu	Örnekler
$L veya $l	16#0A	16#000A	Satır sonu	'$LText', '$0AText'
$N veya $n	16#0A ve 16#0D	16#000A ve 16#000D	Satır kesmesi Yeni satır, karakter dizisindeki iki karakteri gösterir.	'$NText', '$0A$0DText'
$P veya $p	16#0C	16#000C	Sonraki sayfa	'$PText', '$0CText'
$R veya $r	16#0D	16#000D	Satır sonu (CR)	'$RText','$0DText'
$T veya $t	16#09	16#0009	Sekme	'$TText', '$09Text'
$$	16#24	16#0024	Dolar işareti	'100$$', '100$24'
$'	16#27	16#0027	Tek tırnak işareti	'$'Text$'','$27Text$27'

5.4.6 ARRAY veri tipi

Array'ler

Aynı veri tipine ait birden fazla eleman içeren bir Array oluşturabilirsiniz. Array'ler; OB, FC, FB ve DB'nin blok arayüzünde oluşturulabilir. PLC değişken editöründe Array oluşturamazsınız.

Bir Array'i blok arayüzünde oluşturmak için ilgili Array'i adlandırın ve "Array [lo .. hi] of type" veri tipini seçin, ardından "lo", "hi" ve "type" öğelerini aşağıdaki şekilde değiştirin:

- lo - Array'iniz için kullanılan başlangıç indeksi (en düşük indeks)
- hi - Array'iniz için kullanılan bitiş indeksi (en yüksek indeks)
- type - BOOL, SINT, UDINT gibi veri tiplerinden birinin tipi

Tablo 5-38 ARRAY veri tipiyle ilgili kurallar

Veri tipi	Array sözdizimi		
ARRAY	<Veritipi>nin adı [index1_min..index1_max, index2_min..index2_max] • Tüm Array elemanları aynı veri tipine sahip olmalıdır. • İndeks negatif olabilir, ancak alt limit üst limitten daha küçük veya ona eşit olmalıdır. • Array'ler maksimum altı boyutu kapsayabilir. • Çok boyutlu İndeks deklarasyonları, min..maks virgül karakteriyle birbirinden ayrılır. • İç içe geçmiş Array'ler veya Array'lerden oluşturulan Array'lere izin verilmez. • Bir Array'in bellek boyutu = (bir elemanın boyutu * Array'deki elemanların toplam sayısı)		
	Array indeksi	**Geçerli İndeks veri tipleri**	**Array indeksiyle ilgili kurallar**
	Sabit veya değişken	USInt, SInt, UInt, Int, UDInt, DInt	• Limitler: -32768 ila +32767 • Geçerli: Karışık sabitler ve değişkenler • Geçerli: Sabit ifadeler • Geçersiz: Değişken ifadeler

Örnek: Array deklarasyonları	ARRAY[1..20] of REAL	Bir boyut, 20 eleman
	ARRAY[-5..5] of INT	Bir boyut, 11 eleman
	ARRAY[1..2, 3..4] of CHAR	İki boyut, 4 eleman
Örnek: Array adresleri	ARRAY1[0]	ARRAY1 Eleman 0
	ARRAY2[1,2]	ARRAY2 Eleman [1,2]
	ARRAY3[i,j]	i = 3 ve j = 4 ise, ARRAY3 elemanı [3, 4] adreslenir

5.4.7 Struct veri tipi

"Struct" veri tipiyle, başka veri tiplerinden oluşan bir veri yapısı tanımlayabilirsiniz. "Struct" veri tipi, birbirine bağlı proses verilerinden oluşan bir grubu tek bir veri birimi olarak işlemenize olanak sağlar. Yapıları (Struct'lar) veri bloğu editöründe veya bir blok arayüzü editöründe oluşturabilirsiniz.

Array'leri ve yapıları daha büyük yapılar halinde birleştirebilirsiniz. Örneğin Array'ler içeren yapılardan bir yapı oluşturabilirsiniz. Bir yapı, maksimum sekiz seviye derinliğinde iç içe geçirebilir.

5.4.8 PLC veri tipi

PLC veri tipi editöründe, programınızda birden fazla kez kullanabileceğiniz veri yapıları tanımlayabilirsiniz. Proje gezgininde "PLC veri tiplerini" açarak veri tipi ve "Yeni veri tipi ekle" komutuna çift tıklayarak bir PLC veri tipi oluşturabilirsiniz. Standart adı değiştirmek için, yeni oluşturulan PLC veri tipine münferit olarak iki kez tıklayın. Ardından, PLC veri tipi editörünü açmak için iki kez tıklayın.

Kullanıcı tanımlı bir PLC veri tipi yapısı oluşturmak için veri bloğu editöründe olduğu gibi aynı yöntemi izleyebilirsiniz. İstenen veri yapısını oluşturmak için gereken tüm ek veri tipleri için yeni satırlar ekleyin.

Yeni bir PLC veri tipi oluşturduğunuzda, yeni PLC veri tipinin adı DB editöründeki ve kod bloğu arayüz editöründeki seçim açılır listesinde görüntülenir.

PLC veri tiplerini potansiyel olarak aşağıdaki şekilde kullanabilirsiniz:

- Bir kod bloğu arayüzünde veya veri bloklarında veri tipi olarak
- Aynı veri yapısına sahip birden fazla global veri bloğu oluşturmak için bir şablon olarak
- CPU'nun E ve A bellek alanlarında PLC değişken deklarasyonları için veri tipi olarak

Örneğin bir PLC veri tipi, renkleri karıştırmak için kullanılan bir reçete olabilir. Daha sonra, bu PLC veri tipini birden fazla veri bloğuna atayabilirsiniz. Belirli bir renk oluşturmak için her bir veri bloğundaki değişkenleri uyarlayabilirsiniz.

5.4.9 "Variant" işaretçi veri tipi

Variant veri tipi, çeşitli veri tiplerinin veya parametrelerin değişkenlerine referans verebilir. Variant işaretçisi, yapıları ve münferit yapı bileşenlerini gösterebilir. Variant işaretçisi, bellekte yer kaplamaz.

Tablo 5-39 Variant işaretçisinin özellikleri

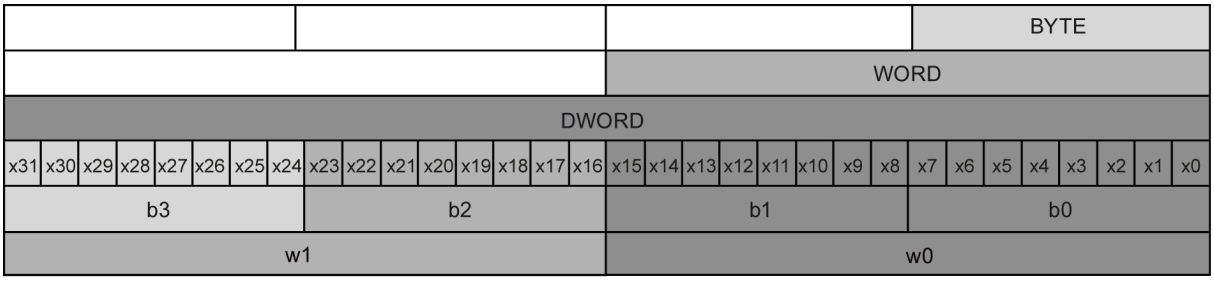
Uzunluk (Bayt)	Gösterim	Format	Örnek giriş
0	Sembolik	İşlemci	MyVariable
		DB adı.Yapı adı.Eleman adı	MeinDB.Strukt1.Druck1
	Mutlak	İşlemci	%MW10
		DB Numarası.İşlemci tipi Uzunluk	P#DB10.DBX10.0 INT 12

5.4.10 Bir değişken veri tipine ait bir "Slice"a erişim

PLC değişkenlerine ve veri bloğu değişkenlerine, bunların boyutuna bağlı olarak Bit, Bayt veya sözcük seviyesinde erişilebilir. Bir veri Slice'ına erişim için kullanılan sözdizimi aşağıdaki gibidir:

- "<PLC değişken adı>".xn (Bit erişimi)
- "<PLC değişken adı>".bn (Bayt erişimi)
- "<PLC değişken adı>".wn (sözcük erişimi)
- "<Veri bloğu adı>".<Değişken adı>.xn (Bit erişimi)
- "<Veri bloğu adı>".<Değişken adı>.bn (Bayt erişimi)
- "<Veri bloğu adı>".<Değişken adı>.wn (Sözcük erişimi)

Bir veri çift sözcüğü boyutundaki bir değişkene 0 - 31 Bit'leri, 0 - 3 Bayt'ları veya 0 - 1 sözcükleri aracılığıyla erişilebilir. Bir sözcük boyutundaki bir değişkene 0 - 15 Bit'leri, 0 - 1 Bayt'ları veya 0 sözcüğü aracılığıyla erişilebilir. Bir sözcük boyutundaki bir değişkene 0 - 7 Bit'leri, veya 0 Bayt'ı aracılığıyla erişilebilir. Bit, Bayt ve sözcük Slice'ları, Bit'lerin, Bayt'ların veya sözcüklerin beklenen işlemciler olduğu her yerde kullanılabilir.

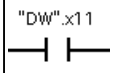
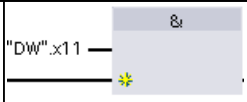
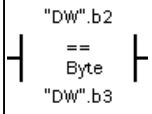
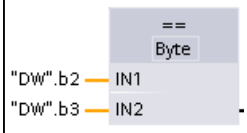
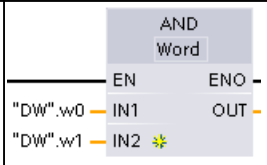
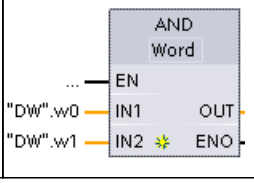


NOT

Slice'lar aracılığıyla şu veri tiplerine erişilebilir: Byte, Char, Conn_Any, Date, DInt, DWord, Event_Any, Event_Att, Hw_Any, Hw_Device, HW_Interface, Hw_Io, Hw_Pwm, Hw_SubModule, Int, OB_Any, OB_Att, OB_Cyclic, OB_Delay, OB_WHINT, OB_PCYCLE, OB_STARTUP, OB_TIMEERROR, OB_Tod, Port, Rtm, SInt, Time, Time_Of_Day, UDInt, UInt, USInt ve Word. Real veri tipindeki PLC değişkenlerine Slice'lar aracılığıyla erişilebilir, ancak Real tipindeki veri bloğu değişkenlerine erişilemez.

Örnekler

PLC değişken tablosundaki "DW", DWORD tipinde bir bildirilmiş değişkendir. Örneklerde Bit, Bayt ve sözcük Slice'ları formunda erişim gösterilmektedir:

	LAD	FBD	SCL
Bit erişimi			**IF "DW".x11 THEN ... END_IF;**
Bayt erişimi			**IF "DW".b2 = "DW".b3 THEN ... END_IF;**
Sözcük erişimi			**out:= "DW".w0 AND "DW".w1;**

5.4.11 Bir AT bindirmesi ile bir değişkene erişim

AT değişken bindirmesi yardımıyla, bir bloğun önceden deklare edilmiş bir değişkenine başka bir veri tipindeki bindirilmiş bir deklarasyonla erişebilirsiniz. Örneğin Bayt, Word veya DWord veri tipindeki bir değişkenin münferit Bit'lerini bir Bool Array'i ile adresleyebilirsiniz. AT bindirmeleri aşağıdaki değişken tipleri için kullanıma sunulur:

- Bir standart erişim bloğundaki değişkenler
- Optimize edilmiş bir bloktaki kalıcı değişkenler

Deklarasyon

Bir parametreyi bindirmek için, bindirilecek parametreden hemen sonra ek bir parametre deklare edin ve "AT" veri tipini seçin. Editör, bindirmeyi oluşturur ve ardından bindirme için veri tipini, yapıyı veya Array'i seçebilirsiniz.

Örnek

Bu örnekte, standart erişime sahip bir FB'nin giriş parametreleri gösterilmektedir. B1 Bayt değişkenine bir Boole Array'i bindirilmiştir:

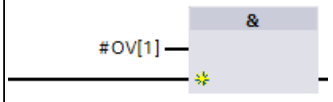
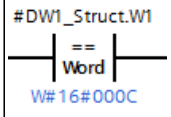
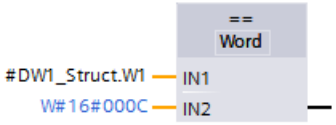
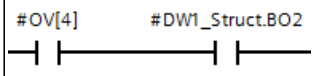
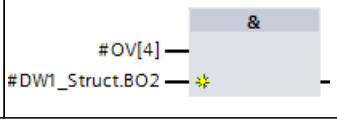
	B1		Byte	0.0
	OV	AT "B1"	Array[0..7] of Bool	0.0
	OV[0]		Bool	0.0
	OV[1]		Bool	0.1
	OV[2]		Bool	0.2
	OV[3]		Bool	0.3
	OV[4]		Bool	0.4
	OV[5]		Bool	0.5
	OV[6]		Bool	0.6
	OV[7]		Bool	0.7

Diğer bir örnek ise, Struct veri tipi bindirilen DWord tipinde bir değişkendir. Struct bir sözcük, bir Bayt ve iki Boole değeri içermektedir:

	DW1		DWord	2.0
	DW1_Struct	AT "DW1"	Struct	2.0
	W1		Word	0.0
	B1		Byte	2.0
	BO1		Bool	3.0
	BO2		Bool	3.1

Blok arayüzünün Ofset sütunu, bindirilen veri tiplerinin orijinal değişkenlere göre konumunu gösterir.

Bindirme tipleri program lojiği içinde doğrudan adreslenebilir:

LAD	FBD	SCL
		**IF #OV[1] THEN** **...** **END_IF;**
		**IF #DW1_Struct.W1 = W#16#000C THEN** **...** **END_IF;**
		**out1 := #DW1_Struct.B1;**
		**IF #OV[4] AND #DW1_Struct.BO2 THEN** **...** **END_IF;**

Kurallar

- Standart erişime sahip FB ve FC bloklarında (optimize edilmiş erişim yok), değişkenlerin bindirilmesi mümkündür.
- Optimize edilmiş FB ve FC bloklarında, kalıcı değişkenlerin üst üste binmesi mümkündür.
- Tüm blok tipleri ve tüm deklarasyon alt bölümleri için parametreleri bindirebilirsiniz.
- Bindirilmiş bir parametre, diğer her blok parametresi gibi kullanılabilir.
- VARIANT tipi parametreleri bindiremezsiniz.
- Bindirilen parametrenin boyutu, üzerine bindirilen parametrenin boyutundan küçük veya ona eşit olmalıdır.
- Bindirilen değişken, bindirildiği değişkenden hemen sonra deklare edilmeli ve çıkış veri tipi olarak "AT" anahtar sözcüğü aracılığıyla işaretlenmiş olmalıdır.

5.5 Memory card'ın kullanılması

NOT

CPU sadece SIMATIC memory card'ları (Sayfa 1343) destekler.

Programları memory card'a kopyalamadan önce, önceden kaydedilmiş tüm programları memory card'dan silin.

Bir memory card'ı aşağıdaki şekilde kullanabilirsiniz:

- Memory card'ı aktarım kartı veya program kartı olarak kullanabilirsiniz. Aktarım kartları ve program kartları kod bloklarını, veri bloklarını, teknoloji nesnelerini ve cihaz konfigürasyonunu içerir. Aktarım kartları ve program kartlarında örneğin Force tabloları, izleme tabloları veya PLC değişken tabloları **bulunmaz**.
  - Bir aktarım kartı (Sayfa 115) ile bir programı, bu işlem için STEP 7'yi kullanmadan CPU'nun dahili yükleme belleğine kopyalayabilirsiniz.
  - Boş bir aktarım kartıyla, şifreyi kaybettiyseniz veya unuttuysanız (Sayfa 126) şifre korumalı bir CPU'ya erişebilirsiniz.
  - Bir program kartını (Sayfa 118) CPU için harici yükleme belleği olarak kullanabilirsiniz.
- Bir memory card'ı Firmware güncellemelerini (Sayfa 124) indirmek için de kullanabilirsiniz.
- Bir memory card'ın gizli PLC konfigürasyon verilerini korumak için kullanılan şifrenin (Sayfa 122) belirlenmesi veya değiştirilmesi için kullanılması.

5.5.1 Memory card'ın CPU'ya takılması

DİKKAT

Memory card'ın ve yuvanın elektrostatik deşarja karşı korunması

Elektrostatik deşarjlar, memory card'ın veya CPU'da bu kart için öngörülen yuvanın hasar görmesine neden olabilir.

Memory card ile çalışırken, iletken, topraklı bir zeminin üzerinde durmanız ve topraklı bir bileklik takmanız gerekir. Memory card iletken bir kapta saklanmalıdır.



Memory card'ın yazma korumalı olmadığından emin olun. Bunun için, koruma sürgüsünü kilitleme pozisyonundan dışarıya doğru itin.

CPU'ya yazma korumalı bir memory card takarsanız, STEP 7, bir sonraki yol alma sırasında bu durum hakkında sizi bilgilendirmek için bir teşhis mesajı görüntüler. CPU yine de hatasız açılır, ancak reçeteler veya veri protokolleri ile birlikte yordamlar, yazma korumalı bir kartta hata mesajlarına neden olur.

⚠ İKAZ

Memory card'ı takmadan önce CPU'nun hiçbir proses yürütmediğinden emin olun.

Çalışan bir CPU'ya herhangi bir memory card takarsanız, CPU derhal STOP işletim durumuna geçer, bu da prosesin kesintiye uğramasına ve dolayısıyla ölüme veya ağır yaralanmalara neden olabilir.

Bir memory card'ı takmadan veya çıkarmadan önce, CPU'nun bir makineyi veya prosesi aktif olarak kontrol etmediğinden emin olun. Uygulamanız veya prosesiniz için bir ACİL KAPAT anahtarlama devresi takın.

NOT

S7-1200 CPU'lar V4.x'e bir V3.0 sürümü program içeren bir aktarım kartı takmayın.

3.0 sürümü bir program içeren aktarım kartları 4.x sürümü S7-1200 CPU'larıyla uyumlu değildir. V3.0 sürümü bir program içeren bir memory card takarsanız bu durum bir CPU hatasına yol açar.

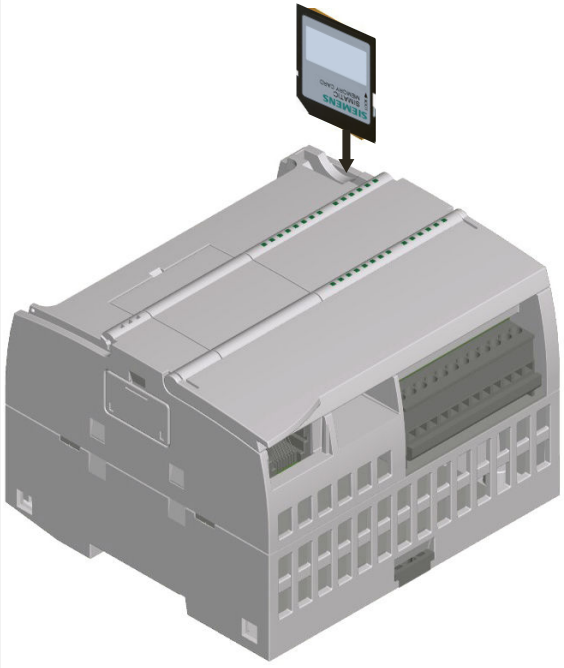
Geçersiz bir program sürümü içeren bir aktarım kartı (Sayfa 115) takarsanız, kartı çıkartın ve CPU'yu STOP işletim durumundan RUN işletim durumuna getirin, bir tam silme işlemi (MRES) gerçekleştirin veya CPU'yu kapatın ve yeniden açın. CPU'nun hata durumunu giderdikten sonra V4.x sürümünde geçerli bir CPU programı yükleyebilirsiniz.

Bir V3.0 sürümü programı bir V4.x sürümü programa aktarmak için, TIA Portal'ı kullanmanız ve donanım konfigürasyonunda cihazı değiştirmeniz gerekir.

NOT

Bir memory card'ı CPU'nun STOP işletim durumunda takarsanız, teşhis tamponunda, memory card'ı değerlendirme işleminin başladığını belirten bir mesaj görüntülenir. CPU'yu RUN işletim durumuna getirirseniz, CPU'nun belleğinde MRES üzerinden tam silme işlemi gerçekleştirirseniz veya CPU'yu kapatır ve yeniden açarsanız CPU memory card'ı değerlendirir.

Tablo 5-40 Memory card'ın yerleştirilmesi

	Bir memory card'ı yerleştirmek için CPU'nun üst kapağını açın ve memory card'ı yuvaya takın. Bir soket bağlayıcı, modülün kolayca takılmasını ve çıkartılmasını sağlar. Memory card, yuvaya sadece bir yönde takılabilecek şekilde tasarlanmıştır.

Bir memory card takılırken CPU davranışı

CPU'ya bir memory card takarken, CPU, aşağıdaki adımları yürütür:

1. STOP işletim durumuna geçme (STOP işletim durumunda değilse)
2. Aşağıdaki seçeneklerle birlikte bir giriş talebi görüntüleme:
   - Kapatma ve yeniden açma
   - RUN işletim durumuna geçme
   - Tam silme işlemi gerçekleştirme
3. Kartı değerlendirme

Memory card'ın CPU tarafından değerlendirilmesi

CPU'nun Cihaz konfigürasyonunun koruma özellikleri (Sayfa 159) kısmında "Dahili yükleme belleğinden harici yükleme belleğine kopyalamayı devre dışı bırak" ayarını seçmezseniz, CPU, ne tip bir memory card taktığınızı belirler:

- **Boş memory card**: Boş bir memory card hiçbir görev dosyası (S7_JOB.S7S) içermez. Boş bir memory card takarsanız, CPU, bir program görev dosyası ekler. Ardından dahili yükleme belleğini harici yükleme belleğine (memory card'daki program dosyası) kopyalar ve dahili yükleme belleğini siler.
- **Boş program kartı**: Boş bir program kartı, boş bir program görev dosyası içerir. Bu durumda CPU, dahili yükleme belleğini harici yükleme belleğine (memory card'daki program dosyası) kopyalar ve dahili yükleme belleğini siler.

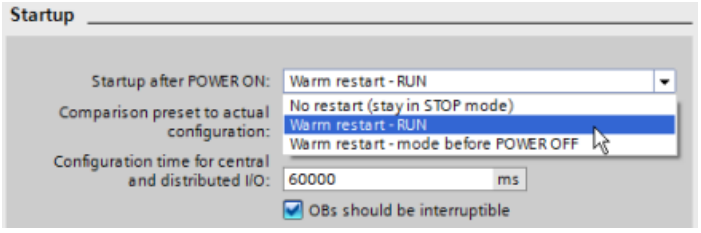
CPU'nun cihaz konfigürasyonunun koruma özellikleri kısmında "Dahili yükleme belleğinden harici yükleme belleğine kopyalamayı devre dışı bırak" ayarını seçtiyseniz, CPU, aşağıdaki şekilde davranır:

- **Boş memory card**: Boş bir memory card hiçbir görev dosyası (S7_JOB.S7S) içermez. Boş bir memory card takarsanız, CPU hiçbir işlem gerçekleştirmez. Bir program görev dosyası oluşturmaz ve dahili yükleme belleğini harici yükleme belleğine (memory card'daki program dosyası) kopyalamaz. Dahili yükleme belleğini silmez.
- **Boş program kartı**: Boş bir program kartı, boş bir program görev dosyası içerir. Bu durumda CPU hiçbir eylem gerçekleştirmez. Dahili yükleme belleğini harici yükleme belleğine (memory card'daki program dosyası) kopyalamaz. Dahili yükleme belleğini silmez.

CPU'ya bir program kartı (Sayfa 118), aktarım kartı (Sayfa 115) veya Firmware güncellemesi içeren bir kart (Sayfa 124) takarsanız, "Dahili yükleme belleğinden harici yükleme belleğine kopyalamayı devre dışı bırak" konfigürasyon ayarı, CPU'nun memory card'ı nasıl değerlendireceği üzerinde etkili olmaz.

5.5.2 Projenin memory card'a kopyalanmadan önce, CPU'nun yol alma parametrelerinin yapılandırılması

Bir programı bir aktarım kartına veya bir program kartına kopyalarsanız, program, CPU için kullanılan yol alma parametrelerini içerir. Programı memory card'a kopyalamadan önce, kapatma ve yeniden açma işleminden sonra CPU'nun işletim durumunu yapılandırmış olduğunuzdan her zaman emin olun. CPU'nun STOP işletim durumunda mı, RUN işletim durumunda mı, yoksa en son işletim durumunda mı (yeniden başlatmadan önce) başlatılması gerektiğini seçebilirsiniz.



5.5.3 Aktarım kartı

DİKKAT
Memory card'ın ve yuvanın elektrostatik deşarja karşı korunması Elektrostatik deşarjlar, memory card'ın veya CPU'da bu kart için öngörülen yuvanın hasar görmesine neden olabilir. Memory card ile güvenli bir şekilde çalışmak için, aşağıdaki koşullardan birinin veya her ikisinin de sağlandığından emin olun: • Topraklı, iletken bir zeminin üzerinde durduğunuzdan emin olun. • Memory card ile çalışırken topraklı bir bileklik takın. Memory card iletken bir kapta saklanmalıdır.

Aktarım kartının oluşturulması

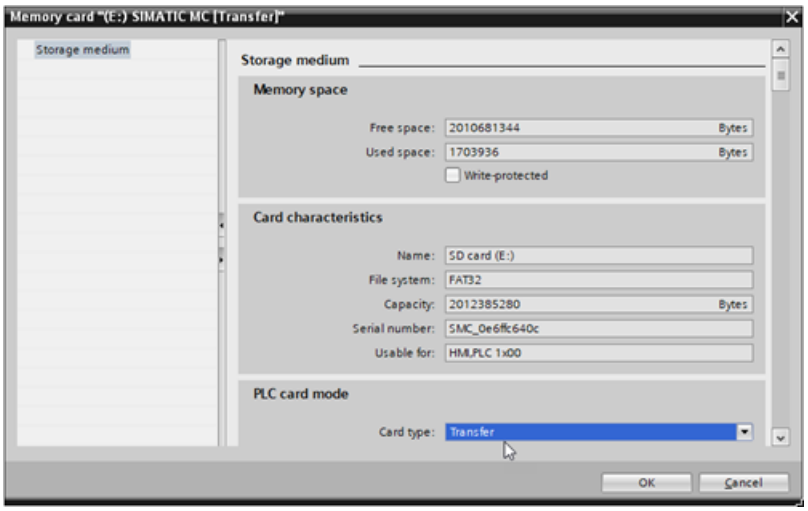
Bir programı aktarım kartına kopyalamadan önce, CPU'nun yol alma parametrelerini yapılandırmayı (Sayfa 115) unutmayın. Bir aktarım kartı oluşturmak için aşağıdaki yöntemi izleyin:

1. Bilgisayarınıza bağlı bir SD kart okuyucuya yazma korumalı olmayan boş bir SIMATIC memory card takın. (Memory card yazma korumalı ise, koruma sürgüsünü kilitleme pozisyonundan dışarıya doğru itin.)
   Halen bir kullanıcı programı, veri protokolleri, reçeteler veya bir Firmware güncellemesi içeren bir SIMATIC memory card kullanıyorsanız, kartı yeniden kullanmak için dosyaları silmeniz **gerekir**. Windows gezginini çağırın ve memory card'ın içeriklerini görüntüleyin. "S7_JOB.S7S" dosyasını ve mevcut tüm klasörleri ("SIMATIC.S7S", "FWUPDATE.S7S", "DataLogs" ve "Recipes" gibi) silin.

DİKKAT
"__LOG__" ve "crdinfo.bin" gizli dosyalarını memory card'dan SİLMEYİN. Memory card, "__LOG__" ve "crdinfo.bin" dosyalarına ihtiyaç duyar. Bu dosyaları silerseniz, memory card'ı artık CPU ile kullanamazsınız.

2. Proje gezgininde (Proje görünümü) "Kart okuyucu/USB bellek" klasörünü genişletin ve kart okuyucunuzu seçin.
3. Farenin sağ tuşuyla memory card'ın sürücü harflerine tıklayın ve bağlam menüsünden "Özellikler" öğesini seçin.

4. "Memory card" iletişim kutusundaki "Kart tipi" açılır listesinden "Aktarım" öğesini seçin. Bunun üzerine STEP 7, boş aktarım kartını oluşturur. CPU şifrenizi kaybettiğiniz (Sayfa 126) için boş bir aktarım kartı oluşturursanız, aktarım kartını kart okuyucudan çıkartın.



5. Proje gezgininde CPU'yu (örn. PLC_1 [CPU 1214C DC/DC/DC]) seçerek ve fareyle memory card'ın üzerine sürükleyerek programı ekleyin. Alternatif olarak CPU'yu kopyalayabilir ve memory card'a yapıştırabilirsiniz. CPU'nun memory card'a kopyalanmasıyla "Önizlemeyi yükle" iletişim kutusu açılır.
6. CPU'yu memory card'a kopyalamak için "Önizlemeyi yükle" iletişim kutusundaki "Yükle" butonuna tıklayın.
7. İletişim kutusunda, CPU'nun (program) hatasız bir şekilde yüklendiğini belirten bir mesaj görüntülenirse, "Tamamla" butonuna tıklayın.

Aktarım kartının kullanılması

⚠ İKAZ

Memory card'ı takmadan önce CPU'nun hiçbir proses yürütmediğinden emin olun.

CPU, bir memory card'ın takılması ile STOP işletim durumuna geçer, bu da bir çevrimiçi prosesin veya bir makinenin çalıştırılmasını etkileyebilir. Bir prosesin veya makinenin öngörülemeyen şekilde çalıştırılması, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Bir aktarım kartını takmadan önce, CPU'nun STOP işletim durumunda olduğundan ve prosesin güvenli bir durumda bulunduğundan emin olun.

NOT

Daha yüksek sürümdeki CPU varyantlarına V3.0 sürümü bir program içeren bir aktarım kartı takmayın.

3.0 sürümü aktarım kartları, sürümü daha yüksek olan S7-1200 CPU'larıyla uyumlu değildir. V3.0 sürümü bir program içeren bir memory card takarsanız bu durum bir CPU hatasına yol açar.

Geçersiz bir program sürümü içeren bir aktarım kartı takılırsa kartı çıkartın. Hatayı silmek için CPU'yu STOP işletim durumundan RUN işletim durumuna getirin, bir tam silme işlemi (MRES) gerçekleştirin veya CPU'yu kapatın ve yeniden açın. Hata silindikten sonra geçerli bir CPU programı yükleyin.

Programı CPU'ya aktarmak için aşağıdaki yöntemi izleyin:

1. Aktarım kartını CPU'ya (Sayfa 112) takın. CPU, RUN işletim durumunda ise STOP işletim durumuna geçer. Memory card'ın değerlendirilmesinin gerektiğini belirtmek için bakım LED'i (MAINT) yanıp söner. Mevcut program şu anda hâlâ CPU'da bulunmaktadır.
2. Memory card'ı değerlendirmek için CPU'yu kapatın ve yeniden açın. CPU'yu yeniden başlatmak için kullanılabilecek alternatif yöntemler, STOP işletim durumundan RUN işletim durumuna geçmek veya STEP 7'de bir tam silme işlemi (MRES) gerçekleştirmektir.
3. CPU, yeniden başlatma işleminden sonra memory card'ı değerlendirir ve programı CPU'nun dahili yükleme belleğine kopyalar.

   Programın kopyalandığını belirtmek için RUN/STOP LED'i sırayla yeşil ve sarı olarak yanıp söner. RUN/STOP LED'i sürekli olarak sarı yanıyorsa ve MAINT LED'i sarı renkte yanıp sönüyorsa, kopyalama işlemi tamamlanmıştır ve memory card'ı çıkartabilirsiniz.
4. Dahili yükleme belleğine aktarılan yeni programı değerlendirmek için CPU'yu yeniden başlatın (açarak veya yeniden başlatma için kullanılan alternatif yöntemlerden biri aracılığıyla).

Ardından CPU, yol alma için kullanılan ve proje için ayarladığınız işletim durumuna (RUN veya STOP) geçer.

NOT

CPU'yu RUN işletim durumuna getirmeden önce aktarım kartını çıkartın.

5.5.4 Program kartı

DİKKAT
Elektrostatik deşarjlar, memory card'ın veya CPU'da bu kart için öngörülen yuvanın hasar görmesine neden olabilir. Memory card ile çalışırken, iletken, topraklı bir zeminin üzerinde durmanız ve topraklı bir bileklik takmanız gerekir. Memory card iletken bir kapta saklanmalıdır.



Memory card'ın yazma korumalı olmadığından emin olun. Bunun için, koruma sürgüsünü kilitleme pozisyonundan dışarıya doğru itin.

Program elemanlarını program kartına kopyalamadan önce, daha önceden kaydedilmiş tüm programları memory card'dan silin.

Program kartının oluşturulması

Memory card, program kartı olarak kullanılırken CPU'nun harici yükleme belleği olarak görev görür. Program kartını çıkartırken CPU'nun dahili yükleme belleği boştur.

NOT

CPU'ya boş bir memory card takarsanız, aşağıdaki işlemlerden birini gerçekleştirdiğiniz takdirde CPU, programı ve Force değerlerini dahili bellekten memory card'a kopyalar:

- CPU'nun kapatılması ve yeniden açılması
- İşletim durumunun STOP'tan RUN'a getirilmesi
- Tam silme işleminin (MRES) gerçekleştirilmesi

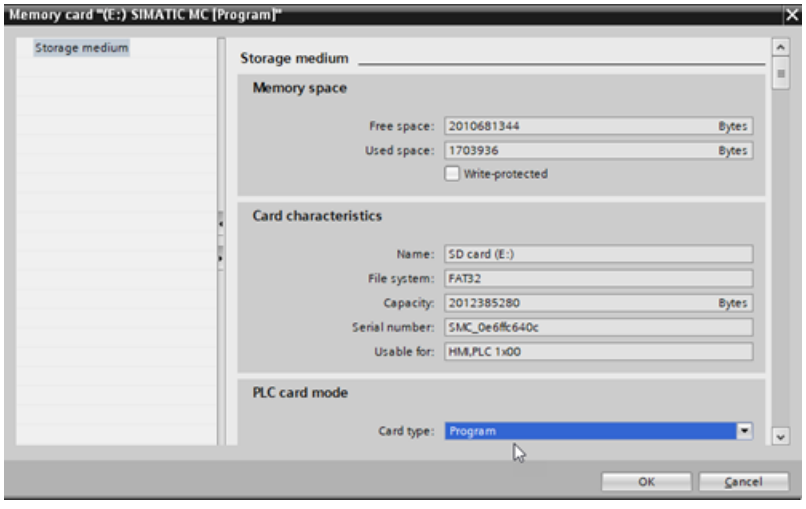
Kopyalama işlemi tamamlandıktan sonra, dahili yükleme belleğindeki program silinir ve CPU, yapılandırılan yol alma türüne geçer.

Bir projeyi program kartına kopyalamadan önce, CPU'nun yol alma parametrelerini yapılandırmayı (Sayfa 115) unutmayın. Bir program kartı oluşturmak için aşağıdaki yöntemi izleyin:

1. Bilgisayarınıza bağlı bir SD kart okuyucuya yazma korumalı olmayan boş bir SIMATIC memory card takın. (Memory card yazma korumalı ise, koruma sürgüsünü kilitleme pozisyonundan dışarıya doğru itin.)
   Halen bir kullanıcı programı, veri protokolleri, reçeteler veya bir Firmware güncellemesi içeren bir SIMATIC memory card kullanıyorsanız, kartı yeniden kullanmak için dosyaları silmeniz **gerekir**. Windows gezgini ile memory card'ın içeriğini görüntüleyebilir ve gerekirse aşağıdaki dosyaları ve klasörleri silebilirsiniz:
   - S7_JOB.S7S
   - SIMATIC.S7S
   - FWUPDATE.S7S
   - DataLogs
   - Recipes
   - UserFiles

DİKKAT
"__LOG__" ve "crdinfo.bin" gizli dosyalarını memory card'dan SİLMEYİN. Memory card, "__LOG__" ve "crdinfo.bin" dosyalarına ihtiyaç duyar. Bu dosyaları silerseniz, memory card'ı artık CPU ile kullanamazsınız.

2. Proje gezgininde (Proje görünümü) "Kart okuyucu/USB bellek" klasörünü genişletin ve kart okuyucuyu seçin.
3. Farenin sağ tuşuyla kart okuyucudaki memory card'ın sürücü harflerine tıklayın ve bağlam menüsünden "Özellikler" öğesini seçin. Bunun üzerine "Memory card" iletişim kutusu görüntülenir.
4. "Memory card" iletişim kutusundaki bağlam menüsünde "Program" öğesini seçin.



5. Proje gezgininde CPU'yu (örn. PLC_1 [CPU 1214C DC/DC/DC]) seçerek ve fareyle memory card'ın üzerine sürükleyerek programı ekleyin. Alternatif olarak CPU'yu kopyalayabilir ve memory card'a yapıştırabilirsiniz. CPU'nun memory card'a kopyalanmasıyla "Önizlemeyi yükle" iletişim kutusu açılır.

6. CPU'yu memory card'a kopyalamak için "Önizlemeyi yükle" iletişim kutusundaki "Yükle" butonuna tıklayın.
7. İletişim kutusunda, yükleme işleminin hatasız bir şekilde tamamlandığını bildiren bir mesaj görüntülenirse, "Tamamla" butonuna tıklayın.

Program kartının bir CPU için harici yükleme belleği olarak kullanılması

⚠ İKAZ

Bir program kartı takılırken ortaya çıkan riskler

CPU, RUN işletim durumundayken bir program kartını takarken dikkatli bir şekilde hareket edin. Cihazların beklenmedik şekilde çalıştırılması ölüme, yaralanmalara ve cihaz hasarlarına neden olabilir.

CPU'nun RUN işletim durumunda bir program kartı takılırsa, CPU, STOP işletim durumuna geçer.

Bir program kartını takmadan önce CPU'nun STOP işletim durumunda bulunduğundan emin olun.

Bir CPU ile bir program kartını kullanmak için aşağıdaki yöntemi izleyin:

1. Program kartını CPU'ya takın. CPU RUN işletim durumundaysa STOP işletim durumuna geçer. Memory card'ın değerlendirilmesinin gerektiğini belirtmek için bakım LED'i (MAINT) yanıp söner.
2. Memory card'ı değerlendirmek için CPU'yu kapatın ve yeniden açın. CPU'yu yeniden başlatmak için kullanılabilecek alternatif yöntemler, STOP işletim durumundan RUN işletim durumuna geçmek veya STEP 7'de bir tam silme işlemi (MRES) gerçekleştirmektir.
3. CPU yeniden başlatıldıktan ve program kartı değerlendirildikten sonra CPU, CPU'nun dahili yükleme belleğini siler.

Ardından CPU, yol alma için kullanılan ve CPU için ayarladığınız işletim durumuna (RUN veya STOP) geçer.

Program kartı, CPU'da takılı durumda kalmalıdır.

⚠ İKAZ

Bir program kartı çıkarılırken ortaya çıkan riskler

CPU, RUN işletim durumundayken bir program kartını çıkarırken dikkatli bir şekilde hareket edin. Cihazların beklenmedik şekilde çalıştırılması ölüme, yaralanmalara ve cihaz hasarlarına neden olabilir.

CPU'nun RUN işletim durumunda bir program kartı çıkarılırsa, CPU, STOP işletim durumuna geçer ve hata LED'i yanıp söner.

Program kartının çıkarılmasıyla program CPU'dan kaldırılır.

Bir SIMATIC memory card'ın kullanım ömrü

Bir SIMATIC memory card'ın kullanım ömrü aşağıdaki faktörlere bağlıdır:

- Bellek bloğu başına silme ve yazma işlemlerinin sayısı
- Yazılan Bayt sayısı
- Ortam sıcaklığı gibi harici etkiler

NOT

Yazma ve silme işlemlerinin SIMATIC memory card'ın kullanım ömrü üzerindeki etkileri

Yazma ve silme işlemleri, özellikle tekrarlanan (periyodik) yazma/silme işlemleri, SIMATIC memory card'ın kullanım ömrünü kısaltır.

Aşağıdaki işlemlerin periyodik olarak uygulanması, yazma işlemlerinin ve verilerin sayısına bağlı olarak memory card'ın kullanım ömrünü kısaltır:

- Değişken arşivi işlemleri (örn. DataLogWrite)
- Reçete işlemleri (örn. RecipeExport)
- Dosya sisteminde yazma/silme işlemleri gerçekleştiren (örn. WRIT_DBL, CREATE) sistem fonksiyonunun çağrılması eylemleri (SFC'ler)
- Dosya sisteminde yazma/silme işlemleri gerçekleştiren (örn. FileWriteC, FileDelete) sistem fonksiyonu blokları (SFB'ler)
- Kalıcı bellekte verileri değiştiren (örn. Tracing, SET-TimeZone) diğer tüm periyodik eylemler

5.5.5 Memory card'ın, gizli PLC konfigürasyon verilerini korumak için kullanılması

Bir SIMATIC memory card'ı,Gizli PLC konfigürasyon verilerini koruma şifresini belirlemek veya değiştirmek için kullanabilirsiniz.

DİKKAT
Memory card'ın ve yuvanın elektrostatik deşarja karşı korunması Elektrostatik deşarjlar, memory card'ın veya CPU'da bu kart için öngörülen yuvanın hasar görmesine neden olabilir. Memory card ile çalışırken, iletken, topraklı bir zeminin üzerinde durmanız ve topraklı bir bileklik takmanız gerekir. Memory card iletken bir kapta saklanmalıdır.

DİKKAT
Memory card'ı yeniden biçimlendirmek için Windows biçimlendirme programını veya başka biçimlendirme programlarını kullanmayın. Bir SIMATIC memory card Windows biçimlendirme programıyla yeniden biçimlendirilirse, bu memory card artık bir S7-1200 CPU'su tarafından kullanılamaz.

Devre dışı bırakmayla bağlantılı riskler

S7-1200 CPU'ları, memory card'ın ve dahili Flash belleğin güvenli şekilde silinmesini desteklemez. Bu nedenle, özel ve gizli bilgilerin kaybolmasını önlemek için, devre dışı bırakma işlemi sırasında CPU ve memory card'ı güvenli bir şekilde tasfiye etmeniz gerekir.

Bunun için gizli PLC konfigürasyon verilerini koruma şifresine sahip bir memory card oluşturabilirsiniz

Bu şifreye sahip memory card'ı oluşturmak için aşağıdaki yöntemi izleyin:

1. Yazma koruması olmayan boş bir SIMATIC memory card'ı, bilgisayarınıza bağlı bir SD kart okuyucuya takın. Kart yazma korumalı ise, koruma sürgüsünü kilitleme pozisyonundan dışarıya doğru itin.
   Halihazırda bir kullanıcı programı veya bir Firmware güncellemesi içeren bir SIMATIC memory card kullanabilirsiniz, ancak memory card'daki dosyalardan bazılarını silmeniz gerekir. Bir memory card'ı yeniden kullanmak için, Gizli PLC konfigürasyon verilerinin koruma dosyasını oluşturmadan önce "S7_JOB.S7S" dosyasını silmeniz gerekir. Windows gezgini ile memory card'ın içeriğini görüntüleyebilir ve "S7_JOB.S7S" dosyasını ve klasörleri silebilirsiniz.

DİKKAT
"__LOG__" ve "crdinfo.bin" gizli dosyalarını memory card'dan SİLMEYİN. Memory card, "__LOG__" ve "crdinfo.bin" dosyalarına ihtiyaç duyar. Bu dosyaları silerseniz, memory card'ı artık CPU ile kullanamazsınız.

2. Memory card'ın kök dizininde "S7_JOB.S7S" adında bir dosya oluşturun. Dosyayı bir metin editörüyle açın ve şu ifadeyi girin: SET_PWD.
3. Memory card'ın kök dizininde şu ada sahip bir klasör oluşturun: SET_PWD.S7S.
4. "SET_PWD.S7S" klasöründe "PWD.TXT" adında bir metin dosyası oluşturun. Dosyanın adı "PWD.TXT" olmalıdır. Dosyanın metin içeriği olarak gizli PLC konfigürasyon verilerini koruma şifrenizi girin. Dosya, gizli PLC konfigürasyon verilerini koruma şifresinin bulunduğu tek bir metin satırı içermelidir. Şifreyi oluştururken şifrelerle ilgili STEP 7 kurallarını uygulayın ve şu karakterleri kullanın:
   - 0123456789
   - A...Z a...z
   - !#$%&()*+,-./:;<=>?@ [\]_{|}~^
5. PLC'de gizli PLC konfigürasyon verilerini koruma şifresini sıfırlamak için dosya boş olmalıdır.
6. Kartı dikkatli bir şekilde okuma cihazından çıkarın.

Gizli PLC konfigürasyon verilerini koruma şifresini aşağıdaki şekilde belirleyebilirsiniz

Gizli PLC konfigürasyon verilerini koruma şifresini belirlemek için aşağıdaki yöntemi izleyin:

1. Gizli PLC konfigürasyon verilerini koruma şifresini belirlemeden önce CPU'nun hiçbir proses yürütmediğinden emin olun.
2. Memory card'ı CPU'nun içine itin. CPU RUN işletim durumundaysa STOP işletim durumuna geçer. Memory card'ın değerlendirilmesinin gerektiğini belirtmek için bakım LED'i (MAINT) yanıp söner.

3. İşlemi başlatmak için CPU'yu yeniden başlatın. CPU yeniden başlatıldıktan sonra gizli PLC konfigürasyon verilerini koruma şifresi belirlenir. RUN/STOP LED'i sürekli olarak sarı renkte yanıyorsa ve MAINT LED'i yanıp sönüyorsa işlem tamamlanmıştır. Ardından memory card'ı çıkarmanız gerekir.
4. Yeni Gizli PLC konfigürasyon verilerini koruma şifresini kullanmak için, memory card'ı çıkardıktan sonra CPU'yu bir kez daha yeniden başlatın.

Mevcut kullanıcı programı Gizli PLC konfigürasyon verilerinin korunması için başka bir şifreye ihtiyaç duyuyorsa yeniden başlatma işleminden sonra kullanıcı programı yüklenmez. Mevcut programı silmeniz ve önceki adımlarda belirlediğiniz Gizli PLC konfigürasyon verilerini koruma şifresini kullanan bir programı hedef sisteme yüklemeniz gerekir.

Mevcut program belirtilen gizli PLC konfigürasyon verilerini koruma şifresine ihtiyaç duyuyorsa, PLC, proje konfigürasyonuna uygun şekilde RUN işletim durumuna geçebilir.

5.5.6 Firmware güncellemesi

Bir SIMATIC memory card ile bir Firmware güncellemesi de gerçekleştirebilirsiniz.

DİKKAT
Memory card'ın ve yuvanın elektrostatik deşarja karşı korunması Elektrostatik deşarjlar, memory card'ın veya CPU'da bu kart için öngörülen yuvanın hasar görmesine neden olabilir. Memory card ile çalışırken, iletken, topraklı bir zeminin üzerinde durmanız ve topraklı bir bileklik takmanız gerekir. Memory card iletken bir kapta saklanmalıdır.

SIMATIC memory card'ı, Firmware güncellemelerini Siemens Industry Online-Support (http://support.industry.siemens.com) web sitesinden indirmek için de kullanabilirsiniz. Bu web sitesindeki "Downloads" bölümüne gidin. Burada, güncellemek istediğiniz özel modül tipini aratın.

Alternatif olarak S7-1200 Downloads (https://support.industry.siemens.com/cs/ww/en/ps/13683/dl) web sitesini doğrudan çağırabilirsiniz.

NOT

Bir S7-1200 CPU V3.0'ı veya önceki bir sürümünü S7-1200 V4.0 Firmware sürümüne (veya daha yeni) yükseltemezsiniz.

Bir Firmware güncellemesini aşağıdaki yöntemlerden birini kullanarak da gerçekleştirebilirsiniz:

- STEP 7'nin çevrimiçi ve teşhis araçları (Sayfa 1120) ile
- Web sunucusunun "Modül bilgileri" standart web sitesi üzerinden (Sayfa 819)
- SIMATIC Automation Tool (https://support.industry.siemens.com/cs/ww/en/view/98161300) ile

DİKKAT

Memory card'ı yeniden biçimlendirmek için Windows biçimlendirme programını veya başka bir biçimlendirme programını kullanmayın.

Bir Siemens memory card Windows biçimlendirme programıyla yeniden biçimlendirilirse, bu memory card artık bir S7-1200 CPU'su tarafından kullanılamaz.

Firmware güncellemesini memory card'ınıza yüklemek için aşağıdaki yöntemi izleyin:

1. Yazma koruması olmayan boş bir SIMATIC memory card'ı, bilgisayarınıza bağlı bir SD kart okuyucuya takın. Kart yazma korumalı ise, koruma sürgüsünü kilitleme pozisyonundan dışarıya doğru itin.

   Halihazırda bir kullanıcı programı veya bir Firmware güncellemesi içeren bir SIMATIC memory card kullanabilirsiniz. Karışıklığı önlemek için, mevcutsa S7_JOB.SYS, SIMATIC.S7S ve FWUPDATE.S7S dosyalarını da silmeniz gerekir.

   DİKKAT

   "__LOG__" ve "crdinfo.bin" gizli dosyalarını memory card'dan SİLMEYİN.

   Memory card, "__LOG__" ve "crdinfo.bin" dosyalarına ihtiyaç duyar. Bu dosyaları silerseniz, memory card'ı artık CPU ile kullanamazsınız.

2. Modülünüze uygun Firmware güncellemesinin Zip dosyasını seçin ve bu Zip dosyasını bilgisayarınıza indirin. Dosyaya çift tıklayın, hedef yol olarak SIMATIC memory card'ın kök dizinini gösterin ve çıkarma işlemini başlatın. Çıkarma işlemi tamamlandıktan sonra, memory card'ın kök dizini (klasör) "FWUPDATE.S7S" dizinini ve "S7_JOB.S7S" dosyasını içerir.

   DİKKAT

   Memory card'ınıza aynı sipariş numarası (MLFB) için birden fazla Firmware güncellemesi kopyalamayın.

   Memory card'a aynı sipariş numarası için birden fazla güncelleme kopyalanırsa, Firmware güncellemesi başarısız olur.

3. Kartı dikkatli bir şekilde okuma cihazından çıkarın.

Firmware güncellemesini kurmak için aşağıdaki yöntemi izleyin:

⚠ İKAZ

Firmware güncellemesini kurmadan önce, CPU'nun hiçbir aktif proses yürütmediğinden emin olun.

CPU, Firmware güncellemesinin kurulması ile STOP işletim durumuna geçer, bu da bir çevrimiçi prosesin veya bir makinenin çalışmasını etkileyebilir. Bir prosesin veya makinenin beklenmeyen şekilde çalışması, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Memory card'ı takmadan önce, CPU'nun çevrimdışı olduğundan ve güvenli bir durumda bulunduğundan mutlaka emin olun.

1. Memory card'ı CPU'nun içine itin. CPU RUN işletim durumundaysa STOP işletim durumuna geçer. Memory card'ın değerlendirilmesinin gerektiğini belirtmek için bakım LED'i (MAINT) yanıp söner.
2. Firmware güncellemesini başlatmak için CPU'yu yeniden başlatın. CPU'yu yeniden başlatmak için kullanılabilecek alternatif yöntemler, STOP işletim durumundan RUN işletim durumuna geçmek veya STEP 7'de bir tam silme işlemi (MRES) gerçekleştirmektir.

   NOT

   Modül için Firmware güncellemesinin tamamlanabilmesi için, modülün harici 24 V DC beslemesinin açık durumda kalmasına dikkat edin.

   CPU yeniden başlatıldıktan sonra Firmware güncellemesi başlar. Güncellemenin kopyalandığını belirtmek için RUN/STOP LED'i yeşil ve sarı renkte yanıp söner. RUN/STOP LED'i sürekli olarak sarı renkte yanıyorsa ve MAINT LED'i yanıp sönüyorsa kopyalama işlemi tamamlanmıştır. Ardından memory card'ı çıkarmanız gerekir.
3. Yeni Firmware'i yüklemek için, memory card'ı çıkardıktan sonra CPU'yu yeniden başlatın (açarak veya yeniden başlatmak için başka bir yöntemle).

Kullanıcı programı ve donanım konfigürasyonu Firmware güncellemesinden etkilenmez. CPU açılırsa, CPU, yapılandırılan yol alma durumuna geçer. (CPU için yol alma işletim durumu olarak "Sıcak başlatma - ŞEBEKE KAPALI'dan önceki işletim türü" yapılandırıldıysa, CPU'nun son işletim durumu STOP olduğu için CPU, STOP işletim durumunda bulunur.)

Güncelleme sırasında, hiçbir istasyon donanım modülüne karşılık gelmeyen UPD dosyaları Firmware güncellemesi işlemi tarafından göz ardı edilir. Bu sayede, tesisinizdeki tüm S7-1200 CPU istasyonlarının güncellenmesini sağlayacak bir Master-Firmware güncellemesi için bir memory card oluşturabilirsiniz. Göz ardı edilen UPD dosyaları için teşhis tamponunda hiçbir giriş oluşturulmaz. Bu sayede, teşhis tamponunun ihtiyaç duyulmayan, çoğunlukla anlamsız girişler nedeniyle ortaya çıkan ve ilgilendiğiniz güncelleme işlemi girişlerini gizleyebilecek olan tıkanma durumları önlenir. Örneğin, bir Firmware güncellemesi gerçekleştirilmeye çalışıldığında (başarılı olsun ya da olmasın), teşhis tamponuna bir giriş yazılır. Daha sonra, bu işlem ihtiyaç duyulmayan girişlerle karmaşık hale gelmeden, teşhis tamponunu beklenmedik anormallikler açısından kolayca tarayabilirsiniz.

Firmware'i güncellemek için yapılan tüm denemelerin sonuçları teşhis tamponunda protokollendirilir. Bir güncelleme başarısız olursa, teşhis tamponundan gelen bir mesajla başarısız olma durumu açıklanır.

5.6 Şifre kaybedildiğinde izlenecek prosedür

Şifre korumalı bir CPU'nun şifresini kaybettiyseniz, şifre korumalı programı boş bir aktarım kartıyla silin. Boş aktarım kartı, CPU'nun dahili yükleme belleğini siler. Ardından STEP 7'den yeni bir kullanıcı programını CPU'ya yükleyebilirsiniz.

Boş bir aktarım kartının oluşturulması ve kullanılmasıyla ilgili bilgileri, Aktarım kartları (Sayfa 115) alt bölümünde bulabilirsiniz.

⚠ İKAZ

Memory card'ı takmadan önce CPU'nun hiçbir proses yürütmediğinden emin olun.

Bir aktarım kartını çalışan bir CPU'ya takarsanız, CPU, STOP işletim durumuna geçer. Kumanda üniteleri, güvenli olmayan işletim durumlarında arızalanabilir ve bunun sonucunda, kumanda edilen cihazların kontrolsüz şekilde çalışmasına neden olabilir. Bunun sonucunda, otomasyon sisteminde, ölüme veya ağır yaralanmalara ve/veya maddi hasara yol açabilen öngörülemeyen bir işletim durumu ortaya çıkar.

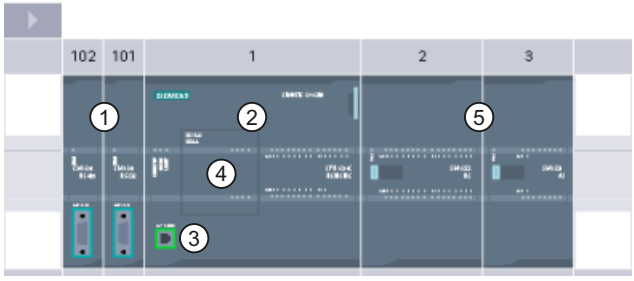
Bir aktarım kartını takmadan önce, CPU'nun STOP işletim durumunda olduğundan ve prosesinizin güvenli bir durumda bulunduğundan emin olun.

CPU'yu RUN işletim durumuna getirmeden önce aktarım kartını çıkartmanız gerekir.

Cihaz konfigürasyonu

6

PLC cihazınız için cihaz konfigürasyonunu, projenize bir CPU ve başka modüller ekleyerek oluşturabilirsiniz.



① İletişim modülü (CM) veya iletişim işlemcisi (CP): maksimum 3, 101, 102 ve 103 yuvalarında

② CPU: Yuva 1

③ CPU'nun PROFINET portu

④ Sinyal kartı (SB), iletişim kartı (CB) veya batarya kartı (BB) maks. 1, CPU'ya takılı

⑤ Dijital veya analog G/Ç için sinyal modülü (SM): maksimum 8, 2 ile 9 arasındaki yuvalarda (1212C, 1212FC ve 1211C CPU'lar hariç tutulmuştur. 1212C ve 1212FC CPU'lar iki sinyal modülüne izin verir. 1211C CPU hiçbir sinyal modülüne izin vermez.)

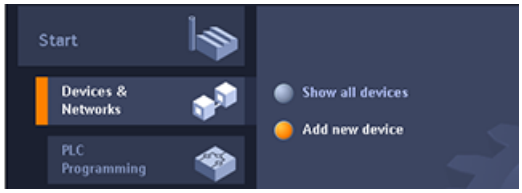
Konfigürasyon kumandası

S7-1200 için kullanılan cihaz konfigürasyonu, muhtemelen hepsi kullanılamayabilen modüllere sahip bir proje için bir maksimum konfigürasyonun projelendirilebileceği "Konfigürasyon kumandasını (Sayfa 133)" da destekler. Bazı durumlarda "Opsiyon yönetimi" olarak da adlandırılan bu fonksiyonla, değişken modüllerin çok sayıda uygulamada kullanımına izin veren bir maksimum donanım projelendirilebilir.

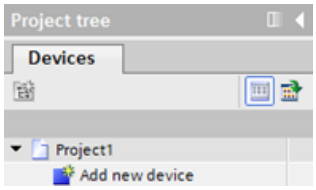
6.1 Bir CPU ekleme

Bir CPU'yu ya STEP 7'nin portal görünümünde ya da proje görünümünde projenize ekleyebilirsiniz:

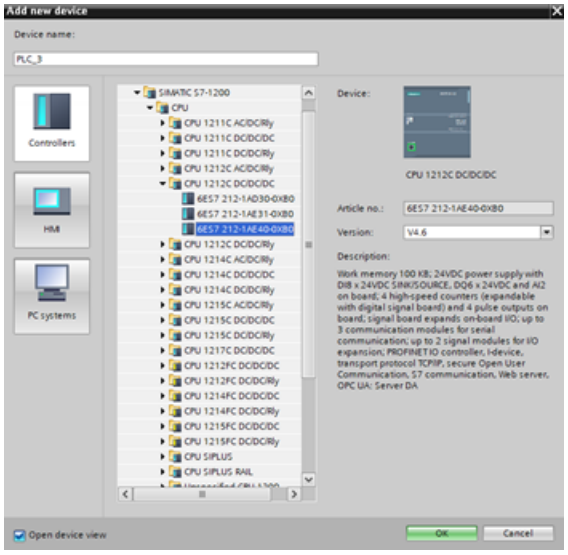
- Portal görünümünde "Cihazlar ve ağlar" portalını seçin ve "Yeni cihaz ekle" öğesine tıklayın.



- Proje görünümündeki proje adının altında yer alan "Yeni cihaz ekle" öğesine çift tıklayın.



"Yeni cihaz ekle" iletişim kutusundaki listeden doğru varyantı ve Firmware sürümünü seçin.

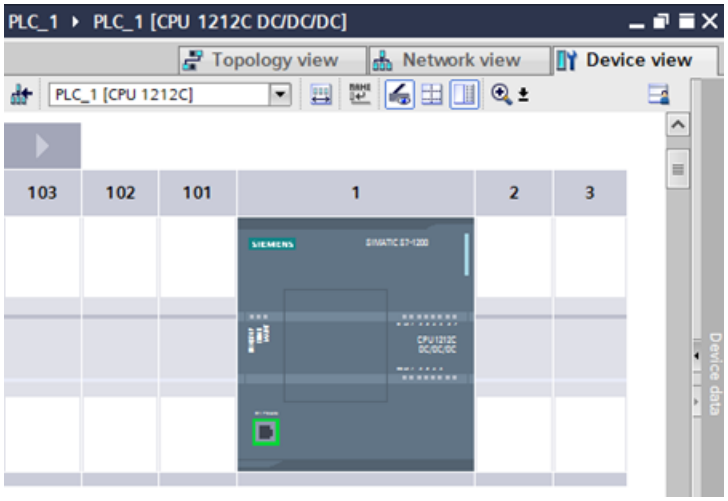


Eklenen CPU'nuz için PLC dijital güvenlik (security) ayarlarının yapılandırılması

Bir S7-1200 CPU V4.x eklerseniz, STEP 7, PLC'niz için dijital güvenlik ayarlarını yapmanızı sağlayan dijital güvenlik asistanını (Sayfa 153) başlatır. PLC dijital güvenlik ayarlarınızı belirlemek için asistandaki adımları takip edin.

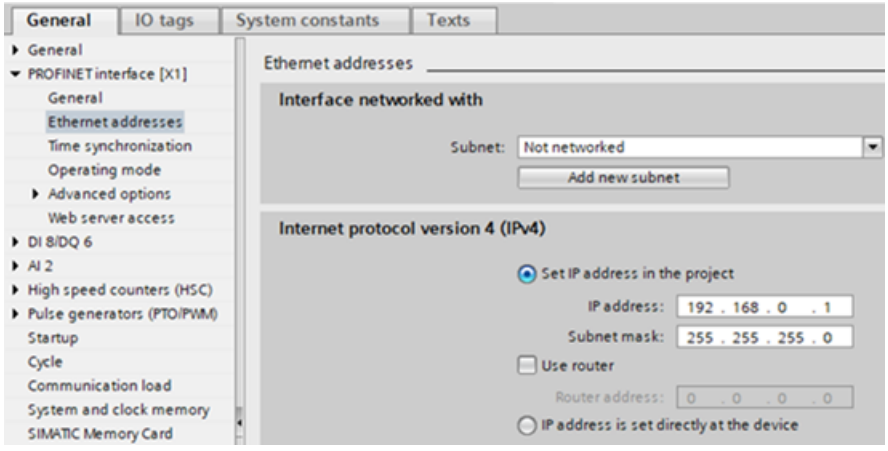
Takılan CPU'nun cihaz konfigürasyonu

STEP 7, bir CPU eklemenizden sonra modül taşıyıcısını oluşturur ve CPU'yu cihaz görünümünde gösterir:



Cihaz görünümünde CPU'ya tıklandıktan sonra CPU'nun özellikleri denetçi penceresinde görüntülenir.

CPU'ya cihaz konfigürasyonunda bir IP adresi atayabilirsiniz. CPU'nuz ağdaki bir yönlendiriciye bağlıysa, yönlendiricinin IP adresi de girilmelidir.



6.2 Konfigürasyonun bağlı bir CPU'dan yüklenmesi

STEP 7, bağlı bir CPU'nun donanım konfigürasyonunu iki şekilde yüklemeye olanak sağlar:

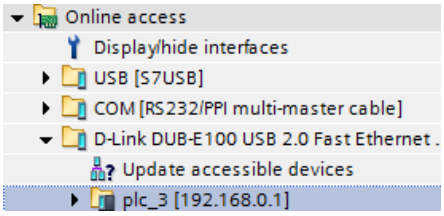
- Bağlı cihazın yeni bir istasyon olarak yüklenmesi
- Belirtilmeyen bir CPU'nun, bağlı CPU'nun donanım algılaması aracılığıyla yapılandırılması

Ancak ilk durumda bağlı CPU'nun hem donanım konfigürasyonunun hem de yazılımının yükleneceğine dikkat edilmelidir.

Cihazın yeni istasyon olarak yüklenmesi

Bağlı bir cihazı yeni istasyon olarak yüklemek için aşağıdaki yöntemi izleyin:

1. Proje gezgininde iletişim arayüzünüzü "Çevrimiçi erişimler" düğümü aracılığıyla genişletin.
2. "Erişilebilir katılımcıları güncelle" öğesine çift tıklayın.
3. Algılanan cihazların arasından PLC'yi seçin.



4. STEP 7'nin çevrimiçi menüsünden "Cihazı yeni istasyon olarak yükle (donanım ve yazılım)" komutunu seçin.

STEP 7, donanım konfigürasyonunu ve program bloklarını yükler.

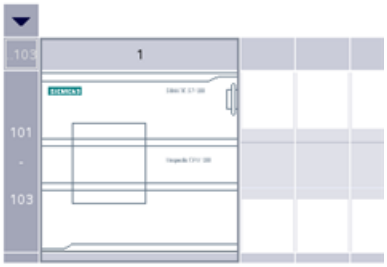
Belirtilmemiş bir CPU'nun donanım konfigürasyonunun algılanması



Bir CPU ile bağlantı mevcutsa, mevcut modüller de dahil olmak üzere bu CPU'nun konfigürasyonunu cihazdan projenize yükleyebilirsiniz. Bunun için yeni bir proje oluşturun ve belirli bir CPU yerine "belirtilmemiş CPU" öğesini seçin.

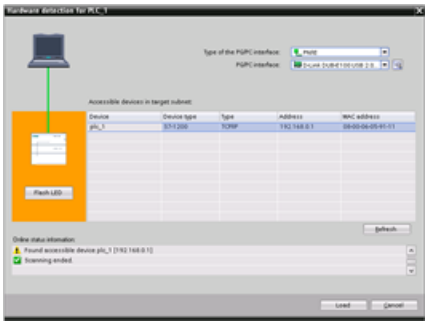
Programlama editöründeki "Çevrimiçi" menüsünden "Donanım algılama" komutunu seçin.

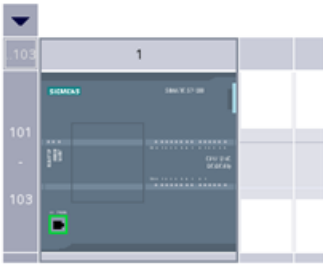
Cihaz konfigürasyonu editöründe, bağlı cihazın konfigürasyonunu algılama opsiyonunu seçin.



The device is not specified.
→ Please use the hardware catalog to specify the CPU,
→ or detect the configuration of the connected device.

STEP 7, Çevrimiçi iletişim kutusunda CPU'yu seçmenizin ve Yükleme butonuna tıklamanızın ardından, olası modüller de (SM, SB veya CM) dahil olmak üzere donanım konfigürasyonunu CPU'dan yükler. Daha sonra CPU ve modüller (Sayfa 144-145) için kullanılan parametreleri yapılandırabilirsiniz.





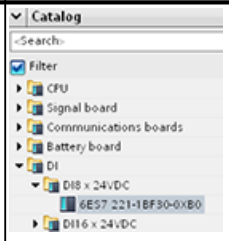
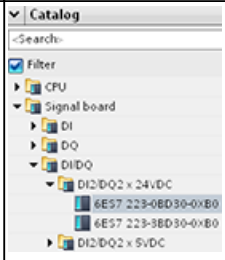
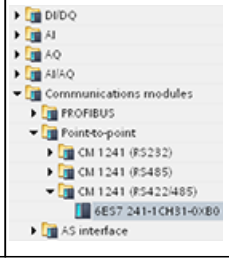
6.3 Konfigürasyona modüller ekleme

Donanım kataloğunda modülleri CPU'ya ekleyebilirsiniz:

- Ek dijital veya analog giriş ve çıkışlar için kullanılan sinyal modülleri (SM). Bu modüller, CPU'nun sağ tarafına bağlanır.
- Sinyal kartları (SB), CPU için sınırlı sayıda ek girişler/çıkışlar sunar. SB, CPU'nun ön tarafına takılır.
- Batarya kartı 1297 (BB), gerçek zaman saatinin uzun süreyle tamponlanmasına olanak sunar. BB, CPU'nun ön tarafına takılır.
- İletişim kartı (CB), ek bir iletişim bağlantısı (örn. RS485) sunar. CB, CPU'nun ön tarafına takılır.
- İletişim modülleri (CM) ve iletişim işlemcileri (CP), örn. PROFIBUS veya GPRS için ek bir iletişim bağlantısı sunar. Bu modüller, CPU'nun sol tarafına bağlanır.

Bir modülü cihaz konfigürasyonuna eklemek için modülü donanım kataloğunda seçin ve üzerine çift tıklayın veya işaretlenen yuvanın üzerine sürükleyin. Modüllerin çalışabilir durumda olması için modülleri cihaz konfigürasyonuna kaydetmeniz ve donanım konfigürasyonunu CPU'ya yüklemeniz gerekir.

Tablo 6-1 Cihaz konfigürasyonuna modül ekleme

Modül	Modül seçme	Modül kullanma	Sonuç
SM			
SB, BB veya CB			
CM veya CP			

"Konfigürasyon kumandası" (Sayfa 133) fonksiyonuyla muhtemelen belirli bir uygulama için güncel donanıma uygun olmayan, fakat aynı kullanıcı programını, CPU modelini ve diğer bileşenlerin yanı sıra yapılandırılmış modüllerden bazılarını da paylaşan başka uygulamalarda kullanılan sinyal modülleri ve sinyal kartları cihaz konfigürasyonuna eklenebilir.

6.4 Konfigürasyon kumandası

6.4.1 Konfigürasyon kumandasının avantajları ve kullanımı

Konfigürasyon kumandası, farklı yapılarda varyasyonlarla kullanılması gereken bir otomasyon çözümü geliştirilirken faydalı olabilir.

Bir STEP 7 cihaz konfigürasyonu ve bir kullanıcı programı, farklı kurulu PLC konfigürasyonlarına yüklenebilir. Bunun için STEP 7 projesinde yeni yapı için sadece az sayıda uyarlama gerekir.

6.4.2 Merkezi yapının ve opsiyonel modüllerin yapılandırılması

STEP 7'deki konfigürasyon kumandası ve S7-1200, her biri bu konfigürasyonun bir alt kümesini kullanan farklı sürümlere (opsiyonlara) sahip bir standart sistem için bir maksimum konfigürasyonun projelendirilmesine olanak sağlar. STEP 7ile PROFINET işletme kitabında (http://support.automation.siemens.com/WW/view/en/49948856) bu proje tipleri "seri makine projeleri" olarak tanımlanır.

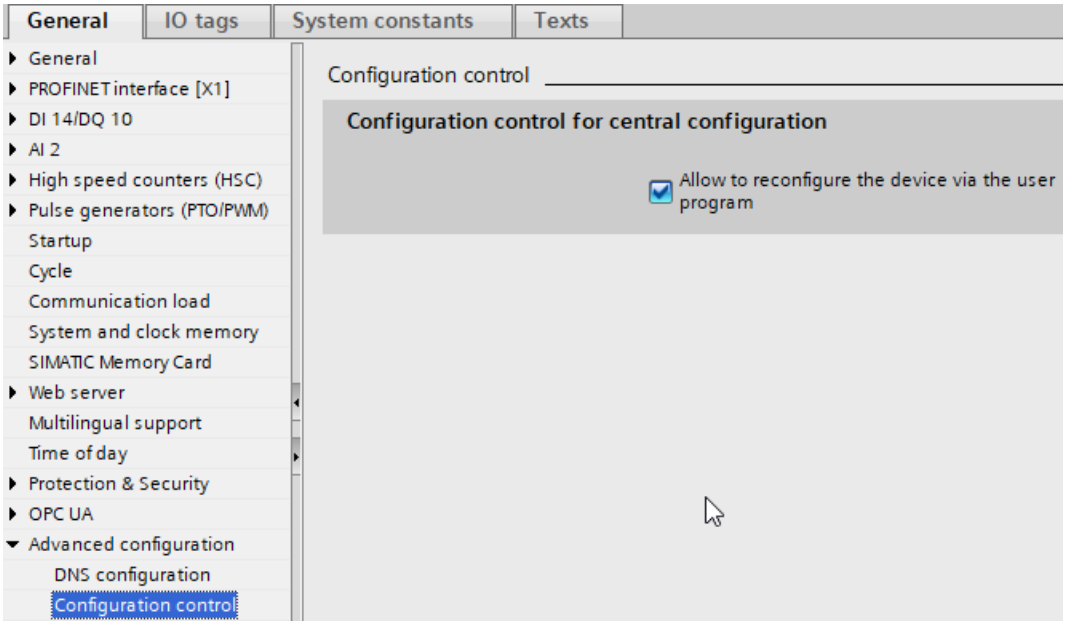
Yol alma OB'sinde programlanan bir kumanda veri seti, CPU'ya gerçek yapıda projelendirmeden farklı olarak hangi modüllerin eksik olduğunu veya projelendirmeden farklı olarak hangi modüllerin başka bir yuvada bulunduğunu bildirir. Konfigürasyon kumandası, modüllerin parametrelendirilmesi üzerinde etkili değildir.

Yapı, gerçek yapı STEP 7'deki maksimum cihaz konfigürasyonunda mevcut olduğu sürece konfigürasyon kumandasıyla esnek bir şekilde değiştirilebilir.

Konfigürasyon kumandasını etkinleştirmek ve gereken kumanda veri setini yapılandırmak için aşağıdaki yöntemi izleyin:

1. CPU, CPU'da uyumsuz bir kumanda veri setinin mevcut olmadığından emin olmak için opsiyonel olarak sıfırlanarak fabrika ayarlarına döndürülebilir.
2. STEP 7'deki cihaz konfigürasyonunda CPU'yu seçin.

3. "Konfigürasyon kumandası" katılımcısındaki CPU özelliklerinde "Kullanıcı programı aracılığıyla cihazın yeniden yapılandırılmasına olanak sağla" opsiyonunu etkinleştirin.

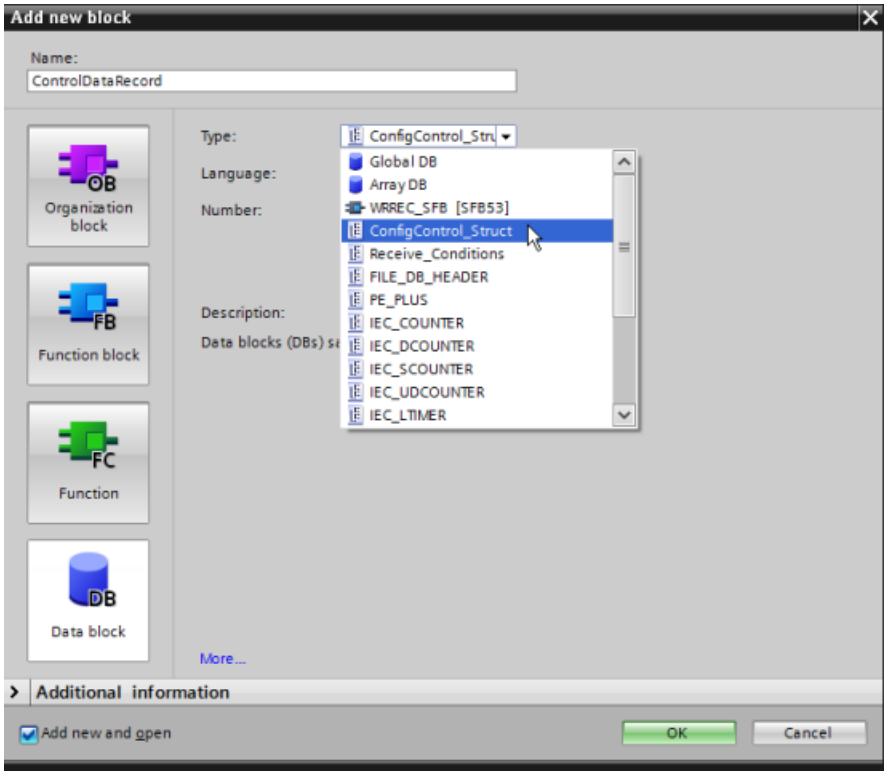


4. Kumanda veri seti için bir PLC veri tipi oluşturun. Konfigürasyon kumandası verileri için dört USint ve S7-1200'ün maksimum donanımı için yuva sayısına uygun olarak diğer USint'lerle birlikte yapı olarak bunu yapılandırın:

ConfigControl_Struct

	Name	Data type	Default value	Comment
1	ConfigControl	Struct		
2	Block_length	USInt	16	Length of control data record, including header
3	Block_ID	USInt	196	Data record number
4	Version	USInt	5	
5	Subversion	USInt	0	
6	Slot_1	USInt	255	Assignment for CPU annex card/Actual annex card
7	Slot_2	USInt	255	Configured slot 2/ Assigned "real" slot
8	Slot_3	USInt	255	Configured slot 3 / Assigned "real" slot
9	Slot_4	USInt	255	Configured slot 4 / Assigned "real" slot
10	Slot_5	USInt	255	Configured slot 5 / Assigned "real" slot
11	Slot_6	USInt	255	Configured slot 6 / Assigned "real" slot
12	Slot_7	USInt	255	Configured slot 7 / Assigned "real" slot
13	Slot_8	USInt	255	Configured slot 8 / Assigned "real" slot
14	Slot_9	USInt	255	Configured slot 9 / Assigned "real" slot
15	Slot_101	USInt	255	Configured slot 101 / Assigned "real" slot
16	Slot_102	USInt	255	Configured slot 102 / Assigned "real" slot
17	Slot_103	USInt	255	Configured slot 103 / Assigned "real" slot

5. Seçilen PLC veri tipiyle bir veri bloğu oluşturun.



6. Bu veri bloğunda Block_length, Block_ID, Version ve Subversion parametrelerini aşağıda gösterildiği şekilde yapılandırın. Dolu olup olmadıklarına veya gerçek yapıdaki konumlarına göre yuvalar için kullanılan değerleri yapılandırın:
   – 0: Yapılandırılmış modül, gerçek konfigürasyonda mevcut değil. (Yuva boş.)
   – 1 ila 9, 101 ila 103: Yapılandırılmış yuvanın gerçek pozisyonu
   – 255 STEP 7 cihaz konfigürasyonu bu yuvada modül içermiyor.

NOT

Konfigürasyon kumandası, sinyal kartındaki HSC'ler ve PTO'lar için kullanılamıyor

CPU'da HSC'ler veya PTO'lar için yapılandırdığınız bir sinyal kartınız varsa, bu sinyal kartını konfigürasyon için kumanda veri setine ait yuva_1'de bir "0" ile devre dışı bırakmamanız gerekir. CPU'nun yapılandırılmış HSC ve PTO cihazları konfigürasyon kumandası için zorunludur.

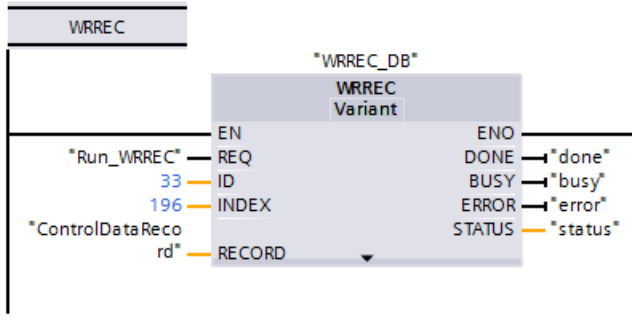
ControlDataRecord

	Name	Data type	Start value	Comment
1	Static			
2	ConfigControl	Struct		
3	Block_length	USInt	16	Length of control data record, including header
4	Block_ID	USInt	196	Data record number
5	Version	USInt	5	
6	Subversion	USInt	0	
7	Slot_1	USInt	255	Assignment for CPU annex card/Actual annex c...
8	Slot_2	USInt	255	Configured slot 2/ Assigned "real" slot
9	Slot_3	USInt	255	Configured slot 3 / Assigned "real" slot
10	Slot_4	USInt	255	Configured slot 4 / Assigned "real" slot
11	Slot_5	USInt	255	Configured slot 5 / Assigned "real" slot
12	Slot_6	USInt	255	Configured slot 6 / Assigned "real" slot
13	Slot_7	USInt	255	Configured slot 7 / Assigned "real" slot
14	Slot_8	USInt	255	Configured slot 8 / Assigned "real" slot
15	Slot_9	USInt	255	Configured slot 9 / Assigned "real" slot
16	Slot_101	USInt	255	Configured slot 101 / Assigned "real" slot
17	Slot_102	USInt	255	Configured slot 102 / Assigned "real" slot
18	Slot_103	USInt	255	Configured slot 103 / Assigned "real" slot

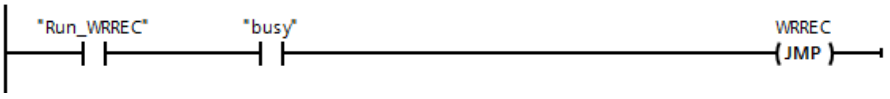
Yuvalar için değerlerin atanmasıyla ilgili olarak, bkz. Konfigürasyon kumandasıyla ilgili örnek (Sayfa 140).

7. Kumanda veri setini donanım ID'si 33'e ait Indeks 196'ya aktarmak için yol alma OB'sinde WRREC gelişmiş yordamını (Veri setini yaz) çağırın. WRREC yordamı tamamlanana kadar beklemek için bir atlama işaretini (JMP) kullanın.

Ağ 1:



Ağ 2:



NOT

Konfigürasyon kumandası ancak WRREC yordamı, kumanda veri setini yol alma OB'sine aktardığı takdirde etkili olur. Konfigürasyon kumandası etkinleştirildiyse ve kumanda veri seti CPU'da mevcut değilse, bu bileşen, yol alma işleminden sonra doğrudan STOP konumuna geçer. Bu nedenle, yol alma OB'sinin, kumanda veri seti aktarılacak şekilde programlanmış olması önemlidir.

Modül düzeni

Aşağıdaki tabloda, yuvaların ataması gösterilmektedir.

Yuva	Modüller
1	Sinyal kartı veya iletişim kartı (CPU ek kartı)
2 ila 9	Sinyal modülleri
101 ila 103	İletişim modülleri

Kumanda veri seti

Bir 196 kumanda veri seti, aşağıda gösterildiği gibi yuva ataması ve gerçek konfigürasyonu içerir:

Bayt	Eleman	Değer	Açıklama
0	Blok uzunluğu	16	Header
1	Blok ID'si	196	
2	Sürüm	5	
3	Alt sürüm	0	
4	Bir CPU ek kartının atanması	Gerçek ek kart, 0 veya 255*	Kumanda elemanı Her elemanda, cihazdaki hangi gerçek yuvanın yapılandırılmış yuvaya atandığını tanımlar.
5	Yapılandırılmış yuva 2'nin atanması	Gerçek yuva, 0 veya 255*	
...	...	...	
12	Yapılandırılmış yuva 9'un atanması	Gerçek yuva, 0 veya 255*	
13	Yapılandırılmış yuva 101'in atanması	Gerçek yuva 255*	Gerçek yuva, fiziksel olarak mevcut iletişim modüllerinde, sinyal modüllerinden farklı olarak, yapılandırılmış yuvayla uyumlu olmalıdır.
14	Yapılandırılmış yuva 102'nin atanması	Gerçek yuva 255*	
15	Yapılandırılmış yuva 103'ün atanması	Gerçek yuva 255*	

***Yuva değerleri:**

0: Yapılandırılmış modül, gerçek konfigürasyonda mevcut değil. (Yuva boş.)

1 ila 9, 101 ila 103: Yapılandırılmış yuvanın gerçek pozisyonu

255 STEP 7 cihaz konfigürasyonu bu yuvada modül içermiyor.

NOT

Bir PLC değişken tipinin oluşturulmasına alternatif

Kullanıcı tanımlı bir PLC değişken tipinin oluşturulmasına alternatif olarak, bir veri bloğunu, doğrudan kumanda veri setinin tüm yapı elemanları oluşturabilirsiniz. Hatta bu veri bloğunda, farklı kumanda veri seti konfigürasyonları olarak görev gören birden çok yapıyı yapılandırabilirsiniz. Her iki uygulama da, kumanda veri setini yol alma sırasında aktarmak için etkili birer yöntemdir.

Kurallar

Aşağıdaki kurallara dikkat edilmelidir:

- Konfigürasyon kumandası, iletişim modülleri için pozisyon değişikliğini desteklemez. Konfigürasyon kumandasını, iletişim modüllerini devre dışı bırakmak için de kullanamazsınız. 101 ile 103 arasındaki yuvaların kumanda veri setinde belirtilen pozisyonları gerçek yapıya karşılık gelmelidir. Cihaz konfigürasyonunuzda yuva için hiçbir modül yapılandırılmadıysa, bu pozisyon için kumanda veri setine 255 değerini girin. Yuva için bir modül yapılandırıldıysa, yapılandırılmış yuvayı bu pozisyon için gerçek yuva olarak girin.
- F-G/Ç modülleri konfigürasyon kumandasını desteklemez. Kumanda veri setindeki bir F-G/Ç modülü için kullanılan yuva pozisyonları, F-G/Ç modülü için kullanılan yapılandırılmış yuva pozisyonuna karşılık gelmelidir. Yapılandırılmış bir F-G/Ç modülünü kumanda veri seti ile taşımaya veya silmeye çalışırsanız, gerçekten kurulu olan tüm F-G/Ç modülleri bir parametrelendirme hatası bildirir ve değiştirmeye izin vermez.
- Dolu yuvaların arasında kullanılmayan yuvalara izin verilmez. Gerçek konfigürasyonda örneğin yuva 4'te bir modül mevcutsa, yuva 2 ve 3'e de modüllerin takılmış olması gerekir. Gerçek konfigürasyonda yuva 102'de bir iletişim modülü mevcutsa, yuva 101'de de buna uygun olarak bir modülün takılmış olması gerekir.
- Konfigürasyon kumandası etkinleştirildiyse, CPU ancak bir kumanda veri seti yüklendiğinde çalışmaya hazır hale gelir. CPU, yol alma OB geçerli bir kumanda veri setini aktarmıyorsa yol alma modundan STOP konumuna geçer. Bu durumda merkezi G/Ç'ler CPU tarafından başlatılmaz ve STOP konumuna geçme nedeni teşhis tamponuna yazılır.
- Başarılı bir şekilde aktarılan bir kumanda veri seti, CPU tarafından kalıcı belleğe yerleştirilir; bu nedenle, konfigürasyon değiştirilmediği sürece 196 kumanda veri setinin bir yeniden başlatma işleminden sonra yeniden yazılması gerekmez.
- Her gerçek yuva, kumanda veri setinde sadece bir kez mevcut olmalıdır.
- Yapılandırılmış her yuvaya sadece bir gerçek yuva atanabilir.

NOT

Bir konfigürasyonun değiştirilmesi

Bir kumanda veri seti, değiştirilen bir konfigürasyonla yazılırsa CPU'da şu akış otomatik olarak başlatılır: Tam silme ve ardından değiştirilen konfigürasyonla yeniden başlatma.

Bunun üzerine CPU, orijinal kumanda veri setini siler ve yeni veri setini kalıcı belleğe kaydeder.

İşletim sırasındaki davranış

STEP 7, çevrimiçi gösterge ve teşhis tamponundaki gösterge için (modül TAMAM veya modül hatalı) sapma gösteren gerçek konfigürasyon yerine cihaz konfigürasyonunu kullanır.
Örnek: Bir modül, teşhis verilerini yayınlar. Bu modül, konfigürasyona göre yuva 4'e, fakat gerçekte yuva 3'e takılıdır. Çevrimiçi görünümde yuva 4'te bir hata görüntülenir. Gerçek konfigürasyonda yuva 3'teki modül, kendi LED göstergesinde bir hata bildirir.

Kumanda veri setinde modüller hatalı (0) olarak yapılandırıldıysa, otomasyon sistemi şu şekilde davranır:

- Kumanda veri setinde mevcut değil olarak yapılandırılmış olan modüller, teşhis verileri bildirmez ve durumları her zaman TAMAM'dır. Değer durumu TAMAM.
- Çıkışlara doğrudan bir yazma erişimi veya mevcut olmayan çıkışların proses görünümüne yazma erişiminin hiçbir etkisi yoktur; CPU erişim hatası bildirmez.
- Girişlere doğrudan bir okuma erişimi veya mevcut olmayan girişlerin proses görünümüne okuma erişimi, her giriş için "0" şeklinde bir değere neden olur; CPU erişim hatası bildirmez.
- Bir veri setinin mevcut olmayan bir modüle yazılmasının hiçbir etkisi yoktur; CPU hata bildirmez.
- Bir veri seti mevcut olmayan bir modülden okunmaya çalışılırsa, CPU geçerli bir veri seti döndüremediği için bir hata mesajı oluşturulur.

Hata mesajları

CPU, kumanda veri seti yazılırken bir hata oluştuğu takdirde aşağıdaki hata mesajlarını döndürür:

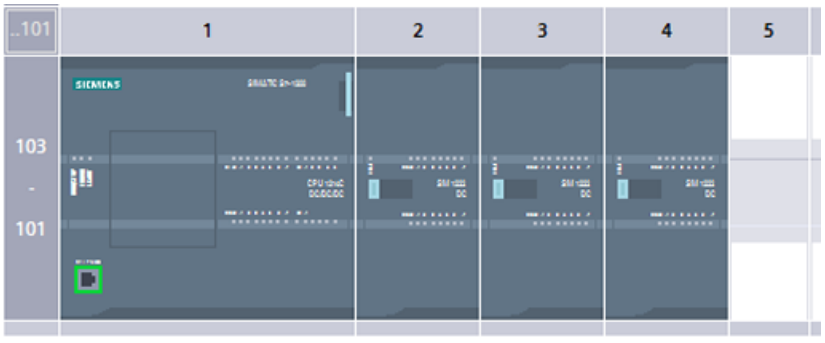
Hata kodu	Anlamı
16#80B1	Geçersiz uzunluk; kumanda veri setinde yanlış bir uzunluk belirtilmiş.
16#80B5	Konfigürasyon kumandası parametreleri atanmadı
16#80E2	Veri seti yanlış OB'de aktarıldı. Veri seti, yol alma OB'sinde aktarılmalıdır.
16#80B0	Kumanda veri setinin blok tipi (Bayt 2) 196 değil.
16#80B8	Parametre hatası; modül geçersiz parametreler bildiriyor, örneğin: • Kumanda veri seti, bir iletişim modülünün veya bir iletişim ek kartının konfigürasyonunu değiştirmeyi deniyor. İletişim modülleri ve bir iletişim ek kartı için kullanılan gerçek konfigürasyon, STEP 7'deki konfigürasyonla eşleşmelidir. • STEP 7 projesindeki yapılandırılmamış bir yuva için atanan değer 255 değil. • Yapılandırılmış bir yuva için atanan değer izin verilen aralıkta değil. • Atanmış konfigürasyon, boş bir "dahili" yuvaya sahip, örn. yuva n atandı ve yuva n-1 atanmadı.

6.4.3 Konfigürasyon kumandasıyla ilgili örnek

Bu örnekte, bir CPU ve üç G/Ç modülünden oluşan bir konfigürasyon açıklanmaktadır. Yuva 3'teki modül, ilk gerçek yapıda mevcut değil ve bu nedenle konfigürasyon kumandasının yardımıyla "gizlenir".
İkinci yapıda, başlangıçta gizlenmiş olan modül kullanılır, ancak şimdi son yuvada yapılandırılmıştır. Değiştirilen bir kumanda veri seti, modüllerin atamasıyla ilgili bilgileri yuvalara aktarır.

Örnek: Yapılandırılmış, fakat kullanılmayan bir modül içeren gerçek yapı

Cihaz konfigürasyonu, gerçek bir yapıda mevcut olabilecek tüm modülleri içerir (maksimum donanım). Bu durumda, cihaz konfigürasyonunda yuva 3'e atanmış olan modül gerçek yapıda mevcut değildir.



Resim 6-1 Üç sinyal modüllü maksimum donanım için kullanılan cihaz konfigürasyonu



Resim 6-2 Yuva 3 için eksik modül ve yuva 3'te yuva 4 için yapılandırılmış modül ile birlikte gerçek yapı

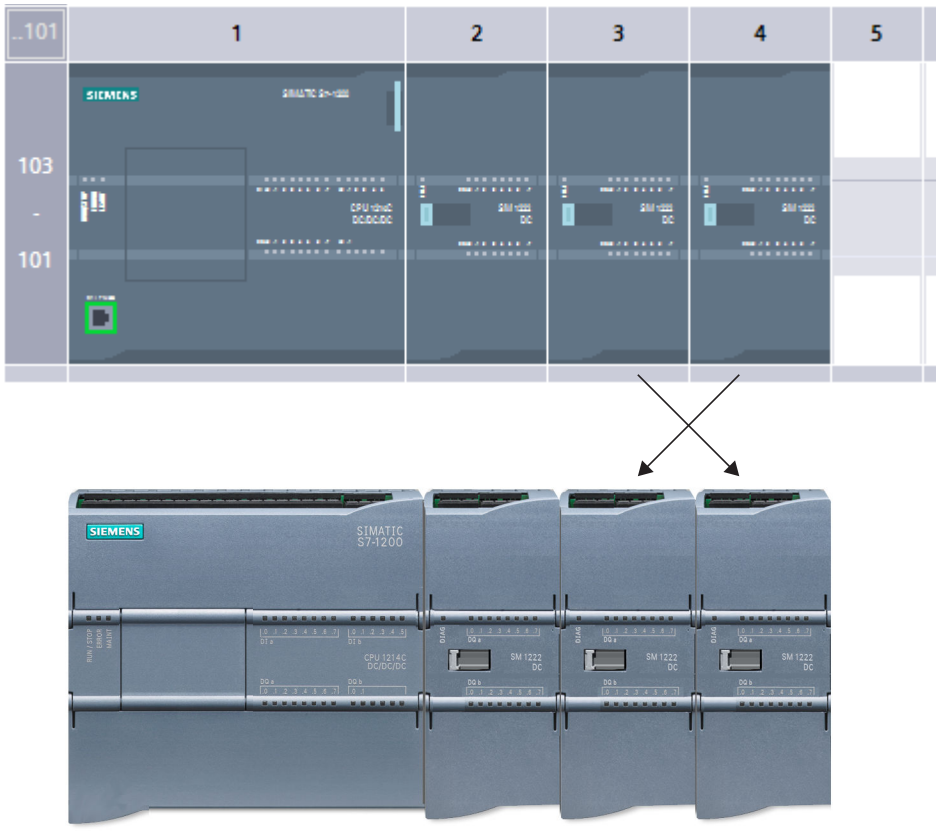
Bir modülün eksik olduğunu göstermek için, yuva 3, kumanda veri setinde 0 değeriyle yapılandırılmalıdır.

ControlDataRecord

	Name	Data type	Start value	Comment
1	Static			
2	ConfigControl	Struct		
3	Block_length	USInt	16	Length of control data record, including header
4	Block_ID	USInt	196	Data record number
5	Version	USInt	5	
6	Subversion	USInt	0	
7	Slot_1	USInt	255	Assignment for CPU annex card/Actual annex ca...
8	Slot_2	USInt	2	Configured slot 2 / Assigned "real" slot
9	Slot_3	USInt	0	Configured slot 3 / Assigned "real" slot
10	Slot_4	USInt	3	Configured slot 4 / Assigned "real" slot
11	Slot_5	USInt	255	Configured slot 5 / Assigned "real" slot
12	Slot_6	USInt	255	Configured slot 6 / Assigned "real" slot
13	Slot_7	USInt	255	Configured slot 7 / Assigned "real" slot
14	Slot_8	USInt	255	Configured slot 8 / Assigned "real" slot
15	Slot_9	USInt	255	Configured slot 9 / Assigned "real" slot
16	Slot_101	USInt	255	Configured slot 101 / Assigned "real" slot
17	Slot_102	USInt	255	Configured slot 102 / Assigned "real" slot
18	Slot_103	USInt	255	Configured slot 103 / Assigned "real" slot

Örnek: Daha sonra başka bir yuvaya eklenen bir modülle birlikte gerçek yapı

İkinci örnekte modül, gerçek yapıda cihaz konfigürasyonunun 3. yuvasında mevcuttur, ancak yuva 4'te bulunmaktadır.



Resim 6-3 Cihaz konfigürasyonundaki yuva 3 ve 4'te bulunan modüller gerçek yapıda birbiriyle değiştirildi

Modüller, cihaz konfigürasyonunu gerçek yapıya uyarlamak için, kumanda veri setinin düzenlenmesi yoluyla doğru yuvalara atanmalıdır.

ControlDataRecord

	Name	Data type	Start value	Comment
1	Static			
2	ConfigControl	Struct		
3	Block_length	USInt	16	Length of control data record, including header
4	Block_ID	USInt	196	Data record number
5	Version	USInt	5	
6	Subversion	USInt	0	
7	Slot_1	USInt	255	Assignment for CPU annex card/Actual annex card
8	Slot_2	USInt	2	Configured slot 2/ Assigned "real" slot
9	Slot_3	USInt	4	Configured slot 3 / Assigned "real" slot
10	Slot_4	USInt	3	Configured slot 4 / Assigned "real" slot
11	Slot_5	USInt	255	Configured slot 5 / Assigned "real" slot
12	Slot_6	USInt	255	Configured slot 6 / Assigned "real" slot
13	Slot_7	USInt	255	Configured slot 7 / Assigned "real" slot
14	Slot_8	USInt	255	Configured slot 8 / Assigned "real" slot
15	Slot_9	USInt	255	Configured slot 9 / Assigned "real" slot
16	Slot_101	USInt	255	Configured slot 101 / Assigned "real" slot
17	Slot_102	USInt	255	Configured slot 102 / Assigned "real" slot
18	Slot_103	USInt	255	Configured slot 103 / Assigned "real" slot

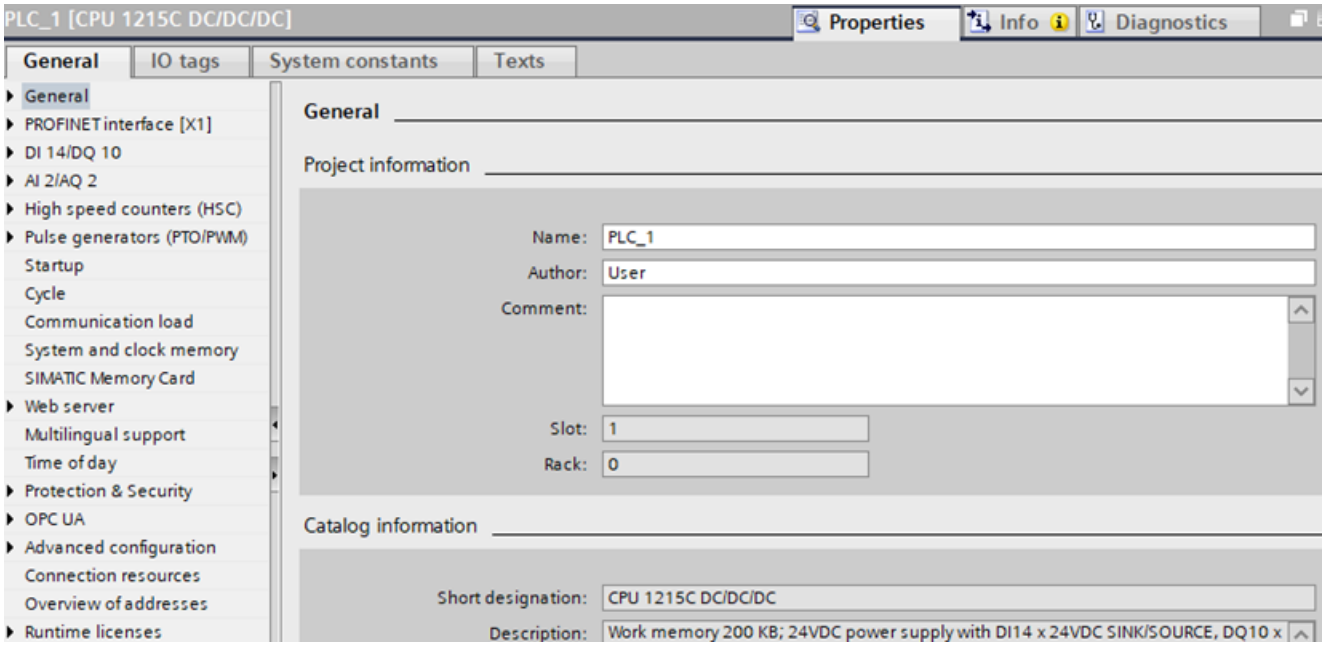
6.5 Bir cihazın değiştirilmesi

Yapılandırılmış bir CPU'nun veya bir modülün cihaz tipini değiştirebilirsiniz. Cihaz konfigürasyonunda farenin sağ tuşuyla cihaza tıklayın ve bağlam menüsünden "Cihazı değiştir" komutunu seçin. İletişim kutusunda değiştirmek istediğiniz CPU'yu veya modülü seçin. "Cihazı değiştir" iletişim kutusunda ilgili iki cihazla ilgili uyumluluk bilgileri görüntülenir. Cihazların CPU sürümünü değiştirmek istiyorsanız, bununla ilgili daha fazla bilgiyi, Bir CPU V3.0'ın CPU V4.x ile değişimi (Sayfa 1365) başlığında bulabilirsiniz.

6.6 CPU işletiminin yapılandırılması

6.6.1 CPU özellikleri

CPU'nun işletim parametrelerini yapılandırmak için CPU'yu cihaz görünümünde seçin ve denetçi penceresinde "Özellikler" sekmesini açın.



Tablo 6-2 CPU özellikleri

Özellik	Açıklama
Genel	Proje bilgilerini, Katalog bilgilerini, Tanımlama ve bakımı ve Sağlama toplamlarını kapsar.
PROFINET arayüzü	CPU ve saat senkronizasyonu için IP adresinin ayarlanması
DE, DA ve AE	Yerel (entegre) dijital ve analog G/Ç davranışının ayarlanması (Sayfa 146) (örn. dijital girişler için kullanılan filtre süreleri (Sayfa 147) ve dijital çıkışların (Sayfa 1113), CPU'nun STOP işletim durumuna ayarlanmasına tepkisi).
Hızlı sayaçlar (Sayfa 509) ve Darbe jeneratörleri (Sayfa 444)	Darbe dizileri (PTO) ve darbe süresi modülasyonu (PWM) için hızlı sayaçların (HSC) ve darbe jeneratörlerinin etkinleştirilmesi ve ayarlanması CPU'nun veya sinyal kartının çıkışlarını darbe jeneratörleri (PWM, PTO veya hareket kumandası yordamları için) yapılandırırsanız, ilgili çıkış adresleri çıkışların belleğinden kaldırılır ve kullanıcı programınızda başka amaçlarla kullanılamaz. Kullanıcı programınız darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU bu değeri fiziksel çıkışa yazar.
Yol alma (Sayfa 63)	**GÜÇ AÇIK durumundan sonra yol alma:** Kapatma ve tekrar çalıştırma sonrasında CPU'nun davranışının ayarlanması, örn. STOP işletim durumunda yol alma veya bir sıcak başlatma sonrasında RUN işletim durumuna geçmek için.
	Ön ayarın gerçek konfigürasyonla karşılaştırılması: CPU'nun yol alma davranışının, S7-1200 istasyonunun gerçek konfigürasyonunun nominal konfigürasyonla eşleşmediği durumlardaki spesifikasyonu: • CPU'nun yalnızca uyumluluk durumunda yol alması • CPU'nun uyumluluk olmasa bile yol alması Yapılandırılmış yuvadaki bir modül, yapılandırılmış modülle uyumlu olmalıdır. Uyumlu, mevcut modülün giriş ve çıkışların sayısıyla eşleşmesi ve elektriksel ve fonksiyonel özellikler açısından

Özellik	Açıklama
Yol alma (Sayfa 63)	uygun olması anlamına gelir. Daha fazla kapasiteye sahip olabilir, ancak daha az kapasiteye sahip olamaz.
	Konfigürasyon zamanı: Merkezi G/Ç'nin ve desantral çevrebirimin yol alması gereken bir maksimum zaman diliminin spesifikasyonu (ön ayar: 60.000 msn). (CM'ler ve CP'ler, yol alma sırasında CPU tarafından gerilimle ve iletişim parametreleriyle beslenir. Bu atama süresi, CM veya CP'ye bağlı G/Ç modüllerinin çevrimiçi hale getirilmesi gereken bir zaman dilimine izin verir.) CPU, merkezi G/Ç ve desantral çevrebirim yol aldıktan ve çalışmaya hazır hale geldikten sonra, atama süresinden bağımsız olarak RUN işletim durumuna geçer. Merkezi G/Ç ve desantral çevrebirim bu süre zarfında çevrimiçi hale getirilmediyse, CPU, merkezi G/Ç ve desantral çevrebirim olmadan yine de RUN işletim durumuna geçer.
	OB'ler durdurulabilir olmalıdır: CPU'daki OB modelinin (tüm OB'ler) durdurulabilir mi, yoksa durdurulamaz mı (Sayfa 80) olacağının belirlenmesi.
Çevrim (Sayfa 84)	Maksimum bir çevrim süresinin veya sabit bir minimum çevrim süresinin belirlenmesi
İletişim yükü	CPU süresinin yüzdesel bir oranının iletişim görevleri için atanması.
Sistem ve zamanlama markörü (Sayfa 88)	Bir Bayt'ın sistem markörü fonksiyonları için ayarlanması ve bir Bayt'ın zamanlama markörü fonksiyonları için ayarlanması (bu sırada her Bit önceden belirtilen bir frekansla açılır ve kapanır).
SIMATIC memory card	CPU'yu, SD kartın yapılandırılmış bir yüzde değerine kadar ne zaman eskidiğini belirleyecek şekilde yapılandırmanızı sağlar. Yüzde oranını eşik değeri olarak yapılandırmak için "SIMATIC memory card'ın eskimesi" opsiyon kutusunu etkinleştirebilirsiniz. GetSMCInfo (Sayfa 435) yordamıyla SIMATIC memory card'ı bu yapılandırılmış değere göre kontrol edebilirsiniz.
Web sunucusu (Sayfa 796)	Web sunucusu fonksiyonunun etkinleştirilmesi ve yapılandırılması.
Çok dilli destek (Sayfa 151)	Web sunucusu için bir proje dilini atar. Bu dil, Web sunucusunun kullanıcı arayüzünde teşhis tamponu mesajlarının görüntülenmesi için kullanılır.
Günün saati	Zaman diliminin seçilmesi ve yaz/kış saatinin ayarlanması.
Koruma ve dijital güvenlik (Sayfa 156)	Okuma/yazma korumasının ve CPU'ya erişim için kullanılan şifrelerin ayarlanması.
OPC UA (Sayfa 764)	OPC UA uygulamasının adının görüntülenmesi. OPC UA sunucu ayarlarının ve güvenlik ayarlarının etkinleştirilmesine ve yapılandırılmasına olanak sağlar. Güvenlik ayarlarında şu opsiyonlardan faydalanabilirsiniz: • Güvenli kanal • Sertifikalar • Kullanıcı kimlik doğrulaması Durum değişiminde teşhis, Ek olayların teşhisi ve Mesaj hacmi yüksek olduğunda teşhisin özetlenmesi opsiyonlarını sunar.
Gelişmiş konfigürasyon	Aşağıdakileri kapsar: • DNS konfigürasyonu (Sayfa 595) - DNS sunucusunun adresinin konfigürasyonu. • Konfigürasyon kumandası (Sayfa 133) - Kullanıcı programında belirli sınırlar içinde bir konfigürasyon değişimine olanak sağlar. • SNMP (Sayfa 730) - SNMP'nin (Simple Network Management Protocol) etkinleştirilmesi.
Bağlantı kaynakları	CPU'nun iletişim bağlantısı kaynaklarına ve yapılandırılmış bağlantı kaynaklarının sayısına bir genel bakış sunar.
Adreslere genel bakış	CPU için yapılandırılmış olan G/Ç adreslerine bir genel bakış sunar.
Runtime lisansları	Gerekli lisansın tipini ve Satın alınan lisansın tipini (OPC-UA) seçme.

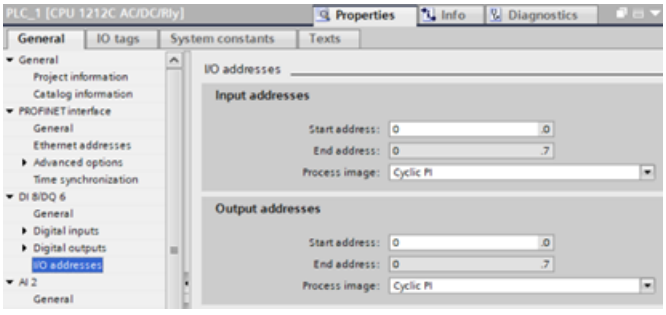
6.6.2 Entegre G/Ç'nin yapılandırılması

Modüllerin işletim parametrelerini yapılandırmak için, modülü cihaz görünümünde seçin ve modül için kullanılan parametreleri ayarlamak üzere denetçi penceresinde "Özellikler" sekmesini açın.

Yapılandırılabilen parametreler

Entegre G/Ç için kullanılan cihaz konfigürasyonunda şunları yapılandırabilirsiniz:

- Dijital G/Ç: Yükselen kenarları algılamak (Sayfa 70), Alçalan kenarları algılamak (Sayfa 70) veya Darbe toplamak (Sayfa 149) için kullanılan girişleri yapılandırabilirsiniz. Çıkışlar dondurulabilir veya yedek değerler (Sayfa 92) devreye sokulabilir.
- Analog G/Ç: Münferit girişler için örn. ölçüm türü (gerilim veya akım), aralık ve düzleştirme gibi parametreleri, ayrıca düşük kalma veya taşma teşhisinin etkinleştirilmesini yapılandırabilirsiniz. Analog çıkışlar, çıkış türü (gerilim veya akım) gibi parametreler ve örn. kısa devre (gerilim çıkışlarında) gibi teşhis veya üst/alt limitlerinin teşhis olanakları sunar. Fiziksel birimlerdeki analog girişlerin ve analog çıkışların aralıklarını "Özellikler" iletişim kutusunda yapılandırmayın. Bu işlemleri program lojiğinde gerçekleştirebilirsiniz (bkz. "Analog değerlerin işlenmesi (Sayfa 98)").
- G/Ç adresleri: Burada, G/Ç için başlangıç adresini yapılandırabilirsiniz. Girişleri ve çıkışları bir kısmi proses görünümüne (TPA0, TPA1, TPA2, TPA3, TPA4), otomatik güncellemeye veya hiçbir kısmi proses görünümünün kullanımına atayabilirsiniz. Proses görünümlerinin ve kısmi proses görünümlerinin bir açıklamasını, "Kullanıcı programının yürütülmesi (Sayfa 60)" başlığında bulabilirsiniz.



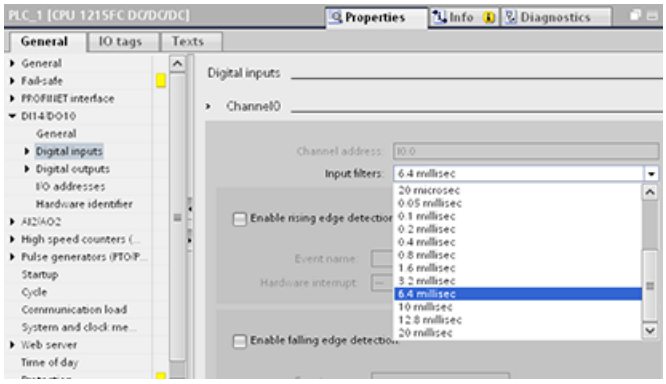
6.6.3 Dijital girişler için filtre sürelerinin ayarlanması

Dijital girişlerin filtreleri, giriş sinyallerindeki örn. kontak titremesinin veya gürültülerin neden olabileceği istenmeyen hızlı değişikliklere tepki göstermeye karşı programınızı korur. Ön ayarlı 6,4 msn'lik filtre süresi, tipik mekanik kontakların neden olduğu istenmeyen anahtarlama işlemlerini engeller. Uygulamanızın belirli fonksiyonları, hızlı sensörlerin girişlerini algılamak ve bunlara tepki göstermek için daha kısa filtre süreleri veya yavaş kontak titremelerini veya daha uzun darbe gürültüsünü filtrelemek için daha uzun filtre süreleri gerektirebilir.

6,4 msn'lik bir giriş filtre süresi, algılanmak için '0'dan '1'e veya '1'den '0'a tek bir sinyal değişiminin yaklaşık 6,4 msn boyunca mevcut olması gerektiği ve 6,4 msn'den daha kısa bir süre boyunca mevcut olan münferit yüksek veya düşük darbelerin algılanmayacağı anlamına gelir. Bir giriş sinyali '0' ile '1' arasındaki ayarlanan filtre süresinden daha hızlı anahtarlama yaparsa, eski değerlerle karşılaştırıldığında yeni değerlere sahip darbelerin toplam süresinin filtre süresini aşması durumunda kullanıcı programındaki girişin değeri değişebilir.

Dijital girişler için kullanılan filtre şu şekilde çalışır:

- "1" değeri girilirse, süre ileriye doğru sayılır ve filtre süresine ulaşıldığında durdurulur. Proses görünümündeki nokta, filtre süresine ulaşılırsa "0" iken "1" olarak değişir.
- "0" değeri girilirse, süre geriye doğru sayılır ve "0" değerine ulaşıldığında durdurulur. Proses görünümündeki nokta, bir "0" değerine ulaşılırsa "1" iken "0" olarak değişir.
- Giriş salınım yaparsa, süre kısmen ileriye ve kısmen geriye doğru sayılır. Sayma değerlerinin net toplamı filtre süresine veya "0" değerine ulaşırsa proses görünümü değişir.
- Bu durumda, "1"den daha fazla "0" değeri olan hızlı değişen bir sinyal, muhtemelen "0"a gider; "0"dan daha fazla "1" değeri mevcutsa, proses görünümü kaydı muhtemelen "1" olur.



Her giriş, tüm amaçlar için geçerli bir filtre konfigürasyonuna sahiptir: Proses girişleri, alarmlar, Darbe toplama (Sayfa 149) ve HSC girişleri. Giriş filtresi için süreleri ayarlamak için "Dijital girişleri" seçin.

Dijital girişler için standart filtre süresi 6,4 msn'dir. "Giriş filtresi" açılır listesinden bir filtre süresi seçebilirsiniz. Geçerli filtre süreleri 0,1 µsn ile 20,0 msn arasındadır.

⚠ İKAZ

Dijital giriş kanalları için filtre süresinin değiştirilmesinde ortaya çıkan tehlikeler

Bir dijital giriş kanalının filtre süresini değiştirirseniz, filtrenin yeni girişlere eksiksiz olarak tepki göstermesi için muhtemelen 20 msn'ye kadar bir süre boyunca giriş sinyali seviyesinin yeni bir geçiş değeri mevcut olmalıdır.

Giriş sinyali seviyesinin kısa geçişleri (20,0 msn'den daha kısa) muhtemelen algılanmadığı için, makine veya proses beklenmedik bir şekilde çalışabilir ve ölüme, yaralanmalara veya maddi hasarlara yol açabilir.

Yeni filtre süresinin hemen etkili olması için CPU'yu kapatın ve açın.

Filtre sürelerinin HSC'ler olarak kullanılan dijital girişler için konfigürasyonu

Hızlı sayaçlar (HSC'ler) olarak kullanılan girişlerde, giriş filtresi süresini, saymanın devre dışı kalma durumlarını önleyeceğiniz uygun bir değere ayarlamanız gerekir.

Siemens aşağıdaki ayarları önerir:

Hızlı sayacın tipi	Önerilen giriş filtresi süresi
1 MHz	0,1 mikro saniye
100 kHz	0,8 mikro saniye
30 kHz	3,2 mikro saniye

6.6.4 Darbe toplama

S7-1200 CPU, dijital girişler için darbe toplama fonksiyonunu sunar. "Darbe toplama" fonksiyonuyla, dijital girişlerin bir çevrimin başında okunması durumunda CPU tarafından kolayca gözden kaçırılabilen çok kısa bir süreye sahip yüksek veya düşük darbeleri kaydedebilirsiniz.

Darbe toplamanın etkinleştirilmesi

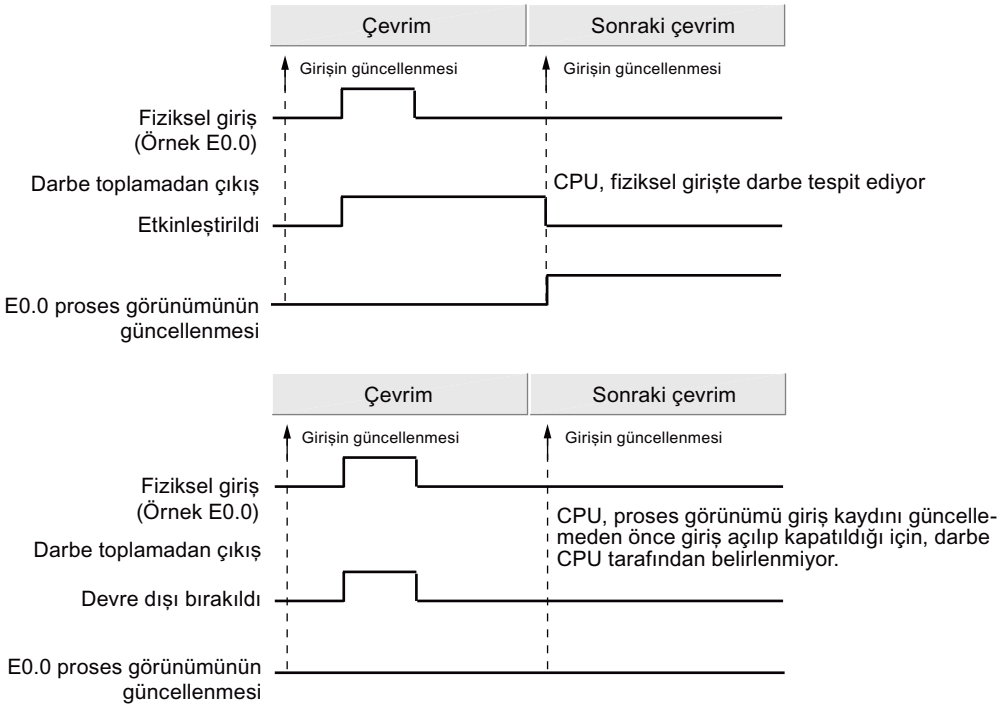
Darbe toplama fonksiyonunu bir dijital giriş için etkinleştirmek üzere aşağıdaki adımları uygulayabilirsiniz:

1. Denetçi penceresinde "Dijital girişler" öğesini seçin.
2. İstediğiniz kanalı seçin.
3. "Darbe toplamayı etkinleştir" öğesini seçin.

Darbe toplamanın temel çalışma şekli

Darbe toplama fonksiyonunu bir giriş için etkinleştirirseniz, girişteki bir durum değişimi, girişin bir sonraki güncellenmesine kadar kaydedilir ve tutulur. Bununla birlikte, CPU girişleri okuyana kadar sadece kısa süreli bir darbenin kaydedilmesi ve tutulması sağlanır.

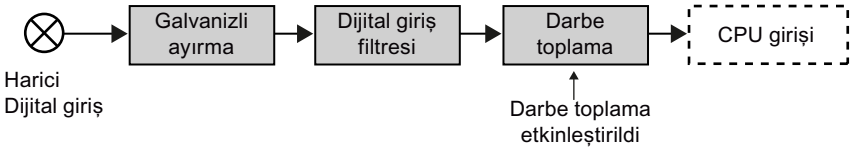
Aşağıdaki şekillerde, S7-1200 CPU'sunun, darbe toplama etkinleştirilmiş ve etkinleştirilmemiş durumdaki davranışı gösterilmektedir:



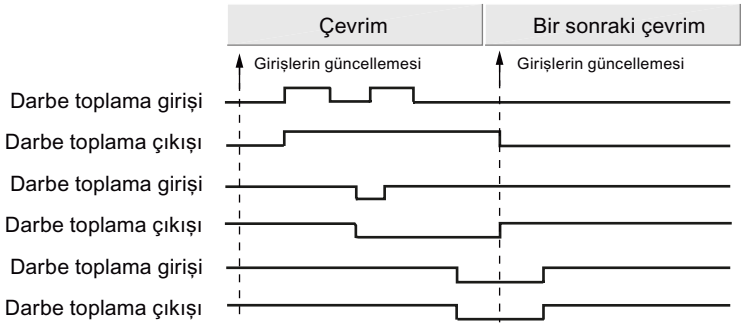
NOT

Darbe toplama fonksiyonu giriş üzerinde ancak giriş filtresinin uygulanmasından sonra etkili olduğu için, giriş filtresi süresi, darbe filtrelenmeyecek şekilde ayarlanmalıdır.

Aşağıdaki resimde, dijital giriş devresinin şematik bir gösterimi yer almaktadır:



Aşağıdaki resimde, etkinleştirilmiş durumdaki darbe toplama fonksiyonunun çeşitli giriş koşullarına tepkisi gösterilmektedir. Belirli bir çevrimde birden fazla darbe mevcutsa sadece ilk darbe okunur. Bir çevrimde birden fazla darbenin olması durumunda yükselen/alçalan kenarlar için kesme olaylarını ayarlamanız gerekir:



6.7 Çok dilli desteğin yapılandırılması

Çok dilli destek ayarları, S7-1200 Web sunucusunun (Sayfa 796) her kullanıcı arayüzü dili için iki proje dili atamanıza olanak sağlar. Bir kullanıcı arayüzünün dili için hiçbir proje dili belirlememe olanağı da mevcuttur.

Proje dili nedir?

Proje dili, TIA Portal'ın, ağ yorumları ve blok yorumları gibi kullanıcı tanımlı proje metinlerini görüntülediği dildir.

Proje gezgininde işaretlenmiş proje için proje dillerini TIA Portal'da **Araçlar > Proje dilleri** menü komutu aracılığıyla seçebilirsiniz.

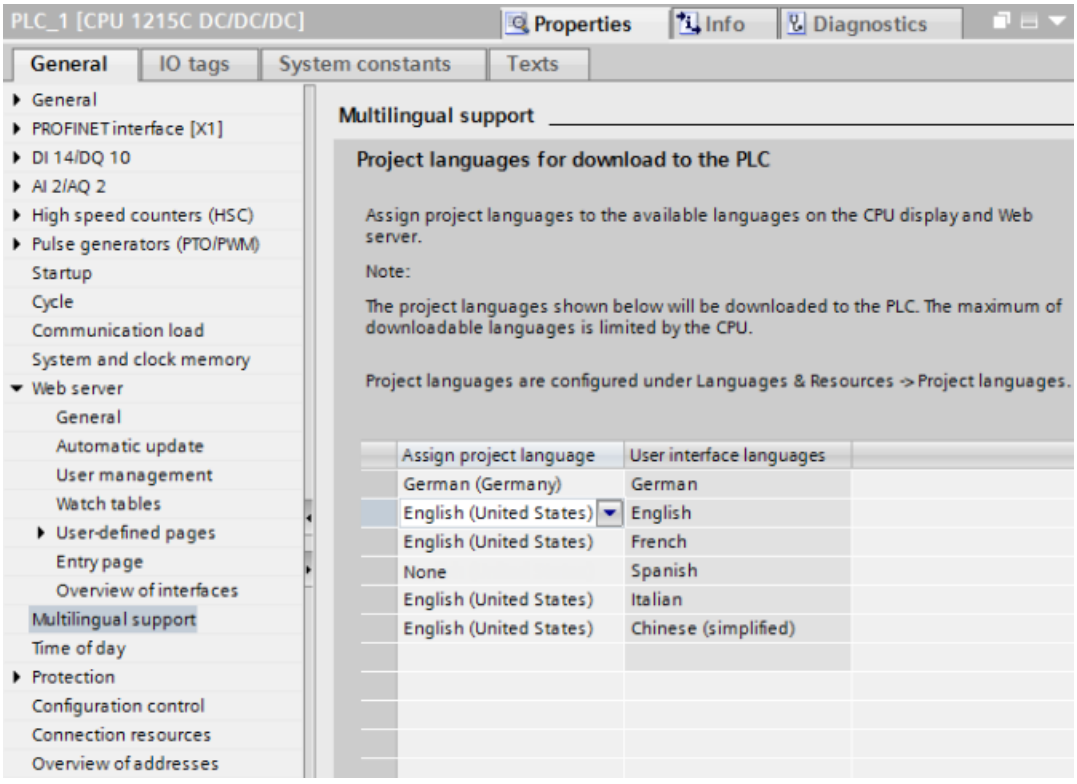
Ardından, **Araçlar > Proje metinleri** menü komutu aracılığıyla, örn. her proje dilinde ağ yorumları gibi kullanıcı metinlerini yapılandırabilirsiniz. TIA Portal için kullanıcı arayüzü dilini değiştirirseniz; ağ yorumları, blok yorumları ve diğer çok dilli proje metinleri ilgili proje dilinde görüntülenir. TIA Portal'ın kullanıcı arayüzü dilini **Opsiyonlar > Ayarlar** menü komutu aracılığıyla ilgili proje diline ayarlayabilirsiniz.

Proje dilleri ve proje metinleri, proje gezginin **Diller ve kaynaklar** düğümleri aracılığıyla da yapılandırılabilir.

Web sunucusu, STEP 7 için teşhis tamponu mesajlarını görüntülemek amacıyla bir veya iki proje dili kullanabilir.

Web sunucusunun proje dili ve kullanıcı arayüzü dili

Web sunucusu, TIA Portal ile aynı kullanıcı arayüzü dillerini destekler, bununla birlikte maksimum iki proje dili kullanmak mümkündür. Web sunucusunu, Web sunucusunun kullanıcı arayüzü diline bağlı olarak teşhis tamponu mesajları için iki proje dilinden biri kullanılacak şekilde yapılandırabilirsiniz. Bu ayarları, CPU'nun cihaz konfigürasyonunda "Çok dilli destek" için kullanılan özellikler aracılığıyla yapılandırabilirsiniz (ağ yorumları, blok yorumları ve diğer çok dilli metinler Web sunucusundan görülmez).



Çok dilli destek için kullanılan özelliklerde, sağ tarafta yer alan kullanıcı arayüzü dilleri düzenlenemez. Burada, hem TIA Portal hem de Web sunucusunun kullanıcı arayüzü için kullanılabilen ön tanımlı diller söz konusudur. "Proje dilini ata" ayarı yapılandırılabilir, yapılandırılmış iki proje dilinizden birini veya "Yok" öğesini seçin. S7-1200 CPU'su sadece iki proje dilini desteklediği için, proje dili, kullanıcı arayüzü diliyle aynı dil olamaz. Bu, desteklenen tüm kullanıcı arayüzü dilleri için geçerlidir.

Aşağıda gösterilen konfigürasyonda, Web sunucusu teşhis tamponu girişleri (Sayfa 817), Web sunucusunun kullanıcı arayüzü Almanca ise görüntülenir, ancak Web sunucusunun kullanıcı arayüzü İspanyolcaya ayarlandıysa teşhis tamponu olayları için hiçbir metin görüntülenmez. Diğer tüm diller için teşhis tamponu girişleri İngilizce dilinde görüntülenir.

6.8 Koruma ve dijital güvenlik

6.8.1 Dijital güvenlik asistanında PLC dijital güvenlik ayarlarının belirlenmesi

TIA Portal'daki dijital güvenlik asistanı, PLC için kullanılan dijital güvenlik ayarlarının yapılandırılması amacıyla merkezi bir yer sunar. Projenize bir S7-1200 CPU V4.x (Sayfa 128) eklerseniz, TIA Portal, dijital güvenlik asistanını çağırır.

Dijital güvenlik asistanı dört bölümden oluşur:

- Gizli PLC verilerinin korunması
- PG/PC ve HMI iletişimi için kullanılan işletim türü
- PLC erişim koruması
- Genel bakış

STEP 7, asistanda "Tamamla" butonuna tıkladığınızda, asistanda yaptığınız ayarları projeye kaydeder. "İptal" butonuna tıklarsanız, STEP 7 yaptığınız değişiklikleri kaydetmez. Asistanda yaptığınız değişiklikler sadece STEP 7 projesinde etkili olur.

Gizli PLC verilerinin korunması

"Gizli PLC konfigürasyon verilerini koruma" fonksiyonuyla, projenizdeki her CPU'yu münferit olarak koruyabilirsiniz. Dijital güvenlik asistanında bu korumayı etkinleştirebilir ve gizli PLC konfigürasyon verilerinin korunması için bir şifre belirleyebilirsiniz.

- Cihaz bu şifreye sahip değilse, hedef sisteme ilk yükleme sırasında TIA Portal, gizli PLC konfigürasyon verilerinin korunması için tanımlanan şifreyi girmenizi ister.
- Cihaz bu şifreye zaten sahipse, STEP 7 projesindeki şifre ile cihazdaki şifre aynı olmalıdır. Şifreler aynı değilse, projeyi CPU'ya yükleyemezsiniz. Gizli PLC konfigürasyon verilerinin korunması için kullanılan şifreyi silebilir veya bunun için cihazdakiyle aynı şifreyi belirleyebilirsiniz.

Gizli PLC konfigürasyon verilerinin korunması özelliğini, CPU'nun cihaz konfigürasyonunda da (Sayfa 154) yapılandırabilirsiniz.

PG/PC ve HMI iletişimi için kullanılan işletim türü

PG/PC ve HMI iletişimi için kullanılan işletim türü, CPU'nuz ile diğer cihazlar arasındaki iletişimin korunması için bir PLC iletişim sertifikası kullanmanıza olanak sağlar:

- TIA Portal ve SIMATIC Automation Tool gibi programlama cihazları (PG'ler)
- HMI cihazları

Sadece güvenli iletişime izin vermek için asistanda "Sadece güvenli PG/PC ve HMI iletişimine izin ver" öğesini seçin.

Güvenli iletişimi desteklemeyen cihazlarla iletişim kurmanız gerekirse, "Sadece güvenli PG/PC ve HMI iletişimi" öğesini seçin. Bu öğe seçildiğinde, PLC'niz, Güvenli iletişim üzerinden veya eski tip iletişim üzerinden iletişim kurabilir.

PG/PC ve HMI iletişimi için kullanılan işletim türünü (Sayfa 158), CPU'nun cihaz konfigürasyonundaki bağlantı mekanizmalarında da yapılandırabilirsiniz.

PLC erişim koruması

Dijital güvenlik asistanıyla, CPU'nun koruma kademesi şifrelerini de (Sayfa 156) belirleyebilirsiniz. Bu, cihaz konfigürasyonundakiyle aynı koruma seviyesi konfigürasyonudur. Dijital güvenlik asistanı, basitleştirilmiş erişim olanağı sunar.

Genel bakış

Dijital güvenlik asistanındaki genel bakış, aşağıdaki alanlarla ilgili ayarlarınızı gösterir:

- Gizli PLC verilerinin korunması (Sayfa 154)
- PG/PC iletişimi ve HMI iletişimi için kullanılan işletim türü (Sayfa 158)
- PLC erişim koruması (Sayfa 156)

Ayarlarınızı kontrol edin ve gerekirse değişiklikler yapmak için "Geri" butonunu kullanın. Yaptığınız ayarlardan memnunsanız, "Tamamla" butonuna tıklayın. STEP 7, yaptığınız ayarları projeye kaydeder.

Dijital güvenlik asistanının, CPU'nun cihaz konfigürasyonundan çağrılması

Alternatif olarak, "Koruma ve dijital güvenlik" alt bölümünde CPU'nun cihaz konfigürasyonu üzerinden dijital güvenlik asistanını manuel olarak çağırabilirsiniz.

6.8.2 Gizli PLC konfigürasyon verilerinin korunması

"Gizli PLC konfigürasyon verilerinin korunması" fonksiyonuyla projenizdeki her CPU'nun konfigürasyonunu koruyabilirsiniz. Cihaz konfigürasyonunun "Koruma ve dijital güvenlik" alt bölümünde bu koruma özelliğini etkinleştirebilir ve gizli PLC konfigürasyon verilerinin korunması için bir şifre belirleyebilirsiniz.

Gizli PLC konfigürasyon verilerinin korunması için koruma belirlerken dikkat etmeniz gerekenler:

- Cihaz bu şifreye sahip değilse, hedef sisteme ilk yükleme sırasında TIA Portal, gizli PLC konfigürasyon verilerinin korunması için tanımlanan şifreyi girmenizi ister.
- Cihaz bu şifreye zaten sahipse, STEP 7 projesindeki şifre ile cihazdaki şifre aynı olmalıdır. Şifreler aynı değilse, projeyi CPU'ya yükleyemezsiniz. Gizli PLC konfigürasyon verilerinin korunması için kullanılan şifreyi silebilir veya bunun için cihazdakiyle aynı şifreyi belirleyebilirsiniz. Cihazdaki şifreyi "Çevrimiçi ve teşhis (Sayfa 1121)" öğesi üzerinden belirleyebilir ve silebilirsiniz.

Dijital güvenlik asistanı

Bu fonksiyonu, dijital güvenlik asistanında (Sayfa 153) da etkinleştirebilir ve şifre belirleyebilirsiniz. Bir CPU V4.x'i ilk defa eklerken dijital güvenlik asistanı başlatılır. Dijital güvenlik asistanını, cihaz konfigürasyonunun "Koruma ve dijital güvenlik" alt bölümü üzerinden de çağırabilirsiniz.

Gizli PLC konfigürasyon verilerini korumanın avantajı

V4.x CPU'lar, TIA Portal ile birlikte her bir CPU için koruma sağlar. Gizli PLC konfigürasyon verilerinin korunması, her PLC'nin proje belleği için daha yüksek seviyede güvenlik sağlar.

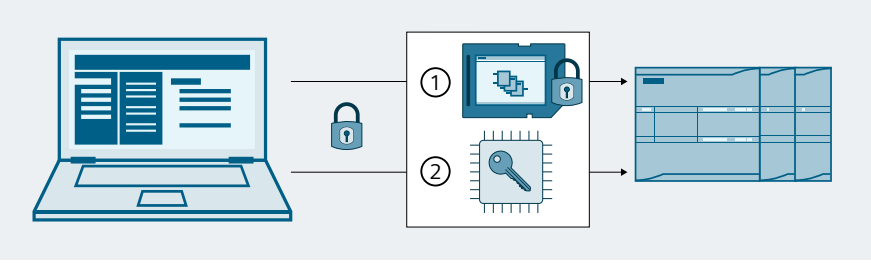
Korumanın çalışma şekli

Koruma, anahtarlı bir kilit gibi çalışır. Gizli PLC konfigürasyon verilerinin korunmasını TIA Portal'da etkinleştirebilir ve bu koruma için bir şifre belirleyebilirsiniz. Projenin hedef sisteme yüklenmesiyle, CPU'da "Gizli PLC konfigürasyon verilerinin korunması" için kullanılan şifre belirlenir. CPU, şifreyi, projenin hedef sisteme yüklenmesi yoluyla veya proje dosyalarını okumak üzere bir memory card (Sayfa 122) aracılığıyla elde etmek zorundadır. Proje dosyaları, gizli konfigürasyon verilerinizi içerir.

Şifreleme:
Gizli PLC konfigürasyon verilerinin korunması özelliğini TIA Portal'da etkinleştirebilir ve bir şifre belirleyebilirsiniz. Bu şifre, gizli PLC konfigürasyon verilerinin korunması için kullanılan araçtır.

Şifre çözme:
Projeyi CPU'ya veya alternatif olarak bir memory card'dan (Sayfa 122) yükleyebilirsiniz.
CPU, şimdi proje dosyalarını okuyabilir (kilit açma).

Bu sayede, projedeki şifrenin şifrelenmesi ve ardından CPU'daki şifre çözme işlemi, gizli PLC konfigürasyon verileriniz için yüksek seviyede bir koruma türü sağlar.



① Şifre korumalı gizli verileri içeren proje, CPU'nun memory card'da

② CPU'nun bellek alanında şifre tarafından oluşturulan ve korunan gizli konfigürasyon verilerine erişimi etkinleştiren anahtar bilgiler.

NOT

CPU'larla ilgili sorumluluk

Gizli PLC konfigürasyon verileri için koruma özelliğini yapılandırdıysanız ve bunu bir CPU'ya yüklediyseniz, devre dışı bırakıldığında CPU'nun güvenli bir şekilde tasfiye edilmesini sağlayın.

Gizli PLC konfigürasyon verileri için koruma özelliğini yapılandırdıysanız ve bunu bir CPU'ya yüklediyseniz ve daha sonra STEP 7 projesindeki CPU'nun Firmware sürümünün V4.5'ten önce olduğu bir projeyi CPU'ya yüklerseniz, CPU yine de şifrelenmiş şifreyi içerir. Devre dışı bırakıldığında bu CPU'nun güvenli bir şekilde tasfiye edilmesini sağlayın.

Devre dışı bırakılan CPU'ları güvenli şekilde tasfiye ederek, hiçbir üçüncü şahsın korunan gizli konfigürasyon verilerine erişemeyeceğinden emin olun.

Çevrimiçi araçlar

CPU çevrimiçi ise, Gizli PLC konfigürasyon verilerinin korunması için kullanılan şifreyi "Çevrimiçi ve teşhis" aracında da belirleyebilir, silebilir veya değiştirebilirsiniz (Sayfa 1121).

Daha fazla bilgi

Fonksiyonlar ve uygulamayla ilgili daha fazla bilgiyi, TIA Portal'ın bilgi sistemindeki "Güvenli iletişim" bölümünde bulabilirsiniz.

Ayrıca bakınız

Gizli PLC konfigürasyon verilerini koruyan bir CPU'nun değişimi (Sayfa 1365)

6.8.3 CPU için erişim koruması

CPU, belirli fonksiyonlara erişimi kısıtlamak için çeşitli güvenlik kademeleri sunar. Koruma kademesi ve bir CPU için kullanılacak şifre ayarlayarak, şifre girmeden erişilebilen fonksiyonları ve bellek alanlarını kısıtlayabilirsiniz.

Her erişim kademesi, bir şifre girilmeden de belirli fonksiyonlara kısıtlamasız şekilde erişim olanağı sunabilir. CPU için standart ayar, "Erişim yok (komple koruma)" şeklindedir. Bir koruma kademesi kullanmak üzere, ilgili kademe için şifre girmeniz gerekir.

Bir ağ şifresi girerseniz, bu şifre, CPU'nun şifre korumasını etkilemez. Şifre koruması, iletişim fonksiyonları da dahil olmak üzere, kullanıcı programının yordamlarının yürütülmesi için geçerli değildir. Doğru şifrenin girilmesi, ilgili kademenin tüm fonksiyonlarına erişime olanak sağlar.

CPU'lar arasındaki iletişim (kod bloklarındaki iletişim fonksiyonları üzerinden), CPU'nun koruma kademesi tarafından kısıtlanmaz.

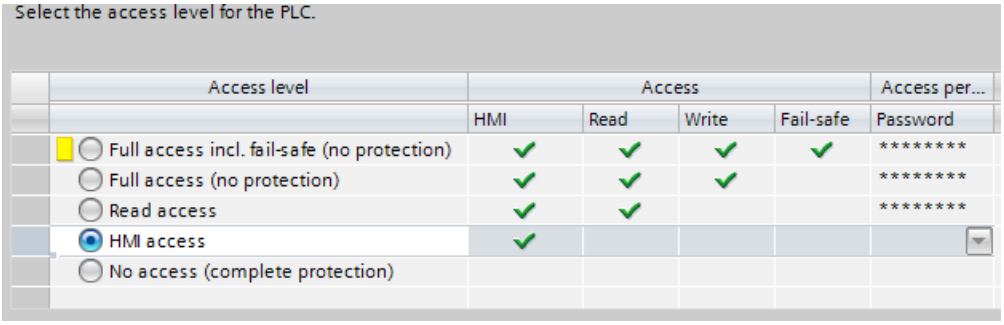
Tablo 6-3 CPU'nun koruma kademeleri

Koruma kademesi	Erişim kısıtlamaları
Hata güvenli özelliği de dahil üzere tam erişim (koruma yok)	Bir F-CPU'ya şifre koruması olmadan kısıtlamasız erişim.
Tam erişim (koruma yok)	Bir standart CPU'ya şifre koruması olmadan kısıtlamasız erişim.
Okuma erişimi	Şifre girilmeden donanım konfigürasyonuna ve bloklara sadece okuma erişimine izin verir. Donanım konfigürasyonunu ve blokları programlama cihazına yükleyebilirsiniz. Ayrıca HMI erişimi ve teşhis verilerine erişim mümkündür. Çevrimdışı/çevrimiçi karşılaştırma sonuçlarını görüntüleyebilir, işletim durumunu (RUN/STOP) değiştirebilir ve saati ayarlayabilirsiniz. CPU'ya hiçbir blok ve hiçbir donanım konfigürasyonu yükleyemezsiniz. Firmware güncellemeleri mümkün değildir.
HMI erişimi	Sadece HMI erişimine izin verilir. HMI erişimi, HMI panellerine erişimin yanı sıra, çoğu çevrimiçi fonksiyonun kullanımına da olanak sağlar. Bu konuyla ilgili ayrıntıları, TIA Portal bilgi sisteminde bulabilirsiniz.
Erişim yok (komple koruma)	Şifresiz erişime izin vermez. Tanımlama verilerini sadece görüntüleyebilirsiniz, örn. "Erişilebilir katılımcılar".

Her koruma kademesinde CPU için bir (geçici) acil IP adresi (Sayfa 762) ayarlayabilirsiniz.

Şifrelerde büyük ve küçük harf yazımına dikkat edilir. Koruma kademesini ve şifreleri yapılandırmak için aşağıdaki yöntemi izleyin:

1. "Cihaz konfigürasyonu"nda CPU'yu seçin.
2. Denetçi penceresinde "Özellikler" sekmesini seçin.
3. Erişim kademesini seçmek ve şifreleri girmek için "Koruma ve dijital güvenlik" özelliğini seçin.



V4.x sürümündeki bir S7-1200 CPU'sunu değiştirirseniz, "Şifre şifrelemesini güncelle" butonuyla, erişim kademesi için kullanılan mevcut şifrelerin kayıt formatını güncelleyin.

Bu konfigürasyonu CPU'ya yüklerseniz, kullanıcı, HMI erişimine sahip olur ve şifre kullanmadan HMI fonksiyonlarına erişebilir. Kullanıcı, verileri okumak veya çevrimdışı/çevrimiçi kod bloklarını karşılaştırmak üzere, "Okuma erişimi" için yapılandırılmış olan şifreyi veya "Tam erişim (koruma yok)" şifresini girmek zorundadır. Kullanıcı, verileri yazmak üzere, "Tam erişim (koruma yok)" için yapılandırılmış olan şifreyi girmek zorundadır. Kullanıcı, hata güvenli CPU'lara tam erişim için, "Hata güvenli özelliği de dahil tam erişim (koruma yok)" için yapılandırılmış olan şifreyi girmek zorundadır.

⚠ İKAZ

Korunan bir CPU'ya yetkili olmayan kişilerin erişimi

Tam CPU erişim yetkilerine veya hata güvenli özelliği de dahil tam erişim olanağına sahip kullanıcılar PLC değişkenlerini okuyabilir ve yazabilir. Web sunucusu kullanıcıları, CPU için kullanılan erişim kademesinden bağımsız olarak PLC değişkenlerini okuma ve yazma yetkilerine sahip olabilir. CPU'ya yetkisi olmayan kişiler tarafından erişim veya PLC değişkenleri için geçersiz değerlerin ayarlanması, proses işletimine zarar verebilir ve hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yetkili kullanıcılar, işletim durumunda değişiklikler yapabilir, PLC verilerini yazabilir ve Firmware güncellemeleri uygulayabilir. Siemens, aşağıdaki güvenlik önlemlerine uyulmasını önerir:

- CPU erişim kademelerini ve Web sunucusu kullanıcı ID'lerini (Sayfa 801) STEP 7'de tanımlandığı şekilde güçlü şifreler aracılığıyla koruyun.
- Web sunucusuna erişimi sadece HTTPS protokolü üzerinden etkinleştirin.
- Web sunucusu kullanıcısının "Herkes" şeklindeki standart asgari yetkilerini genişletmeyin.
- Web sitelerinin kullanıcıları PLC değişkenleri için geçersiz değerler belirleyebileceği için, program lojiğinizdeki değişkenler için bir hata kontrolü ve bir aralık kontrolü gerçekleştirin.

6.8.4 Bağlantı mekanizmalarının yapılandırılması

6.8.4.1 Uzak partnerler için erişim mekanizmasının belirlenmesi

Kullanıcı, uzak bağlantı partnerlerine PUT/GET yordamları aracılığıyla erişim için de uygun yetkiye ihtiyaç duyar.

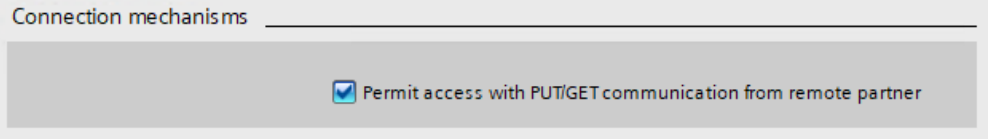
"PUT/GET iletişimi üzerinden etkileşime izin ver" opsiyonu, standart olarak etkinleştirilmemiş durumdadır. Bu durumda CPU verilerine okuma ve yazma erişimi sadece, hem yerel CPU'nun hem de iletişim ortağının yapılandırılmış veya programlanmış olması gereken iletişim bağlantıları ile mümkündür. Örneğin BSEND/BRCV yordamları üzerinden erişim mümkündür.

Bu nedenle, yerel CPU'nun sadece bir sunucu olduğu (yani yerel CPU'da iletişimin iletişim ortağı ile hiçbir konfigürasyonu/programlaması mevcut değildir) bağlantıların kullanımı, CPU'nun çalıştırılması sırasında mümkün değildir. Bu, örneğin aşağıdakiler için geçerlidir:

- PUT/GET, FETCH/WRITE veya FTP aracılığıyla iletişim modülleri üzerinden erişim
- Başka S7-CPU'larının PUT/GET erişimi
- PUT/GET iletişimi üzerinden HMI erişimi

CPU verilerine istemci tarafında erişime izin vermek istiyor ve bu sayede CPU'nun iletişim hizmetlerini kısıtlamak istemiyorsanız aşağıdaki yöntemi izleyin:

1. Erişim koruması için "Erişim yok (komple koruma)"dışında herhangi bir koruma kademesini yapılandırın.
2. "PUT/GET iletişimi üzerinden etkileşime izin ver" opsiyon kutusunu etkinleştirin.



Bu konfigürasyonu CPU'ya yüklerseniz, CPU, uzaktaki partnerlerin PUT/GET iletişimine izin verir.

6.8.4.2 Güvenli PG/PC iletişimi ve HMI iletişimini etkinleştirme ve sertifikalar oluşturma

CPU'nun yalnızca güvenli iletişimi mi kabul edeceğini veya hem güvenli iletişim hem de eski tip iletişim aracılığıyla mı iletişim kurabileceğini yapılandırmak için CPU'nuzun cihaz konfigürasyonundaki "bağlantı mekanizmalarını" kullanın. Güvenli iletişim, TLS (Transport Layer Security) 1.3 üzerinden X.509 sertifikalarını kullanır. CPU, bu sertifikaları CPU ile istemciler arasında güvenli iletişim sunmak için kullanır. İstemciler aşağıdakiler olabilir:

- TIA Portal
- SIMATIC Automation Tool
- HMI cihazları

PG/PC ve HMI için eski tip iletişimi devre dışı bırakmak üzere "Sadece güvenli PG/PC ve HMI iletişimine izin ver" öğesini seçin.

Kendi sertifikalarınızı da oluşturabilirsiniz. CPU için yeni bir sertifika eklemek veya mevcut bir sertifikayı seçmek için PLC iletişim sertifikasının yanındaki "..." öğesine tıklayın. TIA Portal'daki bilgi sisteminde "Sertifikalar oluşturma/sertifikaları uzatma" başlığının altında, sertifikaların konfigürasyonu için kullanılan parametrelerle ilgili daha fazla bilgi bulabilirsiniz.

Cihaz konfigürasyonunun "Koruma ve dijital güvenlik" altındaki konfigürasyon alt bölümleri, ekranda size sunulan dijital güvenlik opsiyonları içinde rehberlik sağlar. Bu alt bölümlerde, TIA

Portal'daki bilgi sisteminde yer alan ve her konfigürasyon görevi ve ilgili güvenlik konseptleriyle ilgili konulara bağlantıları da bulabilirsiniz.

Eski tip iletişim

Güvenli iletişimi desteklemeyen bir cihazla iletişim kurmanız gerekirse, "Sadece güvenli PG/PC ve HMI iletişimi" öğesini seçin. Bu öğe seçildiğinde, PLC'niz, Güvenli iletişim üzerinden veya eski tip iletişim üzerinden iletişim kurabilir.

TIA Portal'da, standart değer olarak güvenli iletişimin en yüksek kademesi kullanılır. Ancak işletime almayla ilgili nedenlerden dolayı, çevrimiçi menüde "Sadece eski tip PG/PC iletişimini kullan" öğesini seçerek TIA Portal'ı eski tip PG/PC iletişimini kullanmaya zorlayabilirsiniz.

Dijital güvenlik asistanı

Dijital güvenlik asistanında (Sayfa 153) V4.x ve daha yeni CPU'lar için "PG/PC ve HMI iletişimi" fonksiyonunu da yapılandırabilirsiniz.

⚠ İKAZ

İşletime alma sırasındaki potansiyel güvenlik riskleri

CPU, işletime alma sırasında, bir bağlantı kurmak için güvenmeniz gereken kendinden imzalı bir sertifikayı kullanma sunar. Bu sertifikaya sadece programlama cihazı ve CPU korumalı bir ağda bulunuyorsa ve bunlar birbirlerine doğrudan bağlıysa güvenin.

Saldırganlar, korumasız bir ortamda bu sertifikaları manipüle edebilir ve örneğin Man-in-the-Middle saldırılarıyla programlama cihazı/kumanda cihazı ile CPU arasındaki iletişime erişebilir.

Korumasız bir iletişim ortamı nedeniyle gerçekleştirilen saldırılar proses akışını bozabilir ve ölüme, ağır yaralanmalara veya maddi hasarlara neden olabilir.

6.8.5 Harici yükleme belleği

Dahili yükleme belleğinden harici bir yükleme belleğine (SIMATIC memory card) kopyalama işlemini engelleyebilirsiniz. Dahili yükleme belleğinin harici bir yükleme belleğine kopyalanmasını engellemek için aşağıdaki yöntemi izleyin:

1. STEP 7'de CPU'nun cihaz konfigürasyonunda "Genel" özelliklerinin altındaki "Koruma ve dijital güvenlik" opsiyonunu seçin.
2. "Harici yükleme belleği" alanındaki "Dahili yükleme belleğinden harici yükleme belleğine kopyalamayı devre dışı bırak" opsiyonunu seçin.

Memory card'ın CPU'ya takılması (Sayfa 112) başlığında, bu özelliğin, CPU'ya bir memory card takılmasını nasıl etkilediğiyle ilgili bir açıklama bulabilirsiniz.

6.8.6 Teknik bilgi koruması

Teknik bilgi koruması ile programınızdaki bir veya daha fazla kod bloğunu (OB, FB, FC veya DB) yetkisiz kişilerin erişimine karşı koruyabilirsiniz. Bir kod bloğuna erişimi sınırlamak için bir şifre girebilirsiniz. Şifre koruması, kod bloğunun yetkisiz kişiler tarafından okunmasını veya değiştirilmesini engeller. Şifre olmadan kod bloğuna sadece aşağıdaki bilgiler kaydedilebilir:

- Blok başlığı, yorum ve modül özellikleri
- Aktarma parametreleri (IN, OUT, IN_OUT, geri dönüş)
- Program çağrı yapısı
- Çapraz referanslardaki global değişkenler (kullanımla ilgili bilgiler olmadan), fakat yerel değişkenler gizlenmiştir

Bir bloğu teknik bilgi koruması için yapılandırırsanız, bu bloktaki koda ancak şifre girildikten sonra erişilebilir.

Bir kod bloğunun teknik bilgi korumasını, ilgili kod bloğunun "Özellikler" görev kartında yapılandırabilirsiniz. Kod bloğunu açtıktan sonra "Özellikler" öğesinin altındaki "Koruma" opsiyonunu seçin.



1. "Teknik bilgi koruması" iletişim kutusunu görüntülemek için "Koruma" butonuna tıklayın.
2. Şifreyi girmek için "Tanımla" butonuna tıklayın.
3. Yeni bir şifre girin ve bu şifreyi onaylayın.
4. İletişim kutusunu kapatmak için "TAMAM" butonuna tıklayın.



6.8.7 Yazma koruması

Bloklar için kullanılan yazma koruması, blokların istenmeden değiştirilmesini engeller. Yazma korumasına sahip bloklar sadece "yazma korumalı" şekilde açılabilir. Ancak blok özellikleri yine de düzenlenebilir.

Bir kod bloğu için yazma korumasını etkinleştirmek için aşağıdaki yöntemi izleyin:

1. Kod bloğunun "Özellikler" görev kartını açın.
2. Kod bloğunun özelliklerini açtıktan sonra "Koruma" öğesini seçin.
3. "Yazma koruması" alanında "Şifre tanımlama" opsiyonunu seçin.
4. "Korumayı tanımla" iletişim alanında "Yeni şifre" ve "Şifreyi yeniden girin" alanlarına şifreyi girin.
5. Yaptığınız girişleri "TAMAM" butonuyla onaylayın.
6. "Yazma koruması" opsiyon kutusunu etkinleştirin.
7. "Erişim koruması" iletişim alanına doğru şifreyi girin.

Bu adımlar uygulandıktan sonra yazma koruması etkinleştirilir ve bloğun bir sonraki açılışında etkili hale gelir.

6.8.8 Kopyalama koruması

Diğer bir güvenlik fonksiyonu, program bloklarını belirli bir memory card'a veya CPU'ya bağlamaktır. Bu fonksiyon, öncelikle fikri mülkiyetin korunması için faydalıdır. Bir program bloğu belirli bir cihaza bağlanırsa, programın veya kod bloğunun kullanımı sadece belirli bir memory card veya CPU ile mümkün olur. Bu fonksiyon, bir program veya kod bloğunu elektronik olarak (örn. internet üzerinden veya e-posta yoluyla) veya bir memory card göndererek dağıtmanıza olanak sağlar. OB'ler (Sayfa 172), FB'ler (Sayfa 174) ve FC'ler (Sayfa 174) için kopyalama koruması mevcuttur. S7-1200 CPU blok korumasının üç türünü destekler:

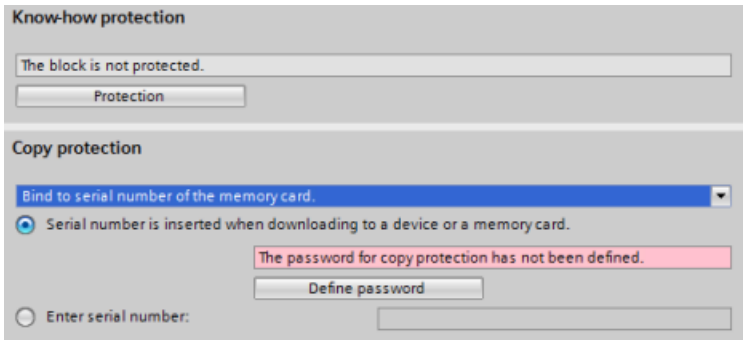
- Bir CPU'nun seri numarasıyla bağlantı
- Bir memory card'ın seri numarasıyla bağlantı
- Zorunlu bir şifreyle dinamik bağlantı

Bir bloğu belirli bir CPU veya memory card'a bağlamak için ilgili kod bloğunun "Özellikler" görev kartını açın.

1. Kod bloğunu açtıktan sonra "Koruma" öğesini seçin.



2. "Kopyalama koruması" açılır listesinden kullanılacak olan kopyalama korumasının türünü seçin.



3. Bir CPU'nun veya memory card'ın seri numarasıyla bağlantı için, seri numarasının yükleme sırasında mı ekleneceğini, yoksa seri numarasının memory card veya CPU tarafından mı girileceğini seçebilirsiniz.

NOT

Seri numarasında büyük ve küçük harf yazımına dikkat edilmelidir.

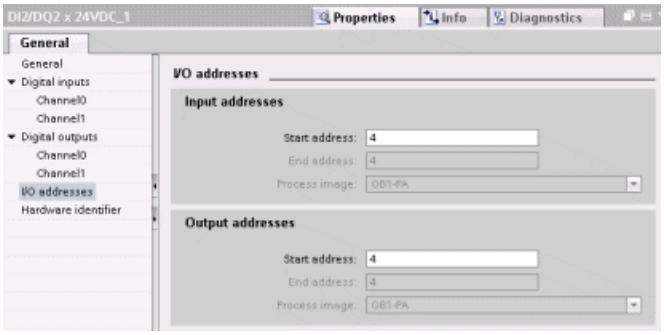
Zorunlu bir şifreyle dinamik bağlantı için, bloğun yüklenebilmesi veya kopyalanabilmesi için gereken şifre belirlenmelidir.
Daha sonra dinamik bağlantılı bir blok yüklenirse (Sayfa 194), yükleme işlemine olanak sağlamak için bu şifrenin girilmesi gerekir. Kopyalama koruması ve teknik bilgi koruması (Sayfa 159) için kullanılan şifrelerin farklı olduğuna dikkat edilmelidir.

6.9 Modül parametrelerinin yapılandırılması

Modüllerin işletim parametrelerini yapılandırmak için, modülü cihaz görünümünde seçin ve modül için kullanılan parametreleri ayarlamak üzere denetçi penceresinde "Özellikler" sekmesini açın.

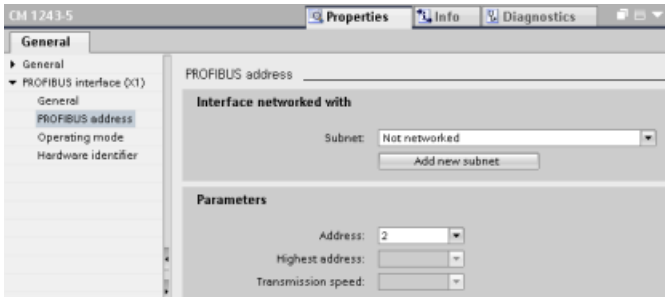
Sinyal modülü (SM) veya sinyal kartının (SB) yapılandırılması

Bir SM'nin veya SB'nin G/Ç'lerini yapılandırmak için bir CPU'nun entegre G/Ç (Sayfa 146)'lerini yapılandırmak için uygulananlarla aynı adımları uygulayın. Sinyal modüllerinin girişlerini, yükselen kenarları algılama (Sayfa 70), alçalan kenarları algılama (Sayfa 70) veya darbe toplama (Sayfa 149) işlemleri için yapılandıramazsınız.



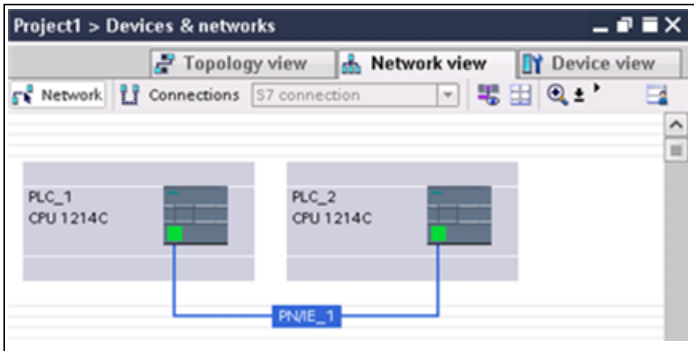
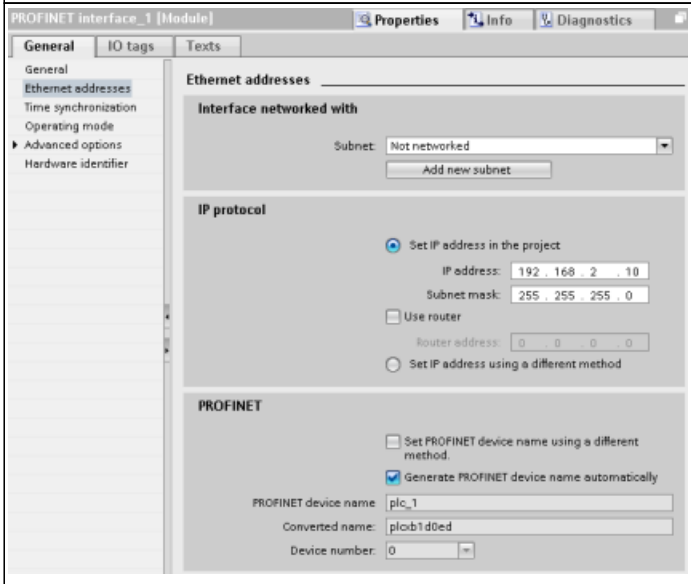
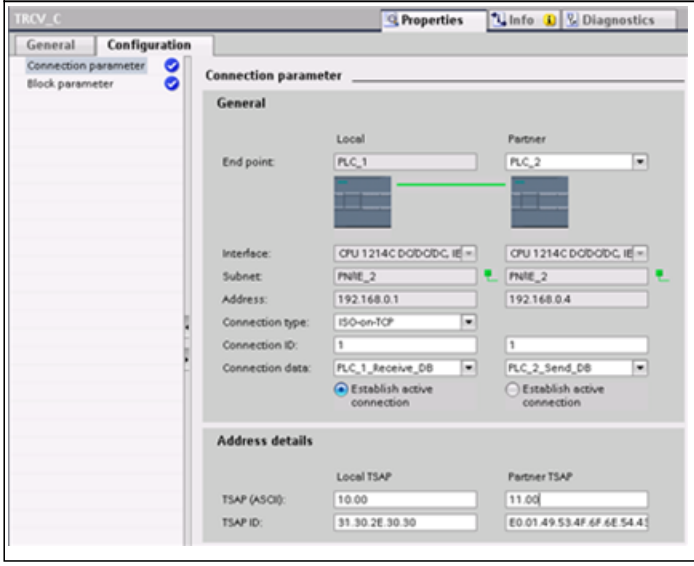
İletişim arayüzünün (CM, CP veya CB) yapılandırılması

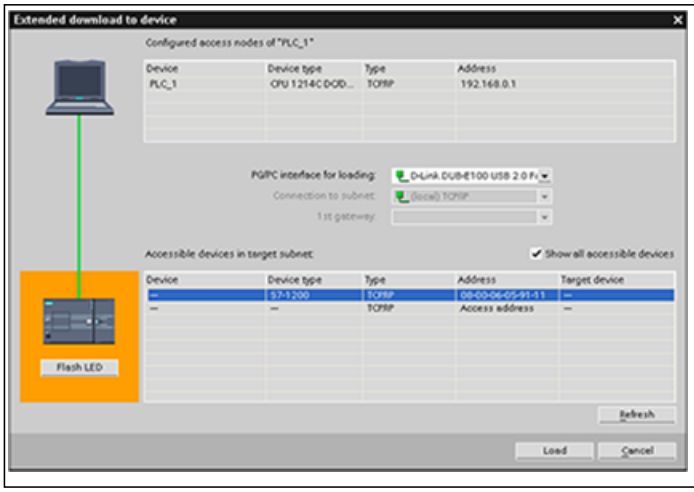
Ağ için kullanılan parametreleri, iletişim arayüzünün tipine bağlı olarak yapılandırın.



6.10 CPU'nun iletişim için yapılandırılması

S7-1200, hem basit hem de karmaşık ağların desteklenmesi yoluyla iletişim ve ağ bağlantısına yönelik gerekliliklerinizi karşılar. S7-1200 ayrıca, örn. kendi iletişim protokolleri olan yazıcılar veya teraziler gibi başka cihazlarla iletişim kurmak için araçlar sunar.

	Cihaz konfigürasyonunun ağ görünümünde, projenizdeki cihazların arasında ağ bağlantıları kurabilirsiniz. Ağ bağlantısı kurulduktan sonra, denetçi penceresinin "Özellikler" sekmesinde ağ parametrelerini yapılandırabilirsiniz. Daha fazla bilgi için bkz. "Bir ağ bağlantısı oluşturma" (Sayfa 562).
	"Özellikler" iletişim kutusunda "Ethernet adresleri" girişini seçin. STEP 7'de, Ethernet adresinin konfigürasyonu için, yazılım projesine, projeye yüklenecek olan CPU'nun IP adresini atayabileceğiniz iletişim kutusu görüntülenir. Not: S7-1200 CPU'sunun önceden yapılandırılmış bir IP adresi yoktur. Bu nedenle CPU'ya manuel olarak bir IP adresi atamanız gerekir. Daha fazla bilgi için bkz. "Internet Protocol (IP) adreslerinin atanması" (Sayfa 566).
	TCP, ISO-on-TCP ve UDP-Ethernet protokollerinde yerel ve orta CPU'nun ve bağlantılarını, yordamın (TSEND_C, TRCV_C veya TCON) "Özellikler" kısmında yapılandırabilirsiniz. Şekilde, bir ISO on TCP bağlantısının "Bağlantı" sekmesindeki "Bağlantı özellikleri" gösterilmektedir. Daha fazla bilgi için bkz. "Yerel CPU ile partner CPU'su arasındaki bağlantı yolunun yapılandırılması" (Sayfa 563).

	Konfigürasyon tamamlandıktan sonra projeyi CPU'ya yükleyin. Tüm IP adresleri, proje yüklenirken yapılandırılır. Daha fazla bilgiyi, "PROFINET ağının test edilmesi" (Sayfa 575) başlığında bulabilirsiniz.

NOT

CPU'nuzla bağlantı için ağ kartınız (NIC) ve aynı ağ sınıfındaki CPU aynı alt ağa ait olmalıdır. Ağ kartınızı CPU'nun ön ayarlı IP adresine göre ayarlayabilir veya CPU'nun IP adresini, ağ sınıfına ve ağ arayüzü kartının alt ağına uygun şekilde değiştirebilirsiniz.

Daha fazla bilgi için bkz. "Internet Protocol (IP) adreslerinin atanması" (Sayfa 566).

6.11 Saat senkronizasyonu

Gerçek zaman saati saat senkronizasyonunun hedefi, diğer tüm Master saatlerin senkronize edilmesi için kullanılan bir Master saate sahip olmaktır. Saatler, Master saat tarafından başlangıçta senkronize edilir ve ardından, zaman içinde birbirine eklenen sapmaların oluşmasını önlemek için düzenli olarak yeniden senkronize edilir.

S7-1200'de ve yerel temel bileşenlerde sadece CPU ve CP modüllerinin bazıları, gerektiğinde senkronize edilmesi gereken bir gerçek zaman saatine sahiptir. CPU'nun gerçek zaman saatini harici bir Master saatle senkronize edebilirsiniz. Harici Master saat, saat bilgisini muhtemelen bir NTP sunucusu veya S7-1200'nin, Master saat içeren bir SCADA sistemine bağlı yerel modül taşıyıcısında bulunan bir CP modülü üzerinden sağlar.

Saat senkronizasyonunu destekleyen diğer tüm S7-1200 CP'ler (https://support.industry.siemens.com/cs/us/en/ps) ile ilgili daha fazla bilgiyi, Siemens'in Industry Online Support web sitesinde bulabilirsiniz.

Gerçek zaman saatinin ayarlanması

S7-1200 CPU'sundaki gerçek zaman saatini beş farklı şekilde ayarlayabilirsiniz:

- NTP sunucusu üzerinden (Sayfa 578)
- STEP 7 üzerinden
- Kullanıcı programı üzerinden
- Bir HMI paneli üzerinden
- SIMATIC Automation Tool ile

CP modüllerinin saat senkronizasyonunu CPU'nun saati aracılığıyla, aşağıdaki şekilde gösterildiği gibi "CPU cihazın modüllerini senkronize eder" opsiyon kutusunu etkinleştirerek yapılandırabilirsiniz:

Standart olarak, saat senkronizasyonu NTP sunucusu üzerinden etkinleştirilmez ve CP saatlerinin saat senkronizasyonu CPU'nun saati ile etkinleştirilmez.

CPU saatinin saat senkronizasyonunu ve CP saatlerinin saat senkronizasyonunu birbirlerinden bağımsız olarak yapılandırabilirsiniz. Bu nedenle, CPU'nun saati yukarıda açıklanan yöntemlerden biriyle ayarlandığında, CP saatlerinin saat senkronizasyonunu CPU üzerinden etkinleştirebilirsiniz.

NTP sunucusunu kullanırsanız, güncelleme aralığını seçebilirsiniz. NTP sunucusunun güncelleme aralığı için standart olarak 10 saniye ayarlanmıştır.

Saat senkronizasyonunu bir modülde etkinleştirirseniz, STEP 7 sizden, henüz gerçekleştirilmediyse "Saat senkronizasyonu" iletişim kutusundaki "CPU cihazın modüllerini senkronize eder" opsiyon kutusunu etkinleştirmenizi ister. STEP 7, saat senkronizasyonu için birden fazla Master saati kaynak olarak etkinleştirdiğiniz takdirde bir de uyarı görüntüler.

NOT

Saat senkronizasyonunun bir CP modülünde etkinleştirilmesi, bu CP modülünün, CPU'nun saatini ayarlamasına neden olur.

"Saat senkronizasyonu" iletişim kutusundaki "CPU cihazın modüllerini senkronize eder" opsiyon kutusunu etkinleştirdiyseniz, CPU Master saat olur. Bu durumda, CP modülleri CPU'nun saatini senkronize eder.

NOT

CPU için sadece bir zaman kaynağını yapılandırın.

CPU birden fazla kaynaktan (örneğin NTP sunucusu ve CP modülü) saat senkronizasyonu alırsa, bu durum, saat güncellemesi sırasında çakışmalara neden olabilir. Birden fazla kaynaktan gelen saat senkronizasyonu, saati temel alan yordamları ve olayları olumsuz etkileyebilir.

Programlama konseptleri

7

7.1 PLC cihazı ile bir otomasyon çözümü tasarlamak için kullanılan direktifler

Bir PLC sistemi geliştirilirken çok sayıda yöntem ve kriter arasından seçim yapabilirsiniz. Aşağıda, birçok projede uygulanabilen genel direktifler verilmiştir. Bu sırada tabii ki şirketinizde geçerli olan prosedür talimatlarına uymanız ve kendi deneyimlerinizi dikkate almanız gerekir.

Tablo 7-1 PLC cihazı ile bir otomasyon çözümü tasarlamak için kullanılan direktifler

Önerilen adımlar	Görevler
Prosesinizin veya tesisinizin bölümlere ayrılması	Prosesinizi veya tesisinizi birbirinden bağımsız bölümlere ayırın. Bu bölümler, çok sayıda otomasyon sisteminin arasındaki sınırları belirler ve fonksiyon alanlarının açıklamalarını ve ayrıca donanımların atanmasını etkiler.
Fonksiyon alanlarının açıklanması	Örneğin girişler/çıkışlar, akışın fonksiyonel açıklaması, bir aktüatör (örn. bir manyetik valf, bir motor veya bir tahrik) tepki vermeden önce ulaşılması gereken durumlar, kullanıcı arayüzünün ve prosesin veya tesisin diğer bölümlerine açılan tüm arayüzlerin bir açıklaması gibi prosesin veya tesisin her bölümü için geçerli olan fonksiyon açıklamalarını girin.
Güvenlik devrelerinin tasarlanması	Güvenlik nedenleriyle sabit şekilde bağlanmış devrelere ihtiyaç duyan cihazları belirleyin. Kumanda cihazlarında, güvenli olmayan çalışma durumları ortaya çıkabilir, bu da beklenmedik yol alma özelliklerine veya sistemin fonksiyon akışlarının değişmesine neden olabilir. Tesisin beklenmeyen veya hatalı bir şekilde çalışması durumunda ağır yaralanmalar veya maddi hasarların oluşması riski söz konusuysa, güvenli olmayan çalışma koşullarını önlemek için kumanda ünitesinden bağımsız olarak çalışan elektro mekanik program müdahaleleri düzenlemeyi hesaba katmanız gerekir. Güvenlik devrelerinin tasarlanması için aşağıdaki yöntemi izleyin: • Tehlike potansiyeline sahip aktüatörlerin hatalı veya beklenmedik şekilde çalışmasını tanımlayın. • İşletimin tehlikesiz olduğu koşulları tanımlayın ve bu koşulların PLC'den bağımsız olarak nasıl algılanacağını belirleyin. • Gerilim açıldığı ve tekrar kapatıldığı takdirde ve hataların algılanması durumunda PLC'nin prosesi nasıl etkileyeceğini tanımlayın. Bu bilgileri sadece normal ve beklenen anormal işletimin tasarlanması için kullanın. Güvenlik nedenleriyle bu "En iyi durum" senaryosuna güvenmeyin. • Manuel müdahale veya tehlikeli akışları PLC'den bağımsız olarak engelleyen elektro mekanik program müdahaleleri aracılığıyla düzeltmeleri tasarlayın. • Durum bilgilerinin, program ve her kullanıcı arayüzü gerekli bilgilere sahip olacak şekilde bağımsız elektrik devreleri tarafından PLC'ye aktarılmasını sağlayın. • Prosesin güvenli bir şekilde gerçekleşebilmesi için diğer güvenlik taleplerini tanımlayın.
Sistem güvenliğinin planlanması	Prosesinize erişim için hangi koruma kademesine (Sayfa 152) ihtiyaç duyduğunuzu belirleyin. CPU'ları ve program bloklarını şifre aracılığıyla yetkisiz kişilerin erişimine karşı koruyabilirsiniz.
Operatör istasyonlarının tanımlanması	Fonksiyon alanlarının açıklamalarındaki gereklilikler aracılığıyla operatör istasyonlarının aşağıdaki planlarını oluşturun: • Prosesle veya sistemle ilgili tüm operatör istasyonlarının pozisyonu • Örn. ekran, şalterler ve lambalar gibi operatör istasyonu cihazlarının mekanik yerleşimi. • PLC'nin ve sinyal modüllerinin ilgili giriş ve çıkışlarıyla birlikte kablaj planları

Önerilen adımlar	Görevler
Konfigürasyon planlarının çizilmesi	Fonksiyon alanlarının açıklamalarındaki gereklilikler aracılığıyla otomasyon sistemi için kullanılan konfigürasyon planlarını oluşturun: • Prosesle veya sistemle ilgili tüm PLC'lerin pozisyonları • Tüm PLC'lerin ve G/Ç modüllerinin mekanik yerleşimi (dolaplar, vb. dahil) • Cihaz numaraları, iletişim adresleri ve giriş ve çıkışların adresleri de dahil olmak üzere tüm PLC'lerin ve G/Ç modüllerinin kablaj planları
Sembolik adların listelenmesi	Mutlak adresler için sembolik adreslerin bir listesini oluşturun. Sadece fiziksel giriş ve çıkışlar yerine, programınızda kullandığınız diğer tüm elemanları da (örn. değişken adları gibi) belirtin.

7.2 Kullanıcı programınızın yapılandırılması

Otomasyon çözümleri için bir kullanıcı programı oluştururken kod bloklarına program için kullanılan yordamları ekleyebilirsiniz:

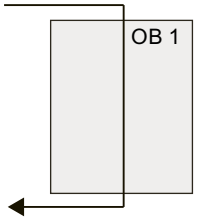
- Bir organizasyon bloğu (OB), CPU'daki belirli bir olaya tepki gösterir ve kullanıcı programının yürütülmesini durdurabilir. Kullanıcı programının (OB 1) periyodik olarak yürütülmesi için standart blok, kullanıcı programınız için temel yapıyı sağlar. Programınıza başka OB'ler eklerseniz, bu OB'ler OB 1'in yürütülmesini durdurur. Diğer OB'ler, örn. yol alma, alarm ve hata işleme için kullanılan fonksiyonlar veya özel bir program kodunun belirli zaman aralıklarında yürütülmesi için kullanılan fonksiyonlar gibi özel fonksiyonları gerçekleştirir.
- Bir fonksiyon bloğu (FB), başka bir kod bloğundan (OB, FB veya FC) çağrıldığında yürütülen bir alt programdır. Çağıran blok, FB'ye parametreleri aktarır ve FB'nin bu çağrısı veya bu örneği için verileri kaydeden belirli bir veri bloğu (DB) tanımlar. Örnek DB değişirse, genel bir FB, bir cihaz grubunun işletimine kumanda edebilir. Örneğin bir FB, her pompanın veya her valfın özel işletim parametrelerini içeren farklı örnek DB'lere sahip birden fazla pompa veya valfa kumanda edebilir.
- Bir fonksiyon (FC), başka bir kod bloğundan (OB, FB veya FC) çağrıldığında yürütülen bir alt programdır. Bir FC'ye hiçbir örnek DB atanmamıştır. Çağıran blok, FC'ye parametreleri aktarır. FC'nin çıkış değerleri bir bellek adresine veya global bir DB'ye yazılmalıdır.

Kullanıcı programı için yapı türünün seçilmesi

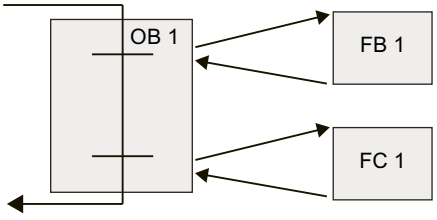
Kullanıcı programınız için, uygulamanızın gerekliliklerine bağlı olarak doğrusal veya modüler bir yapı seçebilirsiniz:

- Doğrusal bir program, otomasyon görevleriniz için tüm yordamları arka arkaya yürütür. Doğrusal bir programda tipik olarak tüm program yordamları çevrim OB'sine (OB 1) kaydedilir.
- Modüler bir program, özel görevleri yürüten özel kod bloklarını çağırır. Modüler bir program yapısı oluşturmak için, karmaşık otomasyon görevini, prosesin teknolojik fonksiyonlarına karşılık gelen daha küçük kısmi görevlere bölebilirsiniz. Her kod bloğu, ilgili kısmi görev için kullanılan program segmentini içerir. Programınızı, başka bir bloktan bir kod bloğu çağırarak yapılandırabilirsiniz.

Doğrusal yapı:



Modüler yapı:



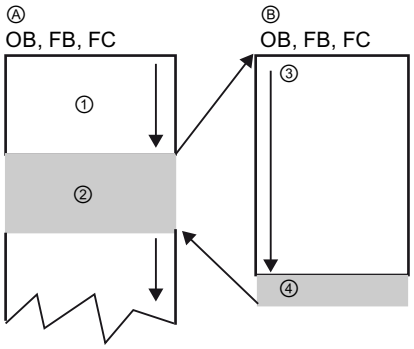
Kullanıcı programında birkaç kez kullanılabilen genel kod blokları oluşturma yoluyla kullanıcı programının tasarlanması ve uygulanması işlemleri kolaylaştırılabilir. Genel kod bloklarıyla çalışmanın birçok avantajı vardır:

- Standart görevler için tekrar kullanılabilen kod blokları oluşturabilirsiniz, örneğin bir pompanın veya bir motorun kumanda ünitesi için. Bu genel kod bloklarını ayrıca, çeşitli uygulamalar veya çözümler için kullanılabilecek bir kütüphanede saklayabilirsiniz.
- Kullanıcı programının modüler olarak, fonksiyonel görevleri temel alan münferit bileşenlere ayrılması yoluyla program yapısı daha anlaşılır hale gelir ve kullanımı kolaylaşır. Modüler bileşenler sadece program tasarlama sürecini standardize etmekle kalmaz, aynı zamanda program kodunun uyarlama ve değişiklik işlemlerini de daha hızlı ve daha kolay hale getirir.
- Modüler bileşenler oluşturmak, programınızı test etmeyi kolaylaştırır. Komple program bir dizi modüler program segmentine ayrıldıysa, her kod bloğunun fonksiyonlarını doğrudan geliştirme sırasında test edebilirsiniz.
- Özel teknolojik fonksiyonlarla ilgili modüler bileşenlerle komple uygulamanın işletime alınması kolaylaştırılır ve kısaltılır.

7.3 Programınızın yapılandırılmasına yönelik blokların kullanımı

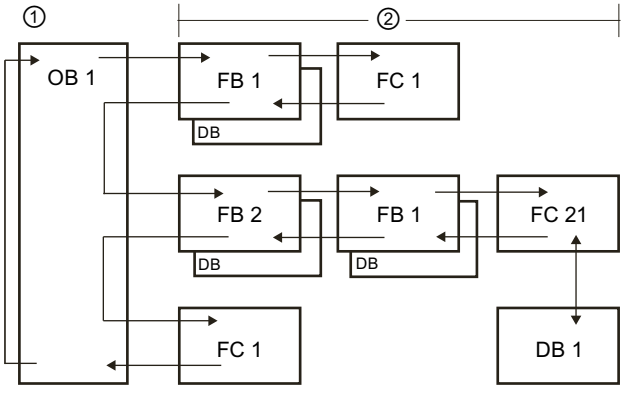
Modüler kod bloklarını, genel görevlerin yürütülmesi için kullanılan FB'ler ve FC'lerin tasarımı aracılığıyla oluşturabilirsiniz. Daha sonra, başka kod bloklarının bu yeniden kullanılabilen modülleri çağırması yoluyla programınızı yapılandırabilirsiniz. Çağıran blok, cihaza özel parametreleri çağrılan bloğa aktarır.

Bir kod bloğu başka bir kod bloğundan çağrılırsa, CPU, çağrılan bloktaki program kodunu yürütür. CPU, çağrılan blok işlendikten sonra, çağıran bloğun yürütülmesi işlemini devam ettirir. İşleme, blok çağrısından sonra bir sonraki yordamın yürütülmesiyle devam eder.



A Çağıran blok
B Çağrılan (veya durdurulan) blok
① Programın yürütülmesi
② Başka bir bloğun yürütülmesini tetikleyen yordam veya olay
③ Programın yürütülmesi
④ Blok sonu (çağıran bloğa dönüş)

Blok çağrıları, yapıyı daha modüler şekilde oluşturmak için iç içe geçirebilir. Gömme derinliği, aşağıdaki örnekte 3 seviyeyi kapsamaktadır: Kod blokları çağrılarak program çevrimi OB artı 3 kod seviye.



① Çevrim başlangıcı
② Gömme derinliği

Not: Maksimum gömme derinliği altı seviyedir. Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

7.3.1 Organizasyon bloğu (OB)

Organizasyon blokları, programınızın yapılandırılması için kullanılır. Bu bloklar, işletim sistemi ile kullanıcı programı arasındaki arayüzü oluşturur. OB'ler olay kontrollüdür. Örn. bir teşhis alarmı veya bir zaman aralığı gibi bir olay, CPU'nun bir OB'yi yürütmesine neden olur. Bazı OB'ler ön tanımlı yol verme olaylarına ve ön tanımlı tutumlara sahiptir.

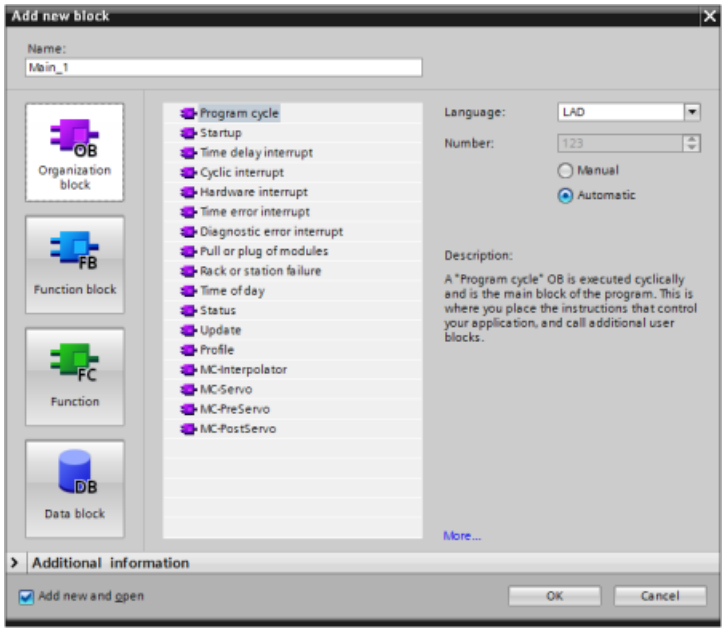
Program çevrimi OB'si, ana programı içerir. Kullanıcı programınıza birden fazla program çevrimi OB'yi kaydedebilirsiniz. RUN işletim durumunda, öncelik kademesi en düşük olan program çevrimi OB'lerini yürütülür ve başka olay türleri tarafından durdurulabilir. Yol alma OB'si, program çevrimi OB'sini durdurmaz, çünkü CPU, yol alma OB'sini RUN işletim durumuna geçilmeden önce yürütür.

CPU, program çevrimi OB işlendikten sonra program çevrimi OB'leri yeniden yürütür. Bu periyodik yürütme, programlanabilir kontrol sistemleri için "normal" işleyiş türüdür. Birçok uygulamada tüm kullanıcı programı tek bir program çevrimi OB'sinin içinde yer alır.

Örn. alarm ve hata işleme için kullanılan fonksiyonlar veya belirli bir program kodunun belirli aralıklarla yürütülmesi için kullanılan fonksiyonlar gibi belirli fonksiyonları yürütmek için başka OB'ler oluşturabilirsiniz. Bu OB'ler, program çevrimi OB'sinin yürütülmesini durdurur.

OB oluşturma

Kullanıcı programınız için yeni OB'leri, "Yeni blok ekle" iletişim kutusunda oluşturabilirsiniz.



Alarm işleme her zaman olay kontrollüdür. CPU, bu tür bir olay ortaya çıkarsa, kullanıcı programının yürütülmesini durdurur ve bu olayın işlenmesi için kullanılan OB'yi çağırır. CPU, durduran OB yürütüldükten sonra, kullanıcı programının yürütülmesine daha önce durdurulduğu noktadan devam eder.

CPU, alarm olaylarının işlenme sırasını öncelikler aracılığıyla belirler. Bir öncelik sınıfına birden fazla alarm olayı atayabilirsiniz. Daha fazla bilgiyi, organizasyon blokları (Sayfa 68) ve kullanıcı programının yürütülmesi (Sayfa 60) ile ilgili konularda bulabilirsiniz.

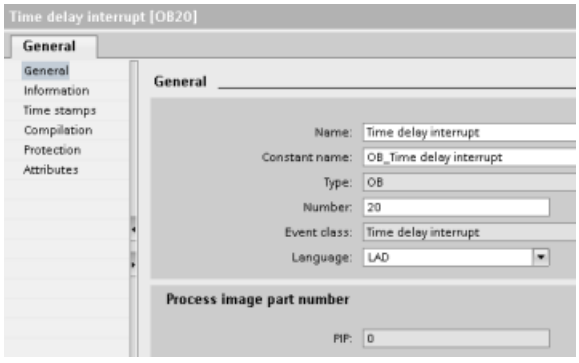
Ek OB'ler oluşturma

Kullanıcı programınız için, program çevrimi ve yol alma OB'si olaylarının sınıfları için de birden fazla OB'ler oluşturabilirsiniz. "Yeni blok ekle" iletişim kutusunda bir OB oluşturabilir ve OB'niz için bir ad girebilirsiniz.

Kullanıcı programınız için çok sayıda program çevrimi OB oluşturursanız, CPU, münferit program çevrimi OB'leri numara sırasıyla yürütür. Bu sırada, en düşük numaraya sahip program çevrimi OB ile başlar (örn. OB 1). Örnek: CPU, ilk program çevrimi OB (OB 1 gibi) tamamlandıktan sonra, bir sonraki büyük numaraya sahip program çevrimi OB'yi yürütür.

Bir OB'nin özelliklerinin yapılandırılması

Bir OB'nin özelliklerini, örneğin OB numarasını veya programlama dilini yapılandırabilirsiniz.



NOT

Bir OB'ye TPA0, TPA1, TPA2, TPA3 veya TPA4 gibi bir kısmi proses görünümü numarası atayabileceğinize dikkat edin. Kısmi proses görünümü numarası için bir sayı girerseniz, CPU, kısmi proses görünümünü oluşturur. Kısmi proses görünümleriyle ilgili açıklamaları, "Kullanıcı programının yürütülmesi (Sayfa 60)" başlığında bulabilirsiniz.

7.3.2 Fonksiyon (FC)

Bir fonksiyon (FC), tipik olarak bir dizi giriş değerine sahip belirli bir işlemi gerçekleştiren bir kod bloğudur. FC, bu işlemin sonuçlarını belirli saklama konumlarına kaydeder. FC'leri örneğin standart işlemleri ve birkaç kez kullanılabilen işlemleri (örn. matematiksel hesaplamalar) veya teknolojik fonksiyonları (örn. Bit bağlantı işlemleri üzerinden münferit kumanda ünitesi işlemleri için) gerçekleştirmek için kullanın. Bir FC, bir programın çeşitli noktalarda birkaç kez de çağrılabilir. Bu yeniden kullanım özelliği, sık tekrarlanan görevlerin programlanmasını kolaylaştırır.

Bir FC'ye hiçbir ilgili örnek veri bloğu (DB) atanmamıştır. FC, işlemlerin hesaplanması için ihtiyaç duyulan geçici veriler için yerel veri belleğini kullanır. Geçici veriler kaydedilmez. Verileri kalıcı olarak kaydetmek için çıkış değeri, örn. markör belleği gibi global bir bellek alanına veya global bir DB'ye atanmalıdır.

7.3.3 Fonksiyon bloğu (FB)

Bir fonksiyon bloğu (FB), kendi parametreleri ve statik verileri için bir örnek veri bloğu kullanan bir kod bloğudur. FB'ler, bir veri bloğunda (DB) veya bir örnek DB'de bulunan değişken bir belleğe sahiptir. Örnek DB, FB'nin bu örneğine (veya bu çağrısına) atanmış olan ve verileri, FB'lerin süresi dolduktan sonra kaydeden bir bellek bloğu sağlar. FB'nin farklı çağrılarına farklı örnek DB'ler atayabilirsiniz. Örnek DB, birden fazla cihazın kumanda ünitesi için genel bir FB kullanmanıza olanak sağlar. Bir kod bloğunun bir FB'yi ve bir örnek DB'yi çağırdığı bir program yapılandırabilirsiniz. Bu durumda CPU, bu FB'deki program kodunu yürütür ve blok parametrelerini ve statik yerel verileri örnek DB'ye kaydeder. CPU, FB'nin yürütülmesi işlemi tamamlandıysa, FB'yi çağıran kod bloğunun yürütülmesine devam eder. Örnek DB, FB'nin bu örneği için kullanılan değerleri kaydeder. Bu değerler, fonksiyon bloğunun sonraki çağrıları için aynı çevrimde veya başka çevrimlerde kullanıma sunulur.

Atanmış belleğe sahip yeniden kullanılabilen kod blokları

İşletimleri bir çevrim dahilinde sona ermeyen faaliyetlerin veya cihazların akışına kumanda etmek için tipik olarak bir FB kullanılır. İşletim parametrelerinin iki çevrim arasında hızlı bir şekilde erişilebilir olmaları amacıyla bu parametrelerin kaydedilmesi için her FB kendi kullanıcı programında bir veya daha fazla örnek DB'ye sahiptir. Bir FB'nin çağrılmasıyla eşzamanlı olarak, FB'nin bu çağrısı veya bu "örneği için" kullanılan blok parametrelerinin ve statik yerel verilerin bulunduğu bir örnek DB belirtebilirsiniz. Örnek DB, bu değerleri FB yürütüldükten sonra kaydeder.

FB genel kumanda görevleri için tasarlanırsa, bu FB, FB'nin farklı çağrıları için farklı örnek DB'ler seçilerek birçok cihaz için kullanılabilir.

Bir FB giriş, çıkış ve geçiş parametrelerini ve ayrıca statik parametreleri bir örnek DB'ye kaydeder.

Ayrıca fonksiyon bloğu arayüzünü RUN işletim durumuna getirebilir ve yükleyebilirsiniz (Sayfa 1144).

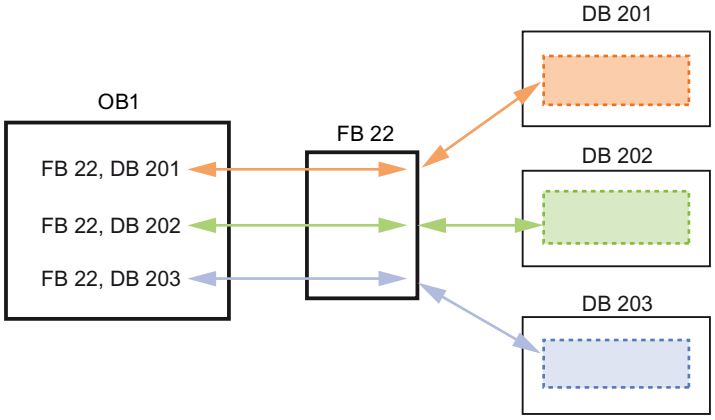
Başlangıç değerinin örnek DB'de atanması

Örnek DB, hem bir standart değeri hem de her parametre için bir başlangıç değerini kaydeder. Başlangıç değeri, FB yürütülürken kullanılması gereken değeri belirtir. Başlangıç değeri, kullanıcı programınızın yürütülmesi sırasında değiştirilebilir.

FB arayüzü, standart değer için, program kodu yazılırken parametre için yeni bir başlangıç değeri girebileceğiniz bir sütuna da sahiptir. FB'deki bu standart değer, daha sonra ilgili örnek DB'nin başlangıç değerine aktarılır. FB arayüzünde bir parametreye yeni bir başlangıç değeri atamazsanız, standart değer örnek DB'den başlangıç değerine kopyalanır.

DB'lerle münferit FB'lerin kullanılması

Aşağıdaki şekilde, bir FB'yi üç kez çağıran ve her çağrıda başka bir veri bloğunun kullanıldığı bir OB gösterilmektedir. Bu yapı aracılığıyla, örn. motorlar gibi aynı türdeki birden fazla cihazın kumandası için, bir cihazın her çağrısına başka bir örnek veri bloğu atanarak genel bir FB kullanılabilir. Her örnek DB, münferit bir cihaz için kullanılan verileri (devir sayısı, ilk açılış süresi ve toplam işletim süresi gibi) kaydeder.



Bu örnekte FB 22, üç farklı cihaza kumanda eder; DB 201, birinci cihazın işletme verilerini, DB 202, ikinci cihazın işletme verilerini ve DB 203 ise üçüncü cihazın işletme verilerini kaydeder.

7.3.4 Veri bloğu (DB)

Kullanıcı programınızda, kod blokları için kullanılan verileri kaydetmek üzere veri blokları (DB'ler) oluşturabilirsiniz. Kullanıcı programındaki tüm program blokları, global bir DB'deki verilere erişebilir, ancak bir örnek DB, belirli bir fonksiyon bloğu (FB) için kullanılan verileri kaydeder.

Bir DB'ye kaydedilen veriler, ilgili kod bloğu yürütüldükten sonra silinmez. İki tür veri bloğu mevcuttur:

- Global bir DB'ye, programınızdaki kod blokları için kullanılan veriler kaydedilir. Her OB, FB veya FC, global bir DB'deki verilere erişebilir.
- Bir örnek DB'ye, belirli bir FB için kullanılan veriler kaydedilir. Bir örnek DB'deki veri yapısı, (Input, Output ve InOut) parametrelerine ve FB'nin statik verilerine karşılık gelir. (FB'nin geçici belleği örnek DB'ye kaydedilmez.)

NOT

Örnek DB, belirli bir FB için kullanılan verileri içermesine rağmen, her kod bloğu, bir örnek DB'nin verilerine erişebilir.

Veri bloklarını RUN işletim durumunda da değiştirebilir ve yükleyebilirsiniz (Sayfa 1144).

Yazma korumalı veri blokları

Bir DB, sadece okunabilecek şekilde yapılandırılabilir:

1. Farenin sağ tuşuyla proje gezgininde DB'ye tıklayın ve bağlam menüsünden "Özellikler" öğesini seçin.
2. "Özellikler" iletişim kutusunda "Öznitelik" elemanını seçin.
3. "Cihazdaki veri bloğu yazma korumalı" opsiyonunu seçin ve "TAMAM" ile onaylayın.

Optimize edilmiş ve standart veri blokları

Veri bloğuna erişimi, optimize edilmiş erişim olarak da yapılandırabilirsiniz. Veri bloğu optimize edildiyse, standart veri bloğu olarak geçerli olur. Bir standart DB, STEP 7 Classic'in ve klasik S7-300 ile S7-400 CPU'larının programlama araçlarıyla uyumludur. Optimize edilmiş erişime sahip veri blokları, sabit şekilde tanımlanmış bir yapıya sahip değildir. Veri elemanları, deklarasyonda sadece bir sembolik ad içerir ve blokta sabit bir adrese sahip değildir. CPU, elemanları otomatik olarak bloğun mevcut bellek alanına, bellekte boşluklar oluşmayacak şekilde kaydeder. Bu sayede bellek kapasitesi optimum şekilde kullanılır.

Bir veri bloğu için optimize edilmiş erişimi ayarlamak için aşağıdaki yöntemi izleyin:

1. STEP 7 proje gezgininde program bloklarının klasörünü genişletin.
2. Farenin sağ tuşuyla veri bloğuna tıklayın ve bağlam menüsünden "Özellikler" komutunu seçin.
3. Özniteliklerde, "Optimize edilmiş blok erişimi" öğesini seçin.

Yeni veri bloklarında standart olarak optimize edilmiş blok erişiminin ayarlandığına dikkat edin. "Optimize edilmiş blok erişimini" devre dışı bırakırsanız, blok için standart erişim ayarlanır.

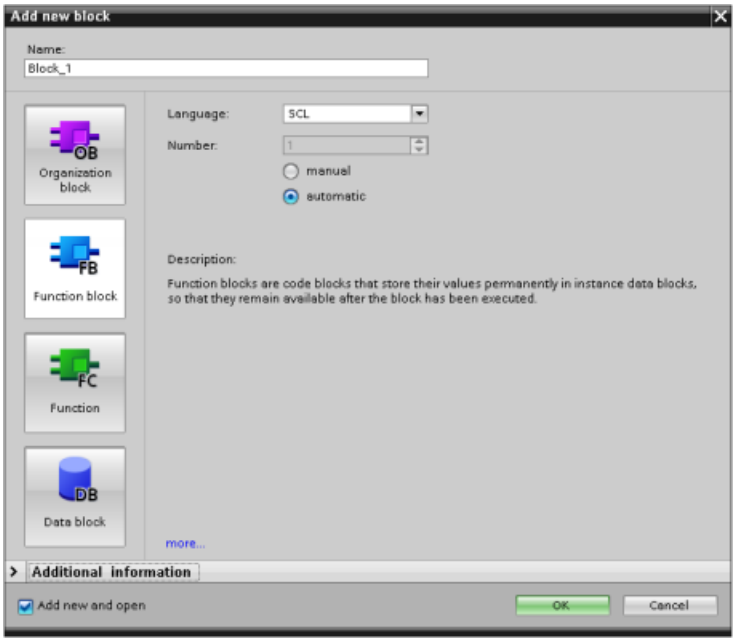
NOT

Bir FB ve buna ait örnek DB için kullanılan blok erişimi tipi

FB'niz için "Optimize edilmiş blok erişimi" ayarlandıysa, bu FB'nin örnek DB'si için de "Optimize edilmiş blok erişimi" opsiyonu ayarlanmış olmalıdır. Aynı şekilde, FB için "Optimize edilmiş blok erişimi" seçeneği ayarlanmadıysa ve bu nedenle standart erişime sahip bir FB söz konusuysa, örnek DB için de "Optimize edilmiş blok erişimi" opsiyonu yerine standart erişimin ayarlanması gerekir.

Erişim tipleri uyumlu değilse, FB yürütülürken FB'nin IN/OUT parametre değerlerinde bir HMI cihazı üzerinden gerçekleştirilen değişiklikler kaybolur.

7.3.5 Yeniden kullanılabilen kod bloklarının oluşturulması



OB'leri, FB'leri, FC'leri ve global DB'leri proje gezginindeki "Program blokları" öğesinin altında yer alan "Yeni blok ekle" iletişim kutusunda oluşturabilirsiniz.

Bir kod bloğu oluşturulurken blok için programlama dilini seçmeniz gerekir. Bir DB için dil seçilmez, çünkü bu bileşen sadece veri kaydeder.

"Yeni ekle ve aç" opsiyon kutusu seçildiğinde (standart ayar) kod bloğu proje görünümünde açılır.

Yeniden kullanmak istediğiniz nesneleri kütüphanelere kaydedebilirsiniz. Her proje, projeye bağlı bir proje kütüphanesine sahiptir. Proje kütüphanesine ek olarak, birçok projede kullanılabilen istediğiniz sayıda global kütüphane oluşturabilirsiniz. Kütüphaneler kendi aralarında uyumlu olduğu için, kütüphane elemanları bir kütüphaneden diğerine kopyalanabilir veya taşınabilir.

Kütüphaneler, örneğin, önce proje kütüphanesine kopyalayacağınız ve daha sonra burada geliştirmeye devam edeceğiniz blok şablonları oluşturmak için kullanılır. Son olarak, blokları proje kütüphanesinden global bir kütüphaneye kopyalayabilirsiniz. Global kütüphaneyi, projedeki diğer iş arkadaşlarınızın kullanımına sunabilirsiniz. İş arkadaşlarınız blokları kullanabilir ve gerektiğinde kendi ilgili gerekliliklerine uyarlayabilir.

Kütüphane işlemleriyle ilgili ayrıntıları, TIA Portal bilgi sisteminde bulabilirsiniz.

7.3.6 Parametrelerin bloklara aktarılması

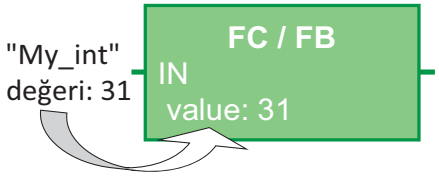
Fonksiyon blokları (FB'ler) ve fonksiyonlar (FC'ler) üç farklı arayüz tipine sahiptir:

- IN
- IN/OUT
- OUT

FB'ler ve FC'ler parametreleri IN ve IN/OUT arayüz tipleri üzerinden alır. Bloklar, parametreleri işler ve IN/OUT ve OUT arayüz tipleri üzerinden değerleri çağrı yapana döndürür.
Kullanıcı programı, parametreleri iki yöntemden biri aracılığıyla aktarır:

Call-by-Value

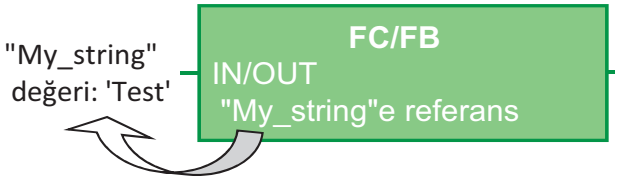
Kullanıcı programı bir parametreyi "Call-by-Value" olarak bir fonksiyona aktarırsa, kullanıcı programı, gerçek parametre değerini, IN arayüz tipi için kullanılan bloğun giriş parametresine kopyalar. Bu işlem, kopyalanan değer için ek bellek gerektirir.



Kullanıcı programı bloğu çağırırsa, değerleri kopyalar.

Call-by-Reference

Kullanıcı programı bir parametreyi "Call-by-Reference" olarak bir fonksiyona aktarırsa, kullanıcı programı, IN/OUT arayüz tipi için kullanılan gerçek parametrenin adresine referans verir ve değeri kopyalamaz. Bu işlem, ek bellek gerektirmez.



Kullanıcı programı bloğu çağırırsa, gerçek parametrelerin adresine referans verir.

NOT

IN/OUT arayüz tipi, genelde, gereken veri belleğinin gereksiz yere büyümesini önlemek amacıyla, yapılandırılmış değişkenler (örneğin ARRAY, STRUCT ve STRING) için kullanılır.

Blok optimizasyonu ve parametre aktarımı

Kullanıcı programı, basit veri tiplerinde (örneğin INT, DINT ve REAL) FC parametrelerini "Call-by-Value" olarak aktarır. Karmaşık veri tipleri (örneğin STRUCT, ARRAY ve STRING) "Call-by-Reference" olarak aktarılır.

Kullanıcı programı, FB parametrelerini normalde, FB'nin örnek veri bloğuna (DB) aktarır:

- Kullanıcı programı, basit veri tiplerini (örneğin INT, DINT ve REAL), parametrelerin örnek DB'ye ve örnek DB'den kopyalanmasıyla "Call-by-Value" olarak aktarır.
- Kullanıcı programı, karmaşık veri tiplerini (örneğin STRUCT, ARRAY ve STRING) IN ve OUT parametre tipleri için örnek DB'ye ve örnek DB'den kopyalar.
- Kullanıcı programı, karmaşık veri tiplerini IN/OUT arayüz tipi için "Call-by-Reference" olarak aktarır.

DB'ler, "Optimize edildi" veya "Standart" (optimize edildi) opsiyonlarıyla oluşturulabilir. Optimize edilen veri blokları, optimize edilmeyen veri bloklarına göre daha kompakttır. Optimize edilen ve optimize edilmeyen DB'lerde, DB'lerin içindeki veri elemanlarının sırası da farklıdır. S7-1200/1500, STEP 7 (TIA Portal) için S7 programlama direktifi, 12/2018 (http://support.automation.siemens.com/WW/view/en/81318674)'deki "Optimize edilen bloklar" alt bölümünde, optimize edilen bloklarla ilgili bir açıklama bulabilirsiniz.

Optimize edilen veya optimize edilmeyen verileri işlemek için FB'ler ve FC'ler oluşturabilirsiniz. "Optimize edilmiş blok erişimi" opsiyon kutusunu, blok için kullanılan özniteliklerden biri olarak etkinleştirebilirsiniz. Kullanıcı programı, program bloklarını standart olarak optimize eder ve program blokları, bloğa aktarılan verileri optimize edilmiş formatta bekler.

Kullanıcı programı karmaşık bir parametreyi (örneğin STRUCT) bir fonksiyona aktarırsa, sistem, veri bloğunun yapıyı içeren optimizasyon ayarını ve program bloğunun optimizasyon ayarını kontrol eder. Hem veri bloğunu hem de fonksiyonu optimize ederseniz, kullanıcı programı yapıyı "Call-by-Reference" olarak aktarır. Aynı durum, hem veri bloğunu hem de fonksiyonu optimize etmediğinizde de geçerlidir.

Ancak fonksiyon ve veri bloğu farklı optimizasyon ayarlarına sahipse (yani bir blok optimize edilmişse ve diğeri optimize edilmemişse), yapı, fonksiyon tarafından beklenen formata dönüştürülmelidir. Örnek: Veri bloğu optimize edilmemişse ve fonksiyon optimize edilmişse, veri bloğundaki bir yapı, fonksiyonun yapıyı işleyebilmesi için optimize edilmiş formata dönüştürülmelidir. Sistem bu dönüştürme işlemini, yapıyı "kopyalayarak" ve fonksiyon tarafından beklenen optimize edilmiş formata dönüştürerek gerçekleştirir.

Özet olarak, kullanıcı programı karmaşık bir veri tipini (örneğin STRUCT) bir fonksiyona IN/OUT parametresi olarak aktarırsa, fonksiyon, kullanıcı programının yapıyı "Call-by-reference" olarak aktarmasını bekler:

- Hem yapıyı içeren veri bloğunu hem de fonksiyonu optimize ederseniz veya optimize etmezseniz, kullanıcı programı, verileri "Call-by-Reference" olarak aktarır.
- Veri bloğu ve fonksiyon için aynı optimizasyon ayarlarını yapılandırmazsanız (bir blok optimize edilmiş, diğeri optimize edilmemişse), sistem, fonksiyona aktarma işleminden önce yapının bir kopyasını oluşturmak zorundadır. Sistem, yapının bu kopyasını oluşturmak zorunda olduğu için, böylece "Call-by-Reference" pratik bir şekilde "Call-by-Value"ya dönüştürülür.

Optimizasyon ayarlarının kullanıcı programları üzerindeki etkisi

Parametrelerin kopyalanması, bir HMI cihazı veya bir alarm OB nedeniyle yapıdaki elemanların değişmesi durumunda bir kullanıcı programında bir soruna yol açabilir. Örnek: Bir fonksiyonun (normalde "Call-by-Reference" olarak aktarılan) bir IN/OUT parametresi mevcuttur, ancak veri bloğunun optimizasyon ayarları ile fonksiyonun optimizasyon ayarları farklıdır:

1. Kullanıcı programı fonksiyonu çağıracağı zaman, sistem, verilerin formatını fonksiyona uyarlamak için yapının bir "kopyasını" oluşturmak zorundadır.
2. Kullanıcı programı, fonksiyonu, yapının "kopyasına" bir referansla çağırır.
3. Fonksiyonun yürütülmesi sırasında bir alarm OB'si ortaya çıkar ve alarm OB'si, başlangıçtaki yapıda bulunan bir değeri değiştirir.
4. Fonksiyon işlenir ve yapı bir IN/OUT parametresi olduğu için sistem, değerleri başlangıçtaki formatta başlangıçtaki yapıya dönüştürür.

Formatı değiştirmek için yapının bir kopyası oluşturulduğu için, Alarm OB tarafından yazılan veriler kaybolur. Aynı durum, bir değerin bir HMI cihazıyla yazılması durumunda ortaya çıkabilir. HMI cihazı, kullanıcı programını durdurabilir ve bir alarm OB ile aynı şekilde bir değeri yazabilir.

Bu sorunu önlemek için birçok olanak mevcuttur:

- En iyi çözüm, daha karışık veri tiplerinin (örneğin STRUCT) kullanımı sırasında program bloğunun optimizasyon ayarlarını ve veri bloğunun optimizasyon ayarlarını birbirlerine uyarlamaktır. Bu sayede, kullanıcı programının parametreleri her zaman "Call-by-Reference" olarak aktarması sağlanır.
- Başka bir çözüm, bir alarm OB'sinin veya bir HMI cihazının yapıdaki bir elemanı doğrudan değiştirmemesi şeklindedir. OB veya HMI cihazı başka bir değişkeni değiştirebilir ve daha sonra bu değişkeni kullanıcı programındaki belirli bir noktada yapıya kopyalayabilirsiniz.

7.4 Veri tutarlılığı

CPU, tüm temel veri tipleri (örn. Word veya DWord) ve sistem tarafından tanımlanan yapılar (örn. IEC_TIMERS veya DTL) için veri tutarlılığını korur. Değerin okuma veya yazma işlemi durdurulamaz. (CPU, örn. DWord veri tipindeki bir değere erişimi, DWord'ün dört Bayt'ı okunana veya yazılana kadar korur.) CPU, program çevrimi OB'leri ile alarm OB'lerinin aynı anda bellekteki aynı adrese yazamamasını sağlamak için, bir alarm OB'sini ancak program çevrimi OB'sindeki okuma veya yazma işlemi sona erdiğinde yürütür.

Kullanıcı programınızda, bir program çevrimi OB'sinin ve bir alarm OB'sinin belleğinde birden çok değerin ortak olarak kullanılması durumunda, kullanıcı programınız, bu değerlerin tutarlı şekilde değiştirilmesini veya okunmasını da sağlamak zorundadır. DIS_AIRT (Alarm işlemeyi devre dışı bırak) ve EN_AIRT (Alarm işlemeyi etkinleştir) yordamlarıyla, program çevrimi OB'nizde ortak olarak kullanılan değerlere erişimi koruyabilirsiniz.

- Bir okuma veya yazma işlemi sırasında hiçbir alarm OB'sinin yürütülemeyeceğinden emin olmak için kod bloğuna bir DIS_AIRT yordamı ekleyin.
- Bir alarm OB tarafından değiştirilebilecek değerleri okuyan veya yazan yordamları ekleyin.
- DIS_AIRT yordamını iptal etmek ve bir alarm OB'sinin yürütülmesine izin vermek için sekansın sonuna bir EN_AIRT yordamı ekleyin.

Bir HMI cihazının veya başka bir CPU'nun bir iletişim talebi de program çevrimi OB'sinin yürütülmesini durdurabilir. İletişim talepleri, veri tutarlılığıyla ilgili sorunlara da yol açabilir. CPU, temel veri tiplerinin, kullanıcı programındaki yordamlar tarafından her zaman tutarlı şekilde okunmasını ve yazılmasını sağlar. Kullanıcı programı iletişim talepleri tarafından

düzenli olarak durdurulacağı için, CPU'nun birçok değerinin tümünün eşzamanlı olarak HMI tarafından güncellenmesi sağlanamaz. Örneğin bir HMI ekranında görüntülenen değerler CPU'nun farklı çevrimlerinden gelebilir.

PtP yordamları (noktadan noktaya), PROFINET yordamları (TSEND_C ve TRCV_C gibi), desantral çevrebirim için kullanılan PROFINET yordamları ve desantral çevrebirim için kullanılan PROFIBUS yordamları, durdurulabilecek veri tamponlarını aktarır. Veri tamponları için veri tutarlılığını, tamponlardaki okuma ve yazma işlemlerini hem program çevrimi OB'sinde hem de alarm OB'sinde engelleyerek sağlayın. Bir alarm OB'sinde bu yordamlar için tampon değerlerinin değiştirilmesi gerekirse, DIS_AIRT yordamıyla durdurma işlemini (bir alarm OB aracılığıyla veya bir HMI cihazı ve başka bir CPU tarafından gerçekleştirilen bir iletişim kesintisi aracılığıyla), bir EN_AIRT yordamı yürütülene kadar geciktirin.

NOT

DIS_AIRT yordamının kullanılması, EN_AIRT yordamı yürütülene kadar alarm OB'lerinin işlenmesini geciktirir ve kullanıcı programınızın alarm gecikmesini (bir olayın oraya çıkmasından alarm OB'sinin yürütülmesine kadar geçen zaman dilimi) etkiler.

7.5 Programlama dili

STEP 7, S7-1200 için şu standart programlama dillerini sunar:

- LAD (kontak planı), bir grafiksel programlama dilidir. Gösterim, devre şemalarını (Sayfa 182) temel alır.
- FBD (fonksiyon planı), Boole cebrinin (Sayfa 183) grafiksel lojik sembollerini temel alan bir programlama dilidir.
- SCL (Structured Control Language); metin tabanlı, daha yüksek seviyede bir programlama dilidir (Sayfa 184).

Bir kod bloğu oluşturursanız, bu blok için programlama dilini seçmeniz gerekir. Fonksiyon bloklarını (FB'ler) (Sayfa 174) programlamak için Sebep-Sonuç Matrisi CEM'i de (Cause-Effect-Matrix) kullanabilirsiniz. Fonksiyon blokları için CEM programlamayla ilgili bilgileri, STEP 7 bilgi sisteminde bulabilirsiniz.

Kullanıcı programı, bu programlama dillerinden birinde oluşturulmuş olan kod bloklarıyla çalışabilir.

7.5.1 Kontak planı (LAD)

Bir devre şemasının NC kontak ve NO kontak ve bobinler gibi elemanları ağlara bağlanır.



Karmaşık işlemler için bağlantılar oluşturmak üzere, paralel devreler için dallar ekleyebilirsiniz. Paralel dallar aşağıya doğru açılır veya doğrudan baraya bağlıdır. Dalları aşağıya doğru sonlandırabilirsiniz.

LAD; aritmetik, zamanlar, sayaçlar ve aktarma gibi çok sayıda fonksiyon için kutu yordamları sunar.

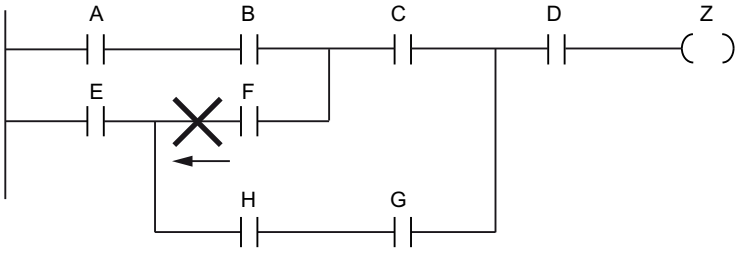
STEP 7, bir LAD ağındaki maksimum yordam sayısını (satırlar ve sütunlar) sınırlamaz.

NOT

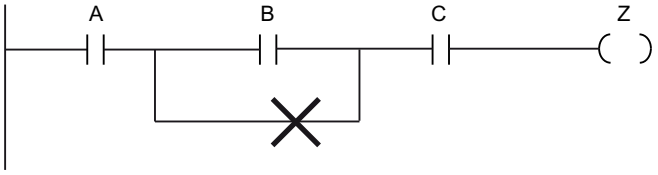
Her LAD ağı, bir bobinle veya bir kutuyla sonlandırılmalıdır.

Bir LAD ağı oluşturulurken şu kurallara dikkat edilmelidir:

- Ters yönde bir sinyal akışına neden olabilecek dallar oluşturamazsınız.

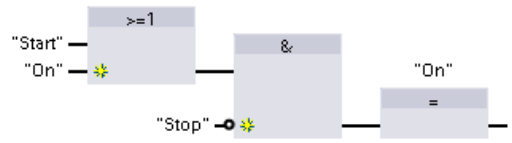


- Kısa devreye yol açabilecek dallar oluşturamazsınız.



7.5.2 Fonksiyon planı (FBD)

Aynı LAD gibi, FBD DE bir grafiksel programlama dilidir. Bağlantı lojiğinin gösterimi, Boole Cebrinde alışılagelmiş olan grafiksel sembolleri temel alır.



Karmaşık işlemler için bağlantılar oluşturmak üzere, kutuların arasına paralel dallar ekleyebilirsiniz.

Aritmetik fonksiyonlar ve diğer karmaşık fonksiyonlar doğrudan lojik kutularıyla bağlantılı olarak gösterilebilir.
STEP 7, bir FBD ağındaki maksimum yordam sayısını (satırlar ve sütunlar) sınırlamaz.

7.5.3 SCL

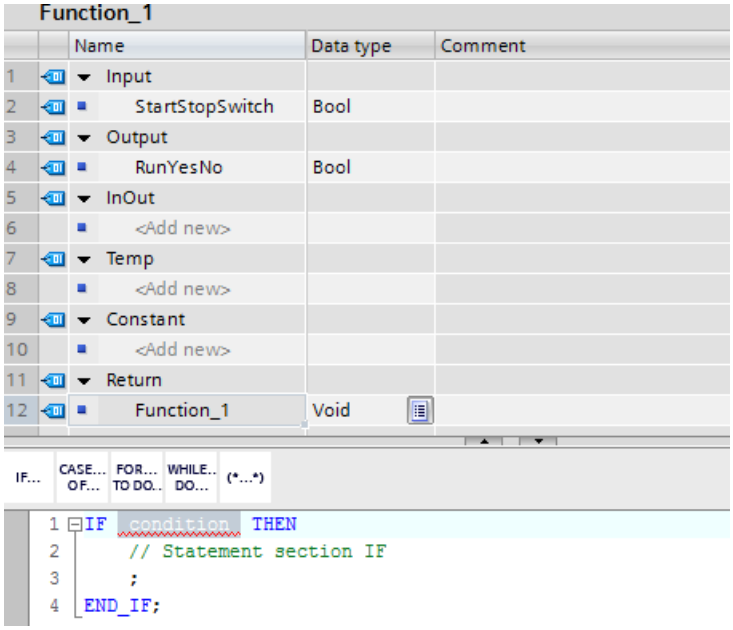
Structured Control Language (SCL), SIMATIC S7-CPU'ları için kullanılan daha yüksek seviyede, PASCAL tabanlı bir programlama dilidir. SCL, STEP 7'nin blok yapısını (Sayfa 171) destekler. Projeniz, şu üç programlama dilinin her birinde program blokları içerebilir: SCL, LAD ve FBD. SCL yordamlarında, programlamanın standart operatörleri kullanılır, örn. atama için (:=), matematiksel fonksiyonlar (toplama için +, çıkarma için -, çarpma için * ve bölme için /). SCL, program kumandası için standart PASCAL işlemlerini de kullanır, örn. IF-THEN-ELSE, CASE, REPEAT-UNTIL, GOTO ve RETURN. SCL programlama dilinin sözdizimiyle ilgili elemanları için tüm PASCAL referanslarını kullanabilirsiniz. Zamanlar ve sayaçlar gibi SCL için kullanılan diğer yordamların birçoğu, LAD ve FBD'deki yordamlara karşılık gelir. Özel yordamlarla ilgili daha fazla bilgiyi, Temel yordamlar (Sayfa 204) ve Gelişmiş yordamlar (Sayfa 311) bölümlerindeki ilgili yordamların altında bulabilirsiniz.

7.5.3.1 SCL programlama editörü

Tüm blok tipleri (OB, FB veya FC) için, blok oluşturulurken bu bloğun SCL programlama dilini kullanması gerektiğini belirtebilirsiniz. STEP 7, aşağıdaki elemanları içeren bir SCL programlama editörüne sahiptir:

- Kod bloğunun parametrelerini tanımlamaya yönelik arayüz bölümü
- Program kodu için kullanılan kod bölümü
- CPU tarafından desteklenen SCL yordamlarının bulunduğu yordam dizini

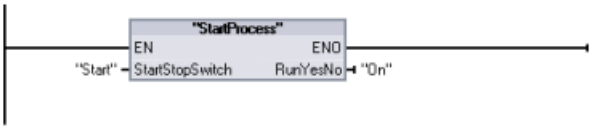
Yordamınız için gereken SCL kodunu doğrudan kod bölümüne girebilirsiniz. Editör, yaygın kod yapıları ve yorumlar için kullanılan butonları içerir. Daha karmaşık yordamlar oluşturmak için, SCL yordamlarını fareyle yordam dizininden programınızın üzerine sürüklemeniz yeterlidir. Ayrıca her metin editöründe bir SCL programı oluşturabilir ve daha sonra dosyayı STEP 7'ye aktarabilirsiniz.



SCL kod bloğunun arayüz bölümünde şu parametre türlerini deklare edebilirsiniz:

- Input, Output, InOut ve Ret_Val: Bu parametreler, giriş ve çıkış değişkenlerini ve kod bloğu için kullanılan geri dönüş değerini tanımlar. Burada girdiğiniz değişken adı, kod bloğunun yürütülmesi sırasında yerel olarak kullanılır. Genelde global değişken adı, değişken tablosunda kullanılmaz.
- Static (sadece FB'ler; yukarıdaki şekilde bir FC gösterilmektedir): Kod bloğu, statik ara sonuçları örnek veri bloğuna kaydetmek için statik değişkenler kullanır. Blok, statik verileri, bu verilerin üzerine yazılana kadar, yani kısmen birçok çevrim boyunca kaydeder. Statik verilerle birlikte, bu kod bloğuna çoklu örnek DB'ler olarak çağrılan blokların adları da kaydedilir.
- Temp: Bu parametrelerde, kod bloğunun yürütülmesi sırasında kullanılan geçici değişkenler söz konusudur.
- Constant: Burada, kod bloğunuz için adlandırılmış sabit değerler söz konusudur.

SCL kod bloğunu başka bir kod bloğundan çağırırsanız, SCL kod bloğunun parametreleri girişler veya çıkışlar olarak ortaya çıkar.



Bu örnekte, "Start" ve "On" için kullanılan değişkenler (sistemin değişken tablosundan gelirler) SCL programının deklarasyon tablosundaki "StartStopSwitch" ve "RunYesNo" değişkenlerine karşılık gelir.

7.5.3.2 SCL ifadeleri ve işlemleri

SCL ifadesinin oluşturulması

Bir SCL ifadesi, bir değeri hesaplamak için kullanılan bir formüldür. İfade, işlemciler ve operatörlerden (*, /, + veya -) oluşur. İşlemcilerde değişkenler, sabitler veya ifadeler söz konusu olabilir.

İfadenin değerlendirilmesi işlemi, aşağıdaki faktörler aracılığıyla belirlenen belirli bir sırayla gerçekleştirilir:

- Her operatör, ön tanımlı bir önceliğe sahiptir ve en yüksek önceliğe sahip işlem önce yürütülür.
- Aynı önceliğe sahip operatörlerde operatörler soldan sağa doğru işlenir.
- Parantezler, birlikte değerlendirilmesi gereken bir dizi operatörü tanımlar.

Bir ifadenin sonucuyla, programınız tarafından kullanılan değişkenlerden birine bir değer atanabilir; sonuç, bir kumanda yordamı için koşul olarak veya başka bir SCL yordamı için parametre olarak ya da bir kod bloğunu çağırmak için kullanılabilir.

Tablo 7-2 SCL'deki operatörler

Tip	İşlem	Operatör	Öncelik
Parantezler	*İfade*)	(,)	1
Aritmetik	Üs	**	2
	Ön işaret (tek bileşenli artı)	+	3
	Ön işaret (tek bileşenli eksi)	-	3
	Çarpma	*	4
	Bölme	/	4
	Modulo fonksiyonu	MOD	4
	Toplama	+	5
	Çıkarma	-	5
Karşılaştırma	Küçüktür	<	6
	Küçüktür veya eşittir	<=	6
	Büyüktür	>	6
	Büyüktür veya eşittir	>=	6
	Eşittir	=	7

Tip	İşlem	Operatör	Öncelik
Karşılaştırma	Eşit değil	<>	7
Bit bağlantısı	Olumsuzlama (tek bileşenli)	NOT	3
	Mantıksal VE bağlantısı	AND veya &	8
	Mantıksal özel VEYA bağlantısı	XOR	9
	Mantıksal VEYA bağlantısı	OR	10
Atama	Atama	:=	11

SCL, daha yüksek seviyeli bir programlama dili olarak temel görevler için standart yordamları kullanır:

- Atama yordamı: :=
- Aritmetik fonksiyonlar: +, -, * ve /
- Global değişkenlerin adreslenmesi: "<Değişken adı>" (Çift tırnak içinde değişken adı veya veri bloğu adı)
- Yerel değişkenlerin adreslenmesi: #<Değişken adı> (önünde "#" sembolüyle birlikte değişken adı)

Aşağıdaki örneklerde, farklı kullanım amaçları için farklı ifadeler gösterilmektedir:

`"C" := #A+#B;`	Bir değişkene iki yerel değişkenin toplamını atar
`"Data_block_1".Tag := #A;`	Bir veri bloğu değişkenine atama
`IF #A > #B THEN "C" := #A;`	IF-THEN yordamı için geçerli koşul
`"C" := SQRT (SQR (#A) + SQR (#B));`	SQRT yordamı için kullanılan parametreler

Aritmetik operatörler farklı sayısal veri tiplerini işleyebilir. Sonucun veri tipi, en yüksek değere sahip işlemcilerin veri tipi tarafından belirlenir. Örnek: INT tipindeki bir işlemci ve REAL tipindeki bir işlemciyle gerçekleştirilen bir çarpma işlemi, sonuç olarak REAL tipinde bir değer verir.

Kumanda yordamları

Bir kumanda yordamı, aşağıdaki görevleri gerçekleştiren özel bir SCL ifadesi türüdür:

- Programın dallanması
- SCL program kodunun bölümlerinin tekrarı
- SCL programının diğer bölümlerine geçiş
- Koşullu yürütme

SCL kumanda yordamları IF-THEN, CASE-OF, FOR-TO-DO, WHILE-DO, REPEAT-UNTIL, CONTINUE, GOTO ve RETURN'ü kapsar.

Münferit bir yordam, genelde bir kod satırını kaplar. Bir satıra birden fazla yordam girebilir veya yordamın daha rahat okunmasını sağlamak için yordamı birden fazla kod satırına bölebilirsiniz. Ayırma işaretleri (sekmeler, satır sonları ve ek boşluk karakterleri gibi) sözdizimi kontrolünde göz ardı edilir. END yordamı, kumanda yordamını sonlandırır.

Aşağıdaki örneklerde bir FOR-TO-DO kumanda yordamı gösterilmektedir. (Her iki kodlama da sözdizimi açısından geçerlidir.)

```
FOR x := 0 TO max DO sum := sum + value(x); END_FOR;
```

```
FOR x := 0 TO max DO
    sum := sum + value(x);
END_FOR;
```

Bir kumanda yordamı, bir atlama işaretini de kapsayabilir. Yordamın başına bir atlama işareti ve onu takiben bir iki nokta üst üste işareti konur:

```
Atlama işareti: <Yordam>;
```

TIA Portal bilgi sisteminde SCL programlama diliyle ilgili eksiksiz bir referans sunulmaktadır.

Koşullar

Bir koşul, sonucu BOOL tipinde (DOĞRU veya YANLIŞ değeri) olan bir karşılaştırma ifadesi veya bir mantıksal ifadedir. Aşağıdaki örneklerde çeşitli türlerde koşullar gösterilmektedir:

`#Sıcaklık > 50`	İlişkisel ifade
`#Sayaç <= 100`	
`#CHAR1 < 'S'`	
`(#Alpha <> 12) AND NOT #Beta`	Karşılaştırma ifadesi ve mantıksal ifade
`5 + #Alpha`	Aritmetik ifade

Bir koşul, aritmetik ifadeler kullanabilir:

- Sonuç sıfır dışında herhangi bir değer ise, ifadenin koşulu DOĞRU olur.
- Sonuç sıfır ise, ifadenin koşulu YANLIŞ olur.

SCL programınızdan diğer kod bloklarının çağrılması

Kullanıcı programınızdaki başka bir kod bloğunu çağırmak için parametrelerle birlikte FB'nin veya FC'nin adını (veya mutlak adresini) girmeniz yeterlidir. Bir FB için, FB ile birlikte çağrılması gereken örnek DB'yi belirtmeniz gerekir.

`<DB adı> (parametre listesi)`	Bir örnek olarak çağrı
`<#Örnek adı> (parametre listesi)`	Çoklu örnek olarak çağrı

```
"MyDB"(MyInput:=10, MyInOut:="Tag1");
```

`<FC adı> (parametre listesi)`	Standart çağrı
`<İşlemci>:=<FC adı> (parametre listesi)`	Bir ifadede çağrı

```
"MyFC"(MyInput:=10, MyInOut:="Tag1");
```

Blokları fareyle proje gezgininden SCL programlama editörünün üzerine sürüklemeniz ve parametrelendirme işlemini tamamlamanız da mümkündür.

Blok yorumlarının SCL koduna eklenmesi

SCL kodunuza bir blok yorumu eklemek için metni (* ile *) işaretlerinin arasına yerleştirin. Yorum başına izin verilen satır sayısı sınırsızdır. Hızlı blok yorumları girmek için SCL editöründeki blok yorumu butonunu kullanın:



Adresleme

LAD ve FBD'de olduğu gibi SCL'de de kullanıcı programınızda değişkenler (sembolik adresleme) veya mutlak adresler kullanabilirsiniz. SCL'de bir değişkeni Array indeksi olarak da kullanabilirsiniz.

Mutlak adresleme

%I0.0 **%MB100**	Mutlak adresin önüne "%" sembolü konmalıdır. "%" karakteri eksikse, STEP 7, derleme sırasında tanımsız bir değişken hatası üretir.

Sembolik adresleme

"PLC_Tag_1"	PLC değişken tablosundaki değişken
"Data_block_1".Tag_1	Bir veri bloğundaki değişken
"Data_block_1".MyArray[#i]	Bir veri bloğu Array'indeki Array elemanı

7.5.3.3 PEEK ve POKE yordamlarıyla indekslenmiş adresleme

SCL, veri bloklarından, G/Ç'lerden veya bellekten okuma veya bu bileşenlere yazma işlemlerini gerçekleştirmenizi sağlayan PEEK ve POKE yordamlarını sunar. İşlem için özel bir Bayt veya Bit ofsetiyle parametreleri belirtebilirsiniz.

NOT

PEEK ve POKE yordamlarını veri bloklarıyla birlikte kullanmak için, standart veri blokları (optimize edilmiş veri blokları değil) kullanmanız gerekir. Ayrıca PEEK ve POKE yordamlarının sadece verileri aktardığına dikkat edin. Adreslerdeki veri tiplerine dikkat edilmez.

```
PEEK(area:=_in_,
     dbNumber:=_in_,
     byteOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından byteOffset tarafından belirtilen Bayt'ı okur.

Bir veri bloğuna verilen referans örneği:

```
%MB100 := PEEK(area:=16#84,
dbNumber:=1, byteOffset:=#i);
```

EB3 girişine verilen referans örneği:

```
%MB100 := PEEK(area:=16#81,
dbNumber:=0, byteOffset:=#i); // when
#i = 3
```

```
PEEK_WORD(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından byteOffset tarafından belirtilen sözcüğü okur.

Örnek:

```
%MW200 := PEEK_WORD(area:=16#84,
dbNumber:=1, byteOffset:=#i);
```

```
PEEK_DWORD(area:=_in_,
           dbNumber:=_in_,
           byteOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından byteOffset tarafından belirtilen çift sözcüğü okur.

Örnek:

```
%MD300 := PEEK_DWORD(area:=16#84,
dbNumber:=1, byteOffset:=#i);
```

```
PEEK_BOOL(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          bitOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından bitOffset ve byteOffset tarafından belirtilen Boole değerini okur.

Örnek:

```
%MB100.0 := PEEK_BOOL(area:=16#84,
dbNumber:=1, byteOffset:=#ii,
bitOffset:=#j);
```

```
POKE(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          value:=_in_);
```

Değeri (Bayt, sözcük veya çift sözcük) belirtilen veri bloğunun, G/Ç'nin veya bellek alanının belirtilen byteOffset'ine yazar.

Bir veri bloğuna verilen referans örneği:

```
POKE(area:=16#84, dbNumber:=2,
byteOffset:=3, value:="Tag_1");
```

AB3 çıkışına verilen referans örneği:

```
POKE(area:=16#82, dbNumber:=0,
byteOffset:=3, value:="Tag_1");
```

```
POKE_BOOL(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          bitOffset:=_in_,
          value:=_in_);
```

Boole değerini, belirtilen veri bloğunun, G/Ç'nin veya bellek alanının belirtilen bitOffset'ine ve byteOffset'ine yazar.

Örnek:

```
POKE_BOOL(area:=16#84, dbNumber:=2,
byteOffset:=3, bitOffset:=5,
value:=0);
```

```
POKE_BLK(area_src:=_in_,
          dbNumber_src:=_in_,
          byteOffset_src:=_in_,
          area_dest:=_in_,
          dbNumber_dest:=_in_,
          byteOffset_dest:=_in_,
          count:=_in_);
```

"count" ifadesinin altında belirtilen sayıda Bayt'ı; belirtilen kaynak veri bloğunun, kaynak G/Ç'nin veya kaynak bellek alanının belirtilen byteOffset'i ile başlayarak, belirtilen hedef veri bloğunun, hedef G/Ç'nin veya hedef bellek alanının belirtilen byteOffset'ine yazar.

Örnek:

```
POKE_BLK(area_src:=16#84,
dbNumber_src:=#src_db,
byteOffset_src:=#src_byte,
area_dest:=16#84,
dbNumber_dest:=#src_db,
byteOffset_dest:=#src_byte,
count:=10);
```

PEEK ve POKE yordamlarında "area", "area_src" ve "area_dest" parametreleri için aşağıdaki değerler geçerlidir. Veri bloğu olarak diğer alanlar için dbNumber parametresi 0 olmalıdır.

16#81	I
16#82	Q
16#83	M
16#84	DB

7.5.4 LAD, FBD ve SCL'deki EN ve ENO

Bir yordam için "sinyal akışının" (EN veENO) belirlenmesi

Bazı yordamlar (örn. matematiksel yordamlar ve aktarma yordamları) EN ve ENO için parametreleri gösterir. Bu parametreler, LAD veya FBD'deki sinyal akışıyla ilgilidir ve yordamın bu çevrimde yürütülüp yürütülmeyeceğini belirlerler. SCL'de ENO parametresini bir kod bloğu için de belirtebilirsiniz.

- EN (serbest bırakma girişi) bir Boole girişidir. Kutunun yürütülebilmesi için bu girişte sinyal akışı (EN = 1) mevcut olmalıdır. Bir LAD kutusunun EN girişi doğrudan sol baraya bağlandıysa, yordam her zaman yürütülür.
- ENO (serbest bırakma çıkışı) bir Boole çıkışıdır. Bir kutunun EN girişinde bir sinyal akışı mevcutsa ve kutu hatasız bir şekilde yürütülürse, ENO çıkışı, sinyal akışını (ENO = 1) bir sonraki elemana aktarır. Kutu yürütülürken bir hata oluşursa, sinyal akışı, hataya neden olan kutu yordamında sonlandırılır (ENO = 0).

Tablo 7-3 EN ve ENO için kullanılan işlemciler

Program editörü	Girişler/çıkışlar	İşlemciler	Veri tipi
LAD	EN, ENO	Sinyal akışı	Bool
FBD	EN	E, E:P, A, M, DB, Temp, sinyal akışı	Bool
	ENO	Sinyal akışı	Bool
SCL	EN[1]	DOĞRU, YANLIŞ	Bool
	ENO[2]	DOĞRU, YANLIŞ	Bool

[1] EN'nin kullanımı sadece FB'lerde mümkündür.

[2] ENO'nun kullanımı SCL kod bloğu ile opsiyoneldir. SCL derleme işlemini, ENO, kod bloğu düzenleme işlemi tamamlanırken ayarlanacak şekilde yapılandırmanız gerekir.

ENO'nun ayarlanmasını SCL'de yapılandırma

SCL derleme işlemini, ENO ayarlanacak şekilde yapılandırmak için aşağıdaki yöntemi izleyin:

1. "Ek özellikler" menüsünde "Ayarlar" komutunu seçin.
2. "PLC programlama" özelliklerini genişletin ve "SCL (Structured Control Language)"yi seçin.
3. "ENO'yu otomatik olarak ayarla" opsiyonunu seçin.

ENO'nun program kodunda kullanımı

ENO'yu, örneğin ENO'yu bir PLC değişkenine atayarak veya ENO'yu yerel bir blokta değerlendirerek program kodunuzda da kullanabilirsiniz.

Örnekler:

```
"MyFunction"
  ( IN1 := … ,
    IN2 := … ,
    OUT1 => #myOut,
    ENO => #statusFlag ); // PLC etiketi statusFlag, ENO'nun
değerini tutar


"MyFunction"
  ( IN1 := …
    IN2 := … ,
    OUT1 => #myOut,
    ENO => ENO );  // "MyFunction"ın blok durumu markörü
                   // yerel blokta saklanır


IF ENO = TRUE THEN
    // kodu ancak MyFunction gerçek ENO'yu döndürürse yürütün
```

Ret_Val veya Status parametrelerinin ENO üzerindeki etkisi

Örn. iletişim yordamları veya karakter dizilerini dönüştürmek için kullanılan yordamlar gibi bazı yordamlar, yordamın işlenmesiyle ilgili bilgileri içeren bir çıkış parametresine sahiptir. Bazı yordamlar örneğin genelde, -32768 ila +32767 aralığında durum bilgilerini içeren Int veri tipinde bir Ret_Val (geri dönüş değeri) parametresine sahiptir. Diğer yordamlar tipik olarak, 16#0000 ila 16#FFFF onaltılı aralığında durum bilgilerini kaydeden Word veri tipindeki Status parametresine sahiptir. Ret_Val veya Status parametrelerinden birindeki sayısal değer, ilgili yordama ait ENO'nun durumunu belirtir.

- Ret_Val: 0 ile 32767 arasındaki bir değer, ENO'yu tipik olarak = 1 (veya DOĞRU) olarak ayarlar. -32768 ile -1 arasındaki bir değer, ENO'yu tipik olarak = 0 (veya YANLIŞ) olarak ayarlar. Ret_Val'i değerlendirmek için gösterimi onaltılı olarak değiştirin.
- Status: 16#0000 ile 16#7FFF arasındaki bir değer, ENO'yu tipik olarak = 1 (veya DOĞRU) olarak ayarlar. 16#8000 ile 16#FFFF arasındaki bir değer, ENO'yu tipik olarak = 0 (veya YANLIŞ) olarak ayarlar.

Birden fazla çevrim boyunca yürütülen yordamlar, yordamın aktif olduğunu, ancak yürütme işleminin henüz tamamlanmadığını belirtmek için sıklıkla Busy (Bool) parametresine sahiptir. Bu tür yordamlar sıklıkla Done (Bool) parametresine ve Error (Bool) parametresine de sahiptir. Done, yordamın hatasız bir şekilde yürütüldüğünü ve Error ise yordamın hatayla tamamlandığını gösterir.

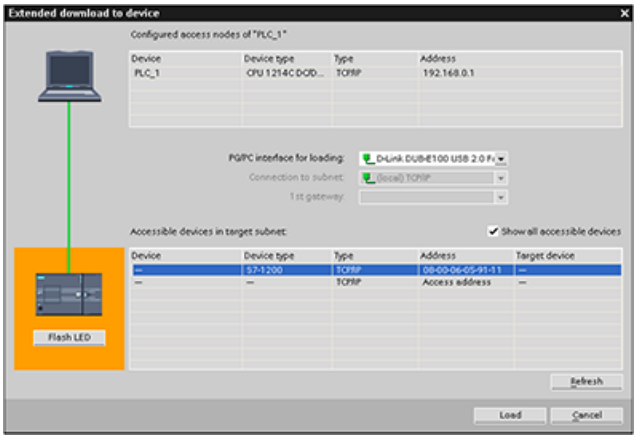
- Busy = 1 (veya DOĞRU) ise, ENO = 1 (veya DOĞRU) olur.
- Done = 1 (veya DOĞRU) ise, ENO = 1 (veya DOĞRU) olur.
- Error = 1 (veya DOĞRU) ise, ENO = 0 (veya YANLIŞ) olur.

Ayrıca bakınız

OK (geçerliliği kontrol et) ve NOT_OK (geçersizliği kontrol et) (Sayfa 225)

7.6 Program elemanlarının yüklenmesi

Projenizin elemanlarını programlama cihazından CPU'ya yükleyebilirsiniz. Bir projeyi yüklerseniz, CPU, kullanıcı programını (OB'ler, FC'ler, FB'ler ve DB'ler) dahili yükleme belleğine veya bir SIMATIC memory card mevcutsa harici yükleme belleğine (kart) yükler.



Projenizi, programlama cihazından CPU'ya yükleyebilir ve bunu aşağıdaki konumlardan birinden yapabilirsiniz:

- Proje gezgini: Farenin sağ tuşuyla program elemanına tıklayın ve ardından "CPU'ya yükle" öğesini seçin.
- Çevrimiçi menü: "Cihaza yükle" opsiyonuna tıklayın.
- Sembol çubuğu: "Cihaza yükle" sembolüne tıklayın.
- Cihaz konfigürasyonu: Farenin sağ tuşuyla CPU'ya tıklayın ve yüklenecek elemanları seçin.

Bir program bloğu için bir zorunlu şifreli dinamik bağlantı (Sayfa 161) belirlendiyse, korunan blokların yüklenmesi için şifrenin girilmesi gerekir. Bu tür kopyalama koruması birden çok blok için ayarlandıysa, korunan blokların her biri için bloğu yükleme şifresinin girilmesi gerekir.

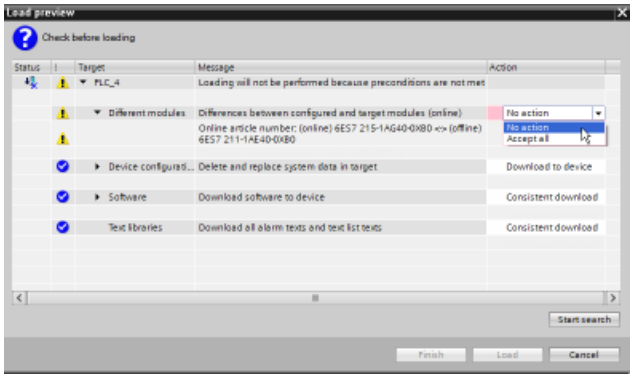
NOT

Bir programın yüklenmesi nedeniyle kalıcı bellekteki değerler silinmez ve değiştirilmez. Kalıcı belleği yükleme işleminden önce silmek için CPU'yu sıfırlayarak fabrika ayarlarına döndürün.

TIA Portal'dan S7-1200 CPU'sundaki bir memory card'a Temel HMI panelleri için bir panel projesi (Sayfa 32) de yükleyebilirsiniz.

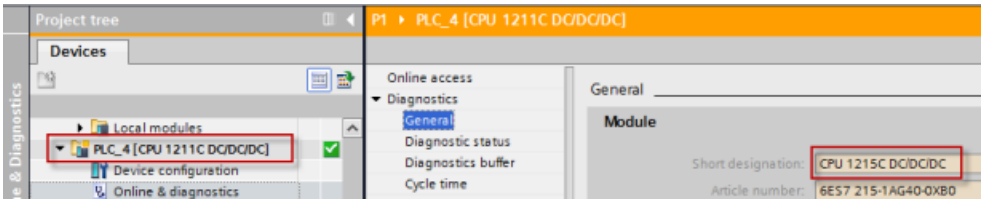
Yapılandırılmış CPU'nun bağlı CPU'ya göre farklı olması durumunda hedef sisteme yükleme

STEP 7 ve S7-1200, bağlı CPU'nun, yapılandırılmış CPU'dan yüklenen verileri, projenin bellek gerekliliklerini ve çevrebirimlerinin uyumluluğunu temel alarak kaydetme özelliğine sahip olması durumunda hedef sisteme yüklemeye izin verir. Çevrebirimleri uyumlu olduğu ve bellek yeterli boyutta olduğu için konfigürasyonu ve programı bir CPU'dan daha büyük bir CPU'ya, örneğin bir CPU 1211C DC/DC/DC'den bir CPU 1215C DC/DC/DC'ye yükleyebilirsiniz. Bu durumda, yükleme işlemi sırasında "Yapılandırılmış bloklarla hedef bloklar (çevrimiçi) arasında farklar mevcut" uyarısı ve "Önizlemeyi yükle" iletişim kutusunda sipariş numaraları ve Firmware sürümleri görüntülenir. Yükleme işlemine devam edilmemesi gerekirse "Eylem yok" öğesini veya yükleme işlemini devam ettirmek istiyorsanız "Tümünü kabul et" öğesini seçmeniz gerekir:



NOT

Yapılandırılmış CPU'yu bağlı durumdaki başka bir CPU'ya yükleme işleminden sonra çevrimiçi olursanız (Sayfa 1117), yapılandırılmış CPU için kullanılan proje, proje ağacında çevrimiçi durum göstergeleriyle birlikte görüntülenir. Ancak "Çevrimiçi ve teşhis" görünümünde gerçekten bağlı olan CPU modül tipi görüntülenir.

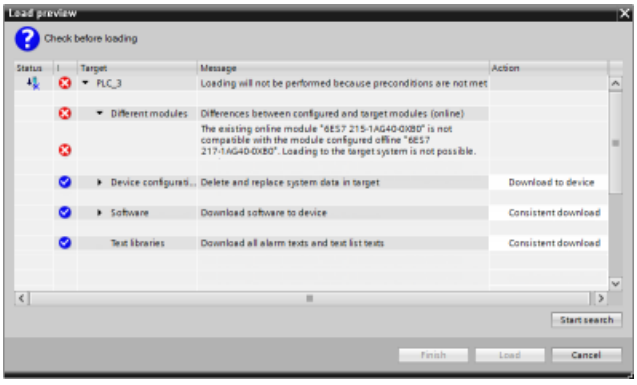


Cihaz konfigürasyonunda, yapılandırılmış CPU, bağlı CPU ile aynı modül tipine sahip olacak şekilde cihazınızı değiştirebilirsiniz (Sayfa 144). Bir cihazı değiştirmeye çalışırsanız, "Cihazı değiştir" iletişim kutusunda, uyumlulukla ilgili eksiksiz bilgiler görüntülenir.

Bağlı CPU'nun, yapılandırılmış CPU'dan yüklenen verileri kaydetme özelliğine sahip olmaması durumunda, STEP 7 ve S7-1200 hedef sisteme yüklemeyi yasaklar. Örneğin donanım konfigürasyonunu ve programı aşağıdaki durumlarda yükleyemezsiniz:

- Çalışma belleği yeterli boyutta olmadığı için bir 1215C DC/DC/DC CPU'dan bir 1212C DC/DC/DC CPU'ya
- Çevrebirimlerinde farklar olduğu için bir 1211C DC/DC/Röle CPU'dan bir 1211C DC/DC/DC CPU'ya
- 1217C CPU 1,5-V çıkışlara sahip olduğu için bir 1217C DC/DC/DC CPU'dan herhangi bir 1211C CPU'ya, 1212C CPU'ya, 1214C CPU'ya veya 1215C CPU'ya
- Firmware sürümü geriye doğru uyumlu olmadığı için bir V4.6 1214C CPU'dan bir V4.5.x veya daha eski bir 1214C CPU'ya

Bu tür durumlarda "Önizlemeyi yükle" iletişim kutusunda bir hata görüntülenir:



Gizli PLC konfigürasyon verilerinin korunduğu CPU'lar

Gizli PLC konfigürasyon verilerinin korunması (Sayfa 154)'u yapılandırdıysanız, şu hususlara dikkat edin:

- Cihaz gizli PLC konfigürasyon verilerinin korunması için tanımlanan şifreye sahip değilse, hedef sisteme ilk yükleme sırasında TIA Portal, gizli PLC konfigürasyon verilerinin korunması için tanımlanan şifreyi girmenizi ister.
- Cihaz gizli PLC konfigürasyon verilerinin korunması için tanımlanan şifreye sahipse, STEP 7 projesindeki şifre ve cihazdaki şifre aynı olmalıdır. Şifreler aynı değilse, projeyi CPU'ya yükleyemezsiniz. Gizli PLC konfigürasyon verilerinin korunması için kullanılan şifreyi silebilir veya bunun için cihazdakiyle aynı şifreyi belirleyebilirsiniz. Gizli PLC konfigürasyon verilerinin korunması için kullanılan şifreyi "Çevrimiçi ve teşhis (Sayfa 1121)" öğesinin altında da belirleyebilir veya silebilirsiniz.

Hedef sisteme yükleme işlemi başarısız olduktan sonra geri yükleme

Yükleme işlemi başarısız olursa, denetçi penceresindeki "Bilgi" sekmesinde bu durumun nedeni görüntülenir. Teşhis tamponu da bilgi sağlar. Bir yükleme işlemi başarısız olduktan sonra, hedef sisteme yükleme işlemini başarıyla gerçekleştirmek için aşağıdaki yöntemi izleyin:

1. Sorunu, hata mesajında açıklandığı şekilde giderin.
2. Verileri hedef sisteme yükleme işlemini yeniden deneyin.

Yükleme işlemi nadiren başarıyla gerçekleştirilir, ancak daha sonra CPU'nun kapatılması ve yeniden açılması işlemi başarısız olur. Bu durumda teşhis tamponunda muhtemelen aşağıdaki gibi bir hata görüntülenir:

- 16# 02:4175 -- CPU hatası: Memory card değerlendirme hatası: CPU konfigürasyon açıklamasında güncel kart tipinin sürümü bilinmiyor veya uyumlu değil: Hiçbir memory card fonksiyonu tamamlanmadı/iptal edilmedi, yeni çalışmaya başlama blokajı belirlendi: ..- Memory card yok, yanlış tip, yanlış içerik veya korumalı

Bu durum ortaya çıkarsa ve sonraki yükleme denemeleri başarısız olursa, dahili yükleme belleğini veya harici yükleme belleğini silmeniz gerekir:

1. Dahili yükleme belleğiyle çalışıyorsanız, CPU'yu sıfırlayarak fabrika ayarlarına döndürün.
2. Bir SIMATIC memory card ile çalışıyorsanız, kartı çıkartın ve memory card'ın içeriğini silin (Sayfa 118). Ardından kartı yerine takın.
3. Donanım konfigürasyonunu ve yazılımı hedef sisteme yükleyin.

7.7 Çevrimiçi CPU'nun ve çevrimdışı projenin senkronize edilmesi

Proje bloklarını CPU'ya yüklerseniz, CPU, son yükleme işleminden sonra çevrimiçi CPU'daki blokların veya değişkenlerin değişip değişmediğini algılayabilir. Bu durumda CPU, size değişiklikleri senkronize etme olanağı sunar. Bu, projeyi CPU'ya yüklemeden önce değişiklikleri çevrimiçi CPU'dan projeye yükleyebileceğiniz anlamına gelir. Çevrimiçi CPU'daki değişiklikler çok sayıda faktörden kaynaklanabilir:

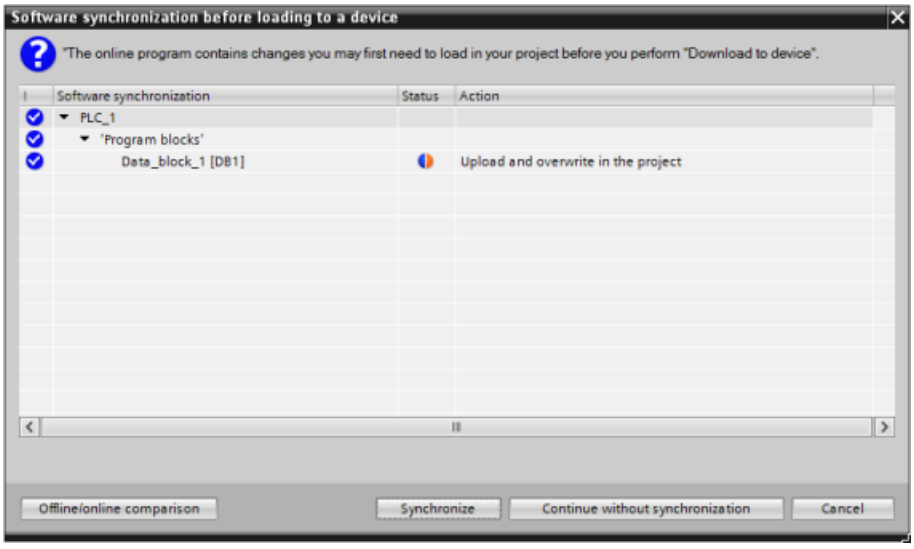
- Örneğin WRIT_DBL yordamı (Sayfa 490) veya bir reçetenin yüklenmesi nedeniyle çalışma zamanı esnasında veri bloğu değişkenlerinin başlangıç değerlerindeki değişiklikler
- Aşağıdaki koşullardan birinin veya daha fazlasının mevcut olduğu, "ikincil" bir projeden (son yükleme işleminin projesinden farklı bir proje) gerçekleştirilen bir yükleme işlemi:
  - Çevrimiçi CPU, projede mevcut olmayan program bloklarını içeriyor.
  - Veri bloğu değişkenleri veya blok öznitelikleri, çevrimdışı projede ve çevrimiçi CPU'da farklı.
  - Çevrimiçi CPU'da, çevrimdışı projede mevcut olmayan PLC değişkenleri mevcut.

NOT

Son yükleme işleminde kullandığınız projede blokları ve değişkenleri düzenlerseniz, senkronizasyon açısından seçim yapmanız gerekmez. STEP 7 ve CPU, çevrimdışı projedeki değişikliklerin çevrimiçi CPU'ya göre daha yeni olduğunu algılar ve normal yükleme işlemini devam ettirir.

Senkronizasyon opsiyonları

Bir projeyi CPU yüklerken, STEP 7'nin çevrimiçi CPU'daki veri bloklarının veya değişkenlerin projedeki değerlerden daha yeni olduğunu algılaması durumunda senkronizasyon iletişim kutusu görüntülenir. Örnek: WRIT_DBL STEP 7 programı yürütülürse ve Data_block_1'deki bir değişken için kullanılan başlangıç değeri değiştiyse, hedef sistem yükleme işlemini başlattığınızda STEP 7'de aşağıdaki senkronizasyon iletişim kutusu görüntülenir:



Bu iletişim kutusunda, farklılıklar bulunan program blokları listelenir. Bu iletişim kutusunda aşağıdaki olanaklar kullanımınıza sunulur:

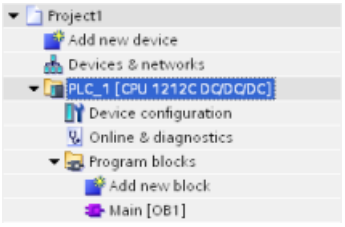
- Çevrimiçi/çevrimdışı karşılaştırma: STEP 7; bu butona tıklarsanız, proje için kullanılan program bloklarını, sistem bloklarını, teknoloji nesnelerini, PLC değişkenlerini ve PLC veri tiplerini çevrimiçi CPU ile karşılaştırmalı olarak (Sayfa 1128) görüntüler. Her nesne için bir tıklama işlemiyle zaman notu da dahil olmak üzere farkların ayrıntılı bir analizini görüntüleyebilirsiniz. Bu bilgilerin yardımıyla, çevrimiçi CPU ile proje arasındaki farklılıkları nasıl ele almak istediğinize karar verebilirsiniz.
- Senkronize et: Bu butona tıklarsanız, STEP 7; veri bloklarını, değişkenleri ve diğer nesneleri çevrimiçi CPU'dan projeye yükler. Ardından, programın yürütülmesi yine projenin CPU ile senkronize olmamasına neden olmadığı sürece, programı hedef sisteme yükleme işlemini devam ettirebilirsiniz.
- Senkronizasyon olmadan devam et: STEP 7, bu butona tıklarsanız projeyi CPU'ya yükler.
- İptal: Bu butona tıklarsanız, yükleme işlemini iptal edersiniz.

7.8 Çevrimiçi CPU'dan yükleme

Program bloklarını bir çevrimiçi CPU'ya veya programlama cihazınıza bağlı bir memory card'a da kopyalayabilirsiniz.

Kopyalanan program blokları için çevrimdışı projeyi hazırlayın:

1. Çevrimiçi CPU'ya karşılık gelen bir CPU ekleyin.
2. CPU düğümlerini, "Program blokları" klasörü görüntülenecek şekilde genişletin.



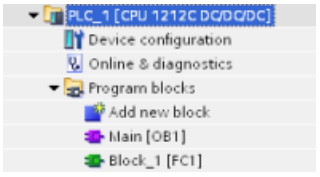
Program bloklarını çevrimiçi CPU'dan çevrimdışı projeye yüklemek için aşağıdaki yöntemi izleyin:

1. Çevrimdışı projede "Program blokları" klasörüne tıklayın.
2. "Çevrimiçi bağlan" butonuna tıklayın.
3. "CPU'dan yükle" butonunu seçin.
4. İşlemi, CPU'dan yükle (Sayfa 1117) iletişim kutusunda onaylayın.





STEP 7, yükleme işleminden sonra, projedeki yüklenen tüm program bloklarını gösterir.



7.8.1 Çevrimiçi CPU'nun çevrimdışı CPU ile karşılaştırması

STEP 7'deki "Karşılaştır" editörüyle (Sayfa 1128), çevrimiçi ile çevrimdışı proje arasındaki farkları belirleyebilirsiniz. Bu özellik, programın CPU'dan yüklenmesinden önce faydalı olabilir.

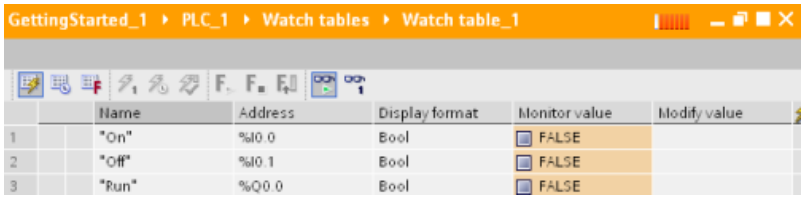
7.9 Programın hatalarının ayıklanması ve test edilmesi

7.9.1 CPU'daki verilerin izlenmesi ve kumanda edilmesi

Çevrimiçi CPU'daki değerleri izleyebilir ve kumanda edebilirsiniz (bkz. aşağıdaki tablo).

Tablo 7-4 Verilerin STEP 7 ile izlenmesi ve kumanda edilmesi

Editör	İzleme	Kumanda	Force fonksiyonu
İzleme tablosu	Evet	Evet	Hayır
Force tablosu	Evet	Hayır	Evet
Programlama editörü	Evet	Evet	Hayır
Değişken tablosu	Evet	Hayır	Hayır
DB editörü	Evet	Hayır	Hayır



Bir izleme tablosunda izleme



LAD editöründe izleme

"Çevrimiçi ve teşhis" bölümünde, CPU'daki verilerin izlenmesi ve kumanda edilmesi (Sayfa 1130) konusuyla ilgili daha fazla bilgi bulabilirsiniz.

7.9.2 İzleme tabloları ve Force tabloları

"İzleme tablolarının" yardımıyla, çevrimiçi CPU tarafından yürütülen bir kullanıcı programının değerlerini denetleyebilir ve değiştirebilirsiniz. Çok sayıda test ortamını kapsamak için projenizde farklı izleme tabloları oluşturabilir ve kaydedebilirsiniz. Bu sayede, örneğin işletime alma sırasında veya servis ve bakım amacıyla testler gerçekleştirebilirsiniz.

Bir izleme tablosu ile kullanıcı programının CPU tarafından yürütülmesi işlemini denetleyebilir ve yürütme işlemine müdahale edebilirsiniz. Sadece kod bloklarının ve veri bloklarının değişkenleri yerine, ayrıca girişler ve çıkışlar (G ve Ç), çevrebirim girişleri (E:P), markörler (M) ve veri blokları da (DB) dahil olmak üzere CPU'nun bellek alanları için de ilgili değerleri çağırabilir ve değiştirebilirsiniz.

İzleme tablosu ile STOP işletim durumunda bulunan bir CPU'nun fiziksel çıkışlarını (A:P) etkinleştirebilirsiniz. Örneğin, CPU'nun kablajını test ederken çıkışlara belirli değerleri atayabilirsiniz.

STEP 7, ayrıca değişkenleri belirli bir değeri almaya zorlamak için bir Force tablosu sunar. Force fonksiyonu ile ilgili daha fazla bilgiyi, "Çevrimiçi ve teşhis" bölümündeki CPU'da değerlere Force fonksiyonunun uygulanması (Sayfa 1138) alt bölümünde bulabilirsiniz.

NOT

Force değerleri CPU'ya kaydedilir ve izleme tablosuna kaydedilmez.

Hiçbir girişe Force fonksiyonu uygulayamazsınız ("E" adresi). Ancak bir çevrebirim girişine Force fonksiyonu uygulayabilirsiniz. Bir çevrebirim girişine Force fonksiyonu uygulamak için adrese bir ":P" ifadesi ekleyin (örnek: "On:P").

STEP 7 ile ayrıca, program değişkenlerini tetikleme koşulları aracılığıyla takip etmek ve kaydetmek (Sayfa 1149) mümkündür.

7.9.3 Kullanımı görüntülemek için çapraz referans

Denetçi penceresinde, bir nesnenin tüm proje dahilinde örn. kullanıcı programında, CPU'da veya HMI cihazlarında nasıl kullanılacağıyla ilgili çapraz referanslar görüntülenir. "Çapraz referans" sekmesinde, seçilmiş bir nesnenin nerede ve başka hangi nesneler tarafından kullanıldığı örnekler görüntülenir. Denetçi penceresi ayrıca, çapraz referanslar dahilinde sadece çevrimiçi olarak mevcut olan bloklar içerir. Çapraz referansları görüntülemek için "Çapraz referanslar" komutunu seçin. (Bu komut, proje görünümünde "Araçlar" menüsünde yer alır.)

NOT

Çapraz referans bilgilerini görüntülemek için editörün kapatılması gerekmez.

Çapraz referans listesinin girişleri farklı şekillerde sıralanabilir. Çapraz referanslar listesi, kullanıcı programında bellek adreslerinin ve değişkenlerin kullanımıyla ilgili bir genel bakış sunar.

- Bir program oluşturur veya bir programı değiştirirseniz, kullanılan işlemciler, değişkenler ve blok çağrıları ile ilgili bir genel bakış sağlayın.
- Çapraz referanslardan doğrudan işlemcilerin ve değişkenlerin kullanıldığı noktaya geçebilirsiniz.
- Bir program testi veya bir hata giderme işlemi sırasında; hangi bellek adresinin hangi bloktaki hangi komut tarafından işlendiği, hangi değişkenin hangi ekranda kullanıldığı ve hangi bloğun başka hangi blok tarafından çağrıldığıyla ilgili bilgilere sahip olabilirsiniz.

Tablo 7-5 Çapraz referans elemanları

Sütun	Açıklama
Nesne	Belirtilen alt nesneleri kullanan veya alt nesneler tarafından kullanılan nesnenin adı
Adet	Kullanım sayısı
Kullanım	Kullanım yeri, örn. bir ağ
Özellik	Referanslanan nesnelerin özel özellikleri, örn. çoklu örnek deklarasyonlarındaki değişken adları
Ne olarak	Örneğin bir örnek DB'nin şablon olarak mı yoksa çoklu örnek olarak mı kullanılacağı gibi nesneyle ilgili ek bilgileri gösterir
Erişim	Erişimin türü, yani işlemcilerde okuma erişiminin (R) ve/veya yazma erişiminin (W) mevcut olup olmadığı
Adres	İşlemcilerin adresi
Tip	Nesnenin hangi tipte ve hangi dilde oluşturulduğu bilgisi
Yol	Nesnenin proje gezginindeki yolu

Kurulu ürünlere bağlı olarak çapraz referans tablosunda ek veya farklı sütunlar görüntülenir.

7.9.4 Çağrı hiyerarşisinin kontrolü için kullanılan çağrı yapısı

Çağrı yapısı, bloğun, kullanıcı programınızın içindeki çağrı hiyerarşisini gösterir. Bu yapı; kullanılan bloklar, başka blokların çağrıları, blokların arasındaki ilişkiler, her bloğa yönelik veri gereklilikleri ve münferit blokların durumu hakkında bir genel bakış sunar. Çağrı yapısındaki bloklar, programlama editörüyle açılabilir ve düzenlenebilir.

Çağrı yapısının görüntülenmesi aracılığıyla, kullanıcı programında kullanılan blokların bir listesini elde edebilirsiniz. STEP 7'de, çağrı yapısının ilk seviyesi vurgulanmış şekilde görüntülenir ve programdaki başka hiçbir blok tarafından çağrılmayan bloklar da görüntülenir. Çağrı yapısının ilk seviyesi, OB'leri ve bir OB tarafından çağrılmayan ilgili FC'leri, FB'leri ve DB'leri içerir. Başka bloklar tarafından çağrılan kod blokları, çağıran bloğun altında girintili olarak görünür. Çağrı yapısında sadece bir kod bloğu tarafından çağrılan bloklar görüntülenir.

Seçerek sadece çağrı yapısı içinde çakışmalara neden olan blokları görüntüleyebilirsiniz. Aşağıdaki koşullar çakışmalara neden olur:

- Kodda daha eski veya daha yeni zaman notlarıyla çağrılar gerçekleştiren bloklar
- Arayüzü değişen bir bloğu çağıran bloklar
- Adresi ve/veya veri tepi değiştirilmiş bir değişken kullanan bloklar
- Bir OB tarafından doğrudan veya dolaylı olarak çağrılmayan bloklar
- Mevcut olmayan veya eksik bir bloğu çağıran bloklar

Birden çok blok çağrısını ve veri bloğunu bir grup halinde birleştirebilirsiniz. Bir açılır liste aracılığıyla farklı çağrı noktalarıyla kurulan bağlantıları görüntüleyebilirsiniz.

Ayrıca zaman notu çakışmalarını göstermek için bir tutarlılık kontrolü gerçekleştirebilirsiniz. Zaman notu çakışmalarına, program oluşturma işlemi sırasında veya sonrasında bir bloğa ait zaman notunun değiştirilmesi neden olabilir. Bu çakışmalar, çağıran ve çağrılan bloklar arasında tutarsızlıklara neden olur.

- Çoğu zaman notu ve arayüz çakışması, kod bloklarının yeniden derlenmesi yoluyla giderilebilir.
- Tutarsızlıklar derleme yoluyla giderilemezse, program editörünü kullanarak, "Ayrıntılar" sütunundaki bağlantı üzerinden sorunun kaynağına gidin. Burada tutarsızlıkları manuel olarak giderebilirsiniz.
- Bloklar kırmızı renkle işaretlenmişse, blokların yeniden derlenmesi gerekir.

Yordamlar

8

8.1 Bit bağlantıları

8.1.1 Bit bağlantısı yordamları

LAD ve FBD, Boolen mantığını çok etkin bir şekilde işler. SCL özellikle karmaşık matematiksel hesaplamalar ve proje kumanda yapılarında çok etkili olsa da, SCL'yi Boolen mantığı için de kullanabilirsiniz.

LAD kontakları

Tablo 8-1 NO kontak ve NC kontaklar

LAD	SCL	Açıklama

"IN" 	**IF in THEN**
	Statement;
	ELSE
	Statement;
	END_IF;
		NO kontak ve NC kontaklar: Kontakları kendi aralarında bağlayabilir ve bu sayede kendi bağlantı lojiğinizi oluşturabilirsiniz. Belirttiğiniz giriş biti E (giriş) veya A (çıkış) bellek algılamasını kullanıyorsa, Bit değeri proses görünümü kaydından okunur. Kumanda prosesinizdeki fiziksel kontak sinyalleri, PLC'nin giriş bağlantılarına bağlanır. CPU, bağlanan giriş sinyallerini sorgular ve girişlerin proses görünümünde ilgili durum değerlerini sürekli olarak günceller.
		Fiziksel bir girişin doğrudan okunması işlemini, E-adresinden sonra ":P" ifadesiyle gerçekleştirebilirsiniz (Örnek: "%E3.4:P"). Doğrudan okuma işleminde Bit verileri değerleri, proses görünümü yerine doğrudan fiziksel girişten okunur. Doğrudan okuma sırasında proses görünümü güncellenmez.

"IN" 	**IF NOT (in)** **THEN** **Statement;** **ELSE** **Statement;** **END_IF;**	

Tablo 8-2 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	Bool	Atanmış Bit

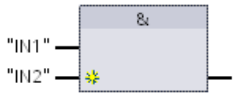
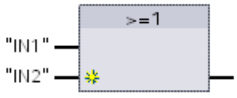
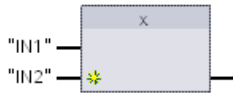
- Atanmış Bit değeri 1'e eşitse NO kontak kapalıdır (AÇIK).
- Atanmış Bit değeri 0'a eşitse NC kontak kapalıdır (AÇIK).
- Seri olarak bağlanan kontaklar mantıksal VE bağlantıları oluşturur.
- Paralel olarak bağlanan kontaklar mantıksal VEYA bağlantıları oluşturur.

FBD'deki VE, VEYA ve XOR kutuları

FBD programlamada LAD kontakları içeren ağlar, kutunun girişleri ve çıkışları için Bit değerlerini belirtebileceğiniz VE (&), VEYA (>=1) ve ÖZEL VEYA (x) kutu ağlarına dönüştürülür. Ayrıca diğer lojik kutularıyla bağlantılar oluşturabilir ve böylece kendi bağlantı lojiğinizi oluşturabilirsiniz. Kutu ağınıza yerleştirildikten sonra, başka girişler eklemek için "Sık kullanılanlar" fonksiyon çubuğundan "Giriş ekle" fonksiyonunu ekleyebilir veya yordam dizinini kutunun giriş tarafının üzerine sürükleyebilirsiniz. Farenin sağ tuşuyla kutunun giriş bağlantısına tıklamanız ve "Giriş ekle" öğesini seçmeniz de mümkündür.

Bir kutunun giriş ve çıkışları başka kutulara bağlanabilir veya bağlanmamış bir giriş için bir Bit adresini veya bir Bit sembolü adını girebilirsiniz. Kutu yordamı yürütülürken giriş durumları ikili kutu bağlantısına ayarlanır ve ardından uygunsa, kutu çıkışı Doğru olarak ayarlanır.

Tablo 8-3 VE, VEYA ve ÖZEL VEYA kutuları

FBD	SCL [1]	Açıklama
	**out := in1 AND in2;**	Çıkışın DOĞRU olması için bir VE kutusunun tüm girişleri DOĞRU olmalıdır
	**out := in1 OR in2;**	Çıkışın DOĞRU olması için bir VEYA kutusunun herhangi bir girişi DOĞRU olmalıdır
	**out := in1 XOR in2;**	Çıkışın DOĞRU olması için bir XOR kutusunun tek sayıdaki girişleri DOĞRU olmalıdır

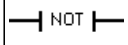
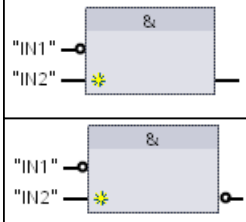
[1] SCL'de: İşlemin sonucunu, başka bir yordamda kullanılabilmesi için bir değişkene atamanız gerekir.

Tablo 8-4 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN1, IN2	Bool	Giriş Bit'i

NOT lojik invertörü

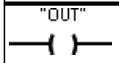
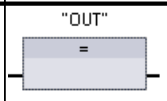
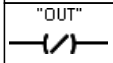
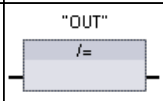
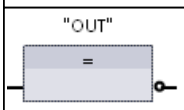
Tablo 8-5 VKE'nin (bağlantı sonucu) ters çevirme

LAD	FBD	SCL	Açıklama
		**NOT**	FBD programlamada, bu kutu bağlantısı için bir lojik invertörü oluşturmak üzere "VKE'yi ters çevir" fonksiyonunu "Sık kullanılanlar" fonksiyon çubuğundan veya yordam dizininden bir girişin veya bir çıkışın üzerine sürükleyebilirsiniz. LAD kontağı NOT, sinyal akışı girişinin mantıksal durumunu tersine çevirir. • NOT kontağına giden bir sinyal akışı mevcut değilse, çıkışta bir sinyal akışı bulunur. • NOT kontağına giden bir sinyal akışı mevcutsa, çıkışta bir sinyal akışı bulunmaz.

Çıkış bobini ve atama kutusu

Bobin çıkışı için kullanılan yordam, bir çıkış Bit'ine bir değer yazar. Belirtilen çıkış Bit'i A bellek algılamasını kullanırsa, CPU, çıkış Bit'ini proses görünümü kaydında açar veya kapatır ve belirtilen Bit'i sinyal akışına uygun şekilde ayarlar. Kumanda aktüatörleriniz için kullanılan çıkış sinyalleri, CPU'nun çıkış klemenslerine bağlanır. CPU, RUN işletim durumunda sürekli olarak giriş sinyallerini sorgular, giriş durumlarını program lojiğine uygun şekilde işler ve ardından, çıkışların proses görünümündeki yeni çıkış değerlerini çıkışlara ayarlayarak tepki verir. CPU, proses görünümü kaydına kaydedilen tepkiyi, bağlanan çıkış klemenslerinin yeni çıkış durumuna aktarır.

Tablo 8-6 Atama yapma ve atamayı iptal etme

LAD	FBD	SCL	Açıklama
		**out := <Boole ifadesi>;**	FBD programlamada, LAD bobinleri atama kutularına (= ve /=) dönüştürülür, bu sırada kutu çıkışı için bir Bit adresi belirtilmelidir. Bir kutunun giriş ve çıkışları başka kutu bağlantılarına bağlanabilir veya bir Bit adresi girebilirsiniz. Bir fiziksel çıkışın doğrudan yazılması işlemini A adresinden sonra ":P" ifadesiyle belirtebilirsiniz (Örnek: "%A3.4:P"). Doğrudan yazma işleminde Bit verileri değerleri proses görünümündeki çıkışa ve doğrudan fiziksel çıkışa yazılır.
	 	**out := NOT <Boole ifadesi>;**	

Tablo 8-7 Parametreler için kullanılan veri tipleri

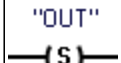
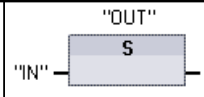
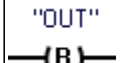
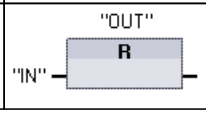
Parametre	Veri tipi	Açıklama
OUT	Bool	Atanmış Bit

- Bir çıkış bobini üzerinden geçen bir sinyal akışı mevcutsa veya bir FBD kutusu "=" etkinleştirildiyse, çıkış Bit'i 1 olarak ayarlanır.
- Bir çıkış bobini üzerinden geçen bir sinyal akışı mevcut değilse veya hiçbir FBD kutusu "=" etkinleştirilmediyse, çıkış Bit'i 0 olarak ayarlanır.
- Tersine çevrilmiş bir çıkış bobini üzerinden geçen bir sinyal akışı mevcutsa veya bir FBD kutusu "/=" etkinleştirildiyse, çıkış Bit'i 0 olarak ayarlanır.
- Tersine çevrilmiş bir çıkış bobini üzerinden geçen bir sinyal akışı mevcut değilse veya hiçbir FBD kutusu "/=" etkinleştirilmediyse, çıkış Bit'i 1 olarak ayarlanır.

8.1.2 Ayarlama ve sıfırlama işlemleri

1 Bit ayarlama ve sıfırlama

Tablo 8-8 S ve R yordamları

LAD	FBD	SCL	Açıklama
		Mevcut değil	Çıkışın ayarlanması: S (Ayarla) etkinleştirildiyse, OUT adresindeki veri değeri 1 olarak ayarlanır. S etkinleştirilmediyse, OUT değişmez.
		Mevcut değil	Çıkışın sıfırlanması: R (Sıfırla) etkinleştirildiyse, OUT adresindeki veri değeri 0 olarak ayarlanır. R etkinleştirilmediyse, OUT değişmez.

1 LAD ve FBD'de: Bu yordamlar, ağdaki herhangi bir konumda eklenebilir.

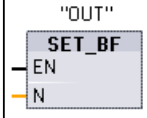
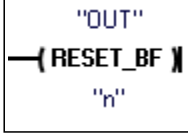
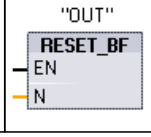
2 SCL'de: Bu fonksiyonu uygulamanızın içinde çoğaltmak için kod yazmanız gerekir.

Tablo 8-9 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN (veya kontak/kumanda lojiğine bağlama)	Bool	İzlenmesi gereken adresin Bit değişkeni
OUT	Bool	Ayarlanması veya sıfırlanması gereken adresin Bit değişkeni

Bit alanının ayarlanması ve sıfırlanması

Tablo 8-10 SET_BF ve RESET_BF yordamları

LAD[1]	FBD	SCL	Açıklama
		Mevcut değil	Bit alanının ayarlanması: SET_BF etkinleştirilirse, 1 veri değeri, OUT adresiyle başlanarak "n" Bit'e yazılır. SET_BF etkinleştirilmezse, OUT değişmez.
		Mevcut değil	Bit alanının sıfırlanması: RESET_BF, 0 veri değerini OUT adresiyle başlayarak "n" Bit'e yazar. RESET_BF etkinleştirilmezse, OUT değişmez.

1 LAD ve FBD'de: Bu yordamlar bir dalda en sağa yerleştirilmelidir.

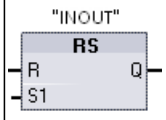
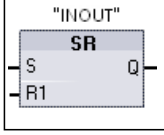
2 SCL'de: Bu fonksiyonu uygulamanızın içinde çoğaltmak için kod yazmanız gerekir.

Tablo 8-11 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
OUT	Bool	Ayarlanması veya sıfırlanması gereken bir Bit alanının başlangıç elemanı (örnek: #MyArray[3])
n	Sabitler (UInt)	Yazılması gereken Bit'lerin sayısı

Flipflop devrelerinin öncelikli olarak ayarlanması ve öncelikli olarak sıfırlanması

Tablo 8-12 RS ve SR yordamları

LAD/FBD	SCL	Açıklama
	Mevcut değil	Flipflop'un sıfırlanması/ayarlanması: RS, ayarlamanın öncelikli olduğu bir Flipflop'tur. Ayarla (S1) ve Sıfırla (R) sinyallerinin her ikisi de doğru ise, INOUT çıkış adresi 1 olur.
	Mevcut değil	Flipflop'un sıfırlanması/ayarlanması: SR, sıfırlamanın öncelikli olduğu bir Flipflop'tur. Ayarla (S) ve Sıfırla (R1) sinyallerinin her ikisi de doğru ise, INOUT çıkış adresi 0 olur.

[1] LAD ve FBD'de: Bu yordamlar bir dalda en sağa yerleştirilmelidir.

[2] SCL'de: Bu fonksiyonu uygulamanızın içinde çoğaltmak için kod yazmanız gerekir.

Tablo 8-13 Parametreler için kullanılan veri tipleri

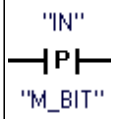
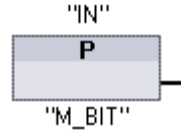
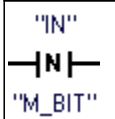
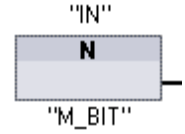
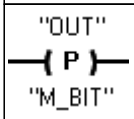
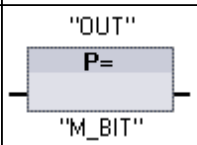
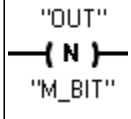
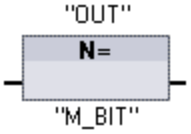
Parametre	Veri tipi	Açıklama
S, S1	Bool	Girişin ayarlanması; 1 baskınlık durumunu belirtir
R, R1	Bool	Girişin sıfırlanması; 1, baskınlık durumunu belirtir
INOUT	Bool	Atanmış Bit değişkeni "INOUT"
Q	Bool	"INOUT" Bit'inin durumunu takip eder

"INOUT" değişkeni, ayarlanacak veya sıfırlanacak olan Bit adresini atar. Opsiyonel Q çıkışı, "INOUT" adresinin sinyal durumunu belirtir.

Yordam	S1	R	"INOUT" Bit'i
RS	0	0	Önceki durum
	0	1	0
	1	0	1
	1	1	1
	S	**R1**	
SR	0	0	Önceki durum
	0	1	0
	1	0	1
	1	1	0

8.1.3 Yükselen kenar ve alçalan kenar işlemleri

Tablo 8-14 "Yükselen kenar" ve "Alçalan kenar" yordamları

LAD	FBD	SCL	Açıklama
		Mevcut değil [1]	İşlemcinin yükselen sinyal kenarı için sorgulanması. LAD: Atanan IN Bit'inde bir yükselen kenar (AÇIK'tan sonra KAPALI) algılanırsa bu kontağın durumu DOĞRU olur. Daha sonra kontağın mantıksal durumu, sinyal akışının çıkış durumunu ayarlamak için sinyal akışının giriş durumuna bağlanır. P kontağı, ağdaki herhangi bir konuma yerleştirilebilir: Bir dalın sonunda. FBD: Atanan giriş Bit'inde bir yükselen kenar (AÇIK'tan sonra KAPALI) algılanırsa bu çıkışın mantıksal durumu DOĞRU olur. P kutusu, sadece bir dalın başlangıcına yerleştirilmelidir.
		Mevcut değil [1]	İşlemcinin negatif sinyal kenarı için sorgulanması. LAD: Atanan giriş Bit'inde bir alçalan kenar (KAPALI'dan sonra AÇIK) algılanırsa bu kontağın durumu DOĞRU olur. Daha sonra kontağın mantıksal durumu, sinyal akışının çıkış durumunu ayarlamak için sinyal akışının giriş durumuna bağlanır. N kontağı, ağdaki herhangi bir konuma yerleştirilebilir: Bir dalın sonunda. FBD: Atanan giriş Bit'inde bir alçalan kenar (KAPALI'dan sonra AÇIK) algılanırsa çıkışın mantıksal durumu DOĞRU olur. N kutusu, sadece bir dalın başlangıcına yerleştirilmelidir.
		Mevcut değil [1]	İşlemcinin yükselen sinyal kenarında ayarlanması. LAD: Bobine giren sinyal akışında yükselen bir kenar (AÇIK'tan sonra KAPALI) algılanırsa, atanan ÇIKIŞ Bit'i DOĞRU olur. Sinyal akışının giriş durumu bobinden her zaman sinyal akışının çıkış durumu olarak geçer. P bobini, ağdaki herhangi bir konumda eklenebilir. FBD: Kutu bir dalın başlangıcında bulunduğu sürece, kutunun giriş bağlantısının mantıksal durumunda veya atanan giriş Bit'inde bir yükselen kenar (AÇIK'tan sonra KAPALI) algılanırsa, atanan OUT Bit'i DOĞRU olur. Girişin mantıksal durumu, kutudan her zaman çıkışın mantıksal durumu olarak geçer. P= kutusu, daldaki herhangi bir konumda eklenebilir.
		Mevcut değil [1]	İşlemcinin negatif sinyal kenarında ayarlanması. LAD: Bobine giren sinyal akışında alçalan bir kenar (KAPALI'dan sonra AÇIK) algılanırsa, atanan ÇIKIŞ Bit'i DOĞRU olur. Sinyal akışının giriş durumu bobinden her zaman sinyal akışının çıkış durumu olarak geçer. N bobini, ağdaki herhangi bir konumda eklenebilir. FBD: Kutu bir dalın başlangıcında bulunduğu sürece, kutunun giriş bağlantısının mantıksal durumunda veya atanan giriş Bit'inde bir alçalan kenar (KAPALI'dan sonra AÇIK) algılanırsa, atanan OUT Bit'i DOĞRU olur. Girişin mantıksal durumu, kutudan her zaman çıkışın mantıksal durumu olarak geçer. N= kutusu, daldaki herhangi bir konumda eklenebilir.

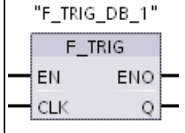
[1] SCL'de: Bu fonksiyonu uygulamanızın içinde çoğaltmak için kod yazmanız gerekir.

Tablo 8-15 P_TRIG ve N_TRIG

LAD/FBD	SCL	Açıklama
	Mevcut değil [1]	VKE'nin (bağlantı sonucu) yükselen sinyal kenarı için sorgulanması. CLK giriş durumunda (FBD) veya CLK sinyal akışında (LAD) bir yükselen kenar (AÇIK'tan sonra KAPALI) algılanırsa, sinyal akışı veya Q çıkışının mantıksal durumu DOĞRU olur. LAD'de, P_TRIG yordamı bir ağın başına veya sonuna yerleştirilmemelidir. FBD'de, P_TRIG yordamı herhangi bir konuma yerleştirilebilir, istisna: Bir dalın sonunda.
	Mevcut değil [1]	VKE"nin negatif sinyal kenarı için sorgulanması. CLK giriş durumunda (FBD) veya CLK sinyal akışında (LAD) bir alçalan kenar (KAPALI'dan sonra AÇIK) algılanırsa, sinyal akışı veya Q çıkışının mantıksal durumu DOĞRU olur. LAD'de, N_TRIG yordamı bir ağın başına veya sonuna yerleştirilmemelidir. FBD'de, N_TRIG yordamı herhangi bir konuma yerleştirilebilir, istisna: Bir dalın sonunda.

[1] SCL'de: Bu fonksiyonu uygulamanızın içinde çoğaltmak için kod yazmanız gerekir.

Tablo 8-16 R_TRIG ve F_TRIG yordamları

LAD/FBD	SCL	Açıklama
	**"R_TRIG_DB"(CLK:=_in_, Q=> _bool_out_);**	Değişkenin yükselen sinyal kenarında ayarlanması. CLK girişinin önceki durumu, atanan örnek DB'ye kaydedilir. CLK giriş durumunda (FBD) veya CLK sinyal akışında (LAD) bir yükselen kenar (AÇIK'tan sonra KAPALI) algılanırsa, sinyal akışı veya Q çıkışının mantıksal durumu DOĞRU olur. LAD'de, R_TRIG yordamı bir ağın başına veya sonuna yerleştirilmemelidir. FBD'de R_TRIG yordamı, bir dalın sonu istisna olmak kaydıyla herhangi bir konuma yerleştirilebilir:
	"F_TRIG_DB"(CLK:=_in_, Q=> _bool_out_);	Değişkenin negatif sinyal kenarında ayarlanması. CLK girişinin önceki durumu, atanan örnek DB'ye kaydedilir. CLK giriş durumunda (FBD) veya CLK sinyal akışında (LAD) bir alçalan kenar (KAPALI'dan sonra AÇIK) algılanırsa, sinyal akışı veya Q çıkışının mantıksal durumu DOĞRU olur. LAD'de, F_TRIG yordamı bir ağın başına veya sonuna yerleştirilmemelidir. FBD'de F_TRIG yordamı, bir dalın sonu istisna olmak kaydıyla herhangi bir konuma yerleştirilebilir:

Yordam programa eklendiği takdirde R_TRIG ve F_TRIG için otomatik olarak "Çağrı opsiyonları" iletişim kutusu açılır. Bu iletişim kutusunda,
kenar markörünün kendi veri bloğuna mı kaydedileceğini (münferit örnek) yoksa yerel değişken olarak (çoklu örnek) blok arayüzüne mi
kaydedileceğini belirleyebilirsiniz. Ayrı bir veri bloğu oluşturulursa, bu veri bloğunu, proje gezginindeki "Program kaynakları" klasöründe
"Program blokları > Sistem blokları" öğelerinin altında bulabilirsiniz.

Tablo 8-17 (Kontaklar/bobinler P veN, P=, N=, P_TRIG and N_TRIG) parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
M_BIT	Bool	Girişin önceki durumunun kaydedildiği markör
IN	Bool	Kenarı algılanan giriş Bit'i
OUT	Bool	Bir kenarın algılandığını belirten çıkış Bit'i
CLK	Bool	Kenarı algılanan sinyal akışı veya giriş Bit'i
Q	Bool	Bir kenarın algılandığını belirten çıkış

Tüm kenar yordamları, izlenen giriş sinyalinin önceki durumunu kaydetmek için bir markör kullanır (M_BIT: P/N kontaklar/bobinler, P_TRIG/N_TRIG) veya (Örnek DB Bit'i: R_TRIG, F_TRIG). Bir kenar, girişin durumu önceki durumla karşılaştırılarak algılanır. Girişteki durumlar istenen yönde bir sinyal değişimini gösteriyorsa, çıkış DOĞRU olarak ayarlanarak bir kenar bildirilir. Aksi takdirde çıkış YANLIŞ olarak ayarlanır.

NOT

Kenar işlemleri, ilk yürütme de dahil olmak üzere, her yürütme işleminde girişi ve markör değerlerini değerlendirir. İlk çevrimde kenar algılamaya izin vermek veya vermemek için programınızdaki girişin ve markörün çıkış durumlarını dikkate almanız gerekir.

Markörün bir yürütme işleminden bir sonraki yürütme işlemine kadar kaydedilmesi gerektiği için, her kenar işlemi için benzersiz bir Bit kullanmanız gerekir. Bu Bit'i programınızdaki başka konumlarda kullanmamanız gerekir. Ayrıca geçici bellekleri ve örn. G/Ç güncellemeleri gibi başka sistem fonksiyonları tarafından değiştirilebilen bellekleri kullanmayın. M_BIT bellek atamaları için sadece markörler (M), global DB'ler veya statik bellekler (bir örnek DB'de) kullanın.

8.2 Zamanların çalışma şekli

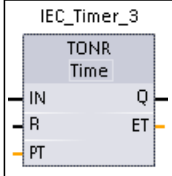
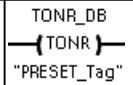
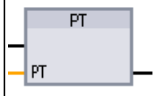
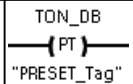
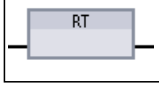
Zaman yordamları ile programlanmış zaman gecikmelerini ayarlayabilirsiniz. Kullanıcı programınızda kullanabileceğiniz zamanların sayısı, sadece CPU'daki bellek alanıyla sınırlıdır. Her zaman, kutu veya bobin yordamının üst kısmında belirtilen 16 Bayt büyüklüğünde, IEC_Timer veri tipinde bir DB yapısı kullanır. Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 8-18 Zaman işlemleri

LAD/FBD kutuları	LAD bobin-leri	SCL	Açıklama
IEC_Timer_0 TP Time IN Q PT ET	TP_DB TP "PRESET_Tag"	**"IEC_Timer_0_DB".TP(IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);**	TP zamanı, ön ayarlı bir süreyle bir darbe üretir.
IEC_Timer_1 TON Time IN Q PT ET	TON_DB TON "PRESET_Tag"	**"IEC_Timer_0_DB".TON (IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);**	TON zamanı, Q çıkışını ön ayarlı bir süre dolduktan sonra AÇIK olarak ayarlar.
IEC_Timer_2 TOF Time IN Q PT ET	TOF_DB TOF "PRESET_Tag"	**"IEC_Timer_0_DB".TOF (IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);**	TOF zamanı, Q çıkışını ön ayarlı bir süre dolduktan sonra sıfırlayarak KAPALI olarak ayarlar.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneklerinde "IEC_Timer_0_DB" örnek DB'nin adı yer alır.

LAD/FBD kutuları	LAD bobin-leri	SCL	Açıklama
		**"IEC_Timer_0_DB".TONR (IN:=_bool_in_, R:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);**	TONR zamanı, Q çıkışını ön ayarlı bir süre dolduktan sonra AÇIK olarak ayarlar. Geçen süre, R girişi geçen süreyi sıfırlamak için etkinleştirilene kadar birden fazla zaman aralığı boyunca toplanır.
Sadece FBD: 		**PRESET_TIMER(PT:=_time_in_, TIMER:=_iec_timer_in_);**	PT bobini (ön ayarlı süre), belirtilen IEC_Timer'a yeni bir PRESET zaman değeri yükler.
Sadece FBD: 		**RESET_TIMER(_iec_timer_in_);**	RT bobini (süreyi sıfırlama) belirtilen IEC_Timer'ı sıfırlar.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneklerinde "IEC_Timer_0_DB" örnek DB'nin adı yer alır.

Tablo 8-19 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
Kutu: IN Bobin: Sinyal akışı	Bool	TP, TON ve TONR: Kutu: 0 = Süreyi devre dışı bırak, 1 = Süreyi etkinleştir Bobin: Sinyal akışı yok = Süreyi devre dışı bırak, Sinyal akışı = Süreyi etkinleştir TOF: Kutu: 0 = Süreyi etkinleştir, 1 = Süreyi devre dışı bırak Bobin: Sinyal akışı yok = Süreyi etkinleştir, Sinyal akışı = Süreyi devre dışı bırak
R	Bool	Sadece TONR kutusu: 0 = Sıfırlama 1= Dolan süreyi ve Q Bit'ini sıfırlayarak 0 değerine ayarla
Kutu: PT Bobin: "PRESET_Tag"	Time	Zaman kutusu veya bobini: Giriş ön ayarlı süresi
Kutu: Q Bobin: DBdata.Q	Bool	Zaman kutusu: Zamanın DB verilerindeki Q kutu çıkışı veya Q Bit'i Zaman bobini: Sadece zamanın DB verilerindeki Q Bit'ini adresleyebilirsiniz
Kutu: ET Bobin: DBdata.ET	Time	Zaman kutusu: Zamanın DB verilerindeki ET kutu çıkışı (dolan süre) veya ET zaman değeri Zaman bobini: Sadece zamanın DB verilerindeki ET zaman değerini adresleye-bilirsiniz

Tablo 8-20 PT ve IN parametrelerindeki değer değişikliklerinin etkisi

Zaman değiştiri-ci	PT ve IN kutu parametrelerindeki ve ilgili bobin parametrelerindeki değişiklikler
TP	• Süre devam ederken PT değişirse, bunun hiçbir etkisi olmaz. • Süre devam ederken IN değişirse, bunun hiçbir etkisi olmaz.
TON	• Süre devam ederken PT değişirse, bunun hiçbir etkisi olmaz. • Süre devam ederken IN değeri YANLIŞ olarak değişirse, süre durdurulur ve sıfırlanır.
TOF	• Süre devam ederken PT değişirse, bunun hiçbir etkisi olmaz. • Süre devam ederken IN değeri DOĞRU olarak değişirse, süre durdurulur ve sıfırlanır.

Zaman değiştirici	PT ve IN kutu parametrelerindeki ve ilgili bobin parametrelerindeki değişiklikler
TONR	• Süre devam ederken PT değişirse, bunun hiçbir etkisi olmaz. Süre devam ettirilirse birtakım etkileri söz konusudur. • Süre devam ederken IN değeri YANLIŞ olarak değişirse, süre durdurulur, ancak sıfırlanmaz. IN değeri tekrar DOĞRU olarak değişirse, süre, toplanan zaman değerinden itibaren ilerlemeye devam eder.

PT (ön ayarlı süre) ve ET (geçen süre) değerleri, belirtilen IEC_TIMER DB'nin verilerinde, mili saniye cinsinden bir süre değerini gösteren işaretli çift tamsayılar olarak kaydedilir. TIME veri tipi, T# kodunu kullanır ve basit zaman birimi (T#200ms veya 200) veya T#2s_200ms gibi birleşik zaman birimleri olarak girilebilir.

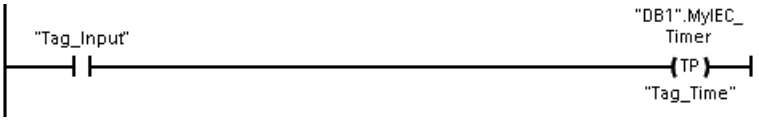
Tablo 8-21 TIME veri tipinin boyutu ve aralığı

Veri tipi	Boyut	Geçerli sayı aralıkları[1]
TIME	32 Bit, DInt verileri olarak kaydedilir	T#-24d_20h_31m_23s_648ms ila T#24d_20h_31m_23s_647ms Şu şekilde kaydedilir: -2.147.483.648 msn ila +2.147.483.647 msn

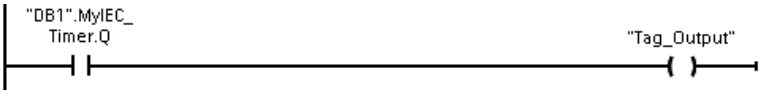
[1] Yukarıda gösterilen TIME veri tipinin negatif aralığı süreler için kullanılamaz. PT (ön ayarlı süre) için tanımlanan negatif değerler, zaman işlemi yürütülürken sıfıra ayarlanır. ET (geçen süre) her zaman pozitif bir değerdir.

Bir zaman bobini örneği

Zaman bobinleri -(TP)-, -(TON)-, -(TOF)- ve -(TONR)- bir LAD ağında son yordam olarak yerleştirilmelidir. Bir zaman örneğinde gösterildiği gibi, takip eden bir ağdaki bir kontak yordamı, Q Bit'ini, bir zaman bobinine ait IEC_Timer DB'nin verilerinde değerlendirir. Aynı şekilde, programınızda geçen sürenin değerini kullanmak istiyorsanız, IEC_Timer DB'nin verilerindeki ELAPSED elemanını adreslemeniz gerekir.



Darbe zamanlayıcısı, Tag_Input'in Bit değeri 0 iken 1 olarak değiştiğinde başlatılır. Süre, Tag_Time zaman değeri tarafından belirtilen süre boyunca ilerler.



Süre ilerlediği sürece DB1.MyIEC_Timer.Q'nun durumu = 1 ve Tag_Output'un durumu = 1 olur. Tag_Time'ın süre değeri dolduysa, DB1.MyIEC_Timer.Q = 0 ve Tag_Output = 0 olur.

Bobin zamanının sıfırlanması -(RT)- ve sürenin yüklenmesi -(PT)-

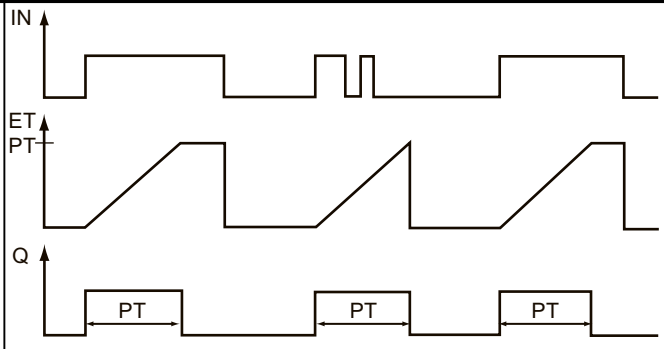
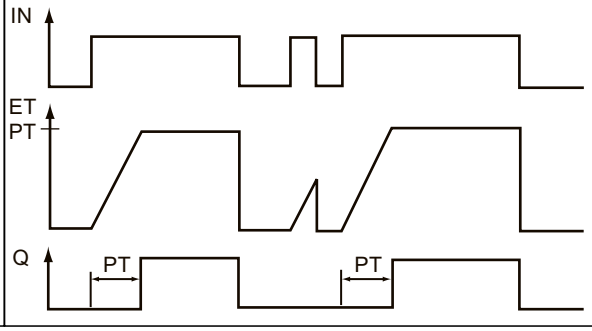
Bu bobin yordamları kutu ve bobin zamanlarıyla birlikte kullanılabilir ve bir akım yolunun ortasına yerleştirilebilir. Bobin çıkışının sinyal durumu, her zaman sinyal girişindeki sinyal durumuyla aynıdır. Bobin -(RT)- etkinleştirilirse, belirtilen IEC_Timer DB'nin verilerindeki ELAPSED zaman elemanı sıfırlanarak 0 değerine ayarlanır. Bobin -(PT)- etkinleştirilirse, belirtilen IEC_Timer DB'nin verilerindeki PRESET zaman elemanına, atanmış süre değeri yüklenir.

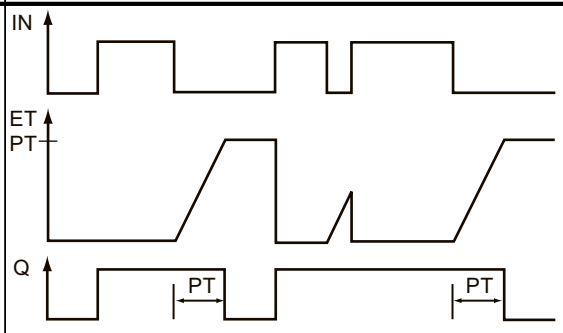
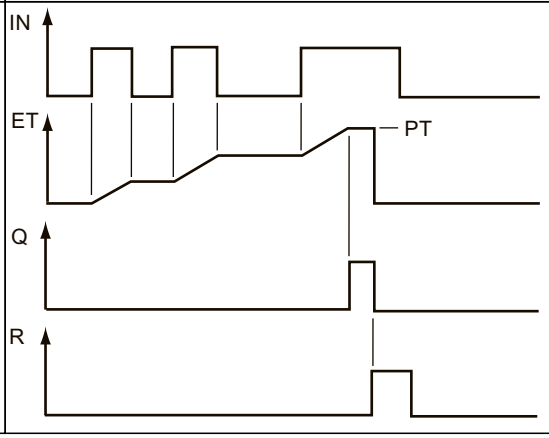
NOT

Zamanları bir FB'ye yerleştirirseniz, "Çoklu örnek veri bloğu" opsiyonunu seçebilirsiniz. Zaman yapılarının adları, çeşitli veri yapılarında farklı olabilir, ancak zaman verileri münferit bir veri bloğunda bulunur ve her zaman için özel bir veri bloğu gerekli değildir. Bu sayede, işleme süresi kısalır ve zamanların yönetimi için gereken veri belleği küçülür. Ortak kullanılan çoklu örnek DB'de, zamanların veri yapıları arasında etkileşim yoktur.

Zamanların çalışma şekli

Tablo 8-22 IEC zamanlarının tipleri

Zaman	Zaman diyagramı
TP: Darbe oluşturma TP zamanı, ön ayarlı bir süreyle bir darbe oluşturur.	
TON: Açma gecikmesi oluşturma TON zamanı, Q çıkışını ön ayarlı bir süre dolduktan sonra AÇIK olarak ayarlar.	

Zaman	Zaman diyagramı
TOF: Kapatma gecikmesi oluşturma TON zamanı, Q çıkışını ön ayarlı bir süre dolduktan sonra sıfırlayarak KAPALI olarak ayarlar.	
TONR: Zaman akümülatörü TONR zamanı, Q çıkışını ön ayarlı bir süre dolduktan sonra AÇIK olarak ayarlar. Geçen süre, R girişi geçen süreyi sıfırlamak için etkinleştirilene kadar birden fazla zaman aralığı boyunca toplanır.	

NOT

CPU'da hiçbir özel zamana belirli bir kaynak atanmamıştır. Bunun yerine zaman, zaman kumandasını gerçekleştirmek için DB belleğindeki kendi zaman yapısını ve sürekli olarak çalışan bir dahili CPU süresini kullanır.

Bir TP, TON, TOF veya TONR yordamının girişindeki bir kenar değişikliği nedeniyle bir zaman başlatıldığında, sürekli çalışan dahili CPU süresinin değeri, bu zamana atanan DB yapısının START elemanına kopyalanır. Bu başlangıç değeri, süre devam ettiği sürece değişmeden kalır ve daha sonra sürenin her güncellenişinde kullanılır. Sürenin her başlatılışında, dahili CPU süresinden yeni bir başlangıç değeri zaman yapısına yüklenir.

Bir süre güncellenirse, geçen süreyi belirlemek için, yukarıda açıklanan başlangıç değeri dahili CPU süresinin güncel değerinden çıkarılır. Geçen süre, Q zaman Bit'inin durumunu belirlemek için daha sonra ön ayarla karşılaştırılır. Daha sonra ELAPSED ve Q elemanları, bu zamanın atandığı DB yapısında güncellenir. Geçen sürenin, ön ayarlı değere bağlı olduğuna dikkat edin (Zaman, ön ayara ulaşıldığı taktirde, geçen süreyi biriktirmeye devam etmez).

Zaman, ancak aşağıdaki koşullar sağlandığında güncellenir:

- Bir zaman yordamı (TP, TON, TOF veya TONR) yürütüldüğünde
- DB'deki zaman yapısının "ELAPSED" elemanı doğrudan bir yordam tarafından refeanslanırsa
- DB'deki zaman yapısının "Q" elemanı doğrudan bir yordam tarafından refeanslanırsa

Zamanların programlanması

Kullanıcı programınız planlanırken ve oluşturulurken, zamanların aşağıdaki sonuçları göz önünde bulundurulmalıdır:

- Aynı çevrimde bir zamanın birden fazla güncellemesine sahip olabilirsiniz. Zaman, zaman yordamının (TP, TON, TOF, TONR) her yürütülüşünde ve zaman yapısının ELAPSED veya Q elemanı yürütülen başka bir yordamın parametresi olarak kullanıldığında güncellenir. Bu, en yeni zaman verilerine ihtiyaç duyduğunuzda (pratik olarak zamanın doğrudan okunması) avantajlıdır. Ancak, bir program çevrimi sırasında tutarlı değerler kullanmak istiyorsanız, zaman yordamınızı bu değerlere ihtiyaç duyan diğer tüm yordamlardan önce düzenleyin ve DB'deki zaman yapısının ELAPSED ve Q elemanları yerine zaman yordamının Q ve ET çıkışlarının değişkenlerini kullanın.
- Zaman güncellemesi olmayan çevrimlere sahip olabilirsiniz. Zamanınızı bir fonksiyonda başlatmanız ve daha sonra bu fonksiyonu bir çevrim veya başka çevrimler için artık çağırmamanız mümkündür. Zaman yapısının ELAPSED veya Q elemanlarını referanslayan başka hiçbir yordam yürütülmezse, zaman artık güncellenmez. Yeni bir güncelleme, ancak zaman yordamı yeniden yürütülürse veya zaman yapısındaki ELAPSED veya Q elemanını parametre olarak kullanan başka bir yordam yürütülürse ortaya çıkar.
- Her ne kadar alışılagelmiş bir durum olmasa da, aynı DB zaman yapısını birden çok zaman yordamına atayabilirsiniz. Genelde, beklenmeyen etkileşimlerin ortaya çıkmasını engellemek için, DB zaman yapısı başına sadece bir zaman (TP, TON, TOF, TONR) kullanmanız gerekir.
- Kendiliğinden sıfırlanan zamanlar, düzenli olarak ortaya çıkması gereken tetik eylemlerinde faydalıdır. Tipik olarak, kendiliğinden sıfırlanan zamanlar, zaman yordamından önce zaman Bit'ini referanslayan bir NC kontak yerleştirilerek oluşturulur. Bu zaman ağı, tipik olarak zaman Bit'iyle çeşitli eylemleri tetikleyen bir veya daha fazla bağımlı ağın üst kısmında yer alır. Zaman dolarsa (geçen süre ön ayarlı değere ulaşır), zaman Bit'i, zaman Bit'i tarafından kumanda edilen bağımlı ağların lojiği yürütülebilecek şekilde bir çevrim boyunca AÇIK durumda olur. Zaman ağının bir sonraki yürütülüşünde NC kontak KAPALI durumdadır, böylece zaman sıfırlanır ve zaman Bit'i silinir. Bir sonraki çevrimde NC kontak AÇIK durumdadır, bu nedenle zaman yeniden başlatılır. Bunun gibi kendiliğinden sıfırlanan zamanlar oluşturulurken, DB'deki zaman yapısının "Q" elemanını zaman yordamından önce NC kontak için parametre olarak kullanmayın. Burada, bunun için zaman yordamının "Q" çıkışına bağlı olan değişkeni kullanın. DB zaman yapısının Q elemanına erişimden kaçınılmasının nedeni, bunun zamanın güncellenmesine neden olmasıdır. Ve zaman NC kontak aracılığıyla güncellenirse, kontak, zaman yordamını derhal sıfırlar. Zaman yordamının Q çıkışı, bu bir çevrim sırasında AÇIK durumda değildir ve bağımlı ağlar yürütülmez.

Bir RUN-STOP-RUN değişiminden veya CPU'nun yeniden başlatılmasından sonra zaman verilerinin kaydedilmesi

RUN işletim durumundaki bir oturum STOP işletim durumuyla veya bir yeniden başlatma işlemiyle sonlandırılırsa ve RUN işletim durumunda yeni bir oturum başlatılırsa, zaman veri yapısı kalıcı olarak tanımlanmadığı sürece (TP, TON, TOF ve TONR zamanları) önceki RUN oturumunda kaydedilen zaman verileri kaybolur.

Programlama editöründe bir zaman yordamını eklerken çağrı opsiyonları iletişim kutusundaki standart ayarı kabul ederseniz, otomatik olarak, **kalıcı olarak tanımlanamayan** bir örnek DB atanır. Zaman verilerinizi kalıcı olarak tanımlamanız için, global bir DB veya bir çoklu örnek DB kullanmanız gerekir.

Zaman verilerinin global bir DB'nin kalıcı veriler olarak atanması yoluyla kaydedilmesi

Bu opsiyon, zamanın nereye yerleştirileceğinden (OB, FC veya FB) bağımsız olarak çalışır.

1. Global DB oluşturma:
   - Proje gezgininde "Yeni blok ekle" öğesine çift tıklayın.
   - Veri bloğu (DB) için kullanılan sembole tıklayın.
   - Tip olarak global DB'yi seçin.
   - DB'deki münferit veri elemanlarını kalıcı olarak tanımlamak istiyorsanız, DB tipi için "Optimize edildi" opsiyon kutusunu etkinleştirmeniz gerekir. "Standart - S7-300/400 ile uyumlu" DB tipi için diğer opsiyon, sadece, DB'nin tüm veri elemanlarının kalıcı olarak tanımlanmasına izin verir veya hiçbir veri elemanının kalıcı olarak tanımlanmasına izin vermez.
   - "TAMAM" butonuna tıklayın.
2. DB'ye zaman yapısı (zaman yapıları) ekleme:
   - Yeni global DB'de IEC_Timer veri tipinde yeni bir statik değişken ekleyin.
   - Bu yapının kalıcı olması için "Kalıcı" sütunundaki opsiyon kutusunu etkinleştirin.
   - Bu işlemi, bu DB'ye kaydetmek istediğiniz tüm zamanlar için yapılar oluşturmak üzere tekrarlayın. Her zaman yapısını benzersiz bir global DB'ye yerleştirebilir veya aynı global DB'ye birden fazla zaman yapısı yerleştirebilirsiniz. Bu global DB'ye zamanlar dışında başka statik değişkenler de yerleştirebilirsiniz. Aynı global DB'ye birden fazla zaman yapısı yerleştirme yoluyla blokların toplam sayısını azaltabilirsiniz.
   - Gerekirse zaman yapılarını yeniden adlandırın.
3. Düzenlemek için, kalıcı bir zaman eklemek istediğiniz program bloğunu (OB, FC veya FB) açın.
4. Zaman yordamını istediğiniz konuma yerleştirin.
5. Çağrı opsiyonlarını içeren iletişim kutusu görüntülenirse "İptal" butonuna tıklayın.
6. Yeni zaman yordamında, üst kısımda global DB'nin ve yukarıda oluşturduğunuz zaman yapısının adını girin (bir ad seçmeyin). Örnek: "Data_block_3.Static_1".

Zaman verilerinin, bir çoklu örnek DB'nin kalıcı veriler olarak kaydedilmesi yoluyla atanması

Bu olanak sadece zamanı bir FB'ye yerleştirmeniz durumunda işe yarar.

Bu varyant, FB özelliklerinde "Optimize edilmiş blok erişiminin" ayarlanıp ayarlanmadığına bağlıdır (sadece sembolik adresleme mümkündür). Mevcut bir FB'nin erişim özniteliğinin nasıl yapılandırıldığını kontrol etmek için, farenin sağ tuşuyla proje gezgininde FB'ye tıklayın, "Özellikler" ve ardından "Öznitelik" öğesini seçin.

FB için "Optimize edilmiş blok erişimi" ayarlandıysa (sadece sembolik adresleme mümkündür):

1. Düzenlemek için FB'yi açın.
2. Zaman yordamını FB'de istediğiniz konuma yerleştirin.
3. Çağrı opsiyonlarını içeren iletişim kutusu görüntülenirse çoklu örnek sembolüne tıklayın. Çoklu örnek opsiyonu, sadece yordamın bir FB'ye eklenmesi durumunda kullanılabilir.
4. Gerekirse çağrı opsiyonları iletişim kutusunda zamana bir ad verin.
5. "TAMAM" butonuna tıklayın. Zaman yordamı editörde görüntülenir ve FB arayüzündeki "Statik" ifadesinin altında IEC_TIMER yapısı görüntülenir.
6. Gerekirse, FB arayüz editörünü açın (görünümü büyütmek için küçük oka tıklamanız gerekebilir).

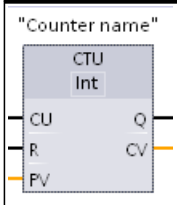
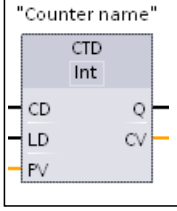
7. "Statik" ifadesinin altında, sizin için oluşturulan zaman yapısını arayın.
8. Bu zaman yapısının "Kalıcı" sütununda, seçimi "Kalıcı" olarak değiştirin. Bu FB başka bir program bloğunda çağrıldığında, zaman yapısının kalıcı olarak tanımlandığı bu arayüz tanımıyla birlikte her zaman bir örnek DB oluşturulur.

FB için "Optimize edilmiş blok erişimi" ayarlanmadıysa, klasik S7-300/400 konfigürasyonlarıyla uyumlu olan ve sembolik olarak ve doğrudan erişime izin veren standart blok erişimi kullanılır. Standart blok erişimine sahip bir FB'ye bir çoklu örnek atamak için aşağıdaki yöntemi izleyin:

1. Düzenlemek için FB'yi açın.
2. Zaman yordamını FB'de istediğiniz konuma yerleştirin.
3. Çağrı opsiyonlarını içeren iletişim kutusu görüntülenirse çoklu örnek sembolüne tıklayın. Çoklu örnek opsiyonu, sadece yordamın bir FB'ye eklenmesi durumunda kullanılabilir.
4. Gerekirse çağrı opsiyonları iletişim kutusunda zamana bir ad verin.
5. "TAMAM" butonuna tıklayın. Zaman yordamı editörde görüntülenir ve FB arayüzündeki "Statik" ifadesinin altında IEC_TIMER yapısı görüntülenir.
6. Bu FB'yi kullanması gereken bloğu açın.
7. Bu FB'yi istediğiniz konuma yerleştirin. Bu sayede, bu FB için bir örnek veri bloğu oluşturulur.
8. FB'yi editöre yerleştirirken oluşturulan örnek veri bloğunu açın.
9. "Statik" ifadesinin altında istediğiniz zaman yapısını arayın. Bu yapıyı kalıcı olarak tanımlamak için, bu zaman yapısının "Kalıcı" sütunundaki opsiyon kutusunu etkinleştirin.

8.3 Sayaçların çalışma şekli

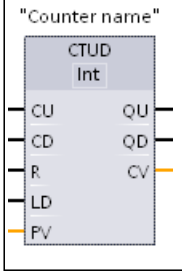
Tablo 8-23 Sayaç yordamları

LAD/FBD	SCL	Açıklama
	**"IEC_Counter_0_DB".CTU(CU:=_bool_in, R:=_bool_in, PV:=_in, Q=>_bool_out, CV=>_out);**	Sayaç yordamları ile program dahilindeki olayları ve harici proses olaylarını sayabilirsiniz. Her sayaç, sayacın verilerini kaydetmek için bir veri bloğuna yerleştirilen bir yapıyı kullanır. Sayaç işlemini editörde eklemek için veri bloğunu atayabilirsiniz. • CTU ileriye doğru sayar. • CTD geriye doğru sayar. • CTUD ileriye ve geriye doğru sayar.
	**"IEC_Counter_0_DB".CTD(CD:=_bool_in, LD:=_bool_in, PV:=_in, Q=>_bool_out, CV=>_out);**	

[1] LAD ve FBD'de: Yordam adının altındaki açılır listeden sayma değeri için kullanılan veri tipini seçin.

[2] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[3] SCL örneklerinde "IEC_Counter_0_DB", örnek DB'nin adıdır.

LAD/FBD	SCL	Açıklama
	**"IEC_Counter_0_DB".CTUD(CU:=_bool_in, CD:=_bool_in, R:=_bool_in, LD:=_bool_in, PV:=_in_, QU=>_bool_out, QD=>_bool_out, CV=>_out_);**	Sayaç yordamları ile program dahilindeki olayları ve harici proses olaylarını sayabilirsiniz. Her sayaç, sayacın verilerini kaydetmek için bir veri bloğuna yerleştirilen bir yapıyı kullanır. Sayaç işlemini editörde eklemek için veri bloğunu atayabilirsiniz. • CTU ileriye doğru sayar. • CTD geriye doğru sayar. • CTUD ileriye ve geriye doğru sayar.

[1] LAD ve FBD'de: Yordam adının altındaki açılır listeden sayma değeri için kullanılan veri tipini seçin.

[2] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[3] SCL örneklerinde "IEC_Counter_0_DB", örnek DB'nin adıdır.

Tablo 8-24 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
CU, CD	Bool	Her seferinde yukarıya veya aşağıya doğru bir birim sayma
R (CTU, CTUD)	Bool	Sayma değerini sıfırlama
LD (CTD, CTUD)	Bool	Ön ayarlı değer için kullanılan şarj kumandası
PV	SInt, Int, DInt, USInt, UInt, UDInt	Ön ayarlı sayma değeri
Q, QU	Bool	CV >= PV ise doğru
QD	Bool	CV <= 0 ise doğru
CV	SInt, Int, DInt, USInt, UInt, UDInt	Güncel sayma değeri

[1] Sayma değerlerinin sayısal aralığı, seçilen veri tipine bağlıdır. Sayma değeri, ön işareti olmayan bir tamsayı değeri ise, sıfıra kadar aşağıya doğru veya aralık sınırına kadar yukarıya doğru sayabilirsiniz. Sayma değeri ön işaretli bir tamsayı değeri ise, alt limite kadar aşağıya doğru ve üst limite kadar yukarıya doğru sayabilirsiniz.

CPU'nun bellek kapasitesi, kullanıcı programında izin verilen sayaçların sayısını sınırlar. Sayaçlar, aşağıdaki bellek alanına ihtiyaç duyar:

- Sayaç yordamı, SInt veya USInt veri tiplerinde 3 Bayt'a ihtiyaç duyar.
- Sayaç yordamı, nt veya UInt veri tiplerinde 6 Bayt'a ihtiyaç duyar.
- Sayaç yordamı, DInt veya UDInt veri tiplerinde 12 Bayt'a ihtiyaç duyar.

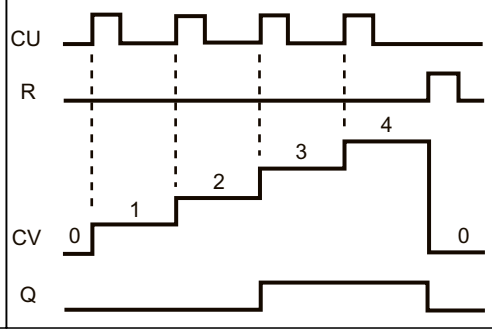
Bu yordamlar, maksimum sayma hızı eklendikleri OB'nin yürütme hızıyla sınırlı olan yazılım tabanlı sayaçlar kullanır. Yordamların ekleneceği OB, CU veya CD girişlerinin tüm geçişlerini algılamak için yeterince sık bir şekilde yürütülmelidir. Daha hızlı saymak için CTRL_HSC yordamını (Sayfa 509) kullanın.

NOT

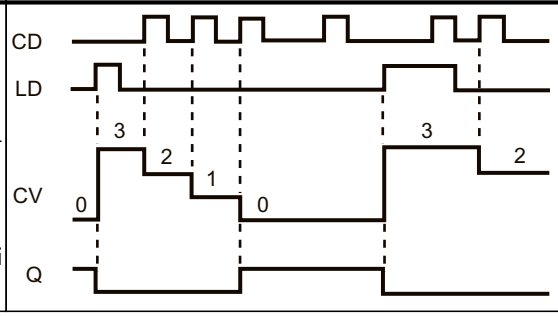
Sayaç yordamlarını bir FB'ye yerleştirirseniz, "Çoklu örnek DB" opsiyonunu seçebilirsiniz. Zaman yapılarının adları, farklı veri yapılarında farklı olabilir, ancak sayaç verileri tek bir DB'de bulunur ve her sayaç için kendine ait bir DB olması gerekmez. Bu sayede, işleme süresi ve sayaç için gereken veri belleği azalır. Ortak kullanılan çoklu örnek DB'de, sayaçların veri yapıları arasında etkileşim yoktur.

Sayaçların çalışma şekli

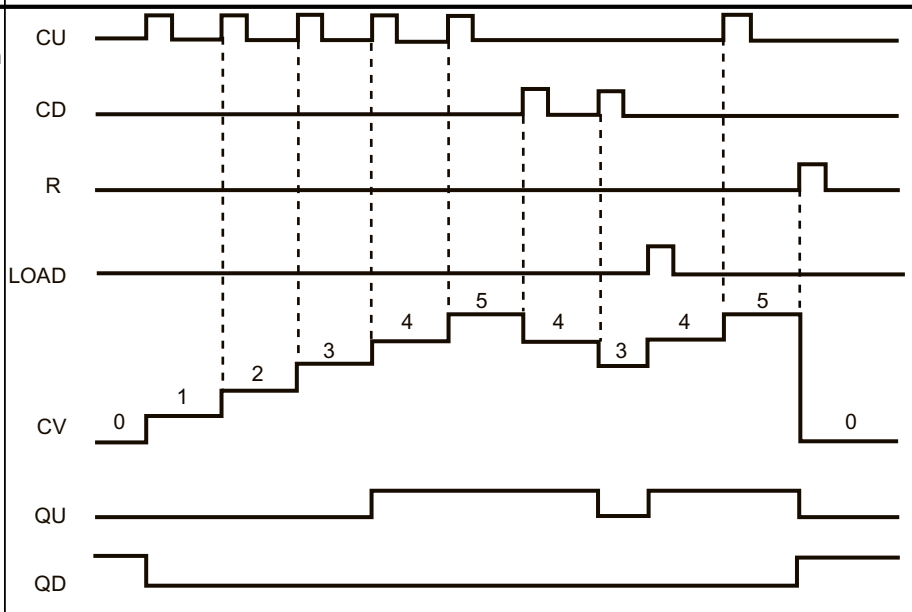
Tablo 8-25 CTU'nun çalışma şekli (ileriye doğru sayım)

Sayaç	Kullanım
CTU sayacı, CU parametrenin değeri 0 iken 1 olarak değişirse, ileriye doğru 1 sayar. CTU zaman diyagramı, ön işareti olmayan bir tamsayı sayma değerinde çalışma şeklini gösterir (burada PV = 3'tür). • CV parametresinin değeri (güncel sayma değeri) PV parametresinin değerinden (ön ayarlı sayma değeri) daha büyükse veya ona eşitse, Q sayaç çıkışı için kullanılan parametre = 1 olur. • R sıfırlama parametresinin değeri 0 iken 1 olarak değişirse, güncel sayma değeri sıfırlanarak 0 yapılır.	

Tablo 8-26 CTD'nin çalışma şekli (geriye doğru sayma)

Sayaç	Kullanım
CTD sayacı, CD parametrenin değeri 0 iken 1 olarak değişirse, geriye doğru 1 sayar. CTD zaman diyagramı, ön işareti olmayan bir tamsayı sayma değerinde çalışma şeklini gösterir (burada PV = 3'tür). • CV parametresinin değeri (güncel sayma değeri) 0'dan küçükse veya ona eşitse, Q sayaç çıkışı için kullanılan parametre = 1 olur. • LOAD parametresinin değeri 0 iken 1 olarak değişirse, PV parametresindeki değer (ön ayarlı değer) yeni CV parametresi olarak (güncel sayma değeri) sayaca yüklenir.	

Tablo 8-27 CTUD'nin çalışma şekli (ileriye ve geriye doğru sayım)

Sayaç	Kullanım
CTUD sayacı, ileriye doğru sayım girişi (CU) veya geriye doğru sayım girişi (CD) 0 iken 1 olarak değişirse ileriye veya geriye doğru 1 sayar. CTUD zaman diyagramı, ön işareti olmayan tamsayı sayma değerindeki (ve PV = 4 iken) çalışma şeklini gösterir. • CV parametresinin değeri PV parametresinin değerinden daha büyükse veya ona eşitse, QU sayaç çıkışı için kullanılan parametre = 1 olur. • CV parametresinin değeri 0'dan küçükse veya ona eşitse, QD sayaç çıkışı için kullanılan parametre = 1 olur. • LOAD parametresinin değeri 0 iken 1 olarak değişirse, PV parametresindeki değer yeni CV parametresi olarak sayaca yüklenir. • R sıfırlama parametresinin değeri 0 iken 1 olarak değişirse, güncel sayma değeri sıfırlanarak 0 yapılır.	

Bir RUN-STOP-RUN değişiminden veya CPU'nun yeniden başlatılmasından sonra sayaç verilerinin kaydedilmesi

RUN işletim durumundaki bir oturum STOP işletim durumu ile veya CPU'nun yeniden başlatılması ile sonlandırılırsa ve RUN işletim durumunda yeni bir oturum başlatılırsa, sayaç verileri yapısı kalıcı olarak tanımlanmadığı sürece (CTU, CTD ve CTUD zamanları), önceki RUN oturumunda kaydedilen sayaç verileri kaybolur.

Programlama editöründe bir sayaç yordamını eklerken çağrı opsiyonları iletişim kutusundaki standart ayarı kabul ederseniz, otomatik olarak, **kalıcı olarak tanımlanamayan** bir örnek DB atanır. Sayaç verilerinizi kalıcı olarak tanımlamanız için, global bir DB veya bir çoklu örnek DB kullanmanız gerekir.

Sayaç verilerinin global bir DB'in kalıcı veriler olarak atanması yoluyla kaydedilmesi

Bu opsiyon, sayacın nereye yerleştirileceğinden (OB, FC veya FB) bağımsız olarak çalışır.

1. Global DB oluşturma:
   - Proje gezgininde "Yeni blok ekle" öğesine çift tıklayın.
   - Veri bloğu (DB) için kullanılan sembole tıklayın.
   - Tip olarak global DB'yi seçin.
   - DB'deki münferit elemanları kalıcı olarak tanımlamak istiyorsanız, "Sadece sembolik olarak adreslenebilir" opsiyon kutusunu etkinleştirmeniz gerekir.
   - "TAMAM" butonuna tıklayın.
2. DB'ye sayaç yapısı (sayaç yapıları) ekleme:
   - Yeni global DB'de sayaç veri tipinde yeni bir statik değişken ekleyin. Bu sırada ön ayar ve sayaç değeri için kullanılan tipi göz önünde bulundurun.
   - Bu yapının kalıcı olması için "Kalıcı" sütunundaki opsiyon kutusunu etkinleştirin.
   - Bu işlemi, bu DB'ye kaydetmek istediğiniz tüm sayaçlar için yapılar oluşturmak üzere tekrarlayın. Her sayaç yapısını benzersiz bir global DB'ye yerleştirebilir veya aynı global DB'ye birden fazla sayaç yapısı yerleştirebilirsiniz. Bu global DB'ye sayaçlar dışında başka statik değişkenler de yerleştirebilirsiniz. Aynı global DB'ye birden fazla sayaç yapısı yerleştirme yoluyla blokların toplam sayısını azaltabilirsiniz.
   - Gerekirse sayaç yapılarını yeniden adlandırın.
3. Düzenlemek için, kalıcı bir sayaç eklemek istediğiniz program bloğunu (OB, FC veya FB) açın.
4. Sayaç yordamını istediğiniz konuma yerleştirin.
5. Çağrı opsiyonlarını içeren iletişim kutusu görüntülenirse "İptal" butonuna tıklayın. Şimdi, yeni sayaç yordamının, "???" yer tutucusuyla birlikte yordam adının üstünde ve altında görüntülenmesi gerekir.
6. Yeni sayaç yordamında, üst kısımda global DB'nin ve yukarıda oluşturduğunuz sayaç yapısının adını girin (bir ad seçmeyin). Örnek: "Data_block_3.Static_1". Bunun üzerine, ön ayar ve sayaç değeri için uygun tip önceden belirtilir (Örnek: Bir IEC_UCounter yapısı için kullanılan UInt).

Sayaç veri tipi	Ön ayar ve sayaç değeri için uygun tip
IEC_Counter	INT
IEC_SCounter	SINT
IEC_DCounter	DINT
IEC_UCounter	UINT
IEC_USCounter	USINT
IEC_UDCounter	UDINT

Sayaç verilerinin, bir çoklu örnek DB'nin kalıcı veriler olarak kaydedilmesi yoluyla atanması

Bu olanak sadece, sayacı bir FB'ye yerleştirmeniz durumunda işe yarar.

Bu varyant, FB özelliklerinde "Optimize edilmiş blok erişiminin" ayarlanıp ayarlanmadığına bağlıdır (sadece sembolik adresleme mümkündür). Mevcut bir FB'nin erişim özniteliğinin nasıl yapılandırıldığını kontrol etmek için, farenin sağ tuşuyla proje gezgininde FB'ye tıklayın, "Özellikler" ve ardından "Öznitelik" öğesini seçin.

FB için "Optimize edilmiş blok erişimi" ayarlandıysa (sadece sembolik adresleme mümkündür):

1. Düzenlemek için FB'yi açın.
2. Sayaç yordamını FB'de istediğiniz konuma yerleştirin.
3. Çağrı opsiyonlarını içeren iletişim kutusu görüntülenirse çoklu örnek sembolüne tıklayın. Çoklu örnek opsiyonu, sadece yordamın bir FB'ye eklenmesi durumunda kullanılabilir.
4. Gerekirse çağrı opsiyonları iletişim kutusunda sayaca bir ad verin.
5. "TAMAM" butonuna tıklayın. Sayaç yordamı, editörde ön ayar ve sayma değeri için kullanılan INT tipiyle ve IEC_COUNTER yapısı FB arayüzünde "Statik" ifadesinin altında görüntülenir.
6. Gerekirse sayaç yordamındaki INT olan tipi başka bir tiple değiştirin. Sayaç yapısı uygun şekilde değişir.
7. Gerekirse, FB arayüz editörünü açın (görünümü büyütmek için küçük oka tıklamanız gerekebilir).
8. "Statik" ifadesinin altında, sizin için oluşturulan sayaç yapısını arayın.
9. Bu sayaç yapısının "Kalıcı" sütununda, seçimi "Kalıcı" olarak değiştirin. Bu FB başka bir program bloğunda çağrıldığında, sayaç yapısının kalıcı olarak tanımlandığı bu arayüz tanımıyla birlikte her zaman bir örnek DB oluşturulur.

FB için "Optimize edilmiş blok erişimi" ayarlanmadıysa, klasik S7-300/400 konfigürasyonlarıyla uyumlu olan ve sembolik olarak ve doğrudan erişime izin veren standart blok erişimi kullanılır. Standart blok erişimine sahip bir FB'ye bir çoklu örnek atamak için aşağıdaki yöntemi izleyin:

1. Düzenlemek için FB'yi açın.
2. Sayaç yordamını FB'de istediğiniz konuma yerleştirin.
3. Çağrı opsiyonlarını içeren iletişim kutusu görüntülenirse çoklu örnek sembolüne tıklayın. Çoklu örnek opsiyonu, sadece yordamın bir FB'ye eklenmesi durumunda kullanılabilir.
4. Gerekirse çağrı opsiyonları iletişim kutusunda sayaca bir ad verin.
5. "TAMAM" butonuna tıklayın. Sayaç yordamı, editörde ön ayar ve sayma değeri için kullanılan INT tipiyle ve IEC_COUNTER yapısı FB arayüzünde "Statik" ifadesinin altında görüntülenir.
6. Gerekirse sayaç yordamındaki INT olan tipi başka bir tiple değiştirin. Sayaç yapısı uygun şekilde değişir.
7. Bu FB'yi kullanması gereken bloğu açın.
8. Bu FB'yi istediğiniz konuma yerleştirin. Bu sayede, bu FB için bir örnek veri bloğu oluşturulur.
9. FB'yi editöre yerleştirirken oluşturulan örnek veri bloğunu açın.
10. "Statik" ifadesinin altında istediğiniz sayaç yapısını arayın. Bu yapıyı kalıcı olarak tanımlamak için, bu sayaç yapısının "Kalıcı" sütunundaki opsiyon kutusunu etkinleştirin.

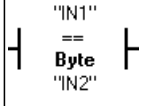
Sayaç yordamındaki tip (ön ayar ve sayma değeri için)	FB arayüzündeki uygun yapı tipi
INT	IEC_Counter
SINT	IEC_SCounter

DINT	IEC_DCounter
UINT	IEC_UCounter
USINT	IEC_USCounter
UDINT	IEC_UDCounter

8.4 Karşılaştırıcıların çalışma şekli

8.4.1 Karşılaştırma işlemleri

Tablo 8-28 Karşılaştırma işlemleri

LAD	FBD	SCL	Açıklama
		**out := in1 = in2;** veya **IF in1 = in2** **THEN out := 1;** **ELSE out := 0;** **END_IF;**	Aynı veri tipindeki iki değeri karşılaştırır. LAD kontak karşılaştırmasının sonucu DOĞRU ise kontak etkinleştirilir. FBD kutu karşılaştırmasının sonucu DOĞRU ise kutu çıkışı da yine DOĞRU olur.

[1] LAD ve FBD'de: Açılır liste aracılığıyla karşılaştırma tipini değiştirmek için yordam adına ("==" gibi) tıklayın. "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-29 Parametreler için kullanılan veri tipleri

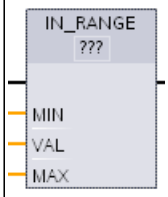
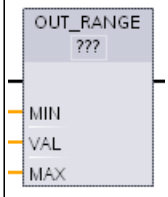
Parametre	Veri tipi	Açıklama
IN1, IN2	Byte, Word, DWord, SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, String, WString, Char, Char, Time, Date, TOD, DTL, sabiti	Karşılaştırılacak değerler

Tablo 8-30 Karşılaştırmaların açıklamaları

İlişki tipi	Karşılaştırma doğrudur, eğer ...
=	IN1, IN2'ye eşit ise
<>	IN1, IN2'ye eşit değil ise
>=	IN1, IN2'den büyük veya ona eşit ise
<=	IN1, IN2'den küçük veya ona eşit ise
>	IN1, IN2'den büyük ise
<	IN1, IN2'den küçük ise

8.4.2 IN_Range (değer, aralık dahilinde) ve OUT_Range (değer, aralık dışında)

Tablo 8-31 Değer, aralık dahilinde ve değer, aralık dışında yordamları

LAD/FBD	SCL	Açıklama
	**out := IN_RANGE(min, val, max);**	Girilecek değerin belirtilen bir değer aralığının içinde mi, yoksa dışında mı olduğunu kontrol eder. Karşılaştırma DOĞRU ise, kutu çıkışı DOĞRU olur.
	**out := OUT_RANGE(min, val, max);**	

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-32 Parametreler için kullanılan veri tipleri

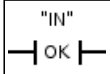
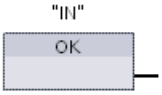
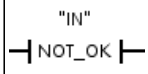
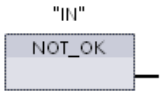
Parametre	Veri tipi[1]	Açıklama
MIN, VAL, MAX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, sabiti	Karşılaştırıcı girişleri

[1] MIN, VAL ve MAX giriş parametreleri aynı veri tipine sahip olmalıdır.

- IN_RANGE karşılaştırması doğrudur, eğer MIN <= VAL <= MAX ise
- OUT_RANGE karşılaştırması doğrudur, eğer VAL < MIN veya VAL > MAX ise

8.4.3 OK (geçerliliği kontrol et) ve NOT_OK (geçersizliği kontrol et)

Tablo 8-33 OK (geçerliliği kontrol et) ve NOT OK (geçersizliği kontrol et) yordamları

LAD	FBD	SCL	Açıklama
		Mevcut değil	Bir giriş verileri referansının IEEE Spesifikasyonu 754'e göre geçerli bir gerçek sayı olup olmadığını kontrol eder.
		Mevcut değil	

[1] LAD ve FBD'de: LAD kontağı DOĞRU ise, kontak etkinleştirilir ve sinyal akışını iletir. FBD kutusu DOĞRU ise, kutu çıkışı DOĞRU olur.

Tablo 8-34 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	Real, LReal	Giriş verileri

Tablo 8-35 Çalışma şekli

Yordam	Bir gerçek sayı açısından gerçekleştirilen kontrol, aşağıdaki DOĞRUDUR, eğer:
OK	Girilecek değer geçerli bir gerçek sayı[1] ise.
NOT_OK	Girilecek değer geçerli bir gerçek sayı[1] değil ise.

[1] Bir Real veya LReal değeri, bu değer +/- INF (sonsuz), NaN (Not a Number, sayı değil) veya standartlaştırılmamış bir değer ise geçersizdir. Standartlaştırılmamış bir değer, sıfıra çok yakın bir sayıdır. CPU, hesaplamalarda, standartlaştırılmamış bir değeri sıfırla değiştirir.

8.4.4 Varyant ve Array karşılaştırma işlemleri

8.4.4.1 Eşittir ve Eşit değildir karşılaştırma işlemleri

S7-1200 CPU'su, bir Variant işlemcisi tarafından gösterilen bir değişkenin veri tipini sorgulamak ve bu veri tipinin, diğer işlemcinin veri tipiyle eşleşip (eşittir) veya eşleşmediğini (eşit değildir) karşılaştırmak için kullanılan yordamları sunar.

S7-1200 CPU'su ayrıca, bir Array elemanının veri tipinin, diğer işlemcinin veri tipiyle eşitliği veya eşitsizliği açısından sorgulanması için kullanılan yordamları sunar.

Bu işlemlerde, <Operand1>, <Operand2> ile karşılaştırılır. <Operand1> , Variant veri tipine sahip olmalıdır. <Operand2>, bir PLC veri tipinin temel bir veri tipi olabilir. LAD ve FBD'de, <Operand1>, yordamın üzerindeki işlemcidir. LAD'de, <Operand2>, yordamın altındaki işlemcidir.

Tüm yordamlar için bağlantı sonucu (VKE), eşittir veya eşit değildir karşılaştırması başarılı olduysa 1'dir (doğru), aksi takdirde 0'dır (yanlış).

Eşittir ve eşit değildir karşılaştırma yordamları aşağıda açıklanmaktadır:

- EQ_Type (veri tipini, bir değişkenin veri tipiyle EŞİTTİR işlemi açısından karşılaştır)
- NE_Type (veri tipini, bir değişkenin veri tipiyle EŞİT DEĞİLDİR işlemi açısından karşılaştır)
- EQ_ElemType (Bir ARRAY elemanının veri tipini, bir değişkenin veri tipiyle EŞİTLİK açısından karşılaştır)
- NE_ElemType (Bir ARRAY elemanının veri tipini, bir değişkenin veri tipiyle EŞİT DEĞİLDİR işlemi açısından karşılaştır)

Tablo 8-36 EQ ve NE yordamları

LAD	FBD	SCL	Açıklama
#Operand1 EQ_Type "Operand2"	#Operand1 EQ_Type "Operand2" — IN2 OUT	Mevcut değil	Operand1'de gösterilen değişkenin, Operand2'deki değişkenle aynı veri tipine sahip olup olmadığını kontrol eder.
#Operand1 NE_Type "Operand2"	#Operand1 NE_Type "Operand2" — IN2 OUT	Mevcut değil	Operand1'de gösterilen değişkenin, Operand2'deki değişkene göre farklı bir veri tipine sahip olup olmadığını kontrol eder.
#Operand1 EQ_ElemType "Operand2"	#Operand1 EQ_ElemType "Operand2" — IN2 OUT	Mevcut değil	Operand1'de gösterilen Array elemanının, Operand2'deki değişkenle aynı veri tipine sahip olup olmadığını kontrol eder.
#Operand1 NE_ElemType "Operand2"	#Operand1 NE_ElemType "Operand2" — IN2 OUT	Mevcut değil	Operand1'de gösterilen Array elemanının, Operand2'deki değişkene göre farklı bir veri tipine sahip olup olmadığını kontrol eder.

Tablo 8-37 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
Operand1	Variant	İlk işlemci
Operand2	Bit karakter dizileri, tamsayılar, kayar noktalı sayılar, zamanlar, tarih ve saat, karakter dizileri, ARRAY, PLC veri tipleri	İkinci işlemci

8.4.4.2 Sıfır karşılaştırma işlemleri

IS_NULL ve NOT_NULL yordamları, girişin bir nesneyi gösterip göstermediğini belirlemek için kullanılabilir.

<Operand>, her iki yordam için de Variant değerine sahip olmalıdır.

Tablo 8-38 IS_NULL (İşaretçi EŞİTTİR SIFIR sorgusu) ve NOT_NULL (İşaretçi EŞİT DEĞİLDİR SIFIR sorgusu)

LAD	FBD	SCL	Açıklama
#Operand IS_NULL	#Operand IS_NULL OUT	Mevcut değil	Operand'daki Variant elemanının gösterdiği değişkenin sıfır olduğunu ve bununla birlikte bir nesne olmadığını kontrol eder.
#Operand NOT_NULL	#Operand NOT_NULL OUT	Mevcut değil	Operand'daki Variant elemanının gösterdiği değişkenin sıfır olmadığını ve bununla birlikte bir nesne olduğunu kontrol eder.

Tablo 8-39 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
Operand	Variant	Sıfır veya sıfır değil kontrolü için kullanılan işlemci.

8.4.4.3 IS_ARRAY (ARRAY açısından kontrol et)

"ARRAY açısından kontrol et" yordamıyla, Variant elemanının, Array veri tipindeki bir değişkeni gösterip göstermediği sorgulanabilir.

<İşlemci>, Variant veri tipine sahip olmalıdır.

Yordamlar, işlemcinin bir Array olması durumunda 1 (doğru) değerini döndürür.

Tablo 8-40 IS_ARRAY (ARRAY açısından kontrol et)

LAD	FBD	SCL	Açıklama
#Operand IS_ARRAY	#Operand IS_ARRAY OUT	IS_ARRAY(_variant_in_)	Operand'daki Variant elemanının gösterdiği değişkenin bir Array olup olmadığını kontrol eder.

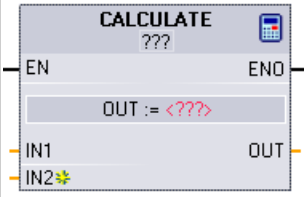
Tablo 8-41 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
Operand	Variant	Bir Array söz konusu olup olmadığını kontrol etmek için kullanılan işlemci.

8.5 Aritmetik fonksiyonlar

8.5.1 CALCULATE (hesapla)

Tablo 8-42 CALCULATE yordamı

LAD/FBD	SCL	Açıklama
	Denklemi oluşturmak için klasik matematiksel SCL ifadelerini kullanın.	CALCULATE yordamıyla, girişleri (IN1, IN2, .. INn) işleyen ve OUT'ta sonucu, önceden belirttiğiniz denkleme uygun şekilde yayınlayan bir matematiksel fonksiyon oluşturabilirsiniz. • Önce bir veri tipi seçin. Tüm girişler ve çıkış aynı veri tipine sahip olmalıdır. • Başka bir giriş eklemek için, son girişteki sembole tıklayın.

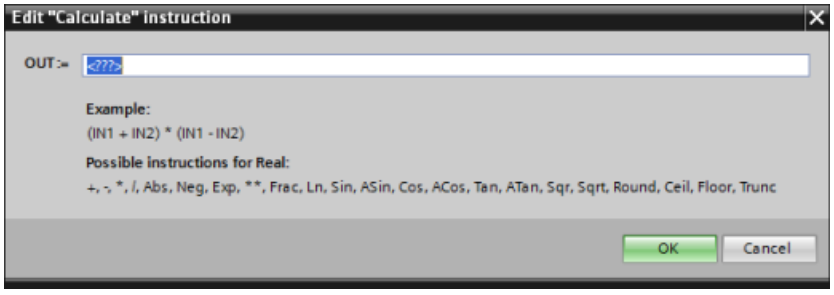
Tablo 8-43 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]
IN1, IN2, ..INn	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord

[1] IN ve OUT parametreleri aynı veri tipine sahip olmalıdır (giriş parametreleri dolaylı şekilde dönüştürülerek). Örnek: Bir giriş için kullanılan SINT tipindeki bir değer, OUT parametresinin INT veya REAL tipinde olması durumunda INT veya REAL tipinde bir değere dönüştürülür.

İletişim kutusunu çağırmak ve matematiksel fonksiyonu tanımlamak için hesap makinesi sembolüne tıklayın. Denkleminizi girişler (IN1 ve IN2 gibi) ve işlemler olarak girebilirsiniz. Fonksiyonu kaydetmek için "TAMAM" butonuna tıklarsanız, iletişim kutusu, otomatik olarak CALCULATE yordamı için kullanılan girişleri oluşturur.

Bu iletişim kutusunda, bir örnek ve OUT parametresinin veri tipini temel alarak kaydedebileceğiniz mümkün olan yordamların bir listesi gösterilmektedir:



NOT

Ayrıca fonksiyonunuzda sabitler için bir giriş oluşturmanız gerekir. Bu durumda sabit değer, CALCULATE yordamının atanmış girişine girilir.

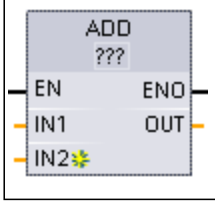
Sabitleri girişler olarak girerek, fonksiyonu değiştirmek zorunda kalmadan CALCULATE yordamını kullanıcı programınızda başka noktalara kopyalayabilirsiniz. Daha sonra, fonksiyonu değiştirmeksizin, yordam için kullanılan girişlerin değerlerini veya değişkenlerini değiştirebilirsiniz.

CALCULATE yordamı yürütülürse ve hesaplamadaki tüm münferit yordamlar başarılı bir şekilde uygulanırsa ENO = 1 olur. Aksi takdirde ENO = 0 olur.

CALCULATE yordamıyla ilgili bir örneği, "Basit bir yordamla karmaşık bir denklem oluşturmak" başlığında bulabilirsiniz.

8.5.2 Topla, Çıkar, Çarp ve Böl yordamları

Tablo 8-44 Topla, Çıkar, Çarp ve Böl yordamları

LAD/FBD	SCL	Açıklama
	**out := in1 + in2;** **out := in1 - in2;** **out := in1 * in2;** **out := in1 / in2;**	• ADD: Toplama (IN1 + IN2 = OUT) • SUB: Çıkarma (IN1 - IN2 = OUT) • MUL: Çarpma (IN1 * IN2 = OUT) • DIV: Bölme (IN1 / IN2 = OUT) Bir tamsayılı bölme işleminde bölümün ondalık basamakları, bir tamsayı çıkış değeri oluşacak şekilde kısaltılır.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-45 Parametreler için kullanılan veri tipleri (LAD ve FBD)

Parametre	Veri tipi[1]	Açıklama
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, sabit	Aritmetik işlemin girişleri
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Aritmetik işlemin çıkışı

[1] IN1, IN2 ve OUT parametreleri aynı veri tipine sahip olmalıdır.



Bir ADD veya MUL girişi eklemek için "Oluştur" sembolüne tıklayın veya mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına tıklayın ve "Giriş ekle" komutunu seçin.

Bir girişi silmek için, mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına (başlangıçtaki iki girişten daha fazla giriş olduğu sürece) tıklayın ve "Sil" komutunu seçin.

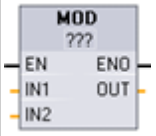
Etkinleştirildiyse (EN = 1), aritmetik yordam, giriş değerleri (IN1 ve IN2) için belirtilen fonksiyonu uygular ve sonucu, çıkış parametresi (OUT) tarafından belirtilen bellek adresine kaydeder. İşlem başarıyla gerçekleştirildikten sonra ENO = 1 olarak ayarlanır.

Tablo 8-46 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Aritmetik işlemin sonucunda elde edilen değer, seçilen veri tipi için geçerli sayı aralığının dışındadır. Sonucun, hedef büyüklüğe uygun en düşük değerli kısmı geri döndürülür.
0	0'a bölme (IN2 = 0): Sonuç tanımsızdır ve sıfır değeri döndürülür.
0	Real/LReal: Değerlerden biri NaN (not a number, sayı değil) ise, NaN değeri döndürülür
0	ADD Real/LReal: Her iki giriş değeri de, farklı ön işaretlere sahip INF (sonsuz) ise, bu işlem geçersizdir ve sonuç olarak NaN döndürülür.
0	SUB Real/LReal: Her iki giriş değeri de aynı ön işarete sahip INF (sonsuz) ise, bu işlem geçersizdir ve sonuç olarak NaN döndürülür.
0	MUL Real/LReal: Bir giriş değeri sıfır ve diğer giriş değeri INF ise, bu işlem geçersizdir ve sonuç olarak NaN döndürülür.
0	DIV Real/LReal: Her iki giriş değeri de sıfır veya INF ise, bu işlem geçersizdir ve sonuç olarak NaN döndürülür.

8.5.3 MOD (bir bölme işleminin kalanı)

Tablo 8-47 Modulo (bir bölme işleminin kalanı) yordamı

LAD/FBD	SCL	Açıklama
	**out := in1 MOD in2;**	MOD yordamıyla, bir tam sayı bölme işleminin kalanını yayınlayabilirsiniz. IN1 girişindeki değer, IN2 girişindeki değere bölünür ve bölme işleminin kalanı, OUT çıkışında yayınlanır.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-48 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
IN1 ve IN2	SInt, Int, DInt, USInt, UInt, UDInt, sabiti	Modulo girişleri
OUT	SInt, Int, DInt, USInt, UInt, UDInt	Modulo çıkışı

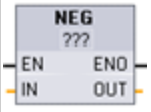
[1] IN1, IN2 ve OUT parametreleri aynı veri tipine sahip olmalıdır.

Tablo 8-49 ENO değerleri

ENO	Açıklama
1	Hata yok
0	IN2 değeri = 0 olur, OUT parametresine sıfır değeri atanır

8.5.4 NEG (ikili tümleyen oluştur)

Tablo 8-50 NEG (ikili tümleyen oluştur) yordamı

LAD/FBD	SCL	Açıklama
	**-(in);**	NEG yordamıyla, IN parametresinin değerinin ön işareti ters çevrilir ve sonuç, OUT parametresine kaydedilir.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-51 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
IN	SInt, Int, DInt, Real, LReal, sabit	Aritmetik işlemin girişi
OUT	SInt, Int, DInt, Real, LReal	Aritmetik işlemin çıkışı

[1] IN ve OUT parametreleri aynı veri tipine sahip olmalıdır.

Tablo 8-52 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Elde edilen değer, seçilen veri tipi için geçerli sayı aralığının dışında bulunmaktadır. SInt ile ilgili örnek: NEG (-128) işlemi +128 olarak sonuçlanır, bu da veri tipi için tanımlanan maksimum değeri aşar.

8.5.5 INC (arttır) ve DEC (azalt)

Tablo 8-53 INC ve DEC yordamları

LAD/FBD	SCL	Açıklama
	**in_out := in_out + 1;**	Ön işareti olan veya olmayan bir tamsayı değerini arttırır: IN_OUT -Değer +1 = IN_OUT -Değer
	**in_out := in_out - 1;**	Ön işareti olan veya olmayan bir tamsayı değerini azaltır: IN_OUT -Değer - 1 = IN_OUT -Değer

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-54 Parametreler için kullanılan veri tipleri

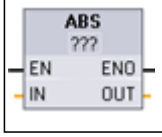
Parametre	Veri tipi	Açıklama
IN/OUT	SInt, Int, DInt, USInt, UInt, UDInt	Aritmetik işlemin girişi ve çıkışı

Tablo 8-55 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Elde edilen değer, seçilen veri tipi için geçerli sayı aralığının dışında bulunmaktadır. SInt ile ilgili örnek: INC (+127) işlemi +128 olarak sonuçlanır, bu da veri tipi için tanımlanan maksimum değeri aşar.

8.5.6 ABS (mutlak değer oluştur)

Tablo 8-56 ABS (mutlak değer oluştur) yordamı

LAD/FBD	SCL	Açıklama
	**out := ABS(in);**	IN parametresindeki ön işaretli bir tamsayının veya gerçek sayının mutlak değerini hesaplar ve sonucu OUT parametresine kaydeder.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-57 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
IN	SInt, Int, DInt, Real, LReal	Aritmetik işlemin girişi
OUT	SInt, Int, DInt, Real, LReal	Aritmetik işlemin çıkışı

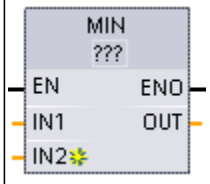
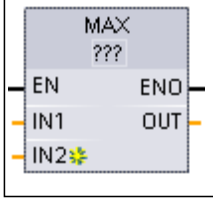
[1] IN ve OUT parametreleri aynı veri tipine sahip olmalıdır.

Tablo 8-58 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Aritmetik işlemin sonucunda elde edilen değer, seçilen veri tipi için geçerli sayı aralığının dışındadır. SInt ile ilgili örnek: ABS (-128) işlemi +128 olarak sonuçlanır, bu da veri tipi için tanımlanan maksimum değeri aşar.

8.5.7 MIN (minimumu çağır) ve MAX (maksimumu çağır)

Tablo 8-59 MIN (minimumu çağır) ve MAX (maksimumu çağır) yordamları

LAD/FBD	SCL	Açıklama
	**out:= MIN(in1:=_variant_in_, in2:=_variant_in_ [,...in32]);**	MIN yordamı, IN1 ve IN2 şeklindeki iki parametrenin değerini karşılaştırır ve daha küçük olan değeri OUT parametresine atar.
	**out:= MAX(in1:=_variant_in_, in2:=_variant_in_ [,...in32]);**	MAX yordamı, IN1 ve IN2 şeklindeki iki parametrenin değerini karşılaştırır ve daha büyük olan değeri OUT parametresine atar.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-60 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
IN1, IN2 [...IN32]	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Time, Date, TOD, sabiti	Aritmetik işlemin girişleri (maks. 32 giriş)
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Time, Date, TOD	Aritmetik işlemin çıkışı

1 IN1, IN2 ve OUT parametreleri aynı veri tipine sahip olmalıdır.



Bir giriş eklemek için "Oluştur" sembolüne tıklayın veya mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına tıklayın ve "Giriş ekle" komutunu seçin.

Bir girişi silmek için, mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına (başlangıçtaki iki girişten daha fazla giriş olduğu sürece) tıklayın ve "Sil" komutunu seçin.

Tablo 8-61 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Sadece Real veri tipinde: • En azından bir giriş gerçek sayı değildir (NaN). • Sonuçta elde edilen OUT çıkışı +/- INF'dir (sonsuz).

8.5.8 LIMIT (limiti ayarla)

Tablo 8-62 LIMIT (limiti ayarla) yordamı

LAD/FBD	SCL	Açıklama
	**LIMIT(MN:=_variant_in_, IN:=_variant_in_, MX:=_variant_in_, OUT:=_variant_out_);**	Limit yordamıyla, IN parametresinin değerinin, MIN ve MAX and if not, clamps the value at MIN or MAX. parametreleriyle belirtilen değer aralığının içinde olup olmadığını kontrol edebilirsiniz.

1 LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-63 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
MN, IN ve MX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Time, Date, TOD· sabiti	Aritmetik işlemin girişleri
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Time, Date, TOD	Aritmetik işlemin çıkışı

1 MN, IN, MX ve OUT parametreleri aynı veri tipine sahip olmalıdır.

IN parametresinin değeri belirtilen aralığın içindeyse, IN için tanımlanan değer, OUT parametresine kaydedilir. IN parametresinin değeri belirtilen aralığın dışındaysa, OUT parametresinde, MIN parametresinin değeri (IN değeri, MIN değerinden küçükse) veya MAX parametresinin değeri (IN değeri, MAX değerinden büyükse) yayınlanır.

Tablo 8-64 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Gerçek: MIN, IN veya MAX için kullanılan değerlerden biri sayı değilse (NaN), NaN ifadesi döndürülür.
0	MIN değeri MAX değerinden büyükse, OUT çıkışına IN değeri atanır.

SCL ile ilgili örnekler:

- MyVal := LIMIT(MN:=10,IN:=53, MX:=40); //Sonuç: MyVal = 40
- MyVal := LIMIT(MN:=10,IN:=37, MX:=40); //Sonuç: MyVal = 37
- MyVal := LIMIT(MN:=10,IN:=8, MX:=40); //Sonuç: MyVal = 10

8.5.9 Üs, logaritma ve trigonometri yordamları

Kayar nokta yordamları aracılığıyla, Real veya LReal veri tipiyle aritmetik fonksiyonları programlayabilirsiniz:

- SQR: Karesini al ($IN^2 = OUT$)
- SQRT: Karekökünü al ($\sqrt{IN} = OUT$)
- LN: Doğal logaritmasını al (LN(IN = OUT)
- EXP: e tabanı = 2,71828182845904523536 ile üs değerini al ($e^{IN} = OUT$),
- EXPT: Üssünü al ($IN1^{IN2} = OUT$)
  EXPT- parametreleri IN1 ve OUT, her zaman Real veya LReal arasında seçim yapmanızı gerektiren aynı veri tipine sahiptir. IN2 üs parametresi için veri tipini birçok veri tipi arasından seçebilirsiniz.
- FRAC: Ondalık basamak değerini döndür (kayar noktalı sayının ondalık basamak değeri IN = OUT)
- SIN: Sinüs değerini al (sin(IN radyan) = OUT)
- ASIN: Arc sinüs değerini al (arcsin(IN) = OUT radyan), burada sin(OUT radyan) = IN
- COS: Kosinüs değerini al (cos(IN radyan) = OUT)
- ACOS: Arc kosinüs değerini al (arcsin(IN) = OUT radyan), burada cos(OUT radyan) = IN
- TAN: Tanjant değerini al (tan(IN radyan) = OUT)
- ATAN: Arc tanjant değerini al (arctan(IN) = OUT radyan), burada tan(OUT radyan) = IN

Tablo 8-65 Aritmetik kayar nokta yordamlarıyla ilgili örnekler

LAD/FBD	SCL	Açıklama
	**out := SQR(in);** veya **out := in * in;**	Karesini al: IN ^{2} = OUT Örnek: IN = 9 ise, OUT = 81 olur.
	**out := in1 ** in2;**	Üssünü al: IN1 ^{IN2} = OUT Örnek: IN1 = 3 ve IN2 = 2 ise, OUT = 9 olur.

[1] LAD ve FBD'de: (Yordam adlarında) "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

[2] SCL'de: SCL'de matematiksel ifadeleri temel matematiksel operatörlerle de oluşturabilirsiniz.

Tablo 8-66 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN, IN1	Real, LReal, sabit	Girişler
IN2	SInt, Int, DInt, USInt, UInt,UDInt, Real, LReal, sabit	Giriş EXPT-Üs
OUT	Real, LReal	Çıkışlar

Tablo 8-67 ENO durumu

ENO	Yordam	Koşul	Sonuç (OUT)
1	Tüm	Hata yok	Geçerli sonuç
0	SQR	Sonuç, Real/LReal için tanımlanan geçerli aralığı aşıyor	+INF
		IN, +/- NaN'dir (sayı değil)	+NaN
	SQRT	IN negatiftir	-NaN
		IN, +/- INF (sonsuz) veya +/- NaN'dir	+/- INF veya +/- NaN
	LN	IN; 0,0, negatif, -INF veya -NaN'dir	-NaN
		IN; +INF veya +NaN'dir	+INF veya +NaN
	EXP	Sonuç, Real/LReal için tanımlanan geçerli aralığı aşıyor	+INF
		IN; +/- NaN'dir	+/- NaN
	SIN, COS, TAN	IN; +/- INF veya +/- NaN'dir	+/- INF veya +/- NaN
	ASIN, ACOS	IN, -1,0 ila +1,0'lık geçerli aralığın dışındadır	+NaN
		IN; +/- NaN'dir	+/- NaN
	ATAN	IN; +/- NaN'dir	+/- NaN
	FRAC	IN; +/- INF veya +/- NaN'dir	+NaN
	EXPT	IN1; +INF'dir ve IN2; -INF değildir	+INF
		IN1 negatiftir veya -INF'dir	IN2 = Real/LReal ise +NaN, aksi takdirde -INF
		IN1 veya IN2, +/- NaN'dir	+NaN
		IN1, 0,0'dır ve IN2, Real/LReal'dir (sadece)	+NaN

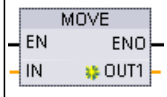
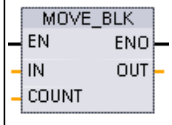
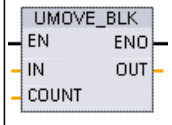
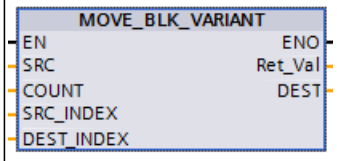
8.6 Verilerin aktarılması için kullanılan yordamlar

8.6.1 MOVE (Değeri kopyala), MOVE_BLK (Aralığı kopyala), UMOVE_BLK (Aralığı kesintisiz olarak kopyala) ve MOVE_BLK_VARIANT (Aralığı kopyala)

Aktarma yordamlarıyla, veri elemanlarını bellekteki yeni bir adrese kopyalayabilir ve verileri bir veri tipinden başka bir veri tipine dönüştürebilirsiniz. Bu nedenle kaynak veriler değişmez.

- MOVE yordamıyla, münferit bir veri elemanı, IN parametresiyle belirtilen kaynak adresten OUT parametresiyle belirtilen hedef adrese kopyalanır.
- MOVE_BLK ve UMOVE_BLK yordamları ek olarak bir COUNT parametresine sahiptir. COUNT ile kaç veri elemanının kopyalanacağı belirlenir. Kopyalanan eleman başına Bayt sayısı, PLC değişken tablosunda IN ve OUT parametrelerinin değişken adlarına hangi veri tipinin atandığına bağlıdır.

Tablo 8-68 MOVE, MOVE_BLK, UMOVE_BLK ve MOVE_BLK_VARIANT yordamları

LAD/FBD	SCL	Açıklama
	**out1 := in;**	Belirli bir adresin altına kaydedilmiş olan bir veri elemanını yeni bir adrese veya birden fazla adrese kopyalar.[1]
	**MOVE_BLK(in:=_variant_in, count:=_uint_in, out=>_variant_out);**	Veri elemanları içeren bir aralığı yeni bir adrese kopyalayan durdurulabilir aktarma işlemi.
	**UMOVE_BLK(in:=_variant_in, count:=_uint_in, out=>_variant_out);**	Veri elemanları içeren bir aralığı yeni bir adrese kopyalayan durdurulamayan aktarma işlemi.
	**MOVE_BLK_VARIANT(SRC:=_variant_in, COUNT:=_udint_in, SRC_INDEX:=_dint_in, DESTINDEX:=_dint_in, DEST=>_variant_out);**	Bir kaynak bellek alanının içeriğini bir hedef bellek alanına kopyalar. Komple bir alan veya bir alanın elemanları aynı veri tipindeki başka bir alana kopyalanabilir. Kaynak ve hedef alanın boyutları (eleman sayısı) farklı olabilir. Bir alanın birden fazla elemanı veya münferit elemanları kopyalanabilir. Kaynak ve hedef alanları göstermek için Variant veri tipleri kullanılır.

[1] MOVE yordamı: LAD veya FBD'ye başka bir çıkış eklemek için çıkış parametresinin yanındaki "Oluştur" sembolüne tıklayın. SCL'de birçok atama yordamı kullanabilirsiniz. Döngü yapılarından birini de kullanabilirsiniz.

Tablo 8-69 MOVE yordamı için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Char, WChar, Array, Struct, DTL, Time, Date, TOD, IEC veri tipleri, PLC veri tipleri	Kaynak adres
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Char, WChar, Array, Struct, DTL, Time, Date, TOD, IEC veri tipleri, PLC veri tipleri	Hedef adres



MOVE çıkışları eklemek için "Oluştur" sembolüne tıklayın veya mevcut OUT parametrelerinden birinde farenin sağ tuşuyla çıkış bağlantısına tıklayın ve "Çıkış ekle" komutunu seçin.

Bir çıkışı silmek için, mevcut OUT parametrelerinden birinde farenin sağ tuşuyla çıkış bağlantısına (başlangıçtaki iki çıkıştan daha fazla çıkış olduğu sürece) tıklayın ve "Sil" komutunu seçin.

Tablo 8-70 MOVE_BLK ve UMOVE_BLK yordamları için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal Byte, Word, DWord, Time, Date, TOD, WChar	Kaynağın başlangıç adresi
COUNT	UInt	Kopyalanması gereken veri elemanı sayısı
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, WChar	Hedefin başlangıç adresi

Tablo 8-71 MOVE_BLK_VARIANT yordamı için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
SRC	Variant (bir alanı veya münferit bir saha elemanını gösterir)	Kopyalama için kullanılan kaynak aralık
COUNT	UDInt	Kopyalanması gereken veri elemanı sayısı
SRC_INDEX	DInt	SRC alanına sıfır tabanlı indeks
DEST_INDEX	DInt	DEST alanına yönelik sıfır tabanlı indeks
RET_VAL	Int	Hata bilgileri
DEST	Variant (bir alanı veya münferit bir saha elemanını gösterir)	Kaynak aralığın içeriklerinin kopyalanacağı hedef aralık

NOT

Verilerin aktarılması için kullanılan yordamlarla ilgili kurallar

- Bool veri tipindeki verileri kopyalamak için SET_BF, RESET_BF, R, S'yi veya birçıkış bobinini (LAD) (Sayfa 206) kullanın.
- Münferit bir temel veri tipinin verilerini kopyalamak için MOVE öğesini kullanın.
- Temel bir veri tipinin bir alanı kopyalamak için MOVE_BLK veya UMOVE_BLK öğesini kullanın.
- Bir yapıyı kopyalamak için MOVE öğesini kullanın.
- Bir karakter dizisini kopyalamak için S_MOVE (Sayfa 320) işlemini kullanın.
- Bir karakter dizisindeki münferit bir karakteri kopyalamak için MOVE işlemini kullanın.
- MOVE_BLK ve UMOVE_BLK işlemlerini, alanları veya yapıları E, A veya M bellek alanlarına kopyalamak için kullanamazsınız.

MOVE_BLK ve UMOVE_BLK işlemleri, alarmların işlenmesi anlamında farklıdır:

- Alarm olayları **bekleme kuyruğuna** alınır ve MOVE_BLK işleminin yürütülmesi sırasında işlenir. MOVE_BLK işlemini, aktarma işleminin hedef adresindeki veriler bir alarm OB'sinin alt programında kullanılmıyorsa veya kullanılıyorsa hedef verilerin tutarlı olması gerekmiyorsa kullanabilirsiniz. Bir MOVE_BLK işlemi durdurulursa, hedef adresteki en son aktarılan veri elemanı eksiksiz ve tutarlı olur. MOVE_BLK işlemi, alarm OB yürütüldükten sonra devam ettirilir.
- Alarm olayları **bekleme kuyruğuna alınır, fakat ancak** UMOVE_BLK işleminin yürütülmesi tamamlandığında işlenir. UMOVE_BLK işlemini, aktarma işlemi tamamlandıysa ve bir alarm OB'sinin alt programı yürütülmeden önce hedef verilerin tutarlı olması gerekiyorsa kullanın. Daha fazla bilgi için Veri tutarlılığı (Sayfa 181) konusuyla ilgili alt bölüme bakın.

ENO, MOVE işlemi yürütüldükten sonra her zaman doğrudur.

Tablo 8-72 ENO'nun durumu

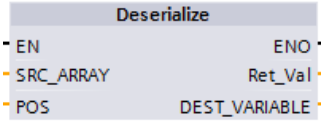
ENO	Koşul	Sonuç
1	Hata yok	Tüm COUNT elemanları başarıyla kopyalandı.
0	Kaynak aralık (IN) veya hedef aralık (OUT), mevcut bellek alanını aşıyor.	Uygun elemanlar kopyalanır. Hiçbir kısmi eleman kopyalanmaz.

Tablo 8-73 MOVE_BLK_VARIANT yordamı için kullanılan koşul kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
80B4	Veri tipleri uyuşmuyor.
8151	SRC parametresine erişim mümkün değil.
8152	SRC parametresinin işlemcisi geçersiz bir tipe sahip.
8153	SRC parametresindeki kod oluşturma işleminde hata
8154	SRC parametresinin işlemcisi Bool veri tipine sahip.
8281	COUNT parametresi geçersiz bir değere sahip.
8382	SRC_INDEX parametresinin değeri, Variant için tanımlanan limitin dışında.
8383	SRC_INDEX parametresinin değeri, alan için tanımlanan üst limiti aşıyor.
8482	DEST_INDEX parametresinin değeri, Variant için tanımlanan limitin dışında.
8483	DEST_INDEX parametresinin değeri, alan için tanımlanan üst limiti aşıyor.
8534	DEST parametresi yazma korumalıdır.
8551	DEST parametresine erişim mümkün değil.
8552	DEST parametresinin işlemcisi geçersiz bir tipe sahip.
8553	DEST parametresindeki kod oluşturma işleminde hata
8554	DEST parametresinin işlemcisi Bool veri tipine sahip.
*Hata kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir.	

8.6.2 Deserialize

Tablo 8-74 DESERIALIZE yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := Deserialize(SRC_ARRAY:=_variant_in_, DEST_VARIABLE=>_variant_out_, POS:=_dint_inout_);**	"Deserialize" yordamıyla, bir PLC veri tipinin (UDT) sıralı gösterimi bir PLC veri tipine geri dönüştürülebilir ve tüm içeriği doldurulabilir. Karşılaştırma DOĞRU ise, kutu çıkışı DOĞRU olur.

Bir PLC veri tipinin sıralı gösterimini içeren bellek alanı "Array of Byte" veri tipinde olmalı ve veri bloğu standart erişim için (optimize edilmiş erişim değil) deklare edilmiş olmalıdır. Dönüştürme işleminden önce, yeterli bellek alanının mevcut olup olmadığı kontrol edilmelidir.

Bu yordamlar, dönüştürülmüş PLC veri tiplerinin birden fazla sıralı gösterimi başlangıçtaki veri tiplerine geri dönüştürülebilir.

NOT

Bir PLC veri tipinin (UDT) sadece tek bir sıralı gösteriminin geri dönüştürülmesi gerekiyorsa, "TRCV: Verileri iletişim bağlantısı üzerinden al" komutu kullanılabilir.

Tablo 8-75 DESERIALIZE yordamı için kullanılan parametreler

Parametre	Tip	Veri tipi	Açıklama
SRC_ARRAY	IN	Variant	Veri akışıyla birlikte global veri bloğu
DEST_VARIABLE	INOUT	Variant	Dönüştürülen PLC veri tipinin (UDT) kaydedilmesi gereken değişken
POS	INOUT	DInt	Dönüştürülen PLC veri tipinin kullandığı Bayt'ların sayısı
RET_VAL	OUT	Int	Hata bilgileri

Tablo 8-76 RET_VAL parametresi

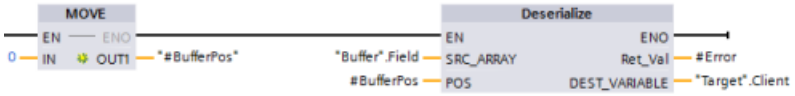
RET_VAL* (W#16#...)	Açıklama
0000	Hata yok
80B0	SRC_ARRAY ve DEST_VARIABLE parametrelerinin bellek alanları üst çakışıyor.
8136	DEST_VARIABLE parametresindeki veri bloğu standart erişime sahip bir blok değil.
8150	SRC_ARRAY parametresinin Variant veri tipi değer içermiyor.
8151	SRC_ARRAY parametresindeki kod oluşturma işleminde hata.
8153	SRC_ARRAY parametresi için yeterli bellek alanı yok.
8250	DEST_VARIABLE parametresinin Variant veri tipi değer içermiyor.
8251	DEST_VARIABLE parametresindeki kod oluşturma işleminde hata.

RET_VAL* (W#16#...)	Açıklama
8254	DEST_VARIABLE parametresi için geçersiz veri tipi.
8382	POS parametresinin değeri, alan için tanımlanan limitin dışında.
*Hata kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir.	

Örnek: "Deserialize" yordamı

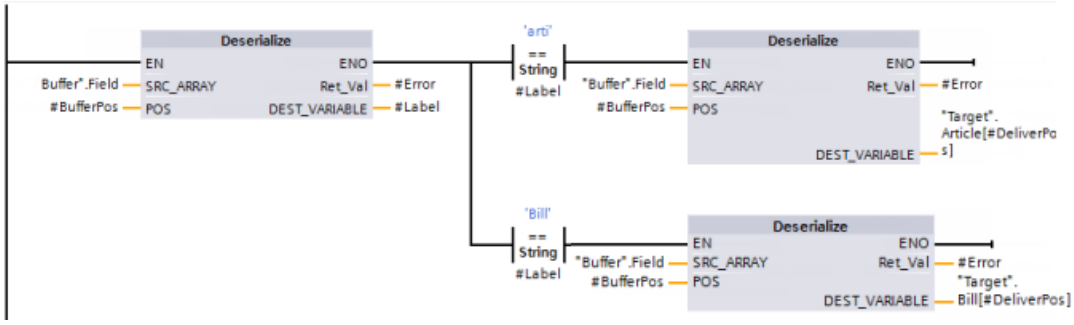
Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:

Ağ 1:



"MOVE" yordamıyla, "0" değeri, "#BufferPos" veri bloğunun değişkenine kopyalanır. Daha sonra Deserialize yordamıyla, müşteri verilerinin sıralı gösterimi "Buffer" veri bloğundan ayrıştırılır ve "Target" veri bloğuna yazılır. Deserialize yordamıyla, dönüştürülen veriler tarafından kullanılan Bayt'ların sayısı hesaplanır ve "#BufferPos" veri bloğunun değişkenlerine kaydedilir.

Ağ 2:



"Deserialize" yordamıyla, "Buffer" ile gösterilen veri akışının sıralı gösterimi ayrıştırılır ve karakterler "#Label" işlemcisine yazılır. Lojik, karakterleri "arti" ve "Bill" karşılaştırma işlemlerinin yardımıyla karşılaştırır. "arti" için karşılaştırma = DOĞRU ise veriler, ayrıştırılan ve "Target" veri bloğunun "Article" veri yapısına yazılan sipariş verileridir. "Bill" için karşılaştırma = DOĞRU ise veriler, ayrıştırılan ve "Target" veri bloğunun "Bill" veri yapısına yazılan hesap kapatma verileridir.

Fonksiyon bloğu arayüzü (veya fonksiyon arayüzü):

Name	Data type
Input	
DeliverPos	Int
Output	
InOut	
Static	
Temp	
BufferPos	DInt
Error	Int
Label	String[4]

Kullanıcı tanımlı PLC veri tipleri:

Bu örnek için her iki PLC veri tipinin de (UDT) yapıları aşağıda gösterilmiştir:

Article

	Name	Data type
1	Number	DInt
2	Declaration	String
3	Colli	Int

Client

	Name	Data type
1	Title	Int
2	Firstname	String[10]
3	Surname	String[10]

Veri blokları:
Bu örnek için her iki veri bloğu da aşağıda gösterilmiştir:

Target

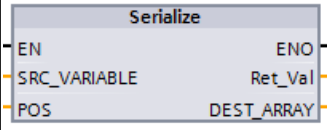
	Name	Data type
1	Static	
2	Client	"Client"
3	Article	Array[0..10] of "Article"
4	Bill	Array[0..10] of Int

Buffer

	Name	Data type
1	Static	
2	Field	Array[0..294] of Byte

8.6.3 Serialize

Tablo 8-77 SERIALIZE yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := Serialize(SRC_VARIABLE=>_variant_in_, DEST_ARRAY:=_variant_out_, POS:=_dint_inout_);**	"Serialize" yordamıyla, birden fazla PLC veri tipi (UDT), herhangi bir veri kaybı olmadan sıralı bir gösterime dönüştürülebilir.

Bu yordam, çok sayıda yapılandırılmış veri elemanını programınızdan geçici olarak bir tampona, örneğin global bir veri bloğuna yerleştirmek ve ardından bunları başka bir CPU'ya aktarmak için kullanılabilir. Dönüştürülen PLC veri tiplerinin yerleştirileceği bellek alanı, "ARRAY of BYTE" veri tipinde ve standart erişimle deklare edilmiş olmalıdır. Dönüştürme işleminden önce, yeterli bellek alanının mevcut olup olmadığı kontrol edilmelidir.

POS parametresi, dönüştürülen PLC veri tiplerini kullanan Bayt'ların sayısıyla ilgili bilgileri içerir.

NOT

Sadece tek bir PLC veri tipi (UDT) aktarılacaksa, bunun için "TSEND: Verileri iletişim bağlantısı üzerinden gönder" yordamı kullanılabilir.

Tablo 8-78 SERIALIZE yordamı için kullanılan parametreler

Parametre	Tip	Veri tipi	Açıklama
SRC_VARIABLE	IN	Variant	Bir seri gösterime dönüştürülmesi gereken PLC veri tipi (UDT)l
DEST_ARRAY	INOUT	Variant	Oluşturulan veri akışının kaydedilmesi gereken veri bloğu
POS	INOUT	DInt	Dönüştürülen PLC veri tiplerinin kullandığı Bayt'ların sayısı. Hesaplanan POS parametresi sıfır tabanlıdır.
RET_VAL	OUT	Int	Hata bilgileri

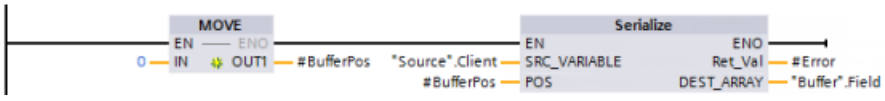
Tablo 8-79 RET_VAL parametresi

RET_VAL* (W#16#...)	Açıklama
0000	Hata yok
80B0	SRC_VARIABLE ve DEST_ARRAY parametrelerinin bellek alanları üst çakışıyor.
8150	SRC_VARIABLE parametresinin Variant veri tipi değer içermiyor.
8152	SRC_VARIABLE parametresindeki kod oluşturma işleminde hata.
8236	DEST_ARRAY parametresindeki veri bloğu standart erişime sahip bir blok değil.
8250	DEST_ARRAY parametresinin Variant veri tipi değer içermiyor.
8252	DEST_ARRAY parametresindeki kod oluşturma işleminde hata.
8253	DEST_ARRAY parametresi için yeterli bellek alanı yok.
8254	DEST_VARIABLE parametresi için geçersiz veri tipi.
8382	POS parametresinin değeri, alan için tanımlanan limitin dışında.
*Hata kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir.	

Örnek: Serialize yordamı

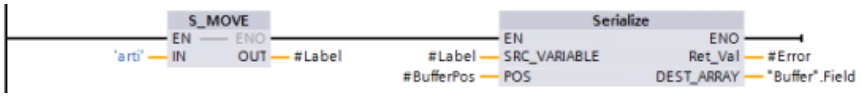
Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:

Ağ 1:



"MOVE" yordamıyla, "0" değeri "#BufferPos" parametresine kopyalanır. "Serialize" yordamıyla müşteri verileri "Source" veri bloğundan serileştirilir ve sıralı gösterimle "Buffer" veri bloğuna yazılır. Yordam, sıralı gösterim tarafından kullanılan Bayt'ların sayısını "#BufferPos" parametresine kaydeder.

Ağ 2:



Ardından lojik, sıralı gösterimin daha sonra daha basit bir şekilde ayrıştırılabilmesi için bir ayırma metni ekler. "S_MOVE" yordamıyla, "arti karakter dizisi "#Label" parametresine kopyalanır. "Serialize" yordamı, bu karakterleri kaynak verilerinden sonra "Buffer" veri bloğuna

yazar. Yordam, "arti" metin karakter dizisindeki Bayt sayısını "#BufferPos" parametresinde halihazırda kaydedilmiş olan sayıya ekler.

Ağ 3:



"Serialize" yordamı, belirli bir ürünün, çalışma zamanı sırasında hesaplanan verilerini "Source" veri bloğundan serileştirir ve bunları, sıralı gösterimle "Buffer" veri bloğuna "arti" karakterlerinden sonra yazar.

Blok arayüzü:

	Name	Data type
	Input	
	DeliverPos	Int
	Output	
	InOut	
	Static	
	Temp	
	BufferPos	DInt
	Error	Int
	Label	String[4]

Kullanıcı tanımlı PLC veri tipleri:

Bu örnek için her iki PLC veri tipinin de (UDT) yapıları aşağıda gösterilmiştir:

Article

	Name	Data type
1	Number	DInt
2	Declaration	String
3	Colli	Int

Client

	Name	Data type
1	Title	Int
2	Firstname	String[10]
3	Surname	String[10]

Veri blokları:

Bu iki örnek için kullanılan iki veri bloğu aşağıda gösterilmektedir:

Source

	Name	Data type
1	Static	
2	Client	"Client"
3	Article	Array[0..10] of "Article"

Buffer

	Name	Data type
1	Static	
2	Field	Array[0..294] of Byte

8.6.4 FILL_BLK (Belleği Bit örüntüsüyle doldur) ve UFILL_BLK (Belleği kesintisiz şekilde Bit örüntüsüyle doldur)

Tablo 8-80 FILL_BLK ve UFILL_BLK yordamları

LAD/FBD	SCL	Açıklama
	**FILL_BLK(** **in:=_variant_in,** **count:=int,** **out=>_variant_out);**	Kesintisiz şekilde doldur yordamı: Bir adres aralığını belirtilen bir veri elemanının kopyalarıyla doldurur.
	**UFILL_BLK(** **in:=_variant_in,** **count:=int,** **out=>_variant_out);**	Kesintisiz şekilde doldur yordamı: Bir adres aralığını belirtilen bir veri elemanının kopyalarıyla doldurur.

Tablo 8-81 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Veri kaynağı adresi
COUNT	UDint, USInt, UInt	Kopyalanması gereken veri elemanı sayısı
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Veri hedefi adresi

NOT

Belleği Bit örüntüsüyle doldur yordamlarıyla ilgili kurallar

- BOOL veri tipindeki verileri kopyalamak için SET_BF, RESET_BF, R, S'yi veya bir çıkış bobinini (LAD) kullanın.
- Belleği münferit bir temel veri tipiyle doldurmak için MOVE işlemini kullanın.
- Bir alanı temel bir veri tipiyle doldurmak için FILL_BLK veya UFILL_BLK yordamını kullanın.
- Bir karakter dizisindeki münferit bir karakteri kopyalamak için MOVE işlemini kullanın.
- FILL_BLK ve UFILL_BLK yordamlarını, E, A veya M bellek alanlarındaki alanları doldurmak için kullanamazsınız.

FILL_BLK ve UFILL_BLK yordamlarıyla, kayna veri elemanı IN hedef adrese kopyalanır, bu sırada OUT parametresindeki başlangıç adresi belirlenir. Kopyalama işlemi, ilgili bitişik adres aralıklarında, kopya sayısı, COUNT parametresinin değeriyle aynı olana kadar devam eder.
FILL_BLK ve UFILL_BLK yordamları, alarmların işlenmesi konusunda farklıdır:

- Alarm olayları **bekleme kuyruğuna** alınır ve FILL_BLK yordamının yürütülmesi esnasında işlenir. FILL_BLK yordamını, aktarma işleminin hedef adresindeki veriler bir alarm OB'sinin alt programında kullanılmıyorsa veya kullanılıyorsa hedef verilerin tutarlı olması gerekmiyorsa kullanabilirsiniz.
- Alarm olayları **bekleme kuyruğuna alınır, fakat ancak** UFILL_BLK yordamının yürütülmesi tamamlandığında işlenir. UFILL_BLK yordamını, aktarma yordamı tamamlandıysa ve bir alarm OB'sinin alt programı yürütülmeden önce hedef verilerin tutarlı olması gerekiyorsa kullanın.

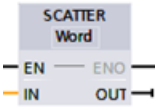
Tablo 8-82 ENO durumu

ENO	Koşul	Sonuç
1	Hata yok	IN elemanı başarılı bir şekilde tüm COUNT hedeflerine kopyalandı.
0	Hedef aralık (OUT), mevcut bellek alanını aşıyor	Uygun elemanlar kopyalanır. Hiçbir kısmi eleman kopyalanmaz.

8.6.5 SCATTER

SCATTER: Bit dizisinin münferit Bit'lere ayrılması

Tablo 8-83 SCATTER

LAD/FBD	SCL	Açıklama
	**SCATTER** **(IN := #SourceWord,** **OUT => #DestinationArray};**	SCATTER: Bit dizisini münferit Bit'lere ayırma yordamı, Byte, Word veya DWord veri tipindeki bir değişkeni münferit Bit'lere ayırır ve bunları Boolean, Struct elemanlarından oluşan bir Array veri tipinde veya sadece Boole elemanları içeren bir PLC veri tipinde kaydeder. Bu sayede, örneğin bir durum sözcüğünü analiz edebilir ve münferit Bit'lerin durumunu indeks yardımıyla okuyabilir ve değiştirebilirsiniz. GATHER ile Bit'leri yeniden bir Bit dizisi halinde birleştirebilirsiniz.

SCATTER yordamı için kullanılan veri tipleri

Aşağıdaki tablo, yordamın parametrelerini gösterir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
EN	Input	Bool	E, A, M, D, L veya sabit	Serbest bırakma girişi
ENO	Output	Bool	E, A, M, D, L	Serbest bırakma çıkışı ENO serbest bırakma çıkışı, aşağıdaki koşullardan biri sağlandığı takdirde "0" sinyal durumunu geri döndürür: • EN serbest bırakma girişi "0" sinyal durumuna sahiptir. • Array, Struct veya PLC veri tipi, yeterli miktarda Bool elemanı sağlamaz.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
IN	Input	Byte, Word, DWord	E, A, M, D, L	Analiz edilen Bit dizisi. Değerler, bir teknoloji nesnesinin G/Ç aralığında veya DB'sinde bulunamaz.
OUT	Output	Bool, Struct veya PLC elemanlarından oluşan Array[*] veri tipi *: 8, 16, 32 veya 64 eleman	E, A, M, D, L	Münferit Bit'lerin kaydedildiği Array, Struct veya PLC veri tipi.

NOT

Bool'dan oluşan çok boyutlu Array

Bir Bit dizisini münferit Bit'lere ayırma yordamında Bool'dan oluşan çok boyutlu bir Array veri tipinin kullanımına izin verilmez.

NOT

Array, Struct veya PLC veri tipinin uzunluğu

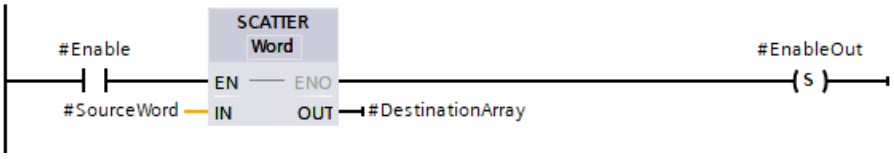
Array, Struct veya PLC veri tipi, tam olarak Bit dizisinde belirtilen eleman sayısına sahip olmalıdır. Bu, örneğin Byte veri tipinde; Array, Struct veya PLC veri tipinin tam olarak 8 elemana (Word = 16 ve DWord = 32) sahip olması gerektiği anlamına gelir.

Örnek: Bir Array'e sahip SCATTER yordamı

Blok arayüzünde aşağıdaki değişkenleri oluşturun:

Değişken	Aralık	Veri tipi
Enable	Giriş	Bool
SourceWord		Word
EnableOut	Çıkış	Bool
DestinationArray		Bool'dan oluşan Array [0..15]

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



Aşağıdaki tabloda, yordamın özel işlemci değerleriyle nasıl çalıştığı gösterilmektedir:

Parametre	İşlemci	Veri tipi
IN	SourceWord	Word (16 Bit)
OUT	DestinationUDT	"DestinationUDT" işlemcisi PLC (UDT) veri tipine sahiptir. 16 elemandan oluşur ve bu sayede, analiz edilmesi gereken Word veri tipiyle aynı büyüklüktedir.

#Enable işlemcisi EN serbest bırakma girişinde "1" sinyal durumunu döndürürse yordam yürütülür. Word veri tipindeki #SourceWord işlemcisi, münferit Bit'lere (16) ayrılır ve #DestinationArray işlemcisinin münferit elemanlarına atanır. Yordam yürütülürken bir hata oluşursa, #EnableOut işlemcisi, ENO serbest bırakma çıkışında "0" sinyal durumunu geri döndürür.

Örnek: Bir PLC (UDT) veri tipine sahip SCATTER yordamı

Şu "myBits" PLC veri tipini oluşturun:

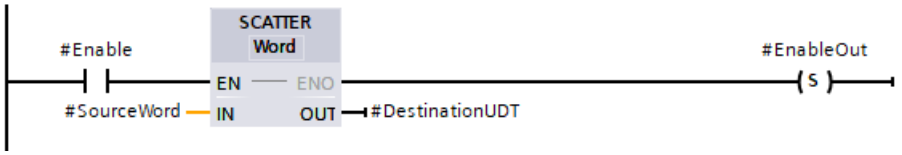
myBits

	Name	Data type
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Blok arayüzünde aşağıdaki değişkenleri oluşturun:

Değişken	Aralık	Veri tipi
Enable	Giriş	Bool
SourceWord		Word
EnableOut	Çıkış	Bool
DestinationUDT		"myBits"

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



Aşağıdaki tabloda, yordamın özel işlemci değerleriyle nasıl çalıştığı gösterilmektedir:

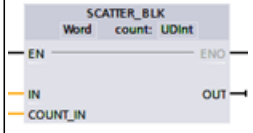
Parametre	İşlemci	Veri tipi
IN	SourceWord	Word (16 Bit)
OUT	DestinationUDT	"DestinationUDT" işlemcisi PLC (UDT) veri tipine sahiptir. 16 elemandan oluşur ve bu sayede, analiz edilmesi gereken Word veri tipiyle aynı büyüklüktedir.

#Enable işlemcisi EN serbest bırakma girişinde "1" sinyal durumunu döndürürse yordam yürütülür. Word veri tipindeki #SourceWord işlemcisi, münferit Bit'lere (16) ayrılır ve #DestinationArray işlemcisinin münferit elemanlarına atanır. Yordam yürütülürken bir hata oluşursa, #EnableOut işlemcisi, ENO serbest bırakma çıkışında "0" sinyal durumunu geri döndürür.

8.6.6 SCATTER_BLK

SCATTER_BLK: <Bit dizisinden> oluşan bir ARRAY'in elemanlarını münferit Bit'lere ayır

Tablo 8-84 SCATTER_BLK

LAD/FBD	SCL	Açıklama
	**SCATTER_BLK (IN:=_byte_in_, COUNT_IN:=_uint_in_, OUT=>_bool_out_);_IN:=_uint_in_,**	<Bit dizisinden> oluşan bir ARRAY'in elemanlarını münferit Bit'lere ayır yordamı, bir Array'in bir veya daha fazla elemanını bölümlere ayırır ve bunları Struct, PLC veya Bool'dan oluşan bir veri tipinde kaydeder. Bu durumda, örneğin durum sözcüklerini analiz edebilir ve münferit Bit'lerin durumunu indeks yardımıyla okuyabilir ve değiştirebilirsiniz. GATHER ile Bit'leri yeniden bir Bit dizisi halinde birleştirebilirsiniz.

SCATTER_BLK yordamı için kullanılan veri tipleri

Aşağıdaki tablo, yordamın parametrelerini gösterir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
EN	Input	Bool	E, A, M, D, L veya sabit	Serbest bırakma girişi
ENO	Output	Bool	E, A, M, D, L	Serbest bırakma çıkışı. ENO = FALSE ise, çıkışa hiçbir veri yazılmaz. ENO serbest bırakma çıkışı, aşağıdaki koşullardan biri sağlandığı takdirde "0" sinyal durumunu geri döndürür: • EN serbest bırakma girişi "0" sinyal durumuna sahiptir. • Kaynak Array, COUNT_IN parametresinde belirtilenden daha az elemana sahiptir. • Hedef Array'in indeksi Bir Byte, Word veya DWord sınırından başlamaz. Bu durumda, Bool'dan oluşan Array veri tipine hiçbir sonuç yazılmaz. • Bool'dan oluşan Array, Struct veya PLC veri tipi, gereken sayıda elemana sahip değildir.
IN	Input	Aşağıdakilerden elde edilen bir eleman: • Byte'tan oluşan Array [*] • Word'den oluşan Array [*] • DWord'den oluşan Array [*] • LWord'den oluşan Array [*]	E, A, M, D, L	Değerler, bir teknoloji nesnesinin G/Ç aralığında veya DB'sinde bulunamaz. Giriş Array'inin (IN parametresi) veri tipini (Sayfa 99), yordamın adının sol alt kısmına girebilirsiniz. IN parametresindeki kaynak Array, COUNT_IN parametresinde belirtilenden daha fazla elemana sahip olabilir.
COUNT_IN	Input	USInt, UInt, UDInt	E, A, M, D, L	OUT parametresine ayrılacak kaynak Array'in eleman sayısı için kullanılan sayaç. "count:" ifadesinin arkasındaki COUNT_IN parametresinin veri tipini, yordamı adının altındaki belirtebilirsiniz. COUNT_IN değeri, bir teknoloji nesnesinin G/Ç aralığında veya veritabanında olamaz.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
OUT	Output	Aşağıdakilerden elde edilen bir eleman: • Bool veri tipinden oluşan Array [*] • Struct veri tipinden oluşan Array [*] • <PLC veri tipinden> oluşan Array [*]	E, A, M, D, L	Münferit Bit'lerin kaydedildiği Array, Struct veya PLC veri tipi.

NOT

Bool'dan oluşan çok boyutlu Array

Dizide Bool'dan oluşan çok boyutlu bir Array veri tipi söz konusu ise, sayma sırasında boyutların tüm Bit'leri dikkate alınır.

Örnek 1: Bool'dan oluşan bir Array [1..10,0..4,1..2], toplam 100 Bit'e ayrılabilen 10x5x2 Bit'e sahiptir.

Örnek 2: Bool'dan oluşan bir Array [0..399], ayrılabilen 400 Bit'e sahiptir.

NOT

Hedef Array'in alt Array sınırı "0" değilse dikkate etmeniz gerekenler:

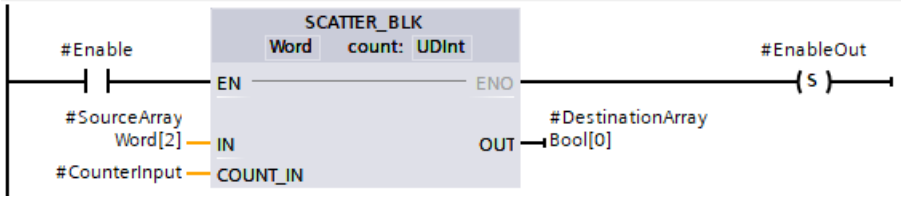
Performansla ilgili nedenlerden dolayı indeks her zaman bir Byte, Word veya DWord sınırından başlamalıdır. Bu, Array indeksinin hesaplamasının Array'in alt sınırından başlaması gerektiği anlamına gelir.

Örnek: Alt sınırı "0" olan bir hedef Array

Blok arayüzünde aşağıdaki değişkenleri oluşturun:

Değişken	Aralık	Veri tipi
Enable	Giriş	Bool
SourceArrayWord		Word'den oluşan Array [0..5]
CounterInput		UDInt
EnableOut	Çıkış	Bool
DestinationArrayBool		Bool'dan oluşan Array [0..95]

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



Aşağıdaki tabloda, yordamın özel işlemci değerleriyle nasıl çalıştığı gösterilmektedir:

Parametre	İşlemci	Veri tipi
IN	SourceArrayWord[2]	Word'den oluşan Array [0..5] (96 Bit ayrılabilir.)
COUNT_IN	CounterInput = 3	3 şeklindeki bir CounterInput değeri, üç sözcüğün veya 48 Bit'in ayrılması gerektiği anlamına gelir. Bu nedenle hedef Array'de en az 48 Bit mevcut olmalıdır.
OUT	DestinationArrayBool[0]	"DestinationArrayBool" işlemcisi, Bool'dan oluşan Array [0..95] veri tipine sahip olmalıdır. Bu, 96 Boole elemanı sağladığı anlamına gelir. OUT parametresi, ayrılan Bit dizilerinin Bit'lerini kaydedebilmek için yeterli sayıda elemana sahip olmalıdır; ancak hedef bellek alanı daha büyük de olabilir.

#Enable işlemcisi EN serbest bırakma girişinde "1" sinyal durumunu döndürürse yordam yürütülür. #SourceArrayWord işlemcisinin üçüncü, dördüncü ve beşinci sözcüğü münferit Bit'lerine (48) ayrılır ve ilk elemandan itibaren #DestinationArrayBool işlemcisinin münferit elemanlarına atanır. Yordam yürütülürken bir hata oluşursa, #EnableOut işlemcisi, ENO serbest bırakma çıkışında "0" sinyal durumunu geri döndürür.

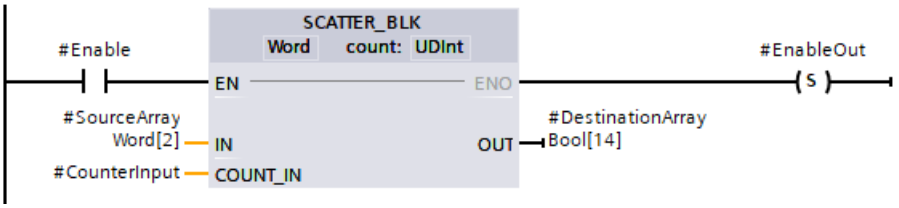
Örnek: Alt sınırı "-2" olan bir hedef Array

Bu örnekte, önceki örnek tekrarlanmakta, ancak hedef Array'in alt sınırı negatif bir değere ayarlanmakta ve #DestinationArrayBool[14]'da yayınlanmaktadır.

Blok arayüzü, DestinationArrayBool'un boyutları istisna olmak kaydıyla önceki örnektekiyle aynıdır.

Değişken	Aralık	Veri tipi
DestinationArrayBool	Çıkış	Bool'dan oluşan Array [-2..93]

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



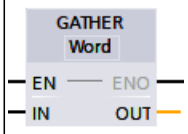
Parametre tablosu, OUT parametresi için kullanılan haricinde önceki örnektekiyle aynıdır.

Parametre	İşlemci	Veri tipi
OUT	DestinationArrayBool[14]	"DestinationArrayBool" işlemcisi, Bool'dan oluşan Array [-2..93] veri tipine sahip olmalıdır. Bu, 96 BOOL elemanı sağladığı anlamına gelir. DestinationArrayBool'un alt sınırı -2'ye ayarlandığı için, DestinationArrayBool[14], Array'in 16. elemanıdır.

8.6.7 GATHER

GATHER: Münferit Bit'leri bir Bit dizisi halinde birleştir.

Tablo 8-85 GATHER

LAD/FBD	SCL	Açıklama
	**GATHER** **(IN := #SourceArray,** **OUT => #DestinationArray};**	GATHER: "Münferit Bit'leri bir Bit dizisi halinde birleştir" yordamı, Bool, Struct veya PLC'den oluşan bir Array veri tipinin Bit'lerini sadece Boole elemanlarıyla bir Bit dizisi halinde birleştirir. Bit dizisi, BYTE, WORD veya DWORD veri tipindeki bir değişkene kaydedilir.

GATHER yordamı için kullanılan veri tipleri

Aşağıdaki tablo, yordamın parametrelerini gösterir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
EN	Input	Bool	E, A, M, D, L veya sabit	Serbest bırakma girişi
ENO	Output	Bool	E, A, M, D, L	Serbest bırakma çıkışı ENO serbest bırakma çıkışı, aşağıdaki koşullardan biri sağlandığı takdirde "0" sinyal durumunu geri döndürür: • EN serbest bırakma girişi "0" sinyal durumuna sahiptir. • Array, Struct veya PLC (UDT) veri tipi, Bit dizisi tarafından belirtilenden daha az veya daha fazla Bool elemanına sahiptir. Bu durumda Bool elemanları aktarılmaz. • Gerekenden daha az Bit mevcuttur.
IN	Input	Bool, Struct veya PLC elemanlarından oluşan Array[*] veri tipi *: 8, 16, 32 veya 64 eleman	E, A, M, D, L	Bit'leri bir Bit dizisi halinde birleştirilecek olan Array, Struct veya PLC veri tipi. Değerler, bir teknoloji nesnesinin G/Ç aralığında veya DB'sinde bulunamaz.
OUT	Output	Byte, Word, DWord	E, A, M, D, L	Bir değişkene kaydedilen birleştirilerek oluşturulan Bit dizisi

NOT

Bool'dan oluşan çok boyutlu Array

"Münferit Bit'leri bir Bit dizisi halinde birleştir" yordamında, Bool'dan oluşan çok boyutlu bir Array veri tipinin kullanımına izin verilmez.

NOT

Array, Struct veya PLC veri tipinin uzunluğu

Array, Struct veya PLC veri tipi, tam olarak Bit dizisinde belirtilen eleman sayısına sahip olmalıdır.

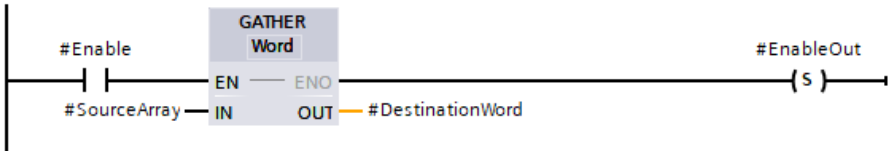
Bu, örneğin Byte veri tipinde; Array, Struct veya PLC veri tipinin tam olarak 8 elemana (Word = 16 ve DWord = 32) sahip olması gerektiği anlamına gelir.

Örnek: Bir Array'e sahip GATHER yordamı

Blok arayüzünde aşağıdaki değişkenleri oluşturun:

Değişken	Aralık	Veri tipi
Enable	Giriş	Bool
SourceArray		Bool'dan oluşan Array [0..15]
EnableOut	Çıkış	Bool
DestinationWord		Word

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



Aşağıdaki tabloda, yordamın özel işlemci değerleriyle nasıl çalıştığı gösterilmektedir:

Parametre	İşlemci	Veri tipi
IN	SourceArray	"SourceArray" işlemcisi, Bool'dan oluşan Array [0..15] veri tipindedir. 16 elemandan oluşur ve böylece, Bit'lerin birleştirilmesi gereken Word veri tipiyle aynı büyüklüktedir.
OUT	DestinationWord	Word (16 Bit)

#Enable işlemcisi EN serbest bırakma girişinde "1" sinyal durumunu döndürürse yordam yürütülür. #SourceArray işlemcisinin Bit'leri bir Word veri tipi halinde birleştirilir. Yordam yürütülürken bir hata oluşursa, #EnableOut işlemcisi, ENO serbest bırakma çıkışında "0" sinyal durumunu geri döndürür.

Örnek: Bir PLC (UDT) veri tipine sahip GATHER yordamı

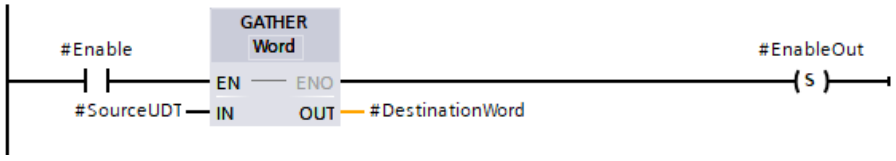
Şu "myBits" PLC veri tipini oluşturun:

myBits		
	Name	Data type
1	GeneralError	Bool
2	DeviceError	Bool
3	CommError	Bool
4	myError1	Bool
5	myError2	Bool
6	myError3	Bool
7	myError4	Bool
8	myError5	Bool
9	myError6	Bool
10	myError7	Bool
11	myError8	Bool
12	myError9	Bool
13	myError10	Bool
14	myError11	Bool
15	myError12	Bool
16	myError13	Bool

Blok arayüzünde aşağıdaki değişkenleri oluşturun:

Değişken	Aralık	Veri tipi
Enable	Giriş	Bool
SourceUDT		"myBits"
EnableOut	Çıkış	Bool
DestinationWord		Word

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



Aşağıdaki tabloda, yordamın özel işlemci değerleriyle nasıl çalıştığı gösterilmektedir:

Parametre	İşlemci	Veri tipi
IN	SourceUDT	"SourceUDT" işlemcisi PLC (UDT) veri tipine sahiptir. 16 elemandan oluşur ve böylece, Bit'lerin birleştirilmesi gereken Word veri tipiyle aynı büyüklüktedir.
OUT	DestinationWord	Word (16 Bit)

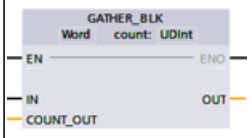
#Enable işlemcisi EN serbest bırakma girişinde "1" sinyal durumunu döndürürse yordam yürütülür. #SourceUDT işlemcisinin Bit'leri bir Word veri tipi halinde birleştirilir. Yordam yürütülürken bir hata oluşursa, #EnableOut işlemcisi, ENO serbest bırakma çıkışında "0" sinyal durumunu geri döndürür.

8.6.8 GATHER_BLK

GATHER_BLK: Münferit bitleri <Bit dizisinden> oluşan bir ARRAY'in birden fazla elemanı şeklinde birleştir

Tablo 8-86 GATHER_BLK

LAD/FBD	SCL	Açıklama

	GATHER_BLK
	(IN := #SourceArrayBool[0],
	COUNT_OUT := #CounterOutput,
	OUT => #DestinationArrayWord[2]);
		"Münferit bitleri <Bit dizisinden> oluşan bir ARRAY'in birden fazla elemanı şeklinde birleştir" yordamı, Bool, Struct veya PLC'den oluşan bir Array veri tipinin Bit'lerini sadece Boole elemanlarıyla <Bit dizisinden> oluşan bir ARRAY veri tipinin bir veya daha fazla elemanı halinde birleştirir.

GATHER_BLK yordamı için kullanılan veri tipleri

Aşağıdaki tablo, yordamın parametrelerini gösterir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
EN	Input	Bool	E, A, M, D, L veya sabit	Serbest bırakma girişi
ENO	Output	Bool	E, A, M, D, L	Serbest bırakma çıkışı. ENO = FALSE ise, çıkışa hiçbir veri yazılmaz. ENO serbest bırakma çıkışı, aşağıdaki koşullardan biri sağlandığı takdirde "0" sinyal durumunu geri döndürür: • EN serbest bırakma girişi "0" sinyal durumuna sahiptir. • Kaynak Array'in indeksi Bir Byte, Word veya DWord sınırından başlamaz. Bu durumda <Bit dizisinden> ARRAY veri tipine hiçbir sonuç yazılmaz. • <Bit dizisinden> oluşan ARRAY[*] veri tipi, gereken sayıda elemana sahip değildir.
IN	Input	Aşağıdakilerden elde edilen bir eleman:	E, A, M, D, L	Bit'leri OUT parametresinde birleştirilen Bool, Struct veya PLC'den oluşan Array veri tipi. Değerler, bir teknoloji nesnesinin G/Ç aralığında veya DB'sinde bulunamaz. Giriş Array'inin (IN parametresi) veri tipi, yordamın adının sol alt kısmına girilir.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
		• Bool veri tipinden oluşan Array [*] • Struct veri tipinden oluşan Array [*] • <PLC veri tipinden> oluşan Array [*]		
COUNT_OUT	Input	USInt, UInt, UDInt	E, A, M, D, L	Tanımlanması gereken hedef Array'in eleman sayısı. COUNT_OUT parametresinde, hedef Array'in kaç elemanının yazılması gerektiğini belirtebilirsiniz. Bu adımla, Bool, Struct veya PLC'den oluşan Array veri tipinin kaç elemanının gerektiğini de dolaylı olarak belirtebilirsiniz. Değer, bir teknoloji nesnesinin G/Ç aralığında veya veritabanında olamaz. COUNT_OUT parametresinin veri tipi, yordamın adının sağ alt kısmında belirtilir.
OUT	Output	Aşağıdakilerden elde edilen bir eleman: • Byte'tan oluşan Array [*] • Word'den oluşan Array [*] • DWord'den oluşan Array [*] • LWord'den oluşan Array [*]	E, A, M, D, L	Bit'lerin kaydedildiği Array. OUT parametresindeki hedef Array, COUNT_OUT parametresinde belirtilenden daha fazla eleman içerebilir.

NOT

Bool'dan oluşan çok boyutlu Array

Dizide Bool'dan oluşan çok boyutlu bir Array veri tipi söz konusu ise, sayma sırasında boyutların tüm Bit'leri dikkate alınır.

Örnek 1: Bool'dan oluşan bir Array [1..10,0..4,1..2], toplam 100 Bit'e ayrılabilen 10x5x2 Bit'e sahiptir.

Örnek 2: Bool'dan oluşan bir Array [0..399], ayrılabilen 400 Bit'e sahiptir.

NOT

Hedef Array'in alt Array sınırı "0" değilse dikkate etmeniz gerekenler:

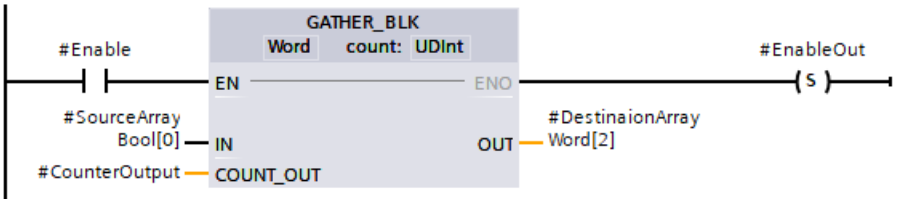
Performansla ilgili nedenlerden dolayı indeks her zaman bir Byte, Word veya DWord sınırından başlamalıdır. Bu, Array indeksinin hesaplamasının Array'in alt sınırından başlaması gerektiği anlamına gelir.

Örnek: Alt sınırı "0" olan bir kaynak Array

Blok arayüzünde aşağıdaki değişkenleri oluşturun:

Değişken	Aralık	Veri tipi
Enable	Giriş	Bool
SourceArrayBool		Bool'dan oluşan Array [0..95]
CounterOutput		UDInt
EnableOut	Çıkış	Bool
DestinationArrayWord		Word'den oluşan Array [0..5]

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:



Aşağıdaki tabloda, yordamın özel işlemci değerleriyle nasıl çalıştığı gösterilmektedir:

Parametre	İşlemci	Veri tipi
IN	SourceArrayBool[0]	"SourceArrayBool" işlemcisi, Bool'dan oluşan Array[0..95] veri tipine sahiptir. Bu, tekrar sözcükler halinde birleştirilebilen 96 Bool elemanı sağladığı anlamına gelir.
COUNT_OUT	CounterOutput = 3	3 şeklindeki bir CounterInput değeri, üç sözcüğün veya 48 Bit'in ayrılması gerektiği anlamına gelir. Bu nedenle hedef Array'de en az 48 Bit mevcut olmalıdır.
OUT	DestinationArrayWord[2]	"DestinationArrayWord" işlemcisi, Word'den oluşan Array[0..5] veri tipine sahiptir. Bu, altı Word elemanının mevcut olduğu anlamına gelir. <Bit dizisinden> oluşan ARRAY veri tipi, birleştirilmesi gereken Bit'leri kaydetmek için yeterli sayıda elemana sahip olmalıdır. Ancak hedef Array daha büyük de olabilir.

#Enable işlemcisi EN serbest bırakma girişinde "1" sinyal durumunu döndürürse yordam yürütülür. #SourceArrayBool işlemcisinin ilk elemanından itibaren 48 Bit, #DestinationArrayWord işlemcisi halinde birleştirilir. Hedef Array'deki başlangıç noktası üçüncü elemandır. Bu, ilk 16 Bit'in, hedef Array'de üçüncü sözcüğe, ikinci 16 Bit'in dördüncü sözcüğe ve üçüncü 16 Bit'in beşinci sözcüğe yazılacağı anlamına gelir. Yordam yürütülürken

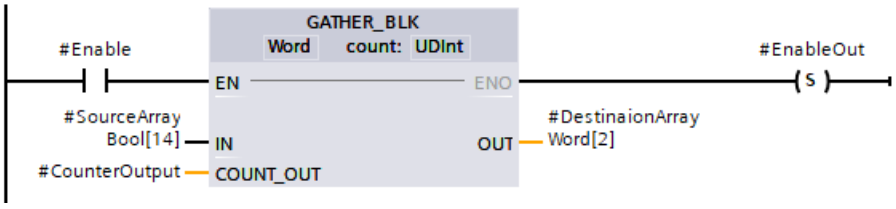
bir hata oluşursa, #EnableOut işlemcisi, ENO serbest bırakma çıkışında "0" sinyal durumunu geri döndürür.

Örnek: Alt sınırı "-2" olan bir kaynak Array

Bu örnekte, önceki örnek tekrarlanmakta, ancak hedef Array'in alt sınırı negatif bir değere ayarlanmakta ve #DestinationArrayWord[2]'de yayınlanmaktadır.
Blok arayüzü, SourceArrayBool'un boyutları istisna olmak kaydıyla önceki örnektekiyle aynıdır.

Değişken	Aralık	Veri tipi
SourceArrayBool	Giriş	Bool'dan oluşan Array [-2..93]

Aşağıdaki örnekte, yordamın nasıl çalıştığı gösterilmektedir:

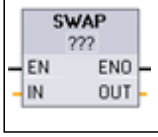


Parametre tablosu, IN parametresi için kullanılan haricinde önceki örnektekiyle aynıdır.

Parametre	İşlemci	Veri tipi
IN	SourceArrayBool[14]	"SourceArrayBool" işlemcisi, Bool'dan oluşan Array [-2..93] veri tipindedir. Başlangıç noktası 16. eleman olduğu için, tekrar sözcükler halinde birleştirilebilen sadece 80 Bool elemanı mevcuttur.

8.6.9 SWAP (sırayı değiştir)

Tablo 8-87 SWAP yordamı

LAD/FBD	SCL	Açıklama
	**out := SWAP(in);**	İki Bayt ve dört Bayt veri elemanlarında Bayt sırasını tersine çevirir. Bir Bayt'ın içindeki Bit dizisi değişmez. ENO, SWAP yordamı yürütüldükten sonra her zaman DOĞRU'dur.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-88 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	Word, DWord	IN parametresindeki normal şekilde sıralanan veri Bayt'ları
OUT	Word, DWord	OUT parametresindeki veri Bayt'larının değiştirilen sırası

Örnek 1	IN parametresi = MB0 (yürütmeden önce)		OUT parametresi = MB4, (yürütmeden sonra)	
Adres	MW0	MB1	MW4	MB5
W#16#1234	12	34	34	12
WORD	MSB	LSB	MSB	LSB

Örnek 2	IN parametresi = MB0 (yürütmeden önce)				OUT parametresi = MB4, (yürütmeden sonra)			
Adres	MD0	MB1	MB2	MB3	MD4	MB5	MB6	MB7
DW#16#12345678	12	34	56	78	78	56	34	12
DWORD	MSB			LSB	MSB			LSB

8.6.10 Belleği oku / belleğe yaz yordamları

8.6.10.1 PEEK ve POKE (sadece SCL)

SCL, veri bloklarından, G/Ç'lerden veya bellekten okuma veya bu bileşenlere yazma işlemlerini gerçekleştirmenizi sağlayan PEEK ve POKE yordamlarını sunar. İşlem için özel bir Bayt veya Bit ofsetiyle parametreleri belirtebilirsiniz.

NOT

PEEK ve POKE yordamlarını veri bloklarıyla birlikte kullanmak için, standart veri blokları (optimize edilmiş veri blokları değil) kullanmanız gerekir. Ayrıca PEEK ve POKE yordamlarının sadece verileri aktardığına dikkat edin. Adreslerdeki veri tiplerine dikkat edilmez.

```
PEEK(area:=_in_,
     dbNumber:=_in_,
     byteOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından byteOffset tarafından belirtilen Bayt'ı okur.

Bir veri bloğuna verilen referans örneği:

```
%MB100 := PEEK(area:=16#84,
dbNumber:=1, byteOffset:=#i);
```

EB3 girişine verilen referans örneği:

```
%MB100 := PEEK(area:=16#81,
dbNumber:=0, byteOffset:=#i); // when
#i = 3
```

```
PEEK_WORD(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından byteOffset tarafından belirtilen sözcüğü okur.

Örnek:

```
%MW200 := PEEK_WORD(area:=16#84,
dbNumber:=1, byteOffset:=#i);
```

```
PEEK_DWORD(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından byteOffset tarafından belirtilen çift sözcüğü okur.

Örnek:

```
%MD300 := PEEK_DWORD(area:=16#84,
dbNumber:=1, byteOffset:=#i);
```

```
PEEK_BOOL(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          bitOffset:=_in_);
```

Belirtilen veri bloğundan, G/Ç'den veya bellek alanından bitOffset ve byteOffset tarafından belirtilen Boole değerini okur.

Örnek:

```
%MB100.0 := PEEK_BOOL(area:=16#84,
dbNumber:=1, byteOffset:=#ii,
bitOffset:=#j);
```

```
POKE(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          value:=_in_);
```

Değeri (Bayt, sözcük veya çift sözcük) belirtilen veri bloğunun, G/Ç'nin veya bellek alanının belirtilen byteOffset'ine yazar.

Bir veri bloğuna verilen referans örneği:

```
POKE(area:=16#84, dbNumber:=2,
byteOffset:=3, value:="Tag_1");
```

AB3 çıkışına verilen referans örneği:

```
POKE(area:=16#82, dbNumber:=0,
byteOffset:=3, value:="Tag_1");
```

```
POKE_BOOL(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          bitOffset:=_in_,
          value:=_in_);
```

Boole değerini, belirtilen veri bloğunun, G/Ç'nin veya bellek alanının belirtilen bitOffset'ine ve byteOffset'ine yazar.

Örnek:

```
POKE_BOOL(area:=16#84, dbNumber:=2,
byteOffset:=3, bitOffset:=5,
value:=0);
```

```
POKE_BLK(area_src:=_in_,
          dbNumber_src:=_in_,
          byteOffset_src:=_in_,
          area_dest:=_in_,
          dbNumber_dest:=_in_,
          byteOffset_dest:=_in_,
          count:=_in_);
```

"count" ifadesinin altında belirtilen sayıda Bayt'ı; belirtilen kaynak veri bloğunun, kaynak G/Ç'nin veya kaynak bellek alanının belirtilen byteOffset'i ile başlayarak, belirtilen hedef veri bloğunun, hedef G/Ç'nin veya hedef bellek alanının belirtilen byteOffset'ine yazar.

Örnek:

```
POKE_BLK(area_src:=16#84,
dbNumber_src:=#src_db,
byteOffset_src:=#src_byte,
area_dest:=16#84,
dbNumber_dest:=#src_db,
byteOffset_dest:=#src_byte,
count:=10);
```

PEEK ve POKE yordamlarında "area", "area_src" ve "area_dest" parametreleri için aşağıdaki değerler geçerlidir. Veri bloğu olarak diğer alanlar için dbNumber parametresi 0 olmalıdır.

16#81	I
16#82	Q
16#83	M
16#84	DB

8.6.10.2 Big ve Little Endian formatını oku ve yaz (SCL) yordamları

S7-1200 CPU'su, Big ve Little Endian formatlarını okumak ve yazmak için kullanılan SCL yordamlarını sağlar. Little Endian formatı, en düşük değere sahip Bit'in bulunduğu Bayt'ın en küçük bellek adresine kaydedileceği anlamına gelir. Big Endian formatı, en yüksek değere sahip Bit'in bulunduğu Bayt'ın en küçük bellek adresine kaydedileceği anlamına gelir.

Little ve Big Endian formatındaki verileri okumak ve yazmak için kullanılan dört SCL yordamı aşağıda açıklanmaktadır:

- READ_LITTLE (verileri Little Endian formatında oku)
- WRITE_LITTLE (verileri Little Endian formatında yaz)
- READ_BIG (verileri Big Endian formatında oku)
- WRITE_BIG (verileri Big Endian formatında yaz)

Tablo 8-89 Big ve Little Endian formatını oku ve yaz yordamları

LAD/FBD	SCL	Açıklama
Mevcut değil	**READ_LITTLE(** **src_array:=_variant_in_,** **dest_Variable =>_out_,** **pos:=_dint_inout)**	Verileri bir bellek alanından okur ve Little Endian Byte formatında münferit bir değişkene yazar.
Mevcut değil	**WRITE_LITTLE(** **src_variable:=_in_,** **dest_array =>_variant_inout_,** **pos:=_dint_inout)**	Verileri münferit bir değişkenden Little Endian Byte formatında bir bellek alanına yazar.
Mevcut değil	**READ_BIG(** **src_array:=_variant_in_,** **dest_Variable =>_out_,** **pos:=_dint_inout)**	Verileri bir bellek alanından okur ve Big Endian Byte formatında münferit bir değişkene yazar.
Mevcut değil	**WRITE_BIG(** **src_variable:=_in_,** **dest_array =>_variant_inout_,** **pos:=_dint_inout)**	Verileri münferit bir değişkenden Big Endian Byte formatında bir bellek alanına yazar.

Tablo 8-90 READ_LITTLE and READ_BIG yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
src_array	Array of Byte	Verilerin okunması gereken bellek alanı
dest_Variable	Bit karakter dizileri, tamsayılar, kayar noktalı sayılar, zamanlar, tarih ve saat, karakter dizileri	Verilerin yazılması gereken hedef değişken
pos	DINT	src_array girişinin okunmaya başlanacağı sıfır tabanlı pozisyon.

Tablo 8-91 WRITE_LITTLE and WRITE_BIG yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
src_variable	Bit karakter dizileri, tamsayılar, kayar noktalı sayılar, TOD, DATA, Char, WChar	Değişkenden gelen kaynak veriler
dest_array	Array of Byte	Verilerin yazılması gereken bellek alanı
pos	DINT	Verilerin dest_array çıkışına yazılmaya başlanacağı sıfır tabanlı pozisyon.

Tablo 8-92 RET_VAL parametresi

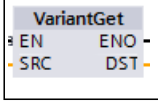
RET_VAL* (W#16#...)	Açıklama
0000	Hata yok
80B4	SRC_ARRAY veya DEST_ARRAY, Array of Byte değildir
8382	POS parametresinin değeri, alan için tanımlanan limitin dışında.
8383	POS parametresinin değeri, alanın limiti içinde; fakat bellek alanının boyutu, alanın limitini aşıyor.
*Hata kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir.	

8.6.11 Variant yordamları

8.6.11.1 VariantGet (VARIANT değişken değerini oku)

"Variant değişken değerini oku" yordamıyla, Variant işaretçisinin SRC parametresinde gösterdiği değişkenlerin değeri okunabilir ve DST parametresinin değişkenine yazılabilir.
SRC parametresi, Variant veri tipine sahiptir. DST parametresi için, Variant dışında her veri tipi programlanabilir.
DST parametresindeki değişkenlerin veri tipi, Variant'ın gösterdiği değişkenlerin veri tipiyle aynı olmalıdır.

Tablo 8-93 VariantGet yordamı

LAD/FBD	SCL	Açıklama
	**VariantGet(SRC:=_variant_in_, DST=>_variant_out_);**	SRC parametresinin gösterdiği değişkeni okur ve DST parametresindeki değişkene yazar.

NOT

Yapıları ve alanları kopyalamak için "MOVE_BLK_VARIANT: Aralığı kopyala" yordamı kullanılabilir.

Tablo 8-94 VariantGet yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
SRC	Variant	İşaretçi, kaynak verilerin üzerinde
DST	Bit karakter dizileri, tamsayılar, kayar noktalı sayılar, zamanlar, tarih ve saat, karakter dizileri, ARRAY elemanları, PLC veri tipleri	Verilerin yazılması gereken hedef

Tablo 8-95 ENO durumu

ENO	Koşul	Sonuç
1	Hata yok	Yordam, SRC'nin gösterdiği değişken verilerini, DST değişkenine kopyaladı.
0	EN serbest bırakma girişi "0" sinyal durumuna sahiptir veya veri tipleri uyuşmuyordur.	Yordam, hiçbir veriyi kopyalamadı.

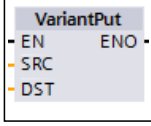
8.6.11.2 VariantPut (VARIANT değişken değerini yaz)

"VARIANT değişken değerini yaz" yordamıyla, SRC parametresindeki değişkenin değeri, VARIANT'ın gösterdiği, DST parametresindeki değişkene yazılabilir.

DST parametresi VARIANT veri tipine sahiptir. SRC parametresi için VARIANT dışında her veri tipi programlanabilir.

SRC parametresindeki değişkenlerin veri tipi, VARIANT'ın gösterdiği veri tipiyle aynı olmalıdır.

Tablo 8-96 VariantPut yordamı

LAD/FBD	SCL	Açıklama
	**VariantPut(** **SRC:=_variant_in_,** **DST=>_variant_in_);**	SRC parametresiyle referanslanan değişkeni, DST parametresinin gösterdiği Variant elemanına yazar.

NOT

Yapıları ve alanları kopyalamak için "MOVE_BLK_VARIANT: Aralığı kopyala" yordamı kullanılabilir.

Tablo 8-97 VariantPut yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
SRC	Bit strings, integers, floating-point numbers, timers, date and time, character strings, ARRAY elements, PLC data types	İşaretçi, kaynak verilerin üzerinde
DST	Variant	Verilerin yazılması gereken hedef

Tablo 8-98 ENO'nun durumu

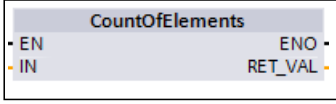
ENO	Koşul	Sonuç
1	Hata yok	Yordam, SRC değişken verilerini, DST değişkenine kopyaladı.
0	EN serbest bırakma girişi "0" sinyal durumuna sahiptir veya veri tipleri uyuşmuyordur.	Yordam, hiçbir veriyi kopyalamadı.

8.6.11.3 CountOfElements (ARRAY elemanlarının sayısını sorgula)

"ARRAY elemanlarının sayısını sorgula" yordamıyla, bir Variant elemanının gösterdiği bir değişkende kaç Array elemanının mevcut olduğu sorgulanabilir.

Yordam, tek boyutlu bir ARRAY'de üst ve alt limit +1 arasındaki farkı verir. Yordam, çok boyutlu bir ARRAY'de tüm boyutların çarpımını geri döndürür.

Tablo 8-99 CountOfElements yordamı

LAD/FBD	SCL	Açıklama
	**Result := CountOfElements(_variant_in_);**	IN parametresinin gösterdiği dizideki Array elemanlarını sayar.

NOT

Variant bir Array of Bool'u gösteriyorsa, yordam, bir sonraki Bayt sınırına kadar doldurma elemanlarını sayar. Array[0..1] of Bool için örneğin 8 değeri döndürülür.

Tablo 8-100 CountOfElements yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
IN	Variant	Sayılması gereken Array elemanlarını içeren değişken
RET_VAL	UDint	Yordamın sonucu

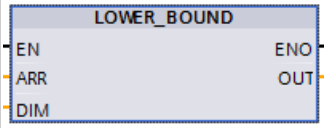
Tablo 8-101 ENO'nun durumu

ENO	Koşul	Sonuç
1	Hata yok	Yordam, Array elemanlarının sayısını geri döndürür.
0	EN serbest bırakma girişi "0" sinyal durumuna sahip veya Variant bir Array elemanını göstermiyor	Yordam, 0 değerini geri döndürür.

8.6.12 Array yordamları

8.6.12.1 LOWER_BOUND: (Alt ARRAY sınırını oku)

Tablo 8-102 LOWER_BOUND yordamı

LAD/FBD	SCL	Açıklama
	**out := LOWER_BOUND(ARR:=_variant_in_, DIM:=_udint_in_);**	Blok arayüzünde ARRAY[*] içeren değişkenleri deklare edebilirsiniz. Bu yerel değişkenler için ARRAY'in limitlerini okuyabilirsiniz. DIM parametresinde gereken boyut değerini belirtmeniz gerekir. LOWER_BOUND (Alt ARRAY sınırını oku) yordamıyla, ARRAY'in alt değişken limitini okuyabilirsiniz.

Parametre

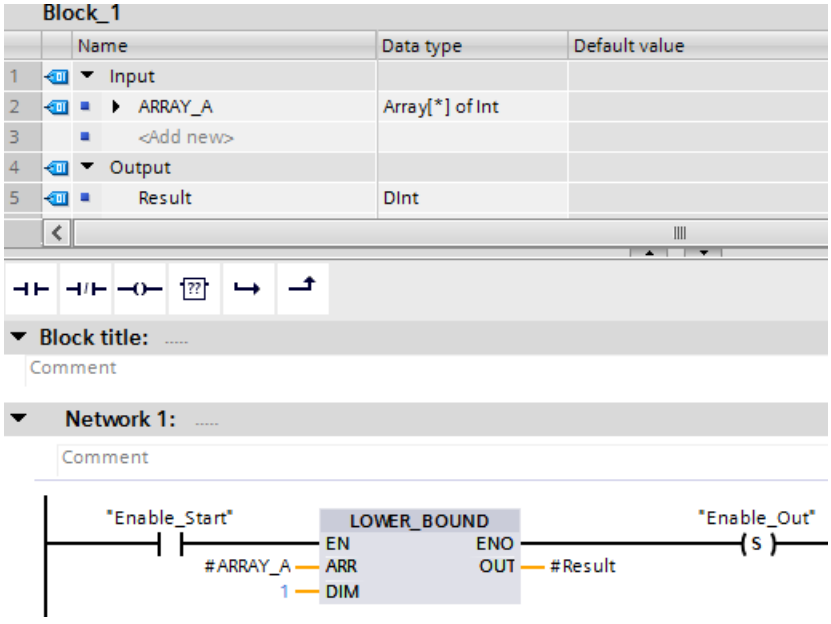
Aşağıdaki tabloda, LOWER_BOUND yordamının parametreleri gösterilmektedir: Alt ARRAY sınırını oku

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
EN	Input	BOOL	E, A, M, D, L	Serbest bırakma girişi
ENO	Output	BOOL	E, A, M, D, L	ENO serbest bırakma çıkışı, aşağıdaki koşullardan biri sağlandığı takdirde 0 sinyal durumuna sahip olur: • EN serbest bırakma girişi 0 sinyal durumuna sahiptir. • DIM girişinde belirtilen boyut mevcut değildir.
ARR	Input	ARRAY [*]	FB: InOut FC aralığı: Input ve InOut aralıkları	Alt değişken sınırı okunması gereken ARRAY.
DIM	Input	UDINT	E, A, M, D, L veya sabit	Alt değişken sınırı okunması gereken ARRAY'in boyutu.
OUT	Output	DINT	E, A, M, D, L	Sonuç

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)":

Örnek

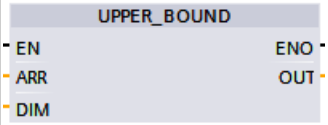
Fonksiyonun (FC) blok arayüzünde ARRAY_A giriş parametresi, değişken boyutlarını içeren tek boyutlu bir Array'dir.



"Enable_Start" işlemcisi 1 sinyal durumunu verirse, CPU, LOWER_BOUND yordamını yürütür. CPU, #ARRAY_A ARRAY'in alt değişken sınırını tek boyutlu Array'den okur. Yordam yürütülürken hatalar oluşursa, "Enable_Out" işlemcisi ayarlanır ve "Result" işlemcisi Array'in alt sınırına ayarlanır.

8.6.12.2 UPPER_BOUND: (Üst ARRAY sınırını oku)

Tablo 8-103 LOWER_BOUND yordamı

LAD/FBD	SCL	Açıklama
	**out := UPPER_BOUND(ARR:=_variant_in_, DIM:=_udint_in_);**	Blok arayüzünde ARRAY[*] içeren değişkenleri deklare edebilirsiniz. Bu yerel değişkenler için ARRAY'in limitlerini okuyabilirsiniz. DIM parametresinde gereken boyut değerini belirtmeniz gerekir. UPPER_BOUND (Üst ARRAY sınırını oku) yordamıyla, ARRAY'in üst değişken limitini okuyabilirsiniz.

Parametre

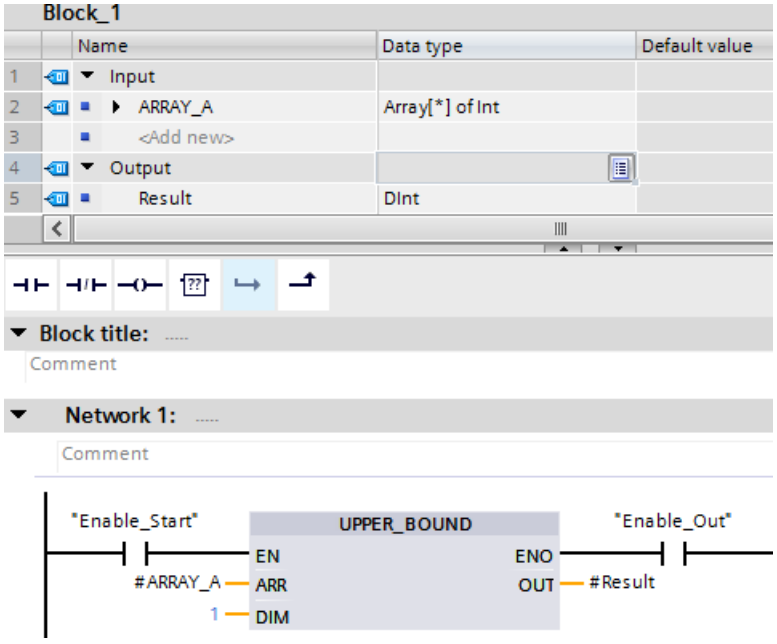
Aşağıdaki tabloda, UPPER_BOUND yordamının parametreleri gösterilmektedir: (Üst ARRAY sınırını oku):

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
EN	Input	BOOL	E, A, M, D, L	Serbest bırakma girişi
ENO	Output	BOOL	E, A, M, D, L	Serbest bırakma çıkışı
ARR	Input	ARRAY [*]	FB: InOut FC aralığı: Input ve InOut aralıkları	Üst değişken sınırı okunması gereken ARRAY.
DIM	Input	UDINT	E, A, M, D, L veya sabit	Üst değişken sınırı okunması gereken ARRAY'in boyutu.
OUT	Output	DINT	E, A, M, D, L	Sonuç

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)":

Örnek

Fonksiyonun (FC) blok arayüzünde ARRAY_A giriş parametresi, değişken boyutlarını içeren tek boyutlu bir Array'dir.

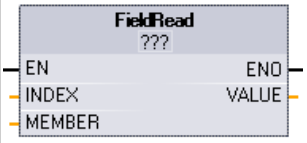


"Enable_Start" işlemcisi 1 sinyal durumunu verirse, CPU, yordamı yürütür. CPU, #ARRAY_A ARRAY'in üst değişken sınırını tek boyutlu Array'den okur. Yordam yürütülürken hatalar oluşursa, "Enable_Out" işlemcisi ve "Result" işlemcisi ayarlanır.

8.6.13 Daha eski sistemlerdeki yordamlar

8.6.13.1 FieldRead (Alanı oku) ve FieldWrite (Alana yaz) yordamları

Tablo 8-104 FieldRead ve FieldWrite yordamları

LAD/FBD	SCL	Açıklama
	**value := member[index];**	FieldRead, INDEX indeks değerine sahip Array elemanını, ilk elemanı MEMBER parametresi tarafından belirtilen Array'den okur. Array elemanının değeri, VALUE parametresi tarafından belirtilen adrese aktarılır.
	**member[index] := value;**	FieldWrite, VALUE parametresi tarafından belirtilen adresteki değeri, ilk elemanı MEMBER parametresi tarafından belirtilen Array'e aktarır. Değer, Array indeksi INDEX parametresi tarafından belirtilen Array elemanına aktarılır.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-105 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
Index	Giriş	DInt	Okunması veya kendisine yazılması gereken Array elemanının indeks numarası.
Member [1]	Giriş	Bir ARRAY değişkeninin bileşenleri olarak ikili sayılar, tamsayılar, kayar noktalı sayılar, zamanlar, DATE, TOD, CHAR ve WCHAR	Global bir veri bloğunda veya bir blok arayüzünde tanımlanan tek boyutlu bir Array'deki ilk elemanın adresi. Örnek: Array indeksi [-2..4] olarak belirtildiyse, ilk elemanın indeksi -2'dir ve 0 değildir.
Value [1]	Çıkış	İkili sayılar, tamsayılar, kayar noktalı sayılar, zamanlar, DATE, TOD, CHAR, WCHAR	Belirtilen Array elemanının kopyalanacağı adres (FieldRead). Belirtilen Array elemanına kopyalanacak değerin adresi (FieldWrite).

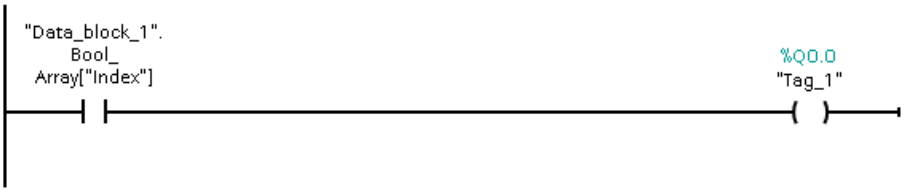
[1] MEMBER parametresi tarafından belirtilen Array elemanının veri tipi ile VALUE parametresinin veri tipi aynı olmalıdır.

ENO serbest bırakma çıkışı, aşağıdaki koşullardan biri sağlandığı takdirde 0'a eşit olur:

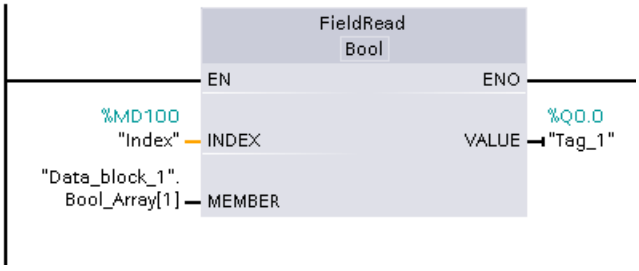
- EN girişi, 0 sinyal durumunda.
- INDEX parametresi tarafından belirtilen Array elemanı, MEMBER parametresinde belirtilen Array'de tanımlanmamıştır.
- İşleme sırasında, örn. taşma gibi hatalar ortaya çıkmaktadır.

Örnek: Array indekslemesi üzerinden verilere erişim

Bir Array'in elemanlarına bir değişkenle erişmek için, değişkeni, program lojiğinizde Array indeksi olarak kullanmanız yeterlidir. Örneğin aşağıdaki ağda, Boole değerlerinden oluşan bir Array'in Boole değeri temel alınarak, "Veri bloğu_1" öğesinde, "Index" PLC değişkeni tarafından belirtilen bir çıkış ayarlanır.



Değişken Array indeksini içeren lojik, FieldRead yordamına karşılık gelir:



FieldRead ve FieldWrite yordamları, indeksleme lojiği aracılığıyla değişken Array'leriyle değiştirilebilir.

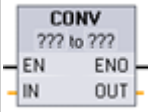
SCL'de FieldRead veya FieldWrite yordamları yoktur, fakat bir değişkenle bir Array'in dolaylı şekilde adreslenmesi desteklenir:

```
#Tag_1 := "Data_block_1".Bool_Array[#Index];
```

8.7 Dönüştürme işlemleri

8.7.1 CONV (değeri dönüştür)

Tablo 8-106 Dönüştür (CONV) yordamı

LAD/FBD	SCL	Açıklama
	**out := <data type in>_TO_<data type out>(in);**	Bir veri tipindeki bir veri elemanını başka bir veri tipine dönüştürür.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tiplerini seçin.

[2] SCL'de: Dönüştürme yordamını, giriş parametresi (in) ve çıkış parametresi (out) için kullanılan veri tipini girerek oluşturun. Örnek: DWORD_TO_REAL, bir çift sözcüğü bir gerçek sayı değerine dönüştürür.

Tablo 8-107 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	Bit karakter dizisi[1], SInt, USInt, Int, UInt, DInt, UDInt, Real, LReal, BCD16, BCD32, Char, WChar	Giriş değeri
OUT	Bit karakter dizisi[1], SInt, USInt, Int, UInt, DInt, UDInt, Real, LReal, BCD16, BCD32, Char, WChar	Yeni bir veri tipine dönüştürülen giriş değeri

[1] Yordam için hiçbir Bit karakter dizisi (Bayt, Word, DWord) seçilemez. Yordamın bir parametresi için Bayt, Word veya DWord veri tipinde bir işlemci girmek için, aynı Bit uzunluğuna sahip ön işaretsiz bir tamsayı seçin. Örnek: Bir Bayt için USInt'yi, bir Word için UInt'yi ve DWord için UDInt'yi seçebilirsiniz.

Kaynak veri tipi (Şundan dönüştür) seçildikten sonra, açılır listede bir dizi olası dönüştürme seçeneği (Şuna dönüştür) görüntülenir. BCD16'dan ve bu elemana dönüştürme işlemleri, Int veri tipiyle sınırlıdır. BCD32'den ve bu elemana dönüştürme işlemleri, DInt veri tipiyle sınırlıdır.

Tablo 8-108 ENO'nun durumu

ENO	Açıklama	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	IN; +/- INF veya +/- NaN'dir	+/- INF veya +/- NaN
0	Sonuç, OUT veri tipi için tanımlanan geçerli aralığı aşıyor	OUT, IN 'in değerine ayarlanır

8.7.2 SCL'deki dönüştürme yordamları

Bu sözdizimine göre bir veri tipinden başka bir veri tipine dönüştürmek için kullanılan SCL yordamları:
out := <data type in>_TO_<data type out>(in);

SCL'deki dönüştürme yordamları

Tablo 8-109 Bool, Byte, Word veya DWord'ün dönüştürülmesi

Veri tipi	Yordam	Sonuç
Bool	**BOOL_TO_BYTE, BOOL_TO_WORD, BOOL_TO_DWORD, BOOL_TO_INT, BOOL_TO_DINT**	Değer, hedef veri tipinin en düşük Bit'ine aktarılır.
Byte	**BYTE_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	BYTE_TO_WORD, BYTE_TO_DWORD	Değer, hedef veri tipinin en düşük Bayt'ına aktarılır.
	BYTE_TO_SINT, BYTE_TO_USINT	Değer, hedef veri tipine aktarılır.
	BYTE_TO_INT, BYTE_TO_UINT, BYTE_TO_DINT, BYTE_TO_UDINT	Değer, hedef veri tipinin en düşük Bayt'ına aktarılır.
Word	**WORD_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	WORD_TO_BYTE	Kaynak değerin en düşük Bayt'ı, hedef veri tipine aktarılır.
	WORD_TO_DWORD	Değer, hedef veri tipinin en düşük sözcüğüne aktarılır.
	WORD_TO_SINT, WORD_TO_USINT	Kaynak değerin en düşük Bayt'ı, hedef veri tipine aktarılır.
	WORD_TO_INT, WORD_TO_UINT	Değer, hedef veri tipine aktarılır.
	WORD_TO_DINT, WORD_TO_UDINT	Değer, hedef veri tipinin en düşük sözcüğüne aktarılır.
DWord	**DWORD_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	DWORD_TO_BYTE, DWORD_TO_WORD, DWORD_TO_SINT	Kaynak değerin en düşük Bayt'ı, hedef veri tipine aktarılır.
	DWORD_TO_USINT, DWORD_TO_INT, DWORD_TO_UINT	Kaynak değerin en düşük sözcüğü, hedef veri tipine aktarılır.
	DWORD_TO_DINT, DWORD_TO_UDINT, DWORD_TO_REAL	Değer, hedef veri tipine aktarılır.

Tablo 8-110 Kısa bir tamsayının (SInt veya USInt) dönüştürülmesi

Veri tipi	Yordam	Sonuç
SInt	**SINT_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	SINT_TO_BYTE	Değer, hedef veri tipine aktarılır.
	SINT_TO_WORD, SINT_TO_DWORD	Değer, hedef veri tipinin en düşük Bayt'ına aktarılır.
	SINT_TO_INT, SINT_TO_DINT, SINT_TO_USINT, SINT_TO_UINT, SINT_TO_UDINT, SINT_TO_REAL, SINT_TO_LREAL, SINT_TO_CHAR, SINT_TO_STRING	Değer dönüştürülür.
USInt	**USINT_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.

Veri tipi	Yordam	Sonuç
USInt	**USINT_TO_BYTE**	Değer, hedef veri tipine aktarılır.
	USINT_TO_WORD, USINT_TO_DWORD, USINT_TO_INT, USINT_TO_UINT, USINT_TO_DINT, USINT_TO_UDINT	Değer, hedef veri tipinin en düşük Bayt'ına aktarılır.
	USINT_TO_SINT, USINT_TO_REAL, USINT_TO_LREAL, USINT_TO_CHAR, USINT_TO_STRING	Değer dönüştürülür.

Tablo 8-111 Bir tamsayının (Int veya UInt) dönüştürülmesi

Veri tipi	Yordam	Sonuç
Int	**INT_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	INT_TO_BYTE, INT_TO_DWORD, INT_TO_SINT, INT_TO_USINT, INT_TO_UINT, INT_TO_UDINT, INT_TO_REAL, INT_TO_LREAL, INT_TO_CHAR, INT_TO_STRING	Değer dönüştürülür.
	INT_TO_WORD	Değer, hedef veri tipine aktarılır.
	INT_TO_DINT	Değer, hedef veri tipinin en düşük Bayt'ına aktarılır.
UInt	**UINT_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	UINT_TO_BYTE, UINT_TO_SINT, UINT_TO_USINT, UINT_TO_INT, UINT_TO_REAL, UINT_TO_LREAL, UINT_TO_CHAR, UINT_TO_STRING	Değer dönüştürülür.
	UINT_TO_WORD, UINT_TO_DATE	Değer, hedef veri tipine aktarılır.
	UINT_TO_DWORD, UINT_TO_DINT, UINT_TO_UDINT	Değer, hedef veri tipinin en düşük Bayt'ına aktarılır.

Tablo 8-112 Bir çift tamsayının (Dint veya UDInt) dönüştürülmesi

Veri tipi	Yordam	Sonuç
DInt	**DINT_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	DINT_TO_BYTE, DINT_TO_WORD, DINT_TO_SINT, DINT_TO_USINT, DINT_TO_INT, DINT_TO_UINT, DINT_TO_UDINT, DINT_TO_REAL, DINT_TO_LREAL, DINT_TO_CHAR, DINT_TO_STRING	Değer dönüştürülür.
	DINT_TO_DWORD, DINT_TO_TIME	Değer, hedef veri tipine aktarılır.
UDInt	**UDINT_TO_BOOL**	En düşük Bit, hedef veri tipine aktarılır.
	UDINT_TO_BYTE, UDINT_TO_WORD, UDINT_TO_SINT, UDINT_TO_USINT, UDINT_TO_INT, UDINT_TO_UINT, UDINT_TO_DINT, UDINT_TO_REAL, UDINT_TO_LREAL, UDINT_TO_CHAR, UDINT_TO_STRING	Değer dönüştürülür.
	UDINT_TO_DWORD, UDINT_TO_TOD	Değer, hedef veri tipine aktarılır.

Tablo 8-113 Bir gerçek sayının (Real veya LReal) dönüştürülmesi

Veri tipi	Yordam	Sonuç
Real	**REAL_TO_DWORD, REAL_TO_LREAL**	Değer, hedef veri tipine aktarılır.
	REAL_TO_SINT, REAL_TO_USINT, REAL_TO_INT, REAL_TO_UINT, REAL_TO_DINT, REAL_TO_UDINT, REAL_TO_STRING	Değer dönüştürülür.
LReal	**LREAL_TO_SINT, LREAL_TO_USINT, LREAL_TO_INT, LREAL_TO_UINT, LREAL_TO_DINT, LREAL_TO_UDINT, LREAL_TO_REAL, LREAL_TO_STRING**	Değer dönüştürülür.

Tablo 8-114 Time, DTL, TOD veya Date'in dönüştürülmesi

Veri tipi	Yordam	Sonuç
Time	**TIME_TO_DINT**	Değer, hedef veri tipine aktarılır.
DTL	**DTL_TO_DATE, DTL_TO_TOD**	Değer dönüştürülür.
TOD	**TOD_TO_UDINT**	Değer dönüştürülür.
Date	**DATE_TO_UINT**	Değer dönüştürülür.

Tablo 8-115 Char veya String'in dönüştürülmesi

Veri tipi	Yordam	Sonuç
Char	**CHAR_TO_SINT, CHAR_TO_USINT, CHAR_TO_INT, CHAR_TO_UINT, CHAR_TO_DINT, CHAR_TO_UDINT**	Değer dönüştürülür.
	CHAR_TO_STRING	Değer, karakter dizisinin ilk karakterine aktarılır.
String	**STRING_TO_SINT, STRING_TO_USINT, STRING_TO_INT, STRING_TO_UINT, STRING_TO_DINT, STRING_TO_UDINT, STRING_TO_REAL, STRING_TO_LREAL**	Değer dönüştürülür.
	STRING_TO_CHAR	Karakter dizisinin ilk karakteri Char'a kopyalanır.

8.7.3 ROUND (sayıyı yuvarla) ve TRUNC (tamsayı oluştur)

Tablo 8-116 ROUND ve TRUNC yordamları

LAD/FBD	SCL	Açıklama
	**out := ROUND (in);**	bir gerçek sayıyı bir tamsayıya dönüştürür. SCL'de ROUND yordamının çıkışı için kullanılan veri tipi, standart olarak DINT'dir. Başka bir çıkış veri tipine yuvarlama işlemi için yordamın adı, veri tipinin açık adıyla birlikte girilmelidir, örneğin ROUND_REAL veya ROUND_LREAL. Gerçek sayının ondalık basamakları, bir sonraki tamsayı değerine yuvarlanır (IEEE - yuvarla). Sayı, iki tamsayının arasındaki aralığın tam olarak yarısıysa (örneğin 10,5), sayı, çift tam sayıya yuvarlanır. Örnek: • ROUND (10.5) = 10 • ROUND (11.5) = 12
	**out := TRUNC(in);**	TRUNC, bir gerçek sayıyı bir tamsayıya dönüştürür. Gerçek sayının ondalık basamakları kısaltılarak sıfıra ayarlanır (IEEE - sıfıra yuvarla).

[1] LAD ve FBD'de: (Yordam adlarında) "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-117 Parametreler için kullanılan veri tipleri

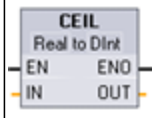
Parametre	Veri tipi	Açıklama
IN	Real, LReal	Kayar noktalı sayı girişi
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Yuvarlanmış veya tamsayı çıkışı

Tablo 8-118 ENO'nun durumu

ENO	Açıklama	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	IN, +/- INF veya +/- NaN'dir	+/- INF veya +/- NaN

8.7.4 CEIL ve FLOOR (kayar noktalı sayıdan bir sonraki daha yüksek veya bir sonraki daha düşük tamsayıyı oluştur)

Tablo 8-119 CEIL ve FLOOR yordamları

LAD/FBD	SCL	Açıklama
	**out := CEIL(in);**	Bir gerçek sayıyı (Real veya LReal), seçilen gerçek sayıdan daha büyük veya ona eşit bir sonraki tamsayıya dönüştürür (IEEE - +sonsuza yuvarla).
	**out := FLOOR(in);**	Bir gerçek sayıyı (Real veya LReal), seçilen gerçek sayıdan daha küçük veya ona eşit bir sonraki tamsayıya dönüştürür (IEEE - -sonsuza yuvarla).

[1] LAD ve FBD'de: (Yordam adlarında) "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-120 Parametreler için kullanılan veri tipleri

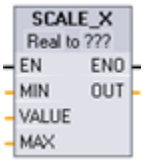
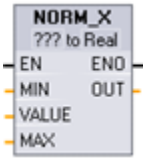
Parametre	Veri tipi	Açıklama
IN	Real, LReal	Kayar noktalı sayı girişi
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Dönüştürülen çıkış

Tablo 8-121 ENO'nun durumu

ENO	Açıklama	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	IN, +/- INF veya +/- NaN'dir	+/- INF veya +/- NaN

8.7.5 SCALE_X (ölçekle) ve NORM_X (standartlaştır)

Tablo 8-122 SCALE_X ve NORM_X yordamları

LAD/FBD	SCL	Açıklama
	**out :=SCALE_X(min:=_in_, value:=_in_, max:=_in_);**	Standartlaştırılmış VALUE gerçek parametresini (0,0 <= VALUE <= 1,0), MIN ve MAX parametreleriyle önceden belirtilen veri tipine ve değer aralığına göre ölçekler: OUT = VALUE (MAX - MIN) + MIN
	**out :=NORM_X(min:=_in_, value:=_in_, max:=_in_);**	VALUE parametresini, MIN ve MAX parametreleriyle belirtilen değer aralığının içinde standartlaştırır: OUT = (VALUE - MIN) / (MAX - MIN), burada (0,0 <= OUT <= 1,0)

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-123 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
MIN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Aralığın minimum değeri girişi
VALUE	SCALE_X: Real, LReal NORM_X: SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Ölçekleme veya standartlaştırma için kullanılan giriş değeri
MAX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Aralığın maksimum değeri girişi
OUT	SCALE_X: SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal NORM_X: Real, LReal	Ölçeklenmiş veya standartlaştırılmış çıkış değeri

[1] SCALE_X'te: MIN, MAX ve OUT parametreleri aynı veri tipine sahip olmalıdır.

NOT

SCALE_X'in VALUE parametresi, (0,0 <= VALUE <= 1,0) aralığında olmalıdır.

VALUE parametresi 0,0'dan küçükse veya 1,0 büyükse:

- Doğrusal ölçekleme işlemi, OUT veri tipinin değer aralığı içinde yer alan OUT değerleri söz konusu olduğu sürece, MIN parametresinin değerinden daha küçük veya MAX parametresinin değerinden daha büyük OUT çıkış değerleri oluşturabilir. SCALE_X yürütülürken, bu durumlarda ENO = DOĞRU olarak ayarlanır.
- OUT'un veri tipi aralığında olmayan ölçeklenmiş sayılar oluşturmak mümkündür. Bu durumda, OUT parametresinin değeri, OUT veri tipine nihai dönüştürme işleminden önce, ölçeklenmiş gerçek sayının en düşük değerli kısmına karşılık gelen bir ara değere ayarlanır. SCALE_X yürütülürken, bu durumlarda ENO = YANLIŞ olarak ayarlanır.

NORM_X'in VALUE parametresi, (MIN <= VALUE <= MAX) aralığında olmalıdır.

VALUE parametresi MIN değerinden daha küçükse veya MAX değerinden daha büyükse, doğrusal ölçekleme yordamı, 0,0'dan daha küçük veya 1,0'dan daha büyük standartlaştırılmış OUT çıkış değerleri oluşturabilir. Bu durumda, NORM_X yürütülürken ENO = DOĞRU olarak ayarlanır.

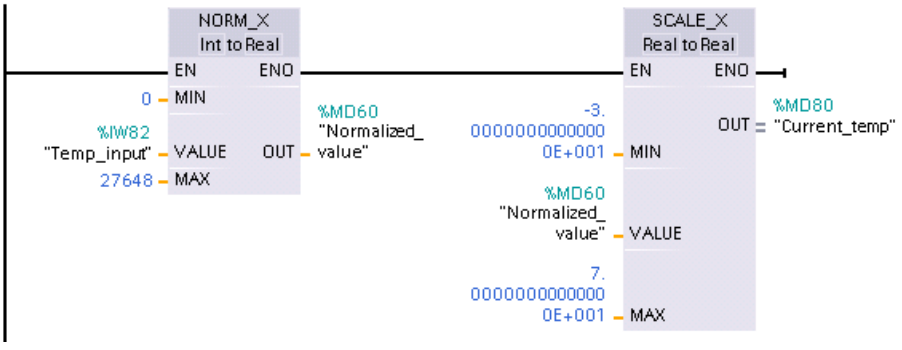
Tablo 8-124 ENO'nun durumu

ENO	Koşul	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	Sonuç, OUT veri tipi için kullanılan geçerli aralığı aşar.	Ara sonuç: OUT'un veri tipine nihai dönüştürme işleminden önce bir gerçek sayının en düşük değerli kısmı.
0	MAX parametresi <= MIN	SCALE_X: OUT'un boyutunun kapladığı VALUE gerçek sayısının en düşük değerli kısmı. NORM_X: VALUE veri tipindeki VALUE, çift sözcük boyutunun atanması için genişletildi.
0	VALUE parametresi = +/- INF veya +/- NaN	VALUE, OUT'a yazılır

Örnek (LAD): Bir analog giriş değerinin standartlaştırılması ve ölçeklenmesi

Analog bir sinyal modülünün veya akım girişli bir sinyal kartının bir analog girişi, geçerli değerlerin 0 ile 27648'lik aralığında yer alır. Bir analog girişin bir sıcaklığı gösterdiğini ve bu arada analog girişin 0 değerinin -30,0 Santigradı ve 27648 değerinin 70,0 Santigradı gösterdiğini kabul edelim.

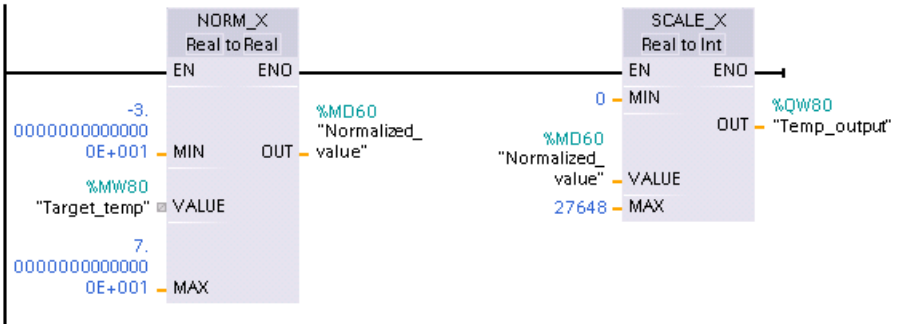
Analog değerin ilgili fiziksel birimlere dönüştürülmesi için, girişi 0,0 ile 1,0 arasında bir değer olarak standartlaştırın ve ardından -30,0 ile 70,0 arasında ölçekleyin. Sonuçta elde edilen değer, analog giriş tarafından Santigrat cinsinden gösterilen sıcaklıktır:



Analog girişin bir analog sinyal modülünden veya gerilim girişli bir sinyal kartından gelmesi durumunda, NORM_X yordamı için MIN değerinin 0 yerine -27648 olacağına dikkat edin.

Örnek (LAD): Bir analog çıkış değerinin standartlaştırılması ve ölçeklenmesi

Analog bir sinyal modülünde veya akım çıkışlı bir sinyal kartında ayarlanması gereken bir analog çıkış, geçerli değerlerin 0 ila 27648'lik aralığında yer almalıdır. Bir analog çıkışın bir sıcaklık ayarını gösterdiğini ve bu arada analog girişin 0 değerinin -30,0 Santigradı ve 27648 değerinin 70,0 Santigradı gösterdiğini kabul edelim. Bellekte bulunan -30,0 ile 70,0 arasındaki bir sıcaklık değerini 0 ile 27648 aralığındaki bir analog çıkış değerine dönüştürmek için, fiziksel birimlerdeki değeri 0,0 ile 1,0 arasında bir değer olarak normalleştirmeniz ve daha sonra 0 ile 27648 arasındaki analog çıkış aralığına göre ölçeklendirmeniz gerekir:



Bir analog sinyal modülü veya gerilim girişli bir sinyal kartı için kullanılan analog çıkışın öngörülmüş olması durumunda, SCALE_X yordamı için MIN değerinin 0 yerine -27648 olacağına dikkat edin.

Gerilim ve akım için Analog girişlerin gösterimi (Sayfa 1271) ve Analog çıkışların gösterimi (Sayfa 1272) ile ilgili daha fazla bilgiyi teknik verilerde bulabilirsiniz.

8.7.6 Varyant dönüştürme yordamları

8.7.6.1 VARIANT_TO_DB_ANY (VARIANT'ı DB_ANY'ye dönüştür)

"VARIANT to DB_ANY" yordamıyla, IN parametresinin işlemcisi okunur ve DB_ANY veri tipine dönüştürülür. IN parametresi, Variant veri tipine sahiptir ve bir örnek veri bloğunu veya bir ARRAY veri bloğunu gösterir. Program oluşturulurken, hangi veri bloğunun IN parametresine karşılık geldiğinin bilinmesi gerekmez. Yordam, çalışma zamanı sırasında veri bloğu numarasını okur ve RET_VAL parametresinin işlemcilerine yazar.

Tablo 8-125 VARIANT_TO_DB_ANY yordamı

LAD/FBD	SCL	Açıklama
Mevcut değil	**RET_VAL := VARIANT_TO_DB_ANY(in := _variant_in_, err => _int_out_);**	Variant IN parametresinin işlemcisini okur ve DB_ANY tipinin fonksiyon sonucuna kaydeder

Tablo 8-126 VARIANT_TO_DB_ANY yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
IN	Variant	Bir örnek veri bloğunu veya Array veri bloğunu gösteren Variant elemanı.
RET_VAL	DB_ANY	Dönüştürülen veri bloğu numarasını içeren yayınlanmış DB_ANY veri tipi.
ERR	Int	Hata bilgileri

Tablo 8-127 ENO'nun durumu

ENO	Koşul	Sonuç
1	Hata yok	Yordam, Variant girişini dönüştürür ve DB_ANY fonksiyon çıkışına kaydeder.
0	EN serbest bırakma girişi "0" sinyal durumuna sahiptir veya IN parametresi geçersizdir.	Yordam hiçbir şey yapmaz.

Tablo 8-128 VARIANT_TO_DB_ANY yordamı için kullanılan hata kodları

Err (W#16#...)	Açıklama
0000	Hata yok
252C	IN parametresinin Variant veri tipi, 0 değerine sahiptir. CPU, STOP işletim durumuna geçer.
8131	Veri bloğu mevcut değil veya çok kısa (ilk erişim).
8132	Veri bloğu çok kısa ve hiçbir Array veri bloğu mevcut değil (ikinci erişim).
8134	Veri bloğu yazma korumalıdır

Err (W#16#...)	Açıklama
8150	IN parametresinin Variant veri tipi, "0" değerini sağlar. Bu hata mesajı için "Bloktaki hatayı düzenle" blok özelliği etkinleştirilmiş olmalıdır. Aksi takdirde CPU STOP işletim durumuna geçer ve 16#252C hata kodunu gönderir
8154	Veri bloğu, yanlış veri tipine sahiptir.
*Hata kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir.	

8.7.6.2 DB_ANY_TO_VARIANT (DB_ANY'yi VARIANT'a dönüştür)

"DB_ANY to VARIANT" yordamıyla, aşağıdaki ön koşulları sağlayan veri bloğunun numarası okunabilir. IN parametresinin işlemcisi DB_ANY veri tipine sahiptir, bu nedenle program oluşturulurken hangi veri bloğunun okunacağının bilinmesi gerekmez. Yordam, çalışma zamanı esnasında veri bloğu numarasını okur ve bir VARIANT işaretçisinin yardımıyla RET_VAL fonksiyon sonucuna yazar.

Tablo 8-129 DB_ANY_TO_VARIANT yordamı

LAD/FBD	SCL	Açıklama
Mevcut değil	**RET_VAL := DB_ANY_TO_VARIANT(in := _db_any_in_, err => _int_out_);**	Variant IN parametresinin veri bloğu numarasını okur ve VARIANT tipinin fonksiyon sonucuna kaydeder.

Tablo 8-130 DB_ANY_TO_VARIANT yordamı için kullanılan parametreler

Parametre	Veri tipi	Açıklama
IN	DB_ANY	Veri bloğu numarasını içeren Variant elemanı
RET_VAL	Variant	Dönüştürülen veri bloğu numarasını içeren yayınlanmış DB_ANY veri tipi.
ERR	Int	Hata bilgileri

Tablo 8-131 ENO'nun durumu

ENO	Koşul	Sonuç
1	Hata yok	Yordam, veri bloğu numarasını Variant formatına dönüştürür ve DB_ANY fonksiyon çıkışına kaydeder
0	EN serbest bırakma girişi "0" sinyal durumuna sahiptir veya IN parametresi geçersizdir.	Yordam hiçbir şey yapmaz.

Tablo 8-132 DB_ANY_TO_VARIANT yordamı için kullanılan hata kodları

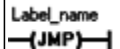
Err (W#16#...)	Açıklama
0000	Hata yok
8130	Veri bloğunun numarası 0'dır.
8131	Veri bloğu mevcut değil veya çok kısa.
8132	Veri bloğu çok kısa ve hiçbir Array veri bloğu mevcut değil.
8134	Veri bloğu yazma korumalıdır.

Err (W#16#...)	Açıklama
8154	Veri bloğu, yanlış veri tipine sahiptir.
8155	Bilinmeyen tip kodu
*Hata kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir.	

8.8 Program kumandası işlemleri

8.8.1 JMP (VKE = 1'de atla), JMPN (VKE = 0'da atla) ve Label (atlama işareti) yordamları

Tablo 8-133 JMP, JMPN ve LABEL yordamları

LAD	FBD	SCL	Açıklama
		Bkz. GOTO (Sayfa 302) yordamı.	VKE (bağlantı sonucu) = 1'de atla: Bir JMP bobinine sinyal akışında (LAD) veya JMP kutusu girişi doğru ise (FBD), programın yürütülmesine, belirtilen atlama işaretinden sonraki ilk yordamla devam edilir.
			VKE = 0'da atla: Bir JMPN bobinine (LAD) sinyal akışı yokken veya JMPN kutu girişi yanlış ise (FBD), programın yürütülmesine, belirtilen atlama işaretinden sonraki ilk yordamla devam edilir.
			Bir JMP veya JMPN atlama yordamı için kullanılan atlama hedefi

[1] Atlama işaretlerinin adlarını, LABEL yordamını doğrudan girerek oluşturabilirsiniz. Parametre sembolü aracılığıyla, JMP veya JMPN yordamı için kullanılabilen atlama işaretlerini seçebilirsiniz. Atlama işaretinin adını doğrudan JMP veya JMPN yordamına girmeniz de mümkündür.

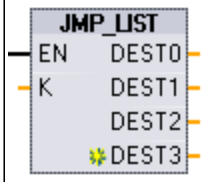
Tablo 8-134 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
Label_name	Atlama işaretinin kodu	Atlama yordamları için kullanılan kod ve atlama hedefi için kullanılan ilgili atlama işareti

- Her atlama işareti, bir kod bloğunun içinde benzersiz olmalıdır.
- Bir kod bloğunun içinde atlama yapabilir, ancak bir kod bloğundan başka bir kod bloğuna atlayamazsınız.
- İleriye doğru veya geriye doğru atlayabilirsiniz.
- Bir kod bloğunun birçok noktasından aynı atlama işaretine atlayabilirsiniz.

8.8.2 JMP_LIST (atlama listesini tanımla)

Tablo 8-135 JMP_LIST yordamı

LAD/FBD	SCL	Açıklama
	**CASE k OF** **0: GOTO dest0;** **1: GOTO dest1;** **2: GOTO dest2;** **[n: GOTO destn;]** **END_CASE;**	JMP_LIST yordamı, program alt bölümlerine kumanda etmek için program atlamalarını dağıtır. K girişinin değerine bağlı olarak, programda ilgili atlama işaretine atlanır. Programın yürütülmesi işlemine, atlama hedefini takip eden program yordamlarıyla devam edilir. K girişinin değeri, atlama işareti sayısı - 1 değerini aşarsa, hiçbir atlama işlemi gerçekleştirilmez ve işlemeye, programda bulunan bir sonraki ağla devam edilir.

Tablo 8-136 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
K	UInt	Atlama dağıtma için kullanılan kumanda değeri
DEST0, DEST1, .., DESTn.	Programdaki atlama işaretleri	Atlama hedeflerinin, K parametresinin belirli değerlerine uygun atlama işaretleri: K'nın değeri 0'a eşitse, DEST0 çıkışına atanmış olan atlama işaretine bir atlama işlemi gerçekleştirilir. K'nın değeri 1'e eşitse, DEST1 çıkışına, vb. atanmış olan atlama işaretine bir atlama işlemi gerçekleştirilir. K girişinin değeri (atlama işaretleri sayısı - 1) aşılırsa, hiçbir atlama işlemi gerçekleştirilmez ve işlemeye, programda bulunan bir sonraki ağla devam edilir.

LAD ve FBD'de: JMP_LIST yordamı, programınızda ilk kez eklenirken atlama işaretleri için iki çıkış mevcuttur. Atlama hedeflerini ekleyebilir veya silebilirsiniz.



Bir atlama hedefi eklemek için, yeni bir hedef çıkış eklemek üzere kutudaki Oluştur sembolüne tıklayın.

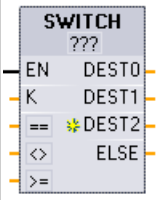
Alternatif olarak, farenin sağ tuşuyla bir çıkışa tıklayabilir ve "Çıkış ekle" yordamını seçebilirsiniz.



Bir atlama hedefini silmek için, farenin sağ tuşuyla bir çıkış bağlantısına tıklayın ve "Sil" yordamını seçin.

8.8.3 SWITCH (atlama dağıtma)

Tablo 8-137 SWITCH yordamı

LAD/FBD	SCL	Açıklama
	**Mevcut değil**	SWITCH yordamı, program alt bölümlerine kumanda etmek için program atlamalarını dağıtır. K girişinin değeri, belirtilen karşılaştırma girişlerinin değerleri arasındaki karşılaştırma sonucuna bağlı olarak, DOĞRU sonucuna sahip ilk karşılaştırma testine karşılık gelen atlama işaretine bir atlama için kullanılır. Karşılaştırmaların hiçbiri doğru değilse, ELSE'in atandığı atlama işaretine bir atlama yapılır. Programın yürütülmesi işlemine, atlama hedefini takip eden program yordamlarıyla devam edilir.

1 LAD ve FBD'de: Kutu adının alt kısmına tıklayın ve açılır listeden bir veri tipi seçin.

2 SCL'de: Bir CASE yordamı veya bir IF-THEN karşılaştırma seti kullanın.

Tablo 8-138 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
K	UInt	Genel karşılaştırma değeri için kullanılan giriş
==, <>, <, <=, >. >=	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, TOD, Date	Özel karşılaştırma türlerine yönelik münferit karşılaştırma değerleri için kullanılan girişler
DEST0, DEST1, .., DESTn, ELSE	Programdaki atlama işaretleri	Atlama hedeflerinin, özel karşılaştırmalara uygun atlama işaretleri: K girişinin altında ve yanında yer alan karşılaştırma girişi önce işlenir ve K değeri ile bu giriş arasındaki karşılaştırma doğru ise, DEST0'ın atandığı atlama işaretine bir atlama yapılmasına neden olur. Bir sonraki karşılaştırma testi, bunun altındaki girişi kullanır ve karşılaştırma doğru ise DEST1'in atandığı atlama işaretine bir atlama yapılmasına neden olur. Diğer karşılaştırmalar aynı şekilde işlenir ve karşılaştırmaların hiçbiri doğru değilse, ELSE çıkışına atanmış olan atlama işaretine bir atlama yapılır.

1 K girişi ve karşılaştırma girişleri (==, <>, <, <=, >, >=) aynı veri tipine sahip olmalıdır.

Girişler ekleme, girişleri silme ve karşılaştırma türlerini belirtme

Programınızdaki LAD veya FBD'de bir SWITCH yordamı ilk kez eklenirken iki karşılaştırma girişi mevcuttur. Karşılaştırma türlerini atayabilir ve girişleri veya atlama hedeflerini ekleyebilir ve silebilirsiniz (bkz. alt kısım).



Kutuda bir karşılaştırma operatörüne tıklayın ve açılır listeden yeni bir operatör seçin.



Yeni bir karşılaştırma girişi ve karşılaştırma çıkışı eklemek için kutudaki Oluştur sembolüne tıklayın.

Alternatif olarak, farenin sağ tuşuyla herhangi bir çıkış parametresine tıklayabilir ve "Giriş ve çıkış ekle" komutunu seçebilirsiniz.



SWITCH yordamındaki bir karşılaştırmayı silmek için, farenin sağ tuşuyla bir girişe veya çıkışa tıklayın ve "Sil" komutunu seçin. Yordam, en az iki karşılaştırma içerir.

Tablo 8-139 SWITCH kutusu için kullanılan veri tipinin seçimi ve izin verilen karşılaştırma yordamları

Veri tipi	Karşılaştırma	Operatör sözdizimi
Bayt, Word, DWord	Eşittir	==
	Eşit değil	<>
SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Time, TOD, Date	Eşittir	==
	Eşit değil	<>
	Büyüktür veya eşittir	>=
	Küçüktür veya eşittir	<=
	Büyüktür	>
	Küçüktür	<

SWITCH kutusunu yerleştirmeyle ilgili kurallar

- Karşılaştırma girişinden önce hiçbir LAD/FBD yordamı bağlanamaz.
- Hiçbir ENO çıkışı mevcut değildir, bu nedenle bir ağda sadece bir SWITCH yordamına izin verilir ve SWITCH yordamı, ağdaki son yordam olmak zorundadır.

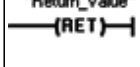
8.8.4 RET (geri dönüş değeri)

Opsiyonel RET yordamıyla, güncel bloğun yürütülmesi işlemi sonlandırılır. Sadece RET bobinine (LAD) sinyal akışında veya RET kutusu doğru ise (FBD), güncel bloğun program yürütme işlemi bu noktada sonlandırılır; RET yordamından sonra başka hiçbir yordam yürütülmez. Güncel blokta bir OB söz konusu ise, "Return_Value" parametresi göz ardı edilir. Güncel blokta bir FC veya bir FB söz konusu ise, "Return_Value " parametresinin değeri, çağrılan kutunun ENO değeri olarak tekrar çağıran rutine aktarılır.

RET yordamını bir bloğa son yordam olarak eklemek zorunda değilsiniz, bu işlem otomatik olarak gerçekleştirilir. Birden fazla RET yordamını tek bir bloğa ekleyebilirsiniz.

SCL için, bkz. RETURN (Sayfa 303) yordamı.

Tablo 8-140 Geri dönüş değeri (RET) program kumanda yordamı

LAD	FBD	SCL	Açıklama
		**RETURN;**	Güncel bloğun yürütülmesi işlemini sonlandırır.

Tablo 8-141 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
Return_Value	Bool	RET yordamının "Return_value" parametresi, çağıran bloktaki blok çağrısı kutusunun ENO çıkışına atanmıştır.

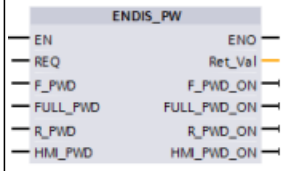
Bir FC kod bloğunda bir RET yordamının kullanımıyla ilgili örnek:

1. Yeni bir proje oluşturun ve bir FC ekleyin.
2. FC'yi düzenleyin:
   – Yordamları ekleyin.
   – "Return_Value" parametresi için aşağıdaki elemanlardan biri de dahil olmak üzere bir RET yordamı ekleyin:
   DOĞRU, YANLIŞ veya gereken geri dönüş değerini belirten bir adres.
   – Başka yordamlar ekleyin.
3. FC'yi MAIN [OB1] ana programından çağırın.

MAIN kod bloğundaki FC kutusunun EN girişi, FC'nin yürütülmesi için doğru olmalıdır.
FC'deki RET yordamı tarafından belirtilen değer, RET yordamında sinyal akışı yer alan FC'nin yürütülmesinden sonra MAIN kod bloğundaki FC kutusunun ENO çıkışında bulunur.

8.8.5 ENDIS_PW (CPU şifresini etkinleştir/devre dışı bırak)

Tablo 8-142 ENDIS_PW yordamı

LAD/FBD	SCL	Açıklama
	**ENDIS_PW(** **req:=_bool_in_,** **f_pwd:=_bool_in_,** **full_pwd:=_bool_in_,** **r_pwd:=_bool_in_,** **hmi_pwd:=_bool_in_,** **f_pwd_on=>_bool_out_,** **full_pwd_on=>_bool_out_,** **r_pwd_on=>_bool_out_,** **hmi_pwd_on=>_bool_out_);**	ENDIS_PW yordamıyla, bir istemcinin bir S7-1200 CPU'su ile olan bağlantısı, serbest bırakılabilir veya istemci doğru şifreyi sağlasa bile bloke edilebilir. Bu yordam, hiçbir Web sunucusu şifresini bloke etmez.

Tablo 8-143 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQ=1 ise fonksiyonu yürüt
F_PWD	IN	Bool	Hata güvenli işletim şifresi: Serbest bırak (=1) veya bloke et (=0)
FULL_PWD	IN	Bool	Tam erişim şifresi: Tam erişim şifresini serbest bırak (=1) veya bloke et (=0)
R_PWD	IN	Bool	Okuma erişimi şifresi: Serbest bırak (=1) veya bloke et (=0)
HMI_PWD	IN	Bool	HMI şifresi: Serbest bırak (=1) veya bloke et (=0)
F_PWD_ON	OUT	Bool	Hata güvenli işletim şifresinin durumu: Serbest bırakıldı (=1) veya bloke edildi (=0)
FULL_PWD_ON	OUT	Bool	Tam erişim şifresinin durumu: Serbest bırakıldı (=1) veya bloke edildi (=0)
R_PWD_ON	OUT	Bool	Okuma erişimi şifresinin durumu: Serbest bırakıldı (=1) veya bloke edildi (=0)
HMI_PWD_ON	OUT	Bool	HMI şifresinin durumu: Serbest bırakıldı (=1) veya bloke edildi (=0)
Ret_Val	OUT	Word	Fonksiyon sonucu

ENDIS_PW yordamı REQ=1 ile çağrılırsa, ilgili şifre giriş parametresi YANLIŞ olan şifre tipleri bloke edilir. Her şifre tipi münferit olarak serbest bırakılabilir veya bloke edilebilir. Örneğin

sadece hata güvenli işlem şifresi serbest bırakılırsa ve diğer tüm şifreler bloke edilirse, CPU'ya erişim küçük bir kullanıcı grubuyla sınırlanabilir.

ENDIS_PW yordamı, bir program çevriminde senkron olarak yürütülür ve şifre çıkış parametreleri her zaman REQ giriş parametresinden bağımsız olarak güncel şifre serbest bırakma durumunu gösterir. Prensip olarak, serbest bıraktığınız şifrelerin bloke edilebilmeleri de gerekir. Aksi takdirde bir hata yayınlanır ve ENDIS_PW yordamı yürütülmeden önce serbest bırakılmış olan tüm şifreler serbest bırakılır. Bu, bir standart CPU'da (hata güvenli işletim şifresinin yapılandırılmadığı bir CPU), 0 şeklindeki bir geri dönüş değerine ulaşmak için F_PWD'nin her zaman 1'e ayarlanmış olması gerektiği anlamına gelir. Bu durumda F_PWD_ON her zaman 1'dir.

NOT

- ENDIS_PW yordamının yürütülmesi, HMI şifresi bloke edildiyse, HMI cihazlarına erişimi engelleyebilir.
- ENDIS_PW yordamı yürütülmeden önce serbest bırakılmış olan istemci oturumları, ENDIS_PW yordamının yürütülmesi nedeniyle mevcut onaylama kademesine bağlı olarak sonlandırılabilir. Bu sayede, örneğin OKUMA şifre korumasıyla onaylanan bir bağlantı ENDIS_PW (REQ=1, R_PWD=0) tarafından iptal edilebilir. Bu senaryoda, Koruma kademesi daha düşük olan diğer bağlantılar da aynı şekilde sonlandırılır. TAM ERİŞİME sahip bağlantılar ise bunun tersine korunur.

Yol alma işleminden sonra CPU erişimi, daha önce CPU korumasının düzenli konfigürasyonunda tanımlanmış olan şifreler aracılığıyla kısıtlanır. Geçerli bir şifreyi bloke etme özelliği, ENDIS_PW yordamının yürütülmesiyle birlikte yeniden ayarlanmalıdır. Ancak ENDIS_PW yordamı hemen yürütülürse ve bunun sonucunda gerekli şifreler bloke edilirse, TIA Portal'a erişim bloke edilebilir. Bir zaman işleminin yardımıyla ENDIS_PW yordamının yürütülmesini geciktirebilir ve şifreler bloke edilmeden önce şifrelerin girilmesi için bir süre tanınmasını sağlayabilirsiniz.

NOT

TIA Portal ile iletişimi bloke edilen bir CPU'nun geri yüklenmesi

Bir PLC'nin yükleme belleğinin bir memory card yardımıyla silinmesiyle ilgili daha fazla bilgiyi, "Kaybolan bir şifrenin geri yüklenmesi (Sayfa 126)" başlığında bulabilirsiniz.

Hatalar, STP yürütme nedeniyle veya STEP 7 tarafından işletim durumunun STOP olarak değişmesi, korumayı etkilemez. Koruma, CPU kapatılıp tekrar açılana kadar geçerlidir. Aşağıdaki tabloda, konuyla ilgili diğer ayrıntıları bulabilirsiniz.

İşlem	İşletim durumu	ENDIS_PW şifre koruması
STEP 7 aracılığıyla tam silme işleminden sonra	STOP	Aktif: Bloke edilmiş şifreler bloke edilmiş durumda kalır.
Açma veya bir memory card'ın değiştirilmesi işleminden sonra	STOP	Kapalı: Hiçbir şifre bloke edilmemiş.
ENDIS_PW yordamı bir çevrim OB'sinde veya yol alma OB'sinde yürütüldükten sonra	STARTUP, RUN	Aktif: Şifreler, ENDIS_PW parametrelerine uygun olarak bloke edilmiştir
RUN veya STARTUP işletim durumu STP yordamı, hatalar veya STEP 7 tarafından STOP olarak değiştirildikten sonra	STOP	Aktif: Bloke edilmiş şifreler bloke edilmiş durumda kalır

⚠ İKAZ

Korunan bir CPU'ya yetkili olmayan kişilerin erişimi

Tam CPU erişim yetkilerine veya hata güvenli özelliği de dahil tam erişim olanağına sahip kullanıcılar PLC değişkenlerini okuyabilir ve yazabilir. Web sunucusu kullanıcıları, CPU için kullanılan erişim kademesinden bağımsız olarak PLC değişkenlerini okuma ve yazma yetkilerine sahip olabilir. CPU'ya yetkisi olmayan kişiler tarafından erişim veya PLC değişkenleri için geçersiz değerlerin ayarlanması, proses işletimine zarar verebilir ve hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yetkili kullanıcılar, işletim durumunda değişiklikler yapabilir, PLC verilerini yazabilir ve Firmware güncellemeleri uygulayabilir. Siemens, aşağıdaki güvenlik önlemlerine uyulmasını önerir:

- CPU erişim kademelerini (Sayfa 156) ve Web sunucusu kullanıcı ID'lerini (Sayfa 801) STEP 7'de tanımlandığı şekilde güçlü şifreler aracılığıyla koruyun.
- Web sunucusuna erişimi sadece HTTPS protokolü üzerinden etkinleştirin.
- Web sunucusu kullanıcısının "Herkes" şeklindeki standart asgari yetkilerini genişletmeyin.
- Web sitelerinin kullanıcıları PLC değişkenleri için geçersiz değerler belirleyebileceği için, program lojiğinizdeki değişkenler için bir hata kontrolü ve bir aralık kontrolü gerçekleştirin.
- Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın.

Tablo 8-144 Koşul kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	Yordam desteklenmiyor.
80D0	Şifre, hata güvenli işletim için yapılandırılmamış:

RET_VAL (W#16#...)	Açıklama
80D1	Şifre, okuma/yazma erişimi için yapılandırılmamış.
80D2	Şifre, okuma erişimi için yapılandırılmamış.
80D3	Şifre, HMI erişimi için yapılandırılmamış.

8.8.6 RE_TRIGR (çevrim denetleme süresini yeniden başlat)

Tablo 8-145 RE_TRIGR yordamı

LAD/FBD	SCL	Açıklama
	**RE_TRIGR();**	RE_TRIGR (Çevrim süresi denetlemesini yeniden başlat) ile çevrim süresi denetlemesi tarafından bir hata mesajı oluşana kadar izin verilen maksimum süre uzatılabilir.

RE_TRIGR yordamıyla, çevrim süresi denetlemesini bir çevrim akışı sırasında yeniden başlatabilirsiniz. Bu sayede, izin verilen maksimum çevrim süresi, RE_TRIGR fonksiyonunun son yürütülmesi ile maksimum bir çevrim süresi kadar uzatılır.

NOT

S7-1200 CPU'sunun 2.2 Firmware sürümünden önce, RE_TRIGR'in, bir program çevrimi OB tarafından yürütülmesi gerekiyordu ve bu komut, hedef sistemin çevrim süresini sınırsız bir şekilde uzatmak için kullanılabiliyordu. RE_TRIGR bir yol alma OB'si, bir alarm OB'si veya bir hata OB'sinden çağrılırsa ENO = YANLIŞ olur ve çevrim süresi denetlemesi sıfırlanmaz.

RE_TRIGR, Firmware sürümü 2.2'den itibaren her OB tarafından yürütülebilir (yol alma OB'leri, alarm OB'ler ve hata OB'ler de dahil). Ancak hedef sistemin çevrim süresi, sadece yapılandırılmış maksimum çevrim süresinin 10 katı kadar uzatılabilir.

PLC cihazının maksimum çevrim süresinin ayarlanması

Maksimum çevrim süresinin değerini, cihaz konfigürasyonundaki "Çevrim süresi" öğesinin altında yapılandırın.

Tablo 8-146 Çevrim süresi değerleri

Çevrim süresi denetlemesi	Minimum değer	En yüksek değer	Ön ayar
Maksimum çevrim süresi	1 msn	6000 msn	150 msn

Çalışma zamanı denetimi

Maksimum çevrim süresi, çevrim tamamlanmadan önce sona ererse bir hata üretilir. Kullanıcı programı bir zaman hatası alarmı OB'si (OB 80) içeriyorsa, CPU, zaman hatası alarm OB'sini yürütür. Bu, belirli bir tepkiyi etkinleştirmek için gereken lojiği içerebilir.

Kullanıcı programı hiçbir zaman hatası alarmı OB'si içermiyorsa, ilk zaman aşımı koşulu göz ardı edilir ve CPU, RUN işletim durumunda kalır. Aynı program çevriminde maksimum çevrim süresinin ikinci kez aşılması durumu ortaya çıkarsa (2 x maksimum çevrim süresi değeri), hedef sistemin STOP işletim durumuna geçmesine neden olan bir hata tetiklenir.

STOP işletim durumunda, programın yürütülmesi durdurulur, ancak CPU sistem iletişimi ve sistem teşhisi çalışmaya devam eder.

8.8.7 STP (Programı sonlandır)

Tablo 8-147 STP yordamı

LAD/FBD	SCL	Açıklama
	**STP();**	STP, CPU'yu STOP işletim durumuna getirir. CPU, STOP işletim durumunda ise, programın yürütülmesi ve proses görünümünde giriş ve çıkışların güncellenmesi durdurulur.

Daha fazla bilgi için bkz.: Çıkışların, RUN işletim durumundan STOP işletim durumuna geçmek için yapılandırılması (Sayfa 92).

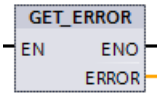
EN = DOĞRU ise, CPU STOP işletim durumuma geçer, programın yürütülmesi sona erer ve ENO'nun durumu anlamsız hale gelir. Aksi takdirde EN = ENO = 0 olur.

8.8.8 GET_ERROR ve GET_ERROR_ID (Hatayı yerel olarak çağır ve Hata ID'sini yerel olarak çağır) yordamları

GET_ERROR yordamları, program blokları yürütülürken ortaya çıkan hatalarla ilgili bilgileri sağlar. Bir GET_ERROR veya GET_ERROR_ID yordamını kod bloğunuza kaydederseniz, program hatalarını program bloğunuzda giderebilirsiniz.

GET_ERROR

Tablo 8-148 GET_ERROR yordamı

LAD/FBD	SCL	Açıklama
	**GET_ERROR(_out_);**	Yerel bir program bloğu yürütülürken bir hatanın ne zaman oluştuğunu gösterir ve ayrıntılı hata bilgileri içeren önceden tanımlanmış bir hata veri yapısına sahiptir.

Tablo 8-149 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
ERROR	ErrorStruct	Hata veri yapısı: Yapıyı yeniden adlandırabilir, ancak yapının içindeki elemanları yeniden adlandıramazsınız.

Tablo 8-150 ErrorStruct veri yapısının elemanları

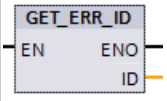
Yapı bileşenleri		Veri tipi	Açıklama
ERROR_ID		Word	Hata ID'si
FLAGS		Bayt	Bir blok çağrısı sırasında bir hatanın oluşup oluşmadığını gösterir. • 16#01: Bir blok çağrısı sırasında hata oluştu. • 16#00: Bir blok çağrısı sırasında hiçbir hata oluşmadı.
REACTION		Bayt	Standart tepki: • 0: Yoksay (yazma hatası), • 1: "0" yedek değeriyle (okuma hatası) devam et, • 2: Yordamı atla (sistem hatası)
CODE_ADDRESS		CREF	Adres ve blok tipiyle ilgili bilgiler
	BLOCK_TYPE	Bayt	Hatanın oluştuğu bloğun tipi: • 1: OB • 2: FC • 3: FB
	CB_NUMBER	UInt	Kod bloğunun numarası
	OFFSET	UDInt	Dahili belleğe referans
MODE		Bayt	Erişim türü: Erişimin türüne bağlı olarak aşağıdaki bilgiler yayınlanabilir:

İşletim türü	(A)	(B)	(C)	(D)	(E)
0					
1					Ofset
2			Area		
3	Adres	Kapsam		Number (numara)	
4			Area		Ofset

Yapı bileşenleri		Veri tipi	Açıklama					
MODE		Bayt	5			Area	DB No.	Ofset
			6	PtrNr.		Area	DB No.	Ofset
			7	PtrNr. / Zug.	Stpl.-No./Ka-psam	Area	DB No.	Ofset
OPERAND_NUMBER		UInt	Makine komutunun işlemci numarası					
POINTER_NUMBER_ LOCATION		UInt	(A) Dahili işaretçi					
SLOT_NUMBER_SCOPE		UInt	(B) Dahili bellekteki bellek alanı					
DATA_ADDRESS		NREF	Bir işlemcinin adresiyle ilgili bilgiler					
	AREA	Bayt	(C) Bellek alanı: • L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE • I: 16#81 • A: 16#82 • M: 16#83 • DB: 16#84, 85, 8A, 8B					
	DB_NUMBER	UInt	(D) Veri bloğunun numarası					
	OFFSET	UDInt	(E) İşlemcilerin bağıl adresi					

GET_ERROR_ID

Tablo 8-151 GetErrorID yordamı

LAD/FBD	SCL	Açıklama
	**GET_ERR_ID();**	Bir program bloğu yürütülürken bir hatanın ne zaman oluştuğunu gösterir ve hatanın ID'sini (kod) bildirir.

Tablo 8-152 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
ID	Word	ERROR_ID ErrorStruct elemanı için kullanılan hata karakteristik değerleri

Tablo 8-153 Error_ID değerleri

ERROR_ID onaltılık	ERROR_ID desimal	Program bloğu yürütme hatası
0	0	Hata yok
2520	9504	Karakter dizisi bozuk
2522	9506	İşlemci aralık dışında okuma hatası
2523	9507	İşlemci aralık dışında yazma hatası
2524	9508	Geçersiz aralık okuma hatası
2525	9509	Geçersiz aralık yazma hatası
2528	9512	Veri hizası okuma hatası (Bit hizası hatalı)
2529	9513	Veri hizası yazma hatası (Bit hizası hatalı)
252C	9516	Başlatılmayan işaretçi hatası
2530	9520	DB yazma korumalı

ERROR_ID onaltılık	ERROR_ID desimal	Program bloğu yürütme hatası
2533	9523	Geçersiz işaretçi
2538	9528	Erişim hatası: DB mevcut değil
2539	9529	Erişim hatası: Yanlış DB
253A	9530	Global DB mevcut değil
253C	9532	Sürüm yanlış veya FC mevcut değil
253D	9533	Yordam mevcut değil
253E	9534	Sürüm yanlış veya FB mevcut değil
253F	9535	Yordam mevcut değil
2550	9552	Erişim hatası: DB mevcut değil
2575	9589	Programın gömme derinliğinde hata
2576	9590	Yerel veriler atama hatası
2942	10562	Fiziksel giriş mevcut değil
2943	10563	Fiziksel çıkış mevcut değil

Çalışma şekli

CPU, standart olarak bir blok yürütme hatasına, hatayı teşhis tamponuna kaydederek tepki gösterir. Ancak bir veya daha fazla GET_ERROR veya GET_ERROR_ID yordamını bir kod bloğuna yerleştirirseniz, bu blok, bloktaki hataları giderebilir. Bu durumda CPU, teşhis tamponunda hata bildirmez. Bunun yerine, hata bilgileri GET_ERROR veya GET_ERROR_ID yordamının çıkışında bildirilir. Ayrıntılı hata bilgilerini GET_ERROR yordamıyla okuyabilirsiniz, GET_ERROR_ID yordamıyla sadece hata ID'si okunur. Normalde ilk hata en önemli hatadır ve sonraki hatalar sadece ilk hatanın sonuçlarıdır.

Bir GET_ERROR veya GET_ERROR_ID yordamının bir bloğun içinde ilk kez yürütülmesiyle, bloğun yürütülmesi sırasında algılanan ilk hatanın yayınlanması sağlanır. Bu hata, bloğun başlatılması ile GET_ERROR veya GET_ERROR_ID yordamının yürütülmesi arasında herhangi bir konumda ortaya çıkmış olabilir. GET_ERROR veya GET_ERROR_ID yordamının sonraki yürütülmesi işlemleriyle GET_ERROR veya GET_ERROR_ID yordamının önceki yürütülmesi işleminden sonraki ilk hatanın yayınlanması sağlanır. Hataların ilerleyişi kaydedilmez ve yordamın yürütülmesi, bir sonraki hatanın yakalanması için hedef sistemi yeniden başlatır.

GET_ERROR yordamının ErrorStruct veri tipi, program lojiğinizin bu değerlere erişebilmesi için veri bloğu editörüne ve blok arayüzlerine eklenebilir. Bu yapıyı tamamlamak için açılır listeden ErrorStruct veri tipini seçin. Benzersiz adların yardımıyla birden fazla ErrorStruct elemanı oluşturabilirsiniz. Bir ErrorStruct'ın elemanları yeniden adlandırılamaz.

ENO tarafından görüntülenen hata koşulu

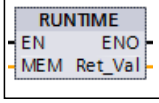
Şu geçerliyse: EN = DOĞRU ve GET_ERROR veya GET_ERROR_ID yordamı yürütülüyor ise, bu durumda:

- ENO = DOĞRU ifadesi, bir kod bloğunun yürütülmesi sırasında bir hatanın oluştuğunu ve hata verilerinin mevcut olduğunu belirtir
- ENO = YANLIŞ ifadesi, bir kod bloğunun yürütülmesi sırasında hiçbir hatanın oluşmadığını belirtir

Hata işleme programının lojiğini, bir hata oluştuktan sonra ayarlanan ENO girişine bağlayabilirsiniz. Bir hata oluştuğunda hata verileri, programınızın erişebileceği çıkış parametresine kaydedilir.
GET_ERROR ve GET_ERROR_ID yordamlarıyla güncel bloğun (çağrılan bloğun) hata bilgileri, çağıran bir bloğa gönderilebilir. Yordamı, çağrılan bloğun nihai yürütme durumunu bildirmek için, çağrılan blok programının son ağına yerleştirin.

8.8.9 RUNTIME (program çalışma zamanını ölç)

Tablo 8-154 RUNTIME yordamı

LAD/FBD	SCL	Açıklama
	**Ret_Val := RUNTIME(_Iread_inout_);**	Tüm programın, münferit blokların veya komut sekanslarının çalışma zamanını ölçer.

Tüm programın çalışma zamanını ölçmek için OB1'deki "Program çalışma süresini ölç" yordamı çağrılmalıdır. Çalışma zamanı ölçümü ilk çağrıyla başlar ve RET_VAL çıkışı, ikinci çağrıdan sonraki ölçülen program çalışma zamanını döndürür. Ölçülen çalışma zamanı, örneğin üst seviyedeki olaylar veya iletişim nedeniyle oluşan kesintiler gibi, programın yürütülmesi sırasında ortaya çıkabilecek tüm CPU proseslerini kapsar. "Program çalışma zamanını ölç" yordamı, CPU'nun dahili bir sayacını okur ve değeri, MEM IN-OUT parametresine yazar. Yordam, dahili sayaç frekansı aracılığıyla güncel program çalışma zamanını hesaplar ve RET_VAL çıkışına yazar.
Münferit blokların veya komut sekanslarının çalışma zamanının ölçülmesi gerekiyorsa, üç ayrı ağa ihtiyaç duyulur. "Program çalışma zamanını ölç" yordamını, programınızın münferit bir ağında çağırın. Çalışma zamanı ölçümünün başlangıç noktasını, yordamın ilk kez çağrılmasıyla ayarlayabilirsiniz. Ardından ihtiyaç duyulan program bloğunu veya komut sekansını bir sonraki ağda çağırın. "Program çalışma zamanını ölç" yordamını başka bir ağda ikinci kez çağırın ve MEM IN-OUT parametresine, yordam ilk kez çağrıldığında kullanılan belleği atayın. Üçüncü ağdaki "Program çalışma zamanını ölç" yordamı, CPU dahilindeki bir sayacı okur, dahili sayaç frekansı aracılığıyla program bloklarının veya komut sekansının güncel çalışma zamanını hesaplar ve bu değeri RET_VAL çıkışına yazar.
"Program çalışma zamanını ölç" yordamı, zamanı hesaplamak için dahili bir yüksek frekans sayacını kullanır. Yordamın, sayaç taşması durumunda döndürdüğü değerler <= 0,0 şeklindedir. Bu çalışma zamanı değerleri göz ardı edilebilir.

NOT

CPU, bir komut sekansının çalışma zamanını tam olarak belirleyemez, çünkü bir komut sekansındaki yordam sırası, optimize edilmiş program derleme esnasında değişir.

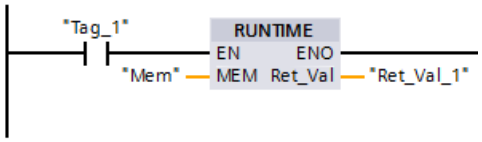
Tablo 8-155 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
MEM	LReal	Çalışma zamanı ölçümünün başlangıç noktası
RET_VAL	LReal	Saniye cinsinden ölçülen çalışma zamanı

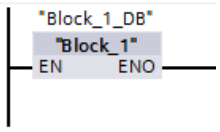
Örnek: RUNTIME yordamı

Aşağıdaki örnekte, RUNTIME yordamının, bir fonksiyon bloğunun yürütme süresini ölçmek için nasıl kullanıldığı gösterilmektedir:

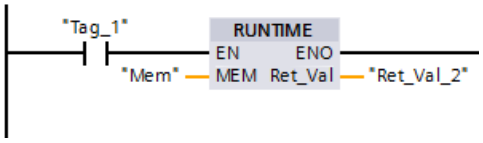
Ağ 1:



Ağ 2:



Ağ 3:



"Tag_1" işlemcisi 1. ağda "1" sinyal durumuna sahipse, RUNTIME yordamı yürütülür. Çalışma zamanı ölçümünün başlangıç noktası, yordamın ilk kez çağrılmasıyla ayarlanır ve yordamın ikinci kez çağrılması için "Mem" işlemcisinde referans olarak tamponlanır.

FB1 fonksiyon bloğu, 2. ağda yürütülür.

FN1 program bloğu yürütüldüyse ve "Tag_1" işlemcisi "1" sinyal durumuna sahipse, RUNTIME yordamı, 3. ağda yürütülür. Yordamın ikinci kez çağrılması, program bloğunun çalışma zamanının hesaplanmasını sağlar ve sonucu, RET_VAL_2 çıkışına yazar.

8.8.10 SCL'deki program kumanda yordamları

Structured Control Language (SCL), kullanıcı programınızın yapılandırılması için üç tür program kumanda yordamı sunar:

- Seçim yordamları: Bir seçim yordamının yardımıyla, program akışına alternatif yordam sıralarıyla kumanda edebilirsiniz.
- Döngüler: Döngü yürütmesi, tekrarlama yordamları aracılığıyla kumanda edebilirsiniz. Bir tekrarlama yordamı, bir programın hangi bölümlerinin belirli koşullara bağlı olarak tekrarlanması gerektiğini belirtir.
- Program atlamaları: Bir program atlaması, belirtilen bir atlama hedefine ve dolaysıyla aynı bloktaki başka bir yordama doğrudan atlamaya neden olur.

Program kumanda yordamları, PASCAL programlama dilinin sözdizimini kullanır.

Tablo 8-156 SCL'deki program kumanda yordamlarının türleri

Program kumanda yordamı		Açıklama
Seçim	IF-THEN (Sayfa 296)	Program akışını, DOĞRU veya YANLIŞ olan bir koşula bağlı olarak iki alternatiften birine dallandırmanıza olanak sağlar.
	CASE (Sayfa 297)	Bir değişkenin değeri temel alınarak *n* adet alternatif dallanmadan birinin seçime bağlı olarak uygulanmasına olanak sağlar.
Döngü	FOR (Sayfa 298)	Kumanda değişkeni belirtilen değer aralığının içinde olduğu sürece, bir yordam dizisinin tekrarlanmasını sağlar.
	WHILE-DO (Sayfa 299)	Bir uygulama koşulu sağlandığı sürece, bir yordam dizisinin tekrarlanmasını sağlar.
	REPEAT-UNTIL (Sayfa 300)	Bir iptal koşulu sağlandığı sürece, bir yordam dizisinin tekrarlanmasını sağlar.
Program atlaması	CONTINUE (Sayfa 301)	Anlık döngü geçişinin uygulanmasını iptal etmeyi sağlar.
	EXIT (Sayfa 302)	Bir döngüden, herhangi bir konumda ve iptal koşulunun sağlanmasından bağımsız olarak çıkmayı sağlar.
	GOTO (Sayfa 302)	Belirtilen atlama işaretine hemen atlamaya neden olur.
	RETURN (Sayfa 303)	Mevcut durumda yürütülen kod bloğundan çıkmaya ve çağıran kod bloğuna geri dönmeye neden olur.

8.8.10.1 IF-THEN

IF-THEN yordamı, bir grup yordamın, mantıksal bir ifadeye ait bir Bool değerinin değerlendirilmesi temel alınarak yürütüldüğü program akışına kumanda eden koşullu bir yordamdır. Birden fazla IF-THEN yordamının yürütülmesini, parantezlerin yardımıyla iç içe geçirebilir veya yapılandırabilirsiniz.

Tablo 8-157 IF-THEN yordamının elemanları

SCL	Açıklama
IF "Koşul" THEN **Yordam_A;** **Yordam_B;** **Yordam_C;** **;**	"Koşul" DOĞRU veya 1 ise, END_IF yordamına kadar aşağıdaki yordamları yürüt. "Koşul" YANLIŞ veya 0 ise, END_IF yordamına kadar atla (program opsiyonel ELSIF veya ELSE yordamları içermediği sürece).
ELSIF "Koşul-n" THEN **Yordam_N;** **;**	Opsiyonel ELSEIF[1] yordamı, değerlendirilmesi gereken başka koşullar sunar. Örnek: IF-THEN yordamındaki "Koşul" YANLIŞ ise, program, "Koşul-n"yi değerlendirir. "Koşul-n" DOĞRU ise, "Yordam_N"yi yürüt.
ELSE **Yordam_X;** **;**	Opsiyonel ELSE yordamı, IF-THEN yordamının "Koşulu" YANLIŞ ise yürütülmesi gereken yordamlar sunar.
END_IF;	END_IF yordamı, IF-THEN yordamını sonlandırır.

[1] Bir IF-THEN yordamına birden fazla ELSIF yordamı ekleyebilirsiniz.

Tablo 8-158 IF-THEN yordamının değişkenleri

Değişkenler	Açıklama
"Koşul"	Gerekli. Mantıksal ifade DOĞRU (1) veya YANLIŞ'tır (0).
"Yordam_A"	Opsiyonel. "Koşul"un DOĞRU olması durumunda yürütülmesi gereken bir veya daha fazla yordam.
"Koşul-n"	Opsiyonel. Opsiyonel ELSIF yordamı tarafından değerlendirilmesi gereken mantıksal ifade.
"Yordam_N"	Opsiyonel. ELSIF yordamına ait "Koşul-n"nin DOĞRU olması durumunda yürütülmesi gereken bir veya daha fazla yordam.
"Yordam_X"	Opsiyonel. IF-THEN yordamına ait "Koşul"un YANLIŞ olması durumunda yürütülmesi gereken bir veya daha fazla yordam.

Bir IF yordamı, aşağıdaki kurallara uygun şekilde yürütülür:

- Mantıksal ifadesi = DOĞRU olan ilk yordam dizisi yürütülür. Diğer yordam dizileri yürütülmez.
- Hiçbir Boole ifadesi = DOĞRU değilse, ELSE tarafından sağlanan yordam dizisi yürütülür (veya hiçbir ELSE dalı mevcut değilse hiçbir yordam dizisi yürütülmez).
- ELSIF yordamlarının sayısı sınırlanmamıştır.

NOT

Bir veya daha fazla ELSIF dalının kullanımı, bir dizi IF yordamına göre, geçerli bir ifadeyi takip eden mantıksal ifadelerin artık değerlendirilmemesi gibi bir avantaja sahiptir. Bu sayede bir programın çalışma zamanı kısaltılabilir.

8.8.10.2 CASE

Tablo 8-159 CASE yordamının elemanları

SCL	Açıklama
CASE "Test değeri" OF **"Değer listesi": Yordam[; Yordam, ...]** **"Değer listesi": Yordam[; Yordam, ...]** **ELSE** **Else yordamı[; Else yordamı, ...]** **END_CASE;**	CASE yordamı, bir ifadenin değerine bağlı, birden fazla yordam grubundan birini yürütür. Her test durumu için diğer yordamlar ve diğer Else yordamları opsiyoneldir.

Tablo 8-160 Parametre

Parametre	Açıklama
"Test değeri"	Test edilecek değer
"Değer listesi"	Gerekli. Münferit bir değer veya virgülle ayrılmış değerler veya değer aralıkları içeren bir liste. (Bir değer aralığını iki nokta aracılığıyla belirtebilirsiniz: 2..8). Aşağıdaki örnekte, değer listelerinin çeşitli varyantları gösterilmektedir: 1: Yordam_A; 2, 4: Yordam_B; 3, 5..7,9: Yordam_C;
Yordam	Gerekli. "Test değerinin" değer listesindeki bir değere karşılık gelmesi durumunda yürütülen bir veya daha fazla yordam.
Else yordamı	Opsiyonel. "Değer listesinin" hiçbir değerinin eşleşmemesi durumunda yürütülen bir veya daha fazla yordam.

Bir CASE yordamı, aşağıdaki kurallara uygun şekilde yürütülür:

- Program, test değeri ifadesinin değerinin, belirtilen bir değer listesinde mevcut olup olmadığını kontrol eder. Bir eşleşme bulunursa, listeye atanmış olan yordam bileşenleri yürütülür.
- Hiçbir eşleşme bulunamazsa, ELSE'i takip eden program alt bölümü yürütülür. Hiçbir ELSE dalı mevcut değilse hiçbir yordam yürütülmez.

Örnek: İç içe geçirilmiş CASE yordamları

CASE yordamları iç içe geçirebilir. İç içe geçirilmiş her CASE yordamı, ilgili bir END_CASE yordamına ihtiyaç duyar.

```
CASE "var1" OF
       1 : #var2 := 'A';
       2 : #var2 := 'B';
ELSE
       CASE "var3" OF

          65..90: #var2 := ,Büyük harf';
          97..122: #var2 := ,Küçük harf';
       ELSE

          #var2:= 'Özel karakter';
       END_CASE;
END_CASE;
```

8.8.10.3 FOR

Tablo 8-161 FOR yordamının elemanları

SCL	Açıklama
FOR "Kumanda değişkeni" := "Başlangıç" TO "Bitiş" BY "Artım" DO Yordam; ; END_FOR;	Bir FOR yordamı, bir kumanda değişkeni belirtilen değer aralığının içinde olduğu sürece bir yordam dizisinin tekrarlanmasını sağlar. FOR içeren bir döngünün tanımı, bir başlangıç ve bir bitiş değeri bilgisini kapsar. Her iki değer de kumanda değişkeniyle aynı tipe sahip olmalıdır. FOR döngülerini iç içe geçirebilirsiniz. END_FOR yordamı, en son yürütülen FOR yordamıyla ilgilidir. BY koşulu opsiyoneldir.

Tablo 8-162 Parametre

Parametre	Açıklama
"Kumanda değişkeni"	Gerekli. Döngü sayacı olarak görev gören bir tamsayı
"Başlangıç"	Gerekli. Kumanda değişkenlerinin başlangıç değerini belirten basit ifade.
"Bitiş"	Gerekli. Kumanda değişkenlerinin bitiş değerini belirten basit ifade.
"Artım"	Opsiyonel. Bir "kumanda değişkeninin" her döngüden sonra değiştiği miktar. "Artım", "kumanda değişkeni" ile aynı veri tipine sahiptir. "Artım" değeri belirtilmediyse, değişkenlerin değeri, her döngüden sonra 1 arttırılır. "Artım" değerini, FOR yordamı yürütülürken değiştiremezsiniz.

FOR yordamı aşağıdaki şekilde yürütülür:

- Döngünün başlangıcında kumanda değişkeni başlangıç değerine ayarlanır (başlangıç ataması) ve her döngü geçişinde, bitiş değerine ulaşılana kadar, belirtilen artım değeri oranında arttırılır (pozitif artım) veya azaltılır (negatif artım).
- Her döngü geçişinden sonra, koşulun (bitiş değerine ulaşıldı) sağlanıp sağlanmadığı kontrol edilir. Bitiş koşulu sağlanmadıysa, yordam sekansı yeniden yürütülür; aksi takdirde döngü sona erer ve yürütme işlemine, döngünün hemen arkasındaki yordamla devam edilir.

BY [Artım] yordamını kullanmayabilirsiniz. Bir artım değeri belirtilmediyse, artım değeri +1 olur.

Döngüyü, "Koşul" ifadesinin durumundan bağımsız olarak sonlandırmak için EXIT (Sayfa 302)'u kullanın. EXIT yordamı, END_FOR yordamının hemen arkasından gelen yordamı yürütür.

CONTINUE (Sayfa 301) yordamıyla, bir FOR döngüsünün sonraki yordamlarına atlayabilir ve döngüyü, sonlandırma koşulunun sağlanıp sağlanmadığını kontrol ederek devam ettirebilirsiniz.

8.8.10.4 WHILE-DO

Tablo 8-163 WHILE yordamı

SCL	Açıklama
WHILE "Koşul" DO **Yordam;** **Yordam;** **...;** **END_WHILE;**	WHILE yordamı, önceden belirtilen bir koşul DOĞRU olana kadar bir yordam dizisini yürütür. WHILE döngülerini iç içe geçirebilirsiniz. END_WHILE yordamı, en son yürütülen WHILE yordamıyla ilgilidir.

Tablo 8-164 Parametre

Parametre	Açıklama
"Koşul"	Değerlendirmesi DOĞRU veya YANLIŞ olarak sonuçlanan bir mantıksal ifade. ("Sıfır" koşulu YANLIŞ olarak değerlendirilir.)
Yordam	Koşul DOĞRU olana kadar yürütülen bir veya daha fazla yordam.

NOT

WHILE yordamı, yordamlar yürütülmeden önce "Koşulun" durumunu değerlendirir. Yordamları, "Koşul"un durumundan bağımsız olarak en az bir kez yürütmek için REPEAT yordamını (Sayfa 300) kullanın.

WHILE yordamı, aşağıdaki kurallara uygun şekilde yürütülür:

- Döngü içeriğinin her yürütülüşünden önce yürütme koşulu değerlendirilir.
- DO'yu takip eden döngü içeriği, yürütme koşulu DOĞRU değerini alana kadar tekrarlanır.
- Koşul YANLIŞ ise, döngü atlanır ve döngüyü takip eden yordam yürütülür.

Döngüyü, "Koşul" ifadesinin durumundan bağımsız olarak sonlandırmak için EXIT (Sayfa 302)'u kullanın. EXIT yordamı, END_WHILE yordamının hemen arkasından gelen yordamı yürütür.
CONTINUE (Sayfa 301) yordamıyla, bir WHILE döngüsünün sonraki yordamlarına atlayabilir ve döngüyü, sonlandırma koşulunun sağlanıp sağlanmadığını kontrol ederek devam ettirebilirsiniz.

8.8.10.5 REPEAT-UNTIL

Tablo 8-165 REPEAT yordamı

SCL	Açıklama
REPEAT **Yordam;** **;** **UNTIL "Koşul"** **END_REPEAT;**	REPEAT yordamı, önceden belirtilen bir koşul DOĞRU olana kadar bir yordam grubunu yürütür. REPEAT döngülerini iç içe geçirebilirsiniz. END_REPEAT yordamı, her zaman en son yürütülen REPEAT yordamıyla ilgilidir.

Tablo 8-166 Parametre

Parametre	Açıklama
Yordam	Koşul DOĞRU olana kadar yürütülen bir veya daha fazla yordam.
"Koşul"	Aşağıdaki iki türde bir veya daha fazla ifade: Değerlendirmesi DOĞRU veya YANLIŞ olarak sonuçlanan bir sayısal ifade veya bir karakter dizisi ifadesi. "Sıfır" koşulu YANLIŞ olarak değerlendirilir.

NOT

REPEAT yordamı, "Koşulun" durumu değerlendirilmeden önce döngünün ilk uygulanışı sırasında yordamları yürütür ("Koşul" YANLIŞ olsa bile). "Koşulun" durumunu yordamlar yürütülmeden önce kontrol etmek için WHILE yordamını (Sayfa 299) kullanın.

Döngüyü, "Koşul" ifadesinin durumundan bağımsız olarak sonlandırmak için EXIT (Sayfa 302)'u kullanın. EXIT yordamı, END_REPEAT yordamının hemen arkasından gelen yordamı yürütür.

CONTINUE (Sayfa 301), bir REPEAT döngüsünün sonraki yordamlarına atlamanıza ve döngüyü, sonlandırma koşulunun sağlanıp sağlanmadığını kontrol ederek devam ettirmenize olanak sağlar.

8.8.10.6 CONTINUE

Tablo 8-167 CONTINUE yordamı

SCL	Açıklama
CONTINUE **Yordam;** **;**	CONTINUE yordamı, bir program döngüsünü sonraki yordamlarına atlar (FOR, WHILE, REPEAT) ve döngüyü, sonlandırma koşulunun sağlanıp sağlanmadığını kontrol ederek devam ettirir. Bu durum söz konusu değilse, döngü devam ettirilir.

CONTINUE yordamı, aşağıdaki kurallara uygun şekilde yürütülür:

- Bu yordam, döngü içeriğinin yürütülmesini derhal sonlandırır.
- Döngünün tekrarlanması için tanımlanan koşulun sağlanıp sağlanmadığına bağlı olarak, döngü içeriği yeniden yürütülür veya tekrarlama yordamı sonlandırılır ve hemen arkadan gelen yordam yürütülür.
- Bir FOR yordamında, kumanda değişkeni bir CONTINUE yordamının hemen arkasından belirtilen artım değeri kadar arttırılır.

CONTINUE yordamını sadece bir döngünün içinde kullanın. İç içe geçirilmiş döngülerde CONTINUE yordamı, her zaman yordamın bulunduğu döngüyle ilişkilidir. CONTINUE, genelde bir IF yordamıyla bağlantılı olarak kullanılır.

Döngünün, sonlandırma kontrolünden bağımsız olarak sonlandırılması gerekiyorsa EXIT yordamını kullanın.

Örnek: CONTINUE yordamı

Aşağıdaki örnekte, bir yüzde değeri hesaplanırken, "0'a bölme" hatasını önlemesi gereken CONTINUE yordamının kullanımı gösterilmektedir:

```
FOR i := 0 TO 10 DO
IF Değer[i] = 0 THEN CONTINUE; END_IF;
    p := Bölüm / Değer[i] * 100;
    s := INT_TO_STRING(p);
     Yüzde := CONCAT(IN1:=s, IN2:="%");
END_FOR;
```

8.8.10.7 EXIT

Tablo 8-168 EXIT yordamı

SCL	Açıklama
EXIT;	EXIT yordamı, bir döngüyü (FOR, WHILE veya REPEAT) herhangi bir konumda ve sonlandırma koşulunun sağlanmasından bağımsız olarak sonlandırmak için kullanılır.

EXIT yordamı, aşağıdaki kurallara uygun şekilde yürütülür:

- Bu yordam, EXIT yordamını doğrudan çevreleyen tekrarlama yordamının hemen sonlandırılmasına neden olur.
- Programın yürütülmesine, döngünün sonundan sonra (örn. END_FOR'dan sonra) devam edilir.

EXIT yordamını bir döngünün içinde kullanın. İç içe geçirilmiş döngülerde EXIT yordamı, işlemeyi bir üstteki gömme seviyesine döndürür.

Örnek: EXIT yordamı

```
FOR i := 0 TO 10 DO
CASE Değer[i, 0] OF
    1..10: Değer [i, 1]:="A";
    11..40: Değer [i, 1]:="B";
    41..100: Değer [i, 1]:="C";
ELSE
EXIT;
END_CASE;
END_FOR;
```

8.8.10.8 GOTO

Tablo 8-169 GOTO yordamı

SCL	Açıklama
GOTO JumpLabel; **Yordam;** **... ;** **JumpLabel: Yordam;**	GOTO yordamı, aynı bloktaki bir atlama işaretine atlayarak yordamları atlar. Atlama işareti ("JumpLabel") ve GOTO yordamı aynı blokta olmak zorundadır. Bir atlama işaretinin adı, bir blok içinde sadece bir kez atanabilir. Her atlama işareti, çeşitli GOTO yordamlarının hedefi olabilir.

Bir döngü alt bölümüne atlamak mümkün değildir (FOR, WHILE veya REPEAT). Bir döngüden dışarıya atlamak mümkündür.

Örnek: GOTO yordamı

Aşağıdaki örnekte şunlar geçerlidir: "Değişken değeri" işlemcisinin değerine bağlı olarak, programın yürütülmesine, ilgili atlama işareti tarafından belirtilen noktadan devam edilir. "Değişken değeri" 2'ye eşitse, program, "MyLabel2" atlama işaretinde yürütülmeye devam eder ve "MyLabel1"e atlar.

```
CASE "Değişken değeri" OF
1 : GOTO Etiketim1;
2 : GOTO Etiketim2;
ELSE GOTO Etiketim3;
END_CASE;

Etiketim1: "Değişken_1" := 1;

Etiketim2: "Değişken_2" := 1;

Etiketim3: "Değişken_4" := 1;
```

8.8.10.9 RETURN

Tablo 8-170 RETURN yordamı

SCL	Açıklama
RETURN;	RETURN yordamı, mevcut durumda yürütülen kod bloğunu koşullar olmaksızın sonlandırır. Programın yürütülmesinde, çağıran bloğa veya işletim sistemine dönülür (bir OB sonlandırılırken).

Örnek: RETURN yordamı:

```
IF "Hata" <> 0 THEN
RETURN;
END_IF;
```

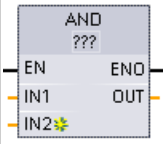
NOT

Kod bloğu, son yordam yürütüldükten sonra otomatik olarak çağıran bloğa döner. Bir RETURN yordamını, kod bloğunun sonuna eklemeyin.

8.9 Sözcük bağlantısı

8.9.1 AND (VE), OR (VEYA) ve XOR (ÖZEL VEYA) bağlantı işlemleri

Tablo 8-171 AND (VE), OR (VEYA) ve XOR (ÖZEL VEYA) bağlantı işlemleri

LAD/FBD	SCL	Açıklama
	**out := in1 AND in2;**	AND: VE işlemi
	out := in1 OR in2;	OR: VEYA işlemi
	out := in1 XOR in2;	XOR: ÖZEL VEYA bağlantısı

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.



Bir giriş eklemek için "Oluştur" sembolüne tıklayın veya mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına tıklayın ve "Giriş ekle" komutunu seçin.

Bir girişi silmek için, mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına (başlangıçtaki iki girişten daha fazla giriş olduğu sürece) tıklayın ve "Sil" komutunu seçin.

Tablo 8-172 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN1, IN2	Byte, Word, DWord	Bağlantı girişleri
OUT	Byte, Word, DWord	Bağlantı çıkışı

[1] Veri tipinin seçilmesi yoluyla IN1, IN2 ve OUT parametreleri aynı veri tipine ayarlanır.

IN1 ve IN2'nin ilgili Bit değerleri, OUT parametresinde bir ikili sonuca bağlanır. ENO, bu yordamlar yürütüldükten sonra her zaman DOĞRU'dur.

8.9.2 INV (tekli tümleyen oluştur)

Tablo 8-173 INV yordamı

LAD/FBD	SCL	Açıklama
	**Mevcut değil**	IN parametresinin ikili tekli tümleyenini hesaplar. Tekli tümleyen, IN parametresinin her Bit değerinin ters çevrilmesi (0'ın 1 olarak ve 1'in 0 olarak değiştirilmesi) yoluyla oluşturulur. ENO, bu yordam yürütüldükten sonra her zaman DOĞRU'dur.

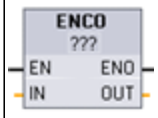
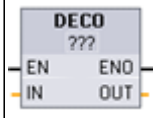
[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-174 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord	Ters çevirme işlemi için kullanılan veri elemanı
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord	Tersine çevrilmiş çıkış değeri

8.9.3 DECO (şifreyi çöz) ve ENCO (şifrele) yordamları

Tablo 8-175 ENCO ve DECO yordamları

LAD/FBD	SCL	Açıklama
	**out := ENCO(_in_);**	Bir Bit örüntüsünü bir ikili sayıya dönüştürür. ENCO yordamı, IN parametresini, IN parametresinin en düşük değerli Bit'inin Bit pozisyonuna karşılık gelen ikili sayıya dönüştürür ve sonucu, OUT parametresinde yayınlar. IN parametresi 0000 0001 veya 0000 0000 ise, OUT parametresinde 0 değeri döndürülür. IN parametresinin değeri 0000 0000'a eşitse, ENO YANLIŞ olarak ayarlanır.
	**out := DECO(_in_);**	Bir ikili sayıyı bir Bit örüntüsüne dönüştürür. DECO yordamı, IN parametresinden bir ikili sayıyı, karşılık gelen Bit pozisyonu OUT parametresinde 1'e ayarlanarak (diğer tüm Bit'ler 0'a ayarlanır) dönüştürür. ENO, DECO yordamı yürütüldükten sonra her zaman DOĞRU'dur. Not: DECO yordamı için kullanılan standart veri tipi DWORD'tür. SCL'de, bir Bayt veya Word değerinin şifresini çözmek için yordam adını DECO_BYTE veya DECO_WORD olarak değiştirebilir ve bir Bayt veya sözcük değişkeni veya adresi atayabilirsiniz.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 8-176 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	ENCO: Byte, Word, DWord DECO: UInt	ENCO: Dönüştürme için kullanılan Bit örüntüsü DECO: Dönüştürme için kullanılan ikili değer
OUT	ENCO: Int DECO: Byte, Word, DWord	ENCO: Dönüştürülmüş değer DECO: Dönüştürülmüş Bit örüntüsü

Tablo 8-177 ENO'nun durumu

ENO	Koşul	Sonuç (OUT)
1	Hata yok	Geçerli Bit numarası
0	IN sıfırdır	OUT, sıfıra ayarlanır

DECO parametresi OUT için kullanılan olası Byte, Word veya DWord veri tipleri, IN parametresinin kullanılabilir aralığını sınırlar. IN parametresinin değeri kullanılabilir aralığı aşıyorsa, en düşük değerli Bit'lerin kökünü almak için bir Modulo işlemi gerçekleştirilir (bkz. alt kısım).

DECO parametresi IN için kullanılan aralık:

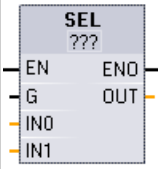
- Bir Byte OUT'taki 1 Bit pozisyonunu ayarlamak için 3 Bit (0-7 aralığındaki değerler) IN kullanılır
- Bir Word OUT'taki 1 Bit pozisyonunu ayarlamak için 4 Bit (0-15 aralığındaki değerler) IN kullanılır
- Bir DWord OUT'taki 1 Bit pozisyonunu ayarlamak için 5 Bit (0-31 aralığındaki değerler) IN kullanılır

Tablo 8-178 Örnekler

DECO için kullanılan IN değeri			DECO için kullanılan OUT değeri (münferit Bit pozisyonunu değiştirme)
Byte OUT 8 Bit	Min. IN	0	00000001
	Maks. IN	7	10000000
Word OUT 16 Bit	Min. IN	0	0000000000000001
	Maks. IN	15	1000000000000000
DWord OUT 32 Bit	Min. IN	0	00000000000000000000000000000001
	Maks. IN	31	10000000000000000000000000000000

8.9.4 SEL (seç), MUX (çokla) ve DEMUX (çoklamayı çöz) yordamları

Tablo 8-179 SEL (seç) yordamı

LAD/FBD	SCL	Açıklama
	**out := SEL(** **g:=_bool_in,** **in0:-_variant_in,** **in1:=_variant_in);**	SEL yordamı, iki giriş değerinden birini, G parametresinin değerine bağlı olarak OUT parametresine atar.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

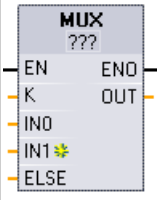
Tablo 8-180 SEL yordamı için kullanılan veri tipleri

Parametre	Veri tipi[1]	Açıklama
G	Bool	• 0, IN0'ı seçer • 1, IN1'i seçer
IN0, IN1	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Girişler
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Çıkış

[1] Giriş değişkenleri ve çıkış değişkeni aynı veri tipinde olmalıdır.

Koşul kodları: ENO, SEL yordamı yürütüldükten sonra her zaman DOĞRU'dur.

Tablo 8-181 MUX (çokla) yordamı

LAD/FBD	SCL	Açıklama
	**out := MUX(** **k:=_unit_in,** **in1:=variant_in,** **in2:=variant_in,** **[...in32:=variant_in,]** **inelse:=variant_in);**	MUX yordamı, birçok giriş değerinden birini, K parametresinin değerine bağlı olarak OUT parametresine kopyalar. K parametresinin değeri (IN*n* - 1)'i aşıyorsa, ELSE parametresinin değeri, OUT parametresine kopyalanır.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.



Bir giriş eklemek için "Oluştur" sembolüne tıklayın veya mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına tıklayın ve "Giriş ekle" komutunu seçin.

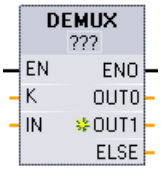
Bir girişi silmek için, mevcut IN parametrelerinden birinde farenin sağ tuşuyla giriş bağlantısına (başlangıçtaki iki girişten daha fazla giriş olduğu sürece) tıklayın ve "Sil" komutunu seçin.

Tablo 8-182 MUX yordamı için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
K	UInt	• 0, IN1'i seçer • 1, IN2'yi seçer • *n*, IN*n*'yi seçer
IN0, IN1, .. INn	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Girişler
ELSE	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Yedek değer girişi (opsiyonel)
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Çıkış

[1] Giriş değişkenleri ve çıkış değişkeni aynı veri tipinde olmalıdır.

Tablo 8-183 DEMUX (çoklamayı çöz) yordamı

LAD/FBD	SCL	Açıklama
	**DEMUX(** **k:=_unit_in,** **in:=variant_in,** **out1:=variant_in,** **out2:=variant_in,** **[...out32:=variant_in,]** **outelse:=variant_in);**	DEMUX, IN parametresine atanmış olan adresin değerini birçok çıkıştan birine kopyalar. K parametresinin değeri, IN değerinin hedefi olarak hangi çıkışın seçileceğini belirtir. K'nın değeri (OUT*n* - 1) sayısından büyükse, IN değeri, ELSE parametresine atanmış olan adrese kopyalanır.

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.



Bir çıkış eklemek için "Oluştur" sembolüne tıklayın veya mevcut OUT parametrelerinden birinde farenin sağ tuşuyla çıkış bağlantısına tıklayın ve "Çıkış ekle" komutunu seçin.

Bir çıkışı silmek için, mevcut OUT parametrelerinden birinde farenin sağ tuşuyla çıkış bağlantısına (başlangıçtaki iki çıkıştan daha fazla çıkış olduğu sürece) tıklayın ve "Sil" komutunu seçin.

Tablo 8-184 DEMUX yordamı için kullanılan veri tipleri

Parametre	**Veri tipi**[1]	**Açıklama**
K	UInt	Seçim değeri: • 0, OUT1'i seçer • 1, OUT2'yi seçer • n, OUTn'yi seçer
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Giriş
OUT0, OUT1, .. OUTn	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	Çıkışlar
ELSE	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Time, Date, TOD, Char, WChar	K değeri (OUTn - 1)'den büyükse çıkışı değiştirin

[1] Giriş değişkeni ve çıkış değişkenleri aynı veri tipinde olmalıdır.

Tablo 8-185 MUX ve DEMUX yordamları için ENO durumu

ENO	**Koşul**	**Sonuç OUT**
1	Hata yok	MUX: Seçilen IN değeri, OUT'a kopyalanır DEMUX: IN değeri, seçilen OUT'a kopyalanır
0	MUX: K, giriş sayısı -1 değerinden büyüktür	• Hiçbir ELSE ifadesi belirtilmedi: OUT değeri değiştirilmez, • ELSE belirtildi, ELSE değeri OUT'a atanır
	DEMUX: K, çıkış sayısı -1 değerinden büyüktür	• Hiçbir ELSE ifadesi belirtilmedi: Çıkışlar değiştirilmez, • ELSE belirtildi, IN değeri ELSE'e kopyalanır

8.10 Kaydır ve döndür

8.10.1 SHR (sağa kaydır) ve SHL (sola kaydır) yordamları

Tablo 8-186 SHR ve SHL yordamları

LAD/FBD	SCL	Açıklama
	**out := SHR(** **in:=_variant_in_,** **n:=_uint_in);** **out := SHL(** **in:=_variant_in_,** **n:=_uint_in);**	Kaydırma yordamları (SHL ve SHR) ile IN parametresinin Bit örüntüsünü kaydırabilirsiniz. Sonuç, OUT parametresine atanır. N parametresi, kaydırılan Bit pozisyonlarının sayısını belirtir: • SHR: Bit örüntüsünü sağa kaydır • SHL: Bit örüntüsünü sola kaydır

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tiplerini seçin.

Tablo 8-187 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	Tamsayılar	Kaydırılması gereken Bit örüntüsü
N	USInt, UDint	Kaydırılması gereken Bit pozisyonlarının sayısı
OUT	Tamsayılar	Kaydırma işleminden sonra Bit örüntüsü

- N=0 olduğunda kaydırma işlemi gerçekleştirilmez. IN değeri OUT'a atanır.
- Kaydırma yordamı tarafından boşaltılan Bit pozisyonlarına sıfırlar kaydırılır.
- Kaydırılması gereken pozisyonların sayısı (N) hedef değerdeki Bit sayısını aşıyorsa (Byte'ta 8, Word'de 16, DWord'de 32), başlangıçtaki tüm Bit değerleri dışarıya doğru kaydırılır ve sıfırlarla değiştirilir (OUT'a sıfır atanır).
- ENO, kaydırma yordamlarında her zaman DOĞRU'dur.

Tablo 8-188 Örnek: Word verileriyle birlikte SHL

Bir sözcüğün Bit'lerini, sağdan sıfırların eklenmesiyle (N = 1) sola doğru kaydırır			
IN	1110 0010 1010 1101	İlk kaydırma işleminden önce OUT değeri:	1110 0010 1010 1101
		İlk sola kaydırma işleminden sonra:	1100 0101 0101 1010
		İkinci sola kaydırma işleminden sonra:	1000 1010 1011 0100
		Üçüncü sola kaydırma işleminden sonra:	0001 0101 0110 1000

8.10.2 ROR (sağa döndür) ve ROL (sola döndür) yordamları

Tablo 8-189 ROR ve ROL yordamları

LAD/FBD	SCL	Açıklama
	**out := ROL(** **in:=_variant_in_,** **n:=_uint_in);** **out := ROR(** **in:=_variant_in_,** **n:=_uint_in);**	Döndürme yordamları (ROR ve ROL) ile IN parametresinin Bit örüntüsünü döndürebilirsiniz. Sonuç, OUT parametresine atanır. N parametresi, döndürülen Bit pozisyonlarının sayısını belirler. • ROR: Bit örüntüsünü sağa döndür • ROL: Bit örüntüsünü sola döndür

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tiplerini seçin.

Tablo 8-190 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	Tamsayılar	Döndürülmesi gereken Bit örüntüsü
N	USInt, UDint	Döndürülmesi gereken Bit pozisyonlarının sayısı
OUT	Tamsayılar	Döndürme işleminden sonra Bit örüntüsü

- N=0 olduğunda döndürme işlemi gerçekleştirilmez. IN değeri OUT'a atanır.
- Hedef değerin bir tarafında dışa doğru döndürülen Bit verileri, hedef değerin diğer tarafında, başlangıçtaki hiçbir Bit değeri kaybolmayacak şekilde içe doğru döndürülür.
- Döndürülmesi gereken Bit pozisyonlarının sayısı (N) hedef değerdeki Bit sayısını aşıyorsa (Byte'ta 8, Word'de 16, DWord'de 32), döndürme işlemi yine de gerçekleştirilir.
- ENO, döndürme yordamları yürütüldükten sonra her zaman DOĞRU'dur.

Tablo 8-191 Örnek: Word verileriyle birlikte ROR

Bit'lerin sağa, dışa doğru ve sola, içe doğru döndürülmesi (N = 1)			
IN	0100 0000 0000 0001	İlk döndürme işleminden önce OUT değeri:	0100 0000 0000 0001
		İlk sağa döndürme işleminden sonra:	1010 0000 0000 0000
		İkinci sağa döndürme işleminden sonra:	0101 0000 0000 0000

Gelişmiş yordamlar

9

9.1 Tarih, saat ve saat fonksiyonları

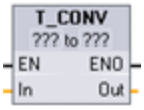
9.1.1 Tarih ve saat yordamları

Tarih ve saat yordamlarını takvim ve zaman hesaplamaları için kullanın.

- T_CONV, bir değeri (tarih ve zaman veri tiplerine veya bu veri tiplerinden) ve (Bayt, sözcük ve çift sözcük veri tiplerine ve bu veri tiplerinden dönüştürür).
- T_ADD, Time ve DTL değerlerini toplar: (Time + Time = Time) veya (DTL + Time = DTL)
- T_SUB, Time ve DTL değerlerini çıkarır: (Time - Time = Time) veya (DTL - Time = DTL)
- T_DIFF, iki DTL değerinin farkını Time değeri olarak yayınlar: DTL - DTL = Time
- T_COMBINE, bir Date değerini ve bir Time_and_Date değerini, bir DTL değeri elde etmek üzere bağlar

DTL ve Time verilerinin formatıyla ilgili bilgileri, Saat ve tarih veri tipleri (Sayfa 102) ile ilgili alt bölümde bulabilirsiniz.

Tablo 9-1 T_CONV (zamanları dönüştür ve ayrıştır) yordamı

LAD/FBD	SCL ile ilgili örnek	Açıklama
	**out := DINT_TO_TIME(in:=_variant_in);** **out := TIME_TO_DINT(in:=_variant_in);**	T_CONV, bir değeri (tarih ve zaman veri tiplerine veya bu veri tiplerinden) ve (Bayt, sözcük ve çift sözcük veri tiplerine ve bu veri tiplerinden dönüştürür).

[1] LAD ve FBD kutularında: "???" öğesine tıklayın ve açılır listeden kaynak ve hedef için kullanılan veri tiplerini seçin.

[2] SCL'de: T_CONV yordamını, yordam dizininden programlama editörünün üzerine sürükleyin. Ardından kaynak ve hedef için kullanılan veri tiplerini seçin.

Tablo 9-2 T_CONV dönüştürme işlemleri için geçerli veri tipleri

IN (veya OUT) veri tipi	OUT (veya IN) veri tipleri
TIME (milisaniye)	DInt, Int, SInt, UDInt, UInt, USInt, TOD Sadece SCL'de: Byte, Word, Dword
DATE (1 Ocak 1990'dan beri geçen gün sayısı)	DInt, Int, SInt, UDInt, UInt, USInt, DTL Sadece SCL'de: Byte, Word, Dword
TOD (Gece yarısı saat 24:00:00.000'dan itibaren geçen milisaniye sayısı)	DInt, Int, SInt, UDInt, UInt, USInt, TIME, DTL Sadece SCL'de: Byte, Word, Dword

NOT

T_CONV ile daha büyük bir veri boyutunu daha küçük bir veri boyutuna dönüştürme

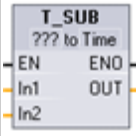
Daha fazla Bayt içeren daha büyük bir veri boyutunu daha az Bayt içeren daha küçük bir veri tipine dönüştürürseniz, veri değerleri bölümlere ayrılabilir. Bu hata oluşursa, ENO, 0'a ayarlanır.

DTL veri tipinden/veri tipine dönüştürme

DTL (Date and Time Long) veri tipi; yıl, ay, tarih ve saat verilerini içerir. DTL verileri; DATE ve TOD veri tipine/veri tipinden dönüştürülebilir.
Ancak sadece DATE veri tipiyle gerçekleştirilen DTL dönüştürme işlemi, sadece yıl, ay ve gün değerlerini etkiler. TOD verileriyle gerçekleştirilen bir DTL dönüştürme işlemi, sadece saat, dakika ve saniye değerlerini etkiler.

T_CONV yordamı DTL'ye dönüştürürse, DTL formatındaki etkilenmeyen veri elemanları değişmez.

Tablo 9-3 T_ADD (zamanları topla) ve T_SUB (zamanları çıkar) yordamları

LAD/FBD	SCL	Açıklama
	**out := T_ADD(in1:=_variant_in, in2:=_time_in);**	T_ADD yordamı, IN1 girişindeki değeri (DTL veya Time veri tipi) IN2 girişindeki değerle (Time veri tipi) toplar. OUT parametresi, sonucu DTL veya Time veri tipinde bir değer olarak sağlar. İki veri tipi yordamı kullanılabilir: • Time + Time = Time • DTL + Time = DTL
	**out := T_SUB(in1:=_variant_in, in2:=_time_in);**	T_SUB yordamı, IN2 girişindeki değeri (Time veri tipi) IN1'nin değerinden (DTL veya Time veri tipi) çıkarır. OUT parametresi, fark değerini DTL veya Time veri tipi olarak sağlar. İki veri tipi yordamı kullanılabilir. • Time - Time = Time • DTL - Time = DTL

[1] LAD ve FBD'de: "???" öğesine tıklayın ve açılır listeden veri tiplerini seçin.

Tablo 9-4 T_ADD ve T_SUB parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1[1]	IN	DTL, Time	DTL veya Time değeri
IN2	IN	Time	Toplanması veya çıkarılması gereken Time değeri
OUT	OUT	DTL, Time	DTL veya Time toplamı veya farkı

[1] Yordamın adının altındaki açılır listeden IN1 için kullanılan veri tipini seçin. IN1 için kullanılan veri tipinin seçilmesi yoluyla, OUT parametresi için kullanılan veri tipi de ayarlanır.

Tablo 9-5 T_DIFF (zaman farkı) yordamı

LAD/FBD	SCL	Açıklama
	**out := T_DIFF(in1:=_DTL_in, in2:=_DTL_in);**	T_DIFF yordamı, DTL değerini (IN2) DTL değerinden (IN1) çıkarır. OUT parametresi, fark değerini Time veri tipi olarak sağlar. • DTL - DTL = Time

Tablo 9-6 T_DIFF parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1	IN	DTL	DTL değeri
IN2	IN	DTL	Çıkarılması gereken DTL değeri
OUT	OUT	Time	Time farkı

Koşul kodları: ENO = 1, hiçbir hatanın oluşmadığı anlamına gelir. ENO = 0 ve OUT parametresi = 0 hata:

- Geçersiz DTL değeri
- Geçersiz Time değeri

Tablo 9-7 T_COMBINE (zamanları bağla) yordamı

LAD/FBD	SCL	Açıklama
	**out := CONCAT_DATE_TOD(In1 := _date_in, In2 := _tod_in);**	T_COMBINE yordamı, bir Date değerini ve bir Time_of_Day değerini, bir DTL değeri elde etmek üzere bağlar.

[1] Gelişmiş yordamların içinde yer alan T_COMBINE yordamının, SCL'deki CONCAT_DATE_TOD fonksiyonuna karşılık geldiğine dikkat edin.

Tablo 9-8 T_COMBINE parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1	IN	Date	Bağlanması gereken Date değeri, DATE#1990-01-01 ile DATE#2089-12-31 arasında olmalıdır
IN2	IN	Time_of_Day	Bağlanması gereken Time_of_Day değerleri
OUT	OUT	DTL	DTL değeri

9.1.2 Saat fonksiyonları

⚠ İKAZ

NTP senkronizasyonu (Network Time Protocol) nedeniyle ağlarınıza yetkisiz kişilerin erişmesi tehlikesi

Bir saldırgan, NTP senkronizasyonu (Network Time Protocol) üzerinden ağlarınıza erişebilirse, CPU sistem saatini değiştirerek prosesinizin üzerindeki kontrolü elinizden alabilir. Bu tür bir proses kontrolü aktarımı nedeniyle ölüm, ağır yaralanma durumları veya maddi hasarlar oluşabilir.

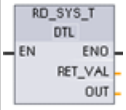
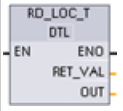
S7-1200 CPU'sunun NTP istemcisi fonksiyonu standart olarak devre dışı bırakılmıştır ve etkinleştirilmiş durumdayken sadece yapılandırılmış IP adreslerinin NTP sunucusu olarak çalışmasına olanak sağlar. CPU, bu fonksiyonu standart olarak devre dışı bırakır ve CPU sistem saatinde uzaktan kontrollü düzeltmelerin yapılmasına izin vermek için bu fonksiyonu yapılandırmanız gerekir.

S7-1200 CPU'su, doğru bir CPU sistem saatine bağlı olan saat alarmlarını ve saat işlemlerini destekler. NTP'yi yapılandırır ve saat senkronizasyonunun bir sunucudan gerçekleştirilmesini kabul ederseniz, sunucuda güvenilir bir kaynağın söz konusu olduğundan emin olmanız gerekir. Aksi takdirde, bilinmeyen bir kullanıcının CPU sistem saatini değiştirebileceği ve bu şekilde prosesinizin üzerindeki kontrolü elinizden alabileceği bir güvenlik ihlali ortaya çıkabilir.

Güvenlikle ilgili bilgileri ve önerileri, Siemens'in "Service & Support" web sitesindeki Operational Guidelines for Industrial Security (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf) bölümümüzde bulabilirsiniz.

Saat yordamlarıyla, CPU'nun sistem saatini ayarlayabilir ve okuyabilirsiniz. DTL (Sayfa 102) veri tipi, tarih ve saat değerlerini sağlar.

Tablo 9-9 Sistem saati için kullanılan yordamlar

LAD/FBD	SCL	Açıklama
	**ret_val := WR_SYS_T(in:=_DTL_in_);**	WR_SYS_T (saati ayarla) yordamı, CPU'daki gerçek zaman saatini, IN parametresinin DTL değeri yardımıyla ayarlar. Bu zaman değeri, ilgili yerel saati ve yaz saati/kış saati geçişini dikkate almaz.
	**ret_val := RD_SYS_T(out=>_DTL_out);**	RD_SYS_T (saati oku) yordamıyla CPU'nun güncel sistem saati okunur. Bu zaman değeri, ilgili yerel saati ve yaz saati/kış saati geçişini dikkate almaz.
	**ret_val := RD_LOC_T(out=>_DTL_out);**	RD_LOC_T (yerel saati oku) yordamı, CPU'nun güncel yerel saatini DTL veri tipi olarak sağlar. Bu saat değeri, yerel zaman dilimine ve yaz saatine/kış saatine (yapılandırıldıysa) uyarlamaya karşılık gelir.

LAD/FBD	SCL	Açıklama
	**ret_val := WR_LOC_T(LOCTIME:=DTL_in_, DST:_in_;**	WR_LOC_T (yerel saati yaz) yordamı, CPU saatinin tarih ve saatini ayarlar. Tarih ve saat bilgilerini, LOCTIME parametresi üzerinden DTL veri tipiyle yerel saat olarak atayabilirsiniz. Yordam, sistem saatini "TimeTransformationRule (Sayfa 316)" DB yapısının yardımıyla hesaplar. Yerel saat ve sistem saati için kullanılan zaman bilgilerinin ayrıntı doğruluğu, ürüne özeldir ve en az bir milisaniye değerindedir. LOCTIME parametresindeki giriş değerleri CPU tarafından desteklenen değerlerden daha küçükse, bu değerler, sistem saati hesaplanırken yuvarlanır. **Not**: "Saat" özelliklerini CPU cihaz konfigürasyonunda ayarlamanız gerekir (zaman dilimi, yaz saatini etkinleştirme, yaz saatinin başlangıcı ve bitişi). Aksi takdirde WR_LOC_T, yaz saatine geçişi yorumlayamaz.

Tablo 9-10 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	DTL	CPU'nun sistem saati için ayarlanması gereken saat
OUT	OUT	DTL	RD_SYS_T: CPU'nun güncel sistem saati RD_LOC_T: Yaz saatine/kış saatine (yapılandırıldıysa) uyarlama da dahil olmak üzere güncel yerel saat
LOCTIME	IN	DTL	WR_LOC_T: Yerel saat
DST	IN	BOOL	WR_LOC_T: **D**aylight **S**aving **T**ime (yaz saati), saat yaz saatine geçerse sadece "çift saat" sırasında değerlendirilir. • TRUE = Yaz saati (ilk saat) • FALSE = Kış saati (ikinci saat)
RET_VAL	OUT	Int	Yürütme koşulu

- Yerel saatin hesaplanması için, cihaz konfigürasyonunun "Genel" sekmesindeki "Saat" ifadesinin altında girdiğiniz zaman dilimi ve yaz saati/kış saati geçişleri kullanılır.
- Zaman dilimi, UTC veya GMT sistem saatini temel alır.
- Yaz saatine geçiş için yaz saati uygulamasına geçilecek ay, hafta, gün ve saat girilmelidir.
- Kış saati geçişi için de kış saati uygulamasına geçilecek ay, hafta, gün ve saat girilmelidir.
- Zaman dilimi farkı her zaman sistem saatiyle ilişkili olarak geçerlidir. Yaz saati kayması sadece yaz saatine geçilmesi durumunda geçerlidir.

NOT

Yaz ve kış saatinin başlangıç zamanının ayarlanması

Cihaz konfigürasyonundaki "Yaz ve kış saatinin başlangıç zamanı" ifadesinin altındaki "Saat" özellikleri için kendi yerel saatinizi belirtmeniz gerekir.

Koşul kodları: ENO = 1, hiçbir hatanın oluşmadığı anlamına gelir. ENO = 0, bir yürütme hatasının oluştuğu ve RET_VAL çıkışında bir koşul kodunun mevcut olduğu anlamına gelir.

RET_VAL (W#16#....)	Açıklama
0000	Güncel yerel saat kış saati olarak belirtilir.
0001	Yaz saati yapılandırıldı ve güncel yerel saat yaz saati olarak belirtilir.
8080	Yerel saat kullanılamıyor veya LOCTIME değeri geçersiz.
8081	LOCTIME parametresi tarafından izin verilmeyen bir yıl değeri atandı veya zaman değeri geçersiz

RET_VAL (W#16#....)	Açıklama
8082	İzin verilmeyen ay değeri (DTL formatında Bayt 2 DTL)
8083	İzin verilmeyen gün değeri (DTL formatında Bayt 3)
8084	İzin verilmeyen saat değeri (DTL formatında Bayt 5 DTL)
8085	İzin verilmeyen dakika değeri (DTL formatında Bayt 6)
8086	İzin verilmeyen saniye değeri (DTL formatında Bayt 7)
8087	İzin verilmeyen nanosaniye değeri (Bayt 8 ila 11 DTL formatında)
8089	Zaman değeri mevcut değil (yaz saatine geçilirken saat geçti)
80B0	Gerçek zaman saati devre dışı kaldı
80B1	"TimeTransformationRule" yapısı tanımlanmadı.

9.1.3 SET_TIMEZONE (zaman dilimini ayarla)

Tablo 9-11 SET_TIMEZONE yordamı

LAD/FBD	SCL	Açıklama
	**"SET_TIMEZONE_DB"(REQ:=_bool_in, Timezone:=_struct_in, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	CPU sistem saatinin ilgili yerel saate dönüştürülmesinde kullanılan yerel zaman dilimi ve yaz saati/kış saati için kullanılan parametreleri ayarlar.

[1] SCL örneğinde "SET_TIMEZONE_DB", örnek DB'nin adıdır.

Tablo 9-12 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQ = 1: Fonksiyonun yürütülmesi
Timezone	IN	TimeTransformationRule	Sistem saatinin yerel saate dönüştürülmesiyle ilgili kurallar
DONE	OUT	Bool	Fonksiyon sonlandı
BUSY	OUT	Bool	Fonksiyon dolu
ERROR	OUT	Bool	Hata algılandı
STATUS	OUT	Word	Fonksiyon sonucu / hata mesajı

CPU için kullanılan zaman dilimi parametrelerini manuel olarak yapılandırmak için, cihaz konfigürasyonunun "Genel" sekmesindeki "Saat" özelliklerini kullanabilirsiniz.

SET_TIMEZONE yordamıyla yerel saati ayarlayabilirsiniz. "TimeTransformationRule" veri yapısının parametreleri, yerel zaman dilimini ve yaz saatiyle kış saati arasında otomatik geçiş için kullanılan zaman ayarlarını atar.

NOT

SET_TIMEZONE yordamının Flash bellek üzerindeki etkisi

SET_TIMEZONE yordamı, Flash bellekte yazma işlemleri gerçekleştirir (dahili yükleme belleğinde veya bir memory card'da). Flash belleğin kullanım ömrünün kısalmasını önlemek için, güncellemelerin daha az sıklıkla yapılması amacıyla SET_TIMEZONE yordamını kullanın.

Koşul kodları: ENO = 1, hiçbir hatanın oluşmadığı anlamına gelir. ENO = 0, bir yürütme hatasının oluştuğu ve STATUS çıkışında bir koşul kodunun mevcut olduğu anlamına gelir.

STATUS (W#16#....)	Açıklama
0	Hata yok
7000	Devam eden bir görev işlemesi yok
7001	Görev işlemesinin başlangıcı. BUSY parametresi = 1, DONE = 0
7002	Ara çağrı (REQ ilgili değil): Yordam zaten aktif; BUSY, "1" değerinde.
808x	x. bileşende hata: Örneğin 8084, DaylightStartWeekif'in 1 ile 5 arasında bir değer olmadığını gösterir.

TimeTransformationRule veri yapısı

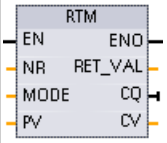
Kış saatiyle yaz saati arasında geçişle ilgili kurallar, TimeTransformationRule veri yapısında belirlenmiştir:

Adı		Veri tipi	Açıklama
TimeTransformationRule		STRUCT	
	Bias	INT	Yerel saat ile UTC arasındaki fark [dakika] Aralık: -1439 ila 1439
	DaylightBias	INT	Yaz saatiyle kış saati arasındaki fark [dakika] Aralık: 0 ila 120
	DaylightStartMonth	USINT	Yaz saatine geçilecek ay Aralık: 1 ila 12
	DaylightStartWeek	USINT	Yaz saatine geçilecek hafta 1 = Ay içinde haftanın gününün ilk görüldüğü zaman, ..., 5 = Ay içinde haftanın gününün son görüldüğü zaman
	DaylightStartWeekday	USINT	Yaz saatine geçilecek gün: 1 = Pazar
	DaylightStartHour	USINT	Yaz saatine geçilecek saat: Aralık: 0 ila 23
	DaylightStartMinute	USINT	Yaz saatine geçilecek dakika: Aralık: 0 ila 59
	StandardStartMonth	USINT	Kış saatine geçilecek ay Aralık: 1 ila 12
	StandardStartWeek	USINT	Kış saatine geçilecek hafta 1 = Ay içinde haftanın gününün ilk görüldüğü zaman, ..., 5 = Ay içinde haftanın gününün son görüldüğü zaman

Adı		Veri tipi	Açıklama
	StandardStartWeekday	USINT	Kış saatine geçilecek gün: 1 = Pazar
	StandardStartHour	USINT	Kış saatine geçilecek saat: Aralık: 0 ila 23
	StandardStartMinute	USINT	Kış saatine geçilecek dakika: Aralık: 0 ila 59
	TimeZoneName	STRING[80]	Zaman diliminin adı: "(GMT+01:00) Amsterdam, Berlin, Bern, Rom, Stockholm, Viyana"

9.1.4 RTM (işletim saati sayacı)

Tablo 9-13 RTM yordamı

LAD/FBD	SCL	Açıklama
	**RTM(NR:=_uint_in_, MODE:=_byte_in_, PV:=_dint_in_, CQ=>_bool_out_, CV=>_dint_out_);**	RTM (işletim saati sayacı) yordamı, CPU'daki işletim saati sayacını ayarlayabilir, başlatabilir, durdurabilir ve okuyabilir.

Tablo 9-14 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
NR	IN	UInt	İşletim saati sayacının numarası: (mümkün olan değerler: 0..9)
MODE	IN	Byte	RTM yürütme modunun numarası: • 0 = Değerleri çağır (daha sonra durum CQ'ya ve güncel değer CV'ye yazılır) • 1 = Başlat (son sayaç değerinden itibaren) • 2 = Durdur • 4 = Ayarla (PV'de belirtilen değere) • 5 = Ayarla (PV'de belirtilen değere) ve sonra başlat • 6 = Ayarla (PV'de belirtilen değere) ve sonra durdur • 7 = CPU'nun tüm RTM değerlerini memory card'a (MC) kaydet
PV	IN	DInt	Belirtilen işletim saati sayacı için saat değerini önceden ayarla
RET_VAL	OUT	Int	Fonksiyon sonucu / hata mesajı
CQ	OUT	Bool	İşletim saati sayacının durumu (1 = çalışıyor)
CV	OUT	DInt	Belirtilen işletim saati sayacı için güncel işletim saati değeri

CPU, kumanda ünitesinin kritik alt sistemlerinin işletim saatlerini takip etmek için maksimum 10 işletim saati sayacı çalıştırır. Münferit işletim saati sayaçlarını sayaç başına bir RTM yordamıyla başlatmanız gerekir. CPU RUN işletim durumundan STOP işletim durumuna geçerse tüm işletim saati sayaçları durdurulur. Münferit sayaçları, RTM yordamının 2. yürütme modu ile de durdurabilirsiniz.

Bir CPU STOP işletim durumundan RUN işletim durumuna geçerse, işletim saati sayaçlarını, sayaç başına bir RTM yordamıyla yeniden başlatmanız gerekir. Bir işletim saati sayacının değeri 2.147.483.647 saatten büyükse, sayma modu sonlandırılır ve bir taşma hatası bildirilir. Bir sayacı sıfırlamak veya değiştirmek için, RTM yordamını sayaç başına bir kez yürütmeniz gerekir.

Bir gerilim kesintisi veya CPU'nun yeniden başlatılması, işletim saati sayaçlarının güncel değerlerinin kalıcı belleğe kaydedildiği bir kapatma işleminin gerçekleştirilmesine neden olur. CPU'nun yol alma işlemi sırasında, işletim saati sayaçlarının kaydedilen değerleri, önceki işletim saatleri kaybolmayacak şekilde sayaçlara yeniden yüklenir. Sonraki işletim saatlerini saymak için işletim saati sayaçları yeniden başlatılmalıdır.

Programınız ayrıca RTM yordamının 7. yürütme modu aracılığıyla, işletim saati sayaçlarının değerlerini bir memory card'a kaydedebilir. RTM yordamının 7. modunun yürütüldüğü anda tüm işletim saati sayaçlarının durumları memory card'a kaydedilir. Kaydedilen bu değerler, zaman içinde, program akışındaki işletim saati sayaçlarının başlatma ve durdurma işlemleri nedeniyle yanlış olabilir. Önemli çalışma zamanı olaylarını yakalamak için, memory card'daki değerleri düzenli olarak güncellemeniz gerekir. RTM değerlerini memory card'a kaydetmenin avantajı, memory card'ı, programınız ve kaydedilen RTM değerleri kullanılabilir durumda olacak şekilde bir yedek CPU'ya takabilme olanağına sahip olmanızdır. RTM değerlerini bir memory card'a kaydetmediyseniz, işletim saati değerleri kaybolur (bir yedek CPU kullanıldığında).

NOT

Memory card'a yazma sırasında program çağırma işlemlerinin sayısının aşırı olmamasını sağlayın

Memory card'ın kullanım ömrünü uzatmak için, memory card'daki yazma işlemlerinin sayısını mümkün olduğunca düşük tutun.

Tablo 9-15 Koşul kodları

RET_VAL (W#16#....)	Açıklama
0	Hata yok
8080	İşletim saati sayacının numarası yanlış
8081	PV parametresine negatif bir değer aktarıldı
8082	İşletim saati sayacında taşma
8091	MODE giriş parametresi izin verilmeyen bir değer içeriyor
80B1	Değer, MC'ye kaydedilemiyor (MODUS = 7)

9.2 Karakter dizileri ve karakter yordamları

9.2.1 String veri tipi

STRING veri tipi

String tipi veriler 2 Bayt'lık bir Header ve onu takip eden ASCII kodunda maksimum 254 karakter Bayt'ı ile birlikte kaydedilir. String verileri için kullanılan bir Header, iki uzunluk Bayt'ına sahiptir. İlk Bayt, bir karakter dizisinin başlatılması sırasında köşeli parantezlerin içinde belirtilen ve önceden 254 olarak ayarlanmış olan maksimum uzunluk değerini içerir. İkinci Header Bayt'ı, karakter dizisindeki geçerli karakter sayısına karşılık gelen gerçek uzunluk değerini içerir. Gerçek uzunluk, maksimum uzunluktan daha küçük veya ona eşit olmalıdır. String formatı için kaydedilen Bayt'ların sayısı, maksimum uzunluktan 2 Bayt daha büyüktür.

String verilerinin başlatılması

String veri tipindeki giriş ve çıkış verileri, yordamlar yürütülmeden önce bellekte geçerli karakter dizileri olarak başlatılmalıdır.

Geçerli String verileri

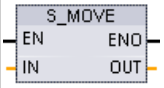
Geçerli String verileri, sıfırdan büyük ve 255'ten küçük olması gereken bir maksimum uzunluğa sahiptir. Gerçek uzunluk, maksimum uzunluktan daha küçük veya ona eşit olmalıdır.

String verileri, E veya A bellek alanlarına atanamaz.

Daha fazla bilgi için bkz.: String veri tipinin formatı (Sayfa 103).

9.2.2 S_MOV (karakter dizisini kaydır)

Tablo 9-16 Karakter dizisini kaydır yordamı

LAD/FBD	SCL	Açıklama
	**out := in;**	IN kaynağının karakter dizisini OUT adresine kopyalayın. S_MOVE'un yürütülmesi, kaynak karakter dizisinin içeriğini etkilemez.

Tablo 9-17 Parametreler için kullanılan veri tipleri

Parametre	Veri tipi	Açıklama
IN	String	Kaynak karakter dizisi
OUT	String	Hedef adres

IN girişindeki karakter dizisinin gerçek uzunluğu, OUT çıkışına kaydedilen bir karakter dizisinin maksimum uzunluğunu aşıyorsa, IN karakter dizisinin, OUT'a uyan kısmı kopyalanır.

9.2.3 Karakter dizilerinin dönüştürülmesi için kullanılan yordamlar

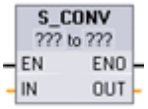
9.2.3.1 S_CONV, STRG_VAL ve VAL_STRG (Karakter dizisine/karakter dizisinden ve sayı değerine/sayı değerinden dönüştür) yordamları

Aşağıdaki yordamlarla sayı karakter dizilerini sayı değerlerine ve sayı değerlerine sayı karakter dizilerine dönüştürebilirsiniz:

- S_CONV yordamıyla bir sayı karakter dizisi bir sayı değerine veya bir sayı değeri bir sayı karakter dizisine dönüştürülür.
- STRG_VAL yordamıyla bir sayı karakter dizisi, format opsiyonlarıyla birlikte bir sayı değerine dönüştürülür.
- VAL_STRG yordamıyla bir sayı değeri, format opsiyonlarıyla birlikte bir sayı karakter dizisine dönüştürülür.

S_CONV (karakter dizisini dönüştür)

Tablo 9-18 Karakter dizilerinin dönüştürülmesi için kullanılan yordamlar

LAD/FBD	SCL	Açıklama
	**out := <Type>_TO_<Type>(in);**	Bir karakter dizisini ilgili değere veya bir değeri ilgili karakter dizisine dönüştürür. S_CONV yordamı, çıkış değerinin biçimlendirilmesiyle ilgili hiçbir opsiyona sahip değildir. Bu nedenle S_CONV , STRG_VAL ve VAL_STRG yordamlarından daha basittir, fakat aynı zamanda bu yordamlara göre daha az esnektir.

1 LAD/FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

2 SCL'de: Gelişmiş yordamlardan S_CONV'u seçin ve dönüştürme için veri tipleri sorgularını yanıtlayın. STEP 7, bunun üzerine ilgili dönüştürme yordamını sağlar.

Tablo 9-19 Veri tipleri (karakter dizisini sayı değerine dönüştürme)

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString	Girilen karakter dizisi
OUT	OUT	String, WString, Char, WChar, SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Yayınlanan sayı değeri

IN karakter dizisi parametresinin dönüştürülmesi işlemi ilk karakterle başlar ve karakter dizisinin sonuna veya "0" ila "9" arasında bir rakam, "+", "-" veya "." olmayan ilk karaktere kadar devam eder. Sonuç, OUT parametresiyle belirtilen adreste sunulur. Yayınlanan sayı değeri OUT parametresinin veri tipi aralığına uymuyorsa, OUT parametresi 0 olarak ve ENO FALSE olarak ayarlanır. Aksi takdirde, OUT parametresi geçerli bir sonuç içerir ve ENO DOĞRU olarak ayarlanır.

Karakter dizisi girişiyle ilgili format kuralları:

- IN'deki karakter dizisi bir desimal karakteri içeriyorsa, "." karakteri kullanılmalıdır.
- Desimal noktasının solunda binler basamağı ayırma karakteri olarak "," virgül işaretine izin verilir, ancak bu işaret göz ardı edilir.
- Başa eklenen boşluk karakterleri göz ardı edilir.

S_CONV (değerden karakter dizisine dönüştürme)

Tablo 9-20 Veri tipleri (sayı değerini karakter dizisine dönüştürme)

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString, Char, WChar, SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Girilen sayı değeri
OUT	OUT	String, WString	Yayınlanan karakter dizisi

Ön işaretli veya ön işareti olmayan bir tamsayı değeri veya bir IN kayar noktalı değer ilgili karakter dizisine dönüştürülür ve OUT parametresinde sunulur. Dönüştürme işlemi gerçekleştirilmeden önce OUT parametresi geçerli bir karakter dizisini referanslamalıdır. Geçerli bir karakter dizisi, ilk Bayt'ta maksimum karakter dizisi uzunluğu, ikinci Bayt'ta gerçek karakter dizisi uzunluğu ve sonraki Bayt'larda gerçek karakterlerden oluşur. Dönüştürülen karakter dizisi, ilk karakterden başlayarak OUT karakter dizisindeki ilk karakterin yerine geçer ve gerçek uzunluğu içeren Bayt'ı, uygun şekilde, yayınlanan karakter dizisinin uzunluğuna uyarlar. Yayınlanan karakter dizisinin maksimum uzunluğunu içeren Bayt değiştirilmez.

Kaç karakterin değiştirileceği, IN parametresinin veri tipine ve sayı değerine bağlıdır. Değiştirilen karakter sayısı, OUT parametresinde belirlenen karakter dizisi uzunluğuna uymalıdır. OUT parametresindeki karakter dizisinin maksimum karakter dizisi uzunluğu (ilk Bayt), dönüştürülen karakterlerin beklenen maksimum sayısından daha büyük veya ona eşit olmalıdır. Aşağıdaki tabloda, S_CONV ile değerlerin karakter dizilerine dönüştürülmesiyle ilgili örnekler gösterilmektedir:

Yayınlanan karakter dizileriyle ilgili format kuralları:

- OUT parametresine yazılan değerler başa eklenen "+" karakterini kullanmaz.
- Sabit nokta gösterimi kullanılır (üs gösterimi kullanılmaz).
- IN parametresi Real veri tipindeyse "." nokta işaretiyle desimal noktası gösterilir.
- Değerler, yayınlanan karakter dizisinde sağa hizalanmış şekilde yerleştirilir ve önlerinde, boş karakter pozisyonlarınım yerini tutan boşluk karakterleri bulunur.

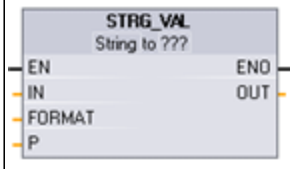
Tablo 9-21 Her veri tipi için maksimum karakter dizisi uzunluğu

Veri tipi IN	S_CONV tarafından atanan karakter pozisyonları	Dönüştürülmüş bir karakter dizisi örneği[1]	Maksimum ve gerçek uzunluk için kullanılan Bayt'larla birlikte karakter dizisinin toplam uzunluğu
USInt	4	"x255"	6
SInt	4	"-128"	6
UInt	6	"x65535"	8
Int	6	"-32768"	8
UDInt	11	"x4294967295"	13
DInt	11	"-2147483648"	13
Real	14	"x-3.402823E+38" "x-1.175495E-38" "x+1.175495E-38" "x+3.402823E+38"	16
LReal	21	"-1.7976931348623E+308" "-2.2250738585072E-308" "+2.2250738585072E-308" "+1.7976931348623E+308"	23

[1] "x" karakterleri, dönüştürülmüş değerdeki atanan sağa hizalanmış alanda bulunan boş pozisyonları kaplayan boşluk karakterlerini gösterir.

STRG_VAL (karakter dizisini sayı değerine dönüştür)

Tablo 9-22 Karakter dizisini sayı değerine dönüştür yordamı

LAD/FBD	SCL	Açıklama
	**"STRG_VAL"(in:=_string_in, format:=_word_in, p:=uint_in, out=>_variant_out);**	Bir sayı karakter dizisini ilgili tamsayı veya kayar noktalı gösterime dönüştürür.

[1] LAD/FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 9-23 STRG_VAL yordamı için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString	Dönüştürülmesi gereken ASCII karakter dizisi
FORMAT	IN	Word	Çıkış formatları için kullanılan opsiyonlar
P	IN	UInt, Byte, USInt	IN: Dönüştürülmesi gereken ilk karaktere verilen referans (ilk karakter = 1)
OUT	OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Dönüştürülmüş sayı değeri

Dönüştürme işlemi, girilen karakter dizisinde P karakter ofsetinde başlar ve karakter dizisinin sonuna veya "+", "-", ".", ",", "e", "E" veya "0" ile "9" arasındaki rakamlar olmayan ilk karaktere kadar gider. Sonuç, OUT parametresinde belirtilen adreste yayınlanır.

String veri tipindeki veriler, yürütme işleminden önce bellekte geçerli karakter dizisi olarak başlatılmalıdır.

STRG_VAL yordamının FORMAT parametresi aşağıda tanımlanmıştır. Serbest Bit pozisyonları sıfıra ayarlanmalıdır.

Tablo 9-24 STRG_VAL yordamının formatı

Bit 16							Bit 8	Bit 7							Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	f	r

f = Gösterim formatı — 1= Üs gösterim
0 = Sabit nokta gösterimi

r = Desimal noktası formatı — (Virgül işareti)
0 = "." (Nokta işareti)

Tablo 9-25 FORMAT parametresinin değeri

FORMAT (W#16#)	Gösterim formatı	Desimal noktası gösterimi
0000 (ön ayar)	Sabit nokta	"."
0001		","
0002	Üs	"."
0003		","
0004 ila FFFF	Geçersiz değerler	

STRG_VAL dönüştürmesi için tanımlanan kurallar:

- Nokta işareti "." desimal noktası için kullanılırsa, desimal noktasının solundaki virgül işaretleri "," binler basamağı ayırma karakteri olarak yorumlanır. Virgül işaretlerine izin verilir ve bu işaretler göz ardı edilir.
- Virgül işareti "," desimal ayırma işareti olarak kullanılırsa, desimal karakterinin solundaki nokta işaretleri "." binler basamağı ayırma karakteri olarak yorumlanır. Bu nokta işaretlerine izin verilir ve bu işaretler göz ardı edilir.
- Başa eklenen boşluk karakterleri göz ardı edilir.

VAL_STRG (sayı değerini karakter dizisine dönüştür)

Tablo 9-26 Sayı değerini karakter dizisine dönüştür yordamı

LAD/FBD	SCL	Açıklama
	**"VAL_STRG"(in:=_variant_in, size:=_usint_in, prec:=_usint_in, format:=_word_in, p:=uint_in, out=>_string_out);**	Ön işaretli veya ön işareti olmayan bir tamsayı değerini veya bir kayar noktalı değeri ilgili karakter dizisi gösterimine dönüştürür.

[1] LAD/FBD'de: "???" öğesine tıklayın ve açılır listeden veri tipini seçin.

Tablo 9-27 VAL_STRG yordamı için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Dönüştürülmesi gereken değer
SIZE	IN	USInt	OUT karakter dizisine yazılması gereken karakterlerin sayısı
PREC	IN	USInt	Ondalık basamakların hassasiyeti veya boyutu. Bu, desimal ayırma işaretini kapsamaz.
FORMAT	IN	Word	Çıkış formatları için kullanılan opsiyonlar
P	IN	UInt, Byte, USInt	IN: OUT karakter dizisinin değiştirilmesi gereken ilk karakterine verilen referans (ilk karakter = 1)
OUT	OUT	String, WString	Dönüştürülen karakter dizisi

Bu yordam, IN parametresinde gösterilen değeri, OUT parametresinde referanslanan bir karakter dizisine dönüştürür. OUT parametresi, dönüştürme işlemi gerçekleştirilmeden önce geçerli bir karakter dizisi olmalıdır.

Dönüştürülen karakter dizisi, P parametresinde belirlenen karakter ofsetinden başlayarak SIZE parametresiyle belirlenen karakter sayısına kadar OUT karakter dizisinin karakterlerinin yerine geçer. SIZE parametresindeki karakterlerin sayısı, P karakter pozisyonundan itibaren sayıldığında OUT karakter dizisinin uzunluğuna uymalıdır. SIZE parametresi sıfır ise, OUT karakter dizisindeki P pozisyonunda bulunan karakterlerin üzerine sınırlama olmaksızın yazılır.

Bu yordam, rakam karakterlerini, metinden oluşan bir karakter dizisine eklemek için faydalıdır. Bu sayede, örneğin "120" sayıları "Pompa basıncı = 120 kPa" karakter dizisine eklenebilir.

PREC parametresi, karakter dizisinin ondalık basamaklarının hassasiyetini veya sayısını belirtir. IN parametresi bir tamsayı değeri ise, PREC, desimal ayırma işaretinin konumunu belirtir.

Örneğin veri değeri 123 ve PREC = 1 ise, sonuç "12.3" olur. Real veri tipi için desteklenen maksimum hassasiyet 7 basamaklıdır.

P parametresi OUT karakter dizisinin gerçek boyutundan daha büyükse, P pozisyonuna kadar boşluk karakterleri eklenir ve sonuç, karakter dizisinin sonuna eklenir. Dönüştürme işlemi, OUT karakter dizisinin maksimum uzunluğuna ulaşıldığında sona erer.

VAL_STRG yordamının FORMAT parametresi aşağıda tanımlanmıştır. Serbest Bit pozisyonları sıfıra ayarlanmalıdır.

Tablo 9-28 VAL_STRG yordamının formatı

Bit 16							**Bit 8**	**Bit 7**							**Bit 0**
0	0	0	0	0	0	0	0	0	0	0	0	0	s	f	r

s = Ön işaret kullanılan karakter — 1= "+" ve "-" ön işareti; 0 = sadece ön işaret "-"

f = Gösterim formatı — 1= Üs gösterim; 0 = Sabit nokta gösterimi

r = Desimal noktası formatı — (Virgül işareti); 0 = "." (Nokta işareti)

Tablo 9-29 FORMAT parametresinin değeri

FORMAT (WORD)	**Ön işaret**	**Gösterim formatı**	**Desimal noktası gösterimi**
W#16#0000	sadece "-"	Sabit nokta	"."
W#16#0001			","
W#16#0002		Üs	"."
W#16#0003			","
W#16#0004	"+" ve "-"	Sabit nokta	"."
W#16#0005			","
W#16#0006		Üs	"."
W#16#0007			","
W#16#0008 ila W#16#FFFF	Geçersiz değerler		

OUT parametresindeki karakter dizileri için tanımlanan format kuralları:

- Dönüştürülen karakter dizisi belirtilen boyuttan küçükse, başta boşluk karakterleri karakter dizisinin en soluna eklenir.
- FORMAT parametresinin ön işaret Bit'i YANLIŞ ise, ön işaretli ve ön işareti olmayan tamsayı değerleri, başta "+" ön işareti olmadan çıkış tamponuna yazılır. "-" ön işareti gerektiğinde kullanılır.
  <başta boşluk karakterleri><başta sıfırlar olmadan rakamlar>'.'<PREC rakamları>
- Parametrenin ön işaret Bit'i DOĞRU ise, ön işaretli ve ön işareti olmayan tamsayı değerleri, her zaman başta bir ön işaretle birlikte çıkış tamponuna yazılır.
  <başta boşluk karakterleri><ön işaret><başta sıfırlar olmadan sıfırlar>'.'<PREC rakamları>
- FORMAT olarak üs gösterim ayarlandıysa, Real veri tipinin değerleri aşağıdaki şekilde çıkış tamponuna yazılır:
  <başta boşluk karakterleri><ön işaret><rakam> '.' <PREC rakamları>'E' <ön işaret><başta sıfır olmadan rakamlar>

- FORMAT olarak sabit nokta gösterimi ayarlandıysa, ön işaretli ve ön işareti olmayan tamsayı değerleri ve Real veri tipindeki değerleri aşağıdaki şekilde çıkış tamponuna yazılır:
  <başta boşluk karakterleri><ön işaret><başta sıfırlar olmadan sıfırlar>'.'<PREC rakamları>
- Desimal noktasının solundaki başta sıfırlar (desimal noktasının hemen yanındaki rakamlar istisna olmak kaydıyla) engellenir.
- Desimal noktasının sağındaki değer, desimal noktasının sağında belirtilmesi gereken ve PREC parametresiyle ayarlanan rakamların sayısına karşılık gelmesi için yuvarlanır.
- Yayınlanan karakter dizisinin boyutu, desimal noktasının sağındaki rakamların sayısından en az üç Bayt daha fazla kapsamalıdır.
- Değerler, yayınlanan karakter dizisinde sağa hizalanmış şekilde yerleştirilir.

ENO ile bildirilen koşullar

Dönüştürme sırasında bir hata oluşursa, yordam aşağıdaki sonuçları verir:

- ENO, 0'a ayarlanır.
- OUT, 0'a veya karakter dizilerinden değerlere dönüştürmeyle ilgili örneklerde belirtilen değere ayarlanır.
- OUT, değişmeden veya OUT bir karakter dizisi ise, örneklerde gösterilen değerde kalır.

Tablo 9-30 ENO durumu

ENO	Açıklama
1	Hata yok
0	İzin verilmeyen veya geçersiz bir parametre, örneğin mevcut olmayan bir DB'ye bir erişim.
0	Maksimum uzunluğu 0 veya 255 olan izin verilmeyen karakter dizisi
0	Uzunluğu maksimum uzunluktan daha büyük olan izin verilmeyen karakter dizisi
0	Dönüştürülen sayı değeri, OUT için belirtilen veri tipi açısından çok büyüktür.
0	OUT parametresi için maksimum karakter dizisi boyutu, P parametresindeki karakter pozisyonuyla başlandığında, SIZE parametresiyle belirtilen sayı için yeterince büyük olmalıdır.
0	İzin verilmeyen P değeri, bu sırada P=0'dır veya P, güncel karakter dizisi uzunluğundan daha büyüktür
0	SIZE parametresi, PREC parametresinden büyük olmalıdır.

Tablo 9-31 S_CONV ile karakter dizilerinin değerlere dönüştürülmesiyle ilgili örnek

IN karakter dizisi	Veri tipi OUT	OUT değeri	ENO
"123"	Int veya DInt	123	DOĞRU
"-00456"	Int veya DInt	-456	DOĞRU
"123,45"	Int veya DInt	123	DOĞRU
"+2345"	Int veya DInt	2345	DOĞRU
"00123AB"	Int veya DInt	123	DOĞRU
"123"	Real	123,0	DOĞRU
"123,45"	Real	123,45	DOĞRU
"1.23e-4"	Real	1,23	DOĞRU
"1.23E-4"	Real	1,23	DOĞRU
"12.345,67"	Real	12345,67	DOĞRU

IN karakter dizisi	Veri tipi OUT	OUT değeri	ENO
"3.4e39"	Real	3,4	DOĞRU
"-3.4e39"	Real	-3,4	DOĞRU
"1.17549e-38"	Real	1,17549	DOĞRU
"12345"	SInt	0	YANLIŞ
"A123"	-/-	0	YANLIŞ
""	-/-	0	YANLIŞ
"++123"	-/-	0	YANLIŞ
"+-123"	-/-	0	YANLIŞ

Tablo 9-32 S_CONV ile değerlerin karakter dizilerine dönüştürülmesiyle ilgili örnekler

Veri tipi	IN değeri	OUT karakter dizisi [1]	ENO
UInt	123	"xxx123"	DOĞRU
UInt	0	"xxxxx0"	DOĞRU
UDInt	12345678	"xxx12345678"	DOĞRU
Real	+9123,456	"xx+9.123456E+3"	DOĞRU
LReal	+9123,4567890123	"xx+9.1234567890123E+3"	DOĞRU
Real	-INF	"xxxxxxxxxxxINF"	YANLIŞ
Real	+INF	"xxxxxxxxxxxINF"	YANLIŞ
Real	NaN	"xxxxxxxxxxxNaN"	YANLIŞ

[1] "x" karakterleri, dönüştürülmüş değerdeki atanan sağa hizalanmış alanda bulunan boş pozisyonları kaplayan boşluk karakterlerini gösterir.

Tablo 9-33 Örnek: STRG_VAL dönüştürme işlemi

IN karakter dizisi	FORMAT (W#16#....)	Veri tipi OUT	OUT değeri	ENO
"123"	0000	Int veya DInt	123	DOĞRU
"-00456"	0000	Int veya DInt	-456	DOĞRU
"123,45"	0000	Int veya DInt	123	DOĞRU
"+2345"	0000	Int veya DInt	2345	DOĞRU
"00123AB"	0000	Int veya DInt	123	DOĞRU
"123"	0000	Real	123,0	DOĞRU
"-00456"	0001	Real	-456,0	DOĞRU
"+00456"	0001	Real	456,0	DOĞRU
"123,45"	0000	Real	123,45	DOĞRU
"123,45"	0001	Real	12345,0	DOĞRU
"123,45"	0000	Real	12345,0	DOĞRU
"123,45"	0001	Real	123,45	DOĞRU
".00123AB"	0001	Real	123,0	DOĞRU
"1.23e-4"	0000	Real	1,23	DOĞRU

IN karakter dizisi	FORMAT (W#16#....)	Veri tipi OUT	OUT değeri	ENO
"1.23E-4"	0000	Real	1,23	DOĞRU
"1.23E-4"	0002	Real	1.23E-4	DOĞRU
"12.345,67"	0000	Real	12345,67	DOĞRU
"12.345,67"	0001	Real	12,345	DOĞRU
"3.4e39"	0002	Real	+INF	DOĞRU
"-3.4e39"	0002	Real	-INF	DOĞRU
"1.1754943e-38" (ve daha küçük)	0002	Real	0,0	DOĞRU
"12345"	-/-	SInt	0	YANLIŞ
"A123"	-/-	-/-	0	YANLIŞ
""	-/-	-/-	0	YANLIŞ
"++123"	-/-	-/-	0	YANLIŞ
"+-123"	-/-	-/-	0	YANLIŞ

VAL_STRG dönüştürme işlemleriyle ilgili aşağıdaki örnekler, aşağıda gösterildiği gibi başlatılmış olan bir OUT karakter dizisini temel alır:

"`Current Temp = xxxxxxxxxx C`"

Burada "`x`" karakteri, dönüştürülen değere atanmış olan boşluk karakterlerini temsil eder.

Tablo 9-34 Örnek: VAL_STRG dönüştürme işlemi

Veri tipi	IN değeri	P	SIZE	FORMAT (W#16#....)	PREC	OUT karakter dizisi	ENO
UInt	123	16	10	0000	0	`Current Temp = xxxxxxx123 C`	DOĞRU
UInt	0	16	10	0000	2	`Current Temp = xxxxxx0.00 C`	DOĞRU
UDInt	12345678	16	10	0000	3	`Current Temp = x12345.678 C`	DOĞRU
UDInt	12345678	16	10	0001	3	`Current Temp = x12345,678 C`	DOĞRU
Int	123	16	10	0004	0	`Current Temp = xxxxxx+123 C`	DOĞRU
Int	-123	16	10	0004	0	`Current Temp = xxxxxx-123 C`	DOĞRU
Real	-0,00123	16	10	0004	4	`Current Temp = xxx-0.0012 C`	DOĞRU
Real	-0,00123	16	10	0006	4	`Current Temp = -1.2300E-3 C`	DOĞRU
Real	-INF	16	10	-/-	4	`Current Temp = xxxxxx-INF C`	YANLIŞ
Real	+INF	16	10	-/-	4	`Current Temp = xxxxxx+INF C`	YANLIŞ
Real	NaN	16	10	-/-	4	`Current Temp = xxxxxxxNaN C`	YANLIŞ
UDInt	12345678	16	6	-/-	3	`Current Temp = xxxxxxxxxx C`	YANLIŞ

9.2.3.2 Strg_TO_Chars ve Chars_TO_Strg (Karakter dizisine/karakter dizisinden ve CHAR elemanlarından oluşan Array'e/Array'den dönüştür) yordamları

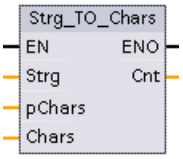
Strg_TO_Chars, bir ASCII karakter dizisini, karakter Bayt'larından oluşan bir Array'e kopyalar. Chars_TO_Strg, ASCII karakterlerinden oluşan bir Array'i bir karakter dizisine kopyalar.

NOT

Aşağıdaki amaçlarla sadece (Array [0..n] of Char) veya (Array [0..n] of Byte) sıfır tabanlı Array tiplerini kullanabilirsiniz:

- Chars_TO_Strg yordamı için kullanılan Chars giriş parametreleri
- Strg_TO_Chars yordamı için kullanılan Chars IN_OUT parametreleri

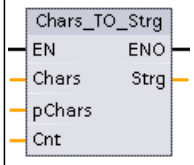
Tablo 9-35 Strg_TO_Chars yordamı

LAD/FBD	SCL	Açıklama
	**Strg_TO_Chars(** **Strg:=_string_in_,** **pChars:=_dint_in_,** **Cnt=>_uint_out_,** **Chars:=_variant_inout_);**	Strg tam giriş karakter dizisinin IN_OUT parametresindeki Chars karakterlerinden oluşan bir Array'e kopyalar. Yordam, pChars parametresi tarafından belirtilen Array eleman numarasından başlayarak Bayt'ların üzerine yazar. Desteklenen tüm maksimum uzunluklardaki (1 .. 254) karakter dizileri kullanılabilir. Hiçbir sonlandırıcı ayırma karakteri yazılmaz, bundan kendiniz sorumlusunuz. Sonlandırıcı ayırma karakterini, doğrudan en son yazılan Array karakterinden sonra olacak şekilde ayarlamak isterseniz, bir sonraki Array eleman numarasını [pChars+Cnt] kullanın.

Tablo 9-36 (Strg_TO_Chars) parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
Strg	IN	String, WString	Kaynak karakter dizisi
pChars	IN	DInt	Hedef Array'e yazılacak olan karakter dizisindeki ilk karakterin Array eleman numarası
Chars	IN_OUT	Varyant	Giriş karakter dizisinden kopyalanacak olan karakterlerden oluşan bir sıfır tabanlı Array'i [0..n] gösteren işaretçi. Array, bir DB'de veya blok arayüzündeki yerel değişkenler olarak deklare edilebilir. Örnek: "DB1".MyArray, DB1'deki MyArray [0..10] of Char eleman değerlerine referans verir.
Cnt	OUT	UInt	Kopyalanan karakter sayısı

Tablo 9-37 Chars_TO_Strg yordamı

LAD/FBD	SCL	Açıklama
	**Chars_TO_Strg(** **Chars:=_variant_in_,** **pChars:=_dint_in_,** **Cnt:=_uint_in_,** **Strg=>_string_out_);**	Karakterlerden oluşan tüm Array'i veya Array'in bir bölümünü bir karakter dizisine kopyalar. Çıkış karakter dizisi, Chars_TO_Strg yürütülmeden önce deklare edilmiş olmalıdır. Daha sonra Chars_TO_Strg yordamı tarafından karakter dizisinin üzerine yazılır. Desteklenen tüm maksimum uzunluklardaki (1 ... 254) karakter dizileri kullanılabilir. Karakter dizisinin maksimum uzunluğu için kullanılan değer, Chars_TO_Strg yordamı tarafından değiştirilmez. Array'den karakter dizisine kopyalama işlemi, karakter dizisinin maksimum uzunluğuna erişilirse sona erer. Karakter Array'indeki null karakteri "$00" veya 16#00 değeri, ayırma işareti olarak görev görür ve karakterlerin karakter dizisine kopyalanması işlemini sonlandırır.

Tablo 9-38 (Chars_TO_Strg) parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
Chars	IN	Varyant	Bir karakter dizisine dönüştürülmesi gereken ve karakterlerden oluşan sıfır tabanlı [0..n] Array ile gösteren işaretçi. Array, bir DB'de veya blok arayüzündeki yerel değişkenler olarak deklare edilebilir. Örnek: "DB1".MyArray, DB1'deki MyArray [0..10] of Char eleman değerlerine referans verir.
pChars	IN	Dint	Array'deki ilk kopyalanacak karakterin eleman numarası. Array elemanı [0], standart değerdir.
Cnt	IN	UInt	Kopyalanması gereken karakter sayısı: 0, tümü anlamına gelir
Strg	OUT	String, WString	Hedef karakter dizisi

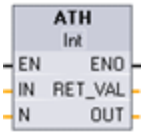
Tablo 9-39 ENO'nun durumu

ENO	Açıklama
1	Hata yok
0	Chars_TO_Strg: Çıkış karakter dizisine, kullanılan Bayt'a uygun olarak karakter dizisi deklarasyonundaki maksimum uzunluk için izin verilenden daha fazla karakter Bayt'ı kopyalamaya çalışma
0	Chars_TO_Strg: Sıfır karakter değeri (16#00), giriş karakterinin Bayt Array'inde bulundu.
0	Strg_TO_Chars: Çıkış Array'ine, kullanılan limite uygun olarak eleman sayısı için izin verilenden daha fazla karakter Bayt'ı kopyalamayı deneme

9.2.3.3 ATH ve HTA (ASCII karakter dizisine/ASCII karakter dizisinden ve onaltılık sayıya/onaltılık sayıdan dönüştür)

ATH (ASCII karakter dizisini onaltılık sayıya dönüştür) ve HTA (Onaltılık sayıyı ASCII karakter dizisine dönüştür) yordamlarıyla, ASCII karakterlerini (0 ila 9 ve A'dan F'ye kadar büyük harfler) ilgili 4-Bit onaltılık sayı gösterimlerine ve ters yönde dönüştürebilirsiniz.

Tablo 9-40 ATH yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := ATH(in:=_variant_in_, n:=_int_in_, out=>_variant_out_);**	ASCII karakterlerini paketlenmiş onaltılık basamaklara dönüştürür.

Tablo 9-41 ATH yordamı için kullanılan veri tipleri

Parametre tipi		Veri tipi	Açıklama
IN	IN	Varyant	ASCII karakter Bayt'ı veri alanını gösteren işaretçi
N	IN	UInt	Dönüştürülmesi gereken ASCII karakter Bayt'ı sayısı
RET_VAL	OUT	Word	Yürütme koşulu
OUT	OUT	Varyant	Dönüştürülen onaltılık Bayt veri alanını gösterir

Dönüştürme işlemi, IN parametresi tarafından belirlenen konumdan başlar ve N Bayt'ı kapsar. Sonuç, OUT parametresinde belirtilen adreste yayınlanır. Sadece 0 ile 9 arasındaki rakamlar, a ile f arasındaki küçük harfler ve A ile F arasındaki büyük harfler şeklindeki geçerli ASCII karakterleri dönüştürülebilir. Diğer tüm karakterler sıfıra dönüştürülür.

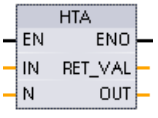
8 Bit ASCII kodlamalı karakterler 4 Bit onaltılık karakterlere dönüştürülür. İki ASCII karakteri, iki adet 4 Bit onaltılık karakter içeren tek bir Bayt'a dönüştürülebilir.

IN ve OUT parametreleri, onaltılık String verileri yerine Bayt Array'leri belirtir. ASCII karakterlerinin dönüştürülmesi ve onaltılık karakterler olarak yayınlanması işlemleri, okundukları sırayla gerçekleştirilir. ASCII karakterlerinin sayısının tek olması durumunda, en son dönüştürülen onaltılık basamağın en sağdaki yarım Bayt'ına sıfırlar yazılır.

Tablo 9-42 Örnekler: ASCII karakter dizisinin onaltılık sayılara dönüştürülmesi (ATH)

IN karakter Bayt'ı	N	OUT değeri	ENO
'0a23'	4	W#16#0A23	DOĞRU
'123AFx1a23'	10	16#123AF01023	YANLIŞ
'a23'	3	W#16#A230	DOĞRU

Tablo 9-43 HTA yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := HTA(in:=_variant_in_, n:=_uint_in_, out=>_variant_out_);**	Paketlenmiş onaltılık basamakları, ilgili ASCII karakter Bayt'larına dönüştürür.

Tablo 9-44 HTA yordamı için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	Varyant	Giriş Bayt'ı veri alanını gösterir
N	IN	UInt	Dönüştürülmesi gereken Bayt sayısı (her giriş biti iki adet 4 Bit yarım Bayt'a sahiptir ve 2N ASCII karakteri oluşturur)
RET_VAL	OUT	Word	Yürütme koşulu
OUT	OUT	Varyant	ASCII karakter Bayt'ı veri alanını gösteren işaretçi

Dönüştürme işlemi, IN parametresi tarafından belirlenen konumdan başlar ve N Bayt'ı kapsar. Her 4 Bit yarım Bayt, tek bir 8 Bit ASCII karakterine dönüştürülür ve çıkışta 2N-ASCII karakter Bayt'ı oluşturur. Çıkışın tüm 2N Bayt'ları, 0 ile 9 arasındaki rakamlar veya A ile F arasındaki büyük harfler şeklinde ASCII karakteri olarak yazılır. OUT parametresi, karakter dizisi değil, bir Bayt Array'i belirtir.

Onaltılık Bayt'ın her yarım Bayt'ı okuma sırasına göre bir karaktere dönüştürülür (önce onaltılık bir basamağın sol yarım Bayt'ı, ardından aynı Bayt'ın sağ yarım Bayt'ı dönüştürülür).

Tablo 9-45 Örnekler: Onaltılık sayıların ASCII karakter dizilerine dönüştürülmesi (HTA)

IN değeri	N	OUT karakter Bayt'ı	ENO (ENO, HTA yürütüldükten sonra her zaman DOĞRU'dur)
W#16#0123	2	'0123'	DOĞRU
DW#16#123AF012	4	'123AF012'	DOĞRU

Tablo 9-46 ATH and HTA koşul kodları

RET_VAL (W#16#....)	Açıklama	ENO
0000	Hata yok	DOĞRU
0007	Geçersiz ATH giriş karakteri: 0-9 arasında bir rakam, a-f arasında bir küçük harf veya A-F arasında bir büyük harf şeklinde bir ASCII karakterinin söz konusu olmadığı bir karakter bulundu	YANLIŞ
8101	İzin verilmeyen veya geçersiz bir giriş işaretçisi, örneğin mevcut olmayan bir DB'ye bir erişim.	YANLIŞ
8120	Giriş karakter dizisi geçersiz bir formata sahip, yani Maks. = 0, Maks. = 255, Güncel > Maks. veya İşaretçideki uzunluk < Maks.	YANLIŞ
8182	Giriş tamponu, N için çok küçük	YANLIŞ
8151	Giriş tamponu için kullanılan veri tipine izin verilmiyor	YANLIŞ
8301	İzin verilmeyen veya geçersiz bir çıkış işaretçisi, örneğin mevcut olmayan bir DB'ye bir erişim.	YANLIŞ
8320	Çıkış karakter dizisi geçersiz bir formata sahip, yani Maks. = 0, Maks. = 255, Güncel > Maks. veya İşaretçideki uzunluk < Maks.	YANLIŞ
8382	Çıkış tamponu, N için çok küçük	YANLIŞ
8351	Çıkış tamponu için kullanılan veri tipine izin verilmiyor	YANLIŞ

9.2.4 Karakter dizisi yordamları

9.2.4.1 MAX_LEN (bir karakter dizisinin maksimum uzunluğu)

Tablo 9-47 Yordam Maksimum Uzunluk

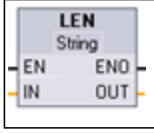
LAD/FBD	SCL	Açıklama
	**out := MAX_LEN(in);**	MAX_LEN (Karakter dizisinin maksimum uzunluğu), IN karakter dizisine OUT çıkışında atanan maksimum uzunluğu belirtir. Yordamın işlenmesi sırasında hatalar oluşursa, boş bir karakter dizisi uzunluğu yayınlanır. String ve WString veri tipleri iki uzunluk içermektedir: İlk Bayt (veya sözcük), maksimum uzunluğu ve ikinci Bayt (veya sözcük) ise güncel uzunluğu belirtir (bu, mevcut durumdaki geçerli karakter sayısıdır). • Karakter dizisinin maksimum uzunluğu, her String veya WString deklarasyonu için köşeli parantezlerin içinde belirtilir. Bir String tarafından doldurulan Bayt'ların sayısı, maksimum uzunluktan 2 Bayt daha büyüktür. Bir WString tarafından kaplanan sözcüklerin sayısı, maksimum uzunluktan 2 sözcük daha büyüktür. • Güncel uzunluk, gerçekten kullanılan karakterlerin sayısını gösterir. Gerçek uzunluk, maksimum uzunluktan daha küçük veya ona eşit olmalıdır. Güncel uzunluk, bir String için Bayt cinsinden ve bir WString için sözcük cinsinden belirtilir. MAX_LEN i yordamıyla, karakter dizisinin maksimum uzunluğunu ve LEN yordamıyla, bir karakter dizisinin güncel uzunluğunu çağırabilirsiniz.

Tablo 9-48 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString	Girilen karakter dizisi
OUT	OUT	DInt	IN karakter dizisindeki izin verilen maksimum karakter sayısı

9.2.4.2 LEN (bir karakter dizisinin uzunluğunu belirle)

Tablo 9-49 LEN yordamı

LAD/FBD	SCL	Açıklama
	**out := LEN(in);**	LEN (Uzunluk), IN karakter dizisinin güncel uzunluğunu OUT çıkışında yayınlar. Boş bir karakter dizisinin uzunluğu sıfırdır.

Tablo 9-50 Parametreler için kullanılan veri tipleri

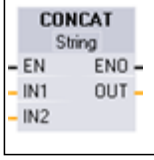
Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString	Girilen karakter dizisi
OUT	OUT	Int, DInt, Real, LReal	IN karakter dizisindeki geçerli karakter sayısı

Tablo 9-51 ENO durumu

ENO	Koşul	OUT
1	Geçerli karakter dizisi yok	Geçerli karakter dizisi uzunluğu
0	IN'in güncel uzunluğu, IN'in maksimum uzunluğunu aşar	Güncel uzunluk 0'a ayarlanır
	IN'in maksimum uzunluğu, atanan bellek alanına uymuyor	
	IN'in maksimum uzunluğu 255'tir (izin verilmeyen uzunluk)	

9.2.4.3 CONCAT (karakter dizilerini birleştir)

Tablo 9-52 Karakter dizilerini birleştir yordamı

LAD/FBD	SCL	Açıklama
	**out := CONCAT(in1, in2);**	CONCAT (Karakter dizilerini birleştir) yordamı, IN1 ve IN2 String parametrelerini, OUT çıkışında mevcut olan bir karakter dizisine bağlar. Bağlama işleminden sonra IN1 karakter dizisi, birleşik karakter dizisinin sol ve IN2 karakter dizisi de sağ kısmı olur.

Tablo 9-53 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1	IN	String, WString	Girilen karakter dizisi 1
IN2	IN	String, WString	Girilen karakter dizisi 2
OUT	OUT	String, WString	Birleşik karakter dizisi (karakter dizisi 1 + karakter dizisi 2)

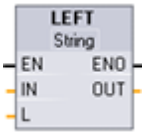
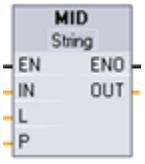
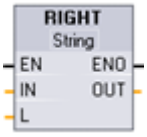
Tablo 9-54 ENO durumu

ENO	Koşul	OUT
1	Hata yok	Geçerli karakter
0	Bağlama işleminden sonra elde edilen karakter dizisi, OUT karakter dizisinin maksimum uzunluğundan büyüktür	Sonuçta elde edilen karakterler, OUT'un maksimum uzunluğuna ulaşılana kadar kopyalanır

ENO	Koşul	OUT
0	IN1'in güncel uzunluğu, IN1'in maksimum uzunluğunu aşar, IN2'nin güncel uzunluğu, IN2'nin maksimum uzunluğunu aşar veya OUT'un güncel uzunluğu, OUT'un maksimum uzunluğunu aşar (geçersiz karakter dizisi)	Güncel uzunluk 0'a ayarlanır
	IN1, IN2 veya OUT'un maksimum uzunluğu, atanan bellek alanına uymuyor	
	IN1 veya IN2'nin maksimum uzunluğu 255, OUT'un maksimum uzunluğu 0 veya 255'tir (String veri tipi)	
	IN1 veya IN2'nin maksimum uzunluğu 65534, OUT'un maksimum uzunluğu 0 veya 65534'tür (WString veri tipi)	

9.2.4.4 LEFT, RIGHT ve MID yordamları (bir karakter dizisindeki kısmi karakter dizilerini oku)

Tablo 9-55 LEFT, RIGHT ve MID yordamları

LAD/FBD	SCL	Açıklama
	**out := LEFT(in, L);**	LEFT (Sol kısmi karakter dizisi), IN parametresinin ilk L karakterinden oluşan bir kısmi karakter dizisini yayınlar. • L, IN karakter dizisinin güncel uzunluğundan büyükse, tüm IN karakter dizisi, OUT çıkışında yayınlanır. • Boş bir karakter dizisi girilirse, OUT çıkışında yine boş bir karakter dizisi yayınlanır.
	**out := MID(in, L, p);**	MID (Orta kısmi karakter dizisi) yordamı, bir karakter dizisinin ortadaki kısmını yayınlar. Orta kısmi karakter dizisi, L karakter uzunluğundadır ve P karakter pozisyonundan (bu pozisyon da dahil) başlar. L ve P'nin toplamı, IN String parametresinin güncel uzunluğunu aşıyorsa, P pozisyonundan IN karakter dizisinin sonuna kadar bir kısmi karakter pozisyonu yayınlanır.
	**out := RIGHT(in, L);**	RIGHT (Sağ kısmi karakter dizisi), bir karakter dizisinin son L karakterini yayınlar. • L, IN karakter dizisinin güncel uzunluğundan büyükse, tüm IN karakter dizisi, OUT çıkışında yayınlanır. • Boş bir karakter dizisi girilirse, OUT çıkışında yine boş bir karakter dizisi yayınlanır.

Tablo 9-56 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString	Girilen karakter dizisi
L	IN	Int	Oluşturulacak kısmi karakter dizisinin uzunluğu: • LEFT, karakter dizisinde en solda bulunan karakterlerin sayısını kullanır • RIGHT, karakter dizisinde en sağda bulunan karakterlerin sayısını kullanır • MID, karakter dizisinde P pozisyonundan itibaren bulunan karakterlerin sayısını kullanır
P	IN	Int	Sadece MID: Kısmi karakter dizisindeki, kopyalanması gereken ilk karakterin pozisyonu P = 1, IN karakter dizisindeki ilk karakterin pozisyonu için
OUT	OUT	String, WString	Yayınlanan karakter dizisi

Tablo 9-57 ENO durumu

ENO	Koşul	OUT
1	Hata yok	Geçerli karakter
0	• L veya P, 0'dan küçüktür veya 0'a eşittir • P, IN'in maksimum uzunluğundan büyüktür • IN'nin güncel uzunluğu, IN'in maksimum uzunluğunu aşıyor veya OUT'un güncel uzunluğu, OUT'un maksimum uzunluğunu aşıyor • IN'nin veya OUT'un maksimum uzunluğu, atanan belleğe uymuyor • IN'nin veya OUT'un maksimum uzunluğu 0 veya 255 (String veri tipi) veya 0 veya 65534 (WString veri tipi)	Güncel uzunluk 0'a ayarlanır
	Kopyalanması gereken kısmi karakter dizisinin uzunluğu (L), OUT karakter dizisinin maksimum uzunluğundan daha büyüktür	OUT'un maksimum uzunluğuna ulaşılana kadar gereken sayıda karakter kopyalanır
	Sadece MID: L veya P, 0'dan küçüktür veya 0'a eşittir	Güncel uzunluk 0'a ayarlanır
	Sadece MID: P, IN'in maksimum uzunluğundan büyüktür	
	IN1'in güncel uzunluğu, IN1'in maksimum uzunluğunu veya IN2'nin güncel uzunluğu, IN2'nin maksimum uzunluğun aşıyor (geçersiz karakter dizisi)	Güncel uzunluk 0'a ayarlanır
	IN1, IN2 veya OUT'un maksimum uzunluğu, atanan bellek alanına uymuyor	
	IN1, IN2 veya OUT'un maksimum uzunluğuna izin verilmiyor: 0 veya 255 (String veri tipi) veya 0 veya 65534 (WString veri tipi)	

9.2.4.5 DELETE (bir karakter dizisindeki karakteri sil)

Tablo 9-58 Kısmi karakter dizisini sil yordamı

LAD/FBD	SCL	Açıklama
	**out := DELETE(in, L, p);**	L karakterini IN karakter dizisinden siler. Karakterlerin silinme işlemine P karakter pozisyonundan (bu pozisyon dahil) başlanır ve kısmi karakter dizisinin geri kalanı OUT parametresinde yayınlanır. • L sıfıra eşitse, girilen karakter dizisi OUT'ta yayınlanır. • L ve P'nin toplamı, girilen karakter dizisinin uzunluğundan daha büyükse, karakter dizisi sonuna kadar silinir.

Tablo 9-59 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN	IN	String, WString	Girilen karakter dizisi
L	IN	Int	Silinecek karakter sayısı
P	IN	Int	Silinmesi gereken ilk karakterin pozisyonu: IN karakter dizisinin ilk karakteri 1 pozisyonuna sahiptir.
OUT	OUT	String, WString	Yayınlanan karakter dizisi

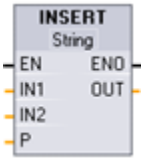
Tablo 9-60 ENO durumu

ENO	Koşul	OUT
1	Hata yok	Geçerli karakter
0	P, IN'nin güncel uzunluğundan büyüktür	IN, OUT'a kopyalanır, hiçbir karakter silinmez

ENO	Koşul	OUT
0	Silme işleminden sonra elde edilen karakter dizisi, OUT karakter dizisinin maksimum uzunluğundan daha büyüktür	Sonuçta elde edilen karakterler, OUT'un maksimum uzunluğuna ulaşılana kadar kopyalanır
	L, 0'dan küçüktür veya P, 0'dan küçüktür veya ona eşittir	Güncel uzunluk 0'a ayarlanır
	IN'in güncel uzunluğu, IN'in maksimum uzunluğunu veya OUT'un güncel uzunluğu, OUT'un maksimum uzunluğun aşıyor	
	IN'nin veya OUT'un maksimum uzunluğu, atanan belleğe uymuyor	
	IN veya OUT'un maksimum uzunluğu 0 veya 255'tir	

9.2.4.6 INSERT (bir karakter dizisine karakter ekle)

Tablo 9-61 Kısmi karakter dizisi ekle yordamı

LAD/FBD	SCL	Açıklama
	**out := INSERT(in1, in2, p);**	IN2 karakter dizisini, IN1 karakter dizisine ekler. Ekleme işlemi, P pozisyonundaki karakterden sonra başlar.

Tablo 9-62 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1	IN	String, WString	Girilen karakter dizisi 1
IN2	IN	String, WString	Girilen karakter dizisi 2
P	IN	Int	IN2 karakter dizisi için kullanılan ekleme noktasından önce IN1 karakter dizisindeki son karakter pozisyonu. IN1 karakter dizisinin ilk karakteri 1 pozisyonuna sahiptir.
OUT	OUT	String, WString	Sonuçta elde edilen karakter dizisi

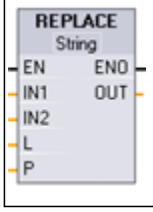
Tablo 9-63 ENO durumu

ENO	Koşul	OUT
1	Hata yok	Geçerli karakter
0	P, IN1'in uzunluğundan büyüktür	IN2, IN1'nin son karakterinden hemen sonra IN1'e bağlanır
	P, 0'dan küçüktür	Güncel uzunluk 0'a ayarlanır
	Ekleme işleminden sonra elde edilen karakter dizisi, OUT karakter dizisinin maksimum uzunluğundan daha büyüktür	Sonuçta elde edilen karakterler, OUT'un maksimum uzunluğuna ulaşılana kadar kopyalanır
	IN1'in güncel uzunluğu, IN1'in maksimum uzunluğunu aşar, IN2'nin güncel uzunluğu, IN2'nin maksimum uzunluğunu aşar veya OUT'un güncel uzunluğu, OUT'un maksimum uzunluğunu aşar (geçersiz karakter dizisi)	Güncel uzunluk 0'a ayarlanır

ENO	Koşul	OUT
0	IN1, IN2 veya OUT'un maksimum uzunluğu, atanan bellek alanına uymuyor	Güncel uzunluk 0'a ayarlanır
	IN1'in veya IN2'nin maksimum uzunluğu 255 veya OUT'un maksimum uzunluğu 0 veya 255'tir (String veri tipi)	
	IN1'in veya IN2'nin maksimum uzunluğu 65534 veya OUT'un maksimum uzunluğu 0 veya 65534'tür (WString veri tipi)	

9.2.4.7 REPLACE (bir karakter dizisindeki karakteri değiştir)

Tablo 9-64 Kısmi karakter dizisini değiştir yordamı

LAD/FBD	SCL	Açıklama
	**out := REPLACE(** **in1:=_string_in_,** **in2:=_string_in_,** **L:=_int_in_,** **p:=_int_in);**	IN1 parametresindeki karakter dizisinde bulunan L karakteri değiştirir. Değiştirme işlemi, IN1 karakter dizisinin P karakter pozisyonunda (bu pozisyon dahil) başlar, değiştirme için kullanılan karakterler IN2 parametresinden alınır.

Tablo 9-65 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1	IN	String, WString	Girilen karakter dizisi
IN2	IN	String, WString	Yedek karakterli karakter dizisi
L	IN	Int	Değiştirilmesi gereken karakter sayısı
P	IN	Int	Değiştirilmesi gereken ilk karakterin pozisyonu
OUT	OUT	String, WString	Sonuçta elde edilen karakter dizisi

L parametresi sıfıra eşitse, IN2 karakter dizisi, IN1 karakter dizisinin P pozisyonuna IN1 karakter dizisindeki karakterler silinmeden eklenir.

P 1'e eşitse, IN1 karakter dizisinin ilk L karakteri, IN2 karakter dizisinin karakterleriyle değiştirilir.

Tablo 9-66 ENO durumu

ENO	Koşul	OUT
1	Hata yok	Geçerli karakter
0	P, IN1'in uzunluğundan büyüktür	IN2, IN1'nin son karakterinden hemen sonra IN1'e bağlanır
	IN1'de P nokta, fakat IN1'deki L karakterden daha az	IN2, IN1'in bitiş karakterini P pozisyonundan itibaren değiştirir
	Değiştirme işleminden sonra elde edilen karakter dizisi, OUT karakter dizisinin maksimum uzunluğundan daha büyüktür	Sonuçta elde edilen karakterler, OUT'un maksimum uzunluğuna ulaşılana kadar kopyalanır
	IN1'in maksimum uzunluğu 0'dır	IN2 karakterleri OUT'a kopyalanır
	L, 0'dan küçüktür veya P, 0'dan küçüktür veya ona eşittir	Güncel uzunluk 0'a ayarlanır

ENO	Koşul	OUT
0	IN1'in güncel uzunluğu, IN1'in maksimum uzunluğunu aşar, IN2'nin güncel uzunluğu, IN2'nin maksimum uzunluğunu aşar veya OUT'un güncel uzunluğu, OUT'un maksimum uzunluğunu aşar	Güncel uzunluk 0'a ayarlanır
	IN1, IN2 veya OUT'un maksimum uzunluğu, atanan bellek alanına uymuyor	
	IN1'in veya IN2'nin maksimum uzunluğu 255 veya OUT'un maksimum uzunluğu 0 veya 255'tir (String veri tipi)	
	IN1'in veya IN2'nin maksimum uzunluğu 65534 veya OUT'un maksimum uzunluğu 0 veya 65534'tür (WString veri tipi)	

9.2.4.8 FIND (Bir karakter dizisindeki karakterleri bul)

Tablo 9-67 Kısmi karakter dizisini bul yordamı

LAD/FBD	SCL	Açıklama
	**out := FIND(in1:=_string_in_, in2:=_string_in);**	IN1 karakter dizisindeki IN2 tarafından belirtilen karakter dizisinin karakter pozisyonunu sağlar. Arama soldan başlar. IN2'nin ilk görüldüğü yerin karakter pozisyonu, OUT'ta yayınlanır. IN2 karakter dizisi IN1 karakter dizisinde bulunamazsa, sıfır değeri yayınlanır.

Tablo 9-68 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
IN1	IN	String, WString	Bu karakter dizisinde ara
IN2	IN	String, WString	Bu karakter dizisini ara
OUT	OUT	Int	IN1 karakter dizisindeki karakter pozisyonu, ilk pozitif arama sonucu

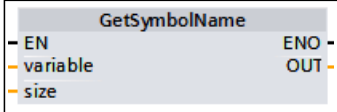
Tablo 9-69 ENO durumu

ENO	Koşul	OUT
1	Hata yok	Geçerli karakter pozisyonu
0	IN2, IN1'den büyüktür	Karakter pozisyonu 0'a ayarlanır
	IN1'in güncel uzunluğu, IN1'in maksimum uzunluğunu veya IN2'nin güncel uzunluğu, IN2'nin maksimum uzunluğun aşıyor (geçersiz karakter dizisi)	
	IN1 veya IN2'nin maksimum uzunluğu, atanan bellek alanına uymuyor	
	IN1 veya IN2'nin maksimum uzunluğu 255 (String veri tipi) veya 65535 (WString veri tipi)	

9.2.5 Çalışma zamanıyla ilgili bilgiler

9.2.5.1 GetSymbolName (giriş parametresindeki bir değişkenin adını oku)

Tablo 9-70 GetSymbolName yordamı

LAD/FBD	SCL	Açıklama
	**OUT := GetSymbolName(variable:=_parameter_in_, size:=_dint_in_);**	GetSymbolName yordamı, blok arayüzündeki bir değişkenin adına karşılık gelen bir karakter dizisi verir. Programınız, yordamı farklı değişkenlerle birden fazla kez çağırabilir. Değişkenin proses değeri ilgili değil. Yordam, okunan adı OUT parametresinde yayınlar.

Parametre

Aşağıdaki tabloda, GetSymbolName yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
VARIABLE	Input	PARAMETER	Input, Output, InOut parametre alt bölümleri	Adı String değeri olarak yayınlanması gereken yerel blok arayüzünden alınan değişken
SIZE	Input	DINT	E, A, M, D, L	OUT parametresinde yayınlanacak karakterlerin sayısını sınırlar: • SIZE > 0: GetSymbolName, adın ilk SIZE karakterlerini verir. • SIZE = 0: GetSymbolName, tam adı verir. • SIZE < 0: GetSymbolName, adın son SIZE karakterlerini verir.
OUT	Return	WSTRING	D, L	Giriş parametresi tarafından sağlanan değişken adının yayınlanması

Blok arayüzünün giriş parametrelerini VARIABLE parametresinde belirtebilirsiniz. Bu parametre için sadece bir arayüz parametresi kullanın ve PLC değişkeni veya veri bloğu değişkeni kullanmayın.

Okunan değişken adının uzunluğunu sınırlamak için SIZE parametresini kullanın. Yordam, adı bölümlere ayırırsa bu durum, adın sonundaki "..." karakterleri (Unicode karakteri 16#2026) ile belirtilir. Bu karakterin 1 uzunluğunda olduğuna dikkat edin.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

Örnek: SIZE parametresinin anlamı

Şu örnekte SIZE parametresinin anlamı gösterilmektedir. Şu değişken adı, blok arayüzünden okunur: "MyPLCTag" (adın başındaki ve sonundaki çift tırnak işaretleri ada aittir).

SIZE	GetSymbolName çıkışı	Anlamı
1	'...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını gösteren işaret: ... • WSTRING'in son karakteri:'
2	'"...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk karakteri:"... • WSTRING'in son karakteri:'
3	'"M...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk iki karakteri:"... "M... • WSTRING'in son karakteri:'
6	'"MyPL...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk beş karakteri: "MyPL... • WSTRING'in son karakteri:'
0	'"MyPLCTag"'	• WSTRING'in ilk karakteri:' • Adın tüm karakterleri: "MyPLCTag" • WSTRING'in son karakteri:'

Örnek: Sembolik bir adın okunması

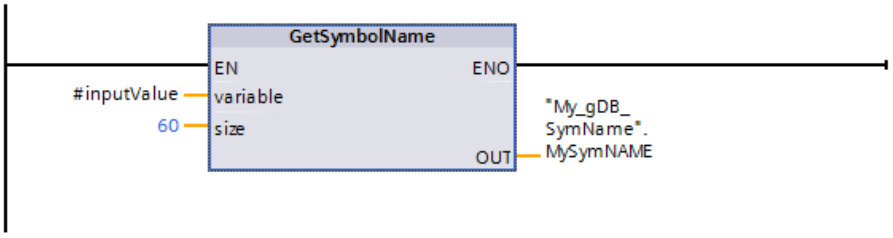
Aşağıdaki örnekte, bir bloğun giriş parametresi üzerinden bağlanan bir değişkenin adını okuyabilirsiniz.

Verileri kaydetmek için global bir veri bloğunda iki değişken oluşturun.

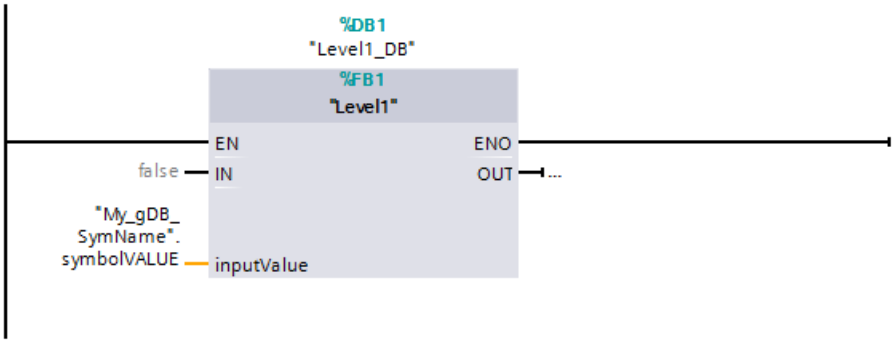
My_gDB_SymName

		Name	Data type	Start value
1		Static		
2		MySymNAME	WString	WSTRING#''
3		symbolVALUE	Byte	16#42

Level1 bloğunda, BYTE veri tipiyle bir inputValue giriş parametresi oluşturun. Level1 bloğunda GetSymbolName yordamını çağırın. Yordamın parametrelerini aşağıdaki şekilde ilişkilendirin:



Level1 bloğunun inputValue parametresini aşağıdaki şekilde bağlayın:



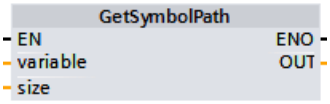
GetSymbolName yordamı, Level1 bloğunda yürütülür. Level1 bloğunun inputValue giriş parametresi, yordamın VARIABLE giriş parametresi yardımıyla kendi bağlantısı açısından incelenir. Bu sırada symbolVALUE değişkeni okunur ve OUT çıkış parametresinde karakter dizisi olarak yayınlanır ("MySymNAME"). Karakter dizisinin uzunluğu, SIZE giriş parametresinin değerine uygun olarak 60 karakterle sınırlanır.

My_gDB_SymName

	Name	Data type	Start value	Monitor value
1	Static			
2	MySymNAME	WString	WSTRING#''	WSTRING#'"My_gDB_SymName".symbolVALUE'
3	symbolVALUE	Byte	16#42	16#42

9.2.5.2 GetSymbolPath (giriş parametresi kaynağının birleşik global adını sorgula)

Tablo 9-71 GetSymbolPath yordamı

LAD/FBD	SCL	Açıklama
	**OUT := GetSymbolPath(variable:=_parameter_in_, size:=_dint_in_);**	GetSymbolPath yordamı, bir bloğun (FB veya FC) yerel arayüzündeki bir giriş parametresinin birleşik global adını okur. Ad, bellek yolu ile değişken adından oluşur. Programınız, yordamı farklı değişkenlerle birden fazla kez çağırabilir. Değişkenin proses değeri ilgili değil. Yordam, okunan adı OUT parametresinde yayınlar.

Parametre

Aşağıdaki tabloda, GetSymbolPath yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
VARIABLE	Input	PARAMETER	Input, Output, InOut parametre alt bölümleri	Çağrı yolunun başlangıcında, kendisi için ilgili güncel parametrenin adını okumak istediğiniz formel parametrenin seçimi.
SIZE	Input	DINT	E, A, M, D, L veya sabit	OUT parametresinde yayınlanacak karakterlerin sayısını sınırlar: • SIZE > 0: GetSymbolPath, adın ilk SIZE karakterlerini verir. • SIZE = 0: GetSymbolPath, tam adı verir. • SIZE < 0: GetSymbolPath, adın son SIZE karakterlerini verir.
OUT	Output	WSTRING	D, L	Giriş parametresi kaynağına ait değişken adının çıkışı.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

Kullanım

GetSymbolPath yordamının kullanımıyla ilgili aşağıdaki ipuçlarına dikkat edin:

- Giriş değişkeninin adını okumaya yardımcı olan blok arayüzünü, yordamın VARIABLE parametresinde belirtin:
  - Bir veri bloğu değişkeni giriş parametresini sağlarsa, GetSymbolPath, DB'nin adını, burada bulunan yapıları ve değişkenlerin adlarını verir.
  - Bir PLC değişkeni giriş parametresini sağlarsa, GetSymbolPath, PLC değişkenlerinin adını verir.
  - Bir sabit, giriş parametresini sağlarsa, GetSymbolPath, sabitin değerini verir.
- Okunan değişken adının uzunluğunu sınırlamak için SIZE parametresini kullanın. Ad bölümlere ayrıldıysa bu durum, adın sonundaki "..." karakterleri (Unicode karakteri 16#2026) ile belirtilir. Bu karakterin 1 uzunluğunda olduğuna dikkat edin.

Örnek: SIZE parametresinin anlamı

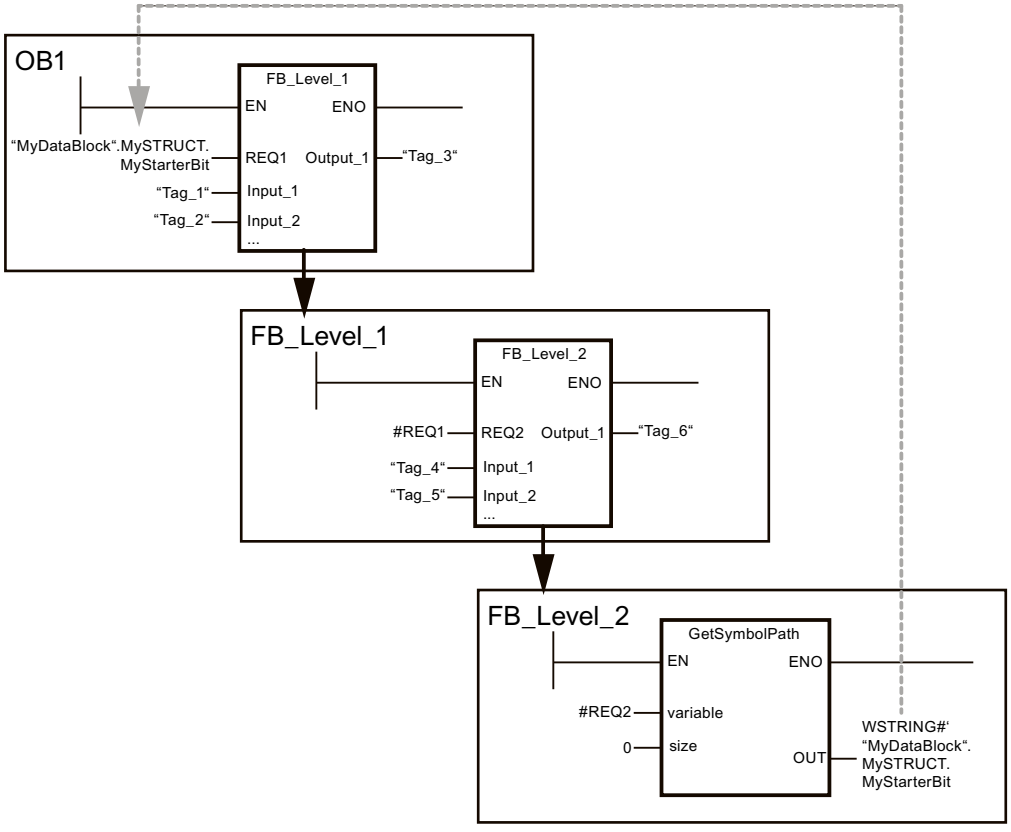
Şu örnekte, SIZE parametresinin anlamı gösterilmektedir. GetSymbolPath, blok arayüzünden şu değişken adını okudu: "MyPLCTag" (adın başındaki ve sonundaki çift tırnak işaretleri ada aittir).

SIZE	GetSymbolPath çıkışı	Anlamı
1	'...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını gösteren işaret: ... • WSTRING'in son karakteri:'
2	'"...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk karakteri:"... • WSTRING'in son karakteri:'
3	'"M...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk iki karakteri:"... "M... • WSTRING'in son karakteri:'
6	'"MyPL...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk beş karakteri: "MyPL... • WSTRING'in son karakteri:'
0	'"MyPLCTag"'	• WSTRING'in ilk karakteri:' • Adın tüm karakterleri: "MyPLCTag" • WSTRING'in son karakteri:'

Örnek: GetSymbolPath'in, blok çağrısının birden fazla kademesi üzerinden çağrılması

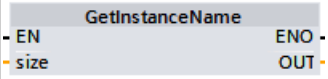
Aşağıdaki örnekte, GetSymbolPath'in birden fazla çağrı kademesi üzerinden kullanımı gösterilmektedir:

- Organizasyon bloğu OB1, FB_Level_1 bloğunu çağırır, bu blok da FB_Level_2 bloğunu çağırır.
- FB_Level_2 bloğu, REQ2 arayüzündeki parametrenin yolunu okumak için GetSymbolPath yordamını yürütür.
- REQ1 arayüzü REQ2 değerini sağladığı için, yordam, REQ1'in giriş parametresine ait yolu belirler.
- MyStarterBit değişkeni, REQ1'in giriş parametresidir. Bit, MyDatablock veri bloğundaki MySTRUCT yapısında bulunur.
  GetSymbolPath, bu bilgileri okur ve OUT parametresinde ("MyDataBlock".MySTRUCT.MyStarterBit) yolunu yayınlar.



9.2.5.3 GetInstanceName (blok örneğinin adını oku)

Tablo 9-72 GetInstanceName yordamı

LAD/FBD	SCL	Açıklama
	**OUT := GetInstanceName(size:=_dint_in_);**	GetInstanceName yordamıyla, bir fonksiyon bloğundaki örnek veri bloğunun adını okuyabilirsiniz.

Parametre

Aşağıdaki tabloda, GetInstanceName yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
SIZE	Input	DINT	E, A, M, D, L veya sabit	OUT parametresinde yayınlanacak karakterlerin sayısını sınırlar: • SIZE > 0: GetInstanceName, adın ilk SIZE karakterlerini verir. • SIZE = 0: GetInstanceName, tam adı verir. • SIZE < 0: GetInstanceName, adın son SIZE karakterlerini verir.
OUT	Output	WSTRING	D, L	Örnek veri bloğunun okunan adı

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

Örnek: SIZE parametresinin anlamı

Okunan örnek adının uzunluğunu sınırlamak için SIZE parametresini kullanın. Yordam, adı bölümlere ayırdıysa bu durum, adın sonundaki "..." karakterleri (Unicode karakteri 16#2026) ile belirtilir. Bu karakterin 1 uzunluğunda olduğuna dikkat edin.

Şu örnekte, SIZE parametresinin anlamı gösterilmektedir. GetInstanceName, blok arayüzünden şu örnek adını okudu: "Level1_DB" (adın başındaki ve sonundaki çift tırnak işaretleri ada aittir).

SIZE	GetSymbolPath çıkışı	Anlamı
1	'...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını gösteren işaret: ... • WSTRING'in son karakteri:'
2	'"...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk karakteri:"... • WSTRING'in son karakteri:'
3	'"L...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk iki karakteri:"... "L... • WSTRING'in son karakteri:'
6	'"Leve...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk beş karakteri: "Leve... • WSTRING'in son karakteri:'

SIZE	GetSymbolPath çıkışı	Anlamı
0	'"Level1_DB"'	• WSTRING'in ilk karakteri:' • Adın tüm karakterleri: "Level1_DB" • WSTRING'in son karakteri:'

GetInstanceName, örnek veri bloğunun adını OUT parametresine yazar. Yordam, örnek veri bloğunun adı, WSTRING'in maksimum uzunluğundan daha uzunsa adı bölümlere ayırır.

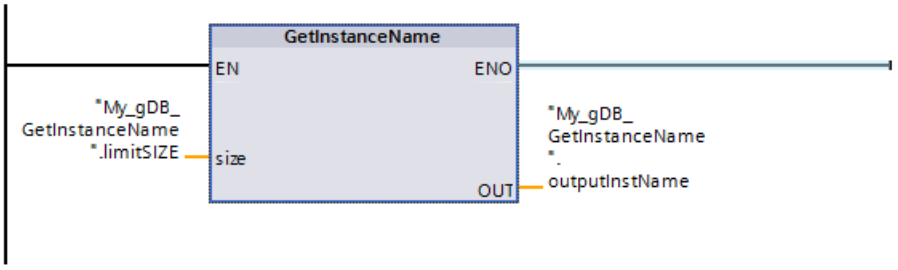
Örnek: Bir örnek veri bloğunun adının okunması

Aşağıdaki örnekte, bir örnek veri bloğunun adını nasıl okuyacağınız gösterilmektedir.
Verileri kaydetmek için global bir veri bloğunda iki değişken oluşturun.
Yordamın parametrelerini aşağıdaki şekilde tanımlayın:

My_gDB_GetInstanceName

		Name	Datentyp	Startwert
1		Static		
2		limitSIZE	DInt	0
3		outputInsName	WString	WSTRING#''

Level1_gin bloğu, GetInstanceName yordamını yürütür, bu sayede Level1_gin bloğunun ilişkili örnek veri bloğu belirlenir ve ad, OUT çıkış parametresinde bir karakter dizisi olarak yayınlanır (outputInstName). Karakter dizisinin uzunluğu, SIZE parametresinin (limitSIZE) 0 değerine uygun olarak sınırsızdır.

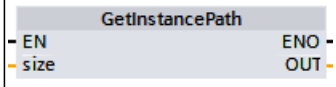


My_gDB_GetInstanceName

		Name	Data type	Start value	Monitor value
1		Static			
2		limitSIZE	DInt	0	0
3		outputInstName	WString	WSTRING#''	WSTRING#'"Level1_DB"'

9.2.5.4 GetInstancePath (blok örneğinin birleşik global adını sorgula)

Tablo 9-73 GetInstancePath yordamı

LAD/FBD	SCL	Açıklama
	**OUT := GetInstancePath(size:=_dint_in_);**	GetInstancePath yordamıyla, bir fonksiyon bloğundaki blok örneğinin birleşik global adını okuyabilirsiniz. Blok örneğinin birleşik global adı, program birden fazla örnek çağırdığı takdirde, tam çağırma hiyerarşisinin yoludur.

Parametre

Aşağıdaki tabloda, GetInstancePath yordamının parametreleri gösterilmektedir:

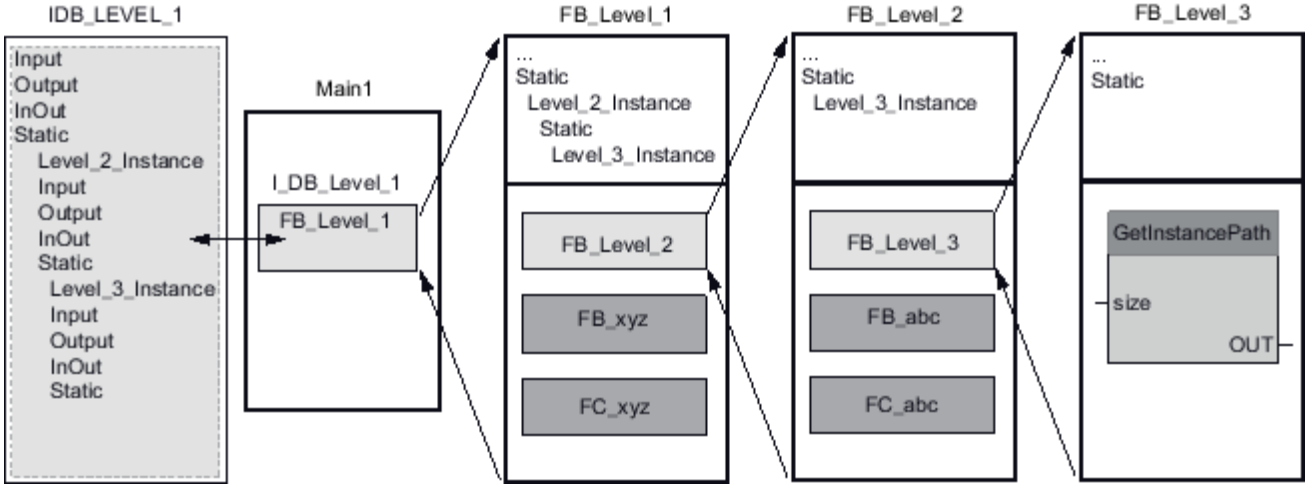
Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
SIZE	Input	DINT	E, A, M, D, L veya sabit	OUT parametresinde yayınlanacak karakterlerin sayısını sınırlar: • SIZE > 0: GetInstancePath, adın ilk SIZE karakterlerini verir. • SIZE = 0: GeInstancePath, tam adı verir. • SIZE < 0: GetInstancePath, adın son SIZE karakterlerini verir.
OUT	Output	WSTRING	D, L	Blok örneğinin okunmuş global adı. Blok örneğinin global adı, WSTRING'in maksimum uzunluğundan (254 karakter) uzunsa, GetInstancePath yordamı, adı bölümlere ayırır.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

Örnek: Birden fazla örneği olan bir FB'ye ait çağrının yolunu sorgulamak için GetInstancePath yordamının çağrılması

Aşağıdaki örnekte FB_Level_3 fonksiyon bloğu, GetInstancePath yordamını çağırmaktadır.

- FB_Level_3 fonksiyon bloğu, verilerini, çağıran FB_Level_2 fonksiyon bloğuna kaydeder.
- FB_Level_2 fonksiyon bloğu, verilerini, çağıran FB_Level_1 fonksiyon bloğuna kaydeder.
- Aynı şekilde FB_Level_1 fonksiyon bloğu, verilerini kendi IDB_LEVEL_1 örnek veri bloğuna kaydeder. Birden fazla örneğin kullanımı sayesinde, FB_Level_1'in örnek veri bloğu, üç fonksiyon bloğunun tüm verilerini içerir.



GetInstancePath yordamı, SIZE parametresinin değerine bağlı olarak bu örnek için aşağıdaki değerleri verir:

SIZE	GetInstancePath çıkışı	Anlamı
1	'...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını gösteren işaret: ... • WSTRING'in son karakteri:'
2	'"...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk karakteri:"... • WSTRING'in son karakteri:'
3	'"I...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk iki karakteri:"... "I... • WSTRING'in son karakteri:'
6	'"IDB_...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk beş karakteri: "IDB_... • WSTRING'in son karakteri:'
0	'"IDB_LEVEL_1".Level_2_Instance.Level_3_Instance'	• WSTRING'in ilk karakteri:' • Adın tüm karakterleri: "IDB_LEVEL_1".Level_2_Instance.Level_3_Instance • WSTRING'in son karakteri:'

NOT

GetInstancePath'in bir örnekli fonksiyon bloklarında kullanımı

GetInstancePath yordamını çağırdığınız fonksiyon bloğu, verileri kendi örnek veri bloğuna kaydediyorsa, GetInstancePath yordamı, münferit örneğin adını global ad olarak verir. Bu durumda OUT parametresindeki sonuç, GetInstanceName (Sayfa 345) yordamına karşılık gelir.

9.2.5.5 GetBlockName (bloğun adını oku)

Tablo 9-74 GetBlockName yordamı

LAD/FBD	SCL	Açıklama
	**RET_VAL := GetBlockName(size:=_dint_in_);**	GetBlockName yordamıyla, yordamın çağrıldığı bloğun adını okuyabilirsiniz.

Parametre

Aşağıdaki tabloda, GetBlockName yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
SIZE	Input	UINT	E, A, M, D, L veya sabit	RET_VAL parametresinde yayınlanacak karakterlerin sayısını sınırlar: • SIZE > 0: GetBlockName, adın ilk SIZE karakterlerini verir. • SIZE = 0: GetBlockName, tam adı verir. • SIZE < 0: GetBlockName, adın son SIZE karakterlerini verir.
RET_VAL	Output	WSTRING	D, L	Örnek veri bloğunun okunan adı

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

Örnek: SIZE parametresinin anlamı

Blok adının uzunluğunu belirli bir karakter sayısıyla sınırlamak için SIZE parametresinde maksimum uzunluk değerini belirtebilirsiniz. GetBlockName yordamı, adı bölümlere ayırırsa bu durum, adın sonundaki "..." karakterleri (Unicode karakteri 16#2026) ile belirtilir. Bu karakterin 1 uzunluğunda olduğuna dikkat edin.

Şu örnekte, SIZE parametresinin anlamı gösterilmektedir. GetBlockName, şu blok adını okudu: Level1_gbn (adın başındaki ve sonundaki çift tırnak işaretleri ada aittir).

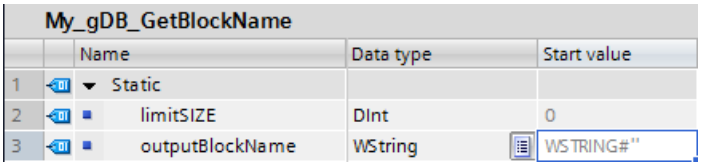
SIZE	GetBlockName çıkışı	Anlamı
1	'...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını gösteren işaret: ... • WSTRING'in son karakteri:'
2	'"...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk karakteri:"... • WSTRING'in son karakteri:'
3	'"L...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk iki karakteri:"... "L... • WSTRING'in son karakteri:'
6	'"Leve...'	• WSTRING'in ilk karakteri:' • Adın bölümlere ayrıldığını göstermek için kullanılan adın ve işaretin ilk beş karakteri: "Leve... • WSTRING'in son karakteri:'
0	'"Level1_gbn"'	• WSTRING'in ilk karakteri:' • Adın tüm karakterleri: "Level1_gbn" • WSTRING'in son karakteri:'

GetBlockName, bloğun adını RET_VAL parametresine yazar. Bloğun adı WSTRING'in maksimum uzunluğundan daha uzunsa, ad bölümlere ayrılır.

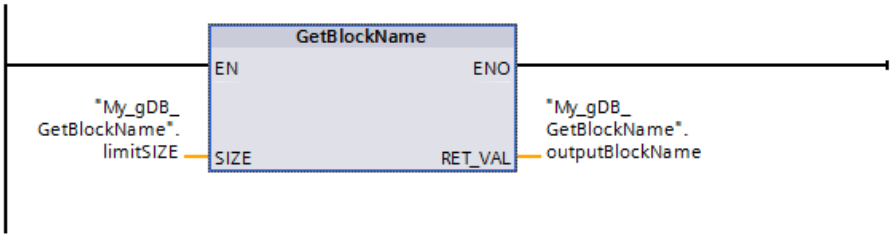
Örnek: Bir blok adının okunması

Aşağıdaki örnekte bir blok adının nasıl okunacağı gösterilmektedir.

1. Verileri kaydetmek için global bir veri bloğunda iki değişken oluşturun.



2. Yordamın parametrelerini aşağıdaki şekilde tanımlayın:



Level1_gbn bloğu, GetBlockName yordamını yürütür. GetBlockName yordamı, Level1_gbn bloğunun adını okur ve adı RET_VAL çıkış parametresinde karakter dizisi olarak yayınlar (outputBlockName). SIZE parametresi 0'a eşit olduğu için (limitSIZE), karakter dizisinin uzunluğu sınırsızdır.

My_gDB_GetBlockName

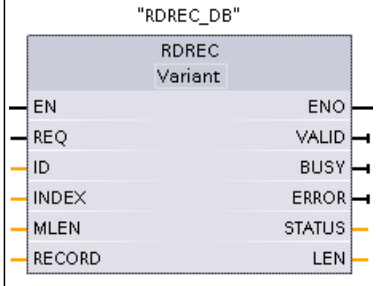
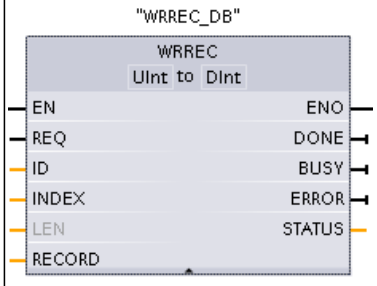
	Name	Data type	Start value	Monitor value
1	Static			
2	limitSIZE	DInt	0	0
3	outputBlockName	WString	WSTRING#''	WSTRING#'"Level1_gbn"'

9.3 Desantral G/Ç (PROFINET, PROFIBUS veya AS-i)

9.3.1 RDREC ve WRREC (veri setlerini oku/yaz)

RDREC (veri setini oku) ve WRREC (veri seti yaz) yordamlarını PROFINET, PROFIBUS ve AS-i için kullanabilirsiniz.

Tablo 9-75 RDREC ve WRREC yordamları

LAD/FBD	SCL	Açıklama
	**"RDREC_DB"(req:=_bool_in_, ID:=_word_in_, index:=_dint_in_, mlen:=_uint_in_, valid=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_dword_out_, len=>_uint_out_, record:=_variant_inout_);**	RDREC yordamıyla, INDEX numaralı bir veri setini, ID tarafından belirtilen bileşenden, örneğin merkezi bir modül taşıyıcısı veya bir desantral bileşenden (PROFIBUS DP veya PROFINET IO) okuyabilirsiniz. Okunması gereken Bayt'ların maksimum sayısını MLEN'de belirtin. RECORD hedef aralığının seçilen uzunluğu, en azından MLEN tarafından belirtilen Bayt'ların uzunluğu kadar olmalıdır.
	**"WRREC_DB"(req:=_bool_in_, ID:=_word_in_, index:=_dint_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_dword_out_, record:=_variant_inout_);**	WRREC yordamıyla bir veri setini (RECORD) INDEX numarasıyla, ID tarafından belirtilen bir DP-Slave'e veya bir PROFINET IO cihazı bileşenine, örn. merkezi modül taşıyıcısındaki bir modül veya bir desantral bileşene (PROFIBUS DP veya PROFINET IO) aktarabilirsiniz. Aktarılması gereken veri setinin Bayt uzunluğunu belirtin. Bu nedenle RECORD kaynak aralığının seçilen uzunluğu, en azından LEN tarafından belirtilen Bayt'ların uzunluğuna sahip olmalıdır.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneklerinde "RDREC_DB" ve "WRREC_DB", örnek DB'nin adlarıdır.

Tablo 9-76 RDREC ve WRREC parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQ = 1: Veri setini aktarma

Parametre ve veri tipi		Veri tipi	Açıklama
ID	IN	HW_IO (Word)	DP-Slave'in/PROFINET IO bileşeninin mantıksal adresi (modül veya alt modül): • Bir çıkış modülünde Bit 15 ayarlanmalıdır (örneğin 5 adresi için: ID:=DW#16#8005). • Bir birleşik modülde, iki adresten küçük olanı belirtilmelidir. **Not**: V3.0'da cihaz ID'si tercihe bağlı olarak iki şekilde belirlenebilir: • "Ağ görünümünde" yer alan şu opsiyonlar seçilerek: – Cihaz (gri alan) – Cihazın "özellikleri" – "Donanım kodu" **Not**: Ancak tüm cihazlar için donanım kodu görüntülenmez. • "Proje gezgininde" şu opsiyonların seçilmesi yoluyla: – PLC değişkenleri – Standart değişken tablosu – Sistemin "Sabitler" sekmesi Yapılandırılmış tüm cihazların donanım kodları görüntülenir. **Not**: V4.0'da arayüz modülü için kullanılan cihaz ID'si (donanım kodu), değişken tablosunda, sistem sabitlerinin altındaki "Cihaz adı [HEAD]" parametresi aracılığıyla belirlenir.
INDEX	IN	Byte, Word, USInt, UInt, SInt, Int, DInt	Veri seti numarası
MLEN	IN	Byte, USInt, UInt	Veri setinden çağrılacak bilgilerin Bayt cinsinden maksimum uzunluğu (RDREC)
VALID	OUT	Bool	Yeni veri seti alındı ve geçerli (RDREC). VALID Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
DONE	OUT	Bool	Veri seti aktarıldı (WRREC). DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• BUSY = 1: Okuma işlemi (RDREC) veya yazma işlemi (WRREC) henüz tamamlanmadı. • BUSY = 0: Veri setinin aktarılması tamamlandı.
ERROR	OUT	Bool	ERROR = 1: Bir okuma hatası (RDREC) veya yazma hatası (WRREC) oluştu. ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	DWord	Blok durumu (Sayfa 366) veya hata bilgileri (Sayfa 501)
LEN	OUT (RDREC) IN (WRREC)	UInt	• Veri setinden çağrılan bilgilerin uzunluğu (RDREC) • Aktarılması gereken veri setinin maksimum Bayt uzunluğu (WRREC)
RECORD	IN_OUT	Variant	• Çağrılan veri seti için kullanılan hedef aralık (RDREC) • Veri seti (WRREC)

RDREC ve WRREC yordamları asenkron çalışır, yani işleme, çok sayıda yordam çağrısını kapsar. Görevi, RDREC veya WRREC'i REQ = 1 ile çağırarak başlatabilirsiniz.

Görev durumu, BUSY çıkış parametresi ve STATUS çıkış parametresinin iki merkezi Bayt'ı aracılığıyla görüntülenir. BUSY çıkış parametresi FALSE değerine ayarlandıysa veri setinin aktarılması tamamlanır.

VALID (RDREC) veya DONE (WRREC) çıkış parametresi TRUE'ya eşit ise (sadece bir çevrim boyunca), bu, veri setinin hedef aralığa RECORD (RDREC) veya hedef cihaza (WRREC) başarıyla

aktarıldığını gösterir. LEN çıkış parametresi, RDREC'de çağrılan verilerin Bayt cinsinden uzunluğunu içerir.

ERROR çıkış parametresi (ERROR = TRUE ise sadece bir çevrim boyunca), veri seti aktarılırken bir hatanın oluştuğunu gösterir. Bu durumda, STATUS çıkış parametresi (ERROR = TRUE ifadesinin geçerli olduğu sadece bir çevrim boyunca) hata bilgilerini içerir.

Veri setleri, donanım cihazının üreticisi tarafından tanımlanır. Bir veri setiyle ilgili ayrıntılı bilgileri, ilgili donanım üreticisinin cihaz dokümantasyonunda bulabilirsiniz.

Maksimum dört RDREC yordamını ve dört WRREC yordamını eş zamanlı olarak kullanabilirsiniz.

NOT

Bir GSD dosyasıyla (Sürüm 3'ten itibaren GSD) bir DPV1 Slave'i yapılandırırsanız ve DP-Master'ın DP arayüzü "S7 uyumlu" olarak ayarlandıysa, kullanıcı programında "RDREC" ile G/Ç modüllerinden hiçbir veri setini okuyamaz ve "WRREC" ile G/Ç modüllerine hiçbir veri setini yazamazsınız. Bu gibi bir durumda DP-Master, yanlış yuvayı adresler (yapılandırılmış yuva + 3).

Yardım: DP-Master'ın arayüzü olarak "DPV1"i ayarlayın.

NOT

"RDREC" ve "WRREC" yordamlarının arayüzleri, "IEC 61131-3 uyarınca PROFIBUS iletişimi ve Proxy fonksiyon blokları için PROFIBUS Direktifi"nde tanımlanmış olan "RDREC" ve "WRREC" FB'lerle aynıdır.

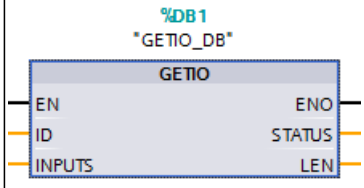
NOT

PROFINET IO için kullanılan bir veri setini okumak veya yazmak için "RDREC" veya "WRREC"i kullanırsanız, CPU, INDEX, MLEN ve LEN parametresindeki negatif değerleri ön işareti olmayan 16-Bit tamsayılar olarak yorumlar.

9.3.2 GETIO (proses görünümünü oku)

"GETIO" yordamıyla, DP-Slave'lerin ve PROFINET IO cihazlarının modüllerinden veya alt modüllerinin girişlerini tutarlı bir şekilde okuyabilirsiniz. "GETIO" yordamı, "DPRD_DAT (Sayfa 370-371)" yordamını çağırır. Veri aktarımı sırasında hiçbir hata oluşmazsa, okunan veriler, INPUTS tarafından belirtilen hedef aralığa girilir.

Tablo 9-77 GETIO (proses görünümünü oku) yordamı

LAD/FBD	SCL	Açıklama
	**"GETIO_DB"(** **id:=_uint_in_,** **status=>_dword_out_,** **len=>_int_out_,** **inputs:=_variant_inout_);**	"GETIO" yordamıyla, bir DP Standart Slave'in/PROFINET IO cihazının tüm girişlerini tutarlı bir şekilde okuyabilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "GETIO_DB", örnek DB'nin adıdır.

Hedef aralık, seçilen bileşenin uzunluğundan daha büyük bir uzunluğa veya bu bileşenle aynı uzunluğa sahip olmalıdır.

Modüler yapıya veya çok sayıda DP koduna sahip bir DP Standart Slave'den veri okuyorsanız, "GETIO" yordamı çağrısı başına sadece yapılandırılmış başlangıç adresindeki bir bileşenin/DP kodunun verilerine erişebilirsiniz.

Parametre

Aşağıdaki tabloda, "GETIO" yordamının parametreleri gösterilmektedir:

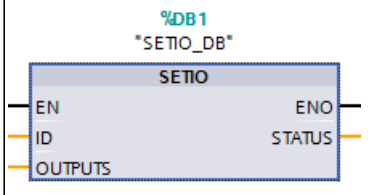
Parametre	Deklarasyon	Veri tipi	Açıklama
ID	IN	HW_SUBMODULE	DP Standart Slave'inin/PROFINET IO cihazının donanım ID'si
STATUS[1]	OUT	DWord	"DPRD_DAT (Sayfa 370-371)"ın DW#16#40xxxx00 formatındaki hata bilgilerini içerir
LEN	OUT	Int	Bayt cinsinden okunan veri miktarı
INPUTS	IN_OUT	Variant	Okunan veriler için hedef aralık: Hedef aralık, seçilen bileşenin DP Standart Slave'in/PROFINET IO cihazının uzunluğundan daha büyük bir uzunluğa veya bu bileşenle aynı uzunluğa sahip olmalıdır. Aşağıdaki veri tiplerini kullanabilirsiniz: • Sistem veri tipleri ve sistem veri tiplerinin Array'i: BYTE, CHAR, SINT, USINT, WORD, INT, UINT, DWORD, DINT, UDINT, REAL, LREAL, LWORD, LINT • Kullanıcı tanımlı veri tipleri (UDT) • Yapılar (STRUCT), ancak sadece optimize edilmemiş veri bloklarında (DB)

1 GETIO hata kodları görüntülenirken DWord veri tipini kullanın.

9.3.3 SETIO (proses görünümünü aktar)

"SETIO" yordamıyla verileri, OUTPUTS parametresi tarafından tanımlanan kaynak aralıktan DP-Slave'ler ve PROFINET IO cihazlarının adreslenmiş modüllerine veya alt modüllerine tutarlı bir şekilde aktarabilirsiniz. DP Standart Slave'in/PROFINET IO cihazının ilgili adres aralığını bir proses görünümünde tutarlı bir aralık olarak yapılandırdıysanız, veriler proses görünümüne aktarılır. "SETIO", bu aktarım işlemi esnasında "DPWR_DAT (Sayfa 370-371)" yordamını çağırır.

Tablo 9-78 SETIO (proses görünümünü oku) yordamı

LAD/FBD	SCL	Açıklama
	**"SETIO_DB"(id:=_uint_in_, status=>_dword_out_, outputs:=_variant_inout_);**	"SETIO" yordamıyla verileri, OUTPUTS parametresi tarafından tanımlanan kaynak aralıktan adreslenmiş DP-Slave'e veya PROFINET IO cihazına tutarlı bir şekilde aktarabilirsiniz.

1 Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

2 SCL örneğinde "SETIO_DB", örnek DB'nin adıdır.

Kaynak aralık, seçilen bileşenin uzunluğundan daha büyük bir uzunluğa veya bu bileşenle aynı uzunluğa sahip olmalıdır.

Modüler yapıya veya çok sayıda DP koduna sahip bir DP Standart Slave'de/PROFINET IO cihazında SETIO çağrısı başına sadece bir DP koduna/bileşenine erişebilirsiniz.

Parametre

Aşağıdaki tabloda, "SETIO" yordamının parametreleri gösterilmektedir:

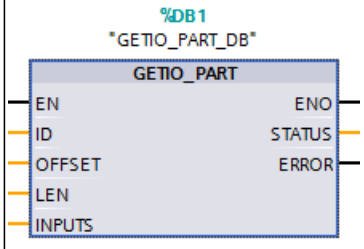
Parametre	Deklarasyon	Veri tipi	Açıklama
ID	IN	HW_SUBMODU-LE	DP Standart Slave'inin/PROFINET IO cihazının donanım ID'si
STATUS[1]	OUT	DWord	"DPWR_DAT (Sayfa 370-371)"ın DW#16#40xxxx00 formatındaki hata bilgilerini içerir
OUTPUTS	IN_OUT	Variant	Yazılması gereken veriler için kaynak aralık: Kaynak aralık, seçilen bileşenin DP Standart Slave'in/PROFINET IO cihazının uzunluğundan daha büyük bir uzunluğa veya bu bileşenle aynı uzunluğa sahip olmalıdır. Aşağıdaki veri tiplerini kullanabilirsiniz: • Sistem veri tipleri ve sistem veri tiplerinin Array'i: BYTE, CHAR, SINT, USINT, WORD, INT, UINT, DWORD, DINT, UDINT, REAL, LREAL, LWORD, LINT • Kullanıcı tanımlı veri tipleri (UDT) • Yapılar (STRUCT), ancak sadece optimize edilmemiş veri bloklarında (DB)

1 SETIO hata kodları görüntülenirken DWord veri tipini kullanın.

9.3.4 GETIO_PART (proses görünümü aralığını oku)

"GETIO_PART" yordamıyla, DP-Slave'lerin ve PROFINET IO cihazlarının modüllerine veya alt modüllerinin girişlerinin ilgili bir bölümünü tutarlı bir şekilde okuyabilirsiniz. GETIO_PART, "DPRD_DAT (Sayfa 370-371)" yordamını çağırır.

Tablo 9-79 GETIO_PART yordamı (proses görünümü aralığını oku)

LAD/FBD	SCL	Açıklama
	**"GETIO_PART_DB"(id:=_uint_in_, offset:=_int_in_, len:=_int_in_, status=>_dword_out_, error=>_bool_out_, inputs:=_variant_inout_);**	GETIO_PART yordamıyla, bir GÇ modülündeki girişlerin ilgili bir bölümünü tutarlı bir şekilde okuyabilirsiniz.

1 Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

2 SCL örneğinde "GETIO_PART_DB", örnek DB'nin adıdır.

ID giriş parametresiyle GÇ modülünü donanım ID'si aracılığıyla seçebilirsiniz.
OFFSET ve LEN parametreleriyle, proses görünümünün okunması gereken bölümünü belirtebilirsiniz. OFFSET ve LEN tarafından belirtilen giriş aralığı modül tarafından tamamen kapsanmıyorsa, blok, DW#16#40**80B7**00 hata kodunu yayınlar.

Hedef aralığın uzunluğu, okunması gereken Bayt miktarından daha büyük veya ona eşit olmalıdır:

- Veri aktarımı sırasında hiçbir hata oluşmazsa, ERROR, FALSE değerini alır. Okunan veriler, INPUTS parametresi tarafından tanımlanan hedef aralığa yazılır.
- Veri aktarımı sırasında bir hata oluşursa, ERROR, TRUE değerini alır. STATUS parametresi, DPRD_DAT'tan hata bilgilerini alır.
- Hedef aralık LEN'den büyükse, yordam, hedef aralığın ilk LEN Bayt'larına yazar. ERROR, FALSE değerini alır.

Parametre

Aşağıdaki tabloda, GETIO_PART yordamının parametreleri gösterilmektedir:

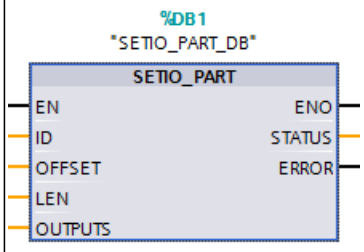
Parametre	Deklarasyon	Veri tipi	Açıklama
ID	IN	HW_SUBMOD-ULE	Modülün donanım kodu
OFFSET	IN	Int	Bileşen için proses görünümündeki okunması gereken ilk Bayt'ın numarası (mümkün olan en küçük değer: 0)
LEN	IN	Int	Okunacak Bayt sayısı
STATUS[1]	OUT	DWord	ERROR = TRUE koşulu sağlanırsa, "DPRD_DAT (Sayfa 370-371)"tan DW#16#40xxxx00 formatında hata bilgilerini alır
ERROR	OUT	Bool	Hata göstergesi: ERROR = TRUE, DPRD_DAT (Sayfa 370-371) çağrılırken bir hata oluşursa
INPUTS	IN_OUT	Variant	Okunan veriler için hedef aralık: Hedef aralık LEN'den büyükse, yordam, hedef aralığın ilk LEN Bayt'larına yazar. Aşağıdaki veri tiplerini kullanabilirsiniz: • Sistem veri tipleri ve sistem veri tiplerinin Array'i: BYTE, CHAR, SINT, USINT, WORD, INT, UINT, DWORD, DINT, UDINT, REAL, LREAL, LWORD, LINT • Kullanıcı tanımlı veri tipleri (UDT) • Yapılar (STRUCT), ancak sadece optimize edilmemiş veri bloklarında (DB)

[1] GETIO_PART hata kodları görüntülenirken DWord veri tipini kullanın.

9.3.5 SETIO_PART (proses görünümü aralığını aktar)

"SETIO_PART" yordamıyla verileri, OUTPUTS tarafından belirtilen kaynak aralıktan DP-Slave'lerin ve PROFINET IO cihazlarının modüllerine ve alt modüllerinin çıkışlarına tutarlı bir şekilde yazabilirsiniz. SETIO_PART, "DPWR_DAT (Sayfa 370-371)" yordamını çağırır.

Tablo 9-80 SETIO_PART (proses görünümü aralığını aktar) yordamı

LAD/FBD	SCL	Açıklama
	**"SETIO_PART_DB"(id:=_uint_in_, offset:=_int_in_, len:=_int_in_, status=>_dword_out_, error=>_bool_out_, outputs:=_variant_inout_);**	SETIO_PART yordamıyla verileri, OUTPUTS parametresi tarafından belirtilen kaynak aralıktan bir GÇ modülünün çıkışlarına tutarlı bir şekilde yazabilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "SETIO_PART_DB", örnek DB'nin adıdır.

ID, giriş parametresiyle, GÇ modülünü, belirtilen donanımı temel alarak seçebilirsiniz.
OFFSET ve LEN, parametreleriyle, proses görünümü aralığının ID tarafından adreslenen bileşen için yazılması gereken kısmını atayabilirsiniz. OFFSET Ve LEN tarafından belirtilen çıkış aralığı modül tarafından tamamen kapsanmıyorsa, blok, DW#16#40**80B7**00 hata kodunu yayınlar.
Hedef aralığın uzunluğu, okunması gereken Bayt miktarından daha büyük veya ona eşit olmalıdır:

- Veri aktarımı sırasında hiçbir hata oluşmazsa, ERROR, FALSE değerini alır.
- Veri aktarımı sırasında bir hata oluşursa, ERROR, TRUE değerini ve STATUS, DPWR_DAT'ın hata bilgilerini alır.
- Kaynak aralık LEN'den büyükse, yordam, OUTPUTS'un ilk LEN Bayt'larını aktarır. ERROR, FALSE değerini alır.

Parametre

Aşağıdaki tabloda, SETIO_PART yordamının parametreleri gösterilmektedir:

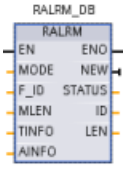
Parametre	Deklarasyon	Veri tipi	Açıklama
ID	IN	HW_SUBMODULE	GÇ modülünün donanım kodu
OFFSET	IN	Int	Bileşen için proses görünümündeki yazılması gereken ilk Bayt'ın numarası (mümkün olan en küçük değer: 0)
LEN	IN	Int	Yazılması gereken Bayt sayısı
STATUS[1]	OUT	DWord	ERROR = TRUE koşulu sağlanırsa "DPWR_DAT (Sayfa 370-371)"ın DW#16#40xxxx00 formatındaki hata bilgilerini içerir
ERROR	OUT	Bool	Hata göstergesi: ERROR = TRUE, DPWR_DAT (Sayfa 370-371) çağrılırken bir hata oluşursa
OUTPUTS	IN_OUT	Variant	Yazılması gereken veriler için kaynak aralık: Kaynak aralık LEN'den büyükse, OUTPUTS'un ilk LEN Bayt'ları aktarılır. Aşağıdaki veri tiplerini kullanabilirsiniz: • Sistem veri tipleri ve sistem veri tiplerinin Array'i: BYTE, CHAR, SINT, USINT, WORD, INT, UINT, DWORD, DINT, UDINT, REAL, LREAL, LWORD, LINT • Kullanıcı tanımlı veri tipleri (UDT) • Yapılar (STRUCT), ancak sadece optimize edilmemiş veri bloklarında (DB)

1 SETIO_PART hata kodları görüntülenirken DWord veri tipini kullanın.

9.3.6 RALRM (alarmı al)

RALRM (alarmı oku) yordamını, PROFINET ve PROFIBUS için kullanabilirsiniz.

Tablo 9-81 RALRM yordamı

LAD/FBD	SCL	Açıklama
	**"RALRM_DB"(mode:=_int_in_, f_ID:=_word_in_, mlen:=_uint_in_, new=>_bool_out_, status=>_dword_out_, ID=>_word_out_, len=>_uint_out_, tinfo:=_variant_inout_, ainfo:=_variant_inout_);**	RALRM (alarmı oku) yordamıyla, bir PROFIBUS veya PROFINET G/Ç modülünden/cihazından teşhis alarmı bilgilerini okuyabilirsiniz. Çıkış parametrelerindeki bilgiler, çağrılan OB'nin başlangıç bilgilerini ve alarm kaynağının bilgilerini içerir. Alarma neden olan olaylarla ilgili bilgileri yayınlamak için RALRM'ı bir alarm OB'sinde çağırın. S7-1200'de şu teşhis alarmı OB'leri desteklenir: Durum, güncelleme, profil, teşhis hatası alarmı, modüllerin çıkartılması veya takılması, modül taşıyıcısı veya istasyon hatası

1 Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

2 SCL örneğinde "RALRM_DB", örnek DB'nin adıdır.

Tablo 9-82 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
MODE	IN	Byte, USInt, SInt, Int	İşletim durumu

Parametre ve veri tipi		Veri tipi	Açıklama
F_ID	IN	HW_IO (Word)	Alarmların alınması gereken bileşenin (modülün), mantıksal başlangıç adresi. **Not**: Cihaz ID'si tercihe bağlı olarak iki şekilde belirlenebilir: • "Ağ görünümünde" yer alan şu opsiyonlar seçilerek: – Cihaz (gri alan) – Cihazın "özellikleri" – "Donanım kodu" **Not**: Tüm cihazlar için donanım kodu görüntülenmez. • "Proje gezgininde" şu opsiyonların seçilmesi yoluyla: – PLC değişkenleri – Standart değişken tablosu – Sistemin "Sabitler" sekmesi – Yapılandırılmış tüm cihazların donanım kodları görüntülenir.
MLEN	IN	Byte, USInt, UInt	Veri setinden alınacak bilgilerin Bayt cinsinden maksimum uzunluğu. MLEN değeri 0'a eşitse, AINFO hedef aralığında mevcut olan kadar alarm bilgisi alınabilir.
NEW	OUT	Bool	Yeni bir alarm alındı.
STATUS	OUT	DWord	RALRM yordamının durumu. Daha fazla bilgiyi, "RDREC, WRREC ve RALRM için kullanılan STATUS parametreleri" (Sayfa 366) başlığında bulabilirsiniz.
ID	OUT	HW_IO (Word)	Teşhis alarmına neden olan G/Ç modülünün donanım kodu **Not**: F_ID parametresinin açıklamasında, cihaz ID'sini nasıl belirleyeceğinizle ilgili bir açıklama bulabilirsiniz.
LEN	OUT	DWord, UInt, UDInt, DInt, Real, LReal	Alınan alarm bilgilerinin uzunluğu AINFO
TINFO	IN_OUT	Variant	Görev bilgileri: OB'nin başlangıç ve yönetim bilgileri için hedef aralık. TINFO uzunluğu her zaman 32 Bayt'tır.
AINFO	IN_OUT	Variant	Alarm bilgileri: Header bilgileri ve ek alarm bilgileri için hedef aralık. MLEN 'in 0'dan büyük olması durumunda, AINFO için en azından MLEN'de belirtilen Bayt sayısı değerinde bir uzunluk belirtmeniz gerekir. AINFO uzunluğu değişkendir.

NOT

"RALRM"yi, yol verme olayı bir G/Ç alarmı olmayan bir OB'de çağırırsanız, yordam, çıkışlarında buna uygun olarak daha az bilgi sağlar.

"RALRM"yi farklı OB'lerde çağırıyorsanız, farklı örnek DB'ler kullandığınızdan emin olun. "RALRM" çağrısından gelen verileri ilgili alarm OB'sinin dışında değerlendiriyorsanız, her OB yol verme olayı için kendine ait bir örnek DB kullanın.

NOT

"RALRM" yordamının arayüzü, EC 61131-3 uyarınca PROFIBUS iletişimi ve Proxy fonksiyon blokları için PROFIBUS Direktifi"nde tanımlanmış olan "RALRM" FB ile aynıdır.

RALRM'yi çağırma

RALRM yordamını, üç farklı işletim türünde (MODE) çağırabilirsiniz.

Tablo 9-83 RALRM yordamının işletim tipleri

MODE	Açıklama
0	• ID, alarmı tetikleyen G/Ç modülünün donanım kodunu içerir. • NEW çıkış parametresi DOĞRU olarak ayarlanır. • LEN, 0 değerinde bir çıkış üretir. • AINFO ve TINFO bilgilerle güncellenmez.
1	• ID, alarmı tetikleyen G/Ç modülünün donanım kodunu içerir. • NEW çıkış parametresi DOĞRU olarak ayarlanır. • LEN, yayınlanan AINFO verilerinin Bayt sayısı boyutunda bir çıkış üretir. • AINFO ve TINFO, alarmla ilgili bilgilerle güncellenir.
2	Alarmı, F_ID giriş parametresinin donanım kodu tetiklediyse şunlar geçerlidir: • ID, alarmı tetikleyen G/Ç modülünün donanım kodunu içerir. Bu değer, F_ID'nin değeriyle aynı olmalıdır. • NEW çıkış parametresi DOĞRU olarak ayarlanır. • LEN, yayınlanan AINFO verilerinin Bayt sayısı boyutunda bir çıkış üretir. • AINFO ve TINFO, alarmla ilgili bilgilerle güncellenir.

NOT

TINFO veya AINFO için çok kısa bir hedef aralık atarsanız, RALRM, tüm bilgileri döndüremez.

MLEN, yayınlanan AINFO veri miktarını sınırlayabilir.

TINFO ve AINFO'nun verilerini nasıl yorumlayacağınızla ilgili bilgileri, STEP 7'nin çevrimiçi bilgi sistemindeki AINFO ve TINFO parametrelerinde bulabilirsiniz.

TInfo organizasyon bloğunun verileri

Aşağıdaki tabloda, TInfo verilerinin RALRM yordamı için nasıl düzenlendiği gösterilmektedir:

OB'ler için de şu geçerlidir: Durum, güncelleme, profil, teşhis hatası alarmı, modüllerin çıkartılması veya takılması, modül taşıyıcısı veya istasyon hatası	0	SI formatı	OB sınıfı	OB-No.	
	4	LADDR			
TI_Submodule - OB'ler: Durum, güncelleme, profil	4			Yuva	
	8	Belirtici		0	
TI_DiagnosticInterrupt - OB: Teşhis hatası alarmı	4			G/Ç durumu	
	8	Kanal		Çoklu hata	0

[1] 28 - 31 aralığındaki Bayt'lar (üretici ve örnek) PROFIBUS tarafından kullanılmaz.

TI_PlugPullModule - OB: Modüllerin çıkartılması veya takılması	4			Olay sınıfı	Hata ID'si
	8	0		0	
TI_StationFailure - OB: Modül taşıyıcısı veya istasyon hatası	4			Olay sınıfı	Hata ID'si
	8	0		0	
OB'ler için de şu geçerlidir: Durum, güncelleme, profil, teşhis hatası alarmı, modüllerin çıkartılması veya takılması, modül taşıyıcısı veya istasyon hatası	12	0			
	16				
	20	Adres		slv_prfl	intr_type
	24	Markör1	Markör2	ID	
	28[1]	Üretici		Örnek	

[1] 28 - 31 aralığındaki Bayt'lar (üretici ve örnek) PROFIBUS tarafından kullanılmaz.

NOT

TINFO verileriyle ilgili ayrıntılı bilgileri STEP 7'nin çevrimiçi bilgi sisteminde bulabilirsiniz.

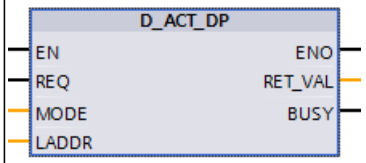
9.3.7 D_ACT_DP (DP-Slave'leri devre dışı bırak/etkinleştir)

"D_ACT_DP" yordamıyla, yapılandırılmış PROFINET IO cihazlarını hedefe yönelik olarak etkinleştirebilir veya devre dışı bırakabilirsiniz. Ayrıca, atanan her PROFINET IO cihazının mevcut durumda etkinleştirilmiş mi yoksa devre dışı mı bırakılmış olduğunu belirleyebilirsiniz.

NOT

D_ACT_DP yordamını sadece PROFINET IO cihazlarında kullanabilirsiniz. Yordamı PROFIBUS DP-Slave'leri ile kullanamazsınız.

Tablo 9-84 D_ACT_DP yordamı

LAD/FBD	SCL	Açıklama
	**"D_ACT_DP_DB"(req:=_bool_in_, mode:=_usint_in_, laddr:=_uint_in_, ret_val=>_int_out_, busy=>_bool_out_);**	D_ACT_DP yordamıyla, yapılandırılmış PROFINET IO cihazlarını etkinleştirebilir veya devre dışı bırakabilir ve atanmış bir PROFINET IO cihazının o anda etkinleştirilmiş mi yoksa devre dışı mı bırakılmış olduğunu belirleyebilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "D_ACT_DP_SFB_DB", örnek DB'nin adıdır.

IE/PB Link PN IO tipindeki bir Gateway'i D_ACT_DP yordamıyla devre dışı bırakamazsınız/etkinleştiremezsiniz. Buna rağmen D_ACT_DP'yi belirtilen Gateway için

kullanırsanız, CPU, W#16#8093 değerini yayınlar (LADDR'de belirtilen adreste etkinleştirilebilen veya devre dışı bırakılabilen bir donanım nesnesi yok).

NOT

Devre dışı bırakma veya etkinleştirme görevi, çevrim kontrol noktasından çok sayıda geçiş gerektirir. Bu nedenle, programlanmış bir döngüde bu tür bir görevin sona ermesini bekleyemezsiniz.

Fonksiyon açıklaması

D_ACT_DP, asenkron bir yordamdır, bu da, görevin işlenmesinin, D_ACT_DP yordamının birkaç kez yürütülecek şekilde uzaması anlamına gelir. Görevi, D_ACT_DP'yi REQ = 1 ile çağırarak başlatabilirsiniz.
RET_VAL ve BUSY çıkış parametreleri, görevin durumunu bildirir.

Uygulama

Bir CPU'da, gerçekten mevcut olmayan veya o anda gerekli olmayan PROFINET IO cihazlarını yapılandırırsanız, CPU, buna rağmen düzenli aralıklarla bu PROFINET IO cihazlarına erişmeye devam eder. CPU erişimi, bu cihazlar devre dışı bırakıldıktan sonra sona erer. İlgili hata olayları artık oluşmaz.

Örnekler

Bir makine üreticisi açısından bakıldığında, makinelerin seri üretiminde çok sayıda olası cihaz opsiyonu mevcuttur. Ancak teslim edilen her makine, seçilen opsiyonlardan oluşan sadece bir kombinasyon içerir.
Üretici, bu olası makine opsiyonlarından her birini bir PROFINET IO cihazı olarak yapılandırır. Üretici bunu, olası tüm opsiyonları içeren ortak bir kullanıcı programı geliştirebilmek ve bu programı sürdürebilmek için yapar. D_ACT_DP, bir makine yol alırken mevcut olmayan tüm PROFINET IO cihazlarını devre dışı bırakmanıza olanak sağlar.
Benzer bir durum, her zaman aralarından sadece az sayıdaki opsiyonun kullanıldığı çok sayıda takım opsiyonuna sahip olan takım tezgahlarında ortaya çıkar. Bu takımlar PROFINET IO cihazları olarak uygulanır. Kullanıcı programı, D_ACT_DP ile o anda ihtiyaç duyulan takımları etkinleştirir ve daha sonra gerekenleri devre dışı bırakır.

Bir görevin tanımlanması

Bir devre dışı bırakma veya etkinleştirme görevini başlattıysanız ve görev tamamen işlenmeden önce D_ACT_DP yordamını yeniden çağırırsanız, yordamın davranışı, yeni görevin aynı görevle ilişkili olup olmadığına bağlı olarak şekillenir. LADDR giriş parametresi aynı ise, çağrı, ardıl çağrı olarak değerlendirilir.

PROFINET IO cihazlarının devre dışı bırakılması

Bir PROFINET IO cihazını D_ACT_DP ile devre dışı bırakırsanız, cihazın proses çıkışları, yapılandırılmış yedek değerlere veya 0'a (güvenli durum) ayarlanır. Atanan PROFINET IO kontrolörü, bu bileşenleri adreslemeye devam etmez. PROFINET IO kontrolöründeki veya CPU'nun hata LED'leri, devre dışı bırakılan PROFINET IO cihazlarını hatalı veya mevcut değil olarak bildirmez.

CPU, devre dışı bırakılan PROFINET IO cihazlarının girişlerinin proses görünümünü 0 ile günceller. Bu nedenle CPU, devre dışı bırakılan PROFINET IO cihazlarını, aynen arızalanan PROFINET IO cihazları gibi ele alır.

Daha önce devre dışı bırakılmış olan bir PROFINET IO cihazının kullanıcı verilerine programınız üzerinden doğrudan erişirseniz, sistem davranışı, blok için seçilen hata işleme şekline bağlı olur:

- Global hata işleme etkinleştirildiyse, sistem, teşhis tamponuna bir erişim hatası yol verme olayı girer ve RUN işletim durumunda kalır.
- Modüle özel hata işleme etkinleştirildiyse, sistem, hata yapısına bir hata sebebi girer. Hata sebebini GET_ERROR_ID (Sayfa 290) yordamıyla sorgulayabilirsiniz.

  Bir okuma erişimi sırasında oluşan bir hata, 0 değerini verir. Hata işlemeyle ilgili daha fazla bilgiyi, "Olayların yürütülmesiyle ilgili öncelikler ve bekleme kuyruğu" (Sayfa 80) başlığında bulabilirsiniz.

Devre dışı bırakılmış bir PROFINET IO cihazına bir yordamla (RD_REC (Sayfa 352) gibi) erişmeye çalışırsanız, mevcut olmayan bir PROFINET IO cihazı için olduğu gibi RET_VAL 'deki hata bilgilerinin aynılarını elde edersiniz.

Devre dışı bırakma işleminden sonra bir PROFINET IO istasyonu D_ACT_DP nedeniyle devreden çıkarsa, işletim sistemi devre dışı kalma durumunu algılamaz.

PROFINET IO cihazlarını etkinleştir

Bir PROFINET IO cihazını D_ACT_DP ile tekrar etkinleştirirseniz, ilgili PROFINET IO kontrolörü, bileşeni (devreden çıkan bir PROFINET IO istasyonunun geri dönüşünde olduğu gibi) yapılandırır ve parametrelendirir. Bu etkinleştirme işlemi, bileşen kullanıcı verilerini aktarabilir hale geldiğinde sonlandırılır.

D_ACT_DP yordamıyla, erişimin mümkün olmadığı (örneğin veriyolundan fiziksel olarak ayrıldığı için) bir PROFINET IO cihazını etkinleştirmeye çalışırsanız, yordam, desantral çevrebirim için yapılandırılmış parametrelendirme süresi dolduktan sonra W#16#80A7 hata kodunu yayınlar. PROFINET IO cihazının etkinleştirilmesi ve etkinleştirilmiş PROFINET IO cihazına erişilememesi durumu, sistem teşhisinde ilgili bir ifadenin görüntülenmesine neden olur.

Ardından PROFINET IO cihazına tekrar erişilebiliyorsa, bu durum standart sistem davranışına neden olur.

NOT

Bir PROFINET IO cihazının etkinleştirilmesi zaman alabilir. Devam eden bir etkinleştirme görevini iptal etmek istiyorsanız, D_ACT_DP'yi LADDR için kullanılan aynı değerle ve MODE = 2 ile başlatın. D_ACT_DP çağrısını MODE = 2 ile, etkinleştirme görevinin başarıyla iptal edilmesi durumu RET_VAL = 0 aracılığıyla görüntülenene kadar tekrarlayın.

Parametre

Aşağıdaki tabloda, D_ACT_DP yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Açıklama
REQ	IN	Bool	Seviye tetiklemeli kumanda parametresi REQ = 1: Etkinleştirme veya devre dışı bırakma işleminin yürütülmesi
MODE	IN	USInt	Görev kodu Olası değerler: • 0: Adreslenmiş bileşenin etkinleştirildiği veya devre dışı bırakıldığıyla ilgili bilgileri talep et (RET_VAL aracılığıyla yayınlama) • 1: PROFINET IO cihazını etkinleştir • 2: PROFINET IO cihazını devre dışı bırak
LADDR	IN	HW_DEVICE	PROFINET IO cihazının donanım kodu (HW_Device) Numara, Ağ görünümündeki PROFINET IO cihazının özelliklerinden veya standart değişken tablosundaki "Sistem sabitleri" sekmesinden öğrenilebilir. Burada, hem cihaz teşhisi için kullanılan kod hem de işletim durumu değişimi için kullanılan ID belirtildiyse, cihaz teşhisi için kullanılan kodu kullanmanız gerekir.
RET_VAL	OUT	Int	Program fonksiyonu yürütürken bir hata oluşursa, geri dönüş değeri bir hata kodu içerir.
BUSY	OUT	Bool	Aktif kod: • BUSY = 1: Görev hâlâ aktif. • BUSY = 0: Görev tamamlandı.

RET_VAL parametresi

Hata kodu* (W#16#...)	Anlamı
0000	Görev hatasız bir şekilde gerçekleştirildi.
0001	PROFINET IO cihazı aktif (bu hata kodu sadece MODE = 0 iken mümkündür).
0002	PROFINET IO cihazı devre dışı (bu hata kodu sadece MODE = 0 iken mümkündür).
7000	REQ = 0 ile ilk çağrı: LADDR'de belirtilen görev aktif değil; BUSY, 0 değerine sahiptir.
7001	REQ = 1 ile ilk çağrı: Program, LADDR'de belirtilen görevi tetikledi. BUSY, 1 değerine sahiptir.
7002	Ara çağrı (REQ ilgili değil). Etkinleştirilen görev hâlâ aktif; BUSY, 1 değerine sahiptir.
8090	• LADDR'de belirtilen adresle bir modül yapılandırmadınız. • CPU'nuzu I-Slave/I-Device olarak çalıştırıyorsunuz ve LADDR'de bu I-Slave'in/I-Device'ın bir adresini belirttiniz.
8092	O anda adreslenen PROFINET IO cihazının devre dışı bırakılması işlemi (MODE = 2), etkinleştirme işlemi aracılığıyla (MODE = 1) iptal edilemez. Bileşeni daha sonra etkinleştirin.
8093	LADDR'de belirtilen adres, etkinleştirilebilen veya devre dışı bırakılabilen bir PROFINET IO cihazına ait değil, veya MODE parametresi bilinmiyor.

Hata kodu* (W#16#...)	**Anlamı**
8094	Bir takım değiştirme portu için potansiyel bir partner olan bir cihazı etkinleştirmeyi denediniz. Bu takım değiştirme portunda o anda zaten başka bir cihaz etkinleştirilmiş durumda. Etkinleştirilen cihaz etkinleştirilmiş durumda kalır.
80A0	CPU ile IO kontrolörü arasındaki iletişimde hata.
80A1	Adreslenen bileşen, parametrelendirilemiyor. (Bu hata kodu, sadece MODE = 1 durumunda mümkündür.) Not: Bu bileşen, etkinleştirilmiş durumdaki cihaz parametrelendirilirken tekrar devre dışı kalırsa, D_ACT_DP yordamı, hata bilgilerini sağlar. Münferit bir modülün parametrelendirme işlemi başarılı olmazsa, D_ACT_DP, W#16#0000 hata bilgisini yayınlar.
80A3	Etkilenen PROFINET IO kontrolörü bu fonksiyonu desteklemez.
80A4	CPU, bu fonksiyonu harici bir PROFINET IO kontrolörü için desteklemez.
80A6	PROFINET IO cihazında yuva hatası; tüm kullanıcı verilerine erişilemiyor (bu hata kodu sadece MODE = 1 iken mevcuttur). Not: D_ACT_DP, bu hata bilgisini sadece, etkinleştirilmiş durumdaki bileşenin, parametrelendirme işleminden sonra ve D_ACT_DP yordamının yürütülmesi tamamlanmadan önce tekrar devreden çıkması durumunda yayınlar. Tek bir modül mevcut değilse, D_ACT_DP, W#16#0000 hata bilgisini yayınlar.
80A7	Etkinleştirme sırasında bir zaman aşımı oluştu: Uzak cihaza erişilemiyor veya Merkezi ve desantral çevrebirimler için parametrelendirme süresini çok kısa olacak şekilden ayarladınız. Uzak cihazın durumu "etkinleştirildi", ancak erişim mümkün değil.
80AA	PROFINET IO cihazında hatalarla etkinleştirme: Konfigürasyondaki farklılıklar
80AB	PROFINET IO cihazında hatalarla etkinleştirme: Parametrelendirme hatası
80AC	PROFINET IO cihazında hatalarla etkinleştirme: Bakım gerekli
80C1	D_ACT_DP başlatıldı ve başka bir adresle devam ettirildi (bu hata kodu MODE = 1 ve MODE = 2 iken mümkündür).
80C3	• Geçici kaynak hatası: CPU, mevcut durumda mümkün olan maksimum sayıda etkinleştirme ve devre dışı bırakma görevini (8) işlemektedir. (Bu hata kodu, sadece MODE = 1 ve MODE = 2 ise mümkündür.) • CPU, şu anda değiştirilmiş bir konfigürasyon alıyor. Şu anda hiçbir PROFINET IO cihazını etkinleştiremezsiniz/devre dışı bırakamazsınız.
80C6	PROFINET: Kullanıcı tarafından girilmeyen görevler, yeniden başlatma sırasında atılır.
Genel hata bilgileri	Hataya erişimle ilgili bilgileri, GET_ERROR_ID (Sayfa 290) yordamının açıklamasında bulabilirsiniz.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.	

9.3.8 RDREC, WRREC ve RALRM için kullanılan STATUS parametreleri

STATUS çıkış parametresi, şu yapıda ARRAY[1...4] OF BYTE olarak hata bilgileri içermektedir:

Tablo 9-85 STATUS çıkış Array'i

Array elemanı	Adı	Açıklama
STATUS[1]	Function_Num	• Hiçbir hata yoksa B#16#00 • DPV1-PDU'nun fonksiyon ID'si: Bir hata oluşursa, , B#16#80 VEYA aracılığıyla bağlanır (veri seti okunurken: B#16#DE ile; veri seti yazılırken: B#16#DF ile). Hiçbir DPV1 protokol elemanı kullanılmıyorsa, B#16#C0 yayınlanır.
STATUS[2]	Error_Decode	Hata ID'sinin adresi
STATUS[3]	Error_Code_1	Hata ID'si
STATUS[4]	Error_Code_2	Hata ID'sinin üreticiye özel olarak genişletilmesi

Tablo 9-86 STATUS[2] değerleri

Error_decode (B#16#....)	Kaynak	Açıklama
00 ila 7F	CPU	Hata yok ve uyarı yok
80	DPV1	IEC 61158-6 uyarınca hata
81 ila 8F	CPU	B#16#8x , yordamının "x." çağrı parametresinde bir hata gösteriyor.
FE, FF	DP profili	Profile özel hata

Tablo 9-87 STATUS[3] değerleri

Error_decode (B#16#....)	Error_code_1 (B#16#....)	Açıklama (DVP1)	Açıklama
00	00		Hata yok, uyarı yok
70	00	Rezerve, reddet	İlk çağrı; aktif veri seti aktarımı yok
	01	Rezerve, reddet	İlk çağrı; veri seti aktarımı başladı
	02	Rezerve, reddet	Ara çağrı; veri seti aktarımı zaten aktif
80	90	Rezerve, aktar	Geçersiz mantıksal başlangıç adresi
	92	Rezerve, aktar	Variant işaretçisi için izin verilmeyen tip
	93	Rezerve, aktar	ID veya F_ID üzerinden adreslenen DP bileşeni yapılandırılmadı.
	96		"RALRM (Sayfa 359)" yordamı; OB'nin yol alma bilgilerini, yönetim bilgilerini, Header bilgilerini ve diğer alarm bilgilerini sağlayamaz. Şu OB'lerde, etkilenen DP-Slave'in güncel teşhis mesajlarını asenkron olarak okumak için (OB'nin yol alma bilgilerindeki adres bilgileri) "DPNRM_DG (Sayfa 378)" yordamını kullanabilirsiniz: • Proses alarmı (Sayfa 70) • Durum (Sayfa 77), güncelleme (Sayfa 77) veya profil (Sayfa 77) • Teşhis hatası alarmı (Sayfa 72) • Modüllerin çıkartılması veya takılması (Sayfa 74)
	A0	Okuma hatası	Modülden okuma sırasında negatif onay
	A1	Yazma hatası	Modüle yazma sırasında negatif onay

Error_decode (B#16#....)	Error_code_1 (B#16#....)	Açıklama (DVP1)	Açıklama
80	A2	Modülün devre dışı kalması	Katman 2'de DP protokol hatası (örnek: Slave'in devre dışı kalması veya veriyolu sorunları)
	A3	Rezerve, aktar	• PROFIBUS DP: Direct-Data-Link-Mapper veya User-Interface/User ile DP protokol hatası • PROFINET IO: Genel CM hatası
	A4	Rezerve, aktar	İletişim veriyolundaki iletişim kesildi
	A5	Rezerve, aktar	-
	A7	Rezerve, aktar	DP-Slave veya modül dolu (geçici hata).
	A8	Sürüm çakışması	DP-Slave veya modül uyumsuz sürümler bildiriyor.
	A9	Fonksiyon desteklenmiyor	Fonksiyon, DP-Slave veya modül tarafından desteklenmiyor
	AA ila AF	Kullanıcıya özel	DP-Slave veya modül, uygulamada üreticiye özel hatalar bildiriyor. DP-Slave'in veya modülün üreticisine ait dokümantasyonu kontrol.
	B0	Geçersiz indeks	Modüldeki veri seti bilinmiyor; izin verilmeyen veri seti numarası ≥ 256
	B1	Yazma sırasında uzunluk hatası	RECORD parametresindeki uzunluk bilgileri yanlış. • RALRM'de: AINFO'da uzunluk hatası **Not**: STEP 7'nin çevrimiçi bilgi sistemi, AINFO tarafından döndürülen tamponların nasıl değerlendirilmesi gerektiğiyle ilgili bilgilere doğrudan erişim olanağı sunar. • RDREC (Sayfa 352) ve WRREC (Sayfa 352)'de: MLEN'de uzunluk hatası
	B2	Geçersiz yuva	Yapılandırılan yuva dolu değil.
	B3	Tip çakışması	Gerçek modül tipi, belirtilen modül tipine karşılık gelmiyor.
	B4	Geçersiz aralık	DP-Slave veya modül, geçersiz veri alanına erişim olduğunu bildiriyor.
	B5	Durum çakışması	DP-Slave veya modül çalışmaya hazır değil
	B6	Erişim reddedildi	DP-Slave veya modül, erişimi reddediyor.
	B7	Geçersiz aralık	DP-Slave veya modül, bir parametre veya değer için geçersiz bir aralık bildiriyor.
	B8	Geçersiz parametre	DP-Slave veya modül, geçersiz bir parametre bildiriyor.
	B9	Geçersiz tip	DP-Slave veya modül, geçersiz bir tip bildiriyor. • RDREC (Sayfa 352)'de: Tampon çok küçük (alt kümeler okunamıyor) • WRREC (Sayfa 352)'de: Tampon çok küçük (alt kümelere yazılamıyor)
	BA ila BF	Kullanıcıya özel	DP-Slave veya modül, erişimde üreticiye özel hatalar bildiriyor. DP-Slave'in veya modülün üreticisine ait dokümantasyonu kontrol.

Error_decode (B#16#....)	Error_code_1 (B#16#....)	Açıklama (DVP1)	Açıklama
80	C0	Okuma sırasında kısıtlama çakışması	• WRREC (Sayfa 352)'de: Veriler, sadece CPU STOP işletim türünde bulunuyorsa yazılabilir. **Not**: Bu, verilerin kullanıcı programından yazılamadığı anlamına gelir. Verileri sadece çevrimiçi olarak bir PG/PC ile yazabilirsiniz. • RDREC (Sayfa 352)'de: Modül, veri setini iletir, ancak ya hiçbir veri mevcut değil ya da veriler ancak CPU STOP işletim türünde ise okunabilir. **Not**: Sadece STOP işletim türünde yazılabiliyorsa, kullanıcı programı aracılığıyla değerlendirme yapmak mümkün değildir. Bu durumda verileri sadece çevrimiçi olarak bir PG/PC ile okuyabilirsiniz.
	C1	Yazma sırasında kısıtlama çakışması	Aynı veri seti için modüle gönderilen önceki yazma talebinin verileri modül tarafından henüz işlenmedi.
	C2	Kaynak tamamen kullanıldı	Modül, şu anda bir CPU için mümkün olan maksimum sayıda görevi işliyor.
	C3	Kaynak mevcut değil	Gereken işletim kaynakları şu anda tamamen kullanılmış durumdadır.
	C4		Dahili geçici hata. Görev gerçekleştirilemedi. Görevi tekrarlayın. Bu hata sıkça ortaya çıkarsa, kurulumunuzu elektrik arızası kaynakları açısından kontrol edin.
	C5		DP-Slave veya modül mevcut değil
	C6		Veri seti aktarımı, öncelik sınıfı sınıflandırması nedeniyle iptal edildi.
	C7		Görev, DP-Master'ın sıcak başlatma veya soğuk başlatma işlemi nedeniyle iptal edildi.
	C8 ila CF		DP-Slave veya modül, üreticiye özel kaynak hataları bildiriyor. DP-Slave'in veya modülün üreticisine ait dokümantasyonu kontrol.
	Dx	Kullanıcıya özel	DP-Slave'e özel. Bkz. DP-Slave'in açıklaması.
81	00 ila FF		İlk çağrı parametresinde hata ("RALRM (Sayfa 359)"de: MODE)
	00		İzin verilmeyen işletim türü
82	00 ila FF		İkinci çağrı parametresinde hata
88	00 ila FF		Sekizinci çağrı parametresinde hata ("RALRM (Sayfa 359)"de: TINFO) **Not**: STEP 7'nin çevrimiçi bilgi sistemi, TINFO tarafından döndürülen tamponların nasıl değerlendirilmesi gerektiğiyle ilgili bilgilere doğrudan erişim olanağı sunar.
	01		Yanlış sözdizimi ID'si
	23		Miktar yapısı aşıldı veya hedef aralık çok küçük
	24		Yanlış aralık ID'si
	32		DB/DI numarası, kullanıcı aralığının dışında
	3A		DB/DI numarası, aralık ID'si DB/DI için SIFIR veya belirtilen DB/DI mevcut değil.

Error_decode (B#16#....)	Error_code_1 (B#16#....)	Açıklama (DVP1)	Açıklama
89	00 ila FF		Dokuzuncu çağrı parametresinde hata ("RALRM (Sayfa 359)"de: AINFO) **Not**: STEP 7'nin çevrimiçi bilgi sistemi, AINFO tarafından döndürülen tamponların nasıl değerlendirilmesi gerektiğiyle ilgili bilgilere doğrudan erişim olanağı sunar.
	01		Yanlış sözdizimi ID'si
	23		Miktar yapısı aşıldı veya hedef aralık çok küçük
	24		Yanlış aralık ID'si
	32		DB/DI numarası, kullanıcı aralığının dışında
	3A		DB/DI numarası, aralık ID'si DB/DI için SIFIR veya belirtilen DB/DI mevcut değil.
8A	00 ila FF		Onuncu çağrı parametresinde hata
8F	00 ila FF		On beşinci çağrı parametresinde hata
FE, FF	00 ila FF		Profile özel hata

STATUS[4] Array elemanı

DPV1 hatalarında DP-Master, CPU'ya ve yordama STATUS'u [4] aktarır. Bu değer, RDREC'de aşağıdaki istisnalarla birlikte DPV1 hatası olmadan 0'a ayarlanır:

- STATUS[4], , MLEN > RECORD'dan gelen hedef aralık uzunluğu ise RECORD'un hedef aralık uzunluğunu içerir.
- Veri setinin gerçek uzunluğu < MLEN < RECORD'dan gelen hedef aralık uzunluğu ise STATUS[4] = MLEN olur.
- STATUS[4] > 255 ise STATUS[4] = 0 olur; ayarlanmalıdır

PROFINET IO'da STATUS[4], 0 değerine sahiptir.

9.3.9 Diğer konular

9.3.9.1 DPRD_DAT ve DPWR_DAT (tutarlı verileri oku/yaz)

DPRD_DAT (tutarlı verileri oku) yordamıyla verilerin bir veya daha fazla Bayt'ını tutarlı bir şekilde okuyabilir ve DPWR_DAT (tutarlı verileri yaz) yordamıyla verilerin bir veya daha Bayt'ı tutarlı bir şekilde aktarabilirsiniz. DPRD_DAT ve DPWR_DAT yordamını PROFINET ve PROFIBUS için kullanabilirsiniz.

Tablo 9-88 DPRD_DAT ve DPWR_DAT yordamları

LAD/FBD	SCL	Açıklama
	**ret_val := DPRD_DAT(laddr:=_word_in_, record=>_variant_out_);**	DPRD_DAT yordamıyla, aşağıdaki saklama konumlarından birinde bulunan modüllerden veya alt modüllerden verilerin bir veya daha fazla Bayt'ını okuyabilirsiniz: • Yerel temel G/Ç • DP-Slave • PROFINET IO cihazı CPU, aktarılan verileri tutarlı şekilde aktarır. Veri aktarımı sırasında hiçbir hata oluşmazsa, okunan veriler CPU tarafından, RECORD parametresi aracılığıyla belirtilen hedef aralığa girilir. Hedef aralık, STEP 7'de seçilen modül için yapılandırdığınız aynı uzunluğa sahip olmalıdır. DPRD_DAT yordamı yürütülürken her seferinde sadece bir modülün veya alt modülün verilerine erişilebilir. Aktarım, yapılandırılan başlangıç adresinde başlar.
	**ret_val := DPWR_DAT(laddr:=_word_in_, record:=_variant_in_);**	RECORD'daki verileri aşağıdaki saklama konumlarına tutarlı bir şekilde aktarmak için DPWR_DAT yordamını kullanın: • Yerel temel cihazdaki adreslenmiş modül veya alt modül • DP Standart Slave • PROFINET IO cihazı Kaynak aralık, STEP 7'de seçilen modül veya alt modül için yapılandırdığınız aynı uzunluğa sahip olmalıdır.

- S7-1200 CPU'su, çevrebirim cihazlarındaki 1, 2 veya 4 Bayt'ın tutarlı şekilde okunmasını veya yazılmasını destekler. DPRD_DAT yordamıyla, 1, 2 veya 4 Bayt'tan farklı uzunluklardaki verileri tutarlı bir şekilde okuyabilir ve DPWR_DAT yordamıyla bu verileri tutarlı bir şekilde yazabilirsiniz.
- Bu yordamları, 1 Bayt veya daha fazla Bayt'lık veri alanları için kullanabilirsiniz. Erişim reddedildiğinde W#16#8090 hata kodu bildirilir.
- PROFINET, maksimum 1024 Bayt tutarlı verileri destekler. S7-1200 ile PROFINET cihazları arasında tutarlı aktarımlar gerçekleştirmek için bu yordamları kullanmanız gerekmez.

NOT

DPRD_DAT ve DPWR_DAT yordamlarını tutarlı verilerle kullanırsanız, bu tutarlı verileri, proses görünümünün otomatik güncellenmesinden almanız gerekir. Bu konuyla ilgili daha fazla bilgiyi, "PLC ile ilgili temel esaslar: Kullanıcı programının yürütülmesi" (Sayfa 60) başlığında bulabilirsiniz.

Tablo 9-89 Parametre

Parametre	Deklarasyon	Veri tipi	Açıklama
LADDR	IN	HW_IO (Word)	Verilerin kendisinden okunması gereken modülün donanım ID'si. (DPRD_DAT) Verilerin yazılması gereken modülün donanım ID'si.(DPWR_DAT) Donanım ID'sini, cihaz görünümündeki modülün özelliklerinde veya sistem sabitlerinde bulabilirsiniz.
RECORD	OUT	Variant	Okunan kullanıcı verilerinin hedef aralığı (DPRD_DAT) veya yazılması gereken kullanıcı verilerinin kaynak aralığı (DPWR_DAT). Bu aralık, seçilen modülde STEP 7 ile yapılandırdığınızla aynı boyutta olmalıdır.
RET_VAL	OUT	Int	Fonksiyonu aktifken bir hata oluşursa, geri dönüş değeri bir hata kodu içerir.

DPRD_DAT'ın çalışma şekli

LADDR parametresiyle DP Standart Slave'inin/PROFINET IO cihazının modülünü seçebilirsiniz. Adreslenmiş modülde bir erişim hatası oluşursa, W#16#8090 hata kodu yayınlanır.

RECORD parametresiyle, okunan verilerin hedef aralığını tanımlayabilirsiniz:

- Hedef aralık, en azından, seçilen modülün girişleri kadar uzun olmalıdır. Sadece girişler aktarılır; diğer Bayt'lar dikkate alınmaz. Modüler yapıya veya çok sayıda DP koduna sahip bir DP Standart Slave'den veri okuyorsanız, DPRD_DAT yordamı çağrısı başına sadece, yapılandırılmış donanım koduna ait bir modülün verilerine erişebilirsiniz. Çok küçük bir hedef aralık seçerseniz, RET_VAL parametresinde W#16#80B1 hata kodu yayınlanır.
- Aşağıdaki veri tipleri kullanılabilir: Byte, Char, Word, DWord, Int, UInt, USInt, SInt, DInt, UDInt. ARRAY veya STRUCT tipindeki kullanıcı tanımlı verilerinden (UDT) birinin veri yapısında bu veri tiplerinin kullanımına izin verilir.
- STRING veri tipi desteklenmez.
- Veri aktarımı sırasında hiçbir hata oluşmadıysa, okunan veriler, RECORD parametresi aracılığıyla belirtilen hedef aralığa girilir.

DPWR_DAT'ın çalışma şekli

LADDR parametresiyle DP Standart Slave'inin/PROFINET IO cihazının modülünü seçebilirsiniz. Adreslenmiş modülde bir erişim hatası oluşursa, W#16#8090 hata kodu yayınlanır.

RECORD parametresiyle, yazılması gereken verilerin kaynak aralığını tanımlayabilirsiniz:

- Kaynak aralık, en azından, seçilen modülün çıkışları kadar uzun olmalıdır. Sadece çıkışlar aktarılır; diğer Bayt'lar dikkate alınmaz. RECORD parametresindeki kaynak aralık yapılandırılmış modülün çıkışlarından daha uzunsa, sadece çıkışların maksimum uzunluğuna kadar olan veriler aktarılır. RECORD parametresindeki kaynak aralık yapılandırılmış modülün çıkışlarından daha kısaysa, RET_VAL parametresinde W#16#80B1 hata kodu yayınlanır.
- Aşağıdaki veri tipleri kullanılabilir: Byte, Char, Word, DWord, Int, UInt, USInt, SInt, DInt, UDInt. ARRAY veya STRUCT tipindeki kullanıcı tanımlı verilerinden (UDT) birinin veri yapısında bu veri tiplerinin kullanımına izin verilir.
- STRING veri tipi desteklenmez.
- Veriler senkron olarak aktarılır, yani yazma işlemi, yordamla birlikte tamamlanır.

Hata kodları

Tablo 9-90 DPRD_DAT ve DPWR_DAT'ın hata kodları

Hata kodu [1]	Açıklama
0000	Hata yok
8090	Şu durumlardan biri geçerlidir: • Belirtilen mantıksal ana adres için hiçbir modül yapılandırmadınız. • Tutarlı verilerin uzunluğuyla ilgili kısıtlamayı göz ardı ettiniz. • LADDR parametresindeki başlangıç adresini onaltılık formatta girmediniz.
8092	RECORD parametresi şu veri tiplerini destekler: Byte, Char, Word, DWord, Int, UInt, USInt, SInt, DInt, UDInt, and arrays of these types.
8093	LADDR'de belirtilen mantıksal adreste, tutarlı verilerin okunabileceği (DPRD_DAT) veya tutarlı verilerin yazılabileceği (DPWR_DAT) hiçbir DP modülü/PROFINET IO cihazı mevcut değildir.
80A0	G/Ç cihazlarına erişim sırasında bir erişim hatası algılandı (DPRD_DAT).
80B1	Belirtilen hedef aralığın (DPRD_DAT) veya kaynak aralığın (DPWR_DAT) uzunluğu, kullanıcı verilerinin STEP 7'de yapılandırılan uzunluğuyla aynı değil.
80B2	Harici DP arayüz modülüyle sistem hatası (DPRD_DAT) ve (DPWR_DAT)

[1] DPRD_DAT ve DPWR_DAT'ın hata kodları görüntülenirken Word veri tipini kullanın.

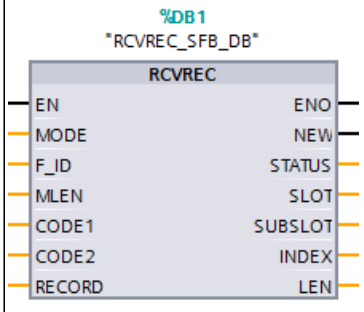
NOT

DPV1 Slave'lerine erişirseniz, bu Slave'lerin hata bilgileri DP-Master tarafından yordama aktarılabilir.

9.3.9.2 RCVREC (veri setini al)

Bir I-Device, üst seviyedeki bir kontrolörden gelen bir veri setini alabilir. Alma işlemi, kullanıcı programında RCVREC (veri setini al) yordamıyla gerçekleştirilir.

Tablo 9-91 RCVREC yordamı

LAD/FBD	SCL	Açıklama
	**"RCVREC_SFB_DB"(mode:=_int_in_, F_ID:=_uint_in_, mlen:=_uint_in_, code1:=_byte_in_, code2:=_byte_in_, new=>_bool_out_, status=>_dword_out_, slot=>_uint_out_, subslot=>_uint_out_, index=>_uint_out_ len=>_uint_out_, record:=_variant_inout_);**	RCVREC yordamıyla, üst seviyedeki bir kontrolörden gelen bir veri seti alınabilir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "RCVREC_SFB_DB", örnek DB'nin adıdır.

Yordam, şu işletim durumlarına sahiptir:

- I-Device'ın bir veri setinin alınması için bir talebe sahip olup olmadığını kontrol etme
- Veri setini, çıkış parametrelerinin kullanımına sunma
- Üst seviyedeki kontrolöre bir yanıt gönderme

Yordam tarafından uygulanan işletim durumunu MODE giriş parametresiyle belirleyebilirsiniz.

I-Device, RUN veya YOL ALMA işletim durumunda olmalıdır.

MLEN ile alınması gereken maksimum Bayt sayısını belirtebilirsiniz. RECORD hedef aralığının seçilen uzunluğu, en azından MLEN tarafından belirtilen Bayt'ların uzunluğu kadar olmalıdır.

Bir veri seti alınırsa (MODE = 1 veya MODE = 2), NEW çıkış parametresi, veri setinin RECORD'a kaydedildiğini bildirir. RECORD'un yeterli uzunluğa sahip olduğuna dikkat edin. LEN çıkış parametresi, alınan veri setinin Bayt cinsinden gerçek uzunluğunu içerir.

Üst seviyedeki kontrolöre pozitif bir yanıt vermek için CODE1 ve CODE2'yi 0'a ayarlayın. Alınan veri setinin reddedilmesi gerekiyorsa, üst seviyedeki kontrolöre giden negatif yanıtı CODE1'in 1. hata koduna ve CODE2'nin 2. hata koduna girin.

NOT

I-Device, bir veri setinin alınması için bir talep aldıysa, bu talebin belirli bir süre içinde sunulduğunu kabul etmeniz gerekir. Kabulün arından, bu süre içinde üst seviyedeki kontrolöre bir yanıt göndermeniz gerekir. Aksi takdirde I-Device'da bir zaman aşımı hatası ortaya çıkar, bunun üzerine I-Device'ın işletim sistemi, üst seviyedeki kontrolöre negatif bir yanıt gönderir. Süre değeriyle ilgili bilgileri, CPU'nuzun teknik verilerinde bulabilirsiniz.

STATUS çıkış parametresi, bir hata oluştuktan sonra hata bilgilerini alır.

İşletim durumları

RCVREC yordamının işletim durumunu MODE giriş parametresiyle belirleyebilirsiniz. Bu adım, aşağıdaki tabloda açıklanmaktadır:

MODE	Anlamı
0	Bir veri setinin alınması için bir talebin mevcut olup olmadığını kontrol etme I-Device'da, üst seviyedeki bir kontrolörün bir veri seti mevcutsa, yordam sadece NEW, SLOT, SUBSLOT, INDEX ve LEN çıkış parametrelerine yazar. Yordamı MODE = 0 ile birden çok kez çağırırsanız, çıkış parametresi sadece tek ve aynı talebi temel alır.
1	I-Device'ın herhangi bir alt yuvası için kullanılan bir veri setini alma I-Device'da, I-Device'ın herhangi bir alt yuvası için üst seviyedeki bir kontrolörün bir veri seti mevcutsa, yordam, çıkış parametresine yazar ve veri setini RECORD parametresine aktarır.
2	I-Device'ın belirli bir alt yuvası için kullanılan bir veri setini alma I-Device'da, I-Device'ın belirli bir alt yuvası için üst seviyedeki bir kontrolörün bir veri seti mevcutsa, yordam, çıkış parametresine yazar ve veri setini RECORD parametresine aktarır.
3	Üst seviyedeki kontrolöre pozitif bir yanıt gönderme Yordam, üst seviyedeki kontrolörün bir veri setinin alınmasına yönelik talebini kontrol eder, mevcut veri setini kabul eder ve üst seviyedeki kontrolöre pozitif bir onay gönderir.
4	Üst seviyedeki kontrolöre negatif bir yanıt gönderme Yordam, üst seviyedeki kontrolörün bir veri setinin alınmasına yönelik talebini kontrol eder, mevcut veri setini reddeder ve üst seviyedeki kontrolöre negatif bir onay gönderir. Reddetme nedenini, CODE1 ve CODE2 giriş parametrelerine girebilirsiniz.

NOT

Veri seti alındıktan sonra (NEW = 1), tam işlemenin uygulanmasını sağlamak için RCVREC yordamını iki kez çağırmanız gerekir. Bu sırada aşağıdaki sıraya dikkat edin:

- MODE = 1 veya MODE = 2 ile ilk çağrı
- MODE = 3 veya MODE = 4 ile ikinci çağrı

Parametre

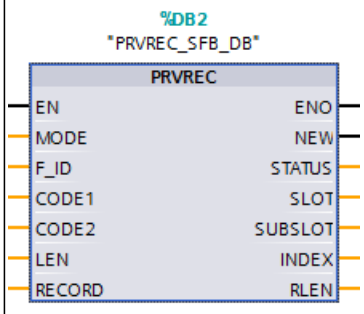
Aşağıdaki tabloda, RCVREC yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Açıklama
MODE	IN	Int	İşletim türü
F_ID	IN	HW_SUBMODULE	Alınması gereken veri seti için I-Device'ın aktarım alanındaki alt yuva (sadece MODE = 2 durumunda geçerlidir). Yüksek sözcük her zaman sıfıra ayarlanır.
MLEN	IN	Int	Alınması gereken veri setinin Bayt cinsinden maksimum uzunluğu
CODE1	IN	Bayt	Sıfır (MODE = 3 durumunda) ve/veya hata kodu 1 (MODE = 4 durumunda)
CODE2	IN	Bayt	Sıfır (MODE = 3 olduğunda) ve/veya hata kodu 2 (MODE = 4 olduğunda)
NEW	OUT	Bool	• MODE = 0: Yeni veri seti alındı • MODE = 1 veya 2: Veri seti RECORD'a aktarıldı
STATUS	OUT	DWord	Hata bilgisi. Daha fazla bilgi için bkz. "STATUS parametreleri" (Sayfa 366).
SLOT	OUT	HW_SUBMODULE	F_ID ile aynı
SUBSLOT	OUT	HW_SUBMODULE	F_ID ile aynı
INDEX	OUT	UInt	Alınan veri setinin numarası
LEN	OUT	UInt	Alınan veri setinin uzunluğu
RECORD	IN_OUT	Varyant	Alınan veri seti için kullanılan hedef aralık

9.3.9.3 PRVREC (veri setini hazırla)

Bir I-Device, üst seviyedeki bir kontrolörden gelen bir veri setinin sağlanmasına yönelik bir talebi alabilir. I-Device, PRVREC (veri setini sağla) yordamıyla, kullanıcı programındaki veri setini sağlar.

Tablo 9-92 PRVREC yordamı

LAD/FBD	SCL	Açıklama
	**"PRVREC_SFB_DB"(** **mode:=_int_in_,** **F_ID:=_uint_in_,** **code1:=_byte_in_,** **code2:=_byte_in_,** **len:=_uint_in_,** **new=>_bool_out_,** **status=>_dword_out_,** **slot=>_uint_out_,** **subslot=>_uint_out_,** **index=>_uint_out_** **rlen=>_uint_out_,** **record:=_variant_inout_);**	PRVREC yordamıyla, üst seviyedeki bir kontrolörden gelen bir veri setinin sağlanmasına yönelik bir talep alınabilir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "PRVREC_SFB_DB", örnek DB'nin adıdır.

Yordam, şu işletim durumlarına sahiptir:

- I-Device'ın bir veri setinin sağlanması için bir talebe sahip olup olmadığını kontrol etme
- Talep edilen veri setini üst seviyedeki kontrolöre aktarma
- Üst seviyedeki kontrolöre bir yanıt gönderme

Yordam tarafından uygulanan işletim durumunu MODE giriş parametresiyle belirleyebilirsiniz.

I-Device, RUN veya YOL ALMA işletim durumunda olmalıdır.

LEN ile gönderilmesi gereken veri setinin sahip olması gereken maksimum Bayt sayısını girin. RECORD hedef aralığının seçilen uzunluğu, en azından LEN tarafından belirtilen Bayt'ların uzunluğu kadar olmalıdır.

Bir veri setinin sağlanmasına yönelik bir talep mevcutsa, (MODE = 0), NEW çıkış parametresi TRUE olarak ayarlanır.

Bir veri setinin sağlanmasına yönelik talep kabul edilirse, üst seviyedeki kontrolöre pozitif bir yanıt göndermek için talep edilen veri setini RECORD'a yazın ve 0 değerini CODE1 ve CODE2'ye yazın. Bir veri setinin sağlanmasına yönelik talebin reddedilmesi gerekiyorsa, üst seviyedeki kontrolöre giden negatif yanıtı CODE1'in 1. hata koduna ve CODE2'nin 2. hata koduna girin.

NOT

I-Device, bir veri setinin sağlanması için bir talep aldıysa, bu talebin belirli bir süre içinde sunulduğunu onaylamanız gerekir. Kabulün arından, bu süre içinde üst seviyedeki kontrolöre bir yanıt göndermeniz gerekir. Aksi takdirde I-Device'da bir zaman aşımı hatası ortaya çıkar, bunun üzerine I-Device'ın işletim sistemi, üst seviyedeki kontrolöre negatif bir yanıt gönderir. Süre değeriyle ilgili bilgileri, CPU'nuzun teknik verilerinde bulabilirsiniz.

STATUS çıkış parametresi, bir hata oluştuktan sonra hata bilgilerini alır.

İşletim durumları

PRVREC yordamının işletim durumunu MODE giriş parametresiyle belirleyebilirsiniz. Bu adım, aşağıdaki tabloda açıklanmaktadır:

MODE	Anlamı
0	Bir veri setinin sağlanması için bir talebin mevcut olup olmadığını kontrol etme I-Device'da üst seviyedeki kontrolörün, bir veri setinin sağlanmasına yönelik bir talebi mevcutsa, yordam sadece NEW, SLOT, SUBSLOT, INDEX ve RLEN çıkış parametrelerine yazar. Yordamı MODE = 0 ile birden çok kez çağırırsanız, çıkış parametresi sadece tek ve aynı talebi temel alır.
1	I-Device'ın herhangi bir alt yuvası için kullanılan bir veri setinin sağlanmasına yönelik bir talebi alma I-Device'da, I-Device'ın herhangi bir alt yuvası için üst seviyedeki bir kontrolörden gelen bu tür bir talep mevcutsa, yordam, çıkış parametresine yazar.
2	I-Device'ın belirli bir alt yuvası için kullanılan bir veri setinin sağlanmasına yönelik bir talebi alma I-Device'da, I-Device'ın belirli bir alt yuvası için üst seviyedeki bir kontrolörden gelen bu tür bir talep mevcutsa, yordam, çıkış parametresine yazar.
3	Bir veri setini sağlama ve üst seviyedeki kontrolöre pozitif bir yanıt gönderme Yordam, üst seviyedeki kontrolörün bir veri setinin sağlanmasına yönelik talebini kontrol eder, veri setini RECORD'da sağlar ve üst seviyedeki kontrolöre pozitif bir onay gönderir.
4	Üst seviyedeki kontrolöre negatif bir yanıt gönderme Yordam, üst seviyedeki kontrolörün bir veri setinin sağlanmasına yönelik talebini kontrol eder, bu talebi reddeder ve üst seviyedeki kontrolöre negatif bir onay gönderir. Reddetme nedenini, CODE1 ve CODE2 giriş parametrelerine girebilirsiniz.

NOT

Bir talep alındıktan sonra (NEW = 1), tam işlemenin uygulanmasını sağlamak için PRVREC yordamını iki kez çağırmanız gerekir. Bu sırada aşağıdaki sıraya dikkat edin:

- MODE = 1 veya MODE = 2 ile ilk çağrı
- MODE = 3 veya MODE = 4 ile ikinci çağrı

Parametre

Aşağıdaki tabloda, PRVREC yordamının parametreleri gösterilmektedir:

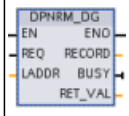
Parametre	Deklarasyon	Veri tipi	Açıklama
MODE	IN	Int	İşletim türü
F_ID	IN	HW_SUBMODULE	Gönderilmesi gereken veri seti için I-Device'ın aktarım alanındaki alt yuva (sadece MODE = 2 durumunda geçerlidir). Yüksek sözcük her zaman sıfıra ayarlanır.
CODE1	IN	Bayt	Sıfır (MODE = 3 durumunda) ve/veya hata kodu 1 (MODE = 4 durumunda)
CODE2	IN	Bayt	Sıfır (MODE = 3 olduğunda) ve/veya hata kodu 2 (MODE = 4 olduğunda)
LEN	IN	UInt	Gönderilmesi gereken veri setinin Bayt cinsinden maksimum uzunluğu
NEW	OUT	Bool	Yeni veri seti, üst seviyedeki kontrolör tarafından talep edildi.
STATUS	OUT	DWord	Hata bilgisi. Daha fazla bilgi için bkz. "STATUS parametreleri" (Sayfa 366).

Parametre	Deklarasyon	Veri tipi	Açıklama
SLOT	OUT	HW_SUBMODULE	F_ID ile aynı
SUBSLOT	OUT	HW_SUBMODULE	F_ID ile aynı
INDEX	OUT	UInt	Gönderilmesi gereken veri setinin numarası
RLEN	OUT	UInt	Gönderilmesi gereken veri setinin uzunluğu
RECORD	IN_OUT	Varyant	Hazırlanan veri seti

9.3.9.4 DPNRM_DG (bir DP-Slave'in teşhis verilerini oku)

DPNRM_DG (teşhis verilerini oku) yordamını PROFIBUS için kullanabilirsiniz.

Tablo 9-93 DPNRM_DG yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := DPNRM_DG(req:=_bool_in_, laddr:=_word_in_, record=>_variant_out_, busy=>_bool_out_);**	DPNRM_DG yordamıyla, bir DP-Slave'in güncel teşhis verilerini, EN 50 170 Cilt 2, PROFIBUS'ta belirtilen formatta okuyabilirsiniz. Okunan veriler, hatasız bir veri aktarımından sonra, RECORD tarafından belirtilen hedef aralığa girilir.

Tablo 9-94 DPNRM_DG parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQ=1: Okuma talebi
LADDR	IN	HW_DPSLAVE	DP-Slave'in yapılandırılmış teşhis adresi: G/Ç cihazının yerine istasyonun adresi olmalıdır. Bu teşhis adresini belirlemek için, "Cihaz konfigürasyonu" ağ görünümünde istasyonu seçin (cihazın görünümünü değil). Adresleri onaltılık formatta girin. Örnek: Teşhis adresi 1022, LADDR:=W#16#3FE anlamına gelir.
RET_VAL	OUT	Int	Fonksiyonu aktifken bir hata oluşursa, geri dönüş değeri bir hata kodu içerir. Hiçbir hata oluşmazsa, gerçekten aktarılan verilerin uzunluğu RET_VAL'e girilir.
RECORD	OUT	Variant	Okunan teşhis verileri için kullanılan hedef aralık. Okunması gereken veri setinin minimum uzunluğu (veya hedef aralık) 6 Bayt'tır. Gönderilmesi gereken veri setinin maksimum uzunluğu 240 Bayt'tır. Standart Slave'ler 240 Bayt'tan daha fazla, maksimum 244 Bayt teşhis verisi sağlayabilir. Daha sonra sadece ilk 240 Bayt hedef aralığa aktarılır ve taşma Bit'i verilerde ayarlanır.
BUSY	OUT	Bool	BUSY=1: Okuma görevi henüz tamamlanmadı.

Okuma görevini, REQ giriş parametresini, çağrıdaki DPNRM_DG yordamına 1 değerini atayarak başlatabilirsiniz. Okuma görevi asenkron olarak gerçekleştirilir, diğer bir deyişle, bunun için DPNRM_DG yordamının birkaç kez çağrılması gerekir. Görevin durumu, RET_VAL ve BUSY çıkış parametreleri tarafından belirtilir.

Tablo 9-95 Slave teşhis verilerinin yapısı

Bayt	Açıklama
0	Katılımcı durumu 1
1	Katılımcı durumu 2
2	Katılımcı durumu 3

Bayt	Açıklama
3	Master katılımcı numarası
4	Üretici ID'si (üst Bayt)
5	Üretici ID'si (alt Bayt)
6 ...	Slave'e özel diğer teşhis bilgileri

Tablo 9-96 DPNRM_DG yordamının hata kodları

Hata kodu	Açıklama	Sınırlama
0000	Hata yok	-
7000	REQ = 0 ile ilk çağrı: Hiçbir veri aktarımı aktif değil; BUSY, 0 değerine sahiptir.	-
7001	REQ = 1 ile ilk çağrı: Hiçbir veri aktarımı aktif değil; BUSY, 1 değerine sahiptir.	Desantral G/Ç
7002	Ara çağrı (REQ ilgili değil): Veri aktarımı halihazırda aktif; BUSY, 1 değerine sahiptir.	Desantral G/Ç
8090	Belirtilen mantıksal ana adres geçersiz: Ana adres mevcut değil.	-
8092	RECORD parametresi şu veri tiplerini destekler: Byte, Char, Word, DWord, Int, UInt, USInt, SInt, DInt, UDInt, and arrays of these types.	-
8093	• Bu yordama, LADDR tarafından belirtilen modül için izin verilmez (S7-1200 için S7-DP modüllerine izin verilir). • LADDR, istasyon yerine G/Ç cihazını belirtir. LADDR'nin teşhis adresini belirlemek için, "Cihaz konfigürasyonu" ağ görünümünde istasyonu seçin (cihazın görünümünü değil).	-
80A2	• Katman 2'de DP protokol hatası (örnek: Slave'in devre dışı kalması veya veriyolu sorunları) • ET200S'de veri seti, DPV0 modunda okunamaz.	Desantral G/Ç
80A3	User-Interface/User ile DP protokol hatası	Desantral G/Ç
80A4	İletişim veriyolunda iletişim sorunu	Hata, CPU ile harici DP arayüz modülü arasında oluşuyor.
80B0	• Bu yordama, modül tipi için izin verilmiyor. • Modül, veri setini algılamıyor. • Veri seti numarası 241'e izin verilmiyor.	-
80B1	RECORD parametresinde belirtilen uzunluk yanlış.	Belirtilen uzunluk > Veri seti uzunluğu
80B2	Yapılandırılan yuva dolu değil.	-
80B3	Gerçek modül tipi, gereken modül tipine karşılık gelmiyor.	-
80C0	Teşhis bilgileri mevcut değil.	-
80C1	Modüldeki aynı veri seti için kullanılan önceki yazma talebinin verileri modül tarafından henüz işlenmedi.	-
80C2	Modül, şu anda bir CPU için mümkün olan maksimum sayıda görevi işliyor.	-
80C3	Gereken kaynaklar (bellek, vb.) şu anda tamamen kullanılmış durumdadır.	-
80C4	Dahili geçici hata. Görevi işlenemedi. Görevi tekrarlayın. Bu hata sıkça ortaya çıkarsa, sisteminizi elektrik arızası kaynakları açısından kontrol edin.	-

Hata kodu	Açıklama	Sınırlama
80C5	Desantral G/Ç mevcut değil	Desantral G/Ç
80C6	Veri seti aktarımı, öncelik sınıfı sınıflandırması nedeniyle iptal edildi (yeniden başlatma veya arka plan).	Desantral G/Ç
8xyy[1]	Genel hata kodları	

Genel hata kodlarıyla ilgili daha fazla bilgiyi, "Genişletilmiş yordamlar, desantral çevrebirim: RDREC, WRREC ve RALRM için kullanılan hata bilgileri" (Sayfa 366) başlığında bulabilirsiniz.

9.4 PROFIenergy

PROFIenergy, PROFINET ile enerji yönetimi için üreticiden ve cihazdan bağımsız bir profildir. Üretimdeki molalarda ve planlanmayan kesintilerde elektrik tüketimini azaltmak için, cihazları PROFIenergy ile koordine edilmiş ve merkezileştirilmiş şekilde kapatmak mümkündür.

PROFINET IO kontrolörü, PROFINET cihazlarını/güç modüllerini, kullanıcı programındaki özel komutlar aracılığıyla kapatır. Ek donanıma ihtiyaç duymazsınız. PROFINET cihazları, PROFIenergy komutlarını doğrudan yorumlar.

S7-1200 CPU'su, PE kontrolörü fonksiyonunu desteklemez. S7-1200 CPU'su, sadece PROFIenergy birimi olarak (I-Device fonksiyonuyla) çalışabilir.

PROFIenergy kontrolörü (PE kontrolörü)

PE kontrolörü, alt seviyedeki cihazların boşta durumunu etkinleştiren veya devre dışı bırakan daha yüksek seviyeli bir CPU'dur (örn. bir S7-1500). PE kontrolörü, belirli üretim bileşenlerini veya tüm üretim hatlarını kullanıcı programı üzerinden devre dışı bırakır ve yeniden etkinleştirir. Alt seviyedeki cihazlar, kullanıcı programından gelen komutları, ilgili yordamlar (fonksiyon blokları) aracılığıyla alır.

Kullanıcı programı, komutları, PROFINET iletişim protokolü üzerinden gönderir. PE komutu, bir PE cihazını uyku moduna ayarlamak için kullanılan bir kumanda komutu veya bir durumu veya ölçüm değerini okumak için kullanılan bir komut olabilir.

PE_I_DEV yordamıyla bir modülün verilerini talep edebilirsiniz. Kullanıcı programı, PE kontrolöründen hangi bilgilerin talep edileceğini belirlemeli ve bu bilgileri veri setlerinin yardımıyla enerji modülünden çağırmalıdır. Modülün kendisi, PE komutlarını doğrudan desteklemez. Modül, enerji ölçümü bilgilerini, ortak olarak kullanılan bir alana kaydeder ve alt CPU (örneğin bir S7-1200) bilgileri PE kontrolörüne göndermek için PE_I_DEV yordamını tetikler.

PROFIenergy birimi (PE cihazı)

PE cihazı (örn. bir S7-1200), PE kontrolörünün (örn. bir S7-1500'ün) PROFIenergy komutlarını alır ve bu komutları (örneğin bir ölçüm değerini yayınlayarak veya bir uyku modunu etkinleştirerek) uygun şekilde yürütür. PE cihazının PROFIenergy özellikli bir cihazda uygulanması, cihaza ve üreticiye bağlıdır.

Referans bilgiler

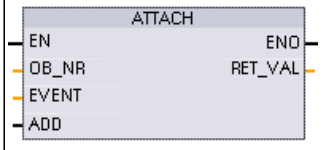
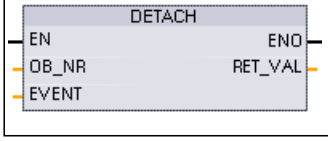
PROFIenergy ile ilgili daha fazla bilgiyi, TIA Portal'daki STEP 7 online yardımında bulabilirsiniz. PROFIenergy yordamlarının kullanıldığı örnekleri, Industry Online Support'taki "PROFIenergy - SIMATIC S7 ile enerji tasarrufu sağlama (http://support.automation.siemens.com/WW/view/en/41986454)" girişinde bulabilirsiniz.

9.5 Alarmlar

9.5.1 ATTACH ve DETACH (OB ve alarm olayını birbirlerine ata/OB ve alarm olayı atamasını kaldır) yordamları

ATTACH ve DETACH yordamlarıyla, alarm olayları aracılığıyla başlatılan alt programları etkinleştirebilir ve devre dışı bırakabilirsiniz.

Tablo 9-97 ATTACH ve DETACH yordamları

LAD/FBD	SCL	Açıklama
	**ret_val := ATTACH(ob_nr:=_int_in_, event:=_event_att_in_, add:=_bool_in_);**	ATTACH yordamı, bir proses olayında bir alarm OB'ye ait alt programın yürütülmesini etkinleştirir.
	**ret_val := DETACH(ob_nr:=_int_in_, event:=_event_att_ in);**	DETACH yordamı, bir proses olayında bir alarm OB'ye ait alt programın yürütülmesini devre dışı bırakır.

Tablo 9-98 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_ATT	Organizasyon bloğu kodu: "Yeni blok ekle" fonksiyonu aracılığıyla oluşturulan mevcut proses alarmı OB'lerinden birini seçin. Parametre alanına ve ardından mevcut OB'leri görüntülemek için kullanılan sembole çift tıklayın.
EVENT	IN	EVENT_ATT	Olay kodu: Dijital girişler veya hızlı sayaçlar için PLC cihaz konfigürasyonunda etkinleştirilmiş olan mevcut proses alarmı olaylarından birini seçin. Parametre alanına ve ardından mevcut olayları görüntülemek için kullanılan sembole çift tıklayın.
ADD (sadece ATTACH)	IN	Bool	• ADD = 0 (standart): Bu olay, bu OB için şu ana kadarki tüm olay atamalarının yerine geçer. • ADD = 1: Bu olay, bu OB için şu ana kadarki tüm olay atamalarına eklenir.
RET_VAL	OUT	Int	Yürütme koşulu

Proses alarmı olayları

Aşağıdaki proses alarmı olayları CPU tarafından desteklenir:

- Yükselen kenar olayları: CPU'nun ilk 12 entegre dijital girişi (DEa.0 ila DEb.3) ve tüm SB dijital girişleri
  - Yükselen bir kenar, dijital bir giriş, girişe bağlı bir cihazın sinyal değişimine tepki olarak KAPALI durumundan AÇIK durumuna geçerse ortaya çıkar.
- Alçalan kenar olayları: CPU'nun ilk 12 entegre dijital girişi (DEa.0 ila DEb.3) ve tüm SB dijital girişleri
  - Alçalan bir kenar, dijital giriş AÇIK durumundan KAPALI durumuna geçerse ortaya çıkar.
- Olaylar Hızlı sayacın (HSC) güncel değeri = Referans değer (CV = RV) (HSC 1 ila 6)
  - Bitişik bir değerin güncel sayma değeri, tam olarak daha önce ayarlanan referans değere karşılık gelen bir değer olarak değişirse, bir HSC için bir alarm CV = RV üretilir.
- HSC yön değişimi olayları (HSC 1 ila 6)
  - Bir yön değişimi olayı, HSC'nin artma iken azalma olarak veya azalma iken artma olarak değiştiği algılanırsa ortaya çıkar.
- Olaylar Harici HSC sıfırlama (HSC 1 ila 6)
  - Belirli HSC işletim türleri, HSC'nin sayma değerini sıfıra döndürmek üzere, harici sıfırlama işlemi için dijital bir girişin atanmasına izin verir. Bir Harici sıfırlama olayı, bu girişin KAPALI iken AÇIK olarak değişmesi durumunda bu tür bir HSC için ortaya çıkar.

Cihaz konfigürasyonunda proses alarmı olaylarının etkinleştirilmesi

Proses alarmları, cihaz konfigürasyonu sırasında etkinleştirilmelidir. Bu olayı konfigürasyon sırasında veya çalışma zamanında atamak istiyorsanız, bir dijital giriş kanalı veya bir HSC için cihaz konfigürasyonunda olayları etkinleştirmek için kullanılan kutuyu seçmeniz gerekir.

PLC cihaz konfigürasyonundaki olası opsiyon kutuları:

- Dijital giriş
  - Yükselen kenarları algılamanın etkinleştirilmesi
  - Alçalan kenarları algılamanın etkinleştirilmesi
- Hızlı sayaç (HSC)
  - Bu hızlı sayacı etkinleştirme
  - Referans değere eşit sayaç değeri için alarm üretme
  - Harici sıfırlama olayı için alarm üretme
  - Yön değişimi olayı için alarm üretme

Yeni proses alarmı OB'leri programınıza ekleme

Standart olarak olayın ilk etkinleştirilmesi sırasında olaya hiçbir OB atanmaz. Bu durum, "Proses alarmı" cihaz konfigürasyonunda "<bağlı değil>" kodu aracılığıyla işaretlenir. Bir proses alarmı olayına sadece proses alarmı OB'leri atanabilir. Mevcut tüm proses alarmı OB'leri, "Proses alarmı" açılır listesinde sıralanır. Hiçbir OB sıralanmamışsa, "Proses alarmı" tipinde bir OB'yi aşağıdaki şekilde oluşturmanız gerekir. Proje gezginindeki "Program blokları" öğesinin altında aşağıdaki yöntemi izleyin:

1. "Yeni blok ekle" öğesine çift tıklayın, "Organizasyon bloğu (OB)"yi ve ardından "Proses alarmı"nı seçin.
2. Opsiyonel olarak OB'yi yeniden adlandırabilir, programlama dilini (LAD, FBD veya SCL) belirleyebilir ve blok numarasını seçebilirsiniz (manuel düzenlemeye geçin ve önerilen blok numarasından farklı bir blok numarası seçin).
3. OB'yi düzenleyin ve programın, olay ortaya çıktığında uygulanması gereken tepkisini ekleyin. Bu OB'den, maksimum gömme derinliğine kadar FC'leri ve FB'leri çağırabilirsiniz. Güvenlik programları için maksimum gömme derinliği dört seviyedir. Maksimum gömme derinliği, diğer programlar için altı seviyedir.

OB_NR parametresi

Mevcut tüm proses alarmı OB'lerinin adları, cihaz konfigürasyonundaki "Proses alarmı:" açılır listesinde ve "ATTACH /DETACH" yordamlarının "OB_NR" parametresi için kullanılan açılır listede sıralanır.

EVENT parametresi

Proses alarmı olayı etkinleştirilirse, bu belirli olaya benzersiz bir standart olay adı atanır. Bu olay adını "Olay adı:" düzenleme alanı aracılığıyla değiştirebilirsiniz, ancak ad benzersiz olmalıdır. Olay adları, "Sabitler" değişken tablosunda değişken adlarına dönüştürülür ve "EVENT" parametresinin açılır listesindeki ATTACH ve DETACH yordam alanlarında görüntülenir. Bu değişkenin değeri, olayı tanımlayan dahili bir numaradır.

Genel çalışma şekli

Her proses alarmı, proses alarmı olayı ortaya çıkarken düzenleme için bekleme kuyruğuna alınan bir proses alarmı OB'sine atanabilir. OB ve olayın atama işlemi, konfigürasyon sırasında veya çalışma zamanında gerçekleştirilebilir.

Konfigürasyon sırasında, etkinleştirilmiş bir olaya bir OB atama veya bu atamayı kaldırma olanağından faydalanabilirsiniz. Konfigürasyon sırasında bir OB'yi bir olaya atamak için "Proses alarmı:" öğesinin altında yer alan mevcut proses alarmı OB'leri açılır listesinde bir OB'yi seçin (açılır listedeki aşağıyı gösteren oka tıklayın). Bu listeden istediğiniz OB'nin adını seçin veya atamayı kaldırmak için "<bağlı değil>" öğesini seçin.

Etkinleştirilmiş bir proses alarmı olayını çalışma zamanı sırasında da atayabilir veya bu atamayı kaldırabilirsiniz. ATTACH ve DETACH yordamlarıyla, çalışma zamanı sırasında ilgili OB'ye (gerekirse birkaç kez) etkinleştirilmiş bir proses alarmı olayını atayabilir veya bu atamayı kaldırabilirsiniz. Mevcut durumda hiçbir OB atanmamışsa (cihaz konfigürasyonunda "<bağlı değil>" seçildiği için veya DETACH yordamının yürütülmesinin sonucu olarak), etkinleştirilmiş olan proses alarmı göz ardı edilir.

DETACH'in çalışma şekli

DETACH yordamıyla, belirli bir olayın veya tüm olayların belirli bir OB'ye atamasını kaldırabilirsiniz. Bir EVENT belirtildiyse, belirtilen OB_NR organizasyon bloğundan sadece bu tek olay ayrılır. Mevcut durumda bu organizasyon bloğuna OB_NR atanmış olan diğer tüm olaylar atanmış şekilde kalır. Hiçbir EVENT belirtilmediyse, mevcut durumda bu OB_NR organizasyon bloğuna atanmış olan tüm olaylar ayrılır.

Koşul kodları

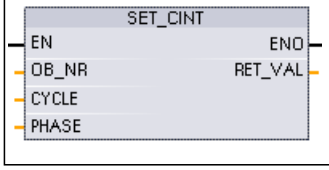
Tablo 9-99 Koşul kodları

RET_VAL (W#16#....)	ENO	Açıklama
0000	1	Hata yok
0001	1	Hiçbir atama mevcut değil (sadece DETACH)
8090	0	OB mevcut değil
8091	0	OB tipi yanlış
8093	0	Olay mevcut değil

9.5.2 Uyandırma alarmları

9.5.2.1 SET_CINT (uyandırma alarmını parametrelendir)

Tablo 9-100 SET_CINT (uyandırma alarmını parametrelendir)

LAD/FBD	SCL	Açıklama
	**ret_val := SET_CINT(ob_nr:=_int_in_, cycle:=_udint_in_, phase:=_udint_in_);**	Program çevrimini durduran periyodik yürütmeyi başlatmak için belirtilen alarm OB ayarlanır.

Tablo 9-101 Parametreler için kullanılan veri tipleri

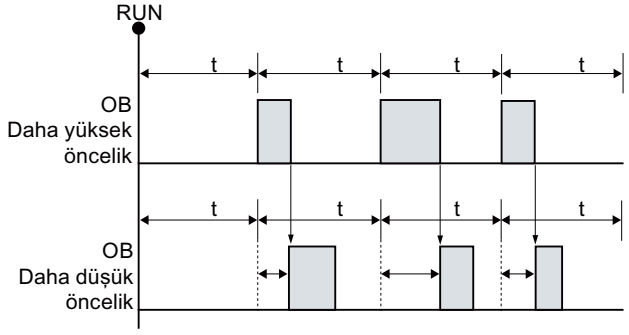
Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_CYCLIC	OB numarası (sembolik olarak adreslenebilir)
CYCLE	IN	UDInt	Milisaniye cinsinden zaman aralığı
PHASE	IN	UDInt	Mikrosaniye cinsinden faz kayması
RET_VAL	OUT	Int	Yürütme koşulu

Örnekler: Zaman parametreleri

- CYCLE süresi = 100 us ise, OB_NR tarafından belirtilen alarm OB, program çevrimini her 100 us'de bir durdurur. Alarm OB yürütülür ve ardından yürütme kumandasını, durdurma işleminin gerçekleştiği noktada tekrar program çevrimine aktarır.
- CYCLE süresi = 0 ise, alarm olayı devre dışı bırakılır ve alarm OB yürütülmez.
- PHASE süresi (faz kayması) CYCLE zaman aralığı başlamadan önce gerçekleşen, belirtilen bir gecikme süresidir. Faz kayması ile düşük önceliğe sahip OB'lerin yürütülmesine zamansal olarak kumanda edebilirsiniz.

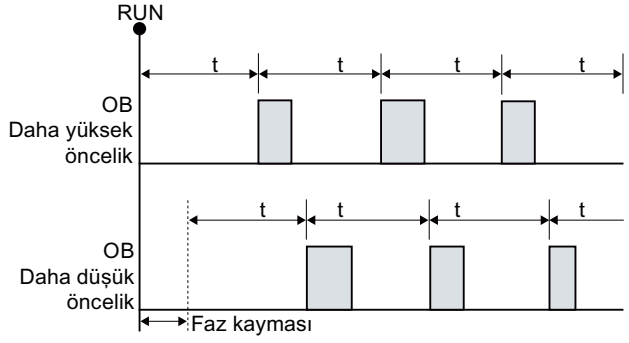
Daha düşük ve daha yüksek öncelikli OB'ler aynı zaman aralığında çağrılırsa, daha düşük öncelikli OB ancak daha yüksek öncelikli OB'nin işlenmesi tamamlandığı takdirde çağrılır. Daha düşük öncelikli OB'nin yürütülmesinin başlangıcı, daha yüksek öncelikli OB'nin işleme süresine bağlı olarak kayabilir.

Faz kayması olmayan OB çağrısı



Daha düşük öncelikli bir OB'nin yürütülmesini sabit çevrim süresiyle başlatmak istiyorsanız, faz kayması değeri, daha yüksek öncelikli OB'nin işleme süresinden daha büyük olmalıdır.

Faz kayması olan OB çağrısı

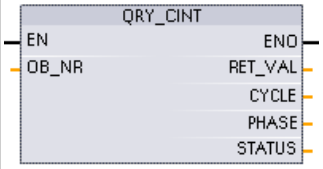


Tablo 9-102 Koşul kodları

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	OB mevcut değil veya yanlış tipte
8091	Geçersiz çevrim süresi
8092	Geçersiz faz kayması süresi
80B2	OB, ilgili bir olaya sahip değil

9.5.2.2 QRY_CINT (uyandırma alarmı parametrelerini sorgula)

Tablo 9-103 **QRY_CINT (uyandırma alarmını sorgula)**

LAD/FBD	SCL	Açıklama
	**ret_val := QRY_CINT(** **ob_nr:=_int_in_,** **cycle=>_udint_out_,** **phase=>_udint_out__,** **status=>_word_out_);**	Bir uyandırma alarmı OB'sinin parametreleri ve yürütme durumu sorgulanır. Döndürülen değerler, QRY_CINT yürütülürken mevcuttu.

Tablo 9-104 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_CYCLIC	OB numarası (sembolik olarak adreslenebilir, örn. OB_MyOBName)
RET_VAL	OUT	Int	Yürütme koşulu
CYCLE	OUT	UDInt	Milisaniye cinsinden zaman aralığı
PHASE	OUT	UDInt	Mikrosaniye cinsinden faz kayması
STATUS	OUT	Word	Uyandırma alarmının durum kodu: • 0 ile 4 arasındaki Bit'ler, bkz. aşağıdaki STATUS tablosu • Diğer Bit'ler, her zaman 0

Tablo 9-105 STATUS parametresi

Bit	Değer	Açıklama
0	0	CPU RUN işletim durumundayken
	1	Yol alma sırasında
1	0	Alarm etkinleştirildi.
	1	Alarm, DIS_IRT yordamı aracılığıyla devre dışı bırakılır.
2	0	Alarm aktif değil veya alarmın süresi doldu.
	1	Alarm aktiftir.
4	0	OB_NR tarafından belirtilen OB mevcut değil.
	1	OB_NR tarafından belirtilen OB mevcut.
Diğer Bit'ler		Her zaman 0

Bir hata oluşursa, RET_VAL, ilgili hata kodunu ve STATUS = 0 parametresini görüntüler.

Tablo 9-106 RET_VAL parametresi

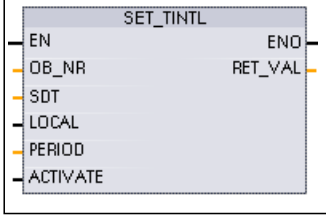
RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	OB mevcut değil veya yanlış tipte.
80B2	OB, ilgili bir olaya sahip değil.

9.5.3 Saat alarmları

> **⚠ İKAZ**
>
> **NTP senkronizasyonu (Network Time Protocol) nedeniyle yetkisiz kişilerin ağa erişimi tehlikesi**
>
> Bir saldırgan, NTP senkronizasyonu (Network Time Protocol) üzerinden ağlarınıza erişebilirse, CPU sistem saatini değiştirerek prosesinizin üzerindeki kontrolü elinizden alabilir. Bu tür bir proses kontrolü aktarımı nedeniyle ölüm, ağır yaralanma durumları veya maddi hasarlar oluşabilir.
>
> S7-1200 CPU'sunun NTP istemcisi fonksiyonu standart olarak devre dışı bırakılmıştır ve etkinleştirilmiş durumdayken sadece yapılandırılmış IP adreslerinin NTP sunucusu olarak çalışmasına olanak sağlar. CPU, bu fonksiyonu standart olarak devre dışı bırakır ve CPU sistem saatinde uzaktan kontrollü düzeltmelerin yapılmasına izin vermek için bu fonksiyonu yapılandırmanız gerekir.
>
> S7-1200 CPU'su, doğru bir CPU sistem saatine bağlı olan saat alarmlarını ve saat işlemlerini destekler. NTP'yi yapılandırır ve saat senkronizasyonunun bir sunucudan gerçekleştirilmesini kabul ederseniz, sunucuda güvenilir bir kaynağın söz konusu olduğundan emin olmanız gerekir. Aksi takdirde, bilinmeyen bir kullanıcının CPU sistem saatini değiştirebileceği ve bu sayede belirli bir çerçevede prosesinizi kontrol edebileceği bir güvenlik ihlali ortaya çıkabilir.
>
> Güvenlikle ilgili bilgileri ve önerileri, Siemens'in "Service & Support" web sitesindeki Operational Guidelines for Industrial Security (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf) bölümümüzde bulabilirsiniz.

9.5.3.1 SET_TINTL (saat alarmını belirle)

Tablo 9-107 SET_TINTL (DTL veri tipindeki saat alarmı için tarih ve saati belirle)

LAD/FBD	SCL	Açıklama
	**ret_val := SET_TINTL(** **OB_NR:=_int_in_,** **SDT:=_dtl_in_,** **LOCAL:=_bool_in_** **PERIOD:=_word_in_** **ACTIVATE:=_bool_in_);**	Bir saat alarmının belirlenmesi. OB, bir yürütme işlemi veya atanmış bir süre içinde tekrarlanan yürütme işlemleri için belirlenebilir.

Tablo 9-108 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)	OB numarası (sembolik olarak adreslenebilir)
SDT	IN	DTL	Başlama tarihi ve saati: Saniye ve milisaniye değerleri göz ardı edilir ve 0'a ayarlanabilir.
LOCAL	IN	Bool	0 = Sistem saatini kullan 1 = Yerel saati kullan (CPU için yerel saat ayarlandıysa, aksi takdirde sistem saatini kullanın)

Parametre ve veri tipi		Veri tipi	Açıklama
PERIOD	IN	Word	Birkaç kez ortaya çıkan alarm olayı için başlama tarihinden ve saatinden itibaren geçen süre. • W#16#0000 = bir kez • W#16#0201 = her dakika • W#16#0401 = her saat • W#16#1001 = her gün • W#16#1201 = her hafta • W#16#1401 = her ay • W#16#1801 = her yıl • W#16#2001 = Ay sonu
ACTIVATE	IN	Bool	Alarm olayını etkinleştirmek için 0 = ACT_TINT yürütülmelidir. 1 = Alarm olayı etkinleştirildi.
RET_VAL	OUT	Int	Yürütme koşulu

Programınız SET_TINTL yordamıyla, atanan alarm OB'sini yürüten bir tarih ve saat alarm olayı belirleyebilir. Başlama tarihi ve başlama saati, SDT parametresi tarafından önceden belirtilir ve tekrarlanan alarmlar için kullanılan süre (örneğin günlük veya haftalık), PERIOD parametresi tarafından belirlenir. Aylık olarak tekrarlanan bir saat ayarlarsanız, başlama tarihi için 1 ile 28 arasındaki bir günü belirtmeniz gerekir. Şubat ayında olmadıkları için 29 ile 31 arasındaki günler kullanılamaz. Bir alarm olayını ay sonunda gerçekleşecek şekilde programlamak istiyorsanız, PERIOD parametresi için ay sonu ayarını kullanın.

SDT parametresindeki DTL verilerinin haftanın günü değeri göz ardı edilir. Bir CPU'nun güncel tarihini ve güncel saatini, bir çevrimiçi CPU'nun "Çevrimiçi ve teşhis" görünümündeki "Saati ayarla" fonksiyonu aracılığıyla ayarlayabilirsiniz. Günü, ayı ve yılı ayarlamanız gerekir. STEP 7, CPU'nun tarih ve saatini temel alarak alarm zaman dilimini hesaplar.

NOT

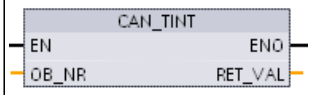
Yaz saatinden kış saatine geçilirken günün ilk saati mevcut değildir. Bu nedenle ikinci saatteki bir başlama zamanını kullanın veya ilk saatin içinde ek bir gecikme alarmı kullanın.

Tablo 9-109 Koşul kodu

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz parametre OB_NR
8091	SDT başlama zamanında geçersiz parametre: (örneğin yaz saatinin başlangıcında atlanan saatteki bir başlama zamanı)
8092	Geçersiz PERIOD parametresi
80A1	Başlama zamanı geçmişte. (Bu hata kodu sadece PERIOD = W #16#0000 iken ortaya çıkar.)

9.5.3.2 CAN_TINT (saat alarmını sil)

Tablo 9-110 CAN_TINT (saat alarmını sil)

LAD/FBD	SCL	Açıklama
	**ret_val:=CAN_TINT(_int_in);**	Belirtilen alarm OB için kullanılan saat alarmı olayını siler.

Tablo 9-111 Parametreler için kullanılan veri tipleri

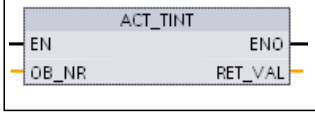
Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)	OB numarası (sembolik olarak adreslenebilir)
RET_VAL	OUT	Int	Yürütme koşulu

Tablo 9-112 Koşul kodları

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz parametre OB_NR
80A0	Bu alarm OB için hiçbir başlama tarihi/saati ayarlanmadı

9.5.3.3 ACT_TINT (saat alarmını etkinleştir)

Tablo 9-113 ACT_TINT (saat alarmını etkinleştir)

LAD/FBD	SCL	Açıklama
	**ret_val:=ACT_TINT(_int_in_);**	Belirtilen alarm OB için kullanılan saat alarmı olayını etkinleştirir.

Tablo 9-114 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)	OB numarası (sembolik olarak adreslenebilir)
RET_VAL	OUT	Int	Yürütme koşulu

Tablo 9-115 Koşul kodları

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz parametre OB_NR
80A0	Saat alarmı OB'si için başlama tarihi ve saati ayarlanmadı
80A1	Etkinleştirilen zaman geçmişte. Bu hata, ancak alarm OB'sinin sadece tek seferlik yürütme için ayarlanmış olması durumunda ortaya çıkar.

9.5.3.4 QRY_TINT (saat alarmının durumunu sorgula)

Tablo 9-116 QRY_TINT (saat alarmını sorgula)

LAD/FBD	SCL	Açıklama
	**ret_val:=QRY_TINT(OB_NR:=_int_in_, STATUS=>_word_out_);**	Belirtilen alarm OB için saat alarmı durumunu sorgular.

Tablo 9-117 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)	Sorgu için kullanılan alarm OB'sinin OB numarası (sembolik olarak adreslenebilir)
RET_VAL	OUT	Int	Yürütme koşulu
STATUS	OUT	Word	Belirtilen alarm OB'sinin durumu

Tablo 9-118 STATUS parametresi

Bit	Değer	Açıklama
0	0	RUN işletim durumunda
	1	Yol alma sırasında
1	0	Alarm etkinleştirildi.
	1	Alarm devre dışı bırakıldı.
2	0	Alarm aktif değil veya alarmın süresi doldu.
	1	Alarm aktiftir.
4	0	Atanan OB_NR mevcut değil.
	1	Atanan OB_NR'ye sahip bir OB mevcut.
6	1	Saat alarmı yerel saati kullanıyor.
	0	Saat alarmı sistem saatini kullanıyor.
Diğer		Her zaman 0

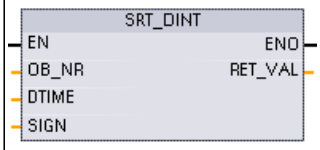
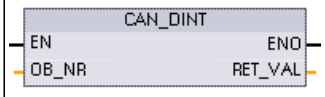
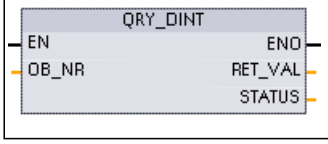
Tablo 9-119 Koşul kodu

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz parametre OB_NR

9.5.4 Gecikme alarmları

Gecikme alarmının işlenmesini SRT_DINT ve CAN_DINT yordamlarıyla başlatabilir ve iptal edebilir ve alarm durumunu QRY_DINT yordamıyla sorgulayabilirsiniz. Her gecikme alarmı, önceden belirtilen bir gecikme süresi dolduktan sonra ortaya çıkan tek seferlik bir olaydır. Zaman gecikmesi olayı, zaman gecikme süresi dolmadan silinirse, alarm programda ortaya çıkmaz.

Tablo 9-120 SRT_DINT, CAN_DINT ve QRY_DINT yordamları

LAD/FBD	SCL	Açıklama
	**ret_val := SRT_DINT(ob_nr:=_int_in_, dtime:=_time_in_, sign:=_word_in_);**	SRT_DINT, DTIME parametresinde belirlenen gecikme süresi dolduğu takdirde, bir OB'yi yürüten bir gecikme alarmını başlatır.
	**ret_val := CAN_DINT(ob_nr:=_int_in_);**	CAN_DINT, başlamış olan bir gecikme alarmını siler. Bu durumda gecikme alarmı OB'si yürütülmez.
	**ret_val := QRY_DINT(ob_nr:=_int_in_, status=>_word_out_);**	QRY_DINT, OB_NR parametresi tarafından belirtilen gecikme alarmının durumunu sorgular.

Tablo 9-121 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
OB_NR	IN	OB_DELAY	Bir zaman gecikmesinden sonra başlatılması gereken organizasyon bloğu (OB): Proje gezgininde "Yeni blok ekle" fonksiyonu aracılığıyla oluşturulan mevcut gecikme alarmı OB'lerinden birini seçin. Parametre alanına ve ardından mevcut OB'leri görüntülemek için kullanılan sembole çift tıklayın.
DTIME [1]	IN	Time	Gecikme değeri (1 ila 60000 msn)
SIGN [1]	IN	Word	S7-1200 tarafından kullanılmaz: Her değer kabul edilir. Hataları önlemek için bir değer atanmalıdır.
RET_VAL	OUT	Int	Yürütme koşulu
STATUS	OUT	Word	QRY_DINT yordamı: Belirtilen gecikme alarmı OB'sinin durumu, bkz. alttaki tablo

[1] Sadece SRT_DINT'te

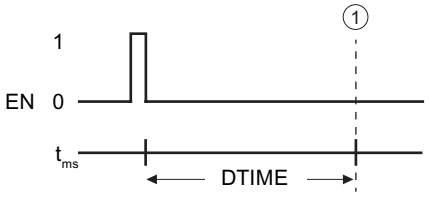
Kullanım

EN=1 ise, SRT_DINT yordamı, dahili zaman gecikmesini (DTIME) başlatır. Zaman gecikme süresi dolarsa, CPU, atanmış gecikme alarmı OB'sinin yürütülmesini tetikleyen bir program durdurması oluşturur. CAN_DINT yordamıyla, başlatılmış bir gecikme alarmını, belirtilen gecikme süresi oluşmadan önce iptal edebilirsiniz. Toplamda maksimum dört gecikme alarmı olayı aktif olabilir.

NOT

SRT_DINT yordamı, EN = 1 ise her çevrimde zaman gecikmesi için belirlenen süreyi başlatır. Zaman gecikmesini başlatmak için EN = 1 olarak ayarlamak yerine EN = 1'i bir kez onaylayın.

SRT_DINT yordamının zaman diyagramı:



① Gecikme alarmı uygulanır

Gecikme alarmı OB'lerinin projenize eklenmesi

Gecikme alarmı OB'leri sadece SRT_DINT ve CAN_DINT yordamlarında ekleyebilirsiniz. Yeni bir projede henüz hiçbir gecikme alarmı OB'si mevcut değildir. Gecikme alarmı OB'leri projenize eklemeniz gerekir. Bir gecikme alarmı OB'si oluşturmak için aşağıdaki yöntemi izleyin:

1. Proje gezgininde "Program blokları" altındaki "Yeni blok ekle" öğesine çift tıklayın, "Organizasyon bloğu (OB)" ve ardından "Gecikme alarmı" öğesini seçin.
2. OB'yi yeniden adlandırabilir, programlama dilini belirleyebilir ve blok numarasını seçebilirsiniz. Otomatik olarak atanan blok numarasından farklı bir blok numarası seçmek istiyorsanız manuel düzenlemeye geçin.
3. Gecikme alarmı OB'sinin alt programını düzenleyin ve gecikme alarmı olayı ortaya çıktığında uygulanması gereken program tepkisini ekleyin. Gecikme alarmı OB'sinden başka FC ve FB kod bloklarını çağırabilirsiniz. Güvenlik programları için maksimum gömme derinliği dört seviyedir. Maksimum gömme derinliği, diğer programlar için altı seviyedir.
4. SRT_DINT ve CAN_DINT yordamlarının OB_NR parametresini düzenlerseniz, yeni atanan gecikme alarmı OB'lerinin adları kullanıma sunulur.

QRY_DINT'in STATUS parametresi

Tablo 9-122 Bir hata (REL_VAL <> 0) mevcutsa, STATUS = 0 olur.

Bit	Değer	Açıklama
0	0	RUN işletim durumunda
	1	Yol alma sırasında
1	0	Alarm etkinleştirildi.
	1	Alarm devre dışı bırakıldı.
2	0	Alarm aktif değil veya alarmın süresi doldu.
	1	Alarm aktiftir.
4	0	OB_NR'de belirtilen OB numarasına sahip bir OB mevcut değil.
	1	OB_NR'de belirtilen OB numarasına sahip bir OB mevcut.
Diğer Bit'ler		Her zaman 0

Koşul kodları

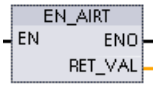
Tablo 9-123 SRT_DINT, CAN_DINT ve QRY_DINT için koşul kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	Yanlış OB_NR parametresi
8091	Yanlış DTIME parametresi
80A0	Gecikme alarmı henüz başlatılmadı.

9.5.5 DIS_AIRT ve EN_AIRT yordamları (daha yüksek öncelikli alarmların ve asenkron hata olaylarının yürütülmesini geciktir/etkinleştir)

Alarm işleme, DIS_AIRT ve EN_AIRT yordamlarıyla etkinleştirilebilir ve devre dışı bırakılabilir.

Tablo 9-124 DIS_AIRT ve EN_AIRT yordamları

LAD/FBD	SCL	Açıklama
	**DIS_AIRT();**	DIS_AIRT, yeni alarm olaylarının işlenmesini geciktirir. Bir OB'de DIS_AIRT'i birden çok kez yürütebilirsiniz.
	**EN_AIRT();**	EN_AIRT, daha önce DIS_AIRT yordamıyla devre dışı bırakılan alarm olaylarını işleme sürecini etkinleştirir. DIS_AIRT yordamının her yürütülme işlemi, EN_AIRT yordamı yürütülerek silinmelidir. EN_AIRT yordamları, ilgili OB için alarmlar yeniden etkinleştirilmeden önce aynı OB'de veya aynı OB'den çağrılan bir FC veya FB'de yürütülmelidir.

Tablo 9-125 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
RET_VAL	OUT	Int	Gecikme sayısı = Bekleme kuyruğundaki yürütülen DIS_AIRT yordamı sayısı.

DIS_AIRT yordamını yürütme sayısı, işletim sistemi tarafından sayılır. Bu yordamların her biri, bu yordamlar bir EN_AIRT yordamıyla silinene veya güncel OB tamamen işlenene kadar geçerli kalır. Örnek: Alarm işlemenin tekrar etkinleştirilmesi için, alarm işlemeyi, DIS_AIRT yordamının beş kez yürütülmesiyle devre dışı bıraktıysanız, bu durumu, EN_AIRT komutunu beş kez yürüterek iptal etmeniz gerekir.

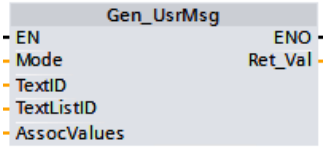
Alarm olaylarının yeniden etkinleştirilmesinden sonra, DIS_AIRT yordamı yürütülürken ortaya çıkan alarmlar işlenir veya güncel OB yürütüldükten sonra alarmlar işlenir.

RET_VAL parametresi, alarm işlemenin ne sıklıkla devre dışı bırakıldığını, kaç adet DIS_AIRT yordamı yürütmesinin bekleme kuyruğuna alındığını belirtir. Alarm işleme, ancak RET_VAL parametresi = 0 olduğunda tekrar etkinleştirilir.

9.6 Alarmlar

9.6.1 Gen_UsrMsg (kullanıcı teşhis mesajları oluştur)

Tablo 9-126 Gen_UsrMsg yordamı

LAD/FBD	SCL	Açıklama
	**ret_val :=Gen_UsrMsg(** **Mode:=_uint_in_,** **TextID:=_uint_in_,** **TextListID:=_uint_in_,** **AssocValues:=_struct_inout_);**	"Gen_UsrMsg" yordamıyla, gelen veya giden bir alarmın söz konusu olabileceği bir kullanıcı teşhis alarmı oluşturabilirsiniz. Kullanıcı teşhis alarmlarının yardımıyla bir kullanıcı girişini teşhis tamponuna yazabilir ve uygun bir alarmın gönderilmesini sağlayabilirsiniz. Teşhis tamponundaki giriş senkron şekilde oluşturulur. Alarm aktarımı asenkron olarak gerçekleşir. Yordam yürütülürken bir hata oluşursa, bu hata, RET_VAL parametresi üzerinden yayınlanır.

Alarmın içeriği

Bir metin listesi, alarmın içeriğini tanımlar:

- Kullanılması gereken metin listesini TextListID parametresiyle tanımlayabilirsiniz. Bunun için proje gezgininde "Metin listeleri" iletişim kutusunu açın. "Metin listeleri" iletişim kutusunda "ID" sütununu görüntüleyin. TextListID parametresindeki ID'yi uygulayın.
- TextID parametresi aracılığıyla metin listesinde bulunan ve teşhis tamponuna yazmak istediğiniz girişi seçin. Bunun için, "Metin listesi girişleri" iletişim kutusundaki TextID parametresinde, "Aralık başlangıcı/Aralık bitişi" sütunlarındaki bir numarayı kullanarak bir girişi seçin. Metin listesi girişlerinin "Aralık başlangıcı" ve "Aralık bitişi" sütunlarının her ikisi için de aynı numarayı kullanmanız gerekir.

Metin listeleriyle ilgili daha fazla bilgiyi, STEP 7 bilgi sisteminde bulabilirsiniz.

Eşlik eden değerlerin tanımlanması

Metin listesi girişi, alarma eklenmesi gereken yeni eşlik eden değerleri tanımlar:

- Eşlik eden değerleri tanımlamak için metin listesi girişine aşağıdaki bilgileri ekleyin: @<Eşlik eden değerin numarası><Eleman tipi><Format bilgisi>@
- AssocValues sistem veri tipi aracılığıyla, alarm oluşturulurken hangi eşlik eden değerin eklenmesi gerektiğini belirtin.

Eşlik eden değerlerin yapısıyla ilgili daha fazla bilgiyi, STEP 7 bilgi sisteminde bulabilirsiniz.

Parametre

Aşağıdaki tabloda, "Gen_UsrMsg" yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
Mode	Input	UInt	E, A, M, D, L veya sabit	Alarmın durumunu seçmek için kullanılan parametreler: • 1: gelen alarm • 2: giden alarm
TextID	Input	UInt	E, A, M, D, L veya sabit	Alarm metni için kullanılması gereken metin listesi girişinin ID'si.
TextListID	Input	UInt	E, A, M, D, L veya sabit	Metin listesi girişini içeren metin listesinin ID'si.
Ret_Val	Return	Int	E, A, M, D, L	Yordamın hata kodu.
AssocValues	InOut	VARIANT	D, L	İşaretçiyi, eşlik eden değerlerin tanımlanması için kullanabileceğiniz AssocValues sistem veri tipinin üzerine getirin.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

AssocValues parametresi

AssocValues sistem veri tipiyle, hangi eşlik eden değerlerin gönderileceğini tanımlayabilirsiniz. Maksimum sekiz eşlik eden değer kullanılabilir. Yapıyı oluşturmak için "AssocValues" veri tipini veri bloğu olarak girin.
Value[x] parametreleri için eşlik eden değerlerin numaralarını girerek eşlik değerlerini seçebilirsiniz. Lütfen aşağıdaki uyarıya dikkat edin:

- Gen_UsrMsg yordamı, TextID ve TextListID için kullanılan değerleri, gönderilmesi gereken eşlik eden değerler olarak ele alır. Bu nedenle "1" ve "2" sayıları, önceden eşlik eden değerlerin adreslenmesi için atanmıştır. "1" ve "2" sayılarını, eşlik eden değerlerin adreslenmesi için kullanmayın.
- Eşlik eden değeri Value [1] parametresindeki numara "3" olarak, Value [2] parametresinde numara "4" olarak, vb. adresleyin.

Bayt	Parametre	Veri tipi	Başlangıç değeri	Açıklama	Eşlik eden değerin numarası
0..1	Value[1]	UINT	0	Alarmın ilk eşlik eden değeri	3
2..3	Value[2]	UINT	0	Alarmın ikinci eşlik eden değeri	4
4..5	Value[3]	UINT	0	...	5
6..7	Value[4]	UINT	0	...	6
8..9	Value[5]	UINT	0	...	7
10..11	Value[6]	UINT	0	...	8
12..13	Value[7]	UINT	0	...	9
14..15	Value[8]	UINT	0	Alarmın sekizinci eşlik eden değeri	10

RET_VAL parametresi

Aşağıdaki tabloda, RET_VAL parametresi için kullanılan çıkış değerleri tanımlanmaktadır. Ayrıca bkz. Gelişmiş yordamlar için ortak hata kodları (Sayfa 501)

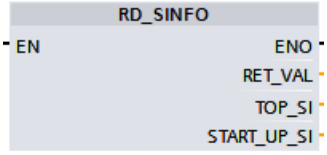
Hata kodu* (W#16#...)	Anlamı
0000	Hata yok
8080	MODE parametresindeki değer desteklenmiyor.
80C1	Çok fazla paralel çağrı nedeniyle kaynaklarda darboğaz.
8528	Parametre 5 (AssocValues), Bayt sınırlarına karşılık gelmiyor.
853A	Parametre 5 (AssocValues) geçersiz bir noktayı referanslıyor.
*Hata kodları, program editöründe tamsayılar veya onaltılık değerler olarak görüntülenebilir.	

9.7 Teşhis (PROFINET veya PROFIBUS)

9.7.1 RD_SINFO (güncel OB'nin başlangıç bilgisini oku)

Açıklama

Tablo 9-127 RD_SINFO yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := RD_SINFO(TOP_SI=>_variant_out_, START_UP_SI=>_variant_out_);**	"RD_SINFO" yordamıyla, şu OB'lerin başlangıç bilgisini okuyabilirsiniz: • Henüz tam olarak yürütülmemiş olan son çağrılan OB • CPU'nun başlattığı son yol alma OB'si İki durumda da zaman notu yoktur. Çağrı OB 100, OB 101 veya OB 102'de gerçekleşirse, iki aynı başlangıç bilgisi mesajı yayınlanır.

Parametre

Aşağıdaki tabloda, "RD_SINFO" yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
RET_VAL	Return	INT	E, A, M, D, L	Hata bilgileri
TOP_SI	Output	VARIANT	D, L	Güncel OB'nin başlangıç bilgisi
START_UP_SI	Output	VARIANT	D, L	Yol alma OB'sinin başlangıç bilgisi en son başlatıldı

Geçerli veri tipleriyle ilgili ayrıntılı bilgiler, "Veri tipleri (Sayfa 99)" bölümünde mevcuttur.

TOP_SI parametresinin SDT'leri

Aşağıdaki tabloda, TOP_SI parametresi için mümkün olan SDT'ler görüntülenmektedir:

Organizasyon blokları (OB)	Sistem veri tipleri (SDT)	Sistem veri tiplerinin numaraları
Herhangi	SI_classic*	592*
	SI_none	593
ProgramCycleOB	ProgramCycleOB	594
TimeOfDay	TimeOfDay	595
TimeDelayOB	SI_Delay	596
CyclicOB	SI_Cyclic	597
ProcessEventOB	SI_HWInterrupt	598
ProfileEventOB StatusEventOB UpdateEventOB	SI_Submodule	601

*SDT SI_classic, S7-1200 için geçerli değildir. TOP_SI parametresi SI_classic tipindeyse, S7-1200 CPU'su von #16#8081 değerinde bir RET_VAL yayınlar.

Organizasyon blokları (OB)	Sistem veri tipleri (SDT)	Sistem veri tiplerinin numaraları
SynchronousCycleOB	SI_SynchCycle	602
IOredundancyErrorOB	SI_IORedundancyError	604
CPUredundancyErrorOB	SI_CPURedundancyError	605
TimeErrorOB	SI_TimeError	606
DiagnosticErrorOB	SI_DiagnosticInterrupt	607
PullPlugEventOB	SI_PlugPullModule	608
PeripheralAccessErrorOB	SI_AccessError	609
RackStationFailureOB	SI_StationFailure	610
ServoOB	SI_Servo	611
IpoOB	SI_Ipo	612
StartupOB	SI_Startup	613
ProgrammingErrorOB IOaccessErrorOB	SI_ProgIOAccessError	614
CPURedundancyErrorOB	SI_CPURedundancyError_V2	617

*SDT SI_classic, S7-1200 için geçerli değildir. TOP_SI parametresi SI_classic tipindeyse, S7-1200 CPU'su von #16#8081 değerinde bir RET_VAL yayınlar.

START_UP_SI parametresinin SDT'leri

Aşağıdaki tabloda, START_UP_SI parametresi için mümkün olan SDT'ler görüntülenmektedir:

Sistem veri tipleri (SDT)	Sistem veri tiplerinin numaraları
SI_classic*	592
SI_none	593
SI_Startup	613

*SDT SI_classic, S7-1200 için geçerli değildir. START_UP_SI parametresi SI_classic tipindeyse, S7-1200 CPU'su von #16#8083 değerinde bir RET_VAL yayınlar.

Yapılar

Aşağıdaki tablolarda, münferit yapıların yapı elemanları tanımlanmaktadır.

Tablo 9-128 SI_classic yapısı

Yapı elemanı	Veri tipi	Açıklama
EV_CLASS	BYTE	• 0 ile 3 arasındaki Bit'ler: Olay ID'si • 4 ile 7 arasındaki Bit'ler: Olay sınıfı
EV_NUM	BYTE	Olay numarası
PRIORITY	BYTE	Öncelik sınıfının numarası (B#16#FE'nin anlamı: OB mevcut değil veya devre dışı bırakıldı veya güncel işletim durumunda başlatılamıyor)
NUM	BYTE	OB numarası
TYP2_3	BYTE	Veri ID'si 2_3: ZI2_3'e girilen bilgileri tanımlar
TYP1	BYTE	Veri ID'si 1: ZI1'e girilen bilgileri tanımlar
ZI1	WORD	Daha fazla bilgi 1
ZI2_3	DWORD	Daha fazla bilgi 2_3

Tablo 9-129 Struktur SI_none

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)

Tablo 9-130 SI_ProgramCycle yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 1	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
Initial_Call	BOOL	OB_Class = 1, 30, 52, 61, 65 için
Remanence	BOOL	OB_Class = 1 için

Tablo 9-131 SI_TimeOfDay yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 10	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
CaughtUp	BOOL	OB_Class = 10 için
SecondTime	BOOL	OB_Class = 10 için

Tablo 9-132 Struktur SI_Delay

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 20	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
Sign	WORD	OB_Class = 20 için

Tablo 9-133 Struktur SI_Cyclic

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 30	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
Initial_Call	BOOL	OB_Class = 1, 30, 52, 61, 65 için
Event_Count	INT	OB_Class = 30, 51, 52, 61, 65, 91, 92 için

Tablo 9-134 SI_HWInterrupt yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 40	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LADDR	HW_IO	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
USI	WORD	OB_Class = 40 için
IChannel	USINT	OB_Class = 40 için
EventType	BYTE	OB_Class = 40 için

Tablo 9-135 SI_Submodule yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LADDR	HW_IO	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
Slot	UINT	OB_Class = 55, 56, 57 için
Specifier	WORD	OB_Class = 55, 56, 57 için

Tablo 9-136 Struktur SI_SynchCycle

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 61	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı

Yapı elemanı	Veri tipi	Açıklama
OB_Nr	UINT	OB numarası (1 ... 32767)
Initial_Call	BOOL	OB_Class = 1, 30, 52, 61, 65 için
PIP_Input	BOOL	OB_Class = 61, 91, 92 için
PIP_Output	BOOL	OB_Class = 61, 91, 92 için
IO_System	USINT	OB_Class = 61, 91, 92 için
Event_Count	INT	OB_Class = 30, 51, 52, 61, 65, 91, 92 için
SyncCycleTime	LTIME	Hesaplanan çevrim süresi

Tablo 9-137 SI_IORedundancyError yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 70	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LADDR	HW_ANY	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
Event_Class	BYTE	OB_Class = 70, 83, 85, 86 için
Fault_ID	BYTE	OB_Class = 70, 80, 83, 85, 86 için

Tablo 9-138 SI_CPURedundancyError yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 72	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
Switch_Over	BOOL	OB_Class = 72 için

Tablo 9-139 SI_TimeError yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 80	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
Fault_ID	BYTE	OB_Class = 70, 80, 83, 85, 86 için
Csg_OBnr	OB_ANY	OB_Class = 80 için
Csg_Prio	UINT	OB_Class = 80 için

Tablo 9-140 SI_DiagnosticInterrupt yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 82	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)

Yapı elemanı	Veri tipi	Açıklama
IO_State	WORD	OB_Class = 82 için
LADDR	HW_ANY	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
Channel	UINT	OB_Class = 82 için
MultiError	BOOL	OB_Class = 82 için

Tablo 9-141 SI_PlugPullModule yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 83	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LADDR	HW_IO	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
Event_Class	BYTE	OB_Class = 70, 83, 85, 86 için
Fault_ID	BYTE	OB_Class = 70, 80, 83, 85, 86 için

Tablo 9-142 SI_AccessError yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 85	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LADDR	HW_IO	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
Event_Class	BYTE	OB_Class = 70, 83, 85, 86 için
Fault_ID	BYTE	OB_Class = 70, 80, 83, 85, 86 için
IO_Addr	UINT	OB_Class = 85 için
IO_LEN	UINT	OB_Class = 85 için

Tablo 9-143 SI_StationFailure yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 86	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LADDR	HW_IO	OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92 için
Event_Class	BYTE	OB_Class = 70, 83, 85, 86 için
Fault_ID	BYTE	OB_Class = 70, 80, 83, 85, 86 için

Tablo 9-144 SI_Servo yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 91	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı

Yapı elemanı	Veri tipi	Açıklama
OB_Nr	UINT	OB numarası (1 ... 32767)
Initial_Call	BOOL	OB_Class = 1, 30, 52, 61, 65 için
PIP_Input	BOOL	OB_Class = 61, 91, 92 için
PIP_Output	BOOL	OB_Class = 61, 91, 92 için
IO_System	USINT	OB_Class = 61, 91, 92 için
Event_Count	INT	OB_Class = 30, 51, 52, 61, 65, 91, 92 için
Synchronous	BOOL	

Tablo 9-145 SI_Ipo SI_Ipo

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 92	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
Initial_Call	BOOL	OB_Class = 1, 30, 52, 61, 65 için
PIP_Input	BOOL	OB_Class = 61, 91, 92 için
PIP_Output	BOOL	OB_Class = 61, 91, 92 için
IO_System	USINT	OB_Class = 61, 91, 92 için
Event_Count	INT	OB_Class = 30, 51, 52, 61, 65, 91, 92 için
Reduction	UINT	OB_Class = 92 için

Tablo 9-146 SI_Startup yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT := 100	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
LostRetentive	BOOL	OB_Class = 100 için
LostRTC	BOOL	OB_Class = 100 için

Tablo 9-147 SI_ProgIOAcessError yapısı

Yapı elemanı	Veri tipi	Açıklama
SI_Format	USINT	• 16#FF = Bilgi yok • 16#FE = Optimize edilmiş başlangıç bilgisi
OB_Class	USINT	"Bilgi yok" veya "Optimize edilmiş başlangıç bilgisi" için kullanılan OB sınıfı
OB_Nr	UINT	OB numarası (1 ... 32767)
BlockNr	UINT	OB_Class = 121, 122 için
Reaction	USINT	OB_Class = 121, 122 için
Fault_ID	BYTE	OB_Class = 121, 122 için
BlockType	USINT	OB_Class = 121, 122 için
Area	USINT	OB_Class = 121, 122 için

Yapı elemanı	Veri tipi	Açıklama
DBNr	DB_ANY	OB_Class = 121, 122 için
Csg_OBNr	OB_ANY	OB_Class = 121, 122 için
Csg_Prio	USINT	OB_Class = 121, 122 için
Width	USINT	OB_Class = 121, 122 için

NOT

"Standart" blok özelliğiyle oluşturma sırasında, SI_classic yapısı için belirtilen yapı elemanları, içerik olarak bir OB'nin geçici değişkenleriyle aynıdır.

Ancak münferit OB'lerin geçici değişkenlerinin farklı adlara ve farklı veri tiplerine sahip olabileceğine dikkat edin. Her OB'nin çağrı arayüzünün, OB talebinin tarihi ve saatiyle ilgili daha fazla bilgiler içerdiğine de dikkat edin.

EV_CLASS yapı elemanının 4 ile 7 arasındaki Bit'leri olay sınıfını içerir. Burada şu değerlere izin verilir:

- 1: Standart OB'lerin yol verme olayları
- 2: Senkron hata OB'lerinin yol verme olayları
- 3: Asenkron hata OB'lerinin yol verme olayları

PRIORITY yapı elemanı, güncel OB'nin öncelik sınıfını sağlar.

Bu iki elemanın yanı sıra, NUM de ilgilidir. NUM , güncel OB'nin numarasını veya en son başlatılan yol alma OB'sini içerir.

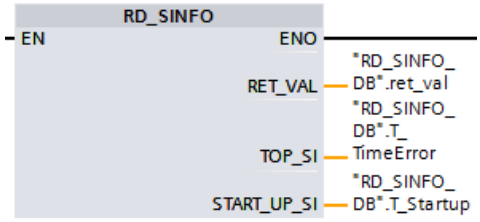
RET_VAL parametresi

Aşağıdaki tabloda, RET_VAL parametre değerlerinin anlamı gösterilmiştir.

Hata kodu* (W#16#...)	Anlamı
8081	Güncel OB'nin başlangıç bilgisi, belirtilen sistem veri tipine karşılık gelmiyor
8083	En son başlatılan yol alma OB'sinin başlangıç bilgisi, belirtilen sistem veri tipine karşılık gelmiyor
*Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenebilir.	

Örnek

Bir zaman hatası alarmı OB'si (OB 80), en son çağrılan ve tümüyle işlenmeyen OB'dir. Yol alma OB'si (OB 100), en son başlatılan yol alma OB'sidir. Yol alma bilgilerini okuma yordam çağrısı aşağıdaki gibidir, bu sırada RD_SINFO_DB , OB tipleri için kullanılan SDT'lerin değişkenlerini içeren veri bloğudur:



Aşağıdaki tabloda, RD_SINFO yordamının TOP_SI parametresine ait yapı elemanlarının, OB 80'in ilgili yerel değişkenlerine atanması gösterilmektedir.

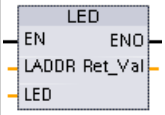
TOP_SI yapı elemanı	**Veri tipi**	**OB 80 - İlgili yerel değişken**	**Veri tipi**
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

Aşağıdaki tabloda, RD_SINFO yordamının START_UP_SI parametresine ait yapı elemanlarının, OB 100'ün ilgili yerel değişkenlerine atanması gösterilmektedir.

START_UP_SI yapı elemanı	**Veri tipi**	**OB 100 - Yerel değişken**	**Veri tipi**
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

9.7.2 LED (LED durumunu okuma)

Tablo 9-148 LED yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := LED(** **laddr:=_word_in_,** **LED:=_uint_in_);**	LED yordamıyla, CPU'daki LED'lerin durumunu (Sayfa 1113) okuyabilirsiniz. Belirtilen LED durumu, RET_VAL çıkışı tarafından döndürülür.

Tablo 9-149 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama		
LADDR	IN	HW_IO	CPU'nun kodu[1]		
LED	IN	UInt	LED tanımlama numarası		
			1	RUN/STOP	Renk 1 = yeşil, Renk 2 = sarı
			2	Hata	Renk 1 = kırmızı
			3	Bakım	Renk 1 = sarı
RET_VAL	OUT	Int	LED'in durumu		

[1] Bağlı CPU'nun kodu için, parametrenin açılır listesinden Local~Common öğesini seçebilirsiniz.

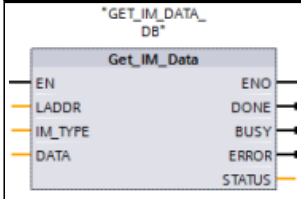
Tablo 9-150 RET_VAL'in durumu

RET_VAL (W#16#...)	Açıklama	
LED durumu 0 ila 9	0	LED mevcut değil
	1	Kapalı
	2	Renk 1 açık (sürekli olarak yanar)
	3	Renk 2 açık (sürekli olarak yanar)
	4	Renk 1, 2 Hz ile yanıp sönüyor
	5	Renk 2, 2 Hz ile yanıp sönüyor
	6	Renk 1 ve 2, değişimli olarak 2 Hz ile yanıp sönüyor
	9	LED'in durumu mevcut değil
8091	LADDR tarafından belirtilen cihaz mevcut değil	
8092	LADDR tarafından belirtilen cihaz LED'leri desteklemiyor	
8093	LED kodu tanımlanmamış	
80Bx	LADDR tarafından belirtilen CPU, LED yordamını desteklemiyor	

9.7.3 Get_IM_Data (tanımlama ve bakım verilerini oku)

Get_IM_Data yordamıyla, belirtilen modül veya alt modül için kullanılan tanımlama ve bakım verilerini (I&M) kontrol edebilirsiniz.

Tablo 9-151 Get_IM_Data yordamı

LAD/FBD	SCL	Açıklama
	**"GET_IM_DATA_DB"(LADDR:=16#0, IM_TYPE:=0, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, DATA:=_variant_inout_);**	Get_IM_Data yordamıyla, belirtilen modül veya alt modül için kullanılan tanımlama ve bakım verilerini (I&M) kontrol edebilirsiniz.

Tablo 9-152 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
LADDR	Input	HW_EA	Bloğun kodu
IM_TYPE	Input	UInt	Tanımlama ve bakım verilerinin (I&M) numarası: • 0: I&M0 (MLFB, seri numarası, sürüm ve diğer bilgiler) • 1: I&M1 (belirticiler) • 2: I&M2 (kurulum tarihi) • 3: I&M3 (tanımlayıcı) • 4: I&M4 (imza)
RET_VAL	Output	Int	Durum (koşul kodu)
DATA	InOut	Varyant	I&M verileri (STRING veya BYTE'tan oluşan bir Array); IM_TYPE = 0 için "IM0_Data" SDT'sinin kullanımı önerilir.

Tanımlama ve bakım verileri (I&M) aracılığıyla sistem konfigürasyonunu kontrol edebilir, donanım değişikliklerini algılayabilir veya bakım verilerini görüntüleyebilirsiniz. (Modül tanımlama verileri) I verileri yazma korumalıdır. Modül bakım verileri (M verileri), örn. kurulum tarihi gibi sistem bilgilerine bağlıdır. M verileri, bakım planlaması sırasında oluşturulur ve modüle yazılır:

- DATA parametresinin veri tipinde bir karakter dizisi söz konusuysa, karakter dizisinin güncel uzunluğu, I&M verilerinin uzunluğuna uygun şekilde belirlenir.
- DATA parametresinin veri tipinde Byte veya Char öğelerinden oluşan bir Array söz konusu ise, I&M verileri, Bayt Array'i olarak kopyalanır.
- DATA parametresinin veri tipinde bir yapı söz konusu ise, I&M verileri Bayt dizisi olarak kopyalanır.
- DATA'daki Bayt/Char öğelerinden oluşan ilgili Array, talep edilen I&M verilerinden daha uzun ise, 16#00 Bayt değeri eklenir.
- Diğer veri tipleri desteklenmez, 8093 hatası döndürülür.

Tablo 9-153 Koşul kodları

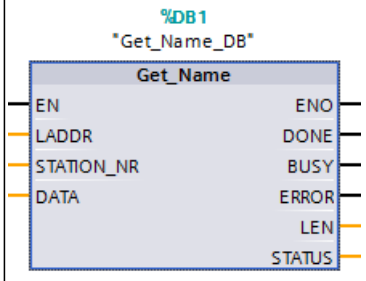
RET_VAL (W#16#...)	Açıklama
0	Hata yok
8091	LADDR mevcut değil
8092	LADDR, I&M verilerini destekleyen hiçbir donanım nesnesini adreslemez
8093	DATA parametresinde belirtilen veri tipi desteklenmez

RET_VAL (W#16#...)	Açıklama
80B1	DATA yordamı, CPU tarafından bu LADDR parametresi için desteklenmez
80B2	IM_TYPE, CPU tarafından desteklenmez
8452	Tam I&M bilgileri, DATA parametresinde belirtilen değişkene uymuyor. Değişkenin Bayt uzunluğuyla ilgili bir kısmi sonuç döndürülür.

9.7.4 Get_Name (Bir PROFINET IO cihazının adını oku)

"Get_Name" yordamı, bir PROFINET IO cihazının, PROFIBUS Slave'lerinin veya AS-i Slave'lerinin adını okur. Ad, ağ görünümünde ve IO cihazının özelliklerinde görüntülenir.

Tablo 9-154 Get_Name yordamı

LAD/FBD	SCL	Açıklama
	**"Get_Name_DB"(** **LADDR:=_uint_in_,** **STATION_NR:=_uint_in_,** **DONE=>_bool_out_,** **BUSY=>_bool_out_,** **ERROR=>_bool_out_,** **LEN=>_dint_out_,** **STATUS=>_word_out_,** **DATA:=_variant_inout_);**	"Get_Name " yordamıyla, bir PROFINET IO cihazının veya PROFIBUS Slave'lerinin adını okuyabilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "Get_Name_DB", örnek DB'nin adıdır.

IO cihazını, desantral IO sisteminin donanım kodu (LADDR parametresinde) ve PROFINET IO cihazının cihaz numarası veya PROFIBUS Slave'inin PROFIBUS adresi (STATION_NR parametresi) aracılığıyla seçebilirsiniz.

Program, yordam yürütüldükten sonra IO cihazının adını DATA parametresi tarafından adreslenen aralığa yazar.

Okunan ad, IO cihazının tipine bağlıdır:

- DP-Slave veya IO cihazı: Kafa modülünün adı
- I-Slave veya I-Device: Arayüz modülünün adı
- HMI paneli: Arayüzün adı
- PC istasyonu: Arayüz modülünün adı
- GSD cihazları: Device Access Point'in (DAP) adı (arayüzün veya Head modülünün adı) görüntülenir

Yordam, adın uzunluğunu LEN parametresine yazar. Ad, DATA parametresinde belirtilen aralıktan daha uzunsa, program sadece, adreslenen aralığın maksimum uzunluğuna karşılık gelen kısmını yazar.

Bir ad için maksimum uzunluk 128 karakterdir.

NOT

CPU adının okunması (Sürüm 1.1)

LADDR ve STATION_NR, parametrelerinin her ikisinde de 0 değerini atarsanız, yordam, CPU'nun adını yazar.

Parametre

Aşağıdaki tabloda, Get_Name yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Açıklama
LADDR	IN	HW_IOSYSTEM	Desantral IO sisteminin donanım kodu (HW-IoSystem) Numara, IO sisteminin sistem sabitleri veya özellikleri tarafından çağrılır.
STATION_NR	IN	UInt	• PROFINET IO cihazı: Cihaz numarası, "Ethernet adresleri" kısmındaki IO cihazının özelliklerinden ağ görünümüne aktarılır. • PROFIBUS-Slave: PROFIBUS adresi, "PROFIBUS adresi" kısmındaki PROFIBUS Slave'inin özelliklerinden ağ görünümüne aktarılır.
DATA	IN_OUT	Variant	İşaretçiyi, aralığın yazılacağı aralığın üzerine götürün.
DONE	OUT	Bool	Yordam başarıyla yürütülür. Modülün ad, DATA parametresindeki aralığa aktarılır.
BUSY	OUT	Bool	Durum parametresi: • 0: Yordamın yürütülmesi tamamlandı. • 1: Yordamın yürütülmesi henüz tamamlanmadı.
ERROR	OUT	Bool	Durum parametresi: • 0: Hata yok • 1: Yordamın yürütülmesi sırasında bir hata oluştu. STATUS parametresi ayrıntılı bilgiler içermektedir.
LEN	OUT	DInt	IO cihazının adının uzunluğu (karakter sayısı).
STATUS	OUT	Word	Durum parametresi: Parametre, sadece bir çağrının süresi boyunca ayarlanır. Durumu görüntülemek için, bu nedenle STATUS parametresini boş bir veri alanına kopyalamanız gerekir.

STATUS parametresi

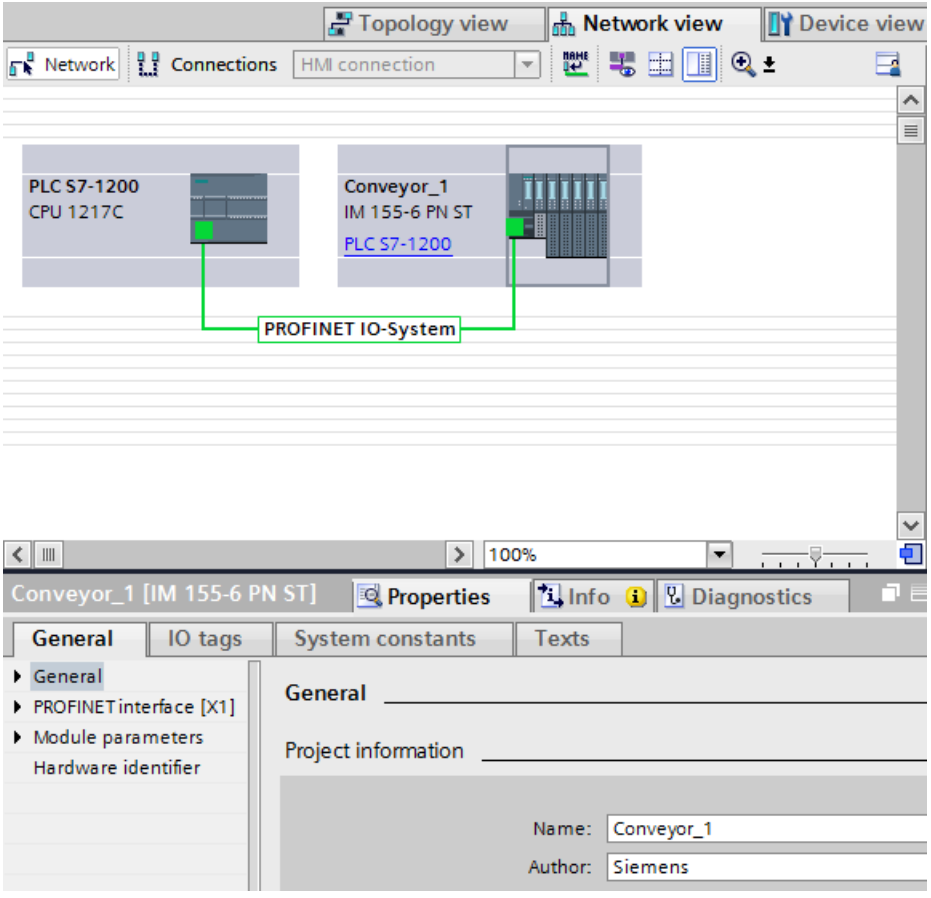
Hata kodu* (W#16#...)	Anlamı
0	Hata yok
7000	Düzenlenen görev yok.
7001	Get_Name asenkron yordamının ilk çağrısı. Yordamın yürütülmesi henüz tamamlanmadı (BUSY = 1, DONE = 0).
7002	Get_Name asenkron yordamının ek çağrısı. Yordamın yürütülmesi henüz tamamlanmadı (BUSY = 1, DONE = 0).
8090	LADDR parametresinde belirtilen donanım kodu, projede mevcut değil.
8092	LADDR parametresindeki değer, hiçbir PROFINET IO sistemini adreslemiyor.
8093	Yordam, DATA parametresindeki veri tipini desteklemiyor.
8095	Cihaz numarası (STATION_NR parametresi), seçilen PROFINET IO sisteminde mevcut değil veya hiçbir IO cihazını adreslemiyor.
80B1	Kullanılan CPU, yordamı desteklemiyor.
80C3	Geçici kaynak hatası: CPU, şu anda mümkün olan maksimum sayıda eşzamanlı blok çağrısını işliyor. Get_Name, ancak blok çağrılarından en az biri işlendiyse yürütülebilir.

Hata kodu* (W#16#...)	Anlamı
8852	DATA parametresinde belirtilen aralık, IO cihazının tam adı için çok kısa. Ad, mümkün olan maksimum uzunluğa kadar yazılabilir. Tam adı okumak için, DATA parametresinde daha uzun bir veri alanı kullanın. Alan, en azından LEN parametresi tarafından önceden belirtildiği kadar karaktere sahip olmalıdır.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.	

Örnek

Aşağıdaki örnekte, bir ET 200SP PROFINET IO cihazının istasyon adını nasıl okuyabileceğiniz gösterilmektedir:

1. ET200SP'nin yapılandırılması:
   - ET 200SP'yi ağ görünümünde "Conveyor_1" istasyon adıyla oluşturun ve CPU ile aynı PROFINET IO sistemine atayın.
   - CPU'yu, ET 200SP için IO kontrolörü olarak atayın.
   - "Ethernet adresleri"nin altındaki özellikler bölümünde standart cihaz numarası "1"i kullanın.

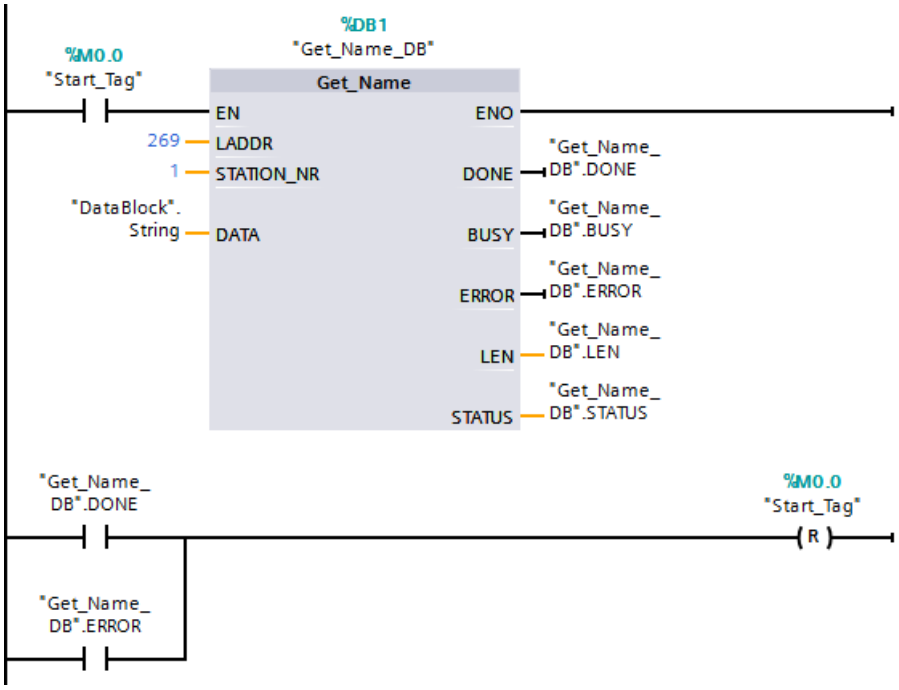


2. Get_Name yordamı için kullanılan parametrelerin atanması:
   - LADDR parametresine IO sisteminin donanım ID'sini girin. Bu örnekte donanım ID'si "269"dur. Donanım ID'sini şu konumda bulabilirsiniz:
     PLC değişkenleri > Tüm değişkenleri göster > "Sistem sabitleri" sekmesi > Yerel PROFINET_IO sistemi.
   - STATION_NR parametresine ET 200SP'nin cihaz numarasını girin. Bu örnekte cihaz numarası "1"dir.
   - Bir veri bloğunun STRING veri tipindeki bir değişkeni DATA parametresine bağlayın.

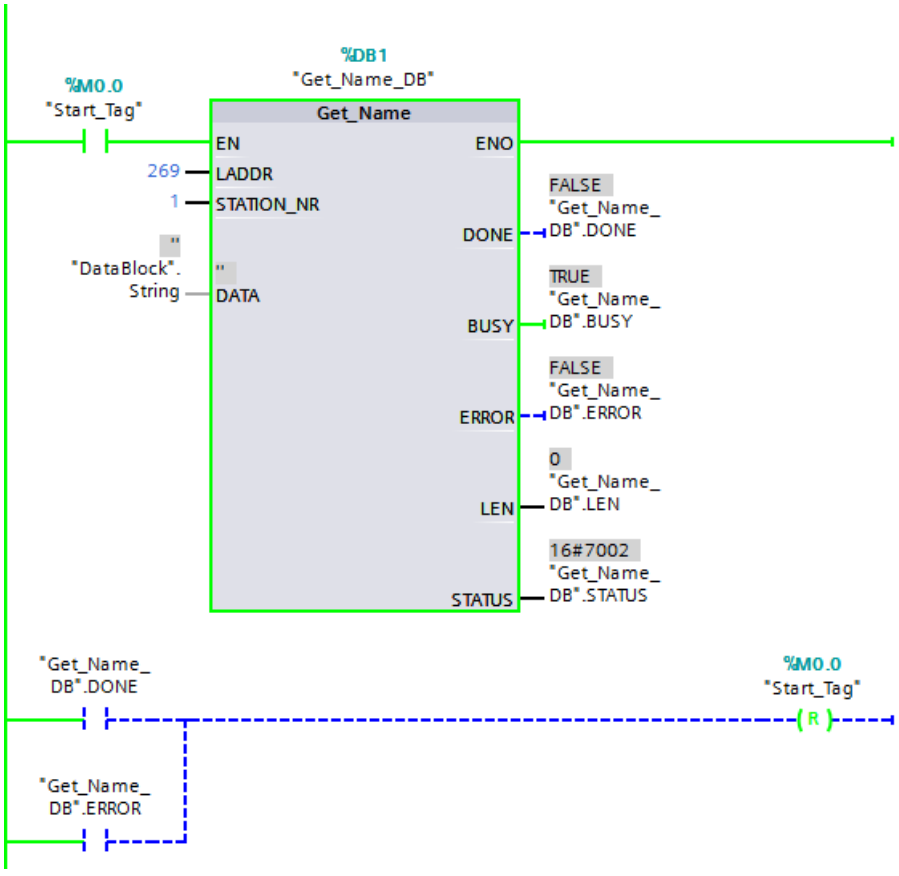
   NOT

   DATA parametresindeki değişkenlerin konfigürasyonunda açılır listeyi kullanırsanız, DB'yi ("Datablock" örneğinde) ve değişkeni ("String[ ]" örneğinde) seçin. Tüm String veri tipini okumak için, aşağıdaki nihai sonucu almak üzere parantezleri silmeniz gerekir: "Datablock".String.

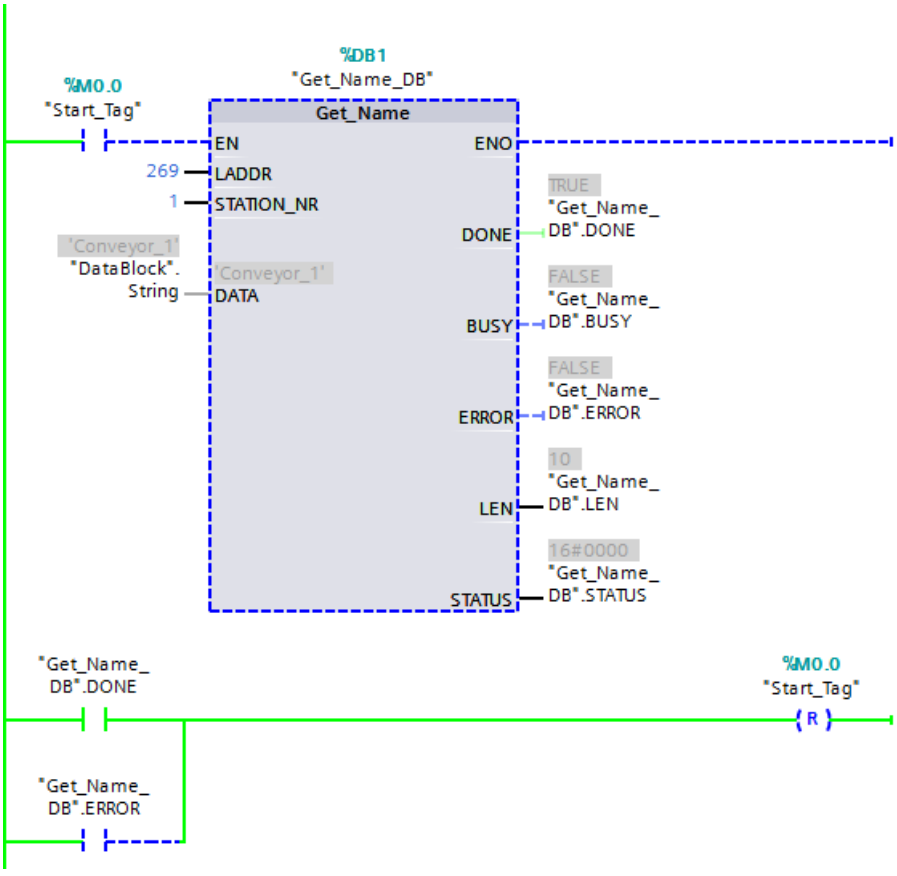
   - Yordamın çıkış parametresi için PLC değişkenlerini (bellek alanı, markör) tanımlayın.



3. Get_Name yordamının yürütülmesi:
   – Yordam yürütülürken BUSY çıkış parametresi 1'e ayarlanabilir, bunun üzerine DONE parametresi 0'a ayarlanır.
   – Hata koduyla ilgili bilgiler, STATUS çıkış parametresinde görüntülenir.



4. Get_Name yordamını yürütmeyi tamamlama:
   – Yordam yürütüldükten sonra "Conveyor_1" programı, ET 200SP'nin istasyon adını, DATA parametresindeki veri bloğuna yazar.
   – Program, istasyon adındaki karakter sayısı olan "10" değerini LEN parametresine yazar.



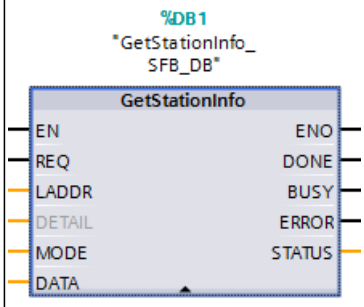
9.7.5 GetStationInfo (bir PROFINET IO cihazının IP veya MAC adresini oku)

"GetStationInfo" yordamı, yerel IO sistemindeki bir PROFINET IO cihazının veya alt seviyedeki bir IO sisteminde (CP/CM modülleri üzerinden bağlıdır) bulunan bir PROFINET IO cihazının IP veya MAC adresini okur.

NOT

GetStationInfo yordamını sadece PROFINET IO cihazlarında kullanabilirsiniz. Yordamı PROFIBUS DP cihazlarıyla kullanamazsınız.

Tablo 9-155 GetStationInfo yordamı

LAD/FBD	SCL	Açıklama
	**"GetStationInfo_SFB_DB"(** **REQ:=_bool_in_,** **LADDR:=_uint_in_,** **DETAIL:=_uint_in_,** **MODE:=_uint_in_,** **DONE=>_bool_out_,** **BUSY=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **DATA:=_variant_inout_);**	GetStationInfo yordamıyla, bir PROFINET IO cihazının IP veya MAC adresini okuyabilirsiniz. Bu yordamla, alt seviyedeki bir IO sisteminde (CP/CM modülleri üzerinden bağlıdır) bulunan bir IO cihazının IP veya MAC adresini de okuyabilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "GetStationInfo_SFB_DB", örnek DB'nin adıdır.

IO cihazını, LADDR parametresindeki istasyonun donanım kodu üzerinden adresleyebilirsiniz. Donanım ID'sini şu konumda bulabilirsiniz:
PLC değişkenleri > Tüm değişkenleri göster > "Sistem sabitleri" sekmesi. "Ad" sütununda IO cihazını ve "Veri tipi" sütununda "Hw_Device"ı arayın.

MODE parametresi aracılığıyla, okunması gereken bilgileri seçebilirsiniz.

DATA parametresinde, yordamın okunan adres verilerini yazdığı veri alanını atayabilirsiniz. IP adresini kaydetmek için "IF_CONF_v4" yapısını kullanabilirsiniz. MAC adresini kaydetmek için "IF_CONF_MAC" yapısını kullanabilirsiniz.

Adres verilerinin okunmasını, REQ kumanda parametresi üzerinden etkinleştirebilirsiniz. Bunun için IO cihazı erişilebilir durumda olmalıdır.

Yordam; BUSY, DONE ve ERROR çıkış parametreleri ve STATUS çıkış parametresi üzerinden okuma görevinin yürütme durumunu gösterir.

NOT

IO cihazını sadece istasyonun donanım kodu üzerinden adresleyin.

İstasyon, IO cihazı ve PROFINET arayüzü kendilerine ait birer donanım koduna sahiptir. GetStationInfo yordamı için sadece istasyonun donanım kodunu kullanın.

Örneğin bir PROFINET arayüzü LADDR parametresi üzerinden adreslenirse, adres verileri okunmaz ve CPU, "8092" hata kodunu oluşturur.

Entegre bir PROFINET arayüzünün veya merkezi konfigürasyondaki bir CM/CP modülünün adres verilerini okumak için "RDREC" yordamını kullanın.

Parametre

Aşağıdaki tabloda, GetStationInfo yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Açıklama
REQ	IN	Bool	REQUEST kumanda parametresi REQ = 1 ile bilgilerin okunmasını etkinleştirir.
LADDR	IN	HW_DEVICE	IO cihazındaki istasyonun donanım kodu Numara, ağ görünümündeki istasyonun özelliklerinden veya standart değişken tablosundaki "Sistem sabitleri" sekmesinden öğrenilir.
DETAIL	IN	HW_SUBMODULE	Bu DETAIL parametresi kullanılmaz. Parametreyi bağlamayın.
MODE	IN	UNIT	Okunması gereken adres verilerinin seçimi: • MODE = 1: IPv4'e göre adres parametresi • MODE = 2: MAC adresi
DATA	IN_OUT	Variant	İşaretçiyi, programın, IO cihazının adres verilerini yazacağı aralığın üzerine götürün. MODE = 1'de "IF_CONF_v4" yapısını ve MODE = 2'de "IF_CONF_MAC" yapısını kullanın.
DONE	OUT	Bool	Program, yordamı başarılı bir şekilde yürüttü. Program, adres verilerini DATA parametresine aktardı.
BUSY	OUT	Bool	STATUS parametresi: • 0: Yordamın yürütülmesi tamamlandı. • 1: Yordamın yürütülmesi henüz tamamlanmadı.
ERROR	OUT	Bool	STATUS parametresi: • 0: Hata yok. • 1: Yordamın yürütülmesi sırasında bir hata oluştu. Ayrıntılı bilgiler, STATUS parametresi aracılığıyla yayınlanır.
STATUS	OUT	Word	STATUS parametresi: Parametre, sadece bir çağrının süresi boyunca ayarlanır. Durumu görüntülemek için, bu nedenle STATUS parametresini boş bir veri alanına kopyalamanız gerekir.

DATA parametresi

- Adres verilerini IPv4'e uygun şekilde kaydetmek için, DATA parametresinde "IF_CONF_v4" yapısını kullanın:

Bayt	Parametre	Veri tipi	Başlangıç değeri	Açıklama
0 ... 1	Id	UINT	30	"IF_CONF_v4" yapısının ID'si
2 ... 3	Length	UNIT	18	BYTE'ta okunan verilerin uzunluğu
4 ... 5	Mode	UNIT	0	"GetStationInfo" yordamında ilgili değildi (0 olarak kalır)
6 ... 9	InterfaceAddress	ARRAY [1..4] of BYTE	-	IO cihazının IP_V4 formatında IP adresi (örnek: 192.168.3.10): • addr[1] = 192 • addr[2] = 168 • addr[3] = 3 • addr[4] = 10

Bayt	Parametre	Veri tipi	Başlangıç değeri	Açıklama
10 ... 13	SubnetMask	ARRAY [1..4] of BYTE	-	IP_V4 formatında IO cihazının alt ağ maskesi (örnek: 255.255.255.0): • addr[1] = 255 • addr[2] = 255 • addr[3] = 255 • addr[4] = 0
14 ... 17	DefaultRouter	ARRAY [1..4] of BYTE	-	Yönlendiricinin IP_V4 formatında IP adresi (örnek: 192.168.3.1): • addr[1] = 192 • addr[2] = 168 • addr[3] = 3 • addr[4] = 1

- MAC adresini kaydetmek için DATA parametresinde "IF_CONF_MAC" yapısını kullanabilirsiniz.

Bayt	Parametre	Veri tipi	Başlangıç değeri	Açıklama
0 ... 1	Id	UINT	3	"IF_CONF_MAC" yapısının ID'si
2 ... 3	Length	UNIT	12	BYTE'ta okunan verilerin uzunluğu
4 ... 5	Mode	UNIT	0	"GetStationInfo" yordamında ilgili değildi (0 olarak kalır)
6 ... 11	MACAddress	ARRAY [1..6] of BYTE	-	IO cihazının MAC adresi (örnek: 08-00-06-12-34-56): • Mac[1] = 8 • Mac[2] = 0 • Mac[3] = 6 • Mac[4] = 12 • Mac[5] = 34 • Mac[6] = 56

STATUS parametresi

Hata kodu* (W#16#...)	Anlamı
0	Hata yok
7000	Düzenlenen görev yok.
7001	GetStationInfo asenkron yordamının ilk çağrısı. Yordamın yürütülmesi henüz tamamlanmadı (BUSY = 1, DONE = 0).
7002	GetStationInfo asenkron yordamının ek çağrısı. Yordamın yürütülmesi henüz tamamlanmadı (BUSY = 1, DONE = 0).
8080	MODE parametresindeki değer desteklenmiyor.
8090	LADDR parametresinde belirtilen donanım kodu yapılandırılmadı.
8092	LADDR parametresi hiçbir PROFINET IO cihazını adreslemiyor.
8093	DATA parametresinde geçersiz veri tipi.
80A0	Talep edilen bilgiler okunmuyor.

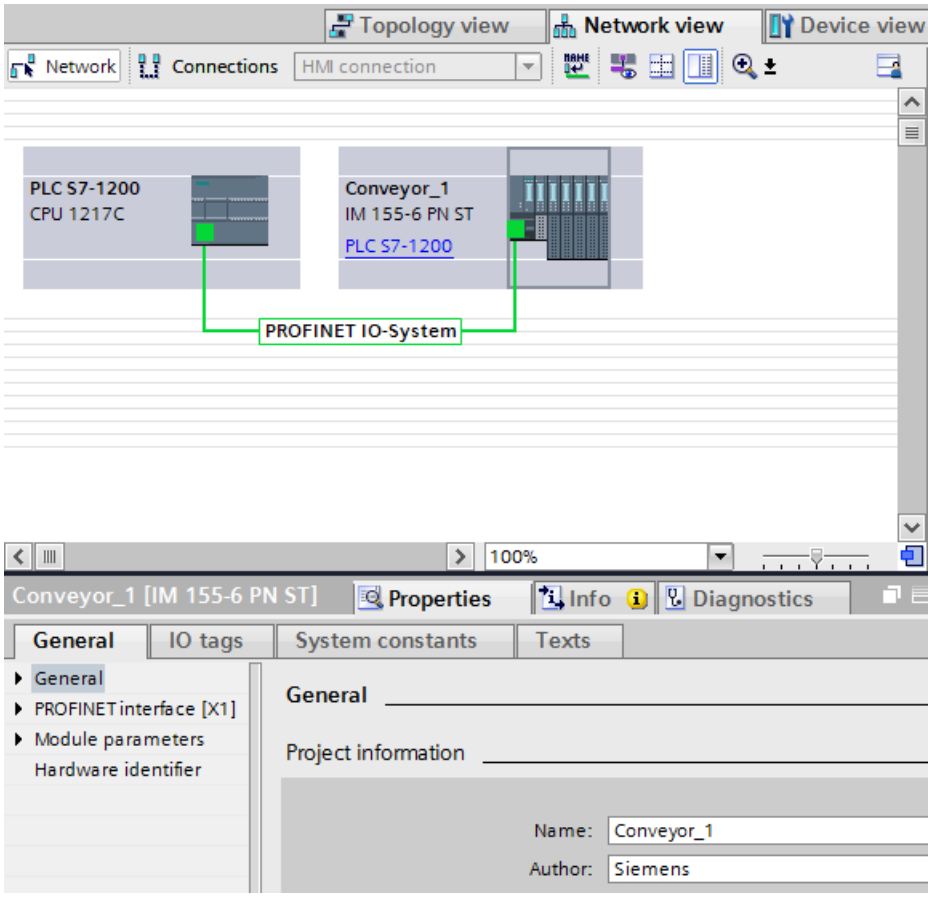
Hata kodu* (W#16#...)	Anlamı
80C0	Adreslenen IO cihazı erişilemez durumda.
80C3	GetStationInfo yordamının (10 örnek) eşzamanlı maksimum çağrı sayısına ulaşıldı.
* Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenir.	

Örnek

Aşağıdaki örnekte, GetStationInfo yordamıyla bir IO cihazının IP adresi verilerini okuyabilir ve bilgileri bir veri bloğuna yazabilirsiniz. IP adresi verileri IP adresini, alt ağ maskesini ve (kullanılıyorsa) yönlendiricinin adres verilerini kapsar.

IO kontrolörü, GetStationInfo yordamını yürütür ve yordam, alt seviyedeki bir IO cihazının (bu örnekte bir ET200SP) IP adresi verilerini okur:

1. ET200SP'nin yapılandırılması:
   – ET 200SP'yi ağ görünümünde "Conveyor_1" istasyon adıyla oluşturun ve CPU ile aynı PROFINET IO sistemine atayın.
   – CPU'yu, ET 200SP için IO kontrolörü olarak atayın.



2. GetStationInfo yordamı için kullanılan parametrelerin atanması:
   – IP adresi verilerini kaydetmek üzere global bir veri bloğunda IF_CONF_v4 veri tipinde beş değişken ve bir yapı oluşturun. Yapıya herhangi bir ad verin. (Örnekte yapının adı "IP_Address" şeklindedir.)

GetStationInfo_Global_DB

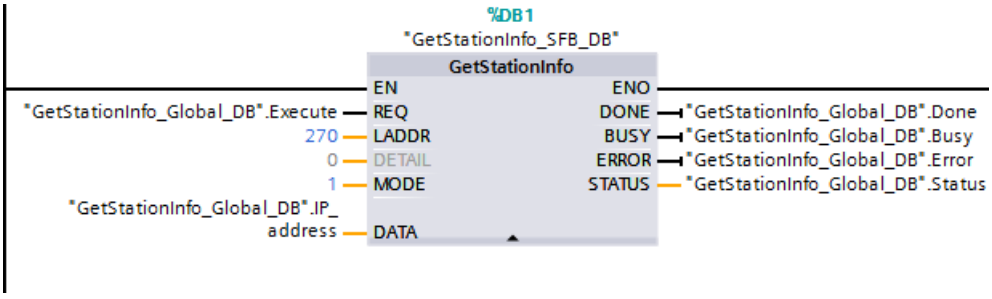
	Name	Data type	Start value
1	Static		
2	Execute	Bool	false
3	IP_address	IF_CONF_v4	
4	Id	UInt	30
5	Length	UInt	18
6	Mode	UInt	0
7	InterfaceAddress	IP_V4	
8	ADDR	Array[1..4] of Byte	
9	ADDR[1]	Byte	16#0
10	ADDR[2]	Byte	16#0
11	ADDR[3]	Byte	16#0
12	ADDR[4]	Byte	16#0
13	SubnetMask	IP_V4	
14	DefaultRouter	IP_V4	
15	Done	Bool	false
16	Busy	Bool	false
17	Error	Bool	false
18	Status	Word	16#0

3. GetStationInfo yordamı için kullanılan parametrelerin atanması:
   – IO cihazının donanım ID'sini LADDR parametresine girin. Donanım kodu, ürünü benzersiz şekilde tanımlar. Bu örnekte donanım ID'si "270"tir. Donanım ID'sini şu konumda bulabilirsiniz: PLC değişkenleri > Tüm değişkenleri göster > "Sistem sabitleri" sekmesi.
   "Ad" sütununda IO cihazını ve "Veri tipi" sütununda "Hw_Device"ı arayın. Atanan değer, LADDR parametresinde girdiğiniz donanım ID'sidir.
   – MODE parametresi için 1 değerini (IPv4'e göre adres parametresini oku) seçin.
   – IF_CONF_v4 yapısını DATA parametresine bağlayın.

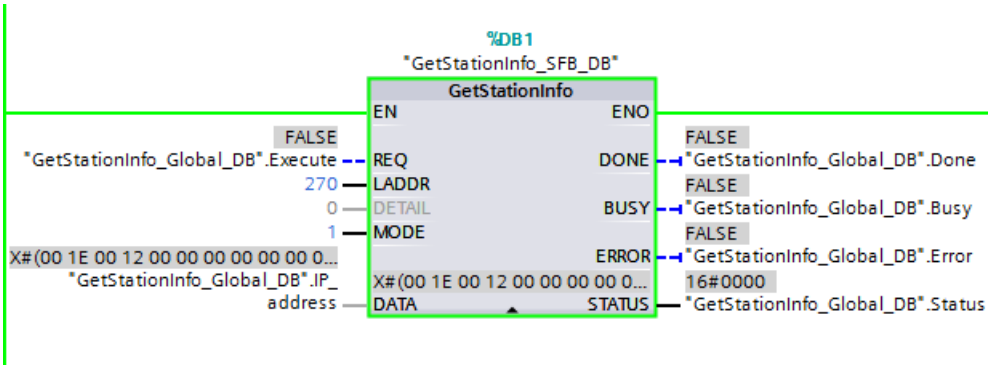
NOT

DATA parametresinde değişkenlerin konfigürasyonunda seçim yaparken açılır listeyi kullanırsanız, DB'yi ("GetStationInfo_Global_DB" örneğinde) ve değişkeni ("IP address" örneğinde) seçin. Tüm IF_CONF_v4 veri tipini okumak için, aşağıdaki sonucu elde etmek üzere, "IP address" ifadesini takip eden noktayı silmeniz gerekir: "GetStationInfo_Global_DB".IP address

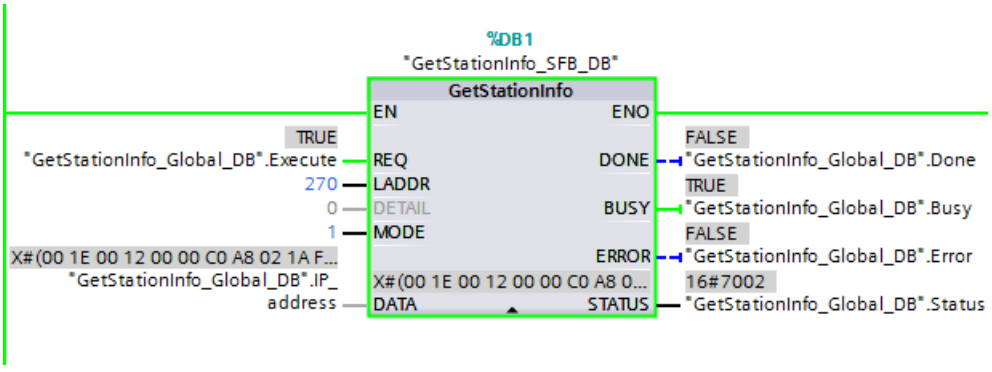
– Yordamın çıkış parametresi için global DB'nizin PLC değişkenlerini tanımlayın.



4. GetStationInfo yordamının yürütülmesi:
   – REQ girişi = 1 ise (FALSE), yordam, DATA giriş/çıkış parametresinde IP adresi verilerini görüntülemez veya STATUS çıkış parametresinde hata koduyla ilgili bilgileri görüntüler.



5. GetStationInfo yordamını yürütmeyi tamamlama:
   – REQ girişi = 1 ise (TRUE), program, yordamı yürütür ve IP adresini veri bloğuna yazar. Program, "C0 A8 02 1A" IP adresini ("192.168.2.26"nın desimal değeri), DATA giriş/çıkış parametresine yazar.



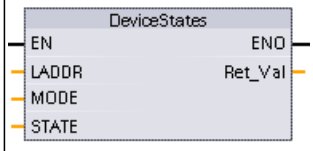
GetStationInfo_Global_DB

	Name	Data type	Start value	Monitor value
1	Static			
2	Execute	Bool	false	TRUE
3	IP_address	IF_CONF_v4		
4	Id	UInt	30	30
5	Length	UInt	18	18
6	Mode	UInt	0	0
7	InterfaceAddress	IP_V4		
8	ADDR	Array[1..4] of Byte		
9	ADDR[1]	Byte	16#0	16#C0
10	ADDR[2]	Byte	16#0	16#A8
11	ADDR[3]	Byte	16#0	16#02
12	ADDR[4]	Byte	16#0	16#1A
13	SubnetMask	IP_V4		
14	DefaultRouter	IP_V4		
15	Done	Bool	false	TRUE
16	Busy	Bool	false	FALSE
17	Error	Bool	false	FALSE
18	Status	Word	16#0	16#0000

9.7.6 DeviceStates yordamı

DeviceStates yordamıyla, belirli bir desantral G/Ç Master'a bağlanmış olan tüm desantral G/Ç Slave cihazlarının durumlarını yayınlayabilirsiniz.

Tablo 9-156 DeviceStates yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := DeviceStates(laddr:=_word_in_, mode:=_uint_in_, state:=_variant_inout_);**	DeviceStates yordamı, bir G/Ç alt sistemindeki G/Ç cihazlarının işletim durumlarını çağırır. Yürütme işleminden sonra STATE parametresi, her münferit G/Ç cihazının hata durumunu bir Bit listesinde (atanan LADDR ve MODE parametresi için) içerir. Bu bilgiler, STEP 7'nin teşhis görünümündeki cihaz durumuna karşılık gelir. DeviceStates yordamının LADDR girişi, desantral bir G/Ç arayüzünün donanım kodunu kullanır. TIA Portal'da, "Sistem sabitleri" sekmesindeki PLC değişken tablosunda "HW_IOSYSTEM" veri tipleri aracılığıyla bir PLC için kullanılan donanım kodlarını bulabilirsiniz.

Tablo 9-157 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
LADDR	IN	HW_IOSYSTEM	Mantıksal adres: (G/Ç sisteminin kodu)

[1] PROFIBUS DP'de durum bilgilerinin uzunluğu 128 Bit'tir. PROFINET G/Ç'de bu uzunluk 1024 Bit'tir.

Parametre ve veri tipi		Veri tipi	Açıklama
MODE	IN	UInt	Beş işletim türünü destekler. MODE girişi, STATE bilgileri için belirtilen konumda hangi verilerin yayınlanacağını belirler. Aşağıdaki işletim türleri kullanılabilir: • 1: Cihaz konfigürasyonu aktif • 2: Cihaz arızalı • 3: Cihaz devre dışı bırakıldı • 4: Cihaz mevcut • 5: Cihazda sorun var
RET_VAL	OUT	Int	Yürütme koşulu
STATE[1]	InOut	Variant	Münferit cihazların hata durumunu alan tampon: STATE parametresi için veri tipi olarak herhangi bir Bit tipini (Bool, Byte, Word veya DWord) veya bir Bit tipinin bir Array'ini seçebilirsiniz. • Yayınlanan STATE verilerinin ilk Bayt'ındaki Bit 0 bir özet Bit'idir. DOĞRU olarak ayarlandıysa, başka verilerin mevcut olduğunu belirtir. • STATE parametresi tarafından yayınlanan veriler, bir Bit pozisyonuyla bir desantral G/Ç adresi arasındaki 1'e 1 eşleşmeyi gösterir. Bu cihaz adresleme, PROFIBUS ve PROFINET için DOĞRU'dur. Örnek: İlk Bayt'taki Bit 4, PROFIBUS adresi 4 veya PROFINET cihaz numarası 4 ile eşleşir.

[1] PROFIBUS DP'de durum bilgilerinin uzunluğu 128 Bit'tir. PROFINET G/Ç'de bu uzunluk 1024 Bit'tir.

Yürütme işleminden sonra STATE parametresi, her münferit G/Ç cihazının hata durumunu Bit listesi olarak (atanan LADDR ve MODE parametresi için) içerir.

Tablo 9-158 Koşul kodları

RET_VAL (W#16#...)	Açıklama
0	Hata yok
8091	LADDR mevcut değil.
8092	LADDR hiçbir G/Ç sistemini adreslemiyor.
8093	STATE parametresi için geçersiz veri tipi: Geçerli veri tipleri (Bool, Byte, Word veya DWord)'dür veya (Bool, Byte, Word veya DWord) öğelerinden oluşan bir Array'dir
80Bx	DeviceStates yordamı, CPU tarafından bu LADDR parametresi için desteklenmez
8452	Tam erişim verileri, atanan STATE parametresi için çok büyük. STATE tamponu bir kısmi sonuç içeriyor.

9.7.6.1 DeviceStates konfigürasyonuyla ilgili örnekler

PROFIBUS'taki örnek

PROFIBUS'taki örnek şu bileşenlerden oluşur:

- Adları "DPSlave_10" ile "DPSlave_25" arasında olan 16 PROFIBUS cihazı
- 16 PROFIBUS cihazının her biri, 10 ile 25 arasındaki PROFIBUS adreslerinden birini kullanır.
- Her Slave cihaz için çok sayıda G/Ç modülü yapılandırılmış.
- Yayınlanan STATE parametre bilgilerinin ilk dört Bayt'ı görüntülenir.

MODE	Örnek 1: Hatasız normal işletim	Örnek 2: Bir modülü çekilmiş DPSlave_12 PROFIBUS Slave cihazı	Örnek 3: DPSlave_12 PROFIBUS Slave cihazı ayrıldı
1: Cihaz konfigürasyonu aktif	0x01FC_FF03	0x01FC_FF03	0x01FC_FF03
2: Cihaz arızalı	0x0000_0000	0x0110_0000	0x0110_0000
3: Cihaz devre dışı bırakıldı	0x0000_0000	0x0000_0000	0x0000_0000
4: Cihaz mevcut	0x01FC_FF03	0x01FC_FF03	0x01EC_FF03
5: Cihazda sorun var	0x0000_0000	0x0110_0000	0x0110_0000

Aşağıdaki dört tabloda, analiz edilmiş dört veri Bayt'ının ikili çözümlemesi gösterilmektedir:

Tablo 9-159 Örnek 1: Hata yok: MODE 1 için (cihaz konfigürasyonu aktif) 0x01FC_FF03 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFC	Bit 15 1111-1100 Bit 8	
Bayt 3 0xFF	Bit 23 1111-1111 Bit 16	
Bayt 4 0x03	Bit 31 0000-0011 Bit 24	

Cihazlar, 10 (Bit 10) ile 25 (Bit 25) arasındaki adreslerde yapılandırılmıştır.
1 ile 9 arasındaki adreslerde hiçbir cihaz yapılandırılmamıştır.
MODE 4 (Cihaz mevcut) ve MODE 1 (Cihaz konfigürasyonu aktif) verileri eşleşir, yani yapılandırılmış cihazlar, mevcut cihazlara karşılık gelmez.

Tablo 9-160 Örnek 2: Bir modül, "DPSlave_12" PROFIBUS Slave cihazından çekildi. MODE 2 (cihaz arızalı) için 0x0110_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x10	Bit 15 000**1**-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 12 (Bit 12) arızalı olarak işaretlendi.
MODE 5 (Cihazda sorun var), MODE 2 (Cihaz arızalı) ile aynı bilgileri yayınlar.

Tablo 9-161 Örnek 2 (devam): Bir modül, "DPSlave_12" PROFIBUS Slave cihazından çekildi. MODE 4 (cihaz mevcut) için 0x01FC_FF03 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFC	Bit 15 111**1**-1100 Bit 8	
Bayt 3 0xFF	Bit 23 1111-1111 Bit 16	
Bayt 4 0x03	Bit 31 0000-0011 Bit 24	

Cihaz 12 (Bit 12), yukarıdaki MODE 2'de (Cihaz arızalı) gösterildiği gibi bir hata içeriyorsa, cihaz buna rağmen ağda çalışır, bu nedenle MODE 4 (Cihaz mevcut), cihazı "mevcut cihaz" olarak gösterir.

Tablo 9-162 Örnek 3: "DPSlave_12" PROFIBUS Slave cihazı, PROFIBUS ağından ayrıldı (kablo çıkartıldı veya gerilim kesildi). "DPSlave_12", arızalı cihaz olarak ve hatalı cihaz olarak algılanmaya devam eder. Aradaki fark, "DPSlave_12"nin artık mevcut cihaz olarak algılanmamasıdır. MODE 4 (cihaz mevcut) için 0x01EC_FF03 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xEC	Bit 15 111**0**-1100 Bit 8	
Bayt 3 0xFF	Bit 23 1111-1111 Bit 16	
Bayt 4 0x03	Bit 31 0000-0011 Bit 24	

Cihaz 12 (Bit 12) mevcut değil olarak işaretlendi. Bu istisna, geri kalan 10 ile 25 arasındaki cihazları mevcut olarak bildirmeye devam eder.

PROFINET'teki örnek

PROFINET'teki örnek şu bileşenlerden oluşur:

- "et200s_1" ile "et200s_16" arasında adlara sahip 16 PROFINET Slave cihazı.
- 16 PROFINET cihazının her biri, 1 ile 16 arasındaki PROFINET cihaz numaralarından birini kullanır.
- Her Slave cihaz için çok sayıda G/Ç modülü yapılandırılmış.
- Yayınlanan STATE parametre bilgilerinin ilk dört Bayt'ı görüntülenir.

MODE	Örnek 1: Hatasız normal işletim	Örnek 2: Modülü çekilmiş et200s_1 PROFINET Slave	Örnek 3: et200s_1 PROFINET Slave ayrıldı
1: Cihaz konfigürasyonu aktif	0xFFFF_0100	0xFFFF_0100	0xFFFF_0100
2 - Cihaz arızalı	0x0000_0000	0x0300_0000	0x0300_0000
3 - Cihaz devre dışı bırakıldı	0x0000_0000	0x0000_0000	0x0000_0000
4 - Cihaz mevcut	0xFFFF_0100	0xFFFF_0100	0xFDFF_0100
5 - Cihazda sorun var	0x0000_0000	0x0300_0000	0x0300_0000

Aşağıdaki dört tabloda, analiz edilmiş dört veri Bayt'ının ikili çözümlemesi gösterilmektedir:

Tablo 9-163 Örnek 1: Hata yok: MODE 1 (cihaz konfigürasyonu aktif) için 0xFFFF_0100 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFF	Bit 15 1111-1111 Bit 8	
Bayt 3 0x01	Bit 23 0000-0001 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Cihazlar, 1 (Bit 1) ile 16 (Bit 16) arasındaki adreslerde yapılandırılmıştır.
1 ile 9 arasındaki adreslerde hiçbir cihaz yapılandırılmamıştır.
MODE 4 (Cihaz mevcut) ve MODE 1 (Cihaz konfigürasyonu aktif) verileri eşleşir, yani yapılandırılmış cihazlar, mevcut cihazlara karşılık gelmez.

Tablo 9-164 Örnek 2: Bir modül, "et200s_1" PROFINET Slave cihazından çekildi. MODE 2 (cihaz arızalı) için 0x0300_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x03	Bit 7 0000-00**1**1 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x00	Bit 15 0000-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 1 (Bit 1) arızalı olarak işaretlendi. Cihaz hâlâ mevcut olduğu için MODE 4 (Cihaz mevcut), normal işletimdekiyle aynı verileri görüntüler.
MODE 5 (Cihazda sorun var), MODE 2 (Cihaz arızalı) ile aynı bilgileri yayınlar.

Tablo 9-165 Örnek 2 (devam): Bir modül, "et200s_1" PROFIBUS Slave cihazından çekildi. MODE 4 (cihaz mevcut) için 0xFFFF_0100 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0xFF	Bit 7 1111-11**1**1 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFF	Bit 15 1111-1111 Bit 8	
Bayt 3 0x01	Bit 23 0000-0001 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 1 (Bit 1), yukarıdaki MODE 2'de (Cihaz arızalı) gösterildiği gibi bir hata içeriyorsa, cihaz buna rağmen ağda çalışır, bu nedenle MODE 4 (Cihaz mevcut), cihazı "mevcut cihaz" olarak gösterir.

Tablo 9-166 Örnek 3: "et200s_1" PROFINET Slave cihazı, PROFINET ağından ayrıldı (kablo çıkartıldı veya gerilim kesildi). MODE 4 (cihaz mevcut) için 0xFDFF_0100 değeri yayınlanır.

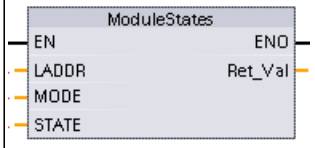
Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0xFD	Bit 7 1111-11**0**1 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFF	Bit 15 1111-1111 Bit 8	
Bayt 3 0x01	Bit 23 0000-0001 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 1 (Bit 1) mevcut değil. 2 (Bit 2) ile 16 (Bit 16) arasındaki cihazlar mevcuttur.

9.7.7 ModuleStates yordamı

ModuleStates yordamıyla bir PROFIBUS veya PROFINET istasyonundaki tüm modüllerin durumunu yayınlayabilirsiniz.

Tablo 9-167 ModuleStates yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := ModuleStates(** **laddr:=_word_in_,** **mode:=_uint_in,** **state:=_variant_inout);**	ModuleStates, G/Ç modüllerinin işletim durumlarını çağırır. Yürütme işleminden sonra STATE parametresi, her münferit G/Ç modülünün hata durumunu bir Bit listesinde (atanan LADDR ve MODE parametresi için) içerir. Bu bilgiler, STEP 7'nin teşhis görünümündeki modül durumuna karşılık gelir. ModuleStates yordamının LADDR girişi, kafa modülünün kendisinin donanım kodu yerine desantral bir G/Ç istasyonunun donanım kodunu kullanır. Donanım kodunu, tüm istasyonu ağ görünümünde seçerek ve ardından özellikler kısmındaki "Donanım kodu" alanına bakarak belirleyebilirsiniz. Bu öğeleri ayrıca PLC değişken tablosunda bulunan "Sistem sabitleri" sekmesindeki "Hw_Device" ve "Hw_DpSlave" veri tipleri aracılığıyla da bulabilirsiniz.

Tablo 9-168 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
LADDR	IN	HW_DEVICE	Mantıksal adres (G/Ç modüllerinin kodu)
MODE	IN	UInt	Beş işletim türünü destekler. MODE girişi, STATE bilgileri için belirtilen konumda hangi verilerin yayınlanacağını belirler. Aşağıdaki işletim türleri kullanılabilir: • 1: Modül konfigürasyonu aktif • 2: Modül arızalı • 3: Modül devre dışı bırakıldı • 4: Modül mevcut • 5: Modülde sorun var
RET_VAL	OUT	Int	Durum (koşul kodu)

[1] Maksimum 128 Bit atanabilir. Gereken Bit sayısı, G/Ç modülünüzün kullanımına bağlıdır.

Parametre ve veri tipi		Veri tipi	Açıklama
STATE[1]	InOut	Variant	Münferit modüllerin hata durumunu alan tampon: STATE parametresi için veri tipi olarak herhangi bir Bit tipini (Bool, Byte, Word veya DWord) veya bir Bit tipinin bir Array'ini seçebilirsiniz. • Yayınlanan STATE verilerinin ilk Bayt'ındaki Bit 0 bir özet Bit'idir. DOĞRU olarak ayarlandıysa, başka verilerin mevcut olduğunu belirtir. • STATE parametresi tarafından yayınlanan veriler, bir Bit pozisyonuyla bir modül pozisyonu arasındaki 1'e 1 eşleşmeyi gösterir. Bu yuva adresleme, PROFIBUS ve PROFINET için DOĞRU'dur. Örnek: Bir kafa modülü, güç modülü, bir çift G/Ç modülü içeren bir ET 200SP'de ilk Bayt'taki Bit 1 kafa modülüyle, Bit 2 güç modülüyle ve Bit 3 ve 4 G/Ç modülleriyle eşleşir.

[1] Maksimum 128 Bit atanabilir. Gereken Bit sayısı, G/Ç modülünüzün kullanımına bağlıdır.

Tablo 9-169 Koşul kodları

RET_VAL (W#16#...)	Açıklama
0	Hata yok
8091	LADDR tarafından belirtilen modül mevcut değil.
8092	LADDR tarafından belirtilen modül, hiçbir G/Ç cihazını adreslememektedir.
8093	STATE parametresi için geçersiz veri tipi: Geçerli veri tipleri (Bool, Byte, Word veya DWord)'dür veya (Bool, Byte, Word veya DWord) öğelerinden oluşan bir Array'dir.
80Bx	ModuleStates yordamı, CPU tarafından bu LADDR parametresi için desteklenmez
8452	Tam erişim verileri, atanan STATE parametresi için çok büyük. STATE tamponu bir kısmi sonuç içeriyor.

9.7.7.1 ModuleStates konfigürasyonuyla ilgili örnekler

PROFIBUS'taki örnek

PROFIBUS'taki örnek şu bileşenlerden oluşur:

- Adları "DPSlave_10" ile "DPSlave_25" arasında olan 16 PROFIBUS cihazı
- 16 PROFIBUS cihazının her biri, 10 ile 25 arasındaki PROFIBUS adreslerinden birini kullanır.
- Her Slave cihaz için çok sayıda G/Ç modülü yapılandırılmış.
- Örnekte, bir kafa modülü, bir güç modülü ve iki G/Ç modülü içeren "DPSlave_12" PROFIBUS Slave'inin LADDR parametresi kullanılmaktadır.
- Yayınlanan STATE parametre bilgilerinin ilk dört Bayt'ı görüntülenir.

MODE	Örnek 1: Hatasız normal işletim	Örnek 2: Modülü çekilmiş DPSlave_12 PROFIBUS Slave cihazı	Örnek 3: DPSlave_12 PROFIBUS Slave cihazı ayrıldı
1: Modül konfigürasyonu aktif	0x1F00_0000	0x1F00_0000	0x1F00_0000
2: Modül arızalı	0x0000_0000	0x0900_0000	0x1F00_0000

MODE	Örnek 1: Hatasız normal işletim	Örnek 2: Modülü çekilmiş DPSlave_12 PROFIBUS Slave cihazı	Örnek 3: DPSlave_12 PROFIBUS Slave cihazı ayrıldı
3: Modül devre dışı bırakıldı	0x0000_0000	0x0000_0000	0x0000_0000
4: Modül mevcut	0x1F00_0000	0x1700_0000	0x0000_0000
5: Modülde sorun var	0x0000_0000	0x0900_0000	0x1F00_0000

Aşağıdaki dört tabloda, analiz edilmiş dört veri Bayt'ının ikili çözümlemesi gösterilmektedir:

Tablo 9-170 Örnek 1: Hata yok: MODE 1 (modül konfigürasyonu aktif) için 0x1F00_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x1F	Bit 7 0001-1111 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x00	Bit 15 0000-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

1 (Bit 1) ile 4 (Bit 4) arasındaki yuvalar, modüller içerir. Yuva 5 (Bit 5) ve sonraki yuvalar modül içermez. MODE 4'ün (Modül mevcut) verileri, MODE 1 (Modül konfigürasyonu aktif) ile eşleşir, yani yapılandırılmış modüller mevcut modüllere karşılık gelmez.

Tablo 9-171 Örnek 2: Bir modül, "DPSlave_12" PROFIBUS Slave cihazından çekildi. MODE 2 (modül arızalı) için 0x0900_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x09	Bit 7 0000-**1**001 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x00	Bit 15 0000-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Sadece modül 3 (Bit 3) arızalı olarak işaretlendi. Diğer tüm modüller çalışır durumdadır.

Tablo 9-172 Örnek 2 (devam): Bir modül, "DPSlave_12" PROFIBUS Slave cihazından çekildi. MODE 4 (modül arızalı) için 0x1700_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x17	Bit 7 0001-**0**111 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x00	Bit 15 0000-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Modül 3 (Bit 3) eksik olarak görüntülenir. Modül 1, 2 ve 4 (Bit 1, 2 ve 4) mevcut olarak gösterilir.

Tablo 9-173 Örnek 3: "DPSlave_12" PROFIBUS Slave cihazı, PROFIBUS ağından ayrıldı (kablo çıkartıldı veya gerilim kesildi). MODE 2 (modül arızalı) için 0x1F00_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x1F	Bit 7 0001-1111 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x00	Bit 15 0000-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

1 ile 4 arasındaki modüllerin (Bit 1 ila 4) tümü, cihaz eksik olduğu için arızalı olarak işaretlendi.
MODE 5 (Modülde sorun var), MODE 2 (Modül arızalı) ile aynı bilgileri görüntüler.

PROFINET'teki örnek

PROFINET'teki örnek şu bileşenlerden oluşur:

- "et200s_1" ile "et200s_16" arasında adlara sahip 16 PROFINET Slave cihazı.
- 16 PROFINET cihazının her biri, 1 ile 16 arasındaki PROFINET cihaz numaralarından birini kullanır.
- Her Slave cihaz için çok sayıda G/Ç modülü yapılandırılmış.
- Örnekte; bir kafa modülü, bir güç modülü ve 18 G/Ç modülü içeren "et200s_1" PROFINET Slave'i kullanılmaktadır.
- Yayınlanan STATE parametre bilgilerinin ilk dört Bayt'ı görüntülenir.

MODE	Örnek 1: Hatasız normal işletim	Örnek 2: Modülü çekilmiş et200s_1 PROFINET Slave	Örnek 3: et200s_1 PROFINET Slave ayrıldı
1: Modül konfigürasyonu aktif	0xFFFF_1F00	0xFFFF_1F00	0xFFFF_1F00
2: Modül arızalı	0x0000_0000	0x0180_0000	0xFFFF_1F00
3: Modül devre dışı bırakıldı	0x0000_0000	0x0000_0000	0x0000_0000
4: Modül mevcut	0xFFFF_1F00	0xFF7F_1F00	0x0000_0000
5: Modülde sorun var	0x0000_0000	0x0180_0000	0xFFFF_1F00

Aşağıdaki dört tabloda, analiz edilmiş dört veri Bayt'ının ikili çözümlemesi gösterilmektedir:

Tablo 9-174 Örnek 1: Hata yok: MODE 1 (modül konfigürasyonu aktif) için 0xFFFF_1F00 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFF	Bit 15 1111-1111 Bit 8	
Bayt 3 0x1F	Bit 23 0001-1111 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

1 (Bit 1) ile 20 (Bit 20) arasındaki yuvalar, modüller içermektedir. Yuva 21 (Bit 21) ve sonraki yuvalar modül içermez.
MODE 4'ün (Modül mevcut) verileri, MODE 1 (Modül konfigürasyonu aktif) ile eşleşir, yani yapılandırılmış modüller mevcut modüllere karşılık gelmez.

Tablo 9-175 Örnek 2: Bir modül, "et200s_1" PROFINET Slave cihazından çekildi. MODE 2 (modül arızalı) için 0x0180_0000 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x80	Bit 15 **1**000-0000 Bit 8	
Bayt 3 0x00	Bit 23 0000-0000 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Sadece modül 15 (Bit 15) arızalı olarak işaretlendi. Diğer tüm modüller çalışır durumdadır.

Tablo 9-176 Örnek 2 (devam): Bir modül, "et200s_1" PROFIBUS Slave cihazından çekildi. MODE 4 (modül arızalı) için 0xFF7F_1F00 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0x7F	Bit 15 **0**111-1111 Bit 8	
Bayt 3 0x1F	Bit 23 0001-1111 Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

Modül 15 (Bit 15) eksik olarak görüntülenir. 1 ile 14 arasındaki modüller (Bit 1 ila 14) ve 16 ile 20 arasındaki modüller (Bit 16 ila 20) eksik olarak görüntülenir.

Tablo 9-177 Örnek 3: "et200s_1" PROFINET Slave cihazı, PROFINET ağından ayrıldı (kablo çıkartıldı veya gerilim kesildi). MODE 2 (modül arızalı) için 0xFFFF_1F00 değeri yayınlanır.

Değerle birlikte Bayt	Değerle birlikte Bit örüntüsü	Notlar
Bayt 1 0xFF	Bit 7 **1111**-**1111** Bit 0	Bit 0 doğrudur; veriler mevcuttur.
Bayt 2 0xFF	Bit 15 **1111**-**1111** Bit 8	
Bayt 3 0x1F	Bit 23 000**1**-**1111** Bit 16	
Bayt 4 0x00	Bit 31 0000-0000 Bit 24	

1 ile 20 arasındaki modüllerin (Bit 1 ila 20) tümü, cihaz eksik olduğu için arızalı olarak işaretlendi. MODE 5 (Modülde sorun var), MODE 2 (Modül arızalı) ile aynı bilgileri görüntüler.

9.7.8 GET_DIAG (teşhis bilgilerini oku)

Açıklama

GET_DIAG yordamıyla, bir donanım cihazının teşhis bilgilerini okuyabilirsiniz. Donanım cihazı, LADDR parametresi aracılığıyla seçilir. MODE parametresiyle, hangi teşhis bilgilerinin okunması gerektiğini seçebilirsiniz.

Tablo 9-178 GET_DIAG yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := GET_DIAG(** **mode:=_uint_in_,** **laddr:=_word_in_,** **cnt_diag=>_uint_out_,** **diag:=_variant_inout_,** **detail:=_variant_inout_);**	Teşhis bilgilerini atanmış bir donanım cihazından okur.

Parametre

Aşağıdaki tabloda, GET_DIAG yordamının parametreleri gösterilmektedir:

Tablo 9-179 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
MODE	IN	UInt	MODE parametresiyle, hangi teşhis verilerinin yayınlanmasını gerektiğini belirtebilirsiniz.
LADDR	IN	HW_ANY (Word)	Cihazın donanım kodu
RET_VAL	OUT	Int	Yordamın durumu
CNT_DIAG	OUT	UInt	Yayınlana teşhis ayrıntılarının sayısı
DIAG	InOut	Variant	İşaretçiyi, seçilen modun teşhis bilgilerini kaydetmek için kullanılan veri alanının üzerine götürün
DETAILS	InOut	Variant	İşaretçiyi, seçilen modla eşleşen teşhis bilgilerini kaydetmek için kullanılan veri alanının üzerine götürün

MODE parametresi

MODE parametresinin değerine bağlı olarak DIAG, CNT_DIAG ve DETAILS çıkış parametrelerinde farklı teşhis verileri yayınlanır:

Tablo 9-180 MODE parametresi

MODE	Açıklama	DIAG	CNT_DIAG	DETAILS
0	Bir modül için desteklenen tüm teşhis bilgilerinin DWord olarak yayınlanması, bu sırada Bit X=1, Mod X'in desteklendiğini belirtir.	DWord olarak desteklenen modların Bit karakter dizisi, burada Bit X=1 mod X'in desteklendiğini belirtir. MODE parametresi = 0 ise, S7-1200 CPU'su, LADDR parametresini göz ardı eder.	0	-
1	Adreslenen donanım nesnesinin ilgili durumunun yayınlanması.	Teşhis durumu: DIS yapısıyla uyumlu olarak yayınlama. (Not: Aşağıdaki "DIS yapısı" ile ilgili bilgilere ve alt bölümün sonundaki GET_DIAG yordamıyla ilgili örneğe de dikkat edin.)	0	-
2	Adreslenen donanım nesnesinin tüm alt seviyedeki modüllerinin durumunun yayınlanması.	Teşhis verilerinin, DNN yapısıyla uyumlu olarak yayınlanması. (Not: Aşağıdaki "DNN yapısı" ile ilgili bilgilere ve alt bölümün sonundaki GET_DIAG yordamıyla ilgili örneğe de dikkat edin.)	0	-

DIS yapısı

Bir MODE parametresi = 1 iken teşhis bilgileri, DIS yapısıyla uyumlu olarak yayınlanır. Aşağıdaki tabloda, münferit parametre değerlerinin anlamı gösterilmiştir:

Tablo 9-181 Teşhis bilgisi kaynağının (DIS, Diagnostic Information Source) yapısı

Parametre	Veri tipi	Değer	Açıklama
MaintenanceState	DWord	Enum	
		0	Bakım gerekli değil
		1	Modül veya cihaz devre dışı bırakıldı.
		2	-
		3	-
		4	-
		5	Bakım gerekli
		6	Bakım talep edildi
		7	Hata
		8	Alt seviyedeki modülde durum bilinmiyor/hata mevcut
		9	-
		10	Girişler/çıkışlar mevcut değil.

Parametre	Veri tipi	Değer	Açıklama
Componentstate Detail	DWord	Bit Array'i	Modülün alt modüllerinin durumu: • Bit 0 ila 15: Modülün durum mesajı • Bit 16 ila 31: CPU'nun durum mesajı
		0 ila 2 (Enum)	Daha fazla bilgi: • Bit 0: Daha fazla bilgi yok • Bit 1: Aktarıma izin verilmiyor
		3	Bit 3 = 1: Teşhis için kullanılan tanımlayıcıları en az bir kanal destekliyor.
		4	Bit 4 = 1: En az bir kanal veya bir bileşenin bakımı yapılmalıdır
		5	Bit 5 = 1: En az bir kanal veya bir bileşen için bakım talep edildi
		6	Bit 6 = 1: En az bir kanalda veya bir bileşende hata mevcut
		7 ila 10	Rezerve (her zaman = 0)
		11 ila 14	Bit 11 = 1: PNIO - Alt modül doğru Bit 12 = 1: PNIO - Yedek modül Bit 13 = 1: PNIO - yanlış modül Bit 14 = 1: PNIO - Modül ayrıldı
		15	Rezerve (her zaman = 0)
		16 ila 31	Modüller için CPU tarafından oluşturulan durum bilgileri: Bit 16 = 1: Modül devre dışı bırakıldı Bit 17 = 1: CiR işlemi aktif Bit 18 = 1: Giriş mevcut değil Bit 19 = 1: Çıkış mevcut değil Bit 20 = 1: Teşhis tamponunda taşma Bit 21 = 1: Teşhis mevcut değil Bit 22 - 31: Rezerve (her zaman 0)
OwnState	Uint16	Enum	OwnState parametresinin değeri, modülün bakım durumunu tanımlar.
		0	Arıza yok
		1	Modül veya cihaz devre dışı bırakıldı.
		2	Bakım gerekli
		3	Bakım talep edildi
		4	Hata
		5	Modüle veya cihaza CPU tarafından erişilemiyor (bir CPU'nun altındaki modüller ve cihazlar için geçerlidir).
		6	Girişler/çıkışlar mevcut değil.
		7	-
IO State	Uint16	Bit Array'i	Modülün G/Ç durumu
		0	Bit 0 = 1: Bakım gerekli değil
		1	Bit 1 = 1: Modül veya cihaz devre dışı bırakıldı.
		2	Bit 2 = 1: Bakım gerekli
		3	Bit 3 = 1: Bakım talep edildi
		4	Bit 4 = 1: Hata
		5	Bit 5 = 1: Modüle veya cihaza CPU tarafından erişilemiyor (bir CPU'nun altındaki modüller ve cihazlar için geçerlidir).
		6	Tanımlayıcı; Bit 0, 2 veya 3 ayarlandıysa Bit 7 = 1

Parametre	Veri tipi	Değer	Açıklama
IO State	Uint16	7	Girişler/çıkışlar mevcut değil.
		8 ila 15	Rezerve (her zaman = 0)
OperatingState	UInt16	Enum	
		0	-
		1	STOP işletim durumunda/Firmware güncellemesinde
		2	STOP işletim durumunda/Belleğin tam silinmesi
		3	STOP/otomatik start işletim durumunda
		4	STOP işletim durumunda
		5	Belleğin tam silinmesi
		6	START işletim durumunda
		7	RUN işletim durumunda
		8	-
		9	HOLD işletim durumunda
		10	-
		11	-
		12	Modül arızalı
		13	-
		14	Gerilim yok
		15	CiR
		16	STOP işletim durumunda/DIS hariç
		17	Giriş
		18	
		19	
		20	

DNN yapısı

Bir MODE parametresi = 2 iken teşhis bilgisi ayrıntıları, DNN yapısıyla uyumlu olarak yayınlanır. Aşağıdaki tabloda, münferit parametre değerlerinin anlamı gösterilmiştir:

Tablo 9-182 Teşhis gezinti düğümünün (DNN, Diagnostic Navigation Node) yapısı

Parametre	Veri tipi	Değer	Açıklama
SubordinateState	UINT	Enum	Alt seviyedeki modülün durumu (bkz. DIS yapısının OwnState parametresi)
SubordinateIOState	WORD	Bitarray	Alt seviyedeki modülün girişlerinin ve çıkışlarının durumu (bkz. DIS yapısının IO State parametresi)
DNNmode	WORD	Bitarray	• Bit 0 = 0: Teşhis etkinleştirildi • Bit 0 = 1: Teşhis devre dışı bırakıldı • Bit 1 ila 15: Rezerve

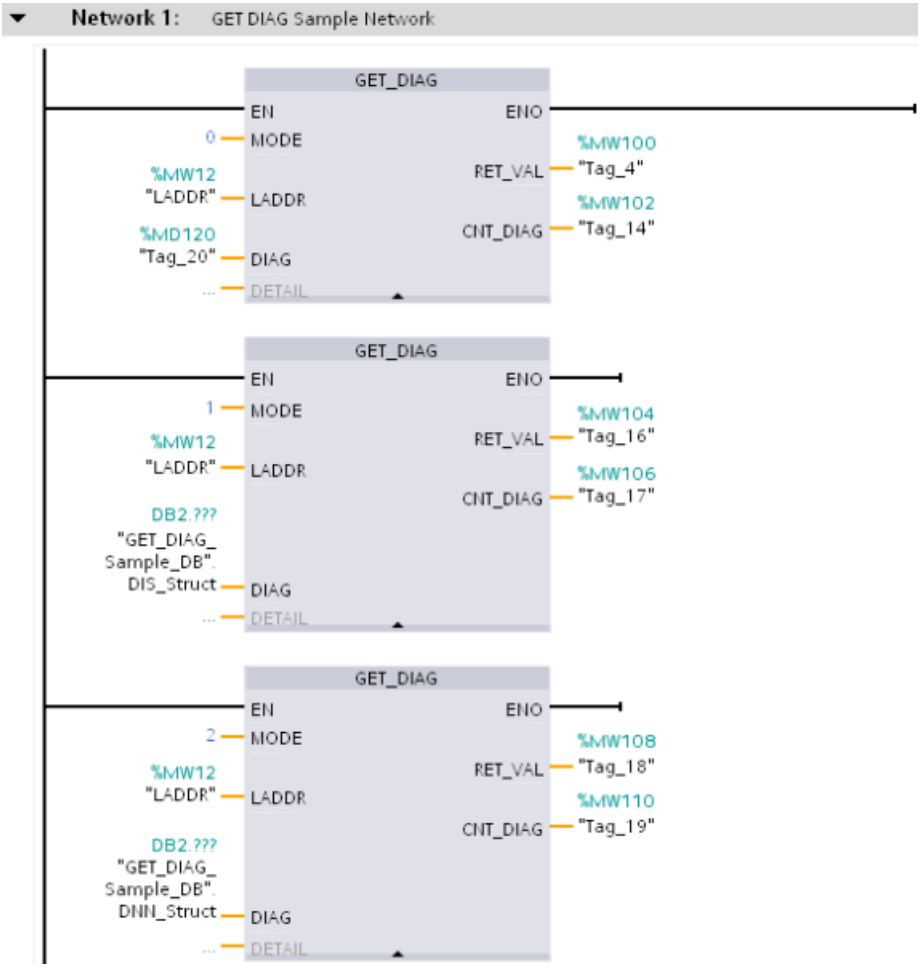
RET_VAL parametresi

Tablo 9-183 RET_VAL parametresinin hata kodları

Hata kodu (W#16#...)	Açıklama
0	Hata yok
8080	MODE parametresindeki değer desteklenmiyor.
8081	DIAG parametresindeki tip, seçilen modülle desteklenmiyor (MODE parametresi).
8082	DETAILS parametresindeki tip, seçilen modülle desteklenmiyor (MODE parametresi).
8090	LADDR mevcut değil.
8091	CHANNEL parametresindeki seçilen kanal mevcut değil.
80C1	Paralel yürütme için yeterli kaynak yok

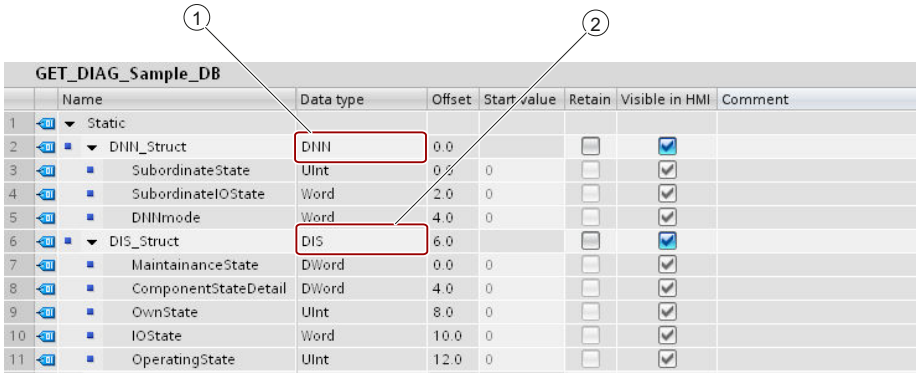
Örnek

Aşağıdaki LAD ağı ve aşağıdaki DB, üç işletim türünü üç modla nasıl kullanacağınızı göstermektedir:



DNN ve DIS yapılarını içeren örnek veri bloğu

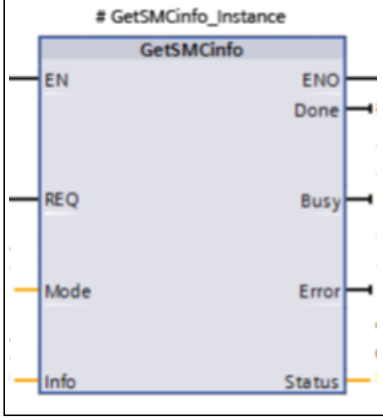
DB'de veri tipi DNN ve DIS'i girmeniz gerekir. Bunlar, veri tipleri açılır listesinde görüntülenmez.



9.7.9 GetSMCInfo (memory card ile ilgili bilgileri oku)

GetSMCInfo yordamı, takılı durumdaki SIMATIC memory card ile ilgili bilgileri çağırır. Okunması gereken bilgi, "Mode" parametresiyle belirtilir.

Tablo 9-184 GetSMCinfo yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := GetSMCinfo(Mode:=_uint_in_, Info:=_variant_inout_);**	"GetSMCinfo" yordamıyla, takılı durumdaki memory card ile ilgili bilgileri okuyabilirsiniz. Yordam, hiçbir memory card takılı değilse W#16#8081 hata kodunu yayınlar. Mode parametresi aracılığıyla, okunması gereken bilgileri seçebilirsiniz.

Parametre

Aşağıdaki tabloda, "GetSMCinfo" yordamının parametreleri gösterilmektedir:

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Request kumanda parametresi REQ = "1" ile bilgilerin okunmasını etkinleştirir.
Mode	IN	UInt	Okumak istediğiniz SIMATIC memory card ile ilgili bilgileri, Mode parametresinin yardımıyla seçebilirsiniz: • 0: KiB (1 KiB = 1024 Bayt) cinsinden bellek boyutu • 1: KiB'de dolu bellek • 2: Bakım bilgileri: Kullanım ömrünün daha önce kullanılan % cinsinden oranı Not: – S7-1200 şimdi bakım bilgilerini sağlar. S7-PLCSIM'de Info'ya her zaman 0x00 değeri girilir (bakım bilgileri mevcut değil). – Bilgiler, SIMATIC memory card ya da CPU'nuzun mevcut durumda kurulu olan Firmware'i tarafından desteklenmiyorsa, Info'ya 0xFF değeri girilir. • 3: Kullanım ömrünün % cinsinden belirlenen eşik değeri aşılırsa, PLC, teşhis tamponunda bir giriş oluşturur ve bakım LED'ini etkinleştirir. Not: – Teşhis alarmlarının üretilmesini devre dışı bıraktıysanız, Info'ya 0xFF değeri girilir. – S7-PLCSIM'de her zaman Info'ya 0xFF değeri girilir. • 10 veya 20: Mode 0'a karşılık gelir, fakat S7-1500-R/H CPU'ları için rezerve edilmiştir. • 11 veya 21: Mode 1'e karşılık gelir, fakat S7-1500-R/H CPU'ları için rezerve edilmiştir. • 12 veya 22: Mode 2'ye karşılık gelir, fakat S7-1500-R/H CPU'ları için rezerve edilmiştir. • 13 veya 23: Mode 3'e karşılık gelir, fakat S7-1500-R/H CPU'ları için rezerve edilmiştir.
Done	OUT	Bool	1: Yordam başarıyla yürütüldü. Okunan bilgiler, Info parametresine aktarılır.
Busy	OUT	Bool	Durum parametresi • 0: Yordamın yürütülmesi tamamlandı • 1: Yordamın yürütülmesi henüz tamamlanmadı
Error	OUT	Bool	Durum parametresi • 0: Hata yok. • 1: Yordamın yürütülmesi sırasında bir hata oluştu Ayrıntılı bilgiler, Status parametresinde yayınlanır
Status	OUT	Word	Hata kodu
Info	INOUT	UDInt	Okunan bilgiler için kullanılan tampon

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

NOT

I&M 0 verilerinin belirlenmesi

Memory card'ın I&M 0 verileri, GetSMCinfo yordamıyla belirlenemez. Bunun için "Get_IM_Data (Sayfa 406)" yordamını kullanın.

Parametre durumu

Hata kodu* (W#16#...)	Anlamı
0	Hata yok
7000	Düzenlenen görev yok.
7001	İlk çağrı: Görevi başlatın (İşletim = 1, Bitti = 0).
7002	Ara çağrı: Görev zaten aktif (İşletim = 1, Bitti = 0).
8080	"Mode" parametresinin geçersiz değeri
8081	Hiçbir SIMATIC memory card takılı değil
8092	Örneğin "GetSMCinfo" CPU tarafından desteklenmediği için hiçbir veri mevcut değil
80C3	"GetSMCinfo" yordamının eşzamanlı maksimum çağrı sayısına ulaşıldı.

* Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenir. Görüntüleme formatlarının değiştirilmesiyle ilgili bilgiler için, bkz. "Ayrıca bkz".

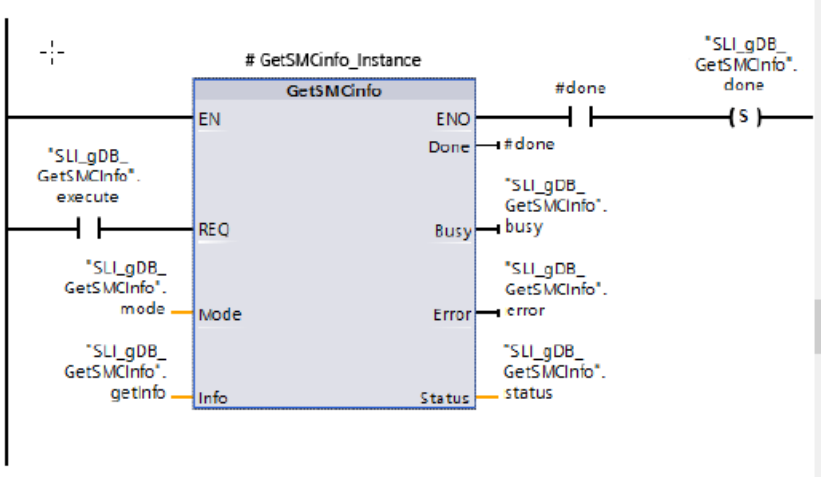
Örnek: Kullanılan SIMATIC memory card'ın bellek boyutunun belirlenmesi

Aşağıdaki örnekte, kullanılan SIMATIC memory card'ın bellek boyutunu belirleyebilirsiniz. Verileri global bir veri bloğuna kaydetmek için aşağıdaki değişkenleri oluşturun:

SLI_gDB_GetSMCinfo

	Name	Data type	Start value
1	Static		
2	execute	Bool	false
3	mode	UInt	0
4	getInfo	UDInt	0
5	done	Bool	false
6	busy	Bool	false
7	error	Bool	false
8	status	Word	16#0

Bir FB oluşturun. FB'nin durum alanında, Bool veri tipiyle yerel bir "#done" değişkeni oluşturun. "GetSMCinfo" yordamının parametrelerini aşağıdaki şekilde ilişkilendirin:



"GetSMCinfo" yordamı sadece, REQ ("execute") giriş parametresinin "DOĞRU" sinyal durumunu döndürmesi durumunda yürütülür. SIMATIC memory card'ı okumak için

kullanılması gereken mod, MODE ("mode") giriş parametresine kaydedilmiştir. Aşağıdaki örnekte, SIMATIC memory card'ın bellek boyutu, "mode" için belirlenen "0" değerine uygun olarak okunur ve INFO ("getInfo") parametresinde KB cinsinden görüntülenir. GetSMCinfo yordamının başarı durumu, DONE ("#done") çıkış parametresinde görüntülenir ve "done" değişkenine kaydedilir.
STATUS ("status") ve ERROR ("error") çıkış parametreleri, örnekteki işleme sürecinin hatasız bir şekilde tamamlandığını belirtmektedir.

SLI_gDB_GetSMCinfo

	Name	Data type	Start value	Monitor value
1	Static			
2	execute	Bool	false	TRUE
3	mode	UInt	0	0
4	getInfo	UDInt	0	2025008
5	done	Bool	false	TRUE
6	busy	Bool	false	FALSE
7	error	Bool	false	FALSE
8	status	Word	16#0	16#0000

9.7.10 Desantral çevrebirim için kullanılan teşhis olayları

NOT

Bir PROFIBUS IO sisteminde CPU; donanım uyumluluğu, kabul edilebilir yedek modüllere (Sayfa 144-145) izin verilecek ve en azından bir modül eksik olacak veya yapılandırılmış modül için kabul edilebilir bir yedek modül söz konusu olmayacak şekilde ayarlanmadıysa, bir yükleme veya kapatma ve tekrar çalıştırma işleminden sonra RUN işletim durumuna geçer.

CPU, aşağıdaki tabloda gördüğünüz gibi, desantral G/Ç sisteminin bileşenleri için yapılandırılabilen teşhisi desteklemektedir. Bu hataların her biri, teşhis tamponunda bir protokol girişi üretir

Tablo 9-185 PROFINET ve PROFIBUS'ta teşhis olaylarının işlenmesi

Hata tipi	İstasyon için teşhis bilgisi var mı?	Teşhis tamponunda giriş var mı?	CPU işletim durumu
Teşhis hatası	Evet	Evet	RUN işletim durumunda kalır
Modül taşıyıcısında veya istas-yonda hata	Evet	Evet	RUN işletim durumunda kalır
Çevrebirim erişim hatası [1]	Hayır	Evet	RUN işletim durumunda kalır
Çevrebirim erişim hatası [2]	Hayır	Evet	RUN işletim durumunda kalır
Çıkartma/takma olayı	Evet	Evet	RUN işletim durumunda kalır

[1] Bir çevrebirim erişim hatasının örnek nedeni: Bir modül çıkartıldı.

[2] Bir çevrebirim erişim hatasının örnek nedeni: İletişim kurmayan bir alt modülle aperiyodik iletişim.

GET_DIAG yordamı (Sayfa 430) aracılığıyla her istasyon için teşhis bilgilerini çağırabilirsiniz. Bu şekilde, cihazda oluşan hataları programla giderebilir ve istenirse, CPU'yu STOP işletim durumuna ayarlayabilirsiniz. Bu yöntem için, durum bilgilerinin okunması gereken donanım cihazını belirtmeniz gerekir.
GET_DIAG yordamı, tüm istasyonun durumunu çağırmak için istasyonun "L adresini" (LADDR) kullanır. Bu L adresini, istasyonun tüm modül taşıyıcısını seçtiğiniz (tüm gri alan) cihaz

konfigürasyonunun ağ görünümünde bulabilirsiniz. L adresi, istasyonun "Özellikler" sekmesinde görüntülenir. Her münferit modül için LADDR parametresini modülün özelliklerinde (cihaz konfigürasyonunda) veya CPU'nun standart değişken tablosunda bulabilirsiniz.

9.8 Darbe

9.8.1 CTRL_PWM (darbe süresi modülasyonu)

Tablo 9-186 CTRL_PWM (darbe süresi modülasyonu) yordamı

LAD/FBD	SCL	Açıklama
	**"CTRL_PWM_DB"(PWM:=_uint_in_, ENABLE:=_bool_in_, BUSY=>_bool_out_, STATUS=>_word_out_);**	Değişken bağıl devreye alma süresine sahip bir sabit çevrim süresi sunar. PWM çıkışı, başlatma işleminden sonra, belirtilen frekansla (çevrim süresi) sürekli olarak çalışır. Darbe süresi, istenen kumandayı sağlamak için ihtiyaca göre değiştirilir.

[1] Yordamı eklerseniz, STEP 7, ilgili DB'yi oluşturmak için "Çağırma seçenekleri" iletişim kutusunu görüntüler.

[2] SCL örneğinde "CTRL_PWM_DB", örnek DB'nin adıdır.

Tablo 9-187 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
PWM	IN	HW_PWM (Word)	PWM kodu: Etkinleştirilen darbe jeneratörlerinin adları, "Sabitler" değişken tablosundaki değişkenler haline gelir ve PWM parametreleri olarak kullanılabilir. (Standart değer: 0)
ENABLE	IN	Bool	1 = Darbe jeneratörünü başlatma 0 = Darbe jeneratörünü durdurma
BUSY	OUT	Bool	Fonksiyon meşgul (standart değer: 0)
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

CTRL_PWM yordamı, parametre bilgilerini DB'ye kaydeder. Veri bloğunun parametreleri, kullanıcı tarafından ayrı olarak değiştirilmez, CTRL_PWM yordamı tarafından kontrol edilir.

İstediğiniz darbe jeneratörünü, PWM parametresi için kullanılan değişken adı aracılığıyla belirtin.

EN girişi DOĞRU ise, PWM_CTRL yordamı, belirtilen PWM'yi ENABLE girişindeki değer aracılığıyla başlatır veya durdurur. Darbe süresi, çıkış sözcüğünün atanmış adresinin değeri tarafından belirtilir.

CPU talebi işlediği için, CTRL_PWM yordamının yürütülmesi sırasında BUSY parametresi her zaman YANLIŞ ifadesini bildirir. Bir hata algılanırsa, ENO YANLIŞ olarak ayarlanır ve STATUS parametresi bir hata kodu içerir.

Darbe süresi, CPU'nun RUN işletim durumuna ilk kez geçişinde, cihaz konfigürasyonunda girilen başlangıç değerine ayarlanır. Darbe süresini değiştirmek için, istediğiniz değerleri, cihaz konfigürasyonunda belirtilen çıkış sözcüğü adresine ("Çıkış adresleri"/"Başlangıç adresi:") yazın. İstediğiniz darbe süresi değerini ilgili çıkış sözcüğüne yazmak için Aktar, Dönüştür, Aritmetik veya PID gibi bir yordamı kullanın. Çıkış sözcüğünün değeri için kullanılan geçerli aralığa uymanız gerekir (yüzde, binde bir, on binde bir veya S7 analog formatı).

NOT

PWM ve PTO'ya atanan dijital G/Ç'lere Force fonksiyonu uygulanamaz.

Darbe süresi modülasyonu (PWM) ve darbe dizisi (PTO) tarafından kullanılan dijital G/Ç'ler cihaz konfigürasyonu sırasında atanır. Bu fonksiyonlara dijital G/Ç'ler atanırsa, atanan G/Ç'lerin adreslerinin değerleri izleme tablosunda Force fonksiyonu aracılığıyla değiştirilemez.

Tablo 9-188 STATUS parametresinin değeri

STATUS	Açıklama
0	Hata yok
80A1	PWM kodu geçerli bir PWM'yi adreslemiyor.

9.8.2 CTRL_PTO (darbe dizisi)

PTO yordamı, belirtilen bir frekansta % 50'lik bir bağıl devreye alma süresiyle birlikte bir kare çıkış sağlar. CTRL_PTO yordamıyla frekansı, bir teknoloji nesnesi (TO) olmadan eksen veri bloğuna (DB) atayabilirsiniz.

Bu yordam, bir darbe jeneratörü gerektirir. Darbe jeneratörünü donanım konfigürasyonunda etkinleştirmeniz ve bir sinyal tipi seçmeniz gerekir. Daha fazla bilgiyi, "PWM veya PTO için bir darbe kanalının yapılandırılması" (Sayfa 445) başlığında bulabilirsiniz.

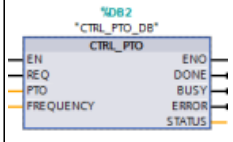
NOT

CTRL_PTO yordamı, yön seçimi için kullanılamaz.

Yön seçimi için, S7-1200 CPU'sundaki MC_MoveAbsolute gibi hareket kumandasının kaydırma fonksiyonlarını kullanabilirsiniz. Önce STEP7 programında bir teknoloji nesnesi oluşturmanız gerekir. Teknoloji nesnesini darbe jeneratörü olarak yapılandırın ve sinyal tipini seçin. Pozitif (+) veya negatif (-) değerler olarak hareket yordamının Direction veya Position girişi. Teknoloji nesnesi, bu değerleri yayınlar ve hareketin uygulanmasına olanak sağlar.

CTRL_PTO yordamına, gelişmiş yordamlar görev kartlarından erişebilirsiniz.

Tablo 9-189 CTRL_PTO (darbe dizisi) yordamı

LAD/FBD [1]	SCL [2]	Açıklama
	**"CTRL_PTO_DB"(REQ:=_bool_in_, PTO:=_uint_in_, FREQUENCY:=_udint_in_, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	PTO yordamı aracılığıyla, bir kare çıkışın frekansına (% 50'lik bir bağıl devreye alma süresiyle birlikte) kumanda edebilirsiniz.

[1] Yordamı eklerseniz, STEP 7, ilgili DB'yi oluşturmak için "Çağırma seçenekleri" iletişim kutusunu görüntüler.

[2] SCL örneğinde "CTRL_PTO_DB", örnek DB'nin adıdır.

Tablo 9-190 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
EN	IN	Bool	1 = Yordam etkinleştirildi 0 = Yordam devre dışı bırakıldı
REQ	IN	Bool	1 = PTO çıkış frekansını FREQUENCY girişinin değerine ayarlayın 0 = PTO'da değişiklik yok
PTO	IN	HW_PTO (Word)	PTO kodu: Darbe jeneratörünün donanım kodu: • Etkinleştirilen darbe jeneratörlerinin adları, "Sabitler" değişken tablosundaki değişkenler haline gelir ve PTO parametreleri olarak kullanılabilir. (Standart değer = 0.) • Donanım ID'sini, cihaz görünümündeki darbe jeneratörünün özelliklerinde bulabilirsiniz. Sistem sabitlerinde, darbe jeneratörlerinin donanım ID'leri de listelenir. (Standart değer = 0.)
FREQUENCY	IN	UDInt	PTO'nun istenen frekansı (Hz cinsinden). Bu değer ancak REQ = 1 olduğu takdirde (standart değer 0 Hz'tir) geçerlidir.
DONE	OUT	Bool	Fonksiyon hatasız bir şekilde tamamlandı (standart değer: 0)

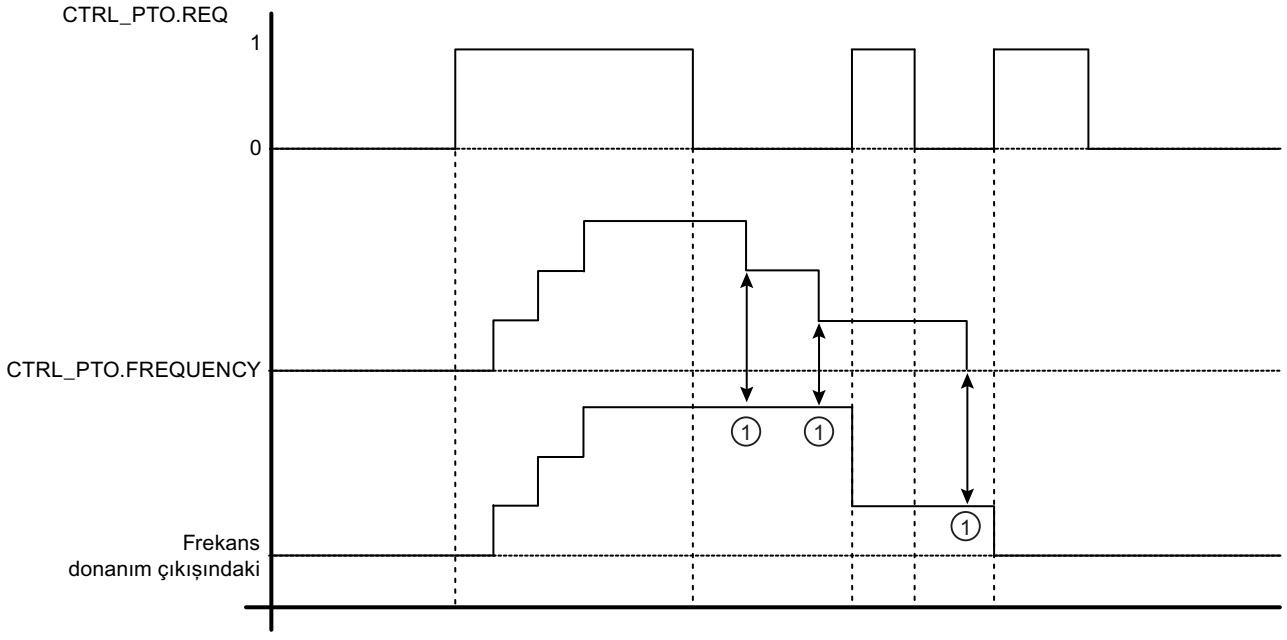
Parametre ve veri tipi		Veri tipi	Açıklama
BUSY	OUT	Bool	Fonksiyon meşgul (standart değer: 0)
ERROR	OUT	Word	Hata algılandı (standart değer: 0)
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

CTRL_PTO yordamı, parametre bilgilerini DB'ye kaydeder. Veri bloğunun parametreleri, kullanıcı tarafından ayrı olarak değiştirilmez, CTRL_PTO yordamı tarafından kontrol edilir.

İstediğiniz etkinleştirilmiş darbe jeneratörünü değişken adı veya PTO parametresindeki donanım kodu aracılığıyla belirtin.

EN girişi DOĞRU ise, CTRL_PTO yordamı, belirtilen PTO'yu başlatır veya durdurur. EN girişi YANLIŞ ise, CTRL_PTO yordamı yürütülmez ve PTO güncel durumunda kalır.

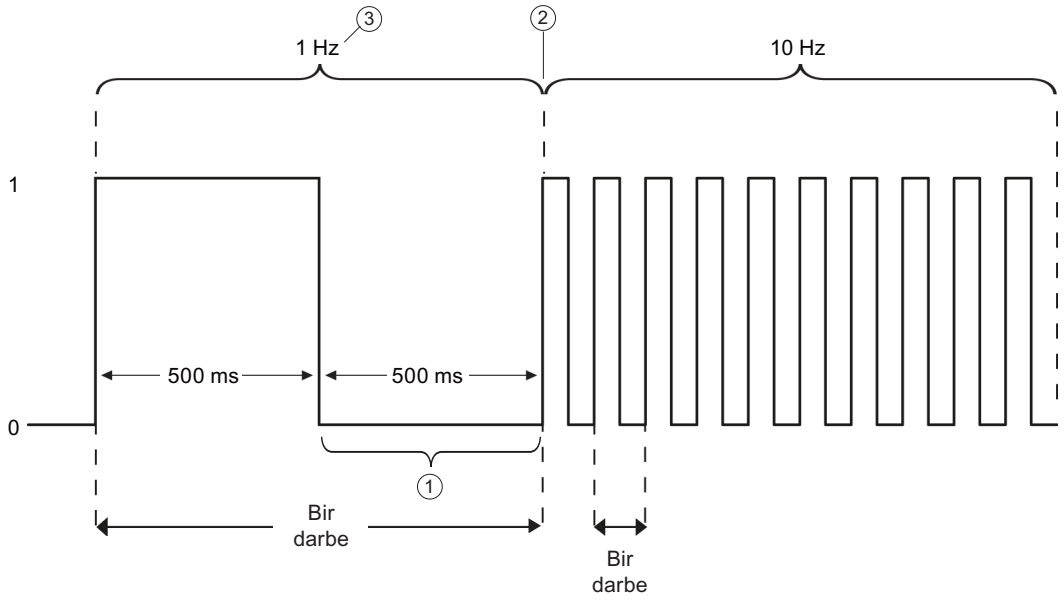
REQ girişini DOĞRU olarak ayarlarsanız, FREQUENCY değeri etkili hale gelir. REQ YANLIŞ ise, PTO'nun çıkış frekansı değiştirilmez ve PTO darbe yayınlamaya devam eder.



1 REQ = 0 olduğu sürece çıkış frekansında değişiklik yok

CTRL_PTO yordamı PTO'yu sadece başlattığı için, CTRL_PTO yordamı derhal sonlandırılır Bu nedenle BUSY çıkışı asla açılmaz. DONE çıkışı, hata oluşmadığı sürece açılır. Bir hata algılanırsa, ERROR parametresi DOĞRU olarak ayarlanır ve STATUS parametresi bir hata kodu içerir.

CTRL_PTO yordamı önceden belirtilen bir frekansla etkinleştirilirse, S7-1200, belirtilen bu frekansta bir darbe dizisi yayınlar. İstenen frekans her zaman değiştirilebilir. S7-1200, frekans değiştirildiğinde güncel darbeyi, frekans değerini istenen yeni değerle değiştirmeden önce sonlandırır. Örnek: İstenen frekans 1 Hz ise (bunun için uygulama amacıyla 1000 msn'ye ihtiyaç duyulur) ve frekans değeri 500 msn sonra 10 Hz olarak değiştirilirse, frekans değişikliği, 1000 msn'lik zaman diliminin sonunda uygulanır.



① Frekans, 500 msn sonra 10 Hz olarak değiştirilir.

② Frekans değerinin 10 Hz'lik yeni değer şeklinde değiştirilebilmesi için 1 Hz'lik darbe sonlandırılmalıdır.

③ 1 Hz, 1000 msn'ye karşılık gelir.

Darbe jeneratörünün donanım nesnesi aşağıdaki kısıtlamalara tabidir: Darbe jeneratörünü PTO olarak sadece bir yordam kullanabilir ve darbe jeneratörünün kullanımı, donanım konfigürasyonu editörü tarafından yönetilir. PTO'ya erişmeye çalışan diğer yordamlar bir hata yayınlar: "0x8090" (belirtilen donanım ID'sine sahip darbe jeneratörü kullanılır).

NOT

PWM ve PTO'ya atanan dijital G/Ç'lere Force fonksiyonu uygulanamaz.

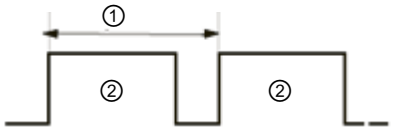
Darbe süresi modülasyonu (PWM) ve darbe dizisi (PTO) tarafından kullanılan dijital G/Ç'ler cihaz konfigürasyonu sırasında atanır. Bu fonksiyonlara dijital G/Ç'ler atanırsa, atanan G/Ç'lerin adreslerinin değerleri izleme tablosunda Force fonksiyonu aracılığıyla değiştirilemez.

Tablo 9-191 STATUS parametresindeki hata kodu değeri

Hata kodu (W#16#...)	Açıklama
0	Hata yok
0x8090	Belirtilen donanım ID'sine sahip darbe jeneratörü kullanılır.
0x8091	Frekans, aralık dışında. İstenen frekans, seçilen darbe çıkışının maksimum frekans değerini aşıyor.

Hata kodu (W#16#...)	Açıklama
0x80A1	PTO kodu (donanım ID'si) geçerli bir PTO'yu adreslemiyor.
0x80D0	Belirtilen donanım ID'sine sahip darbe jeneratörü etkinleştirilmedi. Darbe jeneratörünü, CPU özelliklerindeki "Darbe jeneratörleri (PTO/PWM)" bölümünde etkinleştirin.
0x80D1	Belirtilen donanım ID'sine sahip darbe jeneratöründe PTO seçimi yapılmadı. Donanım konfigürasyonunda PTO'yu seçin.

9.8.3 Darbe çıkışlarının çalışma şekli



① Çevrim süresi

② Darbe süresi

Darbe süresi; çevrim süresinin yüzde biri (0 ila 100), binde biri (0 ila 1000), on binde biri (0 ila 10000) veya S7 analog formatı olarak belirtilebilir.
Darbe süresi, 0 (darbe yok, her zaman kapalı) ve tam sapma (darbe yok, her zaman açık) arasında olabilir.

PWM çıkışı 0 ile tam sapma arasında olabileceği için, birçok açıdan bir analog çıkışa benzeyen bir dijital çıkış sunar. PWM çıkışı, örneğin bir motorun devir sayısına hareketsiz durumdan tam devir sayısı değerine kadar kumanda etmek için veya bir valfın konumuna kapalıdan tam açığa kadar kumanda etmek için kullanılabilir.

Frekansı, donanım konfigürasyonunda yapılandırabilirsiniz (Sayfa 445). Darbe süresini, kullanıcı programı üzerinden kontrol edebilirsiniz

Daha hızlı darbe çıkışlarını kontrol etmek için dört darbe jeneratörü mevcuttur: PWM ve darbe dizisi (PTO). PTO, hareket kumandası yordamlarıtarafından kullanılır. Her darbe jeneratörünü PWM veya PTO'ya atayabilir, ancak her ikisine de eş zamanlı olarak atayamazsınız.

Entegre CPU çıkışlarını veya opsiyonel bir sinyal kartının çıkışlarını kullanabilirsiniz. Aşağıdaki tabloda, çıkışların adresleri listelenmektedir (bu sırada, çıkışların standart konfigürasyonda olması şart koşulur). Çıkışların adreslerini değiştirdiyseniz, adresler, sizin atadığınız adreslere karşılık gelir. PTO opsiyonel olarak kanal başına iki çıkış kullanabilirken, PWM'nin sadece bir çıkışa ihtiyaç duyduğuna dikkat edin. Bir darbe fonksiyonu için bir çıkış gerekli değilse, bu çıkış başka amaçlarla kullanıma sunulur. G/Ç atamasını aşağıdaki tablodan öğrenebilirsiniz.

Aşağıdaki tabloda standart G/Ç atamaları gösterilmektedir. Ancak dört darbe jeneratörü, bir CPU'nun her entegre veya dijital SB çıkışı için yapılandırılabilir. Çeşitli çıkışlar, farklı gerilimleri ve hızları destekler. PWM/PTO çıkışlarını atarken bunu göz önünde bulundurun.

NOT

Darbe dizileri, kullanıcı programındaki başka işlemler tarafından kullanılamaz.

CPU'nun veya sinyal kartının çıkışlarını darbe jeneratörleri (PWM veya hareket kumandası PTO yordamları için) olarak yapılandırırsanız, çıkışların ilgili adresleri çıkışların belleğinden kaldırılır ve kullanıcı programınızda başka amaçlarla kullanılamaz. Kullanıcı programınız darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU bu değeri fiziksel çıkışa yazar.

NOT

PTO yön çıkışları, programdaki başka bir konumda kullanılmak üzere serbest bırakılabilir.

Her PTO için iki çıkış atanmalıdır: Darbe çıkışı olarak bir çıkış ve yön çıkışı olarak bir çıkış. Yön çıkışını değil, sadece darbe çıkışını da kullanabilirsiniz. Bu sayede, yön çıkışını, kullanıcı programınızda başka amaçlar için serbest bırakabilirsiniz.

Tablo 9-192 Darbe jeneratörlerinin standart çıkış atamaları[3]

Açıklama	Darbe	Yön
PTO1		
Entegre G/Ç	A0.0	A0.1
SB-G/Ç	A4.0	A4.1
PWM1		
Entegre çıkışlar	A0.0	-
SB çıkışları	A4.0	-
PTO2		
Entegre G/Ç	A0.2	A0.3
SB-G/Ç	A4.2	A4.3
PWM2		
Entegre çıkışlar	A0.2	-
SB çıkışları	A4.2	-
PTO3		
Entegre G/Ç	A0.4[1]	A0.5[1]
SB-G/Ç	A4.0	A4.1
PWM3		
Entegre çıkışlar	A0.4[1]	-
SB çıkışları	A4.1	-
PTO4		
Entegre G/Ç	A0.6[2]	A0.7[2]
SB-G/Ç	A4.2	A4.3
PWM4		
Entegre çıkışlar	A0.6[2]	-
SB çıkışları	A4.3	-

1 CPU 1211C, A0.4, A0.5, A0.6 ve A0.7 çıkışlarına sahip değildir. Bu nedenle bu çıkışlar CPU 1211C'de kullanılamaz.

2 CPU 1212C, A0.6 ve A0.7 çıkışlarına sahip değildir. Bu nedenle bu çıkışlar CPU 1212C'de kullanılamaz.

3 Bu tablo, CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C ve CPU 1217C'nin PTO/PWM fonksiyonları için geçerlidir.

9.8.4 PWM veya PTO için bir darbe kanalının yapılandırılması

PWM veya PTO işletimini hazırlamak için, önce cihaz konfigürasyonunda CPU'yu, darbe jeneratörünü (PTO/PWM) ve PWM1/PTO1 ila PWM4/PTO4'ü seçme yoluyla bir darbe kanalını yapılandırabilirsiniz. Darbe jeneratörünü etkinleştirin (opsiyon kutusu). Bir darbe jeneratörü etkinleştirilirse, bu belirli darbe jeneratörüne benzersiz bir standart ad atanır. Bu adı, "Ad:" düzenleme alanında değiştirebilirsiniz, ancak ad benzersiz olmalıdır. Etkinleştirilen darbe jeneratörlerinin adları, "Sabitler" değişken tablosundaki değişkenler haline gelir ve şu şekilde kullanılabilir:

- CTRL_PWM yordamının PWM parametreleri
- CTRL_PTO yordamının PTO parametreleri

"Yorum:" düzenleme alanına, bu belirli darbe jeneratörü hakkında bir yorum da girebilirsiniz.

Tablo 9-193 CPU çıkışı: Maksimum frekans (PTO) ve minimum çevrim süresi (PWM)

CPU	CPU çıkış kanalı:	PTO maksimum frekansı	PWM minimum çevrim süresi
1211C	Aa.0 - Aa.3	100 kHz	10 µs
1212C	Aa.0 - Aa.3	100 kHz	10 µs
	Aa.4, Aa.5	20 kHz	50 µsn
1214C ve 1215C	Aa.0 - Aa.3	100 kHz	10 µs
	Aa.4 - Ab.1	20 kHz	50 µsn
1217C	DAa.0 ila DAa.3 (.0+, .0- ila .3+, .3-)	1 MHz	1 µsn
	DAa.4 ila DAb.1	100 kHz	10 µs

Tablo 9-194 Sinyal kartı çıkışı: Maksimum frekans (PTO) ve minimum çevrim süresi (PWM)

Sinyal kartı (SB)	SB çıkış kanalı	PTO maksimum frekansı	PWM minimum çevrim süresi
SB 1222, 200 kHz	DAe.0 ila DAe.3	200 kHz	5 µsn
SB 1223, 200 kHz	DAe.0, DAe.1	200 kHz	5 µsn
SB 1223	DAe.0, DAe.1	20 kHz	50 µsn

NOT

CPU'nun ve sinyal kartının her çıkışına ait minimum çevrim süresi, yukarıdaki tablolarda belirtilmiştir. Ancak bir PWM darbe jeneratörünü, donanımın bu minimum çevrim süresinin altına inen bir çevrim süresiyle yapılandırırsanız TIA Portal tarafından bir uyarı verilmez. Bu, uygulamanızda sorunlara yol açabilir. Yani her zaman çevrim süresinin donanım limitlerinin içinde bulunduğundan emin olun.

NOT

Bir PWM sinyalinin darbe süresini belirliyorsanız, darbe süresinin gerçek zaman değeri (darbenin yüksek olduğu süre), zaman tabanı "milisaniye" olduğu sürece 1 milisaniyeden büyük veya ona eşit olmalıdır. Zaman tabanı "mikrosaniye" ise, darbe süresinin gerçek zaman değeri 1 mikrosaniyeden büyük veya ona eşit olmalıdır. Darbe süresi 1 "zaman tabanı" değerinden küçükse çıkış kapatılır.

Örnek: 10 mikrosaniyelik bir çevrim süresi ve 5 yüzde birlik bir darbe süresi, 0,5 mikrosaniyelik bir darbe süresi oluşturur. Bu değer 1 mikrosaniyenin altında olduğu için PWM sinyali kapalı olur.

Parametrelendirme

Parametrelendirme için tanımlanan aralıkta çıkış darbesinin parametreleri yapılandırılabilir. Aşağıdaki opsiyonlar, PWM'nin mi yoksa PTO'nun mu seçileceğine bağlı olarak kullanılabilir:

- Sinyal tipi: Darbe çıkışını PWM veya PTO olarak yapılandırın. PTO seçimiyle ilgili daha fazla bilgiyi, "Faz konumu" alt bölümünde bulabilirsiniz.
  - PWM
  - PTO (darbe A ve yön B)
  - PTO (darbe ileriye sayma A ve darbe geriye sayma B)
  - PTO (A/B faz kaydırmalı)
  - PTO (A/B faz kaydırmalı - dört kat)
- Zaman tabanı (sadece PWM için geçerlidir): Kullanılması gereken zaman birimlerinin seçimi:
  - Milisaniye
  - Mikrosaniye
- Darbe süresi formatı (sadece PWM için geçerlidir): Darbe süresinin çözünürlüğünün atanması (genişlik):
  - Yüzde bir (0 ila 100)
  - Binde bir (0 ila 1000)
  - On binde bir (0 ila 10000)
  - S7 analog formatı (0 ila 27648)
- Çevrim süresi (sadece PWM için geçerli): Bir çevrimin uygulanması için gerekli sürenin atanması (yüksek darbenin süresi artı düşük darbenin süresi çevrim süresine karşılık gelir). Çevrim süresini, "Çevrim süresinin çalışma zamanında değiştirilmesine izin ver" opsiyon kutusunu etkinleştirerek çalışma zamanı esnasında değiştirebilirsiniz. Daha fazla bilgiyi, bir sonraki "G/Ç adresleri" alt bölümünde bulabilirsiniz. Aralık 1 ile 16.777.215 zaman birimi arasındadır.
- Başlangıç darbe süresi (sadece PWM için geçerlidir): İlk darbenin darbe süresinin atanması. Bu değer, çalışma zamanı sırasında G/Ç adreslerinde yapılandırılan çıkış sözcüğü adresi aracılığıyla değiştirebilirsiniz. Aralık, darbe süresi formatını temel alır.
- Çevrim süresinin çalışma zamanında değiştirilmesine izin ver (sadece PWM için geçerli): Bu opsiyonu seçerseniz, programınız, programın çalışması sırasında PWM sinyalinin çevrim süresini değiştirebilir. Daha fazla bilgiyi, bir sonraki "G/Ç adresleri" alt bölümünde bulabilirsiniz.

NOT

Bir PWM sinyalinin darbe süresini ayarlarken, Ek A'da belirtildiği gibi çıkış kanalının anahtarlama gecikmesini dikkate alın. Çıkışta ölçülen gerçek darbe süresi, seçilen darbe süresinden büyük olabilir. Darbe süresinin artışı, küçük darbe süresinde ve daha yüksek frekanslarda daha belirgindir. Çıkışta ölçülen darbe süresinin gerekliliklerinize uygun olduğundan emin olun.

Darbe süresi değerinin belirlenmesi

"Darbe süresi", "başlangıç darbe süresinin" "çevrim süresi" ile çarpılmasıyla elde edilir. Bir "Zaman tabanı", bir "Darbe süresi formatı", bir "Çevrim süresi" ve "Başlangıç darbe süresi" seçerseniz, "Darbe süresi" toplamının kesirli değer olamayacağını dikkate almanız gerekir. Sonuçta elde ettiğiniz "darbe süresi" bir kesirli değerse, bir tamsayı oluşturmak için "başlangıç darbe sürenizi" uyarlamanız veya "zaman tabanınızı" değiştirmeniz gerekir.

Burada iki örnek verilmiştir:

- Örnek 1: Aşağıdaki değerleri seçebilirsiniz:
  - Zaman tabanı = Milisaniye (msn)
  - Darbe süresi formatı = Yüzde bir (0 ila 100)
  - Çevrim süresi = 3 msn
  - Başlangıç darbe süresi = 75

  Sonuçta elde edilen "darbe süresi" = 0,75 × 3 msn = 2,25 msn

  Bu "darbe süresi" değeri bir kesirli değeridir ve siz CTRL_PWM yordamını yürüttüğünüzde bir hataya neden olur. "Darbe süresi" değeri bir tamsayı değeri olmalıdır.
- Örnek 2: Aşağıdaki değerleri seçebilirsiniz:
  - Zaman tabanı = Mikrosaniye (µsn)
  - Darbe süresi formatı = Yüzde bir (0 ila 100)
  - Çevrim süresi = 3000 µsn
  - Başlangıç darbe süresi = 75

  Sonuçta elde edilen "darbe süresi" = 0,75 × 3000 µsn = 2250 µsn

  Bu "darbe süresi" bir tamsayı değeridir ve CTRL_PWM yordamı bu değerle sorunsuz bir şekilde çalışır.

Donanım çıkışları

Donanım çıkışları alanındaki açılır menüden çıkış kanalını seçin. Konfigürasyona bağlı olarak bir veya iki çıkış seçime sunulur. Bir çıkış kanalını bir darbe jeneratörüne atarsanız, çıkış kanalı başka bir darbe jeneratörü, HSC veya proses görünümü tarafından kullanılamaz.

NOT

Darbe jeneratörü çıkışları, kullanıcı programındaki başka yordamlar tarafından kullanılamaz.

CPU'nun veya sinyal kartının çıkışlarını darbe jeneratörleri (PWM, PTO veya hareket kumandası yordamları için) olarak yapılandırırsanız, ilgili çıkış adresleri çıkışların belleğinden kaldırılır ve programınızda başka amaçlarla kullanılamaz. Programınız, darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU, bu değeri fiziksel çıkışa yazmaz.

G/Ç adresleri

PWM, "darbe süresi" için kullanılan A bellek alanında iki Bayt'a sahiptir. PWM çalışırken, A bellek alanındaki atanmış değeri değiştirebilir ve bununla birlikte darbe süresinde değişiklik yapabilirsiniz.

G/Ç adresleri bölümünde, darbe süresi için kullanılan değeri kaydetmek istediğiniz çıkış sözcüğünün adresini girin.

PWM darbe süresi değerleri için kullanılan standart adresler:

- PWM1: AW1000
- PWM2: AW1002
- PWM3: AW1004
- PWM4: AW1006

PWM'de bu adresteki değer, darbe süresine kumanda eder ve CPU'nun STOP işletim durumundan RUN işletim durumuna her geçişinde (yukarıda atanan) "Başlangıç darbe süresi:" değeri başlatılır. Darbe süresini değiştirmek için bu çıkış sözcüğü değerini, çalışma zamanı sırasında değiştirebilirsiniz. Bu değerin aralığı, parametrelendirmede yapılandırılan darbe süresi formatına bağlıdır.

PWM sinyalinin "çevrim süresinin" A bellek alanında dört Bayt daha atayabilirsiniz. PWM sinyalinin bir diyagramını içeren "Darbe çıkışlarının çalışma şekli" (Sayfa 444) başlığına bakın. "Çevrim süresinin çalışma zamanında değiştirilmesine izin ver" opsiyon kutusu etkinleştirildiyse, ilk iki Bayt, darbe süresi değerini ve son dört Bayt ise çevrim süresi değerini kaydeder.

PWM çalışırken, bu PWM'ye atanmış A bellek alanının sonundaki çift sözcüğün değerini değiştirebilirsiniz. Bu sayede, PWM sinyalinin çevrim süresi değiştirilir. Bu opsiyonu örneğin, CPU, PWM1 için altı Bayt atayacak şekilde etkinleştirebilir ve bunun için AB1008 ila AB1013 aralığını seçebilirsiniz. Programı yüklediğinizde ve PWM'yi başlattığınızda, darbe süresini AW1008 üzerinden ve çevrim süresini AD1010 üzerinden değiştirebilirsiniz.

CPU, STOP işletim durumundan RUN işletim durumuna geçerse, CPU, A bellek alanındaki çevrim süresi değerini, "Parametrelendirme" alt bölümünde "Çevrim süresi" için atanan değerle başlatır. A bellek alanındaki çevrim süresi değeri için kullanılan birimler ve değer aralıkları, "Parametrelendirme" alt bölümündeki konfigürasyona karşılık gelir.

"Çevrim süresinin çalışma zamanında değiştirilmesine izin ver" opsiyon kutusunu etkinleştirdiyseniz, TIA Portal, çıkış adresi için otomatik olarak yeni bir adres seçer. Yeni çıkış adresi, bu darbe jeneratörü için kullanılan standart adresle aynı olamaz. TIA Portal, birbirini takip eden altı Bayt içeren bir sonraki kullanılabilir bloğu kullanır. A bellek alanında, bellek alanının sonundan önce gerçekleştirilen aramada kullanılabilir bir blok bulunamadıysa, arama işlemi, kullanılabilir bir blok bulunana kadar A bellek alanında "0" adresinden yeniden başlatılır.

PTO için yapılandırılan bir darbe jeneratörü, çıkış sözcüğü adresini kullanmaz.

9.9 Reçeteler ve veri protokolleri

9.9.1 Reçeteler

9.9.1.1 Reçetelere genel bakış

Reçete verilerinin kaydedilmesi

- Projenizde oluşturduğunuz bir reçete veri bloğu, CPU'nun **yükleme** belleğine kaydedilmelidir. Bunun için dahili CPU belleği veya program kartı olarak kullanılan harici bir memory card kullanılabilir.
- Oluşturmanız gereken diğer bir DB, aktif reçete veri bloğudur. Bu DB, program lojiğiniz tarafından her seferinde aktif bir reçete veri bloğunun okunacağı veya yazılacağı **çalışma** belleğinde olmalıdır.

Reçete verilerinin yönetimi

Reçete DB, ürün reçete veri setlerinden oluşan bir Array kullanır. Reçete Array'inin her elemanı, ortak bir bileşenler setini temel alan farklı reçete varyantlarını temsil eder.

- Bir reçete veri setindeki tüm bileşenleri tanımlayan bir PLC veri tipi veya bir Struct veri tipi oluşturabilirsiniz. Bu veri tipi şablonu, tüm reçete veri setleri için birkaç kez kullanılır. Ürün reçeteleri, reçete bileşenlerine atanan başlangıç değerlerine bağlı olarak değişiklik gösterir.
- Reçetelerden biri istenildiği zaman READ_DBL yordamıyla reçete DB'den (yükleme belleğindeki tüm reçeteler) aktif reçete DB'ye (çalışma belleğindeki bir reçete) aktarılabilir. Bir reçete veri seti çalışma belleğine taşındıktan sonra, program lojiğiniz bileşen değerlerini okuyabilir ve üretim çalışmasını başlatabilir. Aktarma yoluyla, reçete verileri için gereken CPU çalışma belleği miktarı minimuma indirilir.
- Aktif reçetenin bileşen değerleri bir üretim çalışması sırasında bir HMI cihazı tarafından uyarlanırsa, değiştirilen değerleri WRIT_DBL yordamıyla reçete DB'ye geri yazabilirsiniz.

Reçetelerin dışa aktarılması (reçete DB'den bir CSV dosyasına)

RecipeExport yordamıyla, reçete veri setlerinden oluşan eksiksiz set, bir CSV dosyası formatında dışa aktarılabilir. Kullanılmayan reçete veri setleri de bu sırada dışa aktarılır.

Reçetelerin içe aktarılması (bir CSV dosyasından bir reçete DB'ye)

Reçeteler dışa aktarıldıktan sonra, üretilen CSV dosyasını veri yapısı şablonu olarak kullanabilirsiniz.

1. Mevcut bir CSV reçete dosyasını, CPU Web sunucusunun dosya tarayıcısı sayfası aracılığıyla bir CPU'dan bir PC'ye yükleyebilirsiniz.
2. CSV reçete dosyasını bir ASCII metin editörünün yardımıyla değiştirin. Bileşenlerin başlangıç değerlerini değiştirebilir, ancak veri tiplerini veya veri yapısını değiştiremezsiniz.
3. Değiştirilen CSV dosyasını PC'den tekrar CPU'ya ger yükleyin. Ancak bu amaçla, CPU Web sunucusunun yükleme işlemini gerçekleştirebilmesi için eski CSV dosyasını (aynı addaki) CPU'nun yükleme belleğinden silmeniz veya yeniden adlandırmanız gerekir.
4. Değiştirilen CSV dosyasını CPU'ya yükledikten sonra, yeni başlangıç değerlerini RecipeImport yordamıyla değiştirilen CSV dosyasından (CPU yükleme belleğinde) reçete DB'ye (CPU yükleme belleğinde) aktarabilirsiniz.

9.9.1.2 Bir reçeteyle ilgili örnek

Reçetelerle ilgili örnek

Bu tabloda, reçete bilgilerini, bir reçete DB'de kullanım için nasıl hazırlayacağınız gösterilmektedir. Bir reçete DB için bu örnekte, aralarından üçünün kullanılacağı beş adet veri seti kaydedilmiştir. Dördüncü ve beşinci veri seti, sonraki genişletme işlemleri için serbest bırakılır. Tablodaki her satır; reçete adı, bileşen veri tipleri ve bileşen değerleriyle birlikte bir veri setini temsil eder.

productname	water	barley	wheat	hops	yeast	waterTmp	mashTmp	mashTime	QTest
Pils	10	9	3	280	39	40	30	100	0
Lager	10	9	3	150	33	50	30	120	0
BlackBeer	10	9	3	410	47	60	30	90	1
Not_used	0	0	0	0	0	0	0	0	0
Not_used	0	0	0	0	0	0	0	0	0

Bir reçete veri bloğunun oluşturulması

NOT

Reçete veri bloklarıyla ilgili kurallar

- Reçete DB, PLC veri tipinde veya Struct veri tipinde tek boyutlu bir Array içermelidir. Reçete örneğinde, PLC veri tipinde bir reçete DB'yi nasıl oluşturacağınız gösterilmektedir.
- Örnekte, tüm bileşen içerik maddeleri UINT veri tipine sahiptir. Bileşenlerin veri tipleri, Struct haricinde çeşitli veri tiplerinden oluşa bir karışım da olabilir. Reçete DB'deki bir Array elemanında bir PLC veri tipinde bir Struct'a veya Struct veri tipinde bir Struct'a izin verilmez.

Önce yeni bir PLC veri tipi oluşturun

Buna, adı, reçete tipini belirten yeni bir PLC veri tipi ekleyin. Aşağıdaki şekilde "Beer_Recipe", bir dizi basit veri tipini saklayan yeni karmaşık PLC veri tipidir. "Beer_Recipe" PLC veri tipi, reçete DB'nin her veri setinde ve aktif reçete DB'de de tekrar kullanılan bir veri şablonudur. Tüm örnek reçetelerde ortak olarak mevcut olan bileşen adlarını ve veri tiplerini girin. Münferit bileşen değerleri daha sonra reçete DB'ye eklenir.

Beer_Recipe

	Name	Data type	Default value
1	productname	String[20]	'Beer_Recipe'
2	water	UInt	0
3	barley	UInt	0
4	wheat	UInt	0
5	hops	UInt	0
6	yeast	UInt	0
7	waterTmp	UInt	0
8	mashTmp	UInt	0
9	mashTime	UInt	0
10	QTest	UInt	0

Ardından bir reçete veri bloğu oluşturun

- Reçete DB'nizi, "Sadece yükleme belleğine kaydet" DB özelliği etkinleştirilmiş şekilde global veri bloğu olarak oluşturun.
- Reçete veri bloğunun adı, ilgili CSV dosyasının dosya adı olarak kullanılır. DB adında bulunan karakterler, Windows dosya sisteminin ad kısıtlamalarına uymalıdır. \ / : * ? " < > | karakterlerine ve boşluk karakterine izin verilmez.
- Reçete Array'inin ataması Array [1.. 5] of "Beer_Recipe" olarak "Products" şeklindedir. Array boyutu 5, mümkün olan maksimum reçete varyantı sayısıdır.
- Reçete bileşenleri için kullanılan değerler, DB başlangıç değerleri olarak eklenir.

Bir reçete veri setinin tüm bileşenlerini göstermek için aşağıdaki şekilde "BlackBeer" reçetesi genişletilmiştir.

Recipe_DB

	Name	Data type	Offset	Start value
1	Static			
2	Products	Array [1 .. 5] of "Beer_Recipe"	...	
3	Products[1]	"Beer_Recipe"	...	
4	Products[2]	"Beer_Recipe"	...	
5	Products[3]	"Beer_Recipe"	...	
6	productname	String[20]	...	'BlackBeer'
7	water	UInt	...	10
8	barley	UInt	...	9
9	wheat	UInt	...	3
10	hops	UInt	...	410
11	yeast	UInt	...	47
12	waterTmp	UInt	...	60
13	mashTmp	UInt	...	30
14	mashTime	UInt	...	90
15	QTest	UInt	...	1
16	Products[4]	"Beer_Recipe"	...	
17	Products[5]	"Beer_Recipe"	...	

Reçetelerin dışa aktarılması (reçete DB'den CSV dosyasına)

"RecipeExport (Sayfa 454)" yordamı çalıştırılarak veriler reçete DB'den bir CSV dosyasına aktarılır (bkz. aşağıdaki metin dosyası).

```
Recipe_DB.csv
index,productname,water,barley,wheat,hops,yeast,waterTmp,
mashTmp,mashTime,QTest
1,"Pils",10,9,3,280,39,40,30,100,0
2,"Lager",10,9,3,150,33,50,30,120,0
3,"BlackBeer",10,9,3,410,47,60,30,90,1
4 "Not_used",0,0,0,0,0,0,0,0,0
5 "Not_used",0,0,0,0,0,0,0,0,0
```

Reçetelerin içe aktarılması (CSV dosyasından reçete DB'ye)

1. Mevcut bir CSV reçete dosyasını, CPU Web sunucusunun dosya tarayıcısı sayfası (Sayfa 838) aracılığıyla CPU yükleme belleğinden PC'ye yükleyebilirsiniz.
2. CSV reçete dosyasını bir ASCII metin editörünün yardımıyla değiştirin. Bileşenlerin başlangıç değerlerini değiştirebilir, ancak veri tiplerini veya veri yapısını değiştiremezsiniz.
3. Değiştirilen CSV dosyasını PC'den tekrar CPU'ya ger yükleyin. Ancak bu amaçla, CPU Web sunucusunun yükleme işlemini gerçekleştirebilmesi için eski CSV dosyasını (aynı addaki) CPU'nun yükleme belleğinden silmeniz veya yeniden adlandırmanız gerekir.
4. Değiştirilen CSV dosyasını CPU'ya yükledikten sonra, yeni başlangıç değerlerini RecipeImport yordamıyla değiştirilen CSV dosyasından (CPU yükleme belleğinde) reçete DB'ye (CPU yükleme belleğinde) aktarabilirsiniz.

CSV dosyaları, ilgili reçete DB'sinin yapısıyla tamamen aynı olmalıdır

- CSV dosyasındaki değerler değiştirilebilir, ancak yapı değiştirilemez. RecipeImport yordamı için, veri setlerinin ve bileşenlerin sayısı, hedef reçete DB'nin yapısına tam olarak karşılık gelmelidir. Aksi takdirde RecipeImport yordamının yürütülmesi başarısız olur. Örnek: Reçete DB'de 10 reçete tanımlandıysa, ancak bunların sadece 6'sı kullanılıyorsa, CSV dosyasının 7 ile 10 arasındaki satırları da DB'ye aktarılır. Bu verilerin geçerli olup olmadığını kontrol etmeniz gerekir. Dolayısıyla, örneğin kullanılmayan reçete veri setlerinde ürün adına "Not_used" değişkenini atayabilirsiniz.
- Metin dosyasına veri setleri eklerseniz ve değiştirilen dosyayı içe aktarırsanız, reçete DB'nin atadığınız Array limitinin, tüm reçete veri setleri için yeterli sayıda elemanı kapsamasına dikkat etmeniz gerekir.
- CSV dosyası formatında dışa aktarma sırasında otomatik olarak bir indeks numarası üretilir. Ek veri setleri oluşturursanız, buna uygun olarak sonraki indeks numaralarını ekleyin.
- RecipeImport yordamı yürütülürken, CSV dosyası, doğru yapı açısından ve değerlerin, ilgili reçete DB'de atanan veri tiplerine uyup uymadığı açısından kontrol edilir. Bu sayede, örneğin Boole veri tipi tam sayı değerlerini saklayamaz ve bu gibi bir durumda RecipeImport yordamının yürütülmesi başarısız olur.

Excel'deki CSV reçete verilerinin görüntülenmesi

CSV dosyasını, daha kolay bir şekilde okumak ve düzenlemek için Excel'de açabilirsiniz. Virgüller desimal ayırma işareti olarak algılanmazsa, verileri yapılandırılmış şekilde yayınlamak için Excel'in içe aktarma fonksiyonunu kullanabilirsiniz.

	A	B	C	D	E	F	G	H	I	J	K
1	index	product	water	barley	wheat	hops	yeast	waterTmp	mashTmp	mashTime	QTest
2	1	"Pils"	10	9	3	280	39	40	30	100	0
3	2	"Lager"	10	9	3	150	33	50	30	120	0
4	3	"BlackBeer"	10	9	3	410	47	60	30	90	1
5	4	"Not_used"	0	0	0	0	0	0	0	0	0
6	5	"Not_used"	0	0	0	0	0	0	0	0	0

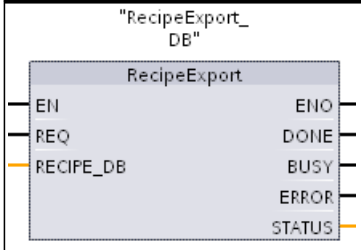
NOT

PLC veri tipi elemanının ad alanındaki virgüller

Bir reçetede kullanılan PLC veri tipi elemanının ad alanına virgül girmeyin. Ad alanına virgül girerseniz, Excel, görüntülenen CSV dosyasına ek sütunlar ekler. Bu sütunlar, reçete veri seti dosyasının başlangıç değerlerini düzenlemeniz sırasında hatalara neden olabilir.

9.9.1.3 Reçete verilerinin aktarılması için kullanılan program yordamları

Tablo 9-195 RecipeExport yordamı

LAD/FBD	SCL	Açıklama
	**"RecipeExport_DB"(req:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, Recipe_DB:=_variant_inout_);**	"RecipeExport" yordamı, bir reçete veri bloğundaki tüm reçete veri setlerini CSV dosya formatında dışa aktarır. CSV dosyası ürün adları, bileşen adları ve başlangıç değerlerini içerir. CSV dosyası, dahili yükleme belleğine veya program kartı olarak kullanılan opsiyonel, harici bir memory card takılı durumdaysa, harici yükleme belleğine da kaydedilir. Dışa aktarma işlemi "REQ" parametresi tarafından başlatılır. Dışa aktarma işleme süreci sırasında BUSY parametresi 1 olarak ayarlanır. RecipeExport yordamı yürütüldükten sonra BUSY değeri tekrar 0'a ayarlanır ve işlemin sona erdiği, DONE parametresinde 1 değeriyle gösterilir. Yürütme sırasında bir hata oluşursa, ERROR ve STATUS parametreleri sonucu gösterir.

Reçetelerin dışa aktarılmasının mümkün olması için bir reçete DB oluşturulmalıdır. Reçete veri bloğunun adı, yeni CSV dosyasının dosya adı olarak kullanılır. Halihazırda aynı ada sahip bir CSV dosyası mevcutsa, dışa aktarma işlemi sırasında bu dosyanın üzerine yazılır.

CPU'nun entegre Web sunucusunun dosya tarayıcısı sayfası (Sayfa 838) aracılığıyla, reçetelerin bulunduğu CSV dosyasına erişebilirsiniz. Dosya, CPU yükleme belleğinin kök dizinindeki reçete klasörüne kaydedilir.

Tablo 9-196 Parametreler için kullanılan veri tipleri

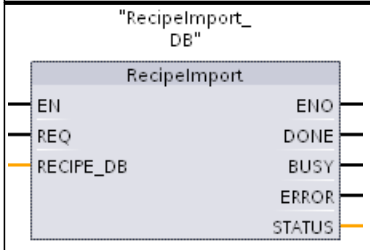
Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQUEST kumanda parametresi: Yükselen bir kenar durumunda dışa aktarmayı etkinleştirir.

Parametre ve veri tipi		Veri tipi	Açıklama
RECIPE_DB	Geçiş	Variant	İşaretçiyi, reçete veri bloğunun üzerine götürün. Daha fazla bilgiyi, Bir reçete DB örneği (Sayfa 451) başlığında bulabilirsiniz. DB adında bulunan karakterler, Windows dosya sisteminin ad kısıtlamalarına uymalıdır. \ / : * ? " < > | karakterlerine ve boşluk karakterine izin verilmez.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur. (Standart değer: Yanlış)
BUSY	OUT	Bool	RecipeExport'un yürütülmesi • 0: İşlenen yordam yok • 1: Yordam işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir. • 0: Uyarı yok ve hata yok • 1: Bir hata oluştu. STATUS parametresi, hata türüyle ilgili bilgileri içerir.
STATUS	OUT	Word	Yürütme koşulu

Tablo 9-197 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#....)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8070	Tüm örnek belleği dolu.
1	8090	Dosya adı geçersiz karakterler içeriyor.
1	8091	RECIPE_DB'de belirtilen veri yapısı işlenemiyor.
1	8092	RECIPE_DB'de belirtilen veri yapısı, 5000 Bayt'ı aşıyor.
1	80B3	MC'de veya dahili yükleme belleğinde yeterli alan yok.
1	80B4	MC yazma korumalıdır.
1	80B6	Reçete DB'nin "Sadece yükleme belleğine kaydet" özniteliği etkinleştirilmedi.
1	80C0	CSV dosyası kısa süreyle bloke edildi.
1	80C1	DB kısa süreyle bloke edildi.

Tablo 9-198 RecipeImport yordamı

LAD/FBD	SCL	Açıklama
	**"RecipeImport_DB"(req:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, Recipe_DB:=_variant_inout_);**	"RecipeImport" yordamı, bir CSV dosyasındaki reçete verilerini CPU yükleme belleğinde RECIPE_DB parametresi tarafından belirtilen dört reçete veri bloğundan birine aktarır. İçe aktarma işlemi sırasında, reçete veri bloğundaki başlangıç değerlerinin üzerine yazılır. İçe aktarma işlemi "REQ" parametresi tarafından başlatılır. İçe aktarma işleme süreci sırasında BUSY parametresi 1 olarak ayarlanır. RecipeImport yordamı yürütüldükten sonra BUSY değeri tekrar 0'a ayarlanır ve işlemin sona erdiği, DONE parametresinde 1 değeriyle gösterilir. Yürütme sırasında bir hata oluşursa, ERROR ve STATUS parametreleri sonucu gösterir.

Tablo 9-199 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQUEST kumanda parametresi: Yükselen bir kenar durumunda içe aktarmayı etkinleştirir.
RECIPE_DB	Geçiş	Variant	İşaretçiyi, reçete veri bloğunun üzerine götürün. Daha fazla bilgiyi, "Bir reçete DB örneği (Sayfa 451)" başlığında bulabilirsiniz. DB adında bulunan karakterler, Windows dosya sisteminin ad kısıtlamalarına uymalıdır. \ / : * ? " < > | karakterlerine ve boşluk karakterine izin verilmez.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur. (Standart değer: Yanlış)
BUSY	OUT	Bool	• 0 - İşlenen yordam yok • 1 - Yordam işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Bir içe aktarma işleminin gerçekleştirilebilmesi için, CSV dosyasının veri yapısına uygun bir yapıya sahip bir reçete DB mevcut olmalıdır.

CSV dosyalarıyla ilgili kurallar:

- CSV dosyası, dahili yükleme belleğinde veya program kartı olarak kullanılan opsiyonel, harici bir memory card takılı durumdaysa, harici yükleme belleğinde da"Recipes" kök dizinde bulunmalıdır.
- CSV dosyasının adı, RECIPE_DB parametresindeki veri bloğunun adına karşılık gelmelidir.
- CSV dosyasının ilk satırı (başlık), reçete bileşenlerinin adını içerir. İlk satır, içe aktarma sırasında göz ardı edilir. CSV dosyasındaki ve veri bloğundaki reçete bileşenlerinin adları, içe aktarma işlemi sırasında birbirleriyle eşleştirilmez.

- Her durumda, CSV dosyasının her satırındaki ilk değer, reçetenin indeks numarasıdır. Münferit reçeteler, indeksin sırasıyla içe aktarılır. Bunun için, CSV dosyasındaki indeks artan sırada olmalı ve boşluk içermemelidir (aksi takdirde STATUS parametresinde 80B0 hata mesajı yayınlanır).
- CSV dosyası, reçete veri bloğunda öngörülenden daha fazla reçete veri seti içermemelidir. Maksimum veri seti sayısı, veri bloğundaki Array limitleri tarafından belirtilir.

Tablo 9-200 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#....)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8070	Tüm örnek belleği dolu.
1	8090	Dosya adı geçersiz karakterler içeriyor.
1	8092	İçe aktarmak için uygun bir CSV dosyası bulunamadı. Olası neden: CSV dosyasının adı, reçete DB'nin adına karşılık gelmiyor.
1	80C0	CSV dosyası kısa süreyle bloke edildi.
1	80C1	Veri bloğu kısa süreyle bloke edildi.
1	80B0	CSV dosyasının indeksindeki numaralandırma sürekli değil, artan sırada değil veya veri bloğundaki maksimum sayıyı (Array limiti) aşıyor.
1	80B1	Reçete veri bloğunun ve CSV dosyasının yapısı aynı değil: CSV dosyası çok fazla alan içeriyor.
1	80B2	Reçete veri bloğunun ve CSV dosyasının yapısı aynı değil: CSV dosyası çok az alan içeriyor.
1	80B6	Reçete DB'nin "Sadece yükleme belleğine kaydet" özniteliği etkinleştirilmedi.
1	80D0 +n	Reçete veri bloğunun ve CSV dosyasının yapısı aynı değil: n alanındaki veri tipi aynı değil (n <= 46).
1	80FF	Reçete veri bloğunun ve CSV dosyasının yapısı aynı değil: n alanındaki veri tipi aynı değil (n > 46).

9.9.1.4 Reçetelerle ilgili örnek program

Reçetelerle ilgili örnek program için tanımlanan ön koşullar

Reçetelerle ilgili örnek program için aşağıdaki ön koşullar sağlanmış olmalıdır:

- Tüm reçete veri setlerini içeren bir reçete DB. Reçete DB; yükleme belleğine kaydedildi.
- Çalışma belleğindeki bir reçetenin kopyasını içeren aktif bir reçete DB.

Reçete DB ve ilgili CSV dosyasıyla ilgili daha fazla bilgiyi, "Bir reçete DB örneği (Sayfa 451)" başlığında bulabilirsiniz.

Aktif reçete DB'yi oluşturun

"Yeni blok ekle" penceresinde:

- "Yeni blok ekle" penceresindeki "Veri bloğu" butonunu seçin.
- "Tip" açılır listesinden, daha önce oluşturulan "Beer_recipe" PLC veri tipini seçin.

Başlangıç değerleri gerekli değildir. Bir reçete, reçete DB'den aktif reçete DB'ye aktarılacaksa, DB veri değerleri belirlenir. Örnekte aktif reçete DB, READ_DBL yordamının verileri için hedeftir ve WRITE_DBL yordamına kaynak verileri sunar. Aşağıdaki şekilde, "Active_Recipe" DB'si gösterilmektedir.

Active_Recipe

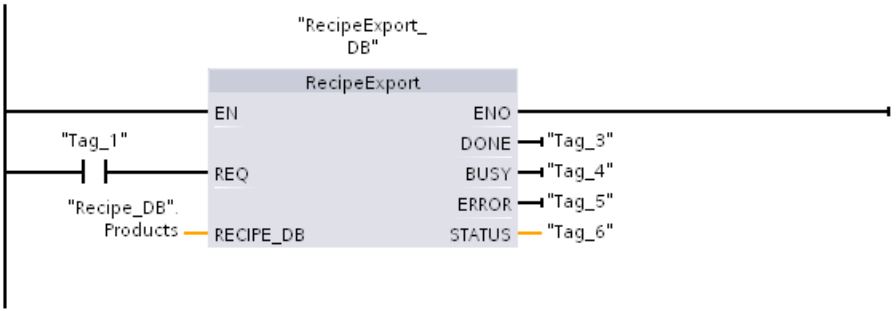
	Name	Data type	Start value
1	Static		
2	productname	String[20]	'Beer_Recipe'
3	water	UInt	0
4	barley	UInt	0
5	wheat	UInt	0
6	hops	UInt	0
7	yeast	UInt	0
8	waterTmp	UInt	0
9	mashTmp	UInt	0
10	mashTime	UInt	0
11	QTest	UInt	0

Örnek DB'ler

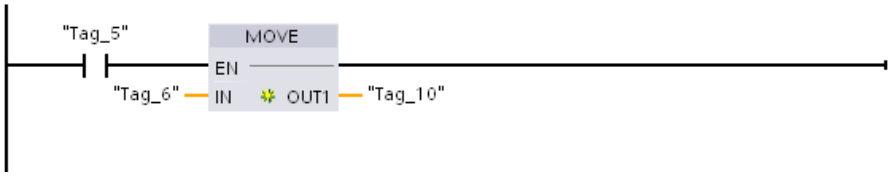
RecipeExport ("RecipeExport_DB") ve RecipeImport ("RecipeImport_DB") yordamları tarafından kullanılan örnek DB'ler, yordamları programınıza eklediğiniz takdirde otomatik olarak oluşturulur. Örnek DB'ler, yordam yürütmeyi kontrol etmek için kullanılır ve program lojiğinde referanslanmaz.

Reçetelerle ilgili örnek program

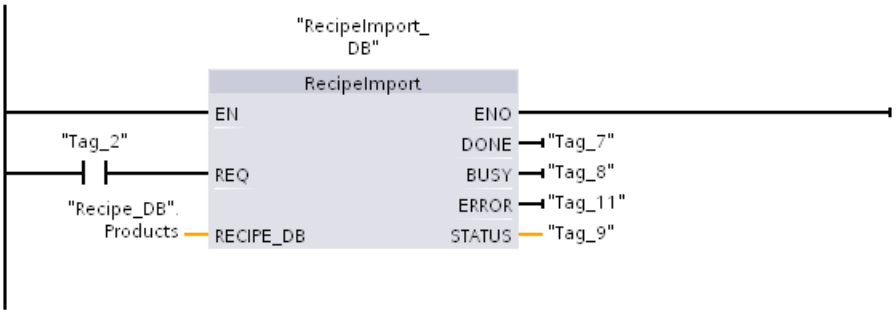
Ağ 1 REQ'teki bir yükselen kenar, dışa aktarma işlemini başlatır. Reçete DB'nin verilerinden bir CSV dosyası oluşturulur ve CPU belleğindeki reçete klasörüne kaydedilir.



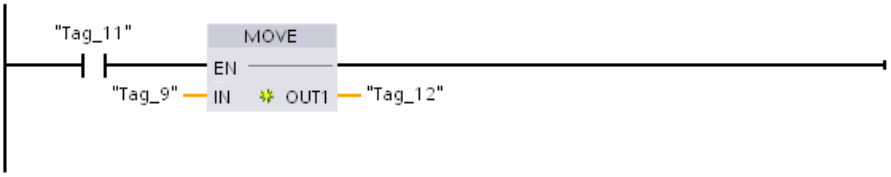
Ağ 2 Sadece bir çevrim boyunca geçerli olduğu için, RecipeExport yordamının yürütülmesine ait STATUS çıkışını belirleyin.



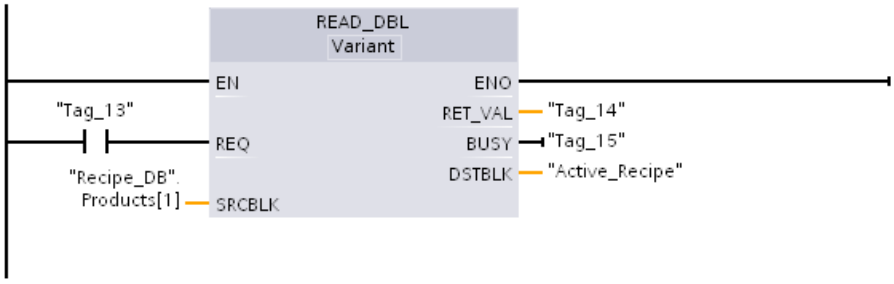
Ağ 3 REQ'teki bir yükselen kenar, içe aktarma işlemini başlatır. CPU belleğinin reçete klasöründeki ilgili CSV dosyasından okunan tüm reçete verileri mevcut reçete DB'ye yüklenir.



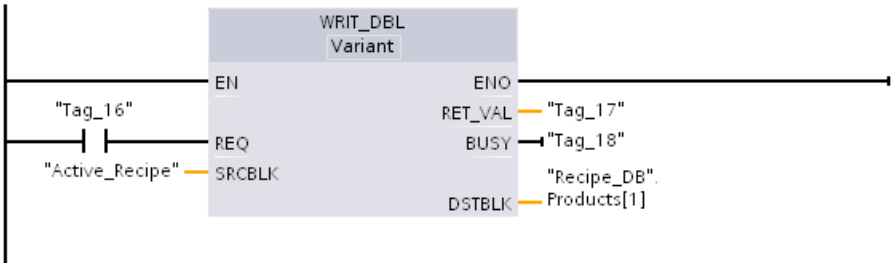
Ağ 4 Sadece bir çevrim boyunca geçerli olduğu için, RecipeImport yordamının yürütülmesine ait STATUS çıkışını belirleyin.



Ağ 5 READ_DBL, bir "Recipe_DB" reçetesinden başlangıç değerlerini kopyalar. Products[1] (CPU yükleme belleğinde) "Active_Recipe" DB'nin (CPU çalışma belleğinde) güncel değerlerine. READ_DBL yordamı yürütüldükten sonra program lojiğiniz, "Active_Recipe" DB'nin belleğindeki adresler aracılığıyla reçetenin bileşen değerlerine erişebilir. Bu sayede, örneğin program lojiğinizin sembolik adresleri ("Active_Recipe".productname) ve ("Active_Recipe".water) güncel reçete adını ve suyun miktarını belirtir.



Ağ 6 Çalışma zamanı esnasında bir HMI cihazı, "Active_Recipe" DB'ye kaydedilen bir bileşen değerini değiştirebilir. İyileştirilmiş reçete verileri, WRIT_DBL yordamı yürütülerek kaydedilebilir. Örnekte, münferit reçete için reçete DB'nin tüm başlangıç değerleri "Recipe_DB" haline gelir. Products[1] "Active_Recipe" DB'deki güncel değerlerle üzerine yazma.



9.9.2 Veri protokolleri

Kumanda programınız, çalışma zamanı verilerinin kalıcı protokol dosyalarına kaydedilmesi için veri protokolü yordamlarını kullanabilir. CPU, veri protokolü dosyalarını Flash belleğe (CPU veya memory card) standart CSV formatında (virgülle ayrılan değerler aracılığıyla) kaydeder. CPU, veri setlerini, önceden tanımlanmış bir boyuttaki yuvarlak şeklindeki bir protokol dosyasında organize eder.

Veri protokolündeki yordamlarla programınızda protokol dosyaları oluşturabilir, bu dosyaları açabilir, kapatabilir ve dosyalara bir veri setini yazabilirsiniz. Hangi program değerlerinin protokollendirileceğini, tek bir protokol veri seti tanımlayan bir veri tamponu oluşturarak karar verebilirsiniz. CPU, veri tamponunuzu, yeni bir protokol veri seti için geçici bellek olarak kullanır. Kumanda programınız, çalışma zamanı esnasında yeni değerleri tampona taşır. Program, tüm güncel veri değerlerini güncellediyse, verileri tampondan bir protokol veri setine aktarmak için DataLogWrite yordamını yürütebilir.

Veri protokolleriyle Web sunucusundan (Sayfa 796) çalışabilirsiniz.

9.9.2.1 Veri protokollerinin veri seti yapısı

DataLogCreate yordamının DATA ve HEADER parametreleri, bir protokol veri setindeki tüm veri elemanlarının veri tipini ve sütun başlığı açıklamasını atar.

DataLogCreate yordamının DATA parametresi

DATA parametresi, yeni bir protokol veri seti için geçici tampon olarak kullanılacak olan belleğe referans verir. Bu bileşene bir M veya DB adresi atanmış olmalıdır.

Komple bir DB'yi (DB oluşturulurken belirttiğiniz bir PLC veri tipinden türetilen) veya bir DB'nin bir kısmını (belirtilen DB elemanında herhangi bir veri tipi, bir veri tipi yapısı, bir PLC veri tipi veya bir veri Array'i söz konusu olabilir) atayabilirsiniz.

DataLogCreate yordamında yapı veri tipleri tek bir gömme seviyesiyle sınırlanmıştır. Karakter dizilerinden oluşan bir Array, bu bağlamda tek bir gömme seviyesi olarak geçerli değildir. Şu anda DataLogCreate yordamı hata döndürmez. Yordam, sadece Array'in ilk karakter dizisini işler. Deklare edilen veri elemanlarının toplam sayısı, HEADER parametresinde belirtilen sütunların sayısına karşılık gelmelidir. Maksimum 253 veri elemanı (zaman notu ile) veya 255 veri elemanı (zaman notu olmadan) atayabilirsiniz. Veri setiniz, bu kısıtlama nedeniyle, bir Excel çalışma sayfasının maksimum 256 sütunundan oluşan limitler dahilinde kalır.

DATA parametresi, "Standart" (S7-300/400 ile uyumlu) veya "Optimize edildi" tipindeki bir DB'de bulunan kalıcı veya kalıcı olmayan veri elemanlarını atayabilir.

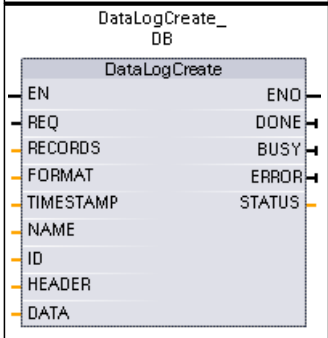
Bir protokol veri setini yazmak için, önce yeni proses değerlerini DATA geçici veri setine yüklemeniz ve ardından, yeni veri seti değerlerinin veri protokolü dosyasına kaydedilmesini sağlayan DataLogWrite yordamını yürütmeniz gerekir.

DataLogCreate yordamının HEADER parametresi

HEADER parametresi, CSV dosyasındaki veri matrisinin en üst satırında yer alan sütun başlıklarını gösterir. HEADER verileri DB veya M belleğinde bulunmalı ve karakterler, CSV dosyaları için kullanılan alışılagelmiş format kurallarına uymalıdır; bu sırada, münferit sütun adlarının virgüllerle ayrılması gerekir. Veri tipinde String'ler, Bayt Array'leri veya karakter dizileri de söz konusu olabilir. String veri tipi maksimum 255 Bayt'la sınırlı iken, karakter veya Bayt Array'leri daha büyük bir boyut kullanmaya olanak sağlar. HEADER parametresi opsiyoneldir. HEADER parametresi atanmadıysa, veri protokolü dosyasında hiçbir başlık satırı oluşturulmaz.

9.9.2.2 Veri protokollerini kontrol etmeye yönelik program yordamları

Tablo 9-201 DataLogCreate yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogCreate_DB"(** **req:=_bool_in_,** **records:=_udint_in_,** **format:=_uint_in_,** **timestamp:=_uint_in_,** **done=>_bool_out_,** **busy=>_bool_out_,** **error=>_bool_out_,** **status=>_word_out_,** **name:=_string_inout_,** **ID:=_dword_inout_,** **header:=_variant_inout_,** **data:=_variant_inout_);**	Bir veri protokolü dosyası oluşturur ve başlatır. CPU, dosyayı \DataLogs klasöründe oluşturur, NAME parametresindeki adı kullanır ve dosyayı yazma işlemleri için dolaylı olarak açar. Data log yordamlarıyla programınızda çalışma zamanı proses verilerini CPU'nun Flash belleğine veya memory card'a kaydedebilirsiniz. Yordamı eklerseniz, STEP 7 ilgili örnek DB'yi otomatik olarak oluşturur.

[1] SCL örneğinde "DataLogCreate_DB", örnek DB'nin adıdır.

Tablo 9-202 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Standart değer: Yanlış)
RECORDS	IN	UDint	En eski girişin üzerine yazılmadan önce yuvarlak şeklindeki veri protokolünü içerebilen maksimum veri seti sayısı: Başlık satırı içeren veri seti, bu sayıya dahil değildir. Veri protokolünün başarılı bir şekilde oluşturulması için yeterli miktarda PLC yükleme belleği mevcut olmalıdır. (Ön ayar - 1)
FORMAT	IN	UInt	Veri protokolünün formatı: • 0 - Dahili format (desteklenmiyor) • 1 - Virgülle ayrılan değerler, "csv-eng" (standart değer)
TIMESTAMP	IN	UInt	Verilerin zaman notunun formatı: Tarih ve saat alanları için sütun başlığı opsiyoneldir. Zaman notu, sistem saatini (Coordinated Universal Time - UTC) veya yerel saati gösterebilir. • 0 - Zaman notu yok • 1 - Sistem saati aa/gg/yyyy,ss:dd:snsn • 2 - Yerel saat aa/gg/yyyy,ss:dd:snsn • 3 - Sistem saati aa/gg/yyyy,ss:dd:snsn • 4 - Yerel saat yyyy-aa--gg,ss:dd:snsn • 5 - Sistem saati yyyy-aa-gg,ss:dd:snsn
NAME	IN	Variant	Veri protokolünün adı: Bir ad girin. Bu varyant, sadece String veri tipini destekler ve sadece DB belleğine veya yerel belleğe kaydedilebilir. (Standart değer: ' ') Bu karakter dizisi, veri protokolü dosyasının adıdır. ASCII karakter setindeki karakterleri kullanın, \ / : * ? " < > | karakterlerine ve boşluk karakterine izin verilmez.

Parametre ve veri tipi		Veri tipi	Açıklama
ID	Geçiş	DWord	Veri protokolünün sayısal kodu: Üretilen bu değeri, başka veri protokolü yordamlarıyla birlikte kullanmak için kaydedebilirsiniz. Parametre ID'si, DataLogCreate yordamında sadece çıkış olarak kullanılır. (Standart değer: 0) Bu parametreye sembolik bir ad üzerinden erişime izin verilmez.
HEADER	Geçiş	Variant	İmleci, CSV dosyasındaki veri matrisinin en üst satırında bulunan veri protokolünün sütun başlığının üzerine götürün. (Standart değer: sıfır). HEADER verileri DB veya M belleğinde olmalıdır. Karakterler, CSV dosyaları için kullanılan alışılagelmiş format kurallarına uymalı ve münferit sütun adları virgüllerle ayrılmalıdır. Veri tipinde String'ler, Bayt Array'leri veya karakter dizileri de söz konusu olabilir. String veri tipi maksimum 255 Bayt'la sınırlı iken, karakter veya Bayt Array'leri daha büyük bir boyut kullanmaya olanak sağlar. HEADER parametresi opsiyoneldir. HEADER parametrelendirilmediyse, veri protokolü dosyasında hiçbir başlık satırı oluşturulmaz.
DATA	Geçiş	Variant	İşaretçiyi; veri setinin, kullanıcı tanımlı veri tipinin (UDT) veya Array'in veri yapısı üzerine götürün. Veri seti verileri, DB veya M belleğinde olmalıdır. DATA parametresi, bir protokol veri setinin münferit veri elemanlarını (sütunlar) ve bunların veri tipini belirtir. DataLogCreate yordamında yapı veri tipleri tek bir gömme seviyesiyle sınırlanmıştır. Karakter dizilerinden oluşan bir Array, bu bağlamda tek bir gömme seviyesi olarak geçerli değildir. Şu anda DataLogCreate yordamı hata döndürmez. Yordam, sadece Array'in ilk karakter dizisini işler. Deklare edilen veri elemanlarının sayısı, HEADER parametresinde belirtilen sütunların sayısına karşılık gelmelidir. Maksimum 253 veri elemanı (zaman notu ile) veya 255 veri elemanı (zaman notu olmadan) atayabilirsiniz. Veri setiniz, bu kısıtlama nedeniyle, bir Excel çalışma sayfasının maksimum 256 sütunundan oluşan limitler dahilinde kalır.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur. (Standart değer: Yanlış)
BUSY	OUT	Bool	• 0 - İşlenen yordam yok • 1 - Yordam işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

CPU, RECORDS ve DATA parametrelerini temel alarak, önceden tanımlanmış bir boyuta sahip bir veri protokolü dosyası oluşturur ve veri setlerini yuvarlak şeklinde bir protokol dosyasında organize eder. DataLogCreate yordamı, DONE yordamının = DOĞRU değerini döndürmesi durumunda tüm veri protokolü için kalıcı bir CPU belleği atar. CPU'da gereken bellek alanı, dosya sistemi yönetimi ve bağlı değerler nedeniyle dosyadan büyüktür. Veri protokolü için kullanılan kalıcı bellek, CPU, belleği aşağıdaki yöntemlerden biriyle tekrar serbest bırakana kadar atanmış durumda kalır:

- Kullanıcı programı, DataLogDelete yordamını çağırır.
- Bir Web sunucusu kullanıcısı, Web sunucusunda veri protokolünü siler.
- Bir SIMATIC Automation Tool kullanıcısı, SIMATIC Automation Tool'da veri protokolünü siler.

Veri protokolü dosyası, örneğin bir kart okuyucu yardımıyla olduğu gibi başka bir yöntemle silinirse, CPU'daki veri protokolü için kullanılan kalıcı bellek serbest bırakılmaz.

DataLogWrite yordamı, RECORDS parametresinde önceden belirtilen maksimum sayıda veri seti kaydedilene kadar veri protokolü dosyasına yeni veri setleri ekler. Ardından, yazılan bir sonraki veri seti, en eski veri setinin üzerine yazılır. Diğer bir DataLogWrite yordamı, en eski ikinci veri setinin üzerine yazılır, vb.

Bellek kullanımı:

- Veri protokolleri sadece yükleme belleğini işgal eder.
- Toplamda tüm veri protokollerinin boyutu, yükleme belleğinin mevcut kapasitesiyle sınırlıdır. Eş zamanlı olarak maksimum sekiz protokol açık olabilir. Veri protokollerinizi Dosya tarayıcısı (Sayfa 838)standart web sitesi üzerinden yönetebilirsiniz. Eş zamanlı olarak kaç veri protokolünün kaydedilebileceğiyle ilgili direktifleri, bu standart web sitesinin açıklamasında bulabilirsiniz.
- RECORDS parametresi için mümkün olan maksimum sayı, bir UDint sayısının limitidir (4.294.967.295). RECORD parametresi için kullanılan gerçek limit, münferit bir veri setinin boyutuna, diğer veri protokollerinin boyutuna ve yükleme belleğinin kullanılabilir kapasitesine bağlıdır. Ayrıca Excel'de bir Excel çalışma sayfasındaki satır sayısı sınırlıdır.

NOT

Bir yordamın veri protokolünü yazma işlemini başlatabilmek için veri protokolü oluşturma işlemi tamamlanmış olmalıdır

- Veri protokollerinin oluşturulmasına yönelik DataLogCreate ve DataLogNewFile işlemleri, birçok program çevrimini kapsar. Protokol dosyasının oluşturulması için ihtiyaç duyulan gerçek süre, veri setlerinin veri seti yapısına ve sayısına bağlıdır. Program lojiğiniz, bir protokol dosyası oluşturma işleminin tamamlandığını bildiren DONE Bit'lerinin DOĞRU olarak değişimini denetlemeli ve kaydetmelidir. Kullanıcı programı, veri protokolü oluşturma işlemi tamamlanmadan önce bir DataLogWrite yordamını yürütürse, yazma işlemiyle, yeni protokol veri seti beklendiği gibi yazılamaz.
- Çok hızlı bir çevrim devam ederken belirli durumlarda veri protokolünün oluşturulması daha uzun sürebilir. Yavaş oluşturma işlemi çok uzun sürüyorsa, periyodik OB'ler için minimum çevrim süresini etkinleştirme opsiyon kutusunun etkinleştirildiğinden ve minimum çevrim süresinin 1 msn veya daha fazla olacak şekilde ayarlandığından emin olmanız gerekir. Daha fazla bilgiyi, Çevrim süresi ve iletişim yükünün yapılandırılması (Sayfa 84) başlığında bulabilirsiniz.

NOT

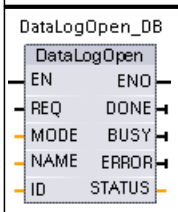
DataLogNewFile yordamı, mevcut bir veri protokolünün veri seti yapısını kopyalar

Veri setlerinin üzerine yazılmasını engellemek istiyorsanız, güncel veri protokolü, maksimum sayıda veri setini kaydettikten sonra DataLogNewFile yordamıyla, güncel veri protokolünü temel alarak yeni bir veri protokolü oluşturabilirsiniz. Bu durumda yeni veri setleri, yeni veri protokolü dosyasına kaydedilir. Eski veri protokolü dosyası ve veri seti verileri Flash bellekte kalır.

Tablo 9-203 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#....)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8070	Tüm dahili örnek belleği dolu.
1	807F	Dahili hata
1	8090	Geçersiz dosya adı
1	8091	Ad parametresi, karakter dizisi değildir.
1	8093	Bu ada sahip bir veri protokolü zaten mevcut. Başka bir ad kullanın, veri protokolünün CSV dosyasının açık olmadığından emin olun ve ardından Web sunucusunun Dosya tarayıcısı (Sayfa 838) sayfası aracılığıyla mevcut veri protokolünü silin.
1	8097	İstenen dosya uzunluğu, dosya sistemi için kullanılan maksimum dosya boyutunu aşıyor.
1	80B2	Başka hiçbir kaynak ID'si yok Not: Bu hatayı önlemek için, bazı mevcut veri protokollerini silin veya veri seti yapısındaki sütun sayısını azaltın.
1	80B3	Yükleme belleği yeterli değil.
1	80B4	Memory card (MC) yazma korumalıdır.
1	80C0	Arşiv dosyası bloke edilmiş
1	80C1	Çok fazla açık dosya mevcut: Eş zamanlı olarak maksimum sekiz veri protokolü dosyası açık olabilir.
1	8253	Geçersiz veri seti sayısı
1	8353	Geçersiz format seçimi
1	8453	Geçersiz zaman notu seçimi
1	8B24	Geçersiz HEADER aralık ataması: Örnek: Yerel belleğe referans verir
1	8B51	HEADER parametresi için geçersiz veri tipi
1	8B52	HEADER parametresinde çok fazla veri elemanı mevcut
1	8C24	Geçersiz DATA aralık ataması: Örnek: Yerel belleği gösterir
1	8C51	DATA parametresi için geçersiz veri tipi
1	8C52	DATA parametresinde çok fazla veri elemanı mevcut

Tablo 9-204 DataLogOpen yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogOpen_DB"(** **req:=_bool_in_,** **mode:=_uint_in_,** **done=>_bool_out_,** **busy=>_bool_out_,** **error=>_bool_out_,** **status=>_word_out_,** **name:=_string_inout_,** **ID:=_dword_inout_);**	Mevcut bir veri protokolü dosyasını açar. Protokole yeni veri setleri yazabilmeniz (Sayfa 467) için bir veri protokolünü açmanız gerekir. Veri protokollerini münferit olarak açabilir ve kapatabilirsiniz. Eş zamanlı olarak maksimum sekiz veri protokolü açık olabilir. Yordamı eklerseniz, STEP 7 ilgili örnek DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "DataLogOpen_DB", örnek DB'nin adıdır.

Tablo 9-205 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Standart değer: Yanlış)
MODE	IN	UInt	İşletim türü: • 0 - Mevcut verilere ekle (standart değer) • 1 - Mevcut tüm veri setlerini sil
NAME	IN	Variant	Mevcut bir veri protokolünün adı: Bu varyant, sadece String veri tipini destekler ve sadece yerel belleğe, DB veya M belleğine veya yerel belleğe kaydedilebilir. (Standart değer: ' ')
ID	Geçiş	DWord	Bir veri protokolünün sayısal kodu. (Standart değer: 0) **Not:** Bu parametreye sembolik bir ad üzerinden erişime izin verilmez.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur. (Standart değer: Yanlış)
BUSY	OUT	Bool	• 0 - İşlenen yordam yok • 1 - İşlem işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Mevcut bir veri protokolünün NAMEN öğesini veya bir ID'sini (giriş olarak ID parametresi) girebilirsiniz. Her iki parametreyi de belirtirseniz ve geçerli bir ID NAME veri protokolüne karşılık geliyorsa, ID kullanılır ve NAME göz ardı edilir.

NAME, DataLogCreate yordamıyla oluşturulan bir veri protokolüne karşılık gelmelidir. Sadece NAME belirtilirse ve NAME geçerli bir veri protokolüne referans veriyorsa, ilgili ID yayınlanır (çıkış olarak ID parametresi).

NOT

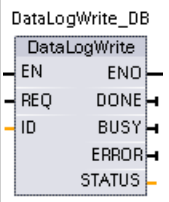
Veri protokolü dosyalarının genel kullanımı

- DataLogCreate ve DataLogNewFile yordamları yürütüldükten sonra veri protokolü dosyaları otomatik olarak açılır.
- PLC cihazı RUN işletim durumundan STOP işletim durumuna geçtikten veya PLC cihazı yeniden başlatıldıktan sonra veri protokolü dosyaları otomatik olarak kapatılır.
- Yeni bir DataLogWrite yordamının yürütülebilmesi için bir veri protokolü dosyası açılmış olmalıdır.
- Eş zamanlı olarak maksimum sekiz veri protokolü dosyası açık olabilir. Sekizden fazla veri protokolü dosyası mevcut olabilir, ancak eş zamanlı olarak maksimum sekizinin açık olması için bunlardan bazıları kapatılmalıdır.

Tablo 9-206 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	0002	Uyarı: Veri protokolü dosyası, bu uygulama programı tarafından zaten açıldı
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8070	Tüm dahili örnek belleği dolu.
1	8090	Veri protokolü tanımı, mevcut veri protokolü dosyasıyla uyuşmuyor.
1	8091	Ad parametresi, karakter dizisi değildir.
1	8092	Veri protokolü mevcut değil.
1	80C0	Veri protokolü dosyası bloke edilmiş.
1	80C1	Çok fazla açık dosya mevcut: Eş zamanlı olarak maksimum sekiz veri protokolü dosyası açık olabilir.

Tablo 9-207 DataLogWrite yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogWrite_DB"(req:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);**	Bir veri setini belirtilen veri protokolüne yazar. DataLogWrite yordamıyla protokole yazabilmeniz için mevcut hedef veri protokolü, açık (Sayfa 465) olmalıdır. Yordamı eklerseniz, STEP 7 ilgili örnek DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "DataLogWrite_DB", örnek DB'nin adıdır.

Tablo 9-208 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Standart değer: Yanlış)
ID	Geçiş	DWord	Veri protokolünün sayısal kodu: DataLogWrite yordamında sadece giriş olarak kullanılır. (Standart değer: 0) **Not:** Bu parametreye sembolik bir ad üzerinden erişime izin verilmez.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - İşlenen yordam yok • 1 - Yordam işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Bir DataLogCreate yordamının DATA parametresi, veri seti tamponunun bellek adresini ve veri yapısını tanımlar. Kumanda programı, yeni veri seti verilerini tampondan bir veri protokolüne aktarmak için güncel çalışma zamanı proses değerleriyle birlikte veri seti tamponunu yüklemeli ve ardından DataLogWrite yordamını yürütmelidir.

ID parametresi bir veri protokolü konfigürasyonunu ve veri seti konfigürasyonunu belirtir. DataLogCreate yordamı, ID sayısını oluşturur.

Yuvarlak şeklindeki veri protokolü dosyası boş veri setleri içeriyorsa, DataLogWrite yordamı bir sonraki boş veri setine yazar. Tüm veri setleri doluysa, DataLogWrite yordamı en eski veri setinin üzerine yazar.

DİKKAT

Bir yordamın veri protokolünü yazma işlemini başlatabilmek için veri protokolü oluşturma işlemi tamamlanmış olmalıdır

Veri protokollerinin oluşturulmasına yönelik DataLogCreate ve DataLogNewFile işlemleri, birçok program çevrimini kapsar. Protokol dosyasının oluşturulması için ihtiyaç duyulan gerçek süre, veri setlerinin veri seti yapısına ve sayısına bağlıdır. Program lojiğiniz, bir protokol dosyası oluşturma işleminin tamamlandığını bildiren DONE Bit'lerinin DOĞRU olarak değişimini denetlemeli ve kaydetmelidir. Veri protokolü oluşturma işlemi tamamlanmadan önce bir DataLogWrite yordamı yürütülürse, yazma işlemi hiçbir yeni protokol veri setini yazmaz.

NOT

Veri protokollerinin dahili CPU belleği üzerindeki etkisi

Bir veri protokolünün her yazma işlemi en az 2 KB bellek kullanır. Programınız sıkça daha küçük miktarlarda verileri yazıyorsa, her yazma işleminde en az 2 KB bellek kullanılır. Daha iyi bir uygulama, küçük veri elemanlarının, sonra daha az sıklıkla veri protokolüne yazılacak olan bir veri bloğunda (DB) toplanması şeklindedir.

Programınız birçok veri protokolü girişini çok sık yazıyorsa, değiştirilebilir bir SD memory card da kullanabilirsiniz.

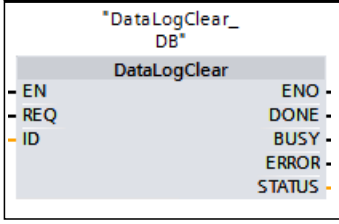
DİKKAT
Bir CPU gerilim kesintisi esnasında olası protokol verileri kaybı Henüz tamamlanmamış bir DataLogWrite yordamının yürütülmesi sırasında bir gerilim kesintisi oluşursa, veri protokolüne aktarılan veri seti muhtemelen kaybolur.

Tablo 9-209 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	0001	Veri protokolünün dolu olduğunu gösterir: Her veri protokolü, önceden belirtilen bir maksimum sayıdaki veri setiyle oluşturulur. Bu maksimum sayıdaki veri setlerinin sonuncusu yazıldı. Bir sonraki yazma yordamı, en eski veri setinin üzerine yazar.
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8070	Tüm dahili örnek belleği dolu.
1	8092	Veri protokolü mevcut değil.
1	80B0	Veri protokolü dosyası açık değil (sadece doğrudan açma işleminin söz konusu olduğu modda).

Açıklama

Tablo 9-210 DataLogClear yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogClear_DB"(REQ:=_bool_in_, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, ID:=_dword_inout_);**	"DataLogClear" yordamı, mevcut bir veri protokolündeki tüm veri setlerini siler. Yordam, CSV dosyasının opsiyonel Header'ını silmez (bkz. DataLogCreate (Sayfa 461-462)" yordamına ait HEADER parametresinin açıklaması "). ID parametresiyle, veri setleri silinecek olan veri protokolünü seçebilirsiniz.

"DataLogClear_DB", örnek DB'nin adıdır.

Ön koşul

Veri setlerini silebilmeniz için veri protokolü açık olmalıdır. Bir veri protokolünü açmak için DataLogOpen (Sayfa 465) yordamını kullanın.

Parametre

Aşağıdaki tabloda, "DataLogClear" yordamının parametreleri gösterilmektedir:

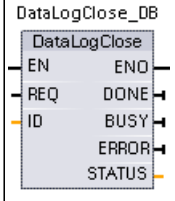
Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L, T, C veya sabit (T ve C sadece S7-1500'de LAD ve FBD'de kullanılabilir)	Yordamın, yükselen bir kenar durumunda yürütülmesi.
ID	InOut	DWORD	E, A, M, D, L	Veri protokolünün sayısal kodu
DONE	Output	BOOL	E, A, M, D, L	Yordam başarıyla yürütüldü.
BUSY	Output	BOOL	E, A, M, D, L	Yordamın yürütülmesi henüz tamamlanmadı.
ERROR	Output	BOOL	E, A, M, D, L	• 0: Hata yok. • 1: Yordamın yürütülmesi sırasında bir hata oluştu. Ayrıntılı bilgiler, STATUS parametresinde yayınlanır.
STATUS	Output	WORD	E, A, M, D, L	Durum parametresi Parametre, sadece bir çağrının süresi boyunca ayarlanır. Durumu görüntülemek için, bu nedenle STATUS parametresini boş bir veri alanına kopyalamanız gerekir.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

STATUS parametresi

Hata kodu* (W#16#...)	Anlamı
0000	Hata yok.
7000	Devam eden bir görev işlemesi yok.
7001	Görev işlemesinin başlangıcı. BUSY parametresi = 1, DONE = 0
7002	Ara çağrı (REQ ilgili değil): Yordam halihazırda aktif; BUSY, 1 değerine sahiptir.
8080	ID parametresiyle seçilen veri protokolü dosyası "DataLogClear" yordamıyla işlenemez.
8092	Veri protokolü mevcut değil.
80A2	Dosya sistemi tarafından geri bildirilen yazma hatası.
80B0	Veri protokolü açık değil.
80B4	Memory card yazma korumalıdır.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir. Görüntüleme formatlarının değiştirilmesiyle ilgili bilgiler için, bkz. "Ayrıca bkz".	

Tablo 9-211 DataLogClose yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogClose_DB"(req:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);**	Açık bir veri protokolü dosyasını kapatır. Kapalı bir veri protokolü için yürütülen bir DataLogWrite yordamı bir hataya neden olur. Bu veri protokolü için kullanılan yazma yordamlarına ancak bir DataLogOpen yordamı yürütüldükten sonra izin verilir. STOP işletim durumuna geçildikten sonra, açık durumdaki tüm veri protokolü dosyaları kapatılır. Yordamı eklerseniz, STEP 7 ilgili örnek DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "DataLogClose_DB", örnek DB'nin adıdır.

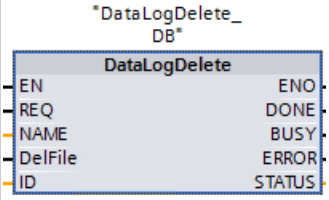
Tablo 9-212 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Standart değer: Yanlış)
ID	Geçiş	DWord	Bir veri protokolünün sayısal kodu. DataLogClose yordamında sadece giriş olarak kullanılır. (Standart değer: 0) **Not:** Bu parametreye sembolik bir ad üzerinden erişime izin verilmez.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - İşlenen yordam yok • 1 - İşlem işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Tablo 9-213 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	0001	Veri protokolü açık değil
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8092	Veri protokolü mevcut değil.

Tablo 9-214 DataLogDelete yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogDelete_DB"(REQ:=_bool_in_, NAME:=_variant_in_, DelFile:=_bool_in_, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, ID:=_dword_inout_);**	"DataLogDelete" yordamıyla bir veri protokolü dosyasını silebilirsiniz. Veri protokolü ve içinde bulunan veri setleri, ancak bu bileşenler "DataLogCreate" veya "DataLogNewFile" yordamıyla oluşturulduysa silinebilir.

"DataLogDelete_DB", örnek DB'nin adıdır.

Parametre

Aşağıdaki tabloda, "DataLogDelete" yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L, T, C veya sabit (T ve C sadece S7-1500'de LAD ve FBD'de kullanılabilir)	Yordamın, yükselen bir kenar durumunda yürütülmesi.
NAME	Input	VARIANT	L, D	Veri protokolünün dosya adı
DELFILE	Input	BOOL	E, A, M, D, L veya sabit	• 0: Veri protokolü kaydedilir. • 1: Veri protokolü silinir.
ID	InOut	DWORD	E, A, M, D, L	Veri protokolünün sayısal kodu
DONE	Output	BOOL	E, A, M, D, L	Yordam başarıyla yürütüldü.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
BUSY	Output	BOOL	E, A, M, D, L	Veri protokolünü silme işlemi henüz tamamlanmadı.
ERROR	Output	BOOL	E, A, M, D, L	• 0: Hata yok. • 1: Yordamın yürütülmesi sırasında bir hata oluştu. Ayrıntılı bilgiler, STATUS parametresinde yayınlanır.
STATUS	Output	WORD	E, A, M, D, L	Durum parametresi Parametre, sadece bir çağrının süresi boyunca ayarlanır. Durumu görüntülemek için, bu nedenle STATUS parametresini boş bir veri alanına kopyalamanız gerekir.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

Kaydedilen veri protokolleri

Bir veri protokolünü kaydederseniz, bu veri protokolü CPU tarafından devre dışı bırakılır ve kullanıcı programı artık buna erişemez. Veri protokolünün temsil ettiği dosya, SIMATIC memory card'da kalır veya memory card mevcut değilse dahili yükleme belleğinde (Sayfa 478) kalır. Dosya, Web sunucusu veya SIMATIC Automation Tool aracılığıyla daha önce olduğu gibi erişilebilir durumdadır. Dosyaya Web sunucusu üzerinden erişmek için veri protokollerinin (Sayfa 833) dosya tarayıcısını (Sayfa 838) ve standart web sitelerini (Sayfa 802) kullanabilirsiniz. Silinmiş bir veri protokolünü ancak bir SIMATIC memory card kullanılıyorsa saklamanız gerekir. CPU'nun dahili belleğine kaydedilmiş olan silinen veri protokolleri, yükleme belleğinin hesaplamalarında dikkate alınmaz.

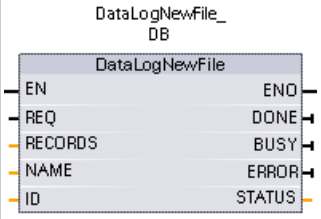
NAME ve ID parametreleri

Silinecek veri protokolünü NAME ve ID parametreleriyle seçin. ID parametresi önce değerlendirilir. İlgili ID'ye sahip bir veri protokolü mevcutsa, NAME parametresi değerlendirilmez. ID parametresinde 0 değeri kullanılırsa, NAME parametresinde, STRING veri tipine sahip bir değer kullanılmalıdır.

RET_VAL parametresi

Hata kodu* (W#16#...)	Anlamı
0	Hata yok.
7000	Devam eden bir görev işlemesi yok.
7001	Görev işlemesinin başlangıcı. BUSY parametresi = 1, DONE = 0
7002	Ara çağrı (REQ ilgili değil): Yordam halihazırda aktif; BUSY, 1 değerine sahiptir.
8091	NAME parametresinde STRING dışında başka bir veri tipi kullanılır.
8092	Veri protokolü mevcut değil.
80A2	Dosya sistemi tarafından geri bildirilen yazma hatası.
80B4	Memory card yazma korumalıdır.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir. Görüntüleme formatlarının değiştirilmesiyle ilgili bilgiler için, bkz. "Ayrıca bkz".	

Tablo 9-215 DataLogNewFile yordamı

LAD/FBD	SCL	Açıklama
	**"DataLogNewFile_DB"(req:=_bool_in_, records:=_udint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, name=:_DataLog_out_, ID:=_dword_inout_);**	Programınızın, mevcut bir veri protokolü dosyasını temel alarak yeni bir veri protokolü dosyası oluşturmasına olanak sağlar. Yordamı eklerseniz, STEP 7 ilgili örnek DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "DataLogNewFile_DB", örnek DB'nin adıdır.

Tablo 9-216 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Standart değer: Yanlış)
RECORDS	IN	UDInt	En eski girişin üzerine yazılmadan önce yuvarlak şeklindeki veri protokolünü içerebilen maksimum veri seti sayısı. (Standart değer: 1) Başlık satırı içeren veri seti, bu sayıya dahil değildir. Veri protokolünün başarılı bir şekilde oluşturulması için yeterli miktarda CPU yükleme belleği mevcut olmalıdır.
NAME	IN	Variant	Veri protokolünün adı: Bir ad girin. Bu varyant, sadece String veri tipini destekler ve sadece yerel belleğe, DB veya M belleğine veya yerel belleğe kaydedilebilir. (Standart değer: ' ') Bu karakter dizisi, veri protokolü dosyasının adı için de kullanılır. Adda bulunan karakterler, Windows dosya sisteminin ad kısıtlamalarına uymalıdır. \ / : * ? " < > | karakterlerine ve boşluk karakterine izin verilmez.)

Parametre ve veri tipi		Veri tipi	Açıklama
ID	Geçiş	DWord	Veri protokolünün sayısal kodu (standart değer: 0): • Yürütme işlemi sırasında giriş ID'si geçerli bir veri protokolünü belirtir. Yeni veri protokolü konfigürasyonu, bu veri protokolünden kopyalanır. • Yürütme işleminde sonra parametre ID'si, yeni oluşturulan veri protokolü dosyasının ID'sini yayınlayan bir çıkışa dönüşür. **Not:** Bu parametreye sembolik bir ad üzerinden erişime izin verilmez.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - İşlenen yordam yok • 1 - İşlem işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

DataLogNewFile yordamını, bir veri protokolü doluysa veya dolmak üzereyse ve veri protokolüne kaydedilen hiçbir veriyi kaybetmek istemiyorsanız yürütebilirsiniz. Dolu veri protokolü dosyası temel alınarak yeni, boş bir veri protokolü dosyası oluşturulabilir. Başlık veri seti, başlangıçtaki veri protokolünün özellikleriyle (veri seti tamponunun DATA ayarları, veri formatı ve zaman notu) başlangıçtaki veri protokolünden çoğaltılır. Başlangıçtaki veri protokolü dosyası, dolaylı olarak kapatılır ve yeni veri protokolü dosyası dolaylı olarak açılır.

DataLogWrite'ın parametreler aracılığıyla tetiklenmesi: Programınız, her DataLogWrite yordamının ERROR ve STATUS parametrelerini denetlemelidir. Son veri seti yazıldıysa ve bir veri protokolü doluysa, DataLogWrite yordamının ERROR Bit'i = 1 ve DataLogWrite yordamının STATUS veri sözcüğü = 1 olur. Bu ERROR ve STATUS değerleri sadece bir çevrim boyunca geçerlidir, bu nedenle denetleme mantığınız, STATUS değerini belirlemek ve ardından STATUS = 1 açısından (veri protokolü dolu) kontrol etmek için ERROR = 1'i zaman aralığı olarak kullanmalıdır.

DataLogNewFile'ın çalışma şekli: Program lojiğiniz "Veri protokolü dolu" sinyalini alırsa, bu durumla bir DataLogNewFile yordamı etkinleştirilir. DataLogNewFile yordamını, mevcut (genelde dolu) ve açık durumdaki bir veri protokolünün ID'siyle yürütmeniz gerekir, ancak NAME parametresi yeni ve benzersiz olmalıdır. DataLogNewFile yordamı yürütüldükten sonra, veri protokolü için, yeni veri protokolünün adına karşılık gelen yeni bir ID değeri yayınlanır (çıkış parametresi olarak). Yeni veri protokolü dosyası, dolaylı olarak açılır ve yeni veri setleri kaydedebilir. Yeni veri protokolü dosyasına yönelik yeni DataLogWrite yordamları, DataLogNewFile yordamı tarafından yayınlanan ID değerini kullanmalıdır.

DİKKAT
Bir yordamın veri protokolünü yazma işlemini başlatabilmek için veri protokolü oluşturma işlemi tamamlanmış olmalıdır Veri protokollerinin oluşturulmasına yönelik DataLogCreate ve DataLogNewFile işlemleri, birçok program çevrimini kapsar. Protokol dosyasının oluşturulması için ihtiyaç duyulan gerçek süre, veri setlerinin veri seti yapısına ve sayısına bağlıdır. Program lojiğiniz, bir protokol dosyası oluşturma işleminin tamamlandığını bildiren DONE Bit'lerinin DOĞRU olarak değişimini denetlemeli ve kaydetmelidir. Veri protokolü oluşturma işlemi tamamlanmadan önce bir DataLogWrite yordamı yürütülürse, yazma işlemiyle, yeni protokol veri seti beklendiği gibi yazılamaz.

Tablo 9-217 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmadan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarıyla ilk çağrı (işleniyor): BUSY = 1, DONE = 0
0	7002	N· çağrı (işleniyor): BUSY = 1, DONE = 0
1	8070	Tüm dahili örnek belleği dolu.
1	8090	Geçersiz dosya adı
1	8091	Ad parametresi, karakter dizisi değildir.
1	8092	Veri protokolü mevcut değil.
1	8093	Veri protokolü zaten mevcut.
1	8097	İstenen dosya uzunluğu, dosya sistemi için kullanılan maksimum dosya boyutunu aşıyor.
1	80B2	Başka hiçbir kaynak ID'si yok Not: Yeni bir veri protokolü için yer açmak için bazı mevcut veri protokollerini silin.
1	80B3	Yükleme belleği yeterli değil.
1	80B4	MC yazma korumalıdır.
1	80C1	Çok fazla açık dosya mevcut.

9.9.2.3 Veri protokolleriyle çalışma

Veri protokolü dosyaları, mevcut Flash belleğe CSV formatında (virgülle ayrılan değerler) kaydedilir. Veri protokollerini PLC Web sunucusu aracılığıyla veya PLC cihazının memory card'ı çıkartarak ve klasik bir PC kart okuyucusuna takarak görüntüleyebilirsiniz.

Veri protokollerini PLC Web sunucusu üzerinden görüntüleme

PLC cihazının PROFINET bağlantısı ve bir PC ağa bağlıysa, entegre PLC Web sunucusuna erişmek için Microsoft Internet Explorer veya Mozilla Firefox gibi bir PC web tarayıcısı kullanabilirsiniz. PLC cihazı, siz PLC Web sunucusunu kullanırken RUN veya STOP işletim türünde olabilir. PLC cihazı RUN işletim durumundaysa, PLC Web sunucusu ağ üzerinden protokol verilerini aktarırken kumanda programınız yürütülmeye devam eder.

Web sunucusuna erişim:

1. Web sunucusunu, hedef CPU'nun cihaz konfigürasyonunda etkinleştirin (Sayfa 798).
2. PC'nizi PROFINET ağı üzerinden hedef sisteme bağlayın (Sayfa 802).
3. Entegre Web sunucusu üzerinden CPU'ya erişin (Sayfa 808).
4. "Dosya tarayıcısı" standart web sitesi (Sayfa 838) ile veri protokolü dosyalarını yükleyin, düzenleyin ve silin.
5. .csv dosyasını Excel gibi bir elektronik tablo programıyla açın.

NOT

Veri protokollerinin yönetimi

Bir dosya sistemine maksimum 1000 veri protokolü kaydedin. Bu sayı aşıldığında, Web sunucusu, veri protokollerini görüntülemek için artık yeterli CPU kaynağına sahip olamayabilir.

Dosya tarayıcısının web sitesi veri protokollerini görüntüleyemediğini fark ederseniz, veri protokollerini görüntüleyebilmek ve silebilmek için CPU'yu STOP işletim durumuna getirmeniz gerekir.

Veri protokollerinizin yönetimi aracılığıyla, sadece gereken sayıda veri protokolü kaydettiğinizden ve 1000 veri protokollük maksimum sayıyı aşmadığınızdan emin olun.

PLC cihazının bir memory card'daki veri protokollerini görüntüleme

S7-1200 CPU'suna "Program kartı" tipinde bir memory card takılıysa, bu memory card'ı çıkartabilir ve bir PC'deki veya PG'deki SD (Secure Digital) veya MMC (MultiMediaCard) gibi bir standart kart yuvasına takabilirsiniz. PLC cihazı, memory card çıkartılırken STOP işletim durumundadır ve kumanda programınız yürütülmez.

Ardından, dosyalar için Windows Gezginini açın ve memory card'daki \Veri protokolü dizinine gidin. Bu dizinde, tüm CSV veri protokolü dosyalarınız bulunur.

Veri protokolü dosyalarını PC'nizdeki yerel bir sürücüye kopyalayın. Ardından bir CSV dosyasının yerel bir kopyasını Excel'le açabilirsiniz; bu, memory card'a kaydedilen orijinal dosya değildir.

DİKKAT
Veri protokolü dosyalarını bir PC kart okuyucu aracılığıyla bir S7-1200 memory card'dan kopyalayabilir, ancak değiştiremez veya silemezsiniz. Veri protokolü dosyalarını görüntülemek, indirmek (kopyalamak) ve silmek için Web sunucusunun standart dosya tarayıcısının kullanılması önerilir. Windows Gezgininde memory card'ın dosya sistemini doğrudan çağırmak, yanlışlıkla veri protokolü dosyalarını veya başka sistem dosyalarını silmeniz veya değiştirmeniz, bu nedenle muhtemelen bir dosyanın zarar görmesi veya memory card'ın kullanılamaz hâle gelmesi tehlikesini barındırır.

DİKKAT
Veri protokollerinin memory card'larının üzerindeki etkisi Sisteminizin genel davranışını ve sağlamlığını sağlamak için veri protokolü hızını maksimum her 200 msn'de bir şeklinde bir değerle sınırlayın.

9.9.2.4 Veri protokolü dosyalarının boyutu için limit

Veri protokolü dosyaları, PLC cihazının yükleme belleğini program, program verileri, konfigürasyon verileri, kullanıcı tanımlı web siteleri ve PLC sistem verileriyle birlikte kullanır. Dahili yükleme belleğini kullanan büyük bir program, yükleme belleğinde çok yer kaplar. Bu nedenle muhtemelen veri protokolü dosyaları için yeterli boş bellek alanı mevcut olmaz. Bu durumda, yükleme belleğinin kapasitesini arttırmak için bir memory card'ı "program kartı" (Sayfa 118) olarak kullanabilirsiniz. S7-1200 CPU'ları dahili veya harici yükleme belleğini kullanabilir, ancak her ikisini de eş zamanlı olarak kullanamaz.

Veri protokolü dosyalarının maksimum boyutuyla ilgili kural

Bir veri protokolü dosyasının maksimum boyutu, boş yükleme belleğinin boyutunu veya 500 Megabayt'ı (hangisi daha küçükse) aşamaz. 500 Megabayt'lık boyut bu durumda Megabayt'ın ondalık tanımıyla bağlantılıdır, bununla birlikte veri protokolü dosyalarının maksimum boyutu 500.000.000 Bayt veya 500 x 1000^2 Bayt olur.

Tablo 9-218 Yükleme belleğinin boyutu

Veri alanı	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C, CPU 1217C	Veri kaydetme
Dahili yükleme belleği Flash bellek	1 MB	2 MB	4 MB	4 MB	Kullanıcı programı ve program verileri, konfigürasyon verileri, veri protokolleri, kullanıcı tanımlı web siteleri ve PLC sistem verileri
Harici yükleme belleği "Program kartı" olarak opsiyonel memory card	SD kartın boyutuna bağlı olarak 4 MB, 12 MB, 24 MB, 256 MB, 2 GB veya 32 GB				

Boş yükleme belleğinin belirlenmesi

İşletim sistemi sürekli olarak belleğe ihtiyaç duyduğu ve belleği tekrar serbest bıraktığı için, boş yükleme belleğinin kapasitesi, normal işletimde değişiklik gösterir. Yükleme belleğinin boyutunu görüntülemek için aşağıdaki yöntemi izleyin.

1. STEP 7 ile S7-1200 hedef sistemi arasında bir çevrimiçi bağlantı kurun.
2. Veri protokolü yordamlarınıza kumanda eden programı yükleyin.
3. Opsiyonel olarak ihtiyaca göre kullanıcı tanımlı web siteleri oluşturun. Veri protokollerine erişim için kullanılan standart web siteleri, PLC cihazının Firmware'ine kaydedilir ve yükleme belleğinde yer kaplamaz.
4. Çevrimiçi fonksiyonlar ve teşhis fonksiyonları (Sayfa 1113) ile veya Web sunucusunun teşhis sayfası (Sayfa 814) ile yükleme belleğinin toplam boyutunu ve boş bellek alanını sorgulayın.

Bir veri protokolü dosyasının boyutunun hesaplanması (tüm veri setleri)

CPU, veri protokolü dosyası oluşturulurken maksimum bellek boyutunu atar. Tüm veri setleri için gereken boyutun yanı sıra, bir veri protokolü Header'ı (kullanılıyorsa), zaman notu Header'ı (kullanılıyorsa), veri seti indeksi Header'ı için kullanılan bellek alanını ve bellek ataması için kullanılan minimum blok boyutunu dikkate almanız gerekir.

Veri protokolü dosyalarınızın boyutunu aşağıdaki formül yardımıyla belirleyin ve maksimum boyutla ilgili kurala uymaya dikkat edin.

Veri protokolünün veri Bayt'ları = ((bir veri setindeki veri Bayt'ları + zaman notu Bayt'ları + 12 Bayt) * veri setlerinin sayısı)

Header

Veri protokolünün Header Bayt'ları = Header karakter Bayt'ları + 2 Bayt

Header karakter Bayt'ları

- Veri Header'ı yok ve zaman notu yok = 7 Bayt
- Veri Header'ı yok ve zaman notu (zaman notu Header'ı ile birlikte) = 21 Bayt
- Veri Header'ı ve zaman notu yok = Ayırma işareti virgüller de dahil olmak üzere, tüm sütun başlıklarındaki karakter Bayt'larının sayısı
- Veri Header'ı ve zaman notu (zaman notu Header'ı ile) = Ayırma işareti virgüller de dahil olmak üzere, tüm sütun başlıklarındaki karakter Bayt'larının sayısı + 21 Bayt

Veriler

Veri protokolünün veri Bayt'ları = ((bir veri setindeki veri Bayt'ları + zaman notu Bayt'ları + 12 Bayt) * veri setlerinin sayısı)

Bir veri setindeki veri Bayt'ları

DataLogCreate yordamının DATA parametresi, bir protokol veri seti için kullanılan veri alanlarının sayısını ve her veri alanının veri tipini atayan bir yapıyı gösterir.

İlgili veri tipinin sayısının bu veri tipi için gereken Bayt sayısıyla çarpın. Aynı işlemi, bir veri setindeki her veri tipi için tekrarlayın ve bir veri setindeki tüm veri elemanlarının toplamını elde etmek için tüm veri Bayt'larını toplayın.

Münferit veri elemanlarının boyutu

Veri protokolü dosyalarındaki veriler, CSV formatında (virgülle ayrılan değerler) karakter Bayt'ları olarak kaydedilir. Aşağıdaki tabloda, her veri elemanını kaydetmek için gereken Bayt sayısı gösterilmektedir.

Veri tipi	Bayt sayısı (veriler artı bir virgül Bayt'ı dahil)
Bool	2
Bayt	5
Sözcük	7
DWord	12
Char	4

String	**Örnek 1**: MyString[10] Maksimum karakter dizisi boyutu 10 karakterle belirtilir. Metin karakterleri + boşluk karakterleriyle otomatik olarak doldurma = 10 Bayt Başlangıçtaki ve sondaki tırnak işaretleri + virgül karakterleri = 3 Bayt 10 + 3 = toplam 13 Bayt **Örnek 2**: Mystring2 Köşeli parantezlerde boyut belirtilmezse, standart olarak 254 Bayt atanır. Metin karakterleri + boşluk karakterleriyle otomatik olarak doldurma = 254 Bayt Başlangıçtaki ve sondaki tırnak işaretleri + virgül karakterleri = 3 Bayt 254 + 3 = toplam 257 Bayt
USInt	5
UInt	7
UDInt	12
SInt	5
Int	7
DInt	12
Real	16
LReal	25
Time	15
DTL	24

Bir veri protokolü dosyasındaki veri setlerinin sayısı

DataLogCreate yordamının RECORDS parametresi, bir veri protokolü dosyasındaki maksimum veri seti sayısını belirler.

Bir veri setindeki zaman notu Bayt'ları

- Zaman notu yok = 0 Bayt
- Zaman notu = 20 Bayt

9.9.2.5 Veri protokolleriyle ilgili örnek program

Veri protokolleri için kullanılan bu örnek program, dinamik bir prosesin tarama verilerini çağırmak için gerekli olan tüm program lojiğini göstermez, ancak veri protokolleri için kullanılan yordamların asıl çalışma şeklini gösterir. Kullanılan protokol dosyalarının yapısı ve sayısı, proses kontrolü gerekliliklerinize bağlıdır.

NOT

Veri protokolü dosyalarının genel kullanımı

- DataLogCreate ve DataLogNew yordamları yürütüldükten sonra veri protokolü dosyaları otomatik olarak açılır.
- PLC cihazı RUN işletim durumundan STOP işletim durumuna geçtikten veya PLC cihazı yeniden başlatıldıktan sonra veri protokolü dosyaları otomatik olarak kapatılır.
- Bir DataLogWrite yordamının yürütülebilmesi için bir veri protokolü dosyası açılmış olmalıdır.
- Eş zamanlı olarak maksimum sekiz veri protokolü dosyası açık olabilir. Sekizden fazla veri protokolü dosyası mevcut olabilir, ancak eş zamanlı olarak sekizden fazlası açık olamaz.

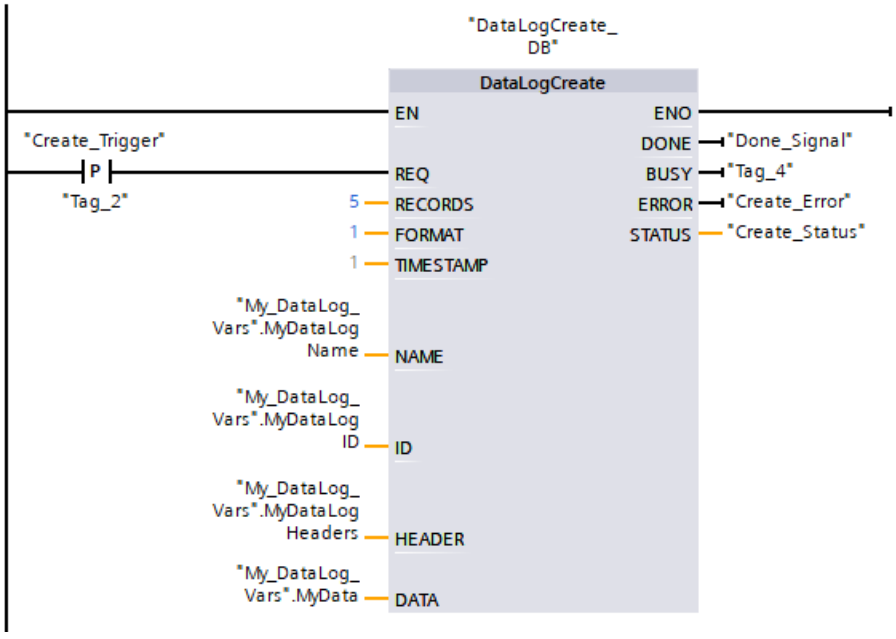
Veri protokolleriyle ilgili örnek program

Bu örnekte veri protokolü adları, başlık satırı metinleri ve MyData yapısı için bir veri bloğu kullanılmaktadır. Üç MyData değişkeni, yeni tarama değerlerini geçici olarak kaydeder. DataLogWrite yordamının yürütülmesi yoluyla, bu DB adreslerindeki proses tarama değerleri bir veri protokolü dosyasına aktarılır.

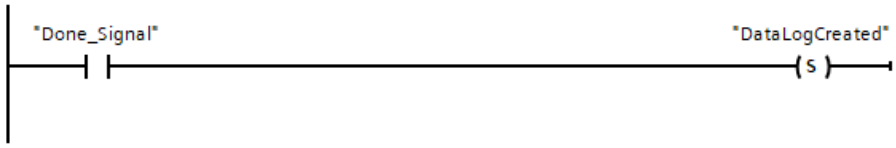
My_DataLog_Vars

	Name	Data type	Start value
1	Static		
2	MyNewDataLogName	String	'DataLog002'
3	MyDataLogName	String	'DataLog001'
4	MyDataLogID	DWord	0
5	MyDataLogHeaders	String	'Count,Temperature,Pressure'
6	MyData	Struct	
7	MyCount	Int	0
8	MyTemperature	Real	0.0
9	MyPressure	Real	0.0

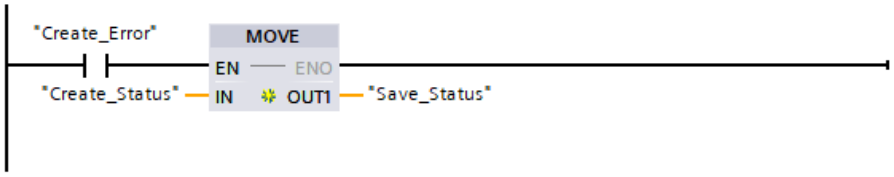
Ağ 1 "Create_Trigger" REQ'sundaki bir yükselen kenar, veri protokolü oluşturma işlemini başlatır.



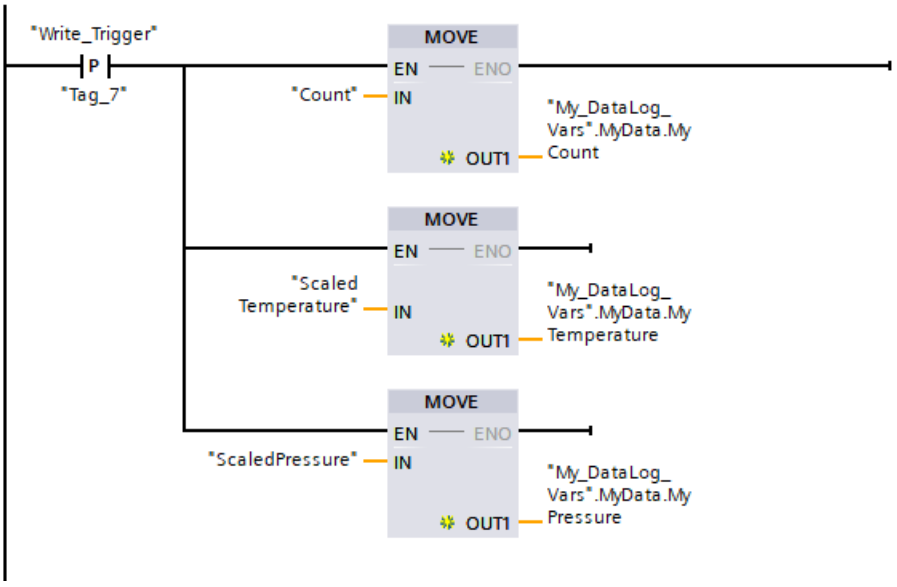
Ağ 2 Sadece bir çevrim boyunca geçerli olduğu için DataLogCreate yordamının DONE çıkışını belirleyin.



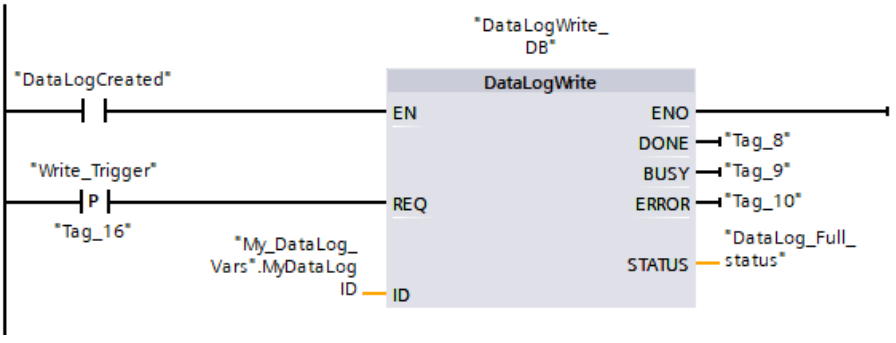
Ağ 3 Bir hata mevcutsa, durum çıkışını kaydedin.



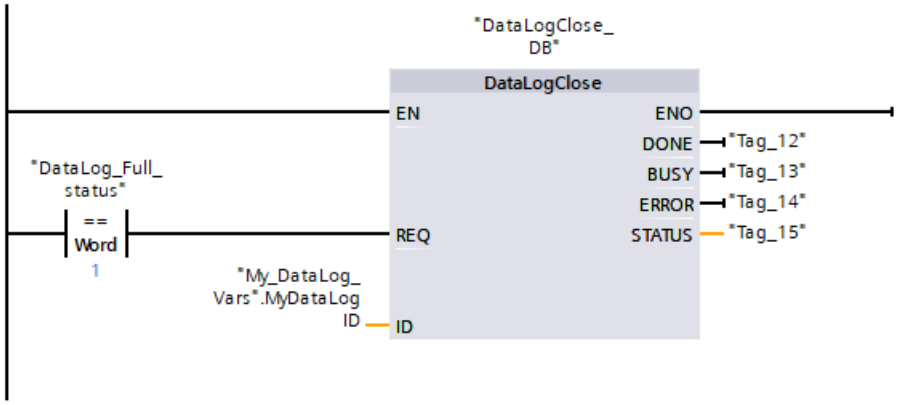
Ağ 4 "Write_Trigger"daki bir yükselen kenar, yeni proses değerlerinin MyData yapısına kaydedileceği zamanı tetikler.



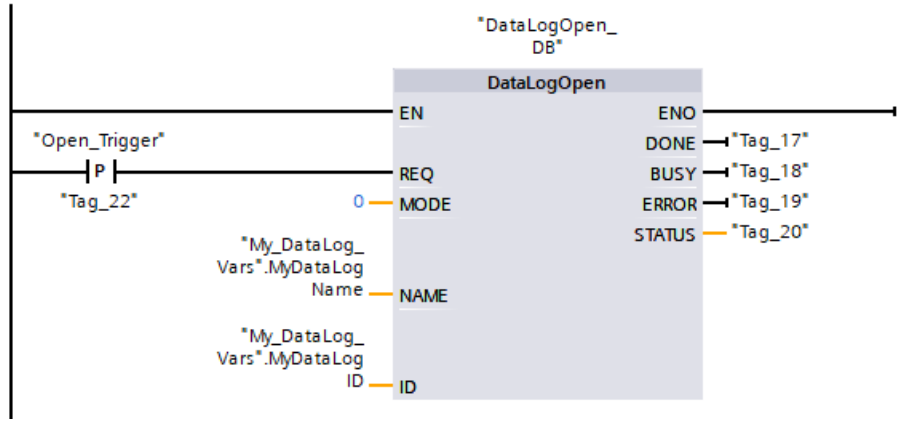
Ağ 5 EN girişinin durumu, DataLogCreate yordamının yürütülmesinin sonlandığı zamanı temel alır. Bir oluşturma işlemi birçok çevrimi kapsar ve bir yazma işleminin gerçekleştirilebilmesi için sonlandırılmış olmalıdır. REQ girişindeki yükselen kenar, bir yazma işlemini tetikleyen olaydır.



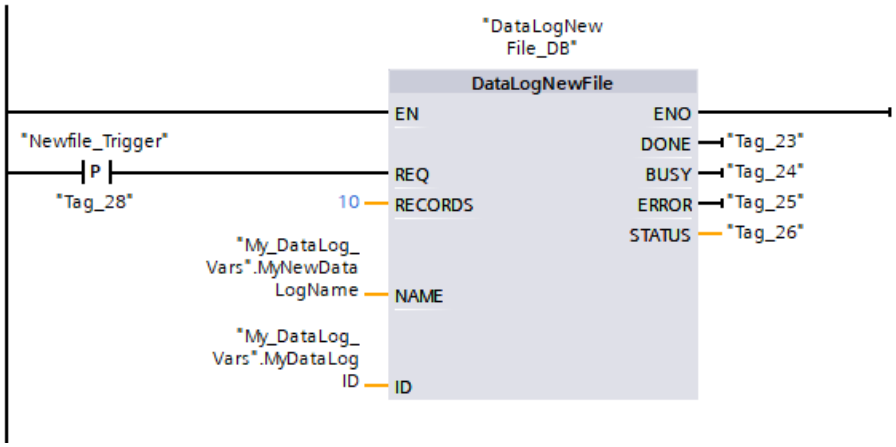
Ağ 6 Son veri setini yazan DataLogWrite işlemi yürütüldükten sonra veri protokolünü kapatın. DataLogWrite STATUS = 1 şeklindeki bir çıkış, protokol dosyasının dolu olduğunu bildirir.



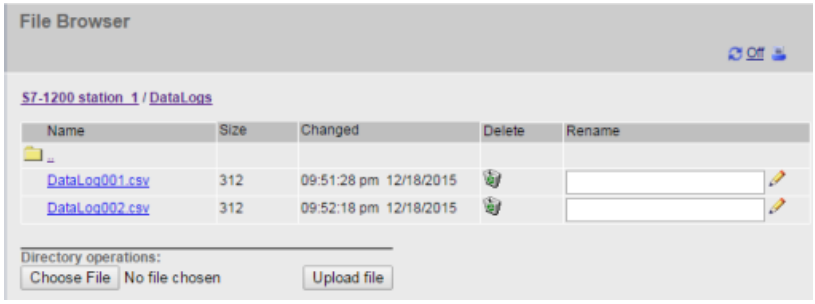
Ağ 7 DataLogOpen yordamının REQ girişindeki bir yükselen kenar, kullanıcının, bir HMI cihazında bulunan ve bir veri protokolü dosyasını açan bir tuşa basma işlemini simüle eder. Tüm veri setlerinin proses verileriyle dolu olduğu bir veri protokolü dosyasını açarsanız, DataLogWrite yordamının bir sonraki yürütülüşünde en eski veri setinin üzerine yazılır. Ancak eski veri protokolünü saklayabilir ve yeni bir veri protokolü oluşturabilirsiniz. Bu, ağ 7'de gösterilir.



Ağ 8 ID parametresi bir IN/OUT tipi parametredir. Önce, yapısını kopyalamak istediğiniz mevcut veri protokolünün ID değerini belirtin. DataLogNewFile yordamı yürütüldükten sonra, ID referansının adresine, yeni veri protokolü için yeni ve benzersiz bir ID değeri geri yazılır. DONE Bit'i = DOĞRU için gereken belirleme gösterilmez. DONE Bit'lerinin lojiğiyle ilgili bir örneği Ağ 1, 2 ve 4'te bulabilirsiniz.



S7-1200 CPU'sunun Web sunucusuyla görüntülenen veri protokolü dosyaları



① "Delete" opsiyonu sadece değiştirme yetkileriyle oturum açtığınız takdirde kullanılabilir.

② "Rename" opsiyonu sadece değiştirme yetkileriyle oturum açtığınız takdirde kullanılabilir.

Tablo 9-219 CSV dosyalarıyla ilgili Excel'de görüntülenen, indirilmiş örnekler

Maksimum beş veri seti içeren bir veri protokolüne yazılan veri seti 1 ve 2

	A	B	C	D	E	F
1	Record	Date	UTC Time	Count	Temperature	Pressure
2	1	3/14/2022	18:31:19	1	9.86E+01	3.52E+01
3	2	3/14/2022	18:36:34	2	1.00E+02	3.73E+01
4						
5						
6						
7						

Maksimum beş veri seti içeren bir veri protokolüne yazılan veri seti 1-5

	A	B	C	D	E	F
1	Record	Date	UTC Time	Count	Temperature	Pressure
2	1	3/14/2022	18:31:19	1	9.86E+01	3.52E+01
3	2	3/14/2022	18:36:34	2	1.00E+02	3.73E+01
4	3	3/14/2022	18:38:15	3	9.99E+02	3.68E+01
5	4	3/14/2022	18:38:51	4	9.95E+01	3.64E+01
6	5	3/14/2022	18:40:08	5	9.92E+01	3.74E+01
7						

Maksimum beş veri seti içeren bir veri protokolüne yazılan veri seti 2-6

Halihazırda dolu olan dosyaya başka bir veri seti yazıldıktan sonra, altıncı yazma işleminde veri seti 6'nın en eski veri setinin (1) üzerine yazılır. Sonraki bir yazma işleminde veri seti 7, veri seti 2'nin üzerine yazılır, vb.

	A	B	C	D	E	F
1	Record	Date	UTC Time	Count	Temperature	Pressure
2	6	3/14/2022	18:42:45	6	9.86E+01	3.58E+01
3	2	3/14/2022	18:36:34	2	1.00E+02	3.73E+01
4	3	3/14/2022	18:38:15	3	9.99E+02	3.68E+01
5	4	3/14/2022	18:38:51	4	9.95E+01	3.64E+01
6	5	3/14/2022	18:40:08	5	9.92E+01	3.74E+01
7						

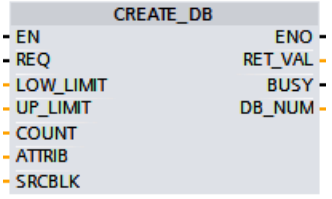
NOT

Dolu olmayan bir veri protokolünün sonunda başka bir //END markörü ayarlanmaz. S7-1200 CPU V4.1'ten önce dolu olmayan veri protokolleri bir //END markörü içeriyordu.

9.10 Veri bloğu kumandası

9.10.1 CREATE_DB (veri bloğu oluştur)

Tablo 9-220 CREATE_DB yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := CREATE_DB(** **REQ:=_bool_in_,** **LOW_LIMIT:=_uint_in_,** **UP_LIMIT:=_uint_in_,** **COUNT:=_udint_in_,** **ATTRIB:=_byte_in_,** **BUSY=>_bool_out_,** **DB_NUM=>_uint_out_);**	"CREATE_DB" yordamıyla yükleme belleğinde veya çalışma belleğinde yeni bir veri bloğu oluşturabilirsiniz. "CREATE_DB" yordamı, kullanıcı programının sağlama toplamını değiştirmez. Sadece çalışma belleğinde oluşturduğunuz bir veri bloğu şu özelliklere sahiptir: • Bir tam silme veya KAPATMA / AÇMA işleminden sonra bu blok artık mevcut olmaz. • Yükleme sırasında veya STOP işletim durumundan RUN işletim durumuna geçişte içerik değiştirilmez.

Veri bloğunun numarası

Oluşturulan veri bloğuna, LOW_LIMIT (alt limit) ve UP_LIMIT (üst limit) parametreleri tarafından tanımlanan bir aralıktaki bir numara atanır. "CREATE_DB" yordamı, DB'ye, belirtilen aralıktaki mümkün olan en küçük numarayı atar. Halihazırda kullanıcı programında bulunan DB'lerin numaralarını atayamazsınız.

Belirli bir numaraya sahip bir DB oluşturmak için, belirtilmesi gereken aralığın üst ve alt limit için aynı numarayı girin. Çalışma belleğinde ve/veya yükleme belleğinde bu numaraya sahip bir DB zaten mevcutsa veya DB, kopyalanmış sürüm olarak mevcutsa, yordam sonlandırılır ve RET_VAL parametresinde bir hata mesajı oluşturulur.

Veri bloğunun başlangıç değerleri

SRCBLK parametresiyle oluşturulacak DB'nin başlangıç değerlerini tanımlayabilirsiniz. SRCBLK parametresi, bir DB'yi veya başlangıç değerlerini aktardığınız bir DB aralığını gösteren bir işaretçidir. SRCBLK parametresinde adreslenen DB, standart erişimle oluşturulmuş olmalıdır ("Optimize edilmiş blok erişimi" özniteliği devre dışı bırakıldı).

- SRCBLK parametresinde belirtilen aralık, oluşturulan DB'den büyükse, oluşturulan DB'nin uzunluğuna kadar olan değerler başlangıç değerleri olarak aktarılır.
- SRCBLK parametresinde belirtilen aralık, oluşturulan DB'den küçükse, geri kalan değerler 0'la doldurulur.

Veri tutarlılığını sağlamak için, "CREATE_DB" yordamı yürütülürken (yani BUSY parametresi TRUE değerine sahip olduğu sürece) bu veri alanını değiştirmemeniz gerekir.

Fonksiyon açıklaması

"CREATE_DB" yordamı asenkron çalışır. İşleme süreci, birden fazla çağrı süresince gerçekleşir. Görevi, "CREATE_DB"yi REQ = 1 ile çağırarak başlatabilirsiniz.
RET_VAL ve BUSY çıkış parametreleri, görevin durumunu bildirir.
Ayrıca bkz.: DELETE_DB (veri bloğunu sil) (Sayfa 494)

Parametre

Aşağıdaki tabloda, "CREATE_DB" yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L veya sabit	Seviye tetiklemeli kumanda parametresi "request to activate" REQ = 1: Veri bloğunu oluşturma talebi
LOW_LIMIT	Input	UINT	E, A, M, D, L veya sabit	Bir DB numarasını atamak için kullanılan aralığın alt limit. Mümkün olan en küçük numara 60000'dir.
UP_LIMIT	Input	UINT	E, A, M, D, L veya sabit	DB'nize bir numara atamak için "CREATE_DB" tarafından kullanılan aralığın üst limit, (mümkün olan en büyük DB numarası: 60999)
COUNT	Input	UDINT	E, A, M, D, L veya sabit	Sayma değeri, oluşturulan DB için üretmek istediğiniz Bayt'ların sayısını belirtir. Bayt sayısı bir çift sayı olmalıdır. Maksimum uzunluk 65534 Bayt'tır.
ATTRIB	Input	BYTE	E, A, M, D, L veya sabit	Veri bloğunun özelliklerini tanımlamak için, ATTRIB parametresindeki Bayt'ın ilk 4 Bit'ini kullanıyorsunuz*:
				• Bit 0 = 0: "Sadece yükleme belleğine kaydet" özniteliği belirlenmedi. • Bit 0 = 1: "Sadece yükleme belleğine kaydet" özniteliği belirlendi. Bu ayarda DB, çalışma belleğinde bir bellek alanı kaplamaz ve programa kaydedilmez. DB'ye Bit komutları aracılığıyla erişilemez. Bit 0 = 1 ise, Bit 2 için yapılan seçim ilgili değildir. STEP 7 V5.x ile uyumluluğu sağlamak için Bit 0 ve 3 birlikte dikkate alınmalıdır (bkz. alt kısım).
				• Bit 1 = 0: "Cihazdaki veri bloğu yazma korumalı" özniteliği belirlenmedi. • Bit 1 = 1: "Cihazdaki veri bloğu yazma korumalı" özniteliği belirlendi.
				• Bit 2 = 0: DB kalıcıdır (sadece yükleme belleğinde ve çalışma belleğinde oluşturulan DB'lerde). En az bir değer kalıcı olarak tanımlandıysa DB kalıcı olarak kabul edilir. • Bit 2 = 1: DB kalıcı değil. Sadece yükleme belleğine veya sadece çalışma belleğine kaydedilen DB'lerde kalıcılık desteklenmez. "CREATE_DB" yordamını "kalıcı ve sadece yükleme belleği" veya "kalıcı ve sadece çalışma belleği" şeklindeki iki kombinasyondan biriyle çağırırsanız, oluşturulması gereken DB kalıcı olarak işaretlenmez.
				• Bit 3 = 0: DB'nin yükleme belleğinde veya çalışma belleğinde oluşturulması (Bit 0 aracılığıyla seçim yapılır, bkz. üst kısım) • Bit 3 = 1: DB'nin hem yükleme belleğinde hem de çalışma belleğinde oluşturulması (Bit 0 önemsiz) STEP 7 V5.x ile uyumluluğu sağlamak için Bit 0 ve 3 birlikte kullanılmalıdır. 3 = 1 ise, Bit 0 ilgili değildir.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama		
ATTRIB	Input	BYTE	E, A, M, D, L veya sabit	**Bit 0**	**Bit 3**	**DB oluşturma**
				0	0	Sadece çalışma belleğinde
				1	0	Sadece yükleme belleğinde
				İlgili değil	1	Çalışma belleği ve yükleme belleği
				• Bit 4 = 0 - Hiçbir başlangıç değeri belirtilmedi (SRCBLK parametresindeki giriş değerleri göz ardı edilir). • Bit 4 = 1 - Başlangıç değerlerini belirt (değerler, SRCBLK parametresi tarafından adreslenen DB'ye karşılık gelir).		
SRCBLK	Input	VARIANT	D	İşaretçiyi, değerleri, oluşturulacak veri bloğunun başlatılması için kullanılacak olan veri bloğunun üzerine getirin.		
RET_VAL	Return	INT	E, A, M, D, L	Hata bilgileri		
BUSY	Output	BOOL	E, A, M, D, L	BUSY = 1: İşlem henüz tamamlanmadı.		
DB_NUM	Output	DB_DYN (UINT)	E, A, M, D, L	Oluşturulan DB'nin numarası.		
* Burada seçilen özellikler, bir veri bloğunun özelliklerindeki özniteliklere karşılık gelir.						

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

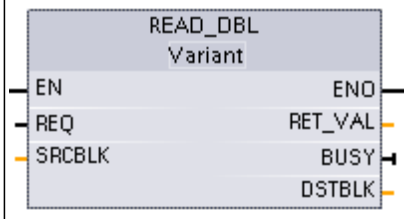
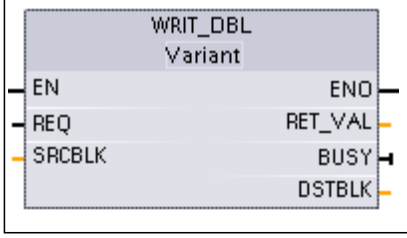
RET_VAL parametresi

Hata kodu* (W#16#...)	Açıklama
0000	Hata yok
0081	Hedef aralık, kaynak aralıktan büyüktür. Kaynak aralık, tümüyle hedef aralığa yazılır. Hedef aralığın geri kalan Bayt'ları değiştirilmez.
7000	REQ = 0 ile ilk çağrı: Hiçbir veri aktarımı aktif değil; BUSY, 0 değerine sahiptir.
7001	REQ = 1 ile ilk çağrı: Veri aktarımı başlatıldı; BUSY, 1 değerine sahiptir.
7002	Ara çağrı (REQ ilgili değil): Veri aktarımı halihazırda aktif; BUSY, 1 değerine sahiptir.
8081	Kaynak aralık, hedef aralıktan büyüktür. Hedef aralık tam olarak açıklanır. Kaynak aralığın geri kalan Bayt'ları göz ardı edilir.
8092	"Veri bloğu oluştur" fonksiyonu şu anda kullanılamıyor, çünkü • "Kullanıcı belleğini sıkıştır" fonksiyonu mevcut durumda aktiftir. • CPU'nuzdaki maksimum veri bloğu sayısına ulaşıldı.
8093	SRCBLK parametresi için hiçbir veri bloğu veya çalışma belleğinde bulunmayan hiçbir veri bloğu belirtilmedi.
8094	ATTRIB parametresinde geçersiz bir değer belirtildi.
80A1	DB hata numarası: • Numara 0. • Numara, DB numaraları için tanımlanan CPU'ya özel üst limiti aşıyor. • Alt limit > üst limit.
80A2	DB uzunluk hatası: • Uzunluk değeri 0. • Uzunluk, bir tek sayıdır. • Uzunluk, CPU için izin verilen değerden daha büyük.

Hata kodu* (W#16#...)	Açıklama
80A3	SRCBLK parametresindeki veri bloğu standart erişimle oluşturulmadı.
80B1	Hiçbir DB numarası boş değil.
80B2	Çalışma belleği yeterli değil.
80B4	Memory card yazma korumalıdır.
80BB	Yükleme belleği yeterli değil.
80C3	Eş zamanlı aktif olan "CREATE_DB" yordamlarının maksimum sayısına ulaşıldı.
Genel hata bilgileri	Ayrıca bkz.: Gelişmiş yordamlar için ortak hata kodları (Sayfa 501)
*Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenebilir.	

9.10.2 READ_DBL ve WRIT_DBL (yükleme belleğindeki veri bloğunu oku/veri bloğunu yükleme belleğine yaz) yordamları

Tablo 9-221 READ_DBL ve WRIT_DBL yordamları

LAD/FBD	SCL	Açıklama
	**READ_DBL(** **req:=_bool_in_,** **srcblk:=_variant_in_,** **busy=>_bool_out_,** **dstblk=>_variant_out_);**	DB başlangıç değerlerini veya değerlerin bir kısmını yükleme belleğinden çalışma belleğindeki bir hedef DB'ye kopyalar. Yükleme belleğinin içeriği, kopyalama işlemi sırasında değiştirilmez.
	**WRIT_DBL(** **req:=_bool_in_,** **srcblk:=_variant_in_,** **busy=>_bool_out_,** **dstblk=>_variant_out_);**	Güncel DB değerlerini veya değerlerin bir kısmını çalışma belleğinden yükleme belleğindeki bir hedef DB'ye kopyalar. Çalışma belleğinin içeriği, kopyalama işlemi sırasında değiştirilmez.

Tablo 9-222 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	BOOL	BUSY = 0 ise, yüksek seviyeli bir sinyal işlemi başlatır.
SRCBLK	IN	VARIANT	READ_DBL: İşaretçiyi yükleme belleğindeki kaynak veri bloğunun üzerine getirin WRIT_DBL: İşaretçiyi çalışma belleğindeki kaynak veri bloğunun üzerine getirin
RET_VAL	OUT	INT	Yürütme koşulu
BUSY	OUT	BOOL	BUSY = 1 ifadesi, okuma/yazma işleminin tamamlanmadığını bildirir.
DSTBLK	OUT	VARIANT	READ_DBL: İşaretçiyi çalışma belleğindeki hedef veri bloğunun üzerine getirin WRIT_DBL: İşaretçiyi yükleme belleğindeki hedef veri bloğunun üzerine getirin

Tipik olarak bir DB, hem yükleme belleğine (Flash), hem de çalışma belleğine (RAM) kaydedilir. Başlangıç değerleri (başlangıç değerleri) her zaman yükleme belleğine ve güncel değerler her zaman çalışma belleğine kaydedilir. READ_DBL ile bir başlangıç değeri setini yükleme belleğinden, programınız tarafından referanslanan çalışma belleğindeki bir DB'nin güncel değerlerine kopyalayabilirsiniz. WRIT_DBL, yükleme belleğine veya memory card'a kaydedilen güncel değerleri çalışma belleğinde güncellemenize olanak sağlar.

NOT

WRIT_DBL ve READ_DBL yordamlarının Flash bellek üzerindeki etkisi

WRIT_DBL yordamı, Flash bellekte yazma işlemleri gerçekleştirir (dahili yükleme belleğinde veya bir memory card'da). Flash belleğin kullanım ömrünün kısalmasını önlemek için, bir üretim prosesindeki değişikliklerin kaydedilmesi gibi güncellemelerin daha az sıklıkla yapılması amacıyla WRIT_DBL yordamını kullanın. Benzer nedenlerle, okuma işlemleri için READ_DBL yordamını sıkça çağırmaktan kaçının.

READ_DBL ve WRIT_DBL için kullanılan veri bloklarını, bu yordamlar çağrılmadan önce STEP 7 programında oluşturmanız gerekir. Kaynak DB'yi "Standard" tipinde bir veri bloğu olarak oluşturduysanız, hedef DB de yine "Standard" tipinde olmalıdır. Kaynak veri bloğu "Optimize edildi" tipinde bir blok olarak oluşturulduysa, hedef veri bloğu da yine "Optimize edildi" tipinde olmalıdır.

Standart DB'ler söz konusu ise, bir değişken adı veya bir P# değeri belirtebilirsiniz. P# değeriyle, belirtilen boyutta (Bayt, Word veya DWord) herhangi bir sayıda elemanı belirtebilir ve kopyalayabilirsiniz. Bu sayede, bir DB'nin bir kısmını veya tüm DB'yi kopyalayabilirsiniz. Optimize edilmiş DB'ler söz konusu ise, sadece bir değişken adı belirtebilirsiniz. P# operatörünü kullanamazsınız. Bir standart DB veya optimize edilmiş DB (veya başka türlerde çalışma bellekleri) için bir değişken adı belirtirseniz, yordam, bu değişken adının referans verdiği verileri kopyalar. Bu sırada; kullanıcı tanımlı bir tip, bir Array veya bir temel eleman söz konusu olabilir. Optimize edilmiş bir DB yerine bir standart DB söz konusu ise, bu yordamlarda sadece Struct veri tipini kullanabilirsiniz. Optimize edilmiş bellekte bir yapı söz konusu ise bir kullanıcı tanımlı veri tipi (UDT) kullanmanız gerekir. "Veri tiplerinin" hem kaynak yapı, hem de hedef yapı için aynı olması, ancak kullanıcı tanımlı bir tiple sağlanır.

NOT

Yapının (Struct veri tipi) "optimize edilmiş" bir DB'de kullanılması

Bir Struct veri tipini "optimize edilmiş" DB'lerle kullanırsanız, önce Struct veri tipi için kullanıcı tanımlı bir veri tipi (UDT) oluşturmanız gerekir. Ardından UDT veri tipinde kaynak ve hedef DB'yi yapılandırabilirsiniz. UDT veri tipi, Struct veri tipindeki veri tiplerinin her iki DB içinde tutarlı kalmasını sağlar.

"Standard" DB'lerde, bir UDT veri tipi oluşturmadan Struct veri tipini kullanabilirsiniz.

READ_DBL ve WRIT_DBL, program çevrimiyle asenkron şekilde yürütülür. İşleme süreci, birden çok READ_DBL ve WRIT_DBL çağrısı boyunca gerçekleşir. DB aktarma görevini REQ = 1'i çağırarak başlatın ve ardından, veri aktarımının ne zaman tamamlanacağını ve doğru olup olmadığını belirlemek için BUSY ve RET_VAL çıkışlarını denetleyin.

NOT

WRIT_DBL ve READ_DBL yordamlarının iletişim yükü üzerindeki etkisi

WRIT_DBL veya READ_DBL yordamı sürekli aktifse, bu sırada, STEP 7, CPU ile olan iletişimini kaybedecek kadar çok fazla iletişim kaynağı kullanabilir. Bu nedenle REQ parametresi için girişte bir yükselen kenar (Sayfa 208) kullanın ve birden fazla çevrim boyunca açık kalacak (sinyal seviyesi High) hiçbir NO kontak veya NC kontak girişi (Sayfa 204) kullanmayın.

Veri tutarlılığı sağlamak için, READ_DBL işlenirken hedef aralığı veya WRIT_DBL işlenirken kaynak aralığı değiştirmeyin (yani BUSY parametresi DOĞRU olduğu sürece).

SRCBLK ve DSTBLK parametrelerindeki kısıtlamalar:

- Bir veri bloğu, referanslanabilmesi için önceden oluşturulmuş olmalıdır.
- BOOL tipindeki bir VARIANT işaretçisinin uzunluğu 8'e bölünebilir olmalıdır.
- STRING tipindeki bir VARIANT işaretçisinin uzunluğu, kaynak ve hedef işaretçide aynı olmalıdır.

Reçeteler ve makine donanımıyla ilgili bilgiler

READ_DBL ve WRIT_DBL yordamlarıyla reçeteler veya makine donanımıyla ilgili bilgileri yönetebilirsiniz. Bu, çok sık değişmeyen değerler için kalıcı verileri kaydetmek amacıyla prensip olarak kullanılabilecek başka bir yöntemdir. Ancak Flash belleği aşırı zorlamamak için yazma işlemlerinin sayısını sınırlandırmanız gerekir. Bu şekilde, en azından sık değişmeyen değerler için kalıcı belleğin kapsamını, kalıcı verilerin normal kapsamının ötesine geçecek şekilde genişletebilirsiniz. WRIT_DBL yordamıyla, reçete bilgilerini veya makine donanımıyla ilgili bilgileri çalışma belleğinden yükleme belleğine kaydedebilir ve bu bilgileri, READ_DBL yordamıyla tekrar yükleme belleğinden çalışma belleğine çağırabilirsiniz.

Tablo 9-223 Koşul kodları

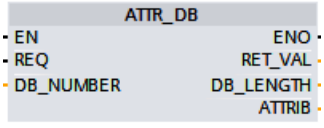
RET_VAL (W#16#...)	Açıklama
0000	Hata yok
0081	Uyarı: Kaynak aralık, hedef aralıktan küçüktür. Hedef aralıktaki ek Bayt'lar değiştirilmezken kaynak verileri tümüyle kopyalanır.
7000	REQ = 0 ile çağrı: BUSY = 0
7001	REQ = 1 ile ilk çağrı (işleniyor): BUSY = 1
7002	N· çağrı (işleniyor): BUSY = 1
8051	Veri bloğu tip hatası
8081	Kaynak aralık, hedef aralıktan büyüktür. Hedef aralık tamamen dolu ve kaynağın geri kalan Bayt'ları göz ardı edilir.
8251	Kaynak veri bloğu tip hatası
82B1	Eksik kaynak veri bloğu
82C0	Kaynak DB, başka bir yordam veya iletişim fonksiyonu tarafından düzenlenir.
8551	Hedef veri bloğu tip hatası

RET_VAL (W#16#...)	Açıklama
85B1	Eksik hedef veri bloğu
85C0	Hedef DB, başka bir yordam veya iletişim fonksiyonu tarafından düzenlenir.
80C3	Mevcut durumda, yürütme için kullanılan bekleme kuyruğunda 50'den fazla READ_DBL yordamı veya 50 WRIT_DBL yordamı bulunuyor.

Ayrıca bkz. Reçeteler (Sayfa 449-450)

9.10.3 ATTR_DB (bir veri bloğunun özniteliğini oku)

Tablo 9-224 ATTR_DB yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := ATTR_DB(REQ:=_bool_in_, DB_NUMBER:=_uint_in_, DB_LENGTH=>_udint_out_, ATTRIB=>_byte_out_);**	"ATTR_DB" yordamıyla, CPU'nun çalışma belleğindeki bir veri bloğuyla (DB) ilgili bilgileri çağırabilirsiniz. Yordam, seçilen DB için ATTRIB parametresinde ayarlanan özniteliği belirler. Optimize edilmiş erişime sahip veri blokları ve sadece yükleme belleğinde bulunan veri blokları için uzunluk okunamaz. Bu tür durumlarda DB_LENGH parametresinin değeri 0'dır. ATTR_DB'yi, optimize edilmiş erişime ve etkinleştirilmiş bellek rezervine sahip veri bloklarında uygulamayın. Hareket kumandası için kullanılan veri bloklarını "ATTR_DB" yordamıyla okumayın. Bu sırada 80B2 hata kodu yayınlanır.

Parametre

Aşağıdaki tabloda, "ATTR_DB" yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L veya sabit	REQ = 1: Blok özniteliklerini okuma talebi
DB_NUMBER	Input	DB_ANY	E, A, M, D, L veya sabit	Test edilecek DB'nin numarası
RET_VAL	Output	INT	E, A, M, D, L	Hata bilgileri
DB_LENGTH	Output	UDINT	E, A, M, D, L	• Seçilen DB'nin içerdiği veri Bayt'larının sayısı • Optimize edilmiş erişime sahip veri bloklarında ve sadece yükleme belleğinde bulunan veri bloklarında 0
ATTRIB	Output	BYTE	E, A, M, D, L	DB özellikleri:
				• Bit 0*= 0: "Sadece yükleme belleğine kaydet" özniteliği belirlenmedi. • Bit 0*= 1: "Sadece yükleme belleğine kaydet" özniteliği belirlendi.
				• Bit 1 = 0: "Cihazdaki veri bloğu yazma korumalı" özniteliği belirlenmedi. • Bit 1 = 1: "Cihazdaki veri bloğu yazma korumalı" özniteliği belirlendi.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
ATTRIB	Output	BYTE	E, A, M, D, L	Bit0 = 1 ise, Bit 2 ilgili değildir ve 1 değerini alır. • Bit 2 = 0: Kalıcı - En az bir değer kalıcı olarak tanımlandıysa DB kalıcı olarak kabul edilir. • Bit 2 = 1: Kalıcı değil - Tüm DB kalıcı değildir.
				• Bit 3*= 0: DB ya yükleme belleğinde (Bit 0 = 1) ya da çalışma belleğindedir (Bit 0 = 0). • Bit 3*= 1: DB, hem yükleme belleğinde hem de çalışma belleğinde oluşturulur.
* Bit 0 ile Bit 3 arasındaki ilişki, "CREATE_DB (veri bloğu oluştur) (Sayfa 486-487)" yordamının parametrelerinde açıklanmaktadır.				

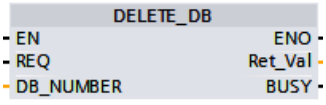
Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

RET_VAL parametresi

Hata kodu* (W#16#...)	Anlamı
0000	Hata yok.
80A1	DB_NUMBER giriş parametresinde hata: Gerçekten seçilen parametre • 0'dır • Kullanılan CPU için izin verilen maksimum DB numarasından daha büyüktür
80B1	Belirtilen numaraya sahip DB, CPU'da mevcut değil.
80B2	Hareket kumandası için kullanılan teknoloji nesnelerin veri blokları "ATTR_DB" yordamıyla okunamıyor.
Genel hata bilgileri	Ayrıca bkz.: Gelişmiş yordamlar için ortak hata kodları (Sayfa 501)
*Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenebilir.	

9.10.4 DELETE_DB (veri bloğunu sil)

Tablo 9-225 DELETE_DB yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := DELETE_DB(REQ := _bool_in_, DB_NUMBER := _uint_in_, BUSY => _bool_out_);**	"DELETE_DB" yordamıyla, kullanıcı programının "CREATE_DB (Sayfa 486-487)" yordamını çağırarak oluşturduğu bir veri bloğunu (DB) silebilirsiniz. Veri bloğu "CREATE_DB" ile oluşturulmadıysa, DELETE_DB , RET_VAL parametresinde W#16#80B5 hata kodunu yayınlar. DELETE_DB çağrısı, seçilen veri bloğunu hemen silmez, çevrim kontrol noktasındaki çevrim OB'si yürütüldükten sonra siler.

Fonksiyon açıklaması

"DELETE_DB" yordamı asenkron olarak çalışır, yani yürütülmesi birden fazla çağrıyı kapsar. Alarm aktarma işlemini, REQ = 1 ile yordamı çağırarak başlatabilirsiniz.

BUSY çıkış parametresi ve RET_VAL çıkış parametresinin Bayt 2 ve 3, görevin durumunu gösterir.

BUSY çıkış parametresi FALSE değerine sahipse, veri bloğu tümüyle silinmiştir.

Parametre

Aşağıdaki tabloda, "DELETE_DB" yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L veya sabit	REQ = 1: DB_NUMBER parametresindeki numaraya sahip DB'nin silinmesi talebi
DB_NUMBER	Input	UINT	E, A, M, D, L veya sabit	Silinmesi gereken DB'nin numarası
RET_VAL	Output	INT	E, A, M, D, L	Hata bilgileri (bkz. "RET_VAL" parametresi)
BUSY	Output	BOOL	E, A, M, D, L	BUSY= 1: İşlem henüz tamamlanmadı.

Geçerli veri tipleriyle ilgili daha fazla bilgi için, bkz. "Veri tipleri (Sayfa 99)".

RET_VAL parametresi

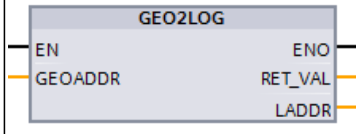
Hata kodu* (W#16#...)	Anlamı
0000	Hata yok.
7000	REQ = 0 ile ilk çağrı: Hiçbir veri aktarımı aktif değil; BUSY, 0 değerine sahiptir.
7001	REQ = 1 ile ilk çağrı: Veri aktarımı başlatıldı; BUSY, 1 değerine sahiptir.
7002	Ara çağrı (REQ ilgili değil): Veri aktarımı halihazırda aktif; BUSY, 1 değerine sahiptir.
80A1	DB_NUMBER giriş parametresinde hata: • Parametredeki değer 0'dır. • Parametredeki değer, kullanılan CPU için izin verilen maksimum DB numarasından daha büyüktür.
80B1	Belirtilen numaraya sahip DB, CPU'da mevcut değil.
80B4	CPU'nun memory card yazma korumalı olduğu için DB silinemiyor.
80B5	DB, "CREATE_DB" ile oluşturulmadı.
80BB	Yükleme belleği yeterli değil.
80C3	"DB'yi sil" fonksiyonu, şu anda kısa süreli bir kaynak sıkıntısı nedeniyle yürütülemiyor.
Genel hata bilgileri	Ayrıca bkz.: Gelişmiş yordamlar için ortak hata kodları (Sayfa 501)
*Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenebilir.	

9.11 Adres işleme

9.11.1 GEO2LOG (donanım kodunu yuva aracılığıyla belirle)

GEO2LOG yordamıyla, yuva bilgilerinin yardımıyla donanım kodu belirlenir.

Tablo 9-226 GEO2LOG yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := GEO2LOG(GEOADDR:=_variant_in_out_, laddr:=_word_out_);**	GEO2LOG yordamıyla, yuva bilgilerinin yardımıyla donanım kodu belirlenir.

GEO2LOG yordamıyla, GEOADDR sistem veri tipiyle ayarlanan yuva bilgilerinin yardımıyla donanım kodu belirlenir:

HWTYPE parametresinde belirlenen donanım tipine bağlı olarak, diğer GEOADDR parametrelerinin aşağıdaki bilgileri değerlendirilir:

- HWTYPE = 1'de (PROFINET IO sistemi):
  - Sadece IOSYSTEM değerlendirilir. GEOADDR'nin diğer parametreleri dikkate alınmaz.
  - PROFINET IO sisteminin donanım kodu yayınlanır.
- HWTYPE = 2'de (PROFINET IO cihazı):
  - IOSYSTEM ve STATION değerlendirilir. GEOADDR'nin diğer parametreleri dikkate alınmaz.
  - PROFINET IO cihazının donanım kodu yayınlanır.

- HWTYPE = 3'te (modül taşıyıcısı):
  - Sadece IOSYSTEM ve STATION değerlendirilir. GEOADDR'nin diğer parametreleri dikkate alınmaz.
  - Modül taşıyıcısının donanım kodu yayınlanır.
- HWTYPE = 4'te (modül):
  - IOSYSTEM, STATION, ve SLOT yayınlanır. GEOADDR'nin SUBSLOT parametresi dikkate alınmaz.
  - Modülün donanım kodu yayınlanır.
- HWTYPE = 5'te (alt modül):
  - GEOADDR'nin tüm parametreleri değerlendirilir.
  - Alt modülün donanım kodu yayınlanır.

GEOADDR sistem veri tipinin AREA parametresi değerlendirilmez.

Tablo 9-227 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		**Veri tipi**	**Açıklama**
GEOADDR	IN/OUT or IN ?	Variant	İşaretçiyi, GEOADDR sistem veri tipinin yapısının üzerine götürün. GEOADDR sistem veri tipi, donanım ID'sinin belirlenmesinde kullanılan yuva bilgilerini içerir. Daha fazla bilgi için bkz. "GEOADDR sistem veri tipi (Sayfa 500)".
RET_VAL	OUT or RETURN ?	Int	Hata bilgilerinin yayınlanması.
LADDR	OUT	HW_ANY	Parçanın veya modülün donanım kodu. Numara otomatik olarak atanır ve donanım konfigürasyonunun özelliklerine kaydedilir.

Geçerli veri tipleriyle ilgili daha fazla bilgiyi, STEP 7 online yardımındaki "Geçerli veri tiplerine genel bakış" başlığında bulabilirsiniz.

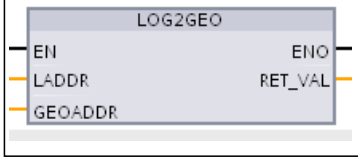
Tablo 9-228 Koşul kodları

RET_VAL* (W#16#...)	**Açıklama**
0	Hata yok.
8091	HWTYPE için kullanılan GEOADDR'de geçersiz değer.
8094	IOSYSTEM için kullanılan GEOADDR'de geçersiz değer.
8095	STATION için kullanılan GEOADDR'de geçersiz değer.
8096	SLOT için kullanılan GEOADDR'de geçersiz değer.
8097	SUBSLOT için kullanılan GEOADDR'de geçersiz değer.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.	

9.11.2 LOG2GEO (yuvayı donanım kodu aracılığıyla belirle)

LOG2GEO yordamıyla, bir donanım kodunun mantıksal adresi aracılığıyla coğrafi adresi (modül yuvası) belirleyebilirsiniz.

Tablo 9-229 LOG2GEO yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := LOG2GEO(** **laddr:=_word_in_,** **GEOADDR:=_variant_in_out_);**	LOG2GEO yordamıyla, bir donanım koduna ait modül yuvasını belirleyebilirsiniz.

LOG2GEO yordamı, donanım kodu aracılığıyla bir mantıksal adrese ait coğrafi adresi belirler:

- LADDR parametresiyle, donanım kodu aracılığıyla mantıksal adresi seçebilirsiniz.
- GEOADDR, LADDR girişinde belirtilen mantıksal adrese ait coğrafi adresi içerir.

NOT

HW tipi bir bileşeni desteklemiyorsa, bir modül 0 için bir alt yuva numarası yayınlanır.

LADDR girişi hiçbir HW nesnesini adreslemiyorsa bir hata bildirilir.

Tablo 9-230 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
LADDR	IN	HW_ANY	IO sisteminin veya modülün donanım kodu. Numara otomatik olarak atanır ve CPU'nun veya donanım konfigürasyonu arayüzünün özelliklerine kaydedilir.
RET_VAL	OUT	Int	Yordamın hata kodu
GEOADDR	IN_OUT	Variant	İşaretçiyi, GEOADDR sistem veri tipinin üzerine götürün. GEOADDR sistem veri tipi, yuva bilgilerini içerir. Daha fazla bilgi için bkz. "GEOADDR sistem veri tipi (Sayfa 500)".

Geçerli veri tipleriyle ilgili daha fazla bilgi için, STEP 7 online yardımındaki "Geçerli veri tiplerine genel bakış" başlığına bakın.

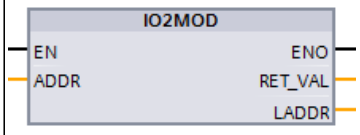
Tablo 9-231 Koşul kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	LADDR parametresinde belirtilen adres geçersizdir.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.	

9.11.3 IO2MOD (donanım kodunu bir G/Ç adresi aracılığıyla belirle)

IO2MOD yordamıyla, bir alt modülün G/Ç adresi aracılığıyla bir modülün donanım kodu belirlenir.

Tablo 9-232 IO2MOD yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := IO2MOD(ADDR:=_word_in_, LADDR:=_word_out_);**	IO2MOD yordamıyla, bir donanım koduna ait modül yuvasını belirleyebilirsiniz.

IO2MOD ile bir alt modülün G/Ç adresi (E, A, PE, PA) aracılığıyla modülün donanım kodu belirlenir.

G/Ç adresini ADDR parametresine girin. Bu parametre için birden çok G/Ç adresi kullanılırsa, donanım kodunun belirlenmesi için sadece ilk adres değerlendirilir. İlk adres doğru belirtilirse, ADDR'deki adres bilgisinin uzunluğu önemli olmaz. Birden çok modülü veya kullanılmayan adresleri kapsayan bir adres aralığı belirtilirse, ilk modülün donanım kodu da belirlenebilir.

ADDR parametresinde bir (alt) modülün hiçbir G/Ç adresi belirtilmezse, RET_VAL parametresinde "8090" hata kodu yayınlanır.

NOT

SCL'de G/Ç adresi girişi

"%QWx:P" G/Ç erişim ID'si ile SCL programlaması mümkün değildir. Bu durumda, sembolik değişken adı veya proses görünümündeki mutlak adres kullanılmalıdır.

Tablo 9-233 Parametreler için kullanılan veri tipleri

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
ADDR	IN or IN/OUT ?	Variant	E, A, M, D, L	Bir alt modüldeki G/Ç adresi (E, A, PE, PA). ADDR parametresi için Slice erişiminin kullanılmamasına dikkat edin. Bu durumda, LADDR parametresinde hatalı değerler yayınlanır.
RET_VAL	OUT or RETURN ?	Int	E, A, M, D, L	Yordamın hata kodu.
LADDR	OUT	HW_IO	E, A, M, D, L	G/Ç alt modülünün belirlenen donanım kodu (mantıksal adres).

Geçerli veri tipleriyle ilgili daha fazla bilgiyi, STEP 7 online yardımındaki "Geçerli veri tiplerine genel bakış" başlığında bulabilirsiniz.

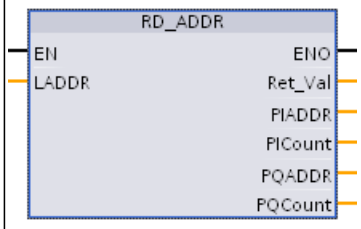
Tablo 9-234 Koşul kodları

RET_VAL* (W#16#...)	Açıklama
0	Hata yok.
8090	ADDR parametresinde belirtilen G/Ç adresi, hiçbir donanım bileşeni tarafından kullanılmıyor.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.	

9.11.4 RD_ADDR (G/Ç adreslerini donanım kodu aracılığıyla belirle)

RD_ADDR yordamıyla, bir alt modülün G/Ç adreslerini çağırabilirsiniz.

Tablo 9-235 RD_ADDR yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := RD_ADDR(** **laddr:=_word_in_,** **PIADDR=>_udint_out_,** **PICount=>_uint_out_,** **PQADDR=>_udint_out_,** **PQCount=>_uint_out_);**	RD_ADDR yordamıyla, bir alt modülün G/Ç adreslerini çağırabilirsiniz.

RD_ADDR yordamı, bir alt modülün donanım kodu yardımıyla, girişlerin veya çıkışların uzunluğunu ve başlangıç adresini belirler:

- LADDR parametresiyle, donanım kodu aracılığıyla giriş veya çıkış modülünü seçebilirsiniz.
- Bir giriş modülünün mü yoksa çıkış modülünün mü söz konusu olduğuna bağlı olarak aşağıdaki çıkış parametreleri kullanılır:
  - Bir giriş modülünde, belirlenen değerler PIADDR ve PICOUNT parametrelerinde yayınlanır.
  - Bir çıkış modülünde, belirlenen değerler PQADDR ve PQCOUNT parametrelerinde yayınlanır.
- PIADDR ve PQADDR parametrelerinin her biri modülün G/Ç adreslerinin başlangıç adresini içerir.
- PICOUNT ve PQCOUNT parametrelerinin her biri girişlerin veya çıkışların Bayt sayısını içerir (8 giriş/çıkış için 1 Bayt, 16 giriş/çıkış için 2 Bayt).

Tablo 9-236 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
LADDR	IN	HW_IO	(Alt) modülün donanım kodu
RET_VAL	OUT	Int	Yordamın hata kodu
PIADDR	OUT	UDInt	Giriş modülünün başlangıç adresi
PICOUNT	OUT	UInt	Girişlerin Bayt sayısı
PQADDR	OUT	UDInt	Çıkış modülünün başlangıç adresi
PQCOUNT	OUT	UInt	Çıkışların Bayt sayısı

Geçerli veri tipleriyle ilgili daha fazla bilgiyi, STEP 7 online yardımındaki "Geçerli veri tiplerine genel bakış" başlığında bulabilirsiniz.

Tablo 9-237 Koşul kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	Modülün, LADDR parametresindeki donanım kodu geçersizdir.
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.	

9.11.5 GEOADDR sistem veri tipi

Coğrafi adres

GEOADDR sistem veri tipi, bir modülün coğrafi adresini (veya yuva bilgilerini) içerir.

- PROFINET IO için kullanılan coğrafi adres:
  PROFINOT IO için coğrafi adres; PROFINET IO sisteminin ID'si, cihaz numarası, yuva numarası ve alt modülden (bir alt modül kullanılıyorsa) oluşur.
- PROFIBUS DP için kullanılan coğrafi adres:
  PROFIBUS DP için coğrafi adres; DP-Master sisteminin ID'si, istasyon numarası ve yuva numarasından oluşur.

Modüllerin yuva bilgisini her modülün donanım konfigürasyonunda bulabilirsiniz.

GEOADDR sistem veri tipinin yapısı

Veri bloğunda veri tipi olarak "GEOADDR" girilirse, GEOADDR yapısı otomatik olarak oluşturulur.

Parametre adı	Veri tipi	Açıklama
GEOADDR	STRUCT	
HWTYPE	UINT	Donanım tipi: • 1: IO sistemi (PROFINET/PROFIBUS) • 2: IO cihazı/DP-Slave • 3: Modül taşıyıcısı • 4: Modül • 5: Alt modül Bir donanım tipi yordam tarafından desteklenmiyorsa, HWTYPE, "0" değerini yayınlar.
AREA	UINT	Aralık kodu: • 0 = CPU • 1 = PROFINET IO • 2 = PROFIBUS DP • 3 = AS-i
IOSYSTEM	UINT	PROFINET IO sistemi (0 = modül taşıyıcısındaki merkezi birim)
STATION	UINT	• AREA aralık kodu = 0 (merkezi modül) ise modül taşıyıcısının numarası. • AREA alan kodu > 0 ise istasyon numarası.
SLOT	UINT	Yuva numarası
SUBSLOT	UINT	Alt modülün numarası. Hiçbir alt modül mevcut değilse veya takılamıyorsa bu parametre "0" değerine sahip olur.

9.12 Gelişmiş yordamlar için ortak hata kodları

Tablo 9-238 Gelişmiş yordamlar için ortak koşul kodları

Koşul kodu (W#16#....)[1]	Açıklama
8x22[2]	Giriş için kullanılan aralık çok küçük
8x23	Çıkış için kullanılan aralık çok küçük
8x24	İzin verilmeyen giriş aralığı
8x25	İzin verilmeyen çıkış aralığı
8x28	İzin verilmeyen giriş Bit'i ataması
8x29	İzin verilmeyen çıkış Bit'i ataması
8x30	Çıkış aralığı, yazma korumalı bir DB'dir.
8x3A	DB mevcut değil.

[1] Bir kod bloğu yürütülürken bu hatalardan biri oluşursa, CPU, RUN işletim durumunda (ön ayar) kalır veya uygun şekilde yapılandırıldıysa STOP işletim durumuna geçer. Opsiyonel olarak, bu kod bloğundaki GetError veya GetErorID yordamlarıyla yerel hata işleme başlatılabilir (CPU, RUN işletim durumunda kalır) ve hataya karşı, programlanmış bir tepki oluşturulabilir.

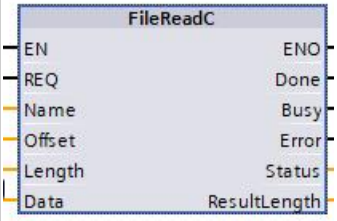
[2] "x", hatanın bulunduğu parametrenin numarasını temsil eder. Parametre numaraları 1 ile başlar.

9.13 Dosyaların kullanımı

9.13.1 FileReadC: Dosyanın memory card'dan okunması

"FileReadC" yordamıyla, verileri memory card'daki bir dosyadan okuyabilir ve CPU'daki bir hedef aralığa yazabilirsiniz. Dosyayı, adı ve eksiksiz yol aracılığıyla belirtebilirsiniz.

Tablo 9-239 FileReadC yordamı

LAD/FBD	SCL	Açıklama
	**"FileReadC_SFB_DB_2"(REQ:=_bool_in_ Name:=_string_in_ Offset:=_udint_in_, Length:=_udint_in_, Done=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, Status=>_word_out_, ResultLength=>_udint_out_, Data:=_variant_inout_);**	"FileReadC" yordamıyla, verileri memory card'daki bir dosyadan okuyabilir ve CPU'daki bir hedef aralığa yazabilirsiniz.

Dosyadan okuma fonksiyonunu, REQ kumanda parametresiyle etkinleştirebilirsiniz. Done, Busy, Error ve Status çıkış parametreleri, görev durumunu belirtir.

Dosya adı ve yol bilgisi için aşağıdaki kurallar geçerlidir:

- Dosya adı maksimum 55 karakter uzunluğunda olmalıdır.
- Klasörler ve dosya adları için şu karakterlere izin verilir: "0" ile "9" arasında rakamlar, "a" ile "z" arasında harfler, "A" ile "Z" arasında harfler, "-", "_"
- Dosya adı maksimum bir nokta (".") içerebilir. Nokta, adı dosya uzantısından ayırır. Dosya adı en az bir karakter içermelidir. Bir dosya uzantısına gerek yoktur.
- Yol adı "/", "\" veya "." ile başlayamaz.

- Yol adı ".." içeremez.
- Yol adı, "UserFiles" klasörünün altında başka alt klasörler içeremez.

Geçerli yollar ve dosya adlarıyla ilgili örnekler: "UserFiles/Lift16_DataBase.txt", "UserFiles/2017-04-13_ErrorLog.bin"

FileReadC ile maksimum 16 MB (16.777.216 Bayt) okuyabilirsiniz. Veriler, segment şeklinde okunur. Segment boyutu cihaza özeldir. Bir S7-1500 CPU'su örneğin 32 KB'lık (32.768 Bayt) bloklar kullanır. Okunacak veri elemanlarının sayısı segment boyutundan büyükse, yordamı programınızda birden fazla kez çağırmanız gerekir. Bunun için, tutarlılık nedenleriyle, okunan verilere ancak yordam son kez çağrıldıktan sonra erişmeniz gerekir.

"Offset" ve "Length" parametreleri, okunacak veri elemanlarının uzunluğunu belirler. Meşgul kaynak, okuma işlemi tamamlandıktan sonra tekrar serbest bırakılır.

"FileReadC" yordamı asenkron çalışır. İşleme süreci, birden fazla çağrı süresince gerçekleşir. İşleme, "REQ" parametresindeki bir yükselen kenarla başlar.

"Busy" ve "Done" parametreleri görev durumunu belirtir.

Yürütme sırasında bir hata oluşursa, bu durum, "Error" ve "Status" parametreleri tarafından gösterilir.

Tablo 9-240 Parametreler için kullanılan veri tipleri

Parametre ve tip		**Veri tipi**	**Açıklama**
REQ	Input	BOOL	REQUEST kumanda parametresi Yükselen bir kenarda memory card'dan bir dosyanın okunmasına olanak sağlar
Name	Input	STRING	Tam yolla birlikte, okunacak dosyanın adı
Offset	Input	UDINT	Kendisinden sonra dosyanın okunacağı Bayt ofseti
Length	Input	UDINT	Okunacak aralığın Bayt cinsinden uzunluğu Length = 0, çağrı başına mümkün olan maksimum sayıda veri elemanının okunacağı anlamına gelir (bu, bir S7-1200 CPU'sunda 8 KB veya "Data" parametresinin boyutudur).
Done	Output	BOOL	1: Yordam başarıyla yürütüldü. Okunan parametreler, "Data" parametresine aktarılır.
Busy	Output	BOOL	Durum parametresi • 0: Yordamın yürütülmesi tamamlandı veya henüz başlatılmadı. • 1: Yordamın yürütülmesi henüz tamamlanmadı.
Error	Output	BOOL	Durum parametresi • 0: Hata yok. • 1: Yordamın yürütülmesi sırasında bir hata oluştu. Ayrıntılı bilgiler, "Status" parametresinde yayınlanır.
Status	Output	WORD	Hata kodu
ResultLength	Output	UDINT	Okunan verilerin Bayt cinsinden uzunluğu
Data	InOut	VARIANT	Okunan veriler için hedef aralık İzin verilen veri tipleri: BYTE ve BYTE'lardan oluşan Array

Tablo 9-241 Koşul kodları

Hata kodu* (W#16#...)	**Anlamı**
0	Hata yok
7000	Devam eden bir görev işlemesi yok
7001	Görev işlemesinin başlangıcı. Busy parametresi = 1, Done = 0.

Hata kodu* (W#16#...)	Anlamı
7002	Ara çağrı (REQ ilgili değil): Yordam halihazırda aktif; Busy, 1 değerine sahiptir.
0081	Uyarı: Offset + Length, dosyanın uzunluğundan büyüktür. Veriler, "Offset"'ten itibaren dosyanın sonuna kadar okunur. "Data", istenenden daha az veri elemanı içeriyor. "ResultLength", "Length" değerinden daha kısadır. "Data"daki veriler "ResultLength"ten sonra değiştirilmez.
8091	Yol mevcut değil veya geçersiz.
8092	"Name" parametresi "STRING" veri tipine sahip değil, çok uzun veya geçersiz karakterler içeriyor.
8093	"Offset" parametresi, okunacak dosyanın sonunun ötesindeki bir noktayı gösterir.
8094	"Length" parametresi, izin verilen maksimum değerden büyüktür. Bir S7-1500 veya S7-1200 CPU'sunda örneğin izin verilen maksimum değer 16 MB, yani 16.777.216 Bayt'tır.
80A1	Okuma hatası. "Data" parametresi tarafından belirtilen hedef aralığın üzerine muhtemelen kısmen yazılır.
80B1	"Data" parametresi tarafından belirtilen hedef aralık, "Length" parametresi tarafından istenen uzunluktan daha kısadır.
80C0	Dosyaya erişilemiyor (yazma korumalı veya başka bir proses tarafından bloke edilmiş).
80C3	Eş zamanlı aktif olan "FileReadC" yordamlarının maksimum sayısına ulaşıldı.
8A30	Hedef aralık yazma korumalıdır, örneğin yazma korumalı bir DB'dir.
8A3A	"Data" parametresi, izin verilmeyen bir aralığı, örneğin yükleme belleğini veya yerel verileri gösteriyor.
8A51	"Data" parametresinin veri tipi geçersiz.
8A52	Data parametresindeki değişken yeterli değil. Kaynak aralık verilerinin bir kısmı muhtemelen yazılmış.
* Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenir. Görüntüleme formatlarının değiştirilmesiyle ilgili bilgiler için, bkz. "Ayrıca bkz".	

9.13.2 FileWriteC: Dosyayı memory card'a yaz

"FileWriteC" yordamıyla, verileri CPU'daki bir kaynak aralıktan memory card'daki "UserFiles" klasöründe bulunan bir dosyaya yazabilirsiniz.

Tablo 9-242 FileWriteC yordamı

LAD/FBD	SCL	Açıklama
	**"FileWriteC_SFB_DB_1"(REQ:=_bool_in_, Name:=_string_in_, Offset:=_udint_in_, Length:=_udint_in_, Done=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, Status=>_word_out_, ResultLength=>_udint_out_, Data:=_variant_inout_);**	"FileWriteC" yordamıyla, verileri CPU'daki bir kaynak aralıktan memory card'daki "UserFiles" klasöründe bulunan bir dosyaya yazabilirsiniz. Dosyayı, adı ve eksiksiz yol aracılığıyla belirtebilirsiniz. Dosya mevcut değilse, CPU tarafından "UserFiles" klasöründe oluşturulur. "UserFiles" klasörü de mevcut değilse, bu klasör de CPU tarafından oluşturulur. Alt klasörler oluşturulmaz. Bu durumda "Status" parametresinde W#16#8091 ifadesi yayınlanır.

Dosyaya yazma fonksiyonunu, "REQ" kumanda parametresiyle etkinleştirebilirsiniz. "Done", "Busy", "Error" ve "Status" çıkış parametreleri görev durumunu belirtir.

Dosya adı ve yol bilgisi için aşağıdaki kurallar geçerlidir:

- Dosya adı maksimum 55 karakter uzunluğunda olmalıdır.
- Klasörler ve dosya adları için şu karakterlere izin verilir: "0" ile "9" arasında rakamlar, "a" ile "z" arasında harfler, "A" ile "Z" arasında harfler, "-", "_"

- Dosya adı maksimum bir nokta (".") içerebilir. Nokta, adı dosya uzantısından ayırır. Dosya adı en az bir karakter içermelidir. Bir dosya uzantısına gerek yoktur.
- Yol adı "/", "\" veya "." ile başlayamaz.
- Yol adı ".." içeremez.
- Yol adı, "UserFiles" klasörünün altında başka alt klasörler içeremez.

Geçerli yollar ve dosya adlarıyla ilgili örnekler: "UserFiles/Lift16_DataBase.txt", "UserFiles/2017-04-13_ErrorLog.bin"

"FileWriteC" ile maksimum 16 MB (16.777.216 Bayt) yazabilirsiniz. Veriler, segment şeklinde yazılır. Segment boyutu cihaza özeldir. Bir S7-1200 CPU'su, örneğin 8 KB'lık (8192 Bayt) bloklar kullanır. Yazılacak veri elemanlarının sayısı segment boyutundan büyükse, yordamı programınızda birden çok kez çağırmanız gerekir. Bunun için, tutarlılık nedenleriyle, yazılan verilere ancak yordam son kez çağrıldıktan sonra erişmeniz gerekir. Mevcut dosya çok kısaysa, gereken boyuta gelecek şekilde uzatılır.

"Offset" ve "Length" parametreleri, dosyada bulunan ve verilerin yazılacağı saklama konumunu belirtir. Meşgul kaynak, yazma işlemi tamamlandıktan sonra tekrar serbest bırakılır.

"FileWriteC" yordamı yazma işlemine, ancak şu koşul sağlandıysa başlar: "Offset" + "Length" ≤ 16 MB.

"FileWriteC" yordamı asenkron çalışır. İşleme süreci, birden fazla çağrı süresince gerçekleşir. İşleme, "REQ" parametresindeki bir yükselen kenarla başlar.

"Busy" ve "Done" parametreleri görev durumunu belirtir.

Yürütme sırasında bir hata oluşursa, bu durum, "Error" ve "Status" parametreleri tarafından gösterilir.

Tablo 9-243 Parametreler için kullanılan veri tipleri

Parametre ve tip		**Veri tipi**	**Açıklama**
REQ	Input	BOOL	REQUEST kumanda parametresi Yükselen bir kenarda memory card'a bir dosyanın yazılmasına olanak sağlar
Name	Input	STRING	Tam yolla birlikte, yazılacak dosyanın adı
Offset	Input	UDINT	Kendisinden itibaren dosyanın yazılacağı Bayt ofseti
Length	Input	UDINT	Yazılacak aralığın Bayt cinsinden uzunluğu "Length" = 0, "Data" parametresi tarafından belirtilen tüm kaynak aralığın yazılacağı anlamına gelir.
Done	Output	BOOL	1: Yordam başarıyla yürütüldü.
Busy	Output	BOOL	Durum parametresi • 0: Yordamın yürütülmesi tamamlandı veya henüz başlatılmadı. • 1: Yordamın yürütülmesi henüz tamamlanmadı.
Error	Output	BOOL	Durum parametresi • 0: Hata yok. • 1: Yordamın yürütülmesi sırasında bir hata oluştu. Ayrıntılı bilgiler, "Status" parametresinde yayınlanır.
Status	Output	WORD	Hata kodu
ResultLength	Output	UDINT	Yazılan verilerin Bayt cinsinden uzunluğu
Data	InOut	VARIANT	Kaynak aralık İzin verilen veri tipleri: BYTE ve BYTE'lardan oluşan Array

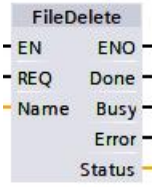
Tablo 9-244 Koşul kodları

Hata kodu* (W#16#...)	**Anlamı**
0	Hata yok
7000	Devam eden bir görev işlemesi yok
7001	Görev işlemesinin başlangıcı. Busy parametresi = 1, Done = 0.
7002	Ara çağrı (REQ ilgili değil): Yordam halihazırda aktif; Busy, 1 değerine sahiptir.
8091	Yol mevcut değil veya geçersiz.
8092	"Name" parametresi "STRING" veri tipine sahip değil, çok uzun veya geçersiz karakterler içeriyor.
8093	• "Offset" parametresi, yazılacak dosyanın sonunun ötesindeki bir noktayı gösterir. • "Offset" sıfırdan büyük olduğu için dosya oluşturma işlemi reddedilir.
8094	• "Length", izin verilen maksimum değerden büyüktür. Bir S7-1500 CPU'sunda örneğin izin verilen maksimum değer 16 MB, yani 16.777.216 Bayt'tır. • "Length" + "Offset", izin verilen maksimum değerden büyüktür.
80A1	Yazma hatası. Memory card'daki dosyada bulunan verilerin üzerine muhtemelen kısmen yazılır.
80B1	"Data" parametresi tarafından belirtilen kaynak aralık, "Length" parametresi tarafından istenen uzunluktan daha kısadır.
80B3	Memory card'da veya dahili yükleme belleğinde yeterli saklama alanı yok.
80B4	Memory card veya dosya yazma korumalıdır.
80C0	Dosyaya erişilemiyor.
80C3	Eş zamanlı aktif olan "FileWriteC" yordamlarının maksimum sayısına ulaşıldı.
8A24	"Data" parametresi, izin verilmeyen bir aralığı, örneğin yükleme belleğini veya yerel verileri gösteriyor.
8A51	"Data" parametresinin veri tipi geçersiz.
8A52	Data parametresindeki değişken yeterli değil. Kaynak aralık verilerinin bir kısmı muhtemelen yazılmış.
* Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenir. Görüntüleme formatlarının değiştirilmesiyle ilgili bilgiler için, bkz. "Ayrıca bkz".	

9.13.3 FileDelete: Memory card'daki dosyayı sil

"FileDelete" yordamıyla mevcut bir dosyayı memory card'dan silebilirsiniz.

Tablo 9-245 FileDelete yordamı

LAD/FBD	SCL	Açıklama
	**"FileDelete_DB_1"(REQ:=_bool_in_, Name:=_string_in_, Done=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, Status=>_word_out_)**	"FileDelete" yordamıyla mevcut bir dosyayı memory card'dan silebilirsiniz.

Dosya açık olmamalıdır. "Name" parametresinde yer tutucular desteklenmez, bu da "UserFiles/*.txt" ve "UserFiles/?.txt" gibi adlara izin verilmediği anlamına gelir.

"FileDelete" yordamına, sadece "Recipes" ve "UserFiles" klasörlerinde izin verilir. Bu klasörlerin içinde başka klasörler olabilir, örn. "UserFiles/Test/file1.txt".

NOT

Veri protokollerinin silinmesi

"DataLog" klasöründeki bir dosyanın "FileDelete" yordamıyla silinmesine izin verilmez. Veri protokollerinin silinmesi için "DataLogDelete" yordamını kullanmanız gerekir.

"FileDelete" asenkron bir yordamdır. İşleme süreci, birden çok çağrıyı kapsayabilir. İşleme, "REQ" parametresindeki bir yükselen kenarla başlar.

"Busy" ve "Done" parametreleri görev durumunu belirtir.

Yürütme sırasında bir hata oluşursa, bu durum, "Error" ve "Status" parametreleri tarafından gösterilir.

Tablo 9-246 Parametreler için kullanılan veri tipleri

Parametre ve tip		Veri tipi	Açıklama
REQ	Input	BOOL	REQUEST kumanda parametresi İşleme, REQ'taki bir yükselen kenarda başlar.
Name	Input	STRING	Silinecek dosyanın yolu ve adı
Done	Output	BOOL	Durum parametresi • 1: Yordam başarıyla yürütüldü.
Busy	Output	BOOL	Durum parametresi • 0: Yordam şu anda yürütülmüyor. • 1: Yordam şu anda yürütülüyor.
Error	Output	BOOL	Durum parametresi • 0: Hata yok • 1: Yordamın yürütülmesi sırasında bir hata oluştu. Ayrıntılı bilgileri, "Status" parametresinde bulabilirsiniz.
Status	Output	WORD	Hata kodu

Tablo 9-247 Koşul kodları

Hata kodu* (W#16#...)	**Anlamı**
0000	Yordam başarıyla tamamlandı.
7000	Devam eden bir görev işlemesi yok
7001	Görev işlemenin başlangıcı: Busy = 1, Done = 0
7002	Ara çağrı (REQ ilgili değil): Yordam halihazırda aktif; Busy, 1 değerine sahiptir.
8090	Dosya bloke edilmiştir, örn. dosya açıksa
8091	Yol mevcut değil veya geçersiz.
8092	Dosya, yolda mevcut değil
80A2	Yazma hatası
80A3	Dosya çok büyük (≥ 2147483648 Bayt) ve "FileDelete" ile silinemiyor.
80B4	Memory card yazma korumalıdır.
80C3	Eş zamanlı aktif olan "FileDelete" yordamlarının maksimum sayısına ulaşıldı.
* Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenir.	

10

Teknoloji yordamları

10.1 Sayma (hızlı sayaç)

"Sayaç" (Sayfa 218) başlığında açıklanan temel sayaç yordamları, sadece S7-1200 CPU'sunun çevriminden daha yavaş ortaya çıkan olayları sayar. Hızlı sayaç (High-Speed Counter, HSC), PLC çevriminden daha hızlı oluşan darbeleri sayma olanağı sunar. Ayrıca, oluşan darbelerin frekansını ve periyodunu sayacak şekilde HSC'yi yapılandırabilir veya hareket kumandası, HSC ile bir motor enkoderi sinyalini okuyabilecek şekilde ayarlayabilirsiniz.

HSC fonksiyonunu kullanmak için, HSC'nin önce CPU'nun cihaz konfigürasyonundaki "Özellikler" sekmesinde etkinleştirilmiş ve yapılandırılmış olması gerekir. HSC'yi nasıl yapılandıracağınızı, "Bir hızlı sayacın yapılandırılması" (Sayfa 524) bölümünden öğrenebilirsiniz.

Donanım konfigürasyonunu yüklemenizin ardından, HSC, ilgili yordamların çağrılması gerekmeden darbeleri sayabilir veya frekans değerini ölçebilir. HSC, sayma veya periyot işletim türünde ise, sayma değeri otomatik olarak her çevrimde kaydedilir ve proses görünümünde (E bellek alanı) güncellenir. HSC frekans işletim türündeyse, proses görünümündeki değer, Hz cinsinden frekans olur.

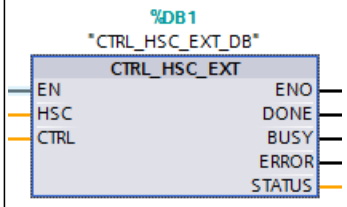
HSC, sayma ve ölçme dışında proses alarmı olayları da üretebilir, fiziksel girişlerin durumuna bağlı olarak çalışabilir ve belirtilen sayaç olayına uygun olarak bir çıkış darbesi oluşturabilir. Kullanıcı programı, CTRL_HSC_EXT teknoloji yordamı aracılığıyla HSC'yi program üzerinden kontrol edebilir. CTRL_HSC_EXT yürütülürse, yordam, HSC parametrelerini günceller ve en güncel değerleri yayınlar. CTRL_HSC_EXT yordamını sayma, periyot ve frekans işletim türlerinde kullanabilirsiniz.

NOT

CTRL_HSC_EXT yordamı, V4.2'den itibaren CPU'lar için projelerde şimdiye kadar kullanılan CTRL_HSC'nin yerine geçmiştir. CTRL_HSC_EXT yordamıyla, CTRL_HSC yordamının tüm fonksiyon kapsamı ve bazı ek fonksiyonlar kullanıma sunulur. Şimdiye kadar kullanılan CTRL_HSC yordamı, sadece uyumlulukla ilgili nedenlerle daha eski S7-1200 projeleri için kullanılabilir ve yeni projelerde kullanılmamalıdır.

10.1.1 CTRL_HSC_EXT (hızlı sayaca kumanda et, geliştirilmiş)

Tablo 10-1 CTRL_HSC_EXT yordamı

LAD/FBD	SCL	Açıklama
	**"CTRL_HSC_1_DB" (hsc:=_hw_hsc_in_, done:=_done_out_, busy:=_busy_ out_, error:=_error_out_, status:=_status_out_, ctrl:=_variant_in_);**	Her CTRL_HSC_EXT yordamı (hızlı sayaca kumanda et, geliştirilmiş), sayaç verilerini kaydetmek için kullanıcı tanımlı global bir DB'ye kaydedilen sistem tanımlı veri yapısı kullanır. HSC_Count, HSC_Period ve HSC_Frequency veri tiplerini CTRL_HSC_EXT yordamının giriş parametreleri olarak atayabilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "CTRL_HSC_1_DB", örnek DB'nin adıdır.

Tablo 10-2 Parametreler için kullanılan veri tipleri

Parametre	Deklarasyon	Veri tipi	Açıklama
HSC	IN	HW_HSC	HSC kodu
CTRL	IN_OUT	Variant	SFB girişi ve iade verileri. Not: Daha fazla bilgiyi şu başlıkta bulabilirsiniz: "CTRL_HSC_EXT (SDT) yordamı için kullanılan sistem veri tipleri (Sayfa 514)".
DONE	OUT	Bool	1 = SFB'nin tamamlandığını gösterir. SFB senkron olduğu için her zaman 1'dir.
BUSY	OUT	Bool	Her zaman 0, fonksiyon dolu değil.
ERROR	OUT	Bool	1 = Bir hatayı belirtir.
STATUS	OUT	Word	Yürütme koşulu Not: Daha fazla bilgi içeren aşağıdaki "Yürütme koşulları" tablosuna bakın.

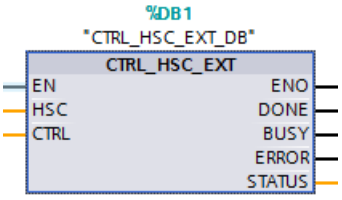
Tablo 10-3 Yürütme koşulları

STATUS (W#16#)	Açıklama
0	Hata yok
80A1	HSC kodu hiçbir HSC'yi adreslemiyor
80B1	NewDirection'da geçersiz değer
80B4	NewPeriod'da geçersiz değer
80B5	NewOpModeBehavior'da geçersiz değer
80B6	NewLimitBehavior'da geçersiz değer
80D0	SBF 124 mevcut değil

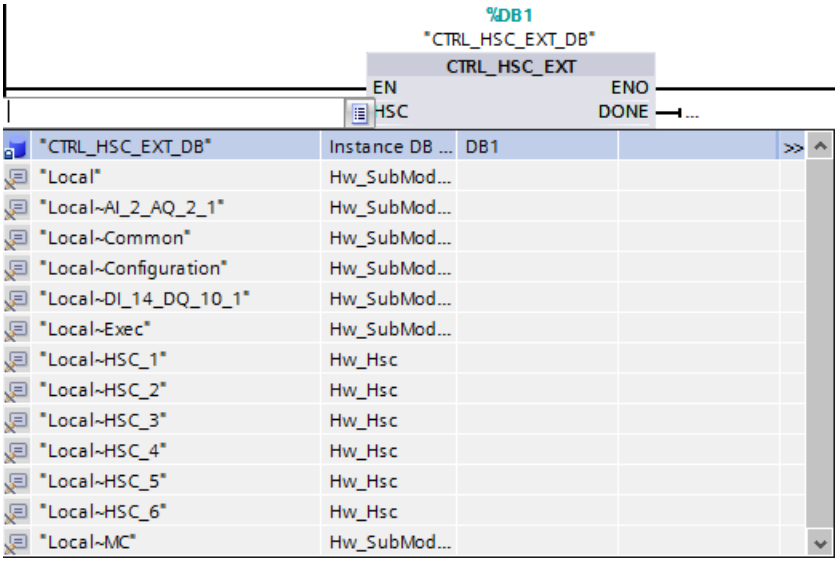
10.1.1.1 CTRL_HSC_EXT Örnek

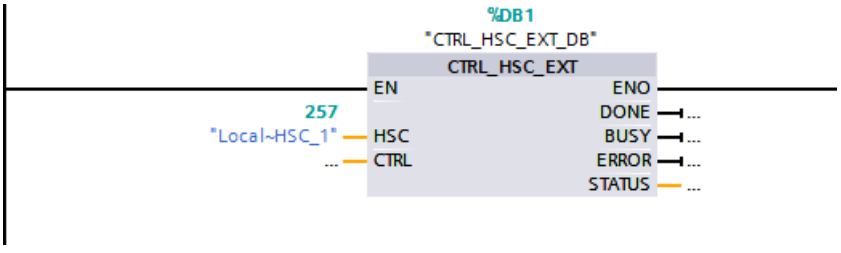
CTRL_HSC_EXT yordamını kullanmak için aşağıdaki yöntemi izleyin:

1. CTRL_HSC_EXT yordamını, şu örnek veri bloğunu da oluşturan LAD ağına ekleyin: "CTRL_HSC_EXT_DB":



2. HSC'nin özelliklerinde bulabileceğiniz HSC donanım kodunu, LAD yordamının "HSC" bağlantısına ekleyin. Bu giriş bağlantısı için açılır menüden altı "Hw_Hsc" nesnesinden birini de seçebilirsiniz. HSC1 için standart değişken adı "Local~HSC_1"dir:



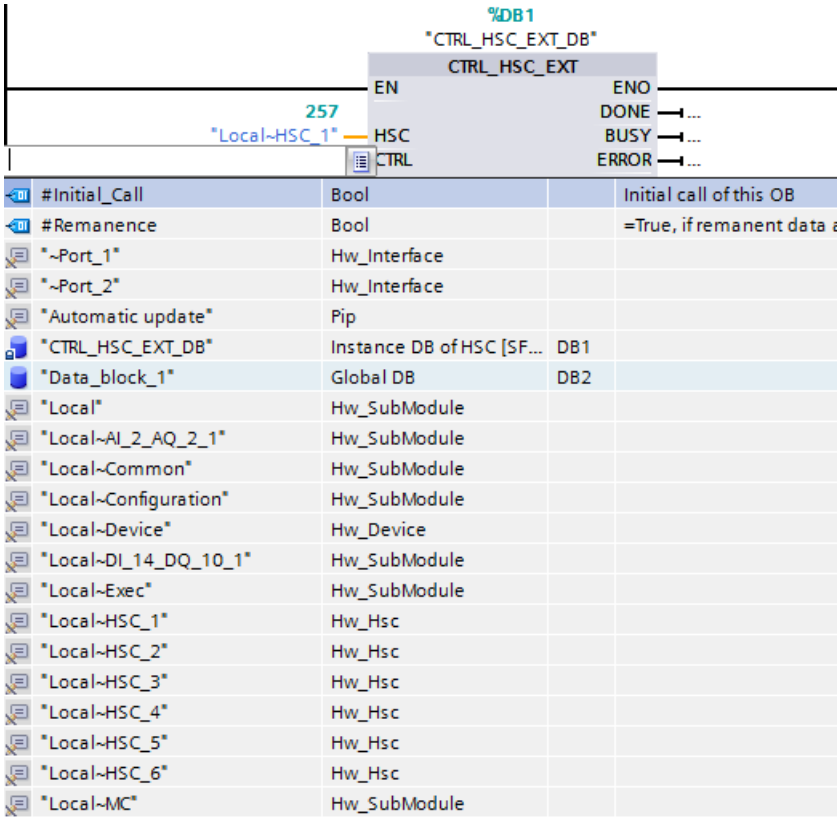


3. "Data_block_1" adıyla global bir veri bloğu oluşturun (mevcut bir global veri bloğunu da kullanabilirsiniz):
   – "Data_block_1"de boş bir satır arayın ve "MyHSC" adıyla bir değişken ekleyin.
   – "Veri tipi" sütununa aşağıdaki sistem veri tiplerinden (SDT) birini ekleyin. HSC'nin yapılandırılan sayma şeklinde karşılık gelen SDT'yi seçin. HSC'lerin SDT'leriyle ilgili daha fazla bilgiyi, bu alt bölümün daha aşağıdaki kısımlarında bulabilirsiniz. Açılır liste bu tipleri içermemektedir. Bu nedenle SDT adını tam olarak gösterildiği şekilde girmeye dikkat edin: HSC_Count, HSC_Period veya HSC_Frequency
   – Veri tipini girdikten sonra, veri yapısında bulunan tüm alanları görüntülemek için "MyHSC" değişkenini genişletebilirsiniz. Burada, tüm alanların veri tipini bulabilir ve standart başlangıç değerlerini değiştirebilirsiniz:

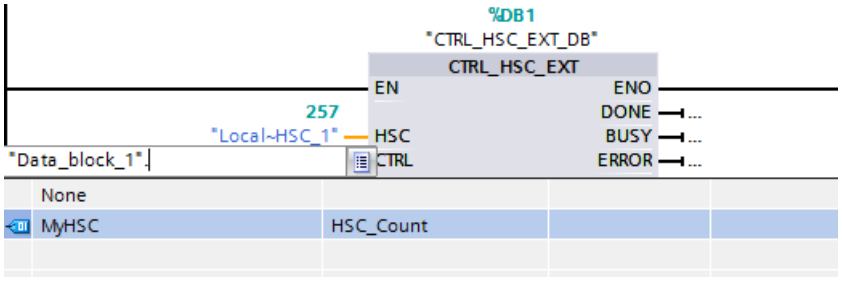
Data_block_1

	Name	Data type	Start value
1	Static		
2	MyHSC	HSC_Count	
3	CurrentCount	DInt	0
4	CapturedCount	DInt	0
5	SyncActive	Bool	false
6	DirChange	Bool	false
7	CmpResult_1	Bool	false
8	CmpResult_2	Bool	false
9	OverflowNeg	Bool	false
10	OverflowPos	Bool	false
11	EnHSC	Bool	false
12	EnCapture	Bool	false
13	EnSync	Bool	false
14	EnDir	Bool	false
15	EnCV	Bool	false
16	EnSV	Bool	false
17	EnReference1	Bool	false
18	EnReference2	Bool	false
19	EnUpperLmt	Bool	false
20	EnLowerLmt	Bool	false
21	EnOpMode	Bool	false
22	EnLmtBehavior	Bool	false
23	EnSyncBehavior	Bool	false
24	NewDirection	Int	0
25	NewOpModeBeha...	Int	0
26	NewLimitBehavior	Int	0
27	NewSyncBehavior	Int	0
28	NewCurrentCount	DInt	0
29	NewStartValue	DInt	0
30	NewReference1	DInt	0
31	NewReference2	DInt	0
32	NewUpperLimit	DInt	0
33	New_Lower_Limit	DInt	0

4. "'Data_block_1'. MyHSC" değişkenini CTRL_HSC_EXT yordamının CTRL giriş bağlantısına atayın:
   – "Data_Block_1"i seçin.



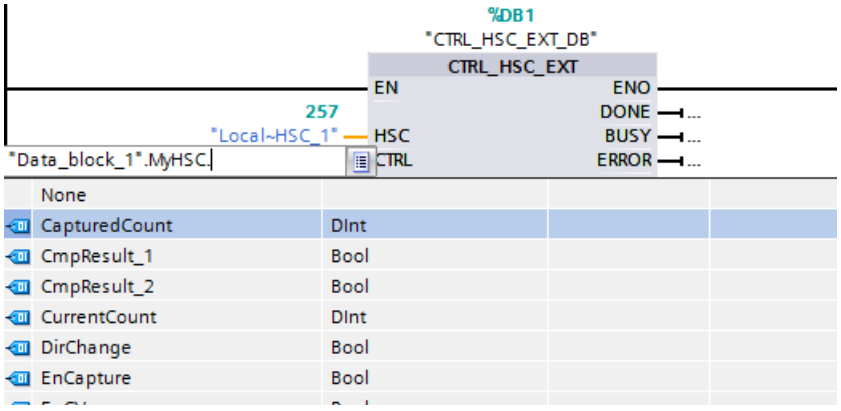
   – "MyHSC"yi seçin.



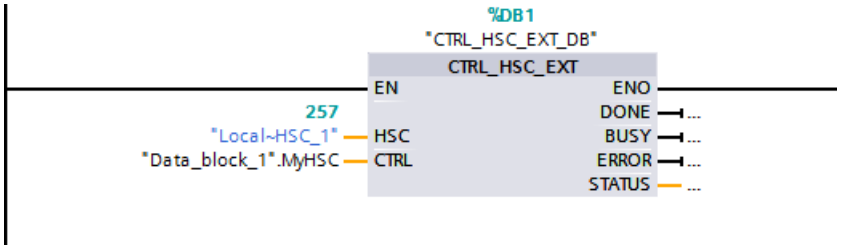
   – "'Data_Block_1'.MyHSC"den sonra gelen noktayı (".") silin. Ardından alanın dışına tıklayın veya bir kez ESC tuşuna ve ardından Enter tuşuna basın.

NOT

"'Data_Block_1'.MyHSC"den sonra gelen noktayı (".") sildikten sonra sadece Enter tuşuna basmayın. Aksi takdirde nokta (".") tekrar alana eklenir.



– CTRL için kullanılan tam giriş aşağıda gösterilmektedir.



PLC'de HSC'yi yapılandırdıktan sonra CTRL_HSC_EXT yordamını yürütebilirsiniz. Bir hata oluştuğunda ENO 0'a ayarlanır ve STATUS çıkışı koşul kodunu belirtir.

Ayrıca bakınız

CTRL_HSC_EXT yordamının sistem veri tipleri (SDT) (Sayfa 514)

10.1.1.2 CTRL_HSC_EXT yordamının sistem veri tipleri (SDT)

Aşağıdaki sistem veri tipleri (SDT'ler) sadece CTRL_HSC_EXT yordamının CTRL bağlantısı için kullanılır. Bunları kullanmak için bir kullanıcı veri bloğu oluşturun ve HSC'nin yapılandırılmış işletim türüne (sayma türü) karşılık gelen, SDT'nin veri tipinde bir nesne ekleyin. STEP 7'de bu veri tipleri açılır menüde görüntülenmez. SDT'nin adını da tam olarak gösterildiği şekilde girin. HSC'ye ait SDT'nin girişleri "En" veya "New" ön ekiyle işaretlenir. "En" ön ekine sahip girişler bir HSC fonksiyonunu etkinleştirir veya ilgili parametreyi günceller. "New" ön eki, güncelleme değerini tanımlar. Yeni değerin etkili olabilmesi için buna uygun olarak "En" Bit'i doğru ve "New" değeri de geçerli olmalıdır. CTRL_HSC_EXT yordamı yürütülürse, program, giriş değişikliklerini uygular ve çıkışları ilgili SDT referansıyla günceller.

"HSC_Count" veri tipi, "Sayma" işletim türü için yapılandırılmış olan bir HSC'ye karşılık gelir. Sayma işletim türü aşağıdaki fonksiyonları sunar:

- Güncel darbe sayma değerine erişim
- Bir giriş olayında güncel darbe sayma değerinin tutulması
- Bir giriş olayında güncel darbe sayma değerinin sıfırlanarak başlangıç değerine döndürülmesi
- Belirli HSC olaylarının ortaya çıktığını bildiren Durum Bit'lerine erişim
- HSC'nin bir yazılım veya donanım girişi üzerinden devre dışı bırakılması
- Sayma yönünün bir yazılım veya donanım girişi üzerinden değiştirilmesi
- Güncel darbe sayma değerinin değiştirilmesi
- Başlangıç değerinin değiştirilmesi (CPU, RUN işletim durumuna geçerse veya senkronizasyon fonksiyonu tetiklenirse kullanılır)
- Karşılaştırmalar için iki bağımsız referans değerin (veya standart değerler) değiştirilmesi
- Üst ve alt sayma sınırlarının değiştirilmesi
- Darbe sayma değerinin bu limitlere ulaşması durumunda HSC'nin çalışma şeklinin değiştirilmesi
- Güncel darbe sayma değerinin bir referans değere (ön ayarlı değer) ulaşması durumunda bir proses alarmı olayının üretilmesi
- Senkronizasyon girişinin (sıfırlama girişi) etkinleştirilmesi durumunda bir proses alarmı olayının üretilmesi
- Sayma yönünün harici bir girişi temel alarak değişmesi durumunda bir proses alarmı olayının üretilmesi
- Belirtilen bir sayma olayında münferit bir çıkış darbesinin üretilmesi

CTRL_HSC_EXT yordamı yürütüldüğünde bir olay ortaya çıkarsa, yordam, bir durum Bit'i ayarlar. Yordam yürütülmeden önce olay yeniden ortaya çıkmadığı sürece, yordamın bir sonraki yürütülüşü sırasında yordam, durum Bit'ini siler.

Tablo 10-4 HSC_Count yapısı

Yapı elemanı	Deklarasyon	Veri tipi	Açıklama
CurrentCount	OUT	Dint	HSC'nin güncel sayma değerini yayınlar
CapturedCount	OUT	Dint	Belirtilen giriş olayında kaydedilen sayma değerini yayınlar
SyncActive	OUT	Bool	Durum Bit'i: Senkronizasyon girişi etkinleştirildi
DirChange	OUT	Bool	Durum Bit'i: Sayma yönü değiştirildi
CmpResult1	OUT	Bool	Durum Bit'i: CurrentCount eşittir Reference1 olayı oluştu
CmpResult2	OUT	Bool	Durum Bit'i: CurrentCount eşittir Reference2 olayı oluştu
OverflowNeg	OUT	Bool	Durum Bit'i: CurrentCount, LowerLimit değerine ulaştı
OverflowPos	OUT	Bool	Durum Bit'i: CurrentCount, UpperLimit değerine ulaştı
EnHSC	IN	Bool	Doğru ise HSC'nin darbeleri saymasına olanak sağlar; yanlış ise sayma fonksiyonu devre dışı bırakılır
EnCapture	IN	Bool	Doğru ise Capture girişini etkinleştirir; yanlış ise Capture girişi etkili olmaz
EnSync	IN	Bool	Doğru ise Sync girişini etkinleştirir; yanlış ise Sync girişi etkili olmaz
EnDir	IN	Bool	NewDirection değerinin etkili olmasına olanak sağlar
EnCV	IN	Bool	NewCurrentCount değerinin etkili olmasına olanak sağlar

Yapı elemanı	Deklarasyon	Veri tipi	Açıklama
EnSV	IN	Bool	NewStartValue değerinin etkili olmasına olanak sağlar
EnReference1	IN	Bool	NewReference1 değerinin etkili olmasına olanak sağlar
EnReference2	IN	Bool	NewReference2 değerinin etkili olmasına olanak sağlar
EnUpperLmt	IN	Bool	NewUpperLimit değerinin etkili olmasına olanak sağlar
EnLowerLmt	IN	Bool	New_Lower_Limitinin etkili olmasına olanak sağlar
EnOpMode	IN	Bool	NewOpModeBehavior değerinin etkili olmasına olanak sağlar
EnLmtBehavior	IN	Bool	NewLimitBehavior değerinin etkili olmasına olanak sağlar
EnSyncBehavior	IN	Bool	Bu değer kullanılmaz.
NewDirection	IN	Int	Sayma yönü: 1 = İleriye sayma; -1 = Geriye sayma; diğer tüm değerler rezerve edilmiştir
NewOpModeBehavior	IN	Int	Taşma durumunda HSC'nin çalışma şekli: 1 = HSC, saymaya son verir (saymaya devam etmesi için HSC devre dışı bırakılmalı ve yeniden etkinleştirilmelidir); 2 = HSC, işletimi devam ettirir; diğer tüm değerler rezerve edilmiştir
NewLimitBehavior	IN	Int	Taşma durumunda CurrentCount değerinin sonucu: 1 = CurrentCount'u ters limite ayarlama; 2 = CurrentCount'u StartValue değerine ayarlama; diğer tüm değerler rezerve edilmiştir
NewSyncBehavior	IN	Int	Bu değer kullanılmaz.
NewCurrentCount	IN	Dint	CurrentCount'un değeri
NewStartValue	IN	Dint	StartValue: HSC'nin başlangıç değeri
NewReference1	IN	Dint	Reference1'in değeri
NewReference2	IN	Dint	Reference2'nin değeri
NewUpperLimit	IN	Dint	Üst sayma limiti
New_Lower_Limit	IN	Dint	Alt sayma limiti

"HSC_Period" veri tipi, "Periyot" işletim türü için yapılandırılmış bir HSC'ye karşılık gelir. CTRL_HSC_EXT yordamı, belirtilen bir ölçüm aralığı boyunca giriş darbelerinin sayısına program erişimi sağlar. Bu yordam, giriş darbelerinin arasındaki zaman diliminin nanosaniye mertebesinde bir çözünürlükle hesaplanmasına olanak sağlar.

Tablo 10-5 HSC_Period yapısı

Yapı elemanı	Deklarasyon	Veri tipi	Açıklama
ElapsedTime	OUT	UDInt	Alttaki açıklamaya bakın.
EdgeCount	OUT	UDInt	Alttaki açıklamaya bakın.
EnHSC	IN	Bool	Doğru ise, periyot ölçümü için HSC'yi etkinleştirir; yanlış ise periyot ölçümünü devre dışı bırakır.
EnPeriod	IN	Bool	NewPeriod'un değerinin etkili olmasına olanak sağlar.
NewPeriod	IN	Int	Milisaniye cinsinden ölçüm aralığı süresini belirtir. Sadece 10, 100 veya 1000 msn değerlerine izin verilir.

ElapsedTime, birbirini takip eden ölçüm aralıklarının son sayaç olaylarının arasındaki süreyi nanosaniye cinsinden belirtir. Bir ölçüm aralığı sırasında hiçbir sayaç olayı ortaya çıkmazsa, ElapsedTime, son sayaç olayından itibaren geçen kümülatif süreyi yayınlar. ElapsedTime, 0 ila 4.294.967.280 nanosaniyelik (0x0000 0000 ila 0xFFFF FFF0) bir aralığa sahiptir.

4.294.967.295 (0xFFFF FFFF) geri dönüş değeri, bir periyot taşmasının ortaya çıktığını bildirir. Taşma, darbe kenarlarının arasındaki sürenin 4,295 saniyeden büyük olduğu ve periyodun bu yordamla hesaplanamayacağı anlamına gelir. 0xFFFF FFF1 ile 0xFFFF FFFE arasındaki değerler rezerve edilmiştir.

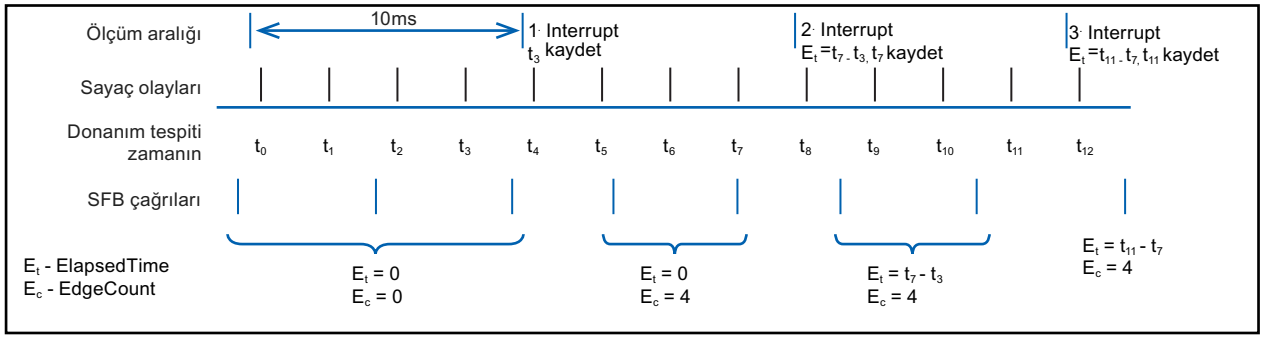
EdgeCount, ölçüm aralığı sırasında alınan sayaç olaylarının sayısını yayınlar. Periyot, ancak EdgeCount değerinin sıfırdan büyük olması halinde hesaplanabilir. ElapsedTime, 0 (hiçbir giriş darbesi alınmadı) ya da 0xFFFF FFFF (periyot taşması) ise, EdgeCount geçerli olmaz.

EdgeCount geçerli ise, nanosaniye cinsinden periyodu şu formül yardımıyla hesaplayabilirsiniz: Periyot = ElapsedTime/EdgeCount

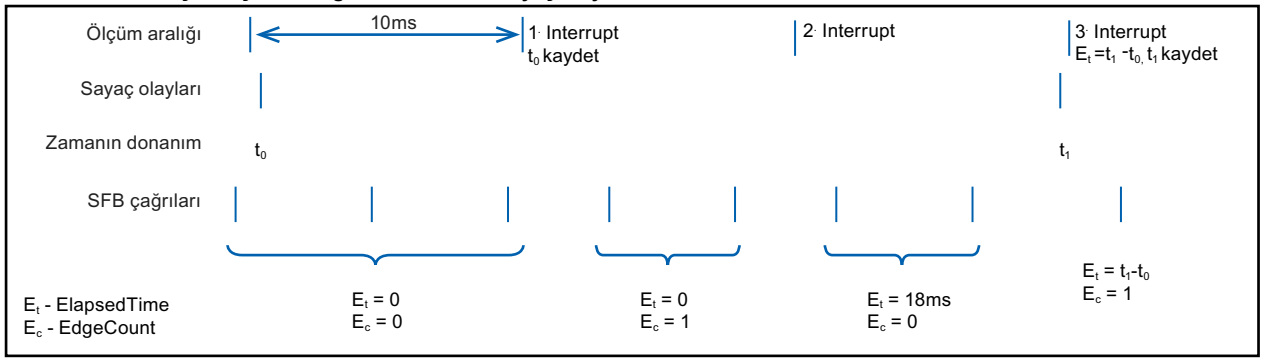
Hesaplanan periyot değeri, ölçüm aralığı esnasında ortaya çıkan tüm darbelerin zaman periyotlarından elde edilen bir ortalama değerdir. Gelen bir darbenin periyodu ölçüm aralığından büyükse (10, 100 veya 1000 msn), periyot hesaplaması için çok sayıda ölçüm aralığına ihtiyaç duyulur.

Aşağıdaki örneklerde, yordamın, periyot ölçümlerini nasıl gerçekleştirdiği gösterilmektedir:

Örnek 1: Bir ölçüm olayında birden çok sayaç olayı



Örnek 2: Birden çok ölçüm aralığında sıfır ve bir sayaç olayı



Kurallar:
1. Eğer $E1_t = 0$ ise zaman aralığı geçersizdir
2. Aksi taktirde, Zaman aralığı = E_t / E_c

"HSC_Frequency" veri tipi, "Frekans" işletim türü için yapılandırılmış bir HSC'ye karşılık gelir. CTRL_HSC_EXT yordamı, belirtilen bir zaman dilimi boyunca ölçülen giriş darbelerinin frekansına program erişimi sağlar.

CTRL_HSC_EXT yordamı, frekans işletim türünde şu olanakları sunar:

Tablo 10-6 HSC_Frequency'nin yapısı

Yapı elemanı	Deklarasyon	Veri tipi	Açıklama
Frekans	OUT	DInt	Ölçüm aralığının süresi boyunca ölçülen Hz cinsinden bir frekans değerini yayınlar. HSC geriye doğru sayarsa, yordam negatif bir frekans yayınlar.
EnHSC	IN	Bool	Doğru ise, frekans ölçümü için HSC'yi etkinleştirir; yanlış ise frekans ölçümünü devre dışı bırakır.
EnPeriod	IN	Bool	NewPeriod'un değerinin etkili olmasına olanak sağlar.
NewPeriod	IN	Int	Milisaniye cinsinden ölçüm aralığı süresini belirtir. Sadece 10, 100 veya 1000 msn değerlerine izin verilir.

CTRL_HSC_EXT yordamı, ElapsedTime ve EdgeCount değerini belirlemek için frekansı periyot işletim türüyle aynı ölçüm tekniğini kullanarak ölçer. Yordam, frekansı Hz cinsinden şu formül yardımıyla ön işaretli tamsayı değeri olarak hesaplar: Frekans = EdgeCount/ElapsedTime

HSC periyot işletim türündeyse, frekans için bir kayar noktalı değere ihtiyaç duyarsanız, yukarıda belirtilen formülü kullanabilirsiniz. ElapsedTime değerinin periyot işletim türünde nanosaniye cinsinden yayınlandığına ve gerektiğinde ölçeklenmesi gerektiğine dikkat edin.

10.1.2 Hızlı sayacın işletimi

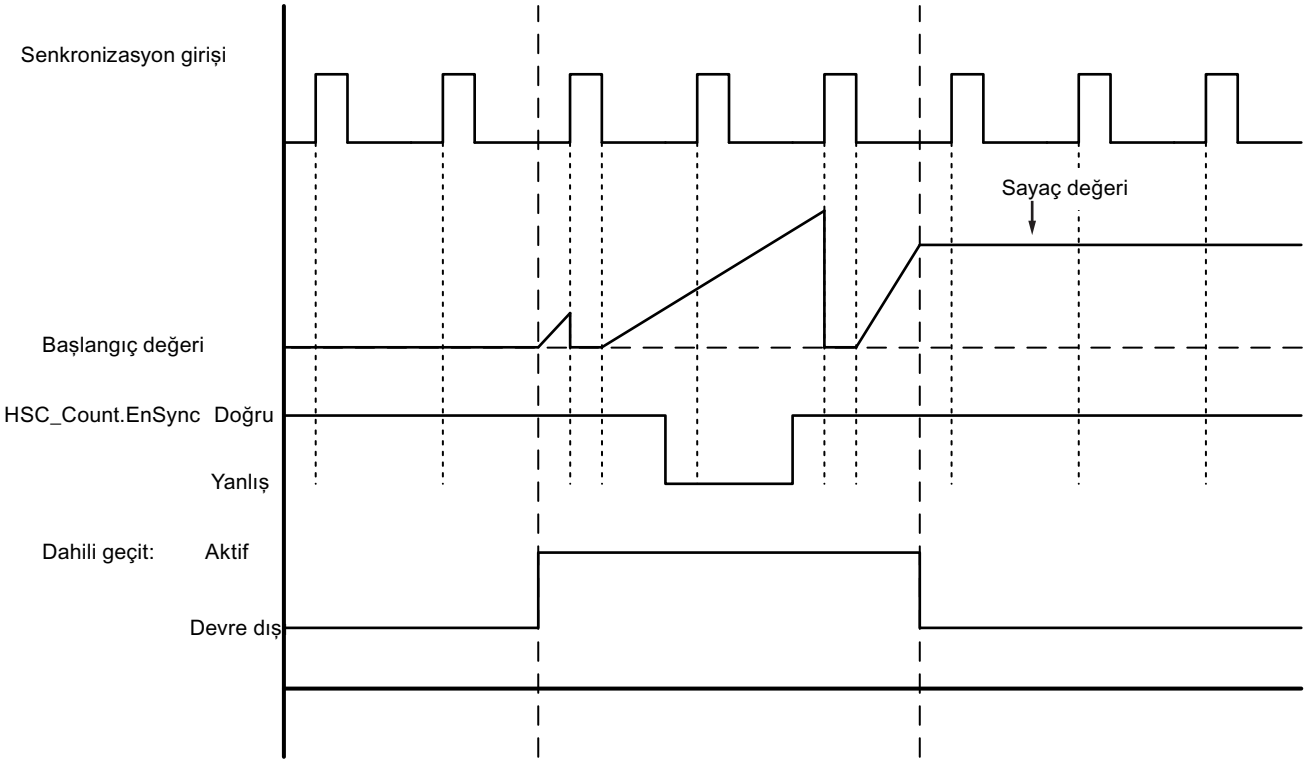
10.1.2.1 Senkronizasyon fonksiyonu

Senkronizasyon fonksiyonuyla sayacı harici bir giriş sinyaliyle başlangıç değerine ayarlayabilirsiniz. Başlangıç değerini, CTRL_HSC_EXT yordamını yürüterek değiştirebilirsiniz. Bu sayede, güncel sayma değeri, harici bir giriş sinyali ortaya çıktığında istenen bir değerle senkronize edilebilir.

Senkronizasyon her zaman giriş sinyali ortaya çıktığında gerçekleştirilir ve dahili geçidin durumundan bağımsız olarak etkili olur. Senkronizasyon fonksiyonunu etkinleştirmek için "HSC_Count.EnSync" Bit'ini doğru olarak ayarlamanız gerekir.

CTRL_HSC_EXT yordamı, senkronizasyon tamamlandıktan sonra HSC_Count.SyncActive durum Bit'ini doğru olarak ayarlar. CTRL_HSC_EXT yordamı, yordamın son yürütülüşünden itibaren hiçbir senkronizasyon işlemi gerçekleşmediyse HSC_Count.SyncActive durum Bit'ini yanlış olarak ayarlar.

Aşağıdaki şekilde, giriş sinyalinin "Aktif High" seviyesi için yapılandırılması durumunda senkronizasyonla ilgili bir örnek gösterilmektedir:



NOT

Yapılandırılan giriş filtreleri, dijital girişin kumanda sinyalini geciktirir.

Bu giriş fonksiyonu sadece HSC'nin sayma işletim türü için yapılandırılmış olması durumunda kullanılabilir.

Senkronizasyon fonksiyonunun konfigürasyonuyla ilgili bilgileri, Giriş fonksiyonları (Sayfa 532) başlığında bulabilirsiniz.

10.1.2.2 Geçit (gate) fonksiyonu

Sayma prosesleri, birçok uygulamada diğer olaylarla uyumlu olarak başlatılmalı veya durdurulmalıdır. Bu tür durumlarda sayma işlemi, dahili geçidin fonksiyonu üzerinden başlatılır ve durdurulur. Her HSC kanalı iki geçide sahiptir: Bir yazılım geçidi ve bir donanım geçidi. Bu geçitlerin durumu, dahili geçidin durumunu belirler (bkz. aşağıdaki tablo).

Yazılım geçidi açıksa ve donanım geçidi açıksa veya yapılandırılmadıysa dahili geçit açılır. Dahili geçit kapalıysa sayma işlemi başlatılır. Dahili geçit kapalıysa, diğer tüm sayma darbeleri göz ardı edilir ve sayma işlemi durdurulur.

Tablo 10-7 Geçit fonksiyonunun durumları

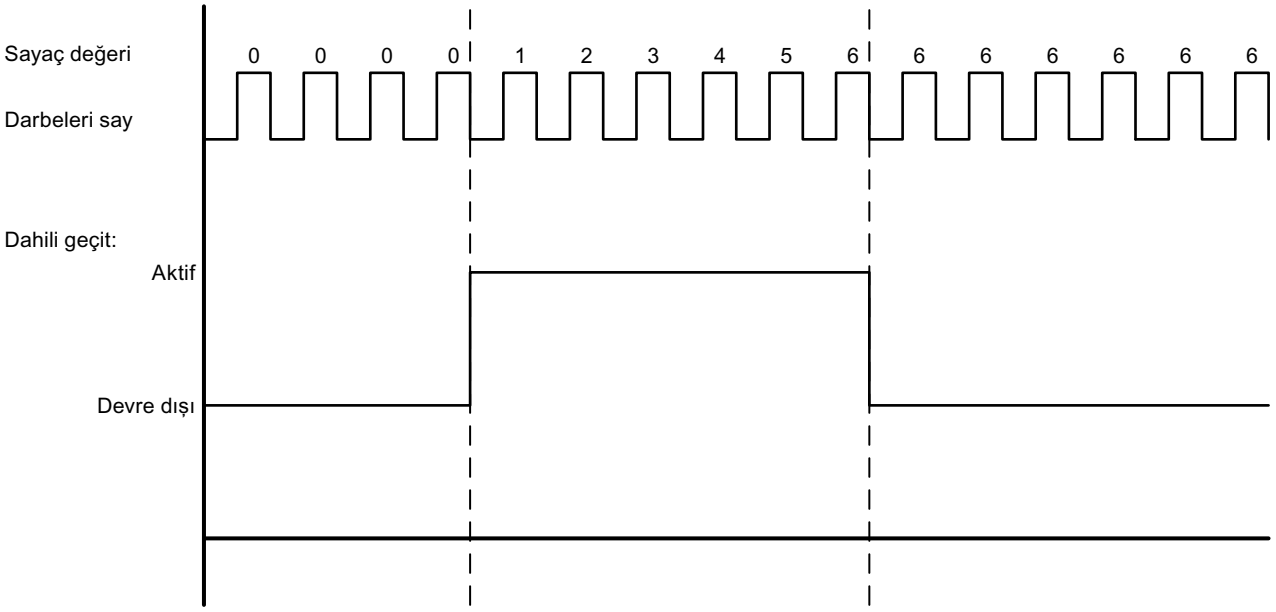
Donanım geçidi	Yazılım geçidi	Dahili geçit
Açık/yapılandırılmamış	Açık	Açık
Açık/yapılandırılmamış	Kapalı	Kapalı
Kapalı	Açık	Kapalı
Kapalı	Kapalı	Kapalı

"açık" terimi, geçidin aktif durumu olarak tanımlanmıştır. Aynı şekilde "kapalı" terimi de geçidin devre dışı durumu olarak tanımlanmıştır.

Yazılım geçidine, CTRL_HSC_EXT yordamının SDT'sinde "HSC_Count.EnHSC" serbest bırakma Bit'i ile kumanda edebilirsiniz. Yazılım geçidini açmak için "HSC_Count.EnHSC" Bit'ini doğru olarak; yazılım geçidini kapatmak içinse "HSC_Count.EnHSC" Bit'ini yanlış olarak ayarlayın. Yazılım geçidinin durumunu uygulamak için CTRL_HSC_EXT yordamını yürütün.

Donanım geçidi opsiyoneldir ve bu bileşeni, HSC'nin özellikleri için kullanılan aralıkta etkinleştirebilir veya devre dışı bırakabilirsiniz. Bir sayma işlemine sadece donanım geçidi ile kumanda etmek için yazılım geçidi açık kalmalıdır. Hiçbir donanım geçidi yapılandırmazsanız, donanım geçidi sürekli açık olarak kabul edilir ve geçit durumu, yazılım geçidinin durumuyla aynı olur.

Aşağıdaki şekilde, bir dijital girişe sahip donanım geçidinin açılması ve kapatılmasıyla ilgili bir örnek gösterilmektedir. Dijital giriş, "Aktif High" seviyesi için yapılandırılmıştır:



NOT

Yapılandırılan giriş filtreleri, dijital girişin kumanda sinyalini geciktirir.

Donanım geçidinin fonksiyonu sadece HSC sayma işletim türü için yapılandırıldığı takdirde kullanılabilir. Periyot ve frekans işletim türlerinde dahili geçidin durumu, yazılım geçidinin durumuyla aynıdır.

Periyot işletim türünde yazılım geçidine, "HSC_Period.EnHSC" tarafından kumanda edilir.

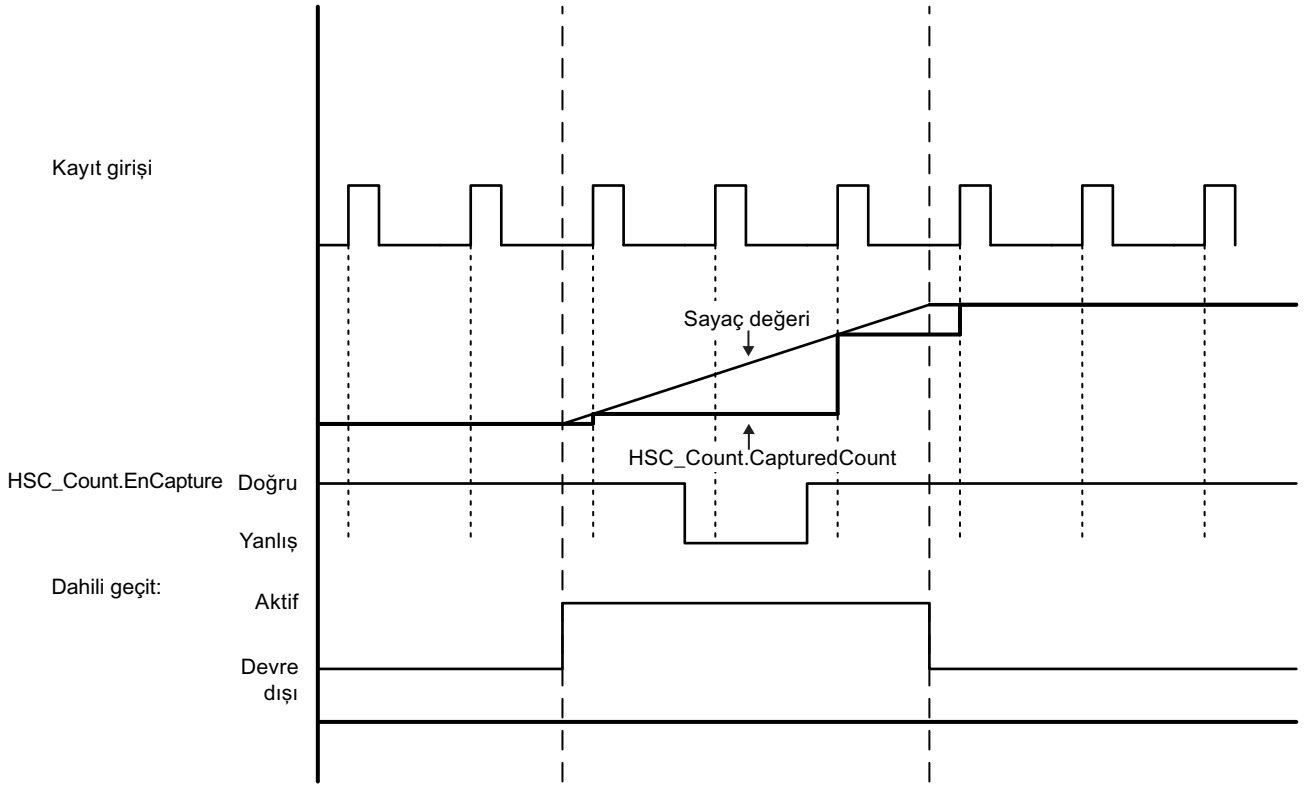
Frekans işletim türünde, yazılım geçidine "HSC_Period.EnHSC" tarafından kumanda edilir.

Geçit fonksiyonunun konfigürasyonuyla ilgili bilgileri Giriş fonksiyonları (Sayfa 532) başlığında bulabilirsiniz.

10.1.2.3 Kaydetme fonksiyonu

Kaydetme fonksiyonuyla, harici bir referans sinyalle güncel sayaç değerini kaydedebilirsiniz. Kaydetme fonksiyonu, "HSC_Count.EnCapture" Bit'i aracılığıyla yapılandırıldıysa ve etkinleştirildiyse, harici bir giriş kenarı ortaya çıktığında güncel sayma değerinin kaydedilmesini sağlar. Kaydetme fonksiyonu, dahili geçidin durumundan bağımsız olarak etkilidir. Geçit kapalı ise, program, değişmeyen sayaç değerini kaydeder. Program, CTRL_HSC_EXT yordamı yürütüldükten sonra "HSC_Count.CapturedCount"ta belirlenen değeri kaydeder.

Aşağıdaki şekilde, yükselen bir kenar durumunda kaydetmek için yapılandırılan kaydetme fonksiyonuyla ilgili bir örnek gösterilmektedir. "HSC_Count.EnCapture" Bit'i CTRL_HSC_EXT yordamı aracılığıyla yanlış olarak ayarlanırsa, kayıt girişi, güncel sayma değerinin kaydedilmesi işlemini tetiklemez.



NOT

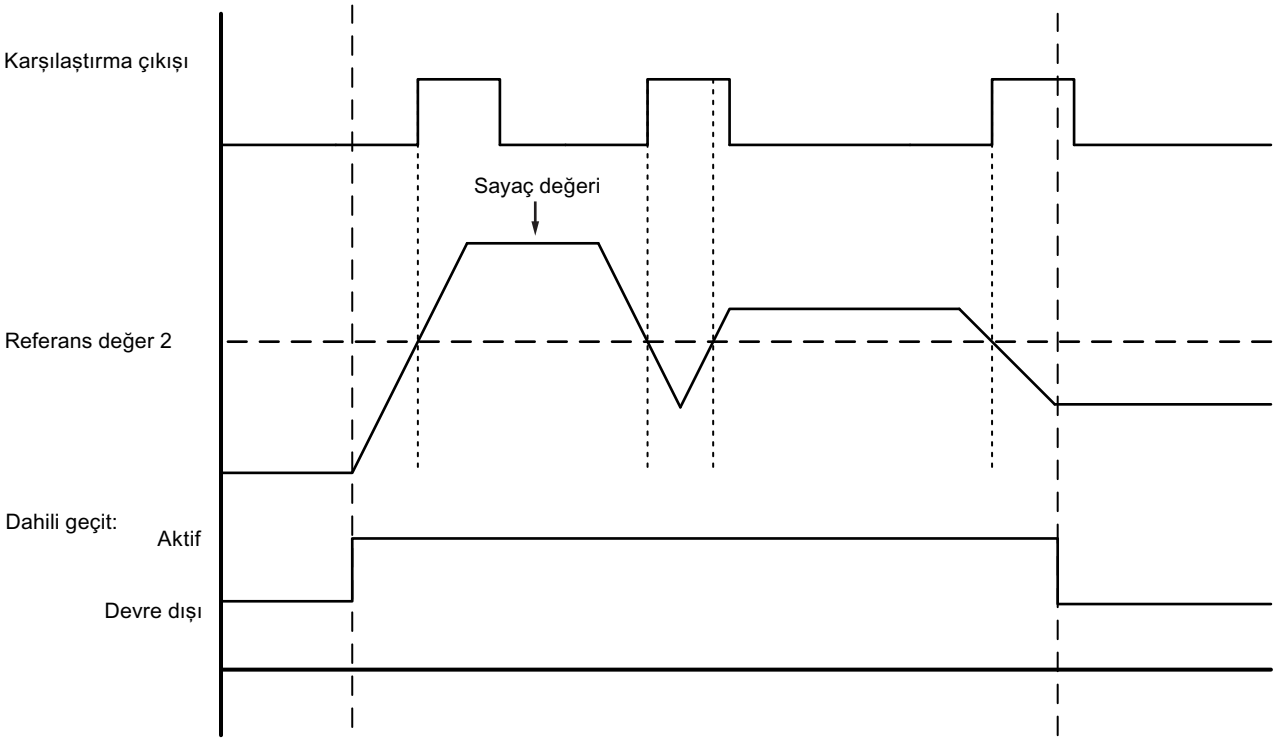
Yapılandırılan giriş filtreleri, dijital girişin kumanda sinyalini geciktirir.

Bu giriş fonksiyonu sadece HSC'nin sayma işletim türü için yapılandırılmış olması durumunda kullanılabilir.

Kaydetme fonksiyonunun konfigürasyonuyla ilgili bilgileri, Giriş fonksiyonları (Sayfa 532) başlığında bulabilirsiniz.

10.1.2.4 Karşılaştırma fonksiyonu

Karşılaştırma çıkışı fonksiyonu, etkinleştirildiği takdirde, yapılandırılan olayın her ortaya çıkışında münferit bir yapılandırılabilir darbe üretir. Olaylara şunlar dahildir: Sayma değeri, bir referans değere veya sayacın taşmasın karşılık gelir. Bir darbe işleniyorsa ve olay yeniden ortaya çıkarsa, olay için hiçbir darbe oluşturulmaz.



NOT

Bu çıkış fonksiyonu sadece HSC'nin sayma işletim türü için yapılandırılmış olması durumunda kullanılabilir.

Karşılaştırma fonksiyonunun konfigürasyonuyla ilgili bilgileri, Çıkış fonksiyonları (Sayfa 533) başlığında bulabilirsiniz.

10.1.2.5 Uygulama

Tipik bir uygulamada HSC aracılığıyla bir açısal enkoderin geri bildirimi denetlenir. Enkoder, tur başına, HSC için saat jeneratörü girişi olarak kullanabileceğiniz belirtilen sayıda sayma değeri sağlar. Ayrıca tur başına bir kez ortaya çıkan ve HSC için senkronizasyon girişi olarak kullanabileceğiniz bir sıfırlama darbesi mevcuttur.

Başlatmak için, kullanıcı programı, ilk referans değeri HSC'ye yüklemeli ve çıkışları bu bileşenin başlangıç durumuna ayarlamalıdır. Çıkışlar, güncel sayma değerinin referans değerden küçük olduğu süre boyunca bu durumda kalır. Senkronizasyon olayı (sıfırlama) ortaya çıktığında veya bir yön değişimi gerçekleştiğinde güncel sayma değeri referans değere eşitse, HSC bir alarmı tetikler.

Her sayaç değeri referans değere karşılık geliyorsa bir alarm olayı ortaya çıkar. Kullanıcı programı, alarm OB'sinde bir sonraki referans değeri HSC'ye yüklemeli ve çıkışları bu bileşenlerin bir sonraki durumuna ayarlamalıdır.

Senkronizasyon girişi tetiklenirse, güncel sayma değeri başlangıç değerine ayarlanır ve bir alarm olayı ortaya çıkar. Bu alarm OB'sinde kullanıcı programı, ilk referans değeri HSC'ye yüklemeli ve çıkışları bu bileşenin başlangıç durumuna ayarlamalıdır. Bu anda, HSC başlangıç durumuna döner ve HSC saymaya devam ederken çevrim tekrarlanır.

Alarmlar HSC'nin sayışından çok daha yavaş bir şekilde oluştuğu için, hızlı yordamların hassas bir şekilde kontrol edilmesini, CPU'nun çevrimi üzerinde nispeten düşük bir etki oluşacak şekilde uygulayabilirsiniz. Alarmları belirli kesme programlarına atayabileceğiniz için, durumun bu şekilde kolayca kontrol edilebilmesi amacıyla her yeni ön ayar ayrı bir kesme programında yüklenebilir. Alternatif olarak tüm alarm olaylarını tek bir kesme programında işleyebilirsiniz.

Bir kullanıcı programı veya harici bir giriş sinyali tarafından tetiklenen geçit fonksiyonu (Sayfa 519), enkoder darbelerinin sayılması fonksiyonunu devre dışı bırakabilir. Geçidi devre dışı bırakma yoluyla milin her hareketini göz ardı edebilirsiniz. Bu, geçit devre dışı kaldığı duruma geçmeden önce enkoderin HSC'ye darbe göndermeye devam edeceği, ancak sayma değerinin en son değerde tutulacağı anlamına gelir. Geçit tekrar aktif duruma gelirse, sayacın devre dışı bırakılmasından önceki son değerden itibaren saymaya devam edilir.

Etkinleştirildiği takdirde kaydetme fonksiyonu (Sayfa 521), harici bir giriş ortaya çıktığında güncel sayma değerinin kaydedilmesini sağlar. Bir proses (örn. bir kalibrasyon rutini) bu fonksiyon yardımıyla, olayların arasında kaç darbenin oluştuğunu belirleyebilir.

Etkinleştirildiği takdirde karşılaştırma fonksiyonu (Sayfa 522), güncel sayma değerinin referans değerlerden veya taşma değerlerinden (sayma sınırlarının aşılması) birine ulaştığı her seferinde tek bir yapılandırılabilir darbe üretir. Belirli bir HSC olayının her ortaya çıkışında başka bir prosesi başlatmak için bu darbeyi sinyal olarak kullanabilirsiniz.

Sayma yönü, kullanıcı programı veya harici bir giriş sinyali tarafından kontrol edilir.

Milin devir sayısını çağırmak için, HSC'yi frekans işletim türü (Sayfa 517) için yapılandırabilirsiniz. Bu fonksiyon, Hz biriminde ön işaretli bir tamsayı değeri sağlar. Başlangıç konumuna geri alma sinyali tur başına bir kez ortaya çıktığı için, sıfırlama sinyalinin frekansını ölçme özelliği, tur başına mil devir sayısı hakkında hızlı bilgi sağlar.

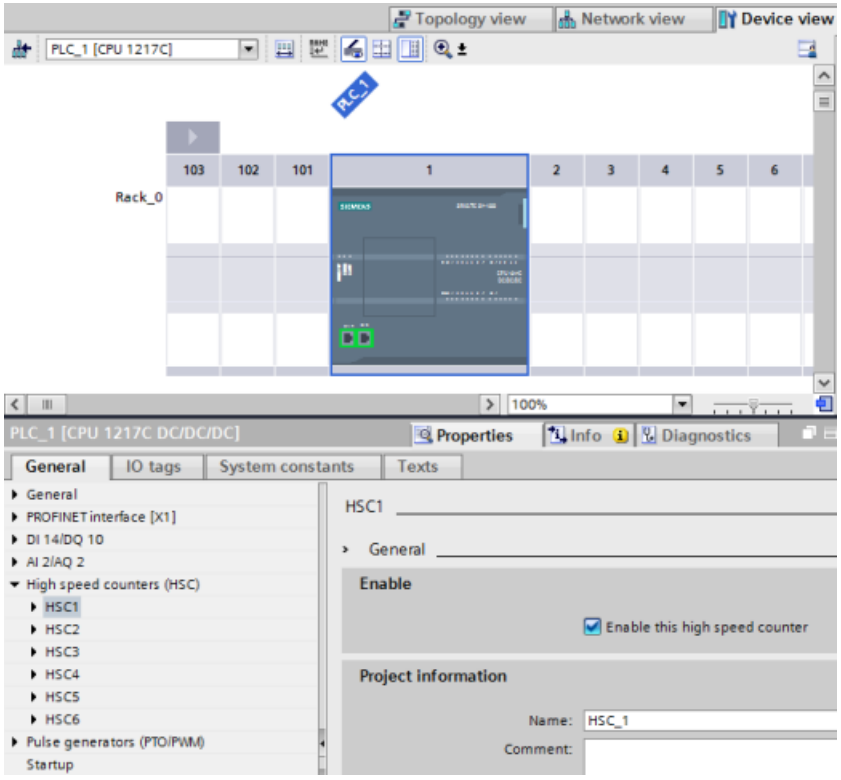
Frekansa kayar noktalı değer olarak ihtiyaç duyarsanız, HSC'yi periyot işletim türü (Sayfa 516) için yapılandırın. Periyot işletim türünde yayınlanan ElapsedTime ve EdgeCount değerleriyle frekans değerini hesaplayabilirsiniz.

10.1.3 Bir hızlı sayacın yapılandırılması

Hızlı sayacı (HSC) ayarlamak için şu adımları uygulayın:

1. Proje gezgininde cihaz konfigürasyonunu seçin.
2. Yapılandırılması gereken CPU'yu seçin.
3. Denetçi penceresinde CPU için kullanılan "Özellikler" sekmesine tıklayın.
4. "Genel" sekmesinde hızlı sayaçlar listesinden etkinleştirmek istediğiniz HSC'yi seçin.

Maksimum altı sayacı yapılandırabilirsiniz (HSC1 ila HSC6). "Bu hızlı sayacı etkinleştir" opsiyonunu seçerek bir HSC'yi etkinleştirin. Etkinleştirmeden sonra STEP 7, bu HSC'ye benzersiz bir standart ad atar. Bu adı, "Ad:" düzenleme alanında değiştirebilirsiniz, ancak ad benzersiz olmalıdır. Etkinleştirilen HSC'lerin adları, "Sistem sabitleri" değişken tablosunda yer alan "HW_Hsc" veri tipinde değişkenler haline gelir ve CTRL_HSC_EXT yordamlarının "HSC" parametresi olarak kullanılabilir. Daha fazla bilgi için bkz. CPU işletiminin yapılandırılması (Sayfa 144).



STEP 7, HSC etkinleştirildikten sonra tek faz sayacını standart konfigürasyon olarak belirler. HSC saat jeneratörü girişinin dijital giriş filtresini belirlemenizin ardından program PLC'ye yüklenebilir ve CPU, sayacı kullanmaya hazır hâle gelir. HSC'nin konfigürasyonunu nasıl değiştireceğinizi, bir sonraki "Sayma türleri" bölümünde öğrenebilirsiniz.

Aşağıdaki tabloda, her konfigürasyon için mevcut olan girişler ve çıkışlara bir genel bakış sunulmaktadır:

Tablo 10-8 HSC için kullanılan sayma türleri

Tip	Giriş 1	Giriş 2	Giriş 3	Giriş 4	Giriş 5	Çıkış 1	Fonksiyon
Dahili yön kumandalı tek faz sayacı	Adım	-	-	-	-	-	Sayma, frekans veya periyot
			Senkronize etme	Geçit	Kaydetme	Karşılaştırma	Sayma
Harici yön kumandalı tek faz sayacı	Adım	Yön	-	-	-	-	Sayma, frekans veya periyot
			Senkronize etme	Geçit	Kaydetme	Karşılaştırma	Sayma
İki faz sayacı	İleriye doğru sayma	Geriye doğru sayma	-	-	-	-	Sayma, frekans veya periyot
			Senkronize etme	Geçit	Kaydetme	Karşılaştırma	Sayma
A/B sayacı	Aşama A	Aşama B	-	-	-	-	Sayma, frekans veya periyot
			Senkronize etme[1]	Geçit	Kaydetme	Karşılaştırma	Sayma
Dörtlü A/B sayacı	Aşama A	Aşama B	-	-	-	-	Sayma, frekans veya periyot
			Senkronize etme[1]	Geçit	Kaydetme	Karşılaştırma	Sayma

[1] Bir açısal enkoderde: Z fazı, referans nokta

10.1.3.1 Sayma türleri

Dört sayma türü veya işletim türü mevcuttur. İşletim türünü değiştirirseniz, HSC için kullanılabilen konfigürasyon opsiyonları da değişir:

- Sayma: Darbe sayısını sayar ve yön kumandasının durumuna bağlı olarak sayma değerini arttırır veya azaltır. Harici G/Ç'ler sayma değerini sıfırlayabilir, sayma fonksiyonunu devre dışı bırakabilir, güncel sayma değerinin kaydedilmesini tetikleyebilir veya belirtilen bir olayda münferit bir darbe oluşturabilir. Çıkış değerleri güncel sayma değeri ve bir kayıt olayı anındaki sayma değeridir.
- Periyot: Belirtilen bir süre boyunca giriş darbelerini sayar. Darbe sayma değerini ve nanosaniye (nsn) cinsinden süreyi yayınlar. Değerler, frekans ölçüm periyodu tarafından belirtilen sürenin sonunda kaydedilir ve hesaplanır. Periyot işletimi CTRL_HSC_EXT yordamı için kullanılabilir, ancak CTRL_HSC yordamı için kullanılamaz.
- Frekans: Giriş darbelerini ve süreyi ölçer ve darbelerin frekansını hesaplar. Program, frekansı Hz biriminde, ön işaretli çift tamsayı olarak yayınlar. Geriye doğru sayılıyorsa değer negatiftir. Değerler, frekans ölçüm periyodu tarafından belirtilen sürenin sonunda kaydedilir ve hesaplanır.
- Hareket kumandası: Teknoloji nesnesi tarafından hareket kumandası için kullanılır ve HSC yordamları için kullanılamaz. Daha fazla bilgiyi, "Hareket kumandası" başlığında bulabilirsiniz.

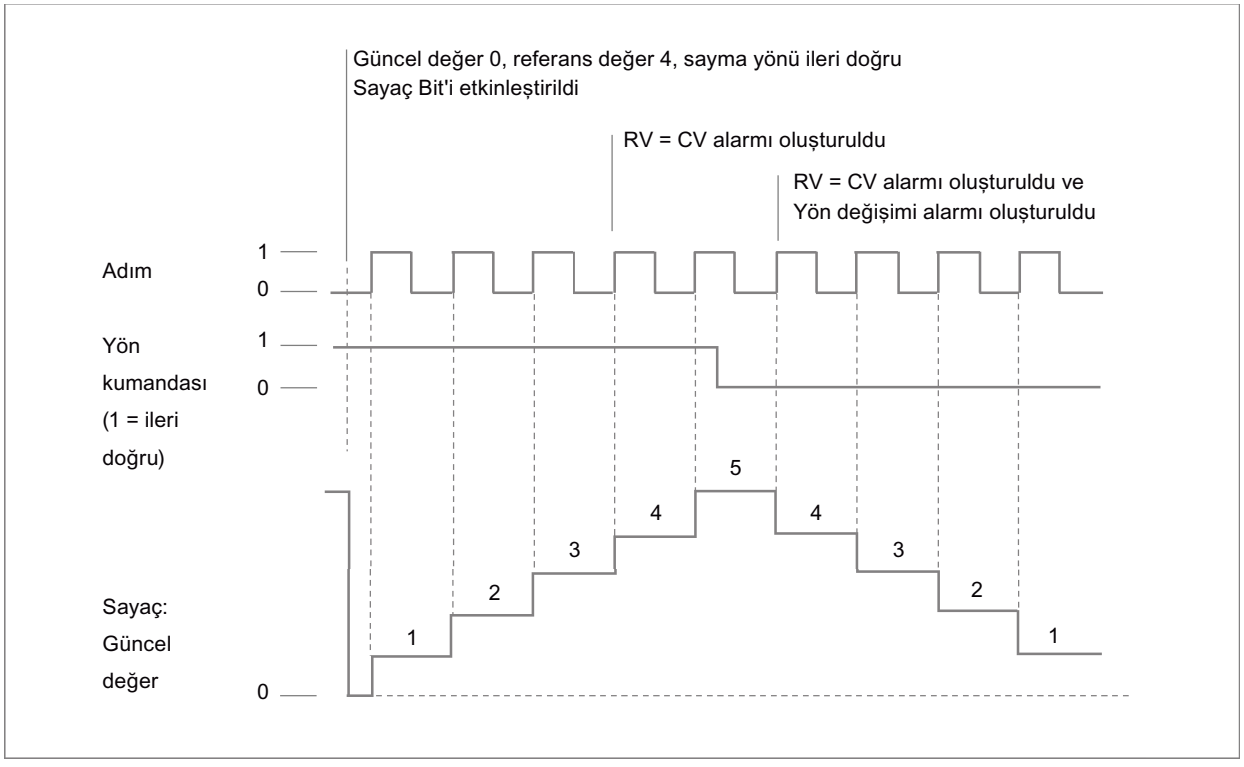
10.1.3.2 İşletim aşaması

HSC için istediğiniz işletim aşamasını seçin. Aşağıdaki dört şekilde, sayaç değerinin ne zaman değiştiği, olayın ne zaman ortaya çıktığı, güncel değerin (CV) referans değere (RV) eşit olduğu ve yön değişimi olayının ne zaman ortaya çıktığı gösterilmektedir.

Tek faz sayacı

Tek faz sayacı (hareket kumandası için temin edilemez) darbeleri sayar:

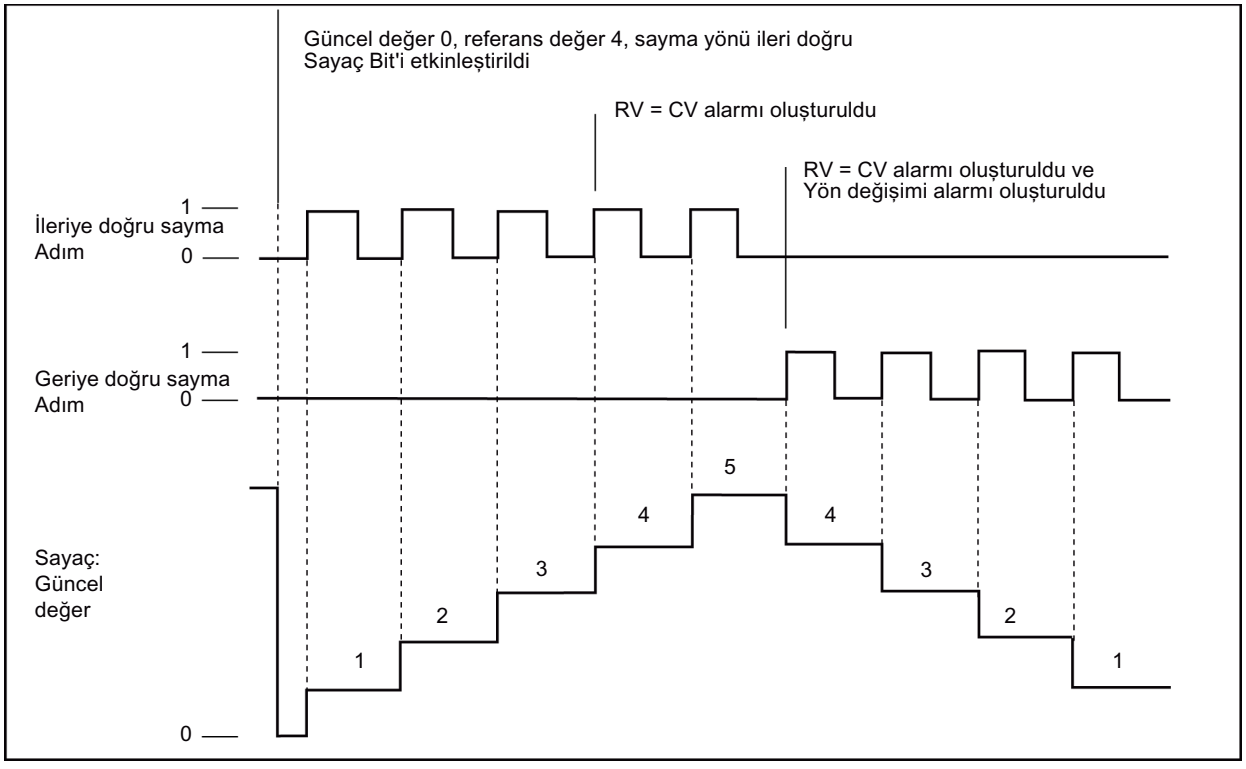
- Kullanıcı programı (dahili yön kumandası):
  - 1 ileriye doğrudur
  - -1 geriye doğrudur
- Donanım girişi (harici yön kumandası):
  - High seviyesi ileriye doğrudur.
  - Low seviyesi geriye doğrudur.



İki faz sayacı

İki faz sayacı:

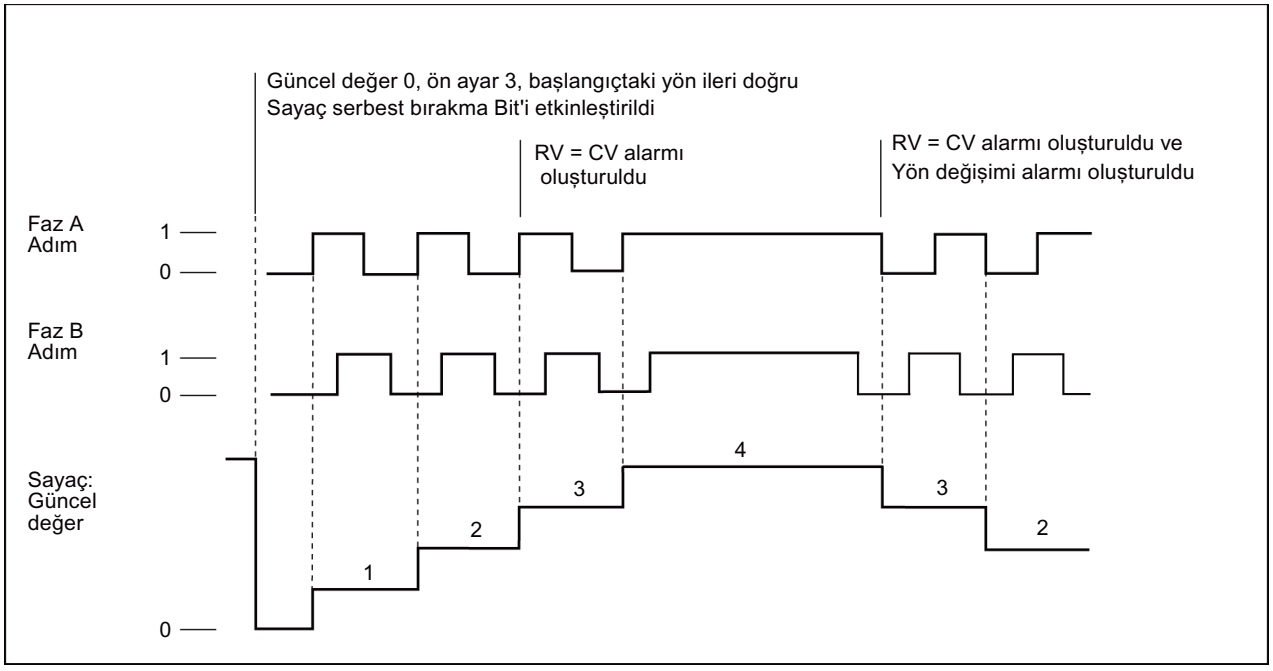
- İleriye doğru sayım girişinde ileriye doğru sayar
- Geriye doğru sayım girişinde geriye doğru sayar



A/B sayacı

A/B sayacının sayma şekli:

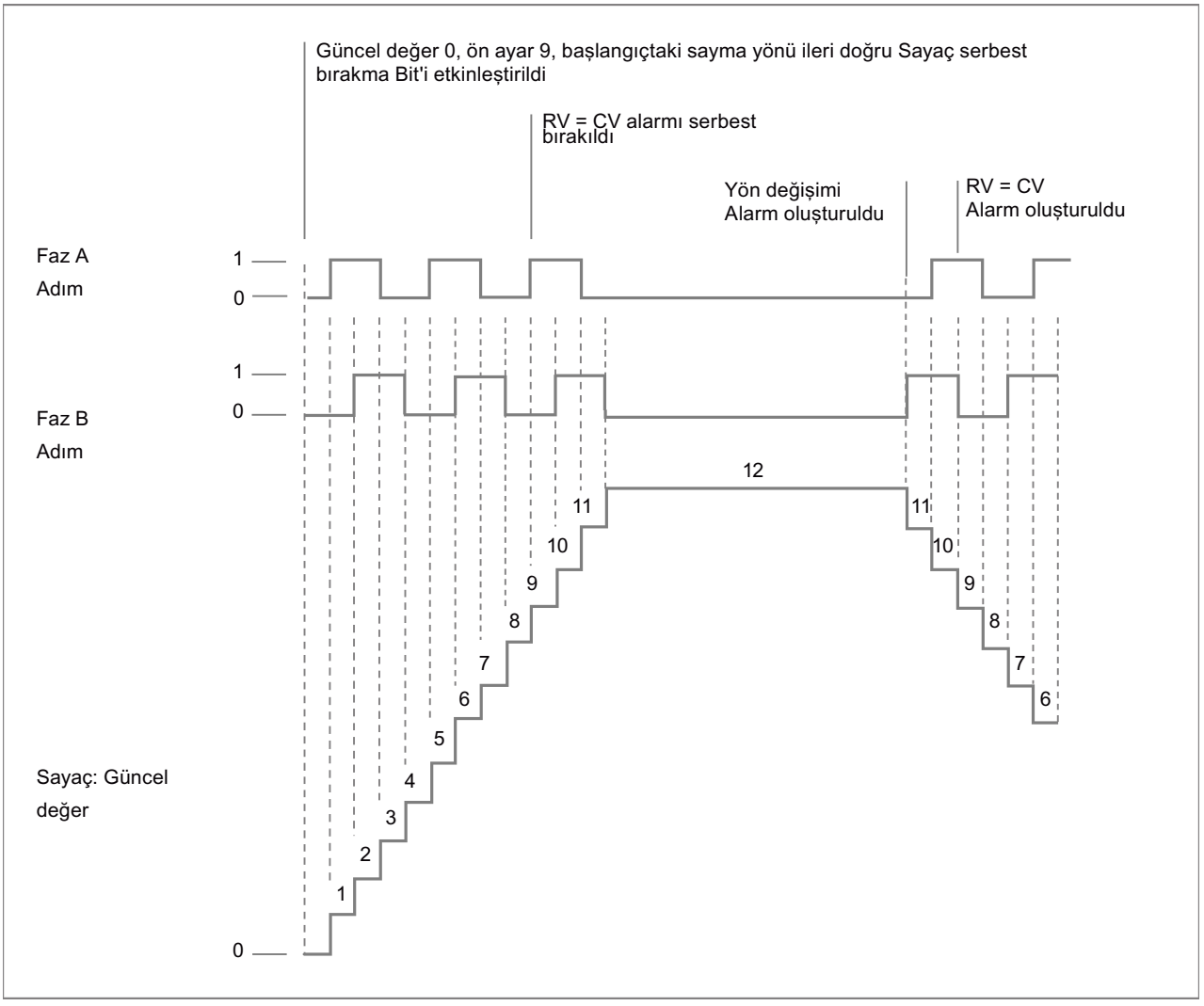
- Sayaç girişi B Low ise, sayaç girişi A'da bir yükselen kenar mevcutken ileriye doğru
- Sayaç girişi B Low ise, sayaç girişi A'da bir alçalan kenar mevcutken geriye doğru



Dörtlü A/B sayacı

Dörtlü A/B sayacının sayma şekli:

- Sayaç girişi B Low ise, sayaç girişi A'da bir yükselen kenar mevcutken ileriye doğru
- Sayaç girişi B High ise, sayaç girişi A'da bir alçalan kenar mevcutken ileriye doğru
- Sayaç girişi A High ise, sayaç girişi B'de bir yükselen kenar mevcutken ileriye doğru
- Sayaç girişi A Low ise, sayaç girişi B'de bir alçalan kenar mevcutken ileriye doğru
- Sayaç girişi A Low ise, sayaç girişi B'de bir yükselen kenar mevcutken geriye doğru
- Sayaç girişi A High ise, sayaç girişi B'de bir alçalan kenar mevcutken geriye doğru
- Sayaç girişi B High ise, sayaç girişi A'da bir yükselen kenar mevcutken geriye doğru
- Sayaç girişi B Low ise, sayaç girişi A'da bir alçalan kenar mevcutken geriye doğru



10.1.3.3 Başlangıç değerleri

CPU, her RUN işletim durumuna geçişinde başlangıç değerlerini yükler. Başlangıç değerleri sadece sayma işletim türünde kullanılır:

- Başlangıçtaki sayaç değeri: CPU STOP işletim durumundan RUN işletim durumuna geçerse veya program senkronizasyon girişini tetiklerse, program, güncel sayma değerini başlangıçtaki sayma değerine ayarlar.
- Başlangıçtaki referans değer: Güncel sayma değeri referans değere ulaşırsa, program, ilgili fonksiyonlar ayarlandığı sürece bir alarm ve/veya bir çıkış darbesi üretir.
- Başlangıçtaki referans değer 2: Güncel sayma değeri referans değer 2'ye ulaşırsa, program, fonksiyon ayarlandığı sürece bir çıkış darbesi üretir.
- Başlangıçtaki üst limit: Maksimum sayma değeri. Standart değer, +2.147.483.647 darbenin mümkün olan en büyük değeridir.
- Başlangıçtaki alt limit: Minimum sayma değeri. Standart değer, -2.147.483.648 darbenin mümkün olan en küçük değeridir.

Yukarıda belirtilen değerler ve sayacın bir limite ulaşıldığındaki davranışı, sadece sayma işletim türünde mevcuttur. Değerleri ve davranışı, CTRL_HSC_EXT yordamının SDT HSC_Count parametresiyle uyarlayabilirsiniz.

10.1.3.4 Giriş fonksiyonları

Adım ve yön girişleri, işletim aşamasını temel alarak sayaç olaylarını ve sayma yönünü belirler. Sayma işletim türünde sadece senkronizasyon, kaydetme ve geçit girişlerini kullanabilir ve girişleri, çeşitli tetikleme türleri için münferit olarak etkinleştirebilir ve yapılandırabilirsiniz.

Senkronizasyon girişi

Senkronizasyon girişi, güncel sayma değerini başlangıç değerine (sayacın başlangıç değeri) ayarlar. Senkronizasyon girişini, genelde sayacı 0 değerine getirmek için kullanabilirsiniz. Senkronizasyonu, giriş bağlantısı aşağıdaki durumlardan birinde ise tetikleyebilirsiniz:

- High
- Low
- Low iken High olarak değişir
- High iken Low olarak değişir
- High iken Low olarak değişir veya Low iken High olarak değişir

Kayıt girişi

Kayıt girişi, kaydedilen sayma değerini, kaydetme girişinin tetiklendiği anda kaydedilen sayma değerine ayarlar. Kaydetme fonksiyonunu, giriş bağlantısı aşağıdaki durumlardan birinde ise tetikleyebilirsiniz:

- Low iken High olarak değişir
- High iken Low olarak değişir
- High iken Low olarak değişir veya Low iken High olarak değişir

Geçit girişi

Geçit girişi, HSC'nin sayma işlemini durdurur. Geçidi, giriş bağlantısı aşağıdaki durumlardan birinde ise tetikleyebilirsiniz:

- High
- Low

10.1.3.5 Çıkış fonksiyonu

Karşılaştırma çıkışı fonksiyonu, HSC için kullanılan tek fonksiyondur ve sadece sayma işletim türünde mevcuttur.

Karşılaştırma çıkışını, aşağıdaki olaylardan biri ortaya çıktığında tek bir darbe üretmek için yapılandırabilirsiniz:

- Sayaç referans değere karşılık geliyor (sayma yönü ileriye doğru)
- Sayaç referans değere karşılık geliyor (sayma yönü geriye doğru)
- Sayaç referans değere karşılık geliyor (sayma yönü ileriye veya geriye doğru)
- Sayaç referans değer 2'ye karşılık geliyor (sayma yönü ileriye doğru)
- Sayaç referans değer 2'ye karşılık geliyor (sayma yönü geriye doğru)
- Sayaç, referans değer 2'ye karşılık geliyor (sayma yönü ileriye veya geriye doğru)
- Pozitif taşma
- Negatif taşma

Çıkış darbelerini 1 ila 500 msn'lik bir çevrim süresiyle yapılandırabilirsiniz; standart çevrim süresi 10 msn'dir. Darbe süresini veya devreye alma süresini % 1 ila 100'lük bir aralıkta belirleyebilirsiniz; standart darbe süresi % 50'dir.
Güncel darbe, çevrimini henüz tamamlamadığı için, belirtilen çevrim süresi içinde birden fazla karşılaştırma çıkışı olayı ortaya çıkarsa, bu olayın çıkış darbeleri kaybolur. Darbe tamamlanınca (yapılandırılan çevrim süresi dolunca), darbe jeneratörü yeni bir darbe oluşturmak için kullanılabilir.

10.1.3.6 Alarm olayları

Olay konfigürasyonu için kullanılan aralıkta açılır menüden bir proses alarmı OB'si seçebilir (veya yeni bir OB oluşturabilir) ve bunu bir HSC olayına atayabilirsiniz. Alarmın önceliği, 2 - 26 aralığında yer alır, burada 2 en düşük öncelik ve 26 en yüksek önceliktir. HSC'nin konfigürasyonuna bağlı olarak aşağıdaki olaylar kullanılabilir:

- Sayaç değeri olayı referans değere karşılık gelir: Referans değere karşılık gelen sayma değeri olayı, HSC'nin sayma değeri referans değere ulaştığında ortaya çıkar. Referans değeri, başlangıç referans değeri için kullanılan aralıkta konfigürasyon sırasında veya CTRL_HSC_EXT yordamıyla "NewReference1" değerini güncelleyerek belirleyebilirsiniz. Daha fazla bilgi için bkz. "İşletim aşaması (Sayfa 527)".
- Senkronizasyon olayı: Bir senkronizasyon işlemi, senkronizasyon girişini etkinleştirdiğiniz ve tetiklediğiniz takdirde her zaman ortaya çıkar.
- Yön değişimi olayı: Bir yön değişimi olayı, sayma yönü değişirse ortaya çıkar. Daha fazla bilgi için bkz. "İşletim aşaması (Sayfa 527)".

10.1.3.7 Donanım girişinin bağlantı yerleşimi

Etkinleştirdiğiniz her HSC girişi için ya CPU'da ya da opsiyonel sinyal kartında (iletişim ve sinyal modülleri HSC girişlerini desteklemez) istediğiniz girişi seçin. Bir girişi seçerseniz, STEP 7, seçimin yanındaki maksimum frekans değerini görüntüler. Tüm sinyal frekanslarının filtreden geçebilmesi için dijital giriş filtresinin ayarlarının muhtemelen uyarlanması gerekir. HSC giriş filtresini nasıl ayarlayacağınızı, "Dijital girişlerin filtre sürelerinin yapılandırılması (Sayfa 147)" bölümünden öğrenebilirsiniz.

NOT

CPU'nun ve SB'nin (Firmware V4'ten itibaren) giriş kanalları, yapılandırılabilir giriş filtresi sürelerine sahiptir

Daha eski Firmware sürümlerinde HSC giriş kanalları ve filtre süreleri de belirlenmiş durumdaydı ve değiştirilemiyordu.

Sürüm V4'ten itibaren giriş kanallarını ve filtre sürelerini belirleyebilirsiniz. Giriş filtresi için standart ayar 6,4 msn'dir, bu değer maksimum sayma frekansını 78 Hz ile sınırlar. Sisteminizin yapısına bağlı olarak daha yüksek veya daha düşük frekansları saymak için filtre ayarlarını değiştirebilirsiniz.

⚠ İKAZ

Dijital giriş kanalları için filtre ayarı değiştirildiğinde oluşabilecek tehlikeler

Bir dijital giriş kanalının o ana kadar kullanılan ayarı değiştirilirse, filtrenin yeni girişlere eksiksiz olarak tepki göstermesi için muhtemelen 20,0 msn'ye kadar bir süre boyunca 0 seviyeli yeni bir giriş değeri mevcut olmalıdır. Bu süre zarfında, 20,0 msn'den kısa olan "0" seviyeli kısa darbeler muhtemelen algılanmaz veya sayılmaz.

Filtre sürelerinin değiştirilmesi, makinenin veya prosesin beklenmeyen bir davranışına neden olabilir, bu da ağır veya hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yeni filtre süresinin hemen etkili olması için CPU kapatılmalı ve yeniden açılmalıdır.

Seçim sırasında aşağıdaki tabloyu kullanın ve CPU'nun ve SB'nin kullanılan giriş kanallarının, proses sinyallerinizdeki maksimum darbe hızlarını desteklemesine dikkat edin.

Tablo 10-9 CPU girişi: Maksimum frekans

CPU	CPU giriş kanalı	İşletim aşaması: Tek faz sayacı veya iki faz sayacı	İşletim aşaması: A/B sayacı veya dörtlü A/B sayacı
1211C	Ea.0 - Ea.5	100 kHz	80 kHz
1212C	Ea.0 - Ea.5	100 kHz	80 kHz
	Ea.6, Ea.7	30 kHz	20 kHz
1214C ve 1215C	Ea.0 - Ea.5	100 kHz	80 kHz
	Ea.6 - Eb.5	30 kHz	20 kHz

CPU	CPU giriş kanalı	İşletim aşaması: Tek faz sayacı veya iki faz sayacı	İşletim aşaması: A/B sayacı veya dörtlü A/B sayacı
1217C	Ea.0 - Ea.5	100 kHz	80 kHz
	Ea.6 - Eb.1	30 kHz	20 kHz
	Eb.2 - Eb.5 (0,2+, 0,2- ila 0,5+, 0,5-)	1 MHz	1 MHz

Tablo 10-10 SB girişi: Maksimum frekans (opsiyonel kart)

Sinyal kartı (SB)	SB giriş kanalı	İşletim aşaması: Tek faz sayacı veya iki faz sayacı	İşletim aşaması: A/B sayacı veya dörtlü A/B sayacı
SB 1221, 200 kHz	Ee.0 - Ee.3	200 kHz	160 kHz
SB 1223, 200 kHz	Ee.0, Ee.1	200 kHz	160 kHz
SB 1223	Ee.0, Ee.1	30 kHz	20 kHz

Bir giriş bir HSC fonksiyonuna atanırken aynı girişi birden fazla HSC fonksiyonuna atayabilirsiniz. Örnek: İki HSC'nin sayma değerinin eş zamanlı senkronizasyonu için E0.3'ün HSC1'in senkronizasyon girişine ve HSC2'nin senkronizasyon girişine atanması geçerli bir konfigürasyondur. Ancak derleme sırasında bir uyarı oluşturur.

Mümkünse, bir girişe aynı HSC'nin birden fazla giriş fonksiyonunu atamayın. Örnek: Sayma değerinin senkronizasyonu ve saymanın devre dışı bırakılması işleminin eş zamanlı senkronizasyonu için E0.3'ün senkronizasyon girişine ve HSC 1'in geçit girişine atanması da geçerli bir konfigürasyondur. Bu konfigürasyonu yapabilirsiniz, ancak bu konfigürasyon istenmeyen sonuçlara yol açabilir.

⚠ İKAZ

Birden çok fonksiyonun tek bir dijital giriş kanalına atanmasında ortaya çıkabilecek riskler

Bir HSC'nin birden çok giriş fonksiyonuna atanması, öngörülemeyen sonuçlara yol açabilir. Bir girişte, birden fazla fonksiyonun atandığı bir tetikleme durumu ortaya çıkarsa, fonksiyonların PLC tarafından yürütülme sırası belirlenemez. Bu, yarışma durumu olarak bilinir ve genelde istenmez.

Filtre sürelerinin değiştirilmesi, makinenin veya prosesin beklenmeyen bir davranışına neden olabilir, bu da ağır veya hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Bir yarışma durumunun oluşmasını engellemek için asla aynı HSC'nin ikiden fazla giriş fonksiyonunu aynı giriş bağlantısına atamayın. Bir HSC'de aynı bağlantıya iki giriş fonksiyonu atandıysa, tetikleme durumlarını, bu giriş fonksiyonları asla eş zamanlı olarak ortaya çıkamayacak şekilde belirleyin. Alçalan bir kenarın bir Low seviyesinin başladığı anda ortaya çıktığına ve yükselen bir kenarın bir High seviyesinin başladığı anda ortaya çıktığına dikkat edin.

NOT

Hızlı sayaçlar (HSC'ler) tarafından kullanılan dijital girişleri ve çıkışları cihaz konfigürasyonu sırasında CPU'ya atayabilirsiniz. HSC cihazlarına girişler ve çıkışlar atarsanız, bu girişlerin ve çıkışların değerlerini bir izleme tablosundaki Force fonksiyonunun yardımıyla değiştiremezsiniz. HSC, bu girişler ve çıkışlar üzerinde tam kontrol sahibidir.

10.1.3.8 Donanım çıkışının bağlantı yerleşimi

Karşılaştırma çıkışını kullanıyorsanız boş bir çıkış seçin. Bir çıkışı bir HSC (bir darbe jeneratörü gibi başka teknoloji nesneleri) tarafından kullanılmak üzere yapılandırdığınızda, bu çıkış sadece bu nesneye ait olur. Başka hiçbir bileşen çıkışı kullanamaz ve çıkışa bir değer için Force fonksiyonu uygulanamaz. Tek bir çıkışı birden fazla HSC için veya bir HSC'de ve bir darbe çıkışında kullanmak üzere yapılandırırsanız, program bir derleme hatası üretir.

10.1.3.9 HSC giriş adresleri

Her HSC, güncel sayma değerinin kaydedildiği E bellek alanında, bir çift sözcük kullanır. HSC'yi frekans işletim türü için yapılandırırsanız, frekans bu giriş adresine kaydedilir. Giriş adreslerinin kullanılabilir aralığı E0.0 ila E1023.7 şeklindedir (maksimum başlangıç adresi E1020.0'dır). HSC, başka bir bileşene atanmış olan bir giriş adresiyle çakışan bir giriş adresi kullanamaz. Proses görünümüyle ilgili daha fazla bilgiyi "Kullanıcı programının yürütülmesi (Sayfa 60)" başlığında bulabilirsiniz.
Aşağıdaki tabloda her HSC için kullanılan standart adresler gösterilmektedir:

Tablo 10-11 HSC standart adresleri

Hızlı sayaç (HSC)	Güncel değerin veri tipi	Güncel değerin standart adresi
HSC1	DInt	ID 1000
HSC2	DInt	ID 1004
HSC3	DInt	ID 1008
HSC4	DInt	ID 1012
HSC5	DInt	ID 1016
HSC6	DInt	ID 1020

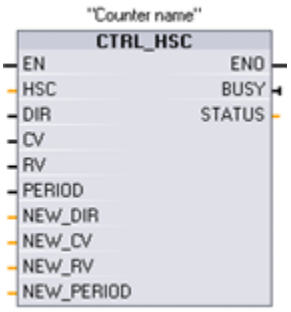
10.1.3.10 Donanım kodu

Her HSC, HSC_CTRL ve HSC_CTRL_EXT yordamları tarafından kullanılan benzersiz bir donanım koduna sahiptir. Donanım kodu için kullanılan PLC değişkenini "Sistem sabitleri" bölümünde bulabilirsiniz. "HSC_1" adındaki bir HSC, "Local~HSC_1" değişkenine ve "Hw_Hsc" veri tipine sahiptir. Bu değişken, açılır menüde CTRL_HSC_EXT yordamları için HSC girişi seçilirken de görüntülenir.

10.1.4 Eski CTRL_HSC yordamı (hızlı sayaca kumanda et)

10.1.4.1 CTRL_HSC (hızlı sayaca kumanda et)

Tablo 10-12 CTRL_HSC (genel sayma görevleri için) yordamı

LAD/FBD	SCL	Açıklama
	**"CTRL_HSC_1_DB" (hsc:=W#16#0, dir:=False, cv:=False, rv:=False, period:=False, new_dir:=0, new_cv:=L#0, new_rv:=L#0, new_period:=0, busy=>_bool_out_, status=>_word_out_);**	Her CTRL_HSC (hızlı sayaca kumanda et) yordamı, sayaç verilerini kaydetmek için bir DB'ye kaydedilen bir yapıyı kullanır. CTRL_HSC yordamını editörde eklerseniz DB'yi atayabilirsiniz.

[1] Yordamı eklerseniz, STEP 7, ilgili DB'yi oluşturmak için "Çağırma seçenekleri" iletişim kutusunu görüntüler.

[2] SCL örneğinde "CTRL_HSC_1_DB", örnek DB'nin adıdır.

Tablo 10-13 Parametreler için kullanılan veri tipleri

Parametre	Deklarasyon	Veri tipi	Açıklama
HSC	IN	HW_HSC	HSC kodu
DIR[1, 2]	IN	Bool	1 = Yeni yön talep etme
CV[1]	IN	Bool	1 = Yeni bir sayma değeri ayarlama talebi
RV[1]	IN	Bool	1 = Yeni bir referans değer ayarlama talebi
PERIOD[1]	IN	Bool	1 = Yeni bir zaman aralığı ayarlama talebi (sadece frekans ölçümünde)
NEW_DIR	IN	Int	Yeni yön: 1= ileriye doğru, -1= geriye doğru
NEW_CV	IN	DInt	Yeni sayma değeri
NEW_RV	IN	DInt	Yeni referans değer
NEW_PERIOD	IN	Int	Yeni zaman aralığı değeri milisaniye cinsinden belirtilir (sadece frekans ölçümünde). Sadece 10, 100 veya 1000 milisaniye değerlerine izin verilir. 1000 = 1 saniye 100 = 0,1 saniye 10 = 0,01 saniye
BUSY[3]	OUT	Bool	Fonksiyon dolu
STATUS	OUT	Word	Yürütme koşulu

[1] Bir parametre değerinin güncellenmesi talep edilmezse, ilgili giriş değerleri göz ardı edilir.

[2] DIR parametresi sadece, yapılandırılan sayma yönü için program kumandası (dahili yön kumandası) önceden belirtildiyse geçerlidir. Bu parametrenin HSC cihaz konfigürasyonunda nasıl kullanacağını belirleyebilirsiniz.

[3] CPU'daki veya SB'deki bir HSC'de BUSY parametresi her zaman 0 değerine sahiptir.

Her HSC için kullanılan parametreleri CPU'nun cihaz konfigürasyonunda yapılandırabilirsiniz: Sayma/frekans fonksiyonu, sıfırlama opsiyonları, alarm olayı konfigürasyonu, donanım G/Ç ve sayma değerinin adresi.

HSC için kullanılan parametrelerden bazıları, sayma işlemi için program kumandasını önceden belirtmek üzere kullanıcı programınız tarafından değiştirilebilir:

- Sayma yönünün bir NEW_DIR değerine ayarlanması
- Güncel sayma değerinin yeni bir NEW_CV değerine ayarlanması
- Referans değerin yeni bir NEW_RV değerine ayarlanması
- Zaman aralığı değerinin (sadece frekans ölçümünde) yeni bir NEW_PERIOD değerine ayarlanması

CTRL_HSC yordamı yürütülürken aşağıdaki Boole markörleri 1'e ayarlandıysa, ilgili NEW_xxx değeri sayaca yüklenir. CTRL_HSC yordamının bir yürütülüşünde birden fazla talep (birden fazla markör eş zamanlı olarak ayarlandı) işlenir.

- DIR = 1, bir NEW_DIR değerini yüklemeye yönelik bir taleptir, 0 = değişiklik yok
- CV= 1, bir NEW_CV değerini yüklemeye yönelik bir taleptir, 0 = değişiklik yok
- RV= 1, bir NEW_RV değerini yüklemeye yönelik bir taleptir, 0 = değişiklik yok
- PERIOD= 1, bir NEW_PERIOD değerini yüklemeye yönelik bir taleptir, 0 = değişiklik yok

Bir hata oluştuğunda ENO 0'a ayarlanır ve STATUS çıkışı bir koşul kodu belirtir:

Tablo 10-14 Execution condition codes

STATUS (W#16#)	Açıklama
0	Hata yok
80A1	HSC kodu hiçbir HSC'yi adreslemiyor
80B1	NEW_DIR'de geçersiz değer
80B2	NEW_CV'de geçersiz değer
80B3	NEW_RV'de geçersiz değer
80B4	NEW_PERIOD'da geçersiz değer
80C0	Hızlı sayaca birden çok erişim Sayma türü (Sayfa 526) için "Periyot" veya "Hareket kumandası" ayarlandıysa bu hata ortaya çıkabilir. Bu sayma türleri, CTRL_HSC yordamı için geçersizdir ve sadece CTRL_HSC_EXT yordamı tarafından desteklenir.
80D0	Hızlı sayaç (HSC) CPU donanım konfigürasyonunda etkinleştirilmedi

10.1.4.2 CTRL_HSC'nin kullanımı

CTRL_HSC yordamı tipik olarak, sayacın proses alarmı olayı tetiklendiğinde yürütülen bir proses alarmı OB'ye yerleştirilir. Örneğin bir CV=RV olayı aracılığıyla sayaç kesmesi tetiklenirse, bir proses alarmı OB, CTRL_HSC yordamını yürütür ve referans değeri bir NEW_RV değerinin yüklenmesi yoluyla değiştirebilir.

Güncel sayma değeri CTRL_HSC için kullanılan parametrelerde öngörülmüştür. Güncel sayma değerinin kaydedildiği proses görünümünün adresi, donanım konfigürasyonunda hızlı sayaca atanır. Sayma değerini program lojiği üzerinden doğrudan okuyabilirsiniz. Programınızda yayınlanan değer, sayacın okunduğu an için doğru bir sayma değeridir. Sayaç, hızlı olayları saymaya devam eder. Bu nedenle gerçek sayma değeri, programınız eski bir sayma değeriyle tamamlanmadan önce değişebilir.

10.1.4.3 HSC'nin güncel sayma değeri

CPU, her HSC'nin güncel değerini bir girişin (E) adresine kaydeder. Aşağıdaki tabloda, her HSC'nin güncel değeri için kullanılan standart adresler gösterilmektedir. Cihaz konfigürasyonunda CPU'nun özelliklerini değiştirerek güncel değer için kullanılan e-adresini değiştirebilirsiniz.

Hızlı sayaçlar, güncel sayma değerini bir DInt değerine kaydeder. Bir DInt sayma değeri, -2147483648 ila +2147483647 şeklinde bir aralığa sahiptir. Aralık sınırlarını yapılandırabilirsiniz. Daha fazla bilgi için bkz. "Başlangıç değerleri (Sayfa 531)".

Sayaç, ileriye doğru saymada maksimum pozitif değerden maksimum negatif değere ve geriye doğru saymada ise maksimum negatif değerden maksimum pozitif değere geçer. Frekans, Hz biriminde yayınlanır (örnek: 123,4 Hz, 123 olarak yayınlanır).

Tablo 10-15 HSC standart adresleri

HSC	Güncel değerin veri tipi	Güncel değerin standart adresi
HSC1	DInt	ED1000
HSC2	DInt	ED1004
HSC3	DInt	ED1008
HSC4	DInt	ED1012
HSC5	DInt	ED1016
HSC6	DInt	ED1020

10.2 Hareket kumandası

10.2.1 Hareket kumandasına genel bakış

TIA Portal, S7-1200 CPU'sunun hareket kumandası fonksiyonu ile birlikte adım motorlarına ve servo motorlara kumanda ederken sizi destekler:

- Konumlama eksenini komut tablosunun teknoloji nesnelerini TIA Portal'da yapılandırabilirsiniz. S7-1200 CPU'su, bu teknoloji nesneleriyle sürücülere kumanda eden çıkışlara kumanda eder.
- Kullanıcı programında, hareket kumandası yordamlarının yardımıyla eksene kumanda edebilir ve sürücünüzün hareket yordamlarını başlatabilirsiniz.

10.2.2 Hareket kumandası için kullanılan donanım bileşenleri

S7-1200 CPU'su ile birlikte bir hareket kumandası uygulaması için kullanılan temel donanım konfigürasyonu aşağıdaki bileşenlerden oluşur.

S7-1200 CPU'su

S7-1200 CPU'su bir programlanabilir kontrol sisteminin fonksiyonlarını, sürücülerin işletimi için kullanılan hareket kumandası fonksiyonlarıyla birleştirir. Hareket kumandası fonksiyonu, sürücülerin kumandasını ve denetimini üstlenir.

Sinyal kartı

Sinyal kartlarıyla, CPU'yu ek girişler ve çıkışlarla genişletebilirsiniz. Dijital çıkışları, ihtiyaca bağlı olarak sürücülerin kumandasına yönelik darbe jeneratörü çıkışları olarak kullanabilirsiniz. Röle çıkışları bulunan CPU'larda, röleler ihtiyaç duyulan anahtarlama frekanslarını desteklemediği için darbe sinyali entegre çıkışlarda yayınlanamaz. Bu CPU'larda darbe dizisi (PTO) ile çalışmak için, dijital çıkışlara sahip bir sinyal kartı kullanmanız gerekir.

Analog çıkışları, ihtiyaca bağlı olarak analog sürücülerin kumandası için kullanabilirsiniz.

PROFINET

PROFINET arayüzü üzerinden S7-1200 CPU'su ile programlama cihazı arasında bir çevrimiçi bağlantı kurun. CPU'nun çevrimiçi fonksiyonlarının yanı sıra hareket kumandası için başka işletime alma ve teşhis fonksiyonları mevcuttur.

PROFINET ayrıca PROFIdrive özelliğine sahip sürücülerin ve enkoderlerin bağlantısı için PROFIdrive profili de destekler.

Sürücüler ve enkoderler

Sürücüler eksenin hareket etmesine izin verir. Enkoderler, eksenin konum kontrolü için mevcut pozisyonu sağlar.

Aşağıdaki tabloda, sürücüler ve enkoderler için mevcut olan bağlantı olanakları gösterilmektedir:

Sürücü bağlantısı	Eksenin ayarlanması/kumandası	Enkoder bağlantısı
Darbe dizisi (PTO) (darbe arayüzüne sahip adım motorları ve servo motorlar)	Konum kontrollü	-
Analog çıkış (AQ)	Konum kontrollü	• Hızlı sayaçtaki (HSC) enkoder • Teknoloji modülündeki (TM) enkoder • PROFINET'teki enkoder
PROFINET	Konum kontrollü	• Sürücüdeki enkoder • Hızlı sayaçtaki (HSC) enkoder • Teknoloji modülündeki (TM) enkoder • PROFINET'teki enkoder

10.2.3 Hareket kumandası yordamları

10.2.3.1 MC yordamlarına genel bakış

Hareket kumandası yordamları, bir eksenin hareketine kumanda etmek için ilgili bir teknoloji veri bloğunu ve CPU'nun bunun için atanan darbe dizisini (PTO) kullanır:

NOT

Hareket kumandası yordamları çeşitli kütüphane sürümlerine aittir.

V1.0 ila V3.0 sürümü hareket kumandasının yordamları, yordamın çıkışına aktif olarak kumanda eder. Blokta bir hata oluşursa, serbest bırakma çıkışı (ENO) KAPALI durumuna ayarlanır. Bloktaki ERROR, ErrorID ve ErrorInfo çıkışları üzerinden bir hata bildirilir. ENO çıkışı aracılığıyla yordamın durumu değerlendirilebilir ve sonraki yordamlar sırayla yürütülebilir.

V4.0 ve V5.0 sürümü hareket kumandasının yordamlarında, bloğun hata durumundan bağımsız olarak EN girişi her zaman ENO çıkışına aktarılır. Bu, sürümü V3.0 veya daha eski olan hareket kumandasıyla çalışan ve ENO durumuna bağlı bir programının doğru çalışmamasına neden olabilir. Bu durumu gidermek için, V4.0 veya daha yüksek sürüm hareket kumandası kullanıyorsanız, yordamın durumunu ENO çıkışıyla değil, DONE ve ERROR çıkışlarıyla değerlendirin.

Referanslar

STEP 7 bilgi sistemi, her hareket kumandası yordamı veya MC yordamı için bir dokümantasyon sağlar. Burada, programlamayla ilgili ve hareket kumandası yordamlarının daha eski ve güncel yorumları arasındaki farklarla ilgili yardıma ulaşabilirsiniz.
Aşağıdaki konular hakkında bilgiler içeren SIMATIC STEP 7 S7-1200 Motion Control (https://support.industry.siemens.com/cs/us/en/view/109773400) başlığına bakın:

- Darbe jeneratörünün yapılandırılması
- Kontrolsüz hareket kumandası
- Eksen yapılandırma
- İşletime alma
- Kontrollü hareket kumandası
- Eksen yapılandırma
- ServoOB'ler
- Devir kontrollü işletim
- Telegram 4 desteği
- Simülasyon ekseni
- Veri aktarımı
- Pulse teknoloji modülüyle (TM) eksen kumandası
- TO_CommandTable_PTO'nun yapılandırılması
- S7-1200'de hareket kumandasının çalışma şekli
- Hareket kumandası için kullanılan CPU çıkışları
- Hareket kumandası için kullanılan donanım ve yazılım limit şalterleri
- Referans nokta hareketi
- Sarsıntı sınırlama
- Aktif komutları denetleme

- MC yordamlarının bir "Done" çıkış parametresiyle denetlenmesi
- MC_Velocity'yi denetleme
- MC_MoveJog'u denetleme
- Hareket kumandası için kullanılan ErrorID'ler ve ErrorInfo'lar

Ayrıca bakınız

MC_Power (ekseni serbest bırak/bloke et) (Sayfa 542)
MC_Reset (hatayı onayla) (Sayfa 542)
MC_Home (eksenin referans nokta hareketini gerçekleştir) (Sayfa 543)
MC_Halt (ekseni duraklat) (Sayfa 543)
MC_MoveAbsolute (ekseni mutlak şekilde konumla) (Sayfa 544)
MC_MoveRelative (ekseni bağıl şekilde konumla) (Sayfa 544)
MC_MoveVelocity (ekseni önceden tanımlanmış hızla hareket ettir) (Sayfa 545)
MC_MoveJog (Ekseni JOG modunda hareket ettir) (Sayfa 545)
MC_CommandTable (eksen kumanda komutunu hareket dizisi olarak yürüt) (Sayfa 546)
MC_ChangeDynamic (eksenin dinamik ayarlarını değiştir) (Sayfa 547)
MC_WriteParam (bir teknoloji nesnesinin parametrelerini yaz) (Sayfa 546)
MC_ReadParam (teknoloji nesnesinin parametrelerini oku) yordamı (Sayfa 547)

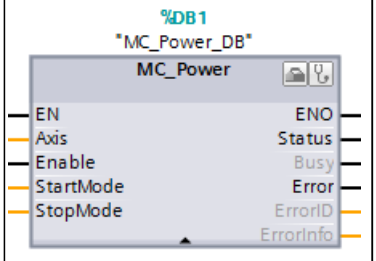
10.2.3.2 MC_Power (ekseni serbest bırak/bloke et)

MC_Power yordamı, hareket kumandası için bir ekseni etkinleştirir ve devre dışı bırakır.

NOT

Eksen bir hata nedeniyle kapatılırsa, hata giderildikten ve onaylandıktan sonra otomatik olarak yeniden etkinleştirilir. Bunun için Enable giriş parametresinin bu işlem sırasında DOĞRU değerini kaydetmiş olması gerekir.

Tablo 10-16 MC_Power yordamı

LAD/FBD	SCL	Açıklama
	**"MC_Power_DB"(Axis:=_multi_fb_in_, Enable:=_bool_in_, StartMode:=_int_in_, StopMode:=_int_in_, Status=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_Power hareket kumandası yordamı bir ekseni etkinleştirir veya devre dışı bırakır. Ekseni etkinleştirebilmeniz veya devre dışı bırakabilmeniz için aşağıdaki koşulları kontrol edin: • Teknoloji nesnesi doğru şekilde yapılandırıldı. • Serbest bırakmayı engelleyen bir hata yok. MC_Power yordamının yürütülmesi, bir hareket kumandası görevi tarafından iptal edilemez. Eksenin devre dışı bırakılması (Enable giriş parametresi = YANLIŞ) yoluyla, ilgili teknoloji nesnesi için tüm hareket kumandası görevleri iptal edilir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğin "MC_Power_DB", örnek DB'nin adıdır.

10.2.3.3 MC_Reset (hatayı onayla)

MC_Reset yordamı, tüm hareket kumandası hatalarını sıfırlar. Onaylanabilen tüm hareket kumandası hataları onaylanır.

Tablo 10-17 MC_Reset yordamı

LAD/FBD	SCL	Açıklama
	**"MC_Reset_DB"(Axis:=_multi_fb_in_, Execute:=_bool_in_, Restart:=_bool_in_, Done=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_Reset yordamı ile "Eksen durdurmalı işletim hatalarını" ve "Konfigürasyon hatalarını" onaylayabilirsiniz. Onaylanması gereken hataları, "Çözüm" başlığı altındaki "ErrorIDs ve ErrorInfos listesinde bulabilirsiniz". MC_Reset yordamını kullanmadan önce, onaylanması gereken mevcut bir konfigürasyon hatasını gidermiş olmanız gerekir (örneğin "Eksen" teknoloji nesnesindeki geçersiz bir hızlanma değerini geçerli bir değerle değiştirerek). Restart yordamı, 3.0 sürümünden itibaren eksen konfigürasyonunu RUN işletim durumunda çalışma belleğine yüklemeye olanak sağlar.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

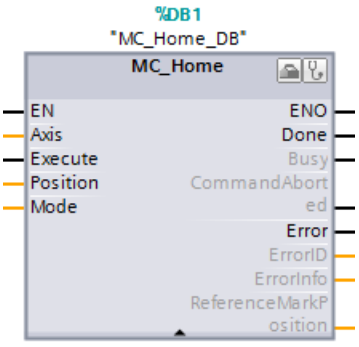
[2] SCL örneğinde "MC_Reset_DB", örnek DB'nin adıdır.

MC_Reset görevi başka hiçbir hareket kumandası görevi tarafından iptal edilemez. Yeni MC_Reset görevi, başka hiçbir aktif hareket kumandası görevini iptal etmez.

10.2.3.4 MC_Home (eksenin referans nokta hareketini gerçekleştir)

MC_Home yordamı, eksen kumandası programı ile eksenin mekanik konumlama sistemi arasındaki ilişkiyi oluşturur.

Tablo 10-18 MC_Home yordamı

LAD/FBD	SCL	Açıklama
	**"MC_Home_DB"(Axis:=_multi_fb_in_, Execute:=_bool_in_, Position:=_real_in_, Mode:=_int_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_, ReferenceMarkPosition=>_real_out_);**	MC_Home yordamıyla, eksen koordinatlarını sürücünün gerçek, fiziksel pozisyonuna uyarlayabilirsiniz. Eksenin mutlak konumlama işlemi için bir referans nokta hareketi gerekir: MC_Home yordamını kullanmak için önce eksen serbest bırakılmalıdır.

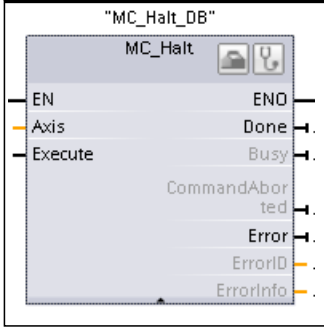
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_Home_DB", örnek DB'nin adıdır.

10.2.3.5 MC_Halt (ekseni duraklat)

MC_Halt yordamı, tüm hareket işlemlerini iptal eder ve eksen hareketinin durdurulmasına neden olur. Durdurma pozisyonu tanımlanmamıştır.

Tablo 10-19 MC_Halt yordamı

LAD/FBD	SCL	Açıklama
	**"MC_Halt_DB"(Axis:=_multi_fb_in_, Execute:=_bool_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_Halt yordamıyla, tüm hareketleri durdurabilir ve eksenin durmasını sağlayabilirsiniz. Durma pozisyonu tanımlanmamıştır. MC_Halt yordamını kullanmak için önce eksen serbest bırakılmalıdır.

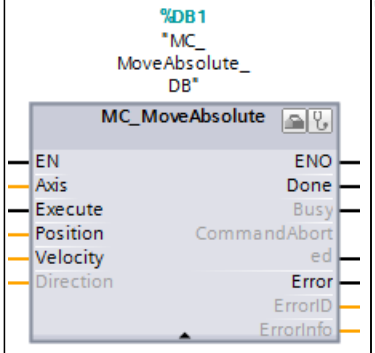
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_Halt_DB", örnek DB'nin adıdır.

10.2.3.6 MC_MoveAbsolute (ekseni mutlak şekilde konumla)

MC_MoveAbsolute yordamı, bir mutlak pozisyon yönündeki hareketi başlatır. Hedef pozisyona ulaşıldıysa görev tamamlanır.

Tablo 10-20 MC_MoveAbsolute yordamı

LAD/FBD	SCL	Açıklama
	**"MC_MoveAbsolute_DB"(Axis:=_multi_fb_in_, Execute:=_bool_in_, Position:=_real_in_, Velocity:=_real_in_, Direction:=_int_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_MoveAbsolute yordamıyla, eksenin bir mutlak pozisyon yönündeki konumlama hareketini başlatabilirsiniz. MC_MoveAbsolute yordamını kullanmak için önce eksen serbest bırakılmalı ve referans noktaya götürülmelidir.

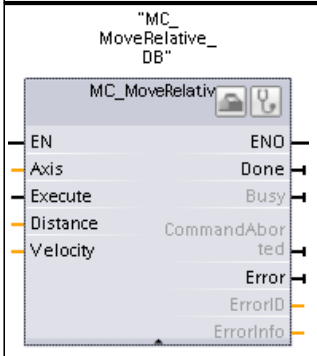
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_MoveAbsolute_DB", örnek DB'nin adıdır.

10.2.3.7 MC_MoveRelative (ekseni bağıl şekilde konumla)

MC_MoveRelative yordamı, başlangıç pozisyonuna göre bir konumlama hareketini başlatır.

Tablo 10-21 MC_MoveRelative yordamı

LAD/FBD	SCL	Açıklama
	**"MC_MoveRelative_DB"(Axis:=_multi_fb_in_, Execute:=_bool_in_, Distance:=_real_in_, Velocity:=_real_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_MoveRelative yordamıyla, başlangıç pozisyonuna göre bir konumlama hareketini başlatabilirsiniz. MC_MoveRelative yordamını kullanmak için önce eksen serbest bırakılmalıdır.

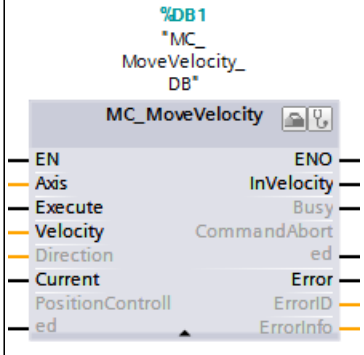
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_MoveRelative_DB", örnek DB'nin adıdır.

10.2.3.8 MC_MoveVelocity (ekseni önceden tanımlanmış hızla hareket ettir)

MC_MoveVelocity yordamı, eksenin belirtilen hızla hareket etmesini sağlar.

Tablo 10-22 MC_MoveVelocity yordamı

LAD/FBD	SCL	Açıklama
	**"MC_MoveVelocity_DB"(Axis:=_multi_fb_in_, Execute:=_bool_in_, Velocity:=_real_in_, Direction:=_int_in_, Current:=_bool_in_, PositionControlled:=_bool_in_, InVelocity=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_MoveVelocity yordamıyla eksen, sabit olarak belirtilen hızla hareket ettirebilirsiniz. MC_MoveVelocity yordamını kullanmak için önce eksen serbest bırakılmalıdır.

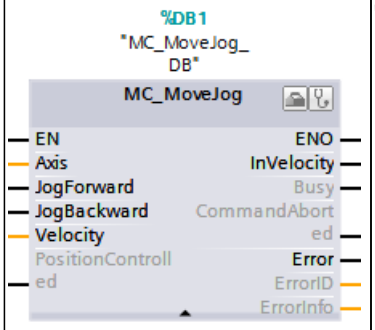
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_MoveVelocity_DB", örnek DB'nin adıdır.

10.2.3.9 MC_MoveJog (Ekseni JOG modunda hareket ettir)

MC_MoveJog yordamı, test ve yol alma amacıyla JOG modunu yürütür.

Tablo 10-23 MC_MoveJog yordamı

LAD/FBD	SCL	Açıklama
	**"MC_MoveJog_DB"(Axis:=_multi_fb_in_, JogForward:=_bool_in_, JogBackward:=_bool_in_, Velocity:=_real_in_, PositionControlled:=_bool_in_, InVelocity=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	MC_MoveJog yordamıyla ekseni, sabit olarak belirtilen hızla JOG modunda hareket ettirebilirsiniz. Bu yordam genelde test ve işletime alma amacıyla kullanılır. MC_MoveJog yordamını kullanmak için önce eksen serbest bırakılmalıdır.

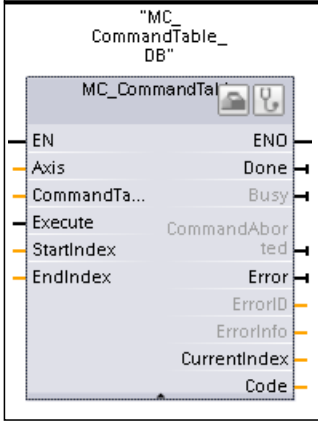
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_MoveJog_DB", örnek DB'nin adıdır.

10.2.3.10 MC_CommandTable (eksen kumanda komutunu hareket dizisi olarak yürüt)

MC_CommandTable yordamı, eksen kumanda komutlarını hareket dizisi olarak yürütür.

Tablo 10-24 MC_CommandTable yordamı

LAD/FBD	SCL	Açıklama
	**"MC_CommandTable_DB"(Axis:=_multi_fb_in_, CommandTable:=_multi_fb_in_, Execute:=_bool_in_, StartStep:=_uint_in_, EndStep:=_uint_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_, CurrentStep=>_uint_out_, StepCode=>_word_out_);**	Bir motor kumanda ekseni için kullanılan ve bir hareket dizisi olarak bağlanabilen bir dizi münferit hareketi uygular. Münferit hareketler, darbe dizisi için bir teknoloji nesnesinin bir komut tablosunda yapılandırılır (TO_CommandTable_PTO).

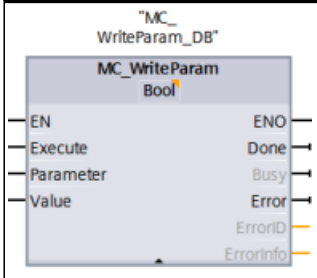
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_CommandTable_DB", örnek DB'nin adıdır.

10.2.3.11 MC_WriteParam (bir teknoloji nesnesinin parametrelerini yaz)

MC_WriteParam yordamıyla, kullanıcı programında eksenin işlevini değiştirmek için belirli parametreleri yazabilirsiniz.

Tablo 10-25 MC_WriteParam yordamı

LAD/FBD	SCL	Açıklama
	**"MC_WriteParam_DB"(Parameter:=_variant_in_, Value:=_variant_in_, Execute:=_bool_in_, Done:=_bool_out_, Error:=_real_out_, ErrorID:=_word_out_, ErrorInfo:=_word_out_);**	MC_WriteParam yordamıyla, açık parametrelere yazabilirsiniz (örn. hızlanma ve kullanıcı DB'sindeki değerler).

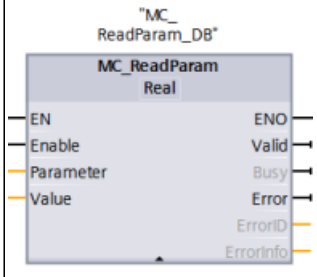
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_WriteParam_DB", örnek DB'nin adıdır.

10.2.3.12 MC_ReadParam (teknoloji nesnesinin parametrelerini oku) yordamı

MC_ReadParam yordamıyla, "Eksen" girişinde tanımlanan eksenin güncel pozisyonunu, hızını, vb. belirten seçilen bir sayıdaki parametreyi okuyabilirsiniz.

Tablo 10-26 MC_ReadParam yordamı

LAD/FBD	SCL	Açıklama
	**"MC_ReadParam_DB"(Enable:=_bool_in_, Parameter:=_variant_in_, Value:=_variant_in_out_, Valid:=_bool_out_, Busy:=_bool_out_, Error:=_real_out_, ErrorID:=_word_out_, ErrorInfo:=_word_out_);**	MC_ReadParam yordamıyla, münferit durum değerlerini çevrimdeki kontrol noktasından bağımsız olarak okuyabilirsiniz.

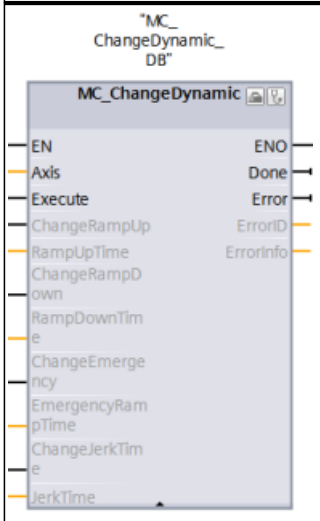
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_ReadParam_DB", örnek DB'nin adıdır.

10.2.3.13 MC_ChangeDynamic (eksenin dinamik ayarlarını değiştir)

MC_ChangeDynamic yordamıyla, bir hareket kumandası ekseninin dinamik ayarlarını değiştirebilirsiniz.

Tablo 10-27 MC_ChangeDynamic yordamı

LAD/FBD	SCL	Açıklama
	**"MC_ChangeDynamic_DB"(Axis:=_param_fb_in_, Execute:=_bool_in_, ChangeRampUp:=_bool_in_, RampUpTime:=_real_in_, ChangeRampDown:=_bool_in_, RampDownTime:=_real_in_, ChangeEmergency:=_bool_in_, EmergencyRampTime:=_real_in_, ChangeJerkTime:=_bool_in_, JerkTime:=_real_in_, Done=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);**	Bir hareket kumandası ekseninin dinamik ayarlarını değiştirir: • İlk açılış süresinin (hızlanma) değerini değiştirme • Geri dönüş süresinin (gecikme) değerini değiştirme • Acil Kapatma geri dönüş süresinin (ACİL KAPAT gecikmesi) değerini değiştirme • Düzleştirme süresinin (sarsıntı) değerini değiştirme

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[2] SCL örneğinde "MC_ChangeDynamic_DB", örnek DB'nin adıdır.

10.3 PID ayarlaması

10.3.1 PID fonksiyonları

STEP 7 ve S7-1200 CPU'su bir PID ayarı sağlar. PID kontrolü ile prosesleri aşağıdaki şekilde kontrol edebilirsiniz:

- Sürekli giriş ve çıkış değişkenlerine sahip teknik prosesler.
- Açma ve kapatma işlemleri için dijital sinyallere ihtiyaç duyan valflar gibi motorla çalıştırılan cihazların ayarlanması.
- Sıcaklık ayarıyla ilgili özel gereklilikler için üniversal PID ayarı.

Referanslar

STEP 7 bilgi sistemi, her PID yordamı için bir dokümantasyon sağlar. Burada, programlamayla ve PID yordamlarının daha eski ve güncel sürümleri arasındaki farklarla ilgili yardım bilgilerini bulabilirsiniz.
S7-1200'ün PID kontrolörüyle ilgili daha fazla bilgiyi, SIMATIC S7-1200, S7-1500 PID ayarı fonksiyon kitabında (https://support.industry.siemens.com/cs/de/en/view/108210036)) bulabilirsiniz. Bu işletme kitabında, şu konularla ilgili ayrıntılı bilgiler bulabilirsiniz:

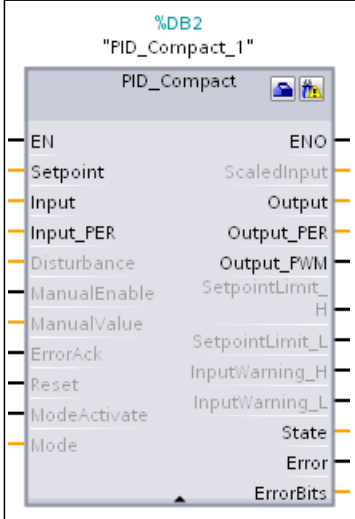
- Ayarlamayla ilgili prensipler
- Bir yazılım kontrolörünün yapılandırılması
- PID_Compact ile çalışma:
  - Proses limitleri
  - ErrorBit parametresi
  - Uyarı parametresi
- PID_3Step ile çalışma:
  - ErrorBit parametresi
  - Uyarı parametresi
- PID_Temp ile çalışma:
  - ErrorBit parametresi
  - Uyarı parametresi
- PID_Compact, PID_3Step ve PID_Temp kontrolörlerinin yapılandırılması
- PID_Compact, PID_3Step ve PID_Temp kontrolörlerinin işletime alınması

10.3.2 PID yordamları

10.3.2.1 PID_Compact yordamı

PID_Compact yordamı, otomatik ve manuel işletim için entegre kendi kendini ayarlama fonksiyonuna sahip üniversal bir PID kontrolör sunar.

Tablo 10-28 PID_Compact yordamı

LAD/FBD	SCL	Açıklama
	**"PID_Compact_1"(** **Setpoint:=_real_in_,** **Input:=_real_in_,** **Input_PER:=_word_in_,** **Disturbance:=_real_in_,** **ManualEnable:=_bool_in_,** **ManualValue:=_real_in_,** **ErrorAck:=_bool_in_,** **Reset:=_bool_in_,** **ModeActivate:=_bool_in_,** **Mode:=_int_in_,** **ScaledInput=>_real_out_,** **Output=>_real_out_,** **Output_PER=>_word_out_,** **Output_PWM=>_bool_out_,** **SetpointLimit_H=>_bool_out_,** **SetpointLimit_L=>_bool_out_,** **InputWarning_H=>_bool_out_,** **InputWarning_L=>_bool_out_,** **State=>_int_out_,** **Error=>_bool_out_,** **ErrorBits=>_dword_out_);**	PID_Compact, otomatik ve manuel işletim için kendi kendini ayarlama fonksiyonuna sahip bir PID kontrolör sağlar. PID_Compact, Anti-Windup ve P ve D oranını ağırlıklandırma özelliğine sahip bir PIDT1 kontrolörüdür.

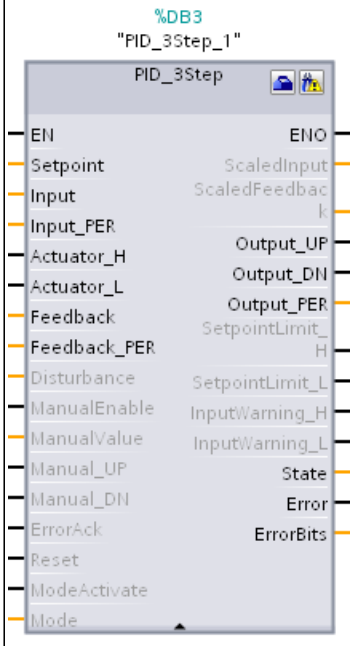
[1] STEP 7, yordamı eklediğiniz takdirde teknoloji nesnesini ve örnek DB'yi otomatik olarak oluşturur. Örnek DB, teknoloji nesnesinin parametrelerini içerir.

[2] SCL örneğinde "PID_Compact_1", örnek DB'nin adıdır.

10.3.2.2 PID_3Step yordamı

PID_3Step yordamı, motorla çalıştırılan valflar ve aktüatörler için optimize edilmiş olan, kendi kendini ayarlama fonksiyonlarına sahip bir PID kontrolör yapılandırır.

Tablo 10-29 PID_3Step yordamı

LAD/FBD	SCL	Açıklama
	**"PID_3Step_1"(** **SetpoInt:=_real_in_,** **Input:=_real_in_,** **ManualValue:=_real_in_,** **Feedback:=_real_in_,** **InputPer:=_word_in_,** **FeedbackPer:=_word_in_,** **Disturbance:=_real_in_,** **ManualEnable:=_bool_in_,** **ManualUP:=_bool_in_,** **ManualDN:=_bool_in_,** **ActuatorH:=_bool_in_,** **ActuatorL:=_bool_in_,** **ErrorAck:=_bool_in_,** **Reset:=_bool_in_,** **ModeActivate:=_bool_in_,** **Mode:=_int_in_,** **ScaledInput=>_real_out_,** **ScaledFeedback=>_real_out_,** **ErrorBits=>_dword_out_,** **OutputPer=>_word_out_,** **State=>_int_out_,** **OutputUP=>_bool_out_,** **OutputDN=>_bool_out_,** **SetpoIntLimitH=>_bool_out_,** **SetpoIntLimitL=>_bool_out_,** **InputWarningH=>_bool_out_,** **InputWarningL=>_bool_out_,** **Error=>_bool_out_,** **ErrorBits=>_dword_out_);**	PID_3Step, motorla çalıştırılan valflar ve aktüatörler için optimize edilmiş olan, kendi kendini ayarlama fonksiyonlarına sahip bir PID kontrolör yapılandırır. İki Boole çıkışı sunar. PID_3Step, Anti-Windup ve P ve D oranını ağırlıklandırma özelliğine sahip bir PID T1 kontrolördür.

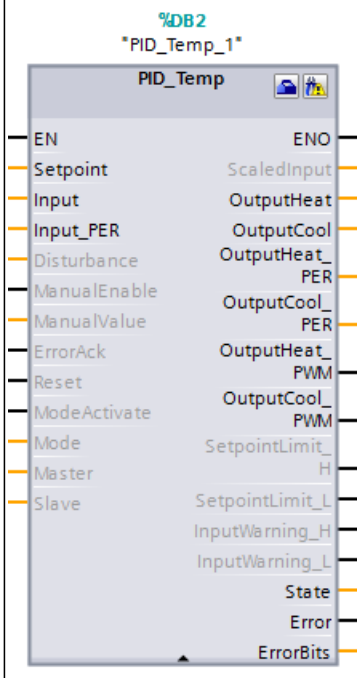
[1] STEP 7, yordamı eklediğiniz takdirde teknoloji nesnesini ve örnek DB'yi otomatik olarak oluşturur. Örnek DB, teknoloji nesnesinin parametrelerini içerir.

[2] SCL örneğinde "PID_3Step_1", örnek DB'nin adıdır.

10.3.2.3 PID_Temp yordamı

PID_Temp yordamı, sıcaklık ayarı için üniversal bir PID kontrolör sağlar.

Tablo 10-30 PID_Temp yordamı

LAD/FBD	SCL	Açıklama
	**"PID_Temp_1"(** **Setpoint:=_real_in_,** **Input:=_real_in_,** **Input_PER:=_int_in_,** **Disturbance:=_real_in_,** **ManualEnable:=_bool_in_,** **ManualValue:=_real_in_,** **ErrorAck:=_bool_in_,** **Reset:=_bool_in_,** **ModeActivate:=_bool_in_,** **Mode:=_int_in_,** **Master:=_dword_in** **Save:=_dword_in** **ScaledInput=>_real_out_,** **OutputHeat=>_real_out_,** **OutputCool=>_real_out_,** **OutputHeat_PER=>_int_out_,** **OutputCool_PER=>_int_out_,** **OutputHeat_PWM=>_bool_out_,** **OutputCool_PWM=>_bool_out_,** **SetpointLimit_H=>_bool_out_,** **SetpointLimit_L=>_bool_out_,** **InputWarning_H=>_bool_out_,** **InputWarning_L=>_bool_out_,** **State=>_int_out_,** **Error=>_bool_out_,** **ErrorBits=>_dword_out_);**	PID_Temp aşağıdaki olanakları sunar: • Çeşitli aktüatörlerle proseste ısıtma ve soğutma • Sıcaklık ayarı için entegre kendi kendini ayarlama fonksiyonu • Prosesteki aynı aktüatör tarafından ayarlanan birden fazla sıcaklığın kaskat bağlanması

[1] STEP 7, yordamı eklediğiniz takdirde teknoloji nesnesini ve örnek DB'yi otomatik olarak oluşturur. Örnek DB, teknoloji nesnesinin parametrelerini içerir.

[2] SCL örneğinde "PID_Temp_1", örnek DB'nin adıdır.

11

İletişim

11.1 Genel bakış

S7-1200, CPU'lar ile programlama cihazları, HMI cihazları ve diğer CPU'lar arasında birçok iletişim türü sunar.

PROFINET

PROFINET, kullanıcı programı aracılığıyla diğer iletişim partnerleri ile Ethernet üzerinden veri alışverişi için kullanılır:

- S7-1200'de PROFINET, 256 alt modülle birlikte 16 IO cihazını destekler ve PROFIBUS, 3 bağımsız PROFIBUS DP-Master'a izin verir ve DP-Master başına maksimum 512 modülle birlikte DP-Master başına 32 Slave'i destekler.
- S7 iletişimi
- User Datagram Protocol (UDP)
- ISO on TCP (RFC 1006)
- Transport Control Protocol (TCP)

PROFINET IO kontrolörü

CPU, IO kumanda ünitesi olarak PROFINET IO üzerinden yerel PN ağında maksimum 16 PN cihazıyla veya bir PN/PN kuplörü (grup) üzerinden iletişim kurar. Bu konuyla ilgili ayrıntılı bilgileri PROFIBUS and PROFINET International, PI (www.us.profinet.com) başlığında bulabilirsiniz.

PROFIBUS

PROFIBUS, kullanıcı programı aracılığıyla diğer iletişim partnerleri ile PROFIBUS ağı üzerinden veri alışverişi için kullanılır:

- CPU, CM 1242-5'te PROFIBUS DP-Slave olarak görev görür.
- CPU, CM 1243-5'te 1. sınıf PROFIBUS DP-Master olarak görev görür.
- PROFIBUS DP-Slave'ler, PROFIBUS DP-Master ve AS-i (sol taraftaki 3 iletişim modülü) ve PROFINET, karşılıklı olarak birbirlerini kısıtlamayan, birbirlerinden ayrı iletişim ağlarıdır.

ASi

S7-1200 CM 1243-2 AS-i Master, bir AS-i ağını bir S7-1200 CPU'suna bağlama olanağı sağlar.

S7 üzerinden CPU-CPU iletişimi

Bir partner istasyonla bir iletişim bağlantısı kurabilir ve GET ve PUT yordamlarının yardımıyla S7 CPU'larıyla iletişim sağlayabilirsiniz.

TeleService iletişimi

GPRS üzerinden TeleService durumunda, STEP 7'nin kurulu olduğu bir mühendislik istasyonu, GSM ağı ve internet üzerinden bir CP 1242-7 ile birlikte bir SIMATIC S7-1200 istasyonuyla iletişim sağlar. Bağlantı, aracı olarak görev gören ve internete bağlı bir Telecontrol sunucusu üzerinden gerçekleşir.

IO-Link

S7-1200 SM 1278 4xIO-Link-Master, IO-Link cihazlarının bir S7-1200 CPU bağlantısına olanak sağlar.

Fiziksel ağ saldırıları nedeniyle oluşabilecek güvenlik risklerine engel olun

⚠ İKAZ

Fiziksel ağ saldırıları nedeniyle oluşabilecek güvenlik risklerine engel olun

Bir saldırgan, ağlarınıza fiziksel olarak erişebilirse, muhtemelen verileri okuyabilir ve yazabilir.

Örn. PROFIBUS, PROFINET, AS-i veya başka bir G/Ç veriyolu, GET/PUT, T-Block ve iletişim modülü (CM) gibi diğer tüm iletişim türleri üzerinden G/Ç alışverişi hiçbir güvenlik fonksiyonuna sahip değildir. Bu tür iletişim türlerini, fiziksel erişimi sınırlama yoluyla korumanız gerekir. Bir saldırgan ağlarınıza bu iletişim türleri üzerinden fiziksel olarak erişebilirse, muhtemelen verileri okuyabilir ve yazabilir.

Bu iletişim türlerini korumazsanız, bu durum ölüme veya ağır yaralanmalara neden olabilir.

Güvenlikle ilgili bilgileri ve önerileri, Siemens'in "Service & Support" web sitesindeki "Operational Guidelines for Industrial Security" (https://new.siemens.com/global/en/products/automation/topic-areas/industrial-security.html) bölümümüzde bulabilirsiniz.

11.2 Ethernet iletişimi için kullanılan iletişim protokolleri ve portlar

Burada, PN/IE arayüzleri üzerinden iletişim için desteklenen protokoller ve portlara ait bir genel bakış sunulmaktadır. Belirtilen portlarda, S7-1200 PLC'si tarafından kullanılan standart port numaraları söz konusudur. Sayısız iletişim protokolü ve uygulama, başka port numaralarını kullanmanıza olanak sağlar. Aşağıdaki tablolarda, S7-1200 PLC'si tarafından kullanılan çeşitli katmanlar, protokoller ve portlar gösterilmektedir.

Tablo 11-1 S7-1200'ün taşıma katmanı için kullanılan portlar ve protokoller

Port (portlar)	Yön	Protokol	Uygulama	Açıklama
25	Giden	TCP	SMTP	SMTP, e-posta göndermek için kullanılır.
80	Gelen	TCP	HTTP	HTTP, CPU dahilindeki Web sunucusuyla iletişim için kullanılır.
102[1]	Gelen/giden	TCP	ISO on TCP	ISO on TCP (RFC 1006 uyarınca) Remote-CPU ile mesaj odaklı veri alışverişi, ES, HMI ile S7 iletişimi için kullanılır.
123	Giden	UDP	NTP	NTP, CPU sistem saatinin bir NTP sunucusunun saatiyle senkronizasyonu için kullanılır.
161	Gelen	UDP	SNMP	SNMP, SNMP-Manager tarafından ağ yönetim verilerinin (SNMP tarafından yönetilen nesneler üzerinden) okunması ve ayarlanması için kullanılır.
443	Gelen	TCP	HTTPS	HTTP, TLS üzerinden CPU dahilindeki Web sunucusu ile iletişim için kullanılır.
465, 587	Giden	TCP	SMTPS	SMTP, E-postaların güvenli bağlantılar üzerinden gönderilmesi için kullanılır.
502	Gelen/giden	TCP	Modbus	Modbus/TCP, kullanıcı programındaki MB_CLIENT/MB_SERVER yordamları tarafından kullanılır.
4840[2]	Gelen	TCP	OPC UA	Şirket düzeyinden alansal düzeye kadar iletişim standardı.
34964[1]	Gelen/giden	UDP	PROFINET Context Manager	PROFINET Context Manager, bir uygulama ilişkisi oluşturmak için bir uç nokta ataması olanağı sunar (PROFINET AR).

[1] Bu portlar, teslimat durumunda, yapılandırılmış IP adresleriyle birlikte hemen kullanılabilen konfigürasyonda açık ve erişilebilir durumdadır. Diğer uygulamalar, S7-1200 kullanıcı programının bir parçası olarak etkinleştirilmeli/yapılandırılmalıdır.

[2] Port 4840, standart porttur, ancak bu port da yapılandırılabilir.

Tablo 11-2 Açık kullanıcı iletişimi (Open User Communication, OUC) ve diğer protokoller tarafından kullanılabilen port aralıkları. S7-1200 kullanıcı programının bir parçası olarak kullanıcı tarafından kesin iletişim parametreleri tanımlanır.

Port aralığı	Yön	Protokol	Uygulama	Açıklama
1 ila 999	Değişir	TCP/UDP	OUC	Port aralığı, halihazırda kullanılmakta olan portlar istisna olmak kaydıyla kısıtlı bir kapsamda kullanılabilir.
2000-5000	Değişir	TCP/UDP	OUC	OUC için önerilen port aralığı
5001-49151	Değişir	TCP/UDP	OUC	Port aralığı, halihazırda kullanılmakta olan portlar istisna olmak kaydıyla kısıtlı bir kapsamda kullanılabilir.
49152-65535	Giden	TCP/UDP	Değişir	Uygulamanın yerel port numarasını belirlememesi durumunda aktif bağlantı uç noktası için kullanılan dinamik port aralığı.

Tablo 11-3 OSI modelinin güvenlik ve ağ katmanında (katman 2 ve 3) S7-1200 tarafından kullanılan protokoller

Protokol	Yön	Ether tipi	Açıklama
PROFINET DCP	Gelen/giden	0x8892	DCP, PROFINET cihazlarını algılamak ve temel ayarları sağlamak için PROFINET tarafından kullanılır.
LLDP	Giden	0x88CC	LLDP, komşu PROFINET cihazlarının arsındaki ilişkileri algılamak ve yönetmek için PROFINET tarafından kullanılır. LLDP, özel Multicast-MAX adresini kullanır: 01-80-C2-00-00-0E.
PROFINET IO	Gelen/giden	0x8892	PROFINET IO telegramları, IO verilerini periyodik olarak PROFINET IO kontrolörü ile IO cihazları arasında Ethernet üzerinden aktarmak için kullanılır.
ICMP	Gelen	0x800	Internet Control Message Protocol, teşhis ve kumanda amacıyla kullanılır.

11.3 Asenkron iletişim bağlantıları

İletişim hizmetlerine genel bakış

CPU, aşağıdaki iletişim hizmetlerini destekler:

İletişim hizmeti	Fonksiyonlar	PROFIBUS DP'de		Ethernet'te
		CM 1243-5 DP-Master modül	CM 1242-5 DP-Slave modül	
PG iletişimi	İşletime alma, test, teşhis	Evet	Hayır	Evet
HMI iletişimi	Kontrol ve gözetleme	Evet	Hayır	Evet
S7 iletişimi	Yapılandırılmış bağlantılar üzerinden veri alışverişi	Evet	Hayır	Evet
PG fonksiyonlarının yönlendirilmesi	Örneğin ağ sınırlarının dışında test teşhis	Hayır	Hayır	Hayır
PROFIBUS DP	Master ile Slave arasında veri alışverişi	Evet	Evet	Hayır
PROFINET IO	G/Ç kumanda üniteleri ile G/Ç cihazları arasında veri alışverişi	Hayır	Hayır	Evet
Web sunucusu	Teşhis	Hayır	Hayır	Evet
SNMP [1] (Simple Network Management Protocol)	Ağ teşhisi ve parametrelendirme için standart protokol	Hayır	Hayır	Evet
S7-Routing	Yönlendirme tablolarının yardımıyla iletişim partnerleri, cihazlar farklı S7 alt ağlarında olsa bile her cihazla iletişim kurabilir.	Hayır	Hayır	Evet

[1] CPU, TRAP'ler olmadan SNMP V1'i destekler.

[2] OPC UA, sadece CPU'nun entegre Ethernet portu üzerinden desteklenir.

İletişim hizmeti	Fonksiyonlar	PROFIBUS DP'de		Ethernet'te
		CM 1243-5 DP-Master modül	CM 1242-5 DP-Slave modül	
TCP/IP üzerinden açık iletişim	TCP/IP protokolü (yüklenebilir FB'lerle) Industrial Ethernet üzerinden veri alışverişi	Hayır	Hayır	Evet
ISO on TCP üzerinden açık iletişim	ISO on TCP protokolüyle (yüklenebilir FB'lerle) Industrial Ethernet üzerinden veri alışverişi	Hayır	Hayır	Evet
UDP üzerinden açık iletişim	UDP protokolü (yüklenebilir FB'lerle) Industrial Ethernet üzerinden veri alışverişi	Hayır	Hayır	Evet
OPC UA sunucusu	OPC UA istemcileriyle Industrial Ethernet üzerinden veri alışverişi	Hayır	Hayır	Evet[2]

1 CPU, TRAP'ler olmadan SNMP V1'i destekler.

2 OPC UA, sadece CPU'nun entegre Ethernet portu üzerinden desteklenir.

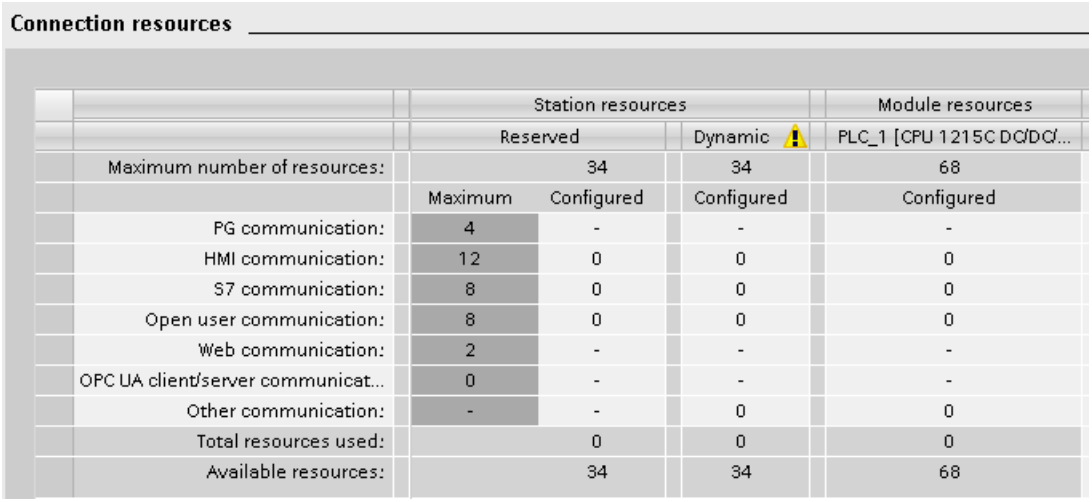
Mevcut bağlantılar

CPU, PROFINET ve PROFIBUS için maksimum aşağıdaki sayıda eş zamanlı, asenkron iletişim bağlantısını destekler. Her kategoriye atanan maksimum bağlantı kaynağı sayısı belirlenmiştir. Değerleri değiştiremezsiniz. Ancak serbestçe kullanılabilen bağlantılar, her kategorideki bağlantı sayısını uygulamanızın ihtiyacına göre arttırmak için kullanılabilir.

Bazı bağlantı tipleri sabit sayıda rezerve edilmiş kaynağa ("garanti edilmiş" olarak da adlandırılır) sahiptir. Bu, CPU'nun, bağlantı tipi için garanti olarak maksimum bu sayıda rezerve edilmiş kaynağı desteklediği anlamına gelir. Bu sayede, örneğin CPU ile eş zamanlı olarak en az 12 HMI bağlantısı kurulabilir. Bir bağlantı tipi için, rezerve edilmiş kaynakların sayısının ötesinde başka bağlantılar kurulabilir, ancak bu bağlantı kaynakları "dinamik" kaynaklar havuzundan gelmek zorundadır.

Dinamik kaynaklar ("boş" olarak da adlandırılır) her bağlantı tipi için kullanılabilen kaynaklardan oluşan bir gruptur. Bu kaynaklar, rezerve edilmiş kaynaklara sahip olmayan bağlantılar tarafından (OPC UA gibi) veya rezerve edilmiş tüm kaynaklarını kullanmış olan bağlantılar tarafından kullanılır.

S7-1200 CPU'ları 34 dinamik kaynağa sahiptir.



Connection resources

	Station resources			Module resources
	Reserved		Dynamic	PLC_1 [CPU 1215C DC/DC/...
Maximum number of resources:		34	34	68
	Maximum	Configured	Configured	Configured
PG communication:	4	-	-	-
HMI communication:	12	0	0	0
S7 communication:	8	0	0	0
Open user communication:	8	0	0	0
Web communication:	2	-	-	-
OPC UA client/server communicat...	0	-	-	-
Other communication:	-	-	0	0
Total resources used:		0	0	0
Available resources:		34	34	68

NOT

CM/CP modüllerini tamamlarsanız, S7-1200 iletişim bağlantılarının toplam sayısı artmaz.

NOT

OPC UA bağlantıları

OPC UA bağlantıları, dinamik kaynaklardan gelen bağlantıları tüketir. Uygulamanız için yeterli miktarda bağlantıya sahip olduğunuzdan emin olun.

Cihaz başına, atanmış bağlantı kaynakları temel alınarak aşağıdaki sayıda bağlantı kullanıma sunulur:

Tip	Rezerve	Maksimum[1]
Programlama cihazı (PG) üzerinden iletişim	4	4
İnsan makina arayüzü (HMI) üzerinden iletişim	12	18
S7 iletişimi	8	14
Open User Communication	8	14
Web iletişimi	2	30
OPC UA istemci/sunucu iletişimi	0	10
Diğer iletişim	0	-

[1] Dinamik bağlantılar birlikte kullanıldığı için, tüm bağlantıları eş zamanlı olarak kullanmak mümkün değildir.

Örnek: CPU, dört kullanılabilir PG bağlantı kaynağına sahiptir. PG, mevcut durumda kullanılan PG fonksiyonlarına bağlı olarak muhtemelen, gerçekte, mevcut bağlantı kaynaklarının bir, iki, üç veya dördünü kullanır. Her zaman bir PG kullanabilirsiniz.

Diğer bir örnek, aşağıdaki şekilde gösterildiği gibi HMI cihazlarının sayısıdır. HMI cihazları 12 adet kullanılabilir bağlantı kaynağına sahiptir. HMI cihazınızın tipine veya modeline ve kullanılan HMI fonksiyonlarına bağlı olarak, her HMI cihazı gerçekte kendi mevcut bağlantı kaynaklarının bir, iki veya üçünü kullanabilir. Kullanılabilen mevcut bağlantı kaynaklarının sayısına bağlı olarak muhtemelen dörtten fazla HMI cihazı eş zamanlı olarak kullanılabilir.

Ancak her zaman en az dört HMI cihazının kullanılması garantidir. Bir HMI cihazı, mevcut bağlantı kaynaklarını (toplam 3 adetten 1'er tane) şu fonksiyonlar için kullanabilir:

- Okuma
- Yazma
- Alarmlar ve teşhisler

Bu sadece bir örnektir. Gerçek bağlantı sayısı, HMI cihazının tipine ve sürümüne bağlı olarak farklı olabilir.

Örnek	HMI 1	HMI 2	HMI 3	HMI 4	HMI 5	Kullanılabilir bağlantı kaynaklarının toplam sayısı
Kullanılan bağlantı kaynakları	2	2	2	3	3	12

NOT

Web sunucusu bağlantıları: CPU, birçok web tarayıcı için bağlantılar sunar. CPU tarafından eş zamanlı olarak desteklenebilen tarayıcı sayısı, bir web tarayıcının kaç bağlantı talep ettiğine/kullandığına ve CPU'da kaç dinamik bağlantının mevcut olduğuna bağlıdır.

NOT

Open User Communication, S7 bağlantısı, HMI cihazı, programlama cihazı, OPC UA ve Web sunucusu üzerinden gerçekleştirilen iletişim bağlantıları, kullanılan fonksiyonlar temelinde birçok bağlantı kaynağını kullanabilir.

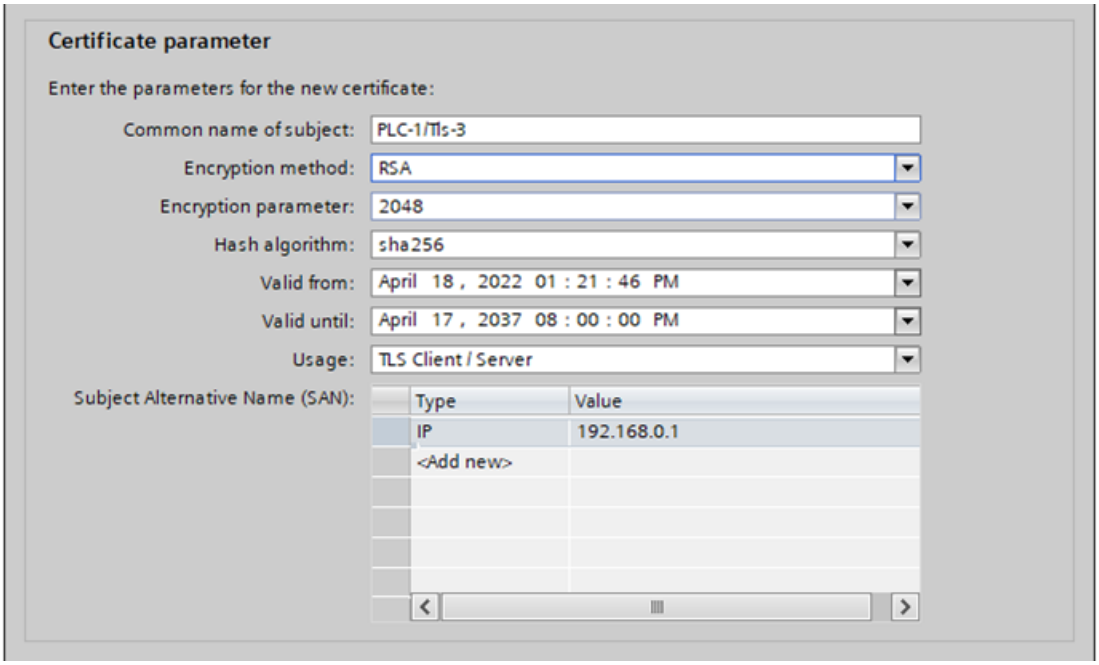
11.4 Desteklenen sertifikalar

Security Communication hizmetleri, belirli güvenlik parametrelerine sahip sertifikalara ihtiyaç duyar.

Bu sertifikaları, TIA Portal'daki aşağıdaki bölümlerde, CPU'nun özellik penceresinin "Genel" sekmesinde oluşturabilir veya seçebilirsiniz:

- "Web sunucusu > Security": Donanım tarafından oluşturulabilen veya yazılım tarafından yüklenebilen Web sunucusu sertifikalarının oluşturulması ve atanması için.
- "Koruma ve dijital güvenlik > Bağlantı mekanizmaları": PLC iletişimi veya PG/PC ve HMI iletişimi için kullanılan sertifikaların oluşturulması veya atanması için.
- "Koruma ve dijital güvenlik > Sertifika yöneticisi": Tüm sertifika tiplerinin oluşturulması veya atanması için. Sertifikaların oluşturulması için standart ayar, Secure Open User Communication (Secure OUC) için kullanılan TLS sertifikalarıdır.
- "OPC UA > Sunucu > Security": OPC UA sunucu sertifikalarının oluşturulması ve atanması için.

"Sertifika yöneticisindeki" "Sertifika parametreleri" penceresiyle ilgili örnek:



Sertifikaların oluşturulması ve atanmasıyla ilgili daha fazla bilgiyi, STEP 7 bilgi sisteminde bulabilirsiniz.

Dijital güvenlik özelliğine sahip tüm iletişim hizmetleri için önerilen sertifika parametreleri:

Sertifika parametreleri		İletişim hizmetleri			
Şifreleme yöntemi	**Şifreleme parametreleri**	**Web sunucusu** (Sayfa 806)	**Secure PG/PC ve HMI iletişimi** (Sayfa 158)	**Secure OUC** (Sayfa 581-582)	**OPC UA** (Sayfa 769)
EC	prime256v1	Destekleniyor	Destekleniyor	Destekleniyor	Desteklenmiyor
EC	secp384r1	Destekleniyor	Destekleniyor	Destekleniyor	Desteklenmiyor
RSA	2048	Destekleniyor	Destekleniyor	Destekleniyor	Destekleniyor

RSA 4096 sertifika parametrelerini, iletişimin tarama çevrimi için kullanımın en azından % 60'ı atanacak şekilde açıkça daha yüksek bir minimum tarama çevrimiyle (Sayfa 84) kullanabilirsiniz.

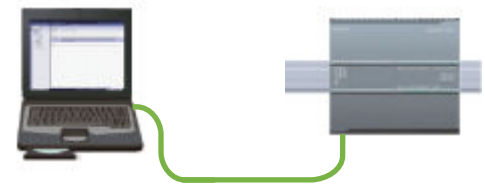
NOT

"Sertifika yöneticisinde", desteklenmeyen sertifika parametrelerini de seçebilmenize rağmen, sadece yukarıdaki tabloda önerilen sertifika parametreleri kullanılmalıdır.

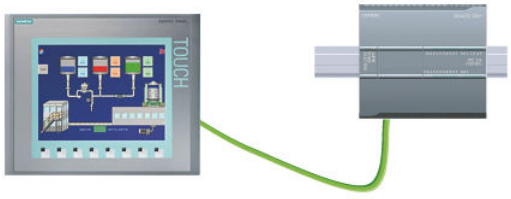
Listede bulunmayan sertifika parametrelerinin kullanımı, iletişim sorunlarına veya iletişim hizmetinin devre dışı kalmasına yol açabilir.

11.5 PROFINET

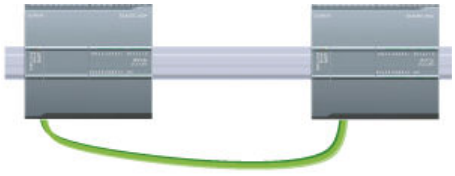
CPU; başka CPU'lar, programlama cihazları, HMI cihazları ve Siemens harici cihazlarla klasik TCP iletişim protokolleri üzerinden iletişim kurabilir.



CPU'ya bağlı programlama cihazı



CPU'ya bağlı HMI cihazı



Başka bir CPU'ya bağlı CPU

Ethernet Switch

CPU 1211C, 1212C ve 1214C bir Ethernet portuna sahiptir ve entegre bir Ethernet Switch'e sahip değildir. Bir programlama cihazı veya HMI cihazı ile bir CPU arasındaki doğrudan bir iletişim, Ethernet Switch kullanmayı gerektirmez. Ancak ikiden fazla CPU veya HMI cihazı içeren bir ağ bir Ethernet Switch'e ihtiyaç duyar.



① CPU 1215C

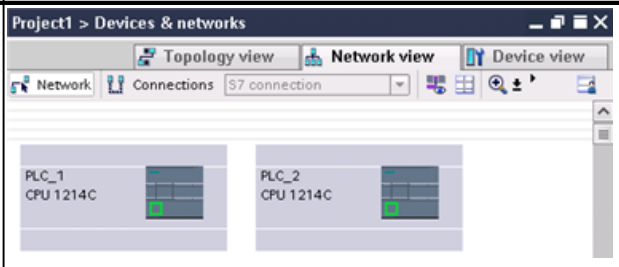
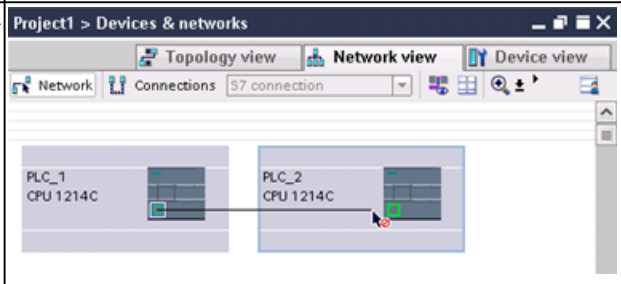
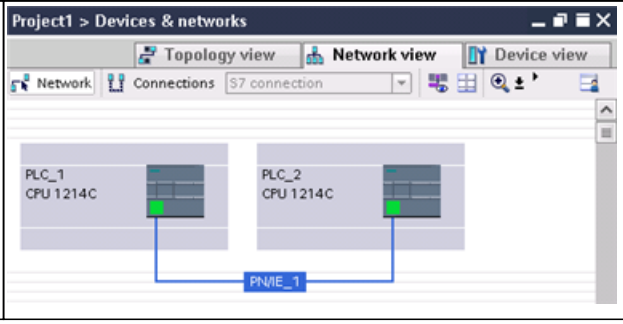
② CSM1277 Ethernet Switch

CPU 1215C ve 1217C, 2 portlu entegre bir Ethernet Switch'e sahiptir. Bir CPU 1215C ve iki ayrı S7-1200 CPU ile bir ağ oluşturabilirsiniz. Birden çok CPU'yu ve HMI cihazını bağlamak için, modül taşıyıcısına takılmış durumdaki 4 portlu bir CSM1277 Ethernet Switch de kullanabilirsiniz.

11.5.1 Ağ bağlantısı oluşturma

Cihaz konfigürasyonunun ağ görünümünde, projenizdeki cihazların arasında ağ bağlantıları kurabilirsiniz. Ağ bağlantısı kurulduktan sonra, denetçi penceresinin "Özellikler" sekmesinde ağ parametrelerini yapılandırabilirsiniz.

Tablo 11-4 Ağ bağlantısı oluşturma

İşlem talimatı	Sonuç
Bağlanması gereken cihazların görüntülenmesi için "Ağ görünümü"nü seçin.	
Bir cihazın arayüzünü seçin ve bağlantıyı ikinci cihazın arayüzüne doğru sürükleyin.	
Ağ bağlantısının oluşturulması için fare tuşunu bırakın.	

11.5.2 Yerel ile partner CPU arasındaki bağlantı yolunun yapılandırılması

Yerel ve uzaktaki partner CPU arasındaki bir bağlantı, iletişim hizmetlerinin oluşturulması için iki iletişim partnerinin mantıksal bir atamasını tanımlar. Bir bağlantı şu elemanları tanımlar:

- Katılan iletişim partnerleri (biri aktif, biri pasif)
- Bağlantının türü (örn. PLC, HMI veya cihaz ile bağlantı)
- Bağlantı yolu

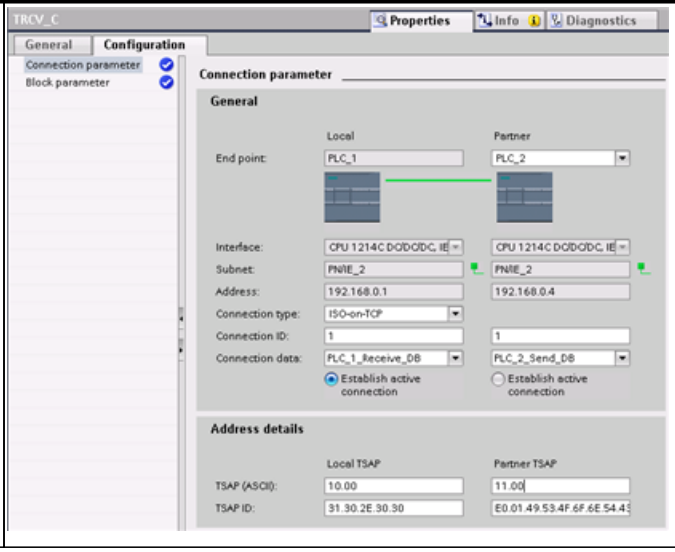
İletişim partnerleri, iletişim bağlantısını ayarlamak ve kurmak için kullanılan yordamları yürütür. Parametrelerle, iletişimin aktif ve pasif uç noktalarını belirtebilirsiniz. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.

Bağlantı kesilirse (örn. bir hat arızası nedeniyle), aktif partner bu bağlantıyı tekrar kurmayı dener. İletişim yordamını tekrar yürütmeniz gerekmez.

Bağlantı yolları

Kullanıcı programına bir TSEND_C, TRCV_C veya TCON yordamı eklendikten sonra, yordamın bir kısmını seçerseniz denetçi penceresinde bağlantının özellikleri görüntülenir. İletişim parametrelerini, "Özellikler" iletişim kutusundaki "Konfigürasyon" sekmesi aracılığıyla belirleyebilirsiniz.

Tablo 11-5 Bağlantı yolunun yapılandırılması (yordamın özellikleri üzerinden)

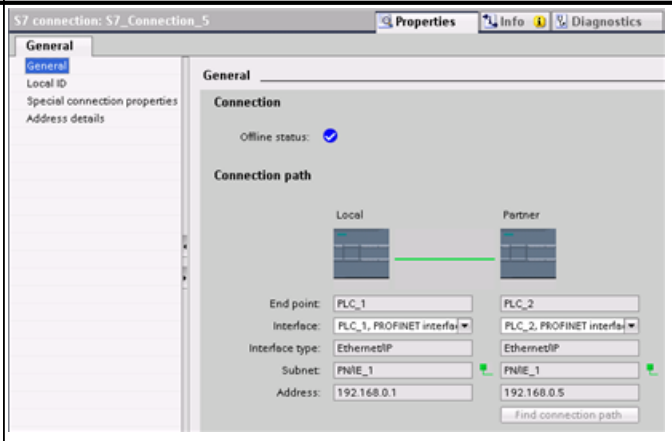
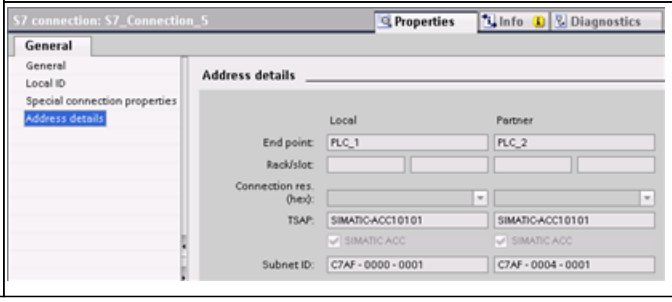
TCP, ISO on TCP ve UDP	Bağlantı özellikleri
TCP, ISO on TCP ve UDP Ethernet protokollerinde yerel ve partner CPU'nun ve bağlantılarını, yordamın (TSEND_C, TRCV_C veya TCON) "Özellikler" kısmında yapılandırabilirsiniz. Şekilde, bir ISO on TCP bağlantısının "Bağlantı" sekmesindeki "Bağlantı özellikleri" gösterilmektedir.	

NOT

Bir CPU için kullanılan bağlantı özelliklerini yapılandırırsanız, STEP 7'de ya partner CPU'daki belirli bir bağlantı DB'yi seçebilir (mevcutsa) ya da partner CPU için bir bağlantı DB kurabilirsiniz. Partner CPU, projede halihazırda oluşturulmuş olmalıdır, "belirtilmemiş" bir CPU söz konusu olmamalıdır.

Yine de bir TSEND_C, TRCV_C veya TCON yordamını partner CPU'nun kullanıcı programına ekleyebilirsiniz. Bir yordam eklerseniz, konfigürasyondan oluşturulan bağlantı DB'yi seçin.

Tablo 11-6 S7 iletişimi için kullanılan bağlantı yolunun yapılandırılması (cihaz konfigürasyonu)

S7 iletişimi (GET ve PUT)	Bağlantı özellikleri
S7 iletişiminde, yerel ile partner CPU arasındaki bağlantıları ağın "Cihazlar ve ağlar" editöründe yapılandırabilirsiniz. "Özellikler"i çağırmak için "Vurgulu: Bağlantı" butonuna tıklayabilirsiniz. "Genel" sekmesinde birçok özellik listelenir: • "Genel" (gösteriliyor) • "Yerel ID" • "Özel bağlantı özellikleri" • "Adres ayrıntıları" (gösteriliyor)	 

"PROFINET" alt bölümündeki "Protokoller" (Sayfa 581-582) başlığında ve "S7 iletişimi" alt bölümündeki "S7 bağlantısının oluşturulması" (Sayfa 753) başlığında daha fazla bilgi ve kullanılabilen iletişim yordamlarının bir listesini bulabilirsiniz.

Tablo 11-7 CPU bağlantısı için kullanılan parametreler

Parametre		Tanım
Adres		Atanmış IP adresleri
Genel	Uç nokta	Partner CPU'nun adı (alıcı)
	Arayüz	Arayüzlerin adı
	Alt ağ	Alt ağların adı
	Arayüz tipi	*Sadece S7 iletişimi*: Arayüzün türü
	Bağlantı tipi	Ethernet protokolünün tipi
	Bağlantı ID'si	ID numarası
	Bağlantı verileri	Yerel CPU ve partner CPU için kullanılan veri belleği
	Aktif bağlantı kurma	Yerel CPU'yu veya partner CPU'yu aktif bağlantı olarak seçmek için kullanılan opsiyon alanı
Adres ayrıntıları	Uç nokta	*Sadece S7 iletişimi*: Partner CPU'nun adı (alıcı)
	Modül taşıyıcısı/yuva	*Sadece S7 iletişimi*: Modül taşıyıcısı ve yuva adresi

[1] Bir S7-1200 CPU'su ile ISO on TCP üzerinden kurulacak bir bağlantıyı yapılandırırken, pasif iletişim katılımcısının TSAP uzantısında sadece ASCII karakterleri kullanın.

Parametre		Tanım
Adres ayrıntıları	Bağlantı kaynağı	*Sadece S7 iletişimi*: TSAP'nin, bir S7-300 veya S7-400 CPU'su ile kurulacak bir S7 bağlantısının konfigürasyonu için kullanılan bileşenleri
	Port (desimal):	TCP ve UPD: Partner CPU'nun desimal formattaki portu
	TSAP [1] ve alt ağ ID'si:	ISO on TCP (RFC 1006) ve S7 iletişimi: Yerel CPU ve partner CPU'nun ASCII ve onaltılık formattaki TSAP'leri

[1] Bir S7-1200 CPU'su ile ISO on TCP üzerinden kurulacak bir bağlantıyı yapılandırırken, pasif iletişim katılımcısının TSAP uzantısında sadece ASCII karakterleri kullanın.

Transport Service Access Points (TSAP'ler)

ISO on TCP protokolü ve S7 iletişimi, TSAP'lerle, tek bir IP adresiyle birden fazla bağlantıya izin verir. TSAP'ler, iletişim uç noktalarının bu bağlantıların bir IP adresine benzersiz bir atamasını belirler.

Kullanılacak TSAP'leri, "Bağlantı parametreleri" iletişim kutusundaki "Adres ayrıntıları" bölümünde tanımlayabilirsiniz. "Yerel TSAP" alanına CPU'daki bir bağlantının TSAP'si girilir. "Partner-TSAP" alanına ise partner CPU'daki bağlantının TSAP'si girilir.

Port numaraları

TCP ve UDP protokollerinde, yerel (aktif) CPU'nun bağlantı parametrelerinin konfigürasyonunda (pasif) partner CPU'nun desantral IP adresi ve port numarası belirtilmelidir.

Kullanılacak portları, "Bağlantı parametreleri" iletişim kutusundaki "Adres ayrıntıları" bölümünde tanımlayabilirsiniz. CPU'daki bir bağlantının portu, "Yerel port" alanına girilir. Partner CPU'daki bağlantı için kullanılan port, "Partner-Port" alanına girilir.

11.5.3 IP adreslerinin atanması

11.5.3.1 Programlama ve ağ cihazları için kullanılan IP adreslerinin atanması

Programlama cihazınız tesisinizin LAN'sine bağlı entegre bir adaptör kartı kullanıyorsa, programlama cihazı ve CPU aynı alt ağda bulunmalıdır. Alt ağı, cihaz için kullanılan IP adresi ve alt ağ maskesinden oluşan bir kombinasyon olarak atayabilirsiniz. Desteğe ihtiyaç duyarsanız, bulunduğunuz yerel ağ yöneticinize başvurun.

C sınıfı bir IP adresinin ağ ID'si, IP adresinin ilk üç sekizlisidir. Örneğin 211.154.184.16'nın ağ ID'si 211.154.184'tür. Hangi IP ağını kullanacağınızı bu bileşen belirler. Alt ağ maskesi normalde 255.255.255.0 değerine sahiptir. Ancak bilgisayarınız bir tesis LAN'sinde bulunduğu için, alt ağ maskesi, benzersiz alt ağlar kurmak için başka değerler alabilir (örn. 255.255.254.0). Alt ağ maskesi, mantıksal AND bağlantısında cihazın IP adresiyle bir IP alt ağının sınırlarını tanımlar.

NOT

Programlama cihazlarınızın, ağ cihazlarınızın ve IP yönlendiricilerinizin dünyayla iletişim kurduğu internette, diğer ağ kullanıcılarıyla çakışmalar oluşmasını engellemek için benzersiz IP adresleri atanmalıdır. Kendinize IP adreslerinin atanmasını sağlamak için, çalışanları şirketinizdeki ağları bilen şirketinizin BT departmanına başvurun.

⚠ İKAZ

Web sunucusu üzerinden CPU'ya yetkisiz erişim

Tam CPU erişim yetkilerine veya hata güvenli özelliği de dahil tam erişim olanağına sahip kullanıcılar PLC değişkenlerini okuyabilir ve yazabilir. Web sunucusu kullanıcıları, CPU için kullanılan erişim kademesinden bağımsız olarak PLC değişkenlerini okuma ve yazma yetkilerine sahip olabilir. CPU'ya yetkisiz kişiler tarafından gerçekleştirilen bir erişim veya PLC değişkenleri için geçersiz değerlerin ayarlanması, proses işletimine zarar verebilir ve hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yetkili kullanıcılar, işletim durumunda değişiklikler yapabilir, PLC verilerini yazabilir ve Firmware güncellemeleri uygulayabilir. Siemens, aşağıdaki güvenlik önlemlerine uyulmasını önerir:

- CPU erişim kademelerini (Sayfa 156) ve Web sunucusu kullanıcı ID'lerini (Sayfa 801) STEP 7'de tanımlandığı şekilde güçlü şifreler aracılığıyla koruyun.
- Web sunucusuna erişimi sadece HTTPS protokolü üzerinden etkinleştirin.
- Web sunucusu kullanıcısının "Herkes" şeklindeki standart asgari yetkilerini genişletmeyin.
- Web sitelerinin kullanıcıları PLC değişkenleri için geçersiz değerler belirleyebileceği için, program lojiğinizdeki değişkenler için bir hata kontrolü ve bir aralık kontrolü gerçekleştirin.
- Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın.

NOT

CPU'nuzu şirket LAN'sine bağlamak istemiyorsanız ikinci bir ağ adaptörü kartı faydalı olur. Bu yapı, özellikle başlangıçtaki testler sırasında veya işletime alma kontrolleri esnasında işe yarar.

Programlama cihazınızın IP adresinin "ağ ortamı" üzerinden atanması veya kontrol edilmesi

Programlama cihazınızın IP adresini atamak veya kontrol etmek için aşağıdaki yöntemi izleyin:

1. Başlat menüsü üzerinden sistem kumanda ünitesini açın.
2. "Ağ ve serbest bırakma merkezini" açın ve CPU'nuza bağlanan ağ adaptörü için kullanılan "LAN bağlantısını" seçin
3. "LAN bağlantısı durumu" iletişim kutusundaki "Özellikler" butonuna tıklayın.
4. "LAN bağlantısının özellikleri" iletişim kutusunda bulunan "Bu bağlantı şu elemanları kullanır:" alanında "İnternet protokolü 4 (TCP/IPv4)"ü seçin.
5. "Özellikler" butonuna tıklayın.
6. "IP adresini otomatik olarak al" opsiyonunu veya statik bir IP adresi girmek için "Şu IP adresini kullan" opsiyonunu seçin.
7. "Şu IP adresini kulla" opsiyonunu seçtiyseniz, IP adresini ve alt ağ maskesini ayarlayın:
   - IP adresini, CPU ile aynı ağ ID'si ve aynı alt ağ maskesi kullanılacak şekilde ayarlayın. Örnek: CPU'nun IP adresi **192.168.0**.1 ise, IP adresi olarak **192.168.0**.200'ü ayarlayabilirsiniz.
   - 255.255.255.0 alt ağ maskesini seçin.
   - Standart Gateway alanını boş bırakın.

Şimdi CPU'ya bağlanabilirsiniz.

NOT

STEP 7'nin CPU'yu bulabilmesi ve onunla iletişim kurabilmesi için ağ arayüz kartı ve CPU aynı alt ağda olmalıdır.

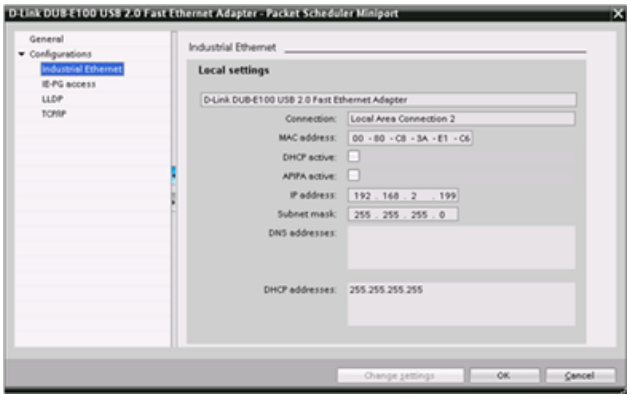
S7-1200 CPU'su ile bağlantı kurmanıza izin veren bir ağ konfigürasyonunu yapılandırırken desteğe ihtiyaç duyuyorsanız BT departmanınıza başvurun.

11.5.3.2 Programlama cihazınızın IP adresinin belirlenmesi

Programlama cihazınızın MAC ve IP adresini şu menü komutlarıyla belirleyebilirsiniz:

1. Proje gezgininde "Çevrimiçi erişimler" düğümünü genişletin.
2. Farenin sağ tuşuyla gereken ağa tıklayın ve "Özellikler" öğesini seçin.
3. "Ağ" iletişim kutusundaki "Konfigürasyonlar" öğesini genişletin ve "Industrial Ethernet"i seçin.

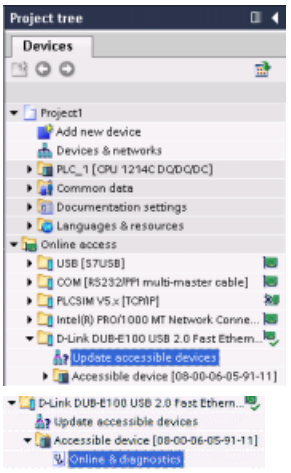
Burada, programlama cihazınızın MAC ve IP adresi görüntülenir.



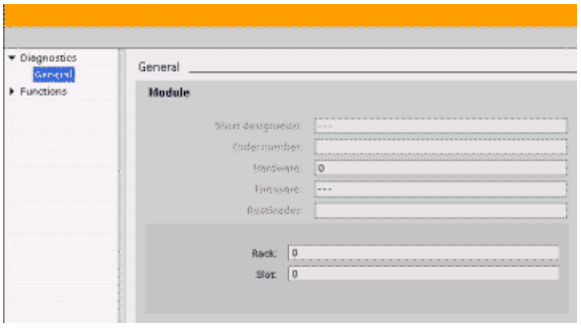
11.5.3.3 Bir IP adresinin bir CPU'ya çevrimiçi olarak atanması

Bir ağ cihazına çevrimiçi olarak bir IP adresi atayabilirsiniz. Bu, ilk cihaz konfigürasyonu sırasında özellikle yararlıdır.

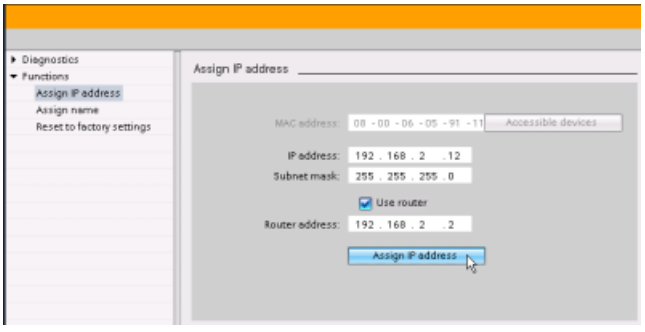
1. Proje gezgininde, CPU'nun yapılandırılmış bir IP adresine sahip olmadığından emin olun. Cihazın bulunduğu ağ için "Çevrimiçi erişim" > <Adaptör kartı öğesini genişletin ve "Erişilebilir katılımcıları güncelle" öğesine çift tıklayın.
STEP 7 bir IP adresi yerine bir MAC adresini görüntülerse hiçbir IP adresi atanmamış demektir.



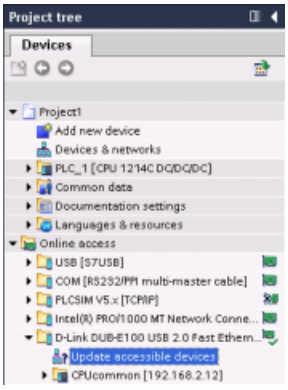
2. Gereken erişilebilir cihazın altındaki "Çevrimiçi ve teşhis" öğesine çift tıklayın.
3. "Çevrimiçi ve teşhis" iletişim kutusunda "Fonksiyonlar" > "IP adresi ata" opsiyonunu seçin.



4. "IP adresi" alanına yeni IP adresinizi girin ve "IP adresi ata" butonuna tıklayın.



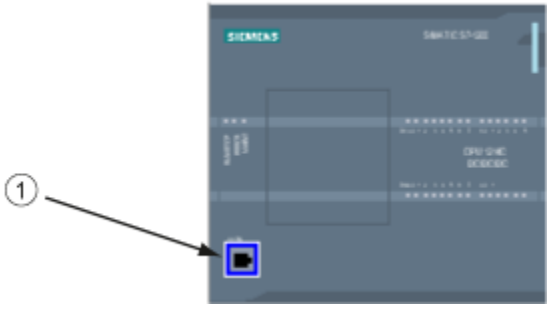
5. Proje gezgininde, STEP 7'nin CPU'ya yeni IP adresini atayıp atamadığını kontrol edin.
Daha önce yapılandırılan IP adresini görüntülemek için "Erişilebilir katılımcıları güncelle" öğesine çift tıklayın.



11.5.3.4 Projenizdeki bir CPU için IP adresinin yapılandırılması

PROFINET arayüzünün yapılandırılması

PROFINET arayüzü için kullanılan parametreleri yapılandırmak için CPU'da yeşil PROFINET alanını seçin. Denetçi penceresinde, PROFINET portu için kullanılan "Özellikler" sekmesi görüntülenir.



① PROFINET portu

IP adresinin yapılandırılması

Ethernet adresi (MAC adresi): Bir PROFINET ağında, tanımlama amacıyla üretici tarafından her cihaza bir MAC adresi (Media-Access-Control-Adresse) atanır. Bir MAC adresi, aktarma sırasıyla ayırma işareti olarak tire (-) veya iki nokta üst üste (:) ile her biri ikişer onaltılık basamaktan oluşan altı gruptan oluşur (örneğin 01-23-45-67-89-AB veya 01:23:45:67:89:AB).

IP adresi: Her cihaz ayrıca bir IP adresine (Internet-Protocol-Adresse) ihtiyaç duyar. Bu adresin yardımıyla bir cihaz verileri daha karmaşık bir ağ üzerinden sağlayabilir.

Her IP adresi dört adet 8 Bit'lik segmente ayrılır ve nokta ayırma işaretleriyle bir desimal formatta gösterilir (örneğin 211.154.184.16). IP adresinin ilk kısmı ağ ID'sidir (aktif ağın kodu), adresin ikinci kısmı Host bilgisayar ID'sidir (ağdaki her cihaz için benzersizdir). 192.168.x.y şeklindeki bir IP adresi, internette olmayan bir özel ağın parçası olarak algılanan standart bir koddur.

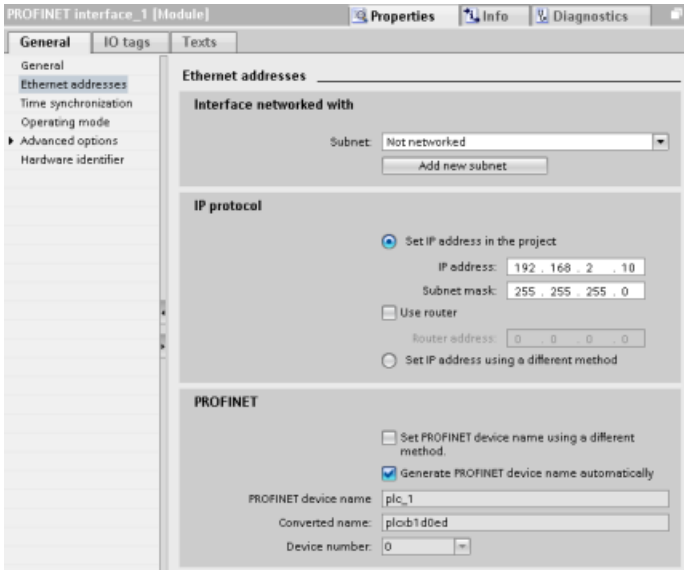
Alt ağ maskesi: Bir alt ağ, birbirine bağlı ağ cihazlarından oluşan bir mantıksal gruplamadır. Bir alt ağın katılımcıları çoğunlukla bir LAN'de (Local Area Network) birbirine yakın şekilde bulunur. Bir maske (alt ağ maskesi veya ağ maskesi), bir IP alt ağının sınırların belirler.

255.255.255.0 şeklindeki bir alt ağ, genel olarak küçük bir yerel ağ için uygundur. Bu, bu ağdaki tüm IP adreslerinin aynı 3 sekizliğe sahip olması gerektiği anlamına gelir. Bu ağdaki münferit cihazlar, son sekizli (8-Bit alanı) aracılığıyla tanımlanır. Bununla ilgili bir örnek olarak, 255.255.255.0 şeklindeki bir alt ağın ve 192.168.2.0 - 192.168.2.255 aralığındaki IP adreslerinin küçük bir yerel ağdaki münferit cihazlara atanması gösterilebilir.

Farklı alt ağların arasındaki tek bağlantı, bir yönlendirici üzerinden çalışır. Alt ağlar kullanılıyorsa, bir IP yönlendiricisi kullanılmalıdır.

IP yönlendiricisi: Yönlendirici, LAN'lerin arasındaki bağlantıdır. Bir bilgisayar, bir yönlendiriciyle bir LAN'de yine başka LAN'lere ait olan başka ağlara mesajlar gönderebilir. Verilerin hedefi LAN'nin dahilinde değilse, yönlendirici, verileri başka bir ağa veya verilerin hedeflerine ulaştırılabileceği ağlardan oluşan bir gruba aktarır.

Yönlendiriciler, veri paketlerini göndermek ve almak için IP adreslerine ihtiyaç duyar.



IP adreslerinin özellikleri: "Özellikler" iletişim kutusunda "Ethernet adresleri" girişini seçin. STEP 7'de, Ethernet adresinin konfigürasyonu için, yazılım projesine, projeye yüklenecek olan CPU'nun IP adresini atayabileceğiniz iletişim kutusu görüntülenir.

Tablo 11-8 IP adresi için kullanılan parametreler

Parametre		Açıklama
Alt ağ	Cihazın bağlı olduğu alt ağın adı. Yeni bir alt ağ oluşturmak için "Yeni alt ağ ekle" butonuna tıklayın. Ön ayar, "bağlı değil" şeklindedir. İki tür bağlantı mümkündür: • Ön ayar "bağlı değil", yerel bir bağlantı kurar. • Ağınızda iki veya daha fazla cihaz varsa, bir alt ağa gerek duyarsınız.	
IP protokolü	IP adresi	CPU'nun atanmış IP adresi
	Alt ağ maskesi	Atanmış alt ağ maskesi
	IP yönlendiricisini kullan	Bir IP yönlendiricisi kullanılıyorsa bu opsiyon kutusunu etkinleştirin
	Yönlendirici adresi	Yönlendiricinin atanmış IP adresi, mevcutsa

NOT

Tüm IP adresleri, proje yüklenirken yapılandırılır. CPU önceden yapılandırılmış bir IP adresine sahip değilse, projeyi hedef cihazın MAC adresiyle atamanız gerekir. CPU'nuz bir yönlendiriciye veya bir ağa bağlıysa, yönlendiricinin IP adresi de girilmelidir.

Programı CPU'ya yükledikten sonra, "IP adresini başka bir yöntemle ayarla" opsiyon alanı aracılığıyla IP adresini çevrimiçi olarak veya "T_CONFIG (Sayfa 669)" yordamı aracılığıyla değiştirebilirsiniz. IP adresinin bu atama şekli sadece CPU için geçerlidir.

⚠ İKAZ

Bir donanım konfigürasyonunun "IP adresinin başka bir yöntemle ayarla" opsiyonuyla yüklenmesi

Bir donanım konfigürasyonu etkinleştirilmiş durumdaki "IP adresini başka bir yöntemle ayarla" opsiyonuyla yüklendikten sonra, CPU'nun işletim durumunu RUN iken STOP olarak veya STOP iken RUN olarak ayarlamak mümkün değildir.

Kullanıcı cihazları bu koşullarda çalışmaya devam eder ve uygun güvenlik önlemleri alınmadığı takdirde, makine veya proses, öngörülemeyen şekilde çalışabilir ve bu da hayati tehlikeye yol açabilecek yaralanmalara veya maddi hasarlara neden olabilir.

CPU'nuzun IP adresinin (IP adreslerinin), CPU'yu gerçek bir otomasyon ortamında kullanmanızdan önce belirlendiğinden emin olun. Bu kontrol için STEP 7 programlama paketi, SIMATIC Automation Tool veya bağlı başka bir HMI cihazını T_CONFIG yordamıyla bağlantılı olarak kullanabilirsiniz.

⚠ İKAZ

PROFINET ağın muhtemelen duracağı koşul

Bir CPU'nun IP adresini çevrimiçi olarak veya kullanıcı programı üzerinden değiştirirseniz, bu durum, PROFINET ağının durmasına neden olabilir.

Bir CPU'nun IP adresi, alt ağın dışındaki bir IP adresi ile değiştirilirse, PROFINET ağı iletişimi kaybeder ve tüm veri alışverişi durur. Kullanıcı cihazları, bu koşullarda çalışmaya devam edecek şekilde yapılandırılabilir. PROFINET iletişiminin kaybolması, uygun güvenlik önlemleri alınmadığı takdirde makine veya prosesin öngörülemeyen şekilde çalışmasına ve bu da hayati tehlikeye yol açabilecek yaralanmalara veya maddi hasarlara neden olabilir.

Bir IP adresinin manuel olarak değiştirilmesi gerekiyorsa, yeni IP adresinin alt ağın dahilinde olduğundan emin olun.

PROFINET portunun yapılandırılması

CPU, PROFINET arayüzünün portlarını standart olarak otomatik anlaşma için yapılandırır. Otomatik anlaşmanın sorunsuz bir şekilde çalışması için her iki istasyonu da otomatik anlaşma için ayarlamanız gerekir. Bir istasyon sabit bir konfigürasyona sahipse (örn. 100 MBit/sn'de tam dupleks) ve diğer istasyon otomatik anlaşma için ayarlandıysa, otomatik anlaşma başarısız olur ve bu da yarı dupleks işletime neden olur.

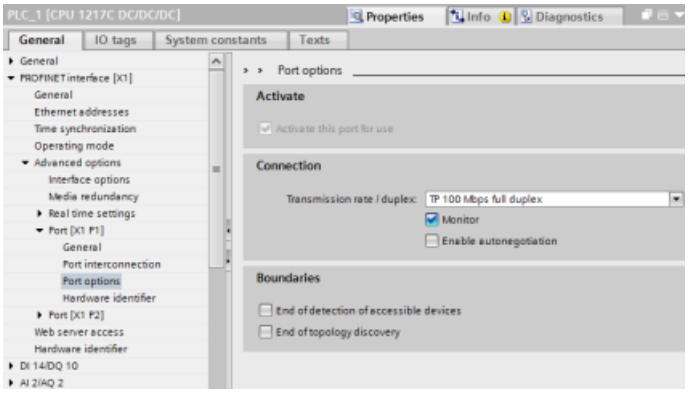
S7-1200, otomatik anlaşmayla ilgili bu kısıtlamayı aşmak için, otomatik anlaşmayı devre dışı bırakmaya yönelik bir opsiyon sunar. Otomatik anlaşmayı devre dışı bırakırsanız, S7-1200 otomatik olarak 100 MBit/sn'de tam dupleks işletime uygun şekilde yapılandırılır.

Aktarma hızını ve tam dupleks opsiyonunu her port için sabit bir değere ayarlayabilirsiniz:

1. Gelişmiş opsiyonları ve yapılandırmak istediğiniz portu seçin. Ardından "Port opsiyonlarını" seçin.
2. "Aktarma hızı/Tam dupleks" alanındaki "Bağlantı" öğesinin altında şu opsiyonlardan birini seçin:
   - Otomatik olarak: CPU ve partner cihaz, otomatik anlaşma aracılığıyla portun aktarma hızını ve tam dupleks ayarını belirler.
   - TP 100 MBit/sn tam dupleks: Otomatik anlaşmayı devre dışı bırakırsanız, port 100 MBit/sn tam dupleks değeriyle çalışır. Otomatik anlaşmayı etkinleştirirseniz, port 100 MBit/sn tam dupleks değeriyle veya CPU ile partner cihaz arasında otomatik anlaşma yoluyla belirlenen başka bir aktarma hızı/tam dupleks değeriyle çalışabilir. "Denetle" opsiyonu seçildiyse, teşhis tamponuna bir mesaj kaydedilir (bkz. alt kısım).
3. Denetleme: Bu opsiyon kutusunu etkinleştirirseniz, portta aşağıdaki durumlardan biri ortaya çıktığında teşhis tamponuna bir mesaj kaydedilir:
   - Portta bağlantı kurulamıyor
   - Kurulan bir bağlantı başarısız oldu
   - Aktarma hızı/tam dupleks ayarı olarak "TP 100 MBit/sn tam dupleks" opsiyonunu seçiyorsunuz ve CPU, otomatik anlaşma aracılığıyla anlaşmaya varılan 100 MBit/sn'den farklı bir aktarma hızı veya üzerinde anlaşmaya varılan yarı dubleksten farklı bir tam dupleks ayarıyla bir bağlantı kuruyor.
4. Otomatik anlaşmayı etkinleştir: "Aktarma hızı/tam dupleks" alanında 100 MBit/sn'de tam dupleks opsiyonunu seçerseniz, otomatik anlaşmayı devre dışı bırakabilirsiniz. Otomatik anlaşmayı devre dışı bırakmak için, "Otomatik anlaşmayı etkinleştir" opsiyon kutusunu devre dışı bırakın.

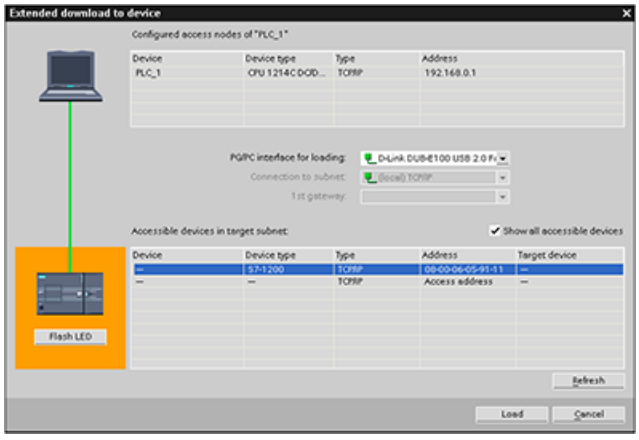
NOT

Otomatik anlaşmayı devre dışı bırakmazsanız, CPU ve partner cihaz, portun aktarma hızı ve tam dupleks ayarını otomatik anlaşma yoluyla belirler.



11.5.4 PROFINET ağının test edilmesi

Konfigürasyon tamamlandıktan sonra CPU'ya programı yükleyin (Sayfa 194). Tüm IP adresleri, proje yüklenirken yapılandırılır.



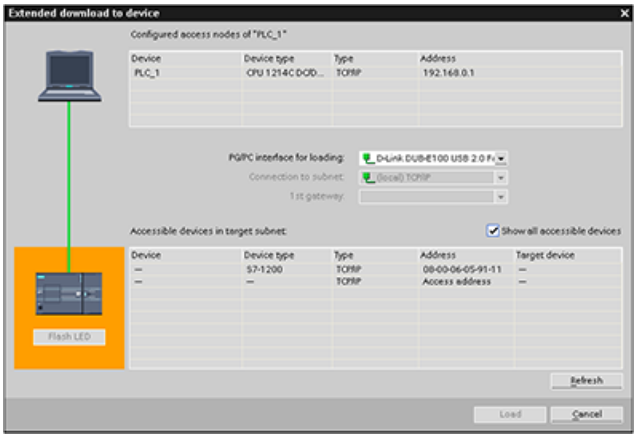
Bir IP adresinin bir cihaza çevrimiçi olarak atanması

S7-1200 CPU'sunun önceden yapılandırılmış bir IP adresi yoktur. Bu nedenle CPU'ya manuel olarak bir IP adresi atamanız gerekir.

- Bir cihaza bir IP adresini çevrimiçi olarak nasıl atayacağınızı, "Cihaz konfigürasyonu: Bir CPU'ya IP adresinin çevrimiçi olarak atanması" (Sayfa 569) bölümünden öğrenebilirsiniz.
- Projenizde bir IP adresi atamak için, IP adresini cihaz konfigürasyonunda yapılandırmanız, konfigürasyonu kaydetmeniz ve PLC'ye yüklemeniz gerekir. Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Projenizdeki bir CPU için IP adresinin yapılandırılması" (Sayfa 571).

"Gelişmiş yükleme" iletişim kutusunda bağlı ağ cihazlarının sorgulanması

S7-1200 CPU'sunun ve ilgili "Gelişmiş yükleme" iletişim kutusunun "Cihaza yükle" fonksiyonu, erişilebilir tüm ağ cihazlarını gösterebilir ve ayrıca tüm cihazlara benzersiz IP adreslerinin atanıp atanmadığını belirtebilir. Tüm erişilebilir ve kullanılabilir cihazları kendi atanmış MAC veya IP adresleriyle görüntülemek için "Tüm erişilebilir katılımcıları göster" opsiyon kutusunu etkinleştirin.



Gereken ağ cihazı bu listede yoksa, bu cihazla kurulan iletişim herhangi bir nedenle kesilmiş demektir. Bu durumda cihaz ve ağ, donanım ve/veya konfigürasyon hatası açısından kontrol edilmelidir.

11.5.5 CPU'nun Ethernet adresinin (MAC adresi) belirlenmesi

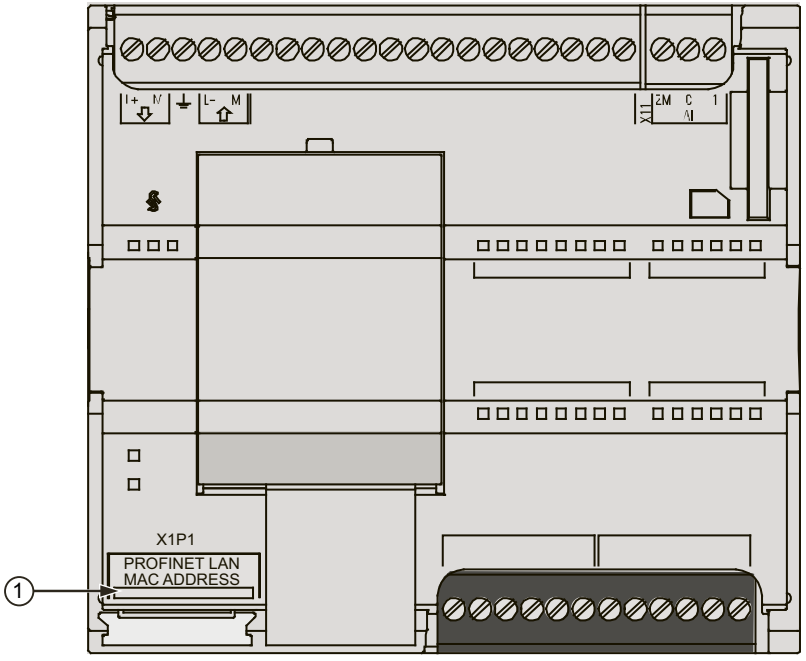
PROFINET ağlarında MAC adresi (Media-Access-Control adresi), ağ arayüzlerinin benzersiz bir şekilde işaretlenmesi için üretici tarafından aranan bir koddur. MAC adresi genelde üreticinin kayıtlı tanımlama numarasını içerir.

MAC adreslerinin düz metin olarak çıktısının alınması için kullanılan standart format (IEEE 802.3), aktarma sırasına göre ayırma işareti olarak tireler (-) veya iki nokta üst üste işaretleri (:) ile ikişer onaltılık basamak içeren altı gruptan oluşur (örneğin 01-23-45-67-89-ab veya 01:23:45:67:89:ab).

NOT

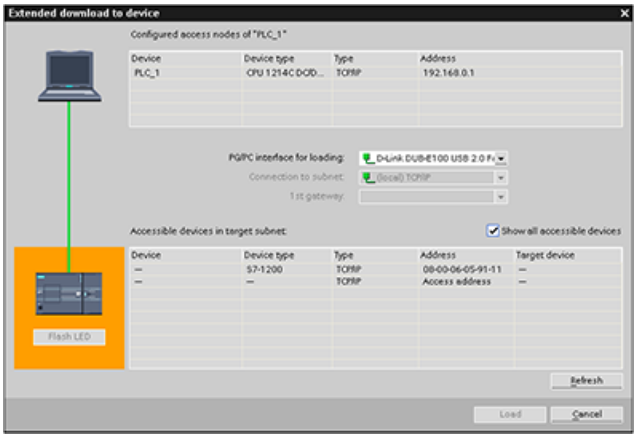
Her CPU'da, fabrikada kalıcı, benzersiz bir MAC adresi ayarlanır. Bir CPU'nun MAC adresini değiştiremezsiniz.

MAC adresi, CPU'nun ön tarafında sol alta basılır. MAC adresini okuyabilmek için alt koruyucu kapağı açmanız gerekir.



① MAC adresi

CPU başlangıçta bir IP adresi yerine, sadece fabrikada ayarlanmış bir MAC adresine sahiptir. PROFINET iletişiminde, tüm cihazlara benzersiz bir IP adresinin atanması gerekir.



CPU'nun "Cihaza yükle" fonksiyonu ve "Gelişmiş yükleme" iletişim kutusu aracılığıyla ağdaki erişilebilir tüm katılımcıları görüntüleyebilir ve ayrıca tüm cihazlara benzersiz IP adreslerinin atanıp atanmadığını kontrol edebilirsiniz. Bu iletişim kutusunda, atanmış MAC veya IP adresleriyle birlikte erişilebilir ve kullanılabilir durumdaki tüm cihazlar görüntülenir. MAC adresleri, benzersiz bir IP adresi olmayan cihazları tanımlamak için özellikle önemlidir.

11.5.6 NTP senkronizasyonunun (Network Time Protocol, NTP) yapılandırılması

Network Time Protocol (NTP), çok yaygındır ve bilgisayar sistemleri için saatleri internet üzerinde bulunan saat sunucuları ile senkronize etmek için kullanılır. NTP yönteminde CPU, alt ağdaki (LAN) NTP sunucusuna düzenli zaman aralıklarıyla saat sorguları (istemci modunda) gönderir. Sunucunun cevapları aracılığıyla en güvenilir ve en doğru saat belirlenir ve istasyonun saati senkronize edilir.

⚠ İKAZ

NTP senkronizasyonu (Network Time Protocol) nedeniyle yetkisiz kişilerin ağa erişimi tehlikesi

Bir saldırgan, NTP senkronizasyonu (Network Time Protocol) üzerinden ağlarınıza erişebilirse, CPU sistem saatini değiştirerek prosesinizin üzerindeki kontrolü elinizden alabilir. Bu tür bir proses kontrolü aktarımı nedeniyle ölüm, ağır yaralanma durumları veya maddi hasarlar oluşabilir.

S7-1200 CPU'su NTP istemci fonksiyonunu standart olarak devre dışı bırakır. NTP fonksiyonu etkinleştirilirse, sadece yapılandırdığınız IP adresleri NTP sunucusu olarak çalışabilir. NTP fonksiyonunu, CPU sistem saati uzak sunucular tarafından düzeltilebilecek şekilde ayarlamanız gerekir.

S7-1200 CPU'su, doğru bir CPU sistem saatine bağlı olan saat alarmlarını ve saat işlemlerini destekler. NTP'yi yapılandırır ve saat senkronizasyonunun bir sunucudan gerçekleştirilmesini kabul ederseniz, sunucuda güvenilir bir kaynağın söz konusu olduğundan emin olmanız gerekir. Aksi takdirde, bilinmeyen bir kullanıcının CPU sistem saatini değiştirebileceği ve bu şekilde prosesinizin üzerindeki kontrolü elinizden alabileceği bir güvenlik ihlali ortaya çıkabilir.

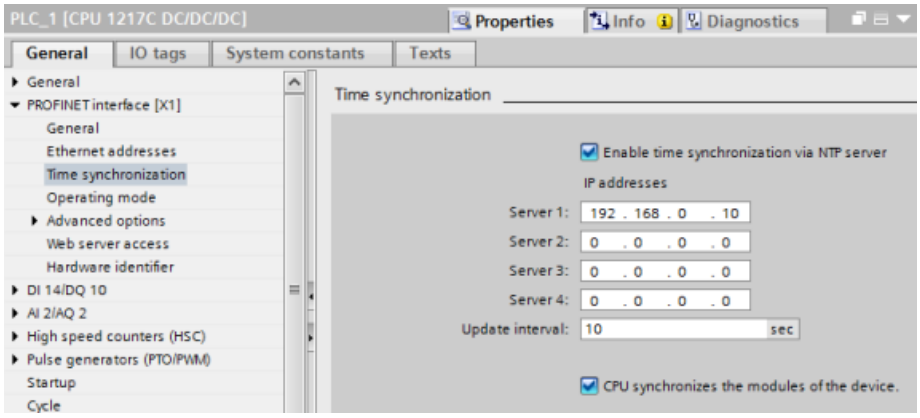
Güvenlikle ilgili bilgileri ve önerileri, Siemens'in "Service & Support" web sitesindeki Operational Guidelines for Industrial Security (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf) bölümümüzde bulabilirsiniz.

Bu yöntemin avantajı, alt ağları kapsayan senkronizasyon için zaman tanımasıdır.

Maksimum dört NTP sunucusunun IP adreslerini yapılandırabilirsiniz. Güncelleme aralığı, saat sorgularının arasındaki süreyi belirler (saniye cinsinden). Aralığın değeri 10 saniye ile bir gün arasındadır.

NTP yönteminde, genelde sunucu üzerinden UTC (Universal Time Coordinated) bilgisi aktarılır, bu da GMT'ye karşılık gelir (Greenwich Mean Time).

CPU cihaz konfigürasyonunun özellikler penceresinde "Saat senkronizasyonu" girişini seçin. STEP 7'de "Saat senkronizasyonu" iletişim kutusu görüntülenir:



NOT

CPU, proje yüklenirken tüm IP adreslerini alır.

Tablo 11-9 Saat senkronizasyonu için kullanılan parametreler

Parametre	Tanım
Saat senkronizasyonunun NTP sunucusu üzerinden etkinleştirilmesi	Saat senkronizasyonunu NTP sunucusu üzerinden etkinleştirmek için bu opsiyon kutusunu etkinleştirin.
Sunucu 1	Ağ saati sunucusu 1 için atanan IP adresi
Sunucu 2	Ağ saati sunucusu 2 için atanan IP adresi
Sunucu 3	Ağ saati sunucusu 3 için atanan IP adresi
Sunucu 4	Ağ saati sunucusu 4 için atanan IP adresi
Saat senkronizasyonu için kullanılan güncelleme aralığı	Aralık değeri (sn)
CPU, cihazın modüllerini senkronize eder.	CP saatini CPU saatiyle senkronize etmek için bu opsiyon kutusunu etkinleştirin.

11.5.7 Yol alma süresi, PROFINET cihazlarının adlandırılması ve adres ataması

PROFINET IO, sisteminiz için yol alma süresini uzatabilir (yapılandırılabilir zaman aşımı değeri). Daha fazla sayıda cihaz ve yavaş cihazlar, RUN işletim durumuna geçmek için ihtiyaç duyulan süreyi etkiler.

S7-1200 PROFINET ağınızda maksimum on altı PROFINET IO cihazınız olabilir.

Her istasyon (veya IO cihazı), yol alma sırasında bağımsız olarak çalışmaya başlar ve bu, CPU'nun toplam yol alma süresini etkiler. Yapılandırılabilir zaman aşımı değerini çok düşük olacak şekilde ayarlarsanız, muhtemelen tüm istasyonların yol alma işlemini tamamlaması için CPU yeterli toplam yol alma süresine sahip olmaz. Bu durum ortaya çıkarsa yanlış istasyon hataları oluşur.

Yapılandırılabilir zaman aşımı için standart değer 60.000 msn'dir (1 dakika). Bu değeri, CPU özellikleri > Yol alma > Konfigürasyon süresi öğelerini kullanarak yapılandırabilirsiniz.

PROFINET cihazlarının STEP 7'deki adları ve adresleri

Tüm PROFINET cihazları bir cihaz adına ve bir IP adresine sahip olmak **zorundadır**. Cihaz adlarını ve IP adreslerini STEP 7'de belirleyebilirsiniz. Cihaz adları, PROFINET DCP (Discovery and Configuration Protocol - Algılama ve konfigürasyon protokolü) üzerinden IO cihazlarına yüklenir.

Sistem yol alırken PROFINET adres ataması

Kumanda ünitesi, cihazların adlarını ağa aktarır ve cihazlar kendi MAC adresleriyle tepki verir. Daha sonra kumanda ünitesi cihaza, PROFINET DCP protokolü üzerinden bir IP adresi atar:

- MAC adresi yapılandırılmış bir IP adresine sahipse, istasyon, yol alma işlemini gerçekleştirir.
- MAC adresi, yapılandırılmış bir IP adresine sahip değilse, STEP 7, projede yapılandırılan adresi atar ve istasyon, yol alma işlemini bundan sonra gerçekleştirir.
- Bu işlem sırasında bir sorun mevcutsa bir istasyon hatası oluşur ve yol alma işlemi gerçekleşmez. Bu durum, yapılandırılabilen zaman aşımı değerinin aşılmasına neden olur.

11.5.8 Açık kullanıcı iletişimi

11.5.8.1 Protokoller

CPU'nun entegre PROFINET portu, bir Ethernet ağı üzerinden iletişim standardını destekler:

- Transport Control Protocol (TCP)
- ISO on TCP (RFC 1006)
- User Datagram Protocol (UDP)

Tablo 11-10 Protokoller ve iletişim yordamları

Protokol	Kullanım örnekleri	Verilerin alma alanına girilmesi	İletişim yordamları	Adresleme türü
TCP	CPU-CPU iletişimi Telegramların taşınması	Geçici mod	Sadece TRCV_C ve TRCV	Yerel cihazlara (aktif) ve partner cihazlara (pasif) port numaraları atar
		Belirtilen uzunlukta veri alma	TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV	
ISO on TCP	CPU-CPU iletişimi Mesajların parçalara ayrılması ve yapısı	Geçici mod	Sadece TRCV_C ve TRCV	Yerel cihazlara (aktif) ve partner cihazlara (pasif) TSAP'ler atar
		Protokol kontrollü	TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV	
UDP	CPU-CPU iletişimi Kullanıcı programındaki iletişim	User Datagram Protocol	TUSEND ve TURCV	Yerel cihazlara (aktif) ve partner cihazlara (pasif) port numaraları atar, ancak burada bir özel bağlantı söz konusu değildir
S7 iletişimi	CPU-CPU iletişimi Verilerin bir CPU'dan okunması/bir CPU'ya yazılması	Belirtilen uzunlukta veri aktarımı ve veri alma	GET ve PUT	Yerel cihazlara (aktif) ve partner cihazlara (pasif) TSAP'ler atar
PROFINET IO	CPU ile PROFINET IO cihazı arasındaki iletişim	Belirtilen uzunlukta veri aktarımı ve veri alma	Entegre	Entegre

11.5.8.2 TCP ve ISO on TCP

Transport Control Protocol (TCP), RFC 793 tarafından tanımlanan bir standart protokoldür: Transmission Control Protocol. TCP'nin temel amacı, iki proses çifti arasında güvenilir, güvenli bir bağlantı sağlamaktır. Bu protokol aşağıdaki özelliklere sahiptir:

- Donanımla yakından bağlantılı olduğu için verimli iletişim protokolü
- Orta büyüklükteki veri miktarlarından büyük veri miktarlarına (maksimum 8192 Bayt) kadar uygundur
- Uygulamalar için belirgin şekilde daha fazla performans özelliği sunar, özellikle hata durumunda geri yükleme, akış kontrolü ve güvenilirlik
- Bağlantı odaklı bir protokol
- Sadece TCP'yi destekleyen harici sistemlerle çok esnek bir şekilde kullanılabilir
- Yönlendirme özellikli
- Sadece statik veri uzunlukları geçerlidir.
- Mesajlar onaylanır.
- Uygulamalar, port numaraları üzerinden adreslenir.
- TELNET ve FTP gibi kullanıcı uygulamalarının çoğu protokolü TCP'yi kullanır.
- SEND/RECEIVE programlama arayüzü nedeniyle, veri yönetimi için programlama çalışmaları gerekir.

International Standards Organization (ISO) on Transport Control Protocol (TCP) (RFC 1006) (ISO on TCP), ISO uygulamalarını TCP/IP ağına aktarmaya olanak sağlayan bir yöntemdir. Bu protokol aşağıdaki özelliklere sahiptir:

- Donanımla yakından bağlantılı verimli iletişim protokolü
- Orta büyüklükteki veri miktarlarından büyük veri miktarlarına (maksimum 8192 Bayt) kadar uygundur
- TCP'nin tersine, mesaj fonksiyonu ve veri sonu algılama özelliği mesaj odaklıdır.
- Yönlendirme özellikli, WAN'de kullanım mümkündür
- Dinamik veri uzunluklarını kullanmak mümkündür.
- SEND/RECEIVE programlama arayüzü nedeniyle, veri yönetimi için programlama çalışmaları gerekir.

TCP protokolü, Transport Service Access Points (TSAP'ler) ile tek bir IP adresiyle çok sayıda bağlantıya (maksimum 64-K bağlantı) izin verir. RFC 1006'da TSAP'ler, iletişim uç noktalarının bu bağlantılarının bir IP adresine benzersiz bir atamasını belirler.

11.5.8.3 Geçici mod

TCP ve ISO on TCP protokolleri, tipik olarak 1 ila 8192 Bayt'lık sabit olarak belirtilen bir uzunluğa sahip veri paketlerini alır. Ancak TRCV_C ve TRCV iletişim yordamları, 1 ila 1472 Bayt arasında değişken uzunluğa sahip veri paketlerinin alınabildiği bir geçici iletişim modu da sunar.

NOT

Verileri bir "optimize edilmiş" DB'ye (sadece sembolik olarak adreslenebilir) kaydederseniz, sadece Byte, Char, USInt ve SInt veri tiplerindeki Array'lerde bulunan verileri alabilirsiniz.

TRCV_C veya TRCV yordamını geçici mod için yapılandırmak için, yordamın ADHOC giriş parametresini ayarlayın.
TRCV_C veya TRCV yordamını geçici modda çok sık çağırmazsanız, bir çağrıda birden çok paket alabilirsiniz. Örnek: Bir çağrıda beş adet 100 Bayt'lık paket almak istiyorsanız, TCP bu beş paketi 500 Bayt'lık bir paket olarak sağlar, bunun tersine ISO on TCP ise paketi, beş adet 100 Bayt'lık pakete böler.

11.5.8.4 İletişim hizmetleri ve kullanılan port numaraları

S7-1200 CPU'su aşağıdaki tabloda listelenen protokolleri destekler. CPU, her protokolde adres parametrelerini, ilgili iletişim katmanını ve iletişim rolü ile iletişim yönünü atar.

Bu bilgiler, otomasyon sistemini korumak için kullanılan güvenlik önlemlerinin kullanılan protokollere (örn güvenlik duvarı) uyarlanmasına olanak sağlar. Sadece Ethernet ve PROFINET ağları güvenlik önlemlerine sahiptir. PROFIBUS güvenlik önlemleri sunmadığı için tablo PROFIBUS protokollerini içermez.

Aşağıdaki tabloda, CPU tarafından kullanılan çeşitli katmanlar ve protokoller gösterilmektedir:

Protokol	**Port numarası**	**(2) Güvenlik katmanı** **(4) Taşıma katmanı**	**Fonksiyon**	**Açıklama**
PROFINET protokolleri				
DCP (Discovery and Configuration Protocol, algılama ve konfigürasyon protokolü)	İlgili değil	(2) Ethernet II ve IEEE 802.1Q ve Ether tipi 0x8892 (PROFINET)	PROFINET algılama ve konfigürasyonun erişilebilir katılımcıları	PROFINET, katılımcıları algılamak ve temel ayarları sunmak için DCP'yi kullanır. DCP, özel Multicast-MAC adresini kullanır: 01-0E-CF-xx-xx-xx burada xx-xx-xx = Organizationally Unique Indentifier'dır (üretici kodu)
LLDP (Link Layer Discovery Protocol)	İlgili değil	(2) Ethernet II ve IEEE 802.1Q ve Ether tipi 0x88CC (PROFINET)	PROFINET Link Layer Discovery Protocol	PROFINET, komşu PROFINET cihazlarının arasındaki ilişkileri algılamak ve yönetmek için LLDP'yi kullanılır. LLDP, özel Multicast-MAC adresini kullanır: 01-80-C2-00-00-0E

11.5.8.5 Açık kullanıcı iletişimi için kullanılan yordamların bağlantı ID'leri

TSEND_C, TRCV_C veya TCON PROFINET yordamını kullanıcı programınıza eklerseniz, STEP 7, cihazların arasındaki iletişim kanalının (veya bağlantının) konfigürasyonu için bir örnek DB oluşturur. Bağlantının parametrelerini yordamın "Özellikler" (Sayfa 563) kısmında yapılandırabilirsiniz. Parametrelerin arasında, bağlantıya ait bağlantı ID'si de yer alır.

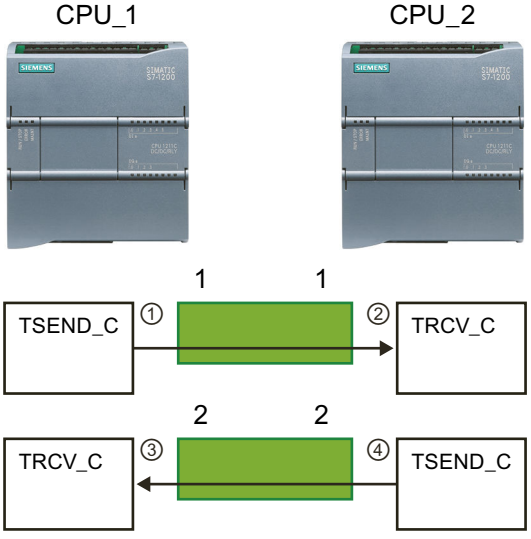
- Bağlantı ID'si, CPU için benzersiz olmalıdır. Kurduğunuz her bağlantı, başka bir DB'ye ve başka bir bağlantı ID'sine ihtiyaç duyar.
- Yerel CPU ve partner CPU, aynı bağlantı için aynı bağlantı ID'sini kullanabilir, ancak bağlantı ID'leri aynı olmak zorunda değildir. Bağlantı ID'si, sadece ilgili CPU'ya ait kullanıcı programının içindeki PROFINET yordamları için geçerlidir.
- CPU'nun bağlantı ID'si için her rastgele numarayı kullanabilirsiniz. Ancak bağlantı ID'lerini sıralı şekilde "1"den itibaren atarsanız, belirli bir CPU için kullanılan bağlantıların sayısını daha kolay anlayabilirsiniz.

NOT

Kullanıcı programınızdaki her TSEND_C, TRCV_C veya TCON yordamı yeni bir bağlantı kurar. Her bağlantı için doğru bağlantı ID'sini kullanmanız önemlidir.

Aşağıdaki örnekte, verilerin gönderilmesi ve alınması için iki ayrı bağlantı kullanan iki CPU arasındaki iletişim gösterilmektedir.

- CPU_1'deki TSEND_C yordamı, ilk bağlantı üzerinden ("her iki CPU'da da bağlantı ID'si 1", CPU_1 ve CPU_2) CPU_2'deki TRCV_C yordamıyla bağlantılıdır.
- CPU_1'deki TRCV_C yordamı, ikinci bağlantı üzerinden ("her iki CPU'da da bağlantı ID'si 2", CPU_1 ve CPU_2) CPU_2'deki TSEND_C yordamıyla bağlantılıdır.



① CPU_1'deki TSEND_C yordamı bir bağlantı kurar ve bu bağlantıya bir bağlantı ID'si atar (CPU_1'de bağlantı ID'si 1).

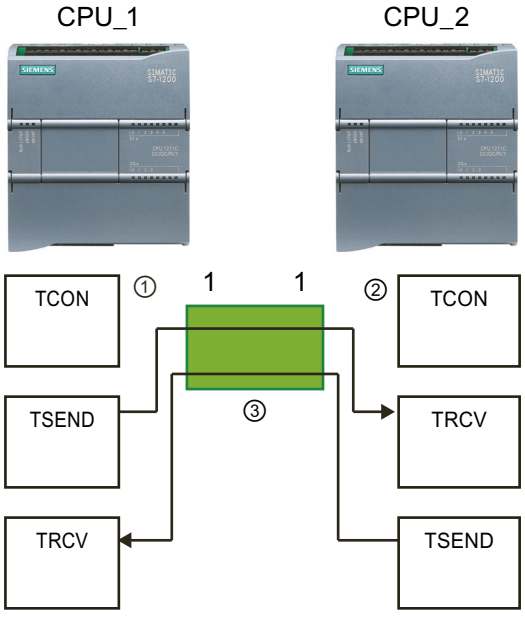
② CPU_2'deki TRCV_C yordamı, CPU_2 için kullanılan bağlantıyı kurar ve bağlantı ID'sini atar (CPU_2'de bağlantı ID'si 1).

③ CPU_1'deki TRCV_C yordamı, CPU_1 için ikinci bir bağlantı kurar ve bu bağlantıya başka bir bağlantı ID'si atar (CPU_1'de bağlantı ID'si 2).

④ CPU_2'deki TSEND_C yordamı, ikinci bir bağlantı kurar ve bu bağlantıya başka bir bağlantı ID'si atar (CPU_2'de bağlantı ID'si 2).

Aşağıdaki örnekte, verilerin gönderilmesi ve alınması için sadece bir bağlantı kullanan iki CPU arasındaki iletişim gösterilmektedir.

- Her CPU, iki CPU arasındaki bağlantıyı yapılandırmak için bir TCON yordamı kullanır.
- CPU_1'deki TSEND yordamı, CPU_1'deki TCON yordamı tarafından yapılandırılmış olan bağlantı ID'si ("bağlantı ID'si 1") üzerindenCPU_2'deki TRCV yordamıyla bağlantılıdır. CPU_2'deki TRCV yordamı, CPU_2'deki TCON yordamı tarafından yapılandırılmış olan bağlantı ID'si ("bağlantı ID'si 1") üzerindenCPU_1'deki TSEND yordamıyla bağlantılıdır.
- CPU_2'deki TSEND yordamı, CPU_2'deki TCON yordamı tarafından yapılandırılmış olan bağlantı ID'si ("bağlantı ID'si 1") üzerindenCPU_1'deki TRCV yordamıyla bağlantılıdır. CPU_1'deki TRCV yordamı, CPU_1'deki TCON yordamı tarafından yapılandırılmış olan bağlantı ID'si ("bağlantı ID'si 1") üzerindenCPU_2'deki TSEND yordamıyla bağlantılıdır.



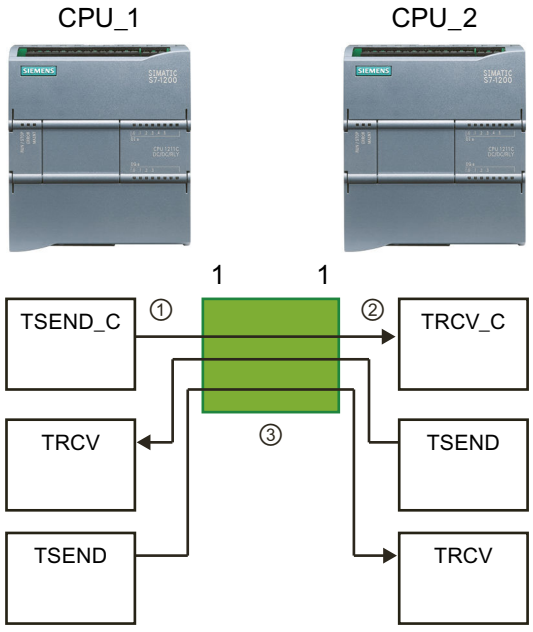
① CPU_1'daki TCON yordamı bir bağlantı kurar ve bu bağlantıya CPU_1 'de bir bağlantı ID'si atar (ID = 1).

② CPU_2'daki TCON yordamı bir bağlantı kurar ve bu bağlantıya CPU_2 'de bir bağlantı ID'si atar (ID = 1).

③ CPU_1'deki TSEND ve TRCV yordamları, TCON tarafından CPU_1'de oluşturulan bağlantı ID'sini (ID = 1) kullanır.

CPU_2'deki TSEND ve TRCV yordamları, TCON tarafından CPU_2'de oluşturulan bağlantı ID'sini (ID = 1) kullanır.

Aşağıdaki örnekte gösterildiği gibi, münferit TSEND ve TRCV yordamlarıyla da bir TSEND_C veya TRCV_C yordamı tarafından kurulan bir bağlantı üzerinden iletişim kurabilirsiniz. TSEND ve TRCV yordamları yeni bir bağlantı kurmaz, bu nedenle bir TSEND_C, TRCV_C veya TCON yordamı tarafından oluşturulan DB'yi ve bağlantı ID'sini kullanmanız gerekir.



① CPU_1'daki TSEND_C yordamı bir bağlantı kurar ve bu bağlantıya bir bağlantı ID'si atar (ID = 1).

② CPU_2 'daki TRCV_C yordamı bir bağlantı kurar ve bu bağlantıya CPU_2'de bir bağlantı ID'si atar (ID = 1).

③ CPU_1'deki TSEND ve TRCV yordamları, TSEND_C tarafından CPU_1'de oluşturulan bağlantı ID'sini (ID = 1) kullanır.
CPU_2'deki TSEND ve TRCV yordamları, TRCV_C tarafından CPU_2'de oluşturulan bağlantı ID'sini (ID = 1) kullanır.

11.5.8.6 PROFINET bağlantısı için kullanılan parametreler

TSEND_C, TRCV_C ve TCON yordamları, partner cihazla bağlantı kurabilmek için bağlantıyla ilgili parametrelere ihtiyaç duyar. Bu parametreler, TCP, ISO on TCP ve UDP protokolleri için TCON_Param yapısı tarafından atanır. Bu parametreyi, "Konfigürasyon" (Sayfa 563) sekmesindeki yordamın "Özellikler" kısmında belirtin. "Konfigürasyon" sekmesine erişilemiyorsa, TCON_Param yapısını, yordamın parametrelerinde belirtmeniz gerekir.

TCP ve UDP için kullanılan iletişim bağlantılarını tam nitelikli etki alanı adları üzerinden yapılandırmak için TCON_QDN ve TCON_QDN_SEC yapısını kullanın. TCP için kullanılan iletişim bağlantılarını Secure Communication ile tam nitelikli etki alanı adları üzerinden yapılandırmak için TCON_QDN_SEC yapısını kullanın.

TCON_Param

Tablo 11-11 Bağlantı açıklamasının yapısı (TCON_Param)

Bayt	Parametre ve veri tipi		Açıklama
0 ila 1	block_length	UInt	Uzunluk: 64 Bayt (sabit)
2 ila 3	id	CONN_OUC (Word)	Bu bağlantıya verilen referans: Değer aralığı: 1 (standart) ila 4095. TSEND_C, TRCV_C veya TCON yordamı için bu parametrenin değerini ID kısmında belirtin.
4	connection_type	USInt	Bağlantı tipi: • 17: TCP (standart) • 18: ISO on TCP • 19: UDP
5	active_est	Bool	Bağlantı türünün ID'si: • TCP ve ISO on TCP: – YANLIŞ: Pasif bağlantı – DOĞRU: Aktif bağlantı (standart) • UDP: YANLIŞ
6	local_device_id	USInt	Yerel PROFINET arayüzünün veya Industrial Ethernet arayüzünün ID'si: 1 (standart)
7	local_tsap_id_len	USInt	local_tsap_id parametresinin Bayt cinsinden uzunluğu; olası değerler: • TCP: 0 (aktif, standart) veya 2 (pasif) • ISO on TCP: 2 ila 16 • UDP: 2
8	rem_subnet_id_len	USInt	Bu parametre kullanılmıyor.
9	rem_staddr_len	USInt	Partner uç nokta adresinin Bayt cinsinde uzunluğu: • 0: belirtilmemiş (rem_staddr parametresi ilgili değil) • 4 (standart): rem_staddr parametresindeki geçerli IP adresi (sadece TCP ve ISO on TCP'de)
10	rem_tsap_id_len	USInt	rem_tsap_id parametresinin Bayt cinsinden uzunluğu; olası değerler: • TCP: 0 (pasif) veya 2 (aktif, standart) • ISO on TCP: 2 ila 16 • UDP: 0
11	next_staddr_len	USInt	Bu parametre kullanılmıyor.

Bayt	Parametre ve veri tipi		Açıklama
12 ila 27	local_tsap_id	Array [1..16] of Byte	Bağlantının yerel adresinin bileşenleri: • TCP ve ISO on TCP: yerel port No. (olası değerler: 1 ila 49151; önerilen değerler: 2000...5000): – local_tsap_id[1] = Onaltılık listede port numarasının High Bayt'ı; – local_tsap_id[2] = Onaltılık listede port Low Bayt'ı; – local_tsap_id[3-16] = ilgili değil • ISO on TCP: yerel TSAP-ID'si: – local_tsap_id[1] = B#16#E0; – local_tsap_id[2] = Yerel uç noktaların modül taşıyıcısı ve yuvası (Bit 0 ila 4: Yuva numarası, Bit 5 ila 7: Modül taşıyıcısının numarası); – local_tsap_id[3-16] = TSAP uzantısı, opsiyonel • UDP: Bu parametre kullanılmıyor. Not: local_tsap_id'nn her değerinin CPU dahilinde benzersiz olduğundan emin olun.
28 ila 33	rem_subnet_id	Array [1..6] of USInt	Bu parametre kullanılmıyor.
34 ila 39	rem_staddr	Array [1..6] of USInt	Sadece TCP ve ISO on TCP'de: Partner uç noktanın IP adresi. (Pasif bağlantılarda ilgili değildir.) Örnek: 192.168.002.003 şeklindeki IP adresi, Array'in aşağıdaki elemanlarına kaydedilir: rem_staddr[1] = 192 rem_staddr[2] = 168 rem_staddr[3] = 002 rem_staddr[4] = 003 rem_staddr[5-6] = ilgili değil
40 ila 55	rem_tsap_id	Array [1..16] of Byte	Bağlantının partner adresinin bileşenleri: • TCP: Partner CPU'nun port numarası. Aralık: 1 ila 49151; önerilen değerler: 2000 ila 5000): – rem_tsap_id[1] = Onaltılık listede port numarasının High Bayt'ı; – rem_tsap_id[2] = Onaltılık listede port Low Bayt'ı; – rem_tsap_id[3-16] = ilgili değil • ISO on TCP: Partner TSAP ID'si: – rem_tsap_id[1] = B#16#E0 – rem_tsap_id[2] = Partner uç noktanın modül taşıyıcısı ve yuvası (Bit 0 ila 4: Yuva numarası, Bit 5 ila 7: Modül taşıyıcısının numarası) – rem_tsap_id[3-16] = TSAP uzantısı, opsiyonel • UDP: Bu parametre kullanılmıyor.
56 ila 61	next_staddr	Array [1..6] of Byte	Bu parametre kullanılmıyor.
62 ila 63	spare	Word	Rezerve: W#16#0000

TCON_IP_V4

Tablo 11-12 Bağlantı açıklamasının yapısı (TCON_IP_V4): TCP ile kullanım için

Bayt	Parametre ve veri tipi		Açıklama
0 ila 1	InterfaceId	HW_ANY	IE arayüzü alt modülünün donanım kodu
2 ila 3	ID	CONN_OUC (Word)	Bu bağlantıya verilen referans: Değer aralığı: 1 (standart) ila 4095. TSEND_C, TRCV_C veya TCON yordamı için bu parametrenin değerini ID kısmında belirtin.
4	ConnectionType	Byte	Bağlantı tipi: • 11: TCP/IP (standart) • 17: TCP/IP (Bu bağlantı tipi, daha eski sistemlerin bağlanması nedeniyle mevcuttur. "11: TCP/IP'nin (standart)" kullanılması önerilir.) • 19: UDP
5	ActiveEstablished	Bool	Aktif/pasif bağlantı kurma: • DOĞRU: Aktif bağlantı (standart) • YANLIŞ: Pasif bağlantı
	V4 IP adresi		
6	ADDR[1]	Byte	Sekizli 1
7	ADDR[1]	Byte	Sekizli 2
8	ADDR[1]	Byte	Sekizli 3
9	ADDR[1]	Byte	Sekizli 4
10 ila 11	RemotePort	UInt	Desantral UDP/TCP port numarası
12 ila 13	LocalPort	UInt	Yerel UDP/TCP port numarası

TCON_IP_V4_SEC

Tablo 11-13 Bağlantı açıklamasının yapısı (TCON_IP_V4_SEC): TCP ile kullanım için

Bayt	Parametre ve veri tipi		Açıklama
0 ila 15	ConnPara	TCON_IP_v4	Bağlantı parametreleri için kullanılan SDT Arayüz koduyla (interface_id) ilgili bilgiler: • Arayüz kodu için ön ayarlı 0 değerini kabul ederseniz, CPU'nun işletim sistemi, kaldırılan IP adresini ve yerel olarak mevcut olan IP güzergahlarını değerlendirir ve ardından CPU'nun, güvenli OUC bağlantısı kurmaya yönelik bir Industrial Ethernet arayüzünü belirtir. Bu durumda teşhis verileri her zaman CPU'nun ilk Industrial Ethernet arayüzüne atanır. • CPU'ya veya bir CP'ye ait bir Industrial Ethernet arayüzünün donanım kodunu arayüz kodu olarak belirtirseniz, güvenli OUC bağlantısı, ilgili Industrial Ethernet arayüzü üzerinden kurulur.
16	ActivateSecure-Conn	Bool	Bu bağlantı için dijital güvenli iletişimin etkinleştirilmesi Bu parametre YANLIŞ değerine (ön ayar) sahipse, sonraki güvenlik parametreleri önemli değildir, bu da bağlantının güvenli olmadığı anlamına gelir. Bu durumda, güvenli olmayan bir TCP veya UDP bağlantısı ayarlayabilirsiniz.
17	TLSServerReq-ClientCert	Bool	Sadece sunucu tarafı için: TLS istemcisinden bir X.509-V3 sertifikasının talep edilmesi. YANLIŞ (standart)

Bayt	Parametre ve veri tipi		Açıklama
18 ila 19	ExtTLSCapabilities	Word	• Bit 0: Sadece istemci tarafı için. Ayarlanmış bir Bit, istemcinin, sunucunun kimliğini kontrol etmek üzere sunucunun X.509-V3 sertifikasındaki sertifika öznesinin alternatif adını (subjectAlternateName) doğruladığı anlamına gelir. Bağlantı kurulduysa sertifikalar kontrol edilir. 16#0 (standart) • Bit 1 ila 15: Gelecekteki yükseltmeler için rezerve edilmiştir
20 ila 23	TLSServerCertRef	UDInt	• Sunucu tarafı: Kendi X.509-V3 sertifikasının ID'si • İstemci tarafı: TLS istemcisi tarafından TLS sunucusunun kimliğini doğrulamak için kullanılan X.509-V3 sertifikasının (genelde bir CA sertifikası) ID'si. Bu parametre 0 ise, TLS istemcisi, sunucunun kimliğini doğrulamak üzere, o anda istemci sertifika belleğine yüklenmiş durumdaki tüm sertifikaları (CA sertifikaları) kullanır. 0 (standart)
24 ila 27	TLSClientCertRef	UDInt	• İstemci tarafı: Kendi X.509-V3 sertifikasının ID'si • Sunucu tarafı: TLS sunucusu tarafından TLS istemcisini doğrulamak için kullanılan X.509-V3 sertifikasının (veya X.509-V3 sertifikalarından oluşan bir grubun) ID'si. Bu parametre 0 ise, TLS sunucusu, istemcinin kimliğini doğrulamak üzere, o anda sunucu sertifika belleğine yüklenmiş durumdaki tüm sertifikaları (CA sertifikaları) kullanır. 0 (standart)

Örnek DB'nin, TCON, TSEND_C ve TRCV_C yordamları için kullanılan CONNECT bağlantı parametresi, kullanılan veri bloğuna bir referans içermektedir.

NOT

IPv4 üzerinden güvenli olmayan TCP veya UDP bağlantıları kurabilirsiniz.

SDT TCON_IP_V4_SEC'i de IPv4 üzerinden, güvenli olmayan bir TCP veya UDP bağlantısı için kullanabilirsiniz.

TCON_IP_RFC

Tablo 11-14 Bağlantı açıklamasının yapısı (TCON_IP_RFC): ISO on TCP ile kullanım için

Bayt	Parametre ve veri tipi		Açıklama
0 ila 1	InterfaceId	HW_ANY	IE arayüzü alt modülünün donanım kodu
2 ila 3	ID	CONN_OUC (Word)	Bu bağlantıya verilen referans: Değer aralığı: 1 (standart) ila 4095. TSEND_C, TRCV_C veya TCON yordamı için bu parametrenin değerini ID kısmında belirtin.
4	ConnectionType	Byte	Bağlantı tipi: • 12: ISO on TCP (standart) • 17: ISO on TCP (Bu bağlantı tipi, daha eski sistemlerin bağlanması nedeniyle mevcuttur. "12: ISO on TCP'nin (standart)" kullanımı için önerilir.)
5	ActiveEstablished	Bool	Aktif/pasif bağlantı kurma: • DOĞRU: Aktif bağlantı (standart) • YANLIŞ: Pasif bağlantı
6 ila 7	Yedek		Kullanılmıyor

Bayt	Parametre ve veri tipi		Açıklama
	V4 IP adresi		
8	ADDR[1]	Byte	Sekizli 1
9	ADDR[1]	Byte	Sekizli 2
10	ADDR[1]	Byte	Sekizli 3
11	ADDR[1]	Byte	Sekizli 4
	Desantral aktarım seçimi		
12 ila 13	TSelLength	UInt	TSelector'un uzunluğu
14 ila 45	TSel	array [1..32] of Byte	TSAP adı için kullanılan karakter alanı
	Yerel aktarım seçimi		
46 ila 47	TSelLength	UInt	TSelector'un uzunluğu
48 ila 79	TSel	array [1..32] of Byte	TSAP adı için kullanılan karakter alanı

TCON_QDN

Tablo 11-15 Bağlantı açıklamasının, TCON_QDN ile uyumlu yapısı

Bayt	Parametre ve veri tipi		Açıklama
0 ila 1	InterfaceId	HW_ANY	Firmware V4.x'ten itibaren S7-1200 CPU'ları: • Takılı bir CP'nin InterfaceId'si: – S7-1200'ün CP'lerinde – ET 200SP'nin CP'lerinde
2 ila 3	ID	CONN_O UC	Bu bağlantıya verilen referans (değer aralığı: 1 ila 4095). TCON yordamı için bu parametrenin değerini ID kısmında belirtin.
4	ConnectionType	BYTE	Bağlantı tipi: • 11: TCP (11 desimal ondalık = 0x0B onaltılı) • 19: UDP (19 desimal = 0x13 onaltılı)
5	ActiveEstablished	BOOL	Bağlantı kurma türünün kodu: • YANLIŞ: Pasif bağlantı kurma • DOĞRU: Aktif bağlantı kurma
6 ila 261	RemoteQDN	STRING'lerden oluşan Array [1..254]	"." ile sonlanması gereken partner uç noktanın tam nitelikli etki alanı adı. Bir SIMATIC ağında adın, sondaki nokta da dahil olmak üzere 254 karakteri aşamayacağına dikkat edin.
262 ila 263	RemotePort	UINT	Kaldırılan bağlantı partnerinin port adresi
264 ila 265	LocalPort	UINT	Yerel bağlantı partnerinin port adresi

TCON_QDN_SEC

Tablo 11-16 TCON_QDN_SEC uyarınca bağlantı açıklamasının yapısı

Bayt	Parametre ve veri tipi		Açıklama
0 ila 271	ConnPara	TCON_QDN	Bağlantı parametreleri
272	ActivateSecureConn	BOOL	Bu bağlantı için dijital güvenli iletişimin etkinleştirilmesi. Bu parametre YANLIŞ değerine sahipse, sonraki güvenlik parametreleri ilgili değildir. Bu durumda, güvenli olmayan bir TCP veya UDP bağlantısı ayarlayabilirsiniz.
273	TLSServerReqClientCert	BOOL	Sadece sunucu tarafında: TLS istemcisinden bir X.509-V3 sertifikasının talep edilmesi.
274 ila 275	ExtTLSCapabilities	WORD	• Bit 0: Sadece istemci tarafı için. Ayarlanmış bir Bit, istemcinin, sunucunun kimliğini kontrol etmek üzere sunucunun X.509-V3 sertifikasındaki sertifikayı düzenleyenin alternatif adını (Subject Alternate Name) doğruladığı anlamına gelir. Bağlantı kurulduysa sertifikalar kontrol edilir. • Bit 1 ila 15: gelecekteki yükseltmeler için rezerve edilmiştir
276 ila 279	TLSServerCertRef	UDINT	• Sunucu tarafında: Kendi X.509-V3 sertifikasının ID'si • İstemci tarafında: TLS istemcisi tarafından TLS sunucusunun kimliğini doğrulamak için kullanılan X.509-V3 sertifikasının (genelde bir CA sertifikası) ID'si. Bu parametre "0" ise, TLS istemcisi, sunucunun kimliğini doğrulamak üzere, o anda istemci sertifika belleğine yüklenmiş durumdaki tüm sertifikaları (CA sertifikaları) kullanır.
280 ila 283	TLSClientCertRef	UDINT	• İstemci tarafında: Kendi X.509-V3 sertifikasının ID'si • Sunucu tarafında: TLS sunucusu tarafından TLS istemcisinin kimliğini doğrulamak için kullanılan X.509-V3 sertifikasının (veya X.509-V3 sertifikalarından oluşan bir grubun) ID'si. Bu parametre 0 ise, TLS sunucusu, istemcinin kimliğini doğrulamak üzere, o anda sunucu sertifika belleğine yüklenmiş durumdaki tüm sertifikaları (CA sertifikaları) kullanır.

11.5.8.7 Desteklenen TLS sürümleri

TLS, veri iletişiminin uygulama katmanında taşıma katmanının güvenliği (Transport Layer Security) anlamına gelir. TLS, S7-1200 CPU ile diğer cihazların arasındaki iletişimde güvenliği ve veri koruma seviyesini arttırır. TLS sürümü, S7-1200 CPU'sunun Firmware sürümüne ve STEP 7 CPU'nun, projenizin cihaz konfigürasyonundaki sürümüne bağlıdır.

STEP 7 projenizdeki CPU sürümünü bulmak için aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda CPU'ya tıklayın.
2. Denetçi penceresindeki "Özellikler" sekmesinde yer alan "Genel" alt bölümüne bakın.
3. Firmware sürümü, katalog bilgileriyle ilgili alt bölümde gösterilir.

TLS'nin en yüksek sürümünü kullanmak için, STEP 7'deki CPU sürümü için CPU'nuzun gerçek Firmware sürümünü yapılandırın. Daha sonra açık kullanıcı iletişiminin partneri, maksimum güvenlik için, desteklenen en yüksek TLS sürümünü kullanabilir.

CPU ve STEP 7 projesinin Firmware sürümü temelinde desteklenen TLS sürümü

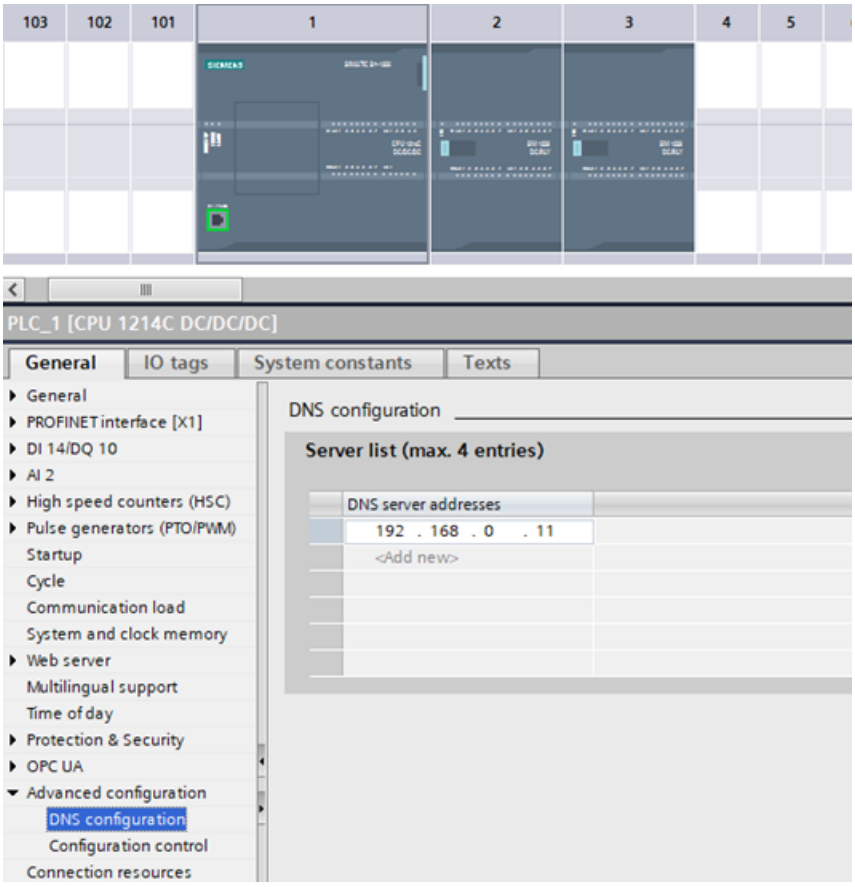
STEP 7 projesinde yapılandırılmış olan Firmware sürümü:	Desteklenen TLS sürümü
V4.5 – V4.6	TLS 1.2, TLS 1.3
V4.3 – V4.4	TLS 1.2

11.5.8.8 Bir DNS'nin yapılandırılması

DNS'yi kullanmak için, ağınızda en azından bir DNS sunucusuna ihtiyacınız bulunmaktadır ve S7-1200 CPU'su için en az bir DNS sunucusu yapılandırmanız gerekir.

Bir DNS sunucusunu aşağıdaki şekilde yapılandırabilirsiniz:

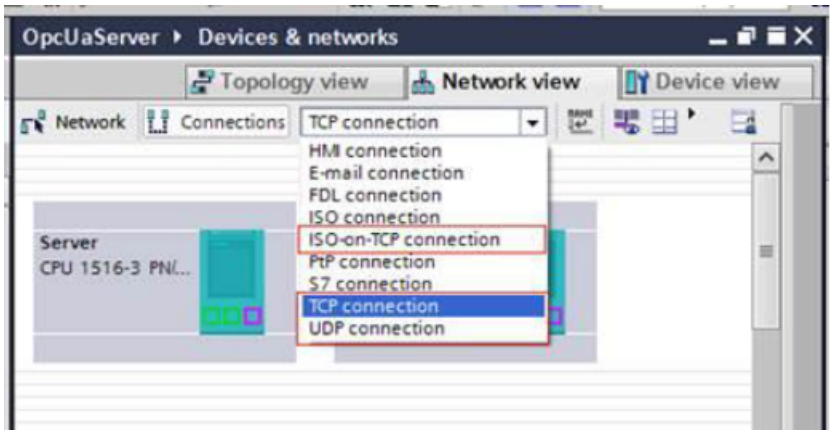
1. S7-1200 CPU'nuz için kullanılan cihaz görünümüne gidin.
2. "Özellikler" sayfasına gidin ve "Genel" sekmesini açın.
3. Konfigürasyon sayfasını görüntülemek için DNS konfigürasyonuna tıklayın.
4. Sunucu listesini içeren tabloda, DNS sunucusu adreslerinin altında ilk satırda yer alan <Yeni ekle> öğesine tıklayın ve DNS sunucunuzun IP adresini girin.



11.5.8.9 TIA Portal'da OUC bağlantısının yapılandırılması

TIA Portal'da, bağlantıyı S7-1200 veya S7-1500 CPU'larını üzerine doğru veya bu CPU'lardan sürüklemek için aşağıdaki Open-User-Communication bağlantılarını seçin (aşağıda gösterildiği gibi).

- ISO on TCP bağlantısı
- TCP bağlantısı
- UDP bağlantısı



Cihazlar arasına bir çizgi çizerseniz, cihazı derlemek ve cihaza yüklemek için bir bağlantı yapılandırılır. Bu bağlantı konfigürasyonu, CPU RUN işletim durumuna geçtiği takdirde S7-1200 Firmware'inin partnerle bir bağlantı kurmasına olanak sağlar. Yapılandırılmış bir bağlantıda TCON yordamının yürütülmesi gerekmez. Yapılandırılmış bir bağlantı için T_DISCON yordamının yürütülmesine de ihtiyaç duyulmaz.

Bu bağlantılar için bir çizgi çizmek için, CPU (veya CP) için kullanılan her iki ağ arayüzü de aynı alt ağda bulunmalıdır. TIA Portal bir bağlantıyı başka ağlardaki cihazların üzerine sürüklemenize engel olmaz. Ancak TIA Portal, derleme veya cihaza yükleme işlemi sırasında bir hata üretir.

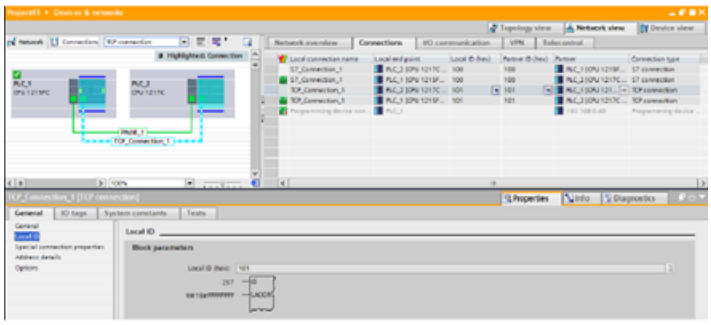
Şimdi, S7-1200 ve S7-1500 CPU'larının arasında sürükleyerek bir OUC bağlantısı kurabilir, konfigürasyonu yükleyebilir ve CPU'ların arasındaki bağlantıyı otomatik olarak oluşturabilirsiniz (bağlantı fiziksel olarak mümkünse).

Yapılandırılmış OUC bağlantıların konfigürasyon opsiyonları

Bağlantının aşağıdaki özelliklerini aşağıda gösterildiği gibi yapılandırabilirsiniz:

- Bağlantı ID'si
- Bağlantının adı
- Aktif bağlantı kurma işlemini gerçekleştiren partner

- Bir bağlantı ucunun seçilmesiyle ağ görünümünün "Özellikler" menüsünde görüntülenen port ayrıntıları

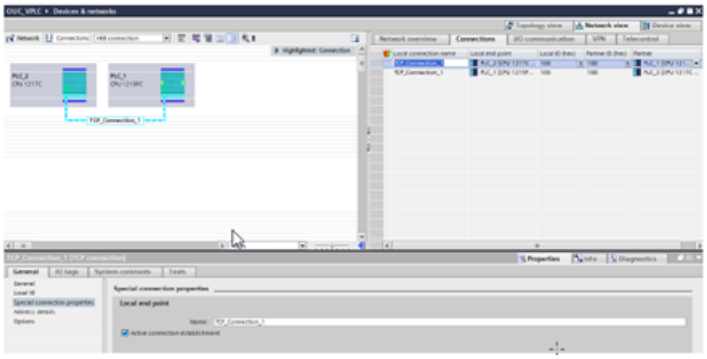


TIA Portal'da bir OUC bağlantısını sürüklerseniz, bir OUC bağlantısı ID'sinin geçerli aralığı içinde bir "Yerel ID" ve bir "Partner ID" atanır. Atanan değeri bağlantı tablosunda veya yerel ID'de değiştirebilirsiniz. Her ID için girilen değer, Tblok yordamları tarafından tanımlanan aralığın içinde olmalıdır (bkz. TSEND).

NOT

Bağlantı ID'si için atanana aralık

Bağlantı ID'si için atanan aralık, bir S7 bağlantı ID'si için atanana ait aynı aralığın içinde bulunur. Bir TSEND yordamına bir S7 bağlantı ID'si girilirse, bu durum uygulama tarafında bir hataya neden olur. Hiçbir OUC bağlantısı kurulmadığı için hata 16#80A1 şeklindedir.



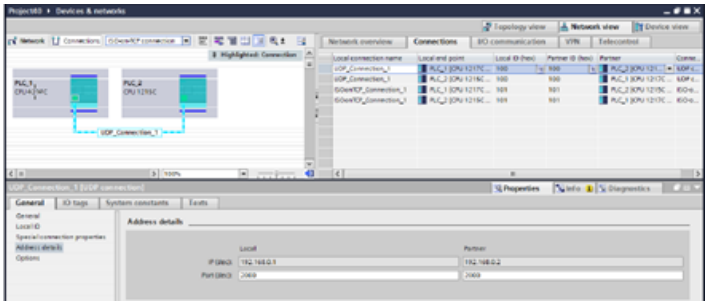
"Özel bağlantı özellikleri" menüsünde "Aktif bağlantı kurma" alanında, konfigürasyon CPU'lara yüklendikten sonra bağlantı mesajları gönderen cihaz tanımlanır. "Aktif bağlantı kurma" opsiyon kutusu etkinleştirilmediyse, yüklenen cihaz, partnerinden bağlantı mesajları bekler. "Aktif bağlantı kurma" opsiyon kutusu etkinleştirilirse TIA Portal, partneri otomatik olarak günceller. Bağlantının sadece bir tarafı aktif taraf olarak belirlenebilir.

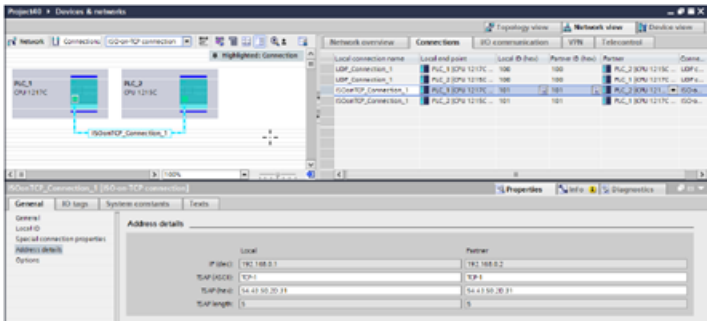
NOT

UDP bağlantıları

UDP bağlantılarında "Aktif bağlantı kurma" opsiyonu mevcut değildir.

"Adres ayrıntıları" özellikler menüsünde, TCP ve ISO on TCP için kullanılan bağlantının iletişiminde kullanılan adreslerin konfigürasyonu görüntülenir. TUSEND ve TURCV yordamlarını bir UDP bağlantısında kullanırsanız, iletişim adresi, yordamlardaki bir parametre tarafından geçersiz kılınır. Ayrıca ISO on TCP bağlantı tiplerinde TSAP'ler "Adres ayrıntıları" özellikler menüsünde aşağıdaki ikinci şekilde gösterildiği gibi değiştirilebilir.





Yapılandırılmış bağlantılarla mevcut Tblok yordamlarının çalışma şekli

Bir bağlantı kurulduktan sonra, yapılandırılmış bağlantı, programlanmış bir bağlantı gibi çalışır. Diğer OUC yordamları, programlanmış bağlantılar aracılığıyla yürütmede olduğu gibi çalışır. Aşağıda, OUC yordamlarıyla ilgili bir açıklama bulacaksınız ve bu yordamlar yapılandırılmış bağlantılar aracılığıyla çalışır.

OUC yordamı	Yapılandırılmış bağlantıyla açıklama
TSEND_C (Sayfa 600)	Connect parametresine bir SDT TCON_Configured aktarılmalıdır. Yapılandırılmış bir bağlantı kullanılıyorsa, TCON_Configured öğesinin ID alanında, yapılandırılmış bağlantı ID'si ve "Bağlantı tipi" alanında 254 değeri ayarlanmalıdır. Yapılandırılmış bir bağlantı kullanılırken, bağlantı zaten kurulmuş olduğu için yordamdaki bağlantı mekanizması atlanmalıdır.
TRCV_C (Sayfa 600)	Connect parametresine bir SDT TCON_Configured aktarılmalıdır. Yapılandırılmış bir bağlantı kullanılıyorsa, TCON_Configured öğesinin ID alanında, yapılandırılmış bağlantı ID'si ve "Bağlantı tipi" alanında 254 değeri ayarlanmalıdır. Yapılandırılmış bir bağlantı kullanılırken, bağlantı zaten kurulmuş olduğu için yordamdaki bağlantı mekanizması atlanmalıdır.
TMAIL_C (Sayfa 644)	Yapılandırılmış bir bağlantıyla çalışmaz
TCON (Sayfa 616)	Bağlantı zaten açık olduğu için yapılandırılmış bir bağlantı için kullanılan bağlantı ID'si belirtilirse bir hata (0x8085) bildirir.
TDISCON (Sayfa 616)	Bir hata (0x80A3) bildirir
TCONSettings (Sayfa 624)	Bir hata (0x8085) bildirir TCONSettings, yapılandırılmış bağlantıları kullanamaz. TCONSettings'in yazma fonksiyonu bir hata döndürür. TCONSettings'in okuma işlemleri, normal kapatma (Sayfa 624) opsiyonunda her zaman YANLIŞ sonucunu döndürür.
TSEND (Sayfa 616)	Çalışma şekli aynı

OUC yordamı	Yapılandırılmış bağlantıyla açıklama
TRCV (Sayfa 616)	Çalışma şekli aynı
TUSEND (Sayfa 664)	Çalışma şekli aynı
TURCV (Sayfa 664)	Çalışma şekli aynı
T_RESET (Sayfa 639)	Çalışma şekli aynıdır. Bağlama işlemi sırasında bağlantı ayrılır ve yeniden bir bağlantı kurulur. Gönderilen Bayt'lar ve alınan Bayt'lar 0 değerine döndürülür.
T_DIAG (Sayfa 640)	Çalışma şekli aynıdır TDIAG_Status varyantının "Tür" alanında, yapılandırılmış bir bağlantı görüntülenmelidir.
T_CONFIG (Sayfa 669)	Çalışma şekli aynıdır (OUC protokolüne göre iletişimle doğrudan bağlantılı değil)
MB_CLIENT (Sayfa 958)	Connect parametresine bir SDT TCON_Configured aktarılmalıdır. Yapılandırılmış bir bağlantı kullanılıyorsa, TCON_Configured öğesinin ID alanında, yapılandırılmış bağlantı ID'si ve "Bağlantı tipi" alanında 254 değeri ayarlanmalıdır. Yapılandırılmış bir bağlantı kullanılırken, bağlantı zaten kurulmuş olduğu için yordamdaki bağlantı mekanizması atlanmalıdır.
MB_SERVER (Sayfa 967)	Connect parametresine bir SDT TCON_Configured aktarılmalıdır. Yapılandırılmış bir bağlantı kullanılıyorsa, TCON_Configured öğesinin ID alanında, yapılandırılmış bağlantı ID'si ve "Bağlantı tipi" alanında 254 değeri ayarlanmalıdır. Yapılandırılmış bir bağlantı kullanılırken, bağlantı zaten kurulmuş olduğu için yordamdaki bağlantı mekanizması atlanmalıdır.

11.5.8.10 TSEND_C ve TRCV_C yordamları

S7-1200 iki seti destekler – TSEND_C yordamları ve TRCV_C yordamları:

- TSEND_C ve TRCV_C yordamları (Sayfa 600): TSEND_C ve TRCV_C yordamları, daha önceki yordamların tüm fonksiyonlarını sunar ve ayrıca TCON_IP_V4, TCON_IP_V4_SEC, TCON_IP_RFC, TCON_QDN, and TCON_QDN_SEC'e uygun yapılara sahip bağlantı parametrelerini kullanma olanağı sağlar.
- Eski TSEND_C ve TRCV_C yordamları (Sayfa 610): TSEND_C ve TRCV_C yordamları, S7-1200'ün V4.1 sürümü tarafından kullanılmıştır ve sadece TCON_Param'a uygun yapılara sahip bağlantı parametreleriyle çalışırlar.

STEP 7, TSEND_C ve TRCV_C yordamlarının çeşitli sürümlerini sağlar. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

TSEND_C yordamı, TCON, TDISCON ve TSEND yordamlarının fonksiyonlarını birleştirir. TRCV_C yordamı, TCON, TDISCON ve TRCV yordamlarının fonksiyonlarını birleştirir. (Bu yordamlarla ilgili daha fazla bilgiyi, "TCON, TDISCON, TSEND VE TRCV (Sayfa 616)" başlığında bulabilirsiniz.)

Gönderilebilen (TSEND_C) veya alınabilen (TRCV_C) minimum veri boyutu bir Bayt'tır; maksimum veri boyutu 8192 Bayt'tır. TSEND_C, Boole adreslerinden veri aktarımını desteklemez ve TRCV_C Boole adreslerindeki verileri almaz. Verilerin bu yordamlarla aktarılmasıyla ilgili daha fazla bilgiyi, Veri tutarlılığı (Sayfa 181) alt bölümünde bulabilirsiniz.

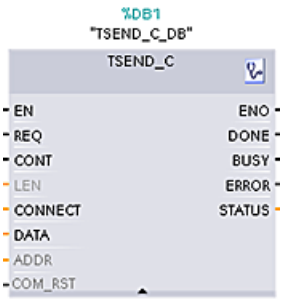
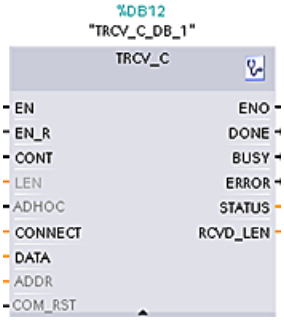
NOT

İletişim parametrelerinin başlatılması

TSEND_C veya TRCV_C yordamını ekledikten sonra, yordamın (Sayfa 563) "Özelliklerinde" iletişim parametrelerini (Sayfa 588) yapılandırabilirsiniz. İletişim partneri için kullanılan parametreleri denetçi penceresinde girerseniz, STEP 7, ilgili verileri yordamın DB'sine girer.

Bir çoklu örnek DB kullanmak istiyorsanız, DB'yi her iki CPU'da da yapılandırmanız gerekir.

Tablo 11-17 TSEND_C ve TRCV_C yordamları

LAD/FBD	SCL	Açıklama
	**"TSEND_C_DB"(req:=_bool_in_, cont:=_bool_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, connect:=_struct_inout_, data:=_variant_inout_, com_rst:=_bool_inout_);**	TSEND_C, bir partnerle bir TCP veya ISO on TCP bağlantısı kurar, verileri gönderir ve ayrıca bağlantıyı sonlandırabilir. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.
	**"TRCV_C_DB"(en_r:=_bool_in_, cont:=_bool_in_, len:=_uint_in_, adhoc:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_uint_out_, connect:=_struct_inout_, data:=_variant_inout_, com_rst:=_bool_inout_);**	TRCV_C, bir CPU ile bir TCP veya ISO on TCP bağlantısı kurar, verileri alır ve ayrıca bağlantıyı sonlandırabilir. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-18 TSEND_C ve TRCV_C parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ (TSEND_C)	IN	Bool	Yükselen bir kenar durumunda gönderme görevini başlatır.
EN_R (TRCV_C)	IN	Bool	Almayı etkinleştirme

Parametre ve veri tipi		Veri tipi	Açıklama
CONT	IN	Bool	İletişim bağlantısına kumanda eder: • 0: İletişim bağlantısını veriler gönderildikten sonra kesme. • 1: İletişim bağlantısı sayısını kurma ve tutma Veri gönderimi sırasında (TSEND_C) (REQ parametresinde yükselen kenar) veya veriler alınırken (TRCV_C) (EN_R parametresinde yükselen kenar), bir bağlantının kurulabilmesi veya tutulabilmesi için CONT parametresi TRUE (DOĞRU) değerine sahip olmalıdır.
LEN	IN	UDInt	Opsiyonel parametre (gizli) Görevle gönderilmesi (TSEND_C) veya alınması (TRCV_C) gereken maksimum Bayt sayısı. DATA parametresinde salt sembolik değerler kullanırsanız, LEN parametresinin 0 değerine sahip olması gerekir.
ADHOC (TRCV_C)	IN	Bool	Opsiyonel parametre (gizli) TCP bağlantı tipi için geçici mod talebi.
CONNECT	IN_OUT	Variant	İşaretçiyi bağlantı açıklamasının üzerine götürme: • TCP veya UDP'de TCON_IP_v4 veya TCON_QDN yapısını kullanabilirsiniz. Açıklama için bkz.: "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)". • Güvenli iletişime sahip TCP'de TCON_IP_V4_SEC veya TCON_QDN_SEC yapısını kullanabilirsiniz. Açıklama için bkz.: "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)". • ISO on TCP'de TCON_IP_RFC yapısını kullanabilirsiniz. Açıklama için bkz.: "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)". • CP 1543-1 / CP 1545-1'in ISO bağlantılarında TCON_ISOnative yapısını kullanabilirsiniz. Açıklama için TIA Portal'daki online yardım bölümüne bakın: "TCON_ISOnative uyarınca bağlantı açıklamasının yapısı". • SMS istemcileriyle kurulan bağlantılarda TCON_PHONE sistem veri tipini kullanabilirsiniz. Açıklama için TIA Portal'daki online yardım bölümüne bakın: "TCON_Phone'a göre bağlantı parametreleri". • CM 1542-5'in FDL bağlantılarında TCON_FDL sistem veri tipini kullanabilirsiniz. Bunun için TIA Portal'daki online yardım bölümüne bakın: "TCON_FDL'ye göre bağlantı parametreleri".
DATA	IN_OUT	Variant	İşaretçiyi, gönderme alanının üzerine aşağıdakilerle götürün. • Gönderilmesi (TSEND_C) gereken verilerin adresi ve uzunluğu • Alınan verilerin (TRCV_C) adresi ve maksimum uzunluğu
ADDR	IN_OUT	Variant	Opsiyonel parametre (gizli) İşaretçiyi, UDP bağlantı tipine sahip alıcıya ait adresin üzerine götürün. Adres bilgileri, TADDR_Param ### yapısında atanır.
COM_RST	IN_OUT	Bool	Opsiyonel parametre (gizli) Yordamı yeniden başlatır: • 0: İlgili değil • 1: Yordamı tümüyle yeniden başlatır. Mevcut bağlantı sonlandırılır veya sıfırlanır ve CONT uyarınca yeniden kurulur. COM_RST parametresi, TSEND_C veya TRCV_C yordamı aracılığıyla değerlendirme sonrasında sıfırlanır ve bu nedenle statik olarak anahtarlanmamalıdır.

Parametre ve veri tipi		Veri tipi	Açıklama
DONE	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Gönderme görevi henüz başlatılmadı veya halen gerçekleştiriliyor. • 1: Gönderme görevi hatasız bir şekilde gerçekleştirildi. Bu durum sadece bir çevrim boyunca görüntülenir.
BUSY	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Gönderme görevi henüz başlatılmadı veya sonlandırıldı. • 1: Gönderme görevi henüz tamamlanmadı. Yeni bir gönderme görevi başlatılamıyor.
ERROR	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Hata yok • 1: Bağlantı kurulurken, veri aktarımı veya bağlantı sonlandırılırken hata oluştu.
STATUS	OUT	Word	Yordamın durumu (ERROR ve STATUS parametrelerinin açıklamasına bakın).
RCVD_LEN (TRCV_C)	OUT	Int	Gerçekten alınan veri miktarı (Bayt cinsinden).

NOT

TSEND_C yordamı, bir gönderme görevini başlatmak için REQ giriş parametresinde 0'dan 1'e geçmeyi gerektirir. Bu durumda BUSY parametresi, işleme sırasında 1'e ayarlanır. Gönderme görevinin tamamlandığı, DONE veya ERROR parametrelerinden biri bir çevrim boyunca 1'e ayarlanarak belirtilir. Bu süre zarfında REQ giriş parametresindeki 0'dan 1'e geçiş göz ardı edilir.

NOT

LEN parametresinin standart ayarı (LEN = 0), DATA parametresinin yardımıyla, aktarılacak verilerin uzunluğunu belirler. TSEND_C yordamı tarafından gönderilen verilerin, TRCV_C yordamının DATA parametresiyle aynı boyuta sahip olması önerilir.

LEN parametresinin standart ayarını kullanırsanız ve verilerin DATA parametresinin boyutundan daha küçük segmentler halinde gönderilmesi gerekiyorsa şunlar geçerlidir. TSEND_C tarafından aktarılan verilerin boyutu, TRCV_C DATA parametresinin boyutuna karşılık gelmiyorsa, TRCV_C, "Dolu" durumunda kalır (durum kodu: TSEND_C tarafından aktarılan verilerin toplam boyutu TRCV_C DATA parametresinin boyutuna karşılık gelene kadar "Dolu" durumunda kalır (durum kodu: 7006).

TRCV_C DATA parametresinin tamponu, yeni alınan verileri ancak veri boyutu, DATA parametresinin tampon boyutuna eşit olduğunda gösterir.

TSEND_C 'nin çalışma şekli

TSEND_C yordamı asenkron olarak yürütülür ve aşağıdaki fonksiyonları sırayla uygular:

1. Bir iletişim bağlantısının ayarlanması ve kurulması:
   TSEND_C, REQ parametresinde bir yükselen kenar algılanırsa ve henüz hiçbir iletişim bağlantısı mevcut değilse bir iletişim bağlantısı ayarlar ve bu bağlantıyı kurar. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir. CONNECT parametresinde belirtilen bağlantı açıklaması, iletişim bağlantısının ayarlanması için kullanılır.
   CPU, STOP işletim durumuna geçerse, mevcut bir bağlantı sonlandırılır ve ayarlanan bağlantı kaldırılır. Bağlantıyı yeniden ayarlamak ve kurmak için TSEND_C yordamını yeniden yürütmeniz gerekir. Olası iletişim bağlantısı sayısıyla ilgili bilgileri, CPU'nuzun teknik veriler kısmında bulabilirsiniz.
2. Verilerin mevcut bir iletişim bağlantısı üzerinden gönderilmesi:
   Gönderme görevi, REQ parametresinde bir yükselen kenar mevcutken yürütülür. Yukarıda açıklandığı gibi, önce iletişim bağlantısı kurulur. Gönderme aralığını, DATA parametresi aracılığıyla belirtebilirsiniz. Bu, adresi ve gönderilecek verilerin uzunluğunu kapsar. DATA parametresinde BOOL veya Array of BOOL veri tipinde bir veri alanı kullanmayın. LEN parametresiyle, bir görevle gönderilen maksimum Bayt sayısını belirtebilirsiniz. DATA parametresinde bir sembolik ad kullanırsanız, LEN parametresinin 0 değerine sahip olması gerekir.
   Gönderilecek veriler, gönderme görevi tamamlanmadan önce düzenlenmemelidir.
3. İletişim bağlantısının sonlandırılması:
   CONT parametresi, REQ parametresinde yükselen kenar ortaya çıktığında 0 değerine sahip idiyse, iletişim bağlantısı, veriler gönderildikten sonra sonlandırılır. Aksi takdirde iletişim bağlantısı korunur.

Gönderme görevi başarıyla gerçekleştirildiğinde DONE parametresi 1 olarak ayarlanır. İletişim bağlantısı, önceden sonlandırılabilir (bunun için CONT parametresine bağlı olmayla ilgili yukarıdaki açıklamaya bakın). DONE parametresindeki sinyal durumu 1, gönderilen verilerin iletişim partneri tarafından okunduğunu göstermez.

COM_RST parametresi 1 olarak ayarlanırsa, TSEND_C sıfırlanır. Bu noktada veriler aktarılırsa veri kaybı oluşabilir.

CONT parametresine bağlı olarak aşağıdaki senaryolar mümkündür:

- CONT = "0":
  Mevcut bir iletişim bağlantısı kuruluyor.
- CONT = 1 değerine sahiptir ve bir iletişim bağlantısı kurulmuştur:
  Mevcut bir iletişim bağlantısı sıfırlanır ve yeniden kurulur.
- CONT = 1 değerine sahiptir ve iletişim bağlantısı kurulmamıştır.
  İletişim bağlantısı kurulmaz.

COM_RST parametresi, değerlendirme sonrasında T_SEND yordamı aracılığıyla sıfırlanır. TSEND_C'yi yürütme sonrasında (DONE = 1) yeniden etkinleştirmek için yordamı REQ = 0 ile bir kez çağırın.

TRCV_C 'nin çalışma şekli

TRCV_C yordamı asenkron olarak yürütülür ve aşağıdaki fonksiyonları sırayla uygular:

1. Bir iletişim bağlantısının ayarlanması ve kurulması:
   EN_R parametresi = 1 ise ve hiçbir iletişim bağlantısı mevcut değilse TRCV_C, bir iletişim bağlantısı ayarlar ve bu bağlantıyı kurar. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.
   CONNECT parametresinde belirtilen bağlantı açıklaması, iletişim bağlantısının ayarlanması için kullanılır. Aşağıdaki bağlantı tipleri kullanılabilir:
   - TCP, ISO on TCP ve UDP protokolleri için kullanılan TCON_Param yapısı
   - TCP / UDP: CONNECT parametresindeki TCON_IP_V4 yapısı aracılığıyla elde edilen bağlantı açıklaması
   - ISO on TCP: CONNECT parametresindeki TCON_IP_RFC yapısı aracılığıyla elde edilen bağlantı açıklaması
   - TCP: CONNECT parametresindeki TCON_IP_V4_SEC yapısı aracılığıyla elde edilen bağlantı açıklaması
   - TCP: TCON_QDN ve TCON_QDN_SEC yapıları aracılığıyla elde edilen bağlantı açıklaması

   CPU, STOP işletim durumuna geçerse, mevcut bir bağlantı sonlandırılır ve ayarlanan bağlantı kaldırılır. Bağlantıyı yeniden ayarlamak ve kurmak için TRCV_C yordamını EN_R = 1 ile yeniden yürütmeniz gerekir.
   EN_R değeri, iletişim bağlantısı oluşturulmadan önce 0'a ayarlanırsa, bağlantı kurulur ve CONT = 0 olsa bile kalır. Ancak hiçbir veri alınmaz (DONE, 0 olarak kalır).
   Olası iletişim bağlantısı sayısıyla ilgili bilgileri, CPU'nuzun teknik veriler kısmında bulabilirsiniz.
2. Verilerin mevcut bir iletişim bağlantısı üzerinden alınması:
   EN_R parametresi 1 değerine ayarlandıysa veri alma etkinleştirilir. Yukarıda açıklandığı gibi, önce iletişim bağlantısı kurulur. Alınan veriler bir alma alanına yazılır. Alma alanının uzunluğunu, kullanılan protokol varyantına bağlı olarak ya LEN parametresiyle (LEN <> 0 ise) ya da DATA parametresinin uzunluk bilgisiyle (LEN = 0 ise) belirtebilirsiniz. DATA parametresinde salt sembolik değerler kullanırsanız, LEN parametresinin 0 değerine sahip olması gerekir.
   Veriler ilk kez alınmadan önce EN_R "0" olarak ayarlanırsa, iletişim bağlantısı CONT = 0 olduğunda da korunur. Ancak hiçbir veri alınmaz (DONE, "0" olarak kalır).
3. İletişim bağlantısının sonlandırılması:
   CONT parametresi, bağlantı kurulurken 0 değerine sahip idiyse, iletişim bağlantısı, veriler alındıktan sonra sonlandırılır. Aksi takdirde iletişim bağlantısı korunur.

Alma görevi başarıyla gerçekleştirildiğinde DONE parametresi 1 olarak ayarlanır. İletişim bağlantısı, önceden sonlandırılabilir (bunun için CONT parametresine bağlı olmayla ilgili yukarıdaki açıklamaya bakın).
COM_RST parametresi ayarlanırsa, TRCV_C sıfırlanır. Yeniden yürütme sırasında veriler alınırsa, bu durum veri kaybına yol açabilir. CONT parametresine bağlı olarak aşağıdaki senaryolar mümkündür:

- CONT = "0":
  Mevcut bir iletişim bağlantısı kuruluyor.
- CONT = 1 değerine sahiptir ve bir iletişim bağlantısı kurulmuştur:
  Mevcut bir iletişim bağlantısı sıfırlanır ve yeniden kurulur.
- CONT = 1 değerine sahiptir ve iletişim bağlantısı kurulmamıştır:
  İletişim bağlantısı kurulmaz.

COM_RST parametresi, değerlendirme sonrasında TRCV_ yordamı aracılığıyla sıfırlanır.

TRCV_C, TRCV yordamıyla aynı alma modlarıyla çalışır. Aşağıdaki tabloda, verilerin alma alanına nasıl yazıldığı gösterilmektedir:

Protokol varyantı	Alma alanındaki verilerin kullanılabilirliği	Bağlantı açıklamasının Connection_type parametresi	LEN parametresi	RCVD_LEN parametresi
TCP (Geçici mod)	Veriler hemen kullanılabilir durumdadır.	B#16#11	TRCV_C yordamının ADHOC girişiyle seçildi	1 ila 1472
TCP (belirtilen uzunlukta veri alma)	LEN parametresinde belirtilen veri uzunluğu tamamen alındığında veriler kullanılabilir.	B#16#11	1 ila 8192	LEN parametresindeki değerle aynıdır
ISO on TCP (protokol kontrollü veri aktarımı)	LEN parametresinde belirtilen veri uzunluğu tamamen alındığında veriler kullanılabilir.	B#16#12	1 ila 8192	LEN parametresindeki değerle aynıdır

NOT

Geçici mod

"Geçici mod" sadece TCP protokol varyantında mevcuttur. TRCV_C yordamını geçici mod için yapılandırmak için, yordamın ADHOC giriş parametresini ayarlayın. Alma alanının uzunluğu, DATA parametresindeki işaretçi tarafından tanımlanır. Gerçekten alınan veri uzunluğu, RCVD_LEN parametresinde yayınlanır. Maksimum 1460 Bayt alınabilir.

NOT

S7-300/400'ün STEP 7 projelerinin geçici mod ile S7-1200'e aktarılması

S7-300/400'ün STEP 7 projelerinde, LEN parametresine 0 değerini atayarak geçici modu seçebilirsiniz. S7-1200'de, TRCV_C yordamını, yordamın ADHOC giriş parametresini ayarlayarak geçici mod için yapılandırabilirsiniz.

S7-300/400'ün, geçici modunu kapsayan bir STEP 7 projesini S7-1200'e aktarıyorsanız, LEN parametresini 65535 olarak değiştirmeniz gerekir.

NOT

TCP (belirtilen uzunlukta veri alma)

LEN parametresinin değeriyle, veri alma için kullanılacak uzunluğu belirtebilirsiniz. LEN parametresinde belirtilen uzunluk tümüyle alındıktan sonra DATA parametresinde belirtilen veriler, alma alanında mevcut olur.

NOT

ISO on TCP (protokol kontrollü veri aktarımı)

ISO on TCP protokol varyantında veriler protokol kontrollü şekilde aktarılır. Alma alanı, LEN ve DATA parametreleri tarafından tanımlanır.

BUSY, DONE ve ERROR parametreleri

NOT

TSEND_C'in asenkron olarak işlenmesi nedeniyle gönderme alanındaki veriler, DONE veya ERROR parametrelerinden biri DOĞRU değerini alana kadar tutarlı şekilde kalmalıdır.

TSEND_C için DONE parametresinin DOĞRU durumu, verilerin başarıyla gönderildiği anlamına gelir. Bu, partner CPU'nun alma tamponunu gerçekten okuduğu anlamına gelmez.

TRCV_C'nin asenkron şekilde işlenmesi nedeniyle, alma alanındaki veriler ancak DONE parametresi = 1 ise tutarlı olur.

Tablo 11-19 TSEND_C veTRCV_C yordamlarının BUSY, DONE ve ERROR parametreleri

BUSY	DONE	ERROR	Açıklama
1	0	0	Gönderme görevi işlenir.
0	1	0	Gönderme işlemi başarıyla gerçekleştirildi.
0	0	1	Bağlantı kurma işlemi veya gönderme görevi hatalarla tamamlandı. Hata sebebi, STATUS parametresine kaydedildi.
0	0	0	Yeni bir gönderme görevi atanmadı.

Yürütme durumunu, BUSY, DONE, ERROR ve STATUS parametreleriyle kontrol edebilirsiniz. BUSY parametresi, düzenleme durumunu gösterir. DONE parametresiyle, bir gönderme görevinin başarıyla gerçekleştirilip gerçekleştirilmediğini kontrol edebilirsiniz. TSEND_C veya TRCV_C yordamı yürütülürken hatalar oluştuysa, ERROR parametresi ayarlanır. Hata bilgileri STATUS parametresinde yayınlanır.

Error ve Status parametreleri

Tablo 11-20 TSEND_C ve TRCV_C'nin ERROR ve STATUS için koşul kodları

ERROR	STATUS * (W#16#...)	Açıklama
0	0000	Gönderme görevi (TSEND_C) veya alma görevi (TRCV_C) hatasız şekilde gerçekleştirildi.
0	0001	İletişim bağlantısı kuruldu.
0	0003	İletişim bağlantısı kapatıldı.
0	7000	Bir gönderme görevinin yürütülmesi aktif değil; iletişim bağlantısı kurulamadı.
0	7001	• Gönderme görevinin (TSEND_C) veya alma görevinin (TRCV_C) yürütülmesinin başlatılması. • Bağlantı kurma. • Bağlantı partnerini bekleme.
0	7002	Görevin yürütülmesi (REQ ilgili değil)
0	7003	Yordam, iletişim bağlantısını sonlandırır.
0	7004	İletişim bağlantısı kuruldu ve denetlendi; hiçbir gönderme görevi (TSEND_C) veya alma görevi (TRCV_C) aktif değil.
0	7005	TSEND_C: Veri transferi devam ediyor.
0	7006	TRCV_C: Yordam, verileri alır.
1	8085	• LEN parametresi, izin verilen en büyük değerden büyüktür. • Yordam, LEN veya DATA parametresindeki değeri ilk çağrıdan sonra değiştirdi.

ERROR	STATUS * (W#16#...)	Açıklama
1	8086	CONNECT parametresindeki ID parametresi, izin verilen aralığın dışında.
1	8087	Maksimum bağlantı sayısına ulaşıldı; başka bağlantı kurmak mümkün değil.
1	8088	LEN parametresindeki değer, DATA parametresinde belirlenen alma alanına karşılık gelmiyor.
1	8089	• CONNECT parametresi bir veri bloğuna referans vermiyor. • CONNECT parametresi, bir bağlantı açıklamasına referans vermiyor. • Manuel olarak oluşturulan bağlantı açıklaması, seçilen bağlantı tipi için yanlış bir yapıya sahip.
1	8091	Maksimum gömme derinliği aşıldı.
1	809A	CONNECT parametresi, bağlantı açıklamasının uzunluğuna karşılık gelmeyen bir alanı gösteriyor.
1	809B	Bağlantı açıklamasındaki InterfaceId, CPU'ya veya CP'ye karşılık gelmiyor.
1	80A1	• Bağlantı veya port kullanılıyor. • İletişim hatası: – Belirtilen bağlantı henüz kurulmadı. – Belirtilen bağlantı sonlandırılır. Bu bağlantıyı aktarmak mümkün değil. – Arayüz yeniden başlatılır.
1	80A2	Yerel port veya çıkartılan port sistem tarafından kullanılıyor. Daha fazla bilgiyi, "TCON ve TDISCON yordamları" (Sayfa 616), "ERROR ve STATUS koşul kodları" başlıklarında bulabilirsiniz.
1	80A3	• Mevcut bir bağlantının yeniden kurulması deneniyor. • Mevcut olmayan bir bağlantının sonlandırılması deneniyor. • İç içe geçirilmiş T_DIAG yordamı, yordamın bağlantıyı kapattığını bildiriyor.
1	80A4	Bağlantının çıkartılan uç noktasının IP adresi geçersiz, yani yerel partnerin IP adresine karşılık geliyor.
1	80A7	İletişim hatası: Gönderme görevi tamamlanmadan önce yordamı COM_RST = 1 ile çağırdınız.
1	80AA	Başka bir blok, aynı bağlantı ID'siyle bir bağlantı kuruyor. Görevi, REQ parametresinde yeni bir yükselen kenarla tekrarlayın.
1	80B3	• UDP protokol varyantı kullanılırken ADDR parametresi veri almıyor. • Bağlantı açıklamasında hata • Başka bir bağlantı açıklaması yerel portu kullanıyor.
1	80B4	Bir bağlantı (ActiveEstablished = FALSE) ISO-on-TCP protokol varyantıyla (ConnectionType = B#16#12) pasif bir şekilde kurulurken şu koşullardan biri veya daha fazlası ihlal edildi: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0
1	80B5	Bağlantı tipi 13 = UDP için sadece bir bağlantının pasif olarak kurulmasına izin veriliyor.
1	80B6	Bağlantı açıklaması için kullanılan veri bloğunun ConnectionType parametresinde parametrelendirme hatası oluştu.
1	80B7	• TCON_Param sistem veri tipinde: Bağlantı açıklaması için kullanılan veri bloğunun şu parametrelerinden birinde hata oluştu: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len. • TCON_IP_V4, TCON_IP_RFC, TCON_IP_V4_SEC, TCON_QDN, ve TCON_QDN_SEC sistem veri tiplerinde: Yordam, partner uç noktasının IP adresini 0.0.0.0 olarak ayarladı.
1	80C3	• Tüm bağlantı kaynakları dolu. • Bu ID'ye sahip bir blok, başka bir öncelik grubunda düzenleniyor.

ERROR	STATUS * (W#16#...)	Açıklama
1	80C4	Geçici iletişim hatası: • Yordam, şu anda bağlantıyı kuramıyor. • Yordam, bağlantı yolundaki güvenlik duvarları gereken portlar için açık olmadığından, bağlantıyı kuramıyor. • Arayüz yeni parametreler alıyor veya yordam, bağlantıyı kuruyor. • "TDISCON (Sayfa 616)" yordamı, yapılandırılan bağlantıyı kaldırıyor. • COM_RST = 1 ile gerçekleştirilen bir çağrı, kullanılan bağlantıyı sonlandırıyor. • Bağlantı partnerinde geçici olarak hiçbir alma kaynağı mevcut değil. Bağlantı partneri almaya hazır değil.
1	80C5	• İletişim partnerinden sağlanan bağlantı sonlandırıldı. • Çıkartılan bağlantı partneri, LSAP'yi serbest bırakmadı.
1	80C6	Ağ hatası: • Yerel cihaz, çıkartılan partnere erişemiyor. • PROFIBUS'ta fiziksel kesinti
1	8722	CONNECT parametresinde hata: Geçersiz kaynak aralık (aralık, veri bloğunda deklare edilmedi).
1	873A	CONNECT parametresinde hata: Bağlantı açıklamasına erişim mümkün değil (veri bloğuna erişim yok).
1	877F	CONNECT parametresinde hata: Dahili hata
1	8822	TSEND_C: DATA parametresi: Geçersiz kaynak aralık, aralık, DB'de mevcut değil.
1	8824	TSEND_C: DATA parametresi: VARIANT işaretçisinde aralık hatası.
1	8832	TSEND_C: DATA parametresi: DB numarası çok büyük.
1	883A	TSEND_C: CONNECT parametresi: Belirtilen bağlantı verilerine erişim mümkün değil (çünkü DB mevcut değil).
1	887F	TSEND_C: DATA parametresi: Dahili hata (örn. VARIANT'a geçersiz referans verildi)
1	893A	TSEND_C: DATA parametresi: Gönderme alanına erişim mümkün değil (çünkü DB mevcut değil).
1	8922	TRCV_C: DATA parametresi: Geçersiz hedef aralık, aralık, DB'de mevcut değil.
1	8924	TRCV_C: DATA parametresi: VARIANT işaretçisinde aralık hatası.
1	8932	TRCV_C: DATA parametresi: DB numarası çok büyük.
1	893A	TRCV_C: CONNECT parametresi: Belirtilen bağlantı verilerine erişim mümkün değil (çünkü DB mevcut değil).
1	897F	TRCV_C: DATA parametresi: Dahili hata (örn. VARIANT'a geçersiz referans verildi).
1	8A3A	TRCV_C: DATA parametresi: Veri alanına erişim yok (örn. çünkü veri bloğu mevcut değil).
* Hata kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir.		

NOT

TCON, TSEND, TRCV ve TDISCON yordamları hata mesajları içeriyor

TSEND_C yordamı, dahili olarak TCON, TSEND ve TDISCON yordamlarını kullanır ve TRCV_C yordamı, TCON, TRCV ve TDISCON yordamlarını kullanır. Bu yordamların hata mesajlarıyla ilgili daha fazla bilgiyi, TCON, TDISCON, TSEND VE TRCV (Sayfa 616) başlığında bulabilirsiniz.

Ethernet bağlantı protokolleri

Her CPU, standart PROFINET iletişimini destekleyen entegre bir PROFINET bağlantısına sahiptir. TSEND_C ve TRCV_C ile TSEND ve TRCV yordamları, TCP ve ISO on TCP Ethernet protokollerinin tümünü destekler.
Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Yerel CPU ile partner CPU arasındaki bağlantı yolunun yapılandırılması (Sayfa 563)".

Ayrıca bakınız

Açık kullanıcı iletişimi için kullanılan yordamların bağlantı ID'leri (Sayfa 585)

11.5.8.11 Daha eski sistemlerdeki TSEND_C ve TRCV_C yordamları

STEP 7 V13 SP1 ve S7-1200 V4.1 CPU'larından önce TSEND_C ve TRCV_C yordamları sadece "TCON_Param"a uygun yapılara sahip bağlantı parametreleriyle çalışıyordu. Her iki yordam türü için de genel konseptler geçerlidir. Programlamayla ilgili bilgileri münferit TSEND_C yordamları ve TRCV_C yordamlarıyla ilgili açıklamalarda bulabilirsiniz.
STEP 7, – ve TSEND_C yordamları ile TRCV_C yordamlarının çeşitli sürümlerini sağlamaktadır. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (Sayfa 616) bölümünde bulabilirsiniz.

Eski TSEND_C yordamı, eski TCON, TDISCON ve TSEND yordamlarının fonksiyonlarını birleştirir. TRCV_C yordamı, TCON, TDISCON ve TRCV yordamlarının fonksiyonlarını birleştirir. (Bu yordamlarla ilgili daha fazla bilgiyi, "Eski TCON, TDISCON, TSEND ve TRCV yordamları (TCP iletişimi) (Sayfa 631)" başlığında bulabilirsiniz.)
Gönderilebilen (TSEND_C) veya alınabilen (TRCV_C) minimum veri boyutu bir Bayt'tır; maksimum veri boyutu 8192 Bayt'tır. TSEND_C, Boole adreslerinden veri aktarımını desteklemez ve TRCV_C Boole adreslerindeki verileri almaz. Verilerin bu yordamlarla aktarılmasıyla ilgili daha fazla bilgiyi, Veri tutarlılığı (Sayfa 181) alt bölümünde bulabilirsiniz.

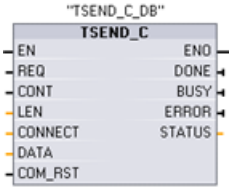
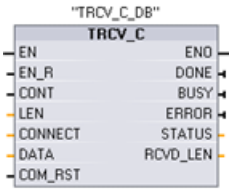
NOT

İletişim parametrelerinin başlatılması

TSEND_C veya TRCV_C yordamını ekledikten sonra, yordamın (Sayfa 563) "Özelliklerinde" iletişim parametrelerini (Sayfa 588) yapılandırabilirsiniz. İletişim partneri için kullanılan parametreleri denetçi penceresinde girerseniz, STEP 7, ilgili verileri yordamın DB'sine girer.

Bir çoklu örnek DB kullanmak istiyorsanız, DB'yi her iki CPU'da da yapılandırmanız gerekir.

Tablo 11-21 TSEND_C ve TRCV_C yordamları

LAD/FBD	SCL	Açıklama
	**"TSEND_C_DB"(req:=_bool_in_, cont:=_bool_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, connect:=_struct_inout_, data:=_variant_inout_, com_rst:=_bool_inout_);**	TSEND_C, bir partnerle bir TCP veya ISO on TCP bağlantısı kurar, verileri gönderir ve ayrıca bağlantıyı sonlandırabilir. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.
	**"TRCV_C_DB"(en_r:=_bool_in_, cont:=_bool_in_, len:=_uint_in_, adhoc:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_uint_out_, connect:=_struct_inout_, data:=_variant_inout_, com_rst:=_bool_inout_);**	TRCV_C, bir CPU ile bir TCP veya ISO on TCP bağlantısı kurar, verileri alır ve ayrıca bağlantıyı sonlandırabilir. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-22 TSEND_C ve TRCV_C parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ (TSEND_C)	IN	Bool	REQ = 1, TSEND_C'nin gönderme görevini CONNECT parametresinde açıklanan bir bağlantı üzerinden yükselen bir kenarda başlatır. İletişim bağlantısını kurmak ve sürdürmek için CONT = 1 de aynı şekilde gereklidir.
EN_R (TRCV_C)	IN	Bool	EN_R = 1 ise, TRCV_C almaya hazırdır. Alma görevi işlenir. İletişim bağlantısını kurmak ve sürdürmek için CONT = 1 de aynı şekilde gereklidir.
CONT	IN	Bool	İletişim bağlantısına kumanda eder: • 0: İletişim bağlantısının sonlandırılması • 1: İletişim bağlantısı sayısını kurma ve tutma Bir bağlantının kurulabilmesi veya tutulabilmesi için, veriler gönderilirken (TSEND_C) (REQ parametresinde yükselen kenar) CONT parametresi, TRUE değerine sahip olmalıdır. Bir bağlantının kurulabilmesi veya tutulabilmesi için, veriler alınırken (TRCV_C) (EN_R parametresinde yükselen kenar) CONT parametresi, TRUE değerine sahip olmalıdır.
LEN	IN	UInt	Gönderilmesi (TSEND_C) veya alınması (TRCV_C) gereken maksimum Bayt sayısı: • Standart = 0: DATA parametresi, gönderilecek (TSEND_C) veya alınacak (TRCV_C) verilerin uzunluğunu belirler. • Geçici mod = 65535: Veri alma için değişken bir uzunluk ayarlanmıştır (TRCV_C).
CONNECT	IN_OUT	TCON_Param	İşaretçiyi bağlantı açıklamasının (Sayfa 588) üzerine götürün

Parametre ve veri tipi		Veri tipi	Açıklama
DATA	IN_OUT	Variant	• Gönderilmesi (TSEND_C) gereken verilerin adresini ve uzunluğunu içerir. • Alınan verilerin (TRCV_C) başlangıç adresini ve maksimum uzunluğunu içerir.
COM_RST	IN_OUT	Bool	Yordamın yeniden başlatılmasına olanak sağlar: • 0: İlgili değil • 1: Fonksiyon bloğunun komple yeniden başlatılması, mevcut bağlantı sonlandırılır.
DONE	OUT	Bool	• 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
BUSY	OUT	Bool	• 0: Görev tamamlandı. • 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz.
ERROR	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Hata yok • 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS	OUT	Word	Durum bilgileri, hata bilgilerini kapsar. (Bkz. alt kısımdaki "Hata ve durum parametreleri".)
RCVD_LEN (TRCV_C)	OUT	Int	Bayt cinsinden gerçekten alınan veri miktarı

NOT

TSEND_C yordamı, bir gönderme görevini başlatmak için REQ giriş parametresinde 0'dan 1'e geçmeyi gerektirir. Bu durumda BUSY parametresi, işleme sırasında 1'e ayarlanır. Gönderme görevinin tamamlandığı, DONE veya ERROR parametrelerinden biri bir çevrim boyunca 1'e ayarlanarak belirtilir. Bu süre zarfında REQ giriş parametresindeki 0'dan 1'e geçiş göz ardı edilir.

NOT

LEN parametresinin standart ayarı (LEN = 0), DATA parametresinin yardımıyla, aktarılacak verilerin uzunluğunu belirler. TSEND_C yordamı tarafından gönderilen verilerin, TRCV_C yordamının DATA parametresiyle aynı boyuta sahip olması önerilir.

LEN parametresinin standart ayarını kullanırsanız ve verilerin DATA parametresinin boyutundan daha küçük segmentler halinde gönderilmesi gerekiyorsa şunlar geçerlidir. TSEND_C tarafından aktarılan verilerin boyutu, TRCV_C yordamının DATA parametresinin boyutuna karşılık gelmiyorsa, TRCV_C, TSEND_C tarafından aktarılan verilerin toplam boyutu, TRCV_C yordamının DATA parametresinin boyutuna karşılık gelene kadar "Dolu" durumunda kalır (durum kodu: 7006).

TRCV_C yordamının DATA parametresinin tamponu, yeni alınan verileri ancak veri boyutu, DATA parametresinin tampon boyutuna eşit olduğunda gösterir.

TSEND_C 'nin çalışma şekli

Aşağıdaki fonksiyonlar, TSEND_C yordamının çalışma şeklini açıklamaktadır:

- Bir bağlantı kurmak için TSEND_C yordamı CONT = 1 ile yürütülmelidir.
- Bağlantı başarıyla kurulduktan sonra TSEND_C yordamı, DONE parametresini bir çevrim için ayarlar.
- Bağlantıyı sonlandırmak için TSEND_C yordamını CONT = 0 ile yürütün. Bu durumda bağlantı hemen iptal edilir. Bu, alıcı için de geçerlidir. Bağlantı sonlandırılır ve alma tamponundaki veriler kaybolabilir.
- Verileri mevcut bir bağlantı üzerinden göndermek için, TSEND_C yordamı, REQ parametresinde bir yükselen kenar mevcutken yürütülmelidir. Gönderme işlemi başarıyla gerçekleştirildikten sonra TSEND_C, DONE parametresini bir çevrim için 1'e ayarlar.
- Bir bağlantı kurmak ve verileri göndermek için TSEND_C yordamı CONT = 1 ve REQ = 1 ile yürütülmelidir. Gönderme işlemi başarıyla gerçekleştirildikten sonra TSEND_C, DONE parametresini bir çevrim için 1'e ayarlar.

TRCV_C 'nin çalışma şekli

Aşağıdaki fonksiyonlar, TRCV_C yordamının çalışma şeklini açıklamaktadır:

- Bir bağlantı kurmak için TRCV_C yordamı CONT = 1 ile yürütülmelidir.
- Verileri almak için, TRCV_C yordamını EN_R parametresi = 1 olacak şekilde yürütün. TRCV_C yordamı, EN_R parametresi = 1 veCONT parametresi = 1 ise verileri sürekli olarak alır.
- Bağlantıyı sonlandırmak için TRCV_C yordamını CONT = 0 ile yürütün. Bağlantı hemen sonlandırılır ve veriler kaybolabilir.

TRCV_C, TRCV yordamıyla aynı alma modlarıyla çalışır. Aşağıdaki tabloda, verilerin alma alanına nasıl yazıldığı gösterilmektedir:

Tablo 11-23 Verilerin alma alanına girilmesi

Protokol varyantı	Verilerin alma alanına girilmesi	Parametre "connection_type"	LEN parametresinin değeri	RCVD_LEN parametresinin değeri (Bayt)
TCP	Geçici mod	B#16#11	65535	1 ila 1472
TCP	Belirtilen uzunlukta veri alma	B#16#11	0 (önerilir) veya 1 ila 8192, 65535 hariç	1 ila 8192
ISO on TCP	Geçici mod	B#16#12	65535	1 ila 1472
ISO on TCP	Protokol kontrollü	B#16#12	0 (önerilir) veya 1 ila 8192, 65535 hariç	1 ila 8192

NOT

Geçici mod

Geçici mod sadece TCP ve ISO on TCP protokol varyantlarında mevcuttur. Geçici modu, LEN parametresine 65535 değerini atayarak ayarlayabilirsiniz. Alma alanı, DATA parametresi tarafından belirtilen alanla aynıdır. Alınan verilerin uzunluğu, RCVD_LEN parametresinde yayınlanır.

Verileri bir "optimize edilmiş" DB'ye (sadece sembolik olarak adreslenebilir) kaydederseniz, sadece Byte, Char, USInt ve SInt veri tiplerindeki Array'lerde bulunan verileri alabilirsiniz.

NOT

S7-300/400'ün STEP 7 projelerinin geçici mod ile S7-1200'e aktarılması

S7-300/400'ün STEP 7 projelerinde, LEN parametresine 0 değerini atayarak geçici modu seçebilirsiniz. S7-1200'de geçici modu, LEN parametresine 65535 değerini atayarak ayarlayabilirsiniz.

S7-300/400'ün, geçici modunu kapsayan bir STEP 7 projesini S7-1200'e aktarıyorsanız, LEN parametresini 65535 olarak değiştirmeniz gerekir.

NOT

Gönderme alanındaki veriler, DONE veya ERROR parametrelerinden biri DOĞRU değerini alana kadar tutarlı şekilde kalmalıdır.

TSEND_C'in asenkron olarak işlenmesi nedeniyle gönderme alanındaki veriler, DONE veya ERROR parametrelerinden biri DOĞRU değerini alana kadar tutarlı şekilde kalmalıdır.

TSEND_C için DONE parametresinin DOĞRU durumu, verilerin başarıyla gönderildiği anlamına gelir. Bu, partner CPU'nun alma tamponunu gerçekten okuduğu anlamına gelmez.

TRCV_C'nin asenkron şekilde işlenmesi nedeniyle, alma alanındaki veriler ancak DONE parametresi = 1 ise tutarlı olur.

Tablo 11-24 TSEND_C veTRCV_C yordamlarının BUSY, DONE ve ERROR parametreleri

BUSY	DONE	ERROR	Açıklama
DOĞRU	İlgili değil	İlgili değil	Görev düzenlenir.
YANLIŞ	DOĞRU	YANLIŞ	Görev başarıyla gerçekleştirildi.
YANLIŞ	YANLIŞ	DOĞRU	Görev hatayla sonlandı. Hata sebebi, STATUS parametresine kaydedildi.
YANLIŞ	YANLIŞ	YANLIŞ	Yeni bir görev atanmaz.

Error ve Status parametreleri

Tablo 11-25 TSEND_C ve TRCV_C'nin ERROR ve STATUS için koşul kodları

ERROR	STATUS	Açıklama
0	0000	Görev hatasız bir şekilde gerçekleştirildi
0	7000	Devam eden bir görev işlemesi yok
0	7001	Görev işlemesini başlatma, bağlantı kurma, bağlantı partnerini bekleme
0	7002	Veriler gönderilir veya alınır
0	7003	Bağlantı sonlandırılır
0	7004	Bağlantı kuruldu, denetleniyor, devam eden bir görev işlemesi
1	8085	LEN parametresi, izin verilen en büyük değerden büyüktür.
1	8086	CONNECT parametresi, izin verilen aralığın dışında.
1	8087	Maksimum bağlantı sayısına ulaşıldı; başka bağlantı kurmak mümkün değil.
1	8088	LEN parametresi, DATA'da belirtilen bellek alanı için geçerli değil.
1	8089	CONNECT parametresi bir veri bloğuna referans vermiyor.
1	8091	Maksimum gömme derinliği aşıldı.

ERROR	STATUS	Açıklama
1	809A	CONNECT parametresi, bağlantı açıklamasının uzunluğuyla eşleşmeyen bir alanı gösteriyor.
1	809B	Bağlantı açıklamasındaki local_device_id, CPU ile eşleşmiyor.
1	80A1	İletişim hatası: • Belirtilen bağlantı henüz kurulmadı • Belirtilen bağlantı şu anda sonlandırılıyor; bu bağlantı üzerinden veri aktarımı mümkün değil • Arayüz yeniden başlatılır
1	80A3	Mevcut olmayan bir bağlantı sonlandırılmaya çalışılıyor
1	80A4	Çıkartılan katılımcının IP adresi geçersiz. Örnek: Çıkartılan katılımcının IP adresi, yerel katılımcının IP adresiyle aynı.
1	80A5	Bağlantı ID'si (Sayfa 585) zaten kullanılıyor.
1	80A7	İletişim hatası: TSEND_C sonlandırılmadan önce TDISCON yordamını çağırdınız.
1	80B2	CONNECT parametresi, kendisi için "Sadece yükleme belleğine kaydet" özniteliğinin ayarlandığı bir veri bloğunu gösterir.
1	80B3	Tutarsız parametreler: • Bağlantı açıklamasında hata • Yerel arayüz (local_tsap_id parametresi) başka bir bağlantı açıklamasında zaten mevcut. • Bağlantı açıklamasındaki ID, parametre olarak belirtilen ID ile eşleşmiyor.
1	80B4	ISO on TCP (connection_type = B#16#12) üzerinden pasif bir bağlantı kurarsanız, girilen TSAP, adresler için aşağıdaki ön koşullardan hiçbirine uygun değilse, 80B4 hata kodu aracılığıyla uyarılırsınız: • Yerel TSAP ilk Bayt için 2 uzunluğuna ve E0 veya E1 şeklinde bir ID değerine (onaltılı) sahipse, ikinci Bayt 00 veya 01 olmalıdır. • Yerel TSAP ilk Bayt için 3 uzunluğuna ve E0 veya E1 şeklinde bir ID değerine (onaltılı) sahipse, ikinci Bayt 00 veya 01 olmalı ve tüm Bayt'lar güncel ASCII karakterleri olmalıdır. • Yerel TSAP, 3 veya daha büyük bir uzunluğa sahipse ve TSAP'nin ID değerinin ilk Bayt'ı E0 veya E1 değerine (onaltılı) eşit değilse, TSAP ID'sinin tüm Bayt'ları güncel ASCII karakterleri olmalıdır. Geçerli ASCII karakterleri, 20 ile 7E (onaltılı) arasındaki Bayt değerleridir.
1	80B7	Aktarılan verilerin veri tipi ve/veya uzunluğu, verilerin yazılması gereken partner CPU'daki aralığa uygun değil.
1	80C3	Tüm bağlantı kaynakları dolu.
1	80C4	Geçici iletişim hatası: • Bağlantı şu anda kurulamıyor • Arayüz yeni parametreler alıyor • Yapılandırılan bağlantı TDISCON tarafından çıkartılıyor.
1	8722	CONNECT parametresi: Geçersiz kaynak aralık: Aralık, DB'de mevcut değil.
1	873A	CONNECT parametresi: Bağlantı açıklamasına erişim mümkün değil (örn. DB kullanılamıyor)
1	877F	CONNECT parametresi: Dahili hata, örn. ANY'ye geçersiz referans verildi.
1	893A	Parametre, yüklenmemiş bir DB'nin numarasını içeriyor.

Ethernet bağlantı protokolleri

Her CPU, standart PROFINET iletişimini destekleyen entegre bir PROFINET bağlantısına sahiptir. TSEND_C ve TRCV_C ile TSEND ve TRCV yordamları, TCP ve ISO on TCP Ethernet protokollerinin tümünü destekler.
Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Yerel CPU ile partner CPU arasındaki bağlantı yolunun yapılandırılması (Sayfa 563)".

11.5.8.12 TCON, TDISCON, TSEND ve TRCV yordamları

S7-1200, TCON, TDISCON, TSEND ve TRCV yordamlarından oluşan iki seti destekler:

- TCON, TDISCON, TSEND ve TRCV yordamları (Sayfa 616): Bu TCON, TDISCON, TSEND ve TRCV yordamları, eski yordamların tüm fonksiyonlarını sağlar ve ayrıca TCON_IP_V4, TCON_IP_V4_SEC, TCON_IP_RFC, TCON_QDN ve TCON_QDN_SEC'e uygun yapılara sahip bağlantı parametrelerini kullanma olanağı sunar.
- Eski TCON, TDISCON, TSEND ve TRCV yordamları (Sayfa 631): TCON, TDISCON, TSEND ve TRCV yordamları, S7-1200'ün V4.1 sürümü tarafından kullanılmıştır ve sadece TCON_Param'a uygun yapılara sahip bağlantı parametreleriyle çalışırlar.

STEP 7; TCON, TDISCON, TSEND ve TRCV yordamlarının çeşitli sürümlerini sağlar. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

TCP ve ISO on TCP protokolleriyle Ethernet iletişimi

NOT
TSEND_C ve TRCV_C yordamları
PROFINET/Ethernet iletişimini programlamayı kolaylaştırmak için TSEND_C ve TRCV_C yordamları, TCON, TDISCON. TSEND ve TRCV yordamlarının fonksiyonlarını birleştirir:

- TSEND_C, TCON, TDISCON ve TSEND yordamlarını birleştirir.
- TRCV_C, TCON, TDISCON ve TRCV yordamlarını birleştirir.

Aşağıdaki yordamlar, iletişim prosesine kumanda eder:

- TCON, istemci ile sunucu PC'si (CPU) arasındaki TCP/IP bağlantısını kurar.
- TSEND ve TRCV, verileri gönderir ve alır.
- TDISCON, bağlantıyı keser.

Gönderilebilen (TSEND) veya alınabilen (TRCV) minimum veri boyutu bir Bayt'tır; maksimum veri boyutu 8192 Bayt'tır. TSEND, Boole adreslerinden veri aktarımını desteklemez ve TRCV Boole adreslerindeki verileri almaz. Bu yordamlarla verilerin aktarılmasıyla ilgili daha fazla bilgi için Veri tutarlılığı (Sayfa 181) alt bölümüne bakın.
TCON, TDISCON, TSEND ve TRCV, bir görevin düzenlenmesi birden fazla yordam yürütme işlemini kapsayacak şekilde asenkron olarak çalışır. Örneğin bir bağlantının ayarlanması ve kurulmasına yönelik bir görevi, TCON yordamını REQ parametresi = 1 olacak şekilde yürüterek başlatabilirsiniz. Ardından görevin yürütme durumunu, TCON yordamının birkaç kez yürütülmesiyle ve DONE parametresinin, görevin tamamlanmasını onaylayıp onaylamadığını kontrol ederek denetleyebilirsiniz.

Aşağıdaki tabloda, BUSY, DONE ile ERROR arasındaki ilişkiler gösterilmektedir. Bir görevin güncel durumunu, tablo aracılığıyla belirleyebilirsiniz:

Tablo 11-26 BUSY, DONE ile ERROR parametreleri arasındaki etkileşimler

BUSY	DONE	ERROR	Açıklama
1	0	0	Görev düzenlenir.
0	1	0	Görev başarıyla gerçekleştirildi.
0	0	1	Görev hatayla sonlandı. Hata sebebi, STATUS parametresinde yayınlanır.
0	0	0	Yeni bir görev atanmaz.

TCON ve TDISCON

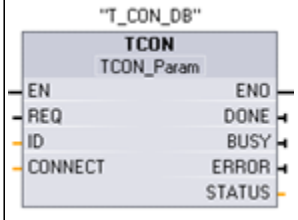
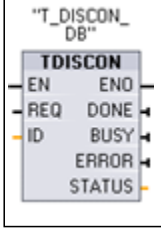
NOT

İletişim parametrelerinin başlatılması

TCON yordamını ekledikten sonra, yordamın (Sayfa 563) "Özellikler" bölümünde iletişim parametrelerini (Sayfa 588) yapılandırabilirsiniz. İletişim partneri için kullanılan parametreleri denetçi penceresinde girerseniz, STEP 7, ilgili verileri, yordamın örnek DB'sine girer.

Bir çoklu örnek DB kullanmak istiyorsanız, DB'yi her iki CPU'da da yapılandırmanız gerekir.

Tablo 11-27 TCON ve TDISCON yordamları

LAD/FBD		Açıklama
	**"TCON_DB"(req:=_bool_in_, ID:=_undef_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, connect:=_struct_inout_);**	TCP ve ISO on TCP: TCON, CPU'dan bir iletişim katılımcısına giden bir iletişim bağlantısını başlatır.
	**"TDISCON_DB"(req:=_bool_in_, ID:=_word_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_);**	TCP ve ISO on TCP: TDISCON, CPU'dan bir iletişim katılımcısına giden bir iletişim bağlantısını sonlandırır.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-28 TCON ve TDISCON parametreleri için kullanılan veri tipleri

Parametre	Deklarasyon	Veri tipi	Açıklama
REQ	IN	Bool	Bir yükselen kenar durumunda ID'de belirtilen bağlantının kurulması için kullanılan görevi başlatır.
ID	IN	CONN_OUC (Word)	Atanan bağlantıya verilen referans. Değer aralığı: W#16#0001 - W#16#0FFF

Parametre	Deklarasyon	Veri tipi	Açıklama
CONNECT (TCON)	IN_OUT	VARIANT	İşaretçiyi bağlantı açıklamasının üzerine götürün: • TCP veya UDP'de TCON_IP_v4 veya TCON_QDN yapısını kullanabilirsiniz. Açıklama için bkz.: "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)". • Güvenli iletişime sahip TCP'de TCON_IP_V4_SEC veya TCON_QDN_SEC yapısını kullanabilirsiniz. Açıklama için bkz.: "PROFINET (Sayfa 588)bağlantısı için kullanılan parametreler (Sayfa 588)". • ISO on TCP'de TCON_IP_RFC yapısını kullanabilirsiniz. Açıklama için bkz.: "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)". • CP 1543-1 / CP 1545-1'in ISO bağlantılarında TCON_ISOnative yapısını kullanabilirsiniz. Açıklama için bkz.: TIA Portal'daki online yardım: "TCON_ISOnative uyarınca bağlantı açıklamasının yapısı". • SMS istemcileriyle kurulan bağlantılarda TCON_PHONE sistem veri tipini kullanabilirsiniz. Açıklama için TIA Portal'daki online yardım bölümüne bakın: "TCON_Phone'a göre bağlantı parametreleri". • CM 1542-5'in FDL bağlantılarında TCON_FDL sistem veri tipini kullanabilirsiniz. Bunun için TIA Portal'daki online yardım bölümüne bakın: "TCON_FDL'ye göre bağlantı parametreleri".
DONE	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Görev henüz başlatılmadı veya halen gerçekleştiriliyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
BUSY	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Görev henüz başlatılmadı veya sonlandırıldı. • 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz.
ERROR	OUT	Bool	ERROR durum parametresi: • 0: Hata yok • 1: Hata oluştu
STATUS	OUT	Word	Yordamın durumu

İletişim bağlantısını ayarlamak ve kurmak için her iki iletişim katılımcısı da TCON yordamını yürütür. Parametrelerle, iletişimin aktif ve pasif uç noktalarını belirtebilirsiniz. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.

Bağlantı bir hat arızası nedeniyle veya çıkartılan katılımcı tarafından iptal edilirse, aktif katılımcı bu bağlantıyı tekrar oluşturmaya çalışır. TCON yordamını tekrar yürütmeniz gerekmez.

TDISCON yordamı yürütülürse veya CPU STOP işletim durumuna geçtiyse, mevcut bağlantı sonlandırılır ve ayarlanan bağlantı silinir. Bağlantıyı ayarlamak ve yeniden kurmak için TCON yordamını yeniden yürütmeniz gerekir.

Tablo 11-29 ERROR ve STATUS 'un TCON ve TDISCON için koşul kodları

ERROR	STATUS * (W#16#...)	Açıklama
0	0000	Bağlantı başarıyla kuruldu.
0	7000	Devam eden bir görev işlemesi yok
0	7001	Görev işlemesini başlatma; bağlantıyı kurma (TCON) veya bağlantıyı sonlandırma (TDISCON).

ERROR	STATUS * (W#16#...)	Açıklama
0	7002	Yordam, bir bağlantı kuru (REQ ilgili değil); bağlantı kurma (TCON) veya bağlantıyı sonlandırma (TDISCON).
1	8085	TCON: Kullanım ID'si kullanılıyor.
1	8086	TCON: ID parametresi, geçerli aralığın dışında.
1	8087	TCON: Maksimum bağlantı sayısına ulaşıldı; başka bağlantı kurmak mümkün değil
1	8089	TCON: CONNECT parametresi, bir bağlantı açıklamasına referans vermiyor veya bağlantı açıklaması manuel olarak oluşturuldu.
1	809A	TCON: Yordam, CONNECT parametresindeki yapıyı desteklemiyor veya uzunluk geçersiz.
1	809B	TCON: • Bağlantı açıklamasındaki InterfaceId elemanı, CPU'ya veya CP'ye karşılık gelmiyor veya "0". • TCON_xxx yapısındaki InterfaceId elemanı, bir CPU veya CM/CP arayüzüne ait bir donanım koduna referans vermiyor.
1	80A1	TCON: TCP/UDP'de: Bağlantı veya port kullanılıyor.
1	80A2	TCON: Sistem, yerel veya çıkartılmış portu kullanıyor. Daha fazla bilgiyi, "Yordamlar için kullanılan ortak parametreler" (Sayfa 679), "Pasif ISO ve TCPiletişimi için kullanılan "Kısıtlı TSAP'ler ve port numaraları" başlıklarında bulabilirsiniz.
1	80A3	TCON: Kullanıcı programı tarafından oluşturulan bir bağlantı (TCON), ID parametresindeki değeri kullanıyor. Bağlantı, aynı ID'yi ve CONNECT parametresinde aynı bağlantı ayarlarını kullanıyor.
1	80A4	TCON: Bağlantının çıkartılan uç noktasının IP adresi geçersiz veya yerel partnerin IP adresine karşılık geliyor.
1	80A7	TCON: İletişim hatası: "TCON" tamamlanmadan önce "TDISCON" yordamını yürüttünüz.
1	80B3	Tutarsız parametrelendirme
1	80B4	TCON: Sadece TCON_IP_RFC ile: Aşağıdaki durumlardan biri gerçekleşti: • Yordam, yerel taşıma seçimini atamadı. • İlk Bayt, 0x0E değerini içermiyor. • Yerel taşıma seçimi "SIMATIC-" ile başlıyor.
1	80B5	TCON: Yordam, bağlantı tipi 13 = UDP için sadece bir bağlantının pasif olarak kurulmasına izin veriyor (TCON_xxx yapısınınActiveEstablished parametresi TRUE değerine sahip).
1	80B6	TCON: Bağlantı açıklaması için kullanılan veri bloğunun ConnectionType parametresinde parametrelendirme hatası oluştu: • Sadece TCON_IP_V4, TCON_IP_V4_SEC, TCON_QDN, TCON_QDN_SEC için geçerli: 0x11, 0x0B ve 0x13 • Sadece TCON_IP_RFC için geçerli: 0x0C ve 0x12
1	80B7	TCON: TCON_IP_V4, TCON_IP_V4_SEC, TCON_QDN, TCON_QDN_SEC'de: • TCP (aktif bağlantı kurma) Çıkartılan port "0"dır. • TCP (pasif bağlantı kurma): Yerel port "0"dır. • UDP: Yerel port "0"dır. • Yordam, partner uç noktasının IP adresini 0.0.0.0 olarak ayarladı. TCON: TCON_IP_RFC'de: • Yordam, yerel (local_tselector) veya çıkartılmış (remote_tselector) taşıma seçimini 32 Bayt'tan fazla bir uzunlukla atadı. • Yordam, TSelLength için taşıma seçimi (yerel veya çıkartılmış) 32'nin üzerinde bir uzunluk atadı. • Belirli bağlantı partnerinin IP adresinde uzunluk hatası • Yordam, partner uç noktasının IP adresini 0.0.0.0 olarak ayarladı.

ERROR	STATUS * (W#16#...)	Açıklama
1	80B8	TCON: Yerel bağlantı açıklamasındaki ID parametresi (CONNECT parametresindeki yapı) ve yordamın ID parametresi farklı.
1	80C3	TCON: Tüm bağlantı kaynakları dolu.
1	80C4	Geçici iletişim hatası: • Yordam, şu anda bağlantıyı kuramıyor (TCON). • Bağlantı yolundaki güvenlik duvarları gereken portlar için açık olmadığı için, yordam, bağlantıyı kuramıyor (TCON). • Arayüz, yeni parametreler (TCON ve TDISCON) alıyor. • "TDISCON" yordamı, yapılandırılan bağlantıyı (TCON) kaldırıyor.
1	80C5	TCON: Çıkartılan partner, aşağıdaki eylemlerden birini gerçekleştirdi: • Bağlantı kurmayı reddetti • Bağlantı sonlandırıldı • Bağlantıyı aktif olarak sonlandırdı
1	80C6	TCON: Yordam, çıkartılan partnere erişemiyor (ağ hatası).
1	80C7	TCON: Yürütme sırasında zaman aşımı
1	80C8	TCON: Kullanıcı programı tarafından oluşturulan bir bağlantı (TCON), ID parametresindeki değeri kullanıyor. Bağlantı, aynı ID'yi kullanıyor, ancak CONNECT parametresindeki bağlantı ayarları farklı.
1	80C9	TCON: Çıkartılan partneri onaylama işlemi başarısız oldu. Bağlantıyı kuracak olan çıkartılan partner, CONNECT parametresinin yapısında belirlenen partnerle uyuşmuyor.
1	80CE	TCON: Yerel arayüzün IP adresi, 0.0.0.0 şeklindedir.
1	80E0	TCON: Yordam, uygun olmayan veya kötü bir mesaj aldı.
1	80E1	TCON: El sıkışmada (Handshake) hata. Olası nedenler: • Kullanıcı tarafından iptal • Güvenlik seviyesi yeterince yüksek değil. • Yordam, yeni anlaşmayı desteklemiyor. • Yordam, SSL/TLS sürümünü desteklemiyor. • Host bilgisayar adını doğrulama işlemi başarısız oldu.
1	80E2	Sertifika desteklenmiyor / sertifika geçersiz / sertifika yok Olası neden: CPU, etkilenen modülde saati ayarlamadı veya modül senkronize edilmedi. Örnek: Modülün tarihi için standart ayar 1/1/2012'dir ve CPU, işletime alma sırasında tarihi ayarlamadı. Sertifikanın geçerlilik süresi 20 Ağustos 2016'da başlıyor ve 20 Ağustos 2024'te bitiyor. Bu durumda modülün tarihi, sertifikanın geçerlilik süresinin dışında kalıyor, dolayısıyla sertifika, modül için geçersiz.
1	80E3	Sertifika iptal edildi.
1	80E4	Geçerli sertifikalandırma kuruluşu bulunamadı.
1	80E5	Sertifikanın süresi doldu.
1	80E6	Transport Layer Security Protocol'da bütünlük hatası
1	80E7	X.509-V3 sertifikasında desteklenmeyen genişletme mevcut
1	80E9	Yordam, sunucu sertifikası olmayan bir TLS sunucusunu desteklemiyor.
1	80EA	Yordam, DTLS protokolünü (UDP) desteklemiyor.
1	80EB	Bir istemci, istemci sertifikası talep edemez.

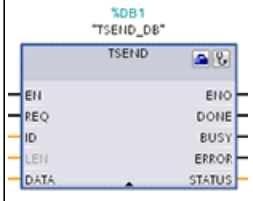
ERROR	STATUS * (W#16#...)	Açıklama
1	80EC	Sunucu, doğrulama işlemini subjectAlternateName'i temel alarak gerçekleştiremiyor (bunu sadece istemciler yapabilir).
1	80ED	TLSServerCertRef_m-ID geçersiz
* Hata kodları, program editöründe tamsayı değerleri veya onaltılık değerler olarak görüntülenebilir.		

TSEND ve TRCV

NOT

PROFINET üzerinden açık kullanıcı iletişiminde, uzak cihazda ilgili TRCV yordamı olmadan bir TSEND yordamını yürütürseniz, TSEND yordamı belirsiz bir süre boyunca "Busy - Meşgul" durumunda kalabilir ve TRCV yordamının verileri almasını bekleyebilir. Bu durumda TSEND yordamının "Busy" çıkışı ayarlanır ve "Status" durumu "0x7002" değerine sahip olur. Bu koşul, 4096 Bayt'tan fazla veri aktardığınız takdirde ortaya çıkabilir. Sorun, TRCV yordamının bir sonraki yürütülüşünde giderilir.

Tablo 11-30 TSEND ve TRCV yordamları

LAD/FBD	SCL	Açıklama
	**"TSEND_DB"(req:=_bool_in_, ID:=_word_in_, len:=_udint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, data:=_variant_inout_);**	TCP ve ISO on TCP: TSEND, verileri bir iletişim bağlantısı üzerinden CPU'dan bir partner istasyona gönderir.
	**"TRCV_DB"(en_r:=_bool_in_, ID:=_word_in_, len:=_udint_in_, adhoc:=_bool_in_, ndr=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_udint_out_, data:=_variant_inout_);**	TCP ve ISO on TCP: TRCV, verileri bir iletişim bağlantısı üzerinden CPU'daki bir partner istasyondan alır.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-31 TSEND ve TRCV parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	TSEND: Yükselen bir kenar durumunda gönderme görevini başlatır. Veriler, DATA ve LEN ile tanımlanan aralıktan aktarılır.
EN_R	IN	Bool	TRCV: CPU'yu alma işlemi için etkinleştirir, EN_R = 1'de TRCV almaya hazırdır. Alma görevi işlenir.

Parametre ve veri tipi		Veri tipi	Açıklama
ID	IN	CONN_OUC (Word)	Bu, ilgili bağlantıya verilen bir referanstır. ID, yerel bağlantı açıklamasındaki ilgili parametre ID'si ile eşleşmelidir. Değer aralığı: W#16#0001 - W#16#0FFF
LEN	IN	UDInt	Gönderilmesi (TSEND) veya alınması (TRCV) gereken maksimum Bayt sayısı: • Standart = 0: DATA parametresi, gönderilecek (TSEND) veya alınacak (TRCV) verilerin uzunluğunu belirler. • Geçici mod = 65535: Veri alma için değişken bir uzunluk ayarlanmıştır (TRCV).
ADHOC	IN	Bool	TRCV: Opsiyonel parametre (gizli) TCP bağlantı tipi için geçici mod talebi.
DATA	IN_OUT	Variant	İşaretçiyi gönderme (TSEND) veya alma (TRCV) için kullanılan veri alanının üzerine götürün. Veri alanı, adresi ve uzunluğu içerir. Adres; E, A veya M bellek alanlarından birini veya bir DB'yi temel alır.
DONE	OUT	Bool	TSEND: • 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
NDR	OUT	Bool	TRCV: • NDR = 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • NDR= 1: Görev, başarıyla gerçekleştirildi.
BUSY	OUT	Bool	• BUSY = 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz. • BUSY = 0: Görev tamamlandı.
ERROR	OUT	Bool	ERROR = 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS	OUT	Word	Durum bilgileri, hata bilgilerini kapsar. (Bkz. alt kısımdaki "Hata ve durum için kullanılan koşul kodları" tablosu.)
RCVD_LEN	OUT	UDInt	TRCV: Bayt cinsinden gerçekten alınan veri miktarı

NOT

TSEND yordamı, bir gönderme görevini başlatmak için REQ giriş parametresinde 0'dan 1'e geçmeyi gerektirir. Bu durumda BUSY parametresi, işleme sırasında 1'e ayarlanır. Gönderme görevinin tamamlandığı, DONE veya ERROR parametrelerinden biri bir çevrim boyunca 1'e ayarlanarak belirtilir. Bu süre zarfında REQ giriş parametresindeki 0'dan 1'e geçiş göz ardı edilir.

TRCV'nin çalışma şekli

TRCV yordamı, alınan verileri, aşağıdaki iki değişken tarafından belirtilen bir alma alanına yazar:

- Aralığın başlangıcını gösterir
- 0 olmadığı sürece LEN girişinde sağlanan aralığın uzunluğu veya değeri

NOT

LEN parametresinin standart ayarı (LEN = 0), DATA parametresinin yardımıyla, aktarılacak verilerin uzunluğunu belirler. TSEND yordamı tarafından gönderilen verilerin, TRCV yordamının DATA parametresiyle aynı boyuta sahip olması önerilir.

LEN parametresinin standart ayarını kullanırsanız ve verilerin DATA parametresinin boyutundan daha küçük segmentler halinde gönderilmesi gerekiyorsa şunlar geçerlidir. İlgili TSEND yordamı, TRCV yordamının DATA parametresini doldurmak için ilgili miktarda veriyi aktarana kadar EN_R Bit'inin 1 değerinde tutulması önerilir. TSEND tarafından aktarılan verilerin boyutu, TRCV yordamının DATA parametresinin boyutuna karşılık gelmiyorsa, TRCV, TSEND tarafından aktarılan verilerin toplam boyutu, EN_R Bit'i = 1 olduğu sürece, TRCVyordamının DATA parametresinin boyutuna karşılık gelene kadar "Dolu" durumunda kalır (durum kodu: 7002). TRCV yordamının EN_R Bit'i darbeler alırsa, verileri almak için TSEND yordamının yürütüldüğü şekilde aynı sayıda darbe almalıdır.

TRCV yordamının DATA parametresinin tamponu, yeni alınan verileri ancak veri boyutu, DATA parametresinin tampon boyutuna eşit olduğunda gösterir.

Tüm görev verileri alındıktan sonra bu, veriler TRCV tarafından alma alanına aktarılır ve NDR 1 olarak ayarlanır.

Tablo 11-32 Verilerin alma alanına girilmesi

Protokol varyantı	Verilerin alma alanına girilmesi	Parametre "connection_type"	LEN parametresinin değeri	RCVD_LEN parametresinin değeri (Bayt)
TCP	Geçici mod	B#16#11	TRCV yordamının ADHOC girişiyle seçildi	1 ila 1472
TCP	Belirtilen uzunlukta veri alma	B#16#11	0 (önerilir) veya 1 ila 8192, 65535 hariç	1 ila 8192
ISO on TCP	Geçici mod	B#16#12	65535	1 ila 1472
ISO on TCP	protokol kontrollü	B#16#12	0 (önerilir) veya 1 ila 8192, 65535 hariç	1 ila 8192

NOT

Geçici mod

Geçici mod sadece TCP ve ISO on TCP protokol varyantlarında mevcuttur. TRCV yordamını geçici mod için yapılandırmak için, yordamın ADHOC giriş parametresini ayarlayın. Alma alanı, DATA parametresi tarafından belirtilen alanla aynıdır. Alınan verilerin uzunluğu, RCVD_LEN parametresinde yayınlanır. TRCV yordamı, bir veri bloğu alındıktan hemen sonra verileri alma alanına yazar ve NDR'yi 1 olarak ayarlar.

Verileri bir "optimize edilmiş" DB'ye (sadece sembolik olarak adreslenebilir) kaydederseniz, sadece Byte, Char, USInt ve SInt veri tiplerindeki Array'lerde bulunan verileri alabilirsiniz.

NOT

S7-300/400'ün STEP 7 projelerinin geçici mod ile S7-1200'e aktarılması

S7-300/400'ün STEP 7 projelerinde, LEN parametresine 0 değerini atayarak geçici modu seçebilirsiniz. S7-1200'de, TRCV yordamını, yordamın ADHOC giriş parametresini ayarlayarak geçici mod için yapılandırabilirsiniz.

S7-300/400'ün, geçici modunu kapsayan bir STEP 7 projesini S7-1200'e aktarıyorsanız, LEN parametresini 65535 olarak değiştirmeniz gerekir.

Tablo 11-33 ERROR ve STATUS 'un TSEND ve TRCV için koşul kodları

ERROR	STATUS	Açıklama
0	0000	• Gönderme görevi hatasız bir şekilde gerçekleştirildi (TSEND) • Yeni veriler kabul edildi: Alınan verilerin güncel uzunluğu, RCVD_LEN 'de (TRCV) gösterilir.
0	7000	• Devam eden bir görev işlemesi yok (TSEND) • Blok, almaya hazır değil (TRCV)
0	7001	• Görev işlemesi başlatılır, veriler gönderilir: Bu düzenleme işlemi sırasında işletim sistemi, DATA gönderme alanındaki verilere erişir (TSEND). • Blok almaya hazır, alma görevi etkinleştirildi (TRCV).
0	7002	• Müteakip çağrı (REQ ilgili değil), görev düzenlenir: Bu düzenleme işlemi sırasında işletim sistemi, DATA gönderme alanındaki verilere erişir (TSEND). • Müteakip çağrı, alma görevi düzenlenir: Bu düzenleme işlemi sırasında veriler alma alanına yazılır. Bu nedenle, bir hata, alma alanında tutarsız verilerin olmasına yol açabilir (TRCV).
1	8085	• LEN parametresi, izin verilen en büyük değerden büyüktür (TSEND) ve (TRCV). • LEN veya DATA parametresi, yordamın ilk yürütülüşünden itibaren değişti (TRCV).
1	8086	ID parametresi izin verilen adres aralığında değil.
1	8088	LEN parametresi, DATA'da belirlenen bellek alanından büyüktür.
1	80A1	İletişim hatası: • Belirtilen bağlantı henüz kurulmadı (TSEND ve TRCV). • Belirtilen bağlantı sonlandırılır. Bu bağlantı üzerinden bir gönderme veya alma görevinin gerçekleştirilmesi mümkün değil (TSEND ve TRCV). • Arayüz yeniden başlatılır (TSEND). • Arayüz yeni parametreler alıyor (TRCV).
1	80C3	Bağlantı kaynaklarında (Sayfa 585) dahili kusur: Bu ID'ye sahip bir blok, başka bir öncelik sınıfında düzenleniyor.
1	80C4	Geçici iletişim hatası: • İletişim partneriyle bağlantı şu anda kurulamıyor. • Arayüz yeni parametrelendirmeler alıyor veya bağlantı şimdi kuruluyor.

Ethernet bağlantı protokolleri

Her CPU, standart PROFINET iletişimini destekleyen entegre bir PROFINET bağlantısına sahiptir. TSEND_C, TRCV_C, TSEND ve TRCV yordamlarının tümü TCP ve ISO on TCP Ethernet protokollerini destekler.

Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Yerel CPU ile partner CPU arasındaki bağlantı yolunun yapılandırılması (Sayfa 563)".

11.5.8.13 TCONSettings

"TCONSettings" yordamıyla aşağıdaki fonksiyonları yürütebilirsiniz:

- Yeni bir OUC bağlantısı için bir bağlantı ID'si talep etme
- Yeni bir OUC bağlantısı için bir bağlantı ID'si talep etme ve aynı anda bu bağlantı için bir özellik belirtme
- Hazırlanmış veya mevcut bir OUC bağlantısının bir özelliğini okuma
- Hazırlanmış veya mevcut bir OUC bağlantısının bir özelliğini yazma

"TCONSettings" yordamıyla aşağıdaki bağlantı özelliklerini okuyabilir veya yazabilirsiniz:

- Bir TCP bağlantısının nasıl sonlandırılacağı

"TCONSettings" yordamı asenkron bir yordamdır. İşleme süreci, birkaç çağrı boyunca gerçekleşir. İşleme, "REQ" parametresindeki bir yükselen kenarla başlar.
"Busy" ve "Done" parametreleri görev durumunu belirtir.
Yürütme sırasında bir hata oluşursa, bu durum, "Error" ve "Status" parametreleri tarafından gösterilir.

Tablo 11-34 TCONSettings'in parametreleri için kullanılan veri tipleri

Parametre ve tip		Veri tipi	Açıklama
REQ	Giriş	Bool	REQUEST kumanda parametresi Yükselen bir kenar durumunda görevi etkinleştirir.
MODE	Giriş	USInt	CPU'dan okumak istediğiniz bilgileri seçmek için "Mode" parametresini kullanır: • 0: Yeni bir OUC bağlantısı için bir bağlantı ID'si talep etme ve gerektiğinde bu bağlantı için bir özelliği yazma (OPTION parametresinde bir özelliğin geçerli bir değeri mevcut ise) • 1: ID aracılığıyla referans verilen OUC bağlantısının bir özelliğinin okunması • 2: ID aracılığıyla referans verilen OUC bağlantısının bir özelliğinin yazılması • 3 ila 255: Rezerve
DONE	Çıkış	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Görev henüz başlatılmadı veya halen gerçekleştiriliyor. • 1: Görev hatasız bir şekilde gerçekleştirildi. Bu durum sadece bir çağrı boyunca görüntülenir.
BUSY	Çıkış	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Görev henüz başlatılmadı veya sonlandırıldı. • 1: Görev henüz tamamlanmadı. Bu örnekle yeni bir görev başlatılamaz.
ERROR	Çıkış	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Hata yok. • 1: Düzenleme sırasında hata oluştu. STATUS, hata türüyle ilgili ayrıntılı bilgiler içerir. Bu durum sadece bir çağrı boyunca görüntülenir.
STATUS	Çıkış	Sözcük	"TCONSettings" yordamının geri dönüş değeri veya hata bilgileri.
ID	InOut	CONN_OUC	Bağlantıya verilen referans: Not: MODE = 0'da ID, bir çıkış parametresidir, MODE = 1'de ve MODE = 2'de ise ID, bir giriş parametresidir.
OPTION	InOut	Varyant	İşaretçiyi, okunacak veya belirtilecek bağlantı özelliğinin açıklamasının üzerine götürün: • TCON_TCPTermination: TCP bağlantısının nasıl sonlandırılması gerektiği.

BUSY, DONE ve ERROR parametreleri

Görevin durumunu BUSY, DONE, ERROR ve STATUS parametreleriyle kontrol edebilirsiniz. BUSY parametresi, düzenleme durumunu gösterir. DONE parametresiyle, bir görevinin başarıyla gerçekleştirilip gerçekleştirilmediğini kontrol edebilirsiniz. ERROR parametresi, "TCONSettings" yürütülürken hatalar oluşursa ayarlanır. Hata bilgileri, STATUS parametresinde yayınlanır.

Aşağıdaki tabloda BUSY, DONE ve ERROR parametrelerinin arasındaki ilişki gösterilmektedir:

BUSY	DONE	ERROR	Açıklama
1	0	0	Görev düzenlenir.
0	1	0	Görev, başarıyla gerçekleştirildi.
0	0	1	Görev hatayla sonlandı. Hata sebebi, STATUS parametresinde yayınlanır.
0	0	0	Yeni bir görev atanmadı.

Tablo 11-35 Durum için TCONSettings'in koşul kodları

STATUS (W#16#...)	Anlamı
0000	"TCONSettings"in yürütülmesi hatasız bir şekilde tamamlandı.
7000	Devam eden bir görev işlemesi yok.
7001	Görev yürütmenin başlangıcı
7002	Ara çağrı (REQ ilgili değil):
8086	ID, izin verilen aralığın dışında.
8087	Maksimum OUC bağlantısı sayısına ulaşıldı; başka bağlantı kurmak mümkün değil
8089	OPTION, geçerli bir veri tipini göstermiyor veya OPTION boş.
8090	OPTION, ID tarafından belirtilen bağlantı için değiştirilmemesi gereken bir bağlantı özelliğini gösteriyor.
8091	MODE için geçersiz değer
8092	OPTION tarafından referanslanan bir veri bloğundaki değere izin verilmez
8093	MODE 0 değerine sahipse, ID de 0 değerine sahip olmalıdır.
809A	OPTION, "TCONSettings" için izin verilmeyen bir veri tipini gösteriyor.
80A3	ID, mevcut olmayan bir iletişim uç noktasını gösteriyor.
80B1	OPTION parametresi, "TCONSettings"in yürütülmesi tamamlanmadan önce değiştirildi. OPTION, "TCONSettings" yürütülürken değiştirilmemelidir.
80C3	Eş zamanlı olarak aktif olan görevlerin maksimum sayısına ulaşıldı.

Eş zamanlı olarak aktif olan görevlerin maksimum sayısı

Eş zamanlı olarak aktif olan görevlerin maksimum sayısı, CPU'nuzun maksimum OUC bağlantısı sayısıyla aynı.

Bağlantı kaynaklarının rezerve edilmesi

TCONSettings'i MODE = 0 ile çağırın. İlgili parametreleri aşağıdaki şekilde atayın:

- ID parametresinde SIFIR değerini girin.
- İlgili bağlantı için hiçbir özelliği belirtmek istemiyorsanız, OPTION parametresini boş bırakın.
  İlgili bağlantı için bir özelliği belirtmek istiyorsanız, OPTION parametresine (Sayfa 624) geçerli bir değer atayın.

TCONSettings'in DONE parametresi DOĞRU olarak değiştikten sonra, parametre ID'sinde yeni bir OUC bağlantısı için bir bağlantı ID'si mevcut olur. OPTION parametresi için bir özellik belirttiyseniz, bağlantı bu özelliği bağlantı için kullanır. TCONSettings yordamı, ID için bir OUC bağlantı kaynağı kullanır ve ilgili teşhis nesnelerini oluşturur. TCONSettings yordamı, bağlantıyı hazırladı, ancak harici iletişim partnerleri henüz bağlantı hakkında hiçbir şey bilmiyor.

Bağlantıyla ilgili hiçbir ayrıntı belirtmediniz, ne bağlantı partnerini, ne protokolü, ne arayüzü, ne de bağlantı açıklamasıyla birlikte DB'yi.

NOT

Bağlantı kurma

TCONSettings, bağlantı kurmuyor.

İlgili bağlantının kurulması

"TCONSettings" başarıyla yürütüldükten sonra ilgili bağlantıyı kurmak istiyorsanız aşağıdaki yöntemi izleyin:

1. "TCONSettings" tarafından önceden belirtilen bağlantı ID'sini kaydedin.
2. "TCON" yordamını tam olarak bu ID ile çağırın.

Kullanılabilir OUC bağlantılarının sayısı değişmez, çünkü TCONSettings yordamı bağlantıyı zaten kullandı.

Bağlantı ID'sinin ve ilgili bağlantı kaynağının etkinleştirilmesi

"TCONSettings" tarafından belirtilen bağlantı ID'sini ve ilgili bağlantı kaynağını yeniden etkinleştirmek istiyorsanız, "TDISCON" yordamını tam olarak bu ID ile çağırın.

CPU, STOP işletim durumuna geçer

CPU STOP işletim durumuna geçerse, "TCONSettings" tarafından belirtilen tüm bağlantı ID'leri ve ilgili bağlantı kaynakları serbest bırakılır.

Hazırlanmış veya mevcut bir bağlantının bir özelliğini okuma

"TCONSettings"i MODE = 1 ile çağırın. İlgili parametreleri aşağıdaki şekilde atayın:

- ID parametresinde, referans istediğiniz bağlantıya verilen referansı belirtin.
- OPTION parametresinde, okumak istediğiniz bağlantı özelliğini belirtin.

DONE parametresi DOĞRU değerini aldıktan sonra, istenen özelliğin güncel değerleri, OPTION tarafından belirtilen veri alanında mevcut olur.

Hazırlanmış veya mevcut bir bağlantının bir özelliğini belirtme

"TCONSettings"i MODE = 2 ile çağırın. İlgili parametreleri aşağıdaki şekilde atayın:

- ID parametresinde, kendisine bir özellik atamak istediğiniz bağlantıya verilen referansı belirtin.
- OPTION parametresinde, önceden tanımlamak istediğiniz bağlantı özelliğini belirtin.

DONE parametresi DOĞRU değerini aldıktan sonra, bağlantıya istenen özellik atanır.

OUC ve Modbus yordamlarından oluşturulan bağlantılar

OUC kütüphanesinin "_C" ile biten yordamları ve MODBUS-TCP kütüphanesinin yordamları, dahili olarak "TCON" yordamının çağrılması yoluyla bağlantılar kurar. Bu tür bağlantıları, "TCON"u doğrudan çağırarak oluşturduğunuz bağlantılar gibi "TCONSettings" ile değiştirebilirsiniz.

Genel olarak okunabilen veya belirtilebilen bağlantı özellikleri

"TCONSettings" yordamıyla aşağıdaki bağlantı özellikleri okunabilir veya belirtilebilir:

- Bir TCP bağlantısının nasıl sonlandırılacağı

Protokol veya arayüz ile gerçekten okunabilen veya belirtilebilen bağlantı özellikleri arasındaki ilişki

Yukarıda belirtilen bağlantı özelliklerini her protokol ve her arayüz okuyamaz veya belirtemez. Aşağıdaki tabloda, münferit protokoller veya arayüzler için hangi bağlantı özelliklerinin kullanılmasının mümkün olduğu gösterilmektedir.

Protokol /arayüz	Bağlantıyı sonlandırma
Programlanan bağlantı:	Evet
Yapılandırılan bağlantı:	Hayır [1]
TCP	Evet
UDP	Hayır [3]
ISO on TCP	Hayır
CPU arayüzü	Evet
CP arayüzü	Hayır
Sanal CP arayüzü	Evet

[1] "TDISCON" yordamını yapılandırılmış bir bağlantı için çağıramazsınız. Bu nedenle, bağlantıyı düzgün bir şekilde sonlandırmak için bir olanak mevcut değildir.
[2] UDP, protokol seviyesinde bağlantısızdır, bu nedenle sonlandırma gerekmez.

Bağlantı özellikleri belirtilirken oluşan çakışmalar

Önceden tanımlanabilen her bağlantı özelliğine, sadece belirli protokoller veya arayüzler için izin verilir. Bu nedenle, bir bağlantı özelliğiyle ilgili olarak sahip olduğunuz bilgilerle, istenen protokol veya istenen arayüz arasında çakışmalar oluşabilir. Bu durumda "TCONSettings", STATUS parametresinde W#16#8090 değerini döndürür.

Bir TCP bağlantısı nasıl sonlandırılabilir?

Mevcut bir TCP bağlantısını şu iki yöntemden biriyle sonlandırabilirsiniz:

- TCP-Reset ile (standart)
  Telegram, başlık satırında ayarlanan RST Bit'i ile birlikte gönderildikten sonra bağlantı kapanır. İlgili kaynaklar derhal silinir ve etkinleştirilir. Geri kalan veriler gönderilir veya kullanıcı programına aktarılır.
- TCP-Finish ile:
  Bir bağlantıyı sonlandırmak için TCP-Finish'i ayarladıysanız ve ardından "TDISCON" yordamını çağırırsanız, bağlantı, "TDISCON" yordamı DONE = DOĞRU ile sonlandırıldıktan sonra, kullanıcının bakış açısından bakıldığında kapanır, yani bağlantı tekrar kullanılabilir duruma gelir. Ancak modülün TCP/IP yığınındaki alt katmanlarda, kaynaklar, bağlantının teşhis nesneleri gibi bir süre daha atanmış durumda kalır.
  Kaynakları etkinleştirmek için tanımlanan süre dolmadan önce çok sayıda bağlantıyı TCP-Finish ile kaldırırsanız ve bağlantıları rezerve ederseniz ("TCONSettings" ile) veya kurarsanız ("TCON" ile), bu durum bir kaynak darboğazına neden olabilir.

TCP-Finish için tanımlanan koşullar

Bir bağlantıyı TCP-Finish ile düzgün bir şekilde sonlandırabilmeniz için aşağıdaki koşullar sağlanmış olmalıdır:

- Kullanılan protokol TCP'dir.
- İlgili arayüz, CPU'da bulunmaktadır.
- Bağlantıyı sonlandırma nedeni, "TDISCON" yordamının çağrılmasıdır.

NOT

TCP bağlantısının STOP işletim durumuna geçişte sonlandırılması

STOP işletim durumuna geçilirken bir TCP bağlantısı her zaman bir TCP-Reset yordamı ile sonlandırılır.

Bağlantıları sonlandırmak için kullanılan SDT: TCON_TCPTermination

Bağlantıları sonlandırmak için kullanılan SDT şu yapıya sahiptir:

Parametre	Veri tipi	Başlangıç değeri	Açıklama
GracefulShutdown	Bool	YANLIŞ	• YANLIŞ: Bağlantıyı sonlandırmak için TCP-Reset yordamını kullanın. • DOĞRU: Bağlantıyı sonlandırmak için TCP-Finish yordamını kullanın.

11.5.8.14 Daha eski TCON, TDISCON, TSEND ve TRCV yordamları

STEP 7 V13 SP1 ve S7-1200 V4.1 CPU'larından önce TCON, TDISCON, TSEND ve TRCV yordamları sadece "TCON_Param"a uygun yapılara sahip bağlantı parametreleriyle çalışıyordu. Her iki yordam türü için de genel konseptler geçerlidir. Programlamayla ilgili bilgileri münferit eski TCON, TDISCON, TSEND ve TRCV yordamlarıyla ilgili açıklamalarda bulabilirsiniz.
STEP 7; TCON, TDISCON, TSEND ve TRCV yordamlarının çeşitli sürümlerini sağlar. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

TCP ve ISO on TCP protokolleriyle Ethernet iletişimi

NOT

TSEND_C ve TRCV_C yordamları

PROFINET/Ethernet iletişimini programlamayı kolaylaştırmak için TSEND_C ve TRCV_C yordamları, TCON, TDISCON. TSEND ve TRCV yordamlarının fonksiyonlarını birleştirir:

- TSEND_C, TCON, TDISCON ve TSEND yordamlarını birleştirir.
- TRCV_C, TCON, TDISCON ve TRCV yordamlarını birleştirir.

Aşağıdaki yordamlar, iletişim prosesine kumanda eder:

- TCON, istemci ile sunucu PC'si (CPU) arasındaki TCP/IP bağlantısını kurar.
- TSEND ve TRCV, verileri gönderir ve alır.
- TDISCON, bağlantıyı keser.

Gönderilebilen (TSEND) veya alınabilen (TRCV) minimum veri boyutu bir Bayt'tır; maksimum veri boyutu 8192 Bayt'tır. TSEND, Boole adreslerinden veri aktarımını desteklemez ve TRCV Boole adreslerindeki verileri almaz. Bu yordamlarla verilerin aktarılmasıyla ilgili daha fazla bilgi için Veri tutarlılığı (Sayfa 181) alt bölümüne bakın.
TCON, TDISCON, TSEND ve TRCV, bir görevin düzenlenmesi birden fazla yordam yürütme işlemini kapsayacak şekilde asenkron olarak çalışır. Örneğin bir bağlantının ayarlanması ve kurulmasına yönelik bir görevi, TCON yordamını REQ parametresi = 1 olacak şekilde yürüterek başlatabilirsiniz. Ardından görevin yürütme durumunu, TCON yordamının birkaç kez yürütülmesiyle ve DONE parametresinin, görevin tamamlanmasını onaylayıp onaylamadığını kontrol ederek denetleyebilirsiniz.
Aşağıdaki tabloda, BUSY, DONE ile ERROR arasındaki ilişkiler gösterilmektedir. Bir görevin güncel durumunu, tablo aracılığıyla belirleyebilirsiniz:

Tablo 11-36 BUSY, DONE ile ERROR parametreleri arasındaki etkileşimler

BUSY	DONE	ERROR	Açıklama
DOĞRU	İlgili değil	İlgili değil	Görev düzenlenir.
YANLIŞ	DOĞRU	YANLIŞ	Görev başarıyla gerçekleştirildi.
YANLIŞ	YANLIŞ	DOĞRU	Görev hatayla sonlandı. Hata sebebi, STATUS parametresine kaydedildi.
YANLIŞ	YANLIŞ	YANLIŞ	Yeni bir görev atanmaz.

TCON ve TDISCON

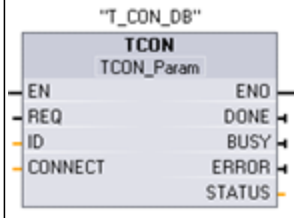
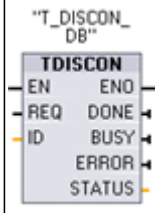
NOT

İletişim parametrelerinin başlatılması

TCON yordamını ekledikten sonra, yordamın (Sayfa 563) "Özellikler" bölümünde iletişim parametrelerini (Sayfa 588) yapılandırabilirsiniz. İletişim partneri için kullanılan parametreleri denetçi penceresinde girerseniz, STEP 7, ilgili verileri, yordamın örnek DB'sine girer.

Bir çoklu örnek DB kullanmak istiyorsanız, DB'yi her iki CPU'da da yapılandırmanız gerekir.

Tablo 11-37 TCON ve TDISCON yordamları

LAD/FBD		Açıklama
	**"TCON_DB"(req:=_bool_in_, ID:=_undef_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, connect:=_struct_inout_);**	TCP ve ISO on TCP: TCON, CPU'dan bir iletişim katılımcısına giden bir iletişim bağlantısını başlatır.
	**"TDISCON_DB"(req:=_bool_in_, ID:=_word_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_);**	TCP ve ISO on TCP: TDISCON, CPU'dan bir iletişim katılımcısına giden bir iletişim bağlantısını sonlandırır.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-38 TCON ve TDISCON parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQ kumanda parametresi, ID tarafından belirtilen bağlantıyı kurarak görevi başlatır. Görev, bir yükselen kenarla başlar.
ID	IN	CONN_OUC (Word)	Çıkartılan katılımcıyla veya kullanıcı programı ile işletim sisteminin iletişim katmanı arasında oluşturulması (TCON) veya sonlandırılması (TDISCON) gereken bağlantıya verilen referans. Bu ID, yerel bağlantı açıklamasındaki ilişkili parametre ID'si ile eşleşmelidir. Değer aralığı: W#16#0001 - W#16#0FFF
CONNECT (TCON)	IN_OUT	TCON_Param	İşaretçiyi bağlantı açıklamasının (Sayfa 588) üzerine götürme
DONE	OUT	Bool	• 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.

Parametre ve veri tipi		Veri tipi	Açıklama
BUSY	OUT	Bool	• 0: Görev tamamlandı. • 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz.
ERROR	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Hata yok • 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS	OUT	Word	Durum bilgileri, hata bilgilerini kapsar. (Bkz. alt kısımdaki "HATA ve DURUM için kullanılan koşul kodları" tablosu.)

İletişim bağlantısını ayarlamak ve kurmak için her iki iletişim katılımcısı da TCON yordamını yürütür. Parametrelerle, iletişimin aktif ve pasif uç noktalarını belirtebilirsiniz. Bağlantı ayarlandıktan ve kurulduktan sonra bu bağlantı otomatik olarak CPU tarafından tutulur ve denetlenir.

Bağlantı bir hat arızası nedeniyle veya çıkartılan katılımcı tarafından iptal edilirse, aktif katılımcı bu bağlantıyı tekrar oluşturmaya çalışır. TCON yordamını tekrar yürütmeniz gerekmez.

TDISCON yordamı yürütülürse veya CPU STOP işletim durumuna geçtiyse, mevcut bağlantı sonlandırılır ve ayarlanan bağlantı silinir. Bağlantıyı ayarlamak ve yeniden kurmak için TCON yordamını yeniden yürütmeniz gerekir.

Tablo 11-39 ERROR ve STATUS 'un TCON ve TDISCON için koşul kodları

ERROR	STATUS	Açıklama
0	0000	Bağlantı başarıyla kuruldu.
0	7000	Devam eden bir görev işlemesi yok
0	7001	Görev işlemesini başlatma, bağlantıyı kurma (TCON) veya bağlantıyı sonlandırma (TDISCON)
0	7002	Müteakip çağrı (REQ ilgili değil), bağlantıyı kurma (TCON) veya bağlantıyı sonlandırma (TDISCON)
1	8086	ID parametresi izin verilen adres aralığının dışındadır.
1	8087	TCON: Maksimum bağlantı sayısına ulaşıldı; başka bağlantı kurmak mümkün değil.
1	809B	TCON: Bağlantı açıklamasındaki local_device_id, CPU ile eşleşmiyor.
1	80A1	TCON: Bağlantı veya port kullanıcı tarafından kullanılıyor.
1	80A2	TCON: Yerel port veya çıkartılan port sistem tarafından kullanılıyor.
1	80A3	Mevcut bir bağlantı yeniden kurulmaya (TCON) veya mevcut olmayan bir bağlantı sonlandırılmaya çalışılıyor (TDISCON).
1	80A4	TCON: Çıkartılan bağlantı noktasının IP adresi geçersiz, yerel IP adresine karşılık geliyor.
1	80A5	TCON: Bağlantı ID'si (Sayfa 585) zaten kullanılıyor.
1	80A7	TCON: İletişim hatası: TCON tamamlanmadan önce TDISCON yordamını yürüttünüz. TDISCON, ilk önce, ID tarafından belirtilen bağlantıyı tamamen sonlandırmalıdır.
1	80B2	TCON: CONNECT parametresi, "Sadece yükleme belleğine kaydet" özniteliğiyle oluşturulan bir veri bloğunu gösteriyor.

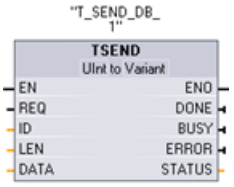
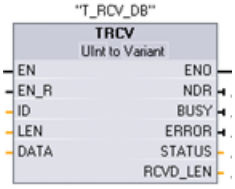
ERROR	STATUS	Açıklama
1	80B4	TCON: ISO on TCP (connection_type = B#16#12) üzerinden pasif bir bağlantı kurarsanız, girilen TSAP, adresler için aşağıdaki ön koşullardan hiçbirine uygun değilse, 80B4 hata kodu aracılığıyla uyarılırsınız: • Yerel TSAP ilk Bayt için 2 uzunluğuna ve E0 veya E1 şeklinde bir ID değerine (onaltılı) sahipse, ikinci Bayt 00 veya 01 olmalıdır. • Yerel TSAP ilk Bayt için 3 uzunluğuna ve E0 veya E1 şeklinde bir ID değerine (onaltılı) sahipse, ikinci Bayt 00 veya 01 olmalı ve tüm Bayt'lar güncel ASCII karakterleri olmalıdır. • Yerel TSAP, 3 veya daha büyük bir uzunluğa sahipse ve TSAP'nin ID değerinin ilk Bayt'ı E0 veya E1 değerine (onaltılı) eşit değilse, TSAP ID'sinin tüm Bayt'ları güncel ASCII karakterleri olmalıdır. Geçerli ASCII karakterleri, 20 ile 7E (onaltılı) arasındaki Bayt değerleridir.
1	80B5	TCON: Bağlantı tipi "13 = UDP" sadece pasif bağlantı kurmaya izin veriyor.
1	80B6	TCON: TCON_Param SDT'sinin CONNECTION_TYPE parametresinde parametrelendirme hatası.
1	80B7	TCON: Bağlantı açıklaması için kullanılan veri bloğunun şu parametrelerinden birinde hata oluştu: • block_length • local_tsap_id_len • rem_subnet_id_len • rem_staddr_len • rem_tsap_id_len • next_staddr_len Not: TCON pasif TCP modunda çalıştırılırken LOCAL_TSAP_ID_LEN "2" Ve REM_TSAP_ID_LEN "0" olmalıdır.
1	80B8	TCON: Yerel bağlantı açıklamasının parametreleri ve parametre ID'si farklı.
1	80C3	TCON: Tüm bağlantı kaynakları dolu.
1	80C4	Geçici iletişim hatası: • Bağlantı şu anda kurulamıyor (TCON). • Yapılandırılan bağlantı TDISCON (TCON) tarafından çıkartılıyor. • Belirtilen bağlantı şu anda kuruluyor (TDISCON). • Arayüz, yeni parametreler (TCON ve TDISCON) alıyor.

TSEND ve TRCV

NOT

PROFINET üzerinden açık kullanıcı iletişiminde, uzak cihazda ilgili TRCV yordamı olmadan bir TSEND yordamını yürütürseniz, TSEND yordamı belirsiz bir süre boyunca "Busy - Meşgul" durumunda kalabilir ve TRCV yordamının verileri almasını bekleyebilir. Bu durumda TSEND yordamının "Busy" çıkışı ayarlanır ve "Status" durumu "0x7002" değerine sahip olur. Bu koşul, 4096 Bayt'tan fazla veri aktardığınız takdirde ortaya çıkabilir. Sorun, TRCV yordamının bir sonraki yürütülüşünde giderilir.

Tablo 11-40 TSEND ve TRCV yordamları

LAD/FBD	SCL	Açıklama
	**"TSEND_DB"(req:=_bool_in_, ID:=_word_in_, len:=_udint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, data:=_variant_inout_);**	TCP ve ISO on TCP: TSEND, verileri bir iletişim bağlantısı üzerinden CPU'dan bir partner istasyona gönderir.
	**"TRCV_DB"(en_r:=_bool_in_, ID:=_word_in_, len:=_udint_in_, ndr=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_udint_out_, data:=_variant_inout_);**	TCP ve ISO on TCP: TRCV, verileri bir iletişim bağlantısı üzerinden CPU'daki bir partner istasyondan alır.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-41 TSEND ve TRCV parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	TSEND: Yükselen bir kenar durumunda gönderme görevini başlatır. Veriler, DATA ve LEN ile tanımlanan aralıktan aktarılır.
EN_R	IN	Bool	TRCV: CPU'yu alma işlemi için etkinleştirir, EN_R = 1'de TRCV almaya hazırdır. Alma görevi işlenir.
ID	IN	CONN_OUC (Word)	Bu, ilgili bağlantıya verilen bir referanstır. ID, yerel bağlantı açıklamasındaki ilgili parametre ID'si ile eşleşmelidir. Değer aralığı: W#16#0001 - W#16#0FFF
LEN	IN	UInt	Gönderilmesi (TSEND) veya alınması (TRCV) gereken maksimum Bayt sayısı: • Standart = 0: DATA parametresi, gönderilecek (TSEND) veya alınacak (TRCV) verilerin uzunluğunu belirler. • Geçici mod = 65535: Veri alma için değişken bir uzunluk ayarlanmıştır (TRCV).

Parametre ve veri tipi		Veri tipi	Açıklama
DATA	IN_OUT	Varyant	İşaretçiyi gönderme (TSEND) veya alma (TRCV) için kullanılan veri alanının üzerine götürün. Veri alanı, adresi ve uzunluğu içerir. Adres; E, A veya M bellek alanlarından birini veya bir DB'yi temel alır.
DONE	OUT	Bool	TSEND: • 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
NDR	OUT	Bool	TRCV: • NDR = 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • NDR= 1: Görev, başarıyla gerçekleştirildi.
BUSY	OUT	Bool	• BUSY = 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz. • BUSY = 0: Görev tamamlandı.
ERROR	OUT	Bool	ERROR = 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS	OUT	Word	Durum bilgileri, hata bilgilerini kapsar. (Bkz. alt kısımdaki "Hata ve durum için kullanılan koşul kodları" tablosu.)
RCVD_LEN	OUT	Int	TRCV: Bayt cinsinden gerçekten alınan veri miktarı

NOT

TSEND yordamı, bir gönderme görevini başlatmak için REQ giriş parametresinde 0'dan 1'e geçmeyi gerektirir. Bu durumda BUSY parametresi, işleme sırasında 1'e ayarlanır. Gönderme görevinin tamamlandığı, DONE veya ERROR parametrelerinden biri bir çevrim boyunca 1'e ayarlanarak belirtilir. Bu süre zarfında REQ giriş parametresindeki 0'dan 1'e geçiş göz ardı edilir.

TRCV'nin çalışma şekli

TRCV yordamı, alınan verileri, aşağıdaki iki değişken tarafından belirtilen bir alma alanına yazar:

- Aralığın başlangıcını gösterir
- 0 olmadığı sürece LEN girişinde sağlanan aralığın uzunluğu veya değeri

NOT

LEN parametresinin standart ayarı (LEN = 0), DATA parametresinin yardımıyla, aktarılacak verilerin uzunluğunu belirler. TSEND yordamı tarafından gönderilen verilerin, TRCV yordamının DATA parametresiyle aynı boyuta sahip olması önerilir.

LEN parametresinin standart ayarını kullanırsanız ve verilerin DATA parametresinin boyutundan daha küçük segmentler halinde gönderilmesi gerekiyorsa şunlar geçerlidir. İlgili TSEND yordamı, TRCV yordamının DATA parametresini doldurmak için ilgili miktarda veriyi aktarana kadar EN_R Bit'inin 1 değerinde tutulması önerilir. TSEND tarafından aktarılan verilerin boyutu, TRCV yordamının DATA parametresinin boyutuna karşılık gelmiyorsa, TRCV, TSEND tarafından aktarılan verilerin toplam boyutu, EN_R Bit'i = 1 olduğu sürece, TRCVyordamının DATA parametresinin boyutuna karşılık gelene kadar "Dolu" durumunda kalır (durum kodu: 7002). TRCV yordamının EN_R Bit'i darbeler alırsa, verileri almak için TSEND yordamının yürütüldüğü şekilde aynı sayıda darbe almalıdır.

TRCV yordamının DATA parametresinin tamponu, yeni alınan verileri ancak veri boyutu, DATA parametresinin tampon boyutuna eşit olduğunda gösterir.

Tüm görev verileri alındıktan sonra bu, veriler TRCV tarafından alma alanına aktarılır ve NDR 1 olarak ayarlanır.

Tablo 11-42 Verilerin alma alanına girilmesi

Protokol varyantı	Verilerin alma alanına girilmesi	Parametre "connection_type"	LEN parametresinin değeri	RCVD_LEN parametresinin değeri (Bayt)
TCP	Geçici mod	B#16#11	65535	1 ila 1472
TCP	Belirtilen uzunlukta veri alma	B#16#11	0 (önerilir) veya 1 ila 8192, 65535 hariç	1 ila 8192
ISO on TCP	Geçici mod	B#16#12	65535	1 ila 1472
ISO on TCP	protokol kontrollü	B#16#12	0 (önerilir) veya 1 ila 8192, 65535 hariç	1 ila 8192

NOT

Geçici mod

Geçici mod sadece TCP ve ISO on TCP protokol varyantlarında mevcuttur. Geçici modu, LEN parametresine 65535 değerini atayarak ayarlayabilirsiniz. Alma alanı, DATA parametresi tarafından belirtilen alanla aynıdır. Alınan verilerin uzunluğu, RCVD_LEN parametrelerinde yayınlanır. TRCV yordamı, bir veri bloğu alındıktan hemen sonra verileri alma alanına yazar ve NDR'yi 1 olarak ayarlar.

Verileri bir "optimize edilmiş" DB'ye (sadece sembolik olarak adreslenebilir) kaydederseniz, sadece Byte, Char, USInt ve SInt veri tiplerindeki Array'lerde bulunan verileri alabilirsiniz.

NOT

S7-300/400'ün STEP 7 projelerinin geçici mod ile S7-1200'e aktarılması

S7-300/400'ün STEP 7 projelerinde, LEN parametresine 0 değerini atayarak geçici modu seçebilirsiniz. S7-1200'de geçici modu, LEN parametresine 65535 değerini atayarak ayarlayabilirsiniz.

S7-300/400'ün, geçici modunu kapsayan bir STEP 7 projesini S7-1200'e aktarıyorsanız, LEN parametresini 65535 olarak değiştirmeniz gerekir.

TSEND ve TRCV Error ve Status için koşul kodları

ERROR	STATUS	Açıklama
0	0000	• Gönderme görevi hatasız bir şekilde gerçekleştirildi (TSEND) • Yeni veriler kabul edildi: Alınan verilerin güncel uzunluğu, RCVD_LEN 'de (TRCV) gösterilir.
0	7000	• Devam eden bir görev işlemesi yok (TSEND) • Blok, almaya hazır değil (TRCV)
0	7001	• Görev işlemesi başlatılır, veriler gönderilir: Bu düzenleme işlemi sırasında işletim sistemi, DATA gönderme alanındaki verilere erişir (TSEND). • Blok almaya hazır, alma görevi etkinleştirildi (TRCV).
0	7002	• Müteakip çağrı (REQ ilgili değil), görev düzenlenir: Bu düzenleme işlemi sırasında işletim sistemi, DATA gönderme alanındaki verilere erişir (TSEND). • Müteakip çağrı, alma görevi düzenlenir: Bu düzenleme işlemi sırasında veriler alma alanına yazılır. Bu nedenle, bir hata, alma alanında tutarsız verilerin olmasına yol açabilir (TRCV).
1	8085	• LEN parametresi, izin verilen en büyük değerden büyüktür (TSEND) ve (TRCV). • LEN veya DATA parametresi, yordamın ilk yürütülüşünden itibaren değişti (TRCV).
1	8086	ID parametresi izin verilen adres aralığında değil.
1	8088	LEN parametresi, DATA'da belirlenen bellek alanından büyüktür.
1	80A1	İletişim hatası: • Belirtilen bağlantı henüz kurulmadı (TSEND ve TRCV). • Belirtilen bağlantı sonlandırılır. Bu bağlantı üzerinden bir gönderme veya alma görevinin gerçekleştirilmesi mümkün değil (TSEND ve TRCV). • Arayüz yeniden başlatılır (TSEND). • Arayüz yeni parametreler alıyor (TRCV).
1	80C3	Kaynaklarda dahili kusur: Bu ID'ye sahip bir blok, başka bir öncelik sınıfında düzenleniyor.
1	80C4	Geçici iletişim hatası: • İletişim partneriyle bağlantı şu anda kurulamıyor. • Arayüz yeni parametrelendirmeler alıyor veya bağlantı şimdi kuruluyor.

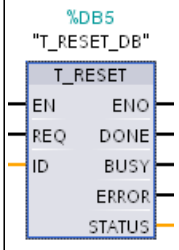
Ethernet bağlantı protokolleri

Her CPU, standart PROFINET iletişimini destekleyen entegre bir PROFINET bağlantısına sahiptir. TSEND_C, TRCV_C, TSEND ve TRCV yordamlarının tümü TCP ve ISO on TCP Ethernet protokollerini destekler.
Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Yerel CPU ile partner CPU arasındaki bağlantı yolunun yapılandırılması (Sayfa 563)".

11.5.8.15 T_RESET (mevcut bağlantıyı sonlandır ve yeni bağlantı kur) yordamı

"T_RESET" yordamıyla mevcut bir bağlantı sonlandırılır ve ardından yeni bir bağlantı kurulur:

Tablo 11-43 T_RESET yordamı

LAD/FBD	SCL	Açıklama
	**"T_RESET_DB"(req:=_bool_in_, id:=_word_in_, done=>_bool_out_, error=>_bool_out_, status=>_word_out_);**	"T_RESET" yordamıyla mevcut bir bağlantı sonlandırılır ve ardından yeni bir bağlantı kurulur.

Bağlantının yerel uç noktaları korunur. Bunlar, aşağıdaki koşullar sağlandığında otomatik oluşturulur:

- Bir bağlantı yapılandırıldıysa ve CPU'ya yüklendiyse
- Bir bağlantı, örneğin "TCON (Sayfa 616)" yordamı çağrılarak kullanıcı programı tarafından kurulduysa.

"T_RESET" yordamı, CPU'nun yerel arayüzünün veya bir CM'nin/CP'nin arayüzünün bağlantı için kullanılıp kullanılmadığından bağımsız olarak tüm bağlantı tipleri için yürütülebilir. TCP ile geçici modda veri aktarımı için kullanılan bağlantılar, bu bağlantılara bir bağlantı ID'si ile referans verilemediği için istisna oluşturur.
"T_RESET" yordamı REQ parametresiyle çağrıldıysa, ID parametresiyle belirtilen bağlantı sonlandırılır ve gerekirse veriler için kullanılan gönderme ve alma tamponları silinir. Bağlantının sonlandırılması nedeniyle, devam eden bir veri aktarımı da iptal edilir. Mevcut durumda bir veri aktarımı gerçekleşiyorsa, bu nedenle veri kaybı yaşanabilir. Bu durumda, aktif bağlantı partneri olarak ayarlanan CPU, kesilen iletişim bağlantısını otomatik olarak tekrar kurmaya çalışır. Bu nedenle, iletişim bağlantısını tekrar kurmak için "TCON (Sayfa 616)" yordamını çağırmak gerekmez.
DONE, BUSY ve STATUS çıkış parametreleri görev durumunu belirtir.

Parametreler için kullanılan veri tipleri

Aşağıdaki tabloda, T_RESET yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L, T, C veya sabit	REQUEST kumanda parametresi, ID aracılığıyla belirtilen bağlantıyı sonlandırma görevini başlatır. Görev, bir yükselen kenar durumunda başlar.
ID	Input	CONN_OUC (WORD)	L, D veya sabit	Sonlandırılması gereken, pasif partnerle kurulan bağlantıya verilen referans. ID, yerel bağlantı açıklamasındaki ilgili parametre ID'siyle eşleşmelidir. Değer aralığı: W#16#0001 - W#16#0FFF
DONE	Output	BOOL	E, A, M, D, L	DONE durum parametresi • 0: Görev henüz başlatılmadı veya halen gerçekleştiriliyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
BUSY	Output	BOOL	E, A, M, D, L	BUSY durum parametresi • 0: Görev tamamlandı. • 1: Görev henüz tamamlanmadı.
ERROR	Output	BOOL	E, A, M, D, L	ERROR durum parametresi • 0: Hata yok. • 1: Düzenleme sırasında hata oluştu. STATUS parametresi hata türüyle ilgili ayrıntılı bilgiler içerir
STATUS	Output	WORD	E, A, M, D, L	STATUS durum parametresi Hata bilgisi (bkz. "STATUSparametresi" tablosu).

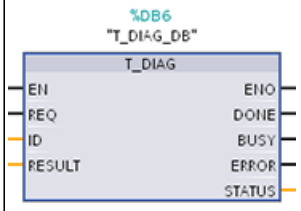
STATUS parametresi

Hata Bit'i	STATUS* (W#16#...)	Açıklama
0	0000	Hata yok.
0	0001	Bağlantı kurulmadı.
0	7001	Bağlantıyı sonlandırma işlemi başlatıldı.
0	7002	Bağlantı sonlandırılır.
1	8081	ID parametresinde bilinmeyen bir bağlantı belirtildi.

11.5.8.16 T_DIAG (bir bağlantının durumunu kontrol et ve bilgileri oku) yordamı

T_DIAG yordamı, bir bağlantının durumunu kontrol eder ve bu bağlantının yerel uç noktasındaki diğer bilgileri okur:

Tablo 11-44 T_DIAG yordamı

LAD/FBD	SCL	Açıklama
	**"T_DIAG_DB"(req:=_bool_in_, id:=_word_in_, done=>_bool_out_, error=>_bool_out_, status=>_dword_out_);**	T_DIAG yordamıyla bir bağlantının durumunu kontrol edebilir ve bu bağlantının yerel uç noktasındaki diğer bilgileri okuyabilirsiniz.

T_DIAG yordamı aşağıdaki şekilde çalışır:

- Bağlantı, ID parametresi tarafından belirtilir. Hem bağlantı editöründe yapılandırılan bağlantı uç noktalarını hem de programlanan bağlantı uç noktalarını (örn. TCON yordamıyla) okuyabilirsiniz.
  Geçici bağlantı uç noktaları (örn. bir mühendislik istasyonu ile bağlantı sırasında oluşturulan uç noktalar) teşhis edilemez, çünkü bu işlemde bağlantı ID'si oluşturulmaz.
- Okunan bağlantı bilgileri, RESULT parametresi tarafından belirtilen bir yapıda kaydedilir.
- STATUS çıkış parametresi, bağlantı bilgilerini okumanın mümkün olup olmadığını belirtir. RESULT parametresindeki yapıda yer alan bağlantı bilgileri sadece T_DIAG yordamı STATUS = W#16#0000 ve ERROR = FALSE ile tamamlandığı takdirde geçerlidir.
  Bir hata oluşursa, bağlantı bilgileri değerlendirilemez.

Olası bağlantı bilgileri

TDiag_Status yapısı, RESULT parametresindeki bağlantı bilgilerini okumak için kullanılabilir. TDiag_Status yapısı sadece bir bağlantı uç noktasıyla ilgili en önemli bilgileri (örn. kullanılan protokol, bağlantı durumu ve gönderilen veya alınan veri Bayt'larının sayısı) içerir.

Yapı ve TDiag_Status yapısının parametreleri aşağıda açıklanmaktadır (bkz. Yapı tablosuTDIAG_Status).

Parametreler için kullanılan veri tipleri

Aşağıdaki tabloda, T_DIAG yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L, T, C veya sabit	Bir yükselen kenar mevcutsa, ID parametresinde belirtilen bağlantının kontrol edilmesi için kullanılan yordamı başlatır.
ID	Input	CONN_OUC (WORD)	L, D veya sabit	Atanan bağlantıya verilen referans. Değer aralığı: W#16#0001 - W#16#0FFF
RESULT	InOut	VARIANT	D	İşaretçiyi, bağlantı bilgilerinin kaydedildiği yapının üzerine götürün. TDiag_Status yapısı, RESULT parametresinde kullanılabilir (bunun hakkındaki bir açıklamayı, TDIAG_Status yapısıyla ilgili tabloda bulabilirsiniz).
DONE	Output	BOOL	E, A, M, D, L	Durum parametresi: • 0: Yordam henüz başlatılmadı veya henüz düzenleniyor. • 1: Yordam hatasız bir şekilde yürütüldü.
BUSY	Output	BOOL	E, A, M, D, L	Durum parametresi: • 0: Yordam henüz başlatılmadı veya sonlandırıldı. • 1: Yordam henüz tamamlanmadı. Yeni bir görev başlatılamaz.
ERROR	Output	BOOL	E, A, M, D, L	Durum parametresi: • 0: Hata yok. • 1: Hata oluştu.
STATUS	Output	WORD	E, A, M, D, L	Yordamın durumu

BUSY, DONE ve ERROR parametreleri

T_DIAG yordamının yürütme durumunu, BUSY, DONE, ERROR ve STATUS parametreleriyle kontrol edebilirsiniz. BUSY parametresi, düzenleme durumunu gösterir. DONE parametresiyle, bir yordamın başarıyla yürütülüp yürütülmediğini kontrol edebilirsiniz. T_DIAG yordamı yürütülürken hatalar oluşursa, ERROR parametresi ayarlanır.

Aşağıdaki tabloda, BUSY, DONE ile ERROR parametrelerinin arasındaki ilişkiler gösterilmektedir:

BUSY	DONE	ERROR	Açıklama
1	-	-	Yordam düzenleniyor.
0	1	0	Yordam başarıyla yürütüldü. RESULT tarafından belirtilen yapıdaki veriler, ancak bu durum söz konusu ise geçerlidir.
0	0	1	Yordam hatayla sonlandı. Hata sebebi, STATUS parametresinde yayınlanır.
0	0	0	Yeni bir yordam atanmadı.

STATUS parametresi

Aşağıdaki tabloda, STATUS parametresindeki değerlerin anlamı gösterilmektedir:

Hata Bit'i	STATUS* (W#16#...)	Açıklama
0	0000	T_DIAG yordamı başarıyla yürütüldü. RESULT parametresinde belirtilen yapıdaki veriler değerlendirilebilir.
0	7000	Hiçbir yordam düzenleme işlemi aktif değil.
0	7001	Yordam düzenleme başlatıldı.
0	7002	Bağlantı bilgileri okunur (REQ Parametresi ilgili değildir).
1	8086	ID parametresindeki değer, geçerli aralığın (W#16#0001 - W#16#0FFF) dışında.
1	8089	RESULT parametresi, geçersiz bir veri tipini gösteriyor (sadece TDIAG_Status ve TDIAG_StatusExt yapıları).
1	80A3	ID parametresi, mevcut olmayan bir bağlantı uç noktasına referans veriyor. Programlanan bağlantılarda bu hata, TDISCON yordamı çağrıldıktan sonra da ortaya çıkabilir.
1	80C4	Dahili hata. Bağlantı uç noktasına erişim kısa bir süre için mevcut değil.

Yapı TDIAG_Status

Aşağıdaki tablo, TDIAG_Status yapısıyla ilgili ayrıntıları içermektedir. Her elemanın değeri, ancak yordam hatasız bir şekilde yürütüldüyse geçerlidir. Bir hata oluşursa, parametrelerin içeriği değişmez:

Adı	Veri tipi	Açıklama
Aşağıdaki parametreler TDIAG_Status yapısında yer alır:		
InterfaceID	HW_ANY	CPU'nun veya CM'nin/CP'nin arayüz ID'si (LADDR).
ID	CONN_OUC	Teşhis edilen bağlantının ID'si. Başarıyla gerçekleştirilen bir çağrıdan sonra bu elemanın değeri, T_DIAG yordamına ait parametre ID'si ile aynı olur.
ConnectionType	BYTE	Bağlantı için kullanılan protokol tipi: • 0x01: Kullanılmıyor. • ... • 0x0B: TCP protokolü (IP_v4) • 0x0C: ISO on TCP protokolü (RFC1006) • 0x0D: TCP protokolü (DNS) • 0x0E: Seçme protokolü • 0x0F: WDC protokolü • 0x10: SMTP protokolü • 0x11: TCP protokolü • 0x12: TCP ve ISO on TCP protokolü (RFC1006) • 0x13: UDP protokolü • 0x14: Rezerve • 0x15: PROFIBUS veriyolu erişim protokolü (FDL) • 0x16: ISO 8073 (orijinal ISO) uyarınca taşıma protokolü • ... • 0x20: SMTP veya SMTPS protokolü - IPv4'ü temel alır • 0x21: SMTP veya SMTPS protokolü - IPv6'yı temel alır • 0x22: SMTP veya SMTPS protokolü - FQDN'yi temel alır (**F**ully **Q**ualified **D**omain **N**ame) • ...

Adı	Veri tipi	Açıklama
		• 0x70: S7 bağlantısı • Diğer: Rezerve
ActiveEstablished	BOOL	• FALSE: Yerel, pasif bağlantı uç noktası • TRUE: Yerel, aktif bağlantı uç noktası
State	BYTE	Bağlantı uç noktasının güncel durumu • 0x00: Kullanılmıyor. • 0x01: Bağlantı sonlandırıldı. Geçici durum, örneğin T_RESET yordamı çağrıldıktan sonra. Bu durumda sistem, otomatik olarak bağlantıyı yeniden kurmayı dener. • 0x02: Aktif bağlantı uç noktası, desantral iletişim partneriyle bağlantı kurmaya çalışır. • 0x03: Pasif bağlantı uç noktası, desantral iletişim partneriyle bağlantı kurulmasın bekler. • 0x04: Bağlantı kuruldu. • 0x05: Bağlantı sonlandırılır. Bu durum, muhtemelen "T_RESET" veya "T_DISCON" yordamı çağrıldığı için gerçekleşir. Diğer olası sebepler protokol hataları veya hat duraklamalarıdır. • 0x06..0xFF: Kullanılmıyor.
Kind	BYTE	Bağlantı uç noktasının modu: • 0x00: Kullanılmıyor. • 0x01: Yapılandırılan ve CPU'ya yüklenen yapılandırılmış statik bağlantı. • 0x02: Yapılandırılan ve CPU'ya yüklenen yapılandırılmış dinamik bağlantı (şu anda desteklenmiyor). • 0x03: Kullanıcı programında "TCON" yordamıyla oluşturulan programlanan bağlantı. TDISCON yordamının çağrılması veya CPU'nun STOP işletim durumuna geçmesi, bağlantı uç noktasını tahrip etti. • 0x04: Örneğin mühendislik istasyonu (ES) veya operatör istasyonu (OS) tarafından kurulan geçici, dinamik bağlantı. (Bu bağlantı tipi, ID mevcut olmadığı için şu anda teşhis edilemiyor). • 0x05..0xFF: Kullanılmıyor.
SentBytes	UDINT	Gönderilen veri Bayt'larının sayısı.
ReceivedBytes	UDINT	Alınan veri Bayt'larının sayısı.

11.5.8.17 TMAIL_C (e-postayı CPU'nun Ethernet arayüzü üzerinden gönder) yordamı

"TMAIL_C" yordamıyla bir e-postayı S7-1200 CPU'sunun Ethernet arayüzü üzerinden gönderebilirsiniz.

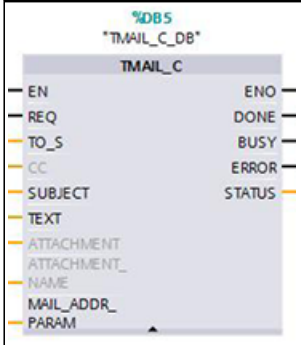
TMAIL_C yordam iki fonksiyona sahiptir:

- CPU arayüzü üzerinden e-posta gönderme
- Bir CP arayüzünden e-posta gönderme

TMAIL_C yordamının kullanılması için aşağıdaki ön koşullar sağlanmış olmalıdır:

- Donanımı yapılandırdınız
- Ağ alt yapısı posta sunucusuyla bir iletişim bağlantısına izin verir

Tablo 11-45 TMAIL_C yordamı

LAD/FBD	SCL	Açıklama
	**"TMAIL_C_DB"(** **req:=_bool_in_,** **to_s:=_string_in_,** **cc:=_string_in_,** **subject:=_string_in_,** **text:=_string_in_,** **attachment:=_variant_in_,** **attachment_name:=_string_in_,** **mail_addr_param:=_string_in_,** **done=>_bool_out_,** **busy=>_bool_out_,** **error=>_bool_out_,** **status=>_word_out_);**	TMAIL_C yordamıyla bir e-postayı S7-1200 CPU'sunun Ethernet arayüzü üzerinden gönderebilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

E-postanın içeriğini ve bağlantı verilerini aşağıdaki parametrelerle belirleyebilirsiniz:

- TO_S ve CC parametreleriyle alıcı adresleri
- SUBJECT ve TEXT parametreleriyle e-postanın içeriği
- ATTACHMENT ve ATTACHMENT_NAME parametrelerindeki VARIANT işaretçileriyle opsiyonel ek
- MAIL_ADDR_PARAM (Sayfa 649) parametresinde posta sunucusu için bağlantı verileri, adresleme ve kimlik doğrulama

TMAIL_C sürüm V6.0 ve daha yeni sürümlerden itibaren ve S7-1200 CPU Firmware sürümü V4.x'ten itibaren TMAIL_C yordamını, bir e-postayı S7-1200 CPU'nuzun entegre bir Ethernet portu üzerinden güvenli iletişimle göndermek için kullanabilirsiniz. Gönderme işlemi için gereken verileri, MAIL_ADDR_PARM (Sayfa 649) parametresinde TMail_V4_SEC veya TMail_QDN_SEC SDT'si ile tanımlayabilirsiniz.

Bir SMS'i doğrudan TMAIL_C yordamıyla gönderemezsiniz. E-postanın posta sunucusundan SMS olarak iletilip iletilemediği, telekomünikasyon servis sağlayıcınıza bağlıdır.

Yordamın çalışma şekli

E-postayı gönderme işlemi, REQ parametresi için kullanılan kenar "0" iken "1" olarak değişirse başlatılır.

TMAIL_C yordamı, görev durumunu "BUSY", "DONE", "ERROR" ve "STATUS" çıkış parametreleri aracılığıyla bildirir.

TMAIL_C yordamı asenkron olarak çalışır, yani yürütülmesi birden fazla çağrıyı kapsar. "TMAIL_C" yordamı çağrılırken bir örnek belirtilmelidir.

Aşağıdaki durumlarda, posta sunucusuyla kurulan bağlantı iptal edilir:

- CPU, TMAIL_C aktifken STOP işletim durumuna geçerse.
- Industrial Ethernet veriyolunda iletişim sorunları ortaya çıkarsa.
  Bu durumda, e-posta gönderme işlemi durdurulur ve e-posta alıcıya ulaşamaz.

Yordam başarıyla yürütüldüyse ve e-posta gönderildiyse de bağlantı sonlandırılır.

DİKKAT
Kullanıcı programının değiştirilmesi Kullanıcı programınızın, doğrudan TMAIL_C yordamının çağrılarıyla ilgili olan kısımlarını ancak aşağıdaki koşullar sağlandığı takdirde değiştirebilirsiniz: • CPU STOP işletim durumunda ise • Mevcut durumda e-posta gönderilmiyorsa (yani REQ = 0 ve BUSY = 0). Bu, özellikle program bloklarının, TMAIL_C çağrılarının veya TMAIL_C örneğinin silinmesi ve değiştirilmesiyle ilgilidir. Bu kısıtlamaya dikkat edilmezse, bağlantı kaynakları bloke edilebilir. Otomasyon sistemi, TCP/IP iletişim fonksiyonlarıyla karşılaştırıldığında Industrial Ethernet üzerinden tanımsız bir duruma geçebilir. Değişiklikler aktarıldıktan sonra CPU'nun sıcak veya soğuk başlatma yoluyla yeniden başlatılması gerekir.

Veri tutarlılığı

TO_S, CC, SUBJECT, TEXT, ATTACHMENT ve MAIL_ADDR_PARAM parametreleri, TMAIL_C yordamı tarafından, bu yordam yürütülürken parametreler ancak görev tamamlandıktan sonra (BUSY = 0) değiştirilebilecek şekilde kullanılır.

SMTP kimlik doğrulama

Kimlik doğrulama, örneğin bir şifre sorgusuyla kimlik kontrolü için kullanılan bir yöntemi tanımlar.

S7-1200 CPU'sunun arayüzünü kullanıyorsanız, "TMAIL_C" yordamı, çoğu posta sunucusu için gerekli olan AUTH-LOGIN SMTP kimlik doğrulama yöntemini destekler. Posta sunucunuzun kimlik doğrulama yöntemiyle ilgili bilgileri, posta sunucunuzun işletme kitabında veya Internet servis sağlayıcınızın web sitesinde bulabilirsiniz.

- AUTH-LOGIN kimlik doğrulama yöntemini kullanabilmek için "TMAIL_C" yordamı, posta sunucusunda oturum açmak için kullanması gereken kullanıcı adına ihtiyaç duyar. Bu kullanıcı adı, posta sunucunuzda bir posta hesabı oluşturmak için faydalandığınız kullanıcı adıyla aynıdır. Bu bilgi, UserName parametresiyle MAIL_ADDR_PARAM parametresindeki yapıya aktarılır.
  MAIL_ADDR_PARAM parametresinde bir kullanıcı adı girilmezse, AUTH-LOGIN kimlik doğrulama yöntemi kullanılmaz. Bu durumda e-posta kimlik doğrulaması yapılmadan gönderilir.
- TMAIL_C yordamı, oturum açmak için ayrıca ilgili şifreye de ihtiyaç duyar. Bu, posta hesabınızı oluştururken belirttiğiniz şifredir. Bu bilgi, PassWord parametresiyle MAIL_ADDR_PARAM parametresindeki yapıya aktarılır.

Parametreler için kullanılan veri tipleri

Aşağıdaki tabloda, TMAIL_C yordamının parametreleri gösterilmektedir:

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
REQ	Input	BOOL	E, A, M, D, L, T, C veya sabit	REQUEST kumanda parametresi: Bir yükselen kenar durumunda bir e-postanın gönderilmesi işlemini etkinleştirir.
TO_S (Sayfa 656)	Input	STRING	D	Alıcı adresleri 180 karakterlik (Bayt) bir maksimum uzunluğa sahip STRING. E-posta adres formatını öğrenmek için lütfen parametre açıklamasındaki örneğe bakın.
CC (Sayfa 656)	Input	STRING	D	CC alıcı adresleri (opsiyonel) 180 karakterlik (Bayt) bir maksimum uzunluğa sahip STRING. TO_S parametresi için kullanılanla aynı e-posta adres formatı. Burada boş bir karakter dizisi atarsanız, yordam, e-postayı bir CC alıcısına göndermez.
SUBJECT	Input	STRING	D	E-postanın konusu 180 karakterlik (Bayt) bir maksimum uzunluğa sahip STRING.
TEXT	Input	STRING	D	E-postanın metni (opsiyonel) 180 karakterlik (Bayt) bir maksimum uzunluğa sahip STRING. Bu parametrede bir boş karakter dizisi atarsanız, yordam, e-postayı metin olmadan gönderir.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
ATTACHMENT	Input	VARIANT	D	E-posta eki (opsiyonel) Bir karaktere/Bayt'a/sözcüğe/çift sözcüğe/karakter dizisi alanına (ArrayOfChar, ArrayOfByte, ArrayOfWord, ArrayOfDWord, or String) verilen 64 Bayt'lık bir maksimum uzunluğa sahip referans. Not: ATTACHMENT parametresinde hiçbir değeri atamazsanız veya boş bir karakter dizisi atarsanız, yordam, e-postayı ek olmadan gönderir.
ATTACHMENT_NAME	Input	VARIANT	D	E-posta ekinin adı (opsiyonel) Ekin dosya adını gösteren, maksimum 50 karakter (Bayt) uzunluğunda bir karakter dizisine verilen referans. Bu parametreye boş bir karakter dizisi atanırsa, posta eki "attachment.bin" dosya adıyla gönderilir. AttachmentName parametresiyle, iletişim partneri e-postayı aldığında görüntülenecek olan ekin adını atayabilirsiniz. TMail_FileReference SDT'si, AttachmentName parametresi için otomatik olarak FileName parametresini kullanır. TMail_FileReference SDT'sini kullanırsanız, AttachmentName parametresi önemli değildir. Bu parametreyi boş bırakın. TMail_FileReference SDT'sini kullanırken AttachmentName parametresini girerseniz, TMAIL_C yordamı bir hata oluşturur. Konuyla ilgili daha fazla bilgiyi, "Örnek DB'nin hata kodları, SFB_STATUS parametresi" başlığında bulabilirsiniz.
MAIL_ADDR_PARAM (Sayfa 649)	Input	VARIANT	D	E-posta sunucusunun bağlantı parametreleri ve adresi Bağlantı parametrelerini, TMail_V4, TMail_FQDN, TMail_V4_SEC veya TMail_QDN_SEC yapısına sahip SDT ile belirleyebilirsiniz (bkz. Parametre açıklaması).
DONE (Sayfa 656)	Output	BOOL	E, A, M, D, L	Durum parametresi • DONE = 0: Görev henüz başlatılmadı veya halen gerçekleştiriliyor. • DONE = 1: Görev hatasız bir şekilde gerçekleştirildi.

Parametre	Deklarasyon	Veri tipi	Bellek alanı	Açıklama
BUSY (Sayfa 656)	Output	BOOL	E, A, M, D, L	Durum parametresi • BUSY = 0: TMAIL_C'nin işlenmesi durduruldu. • BUSY = 1: E-postanın aktarılması işlemi henüz tamamlanmadı.
ERROR (Sayfa 656)	Output	BOOL	E, A, M, D, L	Durum parametresi • ERROR = 0: Hata oluşmadı. • ERROR = 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS (Sayfa 659)	Output	WORD	E, A, M, D, L	Durum parametresi TMAIL_C yordamının geri dönüş değeri veya hata bilgileri (bkz. Parametre açıklaması).

NOT

Opsiyonel parametreler

Yordam, opsiyonel CC, TEXT ve ATTACHMENT parametrelerini ancak ilgili parametreler uzunluğu > 0 olan bir karakter dizisi içerdiği takdirde gönderir.

MAIL_ADDR_PARAM parametresinde, e-postaların gönderilmesi için kullanılan bağlantıyı tanımlayabilir ve e-posta sunucusunun adresini ve oturum açma bilgilerini kaydedebilirsiniz. MAIL_ADDR_PARAM parametresinde kullandığınız sistem veri tipi (SDT), e-posta sunucusunun adreslenmesi gereken formata bağlıdır:

SDT	Açıklama	Arayüz desteği
TMail_V4	IP adresi üzerinden adresleme (IPv4)	CPU ve CP
TMail_V6	IP adresi üzerinden adresleme (IPv6)	CP
TMail_FQDN	Tam nitelikli etki alanı adları üzerinden adresleme (FQDN)	CP
TMail_V4_SEC	IP adresi üzerinden güvenli adresleme (IPv4)	CPU ve CP
TMail_V6_SEC	IP adresi üzerinden güvenli adresleme (IPv6)	CP
TMail_QDN_SEC	Tam nitelikli etki alanı adları üzerinden güvenli adresleme (FQDN)	CPU ve CP

TMail_V4: Posta sunucusunu IP adresi üzerinden adresleme (IPv4)

Parametre	Veri tipi	Açıklama
TMail_v4	Struct	
InterfaceId	LADDR	Ethernet arayüzünün donanım kodu
ID	CONN_OUC	Bağlantı ID'si
ConnectionType	BYTE	Bağlantı tipi. IPv4 için bağlantı tipi olarak 16#20'yi seçin.
ActiveEstablished	BOOL	Aktif/pasif bağlantı kurma. CPU her zaman SMTP istemcisidir.
CertIndex	BYTE	= 0: SMTP kullanılır (**S**imple **M**ail **T**ransfer **P**rotocol). E-posta bir S7-1200 CPU'sunun arayüzü üzerinden gönderilecekse SMTP kullanılmalıdır. ≠0: Bağlantıyı güvenli hâle getirmek için SMTPS kullanılır (CP arayüzü)
WatchDogTime	TIME	Yürütme süresi denetimi. Bu parametreyle, gönderme işlemi için maksimum yürütme süresi belirlenebilir. Bu değer, bir zaman aşımı değerine ulaşılmadan ve TMAIL_C yordamının yürütülmesi durdurulmadan önce TMAIL_C yordamının hangi süreyle yürütme aşamasında kalabileceğini belirler. TMAIL_C yordamının 6.0 sürümünden itibaren, artık bir WatchDogTime 0 değerine izin verilmektedir; bu değer, yürütme sırasında TMAIL_C yordamına süreyi devre dışı bırakma bildiriminde bulunur. TMAIL_C yordamının yürütülmesini daha deterministik hale getirmek üzere, WatchDogTime için 0'dan farklı bir değer girmeye devam edebilirsiniz. Not: Yavaş bağlantılarda bağlantı kurma işlemi daha uzun sürebilir (yaklaşık bir dakika). WatchDogTime parametresini belirtirken, bağlantı kurmak için gereken süreye dikkat edin.
MailServerAddress	IP_v4	Posta sunucusunun IP adresi. Şu formatta IPv4: XXX.XXX.XXX.XXX (desimal). Örnek: 192.142.131.237
UserName	STRING[254]	Posta sunucusunda oturum açmak için kullanılan ad
PassWord	STRING[254]	Posta sunucusu için kullanılan şifre
From	EMAIL_ADDR	E-posta göndericisinin, şu iki STRING parametresi aracılığıyla belirlenen adresi. Örnek: "myname@mymailserver.com".
LocalPartPlusAt-Sign	STRING[64]	@- karakteri de dahil olmak üzere, gönderici adresinin yerel kısmı. Örnek: "myname@".
FullQualifiedDomainName	STRING[254]	Posta sunucusuna ait Fully Qualified Domain Name (FQDN olarak kısaltılır). Örnek: "mymailserver.com".

TMail_V6: Posta sunucusunu IP adresi üzerinden adresleme (IPv6)

Parametre	Veri tipi	Açıklama
TMail_V6	Struct	
InterfaceId	LADDR	Arayüzün donanım kodu
ID	CONN_OUC	Bağlantı ID'si
ConnectionType	BYTE	Bağlantı tipi. IPv6 için bağlantı tipi olarak 16#21'i seçin.
ActiveEstablished	BOOL	Durum Bit'i. Bağlantı kurulduktan sonra "1" olarak ayarlanır.
CertIndex	BYTE	=0: SMTP kullanılır (**S**imple **M**ail **T**ransfer **P**rotocol). E-posta bir S7-1500 CPU'sunun arayüzü üzerinden gönderilecekse SMTP kullanılmalıdır. ≠0: SMTPS, bağlantıyı, (CP'lerle/CM'lerle) kurulmadan önce güvenli hâle getirmek için kullanılır. CertIndex parametresini, kullanılması gereken sertifikayı belirtmek için kullanabilirsiniz (bkz. "Proje gezgini" > "Global güvenlik ayarları" > "Sertifika yöneticisi").
WatchDogTime	TIME	Yürütme sırasında zaman denetimi. Bu parametreyle, gönderme işlemi için maksimum yürütme süresini belirleyebilirsiniz. Not: Bağlantı yavaşsa, bağlantı kurma işlemi daha uzun sürebilir (yaklaşık bir dakika). WATCH_DOG_TIME parametresini belirtirken, bağlantı kurmak için gereken süreye dikkat edin. Belirtilen süre dolduktan sonra bağlantı sonlandırılır.
MailServerAddress	IP_V6	Posta sunucusunun şu formatta IP adresi (IPv6): XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX (onaltılı). Adres, 2'şer Bayt'lık 8 bloğa ayrılır (toplam 16 Bayt). Örnek: 2001:db8:1f11:08d3:290:27ff:0370:2093
UserName	STRING[254]	Posta sunucusu için kullanılan oturum açma adı
PassWord	STRING[254]	Posta sunucusu için kullanılan şifre
From	EMAIL_ADDR	Göndericinin, şu iki STRING parametresi aracılığıyla belirlenen e-posta adresi. Örnek: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	@- karakteri de dahil olmak üzere, gönderici adresinin yerel bileşeni. Örnek: "myname@"
FullQualifiedDomainName	STRING[254]	Posta sunucusuna ait Fully Qualified Domain Name (FQDN olarak kısaltılır). Örnek: "mymailserver.com"

TMail_FQDN : Posta sunucusunuFQDN üzerinden adresleme

Parametre	Veri tipi	Açıklama
TMail_FQDN	Struct	
InterfaceId	LADDR	Ethernet arayüzünün donanım kodu
ID	CONN_OUC	Bağlantı ID'si
ConnectionType	BYTE	Bağlantı tipi. FQDN için bağlantı tipi olarak 16#22'yi seçin.
ActiveEstablished	BOOL	Aktif/pasif bağlantı kurma. Bir iletişim işlemcisi (CP) her zaman SMTP istemcisidir.
CertIndex	BYTE	= 0: SMTP kullanılır (**S**imple **M**ail **T**ransfer **P**rotocol). ≠0: SMTPS, bağlantıyı, (CP arayüzüyle) kurulmadan önce güvenli hale getirmek için kullanılır. CertIndex, kullanılması gereken sertifikayı belirtir.
WatchDogTime	TIME	Yürütme süresi denetimi. Bu parametreyle, gönderme işlemi için maksimum yürütme süresi belirlenebilir. Not: Yavaş bağlantılarda bağlantı kurma işlemi daha uzun sürebilir (yaklaşık bir dakika). WatchDogTime parametresini belirtirken, bağlantı kurmak için gereken süreye dikkat edin.
MailServerAddress	STRING[254]	Posta sunucusuna ait FQDN (**F**ully **Q**ualified **D**omain **N**ame, tam nitelikli etki alanı adı). Posta sunucusu, FQDN üzerinden adreslenir. Örnek: "www.mymailserver.com."
UserName	STRING[254]	Posta sunucusunda oturum açmak için kullanılan ad
PassWord	STRING[254]	Posta sunucusu için kullanılan şifre
From	Struct	E-posta göndericisinin, şu iki STRING parametresi aracılığıyla belirlenen adresi. Örnek: "myname@mymailserver.com".
LocalPartPlusAt-Sign	STRING[64]	@- karakteri de dahil olmak üzere, gönderici adresinin yerel bileşeni. Örnek: "myname@".
FullQualifiedDoma-inName	STRING[254]	Posta sunucusunun tam nitelikli etki alanının adı (kısaltması FQDN). Örnek: "mymailserver.com".

TMail_V4_SEC: Posta sunucusunu IP adresi üzerinden adresleme (IPv4)

Parametre	Veri tipi	Açıklama
TMail_V4_SEC	Struct	
InterfaceId	LADDR	Ethernet arayüzünün değer aralığının donanım kodu: • 0 (yeni): İşletim sistemi, uygun bir entegre portu kendisi seçer. • Kullanılması gereken entegre portun donanım kodu.
ID	CONN_OUC	Bağlantıya verilen referans: Değer aralığı: • 0 (yeni): İşletim sistemi, dahili aralıktan serbest bir bağlantı ID'si seçer. • 1 ila 4095: Kullanılması gereken bağlantı ID'si
ConnectionType	BYTE	Bağlantı tipi. IPv4 için bağlantı tipi olarak 16#20'yi seçin.
ActiveEstablishment	BOOL	Aktif/pasif bağlantı kurma. CPU her zaman SMTP istemcisidir.

Parametre	Veri tipi	Açıklama
WatchDogTime	TIME	Yürütme sırasında zaman denetimi. Bu parametreyle, gönderme işlemi için maksimum yürütme süresini belirleyebilirsiniz. Bu değer, bir zaman aşımı değerine ulaşılmadan ve TMAIL_C yordamının yürütülmesi durdurulmadan önce TMAIL_C yordamının hangi süre boyunca yürütme aşamasında kalabileceğini belirler. TMAIL_C yordamının 6.0 sürümünden itibaren, artık bir WatchDogTime 0 değerine izin verilmektedir; bu değer, yürütme sırasında TMAIL_C yordamına süreyi devre dışı bırakma bildiriminde bulunur. TMAIL_C yordamının yürütülmesini daha deterministik hale getirmek üzere, WatchDogTime için 0'dan farklı bir değer girmeye devam edebilirsiniz. Not: Bağlantı yavaşsa, bağlantı kurma işlemi daha uzun sürebilir (yaklaşık bir dakika). WatchDogTime parametresini belirtirken, bağlantı kurmak için gereken süreye dikkat edin.
MailServerAddress	IP_V4	Posta sunucusunun gerçek IPv4 formatındaki IP adresi: XXX.XXX.XXX.XXX (desimal) Örnek: 192.142.131.237
UserName	STRING[254]	Kullanıcı adı Kendinizi e-posta sağlayıcısına e-posta kutusunun sahibi olarak tanıtmak üzere e-posta kutunuza erişmek için "kullanıcı adınızı" kullanmanız gerekir.
PassWord	STRING[254]	Kullanıcı şifresi Kendinizi e-posta sağlayıcısına e-posta kutusunun sahibi olarak tanıtmak üzere e-posta kutunuza erişmek için "şifrenizi" kullanmanız gerekir.
From	EMAIL_ADDR	E-posta göndericisinin, şu iki STRING parametresi aracılığıyla belirlenen adresi. Örnek: "myname@mymailserver.com"
LocalPartPlusAtSign	STRING[64]	@- karakteri de dahil olmak üzere, gönderici adresinin yerel bileşeni. Örnek: "myname@"
FullQualifiedDomainName	STRING[254]	Posta sunucusunun tam nitelikli etki alanı adı (kısaltması FQDN). Örnek: "mymailserver.com"
RemotePort	UINT	Posta sunucusunun TCP portu
ActivateSecureConn	BOOL	0: SMTP bağlantısı (güvenli hale getirilmedi). Bu durumlarda aşağıdaki parametreler ilgili değildir. 1: Güvenli SMTP bağlantısı
ExtTLSCapabilities	BYTE	Şu anda kullanılmıyor
TLSServerCertRef	UDINT	Posta sunucusunun, TLS istemcisinin, TLS sunucusunun kimliğini doğrulamak için kullandığı X.509 V3 (CA) sertifikasına verilen referans

TMail_V6_SEC: Posta sunucusunu, IPv6 formatındaki IP adresi üzerinden adresleme

Parametre	Veri tipi	Açıklama
TMail_V6_SEC	Struct	
InterfaceId	LADDR	Ethernet arayüzünün donanım kodu
ID	CONN_OUC	Bağlantı ID'si
ConnectionType	BYTE	Bağlantı tipi. IPv6 için bağlantı tipi olarak 16#21'i seçin.
ActiveEstablishment	BOOL	Aktif/pasif bağlantı kurma. CP her zaman SMTP istemcisi olduğu için, bu parametre "1" olarak ayarlanmalıdır.
WatchDogTime	TIME	Yürütme sırasında zaman denetimi. Bu parametreyle, gönderme işlemi için maksimum yürütme süresini belirleyebilirsiniz. Not: Bağlantı yavaşsa, bağlantı kurma işlemi daha uzun sürebilir (yaklaşık bir dakika). WatchDogTime parametresini belirtirken, bağlantı kurmak için gereken süreye dikkat edin. Belirtilen süre dolduktan sonra bağlantı sonlandırılır.
MailServerAddress	IP_V6	Posta sunucusunun IPv6 formatındaki IP adresi: XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX (onaltılı). Adres, 2'şer Bayt'lık 8 bloğa ayrılır (toplam 16 Bayt). Örnek: 2001:db8:1f11:08d3:290:27ff:0370:2093
UserName	STRING[254]	Kullanıcı adı. Kendini e-posta sağlayıcısına e-posta kutusunun sahibi olarak tanıtmak üzere e-posta kutusuna erişim için gerekir.
PassWord	STRING[254]	Kullanıcı şifresi. Kendini e-posta sağlayıcısına e-posta kutusunun sahibi olarak tanıtmak üzere e-posta kutusuna erişim için gerekir.
From	EMAIL_ADDR	Göndericinin, şu iki STRING parametresi aracılığıyla belirlenen e-posta adresi. Örnek: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	@- karakteri de dahil olmak üzere, gönderici adresinin yerel bileşeni. Örnek: "myname@"
FullQualifiedDomainName	STRING[254]	Posta sunucusuna ait Fully Qualified Domain Name (FQDN olarak kısaltılır). Örnek: "mymailserver.com"
RemotePort	UINT	Posta sunucusunun TCP-TCP portu

Parametre	Veri tipi	Açıklama
ActivateSecureConn	BOOL	0: SMTP bağlantısı (güvenli hale getirilmedi). Bu durumlarda aşağıdaki parametreler ilgili değildir. 1: Güvenli SMTP bağlantısı
ExtTLSCapabilities	BYTE	Şu anda kullanılmıyor
TLSServerCertRef	UDINT	Posta sunucusunun, TLS istemcisinin, TLS sunucusunun kimliğini doğrulamak için kullandığı X.509 V3 (CA) sertifikasına verilen referans.

TMail_QDN_SEC: Posta sunucusunuFQDN üzerinden adresleme

Parametre	Veri tipi	Açıklama
TMail_QDN_SEC	Struct	
InterfaceId	LADDR	Ethernet arayüzünün değer aralığının donanım kodu: • 0 (yeni): İşletim sistemi, uygun bir entegre portu kendisi seçer. • Kullanılması gereken entegre portun donanım kodu.
ID	CONN_OUC	Bağlantıya verilen referans: Değer aralığı: • 0 (yeni): İşletim sistemi, dahili aralıktan serbest bir bağlantı ID'si seçer. • 1 ila 4095: Kullanılması gereken bağlantı ID'si
ConnectionType	BYTE	Bağlantı tipi. FQDN için bağlantı tipi olarak 16#22'yi seçin.
ActiveEstablishment	BOOL	Aktif/pasif bağlantı kurma. CPU her zaman SMTP istemcisidir.
WatchDogTime	TIME	Yürütme sırasında zaman denetimi. Bu parametreyle, gönderme işlemi için maksimum yürütme süresini belirleyebilirsiniz. Bu değer, bir zaman aşımı değerine ulaşılmadan ve TMAIL_C yordamının yürütülmesi durdurulmadan önce TMAIL_C yordamının hangi süre boyunca yürütme aşamasında kalabileceğini belirler. TMAIL_C yordamının 6.0 sürümünden itibaren, artık bir WatchDogTime 0 değerine izin verilmektedir; bu değer, yürütme sırasında TMAIL_C yordamına süreyi devre dışı bırakma bildiriminde bulunur. TMAIL_C yordamının yürütülmesini daha deterministik hale getirmek üzere, WatchDogTime için 0'dan farklı bir değer girmeye devam edebilirsiniz. Not: Bağlantı yavaşsa, bağlantı kurma işlemi daha uzun sürebilir (yaklaşık bir dakika). WatchDogTime parametresini belirtirken, bağlantı kurmak için gereken süreye dikkat edin.
MailServerQDN	STRING[254]	Posta sunucusuna ait FQDN (**F**ully **Q**ualified **D**omain **N**ame, tam nitelikli etki alanı adı). Posta sunucusu, bir "." ile bitmesi gereken FQDN üzerinden adreslenir. Örnek: "www.mymailserver.com."
UserName	STRING[254]	Kullanıcı adı Kendinizi e-posta sağlayıcısına e-posta kutusunun sahibi olarak tanıtmak üzere e-posta kutunuza erişmek için "kullanıcı adınızı" kullanmanız gerekir.

Parametre	Veri tipi	Açıklama
PassWord	STRING[254]	Kullanıcı şifresi Kendinizi e-posta sağlayıcısına e-posta kutusunun sahibi olarak tanıtmak üzere e-posta kutunuza erişmek için "şifrenizi" kullanmanız gerekir.
From	EMAIL_ADDR	E-posta göndericisinin, şu iki STRING parametresi aracılığıyla belirlenen adresi. Örnek: "myname@mymailserver.com"
LocalPartPlusAtSign	STRING[64]	@- karakteri de dahil olmak üzere, gönderici adresinin yerel kısmı. Örnek: "myname@"
FullQualifiedDomainName	STRING[254]	Posta sunucusunun tam nitelikli etki alanı adı (kısaltması FQDN). Örnek: "mymailserver.com"
RemotePort	UINT	Posta sunucusunun TCP portu
ActivateSecureConn	BOOL	0: SMTP bağlantısı (güvenli hale getirilmedi). Bu durumlarda aşağıdaki parametreler ilgili değildir. 1: Güvenli SMTP bağlantısı
ExtTLSCapabilities	BYTE	Şu anda kullanılmıyor
TLSServerCertRef	UDINT	Posta sunucusunun, TLS istemcisinin, TLS sunucusunun kimliğini doğrulamak için kullandığı X.509 V3 (CA) sertifikasına verilen referans

TMAIL_C yordamlarında sürüm 6.0'dan önce ve S7-1200 CPU Firmware'i V4.4'ten önce TO_S ve CC parametreleri girilirken aşağıdaki kurallar geçerlidir:

- Her adresten önce bir boşluk karakteri ve bir küçüktür işareti "<" girin.
- Her adresten sonra bir büyüktür işareti ">" girin.
- TO ve CC'deki adreslerin arasına bir virgül işareti girin.

Aşağıda, TO_S ve CC parametre karakter dizileriyle ilgili örnekler bulabilirsiniz:

- <wenna@mydomain.com>, <ruby@mydomain.com>
- <admin@mydomain.com>, <judy@mydomain.com>

TMAIL_C yordamında Sürüm 6.0'dan ve S7-1200 CPU Firmware'i V4.x'ten itibaren parametreler girilirken sadece şu kural geçerlidir:

- TO ve CC'deki adreslerin arasına bir virgül veya noktalı virgül işareti girin.

Aşağıda, TO_S ve CC parametre karakter dizileriyle ilgili örnekler bulabilirsiniz:

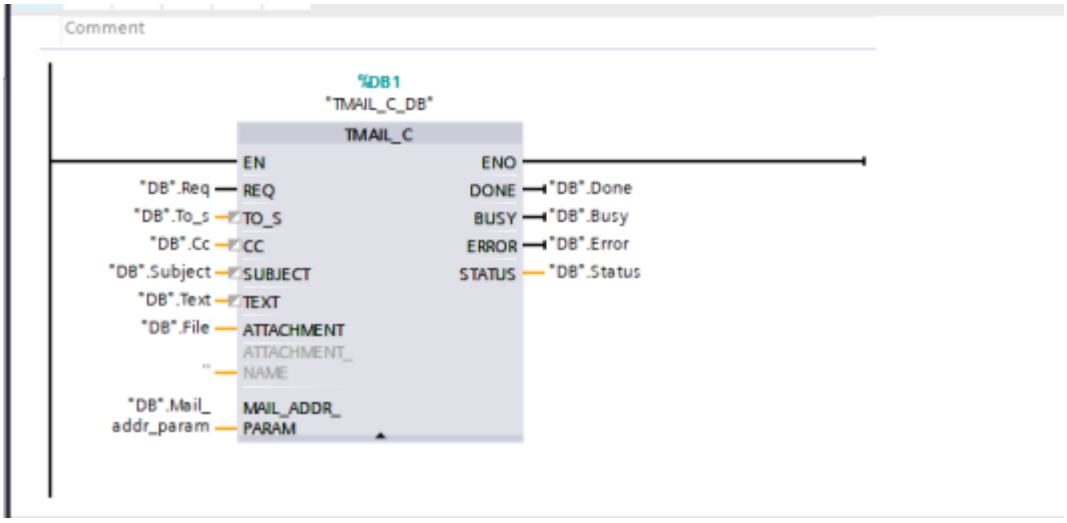
- wenna@mydomain.com, ruby@mydomain.com
- admin@mydomain.com, judy@mydomain.com

Runtime ve bellek alanıyla ilgili nedenlerle TMAIL_C yordamı, TO_S ve CC parametrelerinde sözdizimi kontrolü gerçekleştirmez.

BUSY çıkış parametresinin durumu "1" iken "0" olarak değişirse, DONE, BUSY ve ERROR çıkış parametrelerinin her biri sadece bir çevrim boyunca görüntülenir.
Aşağıdaki tabloda, DONE, BUSY ile ERROR arasındaki ilişkiler gösterilmektedir. Bu tablonun yardımıyla, TMAIL_C yordamının güncel durumunu ve e-postayı gönderme işleminin tamamlandığı anı belirleyebilirsiniz.

DONE	BUSY	ERROR	Açıklama
0	1	0	Görev düzenlenir.
1	0	0	Görev başarıyla gerçekleştirildi.
0	0	1	Görev hatayla sonlandı. Hata sebebi, STATUS (Sayfa 659) parametresine kaydedildi.
0	0	0	TMAIL_C yordamı bir (yeni) görev atamadı.

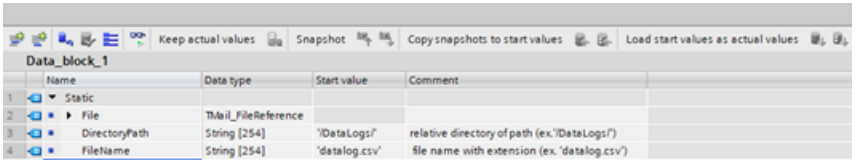
TMAIL_C sürüm V6.0'dan itibaren ve S7-1200 CPU Firmware sürümü V4.x'ten itibaren, TMAIL_C yordamının Attachment parametresi aracılığıyla TMail_FileReference SDT'sini ekleyebilir ve bu bileşene erişebilirsiniz. Daha sonra, SIMATIC memory card'daki (SMC) bir dosya yolunu adresleyebilirsiniz. Memory card mevcut değilse, PLC'nin dahili yükleme belleğinde bulunan reçete ve veri protokolü dizinlerine yine de erişebilirsiniz.
TMail_FileReference SDT'si, AttachmentName parametresi için otomatik olarak FileName parametresini kullanır.



SDT TMail_FileReference

TMail_FileReference SDT'si iki parametreden oluşur ve her iki parametrede de SIMATIC karakter dizileri söz konusudur:

- DirectoryPath parametresinde istediğiniz dosyanın dizinini adresleyebilirsiniz.
- FileName parametresi, önceki parametre tarafından belirtilen dizinde erişmek istediğiniz dosyanın adını ve dosyanın uzantısını (mevcutsa) belirtir.



TMAIL_C yordamında, DirectoryPath parametresinin adreslenmesi için sadece DataLogs, Recipes veya UserFiles dizinleri kullanılabilir. Bu dizinlerin içinde alt dizinleri adresleyebilirsiniz.

Yukarıda belirtilen temel dizinle ilgili kısıtlamanın dışında, bir sonraki "Dosyaların adreslenmesiyle ilgili kurallar" başlıklı alt bölümde, alt dizinler ve dosya adları adreslenirken uymanız gereken kurallar açıklanmaktadır.

Gönderebileceğiniz dosyanın boyutu, TMAIL_C yordamı tarafından sınırlanmaz. Programınızı oluştururken bu hususu göz önünde bulundurun.

Dosyaların adreslenmesiyle ilgili kurallar

Bir dosyayı TMAIL_C yordamıyla TMail_FileReference SDT'si üzerinden düzgün bir şekilde adreslemek için uyulması gereken belirli kurallar mevcuttur. Aşağıdaki alt bölümlerde, DirectoryPath ve FileName parametreleriyle ilgili ayrıntılar açıklanmaktadır. Genel olarak TMail_FileReference SDT'sindeki her iki parametre için de uyulmadığı takdirde TMAIL_C yordamının bir hata durumu oluşturmasına neden olan şu kurallar geçerlidir:

- Bir alt dizin veya dosya adı için boş karakter dizisi kullanamazsınız.
- Parametre karakter dizilerinin hiçbirinde ASCII kontrol karakterleri kullanamazsınız (onaltılık aralık: 0x00 - 0x1F).
- Parametre karakter dizilerinin hiçbirinde aşağıdaki rezerve edilmiş karakterleri kullanamazsınız:
  - < (küçüktür)
  - > (büyüktür)
  - : (İki nokta üst üste)
  - " (çift tırnak işareti)
  - / (Bölü işareti) (Bu karaktere DirectoryPath'te ayırma işareti olarak izin verilir.)
  - \ (Ters bölü işareti)
  - | (dikey çizgi veya Pipe)
  - ? (Soru işareti)
  - * (Yıldız karakteri)
- Hiçbir alt dizin ve hiçbir dosya adı boşluk karakteriyle veya noktayla bitemez.

DirectoryPath

İstediğiniz dizinleri SDT'nin DirectoryPath parametresine girerken aşağıdaki notlara dikkat edin. Kök dizin, PLC'nin Firmware lojiği tarafından türetilir, bunu bilmeniz gerekmez. Opsiyonel olarak bir ön ve bir müteakip bölü işareti (/) girebilirsiniz. İki bölü işaretinden biri girilmezse, Firmware, bölü işaretini otomatik olarak ekler. Bununla birlikte, DirectoryPath için aşağıdaki tüm girişler geçerli olur:

- /DataLogs/
- /DataLogs
- DataLogs/
- DataLogs

Ayrıca kök dizinden daha derin bir yola erişmek de mümkündür. Bunun için '/DataLogs/dir1/' formatını kullanabilirsiniz, burada her bölü işareti yeni bir dizini belirtir. Maksimum derinlik, kök dizin dahil sekiz değerindedir.

"Dosyaların adreslenmesiyle ilgili kurallar" alt bölümünde listelenen kuralların yanı sıra, bağıl yolların kullanılmasının kesinlikle yasak olduğuna dikkat etmeniz gerekir. Dolayısıyla '/DataLogs/' geçersiz bir yol adıdır. Ayrıca güncel dizinin gösterimi için bir alt dizin bileşeninde nokta kullanmak yasaktır (örnek: '/DataLogs/').

FileName

SDT'nin FileName parametresini kullanırken, "Dosyaların adreslenmesiyle ilgili kurallar" alt bölümünde listelenen kurallara dikkat edin. Ayrıca PLC'nin işletim sisteminin, dosya adını 60 karakterden daha küçük bir karakter sayısıyla sınırladığına dikkat etmeniz gerekir. 60 veya daha fazla karakter içeren bir dosya adını adreslerseniz, TMAIL_C yordamının yürütülmesi iptal edilir ve bir hata oluşur.

Bu istisnalar dışında, boyutu veya dosya uzantısından bağımsız olarak herhangi bir dosyayı ekleyebilirsiniz. Adreslenen dosya, bir dosya uzantısı içerebilir, ancak buna mecbur değildir.

Aşağıdaki tabloda, TMAIL_C'nin STATUS parametresindeki geri dönüş değerleri gösterilmektedir:

Geri dönüş değeri STATUS* (W#16#...):	Anlamı	Notlar
0000	TMAIL_C'in yürütülmesi hatasız bir şekilde tamamlandı.	TMAIL_C'nin hatasız bir şekilde tamamlanması, gönderilen e-postanın gerçekten yerine ulaştığı anlamına gelmez. Alıcı adreslerinin hatalı şekilde girilmesi, TMAIL_C yordamında bir durum hatası oluşturmaz. Bu durumda, adresleri doğru şekilde girilmiş olsalar bile e-postanın diğer alıcılara gönderilmesi garanti edilemez.
7001	TMAIL_C aktiftir (BUSY = 1).	İlk çağrı: Görev başladı.
7002	TMAIL_C aktiftir (BUSY = 1).	Ara çağrı: Görev başladı.

Geri dönüş değeri STATUS* (W#16#...):	Anlamı	Notlar
8xxx	TMAIL_C yordamının yürütülmesi, dahili olarak çağrılan iletişim yordamlarına ait bir hata koduyla tamamlandı.	Ayrıntılı bilgileri, TCON, TDISCON, TSEND ve TRCV (Sayfa 616) iletişim yordamları için kullanılan STATUS parametresinin açıklamalarında bulabilirsiniz.
8009	Dahili bir fonksiyonda hata	Dahili bir fonksiyon bir hata döndürdü. Ayrıntılı bilgileri, örnek DB'nin SFB_STATUS parametresinde bulabilirsiniz. Olası değerler aşağıda açıklanmıştır.
8010	Bağlantı kurulurken hata	Değerlendirmeyle ilgili ayrıntılı bilgiler, örnek veri bloğunun SFB_STATUS parametresinde mevcuttur. SFB_STATUS parametresinde görüntülenen hata kodu, TCON (Sayfa 616) yordamı için kullanılan STATUS parametresinin açıklama kısmında açıklanmaktadır.
8011	Veriler gönderilirken hata	Değerlendirmeyle ilgili ayrıntılı bilgiler, örnek veri bloğunun SFB_STATUS parametresinde mevcuttur. SFB_STATUS parametresinde görüntülenen hata kodu, TSEND (Sayfa 616) yordamı için kullanılan STATUS parametresinin açıklama kısmında açıklanmaktadır.
8012	Veriler alınırken hata	Değerlendirmeyle ilgili ayrıntılı bilgiler, örnek veri bloğunun SFB_STATUS parametresinde mevcuttur. SFB_STATUS parametresinde görüntülenen hata kodu, TRCV (Sayfa 616) yordamı için kullanılan STATUS parametresinin açıklama kısmında açıklanmaktadır.
8013	Bağlantı kurulurken hata	Değerlendirmeyle ilgili ayrıntılı bilgiler, örnek veri bloğunun SFB_STATUS parametresinde mevcuttur. SFB_STATUS parametresinde görüntülenen hata kodu, TCON (Sayfa 616) ve TDISCON (Sayfa 616) yordamları için kullanılan STATUS parametresinin açıklama kısmında açıklanmaktadır.
8014	Bağlantı kurulamıyor.	Muhtemelen yanlış bir posta sunucusu IP adresi (MailServerAddress (Sayfa 649)) veya bağlantının kurulması için çok kısa bir süre değeri (WatchDogTime (Sayfa 649)) girildi. CPU'nun ağla bağlantısının olmaması veya CPU konfigürasyonunun hatalı olması da mümkündür.
8015	MAIL_ADDR_PARAM için kullanılan veri tipi yanlış	Geçerli veri tipleri sadece Tmail_v4 ve TMail_FQDN sistem veri tipleridir (yapılar).
8016	ATTACHMENT parametresi için yanlış veri tipi	Geçerli veri tiplerini aşağıdaki listede bulabilirsiniz: • ArrayOfChar • ArrayOfByte • ArrayOfWord • ArrayOfDWord • String

Geri dönüş değeri STATUS* (W#16#...):	Anlamı	Notlar
		Not: ArrayOfChar ve String veri tipleri, ancak TMAIL_C yordamının 5.0 sürümünden itibaren geçerlidir.
8017	ATTACHMENT parametresi için yanlış veri uzunluğu	Veri uzunluğu <= 65534 Bayt olmalıdır.
82xx, 84xx, veya 85xx	Hata mesajı posta sunucusundan gelir ve "8" istisna olmak kaydıyla, SMTP protokolündeki hata numarasına karşılık gelir. Aşağıdaki satırlarda olası hata kodları listelenmiştir.	SMTP hata kodu ve SMTP protokolündeki diğer hata kodlarıyla ilgili daha ayrıntılı bilgileri, internette veya posta sunucusunun hata dokümantasyonunda bulabilirsiniz. Ayrıca kendi örnek DB'nizde posta sunucusunun son hata mesajını BUFFER1 parametresinde görüntüleyebilirsiniz. TMAIL_C yordamıyla en son gönderilen verileri örnek DB'de DATEN öğesinin altında bulabilirsiniz.
8450	İşlem yürütülmedi: Posta kutusu kullanılamıyor/posta kutusuna erişilemiyor	İşlemi daha sonra tekrarlayın.
8451	İşlem iptal edildi: Yerel işleme hatası	İşlemi daha sonra tekrarlayın.
8500	Sözdizimi hatası: Hata algılanmadı. Bu, çok uzun bir komut karakter dizisinden kaynaklanan hatalar için de geçerlidir. Hata, e-posta sunucusunun LOGIN kimlik doğrulama işlemini desteklememesi durumunda da ortaya çıkabilir.	TMAIL_C'nin parametrelerini kontrol edin. Bir e-postayı kimlik doğrulaması olmadan göndermeyi deneyin. Bunun için, UserName parametresinin içeriğini boş bir karakter dizisiyle değiştirin. Kullanıcı adı belirtilmezse, LOGIN kimlik doğrulama yöntemi kullanılmaz.
8501	Sözdizimi hatası: Bir parametrede yanlış giriş	Olası neden: TO_S veya CC parametresinde yanlış adres (ayrıca bkz.: TO_S ve CC parametreleri (Sayfa 656)).
8502	Komut bilinmiyor veya uygulanmadı	Başta FROM parametresi olmak üzere yaptığınız girişleri kontrol edin. Muhtemel bu parametre eksiktir ve "@" veya "." karakterini unutmuşsunuzdur (ayrıca bkz.: TO_S ve CC parametreleri (Sayfa 656)).
8535	SMTP kimlik doğrulama eksik	Muhtemelen yanlış bir kullanıcı adı veya yanlış bir şifre girildi.
8550	Posta sunucusuna erişilemiyor. Erişim yetkileriniz yok.	Yanlış bir kullanıcı adı veya yanlış bir şifre girdiniz veya posta sunucusu oturum açma bilgilerinizi desteklemiyor. Diğer bir hata sebebi, TO_S veya CC parametresindeki "@" karakterinden sonra gelen etki alanı adındaki bir hata olabilir (ayrıca bkz.: TO_S ve CC parametreleri (Sayfa 656)).
8552	İşlem iptal edildi: Atanan bellek alanı aşılmış	İşlemi daha sonra tekrarlayın.
8554	Aktarma başarısız oldu	İşlemi daha sonra tekrarlayın.
* Hata kodları, programlama editöründe tamsayılar veya onaltılık değerler olarak görüntülenebilir.		

TMAIL_C yordamının V6.0 veya daha yüksek bir sürümünden itibaren, örnek DB'nin SFB_STATUS parametresinde şu geri dönüş değerlerinin yayınlanması mümkündür:

Örnek DB'nin SFB_STATUS parametresindeki geri dönüş değerleri (W#16#...)	Anlamı
8085	Bağlantı ID'si (ID parametresi) zaten yapılandırılmış bir bağlantı tarafından kullanılmaktadır.
8086	ID parametresi, geçerli aralığın dışında.
8087	Maksimum bağlantı sayısına ulaşıldı; başka bağlantı kurmak mümkün değil.
8088 *	Dosya mevcut değil veya şu anda kullanılamıyor.
8089 *	Eş zamanlı olarak açılan dosya sayısı sistemin limitini aştığı için dosya açılamıyor. S7-1200'de bu limit, dosya sistemi başına 26 dosyadır.
808A *	DirectoryPath DataLogs, Recipes veya UserFiles'tan farklı bir dizin içeriyor veya adreslenen alt dizinlerden biri, adresleme için daha önce açıklanan kuralları ihlal ediyor. Daha fazla bilgi için, bkz. DirectoryPath (Sayfa 657).
808B *	FileName, izin verilmeyen bir karakter dizisi içeriyor veya boş bırakılmış. Daha fazla bilgiyi, FileName (Sayfa 657) kısmında bulabilirsiniz.
808C *	AttachmentName parametresi, bir dosya yolu ek olarak adreslenirken boş olmalıdır.
8092	TO_S ve CC parametreleri boş veya "From" alt parametresi boş veya tam değil.
8093	MAIL_ADDR_PARAM parametresi, bağlantının yükseltilerek güvenli bir bağlantı haline gelmesini gerektiriyor, ancak posta sunucusu STARTTLS yordamını desteklemiyor.
8095	Posta sunucusundan geçersiz yanıt geldi. Posta sunucusu muhtemelen RFC uyumlu değil.
809A	MAIL_ADDR_PARAM parametresindeki SDT yapısı, entegre bir arayüzde desteklenmiyor.
809B	MAIL_ADDR_PARAM parametresindeki SDT'de geçersiz arayüz ID'si.
80A1	Belirtilen bağlantı veya uzak port zaten kullanılıyor.
80A3	ID, kullanıcı programı tarafından oluşturulan bir bağlantı tarafından kullanılıyor.
80A4	Bağlantının uzak uç noktasına ait IP adresi geçersiz veya yerel partnerin IP adresine karşılık geliyor.
80A7	İletişim hatası: TMAIL_C tamamlanmadan önce TDISCON yordamını yürüttünüz.
80B7	Uzak port 0 veya partner uç noktasının IP adresi için 0.0.0.0 ayarlanmış.
80C3	CPU'daki kaynaklar yetmiyor

* Bu hata kodları, teşhisi kolaylaştırmak için TMAIL_C yordamının dosya yolu düzgün bir şekilde adreslenmediğinde eklenir.

Örnek DB'nin SFB_STATUS parametresindeki geri dönüş değerleri (W#16#...)	Anlamı
80C4	Geçici iletişim hatası: • Bağlantı şu anda kurulamıyor. • Bağlantı yolundaki güvenlik duvarları gereken portlar için açık olmadığından bağlantı kurulamıyor. • Arayüz şu anda yeni parametreler alıyor. • Yapılandırılan bağlantı şu anda bir TDISCON yordamı aracılığıyla çıkartılıyor.
80C5	Posta sunucusu bağlantı kurma işlemini reddediyor veya bağlantıyı sonlandırdı veya aktif bir şekilde iptal etti.
80C6	Bağlantı partneri erişilebilir durumda değil (ağ hatası).
80C7	Yürütme sırasında zaman aşımı
80C8	Mevcut bir bağlantının yeniden kurulması deneniyor.
80C9	Bağlantı partnerini doğrulama işlemi başarısız oldu. Posta sunucusu, MailServerAddress parametresindeki belirlenen partnere karşılık gelmiyor.
80CE	Yerel arayüzün IP adresi 0.0.0.0 şeklindedir.
80D0	MailServerAddress parametresi, DNS kullanıldığında boş bir karakter dizisi içeriyor.
80D1	MailServerAddress parametresi tam nitelikli bir etki alanı ad değildir. Muhtemelen sondaki nokta eksik.
80D2	Hiçbir DNS sunucu adresi yapılandırmadınız.
80D3	Tam nitelikli etki alanı adı (FQDN) çözülemedi. Olası nedenler: • DNS sunucusuna erişilemiyor (örnek: DNS sunucusu kapatıldı veya çıkartılan port erişilemez durumda). • DNS sunucusuyla iletişim sırasında bir hata oluştu. • DNS sunucusu geçerli bir DNS yanıtı döndürdü, ancak yanıt bir IPv4 adresi içermiyor.
80E0	Posta sunucusuyla iletişim, hatalar içeren bir mesaj nedeniyle başarısız oldu. Olası nedenler: • Bir mesajın kimlik doğrulama kodu geçersiz • Mesajın şifresi çözülemedi • Bir mesaj açılırken hata • Dahili kapasite taşması
80E1	El sıkışmada (Handshake) hata. Olası nedenler: • Kullanıcı tarafından iptal • Güvenlik seviyesi yeterince yüksek değil. • Uzatılan anlaşma desteklenmiyor • SSL/TLS sürümü desteklenmez. • Şifre çözme hatası

* Bu hata kodları, teşhisi kolaylaştırmak için TMAIL_C yordamının dosya yolu düzgün bir şekilde adreslenmediğinde eklenir.

Örnek DB'nin SFB_STATUS parametresindeki geri dönüş değerleri (W#16#...)	Anlamı
80E2	Sertifika desteklenmiyor / geçersiz Olası neden: İlgili modülün saati ayarlamadı veya modül senkronize edilmedi. Örnek: Modülün tarihi için standart ayar 1/1/2012'dir ve tarih, işletime alma sırasında ayarlanmadı. Sertifikanın geçerlilik süresi 20 Ağustos 2016'da başlıyor ve 20 Ağustos 2024'te bitiyor. Bu durumda modülün tarihi, sertifikanın geçerlilik süresinin dışında kalıyor, dolayısıyla sertifika, modül için geçersiz.
80E3	Posta sunucusu sertifikası iptal edildi.
80E4	Posta sunucusu sertifikası için geçerli bir sertifikalandırma kuruluşu bulunamadı.
80E5	Posta sunucusu sertifikasının süresi doldu.
80E6	Transport-Layer-Security protokolünde (TLS protokolü) bütünlük hatası
80E7	Posta sunucusu sertifikasında desteklenmeyen uzantı
80E9	TLS sunucusu, sunucu sertifikası olmadan desteklenmiyor

* Bu hata kodları, teşhisi kolaylaştırmak için TMAIL_C yordamının dosya yolu düzgün bir şekilde adreslenmediğinde eklenir.

11.5.8.18 UDP

UDP, RFC 768 açıklanan bir standart protokoldür ve User Datagram Protocol'ün (Kullanıcı Datagram Protokolü) kısaltmasıdır. UDP, bir uygulamanın bir datagramı başka bir uygulamaya yollayabileceği şekilde bir yöntem sunar. Ancak verilerin teslim edilip edilmeyeceği henüz garanti edilmedi. Bu protokol aşağıdaki özelliklere sahiptir:

- Hızlı iletişim protokolü
- Küçük ila orta büyüklükteki veri miktarları (maksimum 1472 Bayt) için uygundur
- UDP, ince bir katman ve az yönetim çalışması gerektirmesiyle TCP'den daha basit bir taşıma protokolüdür
- Birçok harici sistemde çok esnek bir şekilde kullanılabilir
- Yönlendirme özellikli
- Datagramları port numaraları üzerinden iletir
- Mesajlar onaylanmaz: Uygulama, hata durumunda geri yükleme ve güvenlik sorumluluğunu üstlenmelidir
- SEND/RECEIVE programlama arayüzü nedeniyle, veri yönetimi için programlama çalışmaları gerekir

UDP, Broadcast iletişimini destekler. Broadcast fonksiyonunu kullanmak için ADDR konfigürasyonunda IP adresinin ilgili kısmını yapılandırın. Örnek: IP adresi 192.168.2.10 ve alt ağ maskesi 255.255.255.0 olan bir CPU'nun Broadcast adresi 192.168.2.255 şeklinde olur.

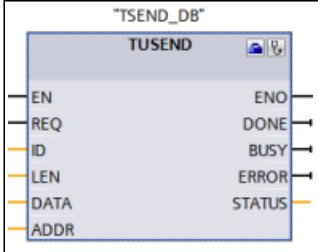
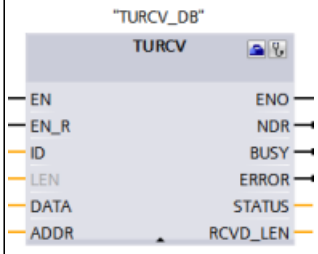
11.5.8.19 TUSEND ve TURCV

Aşağıdaki yordamlar, UDP iletişim prosesine kumanda eder:

- TCON, istemci ile sunucu PC'si (CPU) arasındaki iletişimi kurar.
- TUSEND ve TURCV, verileri gönderir ve alır.
- TDISCON , istemci ile sunucu arasındaki iletişimi keser.

TCON ve TDISCON iletişim yordamlarıyla ilgili daha fazla bilgiyi, "TCP ve ISO on TCP" alt bölümünde TCON, TDISCON, TSEND ve TRCV (Sayfa 616) kısmında bulabilirsiniz.

Tablo 11-46 TUSEND ve TURCV yordamları

LAD/FBD	SCL	Açıklama
	**"TUSEND_DB"(req:=_bool_in_, ID:=_word_in_, len:=_udint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, data:=_variant_inout_);**	TUSEND yordamı, verileri UDP üzerinden, ADDR parametresi tarafından belirtilen bir uzak partnere gönderir. Veri gönderimi görevini başlatmak için TUSEND yordamını REQ = 1 ile çağırın.
	**"TURCV_DB"(en_r:=_bool_in_, ID:=_word_in_, len:=_udint_in_, ndr=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_udint_out_, data:=_variant_inout_);**	TURCV yordamı, verileri UDP üzerinden alır. ADDR parametresi, göndericinin adresini gösterir. TURCV yordamı başarıyla yürütüldükten sonra ADDR parametresi, uzak partnerin (göndericinin) adresini içerir. TURCV, geçici modu desteklemiyor. Verilerin alınması görevini başlatmak için TURCV yordamını EN_R = 1 ile çağırın.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

TCON, TDISCON, TUSEND ve TURCV, bir görevin düzenlenmesi birden fazla yordam yürütme işlemini kapsayacak şekilde asenkron olarak çalışır.

Tablo 11-47 TUSEND ve TURCV parametreleri için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ (TUSEND)	IN	Bool	Yükselen bir kenar durumunda gönderme görevini başlatır. Veriler, DATA ve LEN ile tanımlanan aralıktan aktarılır.
EN_R (TURCV)	IN	Bool	• 0: CPU alamıyor. • 1: CPU'yu alma işlemi için etkinleştirir. TURCV yordamı almaya hazırdır ve alma görevi düzenlenir.
ID	IN	Word	Kullanıcı programı ile işletim sisteminin iletişim katmanı arasındaki ilgili bağlantıya verilen referans. ID, yerel bağlantı açıklamasındaki ilgili parametre ID'si ile eşleşmelidir. Değer aralığı: W#16#0001 - W#16#0FFF.
LEN	IN	UDInt	Gönderilmesi (TUSEND) veya alınması (TURCV) gereken Bayt'ların sayısı. • Standart = 0. DATA parametresi, gönderilmesi veya alınması gereken verilerin uzunluğunu belirler. • Aksi takdirde, değer aralığı: 1 ila 1472

Parametre ve veri tipi		Veri tipi	Açıklama
DONE (TUSEND)	IN	Bool	DONE durum parametresi (TUSEND): • 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
NDR (TURCV)	OUT	Bool	NDR durum parametresi (TURCV): • 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev başarıyla gerçekleştirildi.
BUSY	OUT	Bool	• 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz. • 0: Görev tamamlandı.
ERROR	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Hata yok • 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS	OUT	Word	Durum bilgileri, hata bilgilerini kapsar. (Bkz. alt kısımdaki "Hata ve durum için kullanılan koşul kodları" tablosu.)
RCVD_LEN	OUT	UDInt	Alınan Bayt sayısı (TURCV)
DATA	IN_OUT	Variant	Gönderme alanının (TUSEND) veya alma alanının (TURCV) adresi: • girişlerin proses görünümü • çıkışların proses görünümü • bir markör Bit'i • bir veri bloğu
ADDR	IN_OUT	Variant	İşaretçiyi alıcının (TUSEND'de) veya göndericinin (TURCVde) adresinin üzerine götürün (örn. P#DB100.DBX0.0 Bayt 8) İmleç, herhangi bir bellek alanına referans verebilir. Aşağıda gösterildiği gibi 8 Bayt'lık bir yapı gerekir: • İlk 4 Bayt, desantral IP adresini içerir. • Sonraki 2 Bayt, desantral port numarasını belirtir. • Son 2 Bayt rezerve edildi.

Görev durumu, BUSY ve STATUS çıkış parametrelerinde görüntülenir. STATUS, asenkron çalışan yordamlarda RET_VAL çıkış parametresine karşılık gelir.

Aşağıdaki tabloda, BUSY, DONE (TUSEND), NDR (TURCV) ve ERROR arasındaki ilişkiler gösterilmektedir. Bu tablonun yardımıyla, yordamın (TUSEND veya TURCV) güncel durumunu veya gönderme işleminin (aktarma) veya alma işleminin tamamlandığı anı belirleyebilirsiniz.

Tablo 11-48 BUSY, DONE (TUSEND)/NDR (TURCV) ve ERROR parametrelerinin durumu

BUSY	DONE / NDR	ERROR	Açıklama
DOĞRU	İlgili değil	İlgili değil	Görev düzenlenir.
YANLIŞ	DOĞRU	YANLIŞ	Görev, başarıyla gerçekleştirildi.
YANLIŞ	YANLIŞ	DOĞRU	Görev hatayla sonlandı. Hata sebebi, STATUS parametresine kaydedildi.
YANLIŞ	YANLIŞ	YANLIŞ	Yordama, (yeni) bir görev atanmadı.

[1] Yordamların asenkron fonksiyonu nedeniyle: TUSEND'de gönderme alanındaki veriler, DONE veya ERROR parametrelerinden biri DOĞRU değerini alana kadar tutarlı şekilde kalmalıdır. TURCV'de alma alanındaki veriler, ancak NDR parametresinin DOĞRU değerine sahip olması durumunda tutarlı olur.

Tablo 11-49 TUSEND ve TURCV'nin ERROR ve STATUS için koşul kodları

ERROR	STATUS	Açıklama
0	0000	• Gönderme görevi hatasız bir şekilde gerçekleştirildi (TUSEND). • Yeni veriler kabul edildi. Alınan verilerin güncel uzunluğu, RCVD_LEN 'de (TURCV) gösterilir.
0	7000	• Devam eden bir görev işlemesi yok (TUSEND) • Blok, almaya hazır değil (TURCV)
0	7001	• Görev işlemesi başlatılır, veriler gönderilir (TUSEND): Bu düzenleme işlemi sırasında işletim sistemi, DATA gönderme alanındaki verilere erişir. • Blok almaya hazır, alma görevi etkinleştirildi (TURCV).
0	7002	• Müteakip çağrı (REQ ilgili değil), görev düzenlenir (TUSEND): Bu düzenleme işlemi sırasında işletim sistemi, DATA gönderme alanındaki verilere erişir. • Müteakip çağrı, görev düzenlenir: Düzenleme sırasında TURCV yordamı, verileri alma alanına yazar. Bu nedenle bir hata alma alanında tutarsız verilerin olmasına yol açabilir.
1	8085	LEN parametresi, izin verilen en büyük değerden büyüktür, 0 değerine sahiptir (TUSEND) veya LEN ya da DATA parametresinin değerini ilk yordam yürütme işleminden beri değiştirdiniz (TURCV).
1	8086	ID parametresi izin verilen adres aralığında değil.
1	8088	• LEN parametresi, DATA'da belirlenen bellek alanından (TUSEND) veya alma alanından (TURCV) büyüktür. • Alma alanı çok küçük (TURCV).
1	8089	ADDR parametresi bir veri bloğuna referans vermiyor.
1	80A1	İletişim hatası: • Kullanıcı programı ile işletim sisteminin iletişim katmanı arasındaki belirtilen bağlantı henüz kurulmadı. • Kullanıcı programı ile işletim sisteminin iletişim katmanı arasındaki belirtilen bağlantı şu anda sonlandırıldı. Bu bağlantı üzerinden bir gönderme (TUSEND) veya alma görevinin (TURCV) gerçekleştirilmesi mümkün değil. • Arayüz yeniden başlatılır.
1	80A4	Çıkartılan bağlantı noktasının IP adresi geçersiz, muhtemelen yerel IP adresine karşılık geliyor (TUSEND).
1	80B3	• Belirlenen protokol varyantı (bağlantı açıklamasındaki connection_type parametresi) UDP değildir. Lütfen TSEND veya TRCV yordamını kullanın. • ADDR parametresi: Port numarası için geçersiz ayarlar (TUSEND)
1	80C3	• Bu ID'ye sahip bir blok, başka bir öncelik sınıfında düzenleniyor. • Kaynaklarda dahili kusur
1	80C4	Geçici iletişim hatası: • Kullanıcı programı ile işletim sisteminin iletişim katmanı arasındaki bağlantı şu anda kurulamıyor (TUSEND). • Arayüz yeni parametreler alıyor (TUSEND). • Belirtilen bağlantı şu anda yeniden kuruluyor (TURCV).

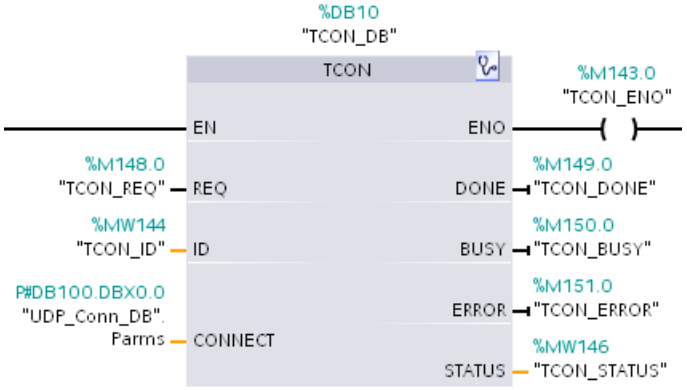
Ethernet bağlantı protokolleri

Her CPU, standart PROFINET iletişimini destekleyen entegre bir PROFINET bağlantısına sahiptir. TUSEND ve TURCV yordamları UDP Ethernet protokolünü destekliyor.

Daha fazla bilgiyi, "Cihaz konfigürasyonu" bölümündeki "Yerel CPU ile partner CPU'su arasındaki bağlantı yolunun yapılandırılması" (Sayfa 563) başlığında bulabilirsiniz.

İşletme

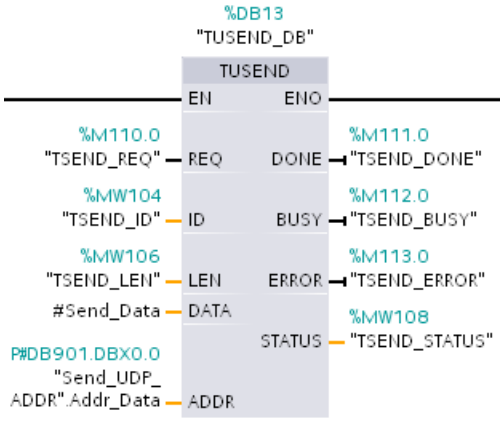
Her iki partner de UDP iletişiminde pasif. "TCON_Param" veri tipinin parametreleri için tipik başlangıç değerleri aşağıdaki şekillerde gösterilmektedir. Port numaraları (LOCAL_TSAP_ID) 2 Bayt formatında yazılmıştır. 161, 34962, 34963 ve 34964 istisna olmak kaydıyla tüm portlara izin verilir.



UDP_Conn_DB

	Name	Data type	Offset	Initial value	Comment
1	Static				
2	Parms	TCON_Param	0.0		
3	BLOCK_LENGTH	UInt	0.0	64	byte length of SDT
4	ID	CONN_OUC	2.0	1	reference to the connection
5	CONNECTION_TYPE	USInt	4.0	19	17: TCP/IP; 18: ISO on TCP
6	ACTIVE_EST	Bool	5.0	false	active/passive connection establishment
7	LOCAL_DEVICE_ID	USInt	6.0	1	1: local IE interface
8	LOCAL_TSAP_ID_L...	USInt	7.0	2	byte length of local TSAP id/port number
9	REM_SUBNET_ID_...	USInt	8.0	0	byte length of remote subnet id
10	REM_STADDR_LEN	USInt	9.0	0	byte length of remote IP address
11	REM_TSAP_ID_LEN	USInt	10.0	0	byte length of remote port/TSAP id
12	NEXT_STADDR_LEN	USInt	11.0	0	byte length of next station address
13	LOCAL_TSAP_ID	Array[1..16] of Byte	12.0		TSAP id/local port number
14	LOCAL_TSAP_I...	Byte		B#16#07	
15	LOCAL_TSAP_I...	Byte		B#16#D0	

TUSEND yordamı, verileri UDP üzerinden TADDR_Param veri tipi tarafından belirtilen uzak partnere gönderir. TURCV yordamı, verileri UDP üzerinden alır. TURCV yordamı başarıyla yürütüldükten sonra "TADDR_Param" veri tipi size uzak partnerin (gönderici) adresini gösterir.



Send_UDP_ADDR

	Name	Data type	Offset	Initial value	Comment
1	Static				
2	Addr_Data	TADDR_Param	0.0		
3	REM_IP_ADDR	Array[1..4] of USint	0.0		remote station address
4	REM_IP_ADDR[1]	USInt		192	
5	REM_IP_ADDR[2]	USInt		168	
6	REM_IP_ADDR[3]	USInt		2	
7	REM_IP_ADDR[4]	USInt		10	
8	REM_PORT_NR	UInt	4.0	2000	remote port number
9	RESERVED	Word	6.0	0	ununsed; has to be 0

11.5.8.20 T_CONFIG

T_CONFIG yordamı; Ethernet adresini, PROFINET cihaz adını veya saat senkronizasyonu için kullanılan NTP sunucusunun IP adreslerini kullanıcı programı aracılığıyla değiştirebilir. Aşağıdaki fonksiyonlar, sürekli veya geçici olarak uyarlanabilir:

- IP adresi
- Alt ağ maskesi
- Yönlendirici adresi
- İstasyon adı
- Maksimum dört NTP sunucusunun IP adresleri

NOT

Programı CPU'ya yükledikten sonra, CPU'nun "Özellikler" bölümündeki "Ethernet adresi" sayfasında, "IP adresi doğrudan cihazda ayarlanır" (Sayfa 675) opsiyon alanı aracılığıyla IP adresini çevrimiçi olarak veya "T_CONFIG" yordamı üzerinden değiştirebilirsiniz.

Programı CPU'ya yükledikten sonra, CPU'nun "Özellikler" bölümündeki "Ethernet adresi" sayfasında, "PROFINET cihaz adı doğrudan cihazda ayarlanır" (Sayfa 677) opsiyon alanı aracılığıyla PROFINET cihaz adını çevrimi olarak veya "T_CONFIG" yordamı üzerinden değiştirebilirsiniz.

CPU'nun "Özellikler" bölümündeki "Saat senkronizasyonu" sayfasında yer alan "Saat senkronizasyonunu NTP sunucusu üzerinden etkinleştir" (Sayfa 678) alanı aracılığıyla maksimum dört NTP sunucusunun IP adreslerini değiştirebilirsiniz.

NOT

Birden fazla T_CONFIG yordamını eş zamanlı olarak yürütemezsiniz.

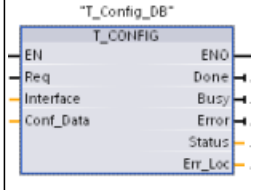
NOT

IP adresindeki veya CPU istasyonunun adındaki değişiklikler geçici veya sürekli olabilir. NTP sunucularının IP adreslerindeki değişiklikler sadece geçici olabilir:

– Sürekli bir değişiklik, ilgili değişikliğin kalıcı olduğunu belirtir, bu da değişikliğin gerilim kesintisinden sonra da mevcut olacağı anlamına gelir.

– Geçici bir değişiklik, ilgili değişikliğin kısa süreli olduğunu ve gerilim kesildikten sonra tekrar başlangıçtaki değerin kabul edileceğini belirtir.

Tablo 11-50 T_CONFIG yordamı

LAD/FBD	SCL	Açıklama
	**"T_CONFIG_DB"(** **Req:=_bool_in_,** **Interface:=_uint_in_,** **Conf_Data:=_variant_in_,** **Done=>_bool_out_,** **Busy=>_bool_out_,** **Error=>_bool_out_,** **Status=>_dword_out_,** **Err_Loc=>_dword_out_);**	T_CONFIG yordamı, IP konfigürasyonunun parametrelerini kullanıcı programı aracılığıyla değiştirir. T_CONFIG asenkron çalışıyor. Yürütme süreci, birkaç çağrı boyunca gerçekleşir.

Tablo 11-51 Parametreler için kullanılan T_CONFIG veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	Input	Bool	Yordamı yükselen kenarda başlatır.
INTERFACE	Input	HW_Interface	Ağ arayüzünün ID'si
CONF_DATA	Input	Variant	Konfigürasyon verilerinin yapısına verilen referans; CONF_DATA maksimum dört sistem veri tipine (SDT) sahip bir yapı aracılığıyla belirlenir.
DONE	Output	Bool	• 0: Görev henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev hatasız bir şekilde gerçekleştirildi.
BUSY	Output	Bool	• 0: Görev tamamlandı. • 1: Görev henüz tamamlanmadı. Yeni bir görev başlatılamaz.
ERROR	Output	Bool	Aşağıdaki değerlere sahip durum parametreleri: • 0: Hata yok • 1: İşleme sırasında hata oluştu. STATUS, hata tipiyle ilgili bilgileri içeriyor.
STATUS	Output	DWord	Durum bilgileri, hata bilgilerini kapsar. (Bkz. alt kısımdaki "Hata ve durum için kullanılan koşul kodları" tablosu.)
ERR_LOC	Output	DWord	Hata konumu (CONF_DATA yapısındaki alan ID'si ve alt alan adresi)

IP konfigürasyon verileri, yukarda belirtilen CONF_DATA parametresinin üzerine getirilen bir Variant işaretçisiyle birlikteCONF_DATA veri bloğuna kaydedilir. T_CONFIG yordamının başarılı şekilde yürütülmesi, IP konfigürasyon verilerinin ağ arayüzüne aktarılmasıyla sona erer.

"T_CONFIG" yordamının durum ve hata mesajları STATUS ve ERR_LOC parametrelerinde yayınlanır:

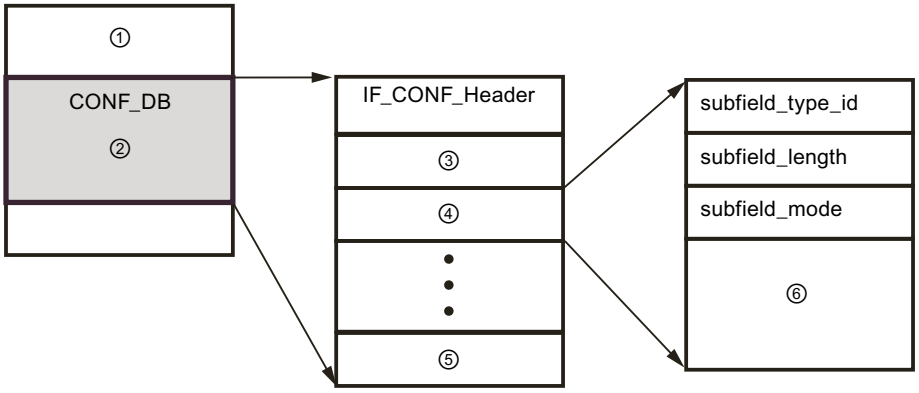
- Hata sebebi, STATUS parametresinde yayınlanır.
- Hata konumu, ERR_LOC parametresinde yayınlanır. Burada şu opsiyonlar mevcuttur:
  - 16#0000_0000: Yordam çağrılırken hata yok veya hata var (örn. yordam parametrelendirilirken veya PROFINET arayüzüyle iletişim sırasına hata).
  - 16#0001_0000: IF_CONF_HEADER sistem veri tipinin parametrelerindeki konfigürasyon verilerinde hat.
  - 16#0001_000x: IF_CONF_V4 veya IF_CONF_NOS ya da IF_CONF_NTP sistem veri tipinin parametrelerindeki konfigürasyon verilerinde hata (x, T_CONFIG yapısındaki hatalı kısmi bloğun pozisyonunu belirtir. T_CONFIG yapısı örneğin IP adresi için kullanılan bir kısmi blok ve istasyon adı için kullanılan bir kısmi blok içeriyorsa ve hata, istasyon adı için kullanılan kısmi blokta bulunuyorsa, ERR_LOC'un değeri, 0001_0002 olur.)

Aşağıdaki tabloda, STATUS ve ERR_LOC parametreleri için olası değerler gösterilmektedir:

STATUS*	ERR_LOC*	Anlamı
0000_0000	0000_0000	Görev işlemesi hatasız bir şekilde gerçekleştirildi.
0070_0000	0000_0000	Devam eden bir görev işlemesi yok.
0070_0100	0000_0000	Görev işlemesinin başlangıcı.
0070_0200	0000_0000	Ara çağrı (REQ ilgili değil).
C08x_yy00	0000_0000	Genel hata bilgileri.

STATUS*	ERR_LOC*	Anlamı
C080_8000	0000_0000	Yordam çağrılırken hata: Interface parametresindeki donanım ID'si geçersizdir.
C080_8100	0000_0000	Yordam çağrılırken hata: Interface parametresindeki donanım ID'si hiçbir PROFINET arayüzünü adreslemiyor.
C080_8700	0000_0000	Yordam çağrılırken hata: CONF_DATA parametresindeki veri bloğunun uzunluğu yanlış.
C080_8800	0001_0000	IF_CONF_HEADER sistem veri tipinde hata: FieldType parametresi geçersiz bir değere sahip. FieldType için 0 değerini kullanın.
C080_8900	0001_0000	IF_CONF_HEADER sistem veri tipinde hata: FieldId parametresi geçersiz bir değere sahip veya çok kez kullanılmış. FieldId için 0 değerini kullanın.
C080_8A00	0001_0000	IF_CONF_HEADER sistem veri tipinde hata: SubfieldCount parametresinde yanlış sayı. Kullanılan IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tiplerinin doğru sayısını girin.
C080_8B00	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: Id parametresi geçersiz bir değere sahip. IF_CONF_V4 için 30 değerini, IF_CONF_NOS için 40 değerini, IF_CONF_NTP için 17 değerini kullanın.
C080_8C00	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: Yanlış sistem veri tipi kullanılmış, sıra yanlış veya bir sistem veri tipi çok kez kullanılmış.
C080_8D00	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: Length parametresi yanlış veya geçersiz bir değere sahip.
C080_8E00	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: Mode parametresi yanlış veya geçersiz bir değere sahip. • IF_CONF_V4 ve IF_CONF_NOS 'ta sadece 1 (sürekli) veya 2 (geçici) değerlerine izin verilir. • IF_CONF_NTP'de sadece 2 değerine (geçici) izin veriliyor.
C080_9000	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: Konfigürasyon verileri aktarılamıyor. Olası neden: • IF_CONF_V4'te: Donanım konfigürasyonunda "IP adresini cihazda ayarla" ayarı seçilmedi. • IF_CONF_NOS'de: Donanım konfigürasyonunda "PROFINET cihaz adlarını cihazda ayarla" ayarı seçilmedi. • IF_CONF_NTP'de: Donanım konfigürasyonunda "Saat senkronizasyonunu NTP sunucusu üzerinden etkinleştir" ayarı seçilmedi ve NTP sunucusu için hiçbir IP adresi ayarlanmadı.
C080_9400	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: Bir parametre değeri belirlenmedi veya geçersiz.
C080_9500	0001_000x	IF_CONF_V4, IF_CONF_NOS veya IF_CONF_NTP sistem veri tipinde hata: İki parametrenin değerleri tutarsız.
C080_C200	0000_0000	Yordam çağrılırken hata: Konfigürasyon verileri aktarılamıyor. Olası neden: PROFINET arayüzüne erişilemiyor.
C080_C300	0000_0000	Yordam çağrılırken hata: Kaynaklar yeterli değil (örn. "T_CONFIG"den farklı parametrelerle çok sayıda çağrı mevcut).
C080_C400	0000_0000	Yordam çağrılırken hata: Geçici iletişim hatası. Yaz saatine geçiş için kullanılan saat göstergesi.
C080_D200	0000_0000	Yordam çağrılırken hata: Çağırmak mümkün değil. Yordam, PROFINET arayüzü tarafından desteklenmiyor.

Aşağıdaki şekilde, aktarılması gereken konfigürasyon verilerinin konfigürasyon DB'ye nasıl kaydedileceği gösterilmektedir.



① Konfigürasyon DB
② Konfigürasyon verileri
③ Alt alan 1
④ Alt alan 2
⑤ Alt alan *n*
⑥ Alt alana özel parametreler

CONF_DB'nin konfigürasyon verileri, bir Header (IF_CONF_Header) içeren bir alandan ve çok sayıda alt alandan oluşmaktadır. IF_CONF_Header, şu elemanları sağlar:

- field_type_id (veri tipi UInt): Sıfır
- field_id (veri tipi UInt): Sıfır kalibrasyonu
- subfield_cnt (veri tipi UInt): Alt alan sayısı

Her alt alan bir Header (subfield_type_id, subfield_length, subfield_mode) ve alt alana özel parametrelerden oluşmaktadır. Her alt alan, tek sayıda Bayt'ı kapsamalıdır. subfield_mode, 1 ve 2 değerlerini destekleyebilir (bkz. aşağıdaki tablolar).

NOT

Şu anda sadece bir alan (IF_CONF_Header) izin veriliyor. field_type_id ve field_id parametreleri 0 değerine sahip olmalıdır. field_type_id ve field_id için başka değerler içeren diğer alanlar gelecekte yapılacak genişletmelere tâbidir.

Tablo 11-52 Desteklenen alt alanlar

subfield_type_id	Veri tipi	Anlamı
30	IF_CONF_V4	IP parametreleri: IP adresi, alt ağ maskesi, yönlendirici adresi
40	IF_CONF_NOS	PROFINET IO cihazı adı (Name of station)
17	IF_CONF_NTP	Network Time Protocol (NTP)

Tablo 11-53 IF_CONF_V4 veri tipinin elemanları

Adı	Veri tipi	Başlangıç değeri	Açıklama
Id	UInt	30	subfield_type_id
Length	UInt	18	subfield_length
Mode	UInt	0	subfield_mode (1: Sürekli veya 2: geçici)
InterfaceAddress	IP_V4	-	Arayüz adresi
ADDR	Array [1..4] of Byte		

Adı		Veri tipi	Başlangıç değeri	Açıklama
	ADDR[1]	Byte	0	IP adresinin High Bayt'ı: 200
	ADDR[2]	Byte	0	IP adresinin High Bayt'ı: 12
	ADDR[3]	Byte	0	IP adresinin Low Bayt'ı: 1
	ADDR[4]	Byte	0	IP adresinin Low Bayt'ı: 144
SubnetMask		IP_V4	-	Alt ağ maskesi
ADDR		Array [1..4] of Byte		
	ADDR[1]	Byte	0	Alt ağ maskesinin High Bayt'ı: 255
	ADDR[2]	Byte	0	Alt ağ maskesinin High Bayt'ı: 255
	ADDR[3]	Byte	0	Alt ağ maskesinin Low Bayt'ı: 255
	ADDR[4]	Byte	0	Alt ağ maskesinin Low Bayt'ı: 0
DefaultRouter		IP_V4	-	Standart yönlendirici
ADDR		Array [1..4] of Byte		
	ADDR[1]	Byte	0	Yönlendiricinin High Bayt'ı: 200
	ADDR[2]	Byte	0	Yönlendiricinin High Bayt'ı: 12
	ADDR[3]	Byte	0	Yönlendiricinin Low Bayt'ı: 1
	ADDR[4]	Byte	0	Yönlendiricinin Low Bayt'ı: 1

Tablo 11-54 IF_CONF_NOS veri tipinin elemanları

Adı	Veri tipi	Başlangıç değeri	Açıklama
Id	UInt	40	subfield_type_id
Length	UInt	246	subfield_length
Mode	UInt	0	subfield_mode (1: Sürekli veya 2: geçici)
NOS (Name of station)	Array[1..240] of Byte	0	İstasyon adı: ARRAY'i ilk Bayt'tan itibaren doldurmanız gerekir. ARRAY atanması gereken istasyon adından uzunsa, gerçek istasyon adından sonra bir sıfır Bayt'ı girmeniz gerekir (IEC 61158-6-10'a uygun olarak). Aksi takdirde NOS reddedilir ve "T_CONFIG (Sayfa 669)" yordamı STATUS öğesine DW#16#C0809400 hata kodunu girer. İlk Bayt'ı 0 ile doldurursanız, istasyon adı silinir.

İstasyon adı aşağıdaki kısıtlamalara tabidir:

- İstasyon adının içindeki bir ad bileşeni, yani iki nokta arasındaki bir karakter dizisi 63 karakterden uzun olamaz.
- Harflerin üzerine konan çift noktalar, parantezler, alt çizgiler, bölü işaretleri, boşluk karakterlerine, vb. izin verilmez. İzin verilen tek özel karakter tiredir.
- İstasyon adı "-" karakteriyle başlayamaz ve bitemez.
- İstasyon adı bir sayıyla başlayamaz.

- İstasyon adları için n.n.n.n formatına (n = 0 ... 999) izin verilmez.
- İstasyon adı "Port-xyz" veya "Port-xyz-abcde" (a, b, c, d, e, x, y, z = 0 ... 9) karakter dizisiyle başlayamaz.

NOT

240 Bayt'tan daha kısa bir ARRAY "NOS" de oluşturabilirsiniz. Ancak minimum uzunluk 2 Bayt'tır. Ardından "Length" değişkenini (alt alanın uzunluğu) uygun şekilde uyarlamanız gerekir.

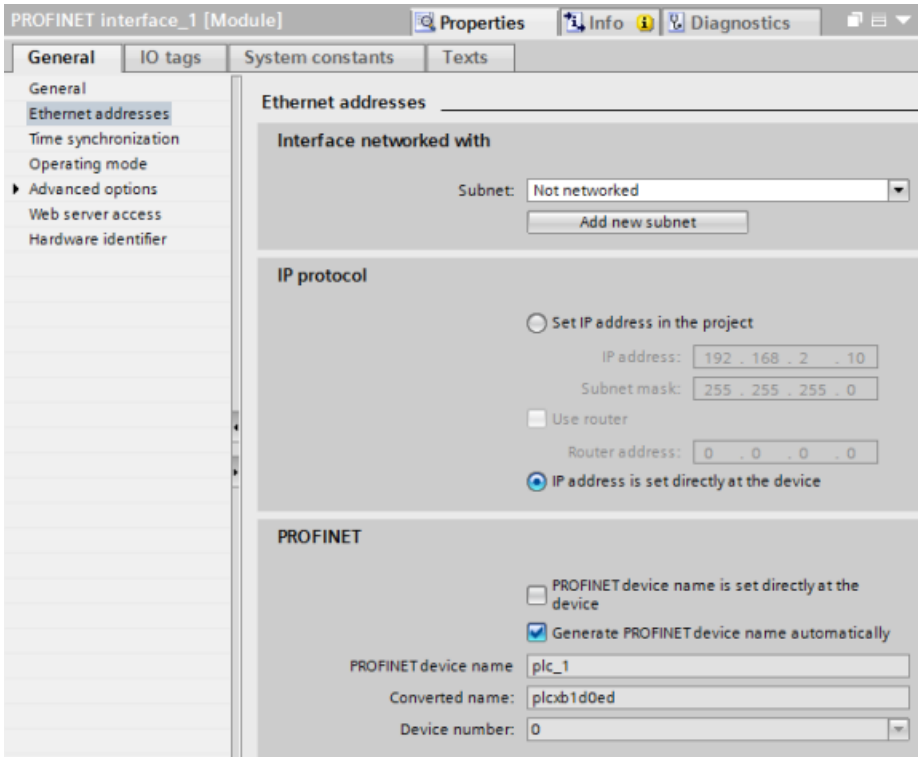
Tablo 11-55 IF_CONF_NTP veri tipinin elemanları

Adı	Veri tipi	Başlangıç değeri	Açıklama
Id	UInt	17	subfield_type_id
Length	UInt	22	subfield_length
Mode	UInt	0	subfield_mode (2: geçici)
NTP_IP	Array[1...4] of IP_V4	-	NTP sunucularının IP adresleri
NTP_IP[1]	IP_V4		NTP sunucusu 1'in IP adresleri
ADDR	Array[1...4] of Byte	0	
ADDR[1]	Byte	0	IP adresinin High Bayt'ı
ADDR[2]	Byte	0	IP adresinin High Bayt'ı
ADDR[3]	Byte	0	IP adresinin Low Bayt'ı
ADDR[4]	Byte	0	IP adresinin Low Bayt'ı
NTP_IP[2]	IP_V4		NTP sunucusu 2'nin IP adresleri
ADDR	Array[1...4] of Byte	0	
ADDR[1]	Byte	0	IP adresinin High Bayt'ı
ADDR[2]	Byte	0	IP adresinin High Bayt'ı
ADDR[3]	Byte	0	IP adresinin Low Bayt'ı
ADDR[4]	Byte	0	IP adresinin Low Bayt'ı
NTP_IP[3]	IP_V4		NTP sunucusu 3'ün IP adresleri
ADDR	Array[1...4] of Byte	0	
ADDR[1]	Byte	0	IP adresinin High Bayt'ı
ADDR[2]	Byte	0	IP adresinin High Bayt'ı
ADDR[3]	Byte	0	IP adresinin Low Bayt'ı
ADDR[4]	Byte	0	IP adresinin Low Bayt'ı
NTP_IP[4]	IP_V4		NTP sunucusu 4'ün IP adresleri
ADDR	Array[1...4] of Byte	0	
ADDR[1]	Byte	0	IP adresinin High Bayt'ı
ADDR[2]	Byte	0	IP adresinin High Bayt'ı
ADDR[3]	Byte	0	IP adresinin Low Bayt'ı
ADDR[4]	Byte	0	IP adresinin Low Bayt'ı

Aşağıdaki örnekte, "addr" alt alanında "InterfaceAddress" (IP adresi), "SubnetMask" ve "DefaultRouter" (IP yönlendirici) parametreleri değiştirilmektedir. Programı CPU'ya yükledikten sonra, CPU'nun "Özellikler" bölümündeki "Ethernet adresi" sayfasında, "T_CONFIG" yordamı üzerinden IP parametrelerini değiştirmek için, "IP adresi doğrudan cihazda ayarlanır" opsiyon alanını kullanabilirsiniz.

CONF_DATA_1

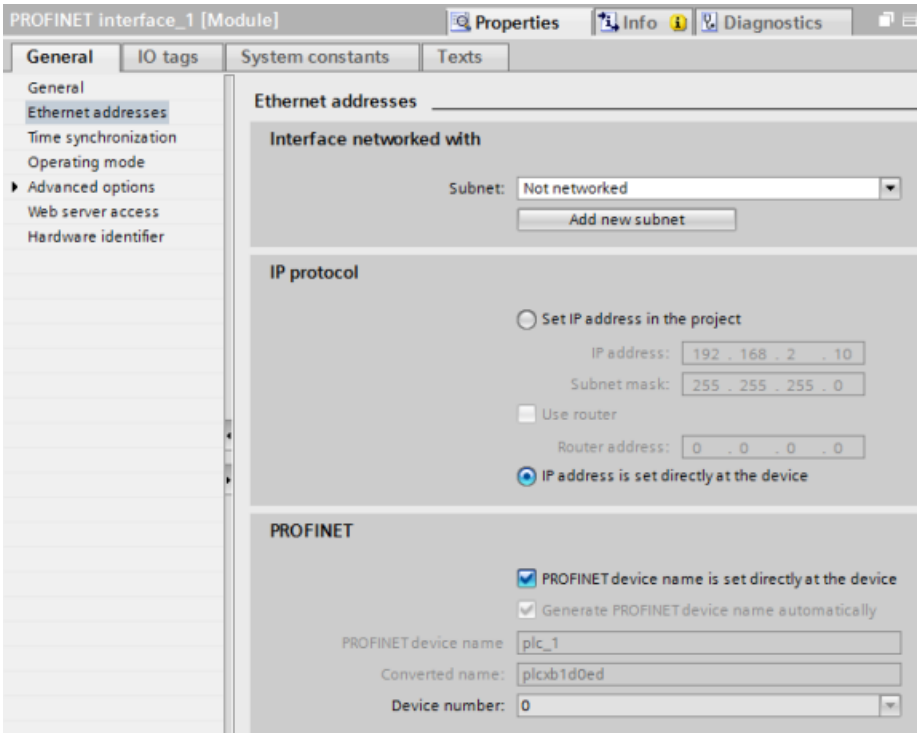
	Name	Data type	Start value
1	Static		
2	Conf_data	Struct	
3	header	IF_CONF_Header	
4	FieldType	UInt	0
5	FieldId	UInt	0
6	SubfieldCount	UInt	1
7	addr	IF_CONF_v4	
8	Id	UInt	30
9	Length	UInt	18
10	Mode	UInt	1
11	InterfaceAddress	IP_V4	
12	ADDR	array [1..4] of Byte	
13	ADDR[1]	Byte	192
14	ADDR[2]	Byte	168
15	ADDR[3]	Byte	2
16	ADDR[4]	Byte	30
17	SubnetMask	IP_V4	
18	ADDR	array [1..4] of Byte	
19	ADDR[1]	Byte	255
20	ADDR[2]	Byte	255
21	ADDR[3]	Byte	255
22	ADDR[4]	Byte	0
23	DefaultRouter	IP_V4	
24	ADDR	array [1..4] of Byte	
25	ADDR[1]	Byte	192
26	ADDR[2]	Byte	168
27	ADDR[3]	Byte	2
28	ADDR[4]	Byte	1



Aşağıdaki örnekte, her iki alt alan "addr" ve "nos" (Name of station) değiştirilmektedir. Programı CPU'ya yükledikten sonra, CPU'nun "Özellikler" bölümündeki "Ethernet adresi" sayfasında, "T_CONFIG" yordamı üzerinden PROFINET cihaz adını değiştirmek için, "PROFINET cihaz adı doğrudan cihazda ayarlanır" opsiyon alanını kullanabilirsiniz.

CONF_DATA_2

	Name	Data type	Start value
1	Static		
2	Conf_data	Struct	
3	header	IF_CONF_Header	
4	FieldType	UInt	0
5	FieldId	UInt	0
6	SubfieldCount	UInt	2
7	addr	IF_CONF_v4	
8	Id	UInt	30
9	Length	UInt	18
10	Mode	UInt	1
11	InterfaceAddress	IP_V4	
12	ADDR	array [1..4] of Byte	
13	SubnetMask	IP_V4	
14	ADDR	array [1..4] of Byte	
15	DefaultRouter	IP_V4	
16	ADDR	array [1..4] of Byte	
17	nos	IF_CONF_NOS	
18	Id	UInt	40
19	Length	UInt	246
20	Mode	UInt	1
21	NOS	array [1..240] of Byte	

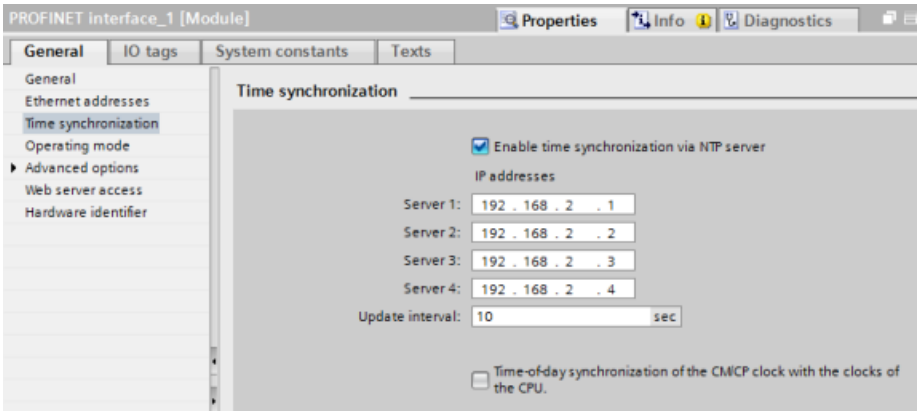

PROFINET interface_1 [Module]
Properties
Info
Diagnostics
General
IO tags
System constants
Texts
General
Ethernet addresses
Time synchronization
Operating mode
Advanced options
Web server access
Hardware identifier
Ethernet addresses
Interface networked with
Subnet:
Not networked
Add new subnet
IP protocol
Set IP address in the project
IP address:
192 . 168 . 2 . 10
Subnet mask:
255 . 255 . 255 . 0
Use router
Router address:
0 . 0 . 0 . 0
IP address is set directly at the device
PROFINET
PROFINET device name is set directly at the device
Generate PROFINET device name automatically
PROFINET device name
plc_1
Converted name:
plcxb1d0ed
Device number:
0

Aşağıdaki örnekte, "ntp" (Network-Time-Protocol - Ağ Zaman Protokolü sunucusu (NTP)) alt alanındaki T_CONFIG yordamı, maksimum dört NTP sunucusunun IP adreslerini değiştirmektedir.

CPU'nun, PROFINET arayüzü [X1]'de, "Özellikler" alanındaki "Saat senkronizasyonu" sayfasında, "NTP sunucusu üzerinden saat senkronizasyonunu etkinleştir" opsiyon kutusunu etkinleştirerek NTP senkronizasyonunu yapılandırın (aşağıdaki şekle bakın). Bunun ardından programı CPU'ya yükledikten sonra, T_CONFIG yordamı ile NTP sunucularının IP adreslerini değiştirebilirsiniz.

CONF_DATA_3

	Name	Data type	Start value
1	Static		
2	Conf_Data	Struct	
3	header	IF_CONF_Header	
4	FieldType	UInt	0
5	FieldId	UInt	0
6	SubfieldCount	UInt	1
7	ntp	IF_CONF_NTP	
8	Id	UInt	17
9	Length	UInt	22
10	Mode	UInt	2
11	NTP_IP	Array[1..4] of IP_V4	
12	NTP_IP[1]	IP_V4	
13	ADDR	Array[1..4] of Byte	
14	ADDR[1]	Byte	192
15	ADDR[2]	Byte	168
16	ADDR[3]	Byte	2
17	ADDR[4]	Byte	5
18	NTP_IP[2]	IP_V4	
19	ADDR	Array[1..4] of Byte	
20	ADDR[1]	Byte	192
21	ADDR[2]	Byte	168
22	ADDR[3]	Byte	2
23	ADDR[4]	Byte	6
24	NTP_IP[3]	IP_V4	
25	ADDR	Array[1..4] of Byte	
26	ADDR[1]	Byte	192
27	ADDR[2]	Byte	168
28	ADDR[3]	Byte	2
29	ADDR[4]	Byte	7
30	NTP_IP[4]	IP_V4	
31	ADDR	Array[1..4] of Byte	
32	ADDR[1]	Byte	192
33	ADDR[2]	Byte	168
34	ADDR[3]	Byte	2
35	ADDR[4]	Byte	8



11.5.8.21 Yordamlar için ortak parametreler

Giriş parametresi REQ

Açık kullanıcı iletişimi için kullanılan yordamların çoğu, yordamı bir yükselen kenarda (0'dan 1'e) başlatan bir REQ girişine sahiptir. Bir yordamın yürütülmesi sırasında REQ girişi 1 (DOĞRU) olmalıdır, ancak REQ girişi istenildiği kadar DOĞRU kalabilir. Yordam, REQ girişi = YANLIŞ ile yürütülene kadar başka bir yordamı tetiklemez, böylece yordam REQ girişinin durumunu sıfırlayabilir. Bu işlem, yordamın bir sonraki yordamı başlatmak için yükselen kenarı algılayabilmesi açısından gereklidir.

Bu yordamlardan birini programınıza eklerseniz, STEP 7 sizden örnek DB'sini girmenizi isteyecektir. Her yordam çağrısı için benzersiz bir DB kullanın. Böylece her yordamın REQ gibi girişleri düzgün bir şekilde işlemesi mümkün olur.

Giriş parametresi ID'si

Bu, STEP 7'deki "Cihazlar ve ağlar" altında "Ağ Görünümü" öğesindeki "Yerel ID"ye (onaltılı) bir referanstır ve bu iletişim bloğu için kullanacağınız ağın kimliğidir. Bu ID, yerel bağlantı açıklamasındaki ilişkili parametre ID'si ile eşleşmelidir.

DONE, NDR, ERROR ve STATUS çıkış parametreleri

Bu yordamlarda, yürütme durumunu gösteren çıkışlar vardır:

Tablo 11-56 Açık kullanıcı iletişimi için yordamların çıkış parametreleri

Parametre	Veri tipi	Standart	Açıklama
DONE	Bool	YANLIŞ	Son talebin hata ile tamamlandığını belirtmek için, bir yürütme boyunca DOĞRU olarak ayarlanır; aksi takdirde YANLIŞ.
NDR	Bool	YANLIŞ	İstenen eylemin hatasız tamamlandığını ve yeni dosyaların alındığını belirtmek için, bir yürütme boyunca DOĞRU olarak ayarlanır; aksi takdirde YANLIŞ.
BUSY	Bool	YANLIŞ	Aşağıdakileri belirtmek için, aktif olduğunda DOĞRU olarak ayarlanır: • Görev henüz tamamlanmadı. • Yeni bir görev başlatılamaz. Görev tamamlandığında YANLIŞ olarak ayarlanır.
ERROR	Bool	YANLIŞ	Son talebin hatalarla tamamlandığını belirtmek için bir yürütme boyunca DOĞRU olarak ayarlanır, karşılık gelen hata kodu STATUS içindedir; aksi takdirde YANLIŞ.
STATUS	Word	0	Sonuç durumu: • DONE veya NDR Bit'i ayarlanırsa, STATUS, 0'a veya bir bilgi koduna ayarlanır. • Bit ERROR olarak ayarlanırsa, STATUS bir hata koduna getirilir. • Yukarıdaki Bit'lerin hiçbiri ayarlanmamışsa, yordam, fonksiyonun güncel durumunu açıklayan durum sonuçlarını geri döndürebilir. STATUS içindeki değer, fonksiyonun yürütülmesi sırasında tutulur.

NOT

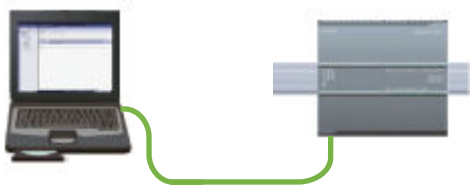
DONE, NDR ve ERROR'un yalnızca bir yürütme için ayarlandığını dikkate alın.

Pasif ISO ve TCP iletişimi için kısıtlı TSAP'ler veya port numaraları

TCON işlemi ile pasif bir iletişim bağlantısı oluşturur ve kurarsanız, aşağıdaki port adreslerini kullanmamalısınız:

- ISO TSAP (pasif):
  - 01.00, 01.01, 02.00, 02.01, 03.00, 03.01
  - 10.00, 10.01, 11.00, 11.01, ... BF.00, BF.01
- TCP portu (pasif) ve UDP portu (pasif):
  - 25, 80, 102, 5001, 34962, 34963, 34964

11.5.9 Bir programlama cihazı ile iletişim



Bir CPU, bir ağda bir STEP 7 programlama cihazı ile iletişim kurabilir.

Bir CPU ile bir programlama cihazı arasında iletişim kurarken aşağıdakileri dikkate alın:

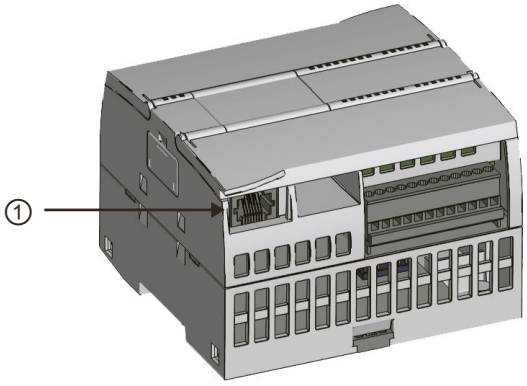
- Konfigürasyon/kurulum: Donanım konfigürasyonu gereklidir.
- İki katılımcı arasındaki iletişim için Ethernet-Switch gerekmez; Ethernet-Switch yalnızca bir ağda ikiden fazla cihaz varsa gerekir.

11.5.9.1 Donanım iletişim bağlantısının oluşturulması

PROFINET arayüzü, bir CPU ile bir programlama cihazı arasındaki fiziksel bağlantıyı kurar. Auto-Cross-Over (otomatik geçiş fonksiyonu) CPU'da uygulandığından, arayüz için standart veya çapraz Ethernet kablosu kullanılabilir. Programlama cihazının CPU'ya doğrudan bağlantısı için bir Ethernet Switch gerekli değildir.

Bir programlama cihazı ile bir CPU arasında donanım bağlantısı kurmak için aşağıdaki yöntemi izleyin:

1. CPU'yu kurun (Sayfa 45).
2. Ethernet kablosunu aşağıda gösterildiği gibi PROFINET konektörüne bağlayın.
3. Ethernet kablosunu programlama cihazınıza bağlayın.



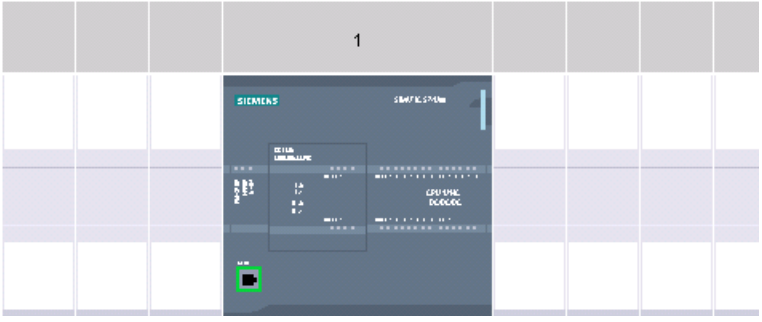
① PROFINET bağlantısı

PROFINET bağlantısı için opsiyonel bir çekme koruması mevcuttur. İlgili sipariş bilgilerini, Yedek parçalar ve diğer donanımlar (Sayfa 1357) altında bulabilirsiniz.

11.5.9.2 Cihazların konfigürasyonu

Daha önce bir CPU içeren bir proje oluşturduysanız, projenizi STEP 7'de açın.

Yoksa, bir proje oluşturun ve modül taşıyıcısına bir CPU ekleyin (Sayfa 128). Aşağıdaki projede "Cihaz görünümü" alanında bir CPU görebilirsiniz.



11.5.9.3 IP adreslerinin atanması

IP adreslerinin atanması

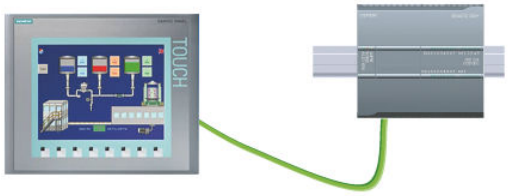
Bir PROFINET ağında, her cihaz ayrı bir IP adresine (İnternet Protokolü adresi) gerek duyar. Bir cihaz bu adresi kullanarak, verileri daha karmaşık bir ağ üzerinden iletebilir:

- Tesis LAN'nize bağlı entegre bir adaptör kartı veya ayrı bir ağa bağlı bir Ethernet USB adaptör kartı kullanan programlama veya diğer ağ cihazları kullanıyorsanız, bu cihazlara IP adreslerinin atanması gerekir. Daha fazla bilgi için, "Programlama ve ağ cihazlarına IP adreslerinin atanması" (Sayfa 566-567) konusuna bakın.
- Ayrıca bir CPU'ya veya ağ cihazına çevrimiçi olarak bir IP adresi de atayabilirsiniz. Bu, ilk cihaz konfigürasyonu sırasında özellikle yararlıdır. Daha fazla bilgi için "Bir CPU'ya IP adresinin atanması" (Sayfa 566-567) konusuna bakın.
- Projenizde CPU veya ağ cihazınızı yapılandırdıktan sonra, PROFINET arayüzünün parametrelerini IP adresini içerecek şekilde yapılandırabilirsiniz. Daha fazla bilgi için, "Projenizdeki bir CPU için IP adresinin yapılandırılması" (Sayfa 569) konusuna bakın.

11.5.9.4 PROFINET ağınızın test edilmesi

Konfigürasyonu tamamladıktan sonra, projenizi CPU'ya yüklemelisiniz. Tüm IP adresleri, proje yüklenirken yapılandırılır.
CPU'nun "Cihaza yükle" fonksiyonu ve ilgili "Gelişmiş yükleme" iletişim kutusunda, erişilebilir tüm ağ cihazları görüntülenebilir ve tüm cihazlara benzersiz IP adresleri atanıp atanmadığı da belirtilebilir. Ayrıntılı bilgi için, "PROFINET ağının test edilmesi" (Sayfa 575) konusuna bakın.

11.5.10 HMI/PLC iletişimi



CPU, PROFINET iletişim bağlantılarını HMI'ler (Sayfa 32) ile destekler. CPU'lar ve HMI'lar arasında iletişimi oluşturmak için aşağıdaki koşulların karşılanması gerekir:

Konfigürasyon/kurulum:

- CPU'nun PROFINET portu, HMI'ye bağlanacak şekilde yapılandırılmalıdır.
- HMI kurulmalı ve yapılandırılmalıdır.
- HMI konfigürasyon verileri, CPU projesine aittir; bunlar, projede yapılandırılabilir ve yüklenebilir.
- İki katılımcı arasındaki iletişim için Ethernet-Switch gerekmez; Ethernet-Switch yalnızca bir ağda ikiden fazla cihaz varsa gerekir.

NOT

4 portlu modül taşıyıcısına monte edilmiş CSM1277 Ethernet Switch, CPU'larınızı ve HMI cihazlarınızı bağlamak için kullanılır. CPU üzerindeki PROFINET bağlantısı bir Ethernet Switch içermez.

Desteklenen fonksiyonlar:

- HMI, CPU verilerini okuyabilir/yazabilir.
- Mesajlar, CPU'dan gelen bilgilere göre tetiklenebilir.
- Sistem teşhisi

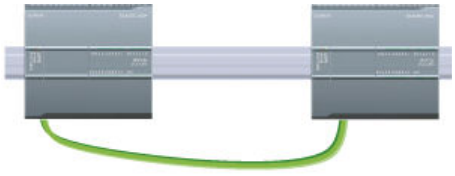
Tablo 11-57 HMI ve CPU arasındaki iletişimi yapılandırmak için gerekli adımlar

Adım	Görevler
1	Donanım iletişim bağlantısının oluşturulması PROFINET arayüzü, bir HMI ile bir CPU arasındaki fiziksel bağlantıyı kurar. Auto-Cross-Over (otomatik geçiş fonksiyonu) CPU'da uygulandığından, arayüz için standart veya çapraz Ethernet kablosu kullanılabilir. HMI'nin CPU'ya doğrudan bağlantısı için bir Ethernet Switch gerekli değildir. Daha fazla bilgi için bkz. "Bir programlama cihazı ile iletişim: Donanım iletişim bağlantısının oluşturulması (Sayfa 681)".
2	Cihazları yapılandırılması Daha fazla bilgi için bkz. "Bir programlama cihazı ile iletişim: Cihazların yapılandırılması (Sayfa 682)".
3	Bir HMI ve bir CPU arasında mantıksal ağ bağlantılarının yapılandırılması Daha fazla bilgi için bkz. "HMI ve PLC arasında iletişim: İki cihaz arasında mantıksal ağ bağlantılarının yapılandırılması" (Sayfa 684).
4	Projenizde IP adresi yapılandırılması Aynı yöntemde devam edin. Yine de HMI ve CPU için IP adreslerini yapılandırmanız gerekir. Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Projenizdeki bir CPU için IP adresinin yapılandırılması" (Sayfa 571).
5	PROFINET ağının test edilmesi Konfigürasyonu, her bir CPU ve her bir HMI cihazı için yüklemeniz gerekir. Ayrıntılı bilgi için bkz. "Cihaz konfigürasyonu: PROFINET ağının test edilmesi" (Sayfa 575).

11.5.10.1 İki cihaz arasında mantıksal ağ bağlantılarının yapılandırılması

Modül taşıyıcısını CPU ile yapılandırdıktan sonra, ağ bağlantılarınızı kurabilirsiniz.
"Cihazlar ve ağlar" portalında, "Ağ görünümü" altında projenizdeki cihazları birbirine bağlayabilirsiniz. İlk önce "Bağlantılar" sekmesine tıklayın, ardından sağdaki açılır listeden bağlantı türünü (örn. ISO on TCP bağlantısı) seçin.
Bir PROFINET bağlantısı oluşturmak için ilk cihazdaki yeşil alana (PROFINET) tıklayın ve ikinci cihazdaki PROFINET alanına bir hat çekin. Fare tuşunu bırakın. PROFINET bağlantınız artık oluşturulmuştur.
Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Ağ bağlantısının oluşturulması" (Sayfa 562).

11.5.11 PLC/PLC iletişimi



TSEND_C ve TRCV_C yordamlarını kullanarak, bir CPU'nun ağdaki başka bir CPU ile iletişime geçmesini sağlayabilirsiniz.

İki CPU arasında iletişim kurarken aşağıdakileri dikkate alın:

- Konfigürasyon/kurulum: Donanım konfigürasyonu gereklidir.
- Desteklenen fonksiyonlar: Ağa bağlanmış bir CPU'ya verilerin okunması/yazılması
- İki katılımcı arasındaki iletişim için Ethernet-Switch gerekmez; Ethernet-Switch yalnızca bir ağda ikiden fazla cihaz varsa gerekir.

Tablo 11-58 İki CPU arasındaki iletişimi yapılandırmak için gerekli adımlar

Adım	Görevler
1	Donanım iletişim bağlantısının oluşturulması Bir PROFINET arayüzü, iki CPU arasındaki fiziksel bağlantıyı kurar. Auto-Cross-Over (otomatik geçiş fonksiyonu) CPU'da uygulandığından, arayüz için standart veya çapraz Ethernet kablosu kullanılabilir. İki CPU'nun bağlantısı için bir Ethernet Switch gerekli değildir. Daha fazla bilgi için bkz. "Bir programlama cihazı ile iletişim: Donanım iletişim bağlantısının oluşturulması (Sayfa 681)".
2	Cihazları yapılandırılması Projenizde iki CPU'yu yapılandırmalısınız. Daha fazla bilgi için bkz. "Bir programlama cihazı ile iletişim: Cihazların yapılandırılması (Sayfa 682)".
3	İki CPU arasında mantıksal ağ bağlantılarının yapılandırılması Daha fazla bilgi için bkz. "PLC ve PLC arasında iletişim: İki cihaz arasında mantıksal ağ bağlantılarının yapılandırılması" (Sayfa 686).
4	Projenizde IP adresi yapılandırılması Aynı yöntemde devam edin. Yine de iki CPU için IP adreslerini yapılandırmanız gerekir (örn. PLC_1 ve PLC_2). Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Projenizdeki bir CPU için IP adresinin yapılandırılması" (Sayfa 571).

Adım	Görevler
5	Gönderim ve alım parametrelerinin yapılandırılması Aralarında bir iletişimi mümkün kılmak için her iki CPU'da TSEND_C ve TRCV_C yordamlarını yapılandırmanız gerekir. Daha fazla bilgiyi, "İki CPU arasında iletişim kurulması başlığı altında bulabilirsiniz: Gönderim ve alım parametrelerinin yapılandırılması (Sayfa 686)".
6	PROFINET ağının test edilmesi Konfigürasyonu, her bir CPU için yüklemeniz gerekir. Ayrıntılı bilgi için bkz. "Cihaz konfigürasyonu: PROFINET ağının test edilmesi" (Sayfa 575).

11.5.11.1 İki cihaz arasında mantıksal ağ bağlantılarının yapılandırılması

Modül taşıyıcısını CPU ile yapılandırdıktan sonra, ağ bağlantılarınızı kurabilirsiniz.

"Cihazlar ve ağlar" portalında, "Ağ görünümü" altında projenizdeki cihazları birbirine bağlayabilirsiniz. İlk önce "Bağlantılar" sekmesine tıklayın, ardından sağdaki açılır listeden bağlantı türünü (örn. ISO on TCP bağlantısı) seçin.

Bir PROFINET bağlantısı oluşturmak için ilk cihazdaki yeşil alana (PROFINET) tıklayın ve ikinci cihazdaki PROFINET alanına bir hat çekin. Fare tuşunu bırakın. PROFINET bağlantınız artık oluşturulmuştur.

Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Ağ bağlantısının oluşturulması" (Sayfa 562).

11.5.11.2 Yerel ile partner CPU arasındaki bağlantı yolunun yapılandırılması

Genel parametrelerin yapılandırılması

İletişim parametrelerini, iletişim yordamının "Özellikler" iletişim kutusunda belirleyebilirsiniz. Bu iletişim kutusu, yordamın bir bölümünü seçtiğinizde sayfanın altında görüntülenir.

Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Yerel CPU ile partner CPU arasındaki bağlantı yolunun yapılandırılması (Sayfa 563)".

Kullanılacak TSAP'leri veya portlarını, "Adres ayrıntıları" bölümündeki "Bağlantı parametreleri" iletişim kutusunda tanımlarsınız. "Yerel TSAP" alanına ise CPU'daki bir bağlantının TSAP veya portu girilir. "Partner TSAP" alanına ise partner CPU'daki bir bağlantının TSAP veya portu girilir.

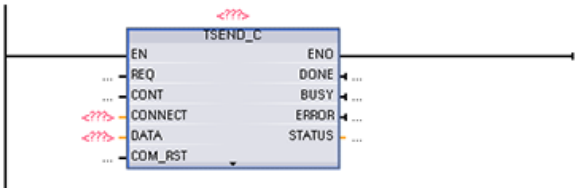
11.5.11.3 Gönderim ve alım parametrelerinin yapılandırılması

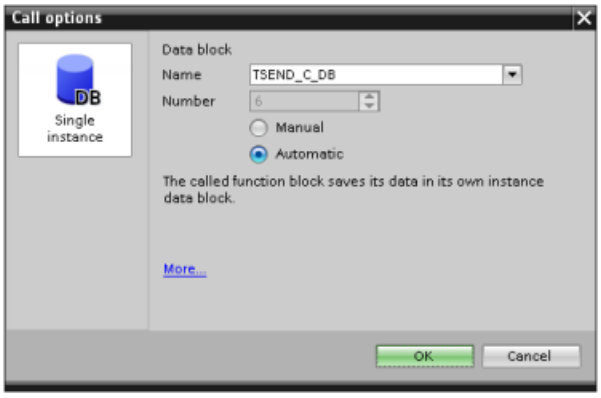
İki CPU arasında bağlantılar, iletişim blokları (örn. TSEND_C ve TRCV_C) üzerinden kurulur. PROFINET iletişiminin CPU'lar tarafından kabul edilebilmesi için mesaj gönderim ve alım parametrelerinin ayarlanması gerekir. Bu parametrelerde, mesajların bir hedef cihaza gönderimi veya bu cihazdan alınması sırasındaki iletişimin nasıl gerçekleşeceği belirlenir.

TSEND_C yordamı

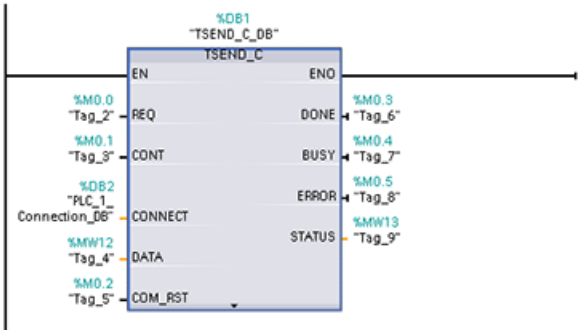
TSEND_C yordamı (Sayfa 600), bir partner istasyonla iletişim bağlantısı kurar. Bağlantı oluşturulur, kurulur ve bir yordam bağlantıyı sonlandırana dek otomatik olarak denetlenir. TSEND_C yordamı, TCON, TDISCON ve TSEND'in fonksiyonlarını birleştirir.

STEP 7'de, cihaz konfigürasyonundaki verilerin bir TSEND_C yordamıyla nasıl gönderileceğini ayarlayabilirsiniz. Önce "Yordamlar" görev kartındaki "İletişim" klasöründe bulunan yordamı, programa ekleyin. TSEND_C yordamı, yordamın parametrelerini saklamak için bir DB atamanızı sağlayan "Çağrı opsiyonları" iletişim kutusuyla birlikte görüntülenir.





Giriş ve çıkışlara, değişkenler belleğinde yer atayabilirsiniz. Aşağıdaki şekilde bu işlem gösterilmiştir:

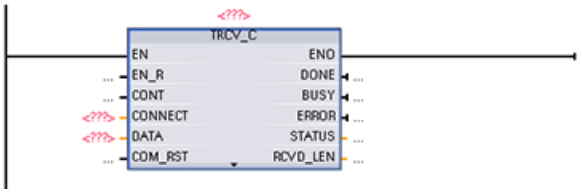


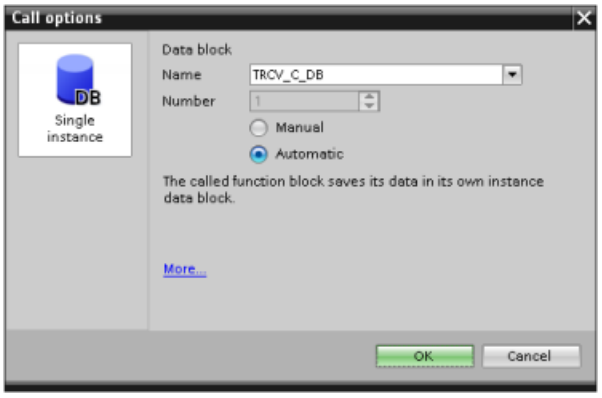
Genel parametrelerin yapılandırılması

TSEND_C yordamının "Özellikler" iletişim kutusundaki parametreleri ayarlayabilirsiniz. Bu iletişim kutusu, TSEND_C yordamın bir bölümünü seçtiğinizde sayfanın altında görüntülenir.

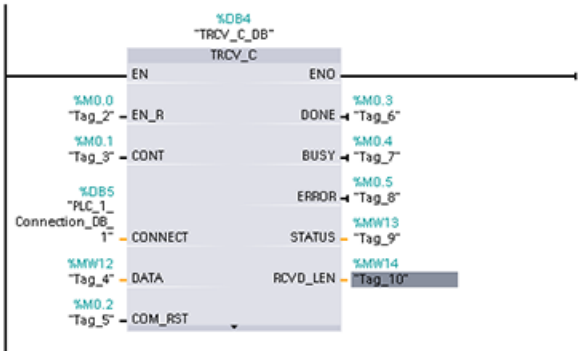
TRCV_C yordamı

TRCV_C yordamı (Sayfa 600), bir partner istasyona bir iletişim bağlantısı kurar. Bağlantı oluşturulur, kurulur ve bir yordam bağlantıyı sonlandırana dek otomatik olarak denetlenir. TRCV_C yordamı, TCON, TDISCON ve TRCV'nin fonksiyonlarını birleştirir.
STEP 7'de, CPU konfigürasyonunda bir TRCV_C yordamının verileri nasıl alması gerektiğini ayarlayabilirsiniz. Önce "Yordamlar" görev kartındaki "İletişim" klasöründe bulunan yordamı, programa ekleyin. TRCV_C yordamı, yordamın parametrelerini saklamak için bir DB ayırmanızı sağlayan "Çağrı opsiyonları" iletişim kutusuyla birlikte görüntülenir.





Giriş ve çıkışlara, değişkenler belleğinde yer atayabilirsiniz. Aşağıdaki şekilde bu işlem gösterilmiştir:



Genel parametrelerin yapılandırılması

TRCV_C yordamının "Özellikler" iletişim kutusundaki parametreleri ayarlayabilirsiniz. Bu iletişim kutusu, TRCV_C yordamının bir bölümünü seçtiğinizde sayfanın altında görüntülenir.

11.5.12 CPU ve PROFINET IO cihazının yapılandırılması

11.5.12.1 PROFINET IO cihazının eklenmesi

PROFINET IO cihazının eklenmesi

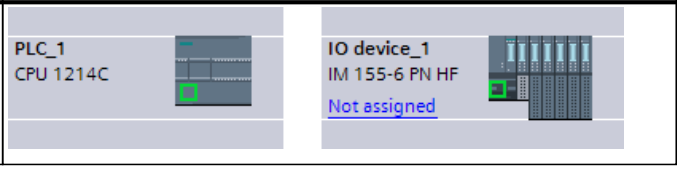
PROFINET IO cihazları eklemek için, "Cihazlar ve Ağlar" portalındaki donanım kataloğunu kullanın.

NOT

Bir PROFINET IO cihazı eklemek için STEP 7 Professional veya Basic V11 üstü sürümlerini kullanabilirsiniz.

Örneğin bir ET200SP IO cihazı eklemek için donanım kataloğunda aşağıdaki kutuları genişletin: Desantral çevrebirim, ET 200SP, arayüz modülleri ve PROFINET. Bunun ardından, ET200SP cihazları listesinden (sipariş numarasına göre sıralanmışlardır) arayüz modülünü seçebilir ve ET200SP IO cihazını ekleyebilirsiniz.

Tablo 11-59 ET200SP IO cihazının cihaz konfigürasyonuna eklenmesi

IO cihazının eklenmesi	Sonuç
	

Artık PROFINET IO cihazını CPU'ya bağlayabilirsiniz:

1. Cihazdaki "Atanmamış" linkine sağ tıklayın ve "IO kumanda ünitesini seç" iletişim kutusunu açmak için bağlam menüsünde "Yeni IO kumanda ünitesini ata" komutunu seçin.
2. Projedeki IO kumanda üniteleri listesinde S7-1200 CPU'nuzu seçin (bu örnekte "PLC_1").
3. Ağ bağlantısını oluşturmak için "Tamam" öğesine tıklayın.

"Cihazlar ve ağlar" portalına gidip, "Ağ görünümü" öğesinde projenizdeki cihazları birbirine bağlayabilirsiniz.

1. Bir PROFINET bağlantısı oluşturmak için ilk cihazdaki yeşil alana (PROFINET) tıklayın ve ikinci cihazdaki PROFINET alanına bir hat çekin.
2. Fare tuşunu bırakın. PROFINET bağlantınız artık oluşturulmuştur.

Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: CPU'nun iletişim için yapılandırılması" (Sayfa 163).

11.5.12.2 CPU'lar ve cihaz adlarının atanması

CPU'lar ve cihaz adlarının atanması

Ayrıca cihazlar arasındaki ağ bağlantıları, CPU'nun cihaza kumanda etmesini sağlamak için CPU'ya PROFINET IO cihazını da atar. Bu atamayı değiştirmek isterseniz, PROFINET IO cihazında görüntülenen PLC adına tıklayın. Bundan sonra, PROFINET IO cihazını güncel CPU'dan ayırabileceğiniz ve yeniden atayabileceğiniz veya atamadan bırakabileceğiniz bir iletişim kutusu açılır.

CPU'ya bağlanmadan önce, PROFINET ağınızdaki cihazlara birer ad atanmalıdır. Cihazlar henüz bir ada sahip değilse veya bir cihazın adı değiştirilecekse, "Ağ görünümünde" PROFINET cihazlarınıza adlar atayın. Bunun için, PROFINET IO cihazına sağ tıklayın ve "Cihaz adını ata" öğesini seçin.

Bir PROFINET IO cihazının adını, hem STEP 7 projesinde hem de PROFINET ağındaki PROFINET IO-Device'a atamalısınız. (PROFINET ağında cihaz adını atamak için "Çevrimiçi ve teşhis" STEP 7 aracını veya işletime alma, konfigürasyon ve teşhis için PRONETA aracını kullanabilirsiniz.) Bir ad eksik kalmışsa veya iki saklama konumundaki adlar eşleşmiyorsa, PROFINET IO veri alışverişi modu işlemez. Daha fazla bilgi için bkz. "Çevrimiçi ve teşhis araçları: Bir PROFINET cihazına çevrimiçi olarak bir ad atanması (Sayfa 1118)".

11.5.12.3 IP adreslerinin atanması

IP adreslerinin atanması

Bir PROFINET ağında, her cihaz ayrı bir IP adresine (İnternet Protokolü adresi) gerek duyar. Bir cihaz bu adresi kullanarak, verileri daha karmaşık bir ağ üzerinden iletebilir:

- Tesis LAN'nize bağlı entegre bir adaptör kartı veya ayrı bir ağa bağlı bir Ethernet USB adaptör kartı kullanan programlama veya diğer ağ cihazları kullanıyorsanız, bu cihazlara IP adreslerinin atanması gerekir. Daha fazla bilgi için, "Programlama ve ağ cihazlarına IP adreslerinin atanması" (Sayfa 566-567) konusuna bakın.
- Ayrıca bir CPU'ya veya ağ cihazına çevrimiçi olarak bir IP adresi de atayabilirsiniz. Bu, ilk cihaz konfigürasyonu sırasında özellikle yararlıdır. Daha fazla bilgi için "Bir CPU'ya IP adresinin atanması" (Sayfa 569) konusuna bakın.
- Projenizde CPU veya ağ cihazınızı yapılandırdıktan sonra, PROFINET arayüzünün parametrelerini IP adresini içerecek şekilde yapılandırabilirsiniz. Daha fazla bilgi için, "Projenizdeki bir CPU için IP adresinin yapılandırılması" (Sayfa 571) konusuna bakın.

11.5.12.4 IO çevrim süresinin yapılandırılması

IO çevrim süresinin yapılandırılması

Bir PROFINET IO cihazı, "IO çevrimi" süresi içinde CPU tarafından yeni verilerle beslenir. Güncelleme süresi, her bir cihaz için ayrı olarak yapılandırılabilir ve bu süre, verilerin CPU'dan cihaza ve cihazdan aktarıldığı süreyi tanımlar.

STEP 7, PROFINET ağının her cihazı için standart ayarlarda "IO çevrimi" için güncelleme süresini otomatik olarak hesaplar ve bu sırada, alınıp verilecek veri hacmini ve ilgili kumanda ünitesine atanan cihaz sayısını dikkate alır. Güncelleme süresinin otomatik olarak hesaplanmasını istemiyorsanız bu ayarı değiştirebilirsiniz.

PROFINET IO cihazının "Özellikler" konfigürasyon iletişim kutusunda "IO çevrimi" için olan parametreleri belirlersiniz. Bu iletişim kutusu, yordamın bir bölümünü seçtiğinizde sayfanın altında görüntülenir.

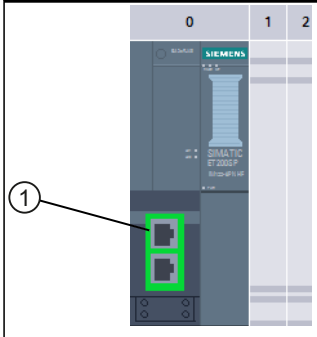
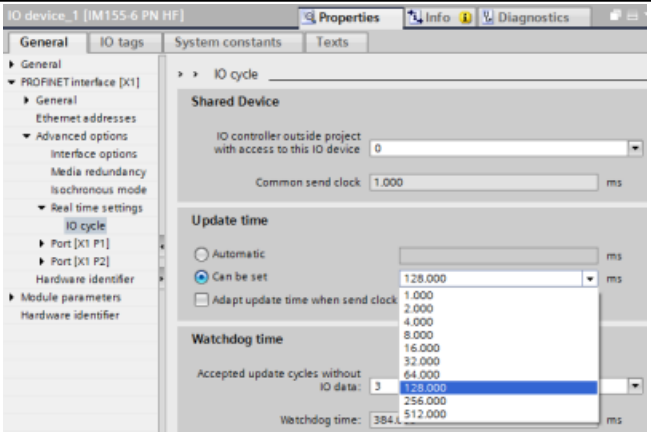
PROFINET IO cihazının "Cihaz görünümü" öğesinde PROFINET portuna tıklayın. "PROFINET arayüzü" iletişim kutusunda aşağıdaki menü seçimiyle "IO çevrimi" parametrelerini çağırırsınız:

- "Gelişmiş opsiyonlar"
- "Gerçek zamanlı ayarlar"
- "IO çevrimi"

IO çevrimi için "güncelleme süresini" şu şekilde tanımlayın:

- Uygun bir güncelleme süresinin otomatik olarak hesaplanmasını istiyorsanız, "Otomatik" öğesini seçin.
- Güncelleme süresini kendiniz ayarlamak istiyorsanız "Ayarlanabilir" öğesini seçin ve istediğiniz güncelleme süresini msn cinsinden girin.

Tablo 11-60 ET200SP için PROFINET IO çevrim süresinin yapılandırılması

ET 200SP PROFINET IO cihazı	"ET 200SP PROFINET IO çevrimi" iletişim alanı
	

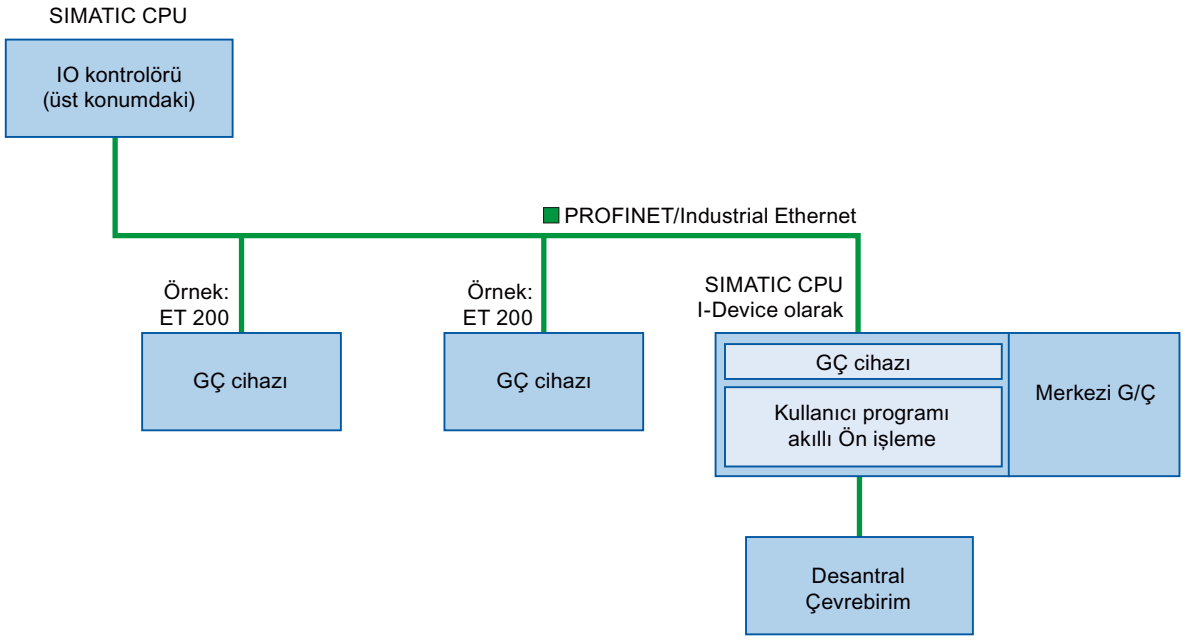
① PROFINET portu

11.5.13 CPU ve PROFINET I-Device'ın yapılandırılması

11.5.13.1 I-Device fonksiyonu

CPU'nun "I-Device" (akıllı IO cihazı) fonksiyonu, bir IO kontrolörü ile veri alışverişini ve örneğin kısmi prosesler için akıllı bir ön işleme ünitesi görevi görerek bu CPU'nun işleyişini kolaylaştırır. I-Device, bir IO cihazı olarak daha üst seviyedeki bir IO kontrolörüne bağlanır.

Ön işleme, CPU'nun kullanıcı programı tarafından yürütülür. Merkezi veya desantral çevrebirim G/Ç'de (PROFINET IO veya PROFIBUS DP) toplanan proses değerleri, kullanıcı programı tarafından önceden işlenir ve bir PROFINET IO arayüzü üzerinden bir üst seviyede istasyonun CPU'suna sunulur.



"I-Device" için adlandırma kuralları

I-Device fonksiyonuna sahip bir CPU veya CP, kolaylık sağlama adına aşağıdaki açıklamalarda "I-Device" olarak anılacaktır.

11.5.13.2 I-Device'ın özellikleri ve avantajları

Kullanım aralıkları

Bir I-Device için kullanım alanı:

- Desantral işleme:
  Karmaşık otomasyon görevleri daha küçük kısmi tesislere/kısmi proseslere bölünebilir. Böylece prosesler daha kolay yönetilebilir ve kısmi görevler basitleştirilir.
- Kısmi proseslerin ayrılması:
  Karmaşık, geniş çapta desantral ve kapsamlı prosesler, I-Device'lar yardımıyla daha yalın arayüzlere sahip birden çok alt prosese bölünebilir. Gerekli olduğunda, kısmi prosesler daha sonra bir ana projede birleştirilebilen münferit STEP 7 projelerine kaydedilebilir.
- Teknik bilgi koruması:
  Bileşenler, bir STEP 7 projesi olarak değil, I-Device arayüz tanımı için yalnızca bir GSD dosyası ile teslim edilebilir. Artık yayımlanması gerekmediği için, kullanıcı, programını koruyabilir.

Özellikler

I-Device'ın özellikleri:

- STEP 7 projeleri ile ayrılmış olarak çalışılması:
  Bir I-Device'ın oluşturucuları ve kullanıcıları, STEP 7 otomasyon projeleriyle tamamen ayrı olarak çalışabilirler. GSD dosyası, STEP 7 projeleri arasındaki arayüzü oluşturur. Bu sayede, standartlaştırılmış bir arayüz üzerinden standart IO kontrolörlerine bağlantı kurulabilir.
- Gerçek zamanlı iletişim
  I-Device, bir PROFINET IO arayüzü üzerinden deterministik bir PROFINET IO sistemi ile teslim edilir.

Avantajlar

Bir I-Device aşağıdaki avantajları sunar:

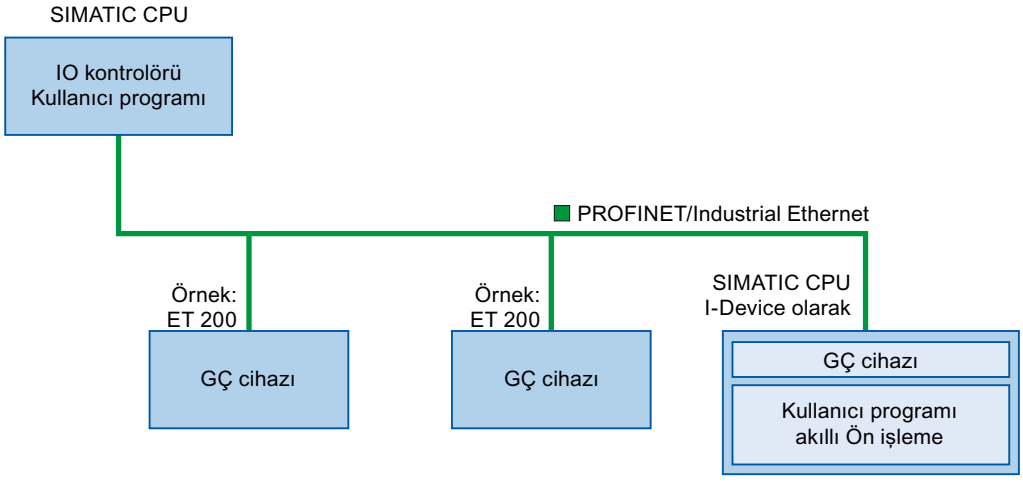
- IO kontrolörlerinin basit bağlantısı
- IO kontrolörler arasında gerçek zamanlı iletişim
- Bilgi işlem gücünün I-Device'lara dağıtılmasıyla IO kontrolörünün rahatlaması
- Proses verilerinin yerel olarak işlenmesi sayesinde daha düşük iletişim yükü
- Alt görevleri ayrı STEP 7 projelerde işleyerek daha kolay yönetilebilir projeler

11.5.13.3 Bir I-Device'ın özellikleri

Bir I-Device, standart bir IO cihazı gibi bir G/Ç sistemine eklenir.

Alt seviyede PROFINET IO sistemi olmayan I-Device

Bir I-Device'ın kendisine ait desantral çevrebirimleri yoktur. Bir I-Device'ın bir IO cihazı olarak konfigürasyonu ve parametrelendirilmesi, bir desantral G/Ç sistemiyle aynıdır (örn. ET 200).



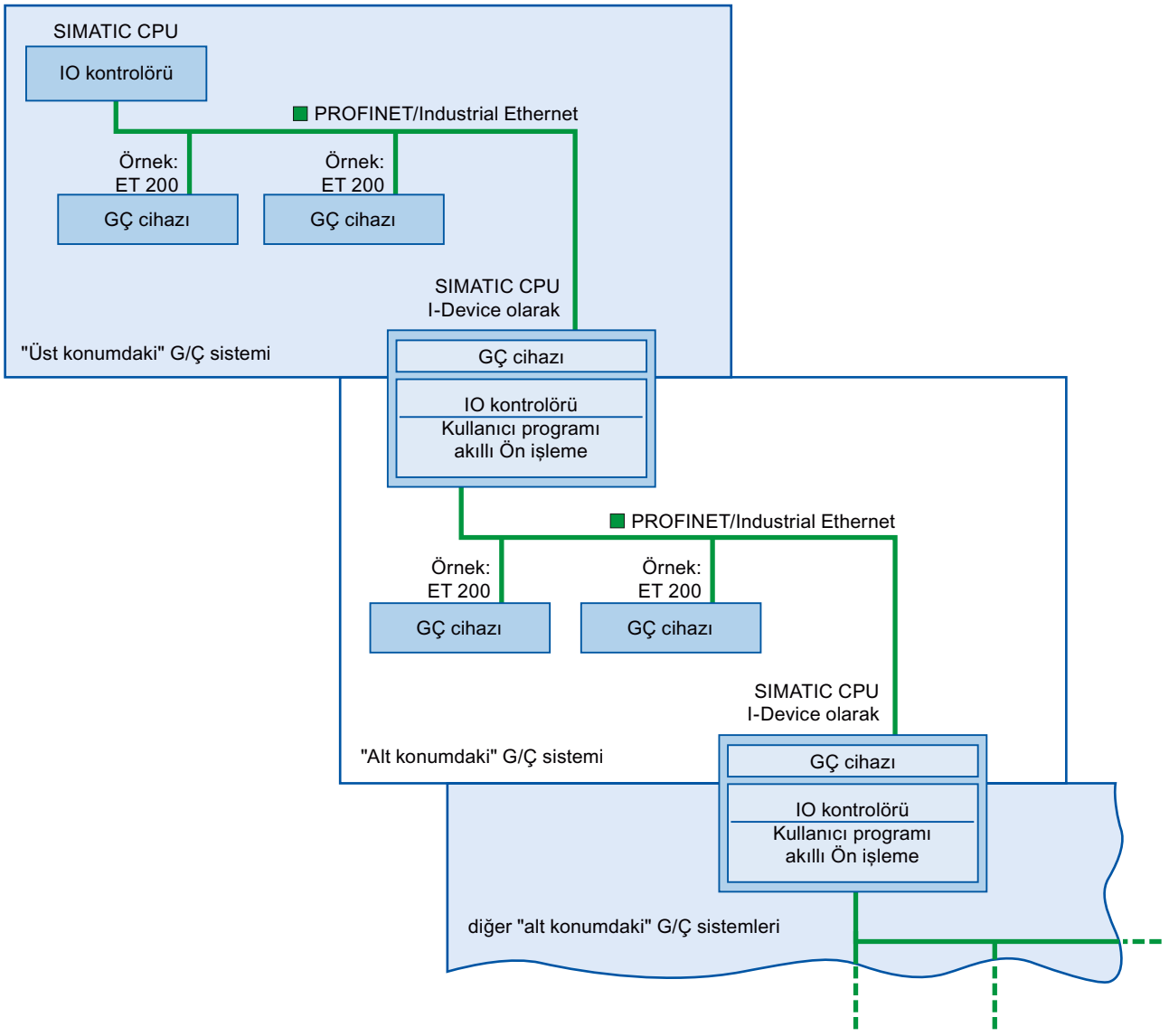
Alt seviyede PROFINET IO sistemine sahip I-Device

Konfigürasyona bağlı olarak, bir I-Device, IO cihazı fonksiyonelliğine ek olarak bir PROFINET arayüzünde IO kontrolörü olarak da kullanılabilir.

Bununla birlikte bir I-Device, PROFINET arayüzü üzerinden daha yüksek seviyeli bir G/Ç sistemine ait olabilir ve bir IO kontrolörü olarak kendi alt seviyeli G/Ç sistemini destekleyebilir.

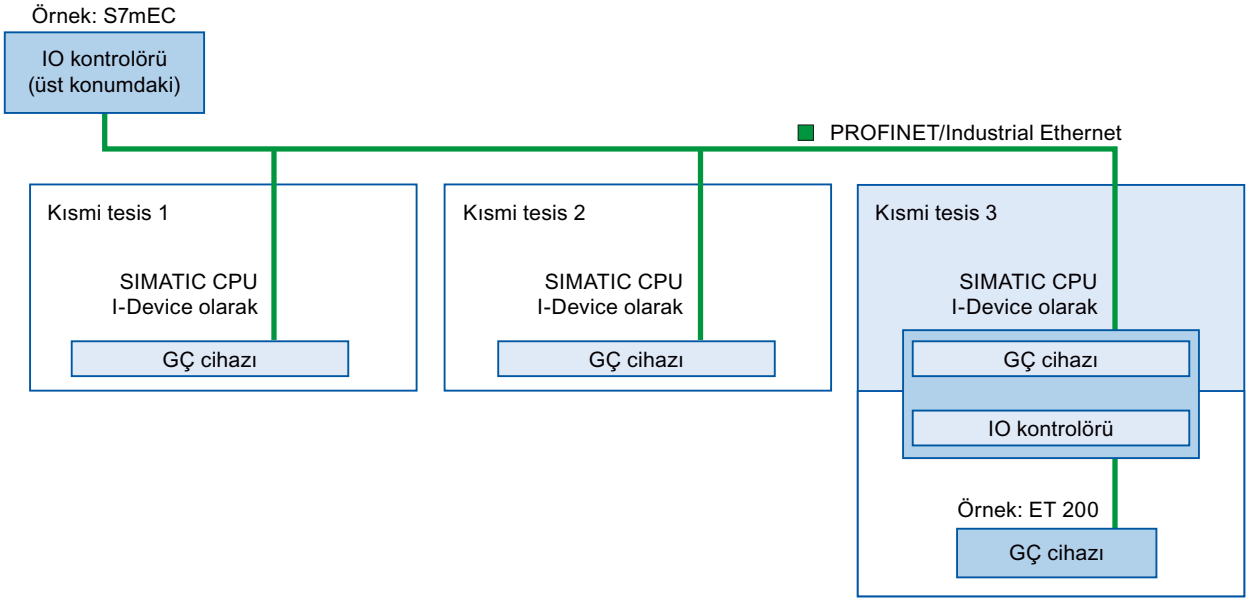
Alt seviyedeki G/Ç sistemi de kendi I-Device'larını içerebilir (aşağıdaki şekle bakın). Bu ise hiyerarşik olarak yapılandırılmış G/Ç sistemlerini mümkün kılar.

Bir IO kontrolörü fonksiyonuna ek olarak, bir I-Device, bir PROFIBUS arayüzü üzerinden bir alt seviyedeki PROFIBUS sistemi için bir DP-Master olarak da işletilebilir.



Örnek: IO cihazı ve IO kontrolörü olarak I-Device

Bir I-Device'ın IO cihazı ve IO kontrolörü olarak fonksiyonu burada örnek olarak bir yazdırma işlemi kullanılarak açıklanmaktadır. I-Device, bir kısmi tesisi (bir kısmi prosesi) kumanda eder. Örneğin bir basılı ürünler paketinde, çok sayıdaki kısmi tesisten biri el ilanları veya broşürler gibi ek sayfalar eklemek için kullanılabilir.



Kısmi tesis 1 ve kısmi tesis 2'nin her biri, merkezi G/Ç'ye sahip birer I-Device'dan oluşur. Desantral G/Ç sistemi (örn. ET 200) ile birlikte I-Device, kısmi tesis 3'ü oluşturur.

I-Device'daki kullanıcı programı, proses verilerinin ön işlenmesinden sorumludur. Bu görev için I-Device'ın kullanıcı programı, üst seviyedeki IO kontrolöründen standart ayarlara (örn. kumanda verileri) gerek duyar. I-Device, sonuçları (örn. kısmi görevin durumu) daha üst seviye IO kontrolörüne iletir.

11.5.13.4 Daha yüksek ve daha düşük seviye G/Ç sistemi arasında veri alışverişi

Transfer alanları, I-Device CPU'sunun kullanıcı programına yönelik arayüzlerdir. Girişler kullanıcı programında işlenir, çıkışlar ise işlemenin sonucudur.

IO kontrolörü ile I-Device arasındaki iletişim için veriler transfer alanlarında tutulur. Bir transfer alanı, IO kontrolörü ile I-Device arasında sürekli olarak alınıp verilen bir bilgi birimi içerir. Transfer alanlarının konfigürasyonu ve kullanımına ilişkin daha fazla bilgiyi "I-Device yapılandırması" (Sayfa 699) bölümünde bulabilirsiniz.

Ağ kaybı durumunda, kumanda ünitesi ve I-Device'daki girişlerin transfer alanlarının farklı davranışları

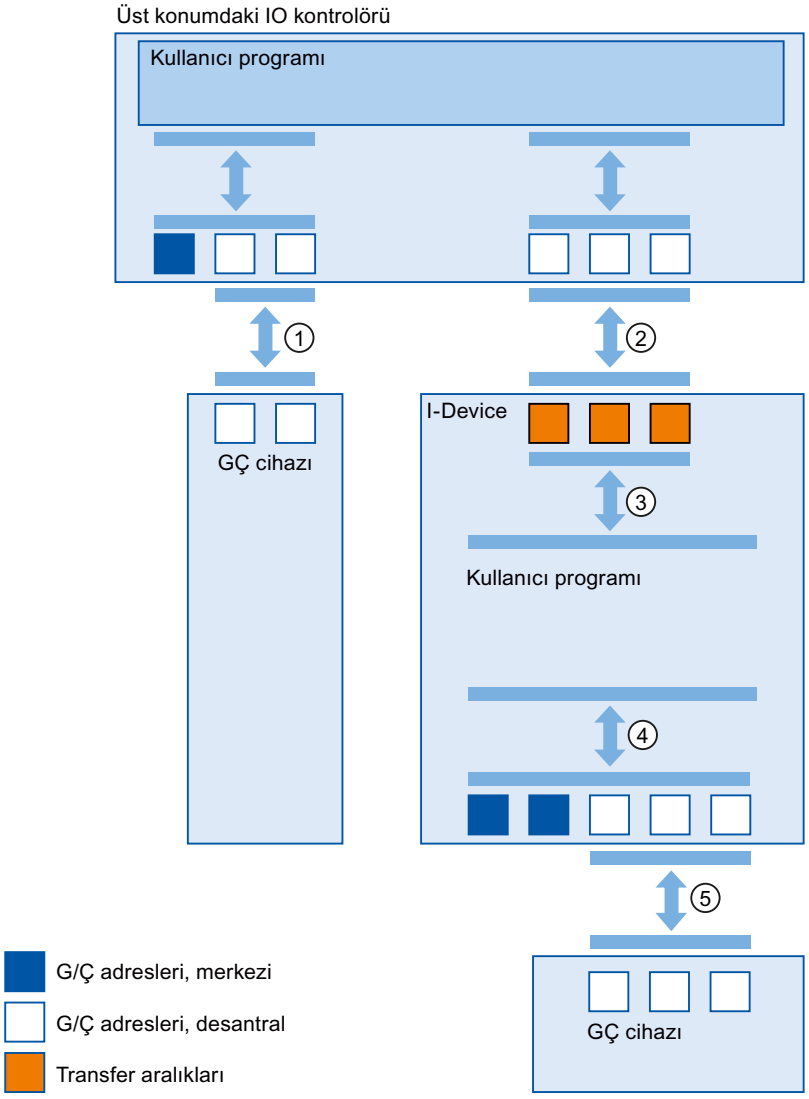
Ağ bağlantısı kesildiğinde, kumanda ünitesi, CPU girişlerin transfer alanlarına sıfır değerini yazar. I-Device üzerindeki transfer alanları son değerlerini korur.

I-Device'ın genel durumunda (onaylanmamış I-Device) bu durumu önlemek için, sisteminizi uygun şekilde yapılandırabilirsiniz. Bunu uygulamak için, yaklaşan bir olay için modül taşıyıcısı veya istasyon arızası için bir OB'deki I-Device girişlerinin transfer alanlarını silin. Aşağıdaki yöntemi izleyin:

1. Projenize bir modül taşıyıcısı veya istasyon arızası OB'si ekleyin. (Bu OB, standart olarak OB 86 numarasına sahiptir.)
2. LAD'nin yol alma değişkeni I-Device'ın donanım ID'sinin değerini ve Event_Class'ın yol alma değişkeni bir "gelen" olayı gösteriyorsa, I-Device girişlerinin değerlerine sıfır değerinin yazılmasını sağlamak üzere OB'ye lojik ekleyin:
   - I-Device'ın donanım ID'sini, "Sistem sabitleri" sekmesindeki standart değişkenler tablosunda bulabilirsiniz. Donanım ID'si bir tür "HW_Device"dir ve değişkenin adı, onun bir I-Device olduğunu gösterir (örn. "Local~PROFINET_interface_1~IODevice").
   - Event_Class'taki "16#39" değeri, bir "gelen" olayı gösterir. "Event_Class" giriş değişkeni "16#39" değerini içeriyorsa, bu modül taşıyıcısı veya istasyon arızası OB'sinin artık aktif olduğunu (ve silinmediğini) belirtir.

Veri alışverişi

Aşağıdaki şekil, üst seviye ve alt seviyedeki G/Ç sistemi arasındaki veri alışverişini göstermektedir. Münferit iletişim ilişkileri aşağıda sayılarla belirtilmektedir:



① **Üst seviye IO kontrolörü ile normal IO cihazı arasında veri alışverişi**
IO kontrolörü ve GÇ cihazları, PROFINET üzerinden veri alışverişinde bulunur.

② **Üst seviye IO kontrolörü ile I-Device arasında veri alışverişi**
IO kontrolörü ve I-Device, PROFINET üzerinden veri alışverişinde bulunur.
Daha üst seviyedeki bir IO kontrolörü ile bir I-Device arasındaki veri alışverişi, IO kontrolörü ile IO cihazı arasındaki olağan ilişkiyi baz alır.
Üst seviyedeki IO kontrolörü için, I-Device'ların transfer alanları, önceden yapılandırılmış bir istasyonun alt modüllerini temsil eder.
Üst seviye IO kontrolörünün çıkış verileri, I-Device'ın giriş verileridir. Öte yandan IO kontrolörünün giriş verileri, I-Device'ın çıkış verileridir.

③ **Kullanıcı programı ile transfer alanı arasındaki transfer ilişkisi**
Giriş ve çıkış verileri, kullanıcı programı ve transfer alanı arasında alınıp verilir.

④ **Kullanıcı programı ile I-Device'ın G/Ç'si arasında veri alışverişi**
Giriş ve çıkış verileri, kullanıcı programı ile merkezi/desantral çevrebirimler arasında alınıp verilir.

⑤ **I-Device ve bir alt seviye IO cihazı arasında veri alışverişi**
I-Device ve onun IO cihazları arasında veriler alınıp verilir. Bu veriler, PROFINET üzerinden aktarılır.

11.5.13.5 I-Device yapılandırması

Konfigürasyon için temel olarak iki opsiyon vardır:

- Bir projede bir I-Device'ın konfigürasyonu
- Başka bir projede veya mühendislik sisteminde kullanılan bir I-Device'ın konfigürasyonu.

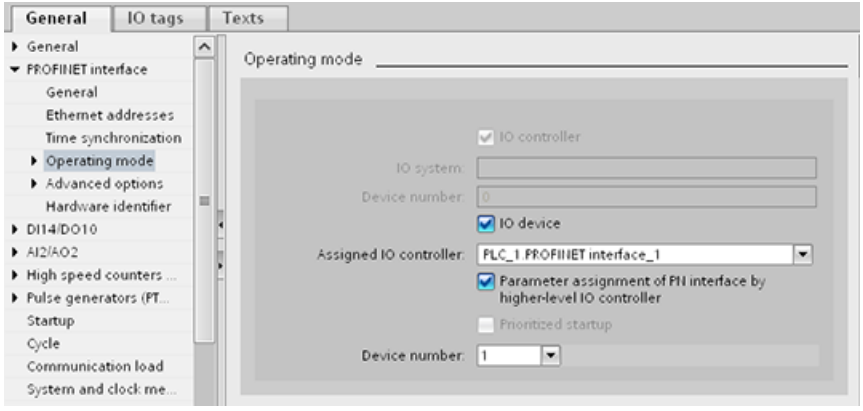
STEP 7'de, yapılandırılmış bir I-Device'ı bir GSD dosyasına dışa aktararak, bir I-Device'ı başka bir proje veya başka bir mühendislik sisteminde kullanmak için yapılandırabilirsiniz. GSD dosyasını, diğer GSD dosyaları gibi başka projelere veya mühendislik sistemlerine aktarabilirsiniz. Diğer verilere ek olarak, bu GSD dosyasında veri alışverişi için transfer alanları da saklanabilir.

NOT

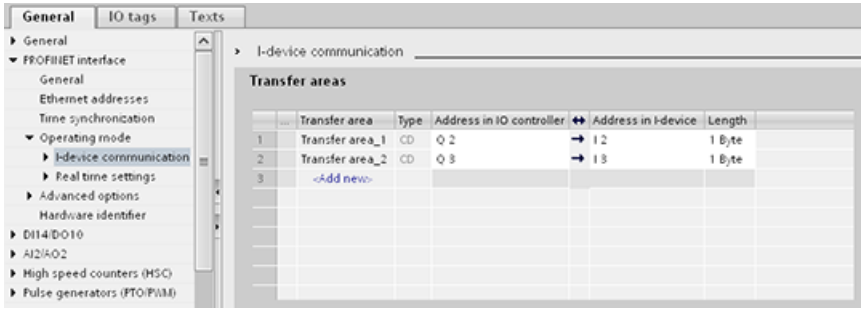
S7-1200'ü paylaşılan bir I-Device ve bir kontrolör olarak kullanıyorsanız, iletişim performansı üzerindeki etkiyi azaltmak için PROFINET I-Device ve PROFINET IO'nun güncelleme sürelerini uzattığınızdan emin olmalısınız. Sistem çok stabildir ve tek bir PROFINET I-Device için 2 msn güncelleme süresi ve tek bir PROFINET IO için 2 msn güncelleme süresi seçtiğinizde düzgün çalışır.

Bir projede bir I-Device'ın konfigürasyonu

1. Fare ile PROFINET CPU'yu donanım kataloğundan ağ görünümüne sürükleyin.
2. IO cihazı olarak da yapılandırılabilen bir PROFINET CPU'yu fare yardımıyla donanım kataloğundan ağ görünümüne sürükleyin. Bu cihaz bir I-Device olarak yapılandırılmıştır (örn. bir CPU 1215C).
3. I-Device için PROFINET arayüzünü seçin.
4. Gezinme alanındaki denetçi penceresinde "İşletim türü" öğesini seçin ve "IO cihazı" opsiyon kutusunu etkinleştirin.
5. Şimdi, "Atanmış IO kontrolörü" açılır listesinden IO kontrolörünü seçebilirsiniz.
   IO kontrolörünü seçtikten sonra, ağ görünümü iki cihaz arasındaki ağı ve IO sistemini gösterir.



6. "Üst seviye IO kontrolörü üzerinden PN arayüzü parametrelendirmesi" opsiyon kutusuyla, arayüz parametrelerinin I-Device'ın kendisi tarafından mı yoksa daha yüksek seviyede bir IO kontrolörü tarafından mı atanacağını belirlersiniz.
   I-Device'ı alt seviyedeki IO sisteminde çalıştırırsanız, I-Device'ın PROFINET arayüzünün parametreleri (örn. port parametresi) üst seviyedeki IO kontrolörü tarafından atanamaz.
7. Transfer alanlarını yapılandırın. Transfer alanlarını "I-Device iletişimi" gezinme alanında bulabilirsiniz:
   – "Transfer alanı" sütununda, ilk alana tıklayın. STEP 7, daha sonra değiştirebileceğiniz bir standart ad atar.
   – İletişim ilişkisi türünü seçin: Bu aşamada sadece CD veya F-CD seçebilirsiniz.
   – Adresler otomatik olarak önceden ayarlanmıştır. Gerektiğinde bu adresleri düzeltebilir ve sürekli olarak aktarılan transfer alanının uzunluğunu belirleyebilirsiniz.



8. Her transfer alanı için gezinme alanında ayrı bir giriş oluşturulur. Bu veri girişlerinden birini seçerseniz, transfer alanının ayrıntılarını uyarlayabilir, düzeltebilir ve bunlarla ilgili yorum girebilirsiniz.

NOT

Bir S7-1200'ü I-Device olarak yapılandırırsanız, aktarım alanının maksimum boyutu 1024 giriş veya çıkış Bayt'ı olur. Kumanda cihazındaki yerel G/Ç ve adres bölümü sınırlamalarına bağlı olarak sınırlayıcı faktörler de söz konusudur.

Bir I-Device'ın bir GSD dosyasıyla yapılandırılması

Bir I-Device'ı başka bir projede veya I-Device'ı başka bir mühendislik sisteminde kullanırsanız, üst seviyedeki IO kontrolörü ve I-Device'ı yukarıda açıklandığı gibi yapılandırmalısınız.

Transfer alanlarını yapılandırdıktan sonra, I-Device için yeni bir GSD dosyası oluşturulmasını sağlamak adına, "Dışa Aktar" butonuna tıklayın. Bu GSD dosyası, diğer projelerde yapılandırılan I-Device'ı temsil eder.

Denetçi penceresindeki "I-Device iletişimi" alanında "Dışa Aktar" butonunu kullanın.

Donanım konfigürasyonu derlenir ve dışa aktarma iletişim kutusu açılır.

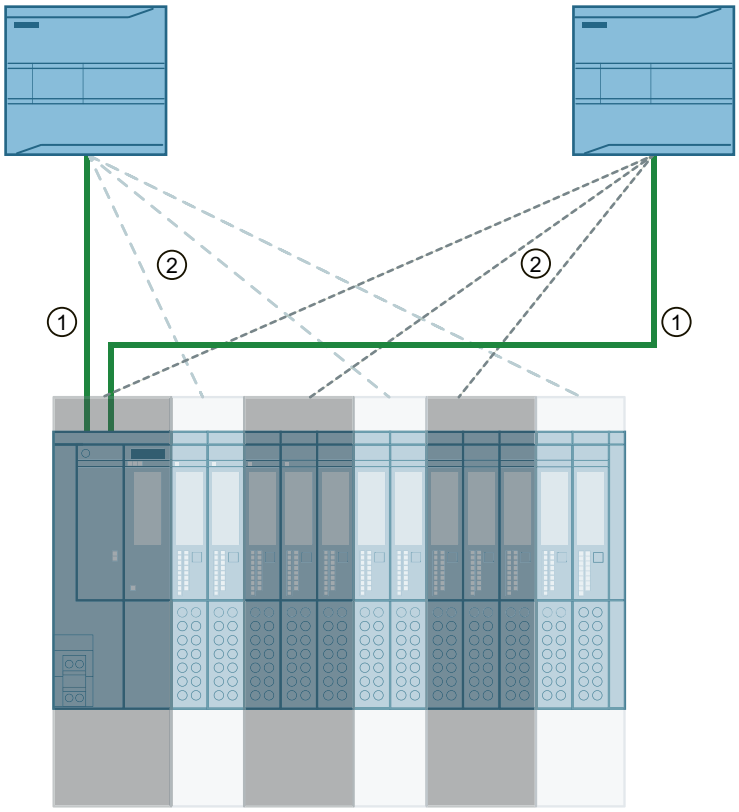
I-Device-Proxy'ye bir ad atayın ve öngörülen alanlara bir açıklama girin. İşlemi tamamlamak için "Dışa Aktar" butonuna tıklayın.

Bunun ardından örneğin GSD dosyasını başka bir projeye içe aktarabilirsiniz.

11.5.14 Shared Device'lar

11.5.14.1 Shared Device fonksiyonu

Daha büyük ve daha dağıtılmış sistemlerde sıklıkla çok sayıda IO kontrolörü kullanılır.
"Shared Device" fonksiyonu olmadan, bir IO cihazının her bir G/Ç modülü aynı IO kontrolörüne atanır. Fiziksel olarak birbirine yakın duran sensörlerin farklı IO kontrolörlerine veri sağlaması istendiğinde, birden fazla IO cihazı gerekir.
"Shared Device" fonksiyon ile bir IO cihazının modülleri veya alt modülleri farklı IO kontrolörlerine paylaştırılabilir. Bu sayede esnek otomasyon konseptleri mümkün olur. Örneğin, bir IO cihazında birbirine yakın G/Ç modüllerini bağlama olanağınız söz konusudur.



① PROFINET
② Mantıksal atama

Prensip

Shared Device'ın alt modüllerine erişim daha sonra münferit IO kontrolörlerine dağıtılır. Shared Device'ın her bir alt modülü özel olarak bir IO kontrolörüne atanır.

Ön koşul (GSD konfigürasyonu)

- STEP 7
- S7-1200 CPU, Firmware V4.1'den itibaren IO kontrolörü olarak
- IO cihazı tarafından Shared Device fonksiyonun desteklenir, örn. arayüz modülü IM 155-5 PN ST
- IO cihazının konfigürasyonu için kullanılan GSD dosyası kurulur
- I-Device olarak yapılandırılmış bir S7-1200 CPU, Shared Device fonksiyonunu destekler. I-Device için PROFINET GSD dosyasını STEP 7'den dışa aktarmalı ve ardından STEP 7'ye (TIA Portal) içe aktarmalısınız.

Erişimin yapılandırılması

Bir IO cihazının modüllerinin veya alt modüllerinin farklı IO kontrolörlerine atanabilmesini sağlamak için, IO cihazının farklı projelerde bulunması gerekir. Her IO kontrolörü için ayrı bir proje gereklidir.

Arayüz modülünün "Shared Device" parametresi üzerinden IO kontrolörünün erişimi olan modülleri ve alt modülleri belirlersiniz:

- Yerel IO kontrolörünün yapılandırılmış modüle erişimi varsa, listeden IO kontrolörünün adını seçin.
- Yerel IO kontrolörü yerine başka bir projenin IO kontrolörünün yapılandırılan modüle erişimi olması gerekiyorsa, "---" girişini seçin.

Her bir modül veya alt modül, bir projede tam olarak bir IO kontrolörüne atanırsa, konfigürasyon, erişim açısından tutarlı olur.

Modül veya alt modül başka bir IO kontrolörüne atanmıştır

Aşağıdaki alt bölümde, yerel IO kontrolörü açısından "Shared Device" parametresi için "---" ayarının sonuçları açıklanmaktadır.

Bu durumda, yerel IO kontrolörünün bu şekilde yapılandırılan modüle erişimi yoktur. Bu, ayrıntılı olarak şu anlama gelir:

- Modül veya alt modül ile veri alışverişi yok
- Alarm ve teşhis mesajları alınmıyor, yani çevrimiçi görünümde teşhis durumu görüntülenmiyor
- Modül veya alt modülün parametrelendirilmesi yok

Gerçek zamanlı özellikler ayarı

STEP 7, iletişim yükünü ve dolayısıyla ortaya çıkan güncelleme sürelerini hesaplar. Shared Device konfigürasyonları ile bir hesaplamanın mümkün olması için, Shared Device'ın PROFINET arayüzünün IO kontrolörüne atandığı projede proje harici IO kontrolörlerinin sayısını belirtmeniz gerekir.

Shared Device için mümkün olan maksimum G/Ç kontrol cihazı sayısı cihaza bağlıdır. Bu sayı, Shared Device'ın GSD dosyasına kaydedilir.

IO kontrolörü olarak bir S7-1200 CPU ile çok kısa bir gönderim adımı (en az 1 msn) belirleyebilirsiniz. Gönderim adımı, Shared Device tarafından desteklenen en kısa gönderim adımından daha kısa olabilir. Bu durumda, G/Ç kontrolörü Shared Device'ı, cihaz tarafından desteklenen bir gönderim adımı ile işletir (gönderim adımı uyarlaması).

Örnek: Bir CPU, 1 msn'den itibaren gönderim adımlarını destekler. Yapılandırılmış bir IO cihazı, 1,25 msn'den başlayan gönderim adımlarını, başka bir IO cihazı ise 1 msn'den başlayan gönderim adımlarını destekler. Bu durumda, CPU için kısa gönderim adımını 1 msn olarak ayarlama olanağınız vardır. CPU, "yavaş" IO cihazını 1,25 msn'lik gönderim adımı ile işletir.

Konfigürasyon kuralları

- Shared Device'ı kullanan IO kontrolörleri farklı projelerde oluşturulur. Her projede, Shared Device'ın her istasyonda aynı şekilde yapılandırılması sağlanmalıdır. Yalnızca tek bir IO kontrolörü, bir alt modüle tam erişime sahip olabilir. Konfigürasyondaki olası tutarsızlıklar, Shared Device'ın hatasına yol açar.
- Bir modülün veya alt modülün G/Ç adresleri, yalnızca aynı projede IO kontrolörüne bir modül veya alt modül atandığında düzenlenebilir.
- Shared Device, her projede aynı IP parametrelerine ve aynı cihaz adına sahip olmalıdır.
- Gönderim adımı, Shared Device'a erişimi olan tüm IO kontrolörleri için aynı olmalıdır.
- Shared Device'ın bağlı olduğu alt ağın S7 alt ağ ID'si tüm projelerde aynı olmalıdır.
- Aşağıdaki fonksiyonlar yalnızca, Shared Device'ın PROFINET arayüzü yerel IO kontrolörüne atandığında kullanılabilir:
  - Öncelikli yol alma
  - Port özelliklerinin parametrelendirmesi

Sınır koşulları

Aşağıdaki sınır koşulları, bir Shared Device konfigürasyonunun birden çok projeye dağıtılması nedeniyle ortaya çıkar:

- Bu IO kontrolörüne atanmamış modüllerin ve alt modüllerin adresleri, bir Shared Device'a erişimi olan her IO kontrolörünün adres genel görünümünde eksik kalır.
- Atanmamış modüller ve alt modüller, tutarlılık denetimi sırasında Shared Device için konfigürasyonun limit hesaplamasına dahil edilmez. Dolayısıyla Shared Device için maksimum alt modül sayısının veya maksimum periyodik IO verisi miktarının aşılmadığını kendiniz denetlemelisiniz. Maksimum değerlere yönelik bilgi için, kullandığınız cihazların dokümantasyonuna bakın.
- Bir modülün veya alt modülün birden çok IO kontrolörüne atanması gibi konfigürasyon hataları, STEP 7'de algılanmaz.
- Bir Shared-Device konfigürasyonunun yüklendiği CPU'ların, IO cihazının bir Shared-Device olup olmadığı konusunda hiçbir bilgisi yoktur. Diğer IO kontrolörlerine ve dolayısıyla diğer CPU'lara atanan modüller ve alt modüller, yüklenen konfigürasyonda bulunmaz. Dolayısıyla bu modüller ve alt modüller, CPU Web sunucusunda veya CPU ekranında görüntülenmez.

11.5.14.2 Örnek: Bir Shared Device'ın yapılandırılması (GSD konfigürasyonu)

Bu örnek, desantral bir G/Ç sistemini bir Shared Device olarak yapılandırmak için STEP 7'nin nasıl kullanılacağını açıklar.

Farklı IO kontrolörü aileleri için farklı mühendislik araçlarıyla "desantral" bir konfigürasyon mümkündür. Aşağıda açıklanan prosedür, STEP 7'yi temel almakta olup, bir Shared Device kullanan S7-1200 serisinin iki IO kontrolörü ile konfigürasyonu ile sınırlıdır.

Örnekte, her biri bir IO kontrolörüne sahip iki proje oluşturulmaktadır:

- Controller1
- Controller2

Fiziksel olarak aynı IO cihazı olsalar bile Shared Device'ı her iki projede oluşturmalısınız.

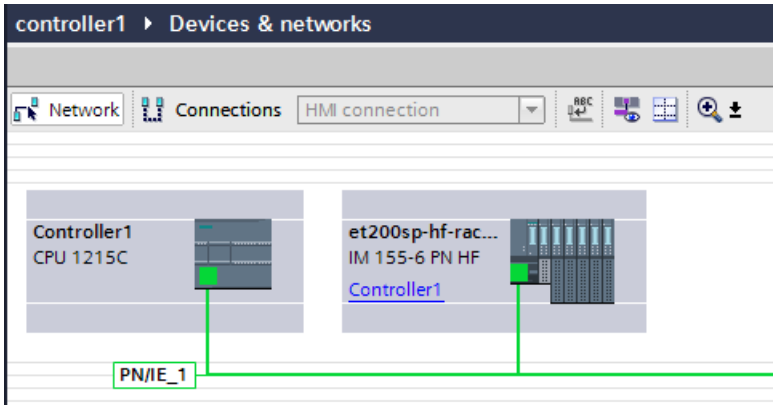
Ön koşullar

- STEP 7
- IO cihazı tarafından Shared Device fonksiyonunun desteklenmesi (örn. ET 200SP IM 155-6 PN HF V3.1).
- IO cihazını bir Shared Device olarak yapılandırmak için GSD dosyası yüklenir.

İzlenecek prosedür: Proje 1'i oluşturun

Bir Shared Device ile ilk projeyi oluşturmak için aşağıdaki yöntemi izleyin:

1. STEP 7'yi başlatın.
2. "Controller1" adlı yeni bir proje oluşturun.
3. Donanım kataloğundan ağ görünümüne bir CPU 1215C sürükleyin. Bunu "Controller1" olarak adlandırın.
4. Donanım kataloğundan "Shared-Device" fonksiyonlu bir IO cihazı (örneğin bir ET 200SP) ekleyin (donanım kataloğu: Diğer saha cihazları > PROFINET IO > G/Ç).
5. IO kontrolörü "Controller1"i, IO cihazına atayın.

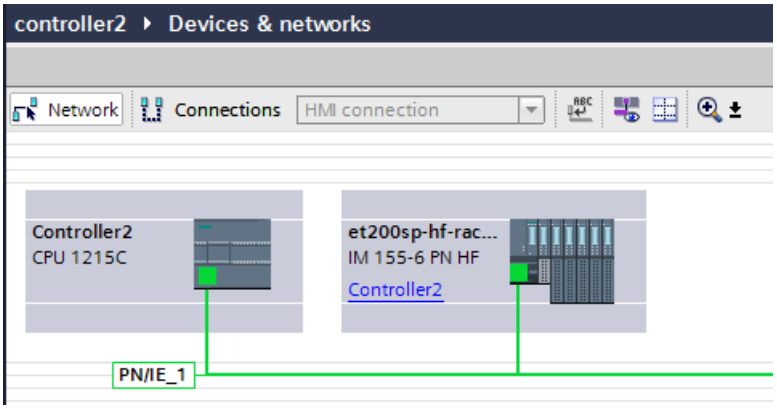


6. IO cihazına çift tıklayın ve donanım kataloğundan gereken tüm modülleri ve alt modülleri cihazın genel bakış tablosuna ekleyin.
7. Modül parametrelerini atayın.
8. Projeyi kaydedin.

İzlenecek prosedür: Proje 2'yi oluşturun

Bir Shared Device ile ikinci projeyi oluşturmak için aşağıdaki yöntemi izleyin:

1. STEP 7'yi yeniden başlatın.
   STEP 7'nin yeni bir örneği açılır.
2. Yeni örnekte "Controller2" adıyla yeni bir proje oluşturun.
3. Ağ görünümüne bir CPU 1215C ekleyin. Bunu, "Controller2" olarak adlandırın.
4. IO cihazını "Controller1" projesinden kopyalayın ve "Controller2" projesinin ağ görünümüne yapıştırın.
5. IO kontrolörü "Controller2"yi, IO cihazına atayın.



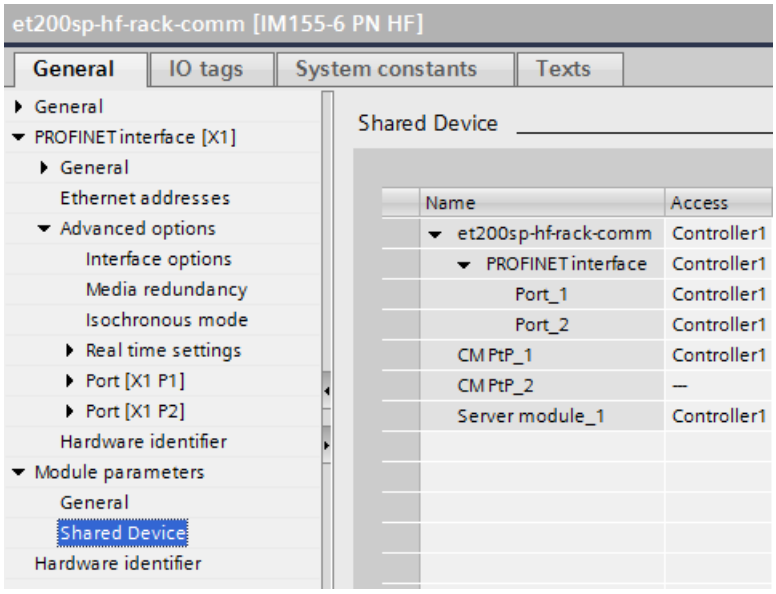
6. Projeyi kaydedin.

Artık her iki projede de aynı şekilde yapılandırılmış birer IO cihazı var ve bu cihazın bir sonraki adımda farklı IO kontrolörü erişimi türleri için yapılandırılması gerekiyor.

İzlenecek prosedür: Shared Device'a erişimi yapılandırın

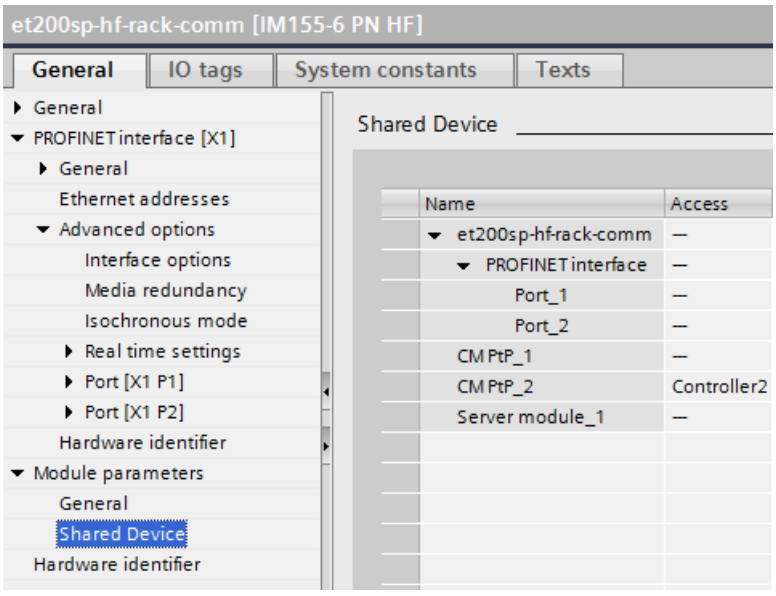
Shared Device'a eklediğiniz modüller ve alt modüller otomatik olarak yerel CPU'ya atanır. Atamayı değiştirmek için aşağıdaki yöntemi izleyin:

1. "Controller1" projesinin ağ veya cihaz görünümünde arayüz modülünü seçin.
2. Denetçi penceresinde "Shared Device" bölümünü seçin.
   Bir tablo, tüm yapılandırılmış modüller için hangi CPU'nun modüllere ve alt modüllere erişimi olduğunu gösterir. Buradaki standart ayar, yerel CPU'nun tüm modüllere ve alt modüllere erişimi olmasıdır.
3. Yerel CPU'nun adres aralığında kalması gereken tüm modüller ve alt modüller için "Controller1" ayarını koruyun.
   "Controller2" projesinden CPU'nun adres alanında yer alacak tüm modüller ve alt modüller için "---" ayarını seçin. Bu, projenin dışındaki bir IO kontrolörünün modüle veya alt modüle erişimi olduğu anlamına gelir.



4. "Controller2" projesinin ağ veya cihaz görünümünde arayüz modülünü seçin.
5. Denetçi penceresinde "Shared Device" bölümünü seçin.
   Bir tablo, tüm yapılandırılmış modüller için hangi CPU'nun modüllere ve alt modüllere erişimi olduğunu gösterir.

6. "Controller1" projesinden CPU'nun adres alanında yer alacak tüm modüller ve alt modüller için "---" ayarını seçin.



7. Son olarak, münferit modüllere ve alt modüllere erişim ayarlarının her iki projede birbirini tamamlayıp tamamlamadığını kontrol edin. Bunun anlamı, yerel CPU'nun bir projede erişimi varsa, diğer projede "---" opsiyonunun ayarlanması gerekmesidir ve bunun tersi de geçerlidir.

   Not: PROFINET arayüzü ve dolayısıyla portlar için "---" opsiyonu, ilişkili parametrelerin yazma korumalı ve değiştirilemez olmalarını sağlar. PROFINET arayüzünün parametreleri ve port parametreleri yalnızca yerel CPU'nun PROFINET arayüzünün atandığı projede düzenlenebilir. Bundan bağımsız olarak, portlar her iki projede birbirlerine bağlanabilir.

8. Tüm projelerde Shared Device için aynı IP adresi parametrelerinin ve aynı cihaz adının belirtilip belirtilmediğini kontrol edin.

   Shared Device'ın bağlı olduğu alt ağ için tüm projelerde aynı S7 alt ağ ID'sinin ayarlanıp ayarlanmadığını kontrol edin (alt ağ özellikleri, denetçi penceresindeki "Genel" alanı).

NOT

Shared Device'da değişiklik yaparsanız: Shared Device'daki her bir projede aynı değişiklikleri gerçekleştirin. Bir modüle veya alt modüle yalnızca bir IO kontrolörünün erişimi olmasına dikkat edin.

İzlenecek prosedür: Gerçek zamanlı ayarları uyarlayın

Tüm IO kontrolörlerinin ve Shared Device'ların uygun gönderim adımı ile çalıştırıldıklarından ve güncelleme sürelerinin iletişim yüküne göre doğru hesaplandığından emin olmak için aşağıdaki ayarları kontrol etmeli ve gerektiğinde uyarlamalısınız:

1. IO kontrolörünün PROFINET arayüzüne ve Shared Device'ın portlarına erişimi olan projeyi seçin.
2. Ağ görünümünde, Shared Device'ın arayüz modülünü seçin.
3. Denetçi penceresinde "PROFINET arayüzü > Gelişmiş opsiyonlar > Gerçek zamanlı ayarlar > IO çevrimi" alanına gidin.
4. "Shared Device" alanında, proje harici IO kontrolörlerinin sayısını belirtin. Maksimum sayı, IO cihazına bağlıdır (GSD dosyasında belirtilir).
5. Shared Device'ın modüllerine ve alt modüllerine erişimi olan her IO kontrolörü için aynı gönderim adımını ayarlamanız gerekir:

- IO kontrolörünü STEP 7 (TIA Portal) ile yapılandırırsanız:
  - İlgili projeyi açın.
  - IO kontrol cihazının PROFINET arayüzünü seçin.
  - Denetçi penceresinde "Gelişmiş opsiyonlar > Gerçek zamanlı ayarlar > IO iletişimi" alanını seçin ve gönderim adımını belirtin.
- IO kontrolörünü başka bir mühendislik aracıyla yapılandırırsanız:
  - STEP 7'de (TIA Portal) Shared Device'ın PROFINET arayüzünü seçin ve Shared Device'ın gönderim adımını okuyun ("Gelişmiş opsiyonlar > Gerçek zamanlı ayarlar" alanı).
  - Mühendislik aracında okunan gönderim adımını girin.

NOT

STEP 7'de (TIA Portal) Shared Device'a erişimi olan tüm IO kontrolörlerini yapılandırdığınızda, IO kontrolöründe, Shared Device tarafından desteklenenden daha kısa gönderim adımları ayarlayabilirsiniz (gönderim adımı uyarlaması).

Derleme ve yükleme

Çeşitli IO kontrolörleri için konfigürasyonları derlemeli ve bunları birbiri ardına CPU'lara yüklemelisiniz.

Ayrı projelerdeki desantral konfigürasyon nedeniyle, erişim parametreleri yanlış ayarlanırsa STEP 7 herhangi bir tutarlılık hatası vermez. Yanlış erişim parametrelendirmesine ilişkin bazı örnekler aşağıda sunulmaktadır:

- Birden çok IO kontrolörü, aynı modüle erişime sahiptir
- IP adresi parametreleri veya gönderim adımları aynı değildir

Bu hatalar sadece kontrolör işletiminde raporlanır ve konfigürasyon hatası olarak yayınlanır.

11.5.14.3 Örnek: Bir I-Device'ın bir Shared Device olarak yapılandırılması

Bu örnekte, V13 SP1 sürümünden itibaren STEP 7 ile bir S7-1200'ün bir I-Device olarak nasıl yapılandırılacağı ve iki projede bir Shared Device olarak nasıl kullanılacağı açıklamaktadır.
Farklı IO kontrolörü aileleri için farklı mühendislik araçlarıyla "desantral" bir konfigürasyon mümkündür. Aşağıda açıklanan prosedür, STEP 7'yi temel almakta olup, bir I-Device'ın aktarım alanını Shared Device olarak kullanan S7-1200 ailesinin iki IO kontrolörü ile konfigürasyonu ile sınırlıdır. I-Device'ın kendisi, bir CPU 1215C'dir.
Örnekte, her biri bir IO kontrolörüne sahip üç proje oluşturulmaktadır:

- S7-1200-I-Device
- Controller1
- Controller2

I-Device'ı, S7-1200 I-Device projesi ile yapılandırırsınız. S7-1200 I-Device'ın PROFINET GSD varyantını Controller1 ve Controller2 projelerinde, ilgili üst seviye IO kontrolöründeki aktarım alanlarını atamak için kullanırsınız.

Shared I-Device konsepti

Shared I-Device konsepti için en az üç ayrı proje gereklidir:

- I-Device projesi: Belirli bir otomasyon görevini gerçekleştirmek için bir I-Device yapılandırır ve programlarsınız. Aktarım alanlarını üst seviyedeki kontrolörler için G/Ç arayüzü olarak tanımlarsınız ve bu aktarım alanlarını farklı GÇ kontrolörlerine atarsınız. Üst seviye IO kontrolörlerine bağlantı için bir PROFINET GSD dosyası hazırlayın ve I-Device'a erişmek için aktarım alanlarını kullanın.
- I-Device'ı ortaklaşa kullanan kontrolörler (iki proje): PROFINET IO sistemi yapılandırırken, I-Device'ı bir PROFINET GSD varyantı olarak kullanırsınız ve IO kontrolörünün aktarım alanlarına eriştiği G/Ç adreslerini belirlersiniz.

NOT

Bir S7-1200'ü I-Device olarak yapılandırırsanız, aktarım alanının maksimum boyutu 1024 giriş veya çıkış Bayt'ı olur. Kumanda cihazındaki yerel G/Ç ve adres bölümü sınırlamalarına bağlı olarak sınırlayıcı faktörler de söz konusudur.

I-Device

Bir S7-1200 CPU için I-Device olarak aşağıdaki parametreleri atarsınız:

- Merkezi ve desantral çevrebirimler
- İstenen aktarım alanları
- Bu I-Device'a erişimi olan IO kontrolörü sayısı (bir Shared Device için her zaman 1'den fazla)

NOT

I-Device'ı üst seviyedeki bir IO kontrolörü olmadan yapılandıramazsınız. Sonuç olarak, aktarım alanındaki adresleri işlemeye dönük kullanıcı programını oluşturmak için yalnızca aktarım alanının yerel G/Ç adreslerini ("I-Device'daki adrese" karşılık gelir) kullanabilirsiniz. I-Device'ı, üst seviyedeki IO kontrolörüne bağlantı haricinde tamamen yapılandırılmış olarak S7-1200 CPU'ya yüklersiniz.

Bir PROFINET GSD dosyasını I-Device konfigürasyonundan dışa aktarırsınız.

I-Device'ı ortaklaşa kullanan kontrolörler

I-Device konfigürasyonundan oluşturulan PROFINET GSD dosyasını, bu Shared I-Device ile bir PROFINET IO sistemini yapılandırırken kullandığınız tüm mühendislik sistemlerine yüklemelisiniz. Bu I-Device'ın tüm kullanımlarını STEP 7 ile yapılandırırsanız, GSD dosyasını STEP 7'de kurmalısınız.
İlgili projelerde I-Device'ı, PROFINET IO sistemindeki bir GSD varyantı olarak yapılandırırsınız. Kurulumdan sonra bu I-Device'ı "Diğer saha cihazları > PROFINET IO > PLC'ler ve CP'ler" altında bulabilirsiniz.
İlgili projelerin her birinde, aktarım alanlarını yalnızca üst seviyedeki IO kontrolörlerine atarsınız (standart ayar: Tümü). Diğer aktarım alanları için "---" (atanmamış) olarak ayarlayın. Bunun sonucunda, yerel IO kontrolörü bu aktarım alanına erişemez durumda olur ve aktarım alanını farklı bir projedeki başka bir IO kontrolörüne atayabilirsiniz.

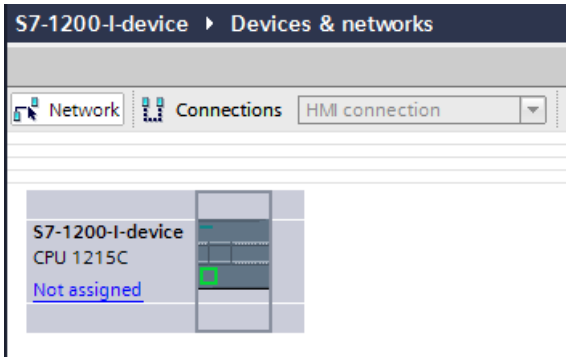
Ön koşullar

- IO cihazı tarafından Shared Device fonksiyonunun desteklenmesi (örn. ET 200SP IM 155-6 PN HF V3.1).
- IO cihazını bir Shared Device olarak yapılandırmak için GSD dosyası yüklenir.

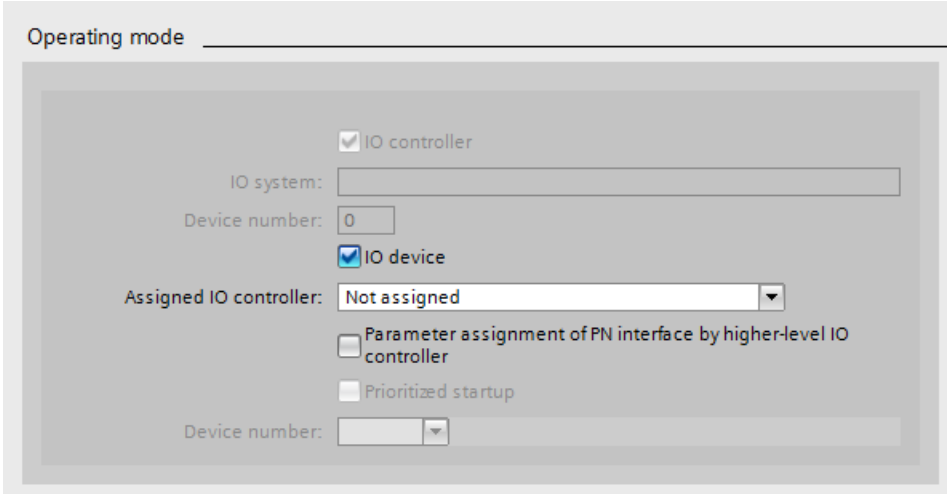
İzlenecek prosedür: S7-1200 I-Device projesi oluşturun

Projeyi, bir Shared I-Device ile oluşturmak için aşağıdaki yöntemi izleyin:

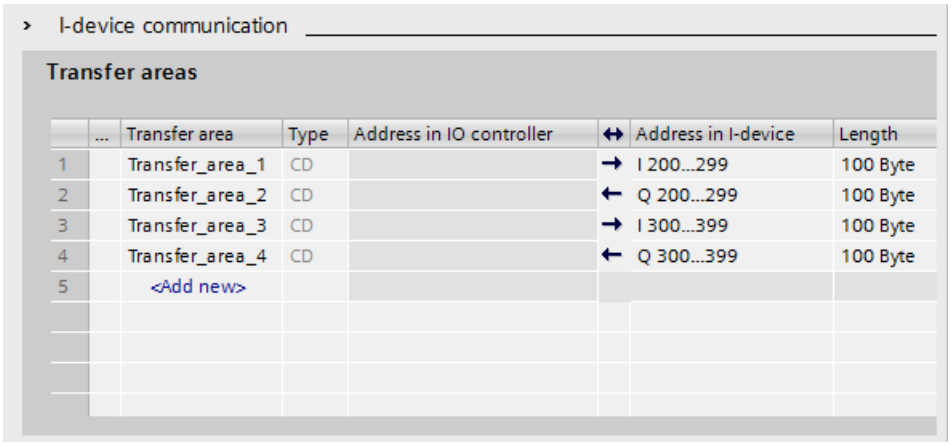
1. STEP 7'yi başlatın.
2. "S7-1200-I-Device" adıyla yeni bir proje oluşturun.
3. Donanım kataloğundan ağ görünümüne bir CPU 1215C sürükleyin. "S7-1200 I-Device " adını atayın.



4. IO cihazına çift tıklayın ve gerekli tüm modülleri ve alt modülleri yapılandırın.
5. Modül parametrelerini atayın. Özellikle PROFINET arayüzü [X1] alanında CPU için aşağıdaki ayarları yapılandırmanız gerekecektir:
   – "İşletim türü" alanında "IO cihazı" opsiyonunu etkinleştirin.

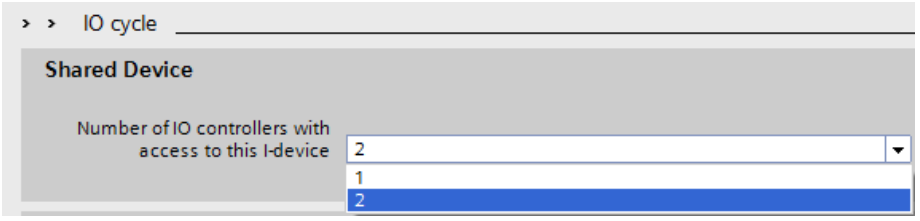


– "İşletim türü > I-Device iletişimi" alanında aktarım alanlarını yapılandırın. "IO kontrolöründeki adres" sütunu, atanan IO kontrolörü olmadığından boş kalır.



Not: Bir giriş alanını bir çıkış alanı olarak değiştirmek veya tam tersini yapmak için ilgili aktarım alanına gitmeniz gerekir.

– İşletim sırasında Shared I-Device'a erişen IO kontrolörü sayısını (en az iki) seçin ("İşletim türü > Gerçek zamanlı ayarlar > IO çevrimi").



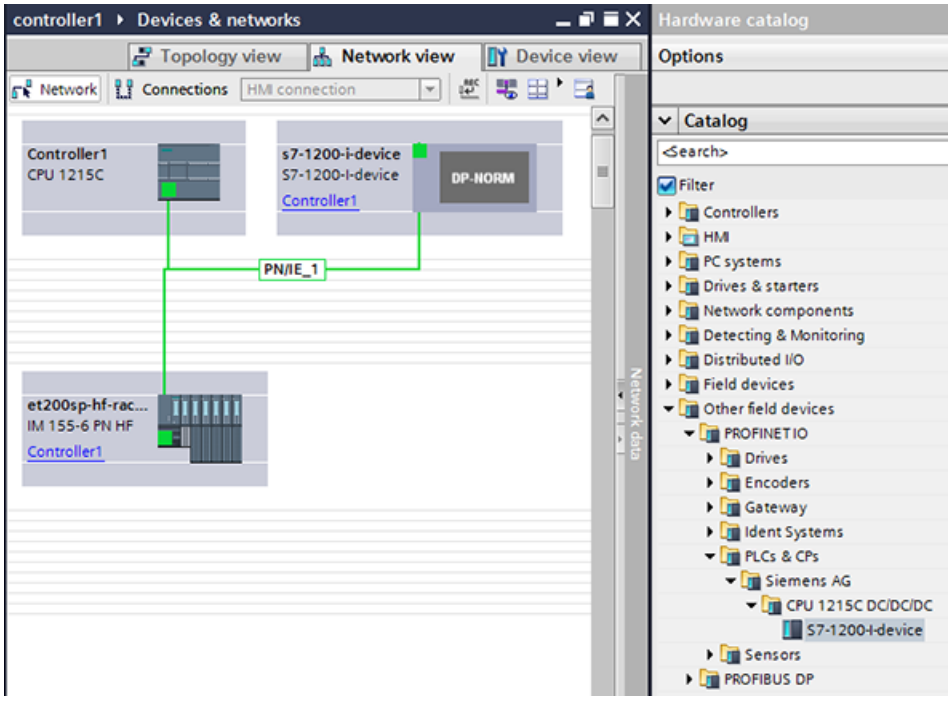
6. Projeyi kaydedin.
7. "Genel" sekmesinde, PROFINET arayüzü [X1] > İşletim türü > I-Device iletişimi öğesini seçin. "Cihaz açıklama dosyasını (GSD) dışa aktar" alanındaki "Dışa aktar" butonuna tıklayın. Dışa aktarma iletişim kutusundaki adı değiştirmezseniz, GSD dosyasına atanan formatta bir ad verilir (örn. "GSDML-V2.31-#Siemens-PreConf_S7-1200-I-Device-20130925-123456").



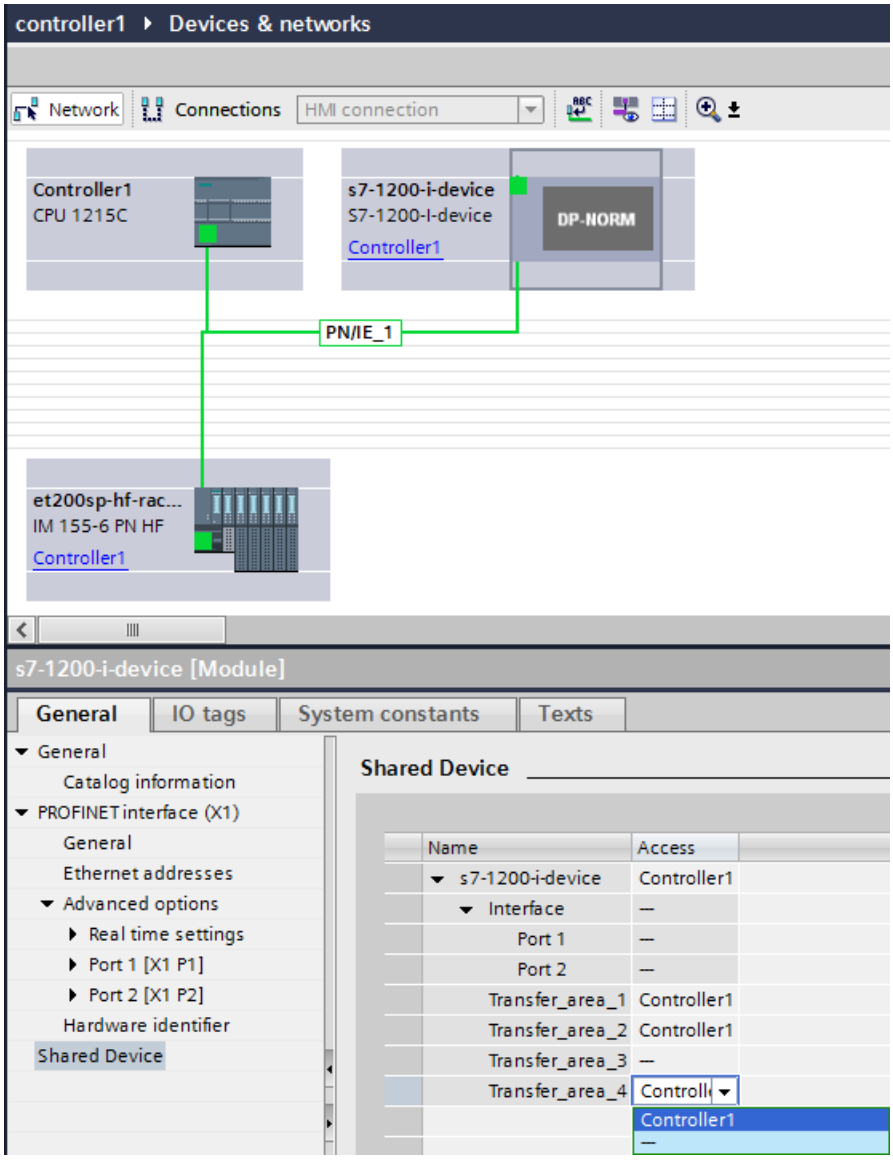
İzlenecek prosedür: Controller1 projesini oluşturun

Bir Shared I-Device ile ilk projeyi oluşturmak için aşağıdaki yöntemi izleyin:

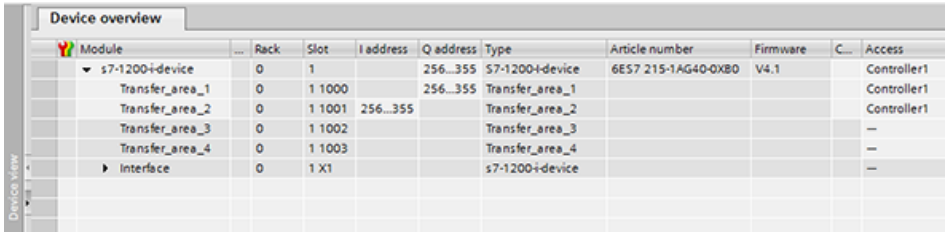
1. STEP 7'yi başlatın.
2. "Opsiyonlar" menüsünde, "Cihaz açıklama dosyalarını yönet (GSD)" girişini seçin.
3. "Cihaz açıklama dosyalarını yönet" iletişim alanında dışa aktarılan PROFINET GSD dosyasını yükleyin.
4. "Controller1" adlı yeni bir proje oluşturun.
5. Ağ görünümüne, CPU 1215C'yi ekleyin. CPU'nun adı "Controller1" olmalıdır.
6. Donanım kataloğundan I-Device'ı ekleyin (donanım kataloğu: Diğer saha cihazları > PROFINET IO > PLC'ler ve CP'ler).
7. I-Device'a "Controller1" IO kontrolörünü atayın.



8. I-Device'ın özelliklerinde "Shared Device" alanını seçin:
   - Tabloda, tüm aktarım alanları ve PROFINET arayüzü yerel IO kontrolörüne (Controller1) atanmış olarak bulunmaktadır.
   - Aktarım alanlarını, Controller1 CPU'sunun **erişimi yoktur** olarak tanımlayın. Bu alanlar için "---" girişini seçin. Bu aktarım alanları, Controller2'nin kullanımına sunulur.



9. Adresleri, cihaza genel bakışta IO kontrolörünün cihaz görünümünden uyarlayabilirsiniz. Cihaza genel bakışı çağırmak için I-Device üzerine çift tıklayın.



10. Projeyi kaydedin.

Prosedür - Controller2 projesini oluşturun

Bir Shared Device ile ikinci projeyi oluşturmak için aşağıdaki yöntemi izleyin:

1. STEP 7'yi yeniden başlatın.
   STEP 7'nin yeni bir örneği açılır.
2. Yeni örnekte "Controller2" adıyla yeni bir proje oluşturun.
3. Ağ görünümüne, CPU 1215C'yi ekleyin. "Controller2" adını atayın.
4. Donanım kataloğundan I-Device'ı ekleyin (donanım kataloğu: Diğer saha cihazları > PROFINET IO > PLC'ler ve CP'ler).
5. I-Device'a "Controller2" IO kontrolörünü atayın.
6. Controller1 projesindeki aktarım alanlarına erişimi ayarlayın. Çifte atama olmadığını kontrol edin.
7. Alt ağın ve PROFINET arayüzünün parametrelerini ayarlayın. Shared I-Device aynı cihazı farklı projelerde kullandığı için bu verilerin eşleşmesi gerekmektedir.
8. Projeyi kaydedin.

Artık her iki projede de aynı şekilde yapılandırılmış bir Shared I-Device bulunmaktadır. IO kontrolörü erişimi ve PROFINET arayüzünün parametreleri bir sonraki adımda çeşitli projelerde kontrol edilmelidir.

Özet - Shared Device'a erişimin parametrelendirilmesi

Aktarım alanları otomatik olarak yerel IO kontrolörüne atanır. Atamayı değiştirmek için aşağıdaki yöntemi izleyin:

1. "Controller1" projesinin ağ görünümünde "S7-1200-I-Device" cihazına tıklayın ve "Shared Device" alanını seçin.
2. Bir tablo, hangi CPU'nun yapılandırılmış aktarım alanlarına erişimi olduğunu gösterir. Buradaki standart ayar, yerel CPU'nun tüm modüllere ve alt modüllere erişimi olmasıdır.
3. Yerel CPU'nun adres aralığında kalması gereken aktarım alanları için "Controller1" ayarını koruyun.
   "Controller2" projesindeki CPU "Controller2" adres alanında olması gereken tüm aktarım alanları için "---" ayarını seçin. Bu, projenin dışındaki bir IO kontrolörünün aktarım alanına erişimi olduğu anlamına gelir.
4. Diğer projeler için de aynı şekilde ilerleyin.
5. Son olarak, münferit modüllere ve alt modüllere erişim ayarlarının her iki projede birbirini tamamlayıp tamamlamadığını kontrol edin. Bunun anlamı, yerel CPU'nun bir projede erişimi varsa, diğer projede "---" opsiyonunun ayarlanması gerekmesidir ve bunun tersi de geçerlidir.
   Not: PROFINET arayüzü ve dolayısıyla portlar için "---" opsiyonu, ilişkili parametrelerin yazma korumalı ve değiştirilemez olmalarını sağlar. PROFINET arayüzünün parametreleri ve port parametreleri yalnızca yerel CPU'nun PROFINET arayüzünün atandığı projede düzenlenebilir. Bundan bağımsız olarak, portlar her iki projede birbirlerine bağlanabilir.
6. Tüm projelerde Shared Device için aynı IP adresi parametrelerinin ve aynı cihaz adının belirtilip belirtilmediğini kontrol edin.
   Shared Device'ın bağlı olduğu alt ağ için tüm projelerde aynı S7 alt ağ ID'sinin ayarlanıp ayarlanmadığını kontrol edin (alt ağ özellikleri, denetçi penceresindeki "Genel" alanı).

NOT

I-Device'da değişiklik yaparsanız (örneğin aktarım alanlarının sayısını veya uzunluğunu değiştirmek gibi), I-Device'ı bir GSD dosyası olarak yeniden dışa aktarın. I-Device'ın paylaşılan bir cihaz olarak kullanıldığı her bir projede GSD dosyasını yeniden yükleyin. Bir aktarım alanına yalnızca bir IO kontrolörünün erişimi olmasını sağlayın.

NOT

S7-1200'ü paylaşılan bir I-Device ve bir kontrolör olarak kullanıyorsanız, iletişim performansı üzerindeki etkiyi azaltmak için PROFINET I-Device ve PROFINET IO'nun güncelleme sürelerini uzattığınızdan emin olmalısınız. Sistem çok stabildir ve tek bir PROFINET I-Device için 2 msn güncelleme süresi ve tek bir PROFINET IO için 2 msn güncelleme süresi seçtiğinizde düzgün çalışır.

PROFINET IO cihazının veya IO'nun "Özellikler" konfigürasyon iletişim kutusunda "IO çevrimi" için olan parametreleri belirlersiniz. Daha fazla bilgi için bkz. "IO çevrim süresinin yapılandırılması" (Sayfa 690).

Prosedür - Gerçek zamanlı ayarları uyarlayın

Tüm IO kontrolörlerinin ve Shared Device'ların uygun gönderim adımı ile çalıştırıldıklarından ve güncelleme sürelerinin iletişim yüküne göre doğru hesaplandığından emin olmak için aşağıdaki ayarları kontrol etmeli ve gerektiğinde uyarlamalısınız:

1. Shared Device'ın modüllerine ve alt modüllerine erişimi olan her IO kontrolörü için aynı gönderim adımını ayarlamanız gerekir:

- IO kontrolörünü STEP 7 (TIA Portal) ile yapılandırırsanız, aşağıdaki yöntemi izleyin:
  - İlgili projeyi açın.
  - IO kontrol cihazının PROFINET arayüzünü seçin.
  - Denetçi penceresinde "Gelişmiş opsiyonlar > Gerçek zamanlı ayarlar > IO çevrimi" alanını seçin ve gönderim adımını belirtin.
- IO kontrolörünü başka bir mühendislik aracıyla yapılandırırsanız aşağıdaki yöntemi izleyin:
  - STEP 7'de (TIA Portal) Shared Device'ın PROFINET arayüzünü seçin ve Shared Device'ın gönderim adımını okuyun ("Gelişmiş opsiyonlar > Gerçek zamanlı ayarlar" alanı).
  - Mühendislik aracında okunan gönderim adımını girin.

NOT

STEP 7'de (TIA Portal) Shared I-Device'a erişimi olan **tüm** IO kontrolörlerini yapılandırdığınızda, IO kontrolöründe, Shared Device tarafından desteklenenden daha kısa gönderim adımları ayarlayabilirsiniz (gönderim adımı uyarlaması).

Derleme ve yükleme

Çeşitli IO kontrolörleri için konfigürasyonları derlemeli ve bunları birbiri ardına CPU'lara yüklemelisiniz.

Ayrı projelerdeki desantral konfigürasyon nedeniyle, erişim parametreleri yanlış ayarlanırsa STEP 7 herhangi bir tutarlılık hatası vermez. Yanlış erişim parametrelendirmesine ilişkin bazı örnekler aşağıda sunulmaktadır:

- Birden çok IO kontrolörü, aynı modüle erişime sahiptir.
- IP adresi parametreleri veya gönderim adımları aynı değildir.

Bu hatalar sadece kontrolör işletiminde raporlanır ve konfigürasyon hatası olarak yayınlanır.

11.5.15 Medya yedeklilik protokolü (MRP)

Aşağıdaki CPU'lar, MRP yöneticisi ve istemci olarak MRP fonksiyonelliğini destekler.

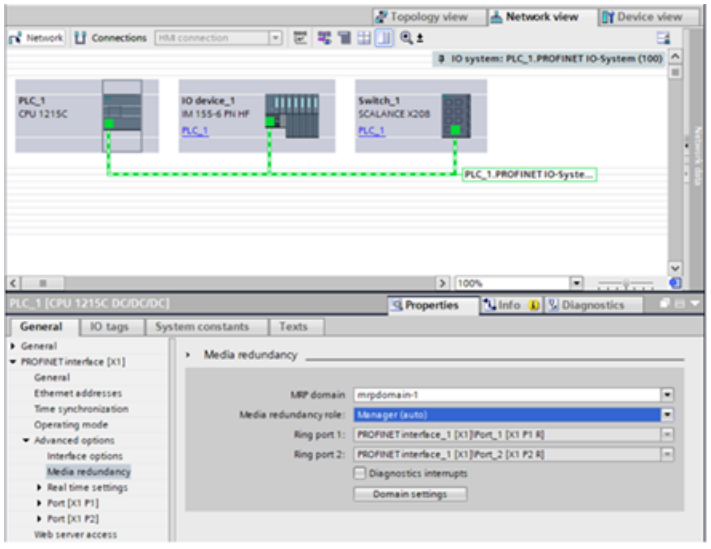
- CPU 1215C
- CPU 1217C
- CPU 1215FC

1215/1217, MRP paketlerinin arayüz üzerinden iletildiği ve bağlantı kesintilerinin ağdaki bir yöneticiye bildirildiği bir MRP halkasında bir "İstemci" olarak çalıştırılmalıdır.

Bir "Yönetici" olarak, 1215/1217'nin ağ üzerinde MRP paketleri göndermesi, halkadaki açık portları algılaması, bloke edilen portları yönetmesi ve diğer potansiyel yöneticilerle yönetici durumuna ilişkin görüşmesi gerekir.

S7-1200 CPU ailesinin, MRP istemcisi veya MRP yöneticisi olarak işletimi başlatmak için MRP protokolünü ve konfigürasyon parametrelerini desteklemek üzere üç modeli (yukarıya bakın) bulunmaktadır. Bu CPU'larda iki PN portu vardır.

Bu üç CPU ile S7-1200 ile istemci veya yönetici olarak bir MRP halkası oluşturabilirsiniz. CPU yönetici olarak hareket ettiğinde, ilgili halkanın kesintiye uğramadığından emin olmak ve denetlemek için test telegramları kullanırlar. İstemci olarak hareket ederken, testi kendisi yapmak yerine bu test telegramlarını bir sonraki üniteye iletir. Böylece S7-1200, gerektiğinde her iki rolde de kullanılabilir.



Resim 11-1 TIA Portal'da medya yedekliliği

MRP yöneticisi (otomatik) ve yönetici
S7-1200'ü (1215C/1217C), bir medya yedekleme halkasında (MRP), yönetici ve (Otomatik) yönetici olarak kullanabilirsiniz. MRP, cihazların bir halka konfigürasyonunda bağlanmasına olanak tanır. MRP yöneticisi tipik olarak, halkanın bir portunu bloke eder ve verilerin doğrusal yönde akmasına yol açar. Bu halka konfigürasyonunda bir kesinti ortaya çıktığında, yönetici kesintiyi algılar ve portun blokajını kaldırarak verilerin diğer yönde akmasını sağlar. MRP ile bir tel kopması veya bir cihazın devre dışı kalması durumunda bile bir ağ çalışır durumda kalabilir. MRP spesifikasyonu, bir halka konfigürasyonunda maksimum 50 cihaza izin verir.

MRP yöneticisi
MRP yöneticisi, S7-1200 PLC'nin yedeklilik yöneticisi olarak çalışmasını sağlar. Bu rol için yapılandırıldığında S7-1200, istemci PLC'ler ile iletişim kurmak ve halkadaki bağlantının kopmadığından emin olmak için test telegramlarından yararlanır. Bir istemcide kesinti algılanırsa, MRP yöneticisi olarak görev yapan S7-1200, halka portundaki istemci cihazları değişiklik hakkında bilgilendirir ve hemen halka portlarını kullanır. TIA Portal'da, bir MRP halkasında yalnızca bir cihaz MRP yöneticisi olarak ayarlanabilir.

MRP yöneticisi (otomatik)
Birden çok cihaza MRP yöneticisi (otomatik) rolü atanırsa, bunlar, kendi aralarında yönetici durumuna ilişkin görüşme yürütür. Üzerinde anlaşmaya varılan MRP yöneticisi konfigürasyondan çıkarıldığında, geriye kalan "Yönetici (otomatik)" ayarına sahip cihazlar, başlangıçtaki konfigürasyon geri yüklenene kadar kendi aralarında yönetici rolünü yeniden görüşür. Başlangıçtaki yönetici başarılı bir şekilde geri dönerse, yönetici durumunu geri alır ve orijinal konfigürasyon devam eder. MRP yöneticisi konfigürasyonları, S7-1200'de (1215C/1217C) yalnızca V4.x'ten itibaren mümkündür.

NOT

Halka yeniden yapılandırması

Halkanın yeniden yapılandırılması 200 msn'ye kadar olabilir. Dolayısıyla, her cihazda PROFINET denetleme süresi için 200 msn'den büyük bir değer ayarlanmalıdır.

NOT

MRP yöneticisi (otomatik) standart ayardır

Herhangi bir proje mevcut değilse, CPU, standart olarak MRP yöneticisi (otomatik) modundadır. Çalışan bir cihazı, halka olmayan bir topolojiye bağladığınızda ve ağınızda test telegramlarını fark ettiğinizde, bu bilgi önemli olacaktır.

S7-1200 (1214C, 1212C ve 1211C CPU'lar), MRP yöneticisi ve MRP yöneticisi (otomatik) fonksiyonlarını etkinleştiremez. TIA Portal'daki menü opsiyonları bu CPU'larda mevcut değildir.

S7-1200, IRT özellikli olmadığından (izokron gerçek zamanlı) S7-1200, MRPD'yi (planlı kopyalama için medya yedekliliği) desteklemez.

Teşhis alarmları, yönetici ve istemcinin aşağıdakiler gibi ilgili MRP değişiklik raporları sağlamasını olanaklı kılmak için, TIA Portal'daki medya yedekleme konfigürasyonu penceresinde açılıp kapatılabilir:

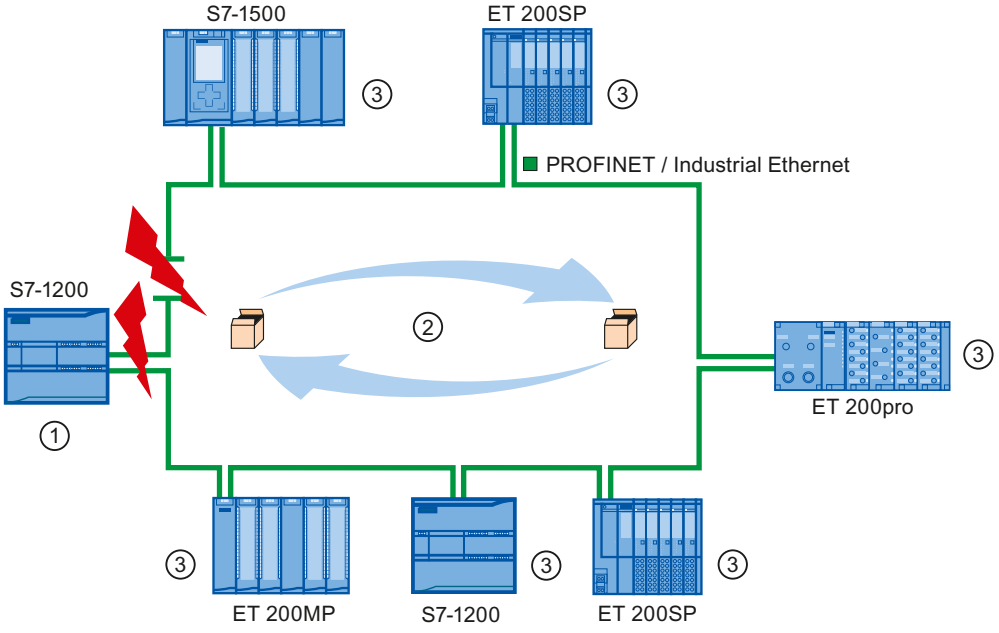
- Halka portun bir komşu cihazı, MRP'yi desteklemez.
- Bir halka portu, halka portu olmayan bir port komşusuna bağlanır.
- Bir halka portu, başka bir MRP etki alanına bağlıdır.
- (Yalnızca yönetici) MRP halkasında kesinti olduğunda ve başlangıçtaki orijinal konfigürasyon geri döndüğünde, kesintilerde/geri dönüşte teşhis alarmları.

11.5.15.1 Halka topolojilerinde medya yedekliliği

Optik veya elektriksel doğrusal veriyolu topolojilerine sahip bir Industrial Ethernet ağındaki ağ kullanılabilirlik oranını artırmak için, ağdaki uçları birbirine bağlayarak bir doğrusal veriyolu topolojisini halka topolojisine dönüştürebilirsiniz.

Bir halka topolojisindeki cihazlar, IO cihazları, IO kontrolörleri, harici Switch'ler ve/veya iletişim modüllerinin entegre Switch'leri olabilir.

Medya yedekliliği olan bir halka topolojisi oluşturmak için, bir doğrusal veriyolu topolojisinin iki serbest ucunu tek bir cihazda bir araya getirmelisiniz. Doğrusal veriyolu topolojisini kapatmak ve bir halka oluşturmak için halkadaki bir cihazın iki portuna (halka portları) gerek duyarsınız. Meydana gelen halkadaki bir cihaz daha sonra MRP yöneticisi rolünü üstlenir. Halkadaki diğer tüm cihazlar MRP istemcileri olur.



① MRP yöneticisi (bu, bir S7-1500 CPU, S7-1200 CPU 1215C, S7-1200 CPU 1215FC veya S7-1200 CPU 1217C olabilir)

② Test telegramları

③ MRP istemcileri

Bir cihazın halka portları, halka topolojisindeki iki komşu cihazla bağlantıyı kuran portlardır. Halka portlar, ilgili cihazın konfigürasyonunda seçilir ve belirtilir (muhtemelen ön ayar da olabilir).

Bir halka topolojisindeki medya yedekliliğinin çalışma şekli

Halka herhangi bir noktada kesintiye uğrarsa, münferit cihazlar arasındaki veriyolları otomatik olarak yeniden yapılandırılır. Yeniden yapılandırma sonrası cihazlar yeniden kullanılabilir.

MRP yöneticisinde, bir kesintisiz ağ işletimi sırasında normal iletişim için iki halka portundan biri engellenir, böylece veri telegramları dolaşımda kalmaz. MRP yöneticisi, kesintiler açısından halkayı denetler. Bunun için yönetici, halka portu 1 ve halka portu 2'den test telegramları gönderir. Test telegramları, MRP yöneticisinin diğer halka portuna ulaşana dek halkanın etrafında her iki yönde hareket eder.

Halkadaki herhangi bir kesinti, iki cihaz arasındaki bağlantının kesilmesinden veya halkadaki bir cihazın arızalanmasından kaynaklanıyor olabilir.

MRP yöneticisinin test telegramları, halkadaki bir kesinti sırasında artık diğer halka portuna gelmezse, MRP yöneticisi iki halka portu birbirine bağlar. Bu yedek yol, geri kalan tüm cihazlar arasında doğrusal bir veriyolu topolojisi şeklinde çalışan bir bağlantıyı yeniden oluşturur.

Halkanın kesintiye uğraması ile fonksiyonel bir doğrusal topolojinin geri yüklenmesi arasındaki süre, yeniden yapılandırma süresi olarak adlandırılır.

Kesinti giderildikten sonra, orijinal aktarım yolları geri yüklenir, MRP yöneticisinin iki halka portunun bağlantısı kesilir ve MRP istemcileri değişiklik hakkında bilgilendirilir. MRP istemcileri daha sonra diğer cihazlara giden başlangıçtaki orijinal yolları tekrar kullanır.

Medya yedeklilik yöntemi

SIMATIC kapsamındaki standart medya yedeklilik yöntemi, 200 msn'lik tipik yeniden yapılandırma süresine sahip medya yedeklilik protokolüdür (MRP). Sistemde halka başına 50'ye varan sayıda cihaz katılabilir.

NOT

Halka yeniden yapılandırması

Halkanın yeniden yapılandırılması 200 msn'ye kadar olabilir. Her cihaz için PROFINET denetleme süresi, 200 msn'den büyük bir değere ayarlanmalıdır.

11.5.15.2 Medya yedeklilik protokolünün (MRP) kullanımı

MRP prosedürü, IEC 61158 tip 10 "PROFINET" kapsamında belirtilen MRP medya yedeklilik protokolüne uyarınca çalışır.

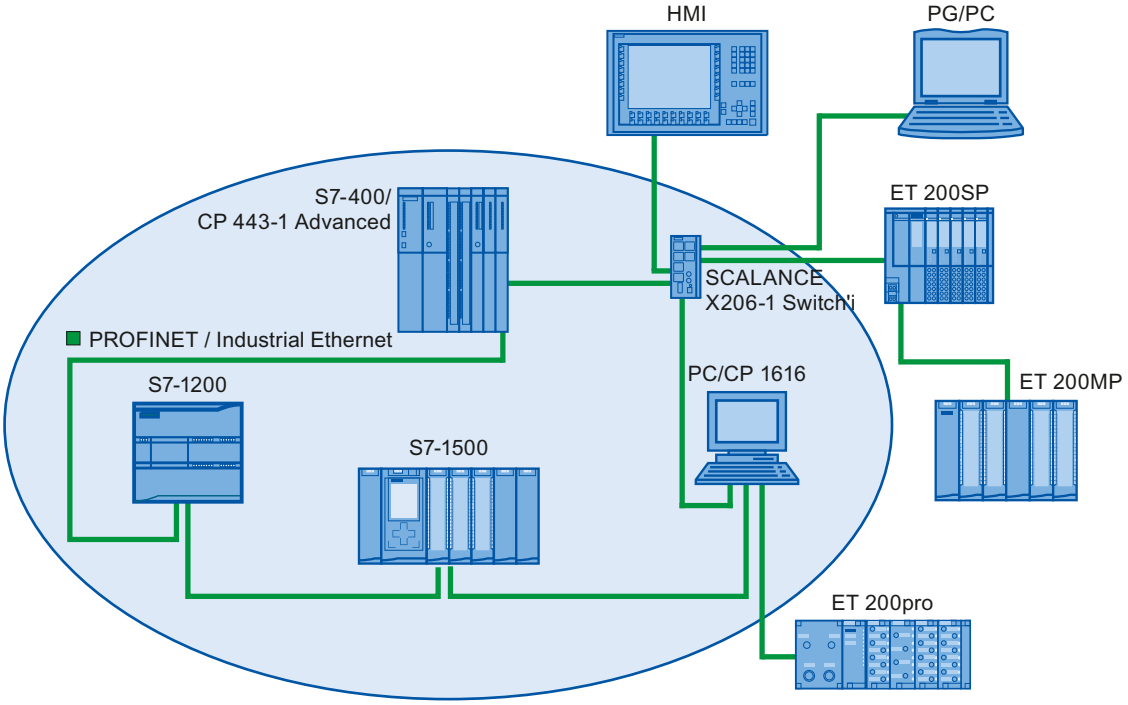
Ön koşullar

MRP ile hatasız bir işletim için aşağıdaki ön koşullar karşılanmalıdır:

- MRP'yi kullanmak istediğiniz halka, yalnızca bu fonksiyonu destekleyen cihazlar içermelidir.
- "MRP", halkadaki tüm cihazlar için etkinleştirilmelidir.
- Tüm cihazlar birbirlerine halka portları üzerinden bağlanmalıdır.
- En az bir MRP yöneticisi veya "Yönetici" rolü mevcut olmalıdır.
- Halkanın maksimum 50 cihaz içermesine izin verilir. Aksi halde 200 msn veya daha uzun yeniden yapılandırma süreleri oluşabilir.
- Halkadaki tüm partner portlar aynı ayarlara sahip olmalıdır.

Topoloji

Aşağıdaki şekilde, MRP'li bir halkadaki cihazlar için olası bir topolojiyi gösterilmektedir. Gölgelendirilmiş oval alandaki cihazlar, yedeklilik etki alanındadır.
Bu, MRP'li bir halka topolojisi örneğidir:



Aşağıdaki kurallar, medya yedekliliği ve MRP içeren bir halka topolojisi için geçerlidir:

- Halkadaki tüm cihazlar aynı yedeklilik etki alanına aittir.
- Halkadaki bir cihaz, MRP yöneticisi veya MRP yöneticisi (otomatik) rolü için yapılandırılmalıdır.
  - Halkadaki yalnızca bir cihaz MRP yöneticisi olabilir. Diğer tüm cihazlar MRP istemcisi olmalıdır.
  - Bir MRP yöneticisi yoksa, birden fazla cihaz MRP yöneticisi (otomatik) olarak yapılandırılabilir.

MRP özelliğine sahip olmayan cihazları, halka portları olarak yapılandırılmamış portlar üzerinden ağa bağlayabilirsiniz. Bu olanak, yalnızca ikiden fazla porta sahip cihazlar için kullanılabilir (örn. SCALANCE X Switch'e veya CP1616'e sahip bir PC).

Sınır koşulları

Aşağıdaki iletişim olanaklarına sahipsiniz:

- MRP özelliği ile MRP ve RT: RT işletimi mümkündür.

  NOT

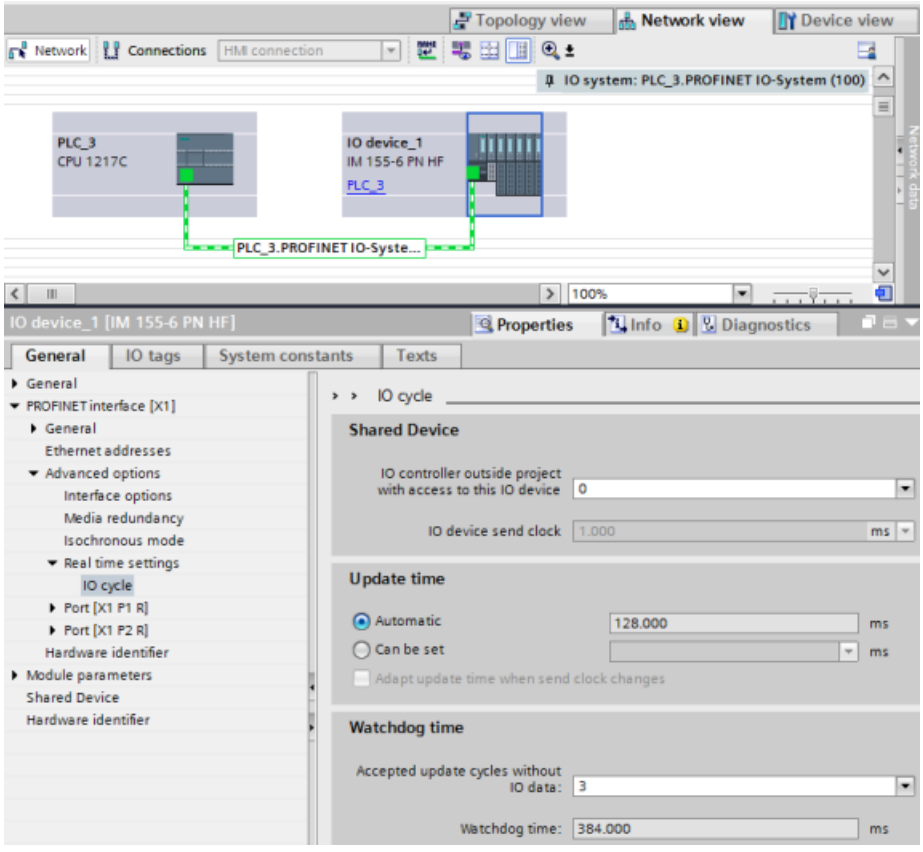
  Halkanın yeniden yapılandırma süresi, IO cihazının seçilen denetleme süresinden uzunsa, RT iletişimi kesintiye uğrar (istasyon arızası). IO cihazlarınız için 200 msn'den daha uzun bir denetleme süresi seçmelisiniz. Daha fazla bilgi için aşağıdaki "Denetleme süresi" alt bölümüne bakın.

- MRP ve TCP/IP (TSEND, HTTP, ...): MRP ile TCP/IP iletişimi mümkündür, çünkü kayıp veri paketleri yeniden gönderilebilir.
- MRP ve öncelikli yol alma:
  - MRP'li bir halkada yapılandırırsanız, söz konusu cihazlardaki PROFINETuygulamalarında "öncelikli yol alma" fonksiyonunu kullanamazsınız.
  - "Öncelikli yol alma" fonksiyonunu kullanmak istiyorsanız, konfigürasyonda MRP'yi devre dışı bırakmalısınız (cihaz halkanın bir parçası olamaz).
- İkiden fazla portlu PROFINET cihazlarında MRP: Bir halkada ikiden fazla porta sahip bir PROFINET cihazı işletiyorsanız, halkada olmayan portlara bir senkronizasyon sınırı belirlemeniz gerekir. Senkronizasyon sınırı ayarlamasıyla senkronizasyon etki alanı için bir sınır tanımmış olursunuz. Senkronizasyon telegramlarını bir senkronizasyon etki alanındaki senkronize cihazlara iletemezsiniz.

Denetleme süresi

Bir IO kontrolörünün veya IO cihazının IO verilerini almadan izin verdiği zaman aralığı, PROFINET denetleme süresi olarak adlandırılır. Denetleme süresi sırasında IO cihazına IO kontrolörü tarafından veri sağlanmazsa, cihaz, eksik telegramları algılar ve yedek değerler yayınlar. Bu, IO kontrolöründe bir istasyon arızası olarak bildirilir.
PROFINET IO cihazları denetleme süresini yapılandırabilirsiniz. Denetleme süresini doğrudan değil, "IO verileri olmadan kabul edilen güncelleme çevrimleri" olarak girin. Burada ortaya çıkan denetleme süresi, güncelleme çevrimlerinin sayısından otomatik olarak hesaplanır.
Denetleme süresini atamak için aşağıdaki yöntemi izleyin:

1. Ağ veya cihaz görünümünde, IO cihazının PROFINET arayüzünü seçin.
2. Arayüz özelliklerinde iken şuraya gidin: Gelişmiş opsiyonlar > Gerçek zamanlı ayarlar > IO Çevrimi.
3. Açılır listeden gerekli çevrim sayısını seçin.



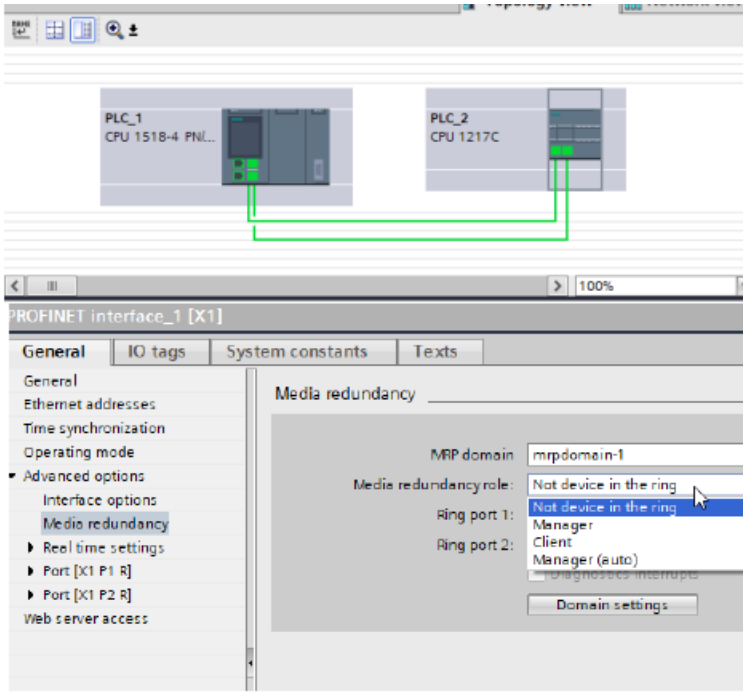
11.5.15.3 Medya yedekliliğinin yapılandırılması

Uygulamanızdaki tüm bileşenler, medya yedeklilik protokolünü (MRP) desteklemelidir.

Prosedür

Medya yedekliliğini yapılandırmak için aşağıdaki yöntemi izleyin:

1. Uygun port ara bağlantılarına sahip bir halka oluşturun (örn. topoloji görünümünde).
2. Medya yedekliliğini yapılandırmak istediğiniz bir PROFINET cihazı seçin.
3. Denetçi penceresinde, "PROFINET arayüzü [X1] > Gelişmiş opsiyonlar Medya yedekliliği" öğesine gidin.



4. "Medya yedekliliği rolü" altında, cihaza "Yönetici (otomatik)", "İstemci" veya "Halkada cihaz yok" rollerinden birini atayın.
   TIA Portal'daki topoloji görünümünde bir halka yapılandırırsanız, TIA Portal, sizin için medya yedekliliği rolünü otomatik olarak belirler. Bir cihaz, yönetici olabiliyorsa, TIA Portal, medya yedekliliği rolünü "Yönetici (otomatik)" olarak belirler. S7-1200 ile "İstemci" veya "Yönetici" otomatik olarak medya yedekliliği rolü olarak tanımlanır.

   NOT

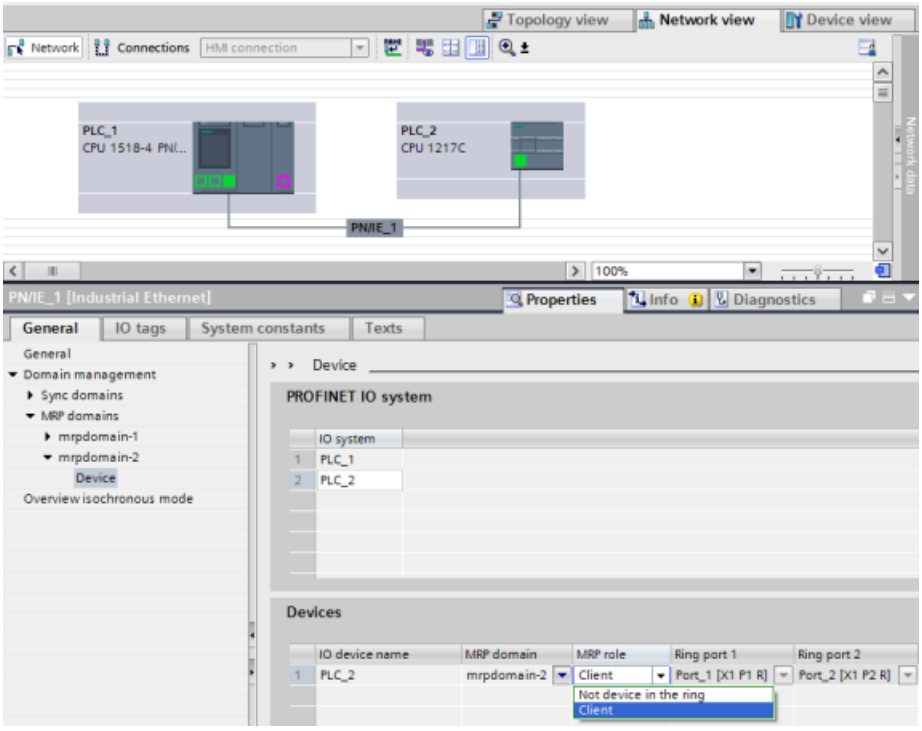
   S7-1200 CPU 1215C/1215FC/1217C sisteminde, "Yönetici" veya "Yönetici (otomatik)" MRP rolleri atayabilirsiniz.

5. Halkadaki tüm PROFINET cihazları için 2 ile 4 arasındaki adımları tekrarlayın.

Veya:

1. Ağ görünümünde PROFINET IO sistemini işaretleyin.
2. PROFINET IO sistemine tıklayın.

3. Denetçi penceresinde gerekli MRP etki alanının cihazına gidin.



4. PROFINET cihazlar için "Yönetici (otomatik)", "İstemci" veya "Halka katılımcısı olmayan" rollerinden birini atayın.

"Medya yedekliliği" opsiyonu: MRP rolü

Kullanılan cihaza bağlı olarak "Yönetici", "Yönetici (otomatik)", "İstemci" ve "Halka katılımcısı olmayan" rolleri mevcuttur.

Kurallar:

- Bir halkada, "Yönetici" rolüne sahip yalnızca bir cihaz olabilir. "Yönetici" veya "Yönetici (otomatik)" rolüne sahip başka hiçbir cihaza izin verilmez. Halkadaki diğer tüm cihazlar "İstemci" rolündedir. Halkaya dahil olmayan diğer tüm cihazlar "Halka katılımcısı olmayan" rolüne sahip olabilir.
- Halkada "Yönetici" rolüne sahip bir cihaz yoksa, halkanın "Yönetici (otomatik)" rolüne sahip en az bir cihazı olmalıdır. Bir halka, "İstemci" ve "Yönetici (otomatik)" rollerine sahip herhangi bir sayıda cihaz içerebilir.

"Medya yedekliliği" opsiyonu: Halka portu 1 ve halka portu 2

Halka portu 1 veya halka portu 2 olarak yapılandırmak istediğiniz portları seçin. Açılır liste, her cihaz tipi için olası portların seçimini gösterir. Portlar fabrikada ayarlanmışsa, ilgili alanlar kullanılamaz.

NOT

S7-1200 CPU'da yalnızca iki port olduğundan, halka portlarının S7-1200 için yapılandırılması gerekli değildir.

Teşhis alarmları

MRP durumu için teşhis kesintileri yerel CPU'da yayınlanacaksa, "Teşhis alarmları" opsiyon kutusunu etkinleştirin. Aşağıdaki teşhis alarmları yapılandırılabilir:

- Kablaj veya port hatası:
  CPU, halka portlarında aşağıdaki hatalar için teşhis alarmları oluşturur:
  - Halka portun bir komşu cihazı, MRP'yi desteklemez.
  - Bir halka portu, halka portu olmayan bir port komşusuna bağlanmıştır.
  - Bir halka portu, başka bir MRP etki alanından bir halka portuna bağlıdır.
- Kesinti/Geri dönüş (yalnızca MRP yöneticisinde):
  Halka kesintiye uğrar ve orijinal konfigürasyon yayınlanırsa, CPU teşhis alarmları üretir. Bu alarmların her ikisi de 0,2 saniye içinde meydana gelirse, bu durum, halkada bir kesinti olduğunu gösterir.

Teşhis hatası OB'sinde (OB 82) uygun reaksiyonu programlayarak kullanıcı programında bu olaylara ilgili tepkiyi verebilirsiniz.

NOT

MRP yöneticisi olarak harici cihazlar

Bir halkada MRP yöneticisi olarak bir harici cihaz kullanıldığında hatasız işletimi sağlamak için, halkayı kapatmadan önce halkadaki diğer tüm cihazlara "İstemci" sabit rolünü atamanız gerekir. Aksi halde veri telegramlarının dolaşımı ve ağ arızası meydana gelebilir.

11.5.16 S7-Routing

STEP 7'deki ağ görünümünde, farklı S7 alt ağlarındaki cihazları bağlayarak karmaşık bir iletişim topolojisi oluşturabilirsiniz. Klasik S7-300/S7-400 CPU'ları ve CP'lerin yanı sıra en yeni S7 CPU'ları ve CP'leri birbirine bağlayabilir ve ağa, HMI cihazlarını ve OPC sunucusu gibi PC istasyonlarını da entegre edebilirsiniz.

STEP 7'de hangi cihazların iletişim kurması gerektiğine karar verdikten ve gerekli bağlantıları kurduktan sonra, mühendislik sistemi (ES), donanım konfigürasyonunun bir bileşeni olarak çeşitli S7 yönlendiricilere uygun yönlendirme tablolarını yükleyebilir. Yönlendirme tablosunu çeşitli S7 yönlendiricilerine yükledikten sonra, ES ve diğer iletişim partnerleri, cihazlar farklı S7 alt ağlarında olsa bile her cihazla iletişim kurabilir. Bu, araya bağlanan CPU'lar ve/veya CP'lerin S7 yönlendiricileri gibi davranması nedeniyle mümkün olmaktadır. CPU'lar ve/veya CP'ler, gelen bağlantı taleplerini, bağlantı talebi hedef cihaza ulaşana ve cihazlar S7 bağlantısını kurana dek bir sonraki S7 yönlendiriciye iletir.

CPU, yerel temel cihazdaki CP cihazlarının gerektirdiği yönlendirme tablolarını aktarmak için veri setlerini yazma prosedürünü kullanır. Yönlendirme tabloları, bir uzak S7 alt ağının ID'sini içeren bir bağlantı talebi sırasında bir cihazdan diğerine giden yolu belirler. Bağlantı talebini alan cihaz, yönlendirme tablosuna sorar, yol üzerindeki bir sonraki istasyonu S7 hedef alt ağına bulur ve bağlantı talebini bir sonraki üniteye aktarır. Bu şekilde, bağlantı talebi sonunda amaçlanan hedefine ulaşır ve ilgili yanıt, rotayı aksi yönde kat eder.

S7-1200 CPU'lar, tek bir PN arayüze ve yerel iletişim veriyoluna bağlı üç adede kadar CP cihazına sahiptir. Bu nedenle, S7-1200 istasyonu içinde yönlendirme için iki olanağınız vardır:

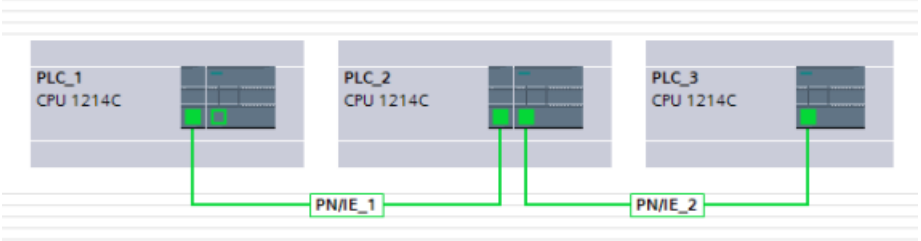
- CPU ve CP arasında yönlendirme
- Bir CP'den başka bir CP'ye yönlendirme

S7-Routing (https://support.industry.siemens.com/cs/us/en/view/584459) fonksiyonunu destekleyen diğer tüm S7-1200 CP'ler ile ilgili daha fazla bilgiyi, Siemens'in Industry Online Support web sitesinde bulabilirsiniz.

11.5.16.1 CPU ve CP arayüzleri arasında S7-Routing

S7-1200 CPU'ları tek bir PN arayüzü ile sınırlı olduğundan, bağımsız bir CPU yönlendirici görevi göremez. Bağımsız bir CPU'yu yalnızca bir S7 alt ağına bağlayabilirsiniz. CPU'nun yerel temel cihazına CP modülleri kurarsanız, birden çok S7 alt ağını birbirine bağlayabilir ve yönlendirmeyi kullanabilirsiniz.

Aşağıdaki örnek sistemde mühendislik sistemi (ES), PLC_1'in PLC_3 ile iletişim kurabilmesini sağlamak için, mesajların PLC_2 üzerinden yönlendirmesi gerekmektedir. ES, PLC_2 için yönlendirme tablosunu yüklemeli ve PLC_2, yerel temel cihazda CP modülü için yönlendirme tablosunu sağlamalıdır. Bu yönlendirme tabloları mevcutsa, PLC_1 ve PLC_3 doğrudan birbirlerine bağlı olmasalar bile, birbirleriyle iletişime geçebilirler.

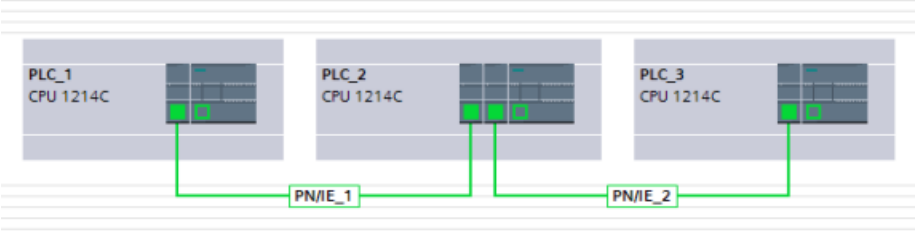


Bir S7 alt ağından diğer S7 alt ağına yönlendirmeyi kontrol etmek için PLC_1, PLC_3 ile bir taşıma bağlantısı ve PLC_3, PLC_1 ile bir bağlantı kurmalıdır. PLC'nin PN/IE arayüzünden bir CP

modülüne ve bir CP modülünden PLC'nin PN/IE arayüzüne yönlendirmenin mümkün olmasını sağlayın.

11.5.16.2 İki CP arayüzü arasında S7-Routing

S7-1200 CPU'ları CP modüllerine kadar desteklediği için, üç modülü de farklı S7 alt ağlarına bağlayabilirsiniz. CPU'nun yerel temel cihaza en az iki CP modülü kurar ve bunları farklı S7 alt ağlarına bağlarsanız, yönlendirmeyi kullanabilirsiniz.
Aşağıdaki örnek sistemde, PLC_1'in PLC_3 ile iletişim kurabilmesi için mühendislik sisteminin (ES), PLC_2'den gelen mesajları CP modülünden yerel temel cihazdaki CP modülüne yönlendirmesi gerekmektedir. ES, PLC_2 için yönlendirme tablosunu yüklemeli ve PLC_2, iki CP modülü için yönlendirme tablosunu sunmalıdır. Bu yönlendirme tabloları mevcutsa, PLC_1 ve PLC_3 doğrudan birbirlerine bağlı olmasalar bile, birbirleriyle iletişime geçebilirler. Buna ek olarak, CP modülünden CP modülüne olan yönlendirmenin, PLC_2'nin PN/IE arayüzü üzerinden mesajlar gönderilmeden gerçekleştiğine dikkat edin.



11.5.17 SNMP

Simple Network Management Protocol (SNMP - Basit ağ yönetimi protokolü), IP ağlarında yönetilen cihazlara ilişkin verileri toplamak ve düzenlemek ve ilgili cihazın davranışını değiştirmeye yönelik olarak bu verileri değiştirmek amacıyla kullanılan bir internet standardı protokolüdür. Yönlendiriciler, Switch'ler, sunucular, iş istasyonları, yazıcılar, modem rafları ve benzerleri, SNMP'yi genellikle destekleyen cihazlardır.
Ağ yönetim sistemleri, ağ yöneticinin müdahalesini gerektiren koşullara dönük ağa bağlı cihazları denetlemek için SNMP'yi kullanır. SNMP, ağ topolojisini algılamak ve teşhis işlemleri için çeşitli hizmetler ve araçlar kullanır. SNMP yetkin cihazların özelliklerine ilişkin bilgileri, MIB (Management Information Base - Yönetim Bilgi Tabanı) dosyalarında bulabilirsiniz. MIB dosyalarına erişmek için ilgili yetkilere gerek duyarsınız.
SNMP, yönetim verilerini değişkenler olarak, sistem konfigürasyonunda açıklanmış olan yönetilen sistemlere aktarır. Yönetim uygulamaları, bu değişkenleri çağırabilir (ve kısmen ayarlayabilir).
SNMP, UDP taşıma protokolünden (User Datagram Protocol - Kullanıcı Datagram Protokolü) yararlanır ve iki ağ bileşenine sahiptir:

- SNMP yöneticisi: Ağ katılımcılarını denetler.
- SNMP istemcisi: Münferit ağ katılımcılarındaki çeşitli ağa özgü bilgileri kaydeder ve bunları yapılandırılmış bir biçimde Yönetim Bilgi Tabanında (MIB) saklar. Bu verileri ayrıntılı ağ teşhisi amaçları için kullanabilirsiniz.

S7-1200 CPU'larda, güvenlik nedenleriyle SNMP değişkenlerine okuma ve yazma erişimini engelleyen SNMP, standart olarak devre dışı bırakılmıştır. SNMP değişkenlerine uzaktan erişim ve bunların düzenlenmesi için SNMP'yi etkinleştirmeniz gerekmektedir.

NOT

V4.6'dan önceki sürüme sahip S7-1200 CPU'larda, SNMP standart olarak aktiftir. Firmware'i V4.6'dan önceki bir sürüme düşürürseniz, SNMP'yi yalnızca STEP 7 programı aracılığıyla devre dışı bırakılabilirsiniz.

SNMP'yi etkinleştirdikten sonra, SNMP'yi belli bazı durumlarda devre dışı bırakmak gerekebilir. Buna ilişkin örnekler:

- Ağınızın güvenlik ayarları SNMP kullanımına izin vermiyor.
- Kendi SNMP çözümünüzü kullanırsınız (örn. kendi iletişim yordamlarınızdan yararlanırsınız).

Cihazınızda SNMP devre dışı bırakılmışsa, SNMP değişkenlerine erişim gerektiren bazı özellikler ve araçlar artık kullanılamaz veya fonksiyonlarını yitirir. Örnek olarak, SIMATIC Automation Tool (SAT) V4.x veya önceki sürümlerde bazı fonksiyonlarda olduğu gibi, TIA Portal'daki topoloji algılama çalışmaz.

Cihaz konfigürasyonu ve STEP 7 programı üzerinden, SNMP'yi etkinleştirebilir veya devre dışı bırakabilirsiniz.

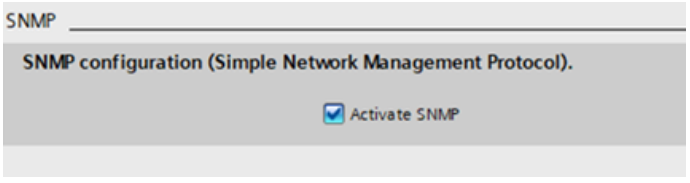
NOT

CPU'nun fabrika ayarına sıfırlanması, SNMP'yi devre dışı bırakır.

Cihaz konfigürasyonu üzerinden SNMP'nin etkinleştirilmesi ve devre dışı bırakılması

SNMP'yi etkinleştirmek veya devre dışı bırakmak için aşağıdaki yöntemi izleyin:

1. Proje gezgininde, CPU'nuzun altında Cihaz konfigürasyonuna çift tıklayın.
2. Cihaz özelliklerinin Genel sekmesinde Genişletilmiş konfigürasyon öğesini genişletin ve burada SNMP'yi seçin.
3. SNMP'yi etkinleştirmek için "SNMP'yi etkinleştir" opsiyon kutusunu seçin veya SNMP'yi devre dışı bırakmak için opsiyon kutusundaki seçim işaretini kaldırın.

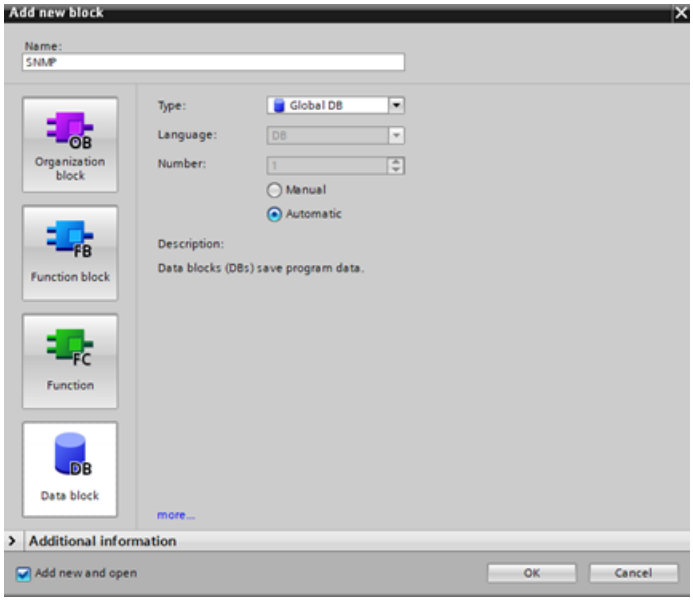


4. Seçilen konfigürasyonu CPU'ya yükleyin.

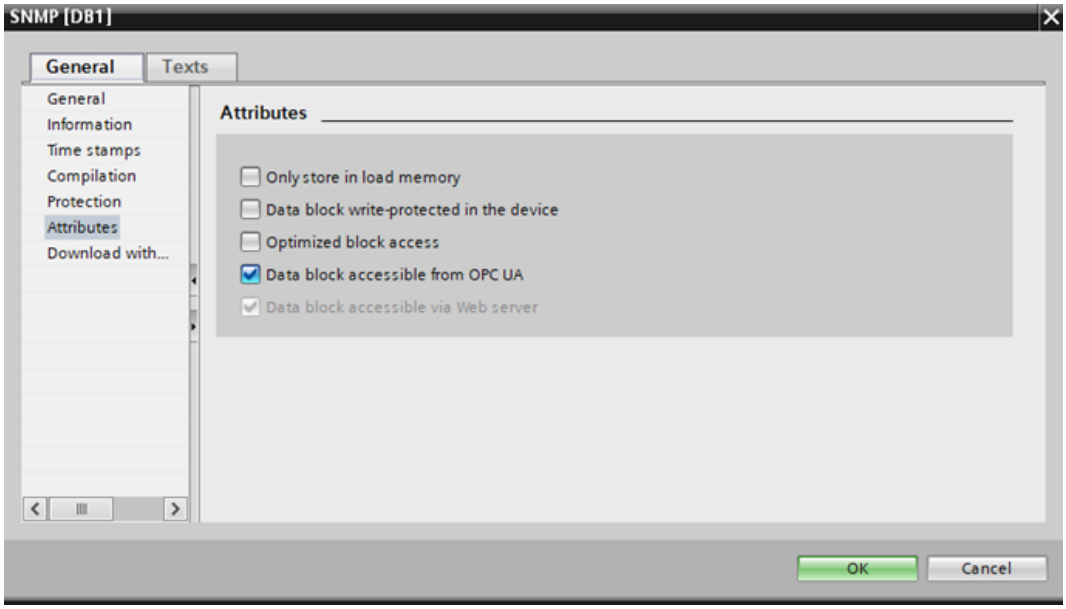
STEP 7 programı üzerinden SNMP'yi etkinleştirme ve devre dışı bırakma

SNMP'yi etkinleştirmek veya devre dışı bırakmak için aşağıdaki yöntemi izleyin:

1. Bir veri bloğu (DB) oluşturun:

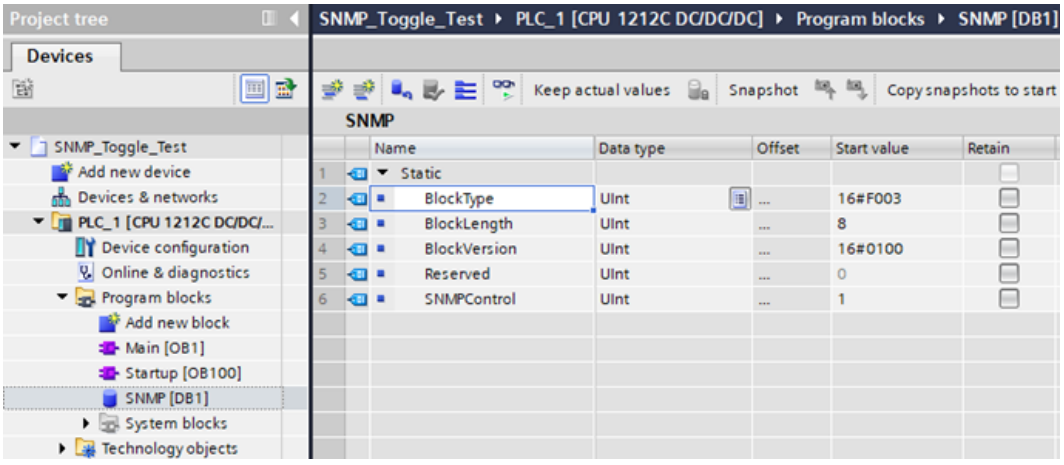


2. Yeni oluşturulan DB'nin özelliklerini seçin.
3. "Öznitelikler" sekmesini açın. "Optimize edilmiş blok erişimi" opsiyon kutusunu devre dışı bırakın:

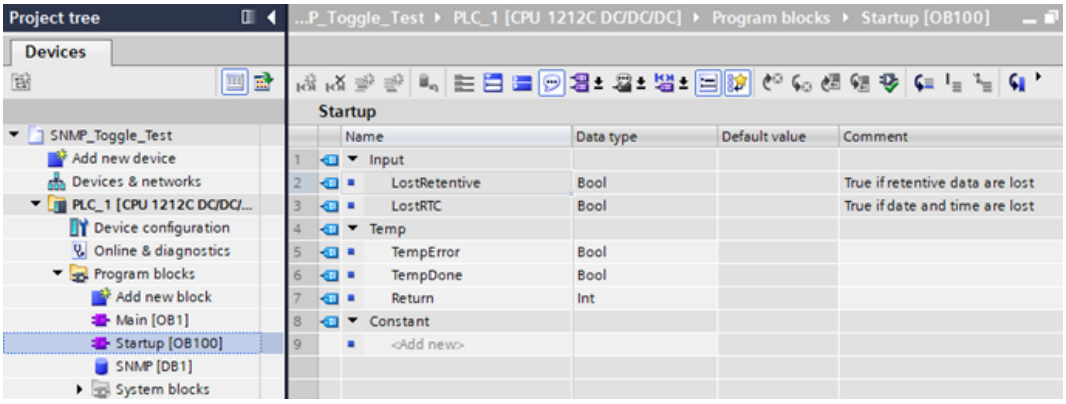


4. "TAMAM" butonuna tıklayın.
   İlgili talep ortaya çıkarsa, programınızı yeniden derleyin.

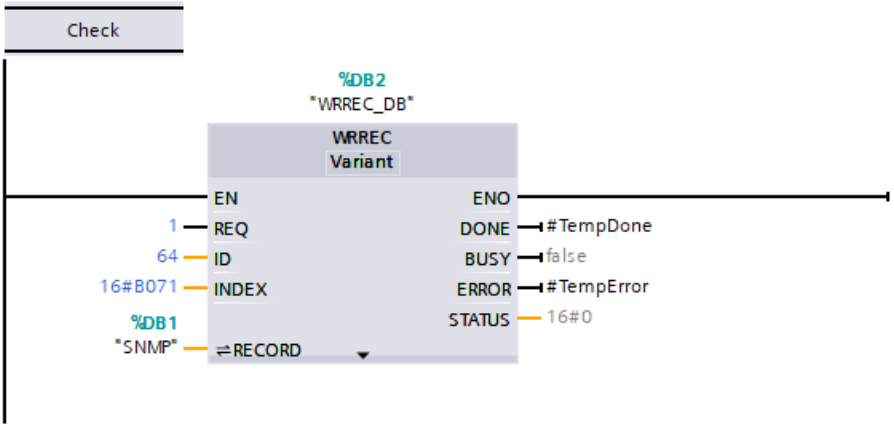
5. DB veri bloğu arayüzünde, gösterilen değerlerle aşağıdaki statik değişkenleri oluşturun. Bu değişkenleri programınızda kullanarak, dahili SNMP uygulamasını etkinleştirmek veya devre dışı bırakabilirsiniz. SNMP'yi etkinleştirmek veya devre dışı bırakmak için, SNMPControl değişkeninin Başlangıç değerini değiştirin. Başlangıç değeri 1'in girişi, SNMP'yi etkinleştirir, öte yandan Başlangıç değeri 0 ise SNMP'yi devre dışı bırakır.



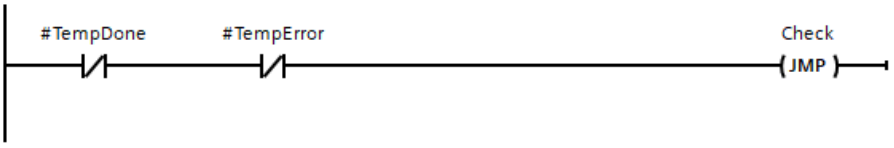
6. Yol alma OB'sinde (OB100), geçici değişkenleri gösterildiği gibi ekleyin:



7. LAD editörünü kullanarak yol alma OB'sinin (OB100) ağ 1'ine bir atlama işareti ekleyin. Aşağıdaki örnekte atlama işaretinin adı: "Kontrol et". Bunun ardından, aynı ağ üzerinde, gösterilen giriş ve çıkışlarla bir WRREC (Write Record) yordamı ekleyin:



8. Ağ 2 için aşağıdaki kodu girin. Bu kod, yol alma OB'sinden çıkmadan önce WRREC çağrısının tamamlanmasını ve SNMP'nin etkinleştirilmesini veya devre dışı bırakılmasını sağlar:



9. STEP 7 proje yazılımını CPU'ya yükleyin.

11.5.18 Teşhis

Bu iletişim ağlarıyla teşhis için organizasyon bloklarının (OB'ler) nasıl kullanılacağına ilişkin bilgileri "Organizasyon blokları (OB'ler) (Sayfa 68)" altında bulabilirsiniz.

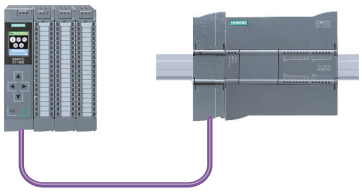
11.5.19 Desantral çevrebirimler için yordamlar ve teşhis yordamları

Desantral çevrebirimler için yordamlar ve teşhis yordamlarına ilişkin bilgileri şurada bulabilirsiniz: Desantral G/Ç (PROFINET, PROFIBUS veya AS-i) (Sayfa 352).
Teşhis olaylarının işlenmesine ilişkin bilgileri, "Desantral çevrebirim için kullanılan teşhis olayları (Sayfa 438)" altında bulabilirsiniz.

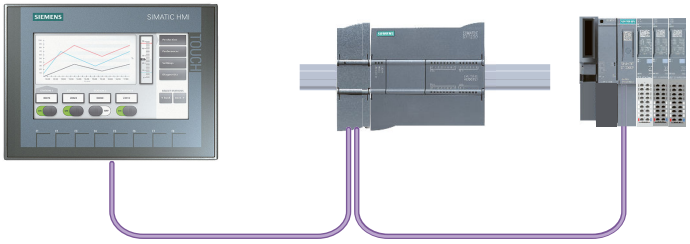
11.6 PROFIBUS

Bir PROFIBUS sistemi, bir RS485 seri veriyolu üzerinde birden fazla konuma dağıtılmış olan Slave cihazları çağırmak için bir veriyolu Master'i kullanır. Bir PROFIBUS Slave, verileri işleyen ve çıktıyı Master'a gönderen herhangi bir çevrebirim cihazıdır (G/Ç dönüştürücü, valf, motor sürücüsü veya ölçüm cihazı). Slave, ağda bir pasif istasyonu temsil eder, çünkü veriyolu erişim yetkisi yoktur. Yalnızca alınan mesajları onaylayabilir veya talep üzerine Master'a yanıt mesajları gönderebilir. Tüm PROFIBUS Slave'leri aynı önceliğe sahiptir ve tüm ağ iletişimi Master'dan kaynaklanır.
Bir PROFIBUS Master, ağda bir "aktif istasyonu" temsil eder. PROFIBUS DP'de, iki Master sınıfı tanımlanmaktadır. 1. sınıf bir Master, kendisine atanmış Slave'lerle olağan iletişimi veya veri alışverişini yönetir (bu, genellikle merkezi olarak programlanabilir bir kumanda ünitesi (PLC) veya özel yazılım yüklenen bir PC'dir). 2. sınıf bir Master, temelde Slave'leri devreye alma ve teşhis amaçlarına yönelik kullanılan özel bir cihazdır (bu, genellikle bir konfigürasyon cihazıdır, örneğin işletime alma, bakım ve teşhis için bir dizüstü bilgisayar veya bir programlama konsoludur).
S7-1200, bir DP-Slave olarak CM 1242-5 iletişim modülü ile bir PROFIBUS ağına bağlanır. CM 1242-5 (DP-Slave), V0/V1 DP-Master'ların iletişim partneri olabilir. Modülü bir harici sistemde yapılandırmak istiyorsanız, modülün teslim kapsamındaki CD'de ve Siemens Industry Online Support (https://support.industry.siemens.com/cs/ww/en/view/49852283) internet sayfalarında CM 1242-5 (DP-Slave) için bir GSD dosyası bulacaksınız.

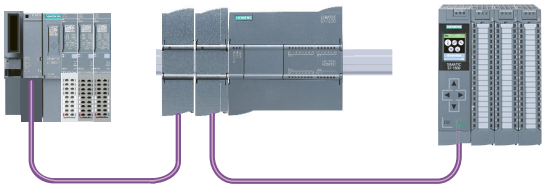
Aşağıdaki şekilde S7-1200, bir S7-1500 kontrolörün DP-Slave olarak çalışmasını görmektesiniz:



S7-1200, bir DP-Master olarak CM 1243-5 iletişim modülü ile bir PROFIBUS ağına bağlanır. CM 1243-5 (DP-Master), V0/V1 DP-Slave'lerin iletişim partneri olabilir. Aşağıdaki şekilde, S7-1200 bir Master'dır ve bir ET200SP DP-Slave'i kumanda eder.



Bir CM 1242-5 ve bir CM 1243-5 birlikte kurulursa, bir S7-1200 aynı anda hem daha üst seviyedeki bir DP-Master sisteminin Slave'i hem de daha alt seviyedeki bir DP-Slave sisteminin Master'ı olarak çalışabilir.



İstasyon başına maksimum üç PROFIBUS CM yapılandırabilirsiniz; bunlar, DP-Master-CM'lerin veya DP-Slave-CM'lerin herhangi bir kombinasyonu olabilir. DP-Master'ların her biri maksimum 32 Slave'i kontrol edebilir.

PROFIBUS CM'lerin konfigürasyon verileri yerel CPU'da saklanır. Bunun anlamı, bu iletişim modüllerinin gerektiğinde kolayca değiştirilebilmesidir.

PROFIBUS'u kullanmak için PROFIBUS Master CM Firmware'i en azından V1.3 olmalıdır.

NOT

PROFIBUS CM Firmware'ini her zaman en güncel sürümünde tutun (http://support.automation.siemens.com/WW/view/en/42131407). Bir Firmware güncellemesini aşağıdaki yöntemlerden birini kullanarak gerçekleştirebilirsiniz:

- STEP 7'nin çevrimiçi ve teşhis araçları (Sayfa 1120) ile
- Bir SIMATIC memory card (Sayfa 124) ile
- Web sunucusunun "Modül bilgileri" standart web sitesi üzerinden (Sayfa 819)
- SIMATIC Automation Tool (https://support.industry.siemens.com/cs/ww/en/view/98161300) ile

---

11.6.1 PROFIBUS CM'ler üzerinden iletişim hizmetleri

PROFIBUS CM'ler, PROFIBUS DP-V1 protokolünü kullanır.

DP-V1 için iletişim tipleri

DP-V1 için aşağıdaki iletişim tipleri mevcuttur:

- Periyodik iletişim (CM 1242 5 ve CM 1243 5)
  Her iki PROFIBUS modülü de DP-Slave ve DP-Master arasında proses verilerinin aktarımı için periyodik iletişimi destekler.
  Periyodik iletişim, CPU'nun işletim sistemi tarafından yürütülür. Bunun için herhangi bir yazılım modülü gerekmez. G/Ç verileri, doğrudan CPU'nun proses görünümünden/görünümüne okunur veya yazılır.
- Aperiyodik iletişim (yalnızca CM 1243-5)
  DP-Master, yazılım modülleri ile aperiyodik iletişimi de destekler.
  - Alarm işleme için "RALRM" yordamı mevcuttur.
  - "RDREC" ve "WRREC" yordamları, konfigürasyon ve teşhis verilerinin aktarımı için mevcuttur.

CM 1243-5 tarafından desteklenmeyen fonksiyonlar: SYNC/FREEZE ve Get_Master_Diag

CM 1243-5'in ek iletişim hizmetleri

CM 1243 5 DP-Master, aşağıdaki ek iletişim hizmetlerini destekler:

- S7 iletişimi
  - PUT/GET hizmetleri
    DP-Master, PROFIBUS aracılığıyla diğer S7 kumanda ünitelerinden veya PC'lerden gelen sorgular için bir istemci ve sunucu görevi görür.
  - PG/OP iletişimi
    PG fonksiyonları ile konfigürasyon verileri ve kullanıcı programlarını bir PG'den indirilebilir ve teşhis verilerini bir PG'ye yükleyebilirsiniz.
    OP iletişimi için olası iletişim partnerleri, HMI panelleri, WinCC flexible özellikli SIMATIC panel PC'ler veya S7 iletişimini destekleyen SCADA sistemleridir.

11.6.2 PROFIBUS CM'ler için kullanıcı kılavuzlarına ilişkin not

Daha fazla bilgi

PROFIBUS CM'ler hakkında ayrıntılı bilgileri, ilgili cihazların işletme kitaplarında bulabilirsiniz. Bunları, internetteki Siemens Industrial Automation müşteri destek sayfalarında aşağıdaki ID'ler altında bulabilirsiniz:

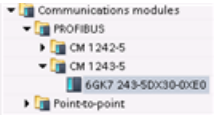
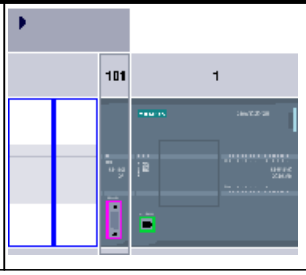
- CM 1242-5 (https://support.industry.siemens.com/cs/ww/en/ps/15667)
- CM 1243-5 (https://support.industry.siemens.com/cs/ww/en/ps/15669)

11.6.3 DP-Master ve DP-Slave'in yapılandırılması

11.6.3.1 CM 1243-5 (DP-Master) ve DP-Slave eklenmesi

PROFIBUS modüllerini CPU'ya eklemek için, "Cihazlar ve Ağlar" portalındaki donanım kataloğunu kullanın. Bu modüller, CPU'nun sol tarafına bağlanır. Donanım konfigürasyonuna bir modül eklemek için, ilgili modülü donanım kataloğunda seçin ve çift tıklayın veya işaretlenen yuvanın üzerine sürükleyin.

Tablo 11-61 PROFIBUS CM 1243-5'in (DP-Master), cihaz konfigürasyonuna eklenmesi

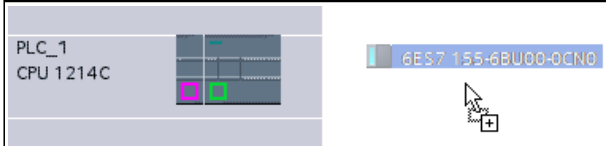
Modül	Modül seçme	Modül kullanma	Sonuç
CM 1243-5 (DP-Master)			

Donanım kataloğunda DP-Slave'leri de ekleyebilirsiniz. Örneğin bir ET200 SP DP-Slave'i eklemek için donanım kataloğunda aşağıdaki klasörleri genişletin:

- Desantral G/Ç
- ET 200SP
- Arayüz modülü
- PROFIBUS

Daha sonra, sipariş numaraları listesinden "6ES7 155-6BU00-0CN0" (IM155-6 DP HF) öğesini seçin ve aşağıdaki şekilde gösterildiği gibi ET200SP DP-Slave'i takın.

Tablo 11-62 Cihaz konfigürasyonuna ET200SP DP-Slave eklenmesi

DP-Slave eklenmesi	Sonuç
	

11.6.3.2 İki PROFIBUS cihazı arasındaki mantıksal ağ bağlantılarının yapılandırılması

CM 1243-5 (DP-Master) modülünü yapılandırdıktan sonra, şimdi ağ bağlantılarınızı yapılandırabilirsiniz.

"Cihazlar ve ağlar" portalında, "Ağ görünümü" altında projenizdeki cihazları birbirine bağlayabilirsiniz. PROFIBUS bağlantısı oluşturmak üzere ilk cihazdaki mor alanı (PROFIBUS) etkinleştirin. Fareyi kullanarak, ikinci cihazdaki PROFIBUS alanına bir hat çekin. Fare tuşunu bırakın. Böylece PROFIBUS bağlantınız artık oluşturulmuştur.

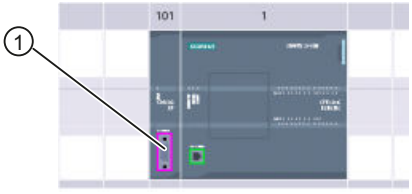
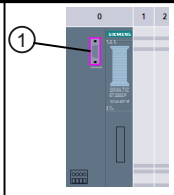
Ayrıntılı bilgi için bkz. "Cihaz konfigürasyonu: Ağ bağlantısının oluşturulması" (Sayfa 562).

11.6.3.3 CM 1243-5 ve DP-Slave'e PROFIBUS adreslerinin atanması

PROFIBUS arayüzünün yapılandırılması

İki PROFIBUS cihazı arasındaki mantıksal ağ bağlantılarını yapılandırılmasının ardından, PROFIBUS arayüzleri için parametreleri yapılandırabilirsiniz. Bunu yapmak için, CM 1243-5 modülündeki mor renkli PROFIBUS alanına tıklayın. Bunun ardından, denetçi penceresindeki "Özellikler" sekmesi PROFIBUS arayüzünü gösterir. DP-Slave'in PROFIBUS arayüzü bu şekilde yapılandırılır.

Tablo 11-63 CM 1243-5 (DP-Master) ve ET200SP DP-Slave'in PROFIBUS arayüzlerinin yapılandırılması

CM 1243-5 (DP-Master)	ET 200SP DP-Slave
	

① PROFIBUS portu

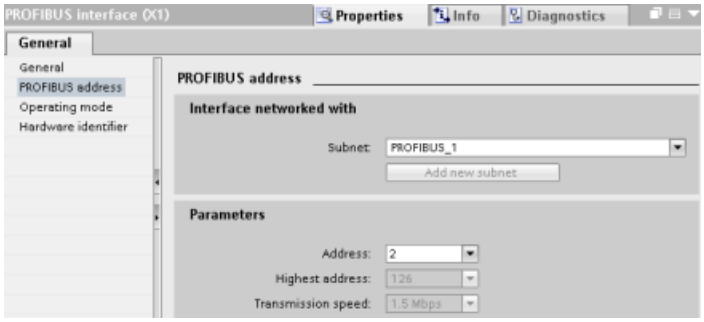
PROFIBUS adresinin atanması

Bir PROFIBUS ağında her cihaza birer PROFIBUS adresi atanır. Bu adres, aşağıdaki istisnalar dışında 0 ile 127 aralığında bulunur:

- Adres 0: Veriyoluna bağlı ağ konfigürasyonu ve/veya programlama araçları için ayrılmıştır
- Adres 1: İlk Master için Siemens tarafından rezerve edilmiştir
- Adres 126: Fabrikada şalter ayarı olmayan ve ağ üzerinden yeniden adreslenmesi gereken cihazlar için ayrılmıştır.
- Adres 127: Ağdaki tüm cihazlarda Broadcast mesajları için rezerve edilmiştir; herhangi bir çalıştırılabilir cihaza atanmasına izin verilmez.

Dolayısıyla, çalıştırılabilir PROFIBUS cihazları için kullanılabilecek adresler 2 ila 125 aralığındadır.

Özellik penceresinde "PROFIBUS adresi" girişini seçin. STEP 7, cihazın PROFIBUS adresini atadığınız PROFIBUS adresi konfigürasyon iletişim kutusunu görüntüler.



Tablo 11-64 PROFIBUS adresi için parametre

Parametre		Açıklama
Alt ağ	Cihazın bağlı olduğu alt ağın adı. Yeni bir alt ağ oluşturmak için "Yeni alt ağ ekle" butonuna tıklayın. Ön ayar, "bağlı değil" şeklindedir. İki tür bağlantı mümkündür: • Ön ayar "bağlı değil", yerel bir bağlantı kurar. • Ağınızda iki veya daha fazla cihaz varsa, bir alt ağa gerek duyarsınız.	
Parametre	Adres	Cihaza atanan PROFIBUS adresi
	En yüksek adres	En yüksek PROFIBUS adresi, PROFIBUS'taki aktif istasyonlara bağlıdır (örn. DP-Master). Örneğin en yüksek PROFIBUS adresi için 15 ayarlanmış olsalar bile pasif DP-Slave'ler, 1'den 125'e kadar bağımsız PROFIBUS adreslerine sahiptir. En yüksek PROFIBUS adresi, token yönlendirme (gönderim haklarının iletilmesi) için önemlidir, token yalnızca aktif istasyonlara iletilir. En yüksek PROFIBUS adresi belirtilmesiyle veriyolu optimize edilir.
	Aktarım hızı	Yapılandırılan PROFIBUS ağındaki aktarım hızı: PROFIBUS aktarım hızları 9,6 kBit/sn ile 12 MBit/sn arasındadır. Aktarım hızı, kullanılan PROFIBUS katılımcılarının özelliklerine bağlıdır. Aktarım hızının en yavaş katılımcının desteklediği hızdan daha yüksek olmasına izin verilmez. Aktarım hızı, normalde PROFIBUS ağındaki Master için ayarlanır ve tüm DP-Slave'ler otomatik olarak aynı aktarım hızını kullanır (otomatik Baud).

11.6.4 Desantral çevrebirimler için yordamlar ve teşhis yordamları

Desantral çevrebirimler için yordamlar ve teşhis yordamlarına ilişkin bilgileri şurada bulabilirsiniz: Desantral G/Ç (PROFINET, PROFIBUS veya AS-i) (Sayfa 352).

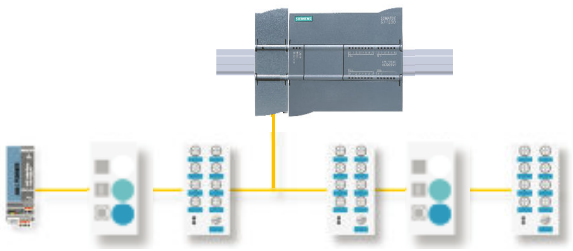
Teşhis olaylarının işlenmesine ilişkin bilgileri, "AUTOHOTSPOT" altında bulabilirsiniz.

11.7 ASi

S7-1200 AS-i Master CM 1243-2, AS-i ağının S7-1200 CPU'ya bağlanmasını sağlar.

Aktüatör/sensör arayüzü veya diğer adıyla AS-i, bir otomasyon sistemindeki en düşük seviyedeki bir Master için kullanılan bir ağ bağlantı sistemidir. CM 1243-2, ağda AS-i Master olarak hizmet verir. Sensörler ve aktüatörler (AS-i Slave'ler), basit bir AS-i kablosuyla CM 1243-2 üzerinden CPU'ya bağlanabilirler. CM 1243-2, AS-i ağ koordinasyonunun tamamını üstlenir ve CM 1243-2'ye atanan G/Ç adresleri üzerinden aktüatörlerden ve sensörlerden gelen verileri ve durum bilgilerini CPU'ya gönderir. Slave tipine göre ikili veya analog değerlere erişebilirsiniz. AS-i Slave'leri, AS-i sisteminin giriş ve çıkış kanallarıdır ve yalnızca CM 1243-2 tarafından çağrıldıklarında aktiftir.

Aşağıdaki şekilde S7-1200, AS-i G/Ç modüllerinin dijital/analog Slave cihazlarını kumanda eden bir AS-i Master'dır.



AS-i'yi, 4.0 sürümüne sahip S7-1200 CPU'ları ile kullanmak istiyorsanız, AS-i Master CM Firmware'i V1.1 sürümüne yükseltmelisiniz.
Bu güncellemeyi Web sunucusu veya SIMATIC Memory Card ile gerçekleştirebilirsiniz.

NOT

Sürüm 4.0'a sahip S7-1200 CPU'larda, Web sunucusu veya SIMATIC Memory Card üzerinden AS-i Firmware'i V1.0'dan V1.1'e yükseltmek için, AS-i Master CM 1243-2'deki AS-i Firmware'ini aşağıdaki şekilde güncellemeniz gerekir:

1. Firmware yükseltmesini AS-i Master CM 1243 2'ye yükleyin.
2. Yükleme işlemi tamamlandıktan sonra, AS-i Master CM 1243-2'deki Firmware yükseltmesini tamamlamak için S7-1200 CPU'yu kapatıp tekrar açın.
3. Her bir ilave AS-i Master CM 1243-2 için 1. ve 2. adımları tekrarlayın. S7-1200 PLC, maksimum üç AS-i Master CM 1243-2'ye izin verir.

NOT

AS-i CM Firmware'ini her zaman en güncel sürümünde tutmanızı (http://support.automation.siemens.com/WW/view/en/43416171) öneririz. Güncel Firmware'i, Siemens web sitesindeki "Servis ve Destek" bölümünde bulabilirsiniz.

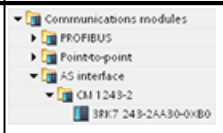
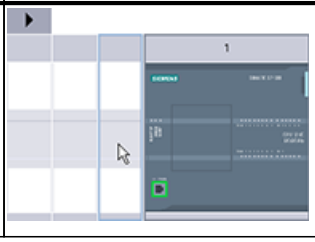
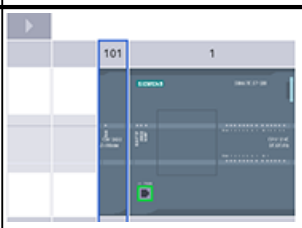
11.7.1 AS-i Master ve Slave cihazların yapılandırılması

AS-i Master CM 1243-2, S7-1200 otomasyon sistemine bir iletişim modülü olarak entegre edilmiştir.
AS-i Master CM 1243-2 hakkında ayrıntılı bilgiyi, "AS-i Master CM 1243-2 ve SIMATIC S7-1200 için AS-i veri ayırma ünitesi DCM 1271" (https://support.industry.siemens.com/cs/ww/en/ps/15750/man) işletme kitabında bulabilirsiniz.

11.7.1.1 AS-i Master modülü CM 1243-2 ve AS-i Slave'in eklenmesi

AS-i Master modülleri CM1243-2'yi, donanım kataloğunda CPU'ya ekleyebilirsiniz. Bu modüller, CPU'nun sol tarafına bağlanır. Maksimum üç adet CM1243-2 AS-i Master modülü kullanılabilir. Donanım konfigürasyonuna bir modül eklemek için, ilgili modülü donanım kataloğunda seçin ve çift tıklayın veya işaretlenen yuvanın üzerine sürükleyin.

Tablo 11-65 Cihaz konfigürasyonuna AS-i Master modülü CM1243-2'nin eklenmesi

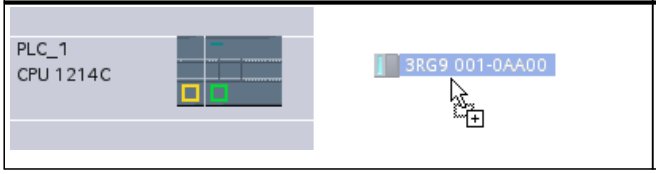
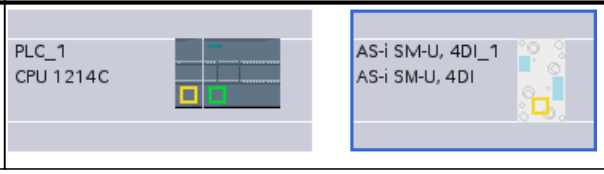
Modül	Modül seçme	Modül kullanma	Sonuç
CM 1243-2 AS-i Master			

Donanım kataloğunda AS-i Slave'leri de ekleyebilirsiniz. Örneğin "G/Ç modülü, kompakt, dijital, giriş" özelliklerine sahip bir Slave'i eklemek için Filtreyi (aktif ise) devre dışı bırakın ve aşağıdaki kutuları genişletin:

Saha cihazları > AS-Interface > IP6x giriş/çıkış modülleri, kompakt modüller > Dijital > Giriş > Kullanıcı modülleri > AS-i SM-U, 4DI

Bunun ardından, sipariş numaraları listesinden "3RG9 001-0AA00" öğesini seçin ve aşağıdaki şekilde gösterildiği gibi "G/Ç modülü, kompakt, dijital, giriş" özelliklerine sahip Slave'i takın.

Tablo 11-66 Cihaz konfigürasyonuna AS-i Slave eklenmesi

AS-i Slave takılması	Sonuç
	

11.7.1.2 İki AS-i cihazı arasındaki mantıksal ağ bağlantılarını yapılandırılması

AS-i Master CM1243-2'yi yapılandırdıktan sonra, şimdi ağ bağlantılarınızı yapılandırabilirsiniz.

"Cihazlar ve ağlar" portalında, "Ağ görünümü" altında projenizdeki cihazları birbirine bağlayabilirsiniz. AS-i bağlantısını oluşturmak için ilk cihazda sarı alanı (AS-i) etkinleştirin. Fareyi kullanarak, ikinci cihazdaki AS-i alanına bir hat çekin. Fare tuşunu bırakın. Böylece AS-i bağlantınız artık oluşturulmuştur.

Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Ağ bağlantısının oluşturulması" (Sayfa 562).

11.7.1.3 AS-i Master CM1243-2'nin özelliklerinin yapılandırılması

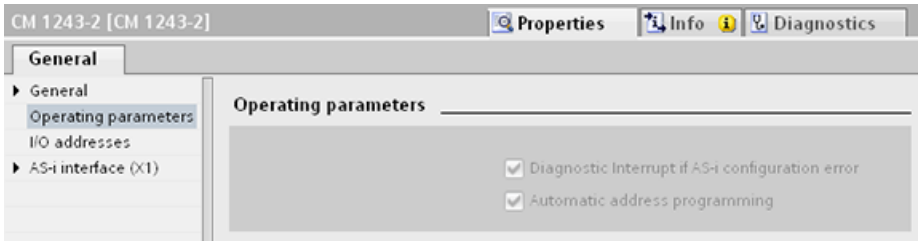
AS-i arayüzünün parametrelerini yapılandırmak için AS-i Master modülü CM1243-2'deki sarı AS-i alanına tıklayın. Bunun ardından, denetçi penceresindeki "Özellikler" sekmesi AS-i arayüzünü gösterir.
STEP 7 denetçi penceresinde genel bilgileri, adresleri ve işletim parametrelerini görüntüleyebilir, yapılandırabilir ve değiştirebilirsiniz:

Tablo 11-67 AS-i Master modülü CM1243-2'nin özellikleri

Özellik	Açıklama
Genel	AS-i Master CM 1243-2'nin adı
İşletim parametreleri	AS-i Master'ın yanıtı için parametre
G/Ç adresleri	Slave G/Ç adresleri için adres aralığı
AS-i arayüzü (X1)	Atanmış AS-i ağı

NOT

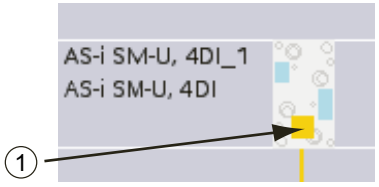
"AS-i konfigürasyonundaki hatalar için teşhis alarmı" ve "Otomatik adres programlaması" daima aktiftir ve dolayısıyla gri olarak gösterilir.



11.7.1.4 Bir AS-i Slave'e bir AS-i adresinin atanması

AS-i Slave arayüzü yapılandırması

AS-i arayüzünün parametrelerini yapılandırmak için AS-i Slave'deki sarı AS-i alanına tıklayın. Bunun ardından, denetçi penceresindeki "Özellikler" sekmesi AS-i arayüzünü gösterir.



① AS-i portu

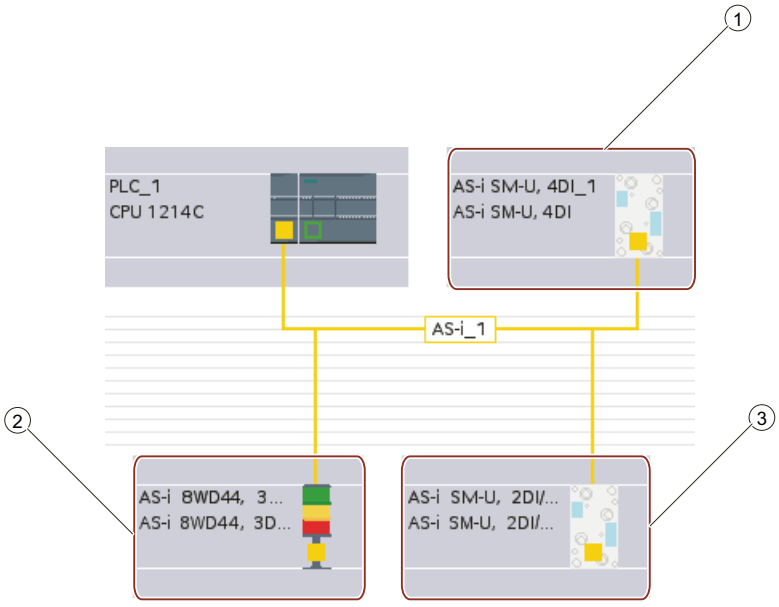
AS-i Slave adresi atanması

Bir AS-i ağında, her cihaza bir AS-i Slave adresi atanır. Bu adres, 0 ile 31 arasında değişebilir. Ancak, 0 adresi özellikle yeni Slave cihazlar için ayrılmıştır. Slave adresleri, toplamda maksimum 62 Slave cihaz için 1(A veya B) ila 31(A veya B) arasındadır.

"Standart" AS-i cihazları, tüm adresi kullanır ve A veya B tanımlayıcısı yoksa sayısal adrese sahiptir. "A/B düğümü" tipindeki AS-i cihazları ise adresin A veya B parçasını kullanır, bu sayede 31 adresin her biri iki kez kullanılabilir. Adres aralığı, 1A ila 31A artı 1B ila 31B'yi kapsar.

1 ila 31 aralığındaki her adres bir AS-i Slave'e atanabilir. Diğer bir deyişle: Slave'lerin adres 21 ile başlaması veya ilk Slave'in gerçekten adres 1'i alıp almaması önemli değildir.

Aşağıda sunulan örnekte, üç adet AS-i cihazına, "1" (standart cihaz), "2A" (A/B tipi düğüm cihazı) ve "3" (standart cihaz) adresleri atanmıştır:

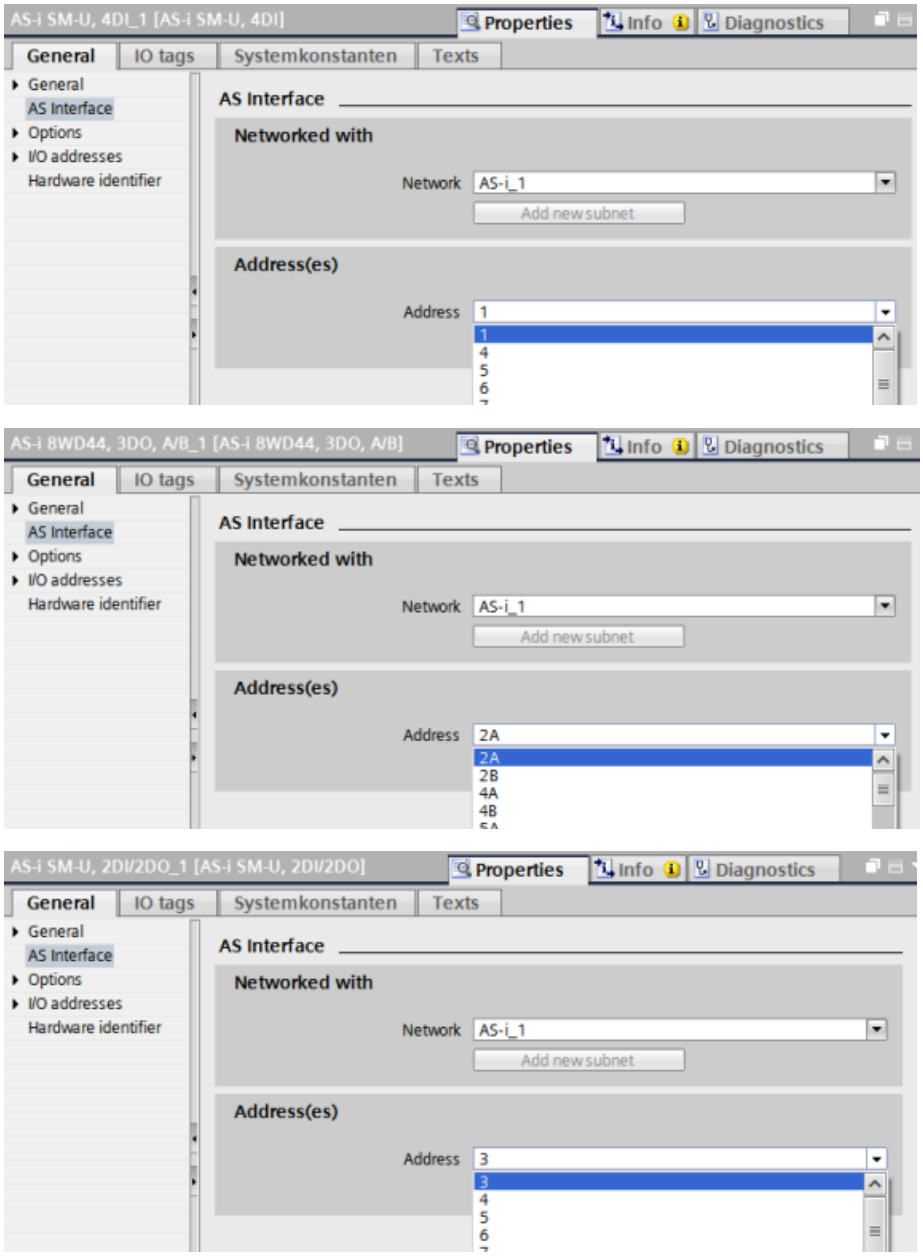


① AS-i Slave adresi 1; cihaz: AS-i SM-U, 4DI; sipariş numarası: 3RG9 001-0AA00

② AS-i Slave adresi 2A; cihaz: AS-i 8WD44, 3DO, A/B; sipariş numarası: 8WD4 428-0BD

③ AS-i Slave adresi 3; cihaz: AS-i SM-U, 2DI/2DO; sipariş numarası: 3RG9 001-0AC00

Buraya AS-i Slave adresini girin:



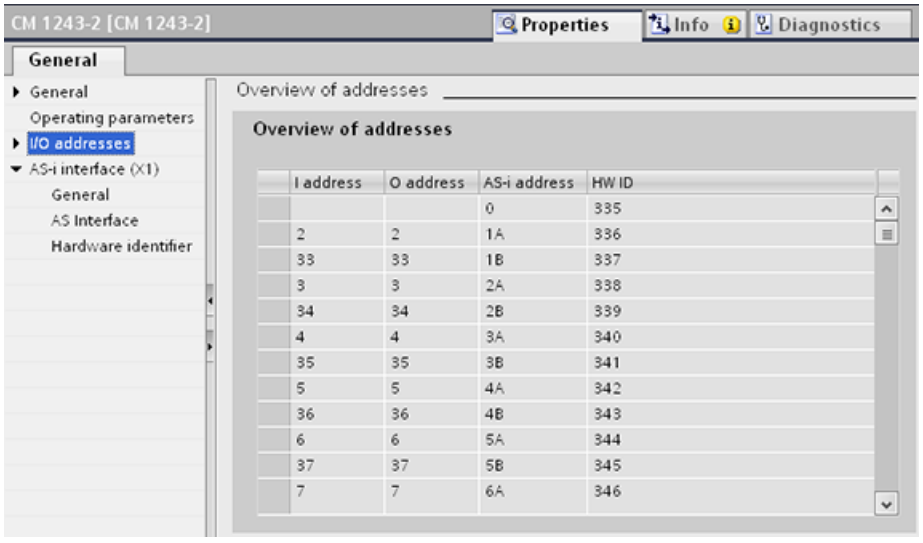
Tablo 11-68 AS-i arayüzü için parametreler

Parametre	Açıklama
Ağ	Cihazın bağlı olduğu ağın adı.
Adres(ler)	1(A veya B) ile 31(A veya B) aralığında toplam 62 adede kadar Slave cihaz için Slave cihaz için atanmış AS-i adresi

11.7.2 Kullanıcı programı ile AS-i Slave'ler arasında veri alışverişi

11.7.2.1 STEP 7 konfigürasyonu

AS-i Master, CPU'nun G/Ç alanında 62 Bayt'lık bir veri alanı ayırır. Buradan dijital verilere Bayt olarak erişilir. Her bir Slave için bir Bayt giriş verisi ve bir Bayt çıkış verisi vardır.
Dijital AS-i Slave'lerin AS-i bağlantılarının, atanmış Bayt'ın veri Bit'lerine atanması, AS-i Master modül CM 1243-2'nin denetçi penceresinde görüntülenir.



Kullanıcı programındaki AS-i Slave'lerin verilerine, görüntülenen G/Ç adresleri üzerinden uygun Bit bağlantı işlemleri (örn. "AND") veya Bit atamaları yardımıyla erişebilirsiniz.

NOT

AS-i Slave'leri STEP 7 ile yapılandırmadığınızda, "Sistem ataması" otomatik olarak etkinleştirilir.

Hiçbir Slave yapılandırmazsanız, fiili veriyolu konfigürasyonu ile ilgili olarak AS-i Master CM1243-2'yi bilgilendirmelisiniz. Bunun için "GERÇEK > HEDEF" çevrimiçi fonksiyonunu kullanın.

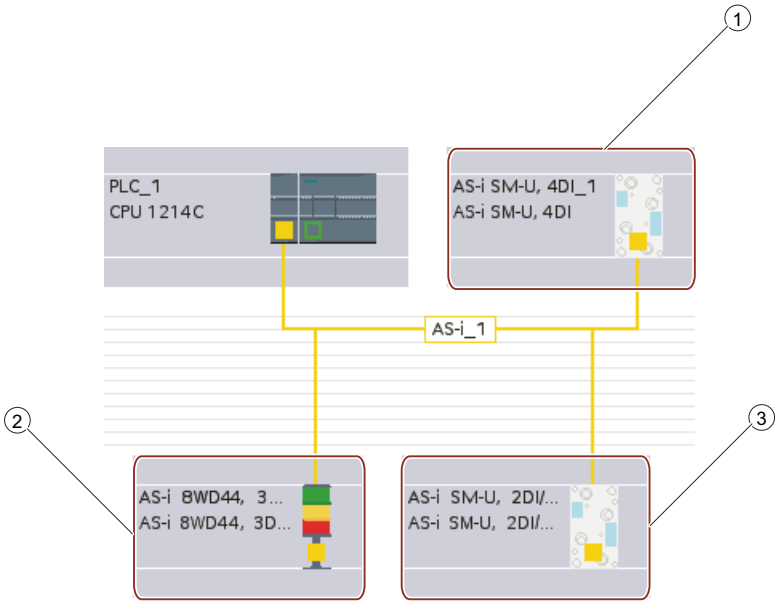
Daha fazla bilgi

AS-i Master CM 1243-2 hakkında ayrıntılı bilgiyi, "AS-i Master CM 1243-2 ve SIMATIC S7-1200 için AS-i veri ayırma ünitesi DCM 1271" (https://support.industry.siemens.com/cs/ww/en/ps/15750/man) işletme kitabında bulabilirsiniz.

11.7.2.2 AS-i Slave'lerin STEP 7 ile yapılandırılması

Dijital AS-i değerlerinin aktarılması

Periyodik işletimde CPU, AS-i Master CM1243-2 üzerinden AS-i Slave'lerin dijital girişlerine ve çıkışlarına erişir. Verilere, G/Ç adresleri veya veri seti aktarımı yoluyla erişilir.



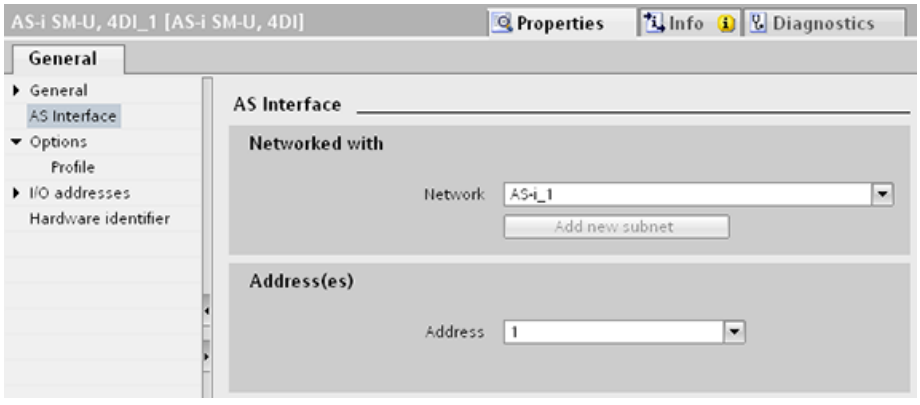
① AS-i Slave adresi 1

② AS-i Slave adresi 2A

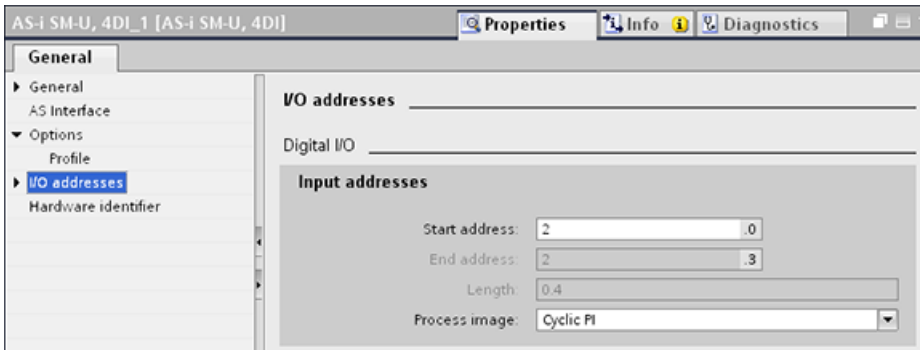
③ AS-i Slave adresi 3

Buradan dijital verilere Bayt olarak erişilir (başka bir deyişle: Her bir dijital AS-i Slave'e birer Bayt atanır). AS-i Slave'lerini STEP 7'de yapılandırdığınızda, ilgili AS-i Slave için kullanıcı programından verilere erişim için olan G/Ç adresi denetçi penceresinde görüntülenir.

Yukarıda AS-i ağındaki dijital giriş modülüne (AS-i SM-U, 4DI), Slave adresi 1 atanmıştır. Dijital giriş modülüne tıkladığınızda, aşağıda gösterildiği üzere cihaz özelliklerinde "AS arayüzü" sekmesinde Slave adresi görüntülenir:

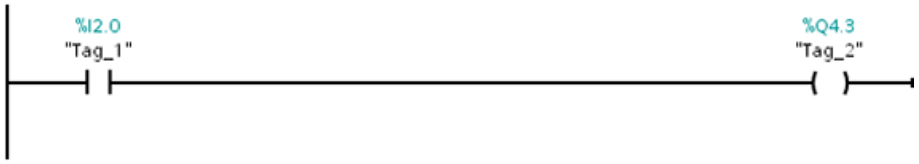


Yukarıda AS-i ağındaki dijital giriş modülüne (AS-i SM-U, 4DI), G/Ç adresi 2 atanmıştır. Dijital giriş modülüne tıkladığınızda, cihaz özelliklerindeki "G/Ç adresleri" sekmesinde G/Ç adresi aşağıda gösterildiği gibi görüntülenir:



Kullanıcı programındaki AS-i Slave'lerin verilerine, onların G/Ç adresleri üzerinden uygun Bit bağlantı işlemleri (örn. "AND") veya Bit atamaları yardımıyla erişebilirsiniz. Aşağıda sunulan basit program, atamanın çalışma şeklini gösterir:

Bu programda giriş 2.0 sorgulanır. AS-i sisteminde, bu giriş Slave 1'e aittir (giriş Bayt'ı 2, Bit 0). Daha sonra ayarlanan çıkış 4.3, AS-i Slave 3'e karşılık gelir (çıkış Bayt'ı 4, Bit 3).



Analog AS-i değerlerinin aktarılması

STEP 7'de bir AS-i Slave'i analog Slave olarak yapılandırdıysanız, bu AS-i Slave'in analog verilerine CPU'nun proses görünümü üzerinden erişebilirsiniz.

Analog Slave'i STEP 7'de yapılandırmadıysanız, AS-i Slave'in verilerine yalnızca aperiyodik fonksiyonlar (veri seti arayüzü) üzerinden erişebilirsiniz. CPU'nun kullanıcı programında, AS-i çağrıları, desantral çevrebirim, yani RDREC (veri seti okuma) ve WRREC (veri seti yazma) yordamları aracılığıyla okunur ve yazılır.

NOT

STEP 7 üzerinden belirlenen ve S7 istasyonuna yüklenen AS-i Slave'lerin bir konfigürasyonu, S7 istasyonu yol aldığında CPU'dan AS-i Master CM1243-2'ye yüklenir. "Sistem ataması" çevrimiçi fonksiyonu (Sayfa 746) ("GERÇEK -> HEDEF") aracılığıyla belirlenen tüm mevcut konfigürasyonların üzerine yazılacaktır.

Daha fazla bilgi

AS-i Master CM 1243-2 hakkında ayrıntılı bilgiyi, "AS-i Master CM 1243-2 ve SIMATIC S7-1200 için AS-i veri ayırma ünitesi DCM 1271" (http://support.automation.siemens.com/WW/view/en/50414115/133300) işletme kitabında bulabilirsiniz.

11.7.3 Desantral çevrebirimler için yordamlar

Desantral çevrebirimler için yordamlar ve teşhis yordamlarına ilişkin bilgileri şurada bulabilirsiniz: Desantral G/Ç (PROFINET, PROFIBUS veya AS-i) (Sayfa 352).
Teşhis olaylarının işlenmesine ilişkin bilgileri, "AUTOHOTSPOT" altında bulabilirsiniz.

11.7.4 AS-i çevrimiçi araçlarıyla çalışma

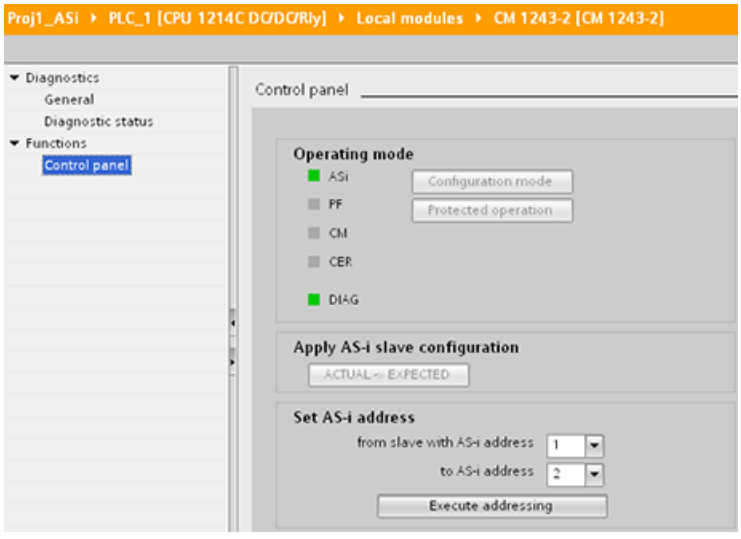
AS-i işletim türlerini görüntülemek ve değiştirmek için, STEP 7'de çevrimiçi olmalısınız.
AS-i'nin işletim türlerini değiştirmek için aşağıdaki yöntemi izleyin:

1. PLC'nizin cihaz konfigürasyonunda AS-i Master modülü CM1243-2'yi seçin.
2. Fonksiyon çubuğundaki "Çevrimiçi bağlan" butonuna tıklayın.
3. "Çevrimiçi" menüsünde veya proje gezgininde "Çevrimiçi ve teşhis" komutunu seçin.

"İşletim durumu" altında Kumanda panelinde iki işletim türü görüntülenir:

- Projelendirme modu:
  - AS-i Slave cihazınızda ve CPU G/Ç adreslerinde gerekli değişiklikleri yapabilirsiniz.
  - Yeşil "CM" LED'i yanar.
- Korumalı işletim:
  - AS-i Slave cihazı ve CPU G/Ç adresleri değiştirilemez.
  - Yeşil "CM" LED'i devre dışıdır.

Projelendirme modu aktif ise AS-i Slave adresini ayarlayabilirsiniz. Herhangi bir adres atanmamış yeni bir Slave, her zaman 0 adresine sahiptir. Master, adres ataması olmadığında onu yeni bir Slave olarak tanır. Kendisine bir adres atanana dek, normal iletişime dahil edilmez.



Konfigürasyon hatası

Sarı "CER" LED'i yanıyorsa, AS-i Slave'in cihaz konfigürasyonunda bir hata vardır. "GERÇEK > HEDEF" butonuyla, AS-i Master modülü CM1243-2'nin Slave cihazı konfigürasyonunun üzerine AS-i saha ağ Slave'inin konfigürasyonunu yazabilirsiniz.

11.8 S7 iletişimi

11.8.1 GET ve PUT (uzak CPU'dan okuma ve yazma)

GET ve PUT yordamlarını kullanarak, PROFINET ve PROFIBUS bağlantıları üzerinden S7 CPU'larla iletişim kurabilirsiniz. Bu iletişim yalnızca, yerel CPU özelliklerinin "Koruma ve dijital güvenlik" özelliğinde partner CPU için "Uzak partnerden PUT/GET iletişimi yoluyla erişime izin ver" fonksiyonu etkinleştirildiğinde mümkündür:

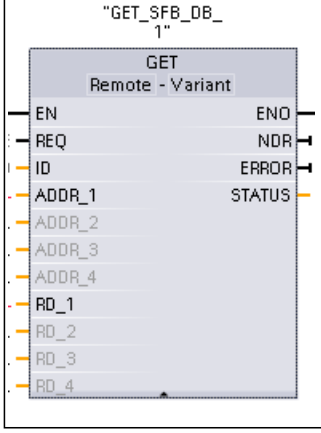
- Bir desantral CPU'daki verilere erişim: Bir S7-1200 CPU, desantral CPU'ların (S7-200/300/400/1200) değişkenlerini adreslemek için yalnızca ADDR_x giriş alanındaki mutlak adreslerden yararlanabilir.
- Standart bir DB'deki verilere erişim: Bir S7-1200 CPU, bir desantral S7 CPU'nun bir standart DB'sindeki DB değişkenlerini adreslemek için, yalnızca ADDR_x giriş alanındaki mutlak adreslerden yararlanabilir.
- Optimize edilmiş bir DB'de verilere erişim: Bir S7-1200 CPU, bir desantral S7-1200 CPU'nun optimize edilmiş bir DB'sindeki DB değişkenlerine erişemez.
- Bir yerel CPU'daki verilere erişim: Bir S7-1200 CPU, GET veya PUT yordamının RD_x veya SD_x giriş alanında giriş olarak mutlak adresleri veya sembolik adresleri kullanabilir.

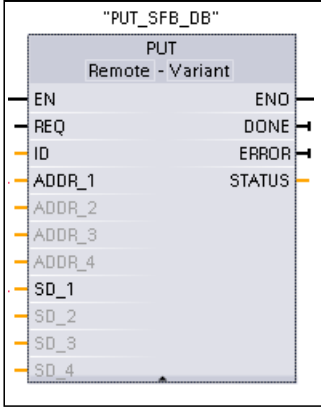
NOT

GET/PUT fonksiyonu, V4.x ile otomatik olarak etkinleştirilmez

GET/PUT erişimini (Sayfa 156) etkinleştirmek için, CPU'nun cihaz konfigürasyonunda, denetçi penceresindeki "Özellikler" sekmesine gidin ve "Koruma ve dijital güvenlik" özelliğini etkinleştirin.

Tablo 11-69 GET ve PUT yordamları

LAD/FBD	SCL	Açıklama
	**"GET_DB"(req:=_bool_in_, ID:=_word_in_, ndr=>_bool_out_, error=>_bool_out_, status=>_word_out_, addr_1:=_remote_inout_, [...addr_4:=_remote_inout_,] rd_1:=_variant_inout_ [,...rd_4:=_variant_inout_]);**	GET yordamıyla bir uzak S7 CPU'dan verileri okursunuz. Uzak CPU, RUN veya STOP işletim durumunda olabilir. Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

LAD/FBD	SCL	Açıklama
	**"PUT_DB"(req:=_bool_in_, ID:=_word_in_, done=>_bool_out_, error=>_bool_out_, status=>_word_out_, addr_1:=_remote_inout_, [...addr_4:=_remote_inout_,] sd_1:=_variant_inout_, [....sd_4:=_variant_inout_]);**	PUT yordamıyla bir uzak S7 CPU'ya veriler yazarsınız. Uzak CPU, RUN veya STOP işletim durumunda olabilir. Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 11-70 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	Input	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır.
ID	Input	CONN_PRG (Word)	S7 bağlantı ID'si (onaltılı)
NDR (GET)	Output	Bool	Yeni veriler hazır: • 0: Yordam, henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev, başarıyla tamamlandı.
DONE (PUT)	Output	Bool	DONE: • 0: Yordam, henüz başlatılmadı veya hâlâ çalışıyor. • 1: Görev, başarıyla tamamlandı.
ERROR STATUS	Output Output	Bool Word	• ERROR=0 STATUS değeri: – 0000H: Ne uyarı var ne de hata. – <> 0000H: Uyarı, STATUS ayrıntılı bilgiler sağlar. • ERROR=1 Hata. STATUS, hatanın doğasına ilişkin ayrıntılı bilgiler getirir.
ADDR_1	InOut	Kaldırıldı	Okunacak (GET) veya gönderilecek (PUT) verileri saklayan yerel CPU'daki bellek alanlarını gösteren işaretçi.
ADDR_2	InOut	Kaldırıldı	
ADDR_3	InOut	Kaldırıldı	
ADDR_4	InOut	Kaldırıldı	
RD_1 (GET) SD_1 (PUT)	InOut	Varyant	Okunacak (GET) veya gönderilecek (PUT) verileri saklayan yerel CPU'daki bellek alanlarına işaretçi. İzin verilen veri tipleri: Bool (yalnızca bir Bit'e izin verilir), Byte, Char, Word, Int, DWord, DInt veya Real. Not: Eğer işaretçi bir DB'ye erişiyorsa, mutlak adresi belirtmeniz gerekir, örneğin: P# DB10.DBX5.0 Bayt 10 Bu durumda, 10 değeri, GET veya PUT yordamı için Bayt sayısını gösterir.
RD_2 (GET) SD_2 (PUT)	InOut	Varyant	
RD_3 (GET) SD_3 (PUT)	InOut	Varyant	
RD_4 (GET) SD_4 (PUT)	InOut	Varyant	

ADDR_x (uzak CPU) ve RD_x veya SD_x (yerel CPU) parametrelerinin uzunluğunun (Bayt sayısı) ve veri tiplerinin eşleştiğinden emin olmalısınız. "Byte" kodundan sonraki sayı, ADDR_x, RD_x veya SD_x parametresi tarafından belirtilen Bayt sayısıdır.

NOT

Bir GET yordamı tarafından alınan toplam Bayt sayısı veya bir PUT yordamı tarafından gönderilen toplam Bayt sayısı sınırlandırılmıştır. Bu sınırlama, dört olası adres ve bellek alanından kaç tanesini kullandığınızı gösterir:

- Eğer ADDR_1 ve RD_1/SD_1 kullanırsanız, bir GET yordamı 222 Bayt çağırabilir ve bir PUT yordamı 212 Bayt gönderebilir.
- Eğer ADDR_1, RD_1/SD_1, ADDR_2 ve RD_2/SD_2 kullanırsanız, bir GET yordamı toplam 218 Bayt çağırabilir ve bir PUT yordamı toplam 196 Bayt gönderebilir.
- Eğer ADDR_1, RD_1/SD_1, ADDR_2, RD_2/SD_2, ADDR_3 ve RD_3/SD_3 kullanırsanız, bir GET yordamı toplam 214 Bayt çağırabilir ve bir PUT yordamı toplam 180 Bayt gönderebilir.
- Eğer ADDR_1, RD_1/SD_1, ADDR_2, RD_2/SD_2, ADDR_3, RD_3/SD_3, ADDR_4, RD_4/SD_4 kullanırsanız, bir GET yordamı toplam 210 Bayt çağırabilir ve bir PUT yordamı toplam 164 Bayt gönderebilir.

Adres ve bellek alanı parametrelerinizin her birinin Bayt sayılarının toplamı, belirtilen limitlerinden az veya bu limitlere eşit olmalıdır. Bu limitleri aşarsanız, GET veya PUT yordamı bir hata yayınlar.

REQ parametresindeki yükselen kenarda, okuma yordamı (GET) veya yazma yordamı (PUT); ID, ADDR_1 ve RD_1 (GET) veya SD_1 (PUT) parametrelerini yükler.

- GET'te: Uzak CPU, talep edilen verileri alma alanlarına (RD_x) yayınlar. Bir sonraki çevrimde bu işlemle başlar. Okuma yordamı hatasız tamamlandıysa, NDR parametresi 1 olarak ayarlanır. Bir önceki yordam bitene kadar yeni bir yordam başlatılamaz.
- PUT'ta: Yerel CPU, verileri (SD_x) uzak CPU'nun belleğindeki adrese (ADDR_x) göndermeye başlar. Yazma yordamı hatasız tamamlandığında, uzak CPU bir yürütme onayı verir. PUT yordamının DONE parametresi daha sonra 1 olarak ayarlanır. Önceki yordam bitene kadar yeni bir yazma yordamı başlatılamaz.

NOT

Verilere erişmeden veya başka bir okuma veya yazma yordamını yürütmeden önce, veri tutarlılığını güvence altına almak adına, her zaman yordamın tamamlanıp tamamlanmadığını kontrol edin (GET ile NDR = 1 veya PUT ile DONE = 1).

ERROR ve STATUS parametreleri, okuma (GET) veya yazma yordamının (PUT) durumunu getirir.

Tablo 11-71 Hata bilgisi:

ERROR	STATUS (desimal)	Açıklama
0	11	• Önceki görev henüz sona ermediğinden, yeni görev etkin olmaz. • Görev, artık daha düşük öncelikli bir öncelik sınıfında işlenmeye geçer.
0	25	İletişim başlar. Görev düzenlenir.

ERROR	STATUS (desimal)	Açıklama
1	1	İletişim sorunları, örneğin: • Bağlantı açıklaması yüklenmedi (yerel veya uzak) • Bağlantı kesintiye uğrar (örnek: Kablo, CPU devre dışı veya CM/CB/CP STOP durumunda) • Partner ile bağlantı henüz kurulmadı
1	2	Partner cihazdan negatif onay. Görev yürütülmez.
1	4	Veri uzunluğu veya veri tipiyle ilgili gönderim alanı işaretçilerinde (GET ile RD_x, PUT ile SD_x) hata
1	8	Partner CPU'da erişim hatası
1	10	Yerel kullanıcı belleğine erişilemiyor (örnek: Silinmiş DB'ye erişim denemesi)
1	12	SFB çağrıldığında: • GET veya PUT'a ait olmayan bir örnek DB belirtildi mi? • Örnek DB belirtilmedi, ancak ortaklaşa kullanılan bir DB söz konusu • Örnek DB bulunamadı (yeni bir örnek DB yükleniyor).
1	20	• Maksimum paralel görev/örnek sayısı aşıldı • Örnekler, CPU-RUN işletimi sırasında aşırı yüklendi Bu durum, GET veya PUT yordamının ilk kez yürütülmesi sırasında mümkündür.
1	27	CPU'da işleme karşılık gelen bir GET veya PUT yordamı yok.

11.8.2 S7 bağlantısının oluşturulması

Bağlantı mekanizmaları

PUT/GET yordamlarını kullanarak uzak katılımcılarla iletişim kurmak için kullanıcının ilgili yetkilere sahip olması gerekir.

Standart olarak, "Uzak partnerden PUT/GET iletişimi yoluyla erişime izin ver" opsiyonu aktif değildir. Bu durumda CPU verilerine okuma ve yazma erişimi, yalnızca yerel CPU ve uzak katılımcı için yapılandırılması veya programlanması gereken iletişim bağlantılarıyla mümkündür. Örneğin BSEND/BRCV yordamları kullanılarak erişim mümkündür.

Yerel CPU yalnızca sunucu olarak kullanılıyorsa, CPU iletişim partneri ile iletişimi programlayamaz veya yapılandıramaz. Dolayısıyla CPU işletimindeyken aşağıdaki bağlantılar mümkün değildir:

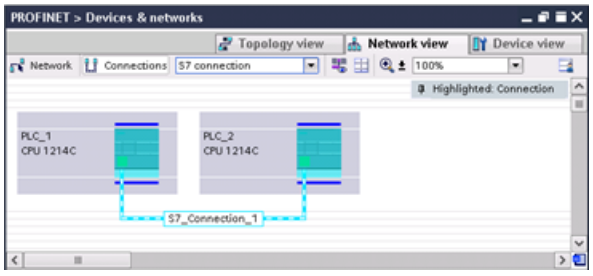
- İletişim modülleri üzerinden PUT/GET, FETCH/WRITE veya FTP erişimi
- Başka S7 CPU'larının PUT/GET erişimi
- PUT/GET iletişimi üzerinden HMI erişimi

CPU verilerine istemci tarafından da erişim mümkün olmalıysa, yani CPU'nun iletişim hizmetleri kısıtlanmayacaksa, bu güvenlik kademesi için S7-1200 CPU için erişim korumasını (Sayfa 156) yapılandırabilirsiniz.

Bağlantı türleri

Seçilmiş bağlantı türü, bir partner istasyonla iletişim bağlantısı kurar. Bağlantı oluşturulur, kurulur ve otomatik olarak denetlenir.

"Cihazlar ve ağlar" portalında, "Ağ görünümü" altında projenizdeki cihazları birbirine bağlayabilirsiniz (Sayfa 562). İlk önce "Bağlantılar" sekmesine tıklayın, ardından sağdaki açılır listeden bağlantı türünü (örn. S7 bağlantısı) seçin. İlk cihazdaki yeşil alana (PROFINET) tıklayın ve ikinci cihazdaki PROFINET alanına bir hat çekin. Fare tuşunu bırakın. PROFINET bağlantınız artık oluşturulmuştur.



Şu butona tıklayarak "Vurgulanmış: Bağlantı", iletişim yordamının "Özellikler" konfigürasyon iletişim kutusunu çağırın.

11.8.3 Yerel ile partner CPU arasındaki bağlantı yolunun yapılandırılması

Genel parametrelerin yapılandırılması

İletişim parametrelerini, iletişim yordamının "Özellikler" iletişim kutusunda belirleyebilirsiniz. Bu iletişim kutusu, yordamın bir bölümünü seçtiğinizde sayfanın altında görüntülenir.

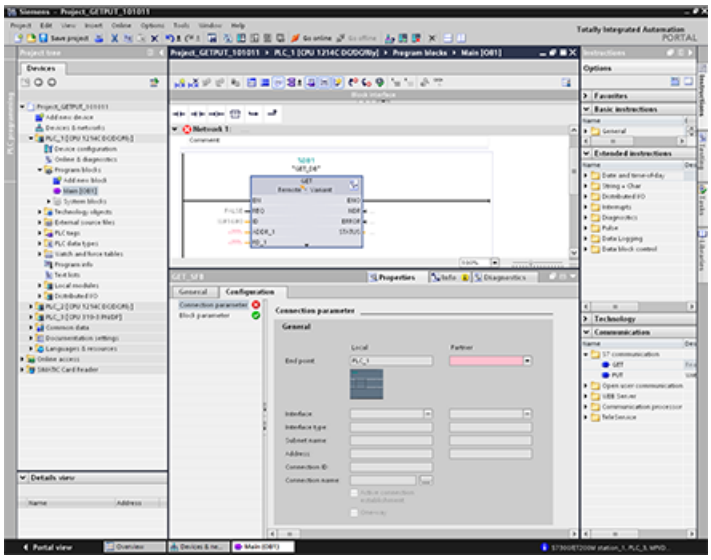
Daha fazla bilgi için bkz. "Cihaz konfigürasyonu: Yerel CPU ile partner CPU arasındaki bağlantı yolunun yapılandırılması (Sayfa 563)".

Kullanılacak TSAP'leri veya portlarını, "Adres ayrıntıları" bölümündeki "Bağlantı parametreleri" iletişim kutusunda tanımlarsınız. "Yerel TSAP" alanına ise CPU'daki bir bağlantının TSAP veya portu girilir. "Partner TSAP" alanına ise partner CPU'daki bir bağlantının TSAP veya portu girilir.

11.8.4 GET/PUT için bağlantı parametrelerinin atanması

GET/PUT yordamları için bağlantı parametrelerinin atanması, CPU'dan CPU'ya S7 iletişimi için olan S7 bağlantılarının konfigürasyonunda kullanıcıya yardımcı olur.

Bir GET veya PUT bloğunu ekledikten sonra, STEP 7, GET/PUT yordamlarının bağlantı parametreleri için atama iletişim kutusunu görüntüler:



Denetçi penceresinde, yordamın bir bölümünü seçtiğinizde bağlantının özellikleri görüntülenir. İletişim parametrelerini, iletişim yordamının "Özellikler" iletişim kutusundaki "Konfigürasyon" sekmesi üzerinden yapılandırabilirsiniz.

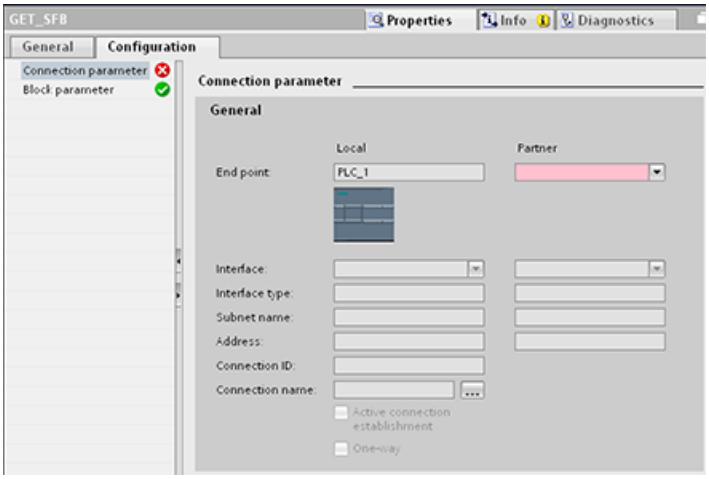
NOT

GET/PUT fonksiyonu, V4.x ile otomatik olarak etkinleştirilmez

GET/PUT erişimini (Sayfa 156) etkinleştirmek için, CPU'nun cihaz konfigürasyonunda, denetçi penceresindeki "Özellikler" sekmesine gidin ve "Koruma ve dijital güvenlik" özelliğini etkinleştirin.

11.8.4.1 Bağlantı parametreleri

"Bağlantı parametreleri" sayfasında, gerekli S7 bağlantısını ve GET/PUT bloğunun "ID" parametresi tarafından referans verilen "Bağlantı ID'si" parametresini yapılandırabilirsiniz. Bu sayfa, yerel uç noktaya ilişkin bilgileri içerir ve burada yerel arayüzü de belirleyebilirsiniz. Ayrıca partner uç noktasını da tanımlayabilirsiniz.
"Blok parametreleri" sayfasında diğer blok parametrelerini yapılandırabilirsiniz.



Tablo 11-72 Bağlantı parametreleri: Genel tanımlar

Parametre		Tanım
Bağlantı parametreleri: Genel	Uç nokta	"Yerel uç nokta": Yerel CPU "Partner uç noktası": Partner CPU'nun adı (desantral) Not: "Partner uç noktası" açılır listesinde sistem, güncel projenin tüm olası S7 bağlantı partnerlerini ve "Belirtilmemiş" opsiyonunu görüntüler. Belirtilmemiş bir partner, şu an için STEP 7 projesinde bulunmayan bir iletişim partneridir (örneğin bir harici üreticinin iletişim partneridir).
	Arayüz	Arayüzlerin adları Not: Yerel ve partner arayüzleri değiştirerek bağlantıyı değiştirebilirsiniz.
	Arayüz tipi	Arayüzün türü
	Alt ağ adı	Alt ağların adı
	Adres	Atanmış IP adresleri Not: "Belirtilmemiş" bir iletişim partneri için bir üçüncü şahıs cihazının uzak adresini girebilirsiniz.
	Bağlantı ID'si	ID numarası: GET/PUT bağlantısının parametre atamasıyla otomatik olarak oluşturulur.
	Bağlantı adı	Yerel CPU ve partner CPU için veri belleği adresi: GET/PUT bağlantısının parametre atamasıyla otomatik olarak oluşturulur.
	Aktif bağlantı kurma	Aktif bağlantı olarak yerel CPU'yu seçmek için opsiyon kutusu
	Tek yönlü	Tek yönlü veya çift yönlü bağlantıyı belirtmek için opsiyon kutusu; yazma korumalı Not: PROFINET ile bir GET/PUT bağlantısında, hem yerel cihaz hem de partner cihaz, sunucu veya istemci görevi görebilir. Bu, çift yönlü bir bağlantıya olanak sağlar, "Tek yönlü" opsiyon kutusu etkinleştirilmez. PROFIBUS ile GET/PUT bağlantısında partner cihaz bazı durumlarda yalnızca sunucu görevi görür (örn. S7-300 ile) ve "Tek yönlü" opsiyon kutusu aktiftir.

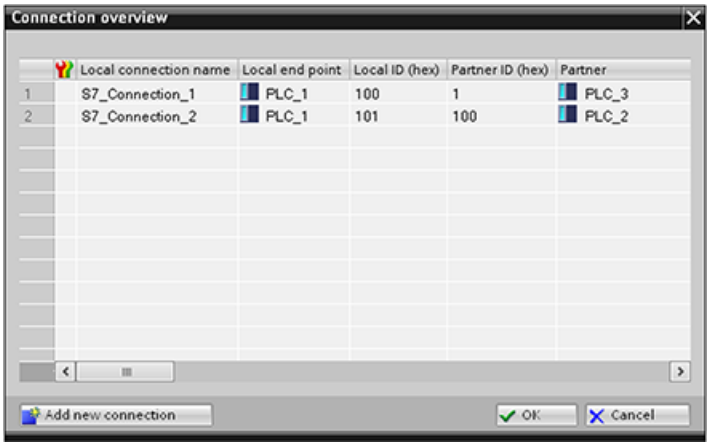
Sistem tanımlı bağlantı ID'lerini değiştirmenin üç yolu vardır:

1. Geçerli ID'yi doğrudan GET/PUT bloğunda değiştirebilirsiniz. Yeni ID zaten var olan bir bağlantıya aitse, bağlantı değiştirilir.
2. Geçerli ID'yi doğrudan GET/PUT bloğunda değiştirebilirsiniz, ancak yeni ID henüz mevcut değildir. Sistem yeni bir S7 bağlantısı oluşturur.
3. Güncel ID'yi "Bağlantıya genel bakış" iletişim kutusundan değiştirebilirsiniz: Veri girişiniz, ilgili GET/PUT bloğunun ID parametresi ile senkronize edilir.

NOT

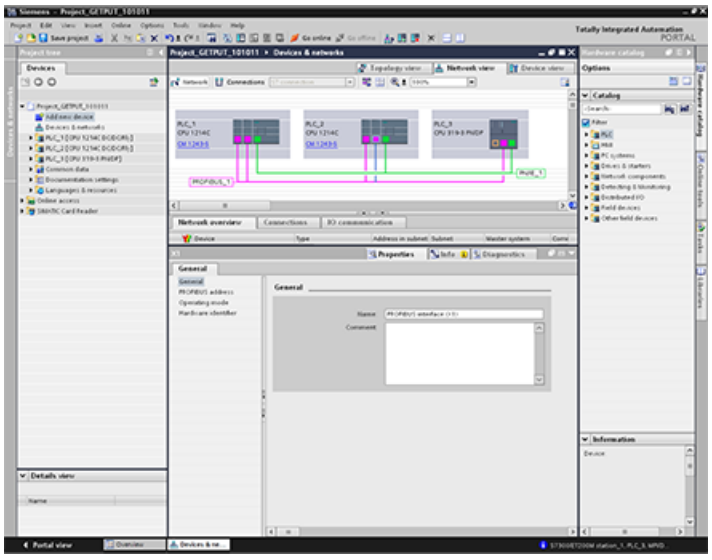
GET/PUT bloğunun "ID" parametresi, bir bağlantı adı olmayıp aşağıdaki örnekteki gibi yazılmış sayısal bir yordamdır: W#16#1

Bağlantı adı, özel bir kumanda elemanı olan "Bağlantıya genel bakış" iletişim kutusu kullanılarak düzenlenebilir. Bu iletişim kutusu, güncel GET/PUT iletişimi için alternatif olarak seçilebilecek tüm mevcut S7 bağlantılarını gösterir. Bu tabloda tamamen yeni bir bağlantı oluşturabilirsiniz. "Bağlantıya genel bakış" kutusunu açmak için "Bağlantı adı" alanının sağındaki butona tıklayın.



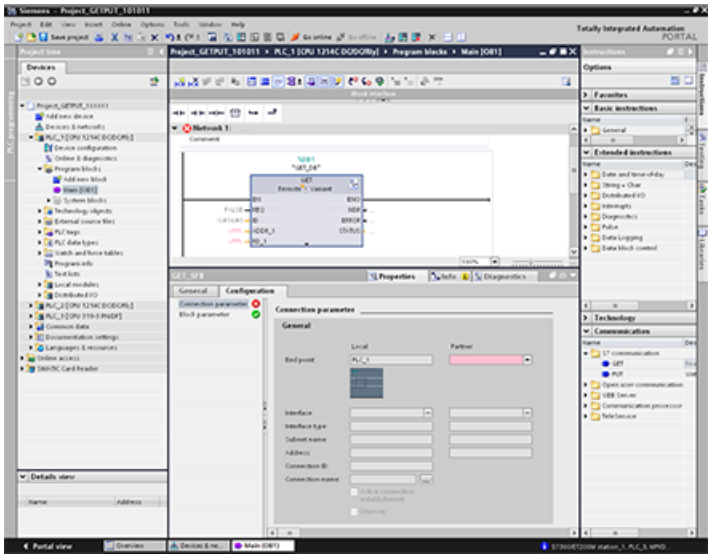
11.8.4.2 CPU'dan CPU'ya S7 bağlantısını yapılandırılması

PLC_1, PLC_2 ve PLC_3'ün konfigürasyonu aşağıdaki şekilde gösterildiği gibi yapılmışsa, "PLC_1" için GET veya PUT blokları eklemelisiniz.

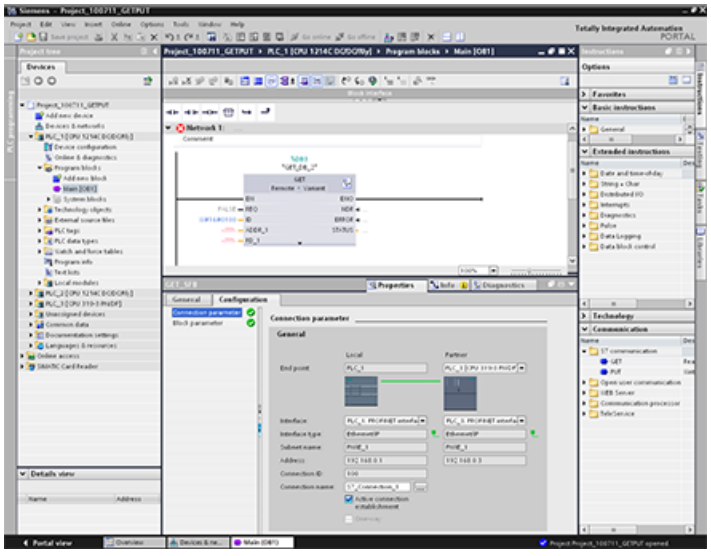


GET veya PUT yordamı için, "Özellikler" sekmesi, aşağıdaki menü opsiyonlarıyla denetçi penceresinde otomatik olarak görüntülenir:

- Bağlantı parametreleri
- Blok parametreleri



"Partner uç noktası" için "PLC_3" opsiyonunu seçin.



Sistem, aşağıdaki değişikliklerle yanıt verir:

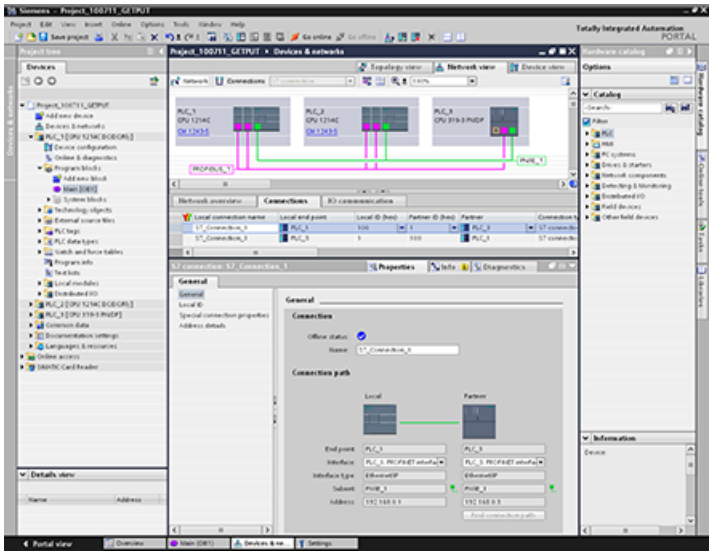
Tablo 11-73 Bağlantı parametreleri: Genel değerler

Parametre		Tanım
Bağlantı parametreleri: Genel	Uç nokta	"Yerel uç nokta", "PLC_1" (yazma korumalı) içerir. "Partner uç noktası" alanı, "PLC_3[CPU319-3PN/DP]" öğesini içerir: • Renk, kırmızıdan beyaza dönüşür. • "Partner" cihaz görseli görüntülenir. • PLC_1 ve PLC_3 cihaz görselleri arasında bir bağlantı çizgisi görüntülenir (yeşil Ethernet hattı).
	Arayüz	"Yerel arayüz", "CPU1214C DC/DC/DC, PROFINET arayüzü (R0/S1)" içerir. "Partner arayüzü" şunları içerir: "CPU319-3PN/DP, PROFINET arayüzü (R0/S2)".
	Arayüz tipi	"Yerel arayüz tipi", "Ethernet/IP"yi içerir; kullanım, yazma korumalıdır. "Partner arayüz tipi", "Ethernet/IP"yi içerir; kullanım, yazma korumalıdır. Arayüz tipi görselleri, yerel ve partner arayüz tiplerinin (yeşil Ethernet sembolü) sağında görüntülenir.
	Alt ağ adı	"Yerel alt ağ adı", "PN/IE_1"i içerir; kullanım, yazma korumalıdır. "Partner alt ağ adı", "PN/IE_1"i içerir; kullanım, yazma korumalıdır.
	Adres	"Yerel adres", yerel IP adresini içerir; kullanım, yazma korumalıdır. "Partner adresi", partner IP adresini içerir; kullanım, yazma korumalıdır.
	Bağlantı ID'si	"Bağlantı ID'si", "100" içerir. Programlama editörünün [OB1] ana programında, GET/PUT bloğunun "bağlantı ID'si" de "100" değerini alır.
	Bağlantı adı	"Bağlantı adı", standart bağlantı adını alır (örnek: "S7_connection_1"); kullanım etkinleştirilmiştir.
	Aktif bağlantı kurma	Yerel CPU'yu, aktif bağlantı olarak seçmek için etkinleştirilmiştir.
	Tek yönlü	Yazma korumalı ve etkinleştirilmemiş. Not: "PLC_1" (bir S7-1200 CPU 1214C DC/DC/röle) ve "PLC_3" (bir S7-300 CPU 319-3PN/DP), PROFINET GET/PUT bağlantısında hem sunucu hem de istemci olarak çalışabilir, yani çift yönlü bir bağlantının mümkün olmasıdır.

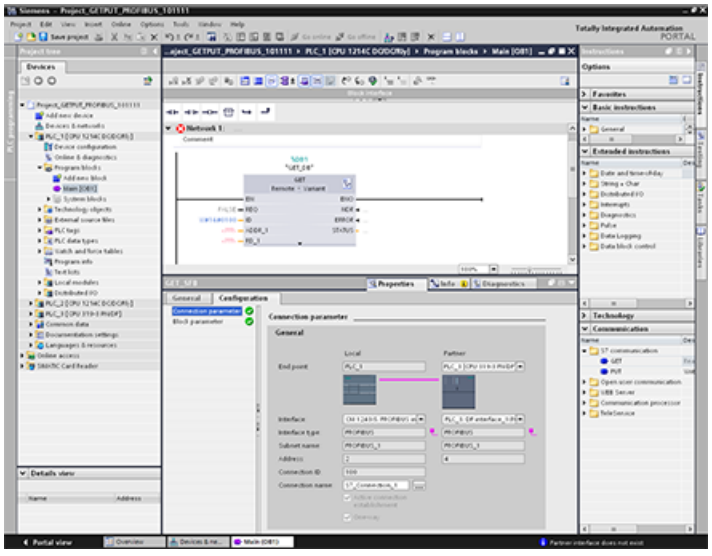
Özellik görünümündeki GET/PUT sembolü kırmızıdan yeşile döner.

Tam PROFINET S7 bağlantısı

"PLC_1" ve "PLC_3" arasındaki "Bağlantılar" tablosundaki "Ağ görünümü"nde çift yönlü bir S7 bağlantısı görüntülenir.



"Partner uç noktası" için "PLC_3" opsiyonunu seçin.



Sistem, aşağıdaki değişikliklerle yanıt verir:

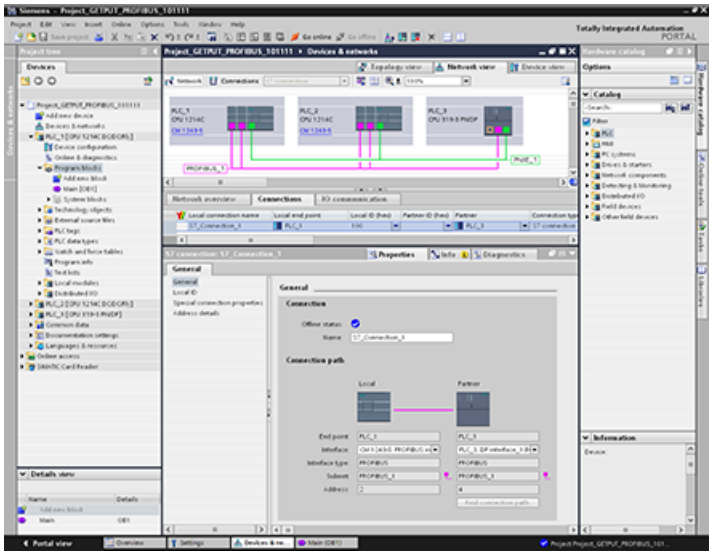
Tablo 11-74 Bağlantı parametreleri: Genel değerler

Parametre		Tanım
Bağlantı parametreleri: Genel	Uç nokta	"Yerel uç nokta", "PLC_1" (yazma korumalı) içerir. "Partner uç noktası" alanı, "PLC_3[CPU319-3PN/DP]" öğesini içerir: • Renk, kırmızıdan beyaza dönüşür. • "Partner" cihaz görseli görüntülenir. • PLC_1 ve PLC_3 cihaz görselleri arasında bir bağlantı çizgisi görüntülenir (mor renkli PROFIBUS hattı).
	Arayüz	"Yerel arayüz", "CPU1214C DC/DC/DC, PROFIBUS arayüzü (R0/S1)" öğesini içerir. "Partner arayüzü" şunları içerir: "CPU319-3PN/DP, PROFIBUS arayüzü (R0/S2)".
	Arayüz tipi	"Yerel arayüz tipi", "PROFIBUS"u alır; kullanım, yazma korumalıdır. "Partner arayüz tipi", "PROFIBUS"u içerir; kullanım, yazma korumalıdır. Arayüz tipi görselleri, yerel ve partner arayüz tiplerinin (mor renkli PROFIBUS sembolü) sağında görüntülenir.
	Alt ağ adı	"Yerel alt ağ adı", "PROFIBUS_1"i içerir; kullanım, yazma korumalıdır. "Partner alt ağ adı", "PROFIBUS_1"i içerir; kullanım, yazma korumalıdır.
	Adres	"Yerel adres", yerel IP adresini içerir; kullanım, yazma korumalıdır. "Partner adresi", partner IP adresini içerir; kullanım, yazma korumalıdır.
	Bağlantı ID'si	"Bağlantı ID'si", "100" içerir. Programlama editörünün [OB1] ana programında, GET/PUT bloğunun "bağlantı ID'si" de "100" değerini alır.
	Bağlantı adı	"Bağlantı adı", standart bağlantı adını alır (örnek: "S7_connection_1"); kullanım etkinleştirilmiştir.
	Aktif bağlantı kurma	Yerel CPU'yu, aktif bağlantı olarak seçmek için etkinleştirilmiştir ve yazma korumalıdır.
	Tek yönlü	Yazma korumalı ve etkinleştirilmiş. Not: "PLC_3" (bir S7-300 CPU319-3PN/DP), PROFIBUS ile bir GET/PUT bağlantısında yalnızca bir sunucu (bir istemci olarak değil) olarak işletilebilir, dolayısıyla yalnızca tek yönlü bir bağlantı mümkündür.

Özellik görünümündeki GET/PUT sembolü kırmızıdan yeşile döner.

Tam PROFIBUS S7 bağlantısı

"PLC_1" ve "PLC_3" arasındaki "Bağlantılar" tablosundaki "Ağ görünümü"nde tek yönlü bir S7 bağlantısı görüntülenir.



11.9 IP adresi üzerinden CPU'ya erişilemiyorsa ne yapmalısınız?

CPU'ya IP adresi üzerinden erişemiyorsanız, CPU için (geçici) bir acil durum IP adresi ayarlayabilirsiniz. Acil durum IP adresini üzerinden CPU bağlantısını yeniden kurabilir ve böylece bir cihaz konfigürasyonunu geçerli bir IP adresi ile yükleyebilirsiniz.

Acil durum IP adresinin gerekli olmasının nedenleri

Bir kişi aşağıdaki sorunlardan herhangi birine sahip bir proje indirirse, CPU'nuz erişilemez olabilir:

- CPU'nun PROFINET arayüzünün IP adresi, ağdaki başka bir cihazın bir kopyasıdır.
- CPU için alt ağ yanlıştır.
- Alt ağ maskesi, CPU'ya erişilemez kılar.

Bu durumlarda, CPU'ya artık STEP 7 üzerinden ulaşılamaz.

Bir acil durum IP adresinin atanması

Aşağıdaki durumlarda bir acil durum IP adresi atayabilirsiniz:

- STEP 7'deki cihaz konfigürasyonunda, IP protokolü için "Projede IP adresini ayarla" seçilidir.
- CPU, STOP işletim durumundadır.

Bu durumlarda, bir DCP aracından yararlanarak cihazın IP adresi için bir acil durum IP adresi ayarlayabilirsiniz. Örneğin SIMATIC Automation Tool, "IP adresi ayarla" adlı bir DCP komuta sahiptir. CPU'nun koruma kademesinden (Sayfa 156) bağımsız bir acil durum IP adresi ayarlayabilirsiniz. Bir DCP aracıyla geçici bir IP adresi ayarladıktan sonra, CPU üzerindeki bakım LED'i yanar. Ayrıca teşhis tamponu bir Ethernet arayüzü için bir acil durum adresinin etkinleştirildiğini gösteren bir girişi de görüntüler.

Bir acil durum IP adresi atandıktan sonra IP adresini geri yüklenmesi

Bir acil durum IP adresi etkinleştirildiğinde veya devre dışı bırakıldığında, teşhis tamponu sizi bilgilendirir. CPU'yu kapatıp tekrar açarak acil durum IP adresini sıfırlayabilirsiniz.

Bir acil durum IP adresi atadıktan sonra, geçerli bir IP adresine sahip CPU için bir STEP 7 projesini yükleyebilirsiniz. Projeyi yükledikten sonra, CPU'yu kapatıp tekrar açın.

11.10 OPC UA sunucusu

S7-1200 CPU'lar, OPC UA Micro-Embedded Profile'ı destekler. OPC UA Micro-Embedded Profile ile daha fazla bilgiyi, TIA Portal bilgi sisteminde bulabilirsiniz. Bundan başka S7-1200 CPU'lar, şunları destekler:

- OPC UA kullanıcı kimlik doğrulaması
- İletişim güvenliği
- Abonelikler
- Program değişkenlerinin okunması ve yazılması
- Yöntem çağrıları

S7-1200 CPU'lar, aşağıdaki alt bölümlerde anlatıldığı üzere OPC UA fonksiyonelliğinin bir alt kümesini destekler. OPC UA sunucuları hakkında daha fazla bilgiyi, OPC Foundation (https://opcfoundation.org/) web sitesinde bulabilirsiniz.

11.10.1 OPC UA sunucusunun konfigürasyonu

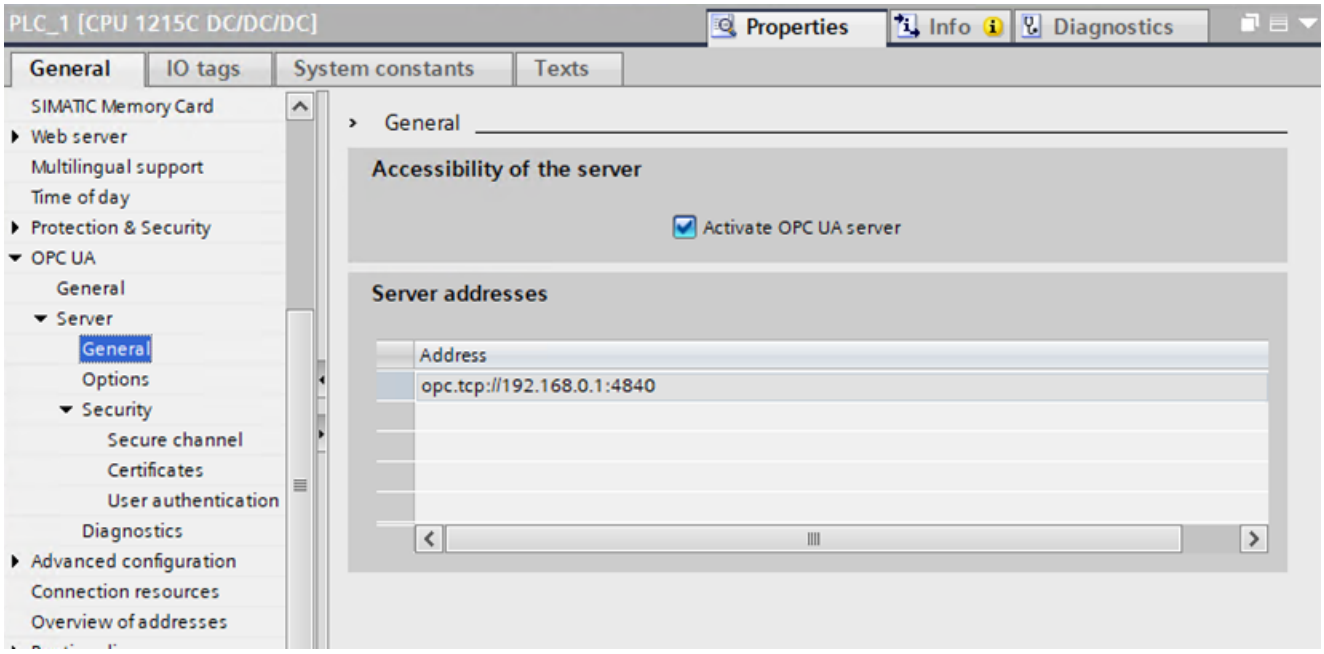
OPC UA sunucusunu kullanmak için, TIA Portal'da CPU'nun cihaz konfigürasyonundaki OPC UA ayarlarını yapılandırmalısınız.

11.10.1.1 OPC UA sunucusunun etkinleştirilmesi

OPC UA sunucusu standart olarak devre dışıdır. CPU'nun donanım özelliklerinde "OPC UA sunucusunu etkinleştir" opsiyon kutusunu seçerek OPC UA sunucusunu etkinleştirmelisiniz.

OPC UA sunucusunu etkinleştirmek için aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda "Genel" sekmesini seçin.
2. "Genel" penceresinde "OPC UA" öğesini seçin.
3. "OPC UA > Sunucu > Genel" penceresinde "OPC UA sunucusunu etkinleştir" opsiyon kutusunu etkinleştirin.



11.10.1.2 İşletim sırasında OPC UA sunucusunun davranışı

Sunucuyu etkinleştirdiğinizde ve projeyi S7-1200 CPU'ya yüklediğinizde, OPC UA sunucusu çalışmaya hazır durumda olur.

CPU'nun STOP işletim durumundaki davranışı

Etkinleştirilmiş bir OPC UA sunucusu, CPU "STOP" işletim durumuna geçse bile işletimde kalır. OPC UA sunucusu, OPC UA istemcilerinden gelen taleplere yanıt vermeye devam eder.

Ayrıntılı olarak sunucu yanıtı:

- PLC değişkenlerin değerleri talep edilirse, CPU "STOP" işletim durumuna geçmeden önce yazılmış olan değerleri geri döndürür.
- "STOP" işletim durumunda iken OPC UA sunucusuna değerler yazabilirsiniz. Ancak kullanıcı programı "STOP" işletim durumunda yürütülmediği için CPU bu değerleri işlemez. Buna karşın bir OPC UA istemcisi "STOP" işletim durumunda yazılan tüm değerleri CPU'nun OPC UA sunucusundan okuyabilir.

İşletimdeki OPC UA sunucusunda iken CPU yüklenmesi

OPC UA sunucusu işletimdeyken belli bazı nesnelerin yüklenmesi, CPU'nun durmasına yol açabilir. CPU durursa, sizden sunucuyu yeniden başlatmanız istenir. Sunucu yeniden başlatıldığında aktif bağlantılar kesilir. Sunucuyu yeniden başlatırken bağlantıları yeniden kurmalısınız.

Yeniden başlatmanın süresi aşağıdaki parametrelerden etkilenir:

- Veri yapısının kapsamı
- OPC UA adres bölümünde görünen değişken sayısı
- Aşağı doğru uyumlu veri tipi tanımı ayarı, OPC UA spesifikasyonu V1.03 uyumludur (tip sözlüğü etkinleştirildi)
- İletişim yükü ve minimum çevrim süresi ayarları (Sayfa 144-145)

OPC UA sunucusu şu şekilde davranır:

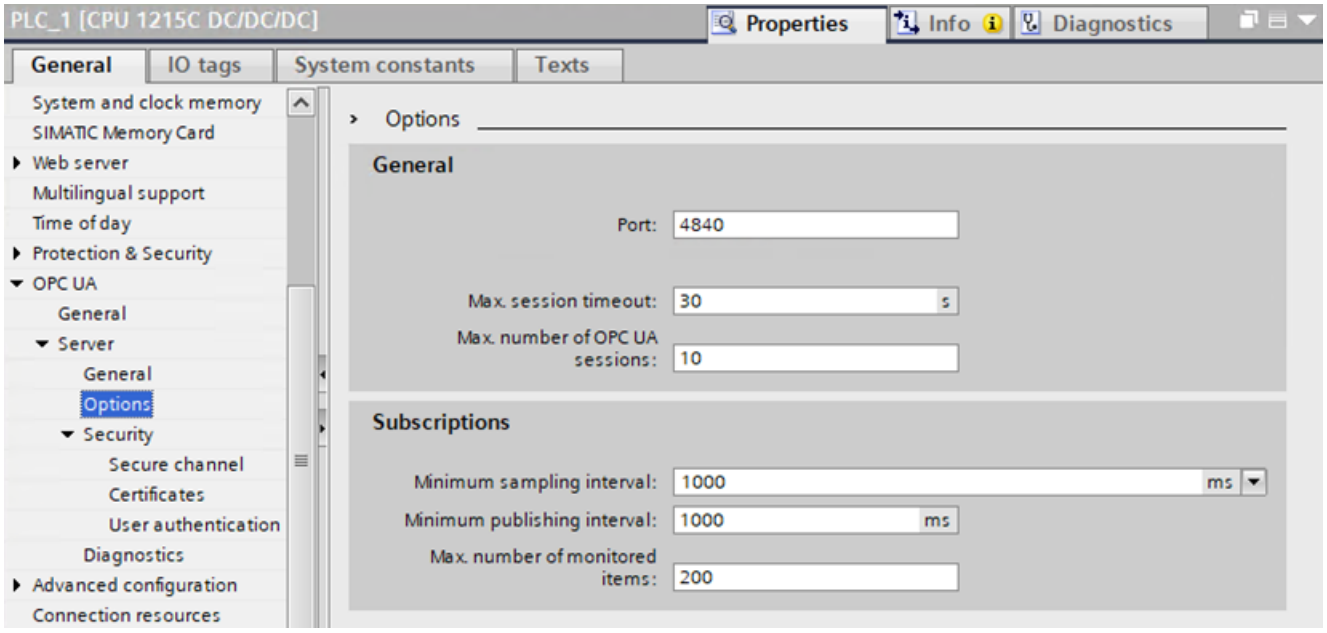
- CPU, "STOP" işletim durumundayken nesneler yüklenirse, OPC UA sunucusu STOP durumuna geçer ve yeniden başlatılır. STEP 7, bir uyarı göstermez.
- CPU, "RUN" işletim türündeyken nesneler yüklenirse, nesneler OPC UA ile ilgiliyse veya olabilecekse, OPC UA sunucusu STOP durumuna geçer. Değiştirilen OPC UA verileri nedeniyle, yeniden başlangıcın ardından OPC UA sunucusu yeniden başlatılır.
  OPC UA ile ilgili nesneler CPU'ya yüklenmeden ve OPC UA sunucusu durdurulmadan önce, STEP 7 önizleme iletişim kutusunda bir uyarı görüntülenir. Sunucunun yeniden başlatılması çalışan işlemle uyumlu değilse, yüklemeyi iptal edebilirsiniz. Bu uyarılar yalnızca OPC UA sunucusu çalışırken görüntülenir. OPC UA sunucusu etkinleştirilmemişse, değişen OPC UA verileri yükleme sürecini etkilemeyecektir.

Yükleme sırasında işletimdeki bir OPC UA sunucusunun davranış örnekleri

Örnek 1:
İstenen eylem: Programa başka bir kod bloğu eklemek istiyorsunuz. Veri blokları, girişler, çıkışlar ve markörler etkilenmez.
Sunucu tepkisi: Çalışmakta olan OPC UA sunucusu kesintiye uğramaz.
Örnek 2:
İstenen eylem: Yeni bir veri modülü yüklemek istiyorsunuz ve veri modülünü OPC UA ile ilgili değil olarak işaretlediniz.
Sunucu tepkisi: Çalışmakta olan OPC UA sunucusu kesintiye uğramaz.
Örnek 3:
İstenen eylem: Bir veri modülünün üzerine yazmak istiyorsunuz.
Sunucu tepkisi: STEP 7, değişikliklerin OPC UA ile ilgili verileri etkileyip etkilemediğini belirleyemez. Dolayısıyla bir uyarı mesajı, sunucunun yeniden başlatıldığına işaret eder.

11.10.1.3 OPC UA sunucusu için ayarlar

OPC UA sunucusunun ayarlarını "Opsiyonlar" iletişim kutusunda uyarlayabilirsiniz.

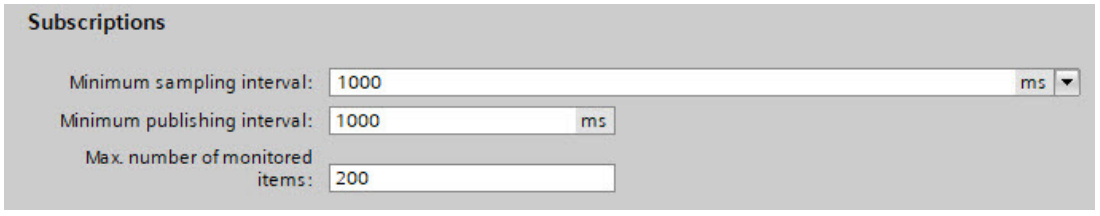


Aşağıdaki "Genel ayarlar" ve "Abonelik ayarları" tabloları, sunucunun yapılandırılabilir ayarlarına yönelik daha fazla bilgi vermektedir.

Her satır için standart değerler, CPU'nun % 20'lik standart iletişim yükü (Sayfa 84) ile çalışır. Gereksiniminize göre CPU iletişim yük yüzdesini artırın. Eylemler için mevcut iletişim yükü (örn. PROFINET), CPU yükünün yüzdesini etkiler.



Genel ayarlar	Aralık	Standart ayar
Portlar	1024 - 49151	4840
Maks. oturum zaman aşımı	1 sn - 600000 sn	30 sn
Maks. OPC UA oturumu sayısı	1 - 10	10



Abonelik ayarları	Aralık	Standart ayar
Minimum tarama süresi	100 msn, 250 msn, 500 msn, 1000 msn, 5000 msn, 10000 msn	1000 msn
Minimum yayımlanma aralığı	200 msn - 20000000 msn	1000 msn
Maks. denetlenen eleman sayısı	100 - 1000	200

11.10.1.4 OPC UA sunucusunun sınırlamaları

OPC UA sunucu arayüzlerini kullanırken aşağıdaki sınırlamalara uymalısınız:

- Sunucu arayüzlerinin sayısı
- OPC UA katılımcılarının sayısı
- Yöntemleri gerçekleştirdiyseniz, sunucu yöntemlerinin veya sunucu yöntemi örneklerinin sayısı

Arayüzler, yöntemler ve abonelikler için olan sınırlamalar

Aşağıdaki tablolar, S7-1200 CPU'lar ve hata güvenli S7-1200 CPU'lar için geçerlidir; bir konfigürasyonu derlerken ve yüklerken bu sınırlamaları dikkate almalısınız. Bir sunucu sınırlamasına uyulmaması, bir hata mesajına yol açar.

Arayüz sınırlamaları OPC UA sunucu arayüzleri	Maks. sayı
OPC UA sunucu arayüzleri[1]	2
Sunucu arayüzündeki kullanıcı tanımlı katılımcılar[2]	2000
Ad alanları[3]	18

[1] Bir sunucu arayüzü, "Yardımcı spesifikasyon" veya "Sunucu arayüzü" tipinde olabilir.

[2] Maksimum sayı, dahili olarak sunucu arayüzü tarafından tanımlanan katılımcıları da kapsar.

[3] Bu, tüm sunucu arayüzü tipleri için tanımlanmış ad alanlarının toplam sayısıdır: "Arayüz", "Yardımcı spesifikasyon" ve "Referans ad alanı".

Yöntem sınırlamaları	Maks. sayı
Sunucu yöntemleri[1]	20
Tüm sunucu arayüzleri ve ilişkili sunucu yöntemleri için sunucu yöntemi örnek DB'leri	20
Sunucu yöntemleri için girdi argümanları	20
Sunucu yöntemleri için çıktı argümanları	20

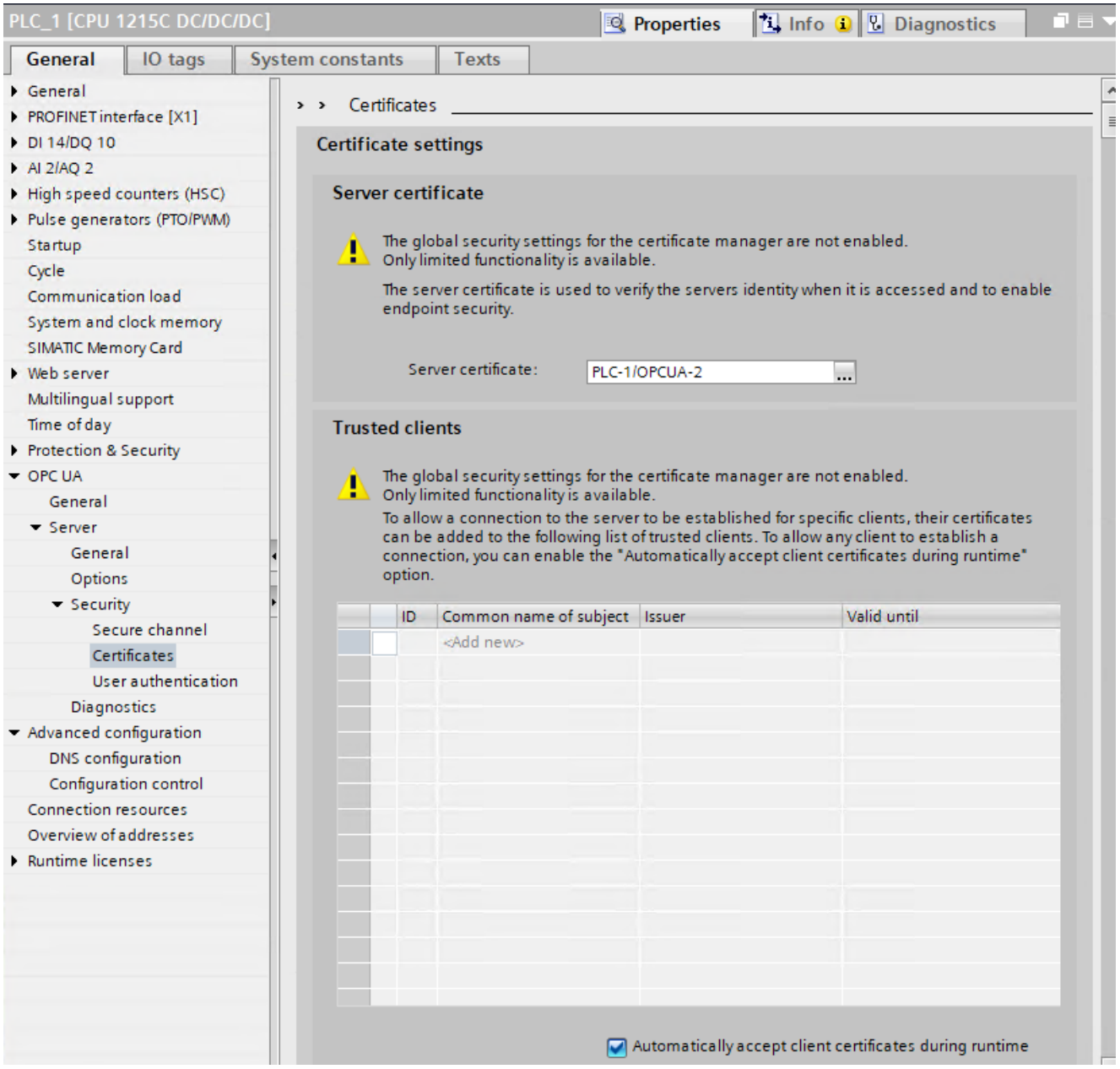
[1] Bu sayı aşıldığında, ilgili yöntem yürütülemez olur ve bir istemci çalışma zamanı hatası verilir. Aynı yöntemin birden çok örneğini kullanabilirsiniz.

Abonelik sınırlamaları	Maks. sayı
Eş zamanlı abonelikler[1]	50
Oturum başına abonelik	5

[1] 'Eş zamanlı abonelikler', projelendirilmiş eş zamanlı oturum sayısının oturum başına maksimum abonelik sayısıyla çarpımına eşittir.

11.10.2 OPC UA sunucusunun güvenliği

Sunucu güvenliği, iletişimin güvenliğini sağlamak için kullanılan çeşitli güvenlik yöntemlerindendir. Diğer iki yöntem ise TIA Portal güvenliği ve PLC güvenliğidir.
OPC UA sunucusu, etkinleştirme için bir sertifika gerektirir. TIA Portal, sunucuyu etkinleştirdiğinizde bir sertifika oluşturur. Bu sertifikayı PLC özelliklerinde değiştirebilirsiniz.



NOT

S7-1200 için sınırlı sayıda sertifika

S7-1200, sistem tarafından belirlenen 64 sertifika sınırı söz konusudur.

Bu sayıya, tüm sertifikalar dahildir (örn. web sertifikaları, OPC UA sertifikaları ve OUC sertifikaları).

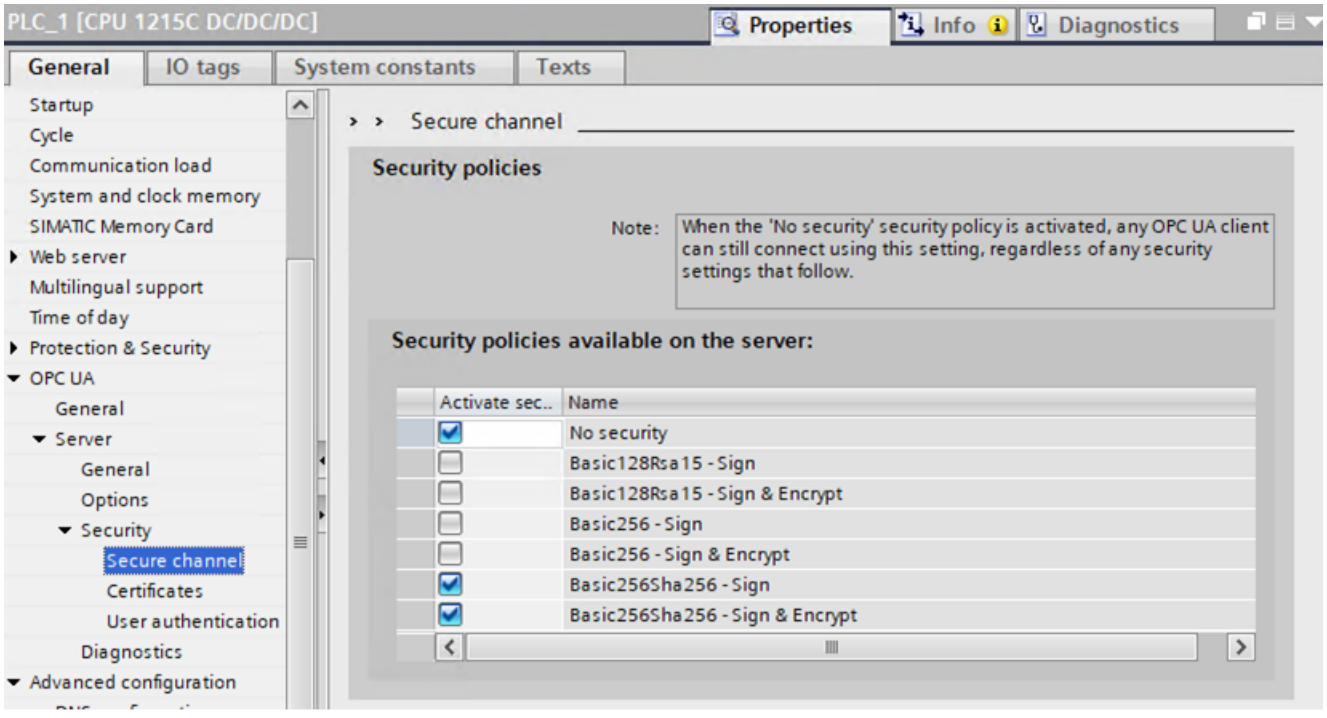
64'ten fazla sertifikanız varsa, TIA Portal, maksimum 64 sertifika sayısını aştığınızı bir hata mesajı ile görüntüler. Bunun ardından bazı sertifikaları PLC konfigürasyonundan kaldırmanız gerekir.

11.10.2.1 Desteklenen güvenlik direktifleri

Çalışma zamanında bir güvenlik ilkesinin seçilmesi, istemci ve sunucu iletişiminin güvenliğini belirler.

OPC UA için güvenlik direktifini seçmek üzere aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda "Genel" sekmesini seçin.
2. "Genel" penceresinde "OPC UA" öğesini seçin.
3. OPC UA altında, "Sunucu > Güvenlik > Güvenli kanal" öğesini seçin.
4. Kullanılabilir güvenlik direktifleri listesinden uygun olan direktifi seçin.



Tablo 11-75 S7-1200 tarafından desteklenen OPC UA güvenlik direktifleri

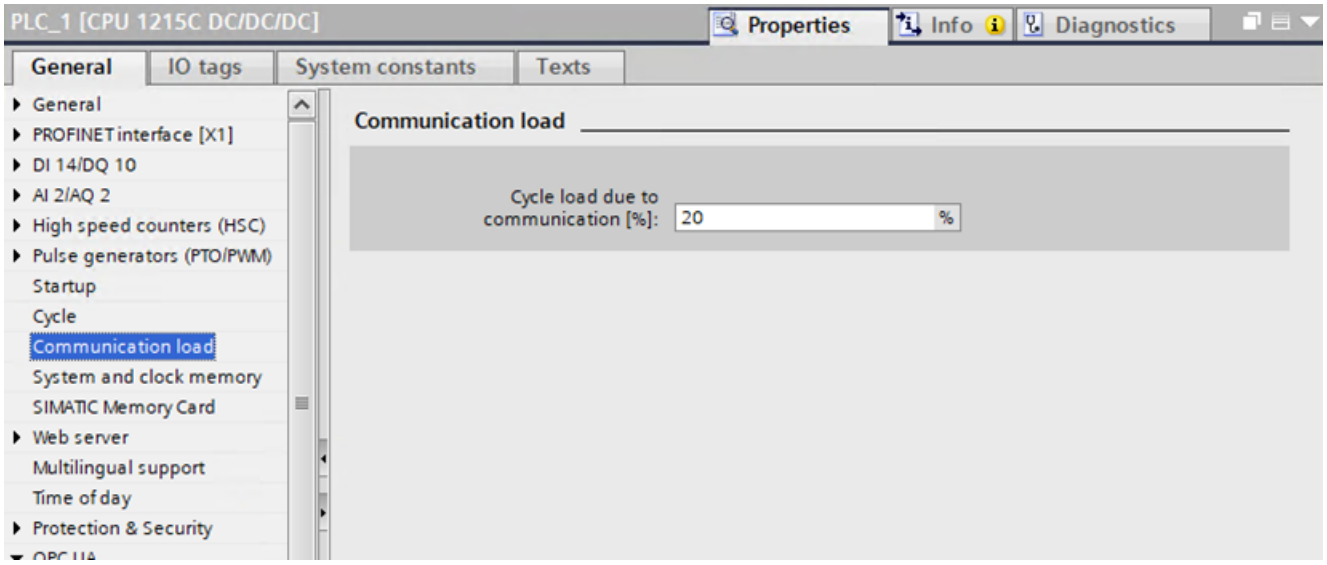
Güvenlik direktifi	Standart olarak etkinleştirilmiş mi?
Güvenlik yok	Evet
Basic128Rsa15 – Sign	Hayır

Güvenlik direktifi	Standart olarak etkinleştirilmiş mi?
Basic128Rsa15 – Sign & Encrypt	Hayır
Basic256 – Sign	Hayır
Basic256 – Sign & Encrypt	Hayır
Basic256Sha256 – Sign	Evet
Basic256Sha256 – Sign & Encrypt	Evet

Güvenli bir OPC UA bağlantısının oluşturulması

OPC UA istemcisi ile S7-1200 sunucusu arasındaki ilk bağlantının kurulumundan sonra, daha zorlu OPC UA güvenlik direktiflerinin iletişim yükü üzerinde olumsuz bir etkisi olmaz.

Ancak, ilk bağlantının kurulması sırasında, harici cihazlarla (TIA Portal, HMI cihazları, vb.) çevrimiçi bağlantıları sürdürmek için iletişim yükü yüzdesini artırmak gerekli olabilir (aşağıya bakın).



İlk OPC UA bağlantısının oluşturulmasında iletişim yükünü artırmamak için gerekirse aşağıdaki olanaklardan birini kullanabilirsiniz:

- Harici cihazlara olan tüm çevrimiçi bağlantıları kesin ve ilk OPC UA bağlantısı oluşturulduktan hemen sonra bağlantıları yeniden kurun.
- Zaman aşımına uğrayan harici cihazlara çevrimiçi olarak yeniden bağlanın.

NOT

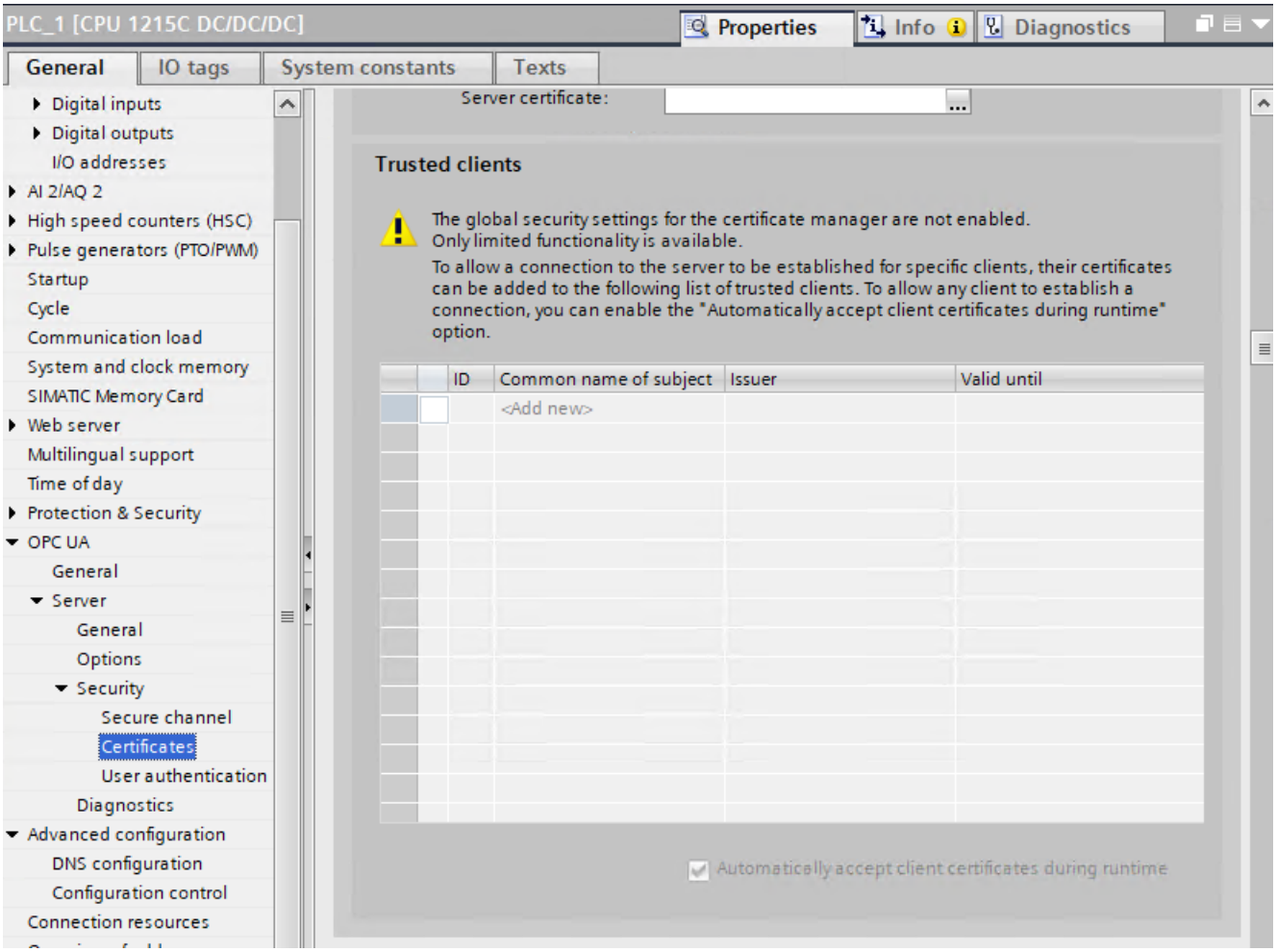
İletişimden gelen çevrim yükü arttıkça, mevcut aktif iletişime göre çevrim süresi uzar.

11.10.2.2 Güvenilir istemciler

OPC UA sunucusunu yalnızca güvenilir istemcilerden gelen bağlantılara izin verecek şekilde yapılandırabilirsiniz. Standart istemci konfigürasyonu, istemci sertifikalarını otomatik olarak kabul eder.

Güvenilir istemciler listesini tanımlarsınız. Sunucu, güvenilir istemcileri tanımak için sertifikalardan yararlanır. Seçildiğinde, yalnızca güvenilir sertifikaları olan istemcilerin çalışma zamanı boyunca sunucuya bağlanmasına izin verilir.

Güvenilir istemcileri belirlemek için, bu istemcilerin sertifikalarını TIA Portal'da "Donanım Özellikleri > OPC UA güvenliği > Güvenli kanal > Güvenilir istemciler" altındaki "Güvenilir istemciler" listesine ekleyin.



11.10.2.3 Kullanıcı kimlik doğrulaması

S7 1200, OPC UA sunucusu için hem konuk kullanıcı kimlik doğrulamasını hem de kullanıcı adı ve şifre kimlik doğrulamasını destekler. Ön ayar, konuk kullanıcı kimlik doğrulama işlemidir.

Konuk kullanıcı kimlik doğrulaması kullanıldığında, bağlanırken istemciden bir kullanıcı adı ve şifre talep edilmez. Konuk kullanıcı kimlik doğrulaması, yalnızca işletime alma aşamasında kullanılmalıdır.

Konuk kullanıcı kimlik doğrulaması devre dışı bırakıldığında, bu durumda yalnızca önceden yapılandırılmış bir kullanıcı adı ve şifre içeren istemcilerin bağlanmasına izin verilir. Siemens, işletim sırasında yalnızca kimliği doğrulanmış kullanıcılara erişim verilmesini önermektedir.

NOT

Atanan kullanıcı rolleri ile birlikte projelerin yüklenmesi

S7-1200 V4.5 ve V4.6 OPC UA sunucusu, "Güvenlik ayarları > Kullanıcılar ve roller" için STEP 7 proje ayarlarıyla uyumlu değildir. V4.5'ten önceki sürümlerde, bir proje yöneticisi ve OPC UA sunucusu erişimine sahip kullanıcı tanımlı bir rol oluşturularak ve OPC UA istemcisinin proje yöneticisinin kullanıcı adı ve şifresiyle kullanıcı kimlik doğrulaması etkinleştirilerek, bir OPC UA istemcisinin bu oturum açma verileri doğrulanabiliyordu. Firmware V4.5 ve V4.6'ya sahip CPU'lar için bu tür projelerin yüklenmesi başarısız olmaktadır.

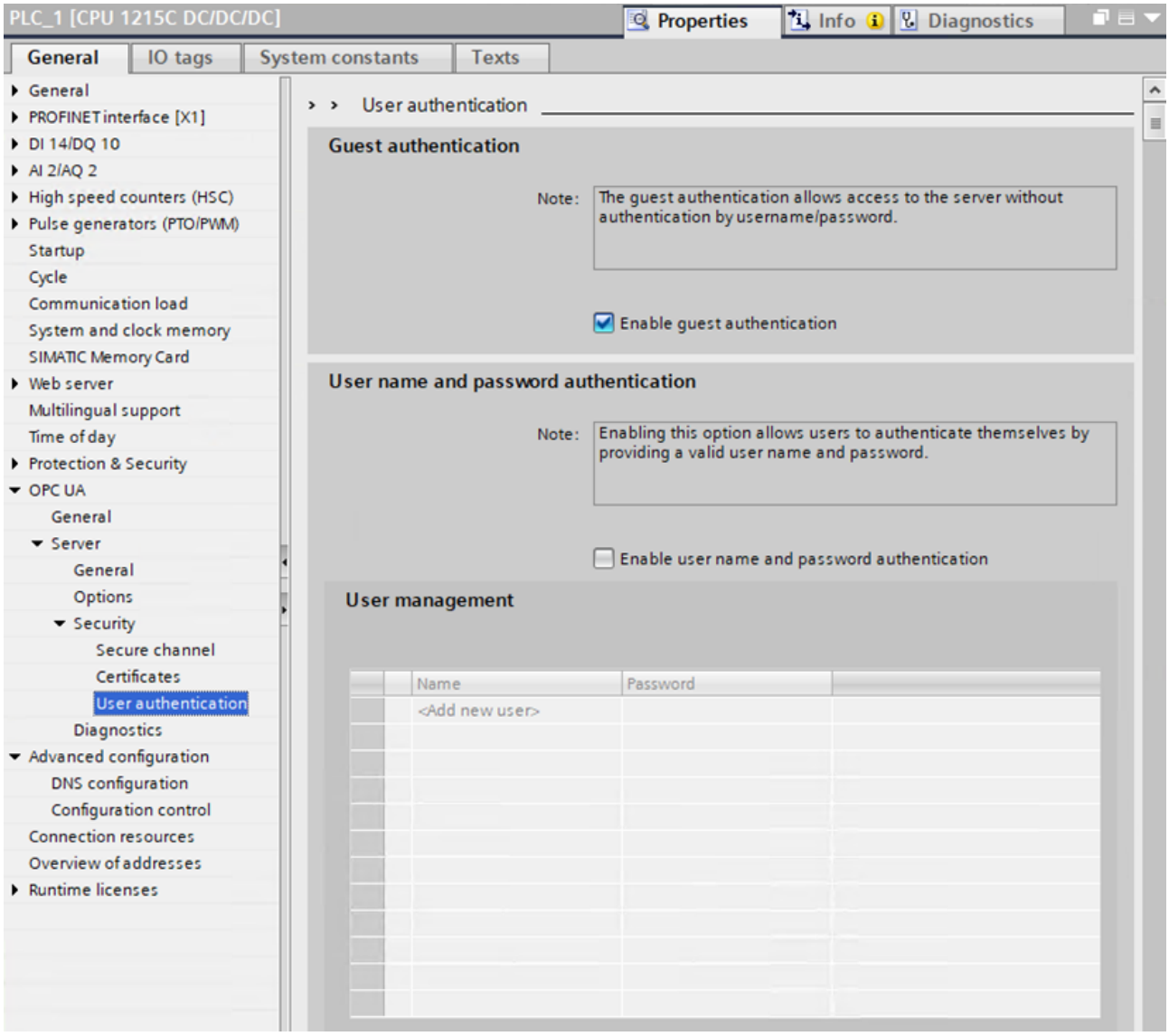
Proje yüklenemezse, yüklemeyi etkinleştirmek için aşağıdaki opsiyonlardan birini seçin:

- "Genel" OPC UA ayarlarında "Projenin güvenlik ayarları üzerinden ek kullanıcı yönetimini etkinleştir" opsiyonunun seçimini kaldırın.
- CPU Firmware'i V4.4'e düşürün

Bu konuyla ilgili ayrıntılı bilgiyi, müşteri destek bölümünde bulabilirsiniz.

OPC UA sunucusu için geçerli kullanıcıları, iki yöntemle tanımlayabilirsiniz:

- CPU'nun cihaz konfigürasyonunda geçerli kullanıcıları tanımlayabilirsiniz.
- Projeniz için global güvenliği etkinleştirdiyseniz, sunucu, TIA Portal projelerinin münferit kullanıcıları için belirttiğiniz roller üzerinden geçerli kullanıcıları tanıyacaktır. Maksimum 21 adet kullanıcı yapılandırabilirsiniz.



11.10.3 OPC UA sunucusunun arayüzü

S7 1200 OPC UA sunucusu, standart SIMATIC sunucu arayüzünü destekler. Ancak, bu seçim üzerinden CPU ve DB değişkenlerinin otomatik olarak "yayımlanmasını" desteklemez. Bunun yerine sunucu arayüzünün yapısını ve içeriğini TIA Portal'da tanımlamanız ve ardından bunu PLC'ye yüklemeniz gerekecektir.

Bir sunucu arayüzü eklemek için aşağıdaki adımları uygulayın:

1. Proje gezgininde PLC adına tıklayın.
2. "OPC UA iletişimi" öğesini seçin.
3. "Sunucu arayüzleri" öğesini seçin.
4. "Yeni sunucu arayüzü ekle" öğesini seçin ve "TAMAM" butonuna tıklayın.
5. PLC programınızdaki değişkenleri yeni sunucu arayüzüne girin.
6. Sunucuyu PLC'nize yükleyin.

Bir sunucu arayüzü eklerken, mevcut tüm değişkenlerin, ekranın OPC UA elemanlarında listelendiğini dikkate alın. Bu elemanları, "OPC UA elemanları" penceresinden "OPC UA sunucu arayüzü" penceresine sürükleyebilirsiniz. Bir tutarlılık denetimi, sunucu arayüzünün içeriğini doğrular. Arayüzü, bir XML dosyasına dışa aktarabilirsiniz.

Projenize bir OPC UA sunucu arayüzü eklediğinizde, OPC UA sunucu arayüzünü belirlemiş olursunuz.

OPC UA sunucu arayüzünü tanımlamak için, OPC UA sunucusu oluşturma iletişim kutusunda aşağıdaki opsiyonlardan birini seçin:

- Şunu seçin: "Sunucu arayüzü", tip: Arayüz
- Şunu seçin: "Yardımcı spesifikasyon", tip: Yardımcı spesifikasyon
- Şunu seçin: "Yardımcı spesifikasyon", tip: Referans ad alanı

11.10.3.1 Desteklenen veri tipleri

OPC Foundation (https://opcfoundation.org/), değişkenlerin değer özniteliğinin yapısını ve bunların değişken tiplerini belirten bir set tanımlar, bu set desteklenen veri tiplerini içerir. S7 1200, bu veri tiplerinin (Sayfa 99) bir alt kümesini ve bu veri tiplerinden türetilmiş tanımlı diğer tipleri destekler.

Aşağıdaki tabloda, S7-1200 tarafından desteklenen veri tipleri listelenmektedir:

SIMATIC tipi	OPC UA tip adı	Katılımcı kodu
Bool	Boole değeri	i=1
SInt	SByte	i=2
USInt	Bayt	i=3
Int	Int16	i=4
UInt	UInt16	i=5
DInt	Int32	i=6
UDInt	UInt32	i=7
Real	Float	i=10
LReal	Double	i=11
WString	String	i=12
DWord	StatusCode	i=19

SIMATIC tipi	OPC UA tip adı	Katılımcı kodu
DATE	UInt16	i=5
TOD	UInt32	i=7
TIME	Int32	i=6
DTL	Yapı	-/-

Bu listenin desteklenen temel katılımcı tiplerini temsil ettiğini dikkate alın. Çok sayıda SIMATIC veri tipinin temel katılımcı tiplerine atanabilmesi nedeniyle, desteklenen tüm katılımcı tiplerini içeren eksiksiz bir liste değildir. Bir temel katılımcı tipine atanabilen her bir SIMATIC veri tipi de desteklenen bir katılımcı tipidir.

S7-1200 CPU'lar, yapılandırılmış veri tiplerinin (yapılar ve Array'ler) yanı sıra sunucu yöntemlerini de destekler.

S7-1200 CPU'lar, birlikleri (unions) desteklemez.

S7 1200, desteklenmeyen türlere sahip bir sunucunun indirilmesini kabul eder. Ancak, istemci desteklenmeyen tipte bir katılımcıyı okumaya veya yazmaya çabalarsa, PLC bir hata yayınlar.

11.10.3.2 PLC gösterimi

OPC UA sunucu arayüzü, PLC'yi tanımlayan özelliklerle PLC'yi betimleyen katılımcılar sunar. Bunlar, OPC UA sunucusu aktif iken her zaman kullanılabilir durumdadır.

SIMATIC sunucusu standart arayüzü aşağıdaki bilgileri sağlar:

Öznitelik	**Değer**		
NodeID (katılımcı ID'si)	Ns=SI;s=PLC		
BrowseName	SI:<PLC> Burada <PLC>, kullanıcının TIA Portal projesinde PLC'ye atamış olduğu addır.		
Referanslar	**NodeClass**	**BrowseName**	**Yorum**
DeviceSet nesnesinin ComponentOf'u			
Nesneler klasörünün OrganizedBy'ı			
HasTypeDefinition	ObjectType	SimaticDeviceType	Cihaz tipinden türetilir
HasProperty	Değişken	DeviceManual	
HasProperty	Değişken	DeviceRevision	
HasProperty	Değişken	EngineeringRevision	
HasProperty	Değişken	HardwareRevision	
HasProperty	Değişken	Sembol	
HasProperty	Değişken	Manufacturer	
HasProperty	Değişken	Model	
HasProperty	Değişken	OperatingMode	
HasProperty	Değişken	OrderNumber	
HasProperty	Değişken	RevisionCounter	
HasProperty	Değişken	SerialNumber	
HasProperty	Değişken	SoftwareRevision	

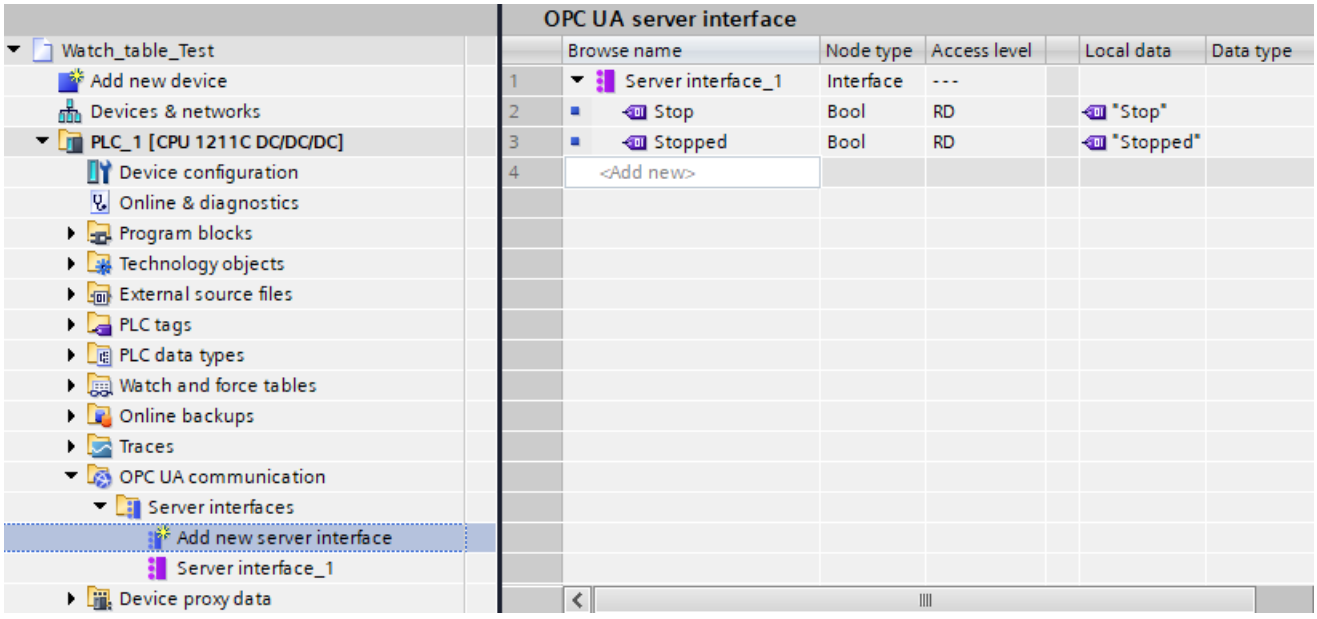
11.10.3.3 Yüklenebilir sunucu arayüzleri

OPC UA sunucu arayüzünün bileşenlerini TIA Portal'da oluşturur ve düzenlersiniz. OPC UA sunucu arayüzlerini iki türlü tanımlayabilirsiniz:

- Yardımcı spesifikasyon olarak harici oluşturulan XML dosyalarının kullanımı
- Programınıza dahil edebileceğiniz veri bloğu elemanları ve genel değişkenleri baz alan ve doğrudan TIA Portal'da tanımlanmış bir sunucu arayüzü kullanımı

Hedef sisteme yüklendiğinde, bu bileşenler bir OPC UA istemcisi tarafından görülebilen sunucu arayüzünü belirler.

OPC UA'nın TAG okumasına ve yazmasına izin vermek için spesifik TAG özniteliklerini ayarlamanız gerekir.



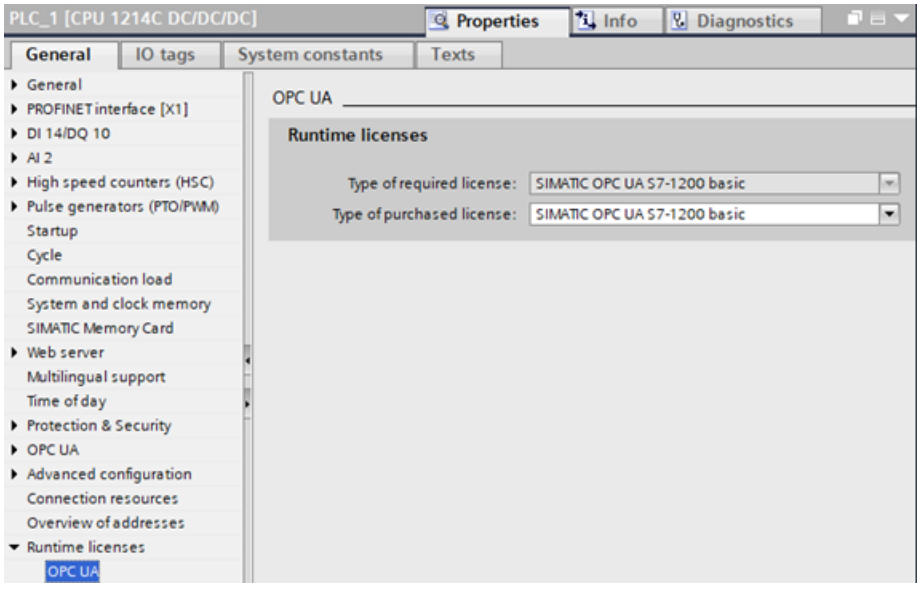
Runtime lisansı gereklidir

S7-1200 CPU için OPC UA sunucusunu çalıştırmak üzere bir Runtime lisansı gerekir. Şu lisanslar sunulmaktadır:

- SIMATIC OPC UA S7-1200 Basic DVD 6ES7823-0BA00-2BA0
- SIMATIC OPC UA S7-1200 Basic DL 6ES7823-0BE00-2BA0

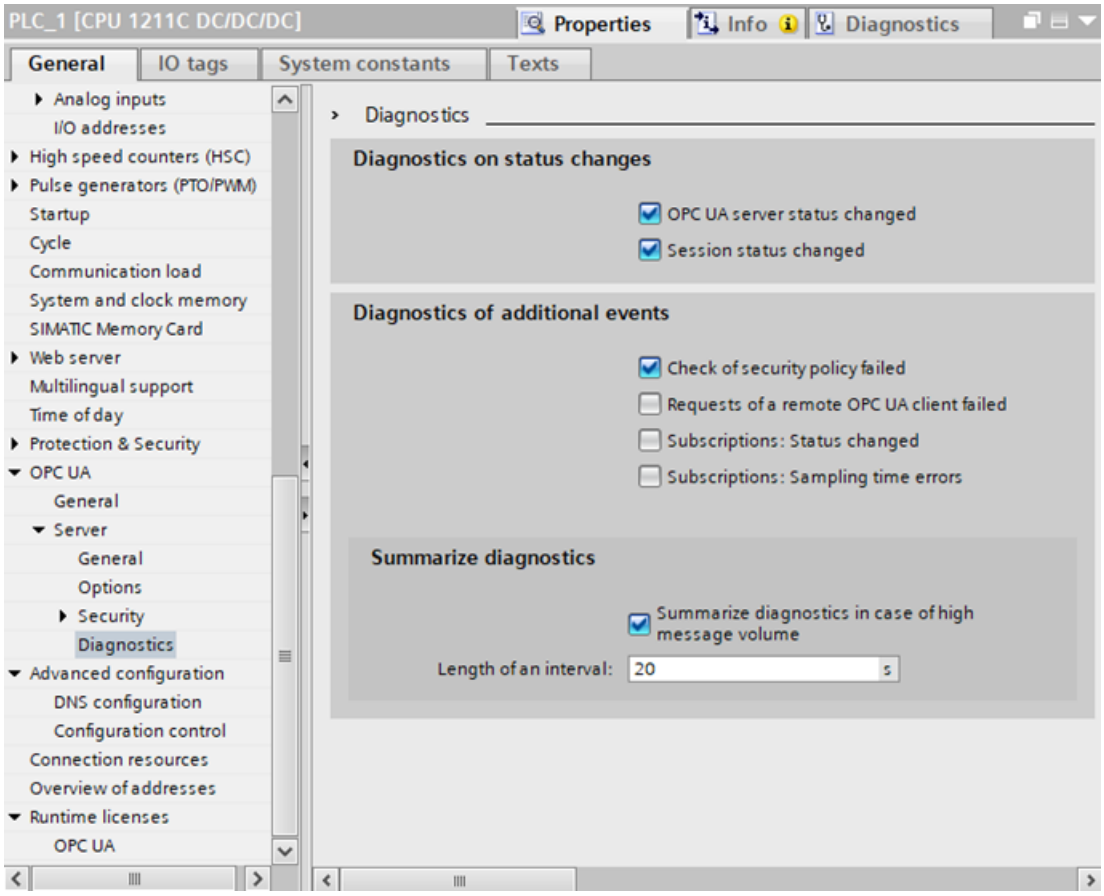
Gereken lisans tipi, "Özellikler > Genel > Runtime lisansları > OPC UA > Gerekli lisans tipi" altında görüntülenir. Gerekli lisansı satın aldığınızı doğrulamak için aşağıdaki yöntemi izleyin:

1. CPU'nun özelliklerinde iken, "Runtime lisansları > OPC UA" üzerine tıklayın.
2. "Satın alınan lisansın tipi" açılır listesinden gerekli lisansı seçin.



11.10.4 OPC UA teşhis tamponu

TIA Portal'da OPC UA teşhisini yapılandırın ve ayarları PLC'ye yükleyin. OPC UA istemcisi ile teşhis tamponundaki girişleri denetleyebilirsiniz.



Durum değişikliğine yönelik teşhis mesajları yoğun frekansta görüntülenmez. Bu mesajların her biri, OPC UA'nın kullanımı için büyük önem taşır.

Teşhis mesajlarının özeti, yalnızca OPC UA sunucusunun teşhis mesajları için geçerlidir. Global CPU güvenlik olayları için farklı ayarlar seçebilirsiniz.

Mesaj hacmi yüksek olduğunda, teşhis mesajlarının sayısını sınırlamak adına, mesaj alımı için saniye cinsinden bir aralık belirleyebilirsiniz. İlgili değer aralığı, 1 ila 7200 saniyedir.

NOT

OPC UA sunucusu etkinleştirilmiş/devre dışı bırakılmış

OPC UA teşhis menüsündeki ayarlar, yalnızca OPC UA sunucusu etkinleştirildiğinde değiştirilebilir.

OPC UA sunucusunu devre dışı bırakırsanız, tüm ayarlar grileşir ve yazma korumalı olur.

11.10.4.1 OPC UA limitine ulaşıldı

Bir sistem limitime ulaşıldığında, örneğin veri sağlayamadığında veya sunucuyu/istemciyi olağan şekilde kullanamadığınızda, sunucu sizi bilgilendirir.

OPC UA sunucusu, abonelik/denetlenen eleman sayısı, kayıtlı eleman sayısı, abonelik tarama hızı ve bağlanmış istemci sayısına yönelik belirli sistem limitlerine sahiptir.

Bu sistem limitlerinden birine ulaşıldığında, sunucu teşhis tamponunda bir veri girişi oluşturur.

Teşhis tamponuna yeni bir veri girişi eklenirse, bir teşhis mesajı görüntülenir:

OPC UA sunucusu: Limit, <Limit adı> tarafından aşıldı.

Teşhis mesajı aşağıdaki limitlerden birini yayınlar:

- Oturum sayısı
- Denetlenen eleman sayısı
- Abonelik sayısı
- Kayıtlı katılımcı sayısı
- Sunucu yöntemleri sayısı
- Çağrı başına denetlenen elemanların sayısı
- Browse başına katılımcıların sayısı
- Read başına katılımcıların sayısı
- RegisterNodes başına katılımcıların sayısı
- TranslateBrowsePathsToNodeIds başına katılımcıların sayısı
- Write başına katılımcıların sayısı
- MethodCall başına katılımcıların sayısı
- Bellek tüketimi

NOT

Genellikle bir hizmet talebi (Servicerequest), önceden yapılandırılmış bir limiti veya sistem limitini aştığında, sunucu bir hizmet hatası ile tepki verir.

İlgili kullanım durumları aşağıda gösterilmektedir:

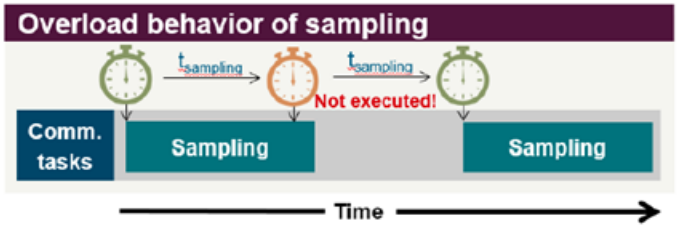
Tablo 11-76 "OPC UA limitlerime ulaşıldı" için uygulama durumu

Kullanıcı eylemi	Beklenen davranış	Limit
İstemci, HWCN'nin izin verdiğinden daha fazla sayıda oturum oluşturur	CPU, teşhis tamponuna aşılan limiti içeren bir teşhis mesajı ekler.	Oturumlar
İstemci, HWCN'nin izin verdiğinden daha sayıda denetlenen eleman oluşturur	CPU, teşhis tamponuna aşılan limiti içeren bir teşhis mesajı ekler.	Denetlenen elemanlar
İstemci, izin verilenden daha fazla abonelik oluşturur (PLC tipine bağlı olarak)	CPU, teşhis tamponuna <Statuscode> durum koduyla bir teşhis mesajı ekler.	İstemci, desteklenmeyen bir kimlik token'ına bağlanmaya çalışıyor

Kullanıcı eylemi	Beklenen davranış	Limit
İstemci, HWCN'nin izin verdiğinden daha fazla sayıda katılımcı kaydeder	CPU, teşhis tamponuna aşılan limiti içeren bir teşhis mesajı ekler.	Kayıtlı katılımcılar
Çok sayıda oturum / abonelik / kayıtlı katılımcı sayısı nedeniyle istemci, sunucudan bir BadOutOfMemory yanıtı tetikliyor	CPU, teşhis tamponuna aşılan limiti içeren bir teşhis mesajı ekler.	Bellek tüketimi
İstemci, bir hizmetin işletim sınırını aşıyor (tek bir talepteki işlem sayısı, sunucu tarafından belirlenmiş olan limiti aşıyor).	CPU, teşhis tamponuna aşılan limiti içeren bir teşhis mesajı ekler.	Browse başına katılımcı Read başına katılımcı Write başına katılımcı MethodCall başına katılımcı RegisterNodes başına katılımcı Translate başına katılımcı

"Aboneliklerin aşırı yük davranışı" için teşhis mesajı

Farklı değişkenlerin bir OPC UA aboneliğinde OPC UA sunucusu, değer değişiklikleri için elemanları önceden tanımlanmış aralıklarla (tarama aralıkları) denetler. Bu denetim (tarama), elemanların sayısından ve veri tipinden bağımsız olarak belirli bir süre ister. Tarama tamamlandığında, sunucu sonuçları yayımlar ve elemanları istemciye gönderir. Bekleme kuyruğu çok fazla eleman içeriyorsa, iletişim yığınına "aşırı yük" binebilir. CPU belirtilen tarama aralığındaki tüm elemanları denetleyemez, bu nedenle bir sonraki tarama görevine atlar.



Ayarlanan tarama hızına ulaşılamazsa, sunucu teşhis tamponunda bir "Tarama aşırı yükü" veri girişi oluşturur.

Aşırı yük durumunda OPC UA teşhis mesajları için örnekler:

OPC UA sunucusu: 100 msn tarama hızına ulaşılamadı. 12345678 numaralı abonelik ID'sinin aşırı yükü

OPC UA sunucusu: Tarama hızına ulaşılamadı. -Son 20 saniye içinde 3 adet oluşum için toplu mesaj

11.10.4.2 OPC UA güvenlik olayları

OPC UA sunucusu, bir güvenlik olayı ortaya çıktığında sizi bilgilendirir, bu sayede, OPC UA sunucusu tehlikeye girerse veya saldırılar algılanırsa tepki verebilirsiniz. OPC UA sunucu/istemci sisteminde belirli OPC UA güvenlik olaylarının görülmesi, teşhis tamponuna bir mesaj girişini tetikler.

Bir güvenlik olayına örnek olarak, yanlış kimlik doğrulama verileri nedeniyle bir oturumun veya bağlantı talebinin reddedilmesini verebiliriz.

OPC UA sunucusu, güvenlik sorunları olması durumunda şu mesajı verir:
"OPC UA sunucusu: Güvenlik denetimleri başarısız"

Bir güvenlik denetimi başarısız olursa, uyarı mesajları oluşturulur ve işlem yürütülmez.

OpenSSL, UMAC 711 veya OPC UA yığın belleği gibi diğer güvenlik bileşenleri genellikle bir StatusCode yayınlar ve sunucu, bu StatusCode'u teşhis mesajına yazar.

Sunucu, her olumsuz yanıt için bir mesaj oluşturur. En sık karşılaşılan kullanım durumları aşağıda gösterilmiştir:

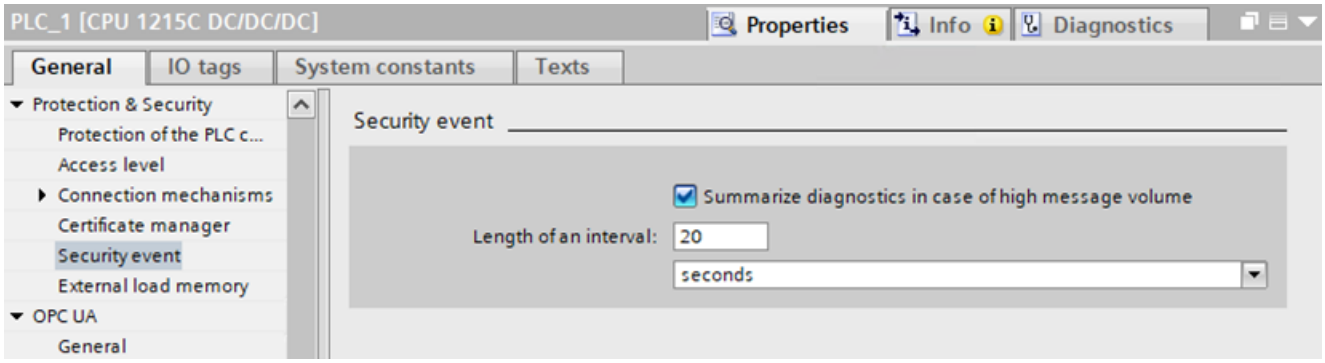
Kullanıcı eylemi	Beklenen davranış	StatusCode
İstemci, desteklenmeyen bir güvenlik modeline bağlanmaya çalışıyor	CPU, teşhis tamponuna <Statuscode> durum koduyla bir teşhis mesajı ekler.	16#8054_0000 (BadSecurityModeRejected)
İstemci, desteklenmeyen bir güvenlik direktifine bağlanmaya çalışıyor	CPU, teşhis tamponuna <Statuscode> durum koduyla bir teşhis mesajı ekler.	16#8055_0000 (BadSecurityPolicyRejected)
İstemci, desteklenmeyen bir kimlik token'ına bağlanmaya çalışıyor	CPU, teşhis tamponuna <Statuscode> durum koduyla bir teşhis mesajı ekler.	16#8020_0000 (BadIdentityTokenInvalid)
İstemci yanlış kimlik doğrulama işlemi ile bağlanmaya çabalıyor (yanlış kullanıcı adı veya şifre)	CPU, teşhis tamponuna <Statuscode> durum koduyla bir teşhis mesajı ekler.	16#8021_0000 (BadIdendityTokenRejected)
İstemci geçersiz bir sertifikaya bağlanmaya çabalıyor (sertifikanın geçersiz olmasının çeşitli olası nedenleri vardır)	CPU, teşhis tamponuna <Statuscode> durum koduyla bir teşhis mesajı ekler.	Geçersizliğe bağlı olarak, örn. 16#8014_0000 (BadCertificateTimeInvalid) 16#801A_0000 (BadCertificateUntrusted)

Güvenlik mesajlarının türleri

İki tür güvenlik mesajı mevcuttur: CPU geneli güvenlik mesajları ve OPC UA mesajlarının güvenliği.

CPU genelinde güvenlik mesajları

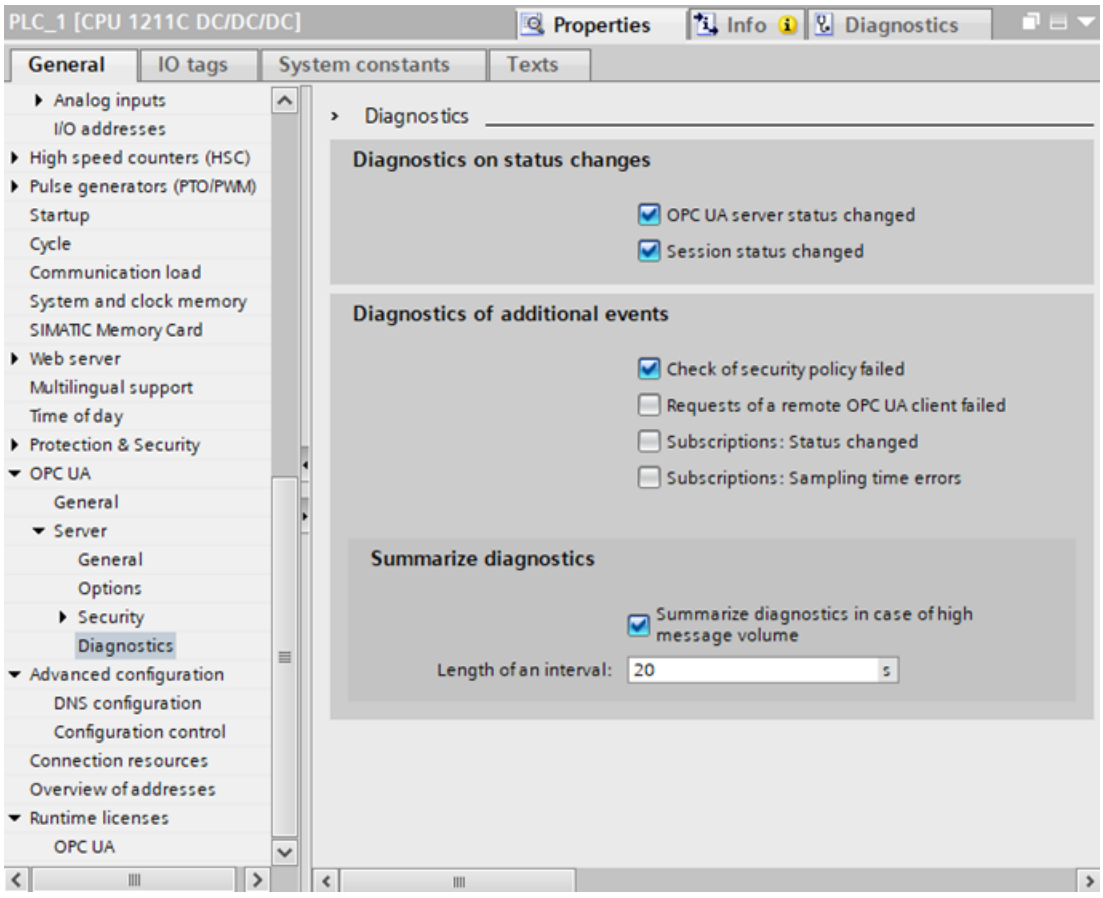
Güvenlik görevlisi tarafından CPU genelinde kritik olarak sınıflandırılan senaryolarda oluşturulan teşhis mesajları, CPU geneli güvenlik mesajları olarak adlandırılır. Bu tür mesajlar özel bir alarm etki alanı kullanır. Bu sayede, HMI cihazlarının bu mesajları filtrelemesine ve onlara farklı şekilde davranması sağlanır. Bu tür güvenlik mesajları, bir kullanıcının, CPU Web sunucusunda veya OPC UA sunucusunda oturum açmaya çabaladığında ortaya çıkabilecek kimlik doğrulama hatalarıdır. TIA Portal'da, "Koruma ve güvenlik" elemanı altında bir konfigürasyon iletişim kutusu bulunmaktadır.



OPC UA mesajlarının güvenliği

OPC UA sunucusu, OPC UA'dan sunucu güvenliğiyle ilgili mesajlar oluşturur ve sertifika doğrulaması sırasında standart alarm etki alanını veya denetimleri kullanır.

Aşağıda gösterildiği gibi, TIA Portal özellikleri penceresinde "Dijital güvenlik ilkesi denetimi başarısız oldu" opsiyonunu etkinleştirin veya devre dışı bırakın.



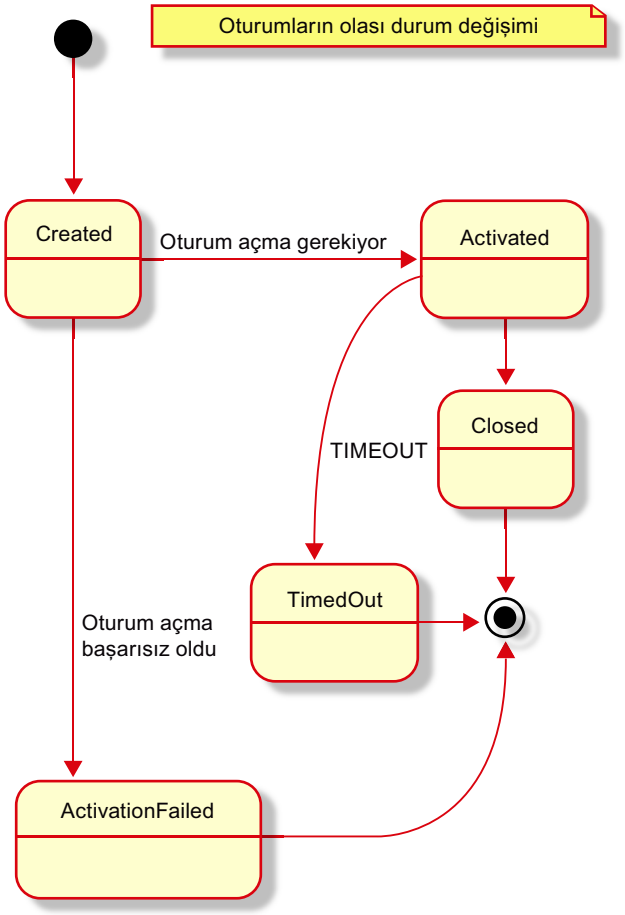
OPC UA sunucusunun bağlantı bilgisi

OPC UA teşhis tamponu ile OPC UA sunucusuna sahip istemcilerin durumunu görüntüleyebilirsiniz. Örneğin teşhis tamponu aşağıdaki bilgileri görüntüler:

- Bir istemcinin OPC UA sunucunuzla ne zaman bağlantı kurduğunu
- İstemcinin OPC UA sunucunuzla bağlantısını ne zaman kestiğini

Bir oturumun durumu değiştiğinde, sunucu teşhis tamponuna bir teşhis mesajı yazar. Oturumların şu durum değişiklikleri mümkündür:

- Created
- Activated
- Closed
- TimedOut
- ActivationFailed



Aşağıdaki kullanım durumları uygulanır:

Tablo 11-77 OPC UA sunucusunda güvenliğin etkinleştirildiği uygulamalı kullanım durumları

Kullanıcı eylemi	Beklenen davranış	Güvenlik direktifleri
OPC UA sunucusunu başlatma (örn. kapatma ve tekrar çalıştırma)	S7-1200 CPU, teşhis tamponuna en düşük güvenlik direktifine (Sayfa 771) sahip bir teşhis mesajı ekler.	Yok, Basic128Rsa15, Basic256, Basic256Sha256,

Tablo 11-78 OPC UA oturumlarının durum değişiklikleri için gerçekleştirilen kullanım durumları

Ön koşul	Kullanıcı eylemi	Beklenen davranış	Durumlar
OPC UA sunucusu çalışıyor, istemci bağlı değil	İstemci, OPC UA sunucusuna bağlanır ve doğru oturum açma verilerini iletir	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	Created, Activated
OPC UA sunucusu çalışıyor, istemci bağlı değil	İstemci, OPC UA sunucusuna bağlanır ve yanlış oturum açma verilerini iletir	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	Created, Activation failed
OPC UA sunucusu çalışıyor, istemci bağlı	İstemci oturumu doğru biçimde kapatır	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	Closed
OPC UA sunucusu çalışıyor, istemci bağlı	İstemci, oturum zaman aşımına uğrayıncaya kadar sunucuya bir mesaj göndermez	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	TimedOut

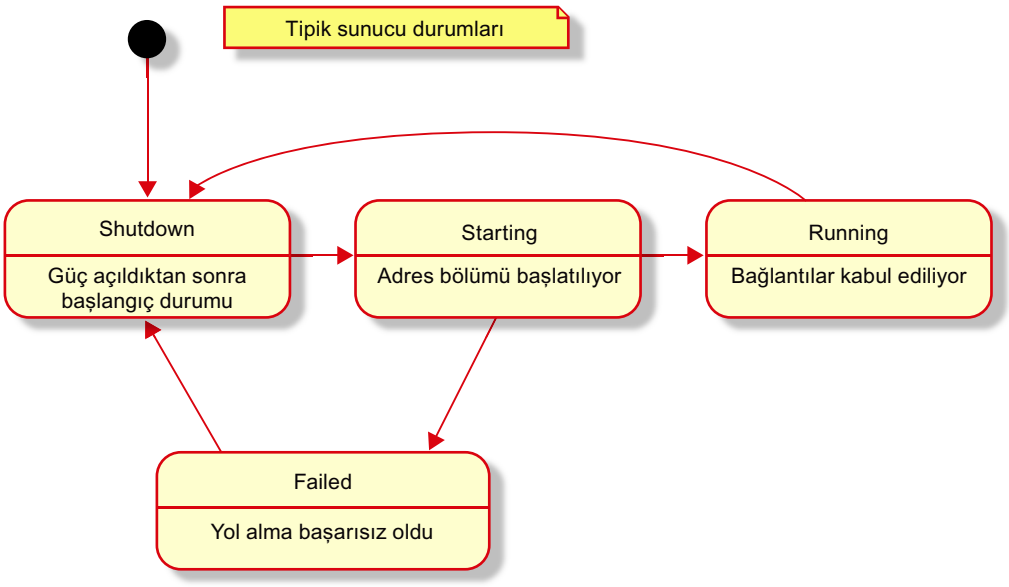
OPC UA sunucusu başlatıldı/durduruldu

OPC UA teşhis tamponu ile OPC UA sunucusunun genel durumunu görüntüleyebilirsiniz. Teşhis tamponundaki mesajlar, OPC UA sunucusunun ne zaman başlayıp durduğunu gösterir. Genel arayüzün açık olup olmadığı veya kaç sunucu arayüzünün veya ad alanının aktif olduğu gibi ek bilgileri de görüntüleyebilirsiniz.

OPC UA sunucusu aşağıdaki durumlardan birinde olabilir:

- Running
- Failed
- NoConfiguration
- Suspended
- Shutdown
- Test
- CommunicationFault
- Unknown
- Starting
- Restarting

Durum değişikliğinin nedenleri: Yükleme/kapatma ve tekrar çalıştırma, kullanıcı programından çağrılan yordam, uzaktan talep. Olası sunucu durumu değişiklikleri şunlardır.



Tablo 11-79 OPC UA oturumlarının durum değişiklikleri için gerçekleştirilen kullanım durumları

Ön koşul	Kullanıcı eylemi	Beklenen davranış	Durumlar
OPC UA sunucusu çalışıyor	Etkinleştirilmiş OPC UA sunucusuyla donanım indirme	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur. Neden: Yükle / kapatma ve tekrar çalıştırma	Shutdown, Starting, Rum.nin
OPC UA sunucusu durduruldu	Etkinleştirilmiş OPC UA sunucusuyla donanım indirme	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	Created, Activated
OPC UA sunucusu çalışıyor	Devre dışı bırakılmış OPC UA sunucusuyla donanım indirme	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	Created, Activation failed
OPC UA sunucusu çalışıyor	Etkinleştirilmiş OPC UA sunucusu ve çok büyük tip sözlüğü (çok fazla yapı) ile donanım indirme	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	Closed
OPC UA sunucusu durduruldu	Etkinleştirilmiş OPC UA sunucusu ve çok büyük tip sözlüğü ile donanım indirme	S7-1200 CPU, durum değişimini gösteren bir teşhis mesajı oluşturur.	TimedOut

NOT

Yazılım indirme

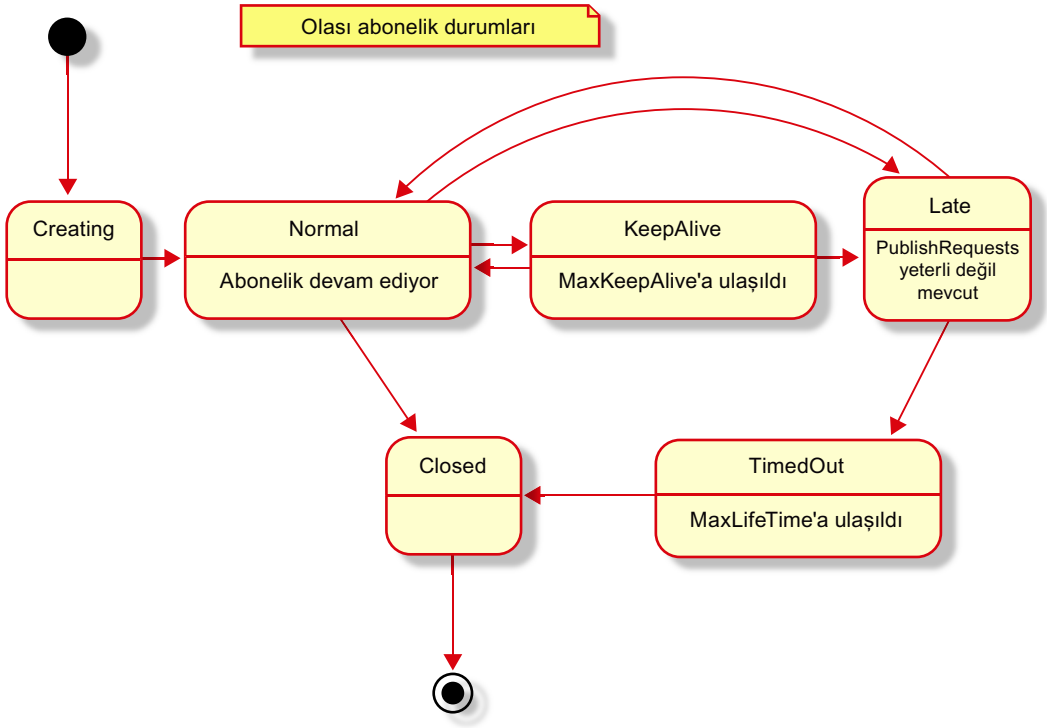
Bir yazılım indirmesi, OPC UA sunucusunun yeniden başlatılmasına yol açar.

OPC UA oturumu/aboneliği zaman aşımı

Bir OPC UA oturumunun veya aboneliğinin ne zaman sona erdiğini ve hangi oturumların veya aboneliklerin hâlâ aktif olduklarını görebilirsiniz.

Bir OPC UA oturumunun veya aboneliğinin durumunu değiştirirseniz, sunucu bu durum değişikliği hakkındaki bilgileri OPC UA teşhis tamponuna yazar. OPC UA aboneliğinin şu durumları mümkündür:

- Created
- Closed
- Normal
- Late
- KeepAlive
- TimedOut



Süregelen bir abonelik sıklıkla "Normal" ve "KeepAlive" durumları arasında geçiş yapar (denetlenen değerler yalnızca ara sıra değişir). "KeepAlive" durumu için herhangi bir mesaj tetiklenmez. Aşağıdaki kullanım durumları uygulanır:

Tablo 11-80 OPC UA sunucusunda güvenliğin etkinleştirildiği uygulamalı kullanım durumları

Kullanıcı eylemi	Beklenen davranış	Güvenlik direktifleri
OPC UA sunucusu başlatıldı (örn. kapatılıp tekrar çalıştırıldı)	S7-1200 CPU, teşhis tamponuna en düşük güvenlik direktifine (Sayfa 771) sahip bir teşhis mesajı ekler.	Yok, Basic128Rsa15, Basic256, Basic256Sha256,

11.10.4.3 Yanlış OPC UA kullanımı

OPC UA sunucusu, "yanlış kullanım" olduğunda sizi bilgilendirir.

Bir istemci amacına uygun olmayan biçimde veri istediğinde veya bir özelliği kullanmaya çabaladığında, "Yanlış kullanım" meydana gelir. Geçersiz bir talep gönderildiğinde, sunucu tarafından bir "Yanlış kullanım" teşhis mesajı görüntülenir.

Geçersiz bir NodeID talep etmek, "Yanlış kullanım" teşhis mesajının görüntülenmesini tetikler. Bir tarayıcı aramasıyla geçersiz bir NodeID'yi algılayabilirsiniz.

"Yanlış kullanım" teşhis mesajı, aşağıdaki olaylar için görüntülenmez:

- Bir istemci isteğe bağlı öznitelikleri okumaya çabalıyor
- Bir istemci, abonelik sayısını aşıyor
- Bir istemci, bir sunucu oturumu için olan limiti aşıyor

Not: "http://opcfoundation.org/UA/" (ns=0) ad alanı, OPC Foundation'ının (veya SDK'nın) özel bir ad alanıdır ve teşhis fonksiyonları çok kısıtlıdır. Bu ad alanındaki her "Yanlış kullanım", bir mesajı tetiklemez (örneğin bilinmeyen bir katılımcının kaydedilmesi gibi).

Sunucu, bir hizmetin yanlış kullanıldığını algıladığında teşhis tamponuna şu teşhis mesajını yazar: "OPC UA sunucusu: <Oturum ID'si> oturum ID'sinde <Hizmet adı> hizmetinin yanlış kullanımı". Yalnızca S7-1200 OPC UA sunucusu tarafından desteklenen hizmetleri dikkate alır:

- FindServers
- GetEndpoints
- FindServersOnNetwork
- CreateSession
- ActivateSession
- CloseSession
- Cancel
- Browse
- BrowseNext
- TranslateBrowsePathsToNodeIds
- RegisterNodes
- UnregisterNodes
- Write
- Read
- Call
- CreateMonitoredItems
- ModifyMonitoredItems
- DeleteMonitoredItems
- SetMonitoringMode
- SetTriggering
- CreateSubscription
- ModifySubscription
- DeleteSubscription
- Publish
- Republish
- SetPublishingMode

- OpenSecureChannel
- CloseSecureChannel

11.10.4.4 OPC UA için özet mesajlar

OPC UA teşhis tamponunda bir "yanlış kullanım" mesajı için bir örnek aşağıda gösterilmiştir:

Tablo 11-81 Hizmetin yanlış kullanımı için OPC UA mesajlarının özeti

Mesaj	Basit olay	Özet olay
Yanlış OPC UA kullanımı		
Örnek	OPC UA sunucusu: Oturum ID'si 12345678'de okunan hizmetin yanlış kullanımı	OPC UA sunucusu: Bir hizmetin yanlış kullanımı. - Özet mesaj. - Son 20 saniye içinde görülen üç oluşum için özet mesaj

11.10.5 OPC UA yöntem çağrıları

Sunucu yöntemleri için kullanıcı programlarının sağlanması

Kullanıcı programı aracılığıyla bir S7-1200 CPU'nun OPC UA sunucusunda OPC UA yöntemleri sağlayabilirsiniz.

OPC UA yöntemleriyle, kontrolör üzerinde spesifik eylemler tetikleyebilir ve verileri tutarlı bir biçimde aktarabilirsiniz.

OPC UA yöntemleri, örneğin S7-1200 yöntem çağrısı üzerinden bir üretim görevi başlatmak için bir OPC UA istemcisini kullanabilir.

"Remote Procedure Calls - Uzaktan prosedür çağrılarının" bir uygulaması olan OPC UA yöntemleri, farklı iletişim katılımcıları arasındaki etkileşimlere yönelik verimli bir mekanizma sunar. Bu mekanizma hem görev onayı hem de geri bildirim değerleri sağlar, dolayısıyla artık el sıkışma (Handshake) mekanizmalarını programlamanıza gerek kalmaz.

Bir OPC UA yöntemi nasıl çalışır?

Bir OPC UA yöntemi, esas itibariyle çalışma zamanı sırasında çağrılan teknik bilgi korumalı bir fonksiyon bloğu gibi çalışır.

OPC UA istemcisi yalnızca tanımlanmış giriş ve çıkışları "görür". Fonksiyon bloğunun içeriği, yöntem veya algoritma, harici OPC UA istemcisinden gizli kalırlar. OPC UA istemcisi, başarılı yürütme ve fonksiyon bloğu (yöntem) tarafından geri döndürülen değerler veya yürütme başarılı değilse bir hata mesajı ile ilgili geri bildirim alır.

Bir programcı olarak, OPC UA yönteminin çalıştırdığı program içeriği üzerinde tam denetime ve sorumluluğa sahipsiniz.

Bir yöntemin ve çalışma zamanı davranışının programlanması için kurallar

- OPC UA yöntemi tarafından geri döndürülen değerlerin, OPC UA istemcisi tarafından sağlanan giriş değerleriyle tutarlı olmalarını sağlayın.
- Parametrelerin adları, yapıları ve izin verilen veri tiplerini atama kurallarına uyun (OPC UA sunucu yordamlarının açıklamasına bakın).
- Çalışma zamanı sırasındaki davranış: OPC UA sunucusu, örnek başına bir çağrı kabul eder. Yöntem örneği, çağrı tamamlanana veya süresi sona erene dek diğer OPC UA istemcileri için kullanılamaz. Sunucuya bağlanmak için izin verilen maksimum süreye ulaşıldığında, örnek zaman aşımına uğrar.

Sunucu yönteminin uygulanması

Bir sunucu yönteminin uygulanması aşağıdaki görevleri içerir:

- Sunucu yöntemi için opsiyonel giriş ve çıkış parametrelerinin ayarlanması
- OPC_UA_ServerMethodPre ile hizmet yöntemi çağrısı sorgulama
- Sunucu yöntemini yazma
- OPC_UA_ServerMethodPost ile sunucu yöntemi ile yanıtlama

Sunucu yöntemi için opsiyonel giriş ve çıkış parametrelerinin ayarlanması

Bir OPC UA yöntemi, opsiyonel giriş veya çıkış parametrelerini belirtebilir. Herhangi bir parametre türü zorunlu değildir. OPC UA istemcisi, çalışma zamanında OPC UA yöntemi için giriş parametrelerini sağlar. OPC UA yöntemi, yöntem tamamlandığında çalışma zamanında OPC UA istemcisi için çıkış parametrelerini geri döndürür.

OPC UA yöntemi için giriş parametrelerini ayarlamak üzere aşağıdaki yöntemi izleyin:

1. FB arayüzünün statik bölümünde UAMethod_InParameters adlı bir yapı belirtin. Bu veri tipi yapısını, "HMI/OPC UA/Web API'den erişilebilir" ve "HMI/OPC UA/Web API'den yazılabilir" olarak işaretleyin.
2. Bu yapıda OPC UA yönteminin giriş parametrelerini belirtin. Giriş parametreleri herhangi bir geçerli ada sahip olabilir. Bir OPC UA yöntemi için giriş parametrelerinin veri tipleri, skaler (Int, Real vb.), yapılandırılmış veri tipleri veya Array'ler olabilir.

OPC UA yöntemi için çıkış parametrelerini ayarlamak üzere aşağıdaki yöntemi izleyin:

1. FB arayüzünün statik bölümünde UAMethod_OutParameters adlı bir yapı belirtin. Bu veri tipi yapısını, "HMI/OPC UA/Web API'sinden erişilebilir" olarak işaretleyin.
2. Bu yapıda OPC UA yönteminin çıkış parametrelerini belirtin. Çıkış parametreleri, herhangi bir geçerli ada sahip olabilir. Bir OPC UA yöntemi için çıkış parametrelerinin veri tipleri, skaler (Int, Real vb.), yapılandırılmış veri tipleri veya Array'ler olabilir.

Aşağıdaki örnek, bir OPC UA yöntemi için kullanıcı tanımlı giriş ve çıkış parametrelerini göstermektedir:

		Name	Data type	Accessible from HMI/OPC UA/Web API	Writable from HMI/OPC UA/Web API
7		Static		☐	☐
8		UAMethod_InParameters	Struct	☑	☑
9		a	Int	☑	☑
10		b	Int	☑	☑
11		UAMethod_OutParameters	Struct	☑	☑
12		SUM	Int	☑	☑

OPC_UA_ServerMethodPre ile sunucu yöntemi çağrısının sorgulanması

Sunucu yönteminizden kullanıcı programınızdaki "OPC_UA_ServerMethodPre" yordamını çağırın.

Bunun ardından yordam, S7-1200 CPU'nun OPC UA sunucusuna, sunucu yönteminin bir OPC UA istemcisinden çağrılıp çağrılmadığını sorar.

Bir OPC UA istemcisi, sunucu yöntemini çağırdığında, sunucu yöntemi, OPC UA istemcisinden tüm giriş parametrelerini alır.

Sunucu yöntemini yazma

OPC_UA_ServerMethodPre ve OPC_UA_ServerMethodPost çağrıları arasındaki sunucu yöntemi alt bölümünde asıl kullanıcı programını sağlarsınız. Diğer tüm kullanıcı programlarıyla aynı opsiyonlara sahipsiniz (örn. diğer fonksiyon bloklarına veya global veri bloklarına erişim gibi). Sunucu yöntemi giriş parametreleri kullanıyorsa, bu parametreler sizin için de kullanılabilir olur. Sunucu yöntemi yalnızca, bir OPC UA istemcisi sunucu yöntemini ve OPC_UA_ServerMethodPre sunucu yöntemini çağırmışsa, ancak o zaman bu alt bölümü çalıştırabilir.

Yöntemin başarıyla yürütülmesinden sonra, sunucu yönteminin çıkış parametrelerini belirleyin (yöntemin bu tür çıkış parametreleri bulunuyorsa).

OPC_UA_ServerMethodPost ile sunucu yöntemine yanıt verilmesi

Sunucu yöntemini sona erdirmek için, "OPC_UA_ServerMethodPost" yordamını çağırın. "OPC_UA_ServerMethodPost" yordamının kullanıcı programının işleyişi hakkında bilgi almak için, parametreleri kullanın.

Kullanıcı programı başarıyla çalıştırılır çalıştırılmaz, OPC UA sunucusu ilgili parametreler tarafından bilgilendirilir. Bunun ardından, OPC UA sunucusu sunucu yönteminin çıkış parametrelerini OPC UA istemcisine gönderir.

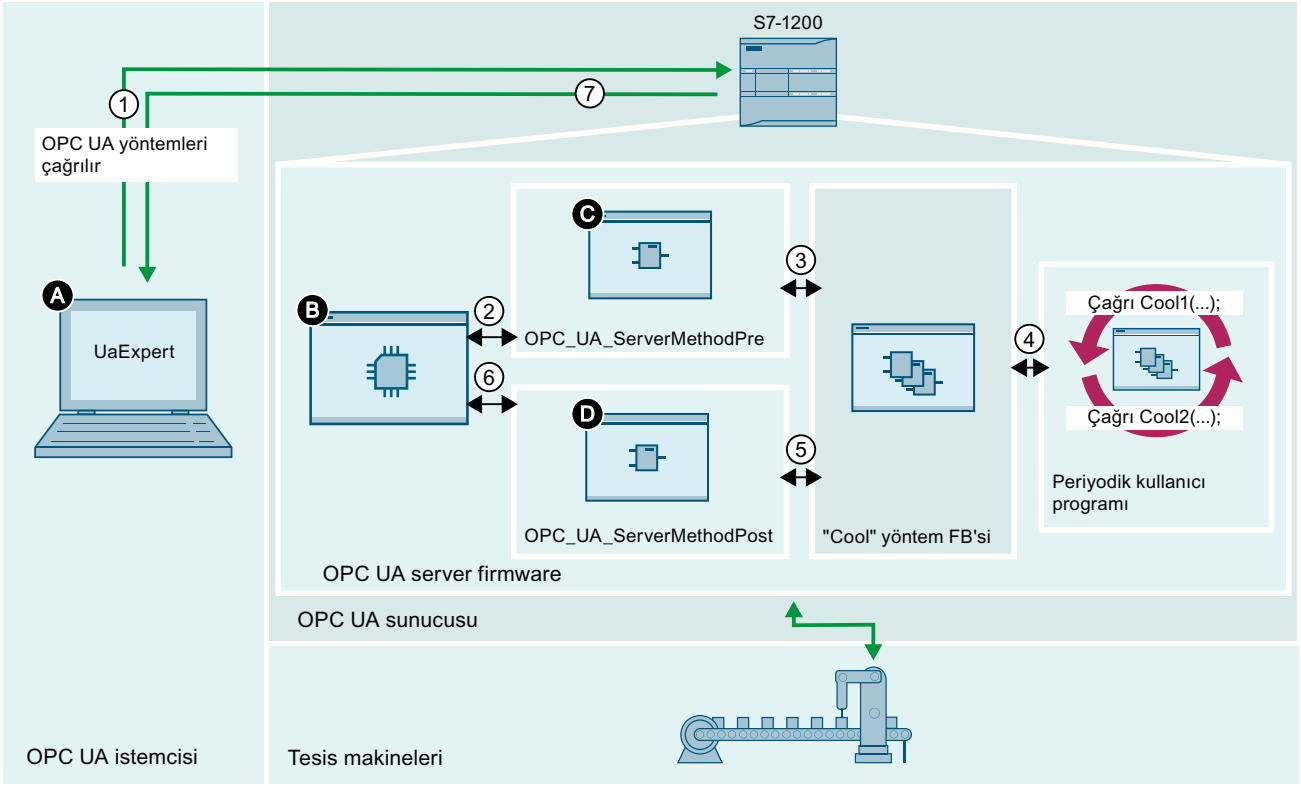
Sunucu yöntemleri hakkında bilgi

Bir OPC UA yöntemi yazarken her zaman "OPC_UA_ServerMethodPre" ve "OPC_UA_ServerMethodPost" çiftlerini kullanın. OPC UA yöntemleri, yalnızca her iki yöntem de eklenirse çalışır.

"OPC_UA_ServerMethodPre" ve "OPC_UA_ServerMethodPost" ile ilgili ayrıntılı bir açıklamayı, TIA Portal'ı bilgi sisteminde bulabilirsiniz.

Sunucu yönteminin entegrasyonu

Aşağıdaki görselde, bir OPC UA istemcisinin "Cool" sunucu yöntemini nasıl çağırdığı gösterilmektedir:



A — OPC UA istemcisi, OPC UA sunucu yöntemini çağırır ve "Tamamlandı" durumunu yönetir.

① — OPC UA istemcisi ile OPC UA sunucu yöntemi arasındaki OPC UA yöntemi çağrıları asenkron olarak gerçekleşir.

B — OPC UA sunucusunun Firmware'i, OPC UA istemcisinden gelen çağrıları bekler, bekleme kuyruğundaki çağrıları yönetir ve "Tamamlandı" bilgisini periyodik kullanıcı programından OPC UA istemcisine iletir.

② — Bu çağrı ile veriler OPC UA sunucusundan kullanıcı programı örneğine, "Cool" yöntemi FB'si ile aktarılmaktadır.

C — OPC_UA_ServerMethodPre yordamı ile CPU'nun OPC UA sunucusuna, OPC UA sunucu yönteminin bir OPC UA istemcisi tarafından çağrılıp çağrılmadığı sorulur. Bir OPC UA istemcisi OPC UA sunucu yöntemini çağırdığında, OPC_UA_ServerMethodPre yordamı, OPC UA sunucu yönteminin o an OPC UA istemcisi tarafından çağrılmakta olduğunu gösteren bir bayrak oluşturur. OPC UA istemcisinden giriş parametreleri varsa, OPC_UA_ServerMethodPre yordamı, bu parametreleri "Cool" yöntem FB'si için kullanılabilir duruma getirir. Kullanıcı programı, "Cool" yöntem FB'si, daha önce OPC_UA_ServerMethodPre yordamını çağırmalıdır.

③ — "Cool" yöntem FB'si, OPC_UA_ServerMethodPre yordamının senkronize çağrısını yürütür. OPC_UA_ServerMethodPre yordamı, OPC UA istemcisinden gelen giriş verilerini saklayan statik bir çoklu örnek değişkenidir. Senkron çağrıdan geri dönüş değeri, bir istemcinin OPC UA sunucu yöntemini çağırıp çağırmadığını gösterir.

④ — Periyodik kullanıcı programı, "Cool" yöntem FB'si gerekli örnek parametreleriyle asenkron olarak çağırır.

⑤ — Bu senkronize çağrıda, OPC UA sunucu yönteminin tamamlandı veya "çalışıyor" şeklindeki durumu denetlenir.

D — OPC_UA_ServerMethodPost yordamı, OPC UA sunucu yöntemi tamamlandıktan sonra yöntem örneğinin çıkış verilerini OPC UA sunucusuna iletir. Bundan başka, OPC_UA_ServerMethodPost yordamı, yöntem örneğine ve OPC UA sunucusuna, yöntemin sona erdiğini bildirir.

⑥ — Bu çağrı ile veriler, kullanıcı programı örneğinden, "Cool" yöntem FB'si verileri OPC UA sunucusuna aktarılır.

⑦ — OPC UA sunucusunun Firmware'i, bilgileri OPC UA istemcisine geri gönderir.

Bu örnekteki çalışma şekli

CPU, periyodik kullanıcı programında ④, "Cool" sunucu yöntem FB'si "Cool1" örneğini yürütür. "Cool1" örneği, bir OPC UA istemcisinin "Cool" yordamını çağırıp çağırmadığını ③ birimine sormak için, "OPC_UA_ServerMethodPre" sunucu yöntem FB'sini çağırır ①.

- "Cool" sunucu yöntem FB'si, OPC_UA_ServerMethodPre yordamını çağırmadıysa, program yürütme, doğrudan ③ ve ④ üzerinden periyodik kullanıcı programına geri döndürülür. "Cool1" yordamından sonra, CPU periyodik kullanıcı programına devam eder.
- Ancak "Cool" sunucu yöntem FB'si daha önce OPC_UA_ServerMethodPre'yi çağırmışsa, OPC_UA_ServerMethodPre, bilgileri doğrudan ③ üzerinden "Cool" sunucu yöntem FB'sine geri döndürür. Şimdi, "Cool" yöntem FB'si yürütülür ve tesis makinelerinin verilerine erişim sağlar.

OPC UA sunucu yöntemi tamamlandıktan sonra, "Cool" sunucu yöntem FB'si, "OPC_UA_ServerMethodPost" yordamını çağırarak ⑤, Firmware'e (B) yordamın ⑥ yürütüldüğünü bildirir. Firmware, bu bilgiyi ⑦ üzerinden çağrıyı yapan OPC UA istemcisine (A) geri gönderir. "Cool1" yordamından sonra, CPU periyodik kullanıcı programına devam eder.

11.10.5.1 Sunucu yöntemlerini kullanmak için sınır koşulları

Sunucu yöntemlerini sağlarken, veri tiplerini aşağıda gösterildiği gibi eşleyin (SIMATIC veri tipi - OPC UA veri tipi). Olağandan farklı atamalar kullanmayın.
STEP 7, yanlış atamaları denetlemez. Veri tiplerinin doğru seçilmesinden ve atanmasından siz sorumlusunuz.
Listelenen veri tiplerini, örneğin kendi oluşturduğunuz sunucu yöntemlerinin (UAMethod_InParameters ve UAMethod_OutParameters) giriş ve çıkış parametreleri için yapıların/Array'lerin/UDT'lerin elemanları olarak da kullanabilirsiniz.

SIMATIC veri tipi	OPC UA veri tipi
BOOL	Boole değeri
SINT	SByte
INT	INT16
DINT	INT32
USINT	Bayt
UINT	UINT16
UDINT	UINT32
REAL	Float
LREAL	Double
WSTRING	String
DINT	Sayım listesi (Int32 şifreleme) ve tüm türetilmiş veri tipleri

12

Web sunucusu

S7-1200'ün Web sunucusu, CPU'nuza ilişkin verilere ve proses verilerine web sitesi erişimi sağlar.

S7-1200 web sitelerine, PC veya mobil cihaz üzerinden erişebilirsiniz. Web sunucusu, küçük ekranlı cihazları bir Temel siteler (Sayfa 809) birikimi ile destekler.

Bağlantıyı kurmak için, S7-1200 CPU'nun IP adresine veya yerel modül taşıyıcısındaki Web sunucusu üzerinden etkinleştirilmiş CP (İletişim işlemcisi) modülünün (Sayfa 805) IP adresine erişmek için bir web tarayıcısını kullanın. S7-1200, aynı anda çok sayıda bağlantıyı destekler.



NOT

Web sunucusuna aynı anda birden fazla bağlantının olması

S7-1200 Web sunucusu, aynı anda 30 adet bağlantıya izin verir (yeterli dinamik bağlantınızın olduğu varsayılmaktadır). Açık tarayıcı örneklerinin her biri, 2 ile 8 arasında bağlantı kullanabilir. Web sunucusu, yedi adede kadar kullanıcının oturum açmasına izin verir, fakat Siemens, aynı anda oturum açmış kullanıcı sayısını mümkünse minimumda tutmanızı önermektedir. Orta kapasitede kullanımda, ortalama 7 eş zamanlı kullanıcı genellikle iyidir.

Standart web siteleri

S7-1200, PC'nizdeki web tarayıcısı (Sayfa 802) veya bir mobil cihaz (Sayfa 804) üzerinden erişebileceğiniz standart web sitelerini (Sayfa 808) içerir:

- Giriş (Sayfa 813) - Standart web sitelerine giriş
- Başlangıç sayfası (Sayfa 813) - CPU hakkında genel bilgiler
- Teşhis (Sayfa 814) - Seri, sipariş ve sürüm numarası, program koruması ve bellek kullanımı da dahil olmak üzere CPU ile ilgili ayrıntılı bilgiler
- Teşhis tamponu (Sayfa 817) – teşhis tamponu
- Modül bilgileri (Sayfa 819)– Yerel ve desantral modüllere ilişkin bilgiler ve modüller için Firmware güncellemeleri olanağı
- İletişim (Sayfa 823) - Ağ adreslerine ilişkin bilgiler, iletişim arayüzlerinin fiziksel özellikleri, istatistikler, parametrelerin yanı sıra bağlantıya genel bakış ve teşhis bilgileri
- Değişken durumu (Sayfa 827) - CPU değişkenleri ve G/Ç, adresler veya PLC değişken adları üzerinden erişilebilir

- İzleme tabloları (Sayfa 829) - STEP 7'de yapılandırdığınız tabloları izlenmesi
- Çevrimiçi yedekleme (Sayfa 830) - Çevrimiçi bir CPU'nun yedeklenmesi veya önceki bir yedeklemeyi geri yükleme olanağı
- Veri protokolleri (Sayfa 833) - PLC'deki tüm veri protokollerinin bir listesini görüntüleme, PLC'den bilgisayarınıza veri protokolünü yükleme, PLC'deki bir veri protokolünü silme ve PLC'den bir veri protokolünü çağırma ve silme olanağı
- Kullanıcı dosyaları (Sayfa 835) - PLC'deki kullanıcı dosyalarının bir listesini görüntüleme, PLC'den bilgisayarınıza bir kullanıcı dosyası yükleme, bilgisayarınızdan PLC'ye bir kullanıcı dosyası yükleme ve PLC'nizdeki bir kullanıcı dosyasını silme olanağı
- Kullanıcı tanımlı web siteleri (Sayfa 840) - CPU verilerine erişmek için kullanıcı tanımlı web sayfalarının oluşturulması
- Dosya tarayıcısı (Sayfa 838) - Veri protokolleri veya reçeteler gibi CPU'da veya memory card'da saklanan dosyalar için tarayıcı
- Oturum açma (Sayfa 809) - Farklı bir kullanıcı olarak oturum açılması veya oturumun kapatılması.

Bu sayfalar S7-1200 CPU'ya entegre edilmiş olup, İngilizce, Almanca, Fransızca, İspanyolca, İtalyanca ve Basitleştirilmiş Çince dillerinde sunulmaktadır. PLC teşhis mesajlarını görüntülemek için TIA Portal'da daha fazla konfigürasyon gereklidir (Bölüm 15). Giriş ve başlangıç sayfaları hariç, diğer tüm sayfaları görüntülemek için STEP 7'de yapılandırdığınız ek kullanıcı yetkileri (Sayfa 801) gereklidir.

Kullanıcı tanımlı web siteleri

S7-1200 ile CPU verilerine erişebilen kullanıcı tanımlı web siteleri de oluşturabilirsiniz. Bu siteleri sizin seçebileceğiniz HTML kodları yazılımı ile oluşturabilir ve CPU verilerine erişmek için HTML kodunuza önceden tanımlanmış "AWP" (Otomasyon web programlama) komutları ekleyebilirsiniz. Kullanıcı tanımlı web siteleri (Sayfa 840) bölümünde, kullanıcı tanımlı web sitelerinin geliştirilmesi ve STEP 7'deki ilgili konfigürasyon ve programlama hakkında kullanıcı tanımlı bilgiler bulabilirsiniz.

Standart veya temel web sitelerine çağırarak bir bilgisayardan veya bir mobil cihazdan kullanıcı tanımlı web sitelerine erişebilirsiniz. Kullanıcı tanımlı web sitelerinizden birini, Web sunucusu için ana sayfa (Sayfa 859) olarak da yapılandırabilirsiniz.

Web API'si

Ayrıca S7 1200 CPU, proses verilerini okumak ve yazmak için arayüz görevi gören bir Web-API (Sayfa 883) de sunar.

Web tarayıcı gereklilikleri

Siemens, Web sunucusunun standart sitelerini test etmiş ve aşağıdaki web tarayıcılarının desteğini incelemektedir.

- Internet Explorer 11
- Microsoft Edge Chromium Based V103
- Mozilla Firefox V102
- Opera V89
- Google Chrome V103
- Android Pie V9 için Android Browser
- Android Pie V9 için Mobile Chrome
- iOS V13 için Mobile Safari ve Chrome

Bir WinCC projesinde HTML tarayıcı kullanımında, Web sunucusu standart siteler için aşağıdaki Siemens HMI panellerini destekler:

- Basic Panels
  - Gen 2 KTP400 - KTP1200
- Comfort Panels
  - TP700 - TP2200
  - KP400 - KP1500
  - KTP400
  - TP700 Comfort Outdoor
- Mobile Panels
  - Gen 2 KTP700[F], KTP900[F]
- Unified Comfort Panels
  - Basic Panel (2. nesil)
  - Mobile Panel (2. nesil)

Standart veya kullanıcı web sitelerinin görüntülenmesini etkileyebilecek tarayıcıya özgü sınırlamalar için Sınırlamalar (Sayfa 885) alt bölümüne bakın.

Web sunucusunun işletim davranışı

Web sunucusunun işletim davranışı, çok sayıda faktörden etkilenebilir. S7-1200 CPU ve programlama cihazı, zamanı, kaynakları ve işleme süresini tüketen diğer görevlerle paylaşmalıdır. Web sunucunuz düşük işletim performansı sergiliyorsa, Web sunucusu performansını iyileştirmek için şu ayarları deneyin:

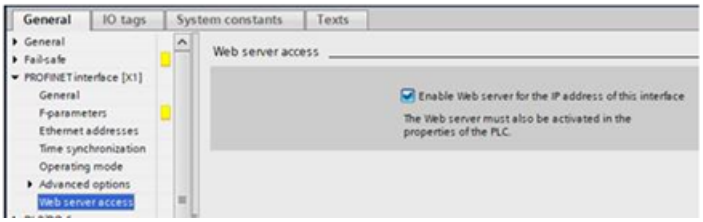
- PLC üzerindeki İletişim yükünü (Sayfa 84) % 20'den % 50'ye çıkarın.
- Bir minimum çevrim süresi (Sayfa 84) yapılandırın. Minimum çevrim süresinin belirlenmesi, S7-1200 CPU ile programlama cihazı arasındaki iletişim süresini uzatır.
- TIA Portal'daki "Güncelleme aralığı" süresini standart olarak 10 saniyenin üzerine çıkarın.
- Web sunucusuna erişim için CP modülü (Sayfa 805) yerine S7-1200 CPU'nun Ethernet arayüzünü kullanın.

12.1 Web sunucusunu etkinleştirilmesi

Web sunucusunu, STEP 7'de bağlanmak istediğiniz CPU'nun cihaz konfigürasyonu üzerinden etkinleştirirsiniz.

Web sunucusunu etkinleştirmek için aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda CPU'yu seçin.
2. Denetçi penceresindeki CPU özellikleri altında "Web sunucusu" öğesini seçin.
3. "Bu cihazın tüm modüllerinde Web sunucusunu etkinleştir" opsiyon kutusunu işaretleyin.
4. İyileştirilmiş güvenlik için, "Yalnızca HTTPS erişimine izin ver" seçerek, TLS (Sayfa 594) üzerinden Web sunucusuna güvenli erişimi zorunlu kılın. Güvenli iletişimi sağlamak üzere HTTPS kullanmak için, bir Web sunucusu sertifikası yapılandırılmalıdır. CPU için cihaz konfigürasyonunda Web sunucusu sertifikalarını yapılandırın. Web sunucusu sertifikalarını yapılandırmak için, CPU özelliklerinde "Web sunucusu > Dijital güvenlik" öğesini seçin.
5. "Otomatik güncelleme" için "Otomatik güncellemeyi etkinleştir" opsiyonunu seçerseniz, standart web siteleri her on saniyede bir yenilenir. Aksi takdirde "Güncelleme aralığı" alanında kullanıcıya özgü bir güncelleme süresi de belirleyebilirsiniz.
6. Kullanılan arayüz erişiminin türüne bağlı olarak, ilgili arayüz için Web sunucusu erişiminin etkinleştirilmesini sağlamalısınız:
   - PLC PROFINET: Web sunucusuna PLC PROFINET arayüzü üzerinden erişilebilmesi için "Bu arayüzün IP adresi için Web sunucusunu etkinleştir" özelliğini PLC'de etkinleştirmelisiniz.
   - WanCP PROFINET portu: Bir WanCP cihazı kullanıyorsanız, özel WanCP PROFINET portu üzerinden Web sunucusuna erişilebilmesi için, "Bu arayüzün IP adresi için Web sunucusunu etkinleştir" özelliğinin etkinleştirilmesini sağlamalısınız.



⚠ İKAZ

Web sunucusu üzerinden CPU'ya yetkisiz erişim

Tam CPU erişim yetkilerine veya hata güvenli özelliği de dahil tam erişim olanağına sahip kullanıcılar PLC değişkenlerini okuyabilir ve yazabilir. Web sunucusu kullanıcıları, CPU için kullanılan erişim kademesinden bağımsız olarak PLC değişkenlerini okuma ve yazma yetkilerine sahip olabilir. CPU'ya yetkisi olmayan kişiler tarafından erişim veya PLC değişkenleri için geçersiz değerlerin ayarlanması, proses işletimine zarar verebilir ve hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yetkili kullanıcılar, işletim durumunda değişiklikler yapabilir, PLC verilerini yazabilir ve Firmware güncellemeleri uygulayabilir. Siemens, aşağıdaki güvenlik önlemlerine uyulmasını önerir:

- CPU erişim kademelerini (Sayfa 156) ve Web sunucusu kullanıcı ID'lerini (Sayfa 801) STEP 7'de tanımlandığı şekilde güçlü şifreler aracılığıyla koruyun.
- Web sunucusuna erişimi sadece HTTPS protokolü üzerinden etkinleştirin.
- Web sunucusu kullanıcısının "Herkes" şeklindeki standart asgari yetkilerini genişletmeyin.
- Web sitelerinin kullanıcıları PLC değişkenleri için geçersiz değerler belirleyebileceği için, program lojiğinizdeki değişkenler için bir hata kontrolü ve bir aralık kontrolü gerçekleştirin.
- Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın.

Cihaz konfigürasyonunu yükledikten sonra, standart web siteleri üzerinden CPU'nun tanıtım ve başlangıç sayfasına erişebilirsiniz. Diğer sayfalara erişmek için bir veya daha fazla Web sunucusu kullanıcısı (Sayfa 801) oluşturmalısınız.

Kullanıcı tanımlı web sitelerini (Sayfa 840) oluşturup etkinleştirdikten sonra, bunlara standart veya temel web sitelerinin gezinme menüsü üzerinden erişebilirsiniz.

NOT

Cihaz değişimi: CPU V3.0'ın CPU V4.x ile değişimi

Mevcut bir CPU V3.0'ı bir CPU V4.x ile değiştirir (Sayfa 1365) ve V3.0 projenizi bir V4.x projesine dönüştürürseniz, STEP 7 ve CPU V4.x'in aşağıdaki Web sunucusu ayarlarını koruduğunu dikkate almalısınız:

- "Bu cihazın tüm modüllerinde Web sunucusunu etkinleştir"
- "Yalnızca HTTPS erişimine izin ver"

NOT

"RUN işletim durumunda yükle (Sayfa 1139)" işlemi yürütüldüğünde, standart ve kullanıcı tanımlı web sitelerindeki veri değerleri güncellenmez ve yükleme tamamlanana dek veri değerlerini tekrar yazamazsınız. Web sunucusu, yükleme işlemi sırasında veri değerleri yazma girişimlerini iptal eder.

12.2 Web sunucusu kullanıcılarının yapılandırılması

Web sunucusu üzerinden CPU'ya erişebilen farklı yetki kademeleri olan kullanıcılar yapılandırabilirsiniz.

Web sunucusu kullanıcılarını ve ilgili yetkileri yapılandırmak için aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda CPU'yu seçin.
2. Denetçi penceresinde, CPU özellikleri altında "Web sunucusu" öğesini seçin ve Web sunucusunu etkinleştirin (Sayfa 798).
3. Web sunucusunun özelliklerinde "Kullanıcı yönetimi" öğesini seçin.
4. Sağlanacak kullanıcı oturumları için kullanıcı adlarını, erişim kademelerini ve şifreleri girin.

Konfigürasyonu CPU'ya yükledikten sonra, yalnızca yetkili kullanıcıların uygun yetkilere sahip oldukları Web sunucusu fonksiyonlarına erişebilirsiniz.

Web sunucusu için erişim kademeleri

STEP 7, standart kullanıcıya şifresiz "Herkes" özelliğini sunar. Standart olarak, bu kullanıcının ek yetkileri yoktur ve yalnızca Başlangıç (Sayfa 813) ve Giriş (Sayfa 813) standart web sitelerini açabilir. Ancak, diğer tüm kullanıcılardaki gibi, "Herkes" kullanıcısı için de ek yetkiler yapılandırabilirsiniz:

- Teşhis çağır
- Değişkenleri oku
- Değişkenleri yaz
- Değişken durumunu oku
- Değişken durumunu yaz
- Kullanıcı tanımlı web sitelerini aç
- Kullanıcı tanımlı Web sitelerine yaz
- Dosyaları oku
- Dosyaları yaz/sil
- İşletim durumu değiştir
- LED'lerin yanıp sönmesini sağla
- Firmware güncellemesini gerçekleştir
- CPU'nun bir yedek kopyasını oluştur
- CPU'yu geri yükle
- Sistem parametrelerini değiştir
- Uygulama parametrelerini değiştir

Web sunucusu için ana sayfa olarak kullanıcı tanımlı bir web sitesini (Sayfa 859) belirlediyseniz, "Herkes" kullanıcısı, "kullanıcı tanımlı web sitelerini aç" yetkisine gerek duyar.

⚠ İKAZ

Web sunucusu üzerinden CPU'ya yetkisiz erişim

Tam CPU erişim yetkilerine veya hata güvenli özelliği de dahil tam erişim olanağına sahip kullanıcılar PLC değişkenlerini okuyabilir ve yazabilir. Web sunucusu kullanıcıları, CPU için kullanılan erişim kademesinden bağımsız olarak PLC değişkenlerini okuma ve yazma yetkilerine sahip olabilir. CPU'ya yetkisi olmayan kişiler tarafından erişim veya PLC değişkenleri için geçersiz değerlerin ayarlanması, proses işletimine zarar verebilir ve hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yetkili kullanıcılar, işletim durumunda değişiklikler yapabilir, PLC verilerini yazabilir ve Firmware güncellemeleri uygulayabilir. Siemens, aşağıdaki güvenlik önlemlerine uyulmasını önerir:

- CPU erişim kademelerini (Sayfa 156) ve Web sunucusu kullanıcı ID'lerini STEP 7'de tanımlandığı şekilde güçlü şifreler aracılığıyla koruyun.
- Web sunucusuna erişimi sadece HTTPS protokolü üzerinden etkinleştirin.
- Web sunucusu kullanıcısının "Herkes" şeklindeki standart asgari yetkilerini genişletmeyin.
- Web sitelerinin kullanıcıları PLC değişkenleri için geçersiz değerler belirleyebileceği için, program lojiğinizdeki değişkenler için bir hata kontrolü ve bir aralık kontrolü gerçekleştirin.
- Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın.

NOT

Cihaz değişimi için şifrelemeyi V4.x'e güncelleyin

V4.x'e sürümü için yapılmış bir cihaz değişiminden sonra, Web sunucusu için kullanıcı şifrelemesini güncellemeniz gerekir. TIA Portal'daki CPU'nun cihaz konfigürasyonunda, Web sunucusu kullanıcı yönetiminde "Şifrelemeyi güncelle" butonuna tıklayın.

12.3 PC üzerinden web sitelerine erişim

Bir PC veya mobil cihaz ile S7-1200 standart web sitelerine, S7-1200 CPU'nun IP adresi veya yerel modül taşıyıcısında Web sunucusu etkinleştirilmiş CP (Sayfa 805)'nin IP adresi üzerinden erişebilirsiniz.

S7-1200 standart web sitelerine PC üzerinden erişmek için aşağıdaki yöntemi izleyin:

1. S7-1200 ve PC'nin ortaklaşa kullanılan Ethernet ağında olduğundan veya olağan bir Ethernet kablosu kullanılarak doğrudan birbirine bağlı olduğundan emin olun.
2. Bir web tarayıcı açın ve "https://ww.xx.yy.zz" URL'sini girin. Bu sırada, "ww.xx.yy.zz", S7-1200 CPU'nun IP adresine veya yerel modül taşıyıcısındaki bir CP'nin IP adresine karşılık gelir.

Web tarayıcı, Standart "Giriş" web sitesini (Sayfa 813) veya ana sayfa (Sayfa 859) olarak yapılandırdıysanız, kullanıcı tanımlı web sitelerinizin standart HTML sayfasını açar.

NOT

Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın. Bundan başka, web ortamınız veya işletim sisteminiz tarafından uygulanabilecek tüm sınırlamaları (Sayfa 885) dikkate alın.

Sayfanın URL'sini girerek standart web sitelerine erişim

Sayfanın URL'sini kullanarak bir standart web sitesine erişebilirsiniz. Bunun için URL'yi "https://ww.xx.yy.zz/<Sayfa>.html" formatında girin; bu bağlamda "ww.xx.yy.zz" S7-1200 CPU'nun IP adresi veya yerel modül taşıyıcısındaki bir CP'nin IP adresini temsil etmektedir.

- https://ww.xx.yy.zz/login.html - O an için oturum açmış kullanıcı yoksa Oturum açma (Sayfa 809) sayfası; aksi takdirde boş sayfa görüntülenir.
- https://ww.xx.yy.zz/start.html - CPU hakkında genel bilgiler içeren Başlangıç sayfası (Sayfa 813)
- https://ww.xx.yy.zz/identification.html - Tanımlama (Sayfa 814) sayfası: Seriler, sipariş ve sürüm numaraları da dahil olmak üzere CPU hakkında ayrıntılı bilgi içerir. Teşhis sayfası ise
- https://ww.xx.yy.zz/diagnostic.html – Teşhis tamponu (Sayfa 817)
- https://ww.xx.yy.zz/module.html - Yerel modül taşıyıcısındaki modüller ve Firmware güncellemesi olanakları (Sayfa 819) ile ilgili bilgiler
- https://ww.xx.yy.zz/communication.html - Ağ adresleri, iletişim arayüzlerinin fiziksel özellikleri ve iletişim istatistiklerine yönelik İletişim bilgileri (Sayfa 823)
- https://ww.xx.yy.zz/variable.html - CPU değişkenleri ve G/Ç (Sayfa 827), adresler, PLC değişken adları veya veri bloğu değişken adları üzerinden erişilebilir
- https://ww.xx.yy.zz/watch.html - İzleme tabloları (Sayfa 829)
- https://ww.xx.yy.zz/datalogs.html - Veri protokolleri (Sayfa 833) yükleme, silme veya çağırıp silme
- https://ww.xx.yy.zz/userfiles.html - Kullanıcı dosyaları, SIMATIC memory card'daki (harici yükleme belleği) dosyalara erişim sağlar
- https://ww.xx.yy.zz/filebrowser.html- CPU'da veya hafıza kartında saklanmakta olan veri protokolü veya reçete dosyalarına erişim için tarayıcı (Sayfa 838)
- https://ww.xx.yy.zz/index.html - -Standart web sitelerinde başlangıç için Giriş sayfası (Sayfa 813)

Örneğin "https://ww.xx.yy.zz/communication.html" yazdığınızda, tarayıcınız size iletişim sitesini görüntüleyecektir.

NOT

Yukarıda özel olarak listelenmeyen herhangi bir standart web sitesinin (örneğin "Çevrimiçi yedekleme" sitesi (Sayfa 830)) doğrudan erişim için bir URL sağlamadığını dikkate alın.

Güvenli erişim

Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın. Standart web sitelerine güvenli erişim (Sayfa 798) için http:// yerine https:// kullanın ve bu erişimi gerekli olarak belirleyin. S7-1200'e https:// üzerinden bağlandığınızda, web sitesi oturumu bir dijital sertifika ile şifreler. Web sunucusu, ilgili verileri hiç kimse tarafından görülmeyecek biçimde güvenli bir bağlantı üzerinden aktarır. Standart web sitelerine erişirken, genellikle "Evet" ile onaylamanız gereken bir güvenlik uyarısı görüntülenir. Her güvenli erişim için güvenlik uyarısının görüntülenmesini önlemek isterseniz, Siemens yazılım sertifikasını web tarayıcınıza içe aktarabilirsiniz (Sayfa 806).

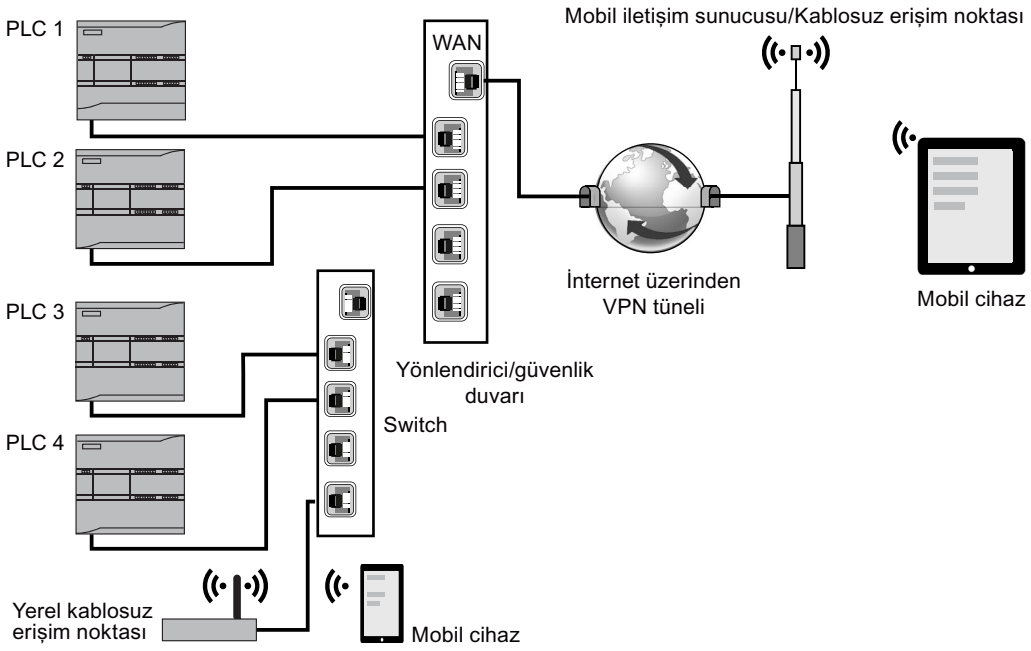
Ayrıca bakınız

Kullanıcı dosyaları (Sayfa 835)

12.4 Web sitelerine bir mobil cihaz üzerinden erişilmesi

S7-1200'e bir mobil cihazdan erişmek için, PLC'nizi internet bağlantılı bir ağa veya yerel bir kablosuz erişim noktasına bağlamanız gerekecektir. Bir mobil cihazı S7-1200 PLC Web sunucusuna bağlarken güvenli bir sanal özel ağ (VPN) kullanmalısınız. Kablosuz yönlendiricideki port yönlendirmenin yardımıyla, PLC'nin IP adresini, bir mobil cihazın PLC'ye internet üzerinden erişebileceği bir adres olarak betimleyebilirsiniz. Port yönlendirmesini ayarlamak için, lütfen yönlendiricinizin yazılım konfigürasyonu için olan yordamlarını dikkate alın. Yönlendiricinizin desteklediği kadar sayıda PLC ve şalt cihazı bağlayabilirsiniz.

Port yönlendirmesi olmadan bir PLC'ye bağlantı, yalnızca yerel olarak kablosuz ağ sinyalinin kapsama alanı içinde mümkündür.



Bu örnekte, yerel kablosuz erişim noktasının kapsama alanındaki bir mobil cihaz, IP adreslerini kullanarak PLC 3 ve PLC 4'e bağlanabilmektedir. Yerel kablosuz ağ sinyali kapsama alanının dışındaki bir mobil cihaz, ilgili PLC'nin iletilen adresini kullanarak internet üzerinden PLC 1 ve PLC 2'ye bağlanabilir.

Standart web sitelerine erişmek için bir mobil iletişim hizmetine veya bir WLAN erişim noktasına erişmeniz gerekir. İnternet üzerinden bir PLC'ye erişim için, mobil cihazınızın web tarayıcısına PLC'ye erişim için ilgili adresi girin, örneğin http://ww.xx.yy.zz:pppp veya https://ww.xx.yy.zz:pppp; burada ww.xx.yy.zz yönlendiricinin adresi ve pppp ise söz konusu PLC için olan port adresidir.

Yerel bir kablosuz erişim noktası aracılığıyla yerel erişim için, yerel modül taşıyıcısına S7-1200 CPU'nun veya Web sunucusu üzerinden etkinleştirilmiş CP (Sayfa 805)'nin IP adresini girin:

- Standart web sitelerine (Sayfa 808) erişim için http://ww.xx.yy.zz veya https://ww.xx.yy.zz
- Temel web sitelerine (Sayfa 809) erişim için: http://ww.xx.yy.zz/basic veya https://ww.xx.yy.zz/basic

Web sunucusunu yalnızca güvenli erişime (HTTPS) (Sayfa 798) izin verecek şekilde yapılandırarak güvenliğin artmasını sağlayın.

12.5 Web sitelerine erişim için bir CP modülünün kullanımı

Web sunucusuna bir PC veya bir mobil cihaz üzerinden erişilmesinden bağımsız olarak, standart web sitelerinin aşağıda sıralanan CP modüllerinden biri üzerinden açılması ancak, söz konusu cihazın STEP 7 'de yapılandırılmış ve S7-1200 CPU ile yerel modül taşıyıcısında kurulu olmasıyla mümkündür:

- CP 1243-1
- CP 1243-7 LTE-EU
- CP 1243-7 LTE-US
- CP 1243-8 IRC

Bu CP modülleri üzerinden web sitelerine erişmek için Standart "Başlangıç" web sitesi (Sayfa 813) kullanılır. Başlangıç sayfası, yerel modül taşıyıcısındaki tüm yapılandırılmış ve kurulu CP modüllerini gösterir, ancak web sitelerine yalnızca yukarıda belirtilen modüller üzerinden erişilebilir.

NOT

Yerel modül taşıyıcısında Web sunucusu üzerinden etkinleştirilen CP'ler varsa standart web sitelerine erişim

Yerel modül taşıyıcısında Web sunucusu etkinleştirilmiş CP'ler takılı ise S7-1200 standart web sitelerine bağlantı bir veya iki dakikaya kadar geciktirilebilir. Siteler ilk önce kullanılamıyor gibi görünüyorsa veya hata mesajları alıyorsanız, bir veya iki dakika bekleyin ve sitelere tekrar erişmeyi deneyin.

12.6 Bir güvenlik sertifikasının indirilmesi ve kurulması

İnternet opsiyonlarında bir güvenlik sertifikası indirebilirsiniz.

Bu sertifikayı sayesinde, https://ww.xx.yy.zz (burada "ww.xx.yy.zz" cihazın IP adresidir) adresini açarken web tarayıcınızdaki güvenlik sorgulamasını yanıtlamak zorunda kalmazsınız. Bir https:// URL'si değil de bir http:// adresi girdiğinizde, sertifikayı indirip yüklemeniz gerekmez.

Bir S7-1200 CPU'nun cihaz konfigürasyonunda, CPU için sertifikalar oluşturabilirsiniz. Bu fonksiyon, cihaz için genel ayarlar, "Koruma ve dijital güvenlik > Sertifika yöneticisi" altında bulunur.

Web sunucusu sertifikalarını, CPU'nun cihaz konfigürasyonunda "Web sunucusu > Dijital güvenlik" üzerinden yapılandırabilirsiniz. Web sunucusu sertifikalarında aşağıdaki olanaklara sahipsiniz:

- Donanım tarafından oluşturulur
- Yazılım tarafından yüklenir; bu ise size kendi özel sertifikanızı kullanmanızı sağlar

Sertifikalar ile ilgili daha fazla bilgiyi, STEP 7 bilgi sisteminde ve bu dokümandaki AUTOHOTSPOT altında bulabilirsiniz.

NOT

S7-1200 için sınırlı sayıda sertifika

S7-1200, sistem tarafından belirlenen 64 sertifika sınırı söz konusudur.

Bu sayıya, tüm sertifikalar dahildir (örn. Web sunucusu sertifikaları, OPC UA sertifikaları ve OUC sertifikaları).

Web sunucusu bir Sertifikasyon Kurumu (CA) sertifikası imzası taşıyorsa ve TIA Portal'da mevcut ise, bu durumda Web sunucusu tarafından iki sertifika kullanılır (biri, Web sunucusu sertifikası için ve diğeri, indirilen CA sertifikası için).

64'ten fazla sertifikanız varsa, TIA Portal, maksimum 64 sertifika sayısını aştığınızı bir hata mesajı ile görüntüler. Bunun ardından bazı sertifikaları PLC konfigürasyonundan kaldırmanız gerekir.

Sertifikanın indirilmesi

Siemens güvenlik sertifikasını PC'nize indirmek için, Giriş sayfasındaki (Sayfa 813) "Sertifikayı indir" linkine tıklayın. İndirme ve içe aktarma işlemleri, kullanılan web tarayıcısına göre değişmektedir. Kendinden imzalı sertifikalar kullanılırken "Sertifikayı indir" bağlantısı aktif değildir.

Sertifikanın içe aktarılması ve yüklenmesi

Siemens sertifikasının içe aktarım ve yüklenmesinde web tarayıcınızın kurallarına uyun. "S7-1200 Controller Family" Siemens güvenlik sertifikasını, web tarayıcınızın içeriği için kullanılan internet opsiyonlarına yükledikten sonra, https:// ww.xx.yy.zz üzerinden Web sunucusuna erişim sırasında artık bir güvenlik sorgulamasını onaylamanız gerekmez.

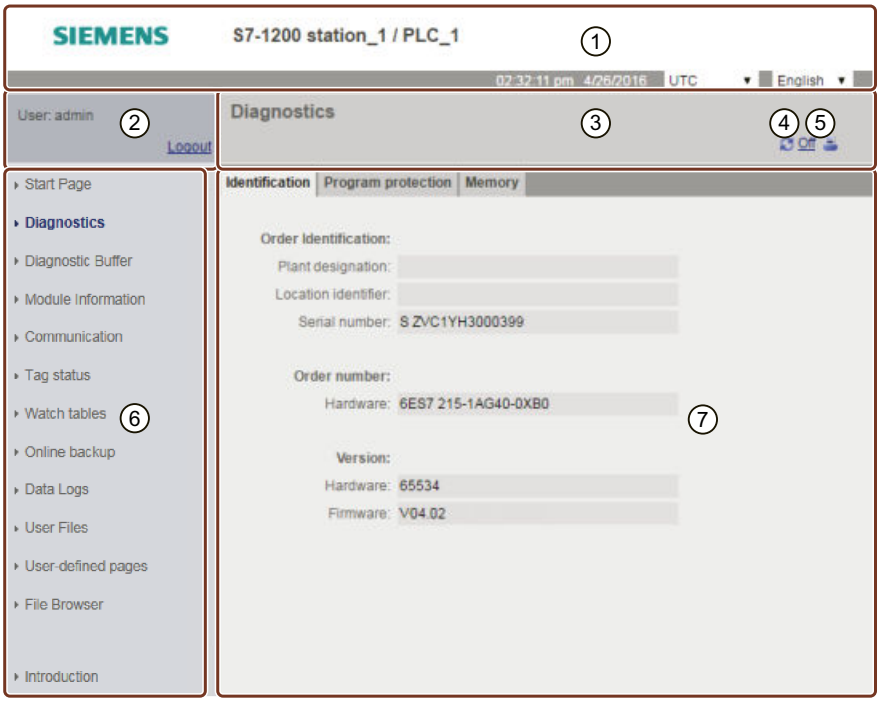
NOT

Güvenlik sertifikası, CPU yeniden başlatmada bile geçerliliğini korur. Cihazın IP adresini değiştirirseniz ve Internet Explorer veya Mozilla Firefox dışında bir tarayıcı kullanmanız durumunda, yeni bir güvenlik sertifikası yüklemeniz gerekecektir.

12.7 Standart web siteleri

12.7.1 Standart web sitelerinin yapısı

Tüm S7-1200 standart web siteleri ortak bir yapıya sahiptir ve site içinde gezinme ve kumanda elemanları için bağlantılara sahiptir. Siteye ister PC'den ister mobil cihazdan erişin, her site aynı içeriğe sahiptir. Ancak yapı ve gezinme elemanları, cihazın ekran boyutuna ve çözünürlüğüne göre farklılık göstermektedir. Bir standart PC' de ve büyük mobil cihazlarda, bir standart web sitesinin yapısı şöyledir:



① PLC yerel saati veya UTC saati ve Gösterge dili (Sayfa 144-145) olanaklarını içeren Web sunucusunun başlık satırı

② Oturum açma veya oturumu kapatma

③ Görüntülenen sayfanın adıyla birlikte standart web sitesinin başlık satırı. Bu örnekte, "CPU teşhis > Tanımlama" sayfasıdır. Bazı standart web sitelerinde, örneğin modül bilgi sitesinde, bu türden birden fazla site çağrılabiliyorsa burada bir gezinme yolu da görüntülenir.

④ Yenile sembolü: Otomatik yenileme içeren siteler için, otomatik yenileme özelliğini etkinleştirin veya devre dışı bırakın; otomatik yenileme özelliği olmayan sitelerde ise ilgili sayfa güncel verilerle yenilenir

⑤ Yazdır sembolü: Görüntülenen sayfadaki bilgilerin yazdırılabilir bir sürümünü oluşturur ve bunu görüntüler

⑥ Başka bir sayfaya gitmek için gezinme alanı

⑦ O an görüntülemekte olduğunuz belirli bir standart web sitesinin içerik alanı. Bu örnekteki sayfa, teşhis sayfasıdır.

NOT

CP modüllerinin standart web siteleri

Belirli CP modülleri (Sayfa 805), görünüm ve fonksiyonellik açısından S7-1200 CPU'nun standart web sitelerine benzer standart web siteleri sunar. CP modülünüzün dokümantasyonunda CP'nizin standart web sitelerine yönelik bir açıklama bulabilirsiniz.

12.7.2 Temel siteler

Web sunucusu, mobil cihazlarda kullanıma yönelik temel sayfalar sunar. Temel sayfalara, cihazın IP adresini ve URL'ye "basic" uzantısını kullanarak erişirsiniz: http://ww.xx.yy.zz/basic veya https://ww.xx.yy.zz/basic

Temel siteler, standart sitelere benzer, yine de birkaç fark mevcuttur. Sitede gezinme alanı, oturum açma alanı ve başlık alanı yoktur, bunun yerine web sitelerinde ileri ve geri kaydırmak için butonlar mevcuttur. Temel sayfalarda ayrıca bir başlangıç sayfası butonu bulunur, bu da sizi bir gezinme sayfasına götürür. Ancak, mobil cihazın gezinme elemanlarını da kullanabilirsiniz. Örneğin, temel teşhis sitesi dikey yönde şöyle görünür:

Bir temel siteyi görüntülemek için minimum çözünürlük 240 x 240 pikseldir.



Bu bölümdeki standart web sitelerinin görsellerinin bir standart PC'deki gösterime karşılık geldiğini unutmayın. Çoğu standart web sitesinin ilgili temel siteleri vardır.

12.7.3 Oturum açma ve kullanıcı yetkileri

Standart web sitelerinin her biri, PC'de gezinme alanının üzerinde bir oturum açma penceresi sunar. Alan sınırlamaları nedeniyle, temel web siteler kendi oturum açma sayfalarını sağlar. S7-1200, farklı erişim kademelerinde (yetkilerine) çoklu kullanıcı oturumlarını destekler:

- Teşhis çağır
- Değişkenleri oku
- Değişkenleri yaz
- Değişken durumunu oku
- Değişken durumunu yaz
- Kullanıcı tanımlı web sitelerini aç
- Kullanıcı tanımlı Web sitelerine yaz
- Dosyaları oku
- Dosyaları yaz/sil
- İşletim durumu değiştir
- LED'lerin yanıp sönmesini sağla
- Firmware güncellemesini gerçekleştir
- PLC'nin bir yedekleme kopyasını oluştur
- Yedek Firmware'i kullanarak PLC'yi geri yükle
- Yalnızca hata güvenli S7-1200 CPU'lar için parametre erişimi (F-Admin)

STEP 7'de, CPU'nun cihaz konfigürasyonunda Web sunucusu kullanıcı yönetiminin özelliklerinde, kullanıcı rollerini yapılandırır, erişim kademelerini (yetkiler) ve şifreler atarsınız (Sayfa 801).

Oturum açma

STEP 7, standart kullanıcıya şifresiz "Herkes" özelliğini sunar. Standart olarak, bu kullanıcının ek yetkileri yoktur ve yalnızca Başlangıç (Sayfa 813) ve Giriş (Sayfa 813) standart web sitelerini açabilir. Fakat "Herkes" kullanıcısına, yapılandırdığınız diğer tüm kullanıcılar gibi ek yetkiler tanıyabilirsiniz:

⚠ İKAZ

Web sunucusu üzerinden CPU'ya yetkisiz erişim

Tam CPU erişim yetkilerine veya hata güvenli özelliği de dahil tam erişim olanağına sahip kullanıcılar PLC değişkenlerini okuyabilir ve yazabilir. Web sunucusu kullanıcıları, CPU için kullanılan erişim kademesinden bağımsız olarak PLC değişkenlerini okuma ve yazma yetkilerine sahip olabilir. CPU'ya yetkisiz kişiler tarafından erişim veya PLC değişkenleri için geçersiz değerlerin ayarlanması, proses işletimine zarar verebilir ve hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Yetkili kullanıcılar, işletim durumunda değişiklikler yapabilir, PLC verilerini yazabilir ve Firmware güncellemeleri uygulayabilir. Siemens, aşağıdaki güvenlik önlemlerine uyulmasını önerir:

- CPU erişim kademelerini (Sayfa 156) ve Web sunucusu kullanıcı ID'lerini (Sayfa 801) STEP 7'de tanımlandığı şekilde güçlü şifreler aracılığıyla koruyun.
- Web sunucusuna erişimi sadece HTTPS protokolü üzerinden etkinleştirin.
- Web sunucusu kullanıcısının "Herkes" şeklindeki standart asgari yetkilerini genişletmeyin.
- Web sitelerinin kullanıcıları PLC değişkenleri için geçersiz değerler belirleyebileceği için, program lojiğinizdeki değişkenler için bir hata kontrolü ve bir aralık kontrolü gerçekleştirin.
- Korumalı ağınızın dışındaki bir konumdan S7-1200 PLC Web sunucusuyla bağlantı kurmak için güvenli bir sanal özel ağ (Virtual Private Network - VPN) kullanın.

Kumanda ünitesini başka bir işletim durumuna geçirmek, belleğe değerler yazmak veya CPU'nun Firmware'ini güncellemek gibi belirli eylemleri gerçekleştirmek istiyorsanız, bunlar için gereken yetkilere ihtiyacınız olur. CPU koruma kademesi (Sayfa 156) için "Erişim yok (tam koruma)" opsiyonunu ayarladıysanız, ayarlanan Web sunucusu kullanıcı yetkilerine bakılmaksızın "Herkes" kullanıcısının Web sunucusuna erişim yoktur.



Oturum açma kutusu, bir PC'de veya büyük ekranlı mobil cihazda görüntülendiğinde herhangi bir standart web sitesinin sol üst kısmındadır.



Temel siteleri görüntüleyen küçük mobil cihazlarda oturum açma sayfası ayrı bir sayfadır. Bu sayfaya, başlangıç sayfasından erişilebilir.

Oturum açmak için aşağıdaki yöntemi izleyin:

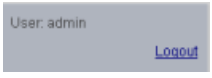
1. "Kullanıcı adı" alanına kullanıcı adını girin.
2. "Şifre" alanına kullanıcı şifresini girin.

Otuz dakika süreyle bir işlem yapılmazsa, oturumunuz otomatik olarak kapatılır. Görüntülenen sayfa sürekli yenileniyorsa, oturum açma süresi dolmaz ve oturumunuz açık kalır.

NOT

Oturum açma sırasında hatalarla karşılaşırsanız, Giriş Sayfasından (Sayfa 806) Siemens güvenlik sertifikasını (Sayfa 813) indirin. Bunun ardından hatasız bir biçimde oturum açabilirsiniz.

Oturumu kapatma





Oturumu kapatmak için, bir PC veya büyük ekranlı mobil cihazdaki herhangi bir sayfada bulunan "Oturumu kapat" linkine tıklamanız yeterlidir.

Başlangıç sayfasından, temel siteler üzerinden oturum açma ve oturumu kapatma sayfasına gidin ve "Oturumu kapat" butonuna tıklayın.

Oturumu kapattıktan sonra, standart web sitelerine yalnızca "Herkes" kullanıcısının yetkileri uyarınca erişebilir ve bunları görüntüleyebilirsiniz. Site için gerekli haklar, münferit standart web sitelerinin açıklamalarında tanımlanmıştır.

NOT

Web sunucusunu kapatmadan önce oturumunuzu kapatın

Web sunucusunda oturum açtıysanız, web tarayıcısını kapatmadan önce mutlaka oturumu kapatmalısınız. Web sunucusu aynı anda en fazla yedi kullanıcıyı destekler.

12.7.4 Giriş

Giriş sayfası, S7-1200 standart web sitelerine başladığınızdaki karşılama sayfasıdır.



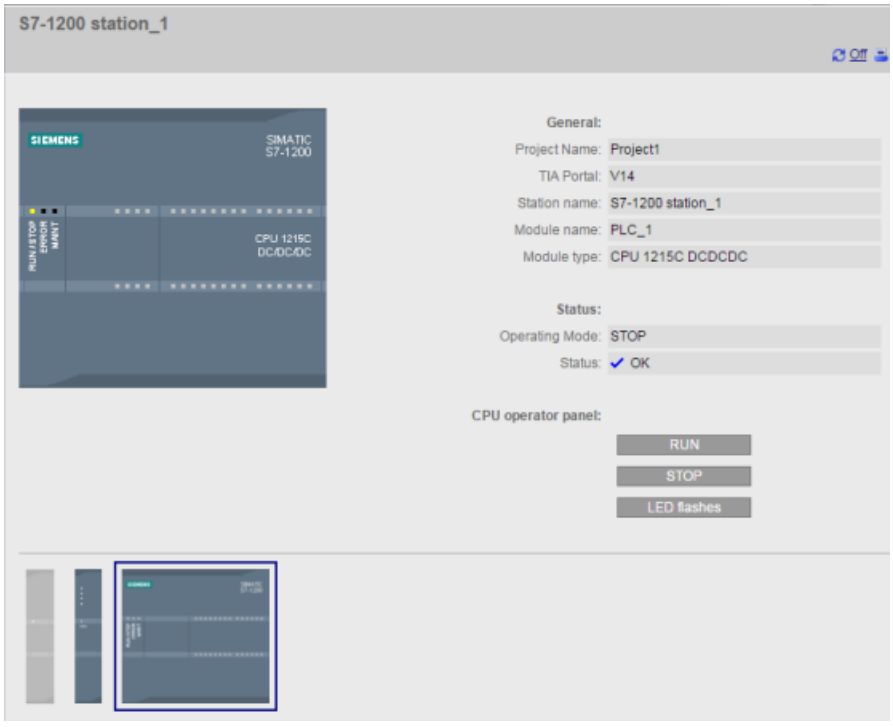
Bu sayfada, S7-1200 standart web sitelerini açmak için "Enter" öğesine tıklayın. Sayfanın üst kısmında, faydalı Siemens web sitelerine linkler ve Siemens güvenlik sertifikasını (Sayfa 806) indirmek için bir link bulursunuz. Burada daha sonraki Web sunucusuna erişimlerde giriş sayfasının atlanması gerektiğini de belirtebilirsiniz.

12.7.5 Start

Başlangıç sayfası, bağlı olduğunuz CPU veya CP'nin birer görselini görüntüler ve cihaz ve projeyi CPU'ya yüklemek için kullandığınız TIA Portal'ın sürümüne ilişkin genel bilgileri listeler. CPU için, "İşletim durumunu değiştir" yetkisi (Sayfa 801) ile oturum açmış (Sayfa 809) olmanız koşuluyla, işletim durumunu değiştirmek ve LED'leri kumanda etmek için butonları kullanabilirsiniz.

Ekranın alt kısmında, Web sunucusu üzerinden etkinleştirilmiş olan CP modüllerinin (Sayfa 805) yapılandırılıp yapılandırılmadığını ve S7-1200 CPU ile yerel modül taşıyıcısında kurulup kurulmadığını görüntülenmektedir Standart web sitelerine erişmek için, imleç ile Web sunucusu üzerinden etkinleştirilen CP modüllerinin üzerine gelin ve bir modüle tıklayın. CP modüllerinin web siteleri hakkında bilgileri, CP modülünüzün dokümantasyonunda bulabilirsiniz. Fare ile üzerine gittiğinizde CP modülünün adı görüntülenir.

Web sunucusu, yerel modül taşıyıcısındaki diğer CM ve CP modüllerini de görüntüler, ancak bunlar herhangi bir web sitesi içermediğinden bunlara tıklanamaz. Bu CM ve CP modülleri, açık gri renkte gösterilerek bunlara tıklanamayacağı ve bunların yalnızca görüntülenebilecekleri belirtilmektedir.



S7-1200 hata güvenli CPU'ların bu sitede fonksiyonel güvenlikle ilgili ek veriler de gösterdiğini dikkate alın.

12.7.6 Teşhis

Teşhis sayfası, CPU'nun tanımlayıcı özelliklerini, teknik bilgi koruması için konfigürasyon ayarlarını ve yükleme belleğinin, çalışma belleğinin ve kalıcı belleğin bellek kullanımını gösterir:

Bu site, üç sekme içerir:

- Tanımlama: STEP 7'deki konumla ilgili bilgiler ve ayrıca modülün ve tesis özelliklerinin tanımlanması
- Program koruması: Yedek parça planlamasında yararlı olabilecek teknik bilgi koruması ve CPU bağlama durumu ile dahili yükleme belleğinden harici yükleme belleğine (SIMATIC memory card) kopyalamalara izin vermek veya bunları engellemek için STEP 7 konfigürasyon ayarı.
- Bellek: Yükleme belleği, çalışma belleği ve kalıcı bellek kullanımı

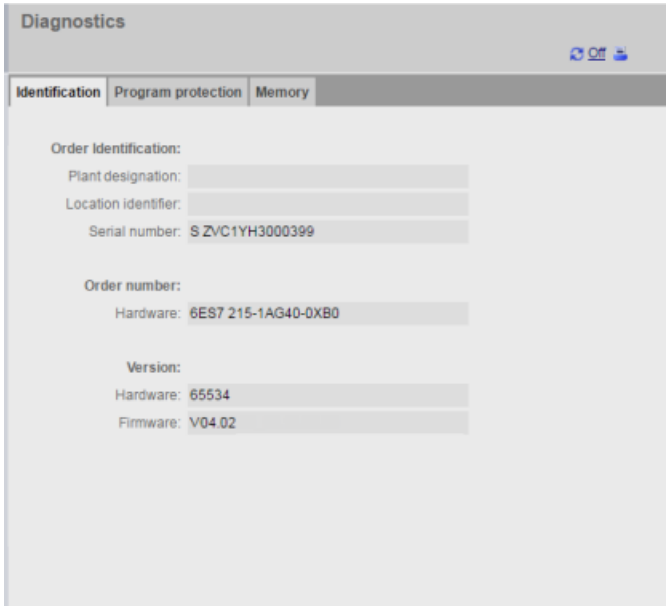
F-CPU'larda hata güvenliği için bir diğer sekme mevcuttur.

Tanımlama sayfasını görüntülemek için, "Teşhis sorgusu" yetkisine (Sayfa 801) ihtiyacınız vardır.

"Tanımlama" sekmesi

"Tanımlama" sekmesi aşağıdaki bilgileri içerir:

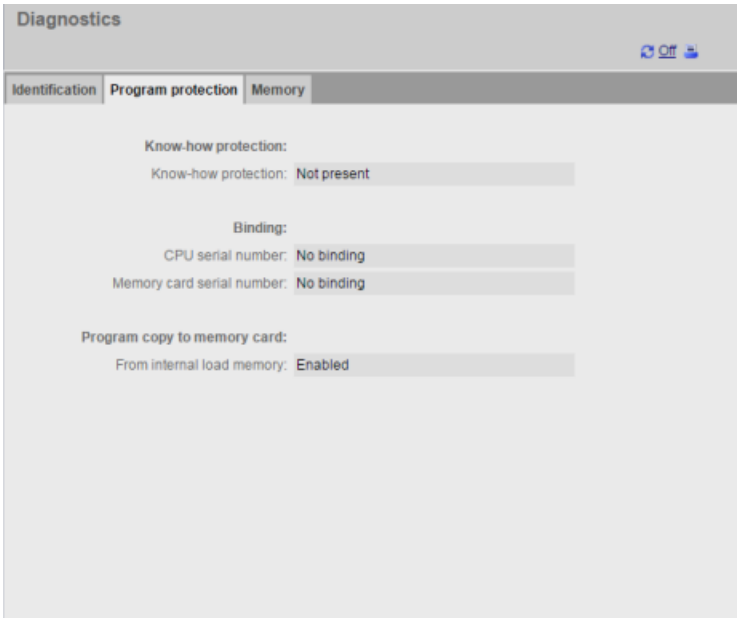
- Tanımlama: Tesis kodunu, yer kodunu ve seri numarasını görüntüler.
- Sipariş numarası: PLC'nin sipariş numarasını gösterir.
- Sürüm: PLC ve Firmware'in sürüm durumunu gösterir.



"Program koruması" sekmesi

"Program koruması" sekmesi aşağıdaki bilgileri içerir:

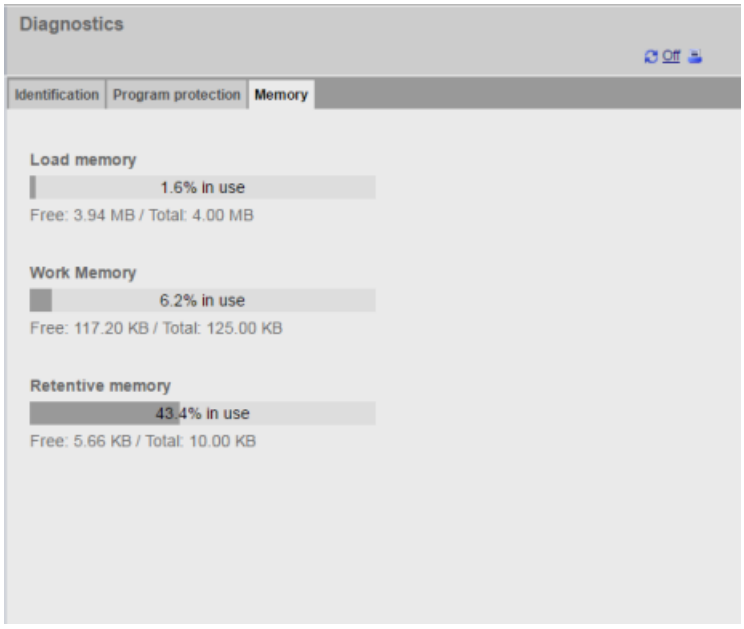
- Teknik bilgi koruması (Sayfa 159): STEP 7'deki program blokları için teknik bilgi korumasını yapılandırıp yapılandırmadığınızı gösterir.
- Bağlama (Sayfa 161): Programı CPU'ya mı yoksa SIMATIC memory card'a mı bağladığınızı gösterir.
- Programın memory card'a kopyalanması (Sayfa 159): Programı, dahili yükleme belleğinden harici yükleme belleğine (SIMATIC memory card) kopyalama opsiyonunu etkinleştirip etkinleştirmediğinizi gösterir.



"Bellek" sekmesi:

"Bellek (Sayfa 86)" sekmesi, aşağıdaki bilgileri içerir:

- Yükleme belleği: Kullanımı yüzde olarak ve boş/toplam belleği ise MB cinsinden gösterir.
- Çalışma belleği: Kullanımı yüzde olarak ve boş/toplam belleği ise kB cinsinden gösterir.
- Kalıcı bellek: Kullanımı yüzde olarak ve boş/toplam belleği ise kB cinsinden gösterir.



"Hata güvenli" sekmesi

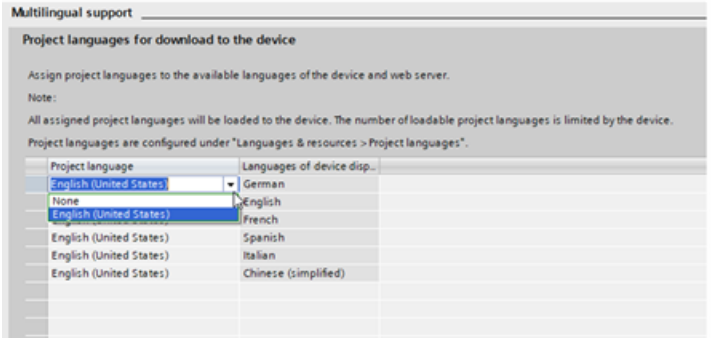
"Hata güvenli" sekmesinin teşhis sayfasına yönelik daha fazla bilgi için bkz. S7-1200 fonksiyonel güvenlik işletme kitabı (https://support.industry.siemens.com/cs/ww/en/view/104547552).

12.7.7 Teşhis tamponu

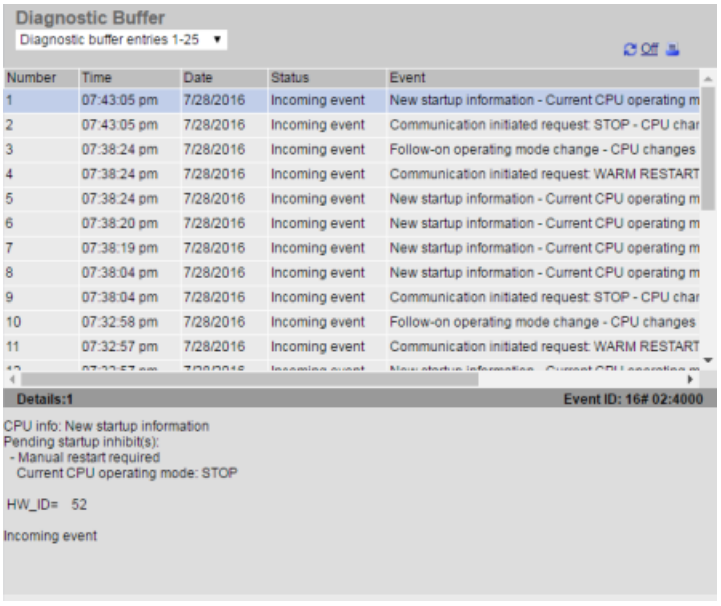
Teşhis tamponu sitesi, teşhis olaylarını görüntüler. En son olay, üstte 1 numara ile görüntülenir. En eski olayın numarası 50'dir. Soldaki seçme alanında, teşhis tamponu girişlerinin hangi aralığının görüntülenmesi gerektiğini, 1 ila 25 girişleri veya 26 ila 50 girişlerini seçerek belirleyebilirsiniz. Sağdaki seçme alanında, saatlerin UTC'de mi yoksa yerel PLC saatinde mi görüntüleneceğini belirlersiniz. Bu teşhis girişleri, sayfanın üst kısmında oluşma saatleri ve tarihleri ile birlikte görüntülenir.

Tek bir teşhis kaydı seçerek sayfanın alt kısmında ilgili daha fazla bilgi görüntüleyebilirsiniz. Teşhis tamponu girişlerinin görüntüleme dilinin, çok dilli destek (Sayfa 151) olarak ayarlanan cihaz konfigürasyonuna bağlı olduğunu dikkate alın.

PLC'ye yüklenen TIA Portal projesinde teşhis metinleri için kullanılan dilleri yapılandırmanız gerekir. Bu konfigürasyon, "Çok dilli destek" altındaki PLC özelliklerinde sunulmaktadır. İndirilen her dil, desteklenen bir Web sunucusu diline atanabilmelidir. PLC indirilen iki dille sınırlandırılmıştır.



Teşhis tamponu sitesi, ayrıca teşhis tamponunu bir CSV dosyası olarak kaydetmeyi sağlayan bir buton da içerir. Standart olarak, Web sunucusu dosyayı virgülle ayrılmış formatta İndirilenler klasörünüze ASLog.csv olarak kaydeder. Dosya, kaydetme anındaki tüm teşhis tamponunu içerir. Dosyayı istediğiniz sıklıkta kaydedebilir ve birden çok dosyayı saklayabilirsiniz. Teşhis tamponu dosyasını, Microsoft Excel veya herhangi bir metin editörüyle açabilirsiniz.



Teşhis tamponu sitesini görüntülemek için, teşhis çağrısı yetkisi (Sayfa 801) gereklidir.

12.7.8 Modül bilgileri

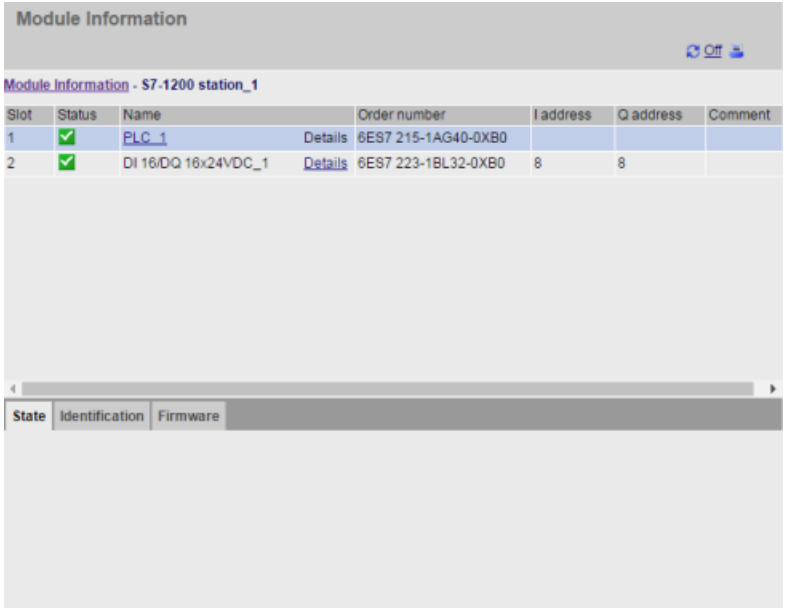
Modül bilgisi sitesi:

- Yerel modül taşıyıcısındaki tüm modüllere ilişkin bilgileri görüntüler. Sayfanın üst kısmında, STEP 7'deki cihaz konfigürasyonuna göre modüllere genel bakış görüntülenir, alt kısmında ise ilgili bağlı modül temelinde seçilen modülün durumu, tanımlaması ve Firmware bilgileri görüntülenir.
- Firmware güncellemesi gerçekleştirme olanağı sağlar.
- Desantral çevrebirim sistemlerine ilişkin bilgiler

Modül bilgileri sitesini görüntülemek için teşhis çağrısı yetkisi (Sayfa 801) gereklidir.

Modül bilgileri: "Durum" sekmesi

Modül bilgisi sitesinin alt kısmındaki "Durum" sekmesinde, üst kısımda seçilen modülün güncel durumuna yönelik bir açıklama görüntülenir. Bu alt bölüm boşsa, modülün bekleyen bir teşhis durumu yoktur.



Modüller için durum sembolleri

Her bir modül için, üstteki durum sütununda modülün durumunu gösteren bir sembol görüntülenir:

Sembol	Anlamı
	Arıza yok
	Devre dışı bırakıldı
	Bakım gerekli
	Bakım talep edildi
	Hata
	CPU, modüle veya cihaza ulaşamıyor (CPU harici cihazlar için).
	CPU, cihazla bir bağlantı kurdu, ancak modül durumu bilinmiyor (CPU harici cihazlar için).
	Alt modül, kendisine ait G/Ç kanallarını kilitlediğinden (CPU harici cihazlarda) giriş ve çıkışlarla ilgili veriler mevcut değil.

Daha fazla ayrıntı gösterilmesi

Belirli bir modülün modül bilgilerinde daha fazla ayrıntı görüntülemek için üst kısımdaki ilgili linki seçebilirsiniz. Alt modüllü modüllerin her bir alt modülü için ayrı bir link mevcuttur. Görüntülenen bilgilerin türü, seçilen modüle göre değişmektedir. Örneğin modül bilgisinde başlangıçta, SIMATIC S7-1200 istasyonunun adı, bir durum göstergesi ve bir yorum görüntülenirler. İlgili linki CPU seviyesine genişletirseniz, modül bilgisi, CPU modelinin dijital ve analog giriş ve çıkışlarının adlarının yanı sıra G/Ç için adres bilgilerini, durum göstergelerini, yuva numaralarını ve yorumları görüntüler.

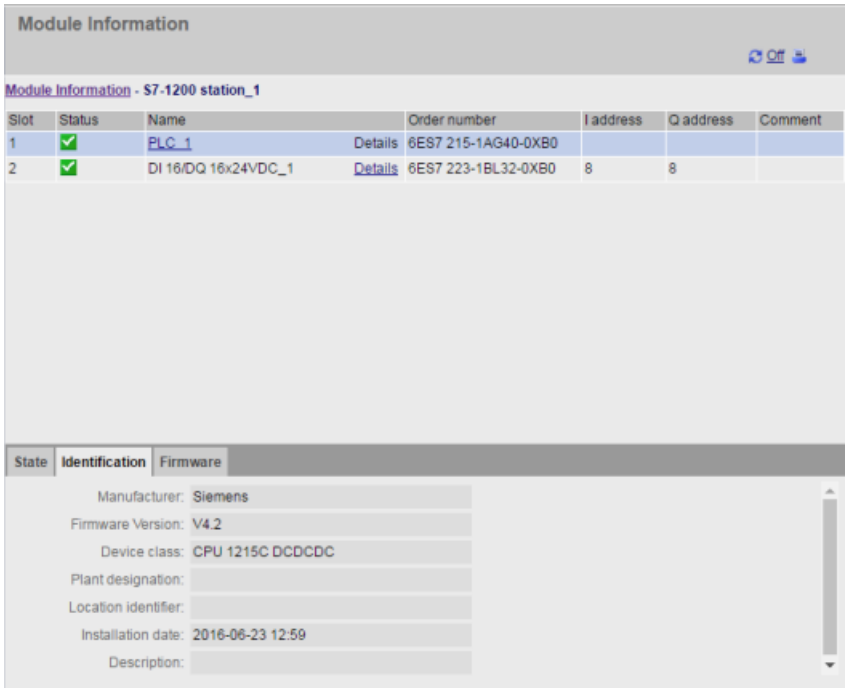
Slot	Status	Name		Order number	I address	Q address	Comment
1.16	✔	HSC_1	Details	6ES7 214-1AG40-0XB0	1000	---	
1.17	✔	HSC_2	Details	6ES7 214-1AG40-0XB0	1004	---	
1.18	✔	HSC_3	Details	6ES7 214-1AG40-0XB0	1008	---	
1.19	✔	HSC_4	Details	6ES7 214-1AG40-0XB0	1012	---	
1.20	✔	HSC_5	Details	6ES7 214-1AG40-0XB0	1016	---	
1.21	✔	HSC_6	Details	6ES7 214-1AG40-0XB0	1020	---	
1.2	✔	AI2_1	Details	6ES7 214-1AG40-0XB0	64	---	
1.1	✔	DI14/DQ10_1	Details	6ES7 214-1AG40-0XB0	0	0	
1.32	✔	Pulse_1	Details	6ES7 214-1AG40-0XB0	---	1000	

Görüntüyü genişletirken, modül bilgi sitesi ilgili yolu belirtir. Bir üst seviyeye dönmek için bu yoldaki herhangi bir linke tıklayabilirsiniz.



Modül bilgileri: "Tanımlama" sekmesi

"Tanımlama" sekmesi, seçilen modülün tanımlama ve bakım verilerini (Identification and Maintenance - Tanımlama ve Bakım, I&M) gösterir.



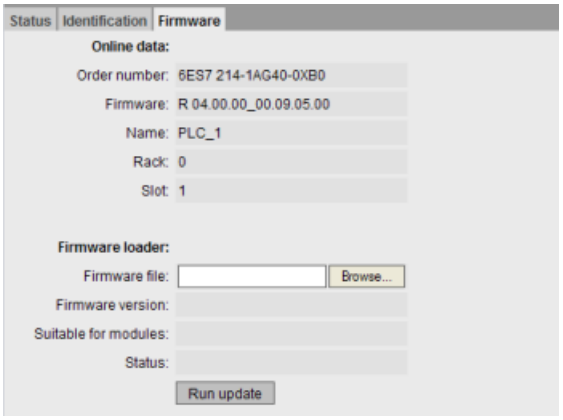
Üst kısımdaki bir F-G/Ç modülüne tıklarsanız, alt kısımda bir "Güvenlik" sekmesi görüntülenir. Bu sekmede, S7-1200 fonksiyonel güvenlik işletme kitabı (https://support.industry.siemens.com/cs/ww/en/view/104547552) bölümünde açıklandığı gibi seçili modüle ilişkin spesifik veriler görüntülenir.

Modül bilgileri: "Firmware" sekmesi

Modül bilgisi sitesinin "Firmware" sekmesi, seçilen modülün Firmware'ine yönelik bilgileri görüntüler. Firmware güncellemesi yapmak için yekiniz (Sayfa 801) varsa, CPU'nun Firmware'ini veya yerel modül taşıyıcısındaki Firmware güncellemelerini ve PROFINET G/Ç modüllerini destekleyen diğer modülleri de güncelleyebilirsiniz. Desantral modüllerde, Firmware bilgilerini görüntüleyebilir, ancak bir Firmware güncellemesi yürütemezsiniz.

NOT

CPU Firmware güncellemesi ile ilgili olarak, yalnızca sürüm 3.0 ve üstü S7-1200 CPU'ları güncelleyebilirsiniz.



Firmware güncellemesini gerçekleştir

Firmware güncellemesini gerçekleştirmek için CPU'nun STOP işletim durumunda olması gerekir. CPU STOP işletim durumunda ise, bir Firmware dosyası seçmek için "Gözat" butonuna tıklayın. Firmware güncellemeleri, Siemens Industry Online Support (http://support.industry.siemens.com) web sitesinde sunulmaktadır.

Güncelleme sırasında, sitede güncellemenin devam ettiğine dair bir mesaj görüntülenir. Güncelleme tamamlandıktan sonra, sitede güncellenen Firmware'in sipariş numarası ve sürüm numarası görüntülenecektir. CPU'nun Firmware'ini veya bir sinyal kartını güncellediyseniz, Web sunucusu CPU'yu yeniden başlatır.

Bir Firmware güncellemesini aşağıdaki yöntemlerden birini kullanarak da gerçekleştirebilirsiniz:

- STEP 7'nin çevrimiçi ve teşhis araçları (Sayfa 1120) ile
- Bir SIMATIC memory card (Sayfa 124) ile
- SIMATIC Automation Tool (https://support.industry.siemens.com/cs/ww/en/view/98161300) ile

NOT

Web sunucusu üzerinden bir Firmware güncellemesi gerçekleştirirken olası problemler

Web sunucusu üzerinden Firmware güncellemesi yapılırken bir iletişim hatası oluşursa, web tarayıcısı, güncel siteden çıkmak isteyip istemediğinizi soran bir mesaj görüntüleyebilir. Olası problemleri önlemek için güncel sitede kalma opsiyonunu seçin.

Web sunucusu üzerinden bir Firmware güncellemesi yapılırken Web sunucusunu kapatırsanız, CPU'nun işletim durumunu RUN işletim durumu olarak değiştiremezsiniz. Burada, CPU'yu RUN işletim durumuna geçirmek için CPU'yu kapatıp açmanız gerekecektir.

"Sıcak başlatma" fonksiyonu

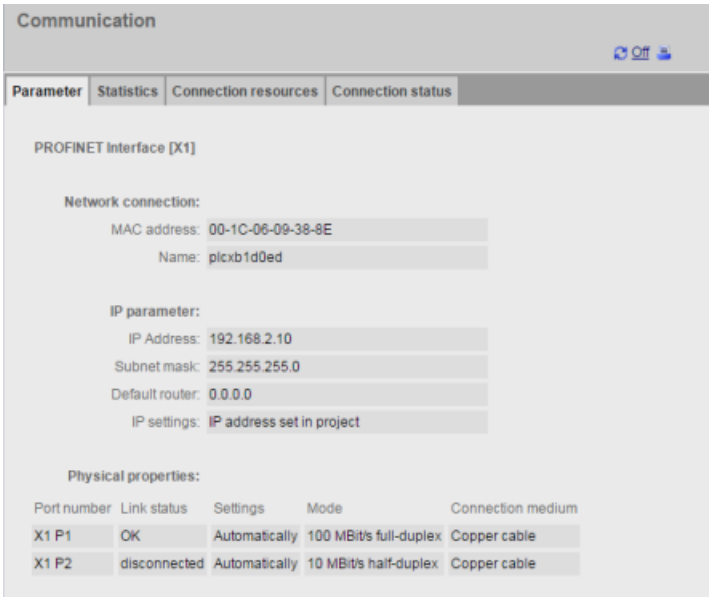
Sıcak başlatma fonksiyonu yalnızca Firmware güncellemesi sırasında kullanılabilir. "Sıcak başlatma" fonksiyonunu diğer bir zamanda çağıramazsınız.

12.7.9 İletişim

İletişim sitesi, bağlı CPU'nun parametreleri, iletişim istatistikleri, kaynaklar ve bağlantılar ile ilgili bilgileri gösterir.
İletişim sitesini görüntülemek için, "Teşhis sorgusu" yetkisine ihtiyacınız vardır.

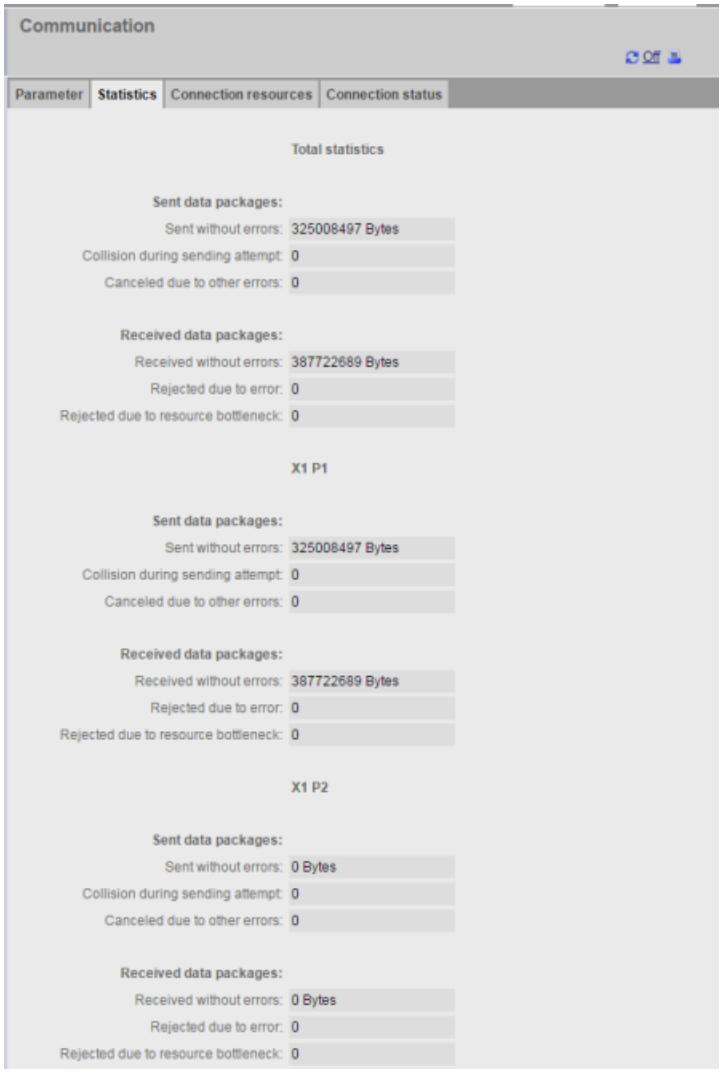
"Parametreler" sekmesi

"Parametreler" sekmesi, CPU'nun MAC adresini, CPU'nun IP adresini ve IP ayarlarını ve fiziksel özellikleri gösterir:



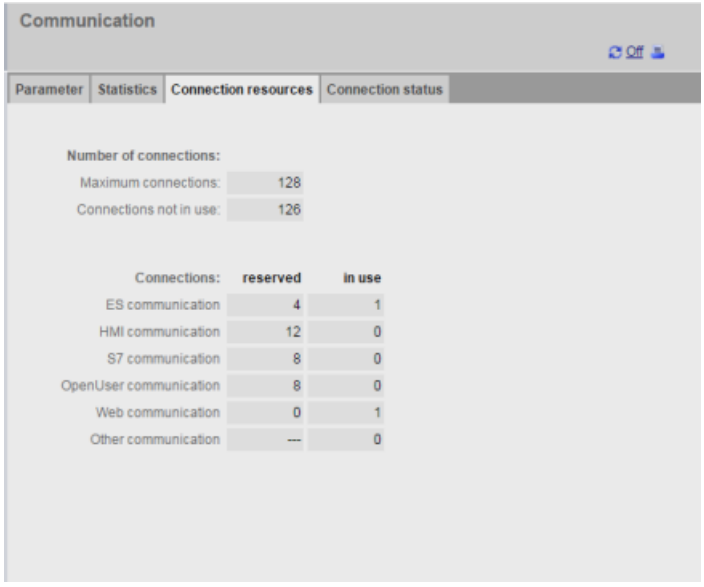
"İstatistikler" sekmesi

"İstatistikler" sekmesi, gönderim ve alım iletişimin istatistiklerini gösterir:



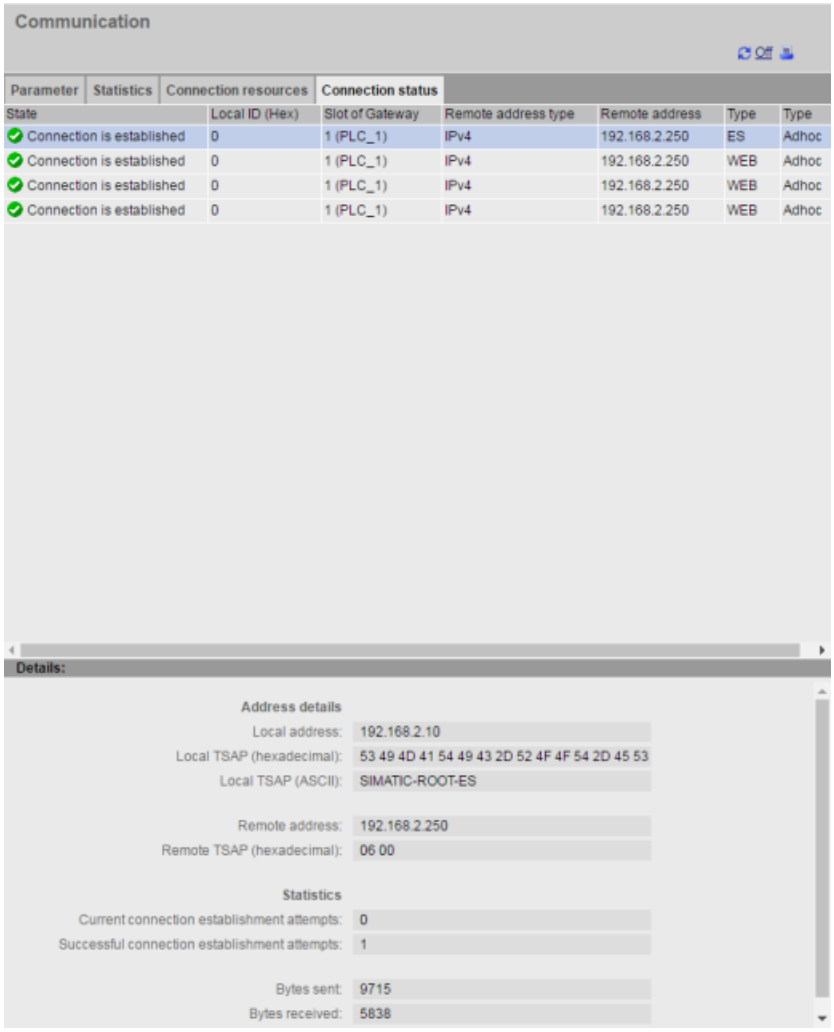
"Bağlantı kaynakları" sekmesi

"Bağlantı kaynakları" sekmesi, bağlantı kaynaklarının toplam sayısı ve bunların çeşitli iletişim türlerine nasıl atandığı ile ilgili bilgiler gösterir:



"Bağlantı durumu" sekmesi

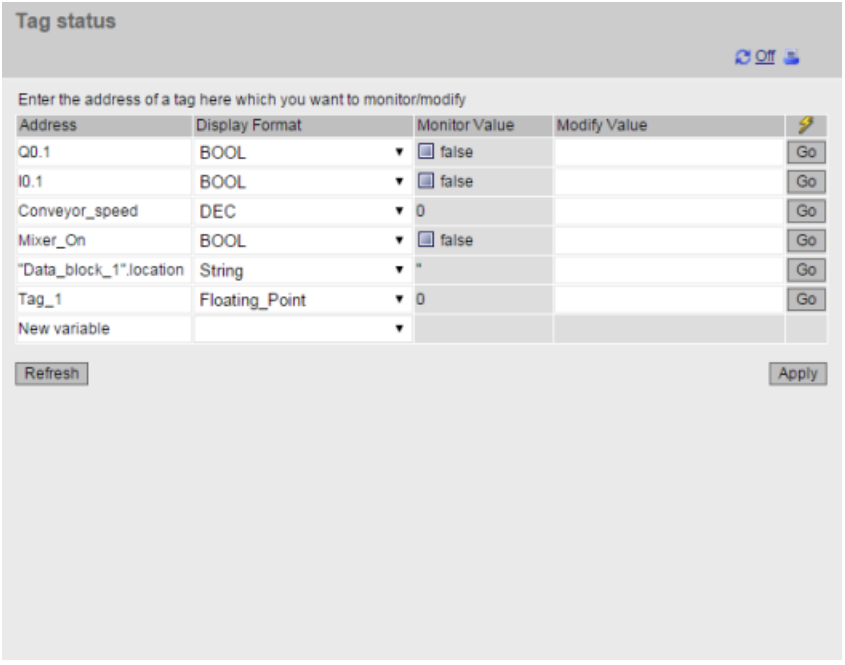
"Bağlantı durumu" sekmesi, CPU bağlantılarını ve seçilen bağlantı ayrıntılarını gösterir.



12.7.10 Değişken durumu

Değişken durumu sitesinde, CPU'nuzdaki tüm G/Ç veya bellek verilerini görüntüleyebilirsiniz. Bir doğrudan adres (örn. %E0.0), bir PLC değişkeni adı veya belirli bir veri bloğundan bir değişken girebilirsiniz. Veri bloğu değişkenlerinde blok adını çift tırnak içine alın. Denetlenen her değer için, veri görüntüleme formatı seçebilirsiniz. Sayfa sınırlaması kapsamında istediğiniz tüm değerleri belirtene kadar değer girmeye devam edebilirsiniz. Denetlenen değerler otomatik olarak görüntülenir. Denetlenen tüm değerleri istediğiniz zaman "Güncelle" butonuna tıklayarak güncelleyebilirsiniz. STEP 7'de otomatik güncellemeyi etkinleştirdiyseniz (Sayfa 798), bu ayarı devre dışı bırakmak için sayfanın sağ üst kısmındaki "Kapalı" öğesine tıklayabilirsiniz. Otomatik güncelleme devre dışı ise, "Açık" öğesi üzerinden tekrar etkinleştirebilirsiniz.

Değişken durumu sitesini görüntülemek için, "Değişken durumunu oku" yetkisi gereklidir. "Değişken durumunu yaz" yetkisi (Sayfa 809) olan bir kullanıcı olarak oturum açarsanız, veri değerlerini de değiştirebilirsiniz. İlgili "Kumanda değeri" alanına, ayarlamak istediğiniz değerleri girin. Değeri CPU'ya yazmak için bir değerin yanındaki "Git" butonuna tıklayın. Bundan başka birden fazla değer girip "Uygula" öğesine tıklayarak tüm değerleri CPU'ya yazabilirsiniz. Değiştirme işlemine yönelik butonlar ve sütun adları yalnızca "Değişken durumunu yaz " yetkisine sahipseniz görüntülenir.



Değişken durumu sitesinden çıkıp geri dönerseniz, Değişken durumu sitesindeki girişleriniz kaydedilmez. Sayfaya yer imi koyabilir ve aynı girişleri görmek istediğinizde ilgili yer imini tekrar çağırabilirsiniz. Sayfaya yer imi koymazsanız, söz konusu değişkenleri yeniden girmeniz gerekir.

Yer iminin yerine, sıklıkla izlenecek veya kullanılacak değerlerde bir İzleme tablosu (Sayfa 829) kullanmayı düşünebilirsiniz.

NOT

Değişken durumu standart sitesini kullanırken aşağıdakileri dikkate alın:

- Tüm karakter dizisi değişikliklerini tek kesme imi içine alın.
- Değişken durumu sayfası, aşağıdaki karakterlerden herhangi birini içeren değişkenleri izleyebilir ve çalıştırabilir: &, <, (, +, ,(virgül), ., [,], $ veya %. Bunun ön koşulu, değişken adını çift tırnak içine almanızdır, örneğin "Takt_2,5Hz".
- Bir DTL değişkeninin yalnızca bir alanını izlemek veya değiştirmek için, ilgili alanı adrese dahil edin, örneğin "VeriBloğu_1".DTL_Değişken.Yıl. Kumanda değeri için, belirli DTL alanının veri tipine uygun bir tamsayı değeri girin. Örnek: Yıl alanı, UInt veri tipine sahiptir.
- Sayfa başına maksimum değişken girişi sayısı 50'dir.
- Bir değişken adı, Değişken durumu sayfasında bir veri girişi olarak reddedilmesine neden olacak özel karakterler içeriyorsa, o değişkenin adını çift tırnak içine alabilirsiniz. Ancak sayfa, genellikle değişken adını tanır.

Ayrıca bkz.

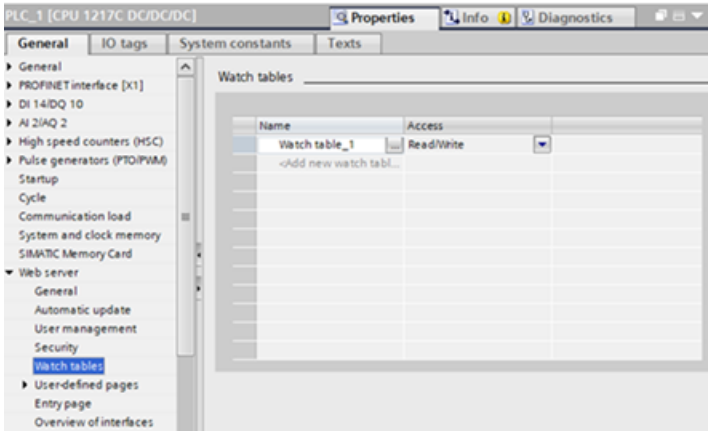
Değişken adlarını ve değerlerini girme kuralları (Sayfa 886)

12.7.11 İzleme tabloları

Web sunucusu üzerinden, STEP 7'de yapılandırdığınız ve CPU'ya yüklediğiniz izleme tablolarına erişebilirsiniz. 50 veya daha az veri girişi içeren izleme tabloları, Web sunucusunda en iyi performansı sağlar.

Web sunucusuna yönelik izleme tablolarını seçmek için STEP 7 konfigürasyonu

STEP 7'de CPU'nun cihaz konfigürasyonuna, Web sunucusu tarafından görüntülenecek olan izleme tablolarını ekleyebilirsiniz. Mevcut izleme tabloları listesinden seçtiğiniz her bir izleme tablosu için Okuma veya Okuma/yazma yetkileri atayın. CPU'ya yükledikten sonra, Okuma yetkileri atadığınız izleme tablolarını yalnızca görüntüleyebilirsiniz, öte yandan Okuma/yazma yetkileri atadığınız izleme tablolarında değişkenleri görüntüleyebilir ve değiştirebilirsiniz.



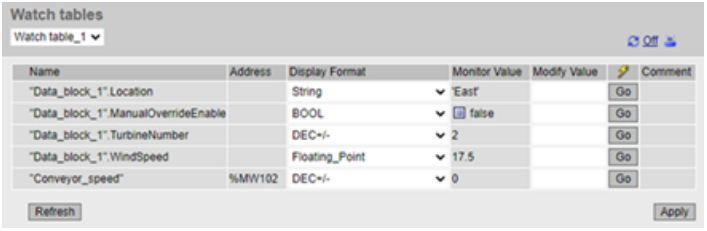
Cihaz konfigürasyonunun "Web sunucusu" alanında iken izleme tablosunun konfigürasyonunu tamamladıktan sonra, donanım konfigürasyonunuzu CPU'ya yükleyin.

İzleme tablolarının Web sunucusu üzerinden görüntülenmesi

"Değişkenleri oku" yetkisine (Sayfa 801) sahipseniz, yapılandırdığınız ve CPU'ya yüklediğiniz izleme tablolarına erişmek için gezinme menüsünde "İzleme tabloları" öğesini seçebilirsiniz. Birden çok izleme tablosu yüklediyseniz, görüntülemek istediğiniz tabloyu açılır listeden seçersiniz. Web sunucusu, STEP 7'de oluşturduğunuz izleme tablosunu ve güncel değerleri görüntüleme formatına uygun olarak görüntüler. Görüntüleme formatını dilediğiniz gibi değiştirebilirsiniz, ancak izleme tabloları sayfasına döndüğünüzde, Web sunucusu standart olarak STEP 7 izleme tablosundaki görüntüleme formatlarına geri döner.

İzleme tablolarındaki değişkenlerin Web sunucusu üzerinden değiştirilmesi

"Oku/yaz" erişim kademesine sahip bir izleme tablosu yüklediyseniz ve "Değişkenleri yaz" yetkisi (Sayfa 801) ile Web sunucusunda oturum açtıysanız, tıpkı STEP 7'deki izleme tablosunda olduğu gibi burada da değişken değerleri yazabilirsiniz. Tek bir değişken değerini değiştirip "Git" öğesine tıklayarak tek bir değeri değiştirebilir veya birden çok değer girip "Uygula" öğesine tıklayarak tüm değerleri bir defada değiştirebilirsiniz.



NOT

Değişkenleri değiştirmek için izleme tablolarının avantajları

Bir kullanıcının izleme tablosu üzerinden CPU'daki değişkenleri ve veri bloğu değişkenlerini değiştirebilmesini sağlamak için, izleme tablosunu STEP 7 cihaz konfigürasyonundaki Web sunucusu özelliklerinde yapılandırmanız ve tabloya Oku/yaz erişimini atamanız gerekir. Böylece "Değişkenleri yaz" yetkisine sahip bir kullanıcının değiştirebileceği değişkenleri, sadece yapılandırılmış Web sunucusu izleme tablolarındaki değişkenlerle sınırlamanız mümkün olur.

Ancak öte yandan, Değişken durumu (Sayfa 827) sayfasında, "Değişken durumunu yaz" yetkisine sahip herhangi bir kullanıcı, CPU'daki herhangi bir değişkene veya veri bloğu değişkenine yazabilir.

Web sunucusunun Kullanıcı yönetimindeki yetkileri (Sayfa 801) dikkatlice yapılandırarak, PLC verilerinize olan erişimi koruyabilirsiniz.

Ayrıca bkz.

Değişken adlarını ve değerlerini girme kuralları (Sayfa 886)

12.7.12 Çevrimiçi yedekleme

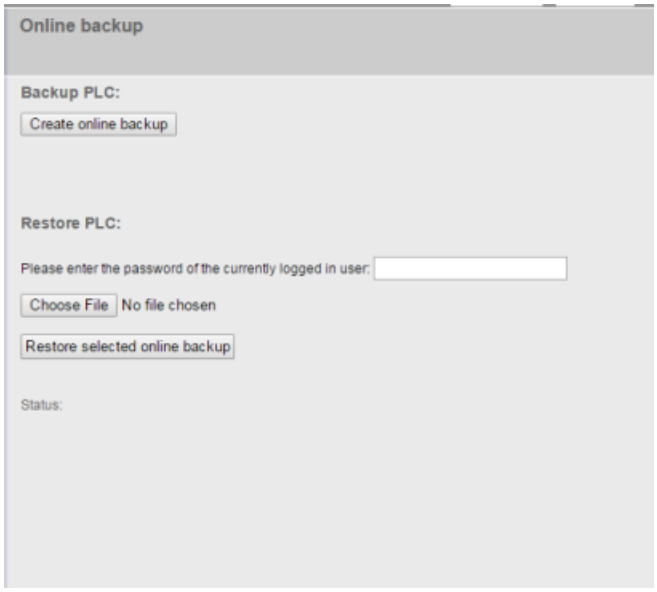
Standart web sitesinde Çevrimiçi yedekleme üzerinden çevrimiçi PLC için STEP 7 projesinin bir yedekleme kopyasını oluşturabilir ve yine aynı şekilde PLC'nin önceden oluşturulmuş bir yedekleme kopyasını geri yükleyebilirsiniz. Bir yedekleme kopyası oluşturmadan veya geri yüklemeden önce, PLC'yi STOP işletim durumuna getirin ve HMI erişimi ve Web sunucusu erişimi gibi PLC ile olan tüm iletişimini durdurun. CPU'nuz STOP işletim durumunda değilse, yedekleme ve geri yükleme fonksiyonları, ilgili işleme devam etmeden önce sizden CPU'yu STOP işletim durumuna geçirmenizi isteyen bir mesaj görüntüler.

Web özellikli CP modüllerinden biri üzerinden "Çevrimiçi yedekleme" sayfasını açtıysanız, bir yedekleme yürütebilirsiniz, ancak geri yükleme yapamazsınız.

NOT

Yedekleme ve geri yükleme fonksiyonlarını STEP 7'de (Sayfa 1154) gerçekleştirebilirsiniz. Bu konuların tam açıklamasında, hangi verileri yedekleyebileceğinizi ve geri yükleyebileceğinizi anlatılmaktadır. SIMATIC Automation Tool (SAT) yedekleme ve geri yükleme fonksiyonlarını da sağlar.

Web sunucusu üzerinden dosyaları yedeklediğinizde, PC'niz veya cihazınız yedekleme dosyalarını standart olarak İndirilenler klasörüne saklar. Dosyaları STEP 7 üzerinden yedeklemeniz durumunda, STEP 7, bu dosyaları STEP 7 projesine saklar. STEP 7 yedekleme dosyalarını Web sunucusu üzerinden ve Web sunucusu yedekleme dosyalarını STEP 7 üzerinden geri yükleyemezsiniz. STEP 7 yedekleme dosyalarını, doğrudan PC'nizin veya cihazınızın indirme klasörüne saklayabilirsiniz. Bu yöntemi kullandığınızda, bu dosyaları Web sunucusundan geri yükleyebilirsiniz.



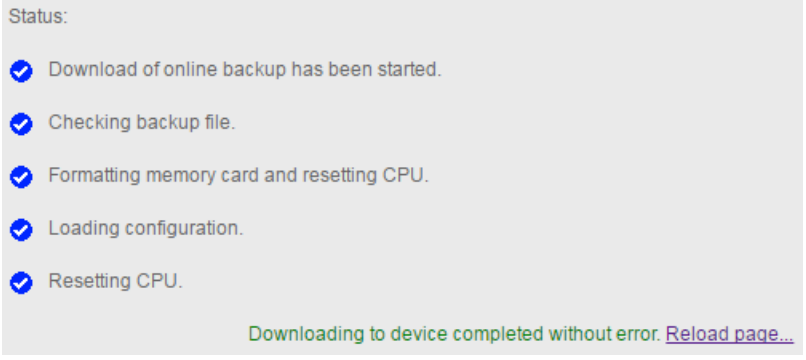
PLC'nin yedeklenmesi

Halen PLC'de saklı olan projenin bir yedekleme kopyasını oluşturmak için, ilgili sayfanın PLC yedeklemesi alanında, "Çevrimiçi yedekleme oluştur" butonuna tıklayın. Bu fonksiyon için, "CPU yedekle" yetkisi (Sayfa 801) gereklidir. Eğer CPU RUN işletim durumundaysa ve onu STOP işletim durumuna değiştirmeniz gerekiyorsa, "İşletim durumunu değiştir" yetkisine de gerek duyarsınız. PC veya cihaz, yedekleme dosyasını, indirmeler için olan standart saklama konumuna saklar. Tarayıcınıza ve cihaz ayarlarınıza bağlı olarak, dosyayı saklamak için onaylamanız gerekebilir.

PLC'nin geri yüklenmesi

Daha önce kaydedilmiş bir yedekleme dosyasını seçmek için, ilgili sayfanın PLC'yi geri yükle alanında, Web sunucusu için olan kullanıcı şifresini girin ve "Gözat" veya "Dosya seç" butonuna tıklayın (tarayıcınıza göre değişir). "Çevrimiçi yedeklemeyi yükle" butonuna tıklayın ve bu dosyayı bağlı PLC'ye yüklemek için gelen mesajı onaylayın. Bu sayfa için, "CPU'yu geri yükle" kullanıcı yetkisine (Sayfa 801) gerek duyarsınız. Eğer CPU RUN işletim durumundaysa ve onu STOP işletim durumuna değiştirmeniz gerekiyorsa, "İşletim durumunu değiştir" yetkisine de gerek duyarsınız.

Geri yükleme işlemi sırasında bazı ilerleme mesajları görüntülenir ve kullanıcı adınızı ve şifrenizi yeniden girmeniz istenir. Başarıyla tamamlanan her adımdan sonra, aşağıdaki tamamlama mesajları ve sayfayı yenilemek için bir link görüntülenir:



⚠ İKAZ

İçeriği bilinmeyen yedekleme kopyalarının geri yüklenmesi

İçeriği bilinmeyen bir yedekleme kopyasını geri yüklemeniz, hatalı fonksiyonlarda veya program hatalarında ciddi maddi hasarlara veya kişisel yaralanmalara neden olabilir.

Bundan başka, CPU'nun cihaz konfigürasyonunda Web sunucusunun etkinleştirilmediği bir yedekleme kopyasını geri yüklerseniz, Web sunucusu üzerinden CPU'ya erişemezsiniz.

Yedekleme kopyasının, bilinen bir konfigürasyon içerdiğinden emin olun.

NOT

CPU'nun IP adresinin farklı olduğu bir yedeklemenin geri yüklenmesi

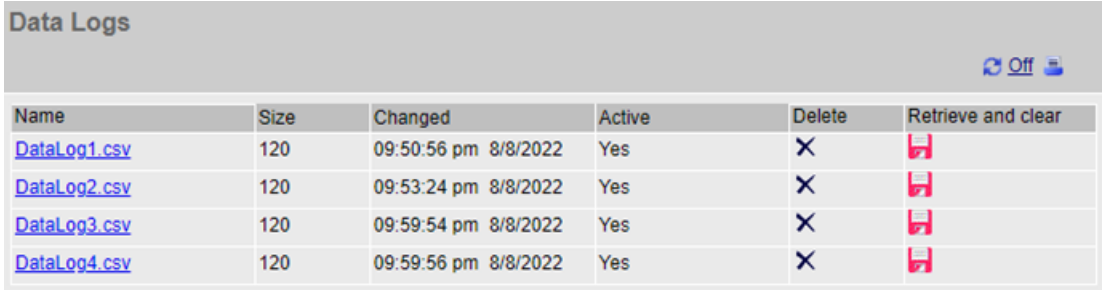
Yedekleme kopyasındaki CPU'nun IP adresinin, güncel CPU'nun IP adresinden farklı olduğu bir yedeklemeyi geri yüklemeyi denerseniz, Web sunucusu geri yüklemenin tamamlandığı mesajını görüntüleyemez. "CPU'yu geri yükle" mesajı beş dakikadan daha uzun süre görüntülenirse, yedekleme dosyasındaki adresle eşleşen yeni IP adresini girin. Bunun ardından CPU bu adresi içerir ve Web sunucusuna erişebilirsiniz.

12.7.13 Veri protokolleri

"Veri protokolleri" sitesinden yararlanarak, veri protokolleriyle etkileşime geçebilirsiniz. Yapacağınız işlemlerin türü, sahip olduğunuz kullanıcı yetkilerine (Sayfa 801) bağlıdır. Dosyaları okuma yetkisine sahip tüm kullanıcılar dosyaları görüntüleyebilir. Dosyaları okuma ve yazma yetkiniz varsa, ayrıca şunları da yapabilirsiniz:

- PLC'nizden bilgisayarınıza bir veri protokolü yükleyebilir
- PLC'nizden bir veri protokolünü silebilir
- Bir veri protokolünü çağırır ve silebilirsiniz

Veri protokolleri, büyük/küçük harfe duyarlı ve artan alfabetik sırada görüntülenir. Veri protokolleri listesi, her biri 50 veri protokolü içeren sayfalara bölünmüştür.



Name	Size	Changed	Active	Delete	Retrieve and clear
DataLog1.csv	120	09:50:56 pm 8/8/2022	Yes	✕	
DataLog2.csv	120	09:53:24 pm 8/8/2022	Yes	✕	
DataLog3.csv	120	09:59:54 pm 8/8/2022	Yes	✕	
DataLog4.csv	120	09:59:56 pm 8/8/2022	Yes	✕	

NOT

Veri protokollerinin yönetimi

Bir dosya sistemine maksimum 1000 veri protokolü kaydedin. Bu sayı aşıldığında, Web sunucusu, veri protokollerini görüntülemek için artık yeterli CPU kaynağına sahip olamayabilir.

Veri protokolü web sitesinde veri protokollerinin görüntülenemediğini fark ederseniz, veri protokollerini görüntülemek ve silmek için CPU'yu STOP işletim durumuna getirmeniz gerekir.

Veri protokollerinizin yönetimi aracılığıyla, sadece gereken sayıda veri protokolü kaydettiğinizden ve 1000 veri protokollük maksimum sayıyı aşmadığınızdan emin olun.

Excel'de bir veri protokolüyle çalışma

Protokol dosyası, Amerikan/İngiliz CSV (virgülle ayrılmış değerler) formatındadır. Amerikan/İngiliz olmayan sistemlerde bu formatı Excel'de açmak için yapmanız gerekenler: Veri protokollerinin CSV formatında Microsoft Excel'in ABD/İngilizce dışındaki sürümlerine içe aktarılması (Sayfa 887).

Aktif durum

İlgili dosya için CPU'da bir veri protokolü kumanda bloğu varsa, veri protokolü sayfasındaki "Aktif" sütununda "Evet" görüntülenir. Bu blok mevcut değilse, "Hayır" görüntülenir. Aktif durum, kullanıcı tarafından manuel olarak değiştirilemez.

Bir veri protokolü açıkken veya STEP 7 programında bir veri protokolüne yazarken, Web sunucusu veri protokolü dosyasını silemez, yükleyemez, çağırıp silemez. Buna ek olarak, Web sunucusu yükle komutunu veya çağır ve sil komutunu kullanarak bir veri protokolü yüklerken, yükleme işlemi tamamlanana veya iptal edilene dek başka bir veri protokolü işlemi gerçekleştiremezsiniz. Bu durumda Web sunucusu, bir "Uygulama meşgul" hata mesajı görüntüler.

Veri protokolü dosyasının indirilmesi

Dosya adına tıklayarak bir veri protokolü dosyasını indirebilirsiniz. Dosya artık mevcut değilse veya başka bir yükleme işlemi gerçekleşiyorsa, Web sunucusu bir hata mesajı görüntüler. Veri protokolü sayfasını yeniden yüklemenize kadar bu hata mesajı sayfada kalır. Aşağıdaki işlemler, Web sunucusunun veri protokolü sayfasını yeniden yüklemesine yol açar:

- Veri protokolü sayfasını yenilenmesi veya veri protokolü sayfasından çıkıp, sayfaya geri dönülmesi
- Bir sayfada görüntülenen veri protokolü sayısının değiştirilmesi
- Bir veri protokolünün başarıyla silinmesi
- Bir veri protokolünün başarıyla çağrılması ve silinmesi

NOT

Veri protokollerine ilişkin hata mesajı

Web sunucusunun otomatik olarak güncellenmesi, hata mesajını sayfadan kaldırmaz.

Aşağıdaki senaryolar, veri protokolü sayfasında hata mesajlarına yol açar:

İşlem	Hata koşulu	Hata mesajı
İndir Çağır/sil	• Dosya mevcut değil • Geçersiz dosya adı • Geçersiz HTTP talep yöntemi • SMC yazma korumalı (yalnızca çağır/sil)	Dosya indirilirken hata oluştu
Sil	• Geçersiz HTTP talep yöntemi • Geçersiz dosya adı • Dosya mevcut değil	Dosya silinirken hata oluştu
Sil	• SMC yazma korumalıdır	Dosya silinirken hata oluştu: Memory card yazma korumalıdır
Sil	• Kullanıcı programı, veri protokolü dosyasını açık tutuyor	Uygulama meşgul
İndir Sil Çağır/sil	• Gerekli yetki eksik • HTTP talep yöntemi, POST veya GET değil • URL'de geçersiz ACTION parametresi	Dosya işlemine izin verilmiyor

İşlem	Hata koşulu	Hata mesajı
İndir Sil Çağır/sil	• Geçersiz veya eksik HTTP yönlendiren alanı	Dosya işlemine izin verilmiyor yönlendiren yok
İndir Sil Çağır/sil	• Kullanıcı programı, veri protokolü dosyasını açık tutuyor • Bir veri protokolünün yüklenmesi şu an devam ediyor	Uygulama meşgul
İndir Sil Çağır/sil	• Beklenmeyen dahili PLC hatası	Dahili hata

Bir veri protokolünün silinmesi

STEP 7 programında açık olan bir veri protokolü silinemez. Silmeden önce veri protokolünü kapatmanız gerekir. Bir veri protokolünün "Sil" sütunundaki ✕ sembole tıklayarak bir veri protokolünü silebilirsiniz. Veri protokolü dosyasını silmek için, silme işlemini "Sil" iletişim alanında onaylayın.

Veri protokolünü çağır ve sil

Bir veri protokolünü açmak ve tüm veri girişlerini kaldırmak için, Çağır ve sil sembolüne tıklayın. Veri protokolü dosyasını çağırmak ve içeriğini silmek için, bu işlemi Çağır ve sil veri protokolü dosyası iletişim alanında onaylayın.

İşlemi onayladıktan sonra, Web sunucusu veri protokolü dosyasının içeriğini indirmenizi sağlar. "Dosyayı sakla" iletişim alanında, veri protokolünün kaydedilip kaydedilmeyeceğini seçebilirsiniz. Seçiminizden sonra, Web sunucusu dosyayı silmeksizin veri protokolü dosyasının içeriğini kaldırır. "Dosyayı sakla" iletişim alanında veri protokolünü silme işlemini iptal edemezsiniz. Bu işlemi yalnızca "Çağır ve sil" iletişim alanında başlangıçtaki onay ile iptal edebilirsiniz.

Bir veri protokolünü yalnızca, ilgili "Aktif" sütununda bir "Hayır" mevcutsa çağırabilir ve silebilirsiniz.

Bir hata mesajı alırsanız veya "Çağır ve sil" sembolü görüntülenmiyorsa, verileri kaldırmak için dosyayı manuel olarak silmeniz gerekecektir.

12.7.14 Kullanıcı dosyaları

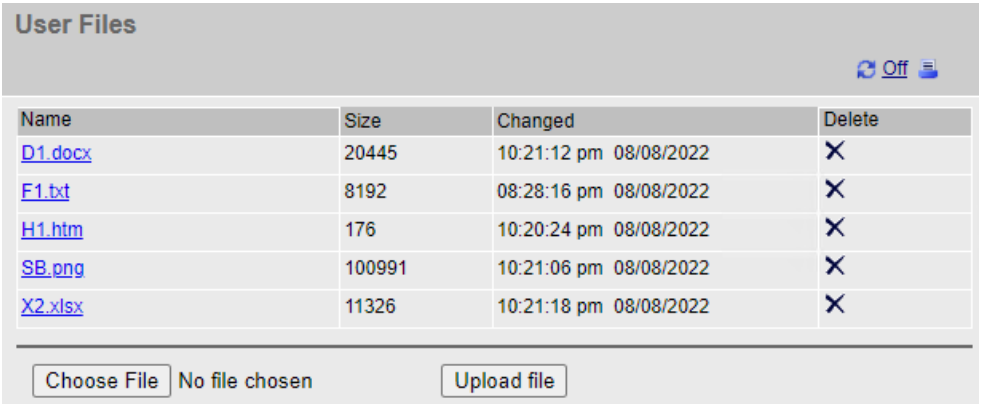
"Kullanıcı dosyaları" sayfasında, SIMATIC memory card'daki (harici yükleme belleği) dosyalara erişebilirsiniz.

Kullanıcı dosyalarına erişim türünüz, kullanıcı yetkilerinize (Sayfa 801) bağlıdır. Dosyaları okuma yetkisine sahip tüm kullanıcılar "Kullanıcı dosyaları" sayfasındaki dosya ve klasörleri görüntüleyebilir. Dosyaları okuma ve yazma yetkiniz varsa, ayrıca şunları da yapabilirsiniz:

- PLC'nizden bilgisayarınıza bir kullanıcı dosyası indirebilirsiniz
- Bilgisayarınızdan PLC'nize bir kullanıcı dosyasını karşıya yükleyebilirsiniz
- PLC'nizden bir kullanıcı dosyasını silebilirsiniz

Kullanıcı dosyalarının görüntülenmesi

Kullanıcı dosyalarını görüntülemek için ana gezinme sayfasında "Kullanıcı dosyaları" öğesine tıklayın.



Dosya listesi, dosyanın güncel boyutunu ve en son değiştirildiği tarihi içerir. Dosya listesi, SIMATIC memory card'ın kök dizinindeki "Kullanıcı dosyaları" dizininden derlenir.

NOT

Kullanıcı dosyası yönetimi

SIMATIC memory card'a en fazla 1.000 kullanıcı dosyası kaydedin. Daha fazla dosya mevcutsa, Web sunucusundaki yalnızca ilk 1000 dosyayı görüntüleyebilirsiniz. Dosyalar artan alfabetik sırada ve büyük/küçük harfe duyarlı olarak görüntülenir.

Kullanıcı dosyaları listesinde sayfa bölüşümü

Kullanıcı dosyaları listesinde, her sayfada 50 giriş görüntülenir. Açılır listeden görüntülemek istediğiniz alanı seçebilirsiniz.



Kullanıcı dosyasının yüklenmesi

Bir kullanıcı dosyası yüklemek için listedeki bir dosyaya tıklayın. Windows Gezgininde, dosyayı seçtiğiniz bir klasöre kaydedin.

Bir kullanıcı dosyasının silinmesi

Bir kullanıcı dosyasını silmek için dosyanın X sembolüne tıklayın ve talebi kabul edin.

Kullanıcı dosyasının karşıya yüklenmesi

Programlama cihazınızdan memory card'ınızdaki kullanıcı dosyalarına bir dosya yüklemek için aşağıdaki yöntemi izleyin:

1. "Dosya seç" öğesine tıklayın.
2. Windows Gezgininde bir dosya seçin.
   Dosya 2 GB'tan küçük olmalıdır.
   Dosya adının yalnızca şu karakterleri içermesine izin verilir: a-z A-Z 0-9 ._- (){}[]$!=~ (boşluk)
3. "Dosyayı karşıya yükle" öğesine tıklayın.

Kullanıcı dosyaları listesinin yazdırılması

"Kullanıcı dosyaları" web sitenizdeki "Yazdır" simgesine tıklayarak SIMATIC memory card'ınızdaki kullanıcı dosyalarının listesini yazdırabilirsiniz.

Kullanıcı dosyalarında hata

"Kullanıcı dosyaları" sayfası, bir işlemin yürütülmesi başarısız olduğunda bir hata mesajı görüntüler:



"Kullanıcı dosyaları" sayfasını yenileme işlemini yapıncaya kadar hata mesajı sayfada kalır.

NOT

Kullanıcı dosyalarıyla ilgili hata mesajı

Web sunucusunun otomatik olarak güncellenmesi, "Kullanıcı dosyaları" sayfasındaki bir hata mesajını kaldırmaz.

"Kullanıcı dosyaları" sayfasındaki hata mesajlarına şunlar neden olur:

İşlem	Hata koşulu	Hata mesajı
Kullanıcı dosyasının indirilmesi	• Dosya mevcut değil • Dahili PLC hatası	Dosya indirilirken hata oluştu
Kullanıcı dosyasının indirilmesi Kullanıcı dosyasının karşıya yüklenmesi Kullanıcı dosyasının silinmesi	• İstenen işlem için yetkiler yeterli değil • Geçersiz HTTP talep yöntemi veya geçersiz eylem parametresi	Dosya işlemine izin verilmiyor
Kullanıcı dosyasının indirilmesi Kullanıcı dosyasının karşıya yüklenmesi Kullanıcı dosyasının silinmesi	• Geçersiz veya eksik HTTP yönlendiren alanı	Dosya işlemine izin verilmiyor - Yönlendiren yok
Kullanıcı dosyasının silinmesi	• Memory card yazma korumalıdır	Dosya işlemine izin verilmiyor
İndir Sil Çağır/sil	• Dosya mevcut değil • Dahili PLC hatası	Dosya silinirken hata oluştu - Memory card yazma korumalı
Kullanıcı dosyasının karşıya yüklenmesi	• Dosya adı eksik veya geçersiz • Dahili PLC hatası	Dosya karşıya yüklenirken hata oluştu
Kullanıcı dosyasının karşıya yüklenmesi	• Dosya adı, PLC'de halen mevcut	Dosya karşıya yüklenirken hata - Ad, halen mevcut
Kullanıcı dosyasının karşıya yüklenmesi	• SMC dolu	Dosya karşıya yüklenirken hata - Memory card dolu
Kullanıcı dosyasının karşıya yüklenmesi	• Geçersiz dosya adı	Dosya karşıya yüklenirken hata - Dosya adında geçersiz karakter
Kullanıcı dosyasının karşıya yüklenmesi	• Dosya, SM dosya sistemi için çok büyük	Dosya karşıya yüklenirken hata oluştu - Dosya çok büyük
Kullanıcı dosyasının karşıya yüklenmesi	• SMC yazma korumalıdır	Dosya karşıya yüklenirken hata - Memory card yazma korumalı

12.7.15 Veri protokolleri ve kullanıcı dosyaları için API

Veri protokolleri ve kullanıcı dosyaları için API

S7-1200 için, veri protokolleri ve kullanıcı dosyalarına yönelik bir API fonksiyonu vardır. Daha fazla bilgiyi, *S7-1500, ET200SP, ET200pro Web sunucusu işletme kitabında* bulabilirsiniz.

12.7.16 Dosya tarayıcısı

Dosya tarayıcısı sayfası, CPU'nun dahili yükleme belleğindeki ve memory card'daki (harici yükleme belleği) dosyalara erişim sağlar. Dosya tarayıcısı sayfası, ilk önce "Recipes" ve "DataLogs" klasörlerini içeren yükleme belleğinin kök klasörünü gösterir, ancak bir memory card kullanıyorsanız, oluşturduğunuz diğer klasörleri de gösterir.

Dosyalara ve klasörlere erişim türünüz, kullanıcı yetkilerinize (Sayfa 801) bağlıdır. Dosyaları okuma yetkisine sahip tüm kullanıcılar, dosya ve klasörleri dosya tarayıcısında görüntüleyebilir. Oturum açma yetkileriniz olursa olsun, "Recipes" veya "DataLogs" klasörünü silemezsiniz. Ancak memory card'da kullanıcıya özgü klasörler oluşturduysanız, dosya yazma ve silme yetkilerinizle oturum açmış olmanız koşuluyla bu klasörleri silebilirsiniz.

Klasördeki münferit dosyalara erişim için bir klasöre tıklayın.



Reçete dosyaları

Yükleme belleğinde bulunan tüm reçete dosyaları, reçete klasöründe görüntülenir. Reçete dosyaları da CSV formatındadır ve Microsoft Excel veya başka bir programda açılabilir. Reçete dosyalarını silmek, değiştirmek ve saklamak, yeniden adlandırmak veya yüklemek için farklı yetkilere ihtiyacınız vardır.

Dosyaların karşıya yüklenmesi ve sayfaları otomatik olarak güncellenmesi

Bir dosyayı karşıya yüklemeye başladığınızda, dosya tarayıcısı sayfasında kaldığınız sürece dosya yükleme işlemi devam edecektir. Otomatik Web sunucusu sayfası yenilemesini on saniyelik aralıklı olarak etkinleştirdiyseniz, her sayfa yenilemede dosya yükleme işleminin ilerlemesini görüntülenir. Örneğin 2 MB'lik bir dosyayı karşıya yüklediğinizde, 2500, 5000, 10.000, 15.000 ve 20.000 Bayt yükleme işleminde dosya boyutu güncellemelerini görürsünüz.

Yükleme işlemi tamamlanmadan önce dosya tarayıcısı sayfasından çıktığınızda ise, Web sunucusu tamamlanmamış eksik dosyayı siler.

Daha fazla bilgi

NOT

Dosya adlandırma kuralları

Web sunucusunun veri protokolü ve reçete dosyalarıyla çalışması sağlamak adına, ASCII karakter setindeki karakterleri kullanmalısınız. \ / : * ? " < > | karakterlerine ve boşluk karakterine izin verilmez.

Dosyalarınız bu adlandırma kuralı ile uyumlu değilse, Web sunucusu örneğin dosyaların karşıya yüklenmesi, silinmesi veya yeniden adlandırılması gibi işlemler sırasında hata verebilir. Harici yükleme belleğinden dosyaları yeniden adlandırmak için bir kart okuyucuya ve Windows dosya gezginine gerek duyarsınız.

Veri protokolü yordamları ile programlama ve reçetelerin içe aktarımı (Sayfa 455) ve dışa aktarımı (Sayfa 454) hakkında daha fazla bilgiyi, Reçeteler ve veri protokolleri (Sayfa 449) bölümünde bulabilirsiniz.

Ayrıca bakınız

Veri protokollerinin CSV formatında Microsoft Excel'in ABD/İngilizce dışındaki sürümlerine içe aktarılması (Sayfa 887)

12.8 Kullanıcı tanımlı web siteleri

S7-1200 Web sunucusu için hedef sistem verileri ile kendiniz uygulamaya özel HTML sayfalarınızı da oluşturabilirsiniz.

⚠ İKAZ

Kullanıcı tanımlı web siteleri üzerinden CPU'ya yetkisiz erişim

Kullanıcı tanımlı web siteleri üzerinden CPU'ya yetkisiz erişim, proses işletimini sekteye uğratabilir ve bu, ölüme veya ciddi yaralanmalara ve/veya maddi hasara neden olabilir.

Kullanıcı tanımlı web sitelerinin güvenli olmayan biçimde kodlanması, güvenlik açıklarının oluşmasına ve böylece Cross Site Scripting (XSS - Siteler arası komut dizisi çalıştırma), Code Injection (Kod enjeksiyonu) veya diğer yöntemlerin uygulanabileceği saldırılara neden olabilir.

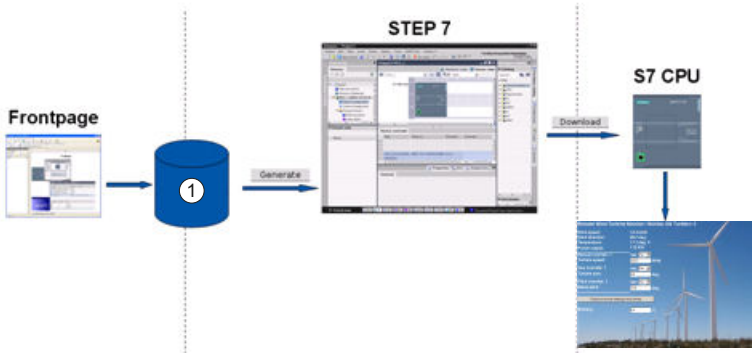
Güvenli kurulum uygulayarak S7 1200 CPU'nuzu yetkisiz erişime karşı koruyun. Bu konudaki gerekli bilgileri, Industrial Security web sitesinde (http://www.siemens.com/industrialsecurity) "Operational Guidelines" adlı bölümde bulabilirsiniz.

Seçtiğiniz bir HTML editörü ile kullanıcı tanımlı web siteleri oluşturabilir ve bu siteleri, standart web siteleri menüsünden çağırılabilecek şekilde CPU'ya yükleyebilirsiniz. Bunun için şu eylemler gerçekleştirilmelidir:

- Microsoft FrontPage gibi bir HTML editöründe HTML sayfaları oluşturun (Sayfa 841)
- HTML kodundaki HTML yorumlarına AWP komutları ekleyin (Sayfa 842): AWP komutları, CPU bilgilerine erişim için Siemens tarafından sağlanan sabit bir komut setidir.
- HTML sayfalarını okumak ve işlemek için STEP 7'yi yapılandırın (Sayfa 857)
- HTML sayfalarından bloklar oluşturun (Sayfa 857)

- HTML sayfalarının kullanımına kumanda etmek için STEP 7'yi programlayın (Sayfa 859)
- Blokları derleyin ve CPU'ya yükleyin (Sayfa 861)
- PC'niz üzerinden kullanıcı tanımlı web sitelerine erişin (Sayfa 861)

Bu proses aşağıda gösterilmiştir:



① Gömülü AWP komutları içeren HTML dosyaları

Ayrıca bakınız

Web sunucusu (Sayfa 796)

12.8.1 HTML sayfalarının oluşturulması

Web sunucusu için kendi HTML sayfalarınızı oluşturmak üzere seçtiğiniz yazılım paketini kullanabilirsiniz. HTML kodunuzun, W3C (World Wide Web Consortium) HTML standartları ile uyumlu olmasını sağlayın. STEP 7, HTML sözdiziminizi kontrol etmez.

WYSIWYG veya Design Layout modunda programlamanızı sağlayan yazılımları kullanabilirsiniz, ancak HTML kodunuzu salt HTML formatında düzenleyebilmeniz gerekir. Çoğu web geliştirme aracı bu tür bir düzenleme olanağı sağlar. Aksi takdirde HTML kodunu bir metin editöründe düzenleyebilirsiniz. UTF-8'i sayfanın karakter seti olarak belirlemek için, HTML sayfanıza aşağıdaki satırı ekleyin:

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

Ayrıca dosyayı, editörde yine aynı şekilde UTF-8 karakter kodlamasıyla saklamalısınız:

STEP 7'de, HTML sayfalarınızın içeriğini STEP 7 veri bloklarında derleyebilirsiniz. Bu veri blokları, web sitelerinin görüntülenmesini düzenleyen bir kumanda veri bloğundan ve derlenmiş web siteleriyle birlikte bir veya daha fazla veri bloğu fragmentinden oluşur. Kapsamalı HTML sayfalarının, özellikle çok sayıda resim içeren sayfaların, yükleme belleğinde (Sayfa 862) DB fragmentleri için önemli miktarda yer kapladığını dikkate alın. CPU'nuzun dahili yükleme belleği, kullanıcı tanımlı web siteleriniz için yeterli değilse, Memory Card (Sayfa 111)'ı harici yükleme belleği olarak kullanılabilirsiniz.

S7-1200'den gelen verileri kullanmak istediğinizde, AWP komutlarını (Sayfa 842) HTML yorumları olarak ekleyerek HTML kodunuzu programlayabilirsiniz. Son olarak, HTML sayfalarınızı PC'nize saklayın ve saklama yolunu not edin.

NOT

AWP komutlarını içeren HTML dosyalarının maksimum dosya boyutu 64 kB'dir. STEP 7'nin sayfalarınızı doğru bir şekilde derleyebilmesi için dosyalarınızın boyutu bu limitin altında kalmalıdır.

Siemens, tüm web kaynak dosyalarının (CCC dosyaları, resimler, JavaScript dosyaları ve HTML dosyaları) maksimum 512 kB boyutunda oluşturulmasını önerir. Aksi takdirde dosyayı Web sunucusundan tarayıcıya gönderirken problemler ortaya çıkabilir. Her web kaynağı dosyasının boyutunu klasörün dosya gezgininde görebilirsiniz.

Kullanıcı tanımlı web sitelerinin güncellenmesi

Kullanıcı tanımlı web siteleri otomatik olarak yenilenmez. Sayfayı yenilenmesi veya yenilenmemesi için HTML kodunu programlayabilirsiniz. PLC verilerini görüntüleyen sayfalarda veriler düzenli yenilemelerle güncel tutulur. Veri girişi için form olarak kullanılan HTML sayfalarında yenileme işlemi, kullanıcının veri girişini etkileyebilir. Sitenizin tamamının otomatik olarak yenilenmesini isterseniz, HTML başlığınıza bu başlık satırını ekleyebilirsiniz; burada "10", yenileme işlemleri için olan saniye sayısını temsil eder:

```
<meta http-equiv="Refresh" content="10">
```

Sayfaları ve verileri güncellemek için JavaScript veya diğer HTML tekniklerini de kullanabilirsiniz. Bunun için lütfen HTML veya JavaScript dokümantasyonuna başvurun.

12.8.2 S7-1200 Web sunucusu tarafından desteklenen AWP komutları

S7-1200 Web sunucusu, kullanıcı tanımlı web sitelerinize aşağıdaki amaçlar doğrultusunda HTML komutları olarak ekleyebileceğiniz AWP komutları sağlar:

- Değişkenleri oku (Sayfa 844)
- Değişkenleri yaz (Sayfa 845)
- Özel değişkenlerin okunması (Sayfa 847)
- Özel değişkenlerin yazılması (Sayfa 849)
- Enum tiplerinin tanımlanması (Sayfa 851)
- Enum tiplerine değişkenlerin atanması (Sayfa 851)
- Veri bloğu fragmentlerinin oluşturulması (Sayfa 853)

Genel sözdizimi

Bir değişkeni okuma komutu hariç diğer AWP komutları aşağıdaki sözdizimine sahiptir:

```
<!-- AWP_ <Komut adı ve parametreler> -->
```

Değişkenleri CPU'ya yazmak için tipik HTML form komutlarıyla birlikte AWP komutlarını kullanırsınız.

Aşağıdaki sayfalarda yer alan AWP komutlarının açıklamalarında şu adlandırma kuralları kullanılır:

- Köşeli parantez [ ] içinde belirtilen elemanlar isteğe bağlıdır.
- Sivri parantez < > içinde gösterilen elemanlar, belirtilmesi istenen parametre değerleridir.
- Soru işaretleri, komutun gerçek bir parçasıdır. Görüntülendikleri gibi mevcut olmalıdırlar.
- Kullanımlarına bağlı olarak, değişken veya veri bloğu adlarındaki özel karakterler bir Escape karakter dizisi ile işaretlenmeli veya tırnak içine (Sayfa 855) alınmalıdır.

AWP komutlarını bir metin editörü veya HTML düzenleme modu kullanarak web sitelerinize yapıştırın.

NOT

AWP komutlarının beklenen sözdizimi

Bir AWP komutunu formüle ederken "<!--" öğesinden sonraki boşluk ve "-->" öğesinden önceki boşluk, komutun doğru derlenmesi için çok önemlidir. Boşlukların atlanması, doğru kodun yanlış derlenmesine neden olabilir. Bu durumda, herhangi bir hata mesajı görüntülenmez.

AWP komutlarına genel bakış

Münferit AWP komutlarının daha ayrıntılı açıklamalarını, sonraki alt bölümlerde bulabilirsiniz, burada öncelikle komutlara bir genel bakış veriyoruz:

Değişkenleri oku

```
:=<Varname>:
```

Değişkenleri yaz

```
<!-- AWP_In_Variable Name='<Varname1>' [Use='<Varname2>'] ... -->
```

Bu AWP komutu yalnızca, ad yan tümcesi değişkenine yazma işleminin mümkün olduğunu deklare eder. HTML kodunuz, <input>, <select> veya başka bir HTML yordamı ile bir HTML formülü içindeki değişkene ad yazma işlemleri gerçekleştirir.

Özel değişkenlerin okunması

```
<!-- AWP_Out_Variable Name='<Typ>:<Name>' [Use='<Varname>'] -->
```

Özel değişkenlerin yazılması

```
<!-- AWP_In_Variable Name='<Typ>:<Name>' [Use='<Varname>']-->
```

Enum tiplerinin tanımlanması

```
<!-- AWP_Enum_Def Name='<Name Enum-Typ>' Values='<Değer>, <Değer>,-
.. ' -->
```

Enum tiplerinin tanımlanması

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<Name Enum-Typ>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<Name Enum-Typ>" -->
```

Fragmentlerin oluşturulması

```
<!-- AWP_Start_Fragment Name='<Name>' [Type=<Typ>][ID=<ID>] -->
```

Fragmentlerin içe aktarımı

```
<!-- AWP_Import_Fragment Name='<Adı>' -->
```

12.8.2.1 Değişkenleri oku

Değişkenleri "HMI'dan erişilebilir" olarak yapılandırmış olmanız koşuluyla, kullanıcı tanımlı web siteleri CPU'dan değişkenleri (PLC değişkenleri) ve veri bloğu değişkenlerini okuyabilir.

Sözdizimi

```
:=<Varname>:
```

Parametre

<Varname>	Okunacak değişken, STEP 7 programınızdan bir PLC değişkeni, bir veri bloğu değişkeni, G/Ç veya bellekteki bir adres olabilir. Bellek ve G/Ç adreslerinde veya Takma adlarda (Sayfa 855), değişken adını çift tırnak içine almayın. PLC değişkenlerinde, değişken adını çift tırnak içine alın. Veri bloğu değişkenlerinde ise blok adını yalnızca çift tırnak içine alın. Değişken adı, tırnak işaretlerinin dışındadır. Veri bloğunun numarası yerine veri bloğunun adını kullanmayı unutmayın. Array elemanlara referanslarda, Array elemanları sözdizimini kullanın.

Örnekler

```
:="Konveyör hızı":
:="Benim_veri_bloğum".Markör 1:
:=I0.0:
:=MW100:
:="Benim_veri_bloğum".Array_Dim1[0]:
:="Benim_veri_bloğum".Array_Dim2[0,0]:
```

Takma ad kullanarak belirtilen bir değişkeni okuma örneği

```
<!-- AWP_Out_Variable Name='Markör1' Use='"Benim_veri bloğum".flag1'
-->
:=Markör1:
```

NOT

PLC değişkenleri ve veri bloğu değişkenleri için diğer takma adları nasıl tanımlayacağınız Değişken referanslama için takma ad kullanımı (Sayfa 850) altında açıklanmıştır.

Bir değişken adı veya veri bloğu adı özel karakterler içeriyorsa, ek olarak tırnak işaretleri veya Escape karakterleri kullanmalısınız. Bu, Özel karakterli değişken adlarının işlenmesi (Sayfa 855) altında açıklanmıştır.

12.8.2.2 Değişkenleri yaz

Kullanıcı tanımlı web siteleri CPU'ya yazabilir. Bunun için, bir AWP komutu ile HTML sayfasının yazılacağı CPU'da bir değişken belirtin. Bu değişken, PLC değişkeni adı veya veri bloğu değişkeni adı kullanılarak belirtilmelidir. Tek bir yordam ile birden çok değişken adı deklare edebilirsiniz. Verileri CPU'ya yazmak için standart HTTP POST komutunu kullanın.

Bu bağlamda tipik kullanım, HTML sayfanızda yazılabilir CPU değişkenlerine ilişkin metin giriş alanları veya liste seçim alanları içeren bir form düzenlemektir. Tüm kullanıcı tanımlı web sitelerinde olduğu gibi, STEP 7 programınıza dahil edilmek üzere STEP 7'de veri blokları oluşturursunuz. Değişkenleri değiştirme yetkisiyle donatılmış bir kullanıcı, daha sonra bu sayfayı ziyaret edip veri giriş alanlarına veriler girdiğinde veya bir seçim listesinden bir alan seçtiğinde, Web sunucusu bu veri girişlerini değişken için uygun veri tipine dönüştürür ve ilgili değeri CPU'daki değişkene yazar. HTML veri giriş alanları ve HTML seçim listeleri için olan ad yan tümcesinin AWP_In_Variable komutunun ad yan tümcesi için tipik olan bir sözdizimi kullandığını dikkate alın. Genellikle bu adı tek tırnak içine ve bir veri bloğuna referansta bulunurken veri bloğu adını çift tırnak içine alırsınız.

Form yönetimi ile ilgili daha fazla bilgi için, HTML dokümantasyonuna başvurun.

Sözdizimi

```
<!-- AWP_In_Variable Name='<Varname1>' [Use='<Varname2>'] ... -->
```

Parametre

<Varname1>	Use yan tümcesi belirtilmemişse, Varname1 yazılacak değişkendir. Bu bağlamda değişken, STEP 7 programınızdan bir PLC değişkeni, belirli bir veri bloğundan bir değişken veya bir veri bloğu adı olabilir. Bir Use yan tümcesi belirtilmişse, Varname1, <Varname2> içinde referanslanan değişken için alternatif bir addır (Sayfa 850). Buradaki ad, HTML sayfasındaki bir yerel addır.
<Varname2>	Bir Use yan tümcesi belirtilmişse, varname2 yazılacak değişkendir. Bu bağlamda değişken, STEP 7 programınızdan bir PLC değişkeni veya belirli bir veri bloğundan bir değişken olabilir.

Hem ad hem de kullanım yan tümceleri için, tam ad tek tırnak içine alınmalıdır. Bir PLC değişkenini tek tırnak ve bir veri bloğu adını ise çift tırnak içine alın. Veri bloğu adı çift tırnak içindedir, ancak veri bloğu değişkeni adı öyle değildir. Veri bloğu değişkenleri için veri bloğunun numarası yerine bloğun adını kullanmayı unutmayın. Array elemanlara referanslarda, Array elemanları sözdizimini kullanın.

Eğer AWP_In_Variable komutuyla bir veri bloğunu yazılabilir yaparsanız, veri bloğundaki her değişken yazılabilir olur.

HTML veri giriş alanları için örnekler

```
<!-- AWP_In_Variable Name='"Hedef kademesi"' -->
<form method="post">
<p>Hedef kademesi girişi: <input name='"Hedef kademesi"' type="text"
/>
</p>
</form>

<!-- AWP_In_Variable Name='"VeriBloğu_1".Frenleme' -->
<form method="post">
<p>Frenleme: <input name='"VeriBloğu_1".Frenleme' type="text" />
%</p>
</form>

<!-- AWP_In_Variable Name='"VeriBloğu_1".Array_Dim2' -->
<form method="post">
<p>İki boyutlu Array-Değeri [2,1]: <input
name='"VeriBloğu_1".Array_Dim2[2,1]' type="text" /> %</p>
</form>
```

Kullanım yan tümceleri için örnekler

```
<!-- AWP_In_Variable Name='"Frenleme"' Use='"VeriBloğu_1".Frenleme'
-->
<form method="post">
<p>Frenleme: <input name='"Frenleme"' type="text" /> %</p>
</form>
```

Düzenlenebilir veri bloğu için örnek

```
<!-- AWP_In_Variable Name='"VeriBloğu_1"' -->
<form method="post">
<p>Frenleme: <input name='"VeriBloğu_1".Frenleme' type="text" /> %
</p>
<p>Türbin hızı: <input name='"VeriBloğu_1".Türbin hızı' size="10"
value='"VeriBloğu_1".Türbin hızı' type="text" />
</p>
</form>
```

Bir HTML seçim listesi için örnek

```
<!-- AWP_In_Variable
Name='"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak'-->
<form method="post">
<select name='"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak'>
<option value=:"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak:>
</option>
<option value=1>Evet</option>
<option value=0>Hayır</option>
</select><input type="submit" value="Ayarları gönder" /></form>
```

NOT

Yalnızca "Kullanıcı tanımlı web sitelerine yaz" yetkisine (Sayfa 801) sahip bir kullanıcının CPU'ya veri yazmasına izin verilir. Web sunucusu, değiştirme yetkisi olmayan bir kullanıcının girdiği komutları yok sayar.

Bir değişken adı veya veri bloğu adı özel karakterler içeriyorsa, ek olarak tırnak işaretleri veya Escape karakterleri kullanmalısınız. Bu, "Özel karakterli değişken adlarının işlenmesi (Sayfa 855)" altında açıklanmıştır.

12.8.2.3 Özel değişkenlerin okunması

Web sunucusu, PLC cihazından değerleri okuma ve bunları HTTP yanıt başlık satırındaki özel değişkenlerde saklama olanağı sunar. Örneğin bir PLC değişkeninden bir yol adını okuyarak, URL'yi HEADER:saklama konumu özel değişkeni aracılığıyla başka bir saklama konumuna yönlendirebilirsiniz.

Sözdizimi

```
<!-- AWP_Out_Variable Name='<Typ>:<Name>' [Use='<Varname>'] -->
```

Parametre

<Tip>	Özel değişkenlerin tipi; mümkün olanlar: HEADER COOKIE_VALUE COOKIE_EXPIRES
<Name>	Tüm HEADER değişken adlarının listesini HTTP dokümantasyonunda bulabilirsiniz. Aşağıda bazı örnekler verilmiştir: Durum: Yanıt kodu Location: Yönlendirme yolu Retry-After: Bir hizmetin, talepte bulunan istemciye sunulmaması sırasında geçebilecek zaman aralığı. COOKIE_VALUE ve COOKIE_EXPIRES tiplerinde, <Name>, belirli bir çerezin adıdır. COOKIE_VALUE:name: Belirtilen çerezin değeri COOKIE_EXPIRES:name: Belirtilen çerezin saniye cinsinden işlem süresi Ad yan tümcesi tek veya çift tırnak içine alınmalıdır. Herhangi bir kullanım yan tümcesi belirtilmezse, özel değişken adı bir PLC değişken adına karşılık gelir. Tam ad yan tümcesini tek tırnak ve PLC değişkenini ise çift tırnak içine alın. Özel değişken adı ve PLC değişken adı tamamen aynı olmalıdır.
<Varname>	Okunacak değişken için PLC değişkeninin veya veri bloğu değişkeninin adı Değişken adı, tek tırnak içine alınmalıdır. Bir PLC değişkenini tek tırnak ve bir veri bloğu adını ise çift tırnak içine alın. Veri bloğu adı çift tırnak içindedir, ancak veri bloğu değişkeni adı öyle değildir. Veri bloğu değişkenleri için veri bloğunun numarası yerine bloğun adını kullanmayı unutmayın.

Bir değişken adı veya veri bloğu adı özel karakterler içeriyorsa, ek olarak tırnak işaretleri veya Escape karakterleri kullanmalısınız. Bu, Özel karakterli değişken adlarının işlenmesi (Sayfa 855) altında açıklanmıştır.

Örnek: Özel değişkenini Use yan tümcesi olmadan oku

```
<!-- AWP_Out_Variable Name='"HEADER:Status"' -->
```

Bu örnekte, "HEADER:Status" HTTP özel değişkeni, "HEADER:Status" PLC değişkeninin değerini alır. Herhangi bir Use yan tümcesi belirtilmemişse, PLC değişken tablosundaki ad, özel değişkenin adıyla tam olarak aynı olmalıdır.

Örnek: Bir Use yan tümcesi ile özel değişkenin okunması

```
<!-- AWP_Out_Variable Name='HEADER:Status' Use='"Durum"' -->
```

Bu örnekte, "HEADER:Status" özel değişkeni, "Status" PLC değişkeninin değerini alır.

12.8.2.4 Özel değişkenlerin yazılması

Web sunucusu, HTTP yanıt başlık satırının özel değişkenlerinden CPU'ya değer yazma olanağı sağlar. Örneğin bir kullanıcı tanımlı web sitesi çerezi, bir sayfaya erişen kullanıcı veya başlık bilgisi hakkında bilgileri STEP 7'de saklayabilirsiniz. Değişkenleri değiştirme yetkisine sahip bir kullanıcı olarak oturum açtıysanız, Web sunucusu size, CPU'ya yazabileceğiniz belirli özel değişkenlere erişim sağlar.

Sözdizimi

```
<!-- AWP_In_Variable Name='<Typ>:<Name>' [Use='<Varname>']-->
```

Parametre

<Tip>	Özel değişkenlerin tipi, mümkün olanlar: HEADER SERVER COOKIE_VALUE
<Name>	Bu örneklerde gösterildiği şekilde yukarıda tanımlanan tipler içindeki özel değişkenler: HEADER:Accept: Kabul edilebilir içerik tipleri HEADER:User-Agent: Talebin geldiği kullanıcı aracısı hakkında bilgi. SERVER:current_user_id: Güncel kullanıcının ID'si; eğer hiçbir kullanıcı yoksa, 0 SERVER:current_user_name: Güncel kullanıcının adı COOKIE_VALUE:<name>: Belirtilen çerezin değeri Ad yan tümcesini tek tırnak içine alın. Herhangi bir kullanım yan tümcesi belirtilmezse, özel değişken adı bir PLC değişken adına karşılık gelir. Tam ad yan tümcesini tek tırnak ve PLC değişkenini ise çift tırnak içine alın. Özel değişken adı, PLC değişkeni adıyla tam olarak aynı olmalıdır. Tüm HEADER değişken adlarının listesini HTTP dokümantasyonunda bulabilirsiniz.
<Varname>	Özel değişkeni yazmak istediğiniz STEP 7 programınızdaki değişken adı. Bu, bir PLC değişkeni veya bir veri bloğu değişkeni olabilir. Değişken adı, tek tırnak içine alınmalıdır. Bir PLC değişkenini tek tırnak ve bir veri bloğu adını ise çift tırnak içine alın. Veri bloğu adı çift tırnak içindedir, ancak veri bloğu değişkeni adı öyle değildir. Veri bloğu değişkenleri için veri bloğunun numarası yerine bloğun adını kullanmayı unutmayın.

Örnekler

```
<!-- AWP_In_Variable Name='"SERVER:current_user_id"' -->
```

Bu örnekte web sitesi, "SERVER:current_user_id" HTTP özel değişkeninin değerini, "SERVER:current_user_id" PLC değişkenine yazmaktadır.

```
<!-- AWP_In_Variable Name=SERVER:current_user_id'
Use='"Kullanıcı-ID'im"' -->
```

Bu örnekte web sitesi, "SERVER:current_user_id" HTTP özel değişkeninin değerini, "Kullanıcı-ID'im" PLC değişkenine yazar.

NOT

Yalnızca değişkenleri değiştirme yetkisine sahip bir kullanıcının CPU'ya veri yazmasına izin verilir. Web sunucusu, değiştirme yetkisi olmayan bir kullanıcının girdiği komutları yok sayar.

Bir değişken adı veya veri bloğu adı özel karakterler içeriyorsa, ek olarak tırnak işaretleri veya Escape karakterleri kullanmalısınız. Bu, "Özel karakterli değişken adlarının işlenmesi (Sayfa 855)" altında açıklanmıştır.

12.8.2.5 Değişken referanslama için takma ad kullanımı

Kullanıcı tanımlı web sitelerinizde In_Variable veya Out_Variable için takma ad kullanabilirsiniz. Örneğin, HTML sayfanızda CPU'dakinden farklı bir sembolik ad kullanabilir veya CPU'daki bir değişkeni özel bir değişkenle denk tutabilirsiniz. AWP kullanım yan tümcesi bu olanağı sağlar.

Sözdizimi

```
<-- AWP_In_Variable Name='<Varname1>' Use='<Varname2>' -->
<-- AWP_Out_Variable Name='<Varname1>' Use='<Varname2>' -->
```

Parametre

<Varname1>	Takma ad veya özel değişken adı Varname1, tek veya çift tırnak içinde belirtilmelidir.
<Varname2>	Takma ad atamak istediğiniz PLC değişkeninin adı. Değişken bir PLC değişkeni, bir veri bloğu değişkeni veya özel bir değişken olabilir. Varname2, tek tırnak içine alınmalıdır. Bir PLC değişkenini tek tırnak içine ve bir veri bloğu adını ise çift tırnak içine alın. Veri bloğu adı çift tırnak içindedir, ancak veri bloğu değişkeni adı öyle değildir. Veri bloğu değişkenleri için veri bloğunun numarası yerine bloğun adını kullanmayı unutmayın.

Örnekler

```
<-- AWP_In_Variable Name='SERVER:current_user_id'
Use='"Data_Block_10".sunucu_kullanıcısı' -->
```

Bu örnekte SERVER:current_user_id özel değişkeni, "Data_Block_10" veri bloğundaki "sunucu_kullanıcısı" değişkenine yazılmaktadır.

```
<-- AWP_Out_Variable Name='Ağırlık'
Use='"Data_Block_10".Kap_verisi.Ağırlık' -->
```

Bu örnekte, Data_Block_10.Tank_data.Weight veri bloğu yapı elemanındaki değere, kullanıcı tanımlı web sitesinin geri kalanında kolayca "Ağırlık" olarak referanslanabilir.

```
<-- AWP_Out_Variable Name='Ağırlık' Use='"Ağırlık_Çiğ süt kabı"' -->
```

Bu örnekte, "Ağırlık_Çiğ süt kabı" PLC değişkenindeki değere, kullanıcı tanımlı web sitesinin geri kalanında kolayca "Ağırlık" olarak referanslanabilir.

Bir değişken adı veya veri bloğu adı özel karakterler içeriyorsa, ek olarak tırnak işaretleri veya Escape karakterleri kullanmalısınız. Bu, Özel karakterli değişken adlarının işlenmesi (Sayfa 855) altında açıklanmıştır.

12.8.2.6 Enum tiplerinin tanımlanması

Enum tiplerini kullanıcı tanımlı sayfalarınızda tanımlayabilir ve elemanları bir AWP komutuna atayabilirsiniz.

Sözdizimi

```
<!-- AWP_Enum_Def Name='<Name Enum-Typ>' Values='<Değer>,
<Değer>,... ' -->
```

Parametre

<Name Enum-Typ>	Numaralandırma tipinin adı, tek veya çift tırnak içine alınır.
<Değer>	<Sabit>:<Name> Sabit, numaralandırma tipinin atanması için olan sayısal değeri tanımlar. Toplam sayı sınırsızdır. Ad, Enum elemanına atanan değerdir.

Enum değeri atamalarının tüm karakter dizisinin tek tırnak içine ve bir Enum tipine yapılan her bir eleman atamasının çift tırnak içine alınacağını unutmayın. Bir Enum tipinin tanımı, kullanıcı tanımlı web siteleri için global geçerliliğe sahiptir. Kullanıcı tanımlı web sitelerinizi Dile özgü klasörlere (Sayfa 876) yerleştirirseniz, Enum tipi tanımı global olarak dil klasöründeki tüm sayfalara uygulanır.

Örnek

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"Alarm yok", 1:"Kap
dolu", 2:"Tank is empty"' -->
```

12.8.2.7 Bir Enum tipi ile CPU değişkenlerine referanslama

Bir Enum tipine CPU'daki bir değişkeni atayabilirsiniz. Bu değişkeni, kullanıcı tanımlı web sitelerinizin herhangi bir yerinde bir okuma işlemi (Sayfa 844) veya yazma işleminde (Sayfa 845) kullanılabilirsiniz. Bir okuma işleminde Web sunucusu, CPU'dan okunan sayısal değeri, bununla eşleşen Enum metin değeriyle değiştirir. Bir yazma işleminde ise Web sunucusu, değeri CPU'ya yazmadan önce metin değerini, bu metinle eşleşen numaralandırma tamsayı değeriyle değiştirir.

Sözdizimi

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<Enum-Typ>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<Enum-Typ>" -->
```

Parametre

<Varname>	Enum tipine atanacak PLC değişkeninin veya veri bloğu değişkeninin adı veya deklare edilmişse bir PLC değişkeninin takma adının (Sayfa 850) adıdır. Varname, tek tırnak içine alınmalıdır. Bir PLC değişkenini tek tırnak ve bir veri bloğu adını ise çift tırnak içine alın. Veri bloğu değişkenleri için veri bloğunun numarası yerine bloğun adını kullanmayı unutmayın. Veri bloğu adı çift tırnak içindedir, ancak veri bloğu değişkeni adı öyle değildir.
<Enum tipi>	Tek veya çift tırnak içine alınmış numaralandırma tipinin adı

Bir numaralandırma tipinin referansı, güncel fragment için geçerlidir.

Bir değişkeni okurken kullanım için örnek

```
<!-- AWP_Out_Variable Name='"Alarm"' Enum="AlarmEnum" -->...
<p>"Alarm" güncel değeri: :="Alarm":</p>
```

CPU'daki "Alarm" değeri, 2'ye eşit ise, HTML sayfası, '"Alarm" güncel değeri kap boş' gösterir, çünkü Enum tipinin tanımı (Sayfa 851) "Kap boş" karakter dizisini 2 sayısal değerine atar.

Bir değişkeni yazarken kullanım için örnek

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"Alarm yok", 1:"Kap
dolu", 2:"Kap boş"' -->
<!-- AWP_In_Variable Name='"Alarm"' Enum='AlarmEnum' -->...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Kap dolu" /></p>
<p><input type="submit" value='Kabı dolu olarak ayarla' /><p>
</form>
```

Enum tipinin tanımı (Sayfa 851), "Kap dolu" metnini sayısal değer 1'e atadığından, 1 değeri CPU'daki "Alarm" PLC değişkenine yazılır.

Bu bağlamda AWP_In_Variable deklarasyonundaki Enum yan tümcesinin, AWP_Enum_Def deklarasyonundaki ad yan tümcesiyle tam olarak eşleşmesi gerektiğini unutmayın.

Takma adlar kullanarak değişken yazma için örnek

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"Alarm yok", 1:"Kap
dolu", 2:"Kap boş"' -->
<!-- AWP_In_Variable Name='"Alarm"' Enum='AlarmEnum'
Use='"Data_block_4".Motor1.Alarm'-->...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Kap dolu" /></p>
<p><input type="submit" value='Kabı dolu olarak ayarla' /><p>
</form>
```

Enum tipinin tanımı (Sayfa 851), "Kap dolu" metnini sayısal değer 1'e atadığı için, 1 değeri, CPU'nun "Data_Block_4" veri bloğundaki "Motor1.Alarm" PLC değişkenine karşılık gelen "Alarm" takma adına yazılır.

Bir değişken adı veya veri bloğu adı özel karakterler içeriyorsa, ek olarak tırnak işaretleri veya Escape karakterleri kullanmalısınız. Bu, Özel karakterli değişken adlarının işlenmesi (Sayfa 855) altında açıklanmıştır.

NOT

Önceki sürümlerde, bir değişkeni tanımlanmış bir Enum tipiyle ilişkilendirmek için ayrı bir AWP_Enum_Ref deklarasyonu gerekiyordu. STEP 7 ve S7-1200, AWP_Enum_Ref deklarasyonlarıyla mevcut kodu desteklemektedir. Ancak bu komut artık gerekli değildir.

12.8.2.8 Fragmentlerin oluşturulması

CPU özelliklerinde Web sunucusu için "Bloklar oluştur" öğesine tıkladığınızda, STEP 7, kullanıcı tanımlı web sitelerini kumanda DB'si ve DB fragmentleri olarak dönüştürür ve saklar. Belirli sayfalar veya belirli sayfaların alt bölümleri için belirli fragmentler oluşturabilirsiniz. "Start_Fragment" AWP komutunu kullanarak bu fragmentleri bir ad ve numara ile tanımlayabilirsiniz. Bir sayfada AWP_Start_Fragment komutunu izleyen her şey, sonraki bir AWP_Start_Command verilene dek veya dosyanın sonuna ulaşıncaya dek o fragmente aittir.

Sözdizimi

```
<!-- AWP_Start_Fragment Name='<Name>'
[Type=<Typ>][ID=<ID>][Mode=<Modus>] -->
```

Parametre

<Name>	Metin karakter dizisi: DB fragmentinin adı Fragment adları; bir harf veya alt çizgi ile başlamalı ve harfler, sayılar ve alt çizgiler içermelidir. Fragment adı, formun düzenli bir ifadesidir: [a-zA-Z_][a-zA-Z_0-9]*
<Tip>	"Manuel" veya "otomatik" Manuel: STEP 7 programı bu fragmenti talep etmelidir ve buna göre tepki verebilir. Fragmentin çalışma şekline, STEP 7 ve kumanda DB'sinin değişkenleri ile kumanda edilmelidir. Otomatik olarak: Web sunucusu, fragmenti otomatik olarak işler. Tip parametresini belirtmezseniz, ön ayar "Otomatik" olur.
<ID>	Tamsayı tanımlama numarası. ID parametresini belirtmezseniz, Web sunucusu standart olarak bir numara atar. Manuel fragmentlerde ID için düşük bir sayı belirtin. ID, STEP 7 programının manuel bir fragmente kumanda etme biçimidir.
<Modus>	"görünür" veya "gizlenir" görünür: Fragmentin içeriği, kullanıcı tanımlı web sitesinde görüntülenir. gizlenir: Fragment nın içeriği, kullanıcı tanımlı web sitesinde görüntülenmez. Tip parametresini belirtmezseniz, ön ayar "görülür" durumdadır.

Manuel fragmentler

Bir kullanıcı tanımlı web sitesi veya sayfanın bir bölümü için manuel bir fragment oluşturursanız, fragmentin ne zaman gönderileceğini STEP 7 programınız kumanda etmelidir. STEP 7 programı, manuel kumanda altında bir kullanıcı tanımlı sayfa için kumanda DB'sinde uygun parametreleri ayarlamalı ve bunun ardından değiştirilmiş kumanda DB'si ile WWW yordamını çağırmalıdır. Kumanda DB'sinin yapısı ve münferit sayfaların ve fragmentlerin nasıl değiştirileceği ile ilgili daha ayrıntılı açıklamaları, "Kullanıcı tanımlı web sitelerin genişletilmiş kumandası (Sayfa 879)" altında bulabilirsiniz.

12.8.2.9 Fragmentlerin içe aktarımı

HTML'nizin bir bölümünden adlandırılmış bir fragment oluşturabilir ve bunun ardından bu fragmenti kullanıcı tanımlı web sitenizin bir setinin başka bir yerine aktarabilirsiniz. Örneğin bir başlangıç sayfasından ve başlangıç sayfasındaki linkler üzerinden erişilen çeşitli diğer HTML sayfalarından oluşan bir dizi kullanıcı tanımlı web sitesi düşünün. Şirket logosunun bu sayfanın her birinde görünmesini istediğinizi varsayalım. Bunun için şirket logosunun resmini yükleyecek bir fragment oluşturabilirsiniz (Sayfa 853). Bu işlemin ardından her bir HTML sayfası şirket logosunu görüntülemek için bu fragmenti içe aktarabilir. Bunun için Import_Fragment AWP komutunu kullanın. Fragmentin HTML kodu yalnızca bir fragmentte bulunur, ancak bu DB fragmenti gerekli duyduğunuz sıklıkla web sitesine içe aktarabilirsiniz.

Sözdizimi

```
<!-- AWP_Import_Fragment Name='<Adı>' -->
```

Parametre

<Name>	Metin karakter dizisi: İçe aktarılacak DB fragmentinin adı

Örnek

Bir resmi görüntülemek için bir fragment oluşturan HTML kodundan alıntı:

```
<!-- AWP_Start_Fragment Name='Şirketimin_logosu' --><p><img
src="Şirket logosu.jpg"></p>
```

Logo görselli fragmenti başka bir *.html dosyasına içe aktaran HTML kodundan alıntı:

```
<!-- AWP_Import_Fragment Name='Şirketimin_logosu' -->
```

Her iki *.html dosyası (fragmenti oluşturan dosya ve fragmenti içe aktaran dosya), STEP 7'de kullanıcı tanımlı sayfaları yapılandırırken (Sayfa 857) belirttiğiniz klasör yapısında bulunur.

12.8.2.10 Tanımların bağlanması

Kullanıcı tanımlı web sitelerinizde kullanılacak değişkenleri deklare ederken, bir değişken deklarasyonunu değişken takma adı (Sayfa 850) ile ilişkilendirebilirsiniz. Ayrıca birden çok In_Variable'ı bir yordama ve birden çok Out_Variable'ı bir yordama bağlayabilirsiniz.

Örnekler

```
<!-- AWP_In_Variable Name='"Dolum seviyesi"' Name='"Ağırlık"'
Name='"Temp"' -->
<!-- AWP_Out_Variable Name='HEADER:Status' Use='"Status"'
         Name='HEADER:Location' Use="Saklama konumu"
         Name='COOKIE_VALUE:name' Use="Benim_çerezim" -->
<!-- AWP_In_Variable Name='Alarm' Use='"VeriBloğu_10".Alarm' -->
```

12.8.2.11 Özel karakterli değişken adlarının işlenmesi

Değişken adları istisnai anlamlara sahip karakterler içeriyorsa, kullanıcı tanımlı web sitelerinde değişken adları belirtilirken özellikle dikkatli olmak gerekir.

Değişkenleri oku

Bir değişkeni okumak için (Sayfa 844) aşağıdaki sözdizimini kullanın:

:=<Varname>:

Değişkenleri okurken aşağıdaki kurallar geçerlidir:

- PLC değişken tablosundaki değişken adları için, değişken adını çift tırnak içine alın.
- Veri bloğu değişkenleri olan değişken adlarında, veri bloğu adı çift tırnak içine alınmalıdır. Değişken, tırnak işaretlerinin dışındadır.
- Doğrudan G/Ç adresleri, bellek adresleri veya takma adlar olan değişken adlarında okunmakta olan değişkeni tırnak içine almayın.
- Sola doğru eğik çizgi ("ters eğik çizgi") içeren değişken adları veya veri bloğu değişkeni adlarında, ters eğik çizgiden önce başka bir ters eğik çizgi daha koyun.
- Bir değişken adı veya veri bloğu değişken adı; iki nokta üst üste, küçüktür işareti, büyüktür işareti veya ve işareti içeriyorsa, okunan değişken için özel karakterler içermeyen bir takma ad tanımlayın ve bu takma adı kullanarak ilgili değişkeni okuyun. Kullanım yan tümcelerinde değişken adlarındaki iki noktadan önce ters eğik çizgi kullanın.

Tablo 12-1 Okunacak değişkenler için örnekler

Veri bloğu adı	Değişken adı	Oku komutu
-/-	ABC:DEF	**<!--AWP_Out_Variable Name='Özel_değişken' Use ="'ABC:DEF'" --> :=Özel_değişken:**
Geçerli değil	T\	**:="T\\":**
Geçerli değil	A \B 'C :D	**<!--AWP_Out_Variable Name='Diğer_Özel_değişken' Use="'A \\B \'C :D'" --> :=Diğer_Özel_değişken:**
Geçerli değil	a<b	**<!--AWP_Out_Variable Name='a_daha_küçük_b' Use="'a<b'" --> :=a_daha_küçük_b:**
VeriBloğu_1	Değişken_1	**:="VeriBloğu_1".Tag_1:**
VeriBloğu_1	ABC:DEF	**<!-- AWP_Out_Variable Name='Özel_değişken' Use="'VeriBloğu_1".ABC\:DEF'--> :=Özel_değişken:**
DB A' B C D$ E	Değişken	**:="DB A' B C D$ E".Tag:**
DB:DB	Değişken:Değişken	**<!--AWP_Out_Variable Name='Benim_değişken' Use ="'DB:DB".Tag\:Tag' --> :=Benim_değişken:**

Adlandırma ve kullanım yan tümceleri

AWP komutları; AWP_In_Variable, AWP_Out_Variable, AWP_Enum_Def, AWP_Enum_Ref, AWP_Start_Fragment ve AWP_Import_Fragment ad yan tümcelerine sahiptir. <input> ve <select> gibi HTML form değişkenlerinin de ad yan tümceleri vardır. AWP_In_Variable ve AWP_Out_Variable, ek kullanım yan tümcelerine sahip olabilirler. Komut ne olursa olsun, ad ve kullanım yan tümcelerinin sözdizimi, özel karakterleri işleme açısından aynıdır:

- Bir ad veya kullanım yan tümcesi için belirttiğiniz, tek tırnak içine alınmalıdır. Buradaki ad bir PLC değişkeni veya veri bloğu adıysa, tüm yan tümceyi tek tırnak içine alın.
- Bir ad veya kullanım yan tümcesi içinde, veri bloğu adları ve PLC değişken adları çift tırnak içine alınmalıdır.
- Bir değişken adı veya veri bloğu adı tek bir tırnak işareti veya ters eğik çizgi içeriyorsa, karakterden önce bir Escape karakteri olarak ters eğik çizgi kullanın. AWP komutlarının derlenmesinde ters eğik çizgi "\" Escape karakteri olarak kullanılır.

Tablo 12-2 Ad yan tümceleri için örnekler

Veri bloğu adı	Değişken adı	Ad yan tümceleri için olanaklar
-/-	ABC'DEF	**Name='"ABC\'DEF"'**
Geçerli değil	A \B 'C :D	**Name='"A \\B \'C :D"'**
VeriBloğu_1	Değişken_1	**Name='"VeriBloğu_1".Tag_1'**
VeriBloğu_1	ABC'DEF	**Name='"VeriBloğu_1".ABC\'DEF'**
VeriBloğu_1	A \B 'C :D	**Name='"VeriBloğu_1".A \\B \'C :D'**
DB A' B C D$ E	Değişken	**Name='"DB A\' B C D$ E".Değişken'**

Ad yan tümcelerindeki aynı yazım kuralları, kullanım yan tümceleri için de geçerlidir.

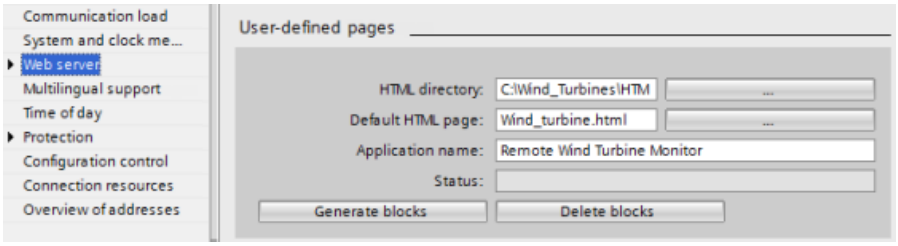
NOT

HTML sayfanızda kullandığınız karakterlerden bağımsız şekilde, HTML sayfasının karakter seti olarak UTF-8 belirleyin ve sayfayı, editörde UTF-8 karakter şifrelemesiyle saklayın.

12.8.3 Kullanıcı tanımlı web sitelerinin kullanımının yapılandırılması

STEP 7'de kullanıcı tanımlı web sitelerini yapılandırırken aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda CPU'yu seçin.
2. CPU'nun denetçi penceresinde Web sunucusu özelliklerini çağırın.
3. Henüz yapılmadıysa, "Bu modülde Web sunucusunu etkinleştir" opsiyon kutusunu etkinleştirin.
4. Web sunucusunun iletişimleri şifrelemesini sağlamak ve web üzerinden erişilebilen CPU'nuzun güvenliğini artırmak için, "Yalnızca HTTPS erişimine izin ver" öğesini seçin.
5. PC'nizde HTML standart sayfasını (başlangıç sayfası) kaydettiğiniz klasörün adını girin.
6. Standart sayfanın adını girin.
7. Uygulamanız için bir ad girin (opsiyonel). Web sunucusu, web sitelerini alt bölümlerine ayırmak veya gruplandırmak için uygulama adından yararlanır. Bir uygulama adı girdiğinizde, Web sunucusu kullanıcı tanımlı sayfanız için aşağıdaki formatta bir URL oluşturur: http[s]://ww.xx.yy.zz/awp/<Uygulama adı>/<Sayfa adı>.html. Herhangi bir uygulama adı girilmezse, URL, http[s]://ww.xx.yy.zz/awp/<Sayfa adı>.html olur.
   Uygulama adında özel karakterlerden kaçının. Bazı karakterler, Web sunucusunun kullanıcı tanımlı sayfaları görüntüleyememesine yol açabilir.



8. "Genişletilmiş" alt bölümünde, AWP komutlarını içeren dosya uzantılarını yazın. STEP 7, standart olarak *.htm, *.html ve *.js uzantılı dosyaları analiz eder. Başka dosya uzantıları kullanıyorsanız, bunları da ekleyin. Bu tipten hiçbir dosya AWP komutu içermiyorsa, proses kaynaklarından tasarruf etmek için dosya uzantılarını girmeyin.
9. Web DB'si için standart numarayı kabul edin veya istediğiniz bir numarayı girin. Bu, web sitelerinin görüntülenmesine kumanda eden kumanda DB'sinin DB numarasıdır.
10. DB fragmenti için standart başlangıç numarasını kabul edin veya istediğiniz bir sayıyı girin. Bu, web siteleriyle birlikte DB fragmentlerinin birincisidir.

Program bloklarının oluşturulması

"Blok oluştur" butonuna tıkladığınızda STEP 7, belirttiğiniz HTML kaynak dizinindeki HTML sayfalarından web sitelerinizi çalıştırmak için veri blokları ve bir kumanda veri bloğu oluşturur. Bu öznitelikleri kendi uygulamanız için gerektiği biçimde (Sayfa 859) ayarlayabilirsiniz. STEP 7 ayrıca tüm HTML sayfalarınızın temsilini saklamak için bir dizi veri bloğu fragmenti de oluşturur. Veri bloklarını oluşturduğunuzda, STEP 7, kumanda veri bloğunun numarasını ve ilk veri bloğu fragmentinin numarasını gösterecek şekilde özellikleri günceller. Veri bloklarını oluşturduktan sonra, kullanıcı tanımlı web siteleriniz STEP 7 programınızın bir fragmenti olur. Bu sayfalara ilişkin bloklar, proje gezgininde "Sistem blokları" klasöründeki program bloklarının altında bulunan Web sunucusu klasöründe görünür.

Program bloklarının silinmesi

Önceden oluşturduğunuz veri bloklarını silmek için, "Veri bloklarını sil" butonuna tıklayın. STEP 7, kullanıcı tanımlı web sitelerini içeren projenizden kumanda veri bloğunu ve tüm veri bloğu fragmentlerini siler.

12.8.4 Ana sayfanın yapılandırılması

CPU'nun cihaz konfigürasyonunda, Web sunucusuna bir PC'den veya bir mobil cihazdan erişmek için, ana sayfa olarak bir kullanıcı tanımlı web sitesini atayabilirsiniz. Bunu yapmazsanız, ana sayfa olarakGiriş (Sayfa 813) standart web sitesi görüntülenir.

Ana sayfa olarak bir kullanıcı tanımlı web sitesini seçmek için aşağıdaki yöntemi izleyin:

1. Cihaz konfigürasyonunda CPU'yu seçin.
2. Denetçi penceresinde, CPU özellikleri altında "Web sunucusu" öğesini seçin ve Web sunucusunu etkinleştirin (Sayfa 798).
3. Web sunucusu özelliklerinde "Ana sayfa" öğesini seçin.
4. Erişildiğinde bir kullanıcı tanımlı sayfa gösterecek şekilde Web sunucusunu yapılandırmak için, açılır listeden "AWP1" öğesini seçin. ("Başlangıç sayfasının" diğer türlü seçiminde, Web sunucusu çağrıldığında standart başlangıç sayfası görüntüler.)

Bundan başka, "Herkes" kullanıcısı için "Kullanıcı tanımlı web sitelerini aç" yetkisini (Sayfa 801) yapılandırmalı ve programınıza WW yordamına (Sayfa 859) bir çağrı eklemelisiniz.

Konfigürasyonu tamamladıktan ve projenizi CPU'ya yükledikten sonra, Web sunucusu, kullanıcı tanımlı web sitelerinizi yapılandırırken (Sayfa 857) ana sayfa olarak seçtiğiniz "standart HTML sayfasını" kullanabilir.

NOT

Bir kullanıcı tanımlı ana sayfa görüntülemek için, CPU'nun RUN işletim durumunda olması gerekir.

12.8.5 Kullanıcı tanımlı web siteleri için WWW yordamının programlanması

Kullanıcı tanımlı web sitelerinin standart web siteleri üzerinden çağrılabilmesi için WWW yordamı, STEP 7 kullanıcı programınıza dahil edilmiş olmalı ve oradan yürütülmelidir. Kumanda veri bloğu, WWW yordamı için giriş parametresidir ve durum ve kumanda bilgilerinin yanı sıra veri bloğu fragmentlerinde temsil edildiği şekliyle sayfaların içeriğini belirtir. STEP 7, kullanıcı tanımlı web sitelerinin konfigürasyonu (Sayfa 857) içindeki "Blok oluştur" butonuna tıkladığınızda kumanda veri bloğunu oluşturur.

WWW yordamının programlanması

Kullanıcı tanımlı web sitelerinin standart web siteleri üzerinden çağrılabilmesi için, STEP 7 programı, WWW yordamını yürütmelidir. Kullanıcı tanımlı web sitelerinin, uygulama gerekliliklerine ve ayarlara göre yalnızca belli bazı koşullar altında kullanılabileceğini belirtebilirsiniz. Bunun ardından program lojiğiniz, WWW yordamının ne zaman çağrılacağına kumanda edebilir.

Tablo 12-3 WWW yordamı

LAD/FBD	SCL	Açıklama
	**ret_val := WWW(ctrl_db:=_uint_in_);**	Standart web sitelerinden kullanıcı tanımlı web sitelerine erişim

Kumanda DB'sinin tamsayı DB'si numarasına karşılık gelen kumanda veri bloğunun (CTRL_DB)) giriş parametresini girmelisiniz. Kullanıcı tanımlı web siteleri için bloklar oluşturduktan sonra, bu kumanda DB'sinin blok numarasını (web DB numarası olarak adlandırılır), CPU'nun Web sunucusu özelliklerinde bulabilirsiniz. DB tamsayısını, WWW yordamının CTRL_DB parametresi olarak belirtin. Geri dönüş değeri (RET_VAL), fonksiyon sonucunu içerir. WWW yordamının asenkron olarak yürütüleceğini ve daha sonra bir hata ortaya çıksa bile RET_VAL çıkışının 0 başlangıç değerine sahip olduğunu dikkate alın. Program, uygulamanın başarıyla başlatıldığından emin olmak için kumanda DB'sinin durumunu sorgulayabilir veya ardından WWW'ya yapılan bir çağrıyla RET_VAL parametresini sorgulayabilir.

Tablo 12-4 Geri dönüş değeri

RET_VAL	Açıklama
0	Hata yok
16#00yx	x: İlgili Bit tarafından temsil edilen talep, bekleme durumunda: x=1: Talep 0 x=2: Talep 1 x=4: Talep 2 x=8: Talep 3 x değerleri, birden çok talebin bekleme durumlarını temsil etmek için mantıksal olarak VEYA olabilir. Örneğin x = 6 ise, 1 ve 2 numaralı talepler bekleme durumundadır. y: 0: Hata yok; 1: Hata mevcut ve kumanda DB'sinde "last_error" ayarlanmıştır (Sayfa 879)
16#803a	Kumanda DB'si yüklenmemiş.
16#8081	Kumanda DB'sinin veri tipi, formatı veya sürümü yanlış.
16#80C1	Web uygulamasını başlatmak için kullanılabilir kaynak mevcut değil.

Kumanda DB'sinin kullanımı

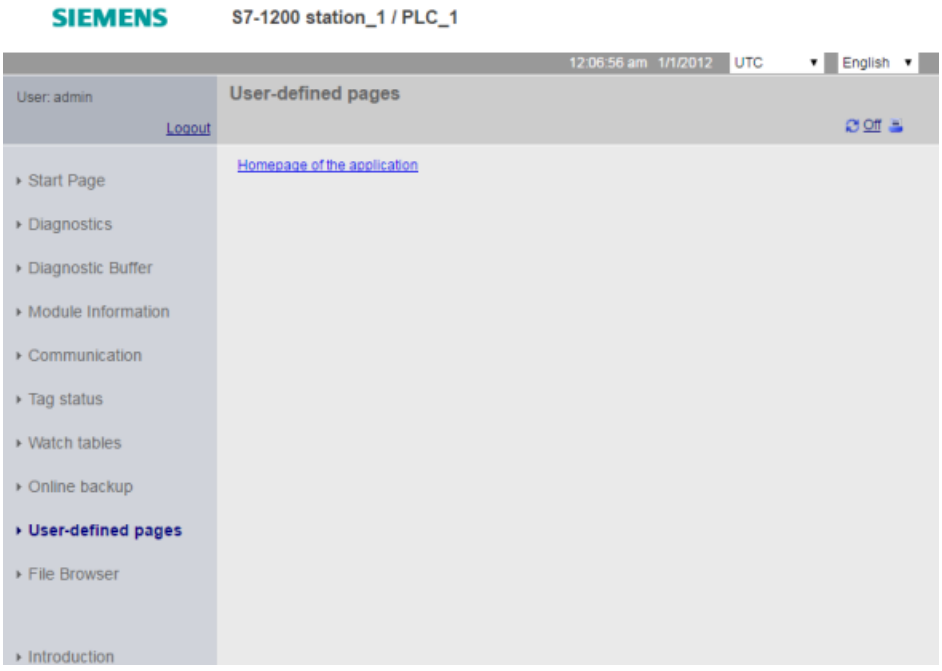
STEP 7, "Blok Oluştur" butonuna tıkladığınızda, kumanda verisi bloğunu oluşturur. Kumanda DB numarası, kullanıcı tanımlı web sitelerinin özelliklerinde görüntülenir. Kumanda DB'sini, proje gezgininde "Program blokları" klasöründe de bulabilirsiniz.
Tipik olarak, STEP 7 programınız kumanda DB'sini başka herhangi bir değişiklik olmaksızın "Blok oluştur" işlemiyle oluşturulduğu biçiminde doğrudan kullanır. Ancak STEP 7 kullanıcı programı, Web sunucusunu devre dışı bırakmak veya ardından yeniden etkinleştirmek için kumanda DB'sinde global komutlar tanımlayabilir. Ayrıca STEP 7 kullanıcı programı, kumanda DB'sindeki bir talep tablosu üzerinde manuel DB fragmentleri olarak (Sayfa 857) oluşturduğunuz kullanıcı tanımlı web sitelerinin davranışına kumanda etmelidir. Bu genişletilmiş görevlere ilişkin daha fazla bilgi bkz. Kullanıcı tanımlı web sitelerin genişletilmiş kumandası (Sayfa 879).

12.8.6 Program bloklarının CPU'ya yüklenmesi

Kullanıcı tanımlı web siteleri için bloklar oluşturduktan sonra, diğer tüm program blokları gibi bunlar da STEP 7 programınızın bir parçası haline gelir. Program bloklarını CPU'ya yüklerken alışıldığı gibi devam edin. Kullanıcı tanımlı web siteleri için program bloklarını yalnızca STOP işletim durumundayken CPU'ya yükleyebileceğinizi dikkate alın.

12.8.7 Kullanıcı tanımlı web sitelerine erişim

Kullanıcı tanımlı web sitelerine, Standart web siteleri (Sayfa 802) üzerinden erişirsiniz. Standart web siteleri, soldaki gezinme menüsünde bir "Kullanıcı tanımlı siteleri" bağlantısı görüntüler. Temel sitelerdeki gezinme, "Kullanıcı tanımlı siteler" linkini de gösterir. "Kullanıcı tanımlı siteler" linkine tıkladığınızda, web tarayıcınız standart sayfanıza bir link sunan sayfaya gidecektir. Kullanıcı tanımlı sitelerde gezinme, ilgili sitelerinizin yapısına karşılık gelir.



NOT

Ayrıca Web sunucusu için bir kullanıcı tanımlı siteyi ana sayfa olarak da tanımlayabilirsiniz (Sayfa 859).

12.8.8 Kullanıcı tanımlı web sitelerindeki sınırlamalar

Standart web siteleri için olan sınırlamalar (Sayfa 885), kullanıcı tanımlı web siteleri için de geçerlidir. Kullanıcı tanımlı web siteleri ile ilgili bazı özel noktalar da vardır.

Yükleme belleğinin kapasitesi

"Blok oluştur" opsiyonuna tıkladığınızda, kullanıcı tanımlı web siteleriniz, yükleme belleğinde yer gerektiren veri blokları haline gelir. Bir memory card taktıysanız, memory card'nızın kapasitesi, kullanıcı tanımlı web siteleri için harici yükleme belleği olarak kullanılabilir.
Bir memory card takmadıysanız, bu bloklar dahili yükleme belleğinde CPU modeline bağlı olarak sınırlı yer kaplar.
STEP 7'deki çevrimiçi ve teşhis fonksiyonları ile yükleme belleğindeki dolu ve boş alanı sorgulayabilirsiniz. Yükleme belleğindeki gerekli alanı, STEP 7'nin kullanıcı tanımlı web siteleriniz için oluşturduğu münferit blokların özelliklerinde de denetleyebilirsiniz.

NOT

Kullanıcı tanımlı web siteleriniz için olan alanı azaltmanız gerekiyorsa, eklenen bazı görselleri kaldırabilirsiniz.

Metin dizilerindeki tırnak işaretleri

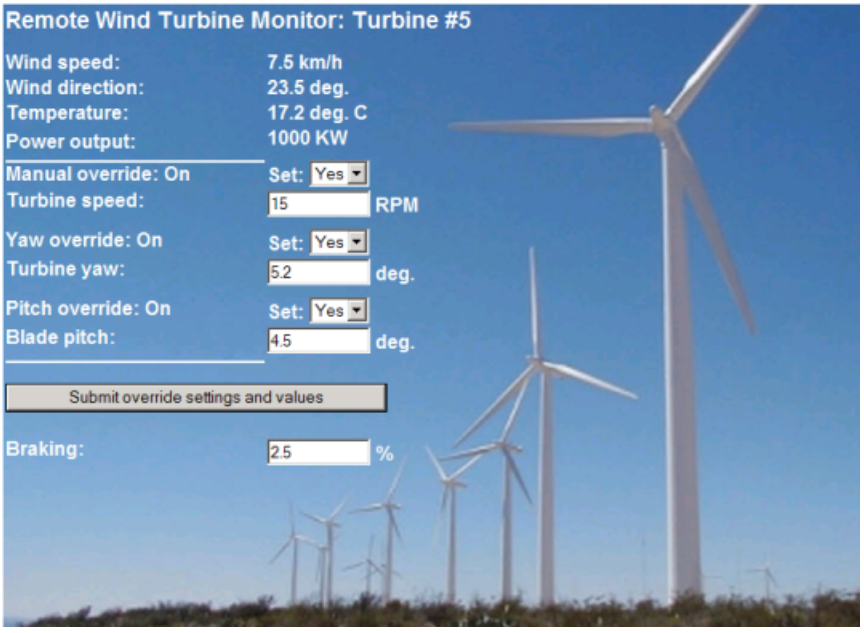
Kullanıcı tanımlı web sitelerindeki veri bloğu değişkenlerinde, gömülü tek veya çift tırnak işaretli metin dizileri kullanılmamalıdır. HTML sözdizimi, ayıma işareti olarak genellikle tek veya çift tırnak işareti istediğinden, metin dizelerindeki tırnak işaretleri kullanıcı tanımlı web sitelerinin görüntülenmesini sekteye uğratabilir.
Kullanıcı tanımlı web sitelerinde kullanılan String tipi veri bloğu değişkenlerinde aşağıdaki kurallara uyulmalıdır:

- STEP 7'deki veri bloğu değişkenlerinin String değerinde tek veya çift tırnak işareti kullanmayın.
- Kullanıcı programının, bu veri bloğu değişkenlerine tırnak içine alınmış karakter dizileri atamadığından emin olun.

12.8.9 Kullanıcı tanımlı web siteleri için örnek

12.8.9.1 Bir rüzgâr türbinini izlemek ve denetlemek için web sitesi

Kullanıcı tanımlı web sitesi örneği olarak, bir rüzgâr türbinini uzaktan izlemeye ve denetlemeye odaklanmış bir web sitesi düşünün:



Açıklama

Bu uygulamada, rüzgâr çiftliğindeki her bir rüzgâr türbini, türbine kumanda etmek üzere bir S7-1200 ile donatılmıştır. STEP 7 programında, her bir rüzgâr türbini için o rüzgâr türbinine özgü verileri içeren bir veri bloğu bulunur.

Bir kullanıcı tanımlı web sitesi, bir PC üzerinden türbine desantral erişim sağlar. Bir kullanıcı, belirli bir rüzgâr türbininin CPU'sunun standart web sitelerini çağırabilir ve türbin verilerini görüntülemek için kullanıcı tanımlı "Remote Wind Turbine Monitor - Uzak rüzgâr türbini monitörü" web sitesine (rüzgâr türbinini desantral izleme için web sitesi) erişebilir. Değişkenleri değiştirme yetkisi olan bir kullanıcı, web sitesi üzerinden türbini manuel moda da ayarlayabilir ve türbin devir sayısı, yönü ve kanat eğim açısı değişkenlerine kumanda edebilir. Değişkenleri değiştirme yetkisi olan bir kullanıcı, türbinin manuel veya otomatik olarak kumada edilmesinden bağımsız olarak bir frenleme değeri de belirleyebilir.

STEP 7 programı, otomatik kumandayı geçersiz kılma için Boole değerlerini inceler ve ayarlanmışsa, türbin devir sayısı, yönü ve kanat eğim açısı için kullanıcı tarafından girilen değerleri kullanır. Aksi durumda program bu değerleri yok sayar.

Kullanılan dosyalar

Bu örnek kullanıcı tanımlı web sitesi, üç dosyadan oluşur:

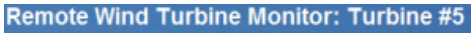
- **Rüzgâr türbini.html**: Bu, yukarıda gösterilen HTML sayfasıdır. Kumanda verilerine AWP komutları aracılığıyla erişilir.
- **Rüzgâr türbini.css**: Bu, HTML sayfası için biçimlendirme spesifikasyonlarını içeren basamaklı stil şablonudur. Bir basamaklı stil şablonu kullanmak isteğe bağlıdır, ancak bu şablon HTML sayfası geliştirmeyi kolaylaştırır.
- **Wind_turbine.jpg**: Bu, HMTL sayfasında görüntülenecek olan arka plan görüntüsüdür. Görsellerin kullanıcı tanımlı web sitelerinde kullanımı elbette isteğe bağlıdır ve resimler CPU'da ek bellek alanı gerektirir.

Kurulumunuzda bu dosyalar mevcut değildir, yine de örnek olarak açıklanmaktadır.

Gerçekleştirme

HTML sayfası, ekran alanları için PLC cihazından değerleri okumak üzere (Sayfa 844) AWP komutlarını ve kullanıcı giriş verilerini PLC cihazına değer olarak yazmak için (Sayfa 845) ise yine AWP komutlarını kullanır. Bu sayfa ayrıca, Enum tiplerinin tanımını (Sayfa 851) ve AÇIK/KAPALI ayarlarını işlemek üzere referans (Sayfa 851) olarak da AWP komutlarını kullanır.

Sayfanın ilk kısmı, rüzgâr türbini numarasını içeren bir başlık satırını görüntüler.



Sayfanın bir sonraki kısmı ise rüzgâr türbinindeki atmosferik koşulları içerir. Türbin konumundaki G/Ç verileri, rüzgâr hızı, rüzgâr yönü ve güncel sıcaklığı sunar.

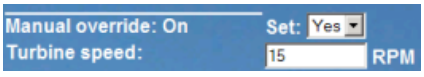


Sayfa, ayrıca S7-1200'den okunan türbin güç çıkışını da gösterir.



Aşağıdaki alt bölümlerde, S7-1200'ün olağan otomatik kumandası geçersiz kılınarak türbinin manuel kumandasına izin verilmektedir. Aşağıdaki geçersiz kılmalar mümkündür:

- Manuel geçersiz kılma: Türbini manuel olarak geçersiz kılmayı etkinleştirir. STEP 7 kullanıcı programı, manuel türbin devir sayısı, yönü veya kanat eğim açısı ayarlarını kullanmak için manuel geçersiz kılma ayarının doğru olmasını talep eder.



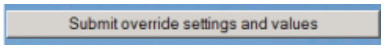
- Hizalamayı geçersiz kılma: Manuel türbin hizalamasını geçersiz kılmayı ve manuel hizalama ayarını etkinleştirir. STEP 7 kullanıcı programı, hizalama ayarının uygulanabilmesi için manuel geçersiz kılma ve hizalama geçersiz kılma ayarının doğru olmasını talep eder.



- Kanat eğim açısını geçersiz kılma: Rotor kanatlarının kanat eğim açısının manuel olarak geçersiz kılınmasını etkinleştirir. STEP 7 kullanıcı programı, kanat eğim açısı ayarının uygulanabilmesi için manuel geçersiz kılma ve kanat eğim açısını geçersiz kılma ayarının doğru olmasını talep eder.



HTML sayfası, geçersiz kılma ayarlarını kumanda ünitesine göndermek için bir "Gönder" butonu içerir.



Frenleme veri giriş alanı, bir frenleme yüzde değerinin manuel olarak ayarlanmasını sağlar. STEP 7 kullanıcı programı, bir frenleme değerini kabul etmek için manuel geçersiz kılma talep etmez.



HTML sayfası, bunun dışında sayfaya erişen kullanıcının kullanıcı ID'sini içeren özel değişkeni PLC değişken tablosundaki bir değişkene yazmak için (Sayfa 849) bir AWP komutu kullanır.

12.8.9.2 Kumanda verilerinin okunması ve görüntülenmesi

"Uzak rüzgâr türbini monitörü" HTML sayfası, bu verileri sayfada görüntülemek için kumanda ünitesinden gelen verileri okumak üzere çok sayıda AWP komutu (Sayfa 844) kullanır. Örneğin, bu örnek web sitesinin ilgili bölümünde güç çıkışını görüntülemek için HTML kodunu ele alalım:



Bir HTML kodu için örnek

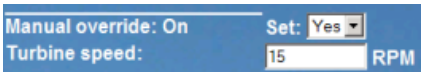
"Uzak rüzgâr türbini monitörü" HTML sayfasındaki aşağıda verilen alıntıda, bir tablo satırının sol hücresinde "Güç çıkışı:" metni görüntülenir ve güç çıkışı değişkeni okunur ve metinle birlikte tablo satırının sağ hücresinde görüntülenir.

AWP komutu :="VeriBloğu_1".güç çıkışı: okuma işlemini yürütür. Bu bağlamda veri bloklarına, veri bloğunun numarasıyla değil adıyla başvurulduğunu dikkate alın (yani "DB1" aracılığıyla değil, "VeriBloğu_1" üzerinden).

```
<tr style="height:2%;">
<td>
<p>Güç çıkışı:</p>
</td>
<td>
<p style="margin-bottom:5px;"> :="VeriBloğu_1".Güç çıkışı: kW</p>
</td>
</tr>
```

12.8.9.3 Enum tipinin kullanılması

"Uzak rüzgâr türbini monitörü" HTML sayfası, HTML sayfasının bir Boole değeri için "ON" veya "OFF" görüntülediği ve kullanıcının bir Boole değeri girdiği üç yer için Enum tiplerini kullanır. "ON" için Enum tipi 1 değeriyle sonuçlanır ve "OFF" için Enum tipi 0 değerini getirir. Örnek: Bir Enum tipi kullanarak "VeriBloğu_1 değerinde".ManuelGeçersizKılmayıSerbestBırak ayarını okumak ve yazmak için HTML kodunu ele alalım.



Bir HTML kodu için örnek

"Uzak rüzgâr türbini monitörü" HTML sayfasından aşağıdaki alıntılar, "Kapalı" ve "Açık" için 0 ve 1 değerleri ile geçersiz kılma durumu adlı bir Enum tipinin deklarasyonunu ve ardından, "VeriBloğu_1" veri bloğundaki "ManuelGeçersizKılmayıSerbestBırak" Boole değişkeni için "Geçersiz kılma durumu" Enum tipi referansını ayarlanmasını göstermektedir.

```
<!-- AWP_In_Variable
Name='"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak'
Enum="Geçersiz kılma durumu" -->
<!-- AWP_Enum_Def Name="Geçersiz kılma durumu"
Values='0:"Kapalı",1:"Açık"' -->
```

HTML sayfası, "ManuelGeçersizKılmayıSerbestBırak" öğesinin güncel durumu için bir tablo hücresinde bir görüntüleme alanı içeriyorsa, değişkenler için yalnızca normal bir okuma komutu kullanılır, ancak önceden deklare edilen ve başvurulan Enum tipi sayesinde sayfa 0 veya 1 değil, aksine "Off" veya "On" görüntüler.

```
<td style="width:24%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manuel geçersiz kılma:
:="VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak:</p>
</td>
```

HTML sayfası, kullanıcının "ManuelGeçersizKılmayıSerbestBırak" değerini değiştirebilmesi için bir açılır liste içerir. Açılır liste, "Evet" ve "Hayır" metnini opsiyon olarak görüntüler. "Evet" metni, "Açık" değerine ve "Hayır" metni "Kapalı" değerine Enum tipi ile bağlanmıştır. Herhangi bir seçim yapılmadığı takdirde, "ManuelGeçersizKılmayıSerbestBırak" değeri değişmeden kalır.

```
<select name='"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak'>
<option value=':"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak:'>
</option>
<option value="Açık">Evet</option>
<option selected value="Kapalı">Hayır</option>
</select>
```

Seçim listesi, HTML sayfasındaki bir formdadır. Kullanıcı "Gönder" butonuna tıkladığında, site, formu karşıya yükler. Böylece kullanıcı "Evet" öğesini seçtiyse, VeriBloğu_1'deki "ManuelGeçersizKılmayıSerbestBırak" Boole değişkenine "1" değeri veya kullanıcı "Hayır" öğesini seçtiyse "0" yazılır.

12.8.9.4 Kullanıcı veri girişlerinin kumanda ünitesine yazılması

"Uzak rüzgâr türbini monitörü" HTML sayfası, Kumanda ünitesine veri yazmak için çeşitli AWP komutlarını (Sayfa 845) içerir. HTML sayfası, Boole değişkenleri için AWP_In_Variables deklare eder, bu sayede, değişkeni değiştirme yetkisi olan bir kullanıcı rüzgâr türbinine manuel olarak kumanda edebilir ve türbin hızını geçersiz kılmayı, türbin yönü ve/veya kanat eğim açısı için manuel geçersiz kılmayı etkinleştirebilir. Bu web sayfası ayrıca, değişken değiştirme yetkisi olan bir kullanıcının türbin hızı, yönü, kanat eğim açısı ve frenleme yüzdesi için daha sonra kayar nokta değerleri ayarlamasına izin veren AWP_In_Variable'ları içerir. Web sitesi, AWP_In_Variable'ları kumanda ünitesine yazmak için bir form (Form POST) göndermek üzere bir HTTP komutu kullanır.

Örneğin frenleme değerini manuel olarak ayarlamak için HTML koduna dikkat edin:



Bir HTML kodu için örnek

"Uzak rüzgâr türbini monitörü" HTML sayfasından alınan aşağıdaki bölümde, ilk önce "VeriBloğu_1" için AWP_In_Variable deklare edilmekte ve bu değişken ise HTML sayfasının "VeriBloğu_1" veri bloğundaki herhangi bir değişkene yazmasını sağlamaktadır. Sayfa, tablo satırının sol hücresinde "Frenleme:" metnini görüntüler. Tablo satırının sağ hücresinde ise "VeriBloğu_1" öğesinin "Frenleme" değişkeni için kullanıcı verilerinin girilebileceği alan bulunur. Bu kullanıcı giriş değeri, girilen metin verilerini HTTP yöntemi "POST" aracılığıyla CPU'ya yazan bir HTML formunda bulunmaktadır. Bunun ardından sayfa, kumanda ünitesinden fiili gecikme değerini okur ve veri giriş alanında görüntüler.

Değişkenleri değiştirme yetkisi olan bir kullanıcı, bu sayfayı, frenlemeye kumanda eden CPU'nun veri bloğuna bir frenleme değeri yazmak için kullanabilir.

```
<!-- AWP_In_Variable Name='"VeriBloğu_1"' -->
...
<tr style="vertical-align: top; height: 2%;">
<td style="width: 22%;"><p>Frenleme:</p></td>
<td>
<form method="POST">
<p><input name='"VeriBloğu_1".Frenleme' size="10" type="text"> %</p>
</form>
</td>
</tr>
```

NOT

Dikkat etmeniz gerekenler: Kullanıcı tanımlı bir sayfa, "String" veri tipinde yazılabilir bir veri bloğu değişkeni için bir veri giriş alanı içeriyorsa, kullanıcı, bu alana metin girerken karakter dizisini tek tırnak içine almalıdır.

NOT

Ancak, bir AWP_In_Variable deklarasyonunda bir veri bloğunun tamamını deklare ederseniz, örn. <!-- AWP_In_Variable Name='"VeriBloğu_1"' -->, bu veri bloğundaki her bir değişkenin kullanıcı tanımlı web sitesi aracılığıyla yazılabileceğine dikkat edin. Bir veri bloğundaki tüm değişkenlerin yazılabilir olmasını istiyorsanız bu şekilde ilerleyin. Aksi takdirde, kullanıcı tanımlı web sitesi üzerinden yalnızca belirli bazı veri bloğu değişkenlerinin yazılabilir olmasını istiyorsanız, bu durumda bu değişkenleri spesifik olarak deklare etmelisiniz, örneğin: <!-- AWP_In_Variable Name='"VeriBloğu_1".Frenleme' -->

12.8.9.5 Özel değişkenlerin yazılması

Kullanıcının değiştirme yetkisi varsa, "Uzak rüzgâr türbini monitörü" web sitesi SERVER:current_user_id özel değişkenini CPU'daki bir PLC değişkenine yazar. Bu durumda PLC değişken değeri, "Uzak rüzgâr türbini monitörü" web sitesine erişen kullanıcının kullanıcı ID'sini alır.
Özel değişken, web sitesinden alınıp PLC cihazına yazılır ve kullanıcı arayüzü gerektirmez.

Bir HTML kodu için örnek

```
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="Kullanıcı
ID'si"-->
```

12.8.9.6 Referans: "Uzak rüzgâr türbini monitörü" web sitesinin HTML kodu

Rüzgâr türbini.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<!--
Bu test programı, bir rüzgâr türbinini denetlemek ve çalıştırmak
için bir web sitesini simüle eder.
STEP 7'de gerekli PLC değişkenleri ve veri bloğu değişkenleri:

PLC değişkeni:
Kullanıcı_ID'si: Int

Veri blokları:
VeriBloğu_1

VeriBloğu_1'in değişkenleri:

Türbin numarası: Int
Rüzgâr hızı: Real
Rüzgâr yönü: Real
Sıcaklık: Real
Güç çıkışı: Real
```

```
ManuelGeçersizKılmayıSerbestBırak: Bool
Türbin hızı: Real
HizalamayıGeçersizKıl: Bool
Hizalama: Real
KanatEğimAçısınıGeçersizKıl: Bool
Kanat eğim açısı: Real
Frenleme: Real
Kullanıcı tanımlı web sitesi, PLC verileri için güncel değerleri
gösterir ve numaralandırma tipi atanan üç Boole değerini ayarlamak
için bir seçim listesi sunar. "Gönder" butonu, seçilen Boole
değerlerinin yanı sıra türbin devir sayısı, yönü ve kanat eğim açısı
için veri giriş alanlarını yükler. Frenleme değeri, "Gönder" butonu
olmadan ayarlanabilir.


Bu web sayfasını kullanmak için gerçek bir STEP 7 programı gerekmez.
Teorik olarak, STEP 7 programı yalnızca atanan Boole değerleri
sabitlenirse türbin devir sayısına, yönüne ve kanat eğim açısı
değerlerine yanıt verir. STEP 7 için tek gereklilik, bu sayfa için
oluşturulan veri bloklarının DB numarasıyla WWW yordamını
çağırmaktır.
-->
<!-- AWP_In_Variable Name='"VeriBloğu_1"' -->
<!-- AWP_In_Variable
Name='"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak'
Enum="Geçersiz kılma durumu" -->
<!-- AWP_In_Variable
Name='"VeriBloğu_1".KanatEğimAçısınıGeçersizKıl' Enum="Geçersiz
kılma durumu" -->
<!-- AWP_In_Variable Name='"VeriBloğu_1".HizalamayıGeçersizKıl'
Enum="Geçersiz kılma durumu" -->
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="Kullanıcı
ID'si"-->
<!-- AWP_Enum_Def Name="Geçersiz kılma durumu"
Values='0:"Kapalı",1:"Açık"' -->


<html>
<head>
<meta http-equiv="content-type" content="text/html;
charset=utf-8"><link rel="stylesheet" href="Rüzgâr türbini.css">
<title>Rüzgâr türbinlerinin uzaktan denetimi</title>
</head>
<body>
<table cellpadding="0" cellspacing="2">
<tr style="height: 2%;">
<td colspan="2">
<h2>Rüzgâr türbinlerinin uzaktan denetimi: Türbin
No.:="VeriBloğu_1".Türbin numarası:</h2>
</td>


<tr style="height: 2%;"><td style="width: 25%;"><p>Rüzgâr
hızı:</p></td>
```

```
<td><p> :="VeriBloğu_1".Rüzgâr hızı: km/h</p></td>
</tr>

<tr style="height: 2%;">
<td style="width: 25%;"><p>Rüzgâr yönü:</p></td>
<td><p> :="VeriBloğu_1".Rüzgâr yönü: Derece</p></td>
</tr>

<tr style="height: 2%;"><td style="width:
25%;"><p>Sıcaklık:</p></td>
<td><p> :="VeriBloğu_1".Sıcaklık: Derece C</p></td>
</tr>

<tr style="height: 2%;">
<td style="width: 25%;"><p>Güç çıkışı:</p></td>
<td><p style="margin-bottom:5px;"> :="VeriBloğu_1".Güç çıkışı:
kW</p>
</td>
</tr>

<form method="POST" action="">
<tr style="height: 2%;" >
<td style="width=25%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manuel geçersiz kılma:
:="VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak:</p>
</td>
<td class="Text">Ayarlama:

<select name='"VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak'>
<option value=':="VeriBloğu_1".ManuelGeçersizKılmayıSerbestBırak:'>
</option>
<option value="Açık">Evet</option>
<option value="Kapalı">Hayır</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;"><td style="width:
25%;"><p>Türbin hızı:</p></td>
<td>
<p style="margin-bottom:5px;"><input name='"VeriBloğu_1".Türbin
hızı' size="10" value=':="VeriBloğu_1".Türbin hızı:' type="text">
U/MIN</p>
</td>
</tr>
```

```
<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;">
<p>Geçersiz kılma Hizalama: :="VeriBloğu_1".HizalamayıGeçersizKıl:
</p>
</td>
<td class="Text">Ayarlar:

<select name='"VeriBloğu_1".HizalamayıGeçersizKıl'>
<option value=':="VeriBloğu_1".HizalamayıGeçersizKıl:'> </option>
<option value="Açık">Evet</option>
<option value="Kapalı">Hayır</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;">
<p>Türbin hizalama:</p>
</td>
<td>
<p style="margin-bottom:5px;"><input name='"VeriBloğu_1".Hizalama'
size="10" value=':="VeriBloğu_1".Hizalama:' type="text"> Derece</p>
</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;">
<p>Geçersiz kılma Kanat eğim açısı:
:="VeriBloğu_1".KanatEğimAçısınıGeçersizKıl: </p>
</td>
<td class="Text">Ayarlama:

<select name='"VeriBloğu_1".KanatEğimAçısınıGeçersizKıl'>
<option value=':="VeriBloğu_1".KanatEğimAçısınıGeçersizKıl:'>
</option>
<option value="Açık">Evet</option>
<option value="Kapalı">Hayır</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
```

```
<td style="width=25%; border-bottom-style: Solid; border-bottom-
width: 2px; border-bottom-color: #ffffff;">
<p>Rotor kanatları Kanat eğim açısı:</p>
</td>
<td>
<p style="margin-bottom:5px;"><input name='"VeriBloğu_1".Kanat eğim
açısı' size="10" value=':="VeriBloğu_1".Kanat eğim açısı:'
type="text"> Derece</p>
</td>

</tr>
<tr style="height: 2%;">
<td colspan="2">
<input type="submit" value="Geçersiz kılma için ayarları ve
değerleri gönder">
</td>
</tr>
</form>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;"><p>Frenleme:</p></td>
<td>
<form method="POST" action="">
<p> <input name='"VeriBloğu_1".Frenleme' size="10"
value=':="VeriBloğu_1".Frenleme:' type="text"> %</p>
</form>
</td>
</tr>
<tr><td></td></tr>

</table>
</body>
</html>
```

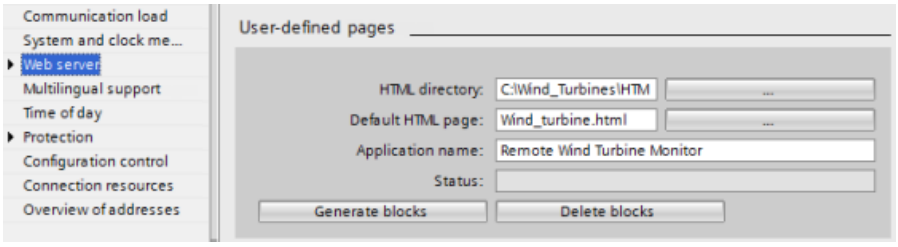
Rüzgâr türbini.css

```
BODY {
background-image: url('./Wind_turbine.jpg');
background-position: 0% 0%;
background-repeat: no-repeat;
background-size: cover;
}
H2 {
font-family: Arial;
font-weight: bold;
font-size: 14.0pt;
color: #FFFFFF;
margin-top:0px;
margin-bottom:10px;
}
P {
font-family: Arial;
font-weight: bold;
color: #FFFFFF;
font-size: 12.0pt;
margin-top:0px;
margin-bottom:0px;
}
TD.Text {
font-family: Arial;
font-weight: bold;
color: #FFFFFF;
font-size: 12.0pt;
margin-top:0px;
margin-bottom:0px;
}
```

12.8.9.7 STEP 7'de örnek web sitesinin konfigürasyonu

"Uzak rüzgâr türbini monitörü" HTML sayfasını, S7-1200 için kullanıcı tanımlı web sayfası olarak eklemek üzere, STEP 7'de HTML sayfası verilerini yapılandırır ve HTML sayfasından veri blokları oluşturursunuz.

Rüzgâr türbinine kumanda eden S7-1200'ün CPU özelliklerine erişirsiniz ve ilgili konfigürasyon bilgilerini Web sunucusunun kullanıcı tanımlı sitelerin özelliklerinde girersiniz:



Konfigürasyon alanları

- HTML dizini: Bu alan, bilgisayarda standart sayfanın (başlangıç sayfası) bulunduğu klasörün tam yol adını belirtir. İstediğiniz klasöre gitmek için "..." butonunu kullanabilirsiniz.
- Standart HTML sayfası: Bu alan, HTML uygulamasının standart sayfasının veya başlangıç sayfasının dosya adını belirtir. İstediğiniz dosyayı seçmek için "..." butonunu kullanırsınız. Bu örnekte, Windturbine.html standart HTML sayfasıdır. "Uzak rüzgâr türbini monitörü" örneği yalnızca tek bir sayfadır, ancak diğer özel uygulamalar standart sayfadan diğer sayfalara bağlanabilir. Bu bağlamda HTML kodundaki standart sayfa, HTML kaynak klasörüne göre diğer sayfalara referanslamalıdır.
- Uygulama adı: Bu, isteğe bağlı bir alandır ve sayfaya erişildiğinde web tarayıcısının adres alanında görüntülenen addır. Bu ad, örneğimizde "Uzak rüzgâr türbini monitörü" olur, ancak başka herhangi bir ad da girebilirsiniz.

Diğer alanların yapılandırılması gerekmez.

Son adımlar

"Uzak rüzgâr türbini monitörü" HTML sayfasını yapılandırıldığı şekilde kullanabilmek için, veri blokları oluşturun, WWW yordamını programlayın (Sayfa 859), giriş parametresi olarak oluşturulan kumanda DB'sinin numarasıyla program bloklarını CPU'ya yükleyin ve CPU'yu RUN işletim durumuna getirin.

Bunun ardından, bir kullanıcı, rüzgâr türbinine kumanda eden S7-1200 standart web sitelerini çağırdığında, gezinme çubuğundaki "Kullanıcı tanımlı siteler" linkini kullanarak "Uzak rüzgâr türbini monitörü" web sitesini görüntüleyebilir. Bu sayfa şimdi, rüzgâr türbinini denetlemek ve kumanda etmek için araçlar sunmaktadır.

12.8.10 Kullanıcı tanımlı, çok dilli web sitelerinin oluşturulması

Web sunucusu, aşağıdaki dillerde kullanıcı tanımlı web sitelerini oluşturmanıza olanak verir:

- Almanca (de)
- İngilizce (en)
- İspanyolca(es)
- Fransızca (fr)
- İtalyanca (it)
- Basitleştirilmiş Çince (zh)

Bu bağlamda, ilgili dillere karşılık gelecek şekilde bir klasör yapısında (Sayfa 876) HTML sayfalarınızı oluşturun ve siteleriniz için özel "siemens_automation_language" çerezini ayarlayın (Sayfa 876). Web sunucusu bu çereze tepki verir ve ilgili dil klasöründeki standart sayfaya geçer.

12.8.10.1 Klasör yapısı oluşturulması

Çok dilli kullanıcı tanımlı web sitelerini oluşturmak için HTML dizininizde bir klasör yapısı oluşturmalısınız. İki harfli klasör adları spesifiktir ve aşağıda gösterildiği gibi belirtilmelidir:



de: Almanca
en: İngilizce
es: İspanyolca
fr: Fransızca
it: İtalyanca
zh: Basitleştirilmiş Çince

Bu seviyedeyken siteleriniz için başka klasörler de düzenleyebilirsiniz, örn. görseller veya komut dizileri için bir klasör gibi.
Herhangi bir dil klasörü ekleyebilirsiniz. Bu altı dilin tamamını kaydetmeniz gerekli değildir. Dil klasörlerinde, HTML sayfalarınızı ilgili dilde oluşturur ve programlarsınız.

12.8.10.2 Dil değiştirmenin programlanması

Web sunucusu, "siemens_automation_language" çerezini kullanarak ilgili dili değiştirir. Bu çerez, HTML sayfalarında tanımlanır, kurulur ve Web sunucusu ilgili dilde bir sayfayı görüntülemek için bu çerez üzerinden söz konusu dilin adıyla dil klasöründen o dili çekip kullanır. HTML sayfası, bu çerez için önceden tanımlanmış dil kodlarından birini ayarlamak üzere JavaScript içermelidir, bu dil kodları şunlardır: "de", "en", "es", "fr", "it" veya "zh". HTML sayfası bu çerezi örn. "de" olarak ayarlarsa, Web sunucusu "de" klasörüne geçer ve STEP 7 konfigürasyonu (Sayfa 879) uyarınca sayfayı HTML standart sayfasının adıyla görüntüler.

Örnek

Aşağıdaki örnekte, dil klasörlerinin her birinde "langswitch.html" adlı bir standart HTML sayfası vardır. HTML dizininde ayrıca "script" adlı bir dizin bulunur. "script " dizini, "lang.js" adlı bir JavaScript dosyası içerir. Her bir "langswitch.html " sayfası, "siemens_automation_language" dil çerezini ayarlamak için bu JavaScript'i kullanır.

"en" dizininde "langswitch.html" için HTML

HTML sayfasının başlık satırı "English" dilini ayarlar, "UTF-8" karakter setini ve "lang.js" JavaScript dosyasının yolunu belirtir.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

Dosyanın içeriği, ilgili kullanıcının Almanca ve İngilizce dilleri arasında seçim yapabileceği bir seçim listesi içerir. İngilizce ("en"), ön seçimli dildir. Kullanıcı, sayfanın dilini değiştirdiğinde sayfa, seçilen opsiyonun değeriyle DoLocalLanguageChange() JavaScript fonksiyonunu çağırır.

```
<!-- Language Selection -->
<table>
   <tr>
      <td align="right" valign="top" nowrap>
       <!-- change language immediately on selection change -->
          <select name="Language"
                    onchange="DoLocalLanguageChange(this)"
                    size="1">
            <option value="de" >German</option>
            <option value="en" selected >English</option>
           </select>
      </td>
   </tr>
</table><!-- Language Selection End-->
```

"de" dizininde "langswitch.html" için HTML

Almanca "langswitch.html" sayfasının başlık satırı, Almanca dilinin ayarlanmış olması hariç, İngilizce sayfasının başlık satırıyla aynıdır.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de"><meta http-
equiv="Content-Type" content="text/html; charset=utf-8">
<title>Sprachumschaltung Deutsche Seite</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

Almanca sayfasındaki HTML kodu, seçilmiş olan dilin standart değerinin Almanca ("de") olması dışında İngilizce sayfasındaki HTML kodu ile aynıdır.

```
<!-- Language Selection -->
<table>
   <tr>
     <td align="right" valign="top" nowrap>
       <!-- change language immediately on change of the selection
-->
       <select name="Language"
               onchange="DoLocalLanguageChange(this)"
               <size="1">
            <option value="de" selected >Deutsch</option>
            <option value="en" >Englisch</option>
       </select>
      </td>
   </tr>
</table><!-- Language Selection End-->
```

"script" dizinindeki "lang.js" JavaScript

"DoLocalLanguageChange()" fonksiyonu, "lang.js" dosyasında bulunur. Bu fonksiyon, "SetLangCookie()" fonksiyonunu çağırır ve ardından HTML sayfasının görüntülendiği pencereyi günceller.

"SetLangCookie()" fonksiyonu ise dokümanın "siemens_automation_language" çereze seçim listesindeki değeri atar. Bu fonksiyon ayrıca uygulamanın yolunu ayarlar, böylece, değiştirilen sayfa yerine talepte bulunan sayfa, çerezin değerini alır.

İsteğe bağlı olarak site, yorum alt bölümünde çerezin geçerliliğine yönelik bir değer sağlayabilir.

```
function DoLocalLanguageChange(oSelect) {
        SetLangCookie(oSelect.value);
        top.window.location.reload();
    }
function SetLangCookie(value) {
        var strval = "siemens_automation_language=";
        // Bu, Web sunucusunun kullandığı çerezdir
        // istenilen dili algılar.
        // Bu ad, Web sunucusu için gereklidir.
        strval = strval + value;
        strval = strval + "; path=/ ;";
        // Uygulama için yol belirtin, çünkü aksi halde
        // yol talepte bulunan tarafa yerleştirilecek
        // ve bu sayfa çerezi almayacaktır.
        /* OPTIONAL
           Bu çerezin güncel tarayıcı oturumundan daha uzun süre
           mevcut olmasını istiyorsanız, bir işlem süresini
```

```
belirtin:
            var now = new Date();
            var endttime = new Date(now.getTime() + expiration);
            strval = strval + "; expires=" +
                     endttime.toGMTString() + ";";
        */
        document.cookie = strval;
    }
```

NOT

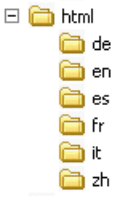
Kullanıcı tanımlı sayfa uygulamanız, hem dile özgü klasörlerdeki HTML dosyalarını (örn. en, de) hem de dile özgü olmayan klasörlerde HTML dosyalarını içeriyorsa, her iki saklama konumdaki dosyalarda Enum tiplerini tanımlamak için AWP_Enum_Def komutunu kullanamazsınız. Bu durumda Enum tiplerini kullanırken, bunları dile özgü klasörlerdeki dosyalarda veya dile özgü olmayan klasörlerdeki dosyalarda tanımlamanız gerekir. Her iki saklama konumundaki dosyalarda Enum deklarasyonları gerçekleştiremezsiniz.

12.8.10.3 STEP 7'nin çok dilli bir sayfa yapısı kullanacak şekilde yapılandırılması

Çok dilli kullanıcı tanımlı web sitelerini yapılandırırken, Kullanıcı tanımlı web sitelerin yapılandırılmasındakine (Sayfa 857) benzer şekilde ilerleyin. Ancak diller için klasörler oluşturduysanız, HTML dizini ayarlarında dil klasörleri için ilgili klasörleri münferit olarak belirtin. HTML dizinini, dil klasörlerinden biri olarak ayarlamazsınız.

HTML standart sayfasını seçerseniz, dil klasörüne gidin ve başlangıç sayfası olmasını istediğiniz HTML sayfasını seçin. Bunun ardından bloklar oluşturur ve bu blokları CPU'ya yüklerseniz, Web sunucusu yapılandırılmış dil klasöründen başlangıç sayfasını görüntüler.

Örnek: Burada gösterilen klasör yapısı C:\ üzerindeyse, HTML dizini için ayar C:\html olur. Eğer önce İngilizce sayfanın görüntülenmesini istiyorsanız, HTML standart sayfasını ayarlamak için en\langswitch.html yoluna gidin.



12.8.11 Kullanıcı tanımlı web sitelerin genişletilmiş kumandası

Kullanıcı tanımlı web siteleriniz için veri blokları oluşturduğunuzda, STEP 7, kullanıcı tanımlı sitelerin görüntülenmesini ve bunlarla etkileşimin sağlanması için bir kumanda DB'si oluşturur. STEP 7 ayrıca, her sayfayı temsil edecek bir DB fragmentleri seti de oluşturur. Normal koşullarda, kumanda DB'sinin yapısını veya onu nasıl değiştireceğinizi bilmenize gerek yoktur.
Örneğin bir web uygulamasını açıp kapatmak veya münferit fragmentleri manuel olarak değiştirmek istediğinizde, bunu kumanda DB'si ve WWW yordamının değişkenleri üzerinden yapabilirsiniz.

Kumanda DB'sinin yapısı

Kumanda DB'si kapsamlı bir veri yapısına sahiptir ve STEP 7 kullanıcı programınızı programladığınızda erişilebilir olur. Burada, kumanda DB'sinin değişkenlerinden yalnızca bazıları açıklanmaktadır.

Commandstate yapısı

"Commandstate", Web sunucusu için global komutları ve global durumları içeren bir yapıdır.

"Commandstate" yapısındaki genel komutlar

Global komutlar genel olarak Web sunucusu için geçerlidir. Web sunucusunu, kumanda DB'sinin parametreleri üzerinden devre dışı bırakabilir veya yeniden başlatabilirsiniz.

Blok değişkeni	Veri tipi	Açıklama
init	BOOL	Kumanda DB'sinin değerlendirilmesi ve web uygulamasının başlatılması
deactivate	BOOL	Web uygulamasının devre dışı bırakılması

Commandstate yapısındaki global durumlar

Global durumlar genel olarak Web sunucusu için geçerlidir ve web uygulamasına ilişkin durum bilgilerini içerir.

Blok değişkeni	Veri tipi	Açıklama
initializing	BOOL	Web uygulaması kumanda DB'sini okur
error	BOOL	Web uygulaması başlatılamadı
deactivating	BOOL	Web uygulaması sonlandırılır
deactivated	BOOL	Web uygulaması sonlandırıldı
initialized	BOOL	Web uygulaması başlatıldı
last_error	INT	WWW için geri dönüş kodu 16#0010 olduğunda bir WWW yordam çağrısından (Sayfa 859) geri döndürülen son hata: 16#0001: Yapı DB'si fragmenti tutarsız 16#0002: Uygulama adı halen mevcut 16#0003: Kaynak yok (bellek) 16#0004: Kumanda DB yapısı tutarsız 16#0005: DB fragmenti mevcut değil 16#0006: DB fragmenti, AWP için değil 16#0007: Numaralandırma verileri tutarsız

Blok değişkeni	Veri tipi	Açıklama
		16#000D: Kumanda DB'sinin boyutu nedeniyle çakışma

Talepler tablosu

Talepler tablosu, münferit DB fragmentlerine uygulanan komutları ve durumları içeren bir veri yapıları Array'idir. AWP_Start_Fragment (Sayfa 853) komutuyla "manuel" tipinde fragmentler oluşturduysanız, STEP 7 kullanıcı programı bu sitelere kumanda DB'si aracılığıyla kumanda etmelidir. Talep durumları, yazma korumalıdır ve güncel fragment hakkında bilgi sağlar. Güncel fragmente kumanda etmek için talep komutlarını kullanırsınız.

Blok değişkeni	Veri tipi	Açıklama
requesttab	ARRAY [1 .. 4] OF STRUCT	Münferit DB fragmentlerine kumanda etmek için veri yapıları Array'i Web sunucusu aynı anda en fazla dört fragmenti işleyebilir. Web sunucusu birden çok fragmenti veya birden çok tarayıcı oturumundan gelen fragmentleri işliyorsa, belirli bir fragment için Array indeksi isteğe bağlıdır.

Requesttab yapısının Struct elemanları

Blok değişkeni	Veri tipi	Açıklama
page_index	UINT	Güncel web sitesinin numarası
fragment_index	UINT	Güncel fragment numarası - başka bir fragment belirlemek mümkün
// Talep komutları		
continue	BOOL	Güncel sayfayı veya fragmenti göndermek için serbest bırakır ve bir sonraki fragment ile devam eder
repeat	BOOL	Güncel sayfayı veya fragmenti yeniden göndermek için serbest bırakır ve aynı fragment ile devam eder
abort	BOOL	HTTP bağlantısını göndermeden kapatır
finish	BOOL	Bu fragmenti gönderin; Sayfa tamamlandı - daha fazla fragment düzenlemeyin
// Talep durumları		Talep durumları, yazma korumalıdır.
idle	BOOL	Yürütülmez, ancak aktif
waiting	BOOL	Fragment, serbest bırakmayı bekliyor
sending	BOOL	Fragment gönderiliyor
aborting	BOOL	Kullanıcı, güncel talebini iptal etti

Çalışma şekli

Programınız kumanda DB'sini her değiştirdiğinde, değiştirilen kumanda DB'sinin numarasıyla bir parametre olarak WWW yordamını çağırır. STEP 7 kullanıcı programı WWW yordamını (Sayfa 859) yürüttüğünde, global komutlar ve talep komutları etkili olur.
STEP 7 kullanıcı programı, fragment_index'i açık bir şekilde belirler, Web sunucusunun belirtilen fragmenti bir talep komutuyla işlemesine neden olur. Aksi takdirde WWW yordamı yürütüldüğünde, Web sunucusu güncel sayfa için güncel fragmenti işleyecektir.
fragment_index'i kullanmak için teknikler şunlardır:

- Güncel fragmentin işlenmesi: Fragment_index'i değiştirmeden bırakır ve devam komutunu ayarlar.
- Güncel fragmentin atlanması: fragment_index'i 0 olarak ayarlar ve devam komutunu ayarlar
- Güncel fragmenti başka bir fragment ile değiştirme: fragment_index için yeni fragment ID'si belirtir ve devam komutunu ayarlar

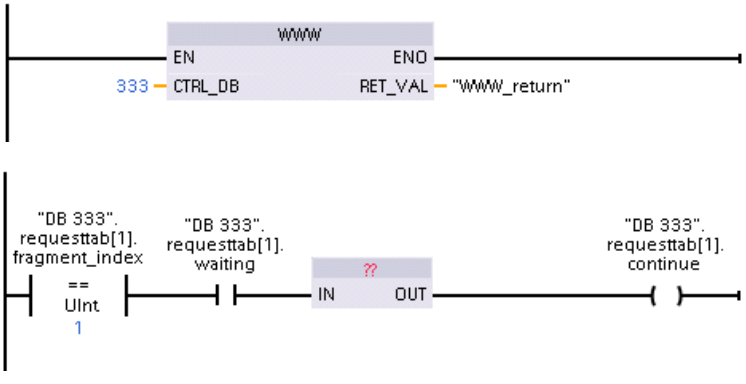
Olası değişen global durumları veya talep durumlarını sorgulamak için, STEP 7 kullanıcı programı, bu durumların güncel değerlerini değerlendirmek üzere WWW yordamını çağırır. Burada tipik kullanım, belirli bir koşul oluşana kadar WWW yordamını düzenli olarak çağırmak olabilir.

NOT

STEP 7 kullanıcı programı birden çok talep komutu oluşturursa, WWW yordamı bunlardan yalnızca birini işler ve bu bağlamda şu sırayı izler: iptal et, sonlandır, tekrar et, devam et. WWW yordamı, işlenen tüm talep komutlarını siler.

Örnekler

Aşağıdaki örnekte, daha önce WWW yordamını çağırdıktan sonra bekleme durumunda ID 1'e sahip bir fragmente kumanda eden bir STEP 7 kullanıcı programı gösterilmektedir. Ancak, uygulamaya özgü diğer koşulların oluşmasını da bekliyor olabilir. Ardından fragment için gerekli olan işlemleri gerçekleştirir, örn. veri bloğu değişkenlerini ayarlar, hesaplamaları yürütür veya diğer uygulamaya özgü görevleri işler. Bundan sonra, Web sunucusunun bu fragmenti yürütmesi için devam markörünü ayarlar.



Program, bu değiştirilmiş kumanda DB'si ile WWW yordamını çağırdığında, web tarayıcısı bu fragment ile kullanıcı tanımlı web sitesini görüntüleyebilir.



Ancak bunun basitleştirilmiş bir örnek olduğunu dikkate alın. Denetlenecek fragment, Array'in dört requesttab yapısından herhangi birinde olabilir. Programınız, bu dört requesttab yapısının tümünü işlemelidir.

12.8.12 Web API'si

S7 1200 CPU, proses verilerini okumak ve yazmak için arayüz görevi gören bir Web-API de sunar. S7-1200 Web API, S7-1500 Web API'nin fonksiyonelliğini gerçekleştirir, bu fonksiyonlar bu dokümanın (https://support.industry.siemens.com/cs/us/en/view/59193560) **web siteleri> Uygulama programlama arayüzü (API)** bölümünde açıklanmaktadır.

S7 1500 dokümanı, belirli bir S7-1500 CPU seti için Web API fonksiyonelliğini açıklamaktadır. Bu fonksiyonellik, V4.x Firmware'e sahip tüm S7 1200 CPU'larda sunulmaktadır. STEP 7'de cihaz konfigürasyonunu V4.x olarak yapılandırmanız gerekir.

S7-1200 CPU'nun eş zamanlı API oturumlarının sayısını 50 ile sınırlı olduğunu unutmayın.

S7-1200 Web API, aşağıdaki veri tipleri haricinde S7-1500 Web API tarafından desteklenen veri tiplerini destekler:

Veri tipi
ARRAY
BCD16
BCD32
VARIANT
REMOTE
HW_DEVICE
TP_TIME
TON_TIME
TOF_TIME
TONR_TIME
CTU_SINT
CTU_INT
CTU_DINT
CTU_USINT
CTU_UINT
CTU_UDINT
CTD_SINT
CTD_INT
CTD_DINT
CTD_USINT
CTD_UINT
CTD_UDINT
CTUD_SINT
CTUD_INT
CTUD_DINT
CTUD_USINT
CTUD_UINT
CTUD_UDINT

S7-1500 Web API tarafından desteklenen veri tiplerini şurada bulabilirsiniz: Bu dokümanda (https://support.industry.siemens.com/cs/us/en/view/59193560), **Web siteleri > Uygulama programlama arayüzü (API) > Proses verilerini okuma ve yazma > Desteklenen veri tipleri**bölümünde.

12.8.12.1 Desteklenen Web API yöntemleri

S7-1200, aşağıdaki Web API yöntemlerini destekler:

```
Desteklenen Web API yöntemleri
Api.Ping
Api.Version
Api.GetCertificateUrl
Api.Browse
Api.Login
Api.Logout
Api.GetPermissions
PlcProgram.Browse
PlcProgram.Write
PlcProgram.Read
```

12.9 Sınırlamalar

Aşağıdaki BT hususları, Web sunucusu kullanımınızı etkileyebilir:

- Genel olarak, standart web sitelerine veya kullanıcı tanımlı web sitelerine erişmek için, CPU'nun IP adresi veya port numarasına sahip bir kablosuz yönlendiricinin IP adresi kullanılmalıdır. Web tarayıcınız bir IP adresine doğrudan bağlantıya izin vermiyorsa, bu sorun için BT yöneticinizle iletişime geçin. Yerel direktifler DNS'yi destekliyorsa, bir DNS kaydı kullanarak IP adresine bağlanabilirsiniz.
- Güvenlik duvarları, proxy ayarları ve konuma özgü diğer kısıtlamalar da CPU'ya erişimi sınırlandırabilir. Bu tür sorunları gidermek üzere BT yöneticinizle iletişime geçebilirsiniz.
- Standart web siteleri JavaScript ve çerezleri kullanır. Web tarayıcınızda JavaScript veya çerezler devre dışıysa, bunları etkinleştirin. Bunları etkinleştiremezseniz, bazı fonksiyonlar kısıtlanacaktır (Sayfa 886). Kullanıcı tanımlı web sitelerinde JavaScript ve çerezlerin kullanımı isteğe bağlıdır. Kullanmak isterseniz, bunları tarayıcınızda etkinleştirilmeleri gerekir.
- Web sunucusu, Secure Sockets Layer - Güvenli soket katmanını (SSL) destekler. Standart web sitelerine ve kullanıcı tanımlı sitelere, http://ww.xx.yy.zz veya https://ww.xx.yy.zz adresli iki URL'den birini kullanarak erişebilirsiniz. "ww.xx.yy.zz" CPU'nun IP adresidir.
- Siemens, Web sunucusuna güvenli erişim için bir güvenlik sertifikası sunmaktadır. Bu sertifikayı, "Giriş" standart web sitesinden (Sayfa 813) indirebilir ve web tarayıcınızın internet opsiyonlarına içe aktarabilirsiniz (Sayfa 806). Sertifikayı içe aktarmadığınız takdirde, https:// adresi üzerinden Web sunucusunu her çağırdığınızda bir güvenlik sorgulaması görüntülenecektir.

Bağlantıların sayısı

Web sunucusu, maksimum 30 aktif bağlantıyı destekler. Bu 30 bağlantı, kullandığınız web tarayıcısına ve sayfa başına nesne sayısına (CSS dosyaları, görseller, JavaScript dosyaları, diğer HTML dosyaları) bağlı olarak farklı şekillerde kullanılabilir. Web sunucusu bir sayfayı görüntülerken bazı bağlantılar açık kalır, diğerleri ise ilk bağlantıdan sonra serbest bırakılır. Örneğin Mozilla Firefox'un maksimum altı kalıcı bağlantılı sürümünü kullandığınızda, Web sunucusu bağlantıları kesmeye başlamadan önce beş tarayıcı veya tarayıcı sekmesi kullanabilirsiniz. Eğer bir site bu altı bağlantının tümünü kullanmıyorsa, ek tarayıcılar veya tarayıcı sekmeleri kullanabilirsiniz.
Ayrıca aktif bağlantı sayısının ilgili sitenin performansını etkileyebileceğini de dikkate almalısınız. Bu nedenle bazı durumlarda web siteleri tam olarak yüklenmeyebilir.

NOT

Web sunucusunu kapatmadan önce oturumunuzu kapatın

Web sunucusunda oturum açtıysanız, web tarayıcısını kapatmadan önce mutlaka oturumu kapatmalısınız. Web sunucusu aynı anda en fazla yedi kullanıcıyı destekler.

Tarayıcınıza bağlı olarak, oturumu kapatmadığınızda, bu aynı anda birkaç bağlantının açılmasına neden olabilir. Web sunucusu tarayıcı pencerelerini oturumu kapatmadan birden çok kez açıp kapatırsanız, 30 bağlantının tamamını açıp kullanıyor olabilirsiniz. Tüm bağlantıları doldurursanız, bir sonraki oturum açmayı denediğinizde "Geçersiz oturum açma" mesajı görüntülenir. Bunun ardından, Web sunucusunun tekrar oturum açmanızı sağlamak adına yeterli bağlantıyı serbest bırakması için 30 dakikaya kadar beklemeniz gerekecektir. Bu sorundan kaçınmak için, Web sunucusunu kapattığınızda her zaman oturumu da kapatın (eğer oturum açtıysanız).

12.9.1 JavaScript kullanımı

Standart web siteleri; HTML, JavaScript ve çerezlerini kullanır. Bulunduğunuz konum, JavaScript ve çerezlerinin kullanımını kısıtlıyorsa, web sitelerinin düzgün çalışması için bunları etkinleştirmelisiniz. Web tarayıcınız için JavaScript'i etkinleştiremezseniz, standart web siteleri çalışamaz. Ancak JavaScript gerektirmeyen temel siteleri kullanabilirsiniz.

Ayrıca bakınız

Standart web sitelerinin yapısı (Sayfa 808)

12.9.2 İnternet opsiyonlarında çerezlere izin verilmezse fonksiyonlar kısıtlıdır

Çerezler web tarayıcınızda devre dışı bırakılmışsa, aşağıdaki sınırlamalar söz konusu olur:

- Oturum açamazsınız.
- Dili değiştiremezsiniz.
- UTC'den PLC saatine geçiş yapamazsınız. Çerezler olmazsa, tüm zamanlar UTC saatleridir.

12.9.3 Değişken adlarını ve değerlerini girme kuralları

Değişken durumu (Sayfa 827) ve İzleme tabloları (Sayfa 829) standart siteleri kullanırken aşağıdakileri dikkate alın:

- Bir DTL değişkeninin tüm değerini değiştirirseniz, örn. "VeriBloğu_1_.DTL_Variable, değişiklik değeri için şu DTL sözdizimini kullanmalısınız:
  DTL#YYYY-AA-GG-SS-DD-SS[.sssssssss]
- Üstel gösterimde Real veya LReal veri tipinde bir değer girerken, aşağıdaki yöntemi izleyin:
  - Pozitif üslü (+3.402823e+25 gibi) bir gerçek sayı değerini (Real veya LReal) girerken, değeri aşağıdaki formatlardan birinde girin:
    +3.402823e25
    +3.402823e+25
  - Negatif üslü (+3.402823e-25 gibi) bir gerçek sayı değeri (Real veya LReal) girerken, değeri şu şekilde girin:
    +3.402823e-25
  - Üstel gösterimde gerçek sayı değerinin mantis kısmının bir ondalık nokta içermesini sağlayın. Ondalık nokta kullanmazsanız, değerin beklenmeyen bir tamsayı değerine değiştirilmesi gerçekleşebilir. Örnek: -1.0e8 değerini girin, -1e8 değil.
- LReal değerlerinin yalnızca 15 basamak içermesine izin verilir (ondalık noktanın konumundan bağımsız olarak). 15 basamaktan fazla girerseniz, yuvarlama hatası gerçekleşir.

Değişken durumu ve İzleme tablosu sayfasındaki sınırlamalar:

- URL için maksimum karakter sayısı 2083'tür. Güncel değişkenler sayfanıza karşılık gelen URL'yi tarayıcınızın adres çubuğunda görürsünüz.
- Karakter görüntüleme formatıyla ilgili olarak, tarayıcı fiili CPU değerlerini geçerli olmayan ASCII karakterleri olarak değerlendirirse, sayfa, önünde Dolar işaretli karakter görüntüler: $.

12.9.4 Veri protokollerinin CSV formatında Microsoft Excel'in ABD/İngilizce dışındaki sürümlerine içe aktarılması

Protokol dosyaları CSV (virgülle ayrılmış değerler) dosya formatında biçiminde saklanır. Sisteminizde Excel'in Amerikan veya İngilizce sürümü yüklü ise, bu dosyaları doğrudan veri protokolü sayfasında, Excel ile açabilirsiniz. Ancak diğer ülkelerde, virgüller genellikle sayısal gösterimde kullanıldığı için bu format o kadar yaygın olarak kullanılmaz.

Excel'in Amerikan/İngilizce sürümünü kullanmıyorsanız, saklanmış bir protokol dosyasını açmak için aşağıdaki yöntemi izleyin:

1. Microsoft Excel'i açın ve boş bir Excel çalışma kitabı oluşturun.
2. "Veriler > Harici verileri içe aktar" menüsünde "Verileri içe aktar" komutunu seçin.
3. Açmak istediğiniz protokol dosyasına gidin ve dosyayı seçin. Metin içe aktarma sihirbazı çağrılır.
4. Metin içe aktarma sihirbazında, "Başlangıçtaki veri tipi" standart opsiyonunu "Sabit genişlik" yerine "Sınırlandırılmış" olarak değiştirin.
5. "Devam" butonunu seçin.
6. "Adım 2" iletişim kutusunda, "Sekme" ayırma işareti tipini "Virgül" olarak değiştirmek için "Virgül" opsiyon kutusunu işaretleyin.
7. "Devam" butonunu seçin.

8. "Adım 3" iletişim kutusunda isteğe bağlı olarak veri formatını MDJ'den (ay/gün/yıl) başka bir formata değiştirebilirsiniz.
9. Dosyayı içe aktarmak için, metin içe aktarma sihirbazındaki kalan adımları tamamlayın.

13

İletişim işlemcisi ve Modbus TCP

13.1 Seri iletişim arayüzleriyle çalışma

İki iletişim modülü (CM'ler) ve bir iletişim kartı (CB), PtP iletişimi için arayüz sağlar:

- CM 1241 RS232 (Sayfa 1341)
- CM 1241 RS422/485 (Sayfa 1342)
- CB 1241 RS485 (Sayfa 1339)

En fazla üç CM (her tipten) artı bir CB, toplam dört iletişim arayüzü bağlayabilirsiniz. CM'yi CPU'nun veya başka bir CM'nin soluna takın. CB'yi CPU'nun ön tarafına takın. Modüllerin takılması ve sökülmesiyle ilgili daha fazla bilgiyi, montaj direktiflerinde (Sayfa 49) bulabilirsiniz.

Seri iletişim arayüzlerinin özellikleri şunlardır:

- Potansiyeli ayrı bağlantı
- Noktadan noktaya protokol desteği
- Noktadan noktaya iletişim işlemcisinin yordamları ile yapılandırma ve programlama
- Gönderim ve alım etkinliğinin LED'ler aracılığıyla gösterilmesi
- Teşhis LED'i (yalnızca CM'ler)
- CPU üzerinden gerilim beslemesi: Harici güç kaynağı gerekmez.

İletişim arayüzlerinin (Sayfa 1329) teknik verilerine bakın.

LED göstergeler

İletişim modüllerinde üç LED göstergesi bulunur:

- Teşhis LED'i (DIAG): Bu LED, CPU tarafından adreslenene dek kırmızı renkte yanıp söner. CPU'nun yol almasından sonra, CM'lere kumanda eder ve onları adresler. Teşhis LED'i yeşil renkte yanıp sönmeye başlar. Bu, CPU'nun CM'ye adreslendiği, ancak henüz ona bir konfigürasyon atamadığı anlamına gelir. Program CPU'ya yüklendiğinde, CPU, konfigürasyonu yapılandırılmış CM'lere yükler. CPU'ya yüklendikten sonra, iletişim modülündeki teşhis LED'i sürekli yeşil renkte yanmalıdır.
- Gönderim LED'i (Tx): İletişim portu üzerinden veri gönderilirken gönderim LED'i yanar.
- Alım LED'i (Rx): Bu LED, iletişim portu üzerinden veri alındığında yanar.

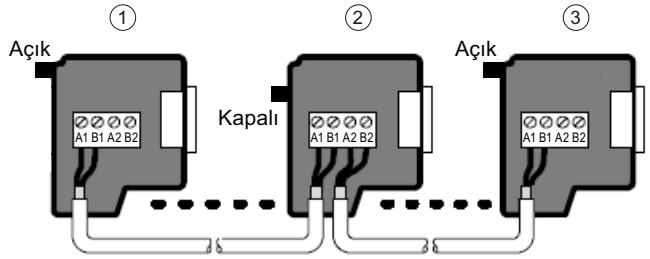
İletişim kartı, gönderim (TxD) ve alım (RxD) için LED'ler sunar. Teşhis LED'i yoktur.

13.2 Bir RS485 veriyolu bağlantı fişinin sonlandırılması

Siemens, birkaç cihazı kolayca bir RS485 ağına bağlayabileceğiniz bir RS485 veriyolu bağlantı fişi (Sayfa 1356) sunmaktadır. Fiş, şebeke için giriş ve çıkış kablolarını bağlamanızı sağlayan iki klemens seti ile donatılmıştır. Fişte ayrıca ağı seçerek sonlandırmak için şalter de bulunur.

NOT

RS485 ağının yalnızca iki ucunu sonlandırırsınız. İki uç cihaz arasındaki cihazlar sonlandırılmaz. Çıplak kablo blendajı: Metal kılavuz üzeri yaklaşık 12 mm çıplak olmalıdır.

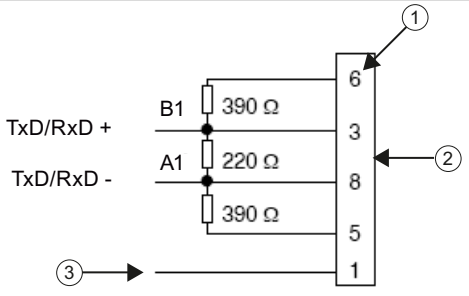
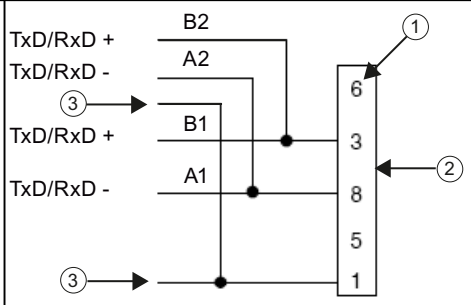


① Şalter pozisyonu = Açık: Sonlandırma direnci devrede

② Şalter pozisyonu = Kapalı: Sonlandırma direnci devrede değil

③ Şalter pozisyonu = Açık: Sonlandırma direnci devrede

Tablo 13-1 RS485 fişinin sonlandırılması

Sonlandırma cihazı (sonlandırma AÇIK)	Sonlandırma cihazı yok (Sonlandırma KAPALI)
	

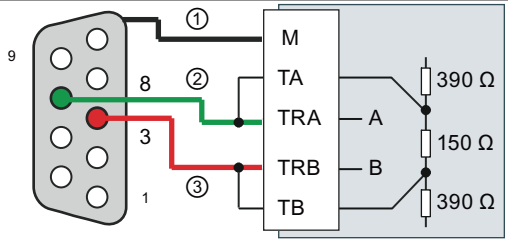
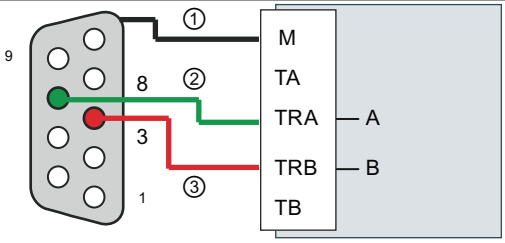
① Pin numarası

② Veriyolu bağlantı fişi

③ Kablo blendajı

CB 1241, ağ sonlandırma için dahili dirençler sunar. Bağlantıyı sonlandırmak için, devredeki dahili dirençleri karşılamak üzere TRA'yı TA'ya ve TRB'yi TB'ye bağlayın. CB 1241'de 9 kutuplu soket bağlayıcı yoktur. Aşağıdaki tablo, iletişim partnerindeki 9 kutuplu bir soket bağlayıcıya yapılan bağlantıları göstermektedir.

Tablo 13-2 CB 1241'in sonlandırılması

Sonlandırma cihazı (sonlandırma AÇIK)	Sonlandırma cihazı yok (Sonlandırma KAPALI)
	

① M'yi kablo blendajına bağlayın

① A = TxD/RxD - (yeşil tel/kutup 8)

② B = TxD/RxD + (kırmızı tel/kutup 3)

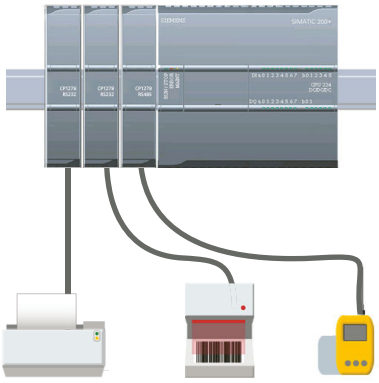
13.3 Noktadan noktaya iletişim (PtP)

CPU, karakter tabanlı seri protokoller için aşağıdaki noktadan noktaya (PtP) iletişimi desteklemektedir.

- PtP, serbestçe programlanabilir iletişim (Sayfa 891)
- PtP, 3964(R) (Sayfa 893)
- USS (Sayfa 942)
- Modbus (Sayfa 954)

13.3.1 PtP, serbestçe programlanabilir iletişim

Serbestçe programlanabilen iletişim için noktadan noktaya protokol, en yüksek seviyede serbestlik ve esneklik sunar, ancak kullanıcı programının hayata geçirilmesi zahmetli olur.



PtP, çeşitli olanaklar sunar:

- Verilerin, örneğin yazıcı gibi harici bir cihaza doğrudan gönderilmesi
- Barkod okuyucular, RFID okuyucular, diğer üreticilerin kamera veya denetleme sistemleri gibi diğer cihazlardan ve diğer birçok cihaz türünden verilerin alınabilmesi
- GPS cihazları, diğer üreticilerin kamera veya denetleme sistemleri, kablosuz modemler ve diğer pek çok cihaz gibi diğer cihazlarla bilgi alışverişinde bulunma, veri gönderimi ve alımı

Bu PtP iletişimi, çeşitli Baud hızlarını ve pariteleri destekleyen standart UART'ları kullanan bir seri iletişimdir. RS232 ve RS422/485 iletişim modülleri (CM 1241) ve yine aynı şekilde RS485 iletişim kartı (CB 1241), PtP iletişimi için elektriksel arayüzler sağlar.

PROFIBUS veya PROFINET üzerinden serbestçe programlanabilen PtP iletişimi

PtP sayesinde çeşitli cihazlarla (RFID okuyucular, GPS ve diğerleri) iletişim kurmak için bir desantral PROFINET veya PROFIBUS çevrebirim modülü kullanabilirsiniz:

- PROFINET (Sayfa 561): S7-1200 CPU'nun Ethernet arayüzü bir PROFINET arayüz modülüne bağlıdır. PtP cihazlarıyla seri iletişim, arayüz modülü ile modül taşıyıcısındaki PtP iletişim modülleri aracılığıyla mümkün olur.
- PROFIBUS (Sayfa 734): S7-1200 CPU ile modül taşıyıcısının sol tarafına bir PROFIBUS iletişim modülü takılır. PROFIBUS iletişim modülü, bir PROFIBUS arayüz modülü içeren bir modül taşıyıcısına bağlanır. PtP cihazlarıyla seri iletişim, arayüz modülü ile modül taşıyıcısındaki PtP iletişim modülleri aracılığıyla mümkün olur.

Bu nedenle S7-1200, iki set PtP yordamını destekler:

- Daha eski noktadan noktaya yordamlar (Sayfa 1045): Bu yordamlar, S7-1200'ün V4.0 sürümünden önce kullanılmaktaydı ve yalnızca bir CM 1241 iletişim modülü veya bir CB 1241 iletişim kartı kullanarak seri iletişim mümkündü.
- Noktadan noktaya yordamlar (Sayfa 908): Bu yordamlar, eski yordamların tüm fonksiyonlarını ve ayrıca desantral PROFINET ve PROFIBUS çevrebirimler aracılığıyla PtP iletişim modüllerini destekleme olanağını da sağlar. Noktadan noktaya yordamları kullanarak desantral çevrebirim modülü aracılığıyla iletişim modüllerine erişebilirsiniz.

  S7 1200 CM 1241 modülleri, noktadan noktaya yordamları kullanabilmek için en azından Firmware V2.1 sürümünü gerektirir. Bu modüller yalnızca S7-1200 CPU'nun sol tarafındaki yerel modül taşıyıcısına bağlanabilir. Noktadan noktaya yordamlar CB 1241 için de kullanılabilir.

Aşağıdaki modüller, desantral çevrebirimler aracılığıyla iletişim için kullanılır:

İstasyon	Modül	Sipariş numarası	Arayüz
ET 200MP	CM PtP RS232 BA	6ES7540-1AD00-0AA0	RS232
	CM PtP RS232 HF	6ES7541-1AD00-0AB0	RS232
	CM PtP RS422/485 BA	6ES7540-1AB00-0AA0	RS422/RS485
	CM PtP RS422/485 HF	6ES7541-1AB00-0AB0	RS422/RS485
ET 200SP	CM PtP	6ES7137-6AA00-0BA0	RS232 ve RS422/RS485

NOT

Noktadan noktaya yordamlar; bir iletişim kartına, yerel seri modüllere (sol tarafta), PROFINET üzerinden seri modüllere ve PROFIBUS üzerinden seri modüllere erişmenizi sağlar. STEP 7, mevcut programları desteklemek için yalnızca eski PtP yordamlarını sunmaktadır. Eski yordamlar, hâlâ güncel S7-1200 CPU'larla çalışmaktadır. Eski programları, yeni yordamlara dönüştürmeye gerek yoktur.

NOT

Saat senkronizasyonu ve PtP iletişimi için CM Firmware sürümü gereklilikleri

Cihaz konfigürasyonunda, Profinet arayüzü için Saat senkronizasyonu (Sayfa 166) özelliklerinde "CPU, cihazın modüllerini senkronize eder" opsiyonu etkinleştirilmişse, bağlı iletişim modüllerinin Firmware sürümlerini en son sürümlere güncellemelisiniz. Daha eski Firmware sürümlerine sahip iletişim modülleri için saat senkronizasyonunun etkinleştirilmesi, iletişim hatalarına yol açabilir.

13.3.2 3964(R) iletişimi

S7-1200 CPU, bir CM 1241 modülü (RS-232) veya bir CM 1241 modülü (RS-422/RS-485) ile 3964(R) protokolünü kullanan bir iletişim partneri arasındaki iletişim için 3964(R) protokolünü destekler. Yukarıda değinilen PtP iletişiminin aksine 3964(R) protokolü, mesajlar için belirli gönderim veya aktarma ve alım özelliklerini belirlediğiniz sıkı bir protokol talep eder, ilgili kontrol karakterleri şunlardır:

- STX Start of Text
  Aktarılacak karakter dizisinin başlangıcı
- DLE Data Link Escape
  Veri aktarımı geçişi
- ETX End of Text
  İletilecek karakter dizisinin sonu
- BCC Block Check Character – Blok kontrol karakteri
- NAK Negative Acknowledge – Negatif onaylama

Protokolün tam bir açıklamasını, işletme kitabındaki seri veri aktarımının ilkelerini ele alan İşletme kitabı - S7-300 noktadan noktaya bağlantı CP 341 kurulumu ve parametrelendirme. (https://support.industry.siemens.com/cs/us/en/view/1117397) bölümünde bulabilirsiniz.

İletişim modülünün yapılandırılması

3964(R) protokolü üzerinden bir partner ile iletişim kurmak için, STEP 7'deki cihaz konfigürasyonuna aşağıdaki iletişim modüllerinden birini eklemeniz gerekir:

- CM 1241, RS232
- CM 1241, RS-422/RS-485

CM modülünün Firmware sürümü, V2.2.0 veya üstü olmalıdır.
İletişim modülü için iletişim portlarını (Sayfa 894), önceliği ve protokol parametrelerini yapılandırırsınız (Sayfa 907).

3964(R) protokolü üzerinden bir partnerle iletişim

Bir Cm'yi 3964(R) protokolü için yapılandırdığınızda, CPU ile iletişim partneri arasında veri aktarmak için standart noktadan noktaya gönderim ve alım yordamlarını kullanırsınız.
CM, gönder yordamının BUFFER parametresindeki verilerinizi 3964(R) protokolüne yerleştirir ve verileri iletişim partnerine gönderir.
CM, 3964(R) protokolünü kullanarak iletişim partnerinden verileri alır, protokol bilgilerini kaldırır ve verileri alım yordamının BUFFER parametresine verir.
Daha fazla bilgi için, aşağıdaki noktadan noktaya yordamlara bakın:

- Send_P2P (gönderim tamponu verilerinin aktarılması) (Sayfa 921)
- Receive_P2P (mesaj alımı etkinleştirme) (Sayfa 924)

Eski noktadan noktaya gönderim ve alım yordamlarını da kullanabilirsiniz:

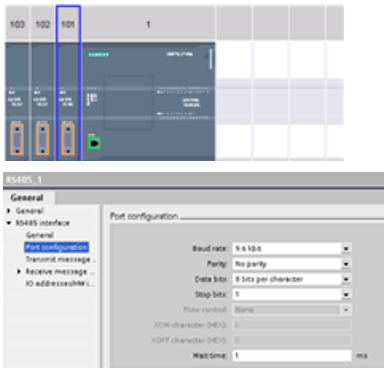
- SEND_PTP (gönderim tamponu aktarımı) (Sayfa 1053)
- RCV_PTP (alım mesajlarını etkinleştirme) (Sayfa 1054)

13.3.3 Serbestçe programlanabilen PtP iletişiminin konfigürasyonu

Serbestçe programlanabilen PtP iletişimine yönelik iletişim arayüzlerini yapılandırmak için aşağıdaki prosedürler mevcuttur:

- STEP 7'deki cihaz konfigürasyonunda port parametrelerini (Baud hızı ve parite), gönderim parametrelerini ve alım parametrelerini yapılandırın. CPU, cihaz konfigürasyonundaki ayarları saklar ve yeniden başlatmadan ve RUN'dan STOP işletim durumuna geçişin ardından ayarları üstlenir.
- Port_Config (Sayfa 910), Send_Config (Sayfa 913) ve Receive_Config (Sayfa 914) yordamlarını kullanarak parametreleri ayarlayın. Yordamlar tarafından belirlenen port ayarları, CPU, RUN işletim türündeyken geçerlidir. Port ayarları, STOP durumuna değiştirdikten veya yeniden başlatmadan sonra cihaz konfigürasyonu ayarlarına geri döner.

Donanım cihazlarının konfigürasyonundan (Sayfa 128) sonra, modül taşıyıcınızdaki CM'lerden birini veya yapılandırılmışsa CB'yi seçerek iletişim arayüzlerini parametrelendirirsiniz.



Denetçi penceresindeki "Özellikler" sekmesinde seçilmiş olan CM veya CB'nin parametreleri görüntülenir. Aşağıdaki parametreleri düzenlemek için, "Port konfigürasyonu" öğesini seçin:

- Baud hızı
- Parite
- Karakter başına veri Bit'i
- Stop Bit'lerinin sayısı
- Akış kontrolü (yalnızca RS232)
- Bekleme süresi

CM 1241 RS232 ve CB RS485 için (yalnızca CM 1241 RS232 tarafından desteklenen akış kontrolü (Sayfa 896) hariç), RS232 ve RS485 iletişim modülleri ve RS485 iletişim kartı aynı port konfigürasyonu parametrelerine sahiptir. Parametre değerleri farklı olabilir.

CM 1241 RS422/485'te, aşağıda gösterildiği gibi ek port konfigürasyonu opsiyonları bulunur. CM 1241 RS422/485'in 422 modu, yazılım akış kontrolünü de destekler.



Aşağıdaki RS422/485 parametrelerini düzenlemek için "Port Konfigürasyonu" öğesini seçin:

- "İşletim türü":
  - Tam dupleks (RS422), dört telli mod (noktadan noktaya bağlantı)
  - Tam dupleks (RS422), dört telli mod (çok noktalı Master)
  - Tam dupleks (RS422), dört telli mod (çok noktalı Slave)
  - Yarı dupleks (RS485), iki telli mod
- "Alım hattının çıkış durumu":
  - Yok
  - İleri yönde ön gerilim (sinyal R(A) 0V, sinyal R(B) 5V)

STEP 7 kullanıcı programı ayrıca, Port_Config (Sayfa 910) yordamı ile portu yapılandırabilir veya mevcut konfigürasyonu değiştirebilir. Yordamın açıklaması, işletim türü ve hattın çıkış durumu ile diğer parametreler hakkında daha ayrıntılı bilgiler içerir.

Parametre	Tanım
Baud hızı	Baud hızı için ön ayarı, 9,6 kBit/sn'dir. Geçerli değerler: 300 Baud, 600 Baud, 1,2 kBit/sn, 2,4 kBit/sn, 4,8 kBit/sn, 9,6 kBit/sn, 19,2 kBit/sn, 38,4 kBit/sn, 57,6 kBit/sn, 76,8 kBit/sn ve 115,2 kBit/sn.
Parite	Ön ayarlı parite, "yok" şeklindedir. Geçerli değerler: Parite yok, çift sayılı, tek sayılı, işaret (parite Bit'i her zaman 1'de), boşluk (parite Bit'i her zaman 0'da).
Karakter başına veri Bit'i	Bir karakterdeki veri Bit'lerinin sayısı. 7 veya 8 ayarları geçerlidir.
Stop Bit'lerinin sayısı	Stop Bit'lerinin sayısı, bir veya iki olabilir. Ön ayar, birdir.
Akış kontrolü	RS232 iletişim modülü için donanım veya yazılım akış kontrolü (Sayfa 896) seçilebilir. Donanım akış kontrolü ayarlanmışsa, RTS sinyalinin her zaman açık olup olmayacağını veya RTS'nin anahtarlanmasını seçebilirsiniz. Yazılım akış kontrolü ayarlanmışsa, XON ve XOFF karakterlerini tanımlayabilirsiniz. RS485 iletişim arayüzleri akış kontrolünü desteklemez. CM 1241 RS422/485'in 422 modu, yazılım akış kontrolünü destekler.
Bekleme süresi	Bekleme süresi, akış kontrolünün türüne bağlı olarak, CM veya CB'nin RTS gönderip, CTS'yi veya XOFF'u aldıktan sonra XON'u almak için beklediği süredir. Eğer iletişim arayüzü beklenen bir CTS veya XON'u almadan önce bekleme süresi sona ererse, CM veya CB tarafından gönderim durdurulur ve kullanıcı programına bir hata mesajı verilir. Bekleme süresini milisaniye cinsinden girin. İzin verilen değer aralığı 0 ila 65535 milisaniyedir.
İşletim durumu	Burada, RS422 veya RS485 işletim modunu ve ağ konfigürasyonunu seçersiniz.
Alım hattının çıkış durumu	Burada ön gerilimi seçersiniz. Geçerli değerler; yok, ileri yönde ön gerilim ve blokaj yönünde ön gerilim. Blokaj yönünde ön gerilim, tel kopması algılamasını olanaklı kılar.

13.3.3.1 Akış kontrolü kumandası

Akış kontrolü, verileri gönderen ve alan birimin farklı kapasitelerini dengeleyerek hiçbir verinin kaybolmamasını sağlar. Akış kontrolü, göndericinin, alıcının işleyebileceğinden daha fazla veri aktarmamasını sağlar. Akış kontrolü, donanım veya yazılım düzeyinde gerçekleşebilir. RS232 iletişim modülü, donanım ve yazılım akış kontrolünü destekler. RS485-CM ve CB'ler, akış kontrolünü desteklemezler. CM 1241 RS422/485'in 422 modu, yazılım akış kontrolünü destekler. Hangi akış kontrolünün oluşturulacağını, arayüzü yapılandırılırken (Sayfa 894) veya PORT_CFG yordamı (Sayfa 1045-1046) üzerinden girersiniz.

Donanım akış kontrolü, RTS (Request-to-send / gönderim talebi) ve CTS (Clear-to-send / gönderim için temiz) iletişim sinyallerine dayalıdır. RS232 iletişim modülünde RTS sinyali, pin 7'den ve CTS sinyali pin 8'den alınır. RS232 iletişim modülü, çıkış olarak RTS sağlayan ve giriş olarak CTS'yi izleyen bir DTE (Data Terminal Equipment / veri uç cihazı) cihazıdır.

Donanım akış kontrolü: RTS anahtarlandı

Bir RS232 iletişim modülü için RTS anahtarlamalı donanım akış kontrolü etkinleştirildiğinde, modül veri gönderimi için RTS sinyalini 1'e ayarlar. Modül, CTS sinyalini denetler ve alıcı cihazın alıma hazır olup olmadığını saptamak için kullanır. CTS sinyali aktifse, CTS sinyali aktif kaldığı sürece modül veri aktarabilir. CTS sinyali devre dışı kalırsa, aktarım durdurulmalıdır.

CTS sinyali tekrar açıldığında, aktarım devam eder. CTS sinyali ayarlanan bekleme süresi içinde tekrar açılmazsa, modül aktarımı iptal eder ve kullanıcı programına bir hata geri döndürür. Arayüz konfigürasyonunda (Sayfa 894) bekleme süresini girmelisiniz.

RTS anahtarlamalı akış kontrolü, aktarımın aktif olduğuna dair bir sinyale gerek duyan cihazlar için yararlıdır. Bunun bir örneği, bir "şalter" sinyali olarak RTS'nin radyo vericisini etkinleştirdiği bir kablosuz modemdir. RTS anahtarlamalı akış kontrolü, geleneksel telefon modemleriyle çalışmaz. Telefon modemlerinde, "RTS her zaman açık" opsiyonunu kullanın.

Donanım akış kontrolü: RTS her zaman açık

"RTS her zaman açık" opsiyonunda CM 1241, standart olarak RTS sinyalini etkinleştirir. Telefon modemi gibi bir cihaz, CM'den gelen RTS sinyalini denetler ve bu sinyali gönderilmeye hazır bir bildirim olarak kullanır. Modem yalnızca RTS aktif iken yani telefon modemi etkin bir CTS algıladığında CM'ye gönderimde bulunur. RTS aktif değilse, telefon modemi CM'ye herhangi bir veri aktarmaz.

Modemin CM'ye herhangi bir zamanda veri gönderebilmesi için, donanım akış kontrolünde "RTS her zaman açık" ayarlanmalıdır. Bunun ardından CM, her zaman RTS sinyalini 1'e ayarlar. Modül herhangi bir karakter alamasa bile CM, RTS'yi kapatmaz. Gönderici cihaz, CM'nin alım tamponunda taşma oluşmamasını sağlamalıdır.

Veri seti hazır (DTR - Data Terminal Ready) ve veri seti hazır (DSR - Data Set Ready) sinyalleri

CM, her türlü donanım akış kontrolü için DTR'yi 1'e getirir. Aktarım, yalnızca DSR sinyali açık olduğunda yapılır. DSR'nin durumu yalnızca gönderim fonksiyonunun başlangıcında değerlendirilir. Aktarım başladıktan sonra DSR devre dışı kalırsa aktarım durmaz.

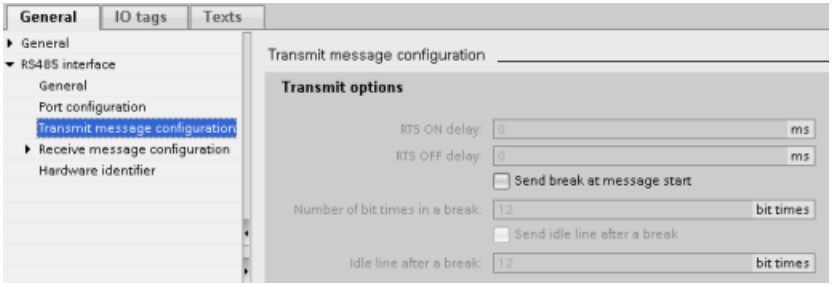
Yazılım akış kontrolü

Yazılım akış kontrolü, akış kontrol mesajlarında özel karakterler kullanır. Bu bağlamda XON ve XOFF'u temsil eden onaltılık karakterleri yapılandırabilirsiniz.
XOFF, bir aktarımın durdurulması gerektiğini belirtir. XON, bir aktarımın devam edebileceğini belirtir. XOFF ve XON, aynı karakter olamaz.
Gönderici cihaz, alıcıdan bir XOFF karakteri alırsa, aktarım durdurulur. Gönderici cihaz XON karakterini aldığında devam eder. CM, arayüz konfigürasyonunda (Sayfa 894) belirtilen bekleme süresi içinde bir XON karakteri almazsa, aktarım işlemini iptal eder ve kullanıcı programına bir hata mesajı yayınlar.
Yazılım akış kontrolü, süregelen bir aktarım sırasında alıcının göndericiye XOFF gönderebilmesi gerektiğinden tam dupleks iletişim gerektirir. Yazılım akış kontrolü, yalnızca ASCII karakterlerinden oluşan mesajlar için mümkündür. İkili protokoller, yazılım akış kontrolünü kullanamaz.

Serbestçe programlanabilen noktadan noktaya iletişimin başlatılabilmesi için mesaj gönderim ve alım parametrelerinin ayarlanması gerekir. Bu parametrelerde, mesajların bir hedef cihaza gönderimi veya bu cihazdan alınması sırasındaki iletişimin nasıl gerçekleşeceği belirlenir.

13.3.3.2 Gönderim parametrelerinin yapılandırılması

CPU'nun cihaz konfigürasyonunda, bir iletişim arayüzünün verileri nasıl göndereceğini ayarlarsınız. Bunun için, seçilen arayüzde "Gönderilen mesaj konfigürasyonu" özelliklerini belirtirsiniz.



Ayrıca, Send_Config (Sayfa 913) yordamı ile kullanıcı programında mesaj gönderimi için olan parametreleri dinamik olarak yapılandırabilir veya değiştirebilirsiniz.

NOT

Kullanıcı programında Send_Config ile girilen parametre değerleri "Gönderilen mesaj konfigürasyonu" özelliklerinin üzerine yazılır. Send_Config yordamı kullanılarak girilen parametreler, güç kapatıldığında CPU'ya kaydedilmez.

Parametre	Tanım
RTS açma gecikmesi	RTS'yi etkinleştirdikten sonra aktarım tetiklenene kadar bekleme süresi. Geçerli aralık 0 ila 65535 msn, standart değer 0'dır. Bu parametre, yalnızca arayüz konfigürasyonunda (Sayfa 894) donanım akış kontrolü ayarlanmışsa geçerlidir. RTS açma gecikmesi geçtikten sonra CTS işlenir. Bu parametre yalnızca RS232 modülleri için geçerlidir.

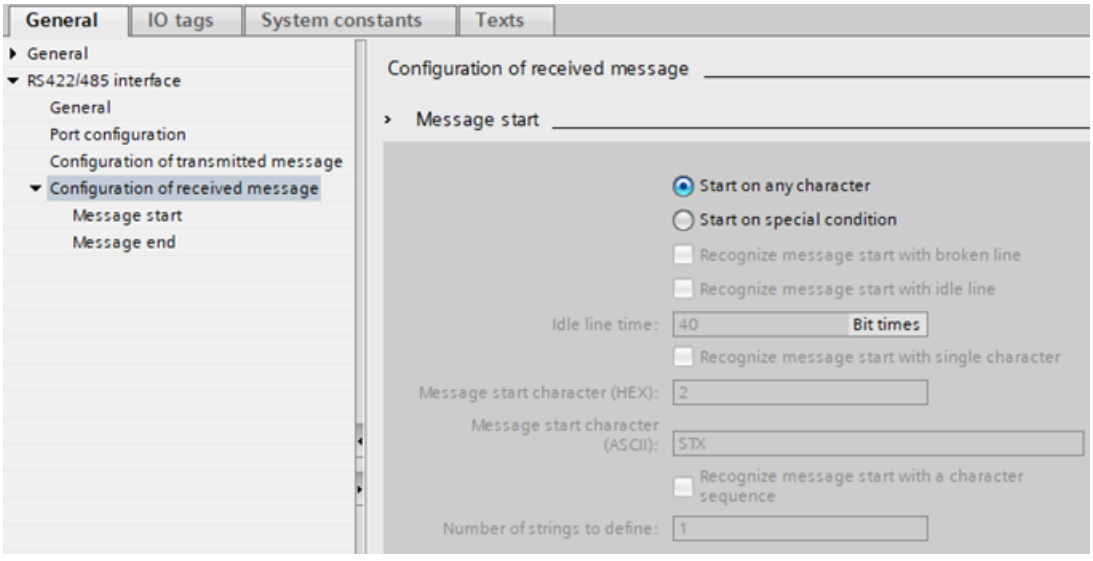
Parametre	Tanım
RTS kapatma gecikmesi	Aktarım bittikten sonra RTS devre dışı bırakılıncaya kadar bekleme süresi. Geçerli aralık 0 ila 65535 msn, standart değer 0'dır. Bu parametre, yalnızca arayüz konfigürasyonunda (Sayfa 894) donanım akış kontrolü ayarlanmışsa geçerlidir. Bu parametre yalnızca RS232 modülleri için geçerlidir.
Mesaj başlangıcında gönderim duraklaması Duraklamadaki Bit adımı sayısı	Bununla, bir mesaj başladığında, gecikmeli RTS açma gecikmesi (ayarlanmışsa) süresi dolduğunda ve CTS aktif olduğunda bir duraklama gönderileceği belirlenir. Aktarımın durdurulduğu sırada kaç Bit adımının bir duraklamayı temsil ettiğini ayarlayabilirsiniz. Ön ayar 12'dir ve maksimum, sekiz saniyelik bir limite kadar 65535'tir.
Duraklamadan sonra boş satır gönderimi Duraklamadan sonra boş satır	Mesaj başlangıcından önce bir boş satırın gönderildiğini belirtir. Bir duraklama yapılandırılmışsa, duraklamadan sonra gönderim gerçekleşir. "Duraklamadan sonra boş satır" parametresi, boş bir satıra kaç Bit adımının karşılık geldiğini belirtir. Ön ayar 12'dir ve maksimum, sekiz saniyelik bir limite kadar 65535'tir.

13.3.3.3 Alım parametrelerinin yapılandırılması

İletişim arayüzünde olduğu gibi yapılandırabilirsiniz:

- Verileri alır
- Mesajın başlangıcını tanır
- Mesajın sonunu algılar

Seçilen arayüz için "Alınan mesajın konfigürasyonu" özelliklerinde bu parametreleri belirtirsiniz.



Bundan başka, kullanıcı programındaki mesaj parametrelerini Receiver_Config (Sayfa 914) yordamı ile dinamik olarak yapılandırabilir veya değiştirebilirsiniz.

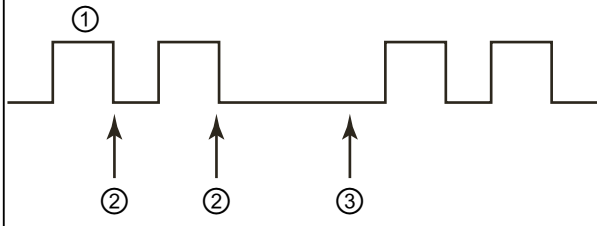
NOT

Kullanıcı programında Receive_Config ile girilen parametre değerleri, "Alındı mesajı konfigürasyonu" özelliklerinin üzerine yazılır. RCV_CFG yordam ile girilen parametreler, kapatma veya STOP durumuna geçiş sırasında CPU'da saklanmaz.

Mesaj başlangıcı için koşullar

İletişim arayüzünün bir mesajın başlangıcını nasıl tanıması gerektiğini belirleyebilirsiniz. Başlangıç karakterleri ve mesaj içeriği, yapılandırılmış bir bitiş koşulu karşılanana kadar alım tamponunda saklanır.

Birden çok başlangıç koşulu belirleyebilirsiniz. Birden çok başlangıç koşulu belirtirseniz, mesajın başlatılmış olarak kabul edilmesi için tüm başlangıç koşullarının karşılanmış olması gerekir. Örneğin bir hat boşta çalışma süresini ve belirli bir başlangıç karakterini yapılandırırsanız, CM veya CB önce hat boşta kalma süresi koşulunun karşılanmasını bekler ve bunun ardından belirli başlangıç karakterini arar. Başka bir karakter alınırsa (belirtilmiş olan başlangıç karakteri değil), CM veya CB, hat boşta çalışma süresini bekleyerek mesajın başlangıcı için aramayı yeniden başlatır.

Parametre	Tanım
Herhangi bir karakterle başla	"Herhangi bir karakter" koşulunun anlamı, herhangi bir karakterin başarılı bir şekilde alınması, bir mesaj başlangıcını belirtmesidir. Bu karakter, bir mesajdaki ilk karakterdir.
Hat duraklaması	"Hat duraklaması" koşulu, bir duraklama işareti karakteri alındıktan sonra mesaj alım işleminin başladığını gösterir.
Hat boşta	"Hat boşta" koşulu, mesaj alımının, alım hattının belirtilen Bit süresi kadar boşta çalıştığında başladığını belirtir. Bu durum oluştuktan sonra mesaj alımı başlar.  ① Karakter ② Hat boşta çalışma süresini yeniden başlatır ③ Hat boşta çalışması algılanır ve mesaj alımı başlar
Özel koşul: Mesajın başlangıcını tek karakterle algılama	Belirli bir karakterin bir mesajın başlangıcını gösterdiğini işaret eder. Bu karakter, bir mesajdaki ilk karakter olur. Belirli bir karakterden önce alınan herhangi bir karakter dikkate alınmaz. Standart karakter, STX'dir.
Özel koşul: Mesaj başlangıcının karakter dizisine göre algılama	En fazla dört yapılandırılmış diziden oluşan belirli bir karakter dizisinin bir mesajın başlangıcını gösterdiğini belirtir. Her karakter dizisi için en fazla beş karakter girebilirsiniz. Her karakter pozisyonu için, belirli bir onaltılık karakter veya karakter dizisindeki karakteri yok sayma (yer tutucu karakter) girilebilir. Bir karakter dizisinin son belirli karakteri, başlangıç koşulu dizisini sonlandırır. Gelen karakter dizileri, bir başlangıç koşulu karşılanana dek yapılandırılmış başlangıç koşullarıyla kıyaslanır. Başlangıç koşulu karşılanırsa, bitiş koşulunun denetlemesi başlar. En fazla dört spesifik karakter dizisi yapılandırabilirsiniz. Farklı karakter dizileri bir mesajın başlangıcını gösterebildiğinde çok dizili bir başlangıç koşulu kullanırsınız. Mesaj, ayarlanan karakter dizilerinden biri karşılandığında başlar.

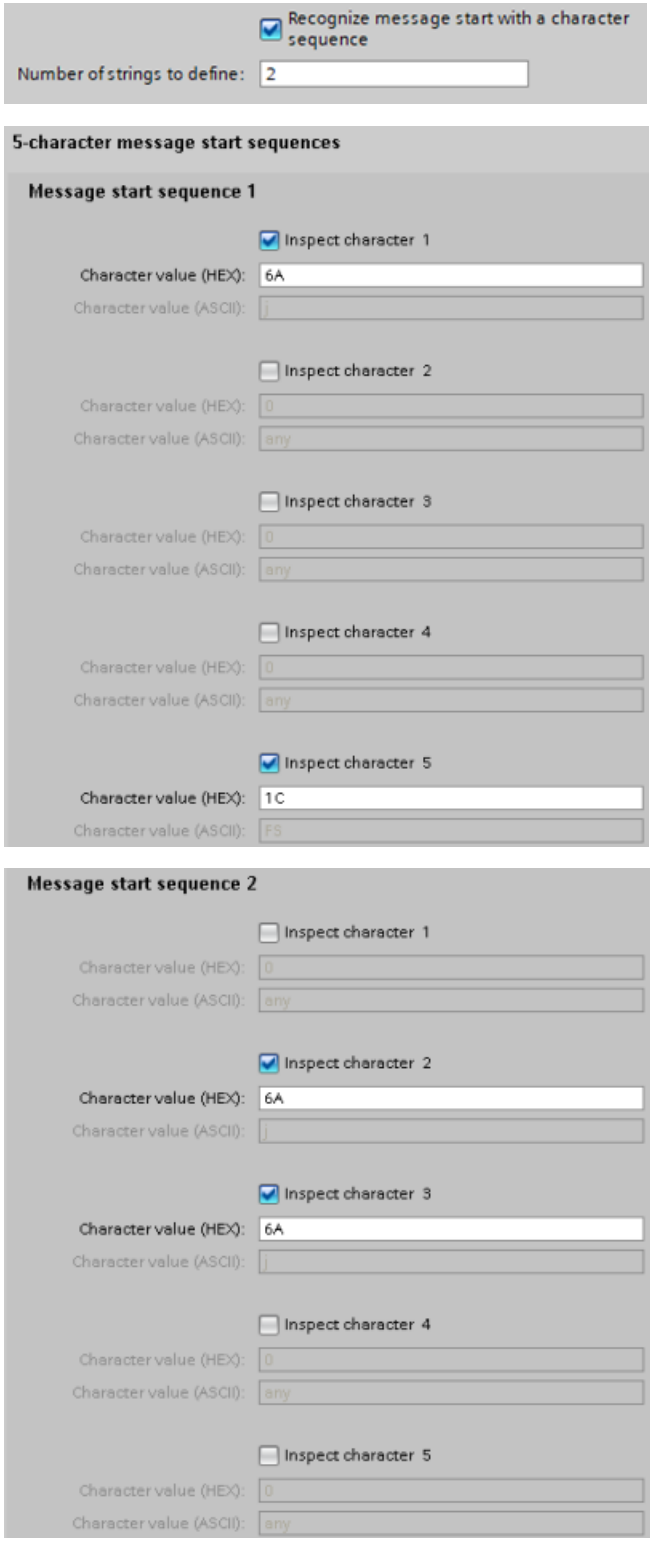
Başlangıç koşulları aşağıdaki sırayla denetlenir:

- Boş satır
- Satır kesmesi
- Karakterler veya karakter dizileri

Birden çok başlangıç koşulu denetlenirse ve bu koşullardan biri karşılanmazsa, CM veya CB denetlemesi ilk talep edilen koşulla yeniden başlatır. CM veya CB, ilgili başlangıç koşullarının karşılandığını belirledikten sonra bitiş koşullarını değerlendirmeye başlar.

Örnek konfigürasyon: İki karakter dizisinden birisiyle mesaj başlangıcı

Mesaj başlangıcı koşullarının aşağıdaki konfigürasyonunu dikkate alın:



Bu konfigürasyonla, aşağıdaki Bit örüntülerinden biri mevcutsa başlangıç koşulu karşılanmış olur:

- İlk karakterin 0x6A ve beşinci karakterin 0x1C olduğu 5 karakterlik bir karakter dizisinin alımı. 2, 3 ve 4 karakterleri herhangi bir değer olabilir. Beşinci karakteri aldıktan sonra, bitiş koşulu denetlemesi başlar.
- Diğer herhangi bir karakterden sonra, iki ardışık 0x6A karakterin alınması. Bu durumda, ikinci 0x6A dizisi (3 karakter) alındıktan sonra, bitiş koşulu denetlemesi başlar. İlk 0x6A'dan önceki karakter başlangıç koşuluna dahil edilir.

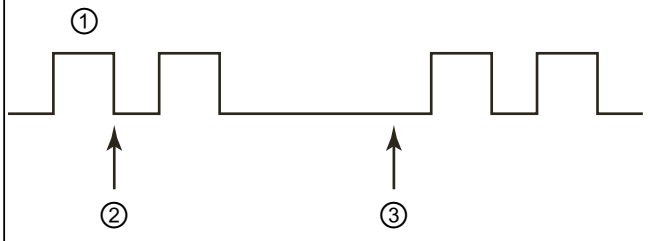
Aşağıda bu başlangıç koşulunu karşılayan karakter dizisi örnekleri verilmektedir:

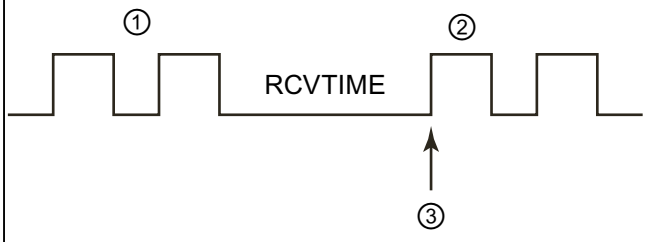
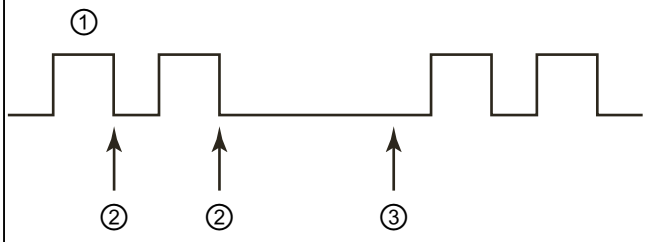
- <herhangi bir karakter> 6A 6A
- 6A 12 14 18 1C
- 6A 44 A5 D2 1C

Mesaj bitişi için koşullar

Bunun dışında, başka iletişim arayüzünün mesaj bitişini nasıl algılaması gerektiğini de tanımlarsınız. Mesajlar için birden çok bitiş koşulu ayarlayabilirsiniz. Mesaj, belirlenen koşullardan biri karşılanır karşılanmaz sona erer.

Örneğin 300 msn mesaj zaman aşımı, 40 Bit süresi karakter aralığı zaman aşımı ve maksimum 50 Bayt uzunluğunu bir bitiş koşulu olarak belirtebilirsiniz. Mesaj alımı 300 msn'den uzun sürdüğünde veya herhangi iki karakter arasındaki mesafe 40 Bit süresini aştığında veya 50 Bayt alındığında mesaj sona erer.

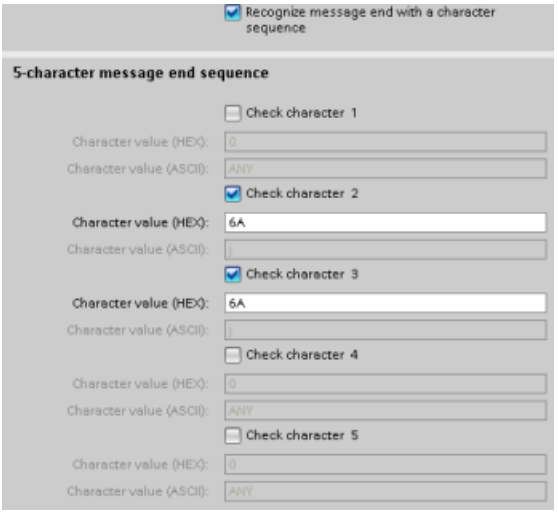
Parametre	Tanım
Mesaj bitişinin mesaj zaman aşımına göre algılanması	Mesaj bitişi için ayarlanan bekleme süresi geçtiğinde, mesajın sonuna ulaşılır. Mesaj zaman aşımı, bir başlangıç koşulu karşılandığında başlar. Ön ayar 200 msn, değer aralığı 0 ila 65535 msn'dir.  ① Alınan karakterler ②Başlama koşulu yerine getirildi: Mesaj zamanlayıcı çalışmaya başlar ③ Mesaj zamanlayıcının süresi dolar ve mesajı sonlandırır

Parametre	Tanım
Yanıt zaman aşımına göre mesaj bitişin algılanması	Geçerli bir başlangıç karakter dizisi alınmadan önce, bir yanıt için yapılandırılmış bekleme süresine ulaşıldığında, mesajın sonuna ulaşılır. Yanıt zaman aşımı, bir aktarım sona erdiğinde ve CM veya CB alım yordamını başlattığında başlar. Ön ayar, yanıt zaman aşımı süresi için 200 msn, değer aralığı 0 ila 65535 msn'dir. RCVTIME tarafından belirtilen yanıt süresi içinde bir karakter alınmazsa, ilgili RCV_PTP yordamına bir hata bildirilir. Yanıt zaman aşımı, spesifik bir bitiş koşulu tanımlamaz. Zaman aşımı, yalnızca bir karakterin belirtilen süre içinde başarıyla alınması gerektiğini gösterir. Bir mesajın fiili sonunu belirtmek için farklı bir bitiş koşulu yapılandırmanız gerekir.  ① Gönderilen karakterler ② Alınan karakterler ③ İlk karakter o ana dek başarıyla alınmış olmalıdır.
Mesaj bitişinin karakter aralığına göre algılanması	Bir mesajın herhangi iki ardışık karakteri arasında ayarlanan maksimum süre geçtiğinde, mesajın sonuna ulaşılır. Ön ayarlı karakter aralığı 12 Bit adımıdır ve maksimum sayı, maksimum sekiz saniyeye kadar 65535 Bit adımıdır.  ① Alınan karakterler ② Karakterler arasındaki zamanlayıcıyı yeniden başlatır ③ Karakterler arasındaki süre için olan zamanlayıcının süresi dolar ve mesajı sonlandırır.
Mesaj bitişinin sabit sayıda karakter alımıyla algılanması	Yapılandırılan sayıda karakter alındığında, mesajın sonuna ulaşılır. Sabit uzunluk için geçerli aralık, 1 ila 4096'dır. S7-1200'de mesaj bitişi için bu koşulun yalnızca V4.0 ve üzeri sürümdeki CPU'lar için geçerli olduğunu dikkate alın.
Mesaj bitişinin maksimum mesaj uzunluğuna göre algılanması	Yapılandırılan maksimum karakter sayısı alındığında, mesajın sonuna ulaşılır. Maksimum uzunluk için geçerli aralık, 1 ile 1024 arasındadır. Bu koşul, "mesaj tamponu taşması" hatasını önlemek için kullanılabilir. Eğer bu bitiş koşulu, zaman aşımı bitiş koşullarıyla birleştirilirse ve zaman aşımı bitiş koşulu oluşursa, maksimum uzunluğa ulaşılmamış olsa bile, alınan tüm geçerli karakterler yayınlanır. Böylece, yalnızca maksimum uzunluk bilinse bile farklı uzunluklardaki protokoller desteklenebilir.
Mesaj uzunluğunun mesajdan okunması	Mesaj uzunluğu, mesajın kendisinde belirtilir. Belirtilen uzunlukta bir mesaj alındığında, mesajın sonuna ulaşılır. Mesaj uzunluğunun nasıl belirleneceği ve değerlendirileceği aşağıda açıklanmaktadır.
Mesaj bitişinin bir karakter ile algılanması	Belirli bir karakter alındığında, mesajın sonuna ulaşılır.

Parametre	Tanım
Mesaj bitişinin bir karakter dizisi ile algılanması	Belirli bir karakter dizisi alındığında, mesajın sonuna ulaşılır. En fazla 5 karakterlik bir dizi belirtebilirsiniz. Her karakter pozisyonu için, belirli bir onaltılık karakter veya karakter dizisindeki karakteri yok sayma girilebilir. Yok sayılan baştaki karakterler bitiş koşuluna ait değildir. Yok sayılan sondaki karakterler bitiş koşuluna aittir.

Örnek konfigürasyon: Mesajı bir karakter dizisi ile bitirmek

Mesaj bitişi koşullarının aşağıdaki konfigürasyonunu dikkate alın:



Bu durumda, iki ardışık karakter 0x6A ve bunun ardından herhangi iki karakter alındığında bitiş koşulu karşılanmış olur. 0x6A 0x6A'dan önceki karakter, bitiş karakteri dizisine ait değildir. 0x6A 0x6A'dan sonra, bitiş karakteri dizisini tamamlamak için iki karakter gerekir. 4 ve 5 karakterlerinden alınan değerler önemsiz olup yalnızca bitiş koşulunun sağlanması için alınması gerekmektedir.

NOT

Karakter dizinizin mesaj sonunu belirtmesini istiyorsanız, onu son karakter pozisyonuna yerleştirin. Yukarıdaki örnekte, 0x6A 0x6A'nın, sondaki karakterler olmadan mesajı bitirmesi isteniyorsa, 0x6A 4 ve 5 karakter pozisyonlarında yapılandırılacaktır.

Mesaj uzunluğunun mesajda belirtilmesi

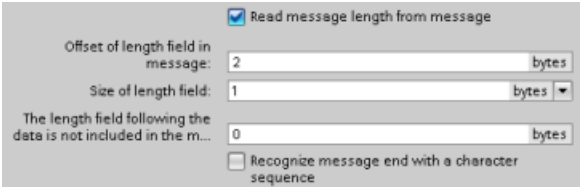
Mesaj uzunluğunun mesajın kendisinde yer aldığı özel koşul seçilirken, mesaj uzunluğu hakkında bilgi içeren üç parametrenin belirtilmesi gerekir.

Fiili mesaj yapısı, kullanılan protokole bağlıdır. Üç parametre şunlardır:

- n: Mesaj içindeki uzunluk belirtiminin başladığı karakter pozisyonu (taban 1)
- Uzunluk değeri: Uzunluk belirtiminin Bayt sayısı (bir, iki veya dört)
- Uzunluk m: Uzunluk belirtiminden sonra uzunluk sayımına dahil olmayan karakterlerin sayısı

Bitiş karakterinin ardışık olması gerekmez. "Uzunluk m" değeri, uzunluğu, uzunluk alanı dışındaki bir sağlama toplamı alanının uzunluğunu belirtmek için kullanılabilir.

Bu alanlar, cihaz özelliklerinde Alınan mesajın konfigürasyonu öğesinde görüntülenir:



Örnek 1: Aşağıdaki protokole göre yapılandırılmış bir mesaj söz konusudur:

STX	Len (n)	3 ila 14 arası karakter, uzunluğa göre sayılmış											
		ADR	PKE		INDEX		PWD		STW		HSW		BCC
1	2	3	4	5	6	7	8	9	10	11	12	13	14
STX	0x0C	xx	xxxx		xxxx		xxxx		xxxx		xxxx		xx

Mesaj alımı uzunluğu için parametreleri aşağıdaki gösterildiği gibi yapılandırın:

- n = 2 (mesaj uzunluğu Bayt 2'de başlıyor.)
- Uzunluk değeri = 1 (mesaj uzunluğu bir Bayt'ta bulunur.)
- Uzunluk m = 0 (uzunluk belirtiminden sonra uzunluk sayımına dahil olmayan başka karakterler gelmez. On iki karakter, uzunluk belirtimini izler.)

Bu örnekte, 3 ile 14 arasındaki karakterler Len (n) tarafından sayılmaktadır.

Örnek 2: Aşağıdaki protokole göre yapılandırılmış başka bir mesaj söz konusudur:

SD1	Len (n)	Len (n)	SD2	5 ila 10 arası karakter, uzunluğa göre sayılmış						FCS	ED
				DA	SA	FA	Veri birimi=3 Bayt				
1	2	3	4	5	6	7	8	9	10	11	12
xx	0x06	0x06	xx	xx	xx	xx	xx	xx	xx	xx	xx

Mesaj alımı uzunluğu için parametreleri aşağıdaki gösterildiği gibi yapılandırın:

- n = 3 (mesaj uzunluğu Bayt 3'te başlıyor.)
- Uzunluk değeri = 1 (mesaj uzunluğu bir Bayt'ta bulunur.)
- Uzunluk m = 3 (uzunluk belirtiminden sonra uzunluk sayımına dahil olmayan üç karakter gelir. Bu örnek protokolde, uzunluk sayımında SD2, FCS ve ED karakterleri dikkate alınmamaktadır. Diğer altı karakter, uzunluk sayımına dahil edilir; yani uzunluk belirtiminden sonraki toplam karakter sayısı dokuz olur.)

Bu örnekte, 5 ile 10 arasındaki karakterler Len (n) tarafından sayılmaktadır.

13.3.4 3964(R) iletişiminin yapılandırılması

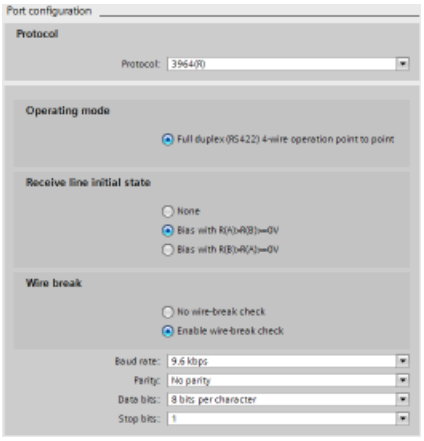
13.3.4.1 3964(R) iletişim portlarının yapılandırılması

3964(R) iletişimi için olan iletişim arayüzlerini yapılandırmada aşağıdaki prosedürler mevcuttur:

- STEP 7'de cihaz konfigürasyonundaki port parametrelerini yapılandırın. CPU, cihaz konfigürasyonundaki ayarları kaydeder ve yeniden başlatmanın ardından ayarları içe aktarır.
- Port_Config (Sayfa 910) yordamını kullanarak port parametrelerini ayarlayın. Yordamlar tarafından belirlenen port ayarları, CPU, RUN işletim türündeyken geçerlidir. Port ayarları, yeniden başlatmanın ardından cihaz konfigürasyonu ayarlarına geri döner.

İletişim arayüzlerini cihaz konfigürasyonuna ekledikten (Sayfa 132) sonra, modül taşıyıcınızdaki CM'lerden birini seçerek iletişim arayüzlerini parametrelendirirsiniz.





Seçilen CM'nin parametreleri, denetçi penceresindeki "Özellikler" sekmesinde görüntülenir. Aşağıdaki parametreleri düzenlemek için, "Port konfigürasyonu" öğesini seçin:

- Protokol: 3964(R)
- İşletim durumu (yalnızca CM 1241 (RS-422/485) modülü)
- Alım hattının çıkış durumu (yalnızca CM 1241 (RS-422/485) modülü)
- Tel kopması (yalnızca CM 1241 (RS-422/485) modülü)
- Baud hızı
- Parite
- Veri Bit'leri
- Stop Bit'leri

Parametre	Tanım
Protokol	3964R veya serbestçe programlanabilir. Portu 3964(R) iletişimi için yapılandırmak üzere 3964R'yi seçin.
İşletim durumu*	Tam dupleks (RS-422), dört telli işletim, noktadan noktaya. (aktif)
Alım hattı çıkış durumu*	Aşağıdaki opsiyonlardan birini etkinleştirin: • Yok • R(A)>R(B)>=0V ile ön gerilim • R(B)>R(A)>=0V ile ön gerilim
Tel kopması*	Aşağıdaki opsiyonlardan birini etkinleştirin: • Tel kopma kontrolü yok • Tel kopma kontrolünü etkinleştirin

* Yalnızca CM 1241 (RS-422/485) modülü

Parametre	Tanım
Baud hızı	Baud hızı için ön ayarı, 9,6 kBit/sn'dir. Geçerli değerler: 300 Baud, 600 Baud, 1,2 kBit/sn, 2,4 kBit/sn, 4,8 kBit/sn, 9,6 kBit/sn, 19,2 kBit/sn, 38,4 kBit/sn, 57,6 kBit/sn, 76,8 kBit/sn ve 115,2 kBit/sn.
Parite	Ön ayarlı parite, "yok" şeklindedir. Geçerli değerler: Parite yok, çift sayılı, tek sayılı, işaret (parite Bit'i her zaman 1'de), boşluk (parite Bit'i her zaman 0'da) ve herhangi bir parite (aktarım için parite Bit'ini 0'a ayarla; alım için parite hatasını göz ardı et).
Karakter başına veri Bit'i	Bir karakterdeki veri Bit'lerinin sayısı. 7 veya 8 ayarları geçerlidir.
Stop Bit'lerinin sayısı	Stop Bit'lerinin sayısı, bir veya iki olabilir. Ön ayar, birdir.

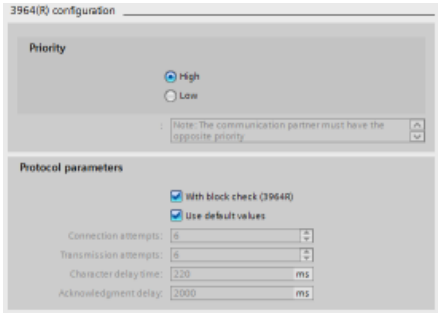
* Yalnızca CM 1241 (RS-422/485) modülü

13.3.4.2 3964(R) önceliğinin ve protokol parametrelerinin yapılandırılması

3964(R) iletişimi için olan iletişim arayüzlerini yapılandırmada aşağıdaki prosedürler mevcuttur:

- Önceliği ayarlamak ve protokol parametrelerini yapılandırmak için, iletişim arayüzünün cihaz konfigürasyonunda "Konfigürasyon, 3964(R)" üzerine tıklayın. CPU, cihaz konfigürasyonundaki ayarları kaydeder ve yeniden başlatmanın ardından ayarları içe aktarır.
- Önceliği ayarlamak ve protokol parametrelerini yapılandırmak için P3964_Config (Sayfa 919) yordamını kullanın. Yordamlarla belirtilen değerler, CPU RUN işletim türündeyken geçerlidir. Yeniden başlatmanın ardından, değerler cihaz konfigürasyonu ayarlarına geri döner.





Seçilen CM'nin parametreleri, denetçi penceresindeki "Özellikler" sekmesinde görüntülenir. Aşağıdaki parametreleri düzenlemek için, "Konfigürasyon, 3964(R)" öğesini seçin:

- Öncelik (yüksek veya düşük):
- Protokol parametreleri
  - Blok kontrolü ile (3964R)
  - Standart değerleri kullan
    Bağlantı denemeleri
    Aktarım denemeleri
    Karakter gecikme süresi
    Onaylama gecikmesi

Parametre	Tanım
Öncelik	Yüksek veya düşük: CM, yüksek veya düşük önceliğe sahiptir ve iletişim partnerinin tersi önceliğe sahip olması gerekir.
Blok kontrolü ile (3964)	Bu seçildiğinde, bir blok kontrol karakteri (BCC) dahil edilerek 3964(R) iletişimlerine aktarım güvenliği uygulanır. Bu opsiyon seçilmemişse, aktarım güvenliği bir blok kontrol karakteri içermez.

Parametre	Tanım
Standart değerleri kullan	Bu seçildiğinde, 3964(R) aşağıdaki protokol parametreleri için standart değerleri kullanır: • Bağlantı denemeleri • Aktarım denemeleri • Karakter gecikme süresi • Onaylama gecikmesi Opsiyon seçili değilse, bu parametrelerin her biri için değerleri yapılandırabilirsiniz.
Bağlantı denemeleri	Bağlantı denemeleri sayısı (standart değer: 6 bağlantı denemesi) 1 ila 255
Aktarım denemeleri	Aktarım denemeleri sayısı (standart değer: 6 bağlantı denemesi) 1 ila 255
Karakter gecikme süresi	Karakter gecikme süresi ayarı (belirlenen veri aktarım hızına bağlı olarak) (standart değer: 220 msn) 1 msn ila 65535 msn
Onaylama gecikmesi	Onaylama gecikme süresi ayarı (belirlenen veri aktarım hızına bağlı olarak) (standart değer: Blok kontrolü etkinleştirilmişse 2000 msn; 550 msn, blok kontrolü etkinleştirilmemişse) 1 msn - 65535 msn

NOT

Öncelik dışında, CM modülü ve iletişim partneri için protokol ayarları aynı olmalıdır.

13.3.5 Noktadan noktaya yordamlar

13.3.5.1 Noktadan noktaya yordamlar için ortak parametreler

Tablo 13-3 PtP yordamlarının genel giriş parametreleri

Parametre	Açıklama
REQ	PtP yordamlarının çoğu, yordamı yükselen bir kenarda (0'dan 1'e) başlatan bir REQ girişine sahiptir. Bir yordamın yürütülmesi sırasında REQ girişi 1 (DOĞRU) olmalıdır, ancak REQ girişi, istenildiği süre kadar DOĞRU kalabilir. Yordam, REQ = YANLIŞ girişi ile çağrılıncaya kadar başka bir yordamı tetiklemez, böylece yordam REQ girişinin durumunu sıfırlayabilir. Bu işlem, yordamın bir sonraki yordamı başlatmak için yükselen kenarı algılayabilmesi açısından gereklidir. Programınıza bir PtP yordamı eklediğinizde, STEP 7 sizden örnek DB'yi belirtmenizi talep eder. Her PtP yordam çağrısı için benzersiz bir DB kullanın. Böylece her yordamın REQ gibi girişleri düzgün bir şekilde işlemesi mümkün olur.
PORT	İletişim cihazının konfigürasyonu sırasında bir port adresi atanır. Konfigürasyondan sonra, parametrenin açılır listesinden standart port için sembolik bir ad seçilebilir. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sabit" sekmesinde atanır.
Bit sürelerinin çözünürlüğü	Yapılandırılan Baud hızındaki Bit sürelerinin sayısı, çeşitli parametreler için belirtilir. Parametre Bit sürelerinde belirtilerek, parametre Baud hızından bağımsızdır. Bit süresi birimlerindeki tüm parametreler maksimum 65535'e kadar belirtilebilir. Ancak, bir CM veya CB'nin ölçebileceği maksimum süre sekiz saniye ile sınırlıdır.

PtP yordamlarının DONE, NDR, ERROR ve STATUS çıkış parametreleri, PtP fonksiyonlarının yürütme durumunu gösterir.

Tablo 13-4 Çıkış parametreleri DONE, NDR, ERROR ve STATUS

Parametre	Veri tipi	Standart	Açıklama
DONE	Bool	YANLIŞ	Son talebin hata ile tamamlandığını belirtmek için, bir yürütme boyunca DOĞRU olarak ayarlanır; aksi takdirde YANLIŞ.
NDR	Bool	YANLIŞ	İstenen eylemin hatasız tamamlandığını ve yeni dosyaların alındığını belirtmek için, bir yürütme boyunca DOĞRU olarak ayarlanır; aksi takdirde YANLIŞ.
ERROR	Bool	YANLIŞ	Son talebin hatalarla tamamlandığını belirtmek için bir yürütme boyunca DOĞRU olarak ayarlanır, karşılık gelen hata kodu STATUS içindedir; aksi takdirde YANLIŞ.
STATUS	Word	0	Sonuç durumu: • DONE veya NDR Bit'i ayarlanırsa, STATUS, 0'a veya bir bilgi koduna ayarlanır. • Bit ERROR olarak ayarlanırsa, STATUS bir hata koduna getirilir. • Yukarıdaki Bit'lerin hiçbiri ayarlanmamışsa, yordam, fonksiyonun güncel durumunu açıklayan durum sonuçlarını geri döndürebilir. STATUS içindeki değer, fonksiyonun yürütülmesi sırasında tutulur.

NOT

DONE, NDR ve ERROR parametrelerinin yalnızca bir yürütme için ayarlandığını dikkate alın. Program lojiğiniz, geçici çıkış durum değerlerini veri tamponlarında saklamalıdır, böylece sonraki program çevrimlerinde durum değişikliklerini saptayabilirsiniz.

Tablo 13-5 Ortak koşul kodları

STATUS (W#16#....)	Açıklama
0000	Hata yok
7000	Fonksiyon kullanılmıyor
7001	Fonksiyon ilk çağrıda aktif.
7002	Fonksiyon, takip eden çağrılarda aktiftir (ilk çağrıdan sonraki sorgular).
8x3A	x parametresinde geçersiz işaretçi
8070	Dahili belleğin tamamı kullanıldı, aynı anda çok fazla yordam işleniyor
8080	Port numarasına izin verilmez.
8081	Zaman aşımı, modül hatası veya başka bir dahili hata
8082	Parametrelendirme arka planda çalıştığı için parametrelendirme başarısız oldu.
8083	Tampon taşması: CM veya CB, uzunluk parametresinin izin verdiğinden daha uzun bir alım mesajı geri döndürüldü.

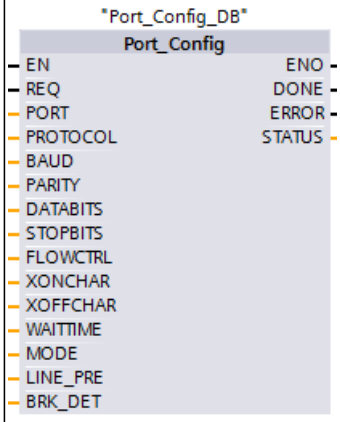
STATUS (W#16#....)	Açıklama
8090	Dahili hata: Yanlış mesaj uzunluğu, yanlış alt modül veya geçersiz mesaj Müşteri Destek birimi ile iletişime geçin.
8091	Dahili hata: Parametrelendirme mesajında yanlış sürüm Müşteri Destek birimi ile iletişime geçin.
8092	Dahili hata: Parametrelendirme mesajında yanlış veri seti uzunluğu Müşteri desteğiyle iletişime geçin.

Tablo 13-6 Ortak hata sınıfları

Sınıfın açıklaması	Hata sınıfları	Açıklama
Port konfigürasyonu	16#81Ax	Arayüz konfigürasyonundaki yaygın hataların açıklaması için
Gönderim konfigürasyonu	16#81Bx	Gönderim konfigürasyonundaki yaygın hataların açıklaması için
Alım konfigürasyonu	16#81Cx 16#82Cx	Alım konfigürasyonundaki yaygın hataların açıklaması için
Gönderim çalışma zamanı	16#81Dx	Gönderim sırasındaki yaygın çalışma zamanı hatalarının açıklaması için
Alım çalışma zamanı	16#81Ex	Alım sırasındaki yaygın çalışma zamanı hatalarının açıklaması için
Sinyal işleme:	16#81Fx	Sinyal işlemeye ilişkin yaygın hataların açıklaması için
İşaretçi hatası	16#8p01 ila 16#8p51	ANY işaretçisi hatası için, burada "p" yordamın parametre numarasıdır
Gömülü protokol hatası	16#848x 16#858x	Gömülü protokol hatası için

13.3.5.2 Port_Config (iletişim parametrelerini dinamik olarak yapılandırılması)

Tablo 13-7 Port_Config yordamı (port konfigürasyonu)

LAD/FBD	SCL	Açıklama
	**"Port_Config_DB"(REQ:=_bool_in_, PORT:=_word_in_, PROTOCOL:=_uint_in_, BAUD:=_uint_in_, PARITY:=_uint_in_, DATABITS:=_uint_in_, STOPBITS:=_uint_in_, FLOWCTRL:=_uint_in_, XONCHAR:=_char_in_, XOFFCHAR:=_char_in_, WAITTIME:=_uint_in_, MODE:=_uint_in_, LINE_PRE:=_uint_in_, BRK_DET:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	Port_Config ile Baud hızı gibi port parametrelerini programınız aracılığıyla değiştirebilirsiniz. Cihaz konfigürasyonu özelliklerinde portun statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. Bunun ardından Port_Config ile programınızdaki konfigürasyonu değiştirebilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

CPU, Port_Config yordamında ayarladığınız değerleri kalıcı olarak saklamaz. Cihaz konfigürasyonunda saklanan parametreler, RUN'dan STOP işletim durumuna değiştiğinde veya tekrar açıldığında CPU tarafından geri yüklenir. Daha fazla bilgi için, İletişim portlarının yapılandırılması (Sayfa 894) ve Akış kontrolünü yönetimi (Sayfa 896) bölümlerine bakın.

Tablo 13-8 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişin yükselen kenarında konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
PROTOCOL	IN	UInt	0 - Serbestçe programlanabilir iletişim (standart değer) 1. Protokol 3964(R)
BAUD	IN	UInt	Portun Baud hızı (standart değer: 6): 1 = 300 Baud, 2 = 600 Baud, 3 = 1200 Baud, 4 = 2400 Baud, 5 = 4800 Baud, 6 = 9600 Baud, 7 = 19200 Baud, 8 = 38400 Baud, 9 = 57600 Baud, 10 = 76800 Baud, 11 = 115200 Baud
PARITY	IN	UInt	Portun paritesi (standart değer: 1): 1 = Parite yok, 2 = Çift sayılı parite, 3 = Tek sayılı parite, 4 = Markör paritesi, 5 = Boşluk paritesi
DATABITS	IN	UInt	Karakter başına Bit (standart değer: 1): 1 = 8 veri Bit'i, 2 = 7 veri Bit'i
STOPBITS	IN	UInt	Stop Bit'leri (standart değer: 1): 1 = 1 Stop Bit'i, 2 = 2 Stop Bit'i
FLOWCTRL*	IN	UInt	Akış kontrolü (standart değer: 1): 1 = Akış kontrolü yok, 2 = XON/XOFF, 3 = Donanım RTS her zaman AÇIK, 4 = Donanım RTS anahtarlanmış
XONCHAR[1]	IN	Char	XON karakteri olarak çalışan karakteri belirtir. Bu, tipik olarak bir DC1 karakteridir (16#11). Bu parametre, yalnızca akış kontrolü etkinleştirildiğinde değerlendirilir. (Standart değer: 16#11)
XOFFCHAR[1]	IN	Char	XOFF karakteri olarak çalışan karakteri belirtir. Bu, tipik olarak bir DC3 karakteridir (16#13). Bu parametre, yalnızca akış kontrolü etkinleştirildiğinde değerlendirilir. (Standart değer: 16#13)
WAITTIME[1]	IN	UInt	Bir XON karakterini aldıktan sonra bir XOFF karakteri için ne kadar süre bekleneceğini veya RTS'yi etkinleştirdikten sonra CTS sinyali için ne kadar süre bekleneceğini belirtir (0 ila 65535 msn). Bu parametre, yalnızca akış kontrolü etkinleştirildiğinde değerlendirilir. (Standart değer: 2000)
MODE[2]	IN	UInt	Modülün seçilen işletim türünü gösterir.

[1] Protokol = 1 ise uygulanmaz (protokol 3964(R))

[2] Protokol = 1 (protokol 3964(R)) ise yalnızca işletim türü 0 ve 1 geçerlidir. Bu ise CM modülünüzün bir RS-232 modülü mü yoksa bir RS-422 modülü mü olduğuna bağlıdır.

Parametre ve veri tipi		Veri tipi	Açıklama
			• 0 = Tam dupleks (RS232) • 1 = Tam dupleks (RS-422), dört telli işletim (noktadan noktaya), gönderici her zaman aktif. • 2 = Tam dupleks (RS-422), dört telli işletim (çok noktalı Master), gönderici her zaman aktif. • 3 = Tam dupleks (RS-422), dört telli işletim (çok noktalı Slave), gönderici aktarım sırasında etkinleştirilir • 4 = Yarı dupleks (RS-485), iki telli işletim
LINE_PRE	IN	UInt	Aktif olmayan hat durumunu (boşta) gösterir. RS422 ve RS485 modülleri için, aktif olmayan hat durumu, R(A) ve R(B) sinyallerine ön gerilim uygulanarak oluşturulur. Şu ayarlar mümkündür: • 0 = Ön gerilim yok (ön ayar yok) (standart) • 1 = Ön gerilim R(A) > R(B) ≥ 0V; yalnızca RS-422 • 2 = Ön gerilim R(B) > R(A) ≥ 0V; RS-422 ve RS-485
BRK_DET	IN	UInt	İletişim hattı kesintisi algılamayı etkinleştirir/devre dışı bırakır. Hat kesintisi algılaması etkinleştirildiyse, iletişim kablosu modüle bağlı değilse, modül bir hata bildirir. RS422 noktadan noktaya işletimde, hat kopması algılaması yalnızca, alım hattının ön ayarı R(A) > R(B) ≥ 0 V olacak şekilde bir ön gerilim uygulanıyorsa mümkündür. • 0 = Hat kopması algılaması yok (standart) • 1 = Hat kopması etkinleştirilmiş
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

[1] Protokol = 1 ise uygulanmaz (protokol 3964(R))

[2] Protokol = 1 (protokol 3964(R)) ise yalnızca işletim türü 0 ve 1 geçerlidir. Bu ise CM modülünüzün bir RS-232 modülü mü yoksa bir RS-422 modülü mü olduğuna bağlıdır.

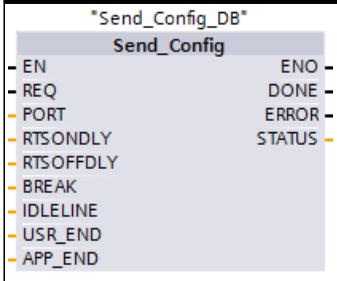
Tablo 13-9 Koşul kodları

STATUS (W#16#....)	Açıklama
81A0	Belirtilen protokol mevcut değil.
81A1	Belirtilen Baud hızı mevcut değil.
81A2	Belirtilen parite opsiyonu mevcut değil.
81A3	Belirtilen sayıda veri Bit'i mevcut değil.
81A4	Belirtilen sayıda Stop Bit'i mevcut değil.
80A5	Belirtilen akış kontrol tipi mevcut değil.
81A6	Bekleme süresi 0 ve akış kontrolü aktif
81A7	XON ve XOFF, geçersiz değerde (örn. aynı değer)
81A8	Blok başlığında hata (örn. yanlış blok tipi veya yanlış blok uzunluğu)
81A9	Bir yapılandırma işlemi devam ettiği için, yeniden yapılandırma reddedildi
81AA	Geçersiz RS422/RS485 işletimi
81AB	Hat kopması algılaması için geçersiz alım hattı ön ayarı
81AC	Geçersiz RS232 duraklatma işlemi

STATUS (W#16#....)	Açıklama
8280	Modülden okuma sırasında negatif onay
8281	Modüle yazma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil

13.3.5.3 Send_Config (seri iletişim için parametrelerin dinamik olarak yapılandırılması)

Tablo 13-10 Send_Config yordamı (gönderim konfigürasyonu)

LAD/FBD	SCL	Açıklama
	**"Send_Config_DB"(REQ:=_bool_in_, PORT:=_word_in_, RTSONDLY:=_uint_in_, RTSOFFDLY:=_uint_in_, BREAK:=_uint_in_, IDLELINE:=_uint_in_, USR_END:=_string_in_, APP_END:=_string_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	Send_Config, bir PtP iletişim arayüzü için seri aktarım parametrelerinin dinamik konfigürasyonuna olanak tanır. Bir CM veya CB'de mevcut mesajlar, Send_Config yürütüldüğünde silinir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Cihaz konfigürasyonu özelliklerinde portun statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. Bunun ardından Send_Config ile programınızdaki konfigürasyonu değiştirebilirsiniz.

CPU, Send_Config yordamında ayarladığınız değerleri kalıcı olarak saklamaz. Cihaz konfigürasyonunda saklanan parametreler, RUN'dan STOP işletim durumuna değiştiğinde veya tekrar açıldığında CPU tarafından geri yüklenir. Bkz. Gönderim parametrelerinin yapılandırılması (Sayfa 898)

Tablo 13-11 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişte yükselen kenardaki konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	Bir CM veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
RTSONDLY	IN	UInt	RTS'yi etkinleştirdikten sonra Tx verilerinin aktarımını gerçekleştirmeden önce beklenecek milisaniye sayısı. Bu parametre, yalnızca donanım akış kontrolü etkinleştirildiğinde geçerlidir. Geçerli aralık, 0 ila 65535 msn'dir. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 0)
RTSOFFDLY	IN	UInt	RTS'yi devre dışı bırakmadan önce, Tx verilerinin aktarımından sonra beklenecek milisaniye sayısı: Bu parametre, yalnızca donanım akış kontrolü etkinleştirildiğinde geçerlidir. Geçerli aralık, 0 ila 65535 msn'dir. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 0)

* CM ve CB 1241 için desteklenmez; parametre için boş bir karakter dizisi ("") kullanın.

Parametre ve veri tipi		Veri tipi	Açıklama
BREAK	IN	UInt	Bu parametre, Bit sürelerinin belirtilen sayısı için her mesajın başlangıcında bir duraklama gönderildiğini belirtir. En fazla 8 saniyeye kadar maksimum 65535 Bit süresi mümkündür. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 12)
IDLELINE	IN	UInt	Bu parametre, her mesajın başlangıcından önce belirtilen sayıda Bit süresi boyunca hattın boşta çalışacağını belirtir. En fazla 8 saniyeye kadar maksimum 65535 Bit süresi mümkündür. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 0)
USR_END*	IN	STRING[2]	Bu, sonlandırıcı ayırma karakterlerin sayısını ve türünü belirtir. Sonlandırıcı ayırma karakteri, gönderim tamponuna gömülüdür (yalnızca karakterler) ve gönderilen mesajın sonunu işaretler (karakterlerin aktarımı, sonlandırıcı ayırma karakterine kadar sürer). Sonlandırıcı ayırma karakteri, mesaj bitimine eklenir. • STRING[2,0,xx,yy] – Sonlandırıcı ayırma karakteri kullanılmıyor (standart) • STRING[2,1,xx,yy] – Sonlandırıcı ayırma karakteri, bir karakterden oluşuyor • STRING[2,2,xx,yy] – Sonlandırıcı ayırma karakteri, iki karakterden oluşuyor Ya USR_END ya da APP_END, sıfır uzunlukta olmalıdır.
APP_END*	IN	STRING[5]	Gönderilen mesaja eklenecek karakterlerin sayısını ve türünü belirtir (yalnızca karakterler eklenir). STRING[5,0,aa,bb,cc,dd,ee] – Bitiş karakteri kullanılmıyor (standart) • STRING[5,1,aa,bb,cc,dd,ee] – Bir bitiş karakteri gönder • STRING[5,2,aa,bb,cc,dd,ee] – İki bitiş karakteri gönder • STRING[5,3,aa,bb,cc,dd,ee]– Üç bitiş karakteri gönder • STRING[5,4,aa,bb,cc,dd,ee] – Dört bir bitiş karakteri gönder • STRING[5,5,aa,bb,cc,dd,ee] – Beş bitiş karakteri gönder
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

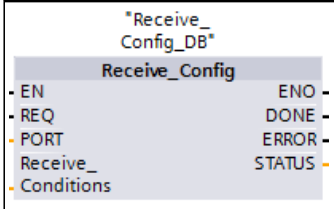
* CM ve CB 1241 için desteklenmez; parametre için boş bir karakter dizisi ("") kullanın.

Tablo 13-12 Koşul kodları

STATUS (W#16#....)	Açıklama
81B0	Bir gönderim alarmının konfigürasyonuna izin verilmez. Müşteri Destek birimi ile iletişime geçin.
81B1	Duraklama süresi, izin verilen maksimum değerden büyüktür.
81B2	Boşta çalışma süresi, izin verilen maksimum değerden büyüktür.
81B3	Blok başlığında hata, örn. yanlış blok tipi veya yanlış blok uzunluğu
81B4	Bir yapılandırma işlemi devam ettiği için, yeniden yapılandırma reddedildi
81B5	Belirtilen bitiş karakteri sayısı, ikiden fazla veya bitiş sayısı beşten fazla
81B6	Firmware'deki gömülü protokoller etkilendiğinden, gönderim konfigürasyonu reddedildi
8280	Modülden okuma sırasında negatif onay
8281	Modüle yazma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil

13.3.5.4 Receive_Config (seri alım parametrelerinin dinamik olarak yapılandırılması)

Tablo 13-13 Receive_Config yordamı (alım konfigürasyonu)

LAD/FBD	SCL	Açıklama
	**"Receive_Config_DB"(REQ:=_bool_in_, PORT:=_uint_in_, Receive_Conditions:=_struct_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	Receive_Config bir PtP iletişim arayüzü için seri alım parametrelerinin dinamik konfigürasyonunu gerçekleştirir. Bu yordam, alınan bir mesajın başlangıcını ve sonunu işaretleyen koşulları yapılandırır. Bir CM veya CB'de mevcut mesajlar, Receive_Config yürütüldüğünde silinir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Cihaz konfigürasyonu özelliklerinde iletişim bağlantısının statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. Bunun ardından Receive_Config ile programınızdaki konfigürasyonu değiştirebilirsiniz.

CPU, Receive_Config yordamında ayarladığınız değerleri kalıcı olarak saklamaz. Cihaz konfigürasyonunda saklanan parametreler, RUN'dan STOP işletim durumuna değiştiğinde veya tekrar açıldığında CPU tarafından geri yüklenir. Bu konuyla ilgili daha fazla bilgi için, bkz. "Alım parametrelerinin yapılandırılması (Sayfa 899)".

Tablo 13-14 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişte yükselen kenardaki konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
CONDITIONS	IN	CONDITIONS	CONDITIONS veri yapısı, aşağıda açıklandığı gibi mesajın başlangıç ve bitiş koşullarını belirtir.
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir çevrim boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatalı tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Receive_P2P yordamı için başlangıç koşulları

Receive_P2P yordamı, noktadan noktaya iletişim mesajlarının başlangıcını ve bitişini belirlemek için Receive_Config yordamı tarafından belirtilen konfigürasyonu kullanır. Mesajın başlangıcı, başlangıç koşulları tarafından belirlenir. Mesajın başlangıcı, bir veya daha fazla başlangıç koşuluna göre belirlenebilir. Birden çok başlangıç koşulu belirtilmişse, mesajın başlaması için tüm koşulların karşılanması gerekir.

Mesaj başlangıcı için koşullara ilişkin bir açıklamayı, "Alım parametrelerinin yapılandırılması (Sayfa 899)" altında bulabilirsiniz.

CONDITIONS parametresinin veri tipi yapısı, bölüm 1 (başlangıç koşulları)

Tablo 13-15 Başlangıç koşulları için CONDITIONS yapısı

Parametre ve veri tipi		Veri tipi	Açıklama
STARTCOND	IN	UInt	Başlangıç koşulunun belirtilmesi (standart değer: 1) • 01H - Başlangıç karakteri • 02H - Herhangi bir karakter • 04H - Hat duraklaması • 08H - Hat boşta • 10H - Karakter dizisi 1 • 20H - Karakter dizisi 2 • 40H - Karakter dizisi 3 • 80H - Karakter dizisi 4
IDLETIME	IN	UInt	Hat boşta çalışmanın zaman aşımı için gereken Bit süresi. (Standart değer: 40). Sadece "Hat boşta" durumuyla bağlantılı olarak. 0 ila 65535
STARTCHAR	IN	Byte	"Başlangıç karakteri" koşulu için başlangıç karakteri. (Standart değer: B#16#2)
STRSEQ1CTL	IN	Byte	Karakter dizisi 1, her karakter için kumandayı yok say/karşılaştır: (Standart değer: B#16#0) Bunlar, başlangıç karakter dizisindeki her karakter için etkinleştirme Bit'leridir. • 01H - Karakter 1 • 02H - Karakter 2 • 04H - Karakter 3 • 08H - Karakter 4 • 10H - Karakter 5 Bir Bit'in belirli bir karakter için devre dışı bırakılması, karakter dizisinin o pozisyonunda her bir karakterin bir eşleşme olduğu anlamına gelir.
STRSEQ1	IN	Char[5]	Karakter dizisi 1, başlangıç karakteri (5 karakter). Standart değer: 0
STRSEQ2CTL	IN	Byte	Karakter dizisi 2, her karakter için kumandayı yok say/karşılaştır. Standart değer: B#16#0)
STRSEQ2	IN	Char[5]	Karakter dizisi 2, başlangıç karakteri (5 karakter). Standart değer: 0
STRSEQ3CTL	IN	Byte	Karakter dizisi 3, her karakter için kumandayı yok say/karşılaştır. Standart değer: B#16#0
STRSEQ3	IN	Char[5]	Karakter dizisi 3, başlangıç karakteri (5 karakter). Standart değer: 0
STRSEQ4CTL	IN	Byte	Karakter dizisi 4, her karakter için kumandayı yok say/karşılaştır. Standart değer: B#16#0
STRSEQ4	IN	Char[5]	Karakter dizisi 4, başlangıç karakteri (5 karakter), standart değer: 0

Örnek

Aşağıda gösterilen alınmış onaltılık kodlanmış mesajı göz önünde bulundurun: "**68** 10 aa **68** bb 10 aa 16". Yapılandırılmış başlangıç karakter dizileri aşağıdaki tabloda gösterilmektedir. İlk karakter 68H başarıyla alındıktan sonra, başlangıç karakter dizileri değerlendirilir. Dördüncü karakterin (ikinci 68H) başarıyla alımından sonra, başlangıç koşulu 1 karşılanmış olur. Başlangıç koşulları karşılandığında, bitiş koşulları değerlendirilir.

Parite, çerçeveleme veya karakterler arasındaki zaman aralıkları gibi çeşitli hatalar nedeniyle başlangıç karakter dizisinin işlenmesi durabilir. Burada başlangıç koşulu karşılanmadığından, bu hatalar mesajın alınmamasına neden olur.

Tablo 13-16 Başlangıç koşulları

Başlangıç koşulu	İlk karakter	İlk karakter +1	İlk karakter +2	İlk karakter +3	İlk karakter +4
1	**68**H	xx	xx	**68**H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Receive_P2P yordamı için bitiş koşulları

Bir mesajın sonu, belirtilen bitiş koşulları tarafından belirlenir. Bir mesajın sonu, yapılandırılmış bir veya daha fazla bitiş koşulunun ilk kez ortaya çıkışıyla belirlenir. "Alım parametrelerinin yapılandırılması (Sayfa 899)" altında "Mesaj başlangıç koşulları" alt bölümü, Receive_Config yordamı için yapılandırabileceğiniz bitiş koşullarını açıklanmaktadır.

Bitiş koşullarını, cihaz konfigürasyonunda iletişim arayüzünün özelliklerinde veya Receive_Config yordamını kullanarak yapılandırabilirsiniz. CPU'nun STOP'tan RUN işletim durumuna her geçişinde, alım parametreleri (başlangıç ve bitiş koşulları) cihaz konfigürasyonu ayarlarına sıfırlanır. STEP 7 kullanıcı programı Receive_Config'i yürüttüğünde, ayarlar Receive_Config'in koşullarına getirilir.

CONDITIONS parametresinin veri tipi yapısı, bölüm 2 (bitiş koşulları)

Tablo 13-17 Bitiş koşulları için CONDITIONS yapısı

Parametre	Parametre tipi	Veri tipi	Açıklama
ENDCOND	IN	UInt 0	Bu parametre, mesaj bitişi koşulunu belirtir: • 01H - Yanıt süresi • 02H - Mesaj süresi • 04H - Karakter aralığı • 08H - Maksimum uzunluk • 10H - N + LEN + M • 20H - Karakter dizisi
MAXLEN	IN	UInt 1	Maksimum mesaj uzunluğu: Yalnızca "Maksimum uzunluk" bitiş koşulu seçildiğinde, kullanılır. 1 ila 1024 Bayt
N	IN	UInt 0	Mesajdaki uzunluk alanının Bayt pozisyonu. Yalnızca N + LEN + M bitiş koşulunda kullanılır. 1 ila 1022 Bayt
LENGTHSIZE	IN	UInt 0	Bayt alanının boyutu (1, 2 veya 4 Bayt). Yalnızca N + LEN + M bitiş koşulunda kullanılır.

Parametre	Parametre tipi	Veri tipi	Açıklama
LENGTHM	IN	UInt 0	Uzunluk alanından sonra, uzunluk alanı değerinin içermediği karakter sayısını belirtin. Bu belirtim, yalnızca N + LEN + M bitiş koşuluyla kullanılır. 0 ila 255 Bayt
RCVTIME	IN	UInt 200	Alınan ilk karakter için ne kadar bekleneceğini belirtirler. Belirtilen süre içinde bir karakter başarıyla alınmadığı takdirde, alım yordamı bir hatayla sonlandırılır. Bu belirtim, yalnızca "Yanıt süresi" koşulunda kullanılır. (0 - 65535 Bit süresi, maksimum 8 saniye) Değerlendirme bir yanıtın ilk karakteri alındığında sona erdiğinden, bu parametre bir bitiş koşulu değildir. Bu yalnızca, bir yanıt beklendiği halde herhangi bir yanıt alınmadığı için alım yordamının sonlandırılması anlamında bir bitiş koşuludur. Burada ayrı bir bitiş koşulu tanımlamanız gerekir.
MSGTIME	IN	UInt 200	Tüm mesajın eksiksiz olarak alınması için, ilk karakteri aldıktan sonra ne kadar süre bekleneceğini belirtin. Bu parametre, sadece "Mesajın zaman aşımı" koşulu seçildiğinde kullanılır. (0 ila 65535 msn)
CHARGAP	IN	UInt 12	Karakterler arasındaki Bit sürelerinin sayısını belirtin. Karakterler arasındaki Bit süresi sayısı belirtilen değeri aşarsa, bitiş koşulu karşılanmış olur. Bu belirtim, yalnızca "Karakter aralığı" koşulunda kullanılır. (0 - 65535 Bit süresi, maksimum 8 saniye)
ENDSEQ1CTL	IN	Byte B#16#0	Karakter dizisi 1, her karakter için kumandayı yok say/karşılaştır: Bunlar, bitiş karakteri dizisindeki her karakter için etkinleştirme Bit'leridir. Karakter 1, Bit 0'dır, karakter 2, Bit 1'dir, ..., karakter 5, Bit 4'tür. Bir Bit'in belirli bir karakter için devre dışı bırakılması, karakter dizisinin o pozisyonunda her bir karakterin bir eşleşme olduğu anlamına gelir.
ENDSEQ1	IN	Char[5] 0	Karakter dizisi 1, başlangıç karakteri (5 karakter).

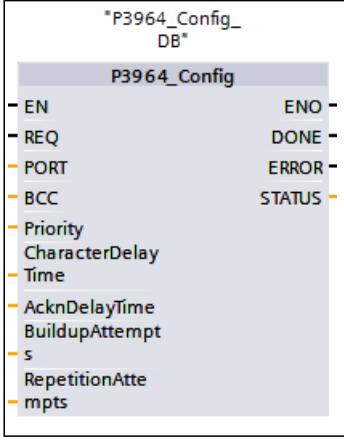
Tablo 13-18 Koşul kodları

STATUS (W#16#....)	Açıklama
81C0	Geçersiz başlangıç koşulu seçildi
81C1	Geçersiz bitiş koşulu seçildi, bitiş koşulu seçilmedi
81C2	Alım alarmı etkinleştirildi ve bu mümkün değil.
81C3	"Maksimum uzunluk" bitiş koşulu etkinleştirildi ve maksimum uzunluk 0 veya > 1024.
81C4	Hesaplanan uzunluk etkinleştirildi ve N, >= 1023'tür.
81C5	Hesaplanan uzunluk etkinleştirildi ve uzunluk 1, 2 veya 4 değil.
81C6	Hesaplanan uzunluk etkinleştirildi ve M değeri > 255'tir.
81C7	Hesaplanan uzunluk etkinleştirildi ve hesaplanan uzunluk > 1024'tür.
81C8	Yanıt zaman aşımı etkinleştirildi ve yanıt zaman aşımı sıfırdır.
81C9	Karakter aralığı için zaman aşımı etkinleştirildi ve zaman aşımı sıfırdır.
81CA	Hat boşta için zaman aşımı etkinleştirildi ve zaman aşımı sıfırdır.
81CB	Bitiş karakteri dizisi etkinleştirildi, ancak tüm karakterler "ilgili değil".

STATUS (W#16#....)	Açıklama
81CC	Başlangıç karakter dizisi (4'ten biri) etkinleştirildi, ancak tüm karakterler "ilgili değil".
81CD	Üzerine yazma koruması hatası, alındı mesajı geçersiz
81CE	STOP'tan RUN işletim durumuna geçişte, alındı mesajları için tampon işleme hatası
81CF	Blok başlığında hata, örn. yanlış blok tipi veya yanlış blok uzunluğu
8281	Modüle yazma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil
82C0	Bir yapılandırma işlemi devam ettiği için, yeniden yapılandırma reddedildi
82C1	Modülün tampona alabileceği mesaj sayısı için belirtilen değer, izin verilen maksimum değerden fazla.
82C2	Firmware'de gömülü protokoller etkilendiğinde alım konfigürasyonu reddedildi.
8351	Bu varyant işaretçi için veri tipine izin verilmiyor

13.3.5.5 P3964_Config (3964(R) protokolü yapılandırması)

Tablo 13-19 P3964_Config yordamı (3964_Config (3964(R) protokolü yapılandırması)

LAD/FBD	SCL	Açıklama
	**"P3964_Config_DB"(REQ:=_bool_in_, PORT:=_uint_in_, BCC:=_usint_in, Priority:= _usint_in, CharacterDelayTime:=_uint_in, AcknDelayTime:= _uint_in, BuildupAttempts:=_usint_in_, RepetitionAttempts:=_usint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	P3964_Config ile çalışma zamanı sırasında öncelik ve protokol parametrelerini değiştirebilirsiniz. Cihaz konfigürasyonu özelliklerinde portun statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. P3964_Config yordamı ile programınızdaki konfigürasyonu değiştirebilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

CPU, P3964_Config yordamında ayarladığınız değerleri kalıcı olarak saklamaz. Cihaz konfigürasyonunda saklanan parametreler, tekrar açıldığında CPU tarafından geri yüklenir. Daha fazla bilgi için bkz. 3964(R) için iletişim önceliklerinin yapılandırılması ve protokol parametreleri (Sayfa 907).

Tablo 13-20 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişin yükselen kenarında konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	UInt	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)

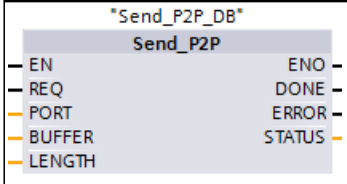
Parametre ve veri tipi		Veri tipi	Açıklama
BCC	IN	USInt	Blok kontrolünün kullanımını etkinleştirir/devre dışı bırakır • 0 = blok kontrolü olmadan • 1 = blok kontrolü ile
Priority	IN	UInt	Öncelik seçimi • 0 = düşük öncelik • 1 = yüksek öncelik CM'nin önceliği, iletişim partnerinin önceliğinin tersi olmalıdır.
CharacterDelayTi-me	IN	UInt	Karakter gecikme süresi ayarı (belirlenen veri aktarım hızına bağlı olarak) (standart değer: 220 msn) 1 msn ila 65535 msn
AcknDelayTime	IN	UInt	Onaylama gecikme süresi ayarı (belirlenen veri aktarım hızına bağlı olarak) (standart değer: 2000 msn) 1 msn ila 65535 msn
BuildupAttempts	IN	UInt	Bağlantı denemeleri sayısı (standart değer: 6 bağlantı denemesi) 1 ila 255
RepetitionAt-tempts	IN	UInt	Aktarım denemeleri sayısı (standart değer: 6 bağlantı denemesi) 1 ila 255
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Tablo 13-21 Koşul kodları

STATUS (W#16#....)	Açıklama
16#8380	Parametrelendirme hatası: "Karakter gecikme süresi" için geçersiz değer.
16#8381	Parametrelendirme hatası: "Yanıt zaman aşımı" için geçersiz değer.
16#8382	Parametrelendirme hatası: "Öncelik" için geçersiz değer.
16#8383	Parametrelendirme hatası: "Blok kontrolü" için geçersiz değer.
16#8384	Parametrelendirme hatası: "Bağlantı denemesi" için geçersiz değer.
16#8385	Parametrelendirme hatası: "Aktarım denemeleri" için geçersiz değer.
16#8386	Çalışma zamanı hatası: Bağlantı denemeleri sayısı aşıldı
16#8387	Çalışma zamanı hatası: Aktarım denemeleri sayısı aşıldı
16#8388	Çalışma zamanı hatası: "Blok kontrol karakterinde" hata Blok kontrol karakterinin dahili olarak hesaplanan değeri, bağlantının sonunda partnerden alınan blok kontrol karakteriyle eşleşmiyor.
16#8389	Çalışma zamanı hatası: Serbest alım tamponu beklenirken geçersiz karakter alındı
16#838A	Çalışma zamanı hatası: Alımda mantıksal hata. DLE'nin alımından sonra, diğer bir rastgele karakter (DLE veya ETX değil) alındı.
16#838B	Çalışma zamanı hatası: Karakter gecikme süresi aşıldı
16#838C	Çalışma zamanı hatası: Serbest alım tamponu için bekleme süresi başladı
16#838D	Çalışma zamanı hatası: NAK'dan sonra 4 sn içinde telegram tekrarı başlamıyor
16#838E	Çalışma zamanı hatası: Boşta çalışmada en az bir karakter alındı (NAK veya STX değil).
16#838F	Çalışma zamanı hatası: Başlatma çakışması - Her iki partner de yüksek önceliğe sahip
16#8391	Parametrelendirme hatası: 3964 - Serbestçe programlanabilen iletişim ayarlandığı için konfigürasyon verileri reddedildi

13.3.5.6 Send_P2P (gönderim tamponu verilerinin aktarılması)

Tablo 13-22 Send_P2P yordamı (noktadan noktaya verilerin gönderimi)

LAD/FBD	SCL	Açıklama
	**"Send_P2P_DB"(REQ:=_bool_in_, PORT:=_word_in_, BUFFER:=_variant_in_, LENGTH:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	Send_P2P veri aktarımını başlatır ve atanmış olan tamponu iletişim arayüzüne aktarır. CM veya CB atanmış olan Baud hızında verileri gönderirken, CPU'nun programı çalışmaya devam eder. Bir seferde yalnızca bir gönder yordamının beklemede olmasına izin verilir. CM veya CB bir mesaj gönderirken, ikinci bir Send_P2P yordamı yürütülürse CM veya CB bir hata verir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

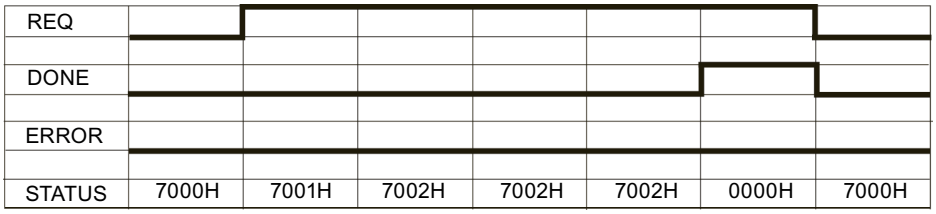
Tablo 13-23 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu serbest bırakma girişinin yükselen kenarında talep edilen aktarımın serbest bırakılması. Böylece tamponun içeriği noktadan noktaya iletişim arayüzüne aktarılır. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
BUFFER	IN	Varyant	Bu parametre, gönderim tamponunun başlangıç adresini gösterir. (Standart değer: 0) **Not:** Boole verileri veya Boole alanları desteklenmez.
LENGTH	IN	UInt	Bayt cinsinden iletilen çerçeve uzunluğu (standart değer: 0) Karmaşık bir yapı gönderiminde her zaman uzunluk 0'ı kullanın. Uzunluk 0 ise, yordam tüm çerçeveyi iletir.
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir çevrim boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatalı tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

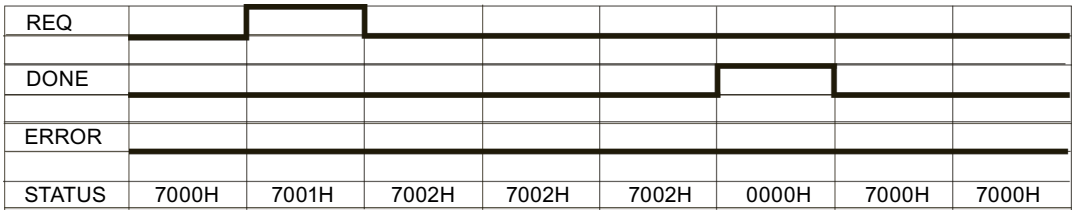
Bir gönder yordamı devam ederken, DONE ve ERROR çıkışları YANLIŞ durumundadır. Bir gönder yordamının sona ermesinden sonra, gönder yordamı durumunu bildirmek için DONE veya ERROR çıkışlarından biri DOĞRU olarak ayarlanır. DONE veya ERROR, DOĞRU durumdayken STATUS çıkışı geçerlidir.

İletişim arayüzü gönderilen verileri kabul ederse, bu yordam, 16#7001 durumunu yayınlar. Takip eden Send_P2P yürütmeleri, CM veya CB hâlâ gönderiyorsa 16#7002 değerini geri döndürür. Gönder yordamının sona ermesinden sonra CM veya CB, gönder yordamı için 16#0000 durumunu yayınlar (herhangi bir hata oluşmadıysa). REQ = 0 içeren Send_P2P takip eden yürütmeleri 16#7000 (serbest) durumunu geri döndürür.

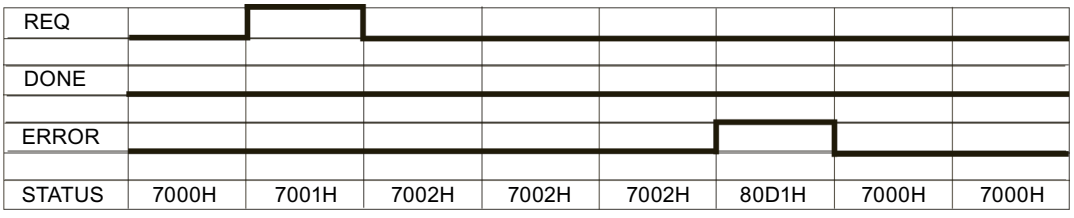
Aşağıdaki diyagramda, çıkış değerleri ile REQ arasındaki ilişki gösterilmektedir. Burada, gönderim işleminin durumunu kontrol etmek için yordamın düzenli olarak çağrıldığını varsayılmaktadır. Aşağıdaki şekilde ise, yordamın her çevrimde çağrıldığını varsayılmaktadır (DURUM değerleri ile temsil ediliyor).



Aşağıdaki şekilde, gönder yordamını başlatmak için REQ hattı darbeli (bir çevrim için) olduğunda DONE ve STATUS parametrelerinin nasıl yalnızca bir çevrim için geçerli olduğu gösterilmektedir.



Aşağıdaki şekilde, bir hata durumunda DONE, ERROR ve STATUS parametrelerinin ilişkisini gösterilmektedir.



DONE, ERROR ve STATUS değerleri, yalnızca Send_P2P aynı örnek DB ile tekrar yürütülene kadar geçerli kalır.

Tablo 13-24 Koşul kodları

STATUS (W#16#....)	Açıklama
81D0	Aktif göndericide yeni talep
81D1	Bekleme süresi içinde CTS olmadığı için gönderim iptal edildi
81D2	DCE cihazından DSR gelmediği için gönderim iptal edildi
81D3	Bekleme kuyruğu taşması nedeniyle gönderim iptal edildi (1024 Bayt'tan fazla gönderim)
81D5	Ters ön gerilim sinyali (tel kopması)
81D6	Gönderim tamponunda sonlandırıcı ayırma karakteri bulunamadığından, gönderim talebi red-dedildi.
81D7	FB ve CM arasında dahili hata / senkronizasyon hatası
81D8	Bağlantı yapılandırılmadığı için, gönderim denemesi reddedildi.

STATUS (W#16#....)	Açıklama
81DF	Aşağıdaki nedenlerden biri dolayısıyla, CM, FB'ye olan arayüzü sıfırladı • Modül yeniden başlatıldı (kapatma/açma) • CPU, bir tutma noktasına ulaştı • Modül yeniden parametrelendirildi Modül, her durumda ilgili kodu durum parametresinde görüntüler. Modül, SEND_P2P için alınan ilk veri setinden sonra durumu ve hatayı sıfırlar.
8281	Modüle yazma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil
8301	Bir ANY işaretçisinde geçersiz sözdizimi ID'si
8322	Bir parametre okurken aralık uzunluğu hatası
8324	Bir parametre okurken aralık hatası
8328	Bir parametre okunurken hizalama hatası
8332	Parametre, izin verilen maksimum numaradan daha büyük bir DB numarası içeriyor (DB numarası hatası).
833A	BUFFER parametresi için DB mevcut değil.

NOT

Profibus iletişimi için veri setinin maksimum uzunluğunun ayarlanması

CM1243-5 iletişim modülü, bir RS232, RS422 veya RS485 noktadan noktaya modül kullanan bir ET200SP veya ET200MP Profibus cihazına kumanda etmek için Profibus Master olarak kullanılıyorsa, bu durumda "max_record_len" veri bloğu değişkenini aşağıdaki gibi belirtik olarak 240 değerine ayarlamalısınız:

Örn. Port_Config, Send_Config veya Receive_Config gibi bir iletişim yordamını yürüttükten sonra, örnek DB'deki ("Send_P2P_DB".max_record_len gibi) max_record_len parametresini 240'a getirin.

max_record_len değerinin belirtik ataması yalnızca Profibus bağlantıları için gereklidir. Profinet bağlantılarında ise max_record_len için geçerli bir değer zaten kullanılmaktadır.

LENGTH ve BUFFER parametreleri arasındaki etkileşim

SEND_P2P yordamı tarafından gönderilebilecek minimum veri boyutu bir Bayttır. BUFFER parametresi, gönderilecek verilerin uzunluğunu belirler. BUFFER parametresi için Bool veri tipini veya Bool tipindeki Array'leri kullanamazsınız.

LENGTH parametresini istediğiniz zaman 0 olarak ayarlayabilir, böylece SEND_P2P'nin BUFFER parametresinin tüm veri yapısını göndermesini sağlayabilirsiniz. BUFFER parametresinin veri yapısının yalnızca bir kısmını göndermek istiyorsanız, LENGTH'i aşağıdaki gibi ayarlayabilirsiniz:

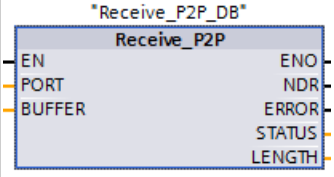
Tablo 13-25 LENGTH ve BUFFER parametreleri

LENGTH	BUFFER	Açıklama
= 0	Kullanılmıyor	Eksiksiz veriler, BUFFER parametresi tarafından tanımlandığı şekilde gönderilir. LENGTH = 0 ise aktarılan Bayt sayısını belirtmeniz gerekmez.
> 0	Yalın veri tipi	LENGTH değeri, bu veri tipinin Bayt sayısını içermelidir. Örnek: Bir Word değerinde, LENGTH iki olmalıdır. Bir Dword veya Real değerinde, LENGTH dört olmalıdır. Aksi takdirde aktarım yürütülmez ve 8088H hatası yayınlanır.

LENGTH	BUFFER	Açıklama
> 0	Yapı	LENGTH değeri, yapının tam Bayt uzunluğundan daha az bir Bayt sayısı içerebilir. Bu durumda yordam, yapının yalnızca ilk n Bayt'ını BUFFER,'dan gönderir, burada n = LENGTH'tir. Bir yapının dahili Bayt sırası her zaman belirlenemediğinden, bu, beklenmeyen sonuçlara neden olabilir. Dolayısıyla tam yapıyı göndermek için bir LENGTH = 0 kullanın.
	Array	LENGTH değeri, Array'in tam Bayt uzunluğundan küçük veya ona eşit bir Bayt sayısı içermelidir, burada veri elemanının Bayt sayısının birkaç katı olması gerekir. Örnek: Word tipinde bir Array'in LENGTH parametresi, ikinin katı ve Real tipinde bir Array'in dört katı olmalıdır. LENGTH belirtilmişse, yordam, Bayt cinsinden LENGTH değerine eşit Array elemanlarının sayısını aktarır. Örneğin BUFFER, 15 Dword elemanlı bir Array içeriyorsa (toplam 60 Bayt) ve siz LENGTH = 20 olarak belirttiyseniz, Array'deki ilk beş Dword elemanı aktarılır. LENGTH değeri, veri elemanlarının Bayt sayısının katı olmalıdır. Aksi takdirde STATUS = 8088H, ERROR = 1 olur ve aktarım gerçekleşmez.
	String	LENGTH parametresi gönderilecek karakter sayısını içerir. Yalnızca String karakterleri aktarılır. String'in maksimum ve gerçek uzunluğuna sahip Bayt'lar gönderilmez.

13.3.5.7 Receive_P2P (mesaj alımı etkinleştirme)

Tablo 13-26 Receive_P2P yordamı (noktadan noktaya verilerin alımı)

LAD/FBD	SCL	Açıklama
	**"Receive_P2P_DB"(PORT:=_word_in_, BUFFER:=_variant_in_, NDR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, LENGTH=>_uint_out_);**	Receive_P2P, CM veya CB'de alınan mesajları denetler. Bir mesaj mevcutsa, CM veya CB'den CPU'ya aktarılır. Bir hata, uygun STATU değerini geri döndürür.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-27 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
BUFFER	IN	Varyant	Bu parametre, alım tamponunun başlangıç adresini gösterir. Bu tampon, maksimum mesaj uzunluğunu alacak kadar büyük olmalıdır. Boole verileri veya Boole alanları desteklenmez. (Standart değer: 0)
NDR	OUT	Bool	Yeni veriler hazırsa ve yordam hatasız tamamlandıysa, bir yürütme boyunca DOĞRU.
ERROR	OUT	Bool	Yordam bir hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)
LENGTH	OUT	UInt	Çıkış mesajının Bayt cinsinden uzunluğu (standart değer: 0)

NDR veya ERROR DOĞRU durumundayken, STATUS değeri geçerlidir. STATUS değeri, CM veya CB'deki alım işleminin sonlandırılmasının nedenini getirir. Bu, tipik olarak alım işleminin başarılı olduğunu ve alım işleminin normal şekilde sona erdiğini gösteren pozitif bir değerdir.

STATUS değeri negatifse (onaltılık değerin en yüksek değerli Bit'i ayarlanmışsa), parite, çerçeveleme veya taşma hatası gibi bir hata durumu nedeniyle alım işlemi sonlandırılmıştır.

Her bir PtP iletişim arayüzü, maksimum 1024 Bayt'ı tampona alabilir. Bu, bir büyük mesaj veya birden çok sayıda küçük mesaj olabilir. CM veya CB'de birden çok mesaj mevcutsa, Receive_P2P yordamı mevcut en eski mesajı yayınlar. Bunun ardından yürütülen bir Receive_P2P yordamı en eski ikinci mesajı yayınlar.

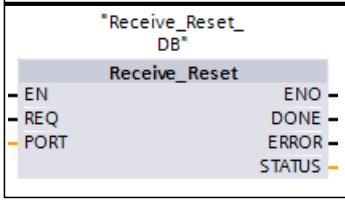
Tablo 13-28 Koşul kodları

STATUS (W#16#...)	Açıklama
0000	Tampon yok
0094	Maksimum karakter uzunluğu alındığı için, mesaj sonlandırıldı
0095	Mesaj zaman aşımı nedeniyle, mesaj sonlandırıldı
0096	Karakter aralığı zaman aşımı nedeniyle mesaj sonlandırıldı
0097	Yanıt zaman aşımı nedeniyle mesaj sonlandırıldı
0098	"N+LEN+M" uzunluk koşulu karşılandığı için, mesaj sonlandırıldı
0099	Bitiş karakteri dizisi karşılandığı için, mesaj sonlandırıldı
8085	LENGTH parametresinin değeri 0 veya 1 kB'den büyük.
8088	LENGTH parametresi veya alınan uzunluk, BUFFER'da belirtilen aralıktan daha büyük veya alınan uzunluk, BUFFER'da belirtilen aralıktan daha büyük.
8090	Hatalı konfigürasyon mesajı, yanlış mesaj uzunluğu, yanlış alt modül, geçersiz mesaj
81E0	Alım tamponu dolu olduğundan, mesaj sonlandırıldı
81E1	Parite hatası nedeniyle, mesaj sonlandırıldı
81E2	Çerçeveleme hatası nedeniyle mesaj sonlandırıldı
81E3	Taşma hatası nedeniyle, mesaj sonlandırıldı
81E4	Hesaplanan uzunluk tampon boyutunu aştığından, mesaj sonlandırıldı
81E5	Ters ön gerilim sinyali (tel kopması)
81E6	Mesaj bekleme kuyruğu dolu. Bu hata, veriler olmadan bildirilir. Hata oluşursa, modül, hatasız veri gönderimi ile hata arasında gidip gelir.
81E7	Dahili hata, yordam ile CM arasında senkronizasyon hatası: Bir sekans hatası olduğunda ayarlanır
81E8	Mesaj sona erdi, mesajın sonu algılanmadan önce karakterler arasındaki zaman aşımına ulaşıldı
81E9	Modbus CRC hatası algılandı (yalnızca Modbus protokolü için CRC oluşturmayı/denetlemeyi destekleyen modüller için)
81EA	Modbus telegramı çok kısa (yalnızca Modbus protokolü için CRC oluşturmayı/denetlemeyi destekleyen modüller için)
81EB	Maksimum mesaj boyutu aşıldığı için, mesaj iptal edildi
8201	Bir ANY işaretçisinde geçersiz sözdizimi ID'si
8223	Bir parametre yazarken aralık uzunluğu hatası. Parametre tamamen veya kısmen bir adres aralığının dışında veya bir Bit aralığının uzunluğu ANY işaretçisi için 8'in katı değil.
8225	Bir parametre yazarken aralık hatası Parametre, sistem fonksiyonu için izin verilmeyen bir aralıktadır.
8229	Bir parametre yazarken hizalama hatası. Referanslanan parametre, 0 olmayan bir Bit adresindedir.
8230	Parametre, yazma korumalı bir global DB'dedir
8231	Parametre, yazma korumalı bir örnek DB'dedir
8232	Parametre, izin verilen maksimum blok numarasından (DB numarası hatası) daha yüksek bir DB numarası içeriyor.

STATUS (W#16#...)	Açıklama
823A	BUFFER parametresi için DB mevcut değil.
8280	Modülden okuma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil

13.3.5.8 Receive_Reset (alım tamponunun silinmesi)

Tablo 13-29 Receive_Reset yordamı (alıcıyı sıfırla)

LAD/FBD	SCL	Açıklama
	**"Receive_Reset_DB"(REQ:=_bool_in_, PORT:=_word_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	Receive_Reset, CM veya CB'deki alım tamponunu siler.

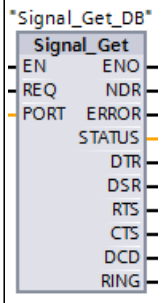
[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-30 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu serbest bırakma girişinin yükselen kenarında alım tamponunun silinmesini etkinleştirir (standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
DONE	OUT	Bool	Bir çevrim boyunca DOĞRU, son talebin hatasız yürütüldüğü anlamına gelir.
ERROR	OUT	Bool	DOĞRU, son talebin hatalarla yürütüldüğü anlamına gelir. Bu çıkış DOĞRU ise STATUS çıkışı ilgili hata kodlarını içerir.
STATUS	OUT	Word	Hata kodu (standart değer: 0)

13.3.5.9 Signal_Get (RS-232 sinyallerinin sorgulanması)

Tablo 13-31 Signal_Get yordamı (RS232 sinyallerinin çağrılması)

LAD/FBD	SCL	Açıklama
	**"Signal_Get_DB"(REQ:=_bool_in_, PORT:=_uint_in_, NDR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, DTR=>_bool_out_, DSR=>_bool_out_, RTS=>_bool_out_, CTS=>_bool_out_, DCD=>_bool_out_, RING=>_bool_out_);**	Signal_Get, RS232 iletişim sinyallerinin güncel durumlarını okur. Bu fonksiyon, yalnızca RS232-CM için geçerlidir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-32 Parametreler için kullanılan veri tipleri

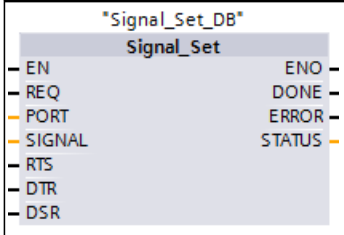
Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişteki yükselen kenarda RS232 sinyal durumları çağrılır (standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır.
NDR	OUT	Bool	Yeni veriler hazırsa ve yordam hatasız tamamlandıysa, bir çevrim boyunca DOĞRU
ERROR	OUT	Bool	Yordam bir hatayla tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)
DTR	OUT	Bool	Veri terminali hazır, modül hazır (çıkış). Standart değer: False
DSR	OUT	Bool	Veri seti hazır, iletişim katılımcısı hazır (giriş). Standart değer: False
RTS	OUT	Bool	Gönderim talebi, modül gönderime hazır (çıkış). Standart değer: False
CTS	OUT	Bool	Gönderime hazır, iletişim katılımcısı verileri alabilir (giriş). Standart değer: False
DCD	OUT	Bool	Veri taşıyıcı algılanır, sinyal seviyesi alınır (her zaman False, desteklenmez)
RING	OUT	Bool	Çağrı göstergesi, gelen çağrı mesajı (her zaman False, desteklenmez)

Tablo 13-33 Koşul kodları

STATUS (W#16#....)	Açıklama
81F0	CM veya CB, RS485'tir ve herhangi bir sinyal mevcut değil
81F4	Blok başlığında hata, örn. yanlış blok tipi veya yanlış blok uzunluğu
8280	Modülden okuma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil

13.3.5.10 Signal_Set (RS-232 sinyallerinin belirlenmesi)

Tablo 13-34 Signal_Set yordamı (RS232 sinyallerinin ayarlanması)

LAD/FBD	SCL	Açıklama
	**"Signal_Set_DB"(REQ:=_bool_in_, PORT:=_word_in_, SIGNAL:=_byte_in_, RTS:=_bool_in_, DTR:=_bool_in_, DSR:=_bool_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	Signal_Set, RS232 iletişim sinyallerinin durumlarını okur. Bu fonksiyon, yalnızca RS232-CM için geçerlidir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-35 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	RS232 sinyallerini ayarlama yordamı, bu girişteki yükselen kenar ile başlatılır (standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
SIGNAL	IN	Byte	Sabit ayarlanacak sinyalleri belirtir: (birden çok sayıda izin verilir) Standart değer: 0 • 01H = RTS • 02H = DTR • 04H = DSR
RTS	IN	Bool	Gönderim talebi, modül ayarlanacak değeri göndermeye hazır (doğru veya yanlış), standart değer: Yanlış
DTR	IN	Bool	Veri terminali, modül ayarlanacak değeri göndermeye hazır (doğru veya yanlış). Standart değer: Yanlış
DSR	IN	Bool	Veri seti hazır (DCE arayüzleri için geçerlidir), kullanılmıyor.
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

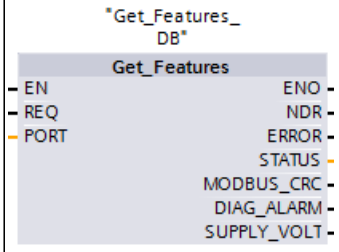
Tablo 13-36 Koşul kodları

STATUS (W#16#....)	Açıklama
81F0	CM veya CB, RS485'tir ve herhangi bir sinyal ayarlanamaz
81F1	Donanım akış kontrolü nedeniyle sinyaller ayarlanamıyor
81F2	Modül bir DTE cihazı olduğu için, DSR ayarlanamıyor
81F3	Modül bir DCE cihazı olduğu için, DTR ayarlanamıyor
81F4	Blok başlığında hata, örn. yanlış blok tipi veya yanlış blok uzunluğu

STATUS (W#16#....)	Açıklama
8280	Modülden okuma sırasında negatif onay
8281	Modüle yazma sırasında negatif onay
8282	DP-Slave veya modül mevcut değil

13.3.5.11 Get_Features

Tablo 13-37 Get_Features yordamı (genişletilmiş fonksiyonların çağrılması)

LAD/FBD	SCL	Açıklama
	**"Get_Features_DB"(REQ:=_bool_in_, PORT:=_word_in_, NDR:=_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MODBUS_CRC=>_bool_out_, DIAG_ALARM=>_bool_out_, SUPPLY_VOLT=>_bool_out);**	Get_Features yordamı, bir modülün genişletilmiş özelliklerini okur.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

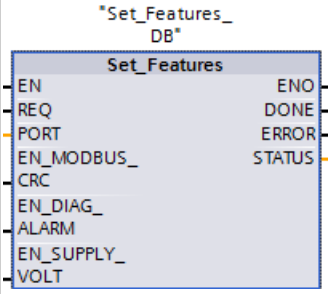
Tablo 13-38 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişte yükselen kenardaki konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
NDR	OUT	Bool	Yeni veriler hazırın olduğunu gösterir.
ERROR	OUT	Bool	Son talep hatalı tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)
MODBUS_CRC*	OUT	Bool	MODBUS CRC oluşturulması ve denetlenmesi
DIAG_ALARM*	OUT	Bool	Teşhis alarmı oluşturma
SUPPLY_VOLT*	OUT	Bool	L+ besleme gerilimi arızası için teşhis sağlanıyor

*Get_Features, fonksiyon sunuluyorsa DOĞRU (1) ve fonksiyon sunulmuyorsa YANLIŞ (0) geri döndürür.

13.3.5.12 Set_Features

Tablo 13-39 Set_Features yordamı (genişletilmiş fonksiyonların ayarlanması)

LAD/FBD	SCL	Açıklama
	**"Set_Features_DB"(** **REQ:=_bool_in_,** **PORT:=_word_in_,** **EN_MODBUS_CRC:=_bool_in_,** **EN_DIAG_ALARM:=_bool_in_,** **EN_SUPPLY_VOLT:=_bool_in_,** **DONE=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_);**	Set_Features, bir modülü destekleyen genişletilmiş fonksiyonları tanımlar.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-40 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişte yükselen kenardaki konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
EN_MODBUS_CRC	IN	Bool	MODBUS CRC oluşturulması ve denetlenmesi • 0: CRC hesaplama KAPALI (Standart) • 1: CRC hesaplama AÇIK Not: Bu parametre, yalnızca sürüm 2.1 CM'leri, CB'li sürüm 4.x CPU'ları ve desantral G/Ç için olan CM PtP modülleri tarafından desteklenir.
EN_DIAG_ALARM	IN	Bool	Teşhis alarmının etkinleştirilmesi • 0: Teşhis alarmı KAPALI • 1: Teşhis alarmı AÇIK (standart)
EN_SUPPLY_VOLT	IN	Bool	L+ besleme gerilimi arızası için teşhisi etkinleştirir: • 0: Teşhis besleme gerilimi devre dışı bırakılmış (standart) • 1: Teşhis besleme gerilimi etkinleştirilmiş
DONE	OUT	Bool	Set_Features'ın çalıştığını belirtir
ERROR	OUT	Bool	Son talep hatalı tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

13.3.6 PtP iletişiminin programlanması

STEP 7'nin sunduğu genişletilmiş yordamlar üzerinden bir kullanıcı programında tasarlanmış ve belirlenmiş bir protokolü kullanarak, kullanıcı programında noktadan noktaya iletişimi yürütebilirsiniz. Bu yordamlar iki kategoriye ayrılır:

- Konfigürasyon yordamları
- İletişim yordamları

Konfigürasyon yordamları

Kullanıcı programının PtP iletişimini başlatabilmesi için, daha önce iletişim arayüzü ve veri gönderme ve alma parametreleri yapılandırılmalıdır.

Arayüz konfigürasyonu ve mesaj konfigürasyonu, her bir CM veya CB için cihaz konfigürasyonunda veya kullanıcı programınızın aşağıda belirtilen yordamları ile gerçekleştirilebilir:

- Port_Config (Sayfa 910)
- Send_Config (Sayfa 913)
- Receive_Config (Sayfa 914)

İletişim yordamları

Noktadan noktaya iletişim yordamları üzerinden kullanıcı programı, iletişim arayüzlerine mesaj gönderebilir ve onlardan mesaj alabilir. Bu yordamlarla verilerin aktarılmasıyla ilgili daha fazla bilgi için Veri tutarlılığı (Sayfa 181) alt bölümüne bakın.

Tümüm PtP yordamları, asenkron çalışır. Bir sorgulama mimarisi yardımıyla, kullanıcı programı gönderim ve alım durumunu saptayabilir. Send_P2P ve Receive_P2P aynı anda çalıştırılabilirler. İletişim modülleri ve iletişim kartı, ilgili gereksinime göre maksimum 1024 Bayt tampon boyutuna kadar gönderim ve alım mesajları tampona alır.

CM'ler ve CB, iletişim katılımcılarına mesaj gönderip, onlardan mesaj alırlar. Mesaj protokolü, belirli bir iletişim arayüzü tarafından alınan veya ona gönderilen bir tamponda yer alır. Tampon ve port, gönderim ve alım parametreleridir:

- Send_P2P (Sayfa 921)
- Receive_P2P (Sayfa 924)

Ek yordamlarla alım tamponu sıfırlanabilir ve özel RS232 sinyalleri sorgulanabilir ve ayarlanabilir:

- Receive_Reset (Sayfa 926)
- Signal_Get (Sayfa 926)
- Signal_Set (Sayfa 927)

13.3.6.1 Sorgulama mimarisi

Alınan mesajları sorgulamak için STEP 7 kullanıcı programı, S7-1200'ün noktadan noktaya yordamlarını periyodik/düzenli olarak çağırmalıdır. Kullanıcı programı aktarım sürecini sorgulayarak aktarımın ne zaman tamamlandığını saptar.

Sorgulama mimarisi: Master'lar

Bir Master için tipik sekans aşağıdaki gibidir:

1. Send_P2P (Sayfa 921) yordamı, CM veya CB'ye bir aktarımı başlatır.
2. Aktarım işleminin durumunu sorgulamak için Send_P2P yordamı ardışık çevrimlerde yürütülür.
3. Send_P2P yordamı aktarımın tamamlandığını bildirdiğinde, kullanıcı kodu yanıtı almaya hazırlanabilir.
4. Receive-P2P (Sayfa 924) yordamı, bir yanıt için sorgulama yapmak üzere tekrarlanarak yürütülür. CM veya CB bir yanıt mesajı kaydettiğinde, Receive_P2P yordamı yanıtı CPU'ya kopyalar ve yeni verilerin alındığını bildirir.
5. Kullanıcı programı yanıtı işleyebilir.
6. 1. adıma geri dönün ve çevrimi tekrarlayın.

Sorgulama mimarisi: Slave

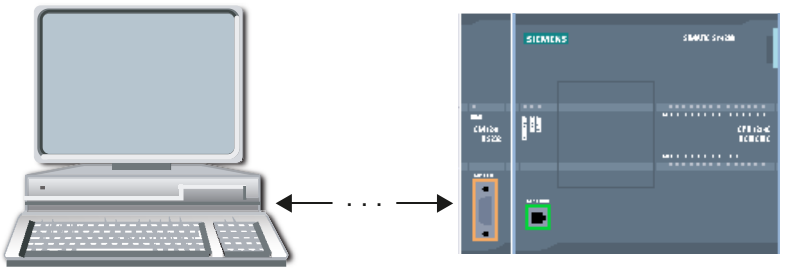
Bir Slave için tipik sekans aşağıdaki gibidir:

1. Kullanıcı programı, her çevrimde Receive_P2P yordamını yürütür.
2. CM veya CB bir talep aldığında, Receive_P2P yordamı yeni verilerin hazır olduğunu bildirir ve talep CPU'ya kopyalanır.
3. Kullanıcı programı talebi işler ve bir yanıt üretir.
4. Send_P2P yordamıyla, yanıt Master'a geri gönderilir.
5. Gönderim işleminin gerçekleştiğinden emin olmak için Send_P2P 'yi tekrar çalıştırın.
6. 1. adıma geri dönün ve çevrimi tekrarlayın.

Master'dan yanıt beklerken işlemin zaman aşımına uğramasını önlemek adına, Slave, Master'dan bir aktarım almak için Receive_P2P 'nin yeteri kadar çağrıldığından emin olmalıdır. Bu görevi başarmak için kullanıcı programı, bir çevrim OB içinden RCV_PTP'yi çağırabilir, bu çevrim süresi, zaman aşımı ayarının süresi dolmadan Master'dan bir aktarım alımına yetecek kadar uzun olmalıdır. OB'nin çevrim süresi, Master'ın zaman aşımı ayarı içinde iki yürütme gerçekleşecek şekilde ayarlanırsa, kullanıcı programı tüm aktarımları kayıpsız alabilir.

13.3.7 Örnek: Noktadan noktaya iletişim

Bu örnekte bir S7-1200 CPU, bir CM 1241 RS232 modülü üzerinden bir terminal emülatörü kullanarak bir PC ile iletişim kurar. Bu örnekte noktadan noktaya konfigürasyon ve STEP 7 programı, CPU'nun PC'den bir mesajın nasıl alındığı ve mesajın yankısının nasıl PC'ye geri döndürüldüğü açıklanmaktadır.



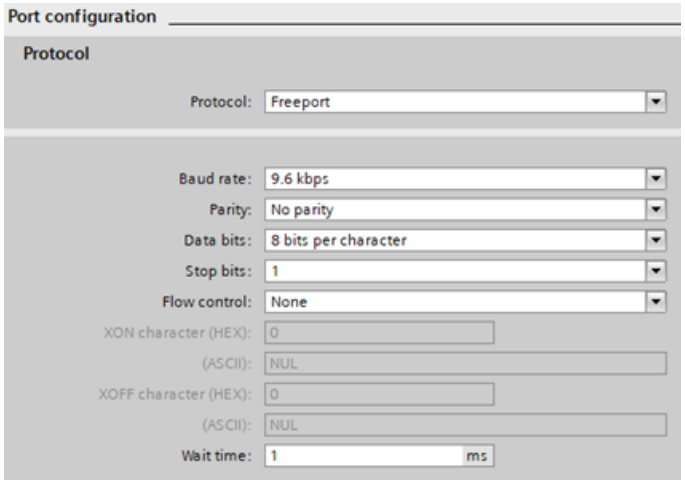
CM 1241 RS232 modülünün iletişim arayüzünü PC'nin RS232 arayüzüne bağlamanız gerekir. Bu ise genellikle COM1'dir. Her iki port da veri terminal ekipmanı (DTE) olduğundan, iki portu bağlarken alım ve gönderim pinlerini (2 ve 3) anahtarlamalısınız. Bunun için aşağıdaki yöntemlerden birini uygulayın:

- Standart bir RS232 kablosuyla birlikte pin 2 ve 3'ü değiştirmek için bir kukla modem adaptörü kullanın.
- Pin 2 ve 3'ü halen değiştirilmiş bir kukla modem kablosu kullanın. Bir kukla modem kablosunu genellikle iki adet 9 kutuplu D soketten tanıyabilirsiniz.

13.3.7.1 İletişim modülünün yapılandırılması

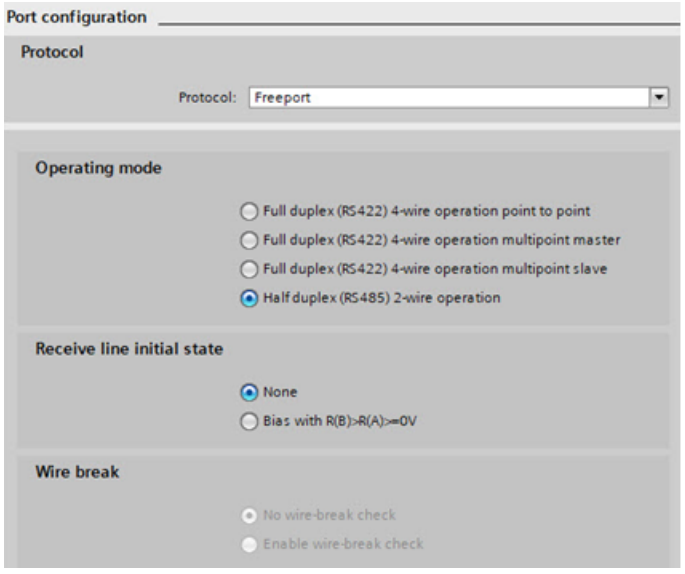
CM 1241'i STEP 7'deki cihaz konfigürasyonunda veya kullanıcı programındaki yordamlarla yapılandırabilirsiniz. Bu örnekte, cihaz konfigürasyonu kullanılmıştır.

- Arayüz konfigürasyonu: Cihaz konfigürasyonunda CM'nin iletişim portuna tıklayın ve portu aşağıdaki gibi yapılandırın:



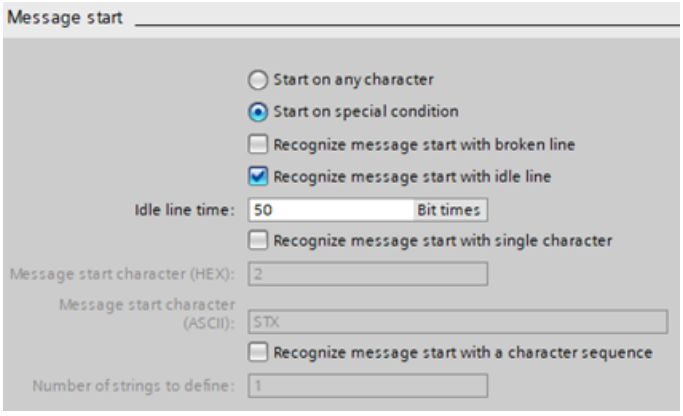
NOT

CM 1241 (RS-422/RS-485) modülü için konfigürasyon ayarları, aşağıda gösterildiği gibi "İşletim türü", "Alım hattının ön ataması" ve "Tel kopması" unsurlarını içerir. Daha fazla bilgi için bkz. RS422 ve RS485'in yapılandırılması (Sayfa 936).

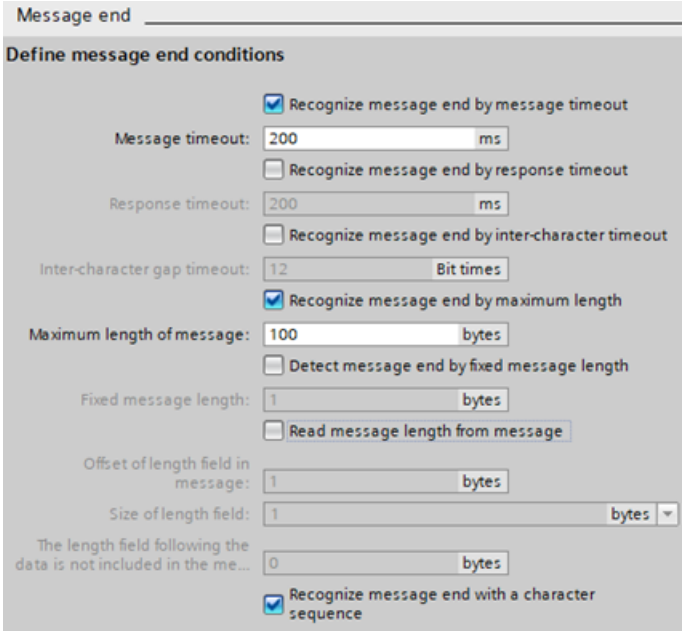


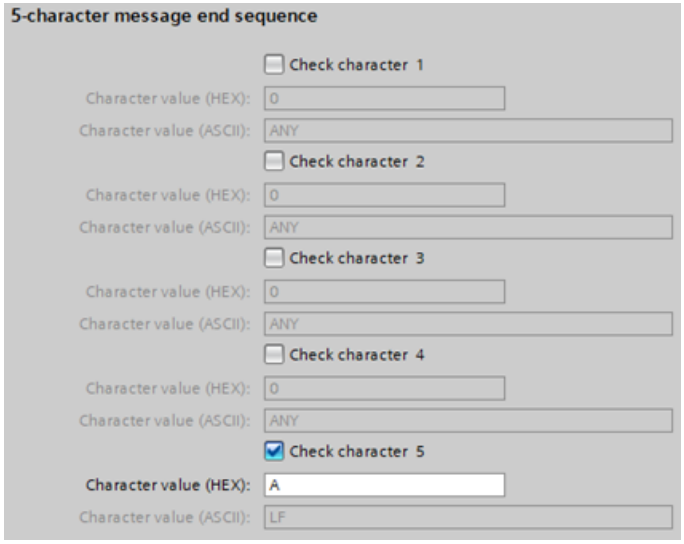
- Gönderim mesajının konfigürasyonu: Mesaj göndermek için ön ayarlı konfigürasyonu kabul edin. Mesaj başlangıcında bir duraklama gönderilmez.

- Mesaj alımının başlangıcının yapılandırılması: CM 1241'i, iletişim hattı en az 50 Bit süresi ile aktif olmadığında (9600 Baud = 50 * 1/9600'de yaklaşık 5 msn) mesaj alımı başlayacak şekilde yapılandırın:



- Mesaj alımı bitişinin yapılandırılması: CM 1241'i, maksimum 100 Bayt veya yeni satır karakteri (10 ondalık veya onaltılık) alındığında mesaj almayı durduracak şekilde yapılandırın. Bitiş karakteri dizisi olarak arka arkaya en fazla beş bitiş karakterine izin verilir. Satır sonu karakteri, dizideki beşinci konumdadır. Dizinin önceki dört bitiş karakteri "ilgili değil" veya seçilmemiş karakterlerdir. CM 1241, "ilgili değil" karakterleri değerlendirmez, ancak mesaj bitişini işaretlemek için önünde 0 veya ek "ilgili değil" karakter bulunan bir satır sonu karakteri bekler.



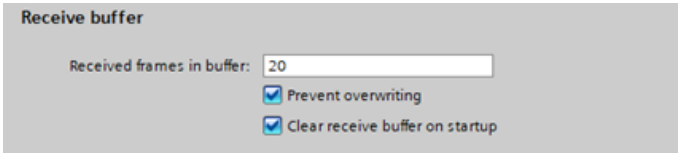


Alım tamponu: Modülün bekleme kuyruğuna aldığı alım mesajlarının sayısını yapılandırın, bekleme kuyruğu dolduğunda modülün daha fazla mesaj almasını durdurun ve STOP'tan RUN (yol alma) işletim durumuna geçişte daha önce alınan mesajları silin.

NOT

Alım tamponunun ayarları

Alım tamponunun ayarları, yalnızca V2.1 sürümüne sahip CM 1241 (RS-232) ve CM 1241 (RS-422/RS-485) için mevcuttur.



13.3.7.2 RS422 ve RS485 işletim türleri

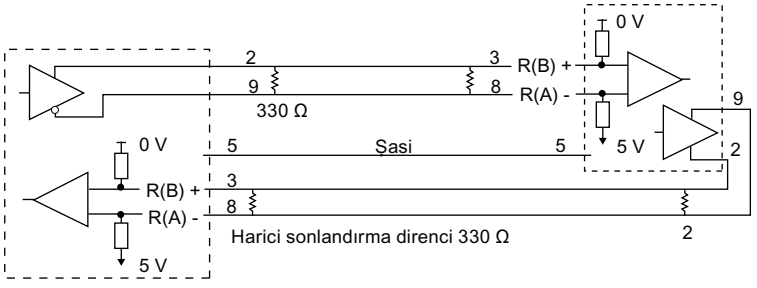
RS422'nin yapılandırılması

RS422 modunda, ağ konfigürasyonunuza bağlı olarak üç işletim türü vardır. Ağınızdaki cihazlara uygun olarak bu işletim türlerinden birini seçin. "Alım hattının çıkış durumu" için çeşitli ayarlar, aşağıda gösterilen durumlara atıfta bulunur.

- Tam dupleks (RS422), dört telli mod (noktadan noktaya bağlantı) Ağınız iki cihaz içeriyorsa bu opsiyonu seçin. "Alım hattı çıkış durumu" için:
  - Ön gerilimi ve sonlandırmayı belirtirken "Yok" öğesini seçin (Durum 3).
  - Dahili ön gerilimi ve sonlandırmayı kullanmak için "ileri yönde ön gerilim" öğesini seçin (Durum 2).
  - Dahili ön gerilimi ve sonlandırmayı kullanmak ve her iki cihaz için kablo kopması algılamasını etkinleştirmek için "Blokaj yönünde ön gerilim" öğesini seçin (Durum 1).
- Tam dupleks (RS422), dört telli mod (çok noktalı Master) Ağınız bir Master ve birden çok Slave içeriyorsa, Master cihaz için bu opsiyonu seçin. "Alım hattı çıkış durumu" için:
  - Ön gerilimi ve sonlandırmayı belirtirken "Yok" öğesini seçin (Durum 3).
  - Dahili ön gerilimi ve sonlandırmayı kullanmak için "ileri yönde ön gerilim" öğesini seçin (Durum 2).
  - Bu işletim türünde kablo kopması algılaması mümkün değildir.
- Tam dupleks (RS422), dört telli mod (çok noktalı Slave): Ağınız bir Master ve birden çok Slave içeriyorsa, tüm Slave cihazları için bu opsiyonu seçin. "Alım hattı çıkış durumu" için:
  - Ön gerilimi ve sonlandırmayı belirtirken "Yok" öğesini seçin (Durum 3).
  - Dahili ön gerilimi ve sonlandırmayı kullanmak için "ileri yönde ön gerilim" öğesini seçin (Durum 2).
  - Dahili ön gerilimi ve sonlandırmayı kullanmak ve Slave'ler için kablo kopması algılamasını etkinleştirmek için "Blokaj yönünde ön gerilim" öğesini seçin (Durum 1).

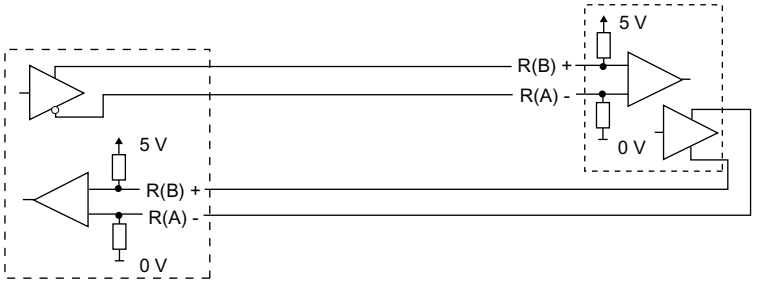
Durum 1: Kablo kopması algılamalı RS422

- İşletim türü: RS422
- Alım hattı çıkış durumu: Blokaj yönünde ön gerilim (ön gerilim, R(A) > R(B) > 0 V)
- Kablo kopması: Kablo kopması algılaması etkinleştirilmiş (gönderici her zaman aktif)



Durum 2: RS422 kablo kopması algılaması yok, ileri yönde ön gerilim

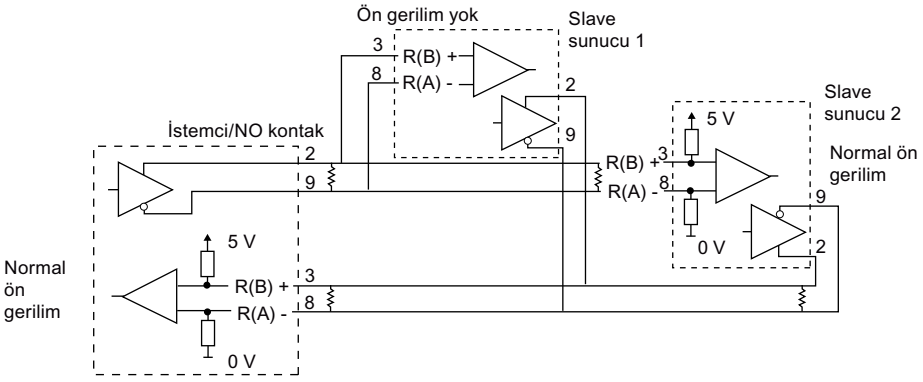
- İşletim türü: RS422
- Alım hattı çıkış durumu: İleri yönde ön gerilim (ön gerilim, R(B) > R(A) > 0 V)
- Kablo kopması: Kablo kopması algılaması yok (gönderici, yalnızca gönderimde etkinleştiriliyor)



Durum 3: RS422: Kablo kopması algılaması yok, ön gerilim yok

- İşletim türü: RS422
- Alım hattı çıkış durumu: Ön gerilim yok
- Kablo kopması: Kablo kopması algılaması yok (gönderici, yalnızca gönderimde etkinleştiriliyor)

Kullanıcı tarafından, ön gerilim ve sonlandırma ağın uç düğümlerinde sağlanır.



RS485'in yapılandırılması

RS485 modunda yalnızca bir işletim türü vardır. "Alım hattının çıkış durumu" için çeşitli ayarlar, aşağıda gösterilen durumlara atıfta bulunur.

- Yarı dupleks (RS485), iki telli mod. "Alım hattı çıkış durumu" için:
  - Ön gerilimi ve sonlandırmayı belirtirken "Yok" öğesini seçin (Durum 5).
  - Dahili ön gerilimi ve sonlandırmayı kullanmak için "ileri yönde ön gerilim" öğesini seçin (Durum 4).

Durum 4: RS485: İleri yönde ön gerilim

- İşletim türü: RS485
- Alım hattı çıkış durumu: İleri yönde ön gerilim (ön gerilim, R(B) > R(A) > 0 V)



Durum 5: RS485: Ön gerilim yok (harici ön gerilim)

- İşletim türü: RS485
- Alım hattı çıkış durumu: Ön gerilim yok (harici ön gerilim gerekli)

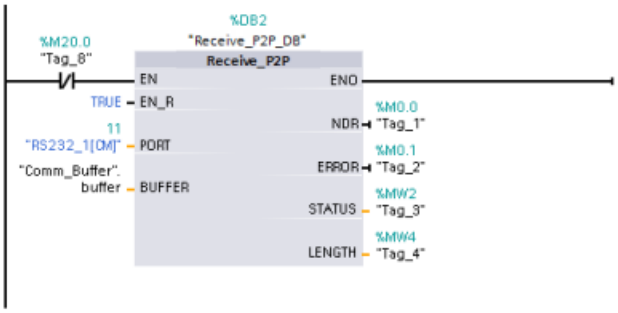


13.3.7.3 STEP 7 programının programlanması

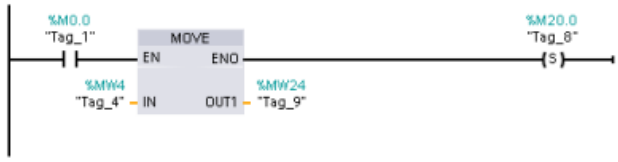
Örnek programda, iletişim tamponu için bir genel veri bloğu, terminal emülatöründen veri almak için bir RCV_PTP yordamı (Sayfa 1054) ve tampon yankısını terminal emülatörüne geri göndermek için bir SEND_PTP yordamı (Sayfa 1053) kullanılmaktadır. Bu örneği programlamada, veri bloğu konfigürasyonunu ve ana program bloğu OB 1'i aşağıda açıklandığı gibi ekleyin.

Global veri bloğu "Comm_Buffer": Bir genel veri bloğu (DB) oluşturun ve "Comm_Buffer" olarak adlandırın. Veri bloğunda, "buffer" adında "Array veri tipi [0 .. 99] Bayt" olarak bir değer oluşturun.

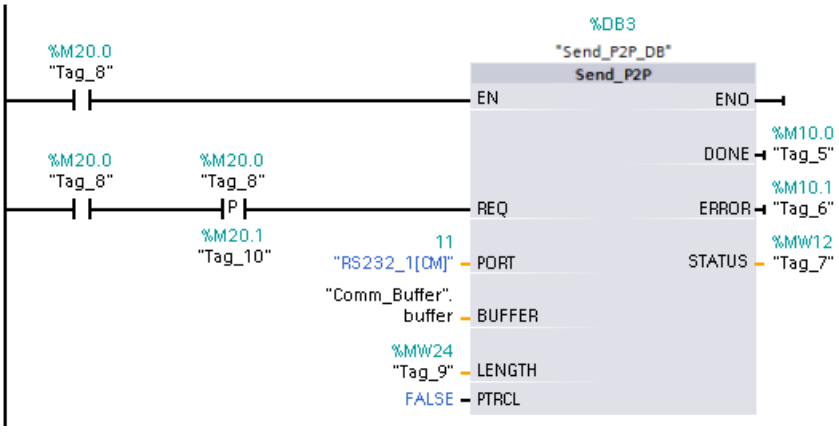
Ağ 1: SEND_PTP aktif değilken, her zaman RCV_PTP yordamını etkinleştirin. MW20.0'daki Tag_8, ağ 4'te aktarım işleminin ne zaman tamamlandığını ve dolayısıyla iletişim modülünün mesajı alımına hazır olduğunu gösterir.



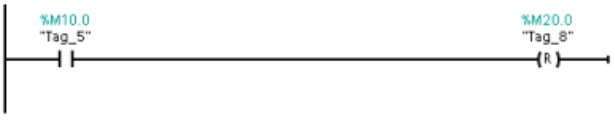
Ağ 2: RCV_PTP yordamı tarafından ayarlanan NDR değerini (M0.0'da Tag_1) kullanarak, alınan Bayt sayısının bir kopyasını oluşturun ve SEND_PTP yordamını tetiklemek için bir markör (M20.0'da Tag_8) ayarlayın.



Ağ 3: M20.0 markörü ayarlandığında SEND_PTP yordamını etkinleştirin. Bu markör ile REQ girişini, bir çevrim boyunca DOĞRU olarak ayarlamak için de kullanırsınız. REQ girişi, SEND_PTP yordamına yeni bir talebin gönderileceğini bildirir. REQ girişi, SEND_PTP'nin yürütülmesi sırasında yalnızca DOĞRU olarak ayarlanabilir. Aktarım tamamlanana dek her çevrimde SEND_PTP yordamı yürütülür. Mesajın son Bayt'ı CM 1241 tarafından gönderildiğinde aktarım tamamlanmış olur. Aktarım tamamlandığında, DONE yürütmesi sırasında SEND_PTP çıkışı (M10.0'da Tag_5) DOĞRU olarak ayarlanır.



Ağ 4: SEND_PTP'nin DONE çıkışını denetleyin ve aktarım işlemi tamamlandığında aktarım markörünü (M20.0'da Tag_8) sıfırlayın. Aktarım markörü sıfırlandığında, sonraki mesajı almak için ağ 1'deki RCV_PTP yordamı etkinleştirilir.



13.3.7.4 Terminal emülatörünün yapılandırılması

Örnek programı çalıştırmak için terminal emülatörünü kurmalısınız. PC'nizdeki örn. HyperTerminal gibi neredeyse tüm terminal emülatörlerini kullanabilirsiniz. Ayarları aşağıdaki gibi değiştirmeden önce terminal emülatörünün devre dışı bırakılmış olduğundan emin olun:

1. PC'deki RS232 bağlantısını (genellikle COM1) kullanmak için terminal emülatörünü ayarlayın.
2. Portu; 9600 Baud için, 8 veri Bit'i, parite yok, 1 stop Bit'i ve akış kontrolü yok olarak yapılandırın.
3. ANSI terminaline öykünmek için terminal emülatörü ayarlarını değiştirin.
4. Terminal emülatörünün ASCII özelliğini, her satır sonunda (kullanıcı Enter tuşuna bastıktan sonra) yeni bir satır gönderecek şekilde yapılandırın.
5. Terminal emülatörünün yazılanları görüntülemesi için, karakterlerin yerel bir yankısını geri döndürün.

13.3.7.5 Örnek programın çalıştırılması

Örnek programı çalıştırmak için aşağıdaki yöntemi izleyin:

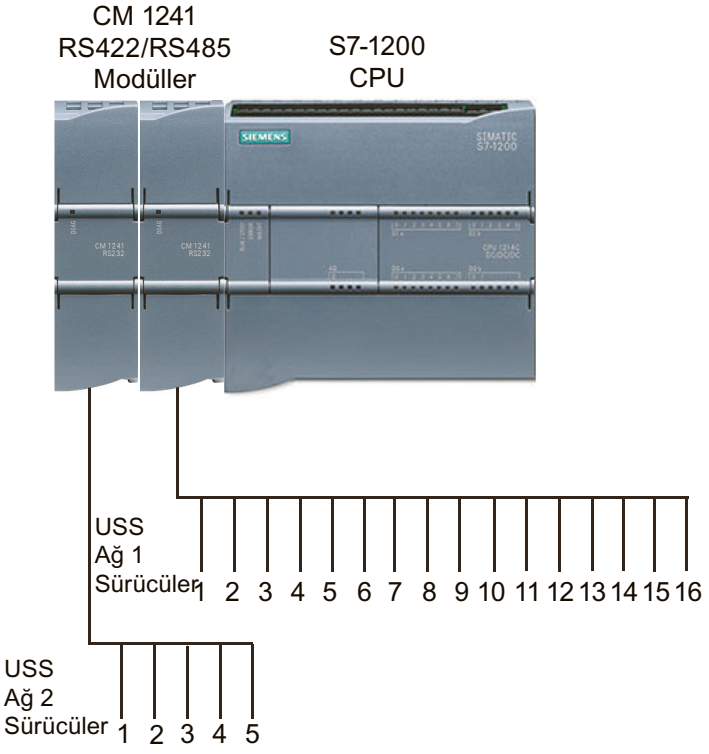
1. STEP 7 programını CPU'ya yükleyin ve RUN işletim durumuna getirin.
2. Konfigürasyon değişikliklerini uygulamak ve CM 1241'e bir terminal oturumu açmak için, terminal emülatöründe bağlan butonuna tıklayın.
3. PC'de karakterleri yazın ve Enter tuşuna basın.

Terminal emülatörü, karakterleri CM 1241'e ve CPU'ya gönderir. Bunun ardından CPU programı, karakterlerin yankısını terminal emülatörü geri gönderir.

13.4 Üniversal seri arayüz (USS) üzerinden iletişim

USS yordamları, üniversal seri arayüz (USS) protokolünü destekleyen motor sürücülerinin işleyişine kumanda eder. USS yordamları ile CM 1241 RS485 iletişim modülünü veya bir CB 1241 RS485 iletişim kartını kullanarak, RS485 bağlantıları üzerinden birden çok sürücüyle iletişim kurabilirsiniz. Bir S7-1200 CPU'ya üç adede kadar CM 1241 RS422/RS485 modülü ve bir CB 1241 RS485 kartı takılabilir. Her bir RS485 portu, on altı adede kadar sürücüyü çalıştırabilir.

USS protokolü, bir seri veriyolu üzerinden iletişim kurmak için bir Master/Slave ağı kullanır. Master, seçili bir Slave'e mesaj göndermek için bir adres parametresi kullanır. Bir Slave'in kendisi, bir talep almadan önce asla gönderimde bulunmaz. Münferit Slave'ler arasında doğrudan mesaj aktarımı mümkün değildir. USS iletişimi yarı dupleks işletimde çalışır. Aşağıdaki USS görselinde, bir sürücünün örnek uygulamasına yönelik bir ağ diyagramı gösterilmektedir.



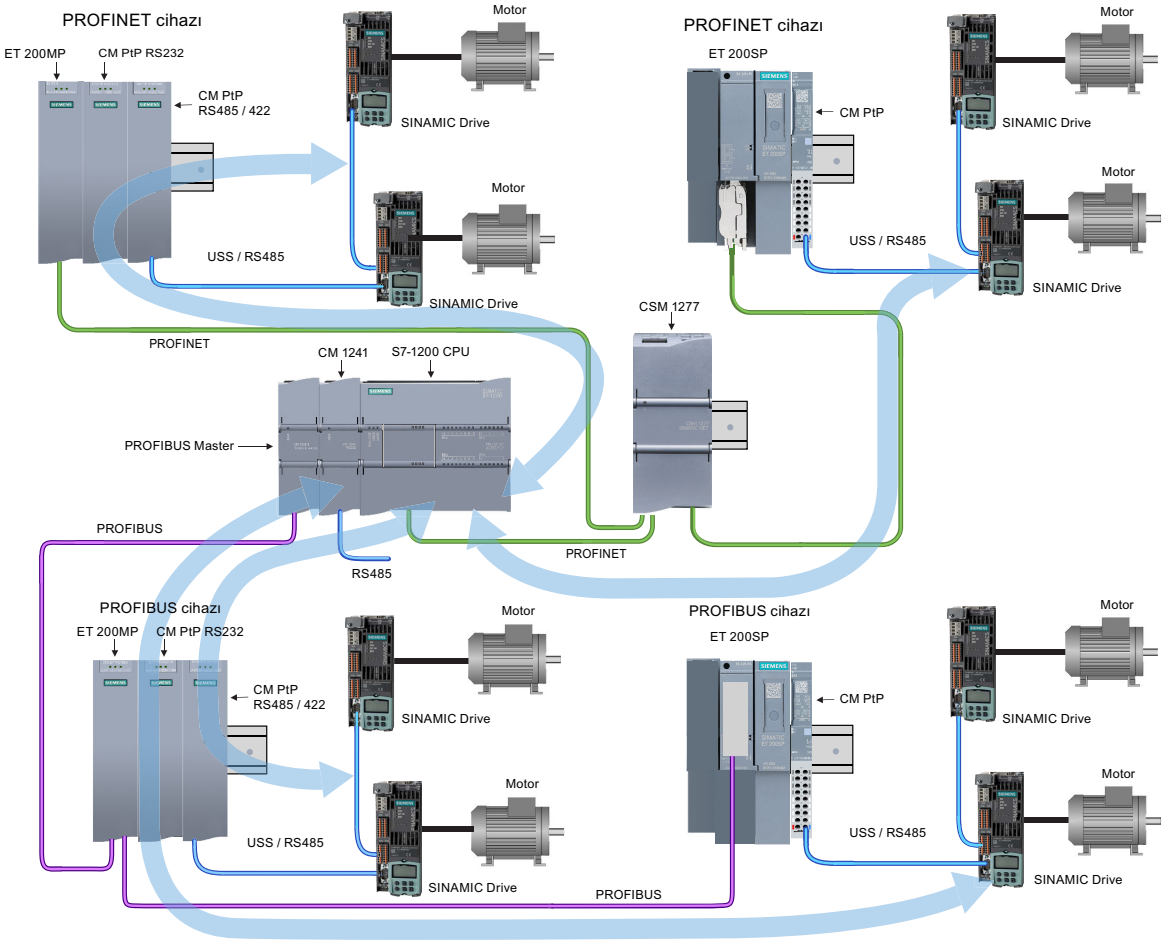
PROFIBUS veya PROFINET üzerinden USS iletişimi

USS iletişimi ile çeşitli cihazlarla (RFID okuma cihazları, GPS cihazları ve diğer cihazlar) iletişim kurmak için bir desantral PROFINET veya PROFIBUS çevrebirim modül taşıyıcısını kullanabilirsiniz:

- PROFINET (Sayfa 561): S7-1200 CPU'nun Ethernet arayüzü bir PROFINET arayüz modülüne bağlıdır. USS sürücüleriyle seri iletişim, arayüz modülü ile modül taşıyıcısındaki PtP iletişim modülleri aracılığıyla mümkün olur.
- PROFIBUS (Sayfa 734): S7-1200 CPU ile modül taşıyıcısının sol tarafına bir PROFIBUS iletişim modülü takılır. PROFIBUS iletişim modülü, bir PROFIBUS arayüz modülü içeren bir modül taşıyıcısına bağlanır. USS sürücüleriyle seri iletişim, arayüz modülü ile modül taşıyıcısındaki PtP iletişim modülleri aracılığıyla mümkün olur.

S7-1200, iki set USS yordamını destekler:

- USS yordamları (Sayfa 944): Bu USS yordamları, eski yordamların tüm fonksiyonlarını artı PROFINET ve PROFIBUS için desantral çevrebirimlere bağlanma opsiyonunu da sağlar. Bu USS yordamları, modül taşıyıcısındaki PtP iletişim modülleri ile desantral çevrebirimler ve USS sürücüleri arasındaki iletişimin yapılandırılmasına olanak tanır. S7 1200 CM 1241 tipi modüllerin bu USS yordamlarını kullanabilmesi için en azından Firmware sürümü V2.1 gereklidir.
- Eski USS yordamları (Sayfa 1061): Bu USS yordamları, S7-1200'ün V4.0 sürümünden önce kullanılmaktaydı ve yalnızca bir CM 1241 iletişim modülü veya bir CB 1241 iletişim kartı kullanarak seri iletişim mümkündü.



Mavi ok ile cihazlar arasındaki çift yönlü iletişim akışı gösterilmektedir.

STEP 7, USS yordamlarının farklı sürümlerini sağlar. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

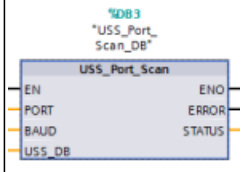
USS protokollerinin kullanım gereklilikleri

Dört USS yordamı, USS protokolünü desteklemek için iki FB ve iki FC kullanır. Her bir USS ağı için, USS_Port_Scan için bir örnek veri bloğu (DB) ve tüm USS_Drive_Control çağrıları için bir örnek veri bloğu kullanılır. STEP 7 bilgi sisteminde veya Siemens Industry Online Support'ta "USS protokolü kullanım gereklilikleri (https://support.industry.siemens.com/cs/ww/en/view/109798671/140076559243)" başlığı altında diğer gereklilikleri bulabilirsiniz.

13.4.1 USS yordamları

13.4.1.1 USS_Port_Scan (USS ağı üzerinden iletişim düzenlenmesi)

Tablo 13-41 USS_Port_Scan yordamı

LAD/FBD	SCL	Açıklama
	**USS_Port_Scan(PORT:=_uint_in_, BAUD:=_dint_in_, ERROR=>_bool_out_, STATUS=>_word_out_, USS_DB:=_fbtref_inout_);**	USS_Port_Scan yordamı, bir USS ağı üzerinden iletişimi yönetir.

Tablo 13-42 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
PORT	IN	Port	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır.
BAUD	IN	DInt	USS iletişimi için Baud hızı.
USS_DB	INOUT	USS_BASE	Programınıza bir USS_Drive_Control yordamı eklendiğinde, oluşturulan ve başlatılan örnek DB'nin adı.
ERROR	OUT	Bool	DOĞRU ise, bu çıkış bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir.
STATUS	OUT	Word	Talebin durum değeri, çevrimin veya başlatmanın sonucunu gösterir. "USS_Extended_Error" değişkenindeki bazı durum kodları için daha fazla bilgi mevcuttur.

Normal olarak programdaki PtP iletişim portu başına yalnızca bir USS_Port_Scan yordamı mevcuttur ve bu fonksiyon bloğuna (FB) yapılan her çağrı, tek bir sürücüye veya bir sürücüden yapılan bir aktarıma kumanda eder. Bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları, aynı örnek DB'yi kullanmalıdır.

Sürücünün zaman aşımına uğramaması için, programınızın USS_Port_Scan yordamını yeterli sayıda yürütmesi gerekir. USS_Port_Scan, sürücülerin zaman aşımına uğramasını önlemek ve en son USS veri güncellemelerini USS_Drive_Control çağrıları için kullanılabilir durumda tutmak için tipik olarak bir uyandırma alarmı OB'sinden çağrılır.

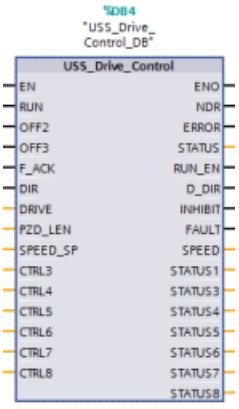
NOT

Bir CB 1241 ile USS protokolü ve USS_Port_Scan yordamı için kütüphaneyi kullandığınızda, LINE_PRE veri bloğu değişkeni için 0 değerini (çıkış durumu yok) ayarlamanız gerekir. LINE_PRE veri bloğu değişkeni için standart değer olan 2, USS_Port_Scan yordamının 16#81AB hata değerini geri döndürmesine yol açar. LINE_PRE veri bloğu değişkeni, USS_Port_Scan yordamının veri bloğunda bulabilirsiniz (genellikle USS_Port_Scan_DB adı kullanılır).

LINE_PRE'nin başlangıç değerinin 0 (sıfır) olarak değiştirildiğini dikkate alın.

13.4.1.2 USS_Drive_Control (sürücü ile veri alışverişi)

Tablo 13-43 USS_Drive_Control yordamı

LAD/FBD	SCL	Açıklama
	**"USS_Drive_Control_DB"(** **RUN:=_bool_in_,** **OFF2:=_bool_in_,** **OFF3:=_bool_in_,** **F_ACK:=_bool_in_,** **DIR:=_bool_in_,** **DRIVE:=_usint_in_,** **PZD_LEN:=_usint_in_,** **SPEED_SP:=_real_in_,** **CTRL3:=_word_in_,** **CTRL4:=_word_in_,** **CTRL5:=_word_in_,** **CTRL6:=_word_in_,** **CTRL7:=_word_in_,** **CTRL8:=_word_in_,** **NDR=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **RUN_EN=>_bool_out_,** **D_DIR=>_bool_out_,** **INHIBIT=>_bool_out_,** **FAULT=>_bool_out_,** **SPEED=>_real_out_,** **STATUS1=>_word_out_,** **STATUS3=>_word_out_,** **STATUS4=>_word_out_,** **STATUS5=>_word_out_,** **STATUS6=>_word_out_,** **STATUS7=>_word_out_,** **STATUS8=>_word_out_);**	USS_Drive_Control yordamı, talep mesajları üreterek ve sürücünün yanıt mesajlarını değerlendirerek bir sürücü ile veri alışverişinde bulunur. Her bir sürücü için ayrı bir fonksiyon bloğu kullanılmalıdır, ancak bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları aynı örnek veri bloğunu kullanmalıdır. İlk USS_Drive_Control yordamını eklediğinizde, DB adını girmelisiniz. Bunun ardından, ilk yordamı eklediğinizde oluşturulan bu DB'ye bakın. Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[1] LAD ve FBD: Tüm parametreleri görmek için kutuyu genişletin. Bunun için kutunun alt kısmına tıklayın. Gri olarak görüntülenen parametre bağlantıları opsiyoneldir, parametrelendirme gerekmez.

Tablo 13-44 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
RUN	IN	Bool	Sürücünün başlatma Bit'i: Bu parametre DOĞRU ise, bu giriş, sürücünün önceden ayarlanmış devir sayısında çalışmasını sağlar. Sürücü, RUN işletim durumundayken YANLIŞ olarak değişirse, motor hareketsiz durum aşamasına kadar yavaşlar. Bu davranış, gerilim beslemesini kapatmaktan (OFF2) ve motoru frenlemekten (OFF3) farklıdır.
OFF2	IN	Bool	"Hareketsiz duruma yavaşla" Bit'i: Eğer bu parametre YANLIŞ ise, bu Bit frenlemeksizin sürücünün yavaşlayarak durma aşamasına gelmesine neden olur.
OFF3	IN	Bool	Hızlı Stop Bit'i: Bu parametre YANLIŞ ise, bu Bit, sürücüyü frenleyerek hızlı durmaya neden olur.
F_ACK	IN	Bool	Hata onaylama Bit'i: Bu Bit, bir sürücünün hata Bit'ini sıfırlamak için kullanılır. Bu Bit, hata silindikten sonra ayarlanır ve böylece sürücü önceki hatanın artık bildirilmesine gerek olmadığını anlar.
DIR	IN	Bool	Sürücü yönünün kumandası: Bu Bit, sürücü ileri yönde çalışacağı zaman ayarlanır (SPEED_SP pozitif olduğunda).
DRIVE	IN	USInt	Sürücünün adresi: Bu giriş, USS sürücüsünün adresidir. Geçerli aralık, sürücü 1 ile sürücü 16 arasındadır.
PZD_LEN	IN	USInt	Sözcük uzunluğu: Bu, Proses Durumu Değerleri (PZD) veri sözcüklerinin sayısıdır. Geçerli değerler 2, 4, 6 veya 8 sözcüktür. Standart değer, 2'dir.
SPEED_SP	IN	Real	Devir sayısı hedef değeri: Bu, yapılandırılan frekansın yüzdesi cinsinden sürücünün devir sayısıdır. Bir pozitif değer, sürücünün ileriye doğru çalıştığı anlamına gelir (DIR doğru ise). Geçerli aralık, 200,00 ile -200,00 arasındadır.
CTRL3	IN	Word	Kumanda sözcüğü 3: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL4	IN	Word	Kumanda sözcüğü 4: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL5	IN	Word	Kumanda sözcüğü 5: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL6	IN	Word	Kumanda sözcüğü 6: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL7	IN	Word	Kumanda sözcüğü 7: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL8	IN	Word	Kumanda sözcüğü 8: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
NDR	OUT	Bool	Yeni veriler hazır: Bu parametre DOĞRU ise, Bit, yeni bir iletişim talebinden gelen verilerin çıkışta mevcut olduğunu bildirir.
ERROR	OUT	Bool	Hata oluştu: DOĞRU ise, bu bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir. Bir hata durumunda diğer tüm çıkışlar sıfıra ayarlanır. İletişim hataları, yalnızca USS_Port_Scan yordamının ERROR ve STATUS çıkışlarında bildirilir.

Parametre ve veri tipi		Veri tipi	Açıklama
STATUS	OUT	Word	Talebin durum değeri, çevrimin sonucunu gösterir. Bu, sürücü tarafından yayınlanan bir durum sözcüğü değildir.
RUN_EN	OUT	Bool	İşletim serbest bırakılır: Bu Bit, sürücünün çalışıp çalışmadığını bildirir.
D_DIR	OUT	Bool	Sürücü yönü: Bu Bit, sürücünün ileriye doğru çalışıp çalışmadığını bildirir.
INHIBIT	OUT	Bool	Sürücü bloke edilmiş: Bu Bit, sürücü blokaj Bit'inin durumunu bildirir.
FAULT	OUT	Bool	Sürücü hatası: Bu Bit, sürücüde bir hata oluştuğunu bildirir. Bu Bit'i silmek için arızayı gidermeli ve F_ACK Bit'ini ayarlamalısınız.
SPEED	OUT	Real	Sürücü devir sayısının gerçek değeri (sürücü durum sözcüğü 2'nin ölçek-lendirilmiş değeri): Bu, yapılandırılan devir sayısının yüzdesi cinsinden sürücünün devir sayısıdır.
STATUS1	OUT	Word	Sürücünün durum sözcüğü 1: Bu değer, bir sürücünün sabit durum Bit'lerini içerir.
STATUS3	OUT	Word	Sürücünün durum sözcüğü 3: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS4	OUT	Word	Sürücünün durum sözcüğü 4: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS5	OUT	Word	Sürücünün durum sözcüğü 5: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS6	OUT	Word	Sürücünün durum sözcüğü 6: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS7	OUT	Word	Sürücünün durum sözcüğü 7: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS8	OUT	Word	Sürücünün durum sözcüğü 8: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.

USS_Drive_Control ilk kez yürütüldüğünde, USS adresi (DRIVE parametresi) tarafından belirtilen sürücü, örnek DB'de başlatılır. Bu başlatmanın ardından, takip eden USS_Port_Scan yordamları bu sürücü numarası üzerinden o sürücü ile iletişime başlayabilir.

Sürücü numarasını değiştirirseniz, örnek DB'nin başlatılması için CPU'nun önce STOP ve ardından tekrar RUN işletim durumuna alınması gerekir. Giriş parametreleri USS gönderim tamponunda yapılandırılır ve varsa çıkışlar "önceki" geçerli bir yanıt tamponundan okunur. USS_Drive_Control yordamının yürütülmesi sırasında herhangi bir veri aktarımı gerçekleşmez. USS_Port_Scan'in yürütülmesinden sonra sürücüler iletişim kurar. USS_Drive_Control yalnızca gönderilecek mesajları yapılandırır ve önceki bir talep ile alınmış olabilecek verileri değerlendirir.

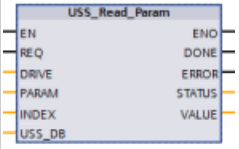
Sürücünün dönüş yönüne, DIR (Bool) girişi veya SPEED_SP girişindeki (Real) ön işaret (pozitif veya negatif) üzerinden kumanda edebilirsiniz. Aşağıdaki tabloda, motorun ileri doğru çalıştığını varsayarak sürücünün dönüş yönünü belirlemek için bu girişlerin birlikte çalışması açıklanmaktadır.

Tablo 13-45 SPEED_SP ve DIR parametreleri arasındaki etkileşim

SPEED_SP	DIR	Sürücünün dönüş yönü:
Değer > 0	0	Geriye
Değer > 0	1	İleri doğru
Değer < 0	0	İleri doğru
Değer < 0	1	Geriye

13.4.1.3 USS_Read-Param (sürücüden parametre okuma)

Tablo 13-46 USS_Read_Param yordamı

LAD/FBD	SCL	Açıklama
	**USS_Read_Param(REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, VALUE=>_variant_out_, USS_DB:=_fbtref_inout_);**	USS_Read_Param yordamı, bir sürücüden bir parametre okur. Bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları aynı veri bloğunu kullanmalıdır. USS_Read_Param, ana programdaki bir çevrim OB'sinden çağrılmalıdır.

Tablo 13-47 Parametreler için kullanılan veri tipleri

Parametre tipi		Veri tipi	Açıklama
REQ	IN	Bool	Gönderim talebi: REQ DOĞRU ise, yeni bir okuma talebi gerekli olur. Bu parametre için talep halen beklemedeyse, bu işlem yok sayılır.
DRIVE	IN	USInt	Sürücünün adresi: DRIVE, USS sürücünün adresidir. Geçerli aralık, sürücü 1 ile sürücü 16 arasındadır.
PARAM	IN	UInt	Parametre numarası: PARAM, hangi sürücü parametresine yazılacağını belirtir. Bu parametre için aralık, 0 ile 2047 arasındadır. Bazı sürücülerdeki en yüksek değerli Bayt, 2047'den büyük PARAM değerlerine erişebilir. Genişletilmiş bir aralığa erişim hakkında daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
INDEX	IN	UInt	Parametre indeksi: INDEX, hangi sürücü parametresi indeksine yazılacağını belirtir. Bu, en düşük Bayt'ın gerçek indeks değeri olduğu 16 Bit'lik bir değerdir ve (0 ila 255) aralığındadır. En yüksek değerli Bayt sürücü tarafından da kullanılabilir ve sürücüye özgüdür. Daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
USS_DB	INOUT	USS_BASE	Programınıza bir USS_Drive_Control yordamı eklendiğinde, oluşturulan ve başlatılan örnek DB'nin adı.
VALUE	IN	Word, Int, UInt, DWord, DInt, UDInt, Real	Bu, okunan parametrenin değeridir ve yalnızca DONE Bit'i DOĞRU ise geçerlidir.
DONE[1]	OUT	Bool	Bu parametre DOĞRU ise, VALUE çıkışında daha önce talep edilen okuma parametresinin değeri bulunur. Bu Bit, USS_Drive_Control yordamı sürücüden okuma yanıtı algıladığında ayarlanır. Bu Bit, şu durumlarda sıfırlanır: Yanıt verilerini başka bir USS_Read_Param sorgusu aracılığıyla veya bir sonraki iki USS_Drive_Control çağrısının ikincisinde talep edersiniz.
ERROR	OUT	Bool	Hata oluştu: DOĞRU ise, ERROR bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir. Bir hata durumunda diğer tüm çıkışlar sıfıra ayarlanır. İletişim hataları yalnızca USS_Port_Scan yordamının ERROR ve STATUS çıkışlarında bildirilir.
STATUS	OUT	Word	STATUS, okuma talebinin sonucunu belirtir. "USS_Extended_Error" değişkenindeki bazı durum kodları için daha fazla bilgi mevcuttur.

[1] DONE Bit'i, referans verilen motor sürücüsünden geçerli verilerin okunduğunu ve CPU'ya iletildiğini gösterir. USS kütüphanesinin başka bir parametreyi hemen okuyabildiğini belirtmez. Parametre kanalı ilgili sürücü tarafından kullanılmak üzere serbest bırakılmadan önce, motor sürücüsüne boş bir parametre karakteristik değeri (PKD) talebi gönderilmeli ve ayrıca yordam tarafından onaylanmalıdır. Belirli motor sürücüsü için hemen bir USS_Read_Param veya USS_Write_Param FC'nin çağrılması, 0x818A hatasıyla sonuçlanır.

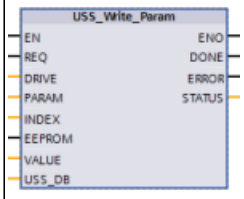
13.4.1.4 USS_Write_Param (sürücüdeki parametreyi değiştirme)

NOT

EEPROM yazma yordamları (bir USS sürücüsündeki EEPROM için)

EEPROM yazma işleminin kullanımında aşırıya kaçmayın. EEPROM'un kullanım ömrünü uzatmak için, EEPROM yazma işlemlerinin sayısını mümkün olduğu kadar düşük tutun.

Tablo 13-48 USS_Write_Param yordamı

LAD/FBD	SCL	Açıklama
	**USS_Write_Param(REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, EEPROM:=_bool_in_, VALUE:=_variant_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, USS_DB:=_fbtref_inout_);**	USS_Write_Param yordamı, sürücüdeki bir parametreyi değiştirir. Bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları aynı veri bloğunu kullanmalıdır. USS_Write_Param, bir ana programın çevrim OB'sinden çağrılmalıdır.

Tablo 13-49 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Gönderim talebi: REQ DOĞRU ise, yeni bir yazma talebi gerekli olur. Bu parametre için talep halen beklemedeyse, bu işlem yok sayılır.
DRIVE	IN	USInt	Sürücünün adresi: DRIVE, USS sürücünün adresidir. Geçerli aralık, sürücü 1 ile sürücü 16 arasındadır.
PARAM	IN	UInt	Parametre numarası: PARAM, hangi sürücü parametresine yazılacağını belirtir. Bu parametre için aralık, 0 ile 2047 arasındadır. Bazı sürücülerdeki en yüksek değerli Bayt, 2047'den büyük PARAM değerlerine erişebilir. Genişletilmiş bir aralığa erişim hakkında daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
INDEX	IN	UInt	Parametre indeksi: INDEX, hangi sürücü parametresi indeksine yazılacağını belirtir. Bu, en düşük Bayt'ın gerçek indeks değeri olduğu 16 Bit'lik bir değerdir ve (0 ila 255) aralığındadır. En yüksek değerli Bayt sürücü tarafından da kullanılabilir ve sürücüye özgüdür. Daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
EEPROM	IN	Bool	Sürücünün EEPROM'unda saklama: DOĞRU ise, sürücü parametresinin yazma işlemi sürücü EEPROM'unda saklanır. YANLIŞ ise, yazılan değer yalnızca geçici olarak saklanır ve sürücüye bir sonraki güç verildiğinde silinir.
VALUE	IN	Word, Int, UInt, DWord, DInt, UDInt, Real	Yazılacak parametrenin değeri. REQ durum değiştirdiğinde geçerli olmalıdır.
USS_DB	INOUT	USS_BASE	Programınıza bir USS_Drive_Control yordamı eklendiğinde, oluşturulan ve başlatılan örnek DB'nin adı.

[1] DONE Bit'i, referans verilen motor sürücüsünden geçerli verilerin okunduğunu ve CPU'ya iletildiğini gösterir. USS kütüphanesinin başka bir parametreyi hemen okuyabildiğini belirtmez. Parametre kanalı ilgili sürücü tarafından kullanılmak üzere serbest bırakılmadan önce, motor sürücüsüne boş bir parametre karakteristik değeri (PKD) talebi gönderilmeli ve ayrıca yordam tarafından onaylanmalıdır. Belirli motor sürücüsü için hemen bir USS_Read_Param veya USS_Write_Param FC'nin çağrılması, 0x818A hatasıyla sonuçlanır.

Parametre ve veri tipi		Veri tipi	Açıklama
DONE[1]	OUT	Bool	DOĞRU ise, DONE, VALUE girişinin sürücüye yazıldığını gösterir. Bu Bit, USS_Drive_Control yordamının sürücüden yazma yanıtı algıladığında ayarlanır. Bu Bit, yanıt verilerini başka bir USS_Drive_Control sorgusu aracılığıyla talep ettiğinizde veya sonraki iki USS_Drive_Control çağrısının ikincisinde sıfırlanır.
ERROR	OUT	Bool	DOĞRU ise, ERROR bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir. Bir hata durumunda diğer tüm çıkışlar sıfıra ayarlanır. İletişim hataları yalnızca USS_Port_Scan yordamının ERROR ve STATUS çıkışlarında bildirilir.
STATUS	OUT	Word	STATUS, yazma talebinin sonucunu belirtir. "USS_Extended_Error" değişkenindeki bazı durum kodları için daha fazla bilgi mevcuttur.

[1] DONE Bit'i, referans verilen motor sürücüsünden geçerli verilerin okunduğunu ve CPU'ya iletildiğini gösterir. USS kütüphanesinin başka bir parametreyi hemen okuyabildiğini belirtmez. Parametre kanalı ilgili sürücü tarafından kullanılmak üzere serbest bırakılmadan önce, motor sürücüsüne boş bir parametre karakteristik değeri (PKD) talebi gönderilmeli ve ayrıca yordam tarafından onaylanmalıdır. Belirli motor sürücüsü için hemen bir USS_Read_Param veya USS_Write_Param FC'nin çağrılması, 0x818A hatasıyla sonuçlanır.

13.4.2 USS durum kodları

USS yordamının durum kodları, USS fonksiyonlarının STATUS çıkışında yayınlanır.

Tablo 13-50 STATUS kodları [1]

STATUS (W#16#....)	Açıklama
0000	Hata yok
8180	Sürücüden gelen yanıtın uzunluğu, sürücüden alınan karakterler ile aynı değil. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
8181	VALUE parametresi Word, Real veya Double Word veri tipine sahip değil.
8182	Kullanıcı, sözcük tipinde bir parametre değeri girdi ve sürücüden çift sözcük veya gerçek formatta yanıt aldı.
8183	Kullanıcı, bir çift sözcük veya gerçek tipinde parametre değeri girdi ve sürücüden sözcük formatında yanıt aldı.
8184	Sürücüden gelen yanıt telegramında yanlış bir sağlama toplamı vardı. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
8185	Geçersiz sürücü adresi (sürücüler için geçerli adres aralığı: 1 ila 16)
8186	Devir sayısı hedef değeri, referansı aralığın dışında (geçerli hız hedef değer aralığı: -%200 ila %200).
8187	Gönderilen talebe, yanlış sürücü numarası yanıt verdi. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
8188	Geçersiz PZD sözcük uzunluğu belirtildi (geçerli aralık = 2, 4, 6 veya 8 sözcük)
8189	Geçersiz Baud hızı belirtildi.
818A	Parametre talep kanalı, bu sürücü için olan başka bir talep tarafından kullanılıyor.
818B	Sürücü, taleplere ve yeniden denemelere yanıt vermedi. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.

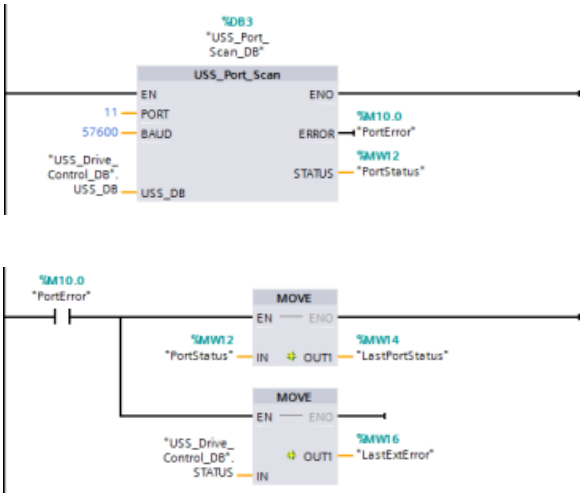
STATUS (W#16#....)	Açıklama
818C	Sürücü, bir parametre talebi için genişletilmiş bir hata yayınladı. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
818D	Sürücü, bir parametre talebi için bir "Geçersiz erişim" hatası yayınladı. Parametre erişiminin neden sınırlandırılmış olabileceği hakkında daha fazla bilgi için, sürücü işletme kitabına bakın.
818E	Sürücü başlatılmadı. USS_Drive_Control yordamı bu sürücü için en azından bir kez çağrılmadıysa, bu hata kodu USS_Read_Param veya USS_Write_Param'a yayınlanır. Bu işlem, USS_Drive_Control'ün ilk çevriminde-ki başlatmanın, bekleyen bir parametre okuma veya yazma talebinin üzerine yazmasını engeller, çünkü sürücü daha sonra yeni bir giriş olarak başlatılır. Bu hatayı düzeltmek için, bu sürücü için USS_Drive_Control yordamını çağırın.
80Ax-80Fx	USS kütüphanesi tarafından PtP iletişimi için çağrılan FB'ler tarafından geri döndürülen spesifik hatalar - Bu hata kodları, USS kütüphanesi tarafından değiştirilmez ve PtP yordamının açıklamalarında tanımlanır.

[1] Yukarıda sıralanan USS yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları (Sayfa 908) da hata geri döndürebilir.

Çeşitli STATUS kodları için daha fazla bilgi, USS_Drive_Control örnek DB'sinin "USS_Extended_Error" değişkeninde sağlanmaktadır. STATUS kodları onaltılık 8180, 8184, 8187 ve 818B için, USS_Extended_Error iletişim hatasının ortaya çıktığı sürücünün sürücü numarasını içerir. USS_Read_Param veya USS_Write_Param yordamı kullanıldığında, STATUS kodu onaltılık 818C için USS_Extended_Error sürücü tarafından yayınlanan bir sürücü hata kodu içerir.

Örnek: İletişim hataları mesajı

İletişim hataları (STATUS = 16#818B) yalnızca USS_Port_Scan yordamıyla bildirilir, USS_Drive_Control yordamıyla bildirilmez. Örnek: Ağ düzgün bir şekilde sonlandırılmamış, bir sürücü, RUN işletim durumuna geçebiliyor, ancak USS_Drive_Control yordamı tüm çıkış parametrelerinde 0 gösteriyor. Bu durumda, iletişim hatasını yalnızca USS_Port_Scan yordamı üzerinden saptayabilirsiniz. Bu hatanın yalnızca bir çevrim boyunca görülebilmesi nedeniyle burada ilgili edinim mantığını girmelisiniz. Bu, aşağıdaki örnekte açıklanmaktadır. Bu örnekte USS_Port_Scan yordamının hata Bit'i DOĞRU ise, markörlerin bellek alanına STATUS ve USS_Extended_Error değerleri konur. STATUS kodunun değeri onaltılık 8180, 8184, 8187 veya 818B ise, sürücü numarası USS_Extended_Error değişkenine konur.



Ağ 1 Port durumu "PortStatus" ve genişletilmiş hata kodu değerleri "USS_Drive_Control_DB".USS_Extended_Error yalnızca bir program çevrimi boyunca geçerli kalır. Bu değerler daha sonraki işleme için toplanır.

Ağ 2 "PortError" kontağı, "PortStatus" değerinin "LastPortStatus" içinde ve "USS_Drive_Control_DB".USS_Extended_Error değerinin ise "LastExtError" içinde saklanmasını tetikler.

Bir sürücünün dahili parametrelerine okuma ve yazma erişimi

USS sürücüleri, bir sürücünün dahili parametrelerine okuma ve yazma erişimlerini destekler. Bu fonksiyon, sürücüye desantral olarak kumanda edilmesine ve konfigürasyonuna olanak tanır. Sürücü parametrelerine erişimler, örneğin aralık dışında değer veya sürücünün güncel işletim türünde izin verilmeyen talepler gibi hatalardan dolayı başarısız olabilir. Sürücü, "USS_Extended_Error" değişkeninde yayınlanan bir hata kodu üretir. Bu hata kodu yalnızca USS_Read_Param veya USS_Write_Param yordamının son yürütülmesi için geçerlidir. STATUS code değeri onaltılık 818C ise, sürücünün hata kodu "USS_Extended_Error" değişkeninde saklanır. USS_Extended_Error'un hata kodunun değeri sürücü modeli göre değişir. Parametreler için okuma ve yazma fonksiyonlarının genişletilmiş hata kodlarına yönelik açıklama için sürücünün işletme kitabına başvurun.

13.4.3 USS sürücü sistemi için genel gereklilikler

USS sürücü ünitesi için aşağıdaki genel gereklilikler geçerlidir:

- Sürücüler için, 4 parametre karakteristik değer sözcüğünün kullanımı ayarlanmalıdır.
- Sürücüler; 2, 4, 6 veya 8 PZD sözcüğü için yapılandırılabilir.
- Sürücüdeki PZD sözcüklerinin sayısı, sürücünün USS_Drive_Control yordamının PZD_LEN girişine karşılık gelmelidir.
- Tüm sürücülerin Baud hızı, USS_Port_Scan yordamının giriş BAUD'una karşılık gelmelidir.
- Sürücü, uzaktan kumanda için ayarlanmalıdır.
- Sürücü COM bağlantısındaki frekans referansı, USS olarak ayarlanmalıdır.
- Sürücü adresi için, 1 ila 16 belirlenmelidir. Bu adres, sürücünün USS_Drive_Control bloğundaki DRIVE girişine karşılık gelmelidir.
- Sürücü referansının polaritesini kullanmak için sürücünün yön kumandası ayarlanmalıdır.
- RS485 ağı, usulüne uygun tamamlanmış olmalıdır.

13.4.4 Örnek: Genel USS sürücü bağlantısı ve kurulumu

MicroMaster sürücünün bağlanması

SIEMENS MicroMaster sürücülere ilişkin bu bilgiler örnek olarak verilmiştir. Diğer sürücülerde, kurulum yordamları için sürücü işletme kitabına başvurun.

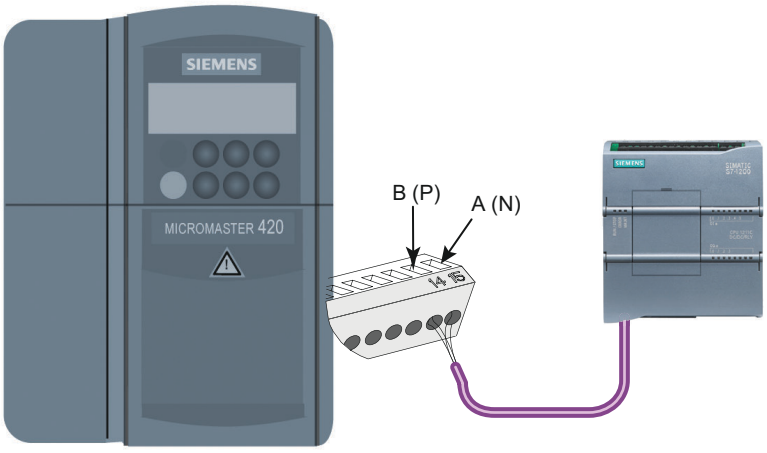
Bir MicroMaster Series 4 (MM4) sürücüyü bağlamak isterseniz, USS işletimi için RS485 kablosunun uçlarını her iki basmalı klemense takın. S7-1200, standart PROFIBUS kablosu ve soket bağlayıcıları kullanılarak MicroMaster sürücüye bağlanabilir.

⚠ DİKKAT

Aynı referans gerilimine sahip olmayan cihazları bağlarsanız, bu işlem, bağlantı hattında istenmeyen akımlara yol açabilir.

Bu istenmeyen akımlar, cihazlarda iletişim hatalarına veya maddi hasara neden olabilmektedir. Bir iletişim kablosu ile bağladığınız tüm cihazların, elektrik devresinde aynı referans kablosuna sahip olmasına veya istenmeyen akımların oluşmaması için elektriksel olarak yalıtılmış olmasına dikkat edin. Blendajlama, şasiye veya 9 kutuplu soket bağlayıcının pin 1'ine bağlanmalıdır. MicroMaster sürücünün 2-0V kablaj klemensi şasiye bağlanmalıdır.

RS485 kablosunun karşı ucundaki iki kabloyu MM4 sürücü klemens bloğuna takın. MM4 sürücüye kablo bağlantısı yaparken, klemens bloklarına erişmek için sürücü kapağını (kapaklarını) çıkarın. Spesifik sürücünüzdeki kapağın (kapakların) nasıl çıkarılacağına ilişkin ayrıntılı bir açıklama için MM4 sürücünün kullanıcı el kitabına başvurun.

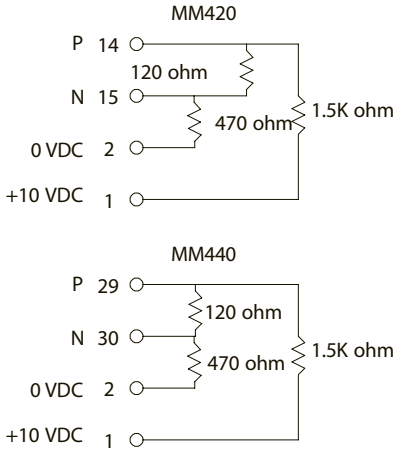


Klemens bloğundaki bağlantılar ardışık olarak numaralandırılmıştır. S7-1200 tarafında, bir PROFIBUS bağlantı fişi kullanın ve kablonun A klemensini sürücünün 15. klemensine (MM420 için) veya sürücünün 30. klemensine (MM440 için) takın. Kablo konektörünün B (P) A (N) klemensini, 14. klemense (MM420 için) veya 29. klemense (MM440 için) bağlayın.

S7-1200, ağdaki bir sonlandırma katılımcısı veya noktadan noktaya bağlantı ise, soket bağlayıcının A1 ve B1 klemenslerini (A2 ve B2 değil) kullanmalısınız, çünkü bu klemensler sonlandırma ayarlarını gerçekleştirir (örn. DP soket bağlayıcı 6ES7972-0BA40-0X40 ile).

⚠ DİKKAT
Cihazı güç kaynağına bağlamadan önce, sürücü kapaklarını dikkatli bir şekilde takın. Cihazı güç kaynağına bağlamadan önce, sürücü kapaklarını dikkatlice yeniden taktığınızdan emin olun.

Eğer sürücü ağda bir sonlandırma katılımcısı olarak yapılandırılırsa, sonlandırma dirençleri de uygun klemenslere kablolanmalıdır. Bu şekilde, MM4 sürücüye ait sonlandırma bağlantılarının örnekleri gösterilmektedir.



MM4 sürücünün kurulumu

Bir sürücüyü S7-1200'e bağlamadan önce, bu sürücünün aşağıdaki sistem parametrelerine sahip olduğuna dikkat edin. Sürücü klavyesi yardımıyla parametreleri ayarlarsınız:

1. Sürücüyü fabrika ayarlarına sıfırlayın (opsiyonel).	P0010 = 30 P0970 = 1
1. adımı atlarsanız, bu parametrelerin belirtilen değerlere ayarlandığından emin olmalısınız:	USS PZD uzunluğu = P2012 indeksi 0 = (2, 4, 6 veya 8) USS PKD uzunluğu = P2013 indeksi 0 = 4
2. Tüm parametreler için okuma/yazma erişimini etkinleştirin (uzman modu).	P0003 = 3
3. Sürücünüzün motor ayarlarını kontrol edin. Ayarlar, ilgili motora göre gerçekleştirilir. P304, P305, P307, P310 ve P311 parametrelerini ayarlayabilmek için önce P010 parametresini 1 (hızlı devreye alma modu) olarak ayarlamalısınız. Parametreleri ayarladıktan sonra, P010 parametresini 0 olarak ayarlayın. P304, P305, P307, P310 ve P311 parametreleri yalnızca hızlı işletime alma modunda değiştirilebilir.	P0304 = Nominal motor gerilimi (V) P0305 = Nominal motor akımı (A) P0307 = Nominal motor gücü (W) P0310 = Nominal motor frekansı (Hz) P0311 = Nominal motor devri
4. Yerel/uzak modu ayarlayın.	P0700 indeks 0 = 5
5. COM bağlantısının frekans referansını USS'e getirin.	P1000 indeks 0 = 5
6. İlk açılış süresi (opsiyonel) Motorun maksimum frekansa hızlanması için gereken saniye cinsinden süre.	P1120 = (0 ila 650,00)
7. Yavaşlayarak durma süresi (opsiyonel) Motorun yavaşlayarak tamamen durması için gereken saniye cinsinden süre.	P1121 = (0 ila 650,00)
8. Seri bağlantı referans frekansını ayarlayın:	P2000 = (1 ila 650 Hz)
9. USS normalizasyonunu ayarlayın:	P2009 indeks 0 = 0
10. RS485 seri arayüzünün Baud hızını ayarlayın:	P2010 indeks 0 = 4 (2400 Baud) 5 (4800 Baud) 6 (9600 Baud) 7 (19200 Baud 8 (38400 Baud) 9 (57600 Baud) 12 (115200 Baud)
11. Slave adresini girin. Her sürücü (maksimum 31) veriyolu üzerinden çalıştırılabilir.	P2011 indeks 0 = (0 ila 31)
12. Seri bağlantı için zaman aşımı değerini ayarlayın. Bu, gelen iki veri telegramı arasında izin verilen maksimum süre olur. Bu fonksiyon, bir iletişim arızası durumunda invertörü kapatır. Geçerli bir telegram alındıktan sonra, süre ölçülür. Belirtilen süre içinde başka veri telegramı alınmazsa, invertör kapanır ve F0070 hata kodunu görüntüler. Değeri sıfıra ayarlarsanız, kumanda ünitesi kapatılır.	P2014 indeks 0 = (0 ila 65,535 msn) 0 = zaman aşımı devre dışı bırakılır
13. Verileri, RAM'den EEPROM'a aktarın:	P0971 = 1 (aktarım başlat). EEPROM'daki parametre ayarlarındaki değişiklikleri saklayın.

13.5 Modbus iletişimi

13.5.1 Modbus RTU ve Modbus TCP üzerinden iletişime genel bakış

Modbus fonksiyon kodları

- Bir Modbus TCP istemcisi veya bir Modbus RTU Master'ı olarak işletilen bir CPU, bir desantral Modbus TCP sunucusu veya Modbus RTU Slave'indeki verileri ve G/Ç durumlarını okuyup yazabilir. Desantral veriler, program lojiğiniz tarafından okunabilir ve ardından işlenebilir.
- Modbus TCP sunucusu veya Modbus RTU Slave'i olarak işletilen bir CPU, üst seviyedeki bir cihazın CPU'nun belleğindeki verileri ve G/Ç durumlarını okumasına ve yazmasına olanak tanır. Bir Modbus TCP istemcisi veya Modbus RTU Master'ı, program lojiğiniz için kullanılabilir bir Slave/sunucu CPU'nun belleğine yeni değerler yazabilir.

⚠ İKAZ

Fiziksel ağ saldırıları nedeniyle oluşabilecek güvenlik risklerine engel olun

Bir saldırgan, ağlarınıza fiziksel olarak erişebilirse, muhtemelen verileri okuyabilir ve yazabilir.

Örn. PROFIBUS, PROFINET, AS-i veya başka bir G/Ç veriyolu, GET/PUT, T-Block ve iletişim modülü (CM) gibi diğer tüm iletişim türleri üzerinden G/Ç alışverişi hiçbir güvenlik fonksiyonuna sahip değildir. Bu tür iletişim türlerini, fiziksel erişimi sınırlama yoluyla korumanız gerekir. Bir saldırgan ağlarınıza bu iletişim türleri üzerinden fiziksel olarak erişebilirse, muhtemelen verileri okuyabilir ve yazabilir.

Bu iletişim türlerini korumazsanız, bu durum ölüme veya ağır yaralanmalara neden olabilir.

Güvenlikle ilgili bilgileri ve önerileri, Siemens'in "Service & Support" web sitesindeki "Operational Guidelines for Industrial Security" (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf) bölümümüzde bulabilirsiniz.

Tablo 13-51 Verileri okuma fonksiyonları: Desantral G/Ç ve program verilerinin okunması

Modbus fonksiyon kodu	Slave'e (sunucuya) yazma fonksiyonları - standart adresleme
01	Çıkış Bit'lerini oku: Talep başına 1 ila 2000 Bit
02	Giriş Bit'lerini oku: Talep başına 1 ila 2000 Bit
03	Tutma kaydını oku: 1 ila 125 talep başına Bit
04	Giriş sözcüklerini oku: 1 ila 125 talep başına Bit

Tablo 13-52 Verileri yazma fonksiyonları: Desantral G/Ç yazma ve program verilerini değiştirme

Modbus fonksiyon kodu	Slave'e (sunucuya) yazma fonksiyonları - standart adresleme
05	Bir çıkış Bit'ine yaz: Talep başına 1 Bit
06	Bir tutma kaydı yaz Talep başına 1 sözcük
15	Bir veya birden fazla çıkış Bit'i yaz: Talep başına 1 ila 1968 Bit
16	Bir veya daha fazla tutma kaydı yaz: 1 ila 123 talep başına Bit

- Modbus fonksiyon kodları 08 ve 11, Slave cihaz iletişimleri için teşhis bilgileri sağlar.
- Modbus fonksiyon kodu 0, tüm Slave'lere bir Broadcast mesajı gönderir (Slave yanıtı olmadan). İletişim bağlantı tabanlı olduğundan, Broadcast fonksiyonu Modbus TCP ile kullanılamaz.
- Modbus fonksiyon kodu 23, bir veya daha fazla tutma kaydı yazabilir ve okuyabilir: 1 ila 121/125 (yazma/okuma) talep başına sözcük Bu fonksiyon kodu, yalnızca Modbus TCP için kullanılabilir.

Tablo 13-53 Modbus ağındaki istasyon adresleri

İstasyon		**Adres**
TCP istasyonu	İstasyon adresi	IP adresi ve port numarası
RTU istasyonu	Standart istasyon adresi	1 ila 247
	Genişletilmiş istasyon adresi	1 ila 65535

Modbus bellek adresleri

Fiilen kullanılabilir Modbus bellek adresi sayısı; CPU varyantına, kullanılabilir çalışma belleğine ve diğer program verilerinin ne kadar CPU belleğini kullanıldığına göre değişir. Aşağıdaki tabloda, adres aralığının nominal değerini gösterilmektedir.

Tablo 13-54 Modbus bellek adresleri

İstasyon		**Adres aralığı**
TCP istasyonu	Standart bellekteki adres	10 K
RTU istasyonu	Standart bellekteki adres	10 K
	Genişletilmiş bellekteki adres	64 K

13.5.2 MODBUS TCP

13.5.2.1 Genel bakış

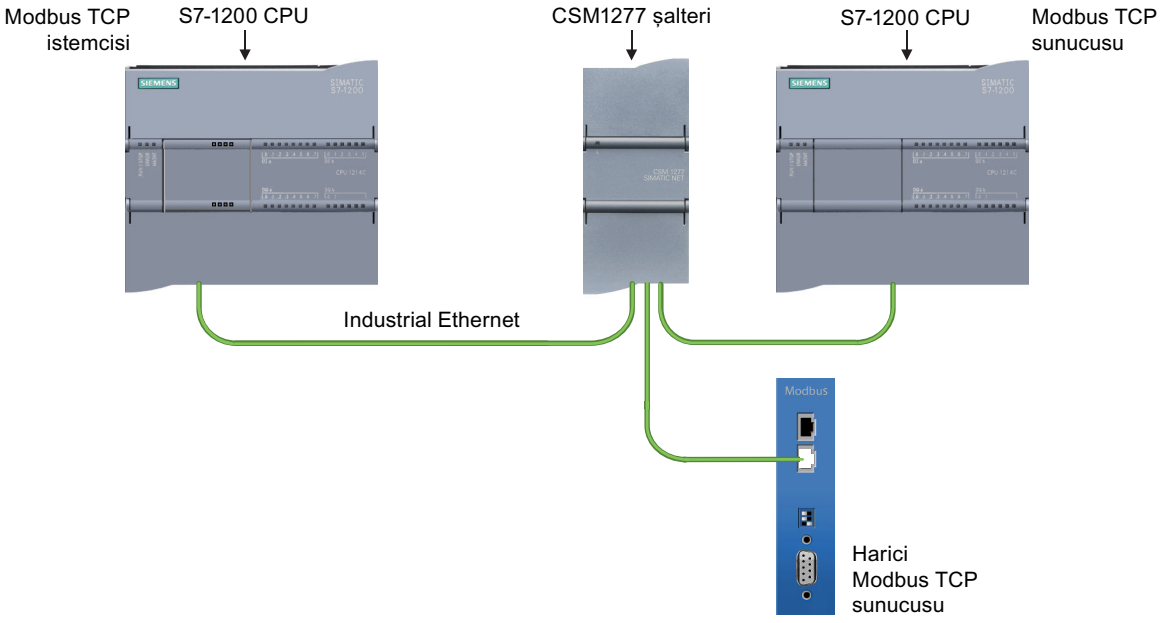
Modbus TCP (Transmission Control Protocol), ağ iletişimine yönelik bir standart protokoldür ve TCP/IP iletişimi için CPU üzerindeki PROFINET bağlantısını kullanır. İletişim için ek bir donanım modülüne gerek yoktur.

Modbus TCP, Modbus iletişim yolu olarak Açık Kullanıcı İletişimi (OUC) bağlantılarından yararlanır. STEP 7 ve CPU arasındaki bağlantının yanı sıra birden çok istemci/sunucu bağlantısı olabilir. Karma istemci ve sunucu bağlantıları, CPU tarafından izin verilen maksimum bağlantı sayısı kadar desteklenir.

Her MB_SERVER bağlantısı, benzersiz bir örnek DB ve IP portu numarası kullanmalıdır. IP portu başına yalnızca bir bağlantı desteklenir. Her bağlantı için, MB_SERVER (benzersiz örnek DB ve IP portu ile) ayrı ayrı çalıştırılmalıdır.

Bir Modbus TCP istemcisi (Master), istemci/sunucu bağlantısını DISCONNECT parametresi üzerinden kumanda etmelidir. Bir Modbus istemcisinin temel eylemleri aşağıda sıralanmaktadır.

1. Belirli bir IP adresi ve IP portu numarası olan bir sunucuya (Slave) bağlantı kurar
2. Bir Modbus mesajının aktarımını başlatır ve sunucudan yanıtları alır
3. Gerektiğinde bir diğer sunucuya olan bağlantıyı etkinleştirmek için istemci ve sunucu arasındaki bağlantı kesintisini tetikler.

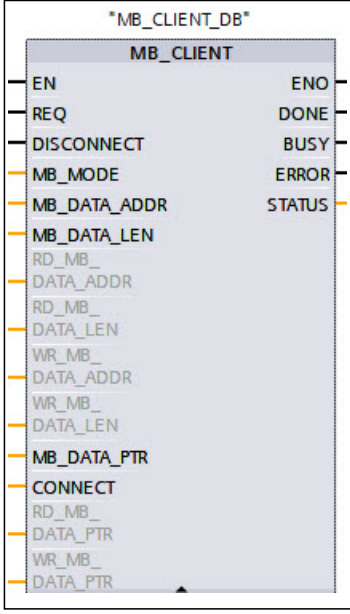


Modbus TCP, yordamlar ve sürümler

Modbus TCP yordamları (Sayfa 958), bir Switch aracılığıyla Modbus TCP istemcileri, Modbus TCP sunucuları ve yabancı Modbus TCP cihazları arasında iletişim sağlar. Daha eski S7-1200 programları, eski Modbus TCP yordamlarını (Sayfa 1069) kullanıyor olabilir.
STEP 7, Modbus TCP için yordamların çeşitli sürümlerini sunmaktadır. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

13.5.2.2 Modbus TCP yordamlar

Tablo 13-55 MB_CLIENT yordamı

LAD/FBD	SCL	Açıklama
	**"MB_CLIENT_DB"(REQ:=_bool_in_, DISCONNECT:=_bool_in_, MB_MODE:=_usint_in_, MB_DATA_ADDR:=_udint_in_, MB_DATA_LEN:=_uint_in_, RD_MB_DATA_ADDR:=_uint_in_, RD_MB_DATA_LEN:=_uint_in_, WR_MB_DATA_ADDR:=_uint_in_, WR_MB_DATA_LEN:=_uint_in_, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MB_DATA_PTR:=_variant_inout_, CONNECT:=_variant_inout_, RD_MB_DATA_PTR:=_variant_inout_, WR_MB_DATA_PTR: =_variant_inout_);**	MB_CLIENT, Modbus TCP istemcisi olarak S7-1200 CPU üzerindeki PROFINET portu aracılığıyla iletişim kurar. İletişim için ek bir donanım modülüne gerek yoktur. MB_CLIENT; bir istemci-sunucu bağlantısı kurabilir, bir Modbus fonksiyon talebi gönderebilir, bir yanıt alabilir ve bir Modbus TCP sunucusundan bağlantıyı kesmeye kumanda edebilir.

Tablo 13-56 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	In	Bool	YANLIŞ = Modbus iletişim talebi yok DOĞRU = Modbus TCP sunucusuyla iletişim talebi
DISCONNECT	IN	Bool	Programınız, bir Modbus sunucu cihazına bağlantının kurulmasına ve sonlandırılmasına kumanda etmek için DISCONNECT parametresini kullanabilir. DISCONNECT = 0 ise ve bağlantı mevcut değilse, MB_CLIENT atanan IP adresi ve port numarası ile bağlantı kurmaya çabalar. DISCONNECT = 1 ise ve bir bağlantı mevcutsa, bağlantıyı kesmeyi dener. Bu girişin etkinleştirildiği her zaman başka hiçbir işlem başlatılmaz.
MB_MODE	IN	USInt	Mod seçimi: Talep türünü atar (okuma, yazma veya teşhis). Daha fazla bilgiyi, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
MB_DATA_ADDR	IN	UDInt	Modül başlangıç adresi: MB_CLIENT tarafından erişilecek verinin başlangıç adresini atar. Geçerli adresler için aşağıdaki Modbus fonksiyonları tablosuna bakın.

Parametre ve veri tipi		Veri tipi	Açıklama
MB_DATA_LEN	IN	UInt	Modbus veri uzunluğu: Bu talep tarafından erişilecek Bit veya sözcük sayısını atar. Geçerli uzunluklar için aşağıdaki Modbus fonksiyonları tablosuna bakın.
MB_DATA_PTR	IN_OUT	Variant	İşaretçi, Modbus veri kaydı üzerinde: Kayıt, bir Modbus sunucusuna gönderilen veya bir Modbus sunucusundan alınan verileri tampona alır. İşaretçi, optimize edilmemiş bir global DB veya markörlerin bellek alanında bir adres atamalıdır.
CONNECT	IN_OUT	Variant	"TCON_IP_v4" sistem veri tipinde bağlantı parametrelerini içeren bir veri bloğu yapısına referanslama. Aşağıdaki veri tipleri de desteklenir: TCON_IP_V4_SEC, TCON_QDN ve TCON_QDN_SEC. Bkz. "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)".
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - İşlenen MB_CLIENT yordamı yok • 1 - MB_CLIENT yordamı işleniyor
ERROR	OUT	Bool	ERROR Bit'i, MB_CLIENT yürütmesi hatalı bir şekilde sona erdikten sonra bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

Modbus fonksiyonu 23

Açıklama

Modbus fonksiyonu 23 ile bir görevde aşağıdakiler yürütülür:

1. Veriler, CPU'dan Modbus sunucusuna aktarılır ve bir veya daha fazla tutma kaydına yazılır.
2. Veriler, Modbus sunucusunun bir veya daha fazla tutma kaydından okunur ve CPU'ya aktarılır.

"MB_CLIENT" yordamı, V6.0 sürümü yordamı olarak Modbus fonksiyonu 23'ü destekler.

Parametre

Modbus fonksiyonu 23 kullanılıyorsa, MB_MODE parametresi 123 değerine sahip olmalıdır.

MB_DATA_ADDR, MB_DATA_LEN ve MB_DATA_PTR parametreleri kullanılmaz ve değer olarak standart ayarlara sahip olmalıdır.

Modbus fonksiyonu 23 kullanılırken, aşağıdaki tabloda açıklanan altı yeni parametre kullanılır. Bu parametrelerin her biri, okuma görevine (RD_) veya yazma görevine (WR_) ait olduklarını göstermek için "RD_" veya "WR_" ile başlar. Standart olarak, bu parametreler gizlidir. Modbus fonksiyonu 23 kullanılırken, bu altı parametrenin tümü kullanılmalıdır. Başka bir Modbus fonksiyonu kullanırsanız, bu altı parametrenin değeri 0 olmalı veya boş olmalıdır. Aksi takdirde, STATUS değeri 16#818D geri döndürülür.

Tablo 13-57 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
RD_MB_DATA_ADDR	IN	UInt	Verilerin okunacağı uzak cihazdaki başlangıç adresi. İzin verilen değerler: 0 ila 65535
RD_MB_DATA_LEN	IN	UInt	Uzak cihazdan okunacak kayıt sayısı. İzin verilen değerler: 1 ila 125

Parametre ve veri tipi		Veri tipi	Açıklama
WR_MB_DATA_ADDR	IN	UInt	Uzak cihazdaki kendisinden itibaren veri yazılacak başlangıç adresi. İzin verilen değerler: 0 ila 65535
WR_MB_DATA_LEN	IN	UInt	Uzak cihaza yazılacak kayıt sayısı. İzin verilen değerler: 1 ila 121
RD_MB_DATA_PTR	IN_OUT	Variant	Modbus sunucusundan okunacak veriler için bir veri tamponuna işaretçi. Veri tipleri olarak MB_DATA_PTR ile aynı veri tiplerine izin verilir.
WR_MB_DATA_PTR	IN_OUT	Variant	Modbus sunucusuna yazılacak veriler için bir veri tamponuna işaretçi. Veri tipleri olarak MB_DATA_PTR ile aynı veri tiplerine izin verilir.

STATUS parametresi

STATUS değerlerinin anlamı, 16#8383, 8189, 818A, 818B genişletilmiştir. STATUS değeri 16#818D eklenir.

Proje güncelleme, yordam güncelleme

Mevcut bir projeyi MB_CLIENT yordamlarıyla güncellerseniz (örn. yordam sürümü V5.2), yordamın yeni sürümü programınızda otomatik olarak kullanılmayacaktır. Modbus fonksiyonu 23'ü kullanmak için yordam sürümünü manuel olarak güncellemeniz gerekir.

REQ parametresi

YANLIŞ = Modbus iletişim talebi yok
DOĞRU = Modbus TCP sunucusuyla iletişim talebi

Hiçbir MB_CLIENT örneği aktif değilse ve DISCONNECT = 0 parametresi REQ = 1 ise, yeni bir Modbus talebi başlatılır. Bağlantı henüz oluşturulmamışsa, yeni bir bağlantı kurulacaktır.

Mevcut talep tamamlanmadan önce aynı MB_CLIENT örneği DISCONNECT = 0 ve REQ = 1 ile tekrar çalıştırılırsa, bir sonraki Modbus aktarımı yürütülmeyecektir. Ancak mevcut talep tamamlandıktan sonra, MB_CLIENT'in REQ=1 ile yürütülmesi koşuluyla yeni bir talep işlenebilir.

Mevcut MB_CLIENT iletişim talebi tamamlandığında, DONE Bit'i bir çevrim boyunca DOĞRU'dur. DONE Bit'i, birden çok MB_CLIENT talebini sıralamak için bir zaman aralığı olarak kullanılabilir.

NOT

MB_CLIENT 'in işlenmesi sırasında giriş verilerinin tutarlılığı

Bir Modbus istemcisi bir Modbus işlemi başlattıktan sonra, tüm giriş durumları dahili olarak saklanır ve bir sonraki her çağrıda karşılaştırılır. Karşılaştırmanın amacı, belirli bir çağrının aktif istemci talebinin orijinal tetikleyicisi olup olmadığını belirlemektir. MB_CLIENT 'in birden çok çağrısı, paylaşılan bir örnek DB kullanılarak gerçekleştirilebilir.

Bir MB_CLIENT işleminin aktif olarak işlendiği süre boyunca girişlerin değişmemesi önemlidir. Bu kural çiğnenirse, MB_CLIENT aktif örneği belirleyemez.

Modbus iletişim fonksiyonunu seçmek için MB_MODE ve MB_DATA_ADDR parametrelerini kullanın.

MB_CLIENT yordamı, fonksiyon kodu girişi yerine bir MB_MODE girişi kullanır. MB_DATA_ADDR desantral verilerin Modbus başlangıç adresini atar.

MB_MODE ve MB_DATA_ADDR kombinasyonu, asıl Modbus mesajında kullanılan fonksiyon kodunu belirler. Aşağıdaki tabloda MB_MODE, MB_DATA_ADDR parametresi ve Modbus fonksiyonu arasındaki ilişki gösterilmektedir:

Tablo 13-58 Modbus fonksiyonları

MB_MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	MB_DATA_ADDR
0	01	1 ila 2000	Çıkış Bit'ini oku: Talep başına 1 ila 2000 Bit	1 ila 9999
101	01	1 ila 2000	Çıkış Bit'ini oku: Talep başına 1 ila 2000 Bit	00000 ila 65535
0	02	1 ila 2000	Giriş Bit'ini oku: Talep başına 1 ila 2000 Bit	10001 ila 19999
102	02	1 ila 2000	Giriş Bit'ini oku: Talep başına 1 ila 2000 Bit	00000 ila 65535
0	03	1 ila 125	Tutma kaydını oku: Talep başına 1 ila 125 sözcük	40001 ila 49999 veya 400001 ila 465535
103	03	1 ila 125	Tutma kaydını oku: Talep başına 1 ila 125 sözcük	00000 ila 65535
0	04	1 ila 125	Giriş sözcüklerini oku: Talep başına 1 ila 125 sözcük	30001 ila 39999
104	04	1 ila 125	Giriş sözcüklerini oku: Talep başına 1 ila 125 sözcük	00000 ila 65535
1	05	1	Bir çıkış Bit'i yaz: Talep başına bir Bit	1 ila 9999
105	05	1	Bir çıkış Bit'i yaz: Talep başına bir Bit	00000 ila 65535
1	06	1	Bir tutma kaydı yaz: Talep başına 1 sözcük	40001 ila 49999 veya 400001 ila 465535
106	06	1	Bir tutma kaydı yaz: Talep başına 1 sözcük	00000 ila 65535
1	15	2 ila 1968	Birden fazla çıkış Bit'i yaz: Talep başına 2 ila 1968 Bit	1 ila 9999
1	16	2 ila 123	Birden fazla tutma kaydı yaz: Talep başına 2 ila 123 sözcük	40001 ila 49999 veya 400001 ila 465535
2	15	1 ila 1968	Bir veya birden fazla çıkış Bit'i yaz: Talep başına 1 ila 1968 Bit	1 ila 9999
115	15	1 ila 1968	Bir veya birden fazla çıkış Bit'i yaz: Talep başına 1 ila 1968 Bit	00000 ila 65535
2	16	1 ila 123	Bir veya birden fazla tutma kaydı yaz: Talep başına 1 ila 123 sözcük	40001 ila 49999 veya 400001 ila 465535

MB_MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	MB_DATA_ADDR
116	16	1 ila 123	Bir veya birden fazla tutma kaydı yaz: Talep başına 1 ila 123 sözcük	00000 ila 65535
11	11	0	Sunucu iletişim durumu sözcüğünü ve olay sayacını oku. Durum sözcüğü meşgul olduğunu gösterir (0 = meşgul değil, 0xFFFF = meşgul). Olay sayacı, bir mesaj başarıyla işlendiğinde her defasında artırılır. MB_CLIENT'in hem MB_DATA_ADDR parametresi hem de MB_DATA_LEN parametresi bu fonksiyonda dikkate alınmaz.	
80	08	1	Teşhis kodu 0x0000 ile sunucu durumunu kontrol eder (kontrol döngüsü testi, sunucu talebin bir yankısını geri döndürür) Talep başına 1 sözcük	
81	08	1	Teşhis kodu 0x000A ile sunucu olay sayacını sıfırlar Talep başına 1 sözcük	
123	23	1 ila 121 (yazma) 1 ila 125 (okuma)	Bir görevde uzak cihazda tutma kayıtlarını yazar ve uzak cihazdaki tutma kayıtlarını okur. Not: Bu Modbus fonksiyonu, yordam sürümü V6.0'dan itibaren "MB_CLIENT" tarafından desteklenmektedir. RD_MB_DATA_ADDR, RD_MB_DATA_LEN, WR_MB_DATA_ADDR, WR_MB_DATA_LEN, RD_MB_DATA_PTR WR_MB_DATA_PTR parametreleri bu amaç doğrultusunda kullanılıyor.	
3 ila10,12 ila 79, 82 ila 100, 107 ila 114, 117 ila 255			Rezerve	

NOT

MB_DATA_PTR , bir Modbus TCP sunucusundan okunan veya sunucuya yazılan verileri saklamak için bir tampon atar

Veri tamponu, optimize edilmemiş bir global DB'de veya markör bellek alanındaki bir adreste olabilir.

Markörlerin bellek alanındaki bir tampon için bir ANY işaretçi formatını kullanın. Bu, P#"Bit adresi" "Veri tipi" "Uzunluk" formatıdır. Bir örnek: P#M1000.0 WORD 500.

MB_DATA_PTR parametresi tarafından bir iletişim tamponunun atanması

- MB_CLIENT'ın iletişim fonksiyonları:
  - Modbus sunucu adreslerinden (00001 ila 09999) 1 Bit'lik verileri okur ve yazar
  - Modbus sunucu adreslerinden (10001 ila 19999) 1 Bit'lik verileri okur
  - Modbus sunucu adreslerinden (30001 - 39999) ve (40001 - 49999) 16 Bit sözcük verilerini okur
  - Modbus sunucu adreslerine (40001 ila 49999) 16 Bit sözcük verilerini yazar
- Sözcük veya Bit boyutu verileri, DB veya markör alanında MB_DATA_PTR tarafından atanan tampona veya tampondan aktarılır.
- Bir DB, MB_DATA_PTR tarafından tampon olarak atanırsa, tüm DB veri elemanlarına veri tipleri atamanız gerekir.
  - 1 Bit'lik Bool veri tipi, bir Modbus Bit adresini temsil eder.
  - WORD, UInt ve Int gibi münferit 16 Bit'lik sözcüklere sahip veri tipleri, bir Modbus sözcük adresini temsil eder.
  - DWORD, DInt ve Real gibi 32 Bit çift sözcük veri tipleri, iki Modbus sözcük adresini temsil eder.
- Karmaşık DB elemanları MB_DATA_PTR tarafından atanabilir, örn.
  - Array'ler
  - Her elemanın benzersiz olduğu adlandırılmış yapılardır.
  - Her elemanın benzersiz bir adı ve 16 veya 32 Bit veri tipi olduğu adlandırılmış karmaşık yapılardır.
- MB_DATA_PTR için veri alanlarının aynı global veri bloğunda (veya markörlerin bellek alanında) olmaları zorunlu değildir. Modbus okuma işlemleri için bir veri bloğu, Modbus yazma işlemleri için başka bir veri bloğu veya her MB_CLIENT için bir veri bloğu oluşturabilirsiniz.

CONNECT parametresi, bir PROFINET bağlantısı oluşturmak için veriler atar

CONNECT parametresinde bu DB'ye referanslamadan önce, global bir veri bloğu kullanmanız ve gerekli bağlantı verilerini kaydetmelisiniz.

1. CONNECT verilerini saklamak için yeni bir global DB oluşturun veya mevcut bir global DB'yi kullanın. Birden çok TCON_IP_v4 veri yapısını bir DB'de saklayabilirsiniz. Her Modbus TCP istemci veya sunucu bağlantısı, bir TCON_IP_v4 veri yapısı kullanır. CONNECT parametresinde bağlantı verilerini referans alırsınız.
2. DB'ye ve bir statik değişkene anlamlı adlar verin. Örneğin veri bloğunu "Modbus bağlantıları" ve statik değişkeni "TCPaktiv_1" (Modbus TCP istemci bağlantısı 1 için) olarak adlandırabilirsiniz.
3. DB editöründe, "Veri tipi" sütunundaki "TCPactive_1" örneğinin statik değişkenine "TCON_IP_v4" sistem veri tipini atayın.
4. Aşağıdaki şekilde gösterildiği gibi bağlantı parametrelerini değiştirebilmeniz için TCON_IP_v4 yapısını genişletin.
5. Bir MB_CLIENT bağlantısı için TCON_IP_v4 yapısındaki verileri değiştirin.
6. MB_CLIENT'in CONNECT parametresi için DB yapı referansını girin. Bu örnekteki referans, "Modbus bağlantıları"dır.TCPaktiv_1.

Modbus connections

		Name	Data type	Start value	Comment
1		Static			
2		TCPactive_1	TCON_IP_v4		
3		InterfaceId	HW_ANY	64	HW-identifier of IE-interface submodule
4		ID	CONN_OUC	1	connection reference / identifier
5		ConnectionType	Byte	16#0B	type of connetion: 11=TCP/IP, 19=UDP (17=TC...
6		ActiveEstablished	Bool	True	active/passive connection establishment
7		RemoteAddress	IP_V4		remote IP address (IPv4)
8		ADDR	array [1..4] of Byte		IPv4 address
9		ADDR[1]	Byte	192	
10		ADDR[2]	Byte	168	
11		ADDR[3]	Byte	2	
12		ADDR[4]	Byte	241	
13		RemotePort	UInt	502	remote UDP/TCP port number
14		LocalPort	UInt	0	local UDP/TCP port number

Her MB_CLIENT bağlantısı için TCP_IP_V4 DB verilerinin değiştirilmesi

- **InterfaceID**: Cihaz konfigürasyonu penceresinde, CPU'nun PROFINET bağlantısının görseline tıklayın. Ardından "Genel" sekmesine tıklayın ve orada gösterilen donanım kodunu kullanın.
- **ID**: 1 ile 4095 arasında bir bağlantı ID'si girin. Modbus TCP iletişimi, açık kullanıcı iletişimi için temel yordamlar olan TCON, TDISCON, TSEND ve TRCV üzerinden gerçekleşir.
- **ConnectionType**: TCP/IP için standart 16#0B'yi (ondalık sayı = 11) kullanın.
- **ActiveEstablished**: Bu değer, 1 veya DOĞRU olmalıdır. MB_CLIENT Modbus iletişimini başlattığı için bağlantı aktiftir.
- **RemoteAddress**: Modbus TCP hedef sunucusunun IP adresini dört ADDR Array elemanına girin. Örneğin, yukarıdaki görseldeki gibi 192.168.2.241 girin.
- **RemotePort**: Standart değer, 502'dir. Bu numara, MB_CLIENT'in bağlanmaya ve iletişim kurmaya çalıştığı Modbus sunucusunun IP port numarasıdır. Bazı üçüncü taraf Modbus sunucuları, başka bir port numarası kullanmanızı gerektirebilir
- **LocalPort**: Bir MB_CLIENT bağlantısı için bu değer, 0 olmalıdır.

Birden çok istemci bağlantısı

Bir Modbus TCP istemcisi, PLC cihazı için izin verilen maksimum açık kullanıcı iletişim (Open User Communication) bağlantısı sayısına kadar eş zamanlı bağlantıları destekler. Modbus TCP istemcileri ve sunucuları da dahil olmak üzere bir PLC cihazı için toplam bağlantı sayısının, desteklenen maksimum açık kullanıcı iletişimi bağlantı sayısını aşmasına (Sayfa 556) izin verilmez.

Münferit eş zamanlı istemci bağlantıları aşağıdaki kurallara uymalıdır:

- Her MB_CLIENT bağlantısı benzersiz bir örnek DB kullanmalıdır.
- Her MB_CLIENT bağlantısı, bir sunucunun benzersiz bir IP adresini atamalıdır.
- Her MB_CLIENT bağlantısı benzersiz bir bağlantı ID'si atamalıdır.
- Sunucu konfigürasyonuna bağlı olarak benzersiz IP portu numaraları gerekebilir.

Her örnek DB için farklı bir bağlantı ID'si kullanılmalıdır. Özetle bunun anlamı, örnek DB ve bağlantı ID'sinin eşleşmiş ve her bağlantı için benzersiz olması gerektiğidir.

Tablo 13-59 MB_CLIENT'ten örnek veri bloğu: Kullanıcı tarafından erişilebilen statik değişkenler

Değişken	Veri tipi	Ön ayar	Açıklama
Blocked_Proc_Timeout	Real	3.0	Bloke edilen bir Modbus istemci örneğini AKTİF olarak kaldırmadan önce bekleme süresi (saniye olarak). Bu, bir istemci talebi gönderildiğinde ve ardından bu talep tamamen işlenmeden önce uygulama istemci fonksiyonunu gerçekleştirmeyi durdurduğunda ortaya çıkabilir. S7-1200'de maksimum değer, 55 saniyedir.
MB_Unit_ID	Sözcük	255	Modbus cihaz kodu: Bir Modbus TCP sunucusu, IP adresi aracılığıyla adreslenir. Dolayısıyla MB_UNIT_ID parametresi Modbus TCP adreslemesi için kullanılmaz. MB_UNIT_ID parametresi, Modbus RTU protokolündeki Slave adres alanına karşılık gelir. Bir Modbus TCP sunucusu, bir Modbus RTU protokolüne Gateway olarak kullanıldığında, Slave cihaz, seri ağda MB_UNIT_ID kullanılarak tanımlanabilir. Bu durumda, MB_UNIT_ID parametresi bu talebi, doğru Modbus RTU Slave adresine iletir. Bazı Modbus TCP cihazları, MB_UNIT_ID parametresinin sınırlı bir değer aralığında olmasını gerektirebilir.
RCV_TIMEOUT	Real	2,0	MB_CLIENT'in bir sunucunun bir talebe yanıt vermesini beklediği saniye cinsinden süre.
Bağlı	Bool	0	Atanan sunucuya bağlantının bağlı olup olmadığını belirtir: 1 = Bağlı, 0 = Bağlı değil

Tablo 13-60 MB_CLIENT protokol hatası

STATUS* (W#16#)	Yerel ve/veya uzak hata	MB_SERVER'ın yanıtında hata kodu (B#16#)	Açıklama
80C8	Yerel	-	Belirtilen zaman dilimi içinde sunucudan yanıt gelmedi. Modbus sunucusuna bağlantıyı kontrol edin. Yapılandırılan yeniden denemeler tamamlanana dek, bu hata bildirilmez. "MB_CLIENT" yordamı, belirtilen süre içinde orijinal olarak iletilen işlem ID'sine sahip bir yanıt almazsa (MB_TRANSACTION_ID statik değişkenine bakın), bu hata kodu geri döndürülür.
8380	Yerel	-	Alınan Modbus telegramı yanlış formatta veya çok az Bayt alındı.
8381	Kaldırıldı	01	Fonksiyon kodu desteklenmiyor.

STATUS* (W#16#)	Yerel ve/veya uzak hata	MB_SERVER'ın yanıtında hata kodu (B#16#)	Açıklama
8382	Yerel	-	• Telegram başlığındaki Modbus telegramının uzunluğu, alınan Bayt sayısına karşılık gelmiyor. • Bayt sayısı, gerçekte iletilen Bayt sayısına karşılık gelmiyor (yalnızca 1-4 fonksiyonlar). Örnek: Bu, "MB_CLIENT" tek sayılı sözcük istediğinde, ancak "MB_SERVER" her zaman çift sayılı sözcük gönderdiğinde geçerlidir. • Alınan telegramdaki başlangıç adresi kayıtlı başlangıç adresine karşılık gelmiyor (fonksiyon 5, 6, 15, 16). • Sözcük sayısı, fiilen aktarılan sözcük sayısına karşılık gelmiyor (fonksiyon 15 ve 16).
	Kaldırıldı	03	Alınan Modbus telegramında geçersiz uzunluk belirtimi. Sunucu tarafını kontrol edin.
8383	Yerel	-	• yordam sürümü < V6.0: Veri okuma veya yazma hatası veya MB_DATA_PTR (Sayfa 977) adres aralığının dışında erişim. • Yordam sürümü ≥ V6.0: Veri okuma veya yazma hatası veya MB_DATA_PTR, RD_MB_DATA_PTR veya adres aralığı dışında erişim.
	Kaldırıldı	02	Veri okuma veya yazma hatası veya sunucunun adres aralığının dışında erişim.
8384	Yerel	-	• Geçersiz istisna kodu alındı. • İstemci tarafından başlangıçta gönderilenden farklı bir veri değeri alındı (fonksiyon 5, 6 ve 8). • Geçersiz durum değeri alındı (fonksiyon 11)
	Kaldırıldı	03	5 fonksiyonu için veri değerinde hata
8385	Yerel	-	• Teşhis kodu desteklenmiyor. • İstemci tarafından başlangıçta gönderilenden farklı bir kısmi fonksiyon kodu alındı (fonksiyon 8).
	Kaldırıldı	03	Teşhis kodu desteklenmiyor.
8386	Yerel	-	Alınan fonksiyon kodu, ilk olarak gönderilenle eşleşmiyor.
8387	Yerel	-	Sunucudan alınan Modbus TCP telegramının protokol ID'si "0" değildir.
8388	Yerel	-	Modbus sunucusu talep edilenden farklı bir veri uzunluğu gönderdi. Bu hata yalnızca Modbus fonksiyonları 5, 6, 15 veya 16 kullanılırken oluşur.
* Durum kodları, program editöründe tamsayı veya onaltılık değerler olarak görüntülenebilir. Görüntüleme formatlarının değiştirilmesiyle ilgili bilgiler için, bkz. "Ayrıca bkz".			

Tablo 13-61 MB_CLIENT yürütmesinin koşul kodları [1]

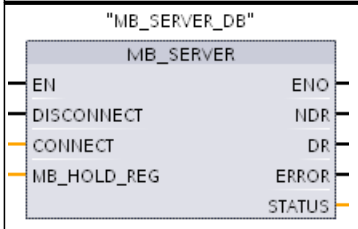
STATUS (W#16#)	MB_CLIENT parametre hatası
7001	MB_CLIENT, bir Modbus sunucusunun atanan TCP portunda bir bağlantı kurma veya sonlandırma talebine yanıt vermesini bekliyor. Bu kod, yalnızca bir bağlantı kurmanın veya sonlandırmanın ilk kez yürütülmesi için yayınlanır.
7002	MB_CLIENT, bir Modbus sunucusunun atanan TCP portunda bir bağlantı kurma veya sonlandırma talebine yanıt vermesini bekliyor. Bu, bir bağlantı kurulumunun veya sonlandırılmasının tamamlanmasını beklerken sonraki tüm yürütmeler için yayınlanır.

[1] Yukarıda listelenen MB_CLIENT hatalarının yanı sıra iletişim için temel T blok yordamları (TCON, TDISCON, TSEND ve TRCV) tarafından da hatalar yayınlanır.

STATUS (W#16#)	MB_CLIENT parametre hatası
7003	Bir bağlantı başarıyla sonlandırıldı (yalnızca bir PLC çevrimi için geçerlidir).
80C8	Sunucu belirlenen süre içinde yanıt vermedi. MB_CLIENT, ayrılan süre içinde orijinal olarak gönderilen işlem ID'si aracılığıyla bir yanıt almalıdır, aksi takdirde bu hata yayınlanır. Modbus sunucu cihazına bağlantıyı kontrol edin. Bu hata yalnızca yeniden denemeler (varsa) gerçekleştirildikten sonra verilir.
8188	MB_MODE parametresi geçersiz bir değere sahip.
8189	• yordam sürümü < V6.0: MB_DATA_ADDR parametresinde verilerin geçersiz adreslenmesi. • Yordam sürümü ≥ V6.0: MB_DATA_ADDR, RD_MB_DATA_ADDR veya WR_MB_DATA_ADDR parametresinde verilerin geçersiz adreslenmesi.
818A	• yordam sürümü < V6.0: MB_DATA_LEN parametresinde geçersiz veri tipi. • Yordam sürümü ≥ V6.0: MB_DATA_LEN, RD_MB_DATA_LEN veya WR_MB_DATA_LEN parametresinde geçersiz veri tipi.
818B	DATA_PTR alanının üzerinde geçersiz işaretçi. Bu, MB_DATA_ADDRESS + MB_DATA_LEN kombinasyonu olabilir.
818C	DATA_PTR işaretçisi, optimize edilmemiş bir DB aralığını işaret ediyor (bu, optimize edilmemiş bir DB aralığı veya markörlerin bellek alanındaki bir alan olmalıdır)
818D	Bir veya daha fazla parametre, standart değerine sahip değil, ancak belirtilen Modbus fonksiyon ile kullanılmıyor. Örnek: MB_MODE 123 ise, MB_DATA_ADDR ve MB_DATA_LEN 0 olmalı ve MB_DATA_PTR boş olmalıdır. MB_MODE, 123'ten farklı bir değere sahipse, "RD_" veya "WR_" ile başlayan tüm parametreler 0 değerine sahip olmalı veya boş olmalıdır.
8200	Port, mevcut bir Modbus talebini işlemekle meşgul.
8380	Alınan Modbus çerçevesi yanlış veya çok az Bayt alındı.
8387	Atanan bağlantı ID parametresi, önceki talepler için kullanılan ID'den farklı. MB_CLIENT'in her örnek DB'sinde yalnızca bir bağlantı ID'si kullanılabilir. Bir sunucudan alınan Modbus TCP protokol ID'si 0 olmadığında da bu kod bir dahili hata olarak yayınlanır.
8388	Bir Modbus sunucusu, talep edilen miktardan farklı bir miktarda veri geri döndürdü. Bu kod sadece Modbus fonksiyonları 15 ve 16 için geçerlidir.

[1] Yukarıda listelenen MB_CLIENT hatalarının yanı sıra iletişim için temel T blok yordamları (TCON, TDISCON, TSEND ve TRCV) tarafından da hatalar yayınlanır.

Tablo 13-62 MB_SERVER yordamı

LAD/FBD	SCL	Açıklama
	**"MB_SERVER_DB"(** **DISCONNECT:=_bool_in_,** **CONNECT:=_variant_in_,** **NDR=>_bool_out_,** **DR=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **MB_HOLD_REG:=_variant_inout_);**	MB_SERVER, S7-1200 CPU üzerindeki PROFINET portu aracılığıyla bir Modbus TCP sunucusu olarak iletişim kurar. İletişim için ek bir donanım modülüne gerek yoktur. MB_SERVER, bir Modbus TCP istemcisi ile bağlantı kurma talebini kabul edebilir, bir Modbus fonksiyon talebi alabilir ve bir yanıt mesajı gönderebilir.

Tablo 13-63 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
DISCONNECT	IN	Bool	MB_SERVER, bir partner cihazla "pasif" bir bağlantı kurmaya çalışıyor. Bunun anlamı, sunucunun talep gönderen IP adreslerinden TCP bağlantı talepleri olup olmadığını pasif olarak izlemekte olduğudur. DISCONNECT = 0 ise ve bağlantı yoksa, pasif bir bağlantı kurulabilir. DISCONNECT = 1 ise ve bir bağlantı varsa, bağlantıyı kesmek için bir girişimde bulunulur. Bu parametre, programınızın bir bağlantının ne zaman kabul edildiğini kontrol etmesini sağlar. Bu girişin etkinleştirildiği her zaman başka hiçbir işlem başlatılmaz.
CONNECT	IN	Variant	"TCON_IP_v4" sistem veri tipinde bağlantı parametrelerini içeren bir veri bloğu yapısına referanslama. Aşağıdaki veri tipleri de desteklenir: TCON_IP_V4, TCON_QDN ve TCON_QDN_SEC. Bkz. "PROFINET bağlantısı için kullanılan parametreler (Sayfa 588)".
MB_HOLD_REG	IN_OUT	Variant	İşaretçi, MB_SERVER'ın Modbus tutma kaydı üzerinde: Tutma kaydı, optimize edilmemiş bir global DB veya markör bellek alanındaki bir adres olmalıdır. Bu bellek alanı, Modbus kayıt fonksiyonları 3 (okuma), 6 (yazma), 16 (yazma) ve 23 (okuma/yazma) kullanılarak bir Modbus istemcisi tarafından erişilebilen verileri saklamak için kullanılır.
NDR	OUT	Bool	Yeni veriler hazır: 0 = Yeni veri yok, 1 = Yeni verinin bir Modbus istemcisinden yazıldığını belirtir
DR	OUT	Bool	Veriler oku: 0 = Yeni veri yok, 1 = Verilerin bir Modbus istemcisi tarafından okunduğunu belirtir
ERROR	OUT	Bool	ERROR Bit'i, MB_SERVER yürütmesi hatalı bir şekilde sona erdikten sonra bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

CONNECT parametresi, bir PROFINET bağlantısı oluşturmak için veriler atar

CONNECT parametresinde bu DB'ye referanslamadan önce, global bir veri bloğu kullanmanız ve gerekli bağlantı verilerini kaydetmelisiniz.

1. CONNECT verilerini saklamak için yeni bir global DB oluşturun veya mevcut bir global DB'yi kullanın. Birden çok TCON_IP_v4 veri yapısını bir DB'de saklayabilirsiniz. Her Modbus TCP istemci veya sunucu bağlantısı, bir TCON_IP_v4 veri yapısı kullanır. CONNECT parametresinde bağlantı verilerini referans alırsınız.
2. DB'ye ve bir statik değişkene anlamlı adlar verin. Örneğin veri bloğunu "Modbus bağlantıları" ve statik değişkeni "TCPpassive_1" (Modbus TCP sunucu bağlantısı 1 için) olarak adlandırın.
3. DB editöründe, "Veri tipi" sütunundaki "TCPactive_1" örneğinin statik değişkenine "TCON_IP_v4" sistem veri tipini atayın.
4. Aşağıdaki şekilde gösterildiği gibi bağlantı parametrelerini değiştirebilmeniz için TCON_IP_v4 yapısını genişletin.
5. Bir MB_SERVER bağlantısı için TCON_IP_v4 yapısındaki verileri değiştirin.
6. MB_SERVER'ın CONNECT parametresi için DB yapı referansını girin. Bu örnekteki referans, "Modbus bağlantıları"dır.TCPaktiv_1.

Modbus connections

	Name	Data type	Start value	Comment
1	Static			
2	TCPpassive_1	TCON_IP_v4		
3	InterfaceId	HW_ANY	64	HW-identifier of IE-interface submodule
4	ID	CONN_OUC	1	connection reference / identifier
5	ConnectionType	Byte	16#0B	type of connetion: 11=TCP/IP, 19=UDP (17=TC...
6	ActiveEstablished	Bool	False	active/passive connection establishment
7	RemoteAddress	IP_V4		remote IP address (IPv4)
8	ADDR	array [1..4] of Byte		IPv4 address
9	ADDR[1]	Byte	192	
10	ADDR[2]	Byte	168	
11	ADDR[3]	Byte	2	
12	ADDR[4]	Byte	241	
13	RemotePort	UInt	0	remote UDP/TCP port number
14	LocalPort	UInt	502	local UDP/TCP port number

Her MB_SERVER bağlantısı için TCP_IP_V4 DB verilerinin değiştirilmesi

- **InterfaceID**: Cihaz konfigürasyonu penceresinde, CPU'nun PROFINET bağlantısının görseline tıklayın. Ardından "Genel" sekmesine tıklayın ve orada gösterilen donanım kodunu kullanın.
- **ID**: Bu bağlantı için 1 ile 4095 arasında benzersiz bir sayı girin. Modbus TCP iletişimi, açık kullanıcı iletişimi için temel yordamlar olan TCON, TDISCON, TSEND ve TRCV üzerinden gerçekleşir. Aynı anda sekiz adede kadar açık kullanıcı iletişimi bağlantısına izin verilir.
- **ConnectionType**: TCP/IP için standart 16#0B'yi kullanın (ondalık değer = 11).
- **ActiveEstablished**: Bu değer, 0 veya YANLIŞ olmalıdır. Bağlantı aktif değil, çünkü MB_SERVER bir Modbus istemcisinden bir iletişim talebi bekliyor.
- **RemoteAddress**: İki opsiyon var:
  - 0.0.0.0 kullanın ve MB_CLIENT, herhangi bir TCP istemcisinden gelen bir Modbus talebine yanıt verecektir.
  - Hedef Modbus TCP istemcisinin IP adresini girin; MB_CLIENT yalnızca o istemcinin IP adresinden gelen talebe yanıt verecektir. Örneğin, yukarıdaki görseldeki gibi 192.168.2.241 girin.

- **RemotePort**: Bir MB_SERVER bağlantısında bu değer 0 olmalıdır.
- **LocalPort**: Standart değer, 502'dir. Bu numara, MB_SERVER'ın bağlanmaya ve iletişim kurmaya çalıştığı Modbus istemcisinin IP port numarasıdır. Bazı üçüncü taraf Modbus istemcileri, farklı bir portu numarası gerektirir.

Modbus ve proses görünümü adresleri

MB_SERVER'da, gelen Modbus fonksiyon kodları (1, 2, 4, 5 ve 15), giriş ve çıkışların proses görünümündeki Bit'leri ve sözcükleri doğrudan okuyabilir ve yazabilir. Veri aktarımı için kullanılan fonksiyon kodlarıyla (3, 6 ve 16), MB_HOLD_REG parametresi bir Bayt'tan büyük bir veri tipi olarak tanımlanmalıdır. Aşağıdaki tabloda, Modbus adreslerinin CPU'daki proses görünümüne atanması gösterilmektedir.

Tablo 13-64 Modbus adreslerinin proses görünümüne atanması

Modbus fonksiyonları						**S7-1200**	
Kodlar	Fonksiyon	Veri alanı	Adres aralığı			Veri alanı	CPU adresi
01	Bit'leri oku	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
02	Bit'leri oku	Giriş	10001	-	18192	Girişlerin proses görünümü	E0.0 ila E1023.7
04	Sözcükleri oku	Giriş	30001	-	30512	Girişlerin proses görünümü	EW0 ila EW1022
05	Bit yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
15	Bit'leri yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7

Gelen Modbus mesaj fonksiyon kodları (3, 6 ve 16), bir markör veya veri bloğu olabilen bir Modbus tutma kaydındaki sözcükleri okur veya yazar. Tutma kaydı türü, MB_HOLD_REG parametresi tarafından belirlenir.

NOT

MB_HOLD_REG parametresinin atanması

Sözcük, tamsayı (16 Bit), geniş karakter, işaretsiz tamsayı (16 Bit), Bayt, kısa tamsayı, işaretsiz kısa tamsayı, karakter, çift sözcük, tamsayı (32 Bit), işaretsiz tamsayı (32 Bit) Array'leri olarak saklanan Modbus tutma kayıtları veya gerçek sayılar herhangi bir bellek alanında saklanabilir.

Yapı olarak tanımladığınız Modbus tutma kayıtları, optimize edilmemiş DB'lerde saklanmalıdır

Markörlerin bellek alanındaki bir Modbus tutma kaydı için Any işaretçisi formatını kullanın. Bu, P#"Bit adresi" "Veri tipi" "Uzunluk" formatıdır. Örnek: P#M1000.0 WORD 500.

Aşağıdaki tabloda, Modbus adreslerinin Modbus fonksiyon kodları 03 (sözcük oku), 06 (sözcük yaz) ve 16 (sözcük yaz) için kullanılan tutma kaydına nasıl eşleneceğine ilişkin örnekler gösterilmektedir. DB adreslerinin gerçek üst sınırı, ilgili CPU modelinin maksimum çalışma belleği ve maksimum M belleği tarafından belirlenir.

Tablo 13-65 Modbus adreslerinin CPU belleğindeki adreslerle nasıl atandığına ilişkin örnekler

Modbus adresi	**MB_HOLD_REG parametrelerine örnekler**		
	P#M100.0 Word 5	**P#DB10.DBx0.0 Word 5**	**"Reçete".Bileşim maddesi**
40001	MW100	DB10.DBW0	"Reçete".Bileşim maddesi[1]
40002	MW102	DB10.DBW2	"Reçete".Bileşim maddesi[2]

Modbus adresi	MB_HOLD_REG parametrelerine örnekler		
	P#M100.0 Word 5	P#DB10.DBx0.0 Word 5	"Reçete".Bileşim maddesi
40003	MW104	DB10.DBW4	"Reçete".Bileşim maddesi[3]
40004	MW106	DB10.DBW6	"Reçete".Bileşim maddesi[4]
40005	MW108	DB10.DBW8	"Reçete".Bileşim maddesi[5]

Modbus uygulama protokolü başlığı

Modbus Uygulama Protokolü başlığı, her Modbus TCP mesajının ilk yedi Bayt'ını içerir. Bu başlık, işlem kodu, protokol kodu, uzunluğu ve cihaz kodunu içerir. MB_SERVER yordamının yanıt mesajı, Modbus talep mesajında alınan aynı işlem kodu, protokol kodu ve cihaz kodu değerlerini içerir. Uzunluk alanı MB_SERVER yordamı ile hesaplanır.

Birden çok sunucu bağlantısı

Birden çok sunucu bağlantısı kurulabilir. Tek bir PLC cihazı, aynı anda birden fazla Modbus TCP istemcisine bağlanabilir.

Bir Modbus TCP sunucusu, PLC cihazı için izin verilen maksimum açık kullanıcı iletişim (Open User Communication) bağlantısı sayısına kadar eş zamanlı bağlantıları destekler. Modbus TCP istemcileri ve Modbus TCP sunucuları dahil olmak üzere bir PLC cihazı için toplam bağlantı sayısı, desteklenen maksimum açık kullanıcı iletişimi bağlantısı sayısını geçmemelidir. Modbus TCP bağlantıları, istemci ve sunucu bağlantıları tarafından ortaklaşa kullanılabilir.

Münferit eş zamanlı sunucu bağlantıları aşağıdaki kurallara uymalıdır:

- Her MB_SERVER bağlantısı, benzersiz bir örnek DB kullanmalıdır.
- Her MB_SERVER bağlantısı, benzersiz bir IP portu numarası atamalıdır. Port başına yalnızca bir bağlantı desteklenir.
- Her MB_SERVER bağlantısı, benzersiz bir bağlantı ID'si atamalıdır.
- MB_SERVER, her bağlantı için ayrı olarak çağrılmalıdır (karşılık gelen örnek DB ile birlikte).

Bağlantı ID'si, her bir bağlantı için benzersiz olmalıdır. Her bir örnek DB için farklı bir bağlantı ID'si kullanılmalıdır. Örnek DB ve bağlantı ID'si eşleştirildi ve her bağlantı için benzersiz olmalıdır.

Tablo 13-66 Modbus teşhis fonksiyon kodları

MB_SERVER'ın Modbus teşhis fonksiyonu		
Kodlar	Kısmi fonksiyon	Açıklama
08	0x0000	Yankı testi sorgu verilerini yayınlar: MB_SERVER yordamı, alınan bir veri sözcüğünün yankısını bir Modbus istemcisine geri döndürür.
08	0x000A	İletişim olayı sayacını sil: MB_SERVER yordamı, Modbus fonksiyonu 11 için kullanılan iletişim olay sayacını siler.
11		İletişim olayı sayacını çağır: MB_SERVER yordamı, Modbus sunucusuna gönderilen başarılı Modbus okuma ve yazma taleplerinin sayısını saymak için dahili bir iletişim olay sayacı kullanır. Fonksiyon 8, fonksiyon 11 ve iletişim hatasına neden olan tüm talepler için sayaç ileriye doğru sayılmaz. Aynı anda yalnızca bir istemci-sunucu bağlantısı olduğundan, Broadcast fonksiyonu Modbus TCP ile kullanılamaz.

MB_SERVER yordamının veri bloğundaki (DB) değişkenler

Bu tabloda, programınızda kullanılabilen MB_SERVER örnek veri bloğundaki genel statik değişkenler gösterilmektedir.

Tablo 13-67 MB_SERVER'in genel statik değişkenleri

Değişken	Veri tipi	Ön ayar	Açıklama
HR_Start_Offset	Word	0	Modbus tutma kaydının başlangıç adresini atar
Request_Count	Word	0	Bu sunucu tarafından alınan tüm taleplerin sayısı
Server_Message_Count	Word	0	Belirli sunucu için alınan taleplerin sayısı
Xmt_Rcv_Count	Word	0	Bir hata oluşan gönderimlerin veya veri alımlarının sayısı. Geçersiz bir Modbus mesajı olan bir mesaj alındığında da sayma değeri arttırılır.
Exception_Count	Word	0	İade edilen bir istisna gerektiren Modbus'a özgü hatalar geri döndürüldü
Success_Count	Word	0	Bu spesifik sunucu için protokol hatası olmaksızın alınan taleplerin sayısı.
Connected	Bool	0	Atanan istemciye bağlantının bağlı olup olmadığını belirtir: 1 = Bağlı, 0 = Bağlı değil
QB_Start	UInt	0	CPU'nun yazabileceği çıkış Bayt'larının başlangıç adresi (AB0 ila AB65535)
QB_Count	UInt	65535	Bir uzak cihazın yazabileceği Bayt sayısı. QB_Count = 0 ise, bir uzak cihaz çıkışlara yazamaz. Örnek: Yalnızca AB10 ila AB17 yazılabilir olacaksa, QB_Start = 10 ve QB_Count = 8.
QB_Read_Start	UInt	0	CPU'nun okuyabileceği çıkış Bayt'larının başlangıç adresi (AB0 ila AB65535)
QB_Read_Count	UInt	65535	Uzak bir cihazın okuyabileceği çıkış Bayt'ı sayısı. QB_Count = 0 ise, bir uzak cihaz çıkışlardan okuyamaz. Örnek: Yalnızca AB10 ila AB17 okunabilirse, QB_Start = 10 ve QB_Count = 8.
IB_Read_Start	UInt	0	CPU'nun okuyabileceği giriş Bayt'larının başlangıç adresi (EB0 ila EB65535)
IB_Read_Count	UInt	65535	Uzak bir cihazın okuyabileceği giriş Bayt'ı sayısı. IB_Count = 0 ise, bir uzak cihaz girişlerden okuyamaz. Örnek: Yalnızca EB10 ila EB17 okunabilirse, IB_Start = 10 ve IB_Count = 8.
NDR_immediate	Bool	YANLIŞ	NDR parametresiyle aynı anlama gelir (yeni veriler hazır). MB_SERVER, bir Modbus TCP yazma talebini işleyen aynı çağrı ile "NDR_immediate"i günceller.
DR_immediate	Bool	YANLIŞ	DR parametresiyle aynı anlama gelir (Verileri oku). MB_SERVER, bir Modbus TCP yazma talebini işleyen aynı çağrı ile "DR_immediate"i günceller.

Programınız, kumanda eden Modbus sunucu işlemlerine ve aşağıdaki değişkenlere veri yazabilir:

- HR_Start_Offset
- QB_Start
- QB_Count
- QB_Read_Start
- QB_Read_Count
- IB_Read_Start
- IB_Read_Count

Aşağıdaki sürüm gereklilikleri, MB_SERVER yordamının veri bloğundaki (DB) değişkenlerin kullanılabilirliği için geçerlidir.

Tablo 13-68 MB_SERVER yordamının veri bloğundaki (DB) değişkenlerin kullanılabilirliği için sürüm gereklilikleri: Yordam, TIA Portal ve S7-1200 CPU

MB_SERVER yordamının sürümü	TIA Portal sürümü	S7-1200 CPU Firmware'inin (FW) sürümü	Veri bloğu değişkenleri
4.2	V14 SP1	CPU FW V4.0 veya üstü	QB_Start
			QB_Count
5.0 veya üstü	V15 veya üstü	CPU FW V4.2 veya üstü	QB_Start
			QB_Count
			QB_Read_Start
			QB_Read_Count
			IB_Read_Start
			IB_Read_Count
			NDR_immediate
			DR_immediate

HR_Start_Offset

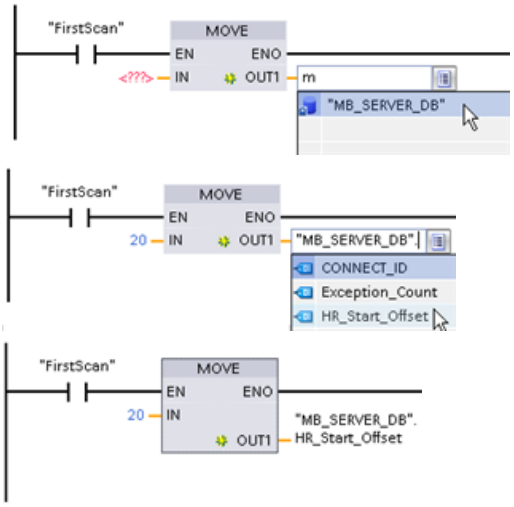
Modbus tutma kaydı adresleri 40001'den başlar. Bu adresler, hedef sistem belleğindeki tutma kaydının başlangıç adresine karşılık gelir. Ancak Modbus tutma kaydı için 40001 dışında bir başlangıç adresi yapılandırmak isterseniz, HR_Start_Offset değişkenini kullanabilirsiniz. Örneğin, MW100'de başlayan ve 100 sözcük uzunluğunda olan bir tutma kaydı yapılandırabilirsiniz. Tutma kaydı başlangıç adresini, 20'lik bir ofset ile 40001 yerine 40021 olarak belirtirsiniz. Bu durumda 40021'in altındaki ve 40119'un üzerindeki her adres, adresleme hatasına yol açacaktır.

Tablo 13-69 Modbus tutma kaydının adresleme örneği

HR_Start_Offset	Adres	Minimum	Maksimum
0	Modbus adresi (sözcük)	40001	40099
	S7-1200 adresi	MW100	MW298
20	Modbus adresi (sözcük)	40021	40119
	S7-1200 adresi	MW100	MW298

HR_Start_Offset, MB-SERVER'ın örnek veri bloğundaki Modbus tutma kaydının başlangıç adresini atayan sözcük verisidir. MB_SERVER'ı programınıza eklenmesinin ardından, bu açık statik değişkeni parametre açılır listesinden seçebilirsiniz.

Örneğin bir LAD ağına MB_SERVER eklediyseniz, bir önceki bir ağa gidip, HR_Start_Offset atayabilirsiniz. MB_SERVER yürütülmeden önce başlangıç adresi atanmalıdır.



Modbus sunucusu için bir değişken girilmesi Standart DB adı üzerinden:

1. İmleci parametre alanına getirin ve m karakterini girin.
2. DB adlarının açılır listesinden "MB_SERVER_DB" öğesini seçin.
3. DB değişkenleri açılır listesinden "MB_SERVER_DB.HR_Start_Offset" öğesini seçin.

Modbus adreslerine doğrudan erişim yerine veri bloklarındaki (DB) veri alanlarına erişim

DB'lerdeki veri alanlarına erişebilirsiniz. Global DB'nin "Öznitelikler" özellikler sayfasında, "Yalnızca yükleme belleğinde sakla" ve "Optimize edilmiş modül erişimi" opsiyon kutularının seçimi kaldırılmalıdır.

Bir Modbus talebi geldiğinde ve karşılık gelen fonksiyon kodunun Modbus veri tipi için herhangi bir veri alanı ayarlanmadığında, MB_SERVER yordamı bir talebi, bu yordamın önceki sürümlerinde olduğu gibi işler: Proses görünümlerine ve tutma kayıtlarına doğrudan erişebilirsiniz.

Fonksiyon kodunun Modbus veri tipi için bir veri alanı belirtilmişse, MB_SERVER yordamı bu veri alanından okur veya bu veri alanına yazar. Okunması veya yazılması görevin türüne göre değişir.

NOT

Bir veri alanı yapılandırıldığında, MB_SERVER yordamı, veri alanının veri tipine karşılık gelen örnek veri bloğundaki statik değişkenler tarafından yapılandırılan ofsetleri veya alanları yok sayar. Bu ofsetler ve alanlar, yalnızca MB_HOLD_REG tarafından referanslanan proses görünümü veya bellek için geçerlidir. Veri alanının başlangıç ve uzunluk parametreleri, ofsetleri ve alanları tanımlamada benzersiz bir yöntem sağlar.

Tek bir Modbus talebi, yalnızca bir veri alanına yazabilir veya buradan okuyabilir. Örneğin, birden fazla veri alanına yayılan tutma kayıtları okunacaksa, birkaç Modbus talebi gerekir.

Veri alanlarının belirlenmesinde aşağıdaki kurallar geçerlidir:

- Farklı DB'lerde en fazla sekiz veri alanı belirleyebilirsiniz, bu DB'lerden her birinin yalnızca bir veri alanı içermesine izin verilir. Bir MODBUS talebi, yalnızca tek bir veri alanından okuyabilir veya bu alana yazabilir. Her veri alanı bir MODBUS adres alanına karşılık gelir. Veri alanlarını, örnek DB'nin "Data_Area_Array" statik değişkeninde belirlersiniz.
- Sekizden az veri alanı kullanılacaksa, gerek duyduğunuz veri alanlarının aralıksız olarak birbiri ardına düzenlemelisiniz. Veri alanlarındaki ilk boş veri girişinde, işlem sırasında veri alanı araması durdurulur. Örneğin, saha elemanları 1, 2, 4 ve 5 veri alanları olarak belirtilmiştir, saha elemanı 3 boş olduğu için "Data_Area_Array", yalnızca saha elemanları 1 ve 2'yi algılar.
- Data_Area_Array, sekiz elemandan oluşur: Data_Area_Array[1] ila Data_Area_Array[8]
- Her Data_Area_Array[x], $1 \leq x \leq 8$ saha elemanı, MB_DataArea tipinde bir UDT'dir ve aşağıdaki şekilde yapılandırılmıştır:

Parametre	Veri tipi	Anlamı
data_ type	UInt	Bu veri alanına eşlenen MODBUS veri tipi için kod: – 0: Boş bir saha elemanı veya kullanılmayan bir veri alanı için kod. Bu durumda; db, start ve length değerleri önemsizdir. – 1: Proses görünümü çıkışı (fonksiyon kodları 1, 5 ve 15 ile ilişkili olarak) – 2: Proses görünümü girişi (fonksiyon kodu 2 ile ilişkili olarak) – 3: Tutma kayıtları (fonksiyon kodları 3, 6 ve 16 ile ilişkili olarak) – 4: Giriş kaydı (fonksiyon kodu 4 ile bağlantılı olarak) Not: Bir MODBUS veri tipi için bir veri alanı tanımladıysanız, MB_SERVER yordamı artık bu MODBUS veri tipine doğrudan erişemez. Bu tür bir veri tipi için bir MODBUS talebinin adresi belirtilen bir veri alanına karşılık gelmiyorsa, STATUS'ta W#16#8383 değeri geri döndürülür.
db	UInt	MODBUS kaydının veya sonra belirtilen Bit'lerin eşlendiği veri bloğunun numarası DB numarası, veri alanlarında bir kez mevcut olmalıdır. DB numarası, birden fazla veri alanında mevcut olmamalıdır. Global DB'nin "Öznitelikler" özellikler sayfasında, "Yalnızca yükleme belleğinde sakla" ve "Optimize edilmiş modül erişimi" opsiyon kutularının seçimi kaldırılmalıdır. Veri alanları da DB'nin Bayt adresi 0 ile başlar. İzin verilen değerler: 1 ila 60999
start	UInt	0.0 adresinden başlamak üzere veri bloğuna eşlenen ilk MODBUS adresi İzin verilen değerler: 0 ila 65535
length	UInt	Bit sayısı (data_type değerleri 1 ve 2 için) veya kayıt sayısı (data_type değerleri 3 ve 4 için) Bir MODBUS veri tipinin MODBUS adres aralıkları çakışmamalıdır. İzin verilen değerler: 1 ila 65535

Veri alanlarını belirlenme örnekleri:

- Birinci örnek: data_type = 3, db = 1, start = 10, length = 6
  CPU, veri bloğu 1'deki (db = 1) tutma kayıtlarını (data_type = 3) eşler ve Modbus adresi 10'u (start = 10) veri sözcüğü 0'a ve son geçerli Modbus adresi 15'i (length = 6) veri sözcüğü 5'e saklar.
- İkinci örnek: data_type = 2, db = 15, start = 1700, length = 112
  CPU, girişleri (data_type = 2) veri bloğu 15'e (db = 15) eşler ve Modbus adresi 1700'ü (start = 1700) veri sözcüğü 0'a ve son geçerli Modbus adresi 1811'i (length = 112) veri sözcüğü 111'e saklar.

Koşul kodları

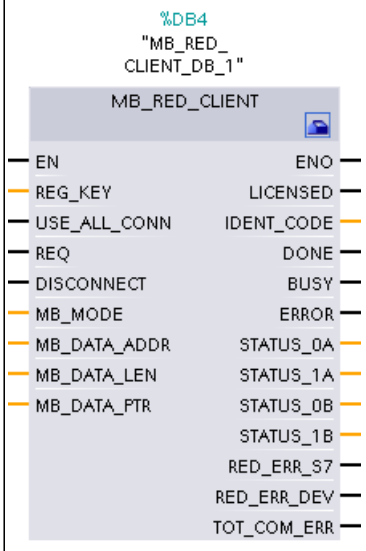
Tablo 13-70 MB_SERVER yürütmesinin koşul kodları [1]

STATUS (W#16#)	Modbus sunucusuna yanıt kodu(B#16#)	Modbus protokol hatası
7001		MB_SERVER, atanan TCP portuna bağlanmak için bir Modbus istemcisini bekliyor. Bu kod, bir bağlantı ilk kurulduğunda veya sonlandırıldığında yayınlanır.
7002		MB_SERVER, atanan TCP portuna bağlanmak için bir Modbus istemcisini bekliyor. Bu kod, bir bağlantı kurulumunun veya sonlandırılmasının tamamlanmasını beklerken sonraki tüm yürütmeler için yayınlanır.
7003		Bir bağlantı başarıyla sonlandırıldı (yalnızca bir PLC çevrimi için geçerlidir).
8187		MB_HOLD_REG geçerli değil, muhtemelen optimize edilmiş bir DB'ye veya 2 Bayt'tan küçük bir aralığa işaret ediyor.
818C		MB_HOLD_REG işaretçisi, optimize edilmemiş bir DB aralığına işaret ediyor (bu, optimize edilmemiş bir global DB alanı veya markörlerin bellek alanındaki bir alan olmalıdır) veya bloke edilmiş işlem için zaman aşımı 55 saniye sınırını aşıyor. (S7-1200'e özgü)
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluğunda hata • Alınan Modbus telegramında geçersiz uzunluk belirtimi. • Modbus telegramının başlığındaki telegram uzunluğu, fiilen alınan Bayt sayısına karşılık gelmiyor. • Modbus telegramının başlığındaki Bayt sayısı, fiilen alınan Bayt sayısına karşılık gelmiyor (fonksiyon 15 ve 16).
8383	02	Veri adresi hatası veya MB_HOLD_REG adres aralığının sınırları dışında erişim
8384	03	Veri değerinde hata
8385	03	Veri teşhis kodu desteklenmiyor (fonksiyon kodu 08)
8389		Geçersiz veri alanı tanımı: • Veri tipi için geçersiz değer • DB numarası geçersiz veya mevcut değil: – Geçersiz DB değeri – DB numarası mevcut değil – DB numarası halen başka bir veri alanı tarafından kullanılıyor – Optimize edilmiş erişime sahip DB – DB, çalışma belleğinde değil • Geçersiz uzunluk değeri • Aynı MODBUS veri tipine ait çakışan MODBUS adres aralıkları

[1] Yukarıda listelenen MB_SERVER hatalarının yanı sıra iletişim için temel T blok yordamları (TCON, TDISCON, TSEND ve TRCV) tarafından da hatalar yayınlanır.

Bu yordamı, bir S7-1200 CPU ile Modbus TCP protokolünü destekleyen bir cihaz arasında bağlantı kurmak için kullanabilirsiniz.

Tablo 13-71 MB_RED_CLIENT yordamı

LAD/FBD	SCL	Açıklama
	**"MB_RED_CLIENT_DB"(** **REG_KEY:=_string_in_,** **USE_ALL_CONN:=_bool_in_** **REQ:=_bool_in_,** **DISCONNECT:=_bool_in_,** **MB_MODE:=_usint_in_,** **MB_DATA_ADDR:=_udint_in_,** **MB_DATA_LEN:=_uint_in_,** **LICENSED=>_bool_out_** **IDENT_CODE=>_string_out_** **DONE=>_bool_out_,** **BUSY=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS_0A=>_word_out_,** **STATUS_1A=>_word_out_,** **STATUS_0B=>_word_out_,** **STATUS_1B=>_word_out_,** **RED_ERR_S7=>_bool_out_,** **RED_ERR_DEV=>_bool_out_,** **TOT_COM_ERR=>_bool_out_,** **MB_DATA_PTR:=_variant_inout_);**	MB_RED_CLIENT yordamı, PROFINET bağlantısı üzerinden bir Modbus TCP istemcisi olarak iletişim kurar. MB_RED_CLIENT yordamını, istemci ve sunucu arasında bir yedekli bağlantı kurmak, Modbus talepleri göndermek, yanıtlar almak ve bağlantının Modbus TCP istemcisi tarafından sonlandırılmasına kumanda etmek için kullanabilirsiniz.

Tablo 13-72 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REG_KEY [1]	IN	STRING[17]	Lisanslama için kayıt kodu MB_RED_CLIENT yordamı, her bir CPU'da ayrı olarak lisanslanmalıdır.
USE_ALL_CONN	IN	Bool	Telegramın gönderileceği yapılandırılmış bağlantıların sayısını girin: • 0: Telegramı tek bir bağlantı üzerinden gönderin, yalnızca bir hata durumunda bir sonraki bağlantıya geçin • 1: Yapılandırılan tüm bağlantılar üzerinden telegram gönder
REQ	IN	Bool	Modbus TCP sunucusuna Modbus sorgusu REQ parametresi adım kontrollüdür. Bunun anlamı, giriş ayarlandığı sürece (REQ = DOĞRU) yordamın iletişim taleplerini göndermesidir. Bağlantı henüz kurulmadıysa, şimdi kurulacak ve bunun ardından hemen Modbus telegramı gönderilecektir. Giriş parametrelerinde yapılan değişiklikler yalnızca sunucu yanıt verdiğinde veya bir hata mesajı verildiğinde etkili olur. Devam eden bir Modbus talebi sırasında REQ parametresi tekrar ayarlanırsa, sonrasında başka iletim gerçekleşmez.

[1] Daha fazla bilgiyi, takip eden "Lisanslama" alt bölümünde bulabilirsiniz:

[2] Daha fazla bilgiyi, takip eden "Giriş parametreleri: MB_MODE, MB_DATA_ADDR, MB_DATA_LEN ve MB_DATA_PTR" alt bölümünde bulabilirsiniz.

[3] Daha fazla bilgiyi, takip eden "Çıkış parametreleri: STATUS_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR" alt bölümünde bulabilirsiniz.

Parametre ve veri tipi		Veri tipi	Açıklama
DISCONNECT	IN	Bool	Bu parametre ile Modbus sunucusuna bağlantının kurulmasını ve sonlandırılmasını kontrol edersiniz: • 0: CONNECT parametresinde yapılandırılan bağlantı partneriyle iletişim bağlantısının kurulması (CONNECT parametresine bakın). • 1: İletişim bağlantısının sonlandırılması Bağlantının sonlandırılmasında başka herhangi bir fonksiyon yürütülmez. Bağlantı başarıyla sonlandırıldıktan sonra, STATUS_x parametresinde 0003 değeri verilir.
MB_MODE [2]	IN	USInt	Modbus talebi modunu (okuma, yazma veya teşhis) veya bir Modbus fonksiyonunun doğrudan seçimini seçer
MB_DATA_ADDR [2]	IN	UDInt	MB_MODE'a bağımlı Modbus adresi
MB_DATA_LEN	IN	UInt	Veri uzunluğu: Veri erişimi için Bit veya kayıt sayısı
MB_DATA_PTR [2]	IN_OUT	Variant	İşaretçi, Modbus sunucusundan alınacak veriler veya Modbus sunucusuna gönderilecek veriler için bir veri tamponu üzerinde.
LICENSED [1]	OUT	Bool	• 0: Yordam, lisanslı değil • 1: Yordam, lisanslı
IDENT_CODE [1]	OUT	STRING[18]	Lisanslama için tanımlama REG_KEY kayıt kodunu istemek için bu karakter dizisini kullanın.
DONE	OUT	Bool	Etkinleştirilen Modbus görevi en az bir bağlantıda hatasız tamamlanır tamamlanmaz, DONE çıkış parametresindeki Bit "1" olarak ayarlanır.
BUSY	OUT	Bool	• 0: Devam eden Modbus talebi yok • 1: Modbus talebi işleniyor BUSY çıkış parametresi, bağlantı kurulurken ve sonlandırılırken ayarlanmaz.
ERROR	OUT	Bool	• 0: Hata yok • 1: Etkinleştirilen Modbus görevi, yapılandırılmış bağlantılardan hiçbirine başarıyla aktarılamadı. STATUS_x parametresi, hatanın sebebini belirtir.
STATUS_0A [3]	OUT	Word	0A bağlantısı için yordamın ayrıntılı durum bilgisi.
STATUS_1A [3]	OUT	Word	1A bağlantısı için yordamın ayrıntılı durum bilgisi.
STATUS_0B [3]	OUT	Word	0B bağlantısı için yordamın ayrıntılı durum bilgisi.
STATUS_1B [3]	OUT	Word	1B bağlantısı için yordamın ayrıntılı durum bilgisi.
RED_ERR_S7 [3]	OUT	Bool	• 0: SIMATIC'te yedeklilik hatası yok • 1: SIMATIC'te yedeklilik hatası
RED_ERR_S7 [3]	OUT	Bool	• 0: Link partneri tarafında yedeklilik hatası yok • 1: Link partneri tarafında yedeklilik hatası
RED_ERR_S7 [3]	OUT	Bool	• 0: En az 1 yapılandırılmış bağlantı kuruldu • 1: Tam iletişim kaybı, yapılandırılmış tüm bağlantılar sonlandırılır

[1] Daha fazla bilgiyi, takip eden "Lisanslama" alt bölümünde bulabilirsiniz:

[2] Daha fazla bilgiyi, takip eden "Giriş parametreleri: MB_MODE, MB_DATA_ADDR, MB_DATA_LEN ve MB_DATA_PTR" alt bölümünde bulabilirsiniz.

[3] Daha fazla bilgiyi, takip eden "Çıkış parametreleri: STATUS_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR" alt bölümünde bulabilirsiniz.

NOT

Bir MB_RED_CLIENT çağrısı sırasında tutarlı giriş verileri

Bir Modbus istemci yordamı çağrıldığında, giriş parametresi değerleri dahili olarak saklanır. Telegram işlenirken değerler değiştirilmemelidir.

Birden çok istemci bağlantısı

CPU'lar birden çok Modbus TCP istemci bağlantısını işleyebilir. Maksimum bağlantı sayısı, kullanılan CPU'ya bağlı olup, CPU'nun teknik verilerinde bulunabilir. Modbus TCP istemcileri ve sunucularından olanlar da dahil olmak üzere bir CPU'dan yapılan toplam bağlantı sayısının, desteklenen maksimum bağlantı sayısını aşmasına izin verilmez.

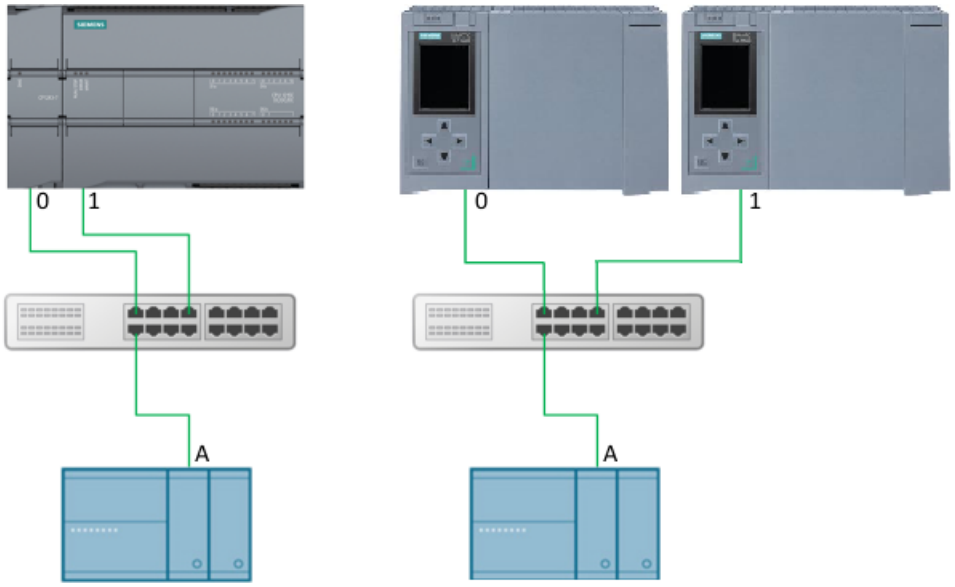
Münferit istemci bağlantıları için aşağıdaki kuralları dikkate alın:

- Her MB_RED_CLIENT bağlantısı, benzersiz bir örnek DB kullanmalıdır.
- Her MB_RED_CLIENT bağlantısı için benzersiz bir sunucu IP adresi belirtilmelidir.
- Her MB_RED_CLIENT bağlantısı, benzersiz bir bağlantı ID'si gerektirir. Bağlantı CPU genelinde benzersiz olmalıdır.

İşletim ve yedeklilik

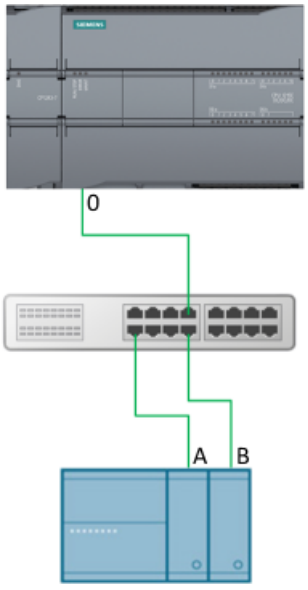
İletişim katılımcıları, bağımsız veya yedekli katılımcılar olabilir. Partnerlerden birinin bağımsız çalışması, tek yönlü yedeklilik olarak adlandırılır. Her iki partner de bağımsız olarak yürütülüyorsa, buna iki yönlü yedeklilik denir:

- Tek taraflı yedeklilik
  - Açıklama: İletişim partnerleri arasındaki her bağlantı için bir bağlantı yapılandırılmalıdır. **SIMATIC S7**'nin bağlantı noktaları, **0** ve **1** olarak adlandırılır. **İletişim partnerlerinin** bağlantı noktaları, **A** ve **B** olarak adlandırılır. R-CPU veya H-CPU'da 1, bağlantı noktası 0'ı ifade eder ve R-CPU veya H-CPU'da 2, bağlantı noktası 1'i anlamına gelir.
  - Konfigürasyon: S7 yedekliyse, S7 bağlantı noktası 0'dan link partnerinin A düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm **A**'ya bağlantı => bağlantı **0A**) ve S7 bağlantı noktası 1'den link partnerinin A düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **1**'den partner/düğüm **A**'ya bağlantı => bağlantı **1A**). Şekilde, bağlantı adları gösterilmektedir:



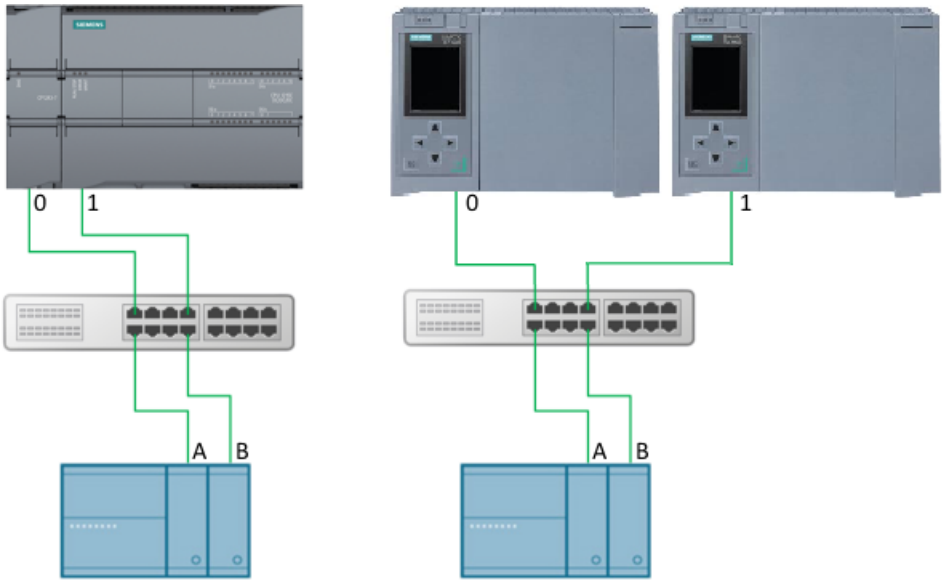
Resim 13-1 S7 tek taraflı yedeklilik

– S7 bağımsızsa ve link partneri yedekliyse, S7 bağlantı noktası 0'dan link partnerinin A düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm **A**'ya bağlantı => bağlantı **0A**) ve S7 bağlantı noktası 0'dan link partnerinin B düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm **B**'ye bağlantı => bağlantı **0B**). Şekilde, bağlantı adları gösterilmektedir:



Resim 13-2 Tek taraflı yedeklilik partneri

- İki yönlü yedeklilik:
  - Açıklama: İletişim partnerleri arasındaki her bağlantı için bir bağlantı yapılandırılmalıdır. **SIMATIC S7**'nin bağlantı noktaları, 0 ve **1** olarak adlandırılır. **İletişim partnerlerinin** bağlantı noktaları, **A** ve **B** olarak adlandırılır.
    R-CPU veya H-CPU'da 1, bağlantı noktası 0'ı ifade eder ve R-CPU veya H-CPU'da 2, bağlantı noktası 1'i anlamına gelir.
  - Konfigürasyon: İki taraflı yedeklilikte, bağlantı noktası 0'dan iki bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm noktası **A**'ya bağlantı => bağlantı **0A** ve S7 bağlantı noktası **0**'dan partner/düğüm noktası **B**'ye bağlantı => bağlantı **0B**) ve S7'nin bağlantı noktası 1'den bağlantı partnerinin A ve B düğüm noktalarına iki bağlantı oluşturulur (S7 bağlantı noktası **1**'den partner/düğüm noktası **A**'ya bağlantı => bağlantı **1A** ve S7 bağlantı noktası **1**'den partner/düğüm noktası **B**'ye bağlantı => bağlantı **1B**). Şekilde, bağlantı adları gösterilmektedir:



Resim 13-3 İki yönlü yedeklilik:

- Telegram işleme: Telegramlar, bir bağlantı veya yapılandırılmış tüm bağlantılar üzerinden gönderilebilir:
  - Telegramların tek bir bağlantı üzerinden gönderilmesi: MODBUS telegramı USE_ALL_CONN = FALSE ayarı ile - o an aktif olan - bir bağlantı üzerinden gönderilir. Bir zaman aşımı (sunucudan yanıt alınamaması) veya bir bağlantı hatası halinde, telegramın diğer yapılandırılmış bağlantılar (maksimum 4) üzerinden gönderilmesine çabalanır. Bu durumda sıralama, 0A, 1A, 0B ve 1B şeklinde olur. Bir telegram bağlantılardan biri üzerinden başarıyla aktarılmışsa, bu bağlantı "aktif" olarak işaretlenir ve sonraki telegram trafiği de bu bağlantı üzerinden gerçekleştirilir. Aktif bağlantıda bir bağlantı hatası olması halinde, telegramı, yapılandırılmış tüm bağlantılar üzerinden göndermek için yeni bir çabada bulunulur. Tüm gönderim denemeleri başarısız olursa, ERROR ve STATUS_x buna karşılık geldiği şekilde ayarlanır.
    Bir yanıt telegramı alındığında, bir olasılık denetimi gerçekleştirilir. Bu denetim başarılı olursa, gerekli eylemler gerçekleştirilir ve görev hatasız olarak yürütülür. DONE çıkışı ayarlanır. Kontrol sırasında hata algılanırsa, görev hatalarla sonlandırılır, ERROR Bit'i ayarlanır ve STATUS_x adresinde bir hata numarası görüntülenir. Bu durumda telegramı, bir sonraki yapılandırılmış bağlantıda göndermek için yeni bir çabada

bulunulmaz. Yapılandırılmış diğer bağlantılara geçiş sadece, bir bağlantı hatası algılandığında veya yanıt alınmadığında yürütülür.

– Telegramların tüm bağlantılar üzerinden gönderilmesi: MODBUS telegramı, yapılandırılmış ve oluşturulmuş tüm bağlantılar üzerinden USE_ALL_CONN = TRUE ayarı ile gönderilir. Yanıt telegramı bağlantılardan birinde alındıktan sonra, bir geçerlilik denetimi gerçekleştirilir. Bu denetim başarılı olursa, gerekli eylemler gerçekleştirilir. En az bir bağlantıda geçerli bir yanıt telegramı alınmışsa, DONE çıkışı ayarlanır.

- RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR yedeklilik çıkışları:
  – RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR yedeklilik Bit'leri durum çıkışlarının durumlarına bağlı olarak ayarlanır:

Number of faulty connections	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	okay	okay	okay	FALSE	FALSE	FALSE
1	okay	okay	okay	Error	FALSE	FALSE	FALSE
	okay	okay	Error	okay	FALSE	FALSE	FALSE
	okay	Error	okay	okay	FALSE	FALSE	FALSE
	Error	okay	okay	okay	FALSE	FALSE	FALSE
2	okay	okay	Error	Error	TRUE	FALSE	FALSE
	okay	Error	okay	Error	FALSE	TRUE	FALSE
	Error	okay	okay	Error	FALSE	FALSE	FALSE
	okay	Error	Error	okay	FALSE	FALSE	FALSE
	Error	okay	Error	okay	FALSE	TRUE	FALSE
	Error	Error	okay	okay	TRUE	FALSE	FALSE
3	Error	Error	Error	okay	TRUE	TRUE	FALSE
	Error	Error	okay	Error	TRUE	TRUE	FALSE
	Error	okay	Error	Error	TRUE	TRUE	FALSE
	okay	Error	Error	Error	TRUE	TRUE	FALSE
4	Error	Error	Error	Error	TRUE	TRUE	TRUE

Resim 13-4 Her iki tarafta yedeklilik kurulumu için alarm Bit'lerinin görüntülenmesi

Number of faulty connections	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	0AFF	okay	0AFF	FALSE	FALSE	FALSE
1	okay	0AFF	Error	0AFF	TRUE	TRUE	FALSE
	Error	0AFF	okay	0AFF	TRUE	TRUE	FALSE
2	Error	0AFF	Error	0AFF	TRUE	TRUE	TRUE

Resim 13-5 Yedeklilik kurulumu için alarm Bit'lerinin bir sayfada görüntülenmesi

NOT

İstemci ve sunucu için port numaraları

Modbus istemcisi 2000'den başlayan bir port numarası kullanır. Modbus sunucusu genellikle 502 numaralı portu üzerinden adreslenir.

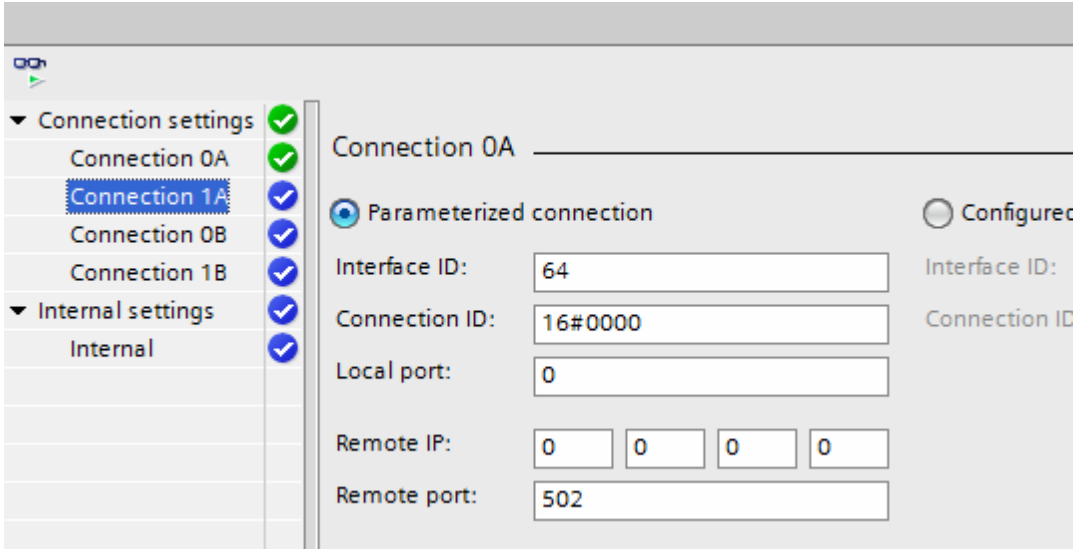
Parametrelendirme

S7-1200'de MB_RED_CLIENT **V1.0** ve **V1.1** yordamını kullanabilirsiniz. CPU, bağlantıları CPU'nun yerel arayüzü veya CM/CP aracılığıyla uygular. CPU, bağlantıları TCON_IP_V4 yapısı üzerinden yapılandırır ve kurar.

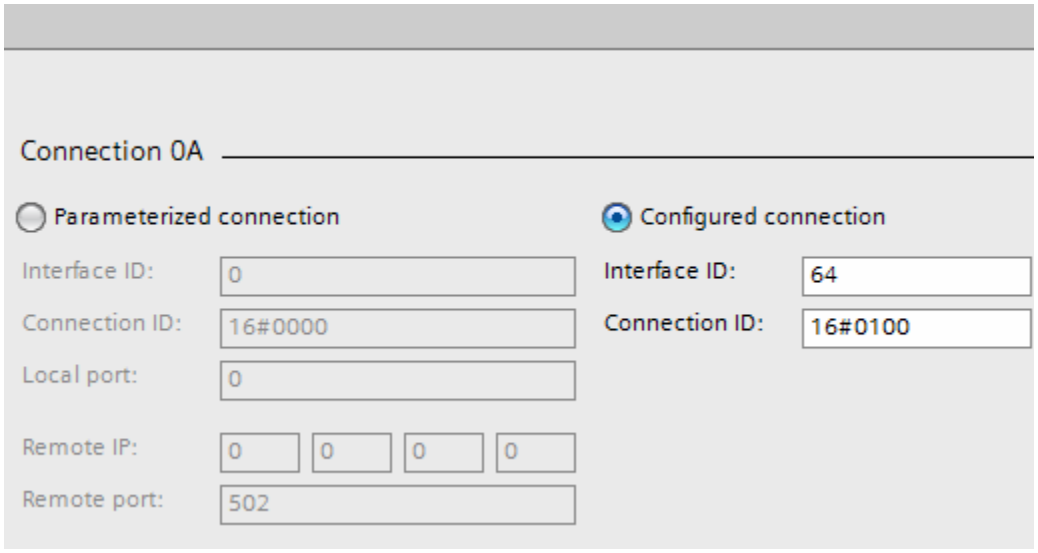
MB_RED_CLIENT konfigürasyonu: MB_RED_CLIENT yordamının konfigürasyon iletişim kutusunda aşağıdaki ayarları yapın:

- 0A, 1A, 0B ve 1B bağlantıları için bağlantı parametreleri (yedeklilik konfigürasyonu hakkında daha fazla bilgiyi, yukarıdaki "Çalışma ve Yedeklilik" bölümünde bulabilirsiniz)
- Dahili parametreler (opsiyonel)

Konfigürasyon iletişim kutusunu, MB_RED_CLIENT yordamıyla veya teknoloji nesneleri üzerinden açabilirsiniz:

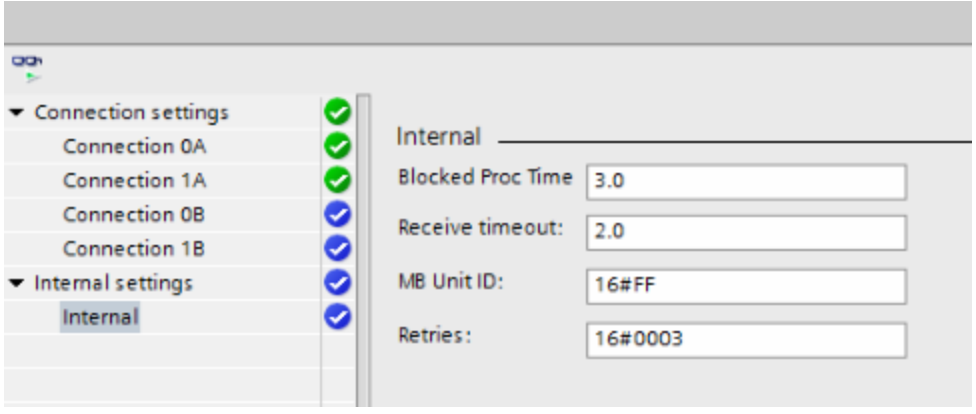


Resim 13-6 Parametrelendirilmiş istemci bağlantısı



Resim 13-7 Yapılandırılmış istemci bağlantısı

Değişken	Başlangıç değeri	Açıklama
Yapılandırılmış bağlantılar		
Interface ID	64	Kullanılan PN arayüzünün donanım kodu
Connection ID	16#0000	Kullanılan bağlantılar için bağlantı ID'leri Bu bağlantı ID'leri, CPU genelinde benzersiz olmalıdır.
Local port	0	İstemcinin yerel port numarası. Standart olarak, istemciye port numarası girilmez.
Remote IP	0.0.0.0	Sunucunun uzak IP adresi
Remote port	502	Sunucunun uzak port numarası Modbus/TCP sunucu için standart port, 502'dir.
Yapılandırılmış bağlantılar		
Interface ID	64	Kullanılan PN arayüzünün donanım kodu
Connection ID	16#0000	Kullanılan bağlantılar için bağlantı ID'leri Bu bağlantılar, ağ görünümünde yapılandırılır.



Resim 13-8 Dahili parametreler (opsiyonel)

Değişken	Veri tipi	Başlangıç değeri	Açıklama
Blocked Proc Time	REAL	3.0	Bir Modbus örneği bloke edilmişse, ACTIVE statik değişkenin sıfırlanmasından önceki saniye olarak bekleme süresi. Örneğin bu, bir istemci talebi gönderildiğinde ve istemci fonksiyonunun yürütülmesi, talep tamamlanmadan önce iptal edildiğinde ortaya çıkabilir. Bekleme süresi, 0,5 sn ile 55 sn arasında olmalıdır.
Receive timeout	REAL	2.0	"MB_RED_CLIENT" yordamının sunucudan yanıt beklediği saniye cinsinden zaman aralığı. Aralık, 0,5 sn ile 55 sn arasında olmalıdır.
MB_Unit_ID	BYTE	255	Modbus cihaz tanımlama: Bir Modbus TCP sunucusu, IP adresi üzerinden adreslenir. Bundan ötürü Modbus TCP adreslemede, MB_UNIT_ID parametresi kullanılmaz. MB_UNIT_ID parametresi, Modbus RTU protokolündeki Slave adresine karşılık gelir. Bir Modbus/TCP sunucusu bir Modbus RTU protokolü için Gateway olarak kullanıldığında, Slave cihaz seri ağda MB_UNIT_ID aracılığıyla tanımlanabilir. Bu durumda MB_UNIT_ID parametresi, talebi doğru Modbus RTU Slave adresine aktarır. Bu bağlamda, bazı Modbus/TCP cihazlarını sınırlı bir değer aralığında başlatmak için, MB_UNIT_ID parametresinin gerekebileceğini dikkate alın.
Retries	WORD	3	W#16#80C8 hatasını geri döndürmeden önce MB_RED_CLIENT yordamını gönderme denemelerinin sayısı.

NOT

MB_Transaction_ID değişkeni

Modbus TCP sunucusunun yanıtındaki işlem ID'si MB_RED_CLIENT talebinin işlem ID'si ile eşleşmiyorsa, MB_RED_CLIENT yordamı RCV_TIMEOUT * RETRIES süresi boyunca Modbus TCP sunucusundan doğru işlem ID'si ile yanıt bekler. Bu süre geçer geçmez W#16#80C8 hatası geri döndürülür.

Lisanslama

MB_RED_CLIENT yordamı belli bir ücrete tabidir ve bu yordamı her bir CPU için ayrı ayrı lisanslamanız gerekmektedir. Lisanslama, iki adımda gerçekleşir:

- Lisansın IDENT_CODE'unun görüntülenmesi
- REG_KEY kayıt anahtarının girilmesi: REG_KEY kayıt anahtarını, her MB_RED_CLIENT yordamında girmeniz gerekir. REG_KEY'i, tüm MB_RED_CLIENT yordamlarının gereken kayıt anahtarını aldıkları bir global veri bloğunda saklayın.

Lisansın IDENT_CODE'unu görüntüleme prosedürü:

1. MB_RED_CLIENT yordamını bir periyodik OB'de kendi gerekliliklerinize göre parametrelendirin. Programı CPU'ya yükleyin ve CPU'yu RUN işletim durumuna getirin.
2. Modbus yordamının örnek DB'sini açın ve "Tümünü izle" butonuna tıklayın.
3. Örnek DB, IDENT_CODE çıkışında 18 karakterlik bir karakter dizisi görüntüler.

	Name	Data type	Start value	Monitor value
1	Input			
2	REG_KEY	String[17]	''	''
3	USE_ALL_CONN	Bool	false	FALSE
4	REQ	Bool	false	FALSE
5	DISCONNECT	Bool	false	FALSE
6	MB_MODE	USInt	0	0
7	MB_DATA_ADDR	UDInt	0	0
8	MB_DATA_LEN	UInt	0	0
9	Output			
10	LICENSED	Bool	false	FALSE
11	IDENT_CODE	String[18]	''	'RTPCFIGDCDIIHJHAH4'
12	DONE	Bool	false	FALSE
13	BUSY	Bool	false	FALSE
14	ERROR	Bool	false	FALSE

Resim 13-9 Lisans

4. Kopyala/yapıştır fonksiyonunu kullanarak, bu karakter dizisini veri bloğundan kopyalayın ve forma yapıştırın (ürünü sipariş ettikten sonra e-posta ile aldığınız veya ayrıca CD'de bulunan).
5. Bu formu bir hizmet isteği üzerinden Müşteri desteği (https://support.industry.siemens.com/my/ww/en/requests/#createRequest)'a gönderin. Bunun ardından, size CPU'nuz için kayıt anahtarı gönderilecektir.

REG_KEY kayıt anahtarını girme prosedürü:

1. "Yeni blok ekle..." yoluyla benzersiz bir sembolik ada sahip yeni bir global veri bloğu ekleyin, örneğin "Lisans-DB".
2. Bu blokta, STRING[17] veri tipiyle bir REG_KEY parametresi oluşturun.

LICENSE_DB

	Name	Data type	Offset	Start value
1	Static			
2	REG_KEY	String[17]	0.0	''

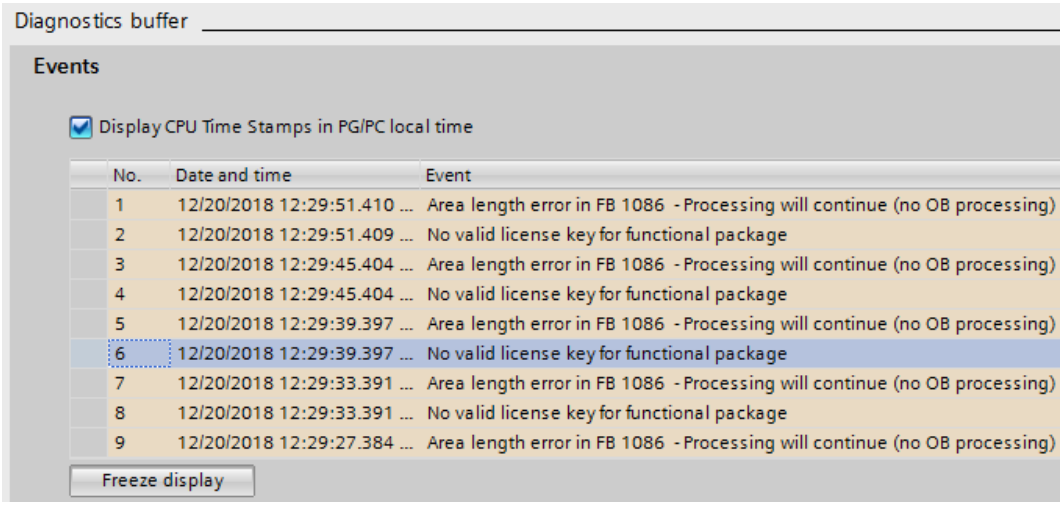
Resim 13-10 REG KEY

3. Aktarılan 17 basamaklı kayıt anahtarını "Başlangıç değeri" sütununa kopyalamak için Kopyala/yapıştır fonksiyonunu kullanın.

4. MB_RED_CLIENT yordamının REG_KEY parametresinde periyodik OB'de lisans DB'sinin adını ve karakter dizisinin adını (örn. License_DB.REG_KEY) girin.
5. Değiştirilmiş veri bloklarını CPU'ya yükleyin, çalışma zamanı boyunca kayıt anahtarını girebilirsiniz. Bunun için CPU'yu STOP'tan RUN işletim durumuna geçirmek gerekli değildir.
6. Modbus/TCP iletişimi, MB_RED_CLIENT yordamı üzerinden artık bu CPU için lisanslanmıştır, LICENSED çıkış Bit'i TRUE'dur.

Eksik veya hatalı lisanslamayı düzeltme prosedürü:

- Yanlış bir kayıt anahtarı girerseniz veya herhangi bir kayıt anahtarı girmezseniz, CPU'nun ERROR LED'i yanıp söner. Buna ek olarak, S7-1200'ün CPU'su, eksik lisansla ilgili olarak teşhis tamponuna bir periyodik veri girişi yapar.



Resim 13-11 Teşhis tamponu

- Eksik veya hatalı kayıt anahtarında CPU, Modbus TCP iletişimini işler. Buna karşın CPU STATUS_x çıkışında her zaman "W#16#0A90" (fonksiyon paketi için geçerli lisans anahtarı yok) görüntüler. LICENSED çıkış Bit'i FALSE'tır.

Giriş parametreleri: MB_MODE, MB_DATA_ADDR, MB_DATA_LEN ve MB_DATA_PTR

MB_MODE, MB_DATA_ADDR ve MB_DATA_LEN kombinasyonu, güncel Modbus mesajında kullanılan fonksiyon kodunu belirler.

- **MB_MODE** okuma veya yazma bilgisini içerir:
  Oku: MB_MODE = 0, 101, 102, 103 ve 104
  Yaz: MB_MODE = 1, 2, 105, 106, 115 ve 116 (Bilgi notu: MB_MODE = 2 ise, Modbus fonksiyonları 15 ile 05 arasında veya Modbus fonksiyonları 16 ile 06 arasında herhangi bir ayrım yoktur).
- **MB_DATA_ADDR** neyin okunacağı veya yazılacağı ve ayrıca MB_RED_CLIENT yordamının uzak adresi hesaplamak için kullandığı adres bilgilerini içerir.
- **MB_DATA_LEN** okunması/yazılması gereken Bit'lerin sayısını içerir.

Aşağıdaki tabloda, MB_RED_CLIENT yordamının MB_MODE, MB_DATA_ADDR, MB_DATA_LEN giriş parametreleri ile Modbus fonksiyonu arasındaki ilişki gösterilmektedir:

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus fonksiyonu	Fonksiyon ve veri tipi
0	1 ila 9.999	1 ila 2.000	01	Uzak adres 0 ila 9.998'de 1 ila 2.000 çıkış Bit'lerini okur
0	10.001 ila 19.999	1 ila 2.000	02	Uzak adres 0 ila 9.998'de 1 ila 2.000 giriş Bit'lerini okur
0	• 40.001 ila 49.999 • 400.001 ila 465.535	1 ila 125	03	• Uzak adres 0 ila 9.998'de 1 ila 125 tutma kaydını okur • Uzak adres 0 ila 65.534'de 1 ila 125 tutma kaydını okur
0	30.001 ila 39.999	1 ila 125	04	Uzak adres 0 ila 9.998'de 1 ila 125 giriş sözcükleri okur
1	1 ila 9.999	1	05	Uzak adres 0 ila 9.998'de arasında 1 çıkış Bit'i yazar
1	• 40.001 ila 49.999 • 400.001 ila 465.535	1	06	• Uzak adres 0 ila 9.998'de arasındaki 1 tutma kaydı yazar • Uzak adres 0 ila 65.534'te arasındaki 1 tutma kaydı yazar
1	1 ila 9.999	2 ila 1.968	15	0 ila 9.998 arasındaki uzak adrese 2 ila 1.968 çıkış Bit'i yazar
1	• 40.001 ila 49.999 • 400.001 ila 465.535	2 ila 123	16	• Uzak adres 0 ila 9.998'de 2 ila 123 tutma kaydı yazar • Uzak adres 0 ila 65.534'te 2 ila 123 tutma kaydı yazar
2	1 ila 9.999	1 ila 1.968	15	Uzak adres 0 ila 9.998'de 1 ila 1.968 çıkış Bit'i yazar

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Modbus fonksiyonu	Fonksiyon ve veri tipi
2	• 40.001 ila 49.999 • 400.001 ila 465.535	1 ila 123	16	• Uzak adres 0 ila 9.998'de 1 ila 123 tutma kaydını okur • Uzak adres 0 ila 65.534'te 1 ila 123 tutma kaydı yazar
11	Bu fonksiyon yürütüldüğünde yordam, MB_DATA_ADDR ve MB_DATA_LEN parametrelerini değerlendirmez.		11	Sunucunun durum sözcüğünü ve olay sayacını okur: • Durum sözcüğü, işleme durumunu gösterir (0 - işleme yok, 0xFFFF - işleniyor). • Modbus talebi başarıyla yürütüldükten sonra, olay sayacı değeri artırılır. Bir Modbus fonksiyonu yürütülürken bir hata oluşursa, sunucudan bir mesaj gönderilir ancak olay sayacı değeri artırılmaz.
80	-	1	08	Teşhis kodu 0x0000 ile sunucu durumunu denetler (çevrim geri dönüş denetimi - sunucu talebi geri gönderir): Çağrı başına 1 WORD
81	-	1	08	Teşhis kodu 0x000A ile sunucunun olay sayacı sıfırlanır: Çağrı başına 1 WORD
101	0 ila 65.535	1 ila 2.000	01	Uzak adres 0 ila 65.535'te 1 ila 2.000 çıkış Bit'lerini okur
102	0 ila 65.535	1 ila 2.000	02	Uzak adres 0 ila 65.535'te 1 ila 2.000 giriş Bit'lerini okur
103	0 ila 65.535	1 ila 125	03	Uzak adres 0 ila 65.535'te 1 ila 125 tutma kaydını okur
104	0 ila 65.535	1 ila 125	04	Uzak adres 0 ila 65.535'te 1 ila 125 giriş sözcüklerini okur
105	0 ila 65.535	1	05	Uzak adres 0 ila 65.535'te arasında 1 çıkış Bit'i yazar
106	0 ila 65.535	1	06	Uzak adres 0 ila 65.535'te 1 tutma kaydı yazar
115	0 ila 65.535	1 ila 1.968	15	Uzak adres 0 ila 65.535'te 1 ila 1.968 çıkış Bit'i yazar
116	0 ila 65.535	1 ila 123	16	Uzak adres 0 ila 65.535'te 1 ila 123 tutma kaydı yazar
3 ila 10, 12 ila 79, 82 ila 100, 107 ila 114, 117 ila 255				Rezerve

Örnek:

Değişken	Anlamı
MB_MODE = 1 MB_DATA_ADDR = 1 MB_DATA_LEN = 1	Uzak adres 0'dan başlayarak fonksiyon kodu 5 olan 1 çıkış Bit'i yazar.
MB_MODE = 1 MB_DATA_ADDR = 1 MB_DATA_LEN = 2	Uzak adres 0'dan başlayarak fonksiyon kodu 15 olan 2 çıkış Bit'i yazar.
MB_MODE = 104 MB_DATA_ADDR = 17834 MB_DATA_LEN = 125	Uzak adres 17.834'ten başlayarak fonksiyon kodu 4 olan 125 giriş Bit'i yazar.

MB_DATA_PTR:

MB_DATA_PTR parametresi, Modbus sunucusundan alınacak veriler veya Modbus sunucusuna gönderilecek veriler için bir veri tamponu üzerinde. Veri tamponu olarak bir global veri bloğu veya bir markör alanı (M) kullanabilirsiniz.

Markör alanındaki (M) bir tampon için, aşağıdaki gibi ANY formatında bir işaretçi kullanın: "P#Bit adresi" "Veri tipi" "Uzunluk" (örneğin: P#M1000.0 WORD 500)

Veri tamponunun bulunduğu bellek alanına bağlı olarak, MB_DATA_PTR farklı veri yapılarına referanslayabilir:

- Optimize edilmiş erişime sahip bir global DB kullanıyorsanız, MB_DATA_PTR, bir temel veri tipindeki bir değişkene veya temel veri tiplerinden oluşan bir Array'e referanslayabilir. Aşağıdaki veri tipleri desteklenir:

Veri tipi	Bit'lerin uzunluğu
Bool	1
Byte, SInt, USInt, Char	8
Word, Int, WChar, UInt	16
DWord, DInt, UDInt, Real	32

  Tüm Modbus fonksiyonları için desteklenen tüm veri tiplerini kullanabilirsiniz. Örnek: MB_RED_CLIENT, Bayt türündeki bir değişkende alınan bir Bit'i, o Bayt'taki diğer Bit'leri değiştirmeden belirtilen bir adrese de yazabilir. Dolayısıyla, Bit odaklı fonksiyonları yürütmek için bir Bit Array'ine sahip olmak zorunluluğu yoktur.
- Bellek alanı olarak standart erişime sahip bir markör alanı veya global DB kullanıyorsanız, MB_DATA_PTR için temel veri tiplerinde bir kısıtlama ortadan kalkmış olur; bunun ardından MB_DATA_PTR PLC veri tipleri (UDT) ve sistem veri tipleri (SDT) gibi karmaşık veri yapılarına da referanslayabilir.

NOT

Bir markör alanının veri tamponu olarak kullanılması

MB_DATA_PTR için veri tamponu olarak bir markör alanı kullanıyorsanız, bu değişkeni dikkate almalısınız. Bu, S7-1200 CPU'larda 8 KB'dir.

Çıkış parametreleri: STATUS_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR

CPU, MB_RED_CLIENT yordamının durum çıkışlarında hata mesajları görüntüler:

NOT

İlgili hata durum kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir:

1. Programlama editöründe istenen bloğu açın.
2. "İzleme açık/kapalı" üzerine tıklayarak programlama durumunu açın. (Henüz bir çevrimiçi bağlantı kurmadıysanız, "Çevrimiçi ol" iletişim kutusu açılır. Bu iletişim kutusunda bir çevrimiçi bağlantı kurabilirsiniz.)
3. İzlemek istediğiniz değişkeni seçin ve bağlam menüsündeki "Görüntüleme formatı" altında istediğiniz görüntüleme formatını seçin.

- STATUS_x parametresi (genel durum bilgileri)

STATUS (W#16#)	Açıklama
0000	Yordam hatasız bir şekilde yürütüldü.
0001	Bağlantı kuruldu.
0003	Bağlantı sonlandırıldı.
0A90	MB_RED_CLIENT yordamı lisanslı değil. Daha fazla bilgiyi, yukarıdaki "Lisanslama" alt bölümünde bulabilirsiniz.
0AFF	Bağlantı yapılandırılmadı ve kullanılmıyor. "0A" bağlantısının yapılandırılması gerekiyor.
7000	Aktif görev yok ve bağlantı kurulmadı (REQ = 0, DISCONNECT = 1).
7001	Bağlantı kurma başlatıldı.
7002	Ara çağrı. Bağlantı kuruluyor.
7003	Bağlantı sonlandırılır.
7004	Bağlantı kuruluyor ve izleniyor. Devam eden bir görev işlemesi yok.
7005	Veriler gönderiliyor.
7006	Veriler alınıyor.

- STATUS_x parametresi (protokol hatası)

STATUS (W#16#)	Açıklama
80C8	Belirtilen zaman dilimi içinde sunucudan yanıt gelmedi. Modbus sunucusuna bağlantıyı kontrol edin. Yapılandırılan yeniden denemeler tamamlanana dek, bu hata bildirilmez. MB_RED_CLIENT yordamı, belirtilen süre içinde orijinal olarak aktarılan işlem ID'sine sahip bir yanıt almazsa (MB_TRANSACTION_ID statik değişkenine bakın), bu hata kodu geri döndürülür.
8380	Alınan Modbus telegramı yanlış formatta veya çok az Bayt alındı.

STATUS (W#16#)	Açıklama
8382	– Telegram başlığındaki Modbus telegramının uzunluğu, alınan Bayt sayısına karşılık gelmiyor. – Bayt sayısı, gerçekte iletilen Bayt sayısına karşılık gelmiyor (yalnızca 1-4 fonksiyonlar). – Alınan telegramdaki başlangıç adresi, kayıtlı başlangıç adresine karşılık gelmiyor (fonksiyonlar 5, 6, 15 ve 16). – Sözcük sayısı, fiilen aktarılan sözcük sayısına karşılık gelmiyor (fonksiyon 15 ve 16).
8383	Veri okuma veya yazma hatası veya MB_DATA_PTR adres aralığının dışında erişim. Daha fazla bilgi için yukarıdaki "MB_DATA_PTR" alt bölüme bakın.
8384	– Geçersiz istisna kodu alındı. – İstemci tarafından başlangıçta gönderilenden farklı bir veri değeri alındı (fonksiyon 5, 6 ve 8) – Geçersiz durum değeri alındı (fonksiyon 11)
8385	– Teşhis kodu desteklenmiyor. – İstemci tarafından başlangıçta gönderilenden farklı bir kısmi fonksiyon kodu alındı (fonksiyon 8).
8386	Alınan fonksiyon kodu, ilk olarak gönderilenle eşleşmiyor.
8387	Sunucudan alınan Modbus TCP telegramının protokol ID'si "0" değildir.
8388	Modbus sunucusu işlenenden farklı bir veri uzunluğu gönderdi. Bu hata yalnızca Modbus fonksiyonları 5, 6, 15 veya 16 kullanılırken oluşur.

- STATUS_x parametresi (parametre hatası)

STATUS (W#16#)	Açıklama
80B6	Geçersiz bağlantı tipi; yalnızca TCP bağlantıları desteklenir.
80BB	ActiveEstablished parametresi geçersiz bir değere sahip. İstemci için yalnızca aktif bağlantı kurulumuna izin verilir (ActiveEstablished = TRUE).
8188	MB_MODE parametresi geçersiz bir değere sahip.
8189	MB_DATA_ADDR parametresinde verilerin geçersiz adreslenmesi
818A	MB_DATA_LEN parametresinde geçersiz veri tipi
818B	MB_DATA_PTR parametresi geçersiz bir işaretçiye sahip. MB_DATA_ADDR ve MB_DATA_LEN parametrelerinin değerlerini de kontrol etmelisiniz. ("MB_DATA_ADDR" hakkında daha fazla bilgi için yukarıdaki "MB_DATA_ADDR" alt bölümüne bakın).
818C	BLOCKED_PROC_TIMEOUT veya RCV_TIMEOUT parametresinde zaman aşımı (yordamın statik değişkenlerine bakın). BLOCKED_PROC_TIMEOUT ve RCV_TIMEOUT 0,5 sn ile 55,0 sn arasında olmalıdır.
8200	– CPU şu an için portta başka bir Modbus talebini işliyor. – Aynı bağlantı parametrelerine sahip başka bir MB_RED_CLIENT örneği mevcut bir Modbus talebini işliyor.

NOT

Dahili olarak kullanılan iletişim yordamlarından gelen hata kodları

MB_RED_CLIENT yordamında, tablolarda listelenen hatalara ek olarak, yordam tarafından kullanılan iletişim yordamlarından (TCON, TDISCON, TSEND, TRCV, T_DIAG ve TRESET) kaynaklanan hatalar da ortaya çıkabilir.

CPU, hata kodlarını örnek veri bloğu aracılığıyla MB_RED_CLIENT yordamına atar. CPU, STATUS altındaki "Static" alt bölümündeki her yordam için hata kodlarını görüntüler.

Hata kodlarının anlamı, ilgili iletişim yordamının dokümantasyonunda mevcuttur.

NOT

Veri gönderirken veya alırken iletişim hatası

Veri gönderirken veya alırken bir iletişim hatası ortaya çıkarsa, CPU mevcut bağlantıyı sonlandırır. Aşağıdaki hatalar ortaya çıkabilir:

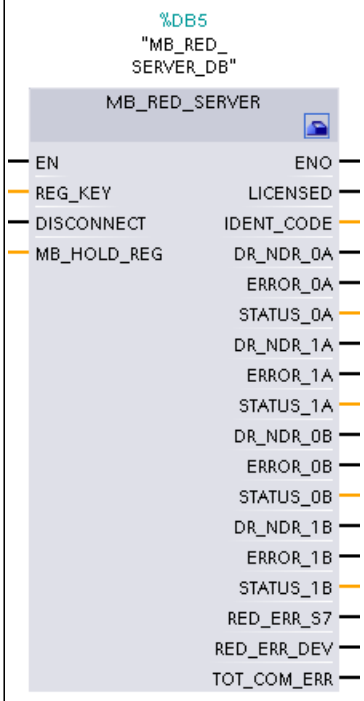
- 80C4 - Geçici iletişim hatası; belirtilen bağlantı geçici olarak sonlandırılır.
- 80C5 - Uzak partner aktif olarak bağlantıyı sonlandırdı.
- 80A1 – Belirtilen bağlantı sonlandırıldı veya henüz kurulmadı.

Bunun anlamı, bağlantı kesildiğinde döndürülen herhangi bir STATUS değerini görebileceğiniz ve bağlantının kesilmesine neden olan STATUS kodunun yalnızca bağlantı kesildiğinde geri döndürüldüğüdür.

Örnek: Veri alırken geçici bir iletişim hatası oluşursa, önce STATUS 7003 (ERROR = False) ve bunun ardından 80C4 (ERROR = True) yayınlanır.

Bu yordamı, bir S7-1200 CPU ile Modbus TCP protokolünü destekleyen bir cihaz arasında bağlantı kurmak için kullanabilirsiniz.

Tablo 13-73 MB_RED_SERVER yordamı

LAD/FBD	SCL	Açıklama
	**"MB_RED_SERVER_DB"(** **DISCONNECT:=_bool_in_,** **LICENSED=>_bool_out_** **IDENT_CODE=>_string_out_** **DR_NDR_0A=>_bool_out,** **ERROR_0A=>_bool_out,** **STATUS_0A=>_word_out_,** **DR_NDR_1A=>_bool_out,** **ERROR_1A=>_bool_out,** **STATUS_1A=>_word_out_,** **DR_NDR_0B=>_bool_out,** **ERROR_0B=>_bool_out,** **STATUS_0B=>_word_out_,** **DR_NDR_1B=>_bool_out,** **ERROR_1B=>_bool_out,** **STATUS_1B=>_word_out_,** **RED_ERR_S7=>_bool_out_,** **RED_ERR_DEV=>_bool_out_,** **TOT_COM_ERR=>_bool_out_,** **MB_HOLD_REG:=_variant_inout_);**	MB_RED_SERVER yordamı, PROFINET bağlantısı üzerinden bir Modbus TCP sunucusu olarak iletişim kurar. MB_RED_SERVER yordamı, bir Modbus TCP istemcisinden gelen bağlantı taleplerini işler, Modbus taleplerini alır ve işler ve yanıtları gönderir.

Tablo 13-74 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REG_KEY [1]	IN	STRING[17]	Lisanslama için kayıt kodu MB_RED_SERVER yordamı, her bir CPU'da ayrı olarak lisanslanmalıdır.
DISCONNECT	IN	Bool	MB_RED_SERVER yordamıyla, bir partner modül ile pasif bir bağlantıya girersiniz. Sunucu, bağlantı açıklamalarında belirtilmiş veya belirtilmemiş olarak gelen IP adreslerinden bir bağlantı talebine yanıt verir. Bir bağlantı talebinin ne zaman kabul edileceğine kumanda etmek için bu parametreyi kullanırsınız: • 0: CPU, iletişim bağlantısı olmadığında pasif bir bağlantı kurar. • 1: Bağlantıyı sonlandırmanın başlatılması. Giriş ayarlandıktan sonra, CPU herhangi bir ek müşteri talebini işlemez ve bağlantı sonlandırılması başlatılır. Bağlantı başarıyla temizlendikten sonra, STATUS_x parametresinde "0003" değeri yayınlanır.

[1] Daha fazla bilgiyi, takip eden "Lisanslama" alt bölümünde bulabilirsiniz:

[2] Daha fazla bilgiyi, takip eden "MB_HOLD_REG giriş parametreleri" alt bölümünde bulabilirsiniz.

[3] Daha fazla bilgiyi, takip eden "Çıkış parametreleri: ERROR_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR" alt bölümünde bulabilirsiniz.

Parametre ve veri tipi		Veri tipi	Açıklama
MB_HOLD_REG [2]	IN_OUT	Variant	İşaretçi, MB_RED_SERVER yordamının Modbus tutma kaydı üzerinde MB_HOLD_REG her zaman iki Bayt'tan daha büyük bir bellek alanına referans vermelidir. Tutma kaydı, bir Modbus istemcisinin Modbus fonksiyonları 3 (okuma), 6 (yazma), 16 (birden çok kez yazma) ve 23 (bir görev sırasında okuma ve yazma) üzerinden erişebileceği değerleri içerir.
LICENSED [1]	OUT	Bool	• 0: Yordam, lisanslı değil • 1: Yordam, lisanslı
IDENT_CODE [1]	OUT	STRING[18]	Lisanslama için tanımlama REG_KEY kayıt kodunu istemek için bu karakter dizisini kullanın.
DR_NDR_0A	OUT	Bool	0A bağlantısı için "Data Read" veya "New Data Ready": • 0: Yeni veriler yok • 1: Modbus istemcisinden okunan veya yazılan veriler
ERROR_0A	OUT	Bool	0A bağlantısı için MB_RED_SERVER yordamı çağrıldığında bir hata oluşursa, ERROR_0A parametresinin çıkışı "1" olarak ayarlanır. STATUS_0A parametresi, bu hata sebebi hakkında ayrıntılı bilgi sunar.
STATUS_0A [3]	OUT	Word	0A bağlantısı için yordamın ayrıntılı durum bilgisi.
DR_NDR_1A	OUT	Bool	1A bağlantısı için "Data Read" veya "New Data Ready": • 0: Yeni veriler yok • 1: Modbus istemcisinden okunan veya yazılan veriler
ERROR_1A	OUT	Bool	1A bağlantısı için MB_RED_SERVER yordamı çağrıldığında bir hata oluşursa, ERROR_1A parametresinin çıkışı "1" olarak ayarlanır. STATUS_1A parametresi, bu hata sebebi hakkında ayrıntılı bilgi sunar.
STATUS_1A [3]	OUT	Word	1A bağlantısı için yordamın ayrıntılı durum bilgisi.
DR_NDR_0B	OUT	Bool	0B bağlantısı için "Data Read" veya "New Data Ready": • 0: Yeni veriler yok • 1: Modbus istemcisinden okunan veya yazılan veriler
ERROR_0B	OUT	Bool	0B bağlantısı için MB_RED_SERVER yordamı çağrıldığında bir hata oluşursa, ERROR_0B parametresinin çıkışı "1" olarak ayarlanır. STATUS_0B parametresi, bu hata sebebi hakkında ayrıntılı bilgi sunar.
STATUS_0B [3]	OUT	Word	0B bağlantısı için yordamın ayrıntılı durum bilgisi.
DR_NDR_1B	OUT	Bool	1B bağlantısı için "Data Read" veya "New Data Ready": • 0: Yeni veriler yok • 1: Modbus istemcisinden okunan veya yazılan veriler
ERROR_1B	OUT	Bool	1B bağlantısı için MB_RED_SERVER yordamı çağrıldığında bir hata oluşursa, ERROR_1B parametresinin çıkışı "1" olarak ayarlanır. STATUS_1B parametresi, bu hata sebebi hakkında ayrıntılı bilgi sunar.
STATUS_1B [3]	OUT	Word	1B bağlantısı için yordamın ayrıntılı durum bilgisi.

[1] Daha fazla bilgiyi, takip eden "Lisanslama" alt bölümünde bulabilirsiniz:

[2] Daha fazla bilgiyi, takip eden "MB_HOLD_REG giriş parametreleri" alt bölümünde bulabilirsiniz.

[3] Daha fazla bilgiyi, takip eden "Çıkış parametreleri: ERROR_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR" alt bölümünde bulabilirsiniz.

Parametre ve veri tipi		Veri tipi	Açıklama
RED_ERR_S7 [3]	OUT	Bool	• 0: SIMATIC'te yedeklilik hatası yok • 1: SIMATIC'te yedeklilik hatası
RED_ERR_S7 [3]	OUT	Bool	• 0: Link partneri tarafında yedeklilik hatası yok • 1: Link partneri tarafında yedeklilik hatası
RED_ERR_S7 [3]	OUT	Bool	• 0: En az 1 yapılandırılmış bağlantı kuruldu • 1: Tam iletişim kaybı, yapılandırılmış tüm bağlantılar sonlandırılır

[1] Daha fazla bilgiyi, takip eden "Lisanslama" alt bölümünde bulabilirsiniz:

[2] Daha fazla bilgiyi, takip eden "MB_HOLD_REG giriş parametreleri" alt bölümünde bulabilirsiniz.

[3] Daha fazla bilgiyi, takip eden "Çıkış parametreleri: ERROR_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR" alt bölümünde bulabilirsiniz.

NOT

Güvenlik uyarıları

Ağdaki her istemci, proses ekranındaki giriş ve çıkışlara ve Modbus tutma kaydında belirtilmiş olan veri bloğu veya Bit bellek alanlarına okuma ve yazma erişimine sahiptir. Bu opsiyonun amacı, bir IP adresine erişimi kısıtlamak ve yetkisiz okuma ve yazmayı önlemektir. Yine de paylaşılan adresin yetkisiz erişim için de kullanılabileceğini dikkate alınmalıdır.

Birden çok sunucu bağlantısı

CPU'lar şunları yapabilir:

- Birden çok sunucu bağlantısının işlenmesi
- Bir sunucu portunda aynı anda farklı istemcilerden gelen birden çok bağlantının kabul edilmesi

Maksimum bağlantı sayısı, kullanılan CPU'ya bağlı olup, CPU'nun teknik verilerinde bulunabilir. Modbus TCP istemcileri ve sunucularından olanlar da dahil olmak üzere bir CPU'dan yapılan toplam bağlantı sayısının, desteklenen maksimum bağlantı sayısını aşmasına izin verilmez.

Sunucu bağlantılarında aşağıdaki kuralları dikkate alın:

- Her MB_RED_SERVER bağlantısı, benzersiz bir örnek DB kullanmalıdır.
- Sunucu portuna bağlanmak isteyen her bir istemci için benzersiz bir bağlantı/bağlantı ID'si gerekir.
- Bağlantı ID'leri CPU genelinde benzersiz olmalıdır.

Modbus adreslerinin proses görünümüne atanması

MB_RED_SERVER yordamı, gelen Modbus fonksiyonlarının (1, 2, 4, 5 ve 15) CPU'nun proses görünümünün (BOOL ve WORD veri tiplerini kullanarak) giriş ve çıkışlarına doğrudan okuma ve yazma erişimi sağlar.

S7-1200 CPU'larda girişlerin proses görünümü ve çıkışların proses görünümü için adres bölümü 1 KB'dir.

Aşağıdaki tabloda, yukarıda listelenen Modbus fonksiyonlarının adres bölümü gösterilmektedir:

Modbus fonksiyonu					
Fonksiyon kodu	**Fonksiyon**	**Veri alanı**	**Adres bölümü**		
01	Oku: Bit	Output	0	-	65.535
02	Oku: Bit	Input	0	-	65.535
04	Oku: WORD	Input	0	-	65.535
05	Yaz: Bit	Output	0	-	65.535
15	Yaz: Bit	Output	0	-	65.535

3, 6, 16 ve 23 fonksiyon kodlarıyla gelen Modbus talepleri, Modbus tutma kayıtlarını yazar veya okur (tutma kaydını MB_HOLD_REG parametresi veya Data_Area_Array üzerinden belirtirsiniz).

Modbus fonksiyonları

Aşağıdaki tabloda, MB_RED_SERVER yordamı tarafından desteklenen tüm Modbus fonksiyonları listelenmektedir:

Fonksiyon kodu	Açıklama
01	Çıkış Bit'lerini oku
02	Giriş Bit'lerini oku
03	Tutma kaydını oku
04	Giriş sözcüklerini oku
05	Çıkış Bit'ini yaz
06	Tutma kaydını yaz
08	Teşhis fonksiyonu: • Yankı testi (kısmi fonksiyon 0x0000): MB_RED_SERVER yordamı bir veri sözcüğü alır ve bunu değiştirmeden Modbus istemcisine geri döndürür. • Olay sayacının sıfırlanması (kısmi fonksiyon 0x000A): MB_RED_SERVER yordamı şu olay sayaçlarını sıfırlar: "Success_Count", "Xmt_Rcv_Count", "Exception_Count", "Server_Message_Count" ve "Request_Count".
11	Teşhis fonksiyonu: İletişim olay sayacını çağır MB_RED_SERVER yordamı iletişim için bir dahili olay sayacı kullanarak, Modbus sunucusuna gönderilen başarıyla tamamlanmış okuma ve yazma taleplerinin sayısını saptar. Olay sayacının değeri, 8 ve 11 fonksiyonları için artırılmaz. Bu, örneğin bir protokol hatası oluştuğunda, iletişim hatasına neden olan talepler için de geçerlidir. Alınan Modbus talebindeki fonksiyon kodu desteklenmiyor.

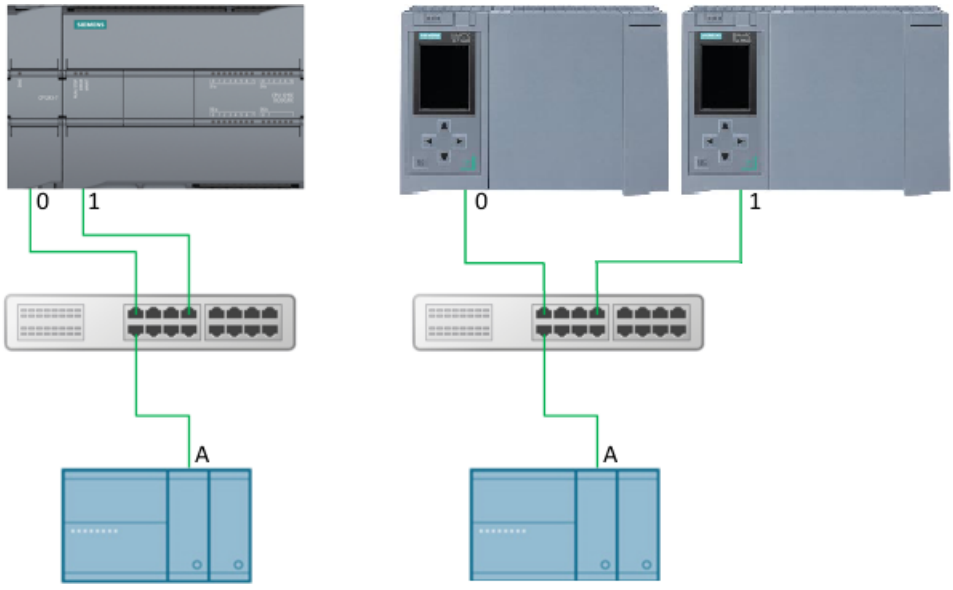
Fonksiyon kodu	Açıklama
15	Çıkış Bit'lerini oku
16	Tutma kaydını yaz
23	Bir taleple bir tutma kaydı yaz ve bir tutma kaydını oku

İşletim ve yedeklilik

İletişim katılımcıları, bağımsız veya yedekli katılımcılar olabilir. Partnerlerden birinin bağımsız çalışması, tek yönlü yedeklilik olarak adlandırılır. Her iki partner de bağımsız olarak yürütülüyorsa, buna iki yönlü yedeklilik denir:

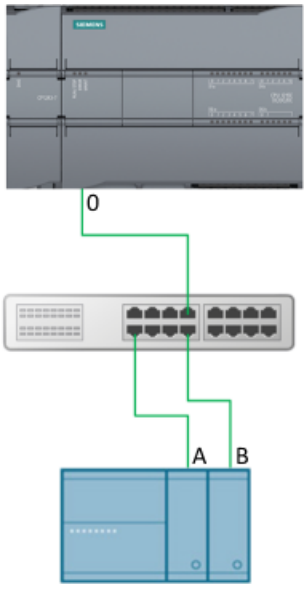
- Tek taraflı yedeklilik
  - İletişim partnerleri arasındaki her bağlantı için bir bağlantı yapılandırılmalıdır. **SIMATIC S7**'nin bağlantı noktaları, **0** ve **1** olarak adlandırılır. **İletişim partnerlerinin** bağlantı noktaları, **A** ve **B** olarak adlandırılır.

    R-CPU veya H-CPU'da 1, bağlantı noktası 0'ı ifade eder ve R-CPU veya H-CPU'da 2, bağlantı noktası 1'i anlamına gelir.
  - Konfigürasyon: S7 yedekliyse, S7 bağlantı noktası 0'dan link partnerinin A düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm **A**'ya bağlantı => bağlantı **0A**) ve S7 bağlantı noktası 1'den link partnerinin A düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **1**'den partner/düğüm **A**'ya bağlantı => bağlantı **1A**). Şekilde, bağlantı adları gösterilmektedir:



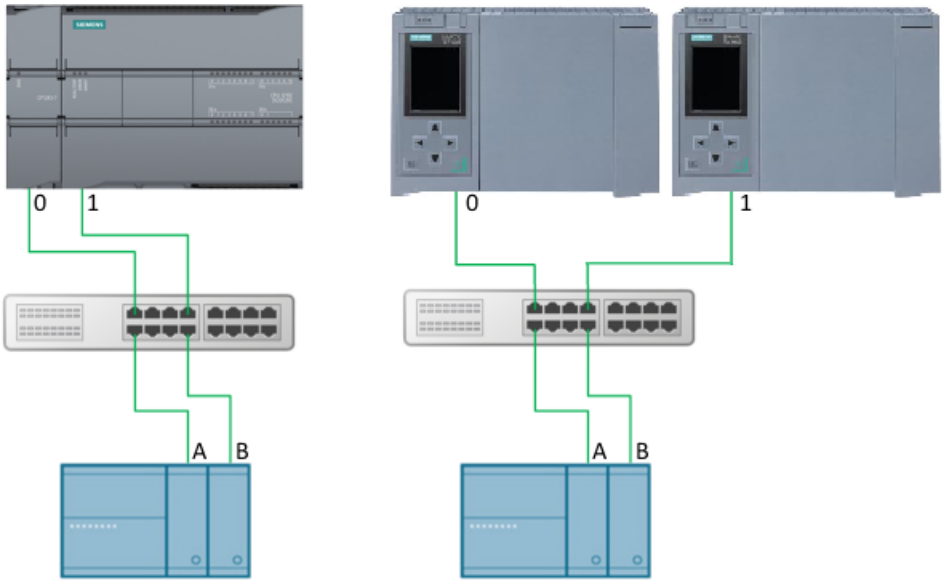
Resim 13-12 S7 tek taraflı yedeklilik

– S7 bağımsızsa ve link partneri yedekliyse, S7 bağlantı noktası 0'dan link partnerinin A düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm **A**'ya bağlantı => bağlantı **0A**) ve S7 bağlantı noktası 0'dan link partnerinin B düğüm noktasına bir bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm **B**'ye bağlantı => bağlantı **0B**). Şekilde, bağlantı adları gösterilmektedir:



Resim 13-13 Tek taraflı yedeklilik partneri

- İki yönlü yedeklilik:
  - Açıklama: İletişim partnerleri arasındaki her bağlantı için bir bağlantı yapılandırılmalıdır. **SIMATIC S7**'nin bağlantı noktaları, 0 ve **1** olarak adlandırılır. **İletişim partnerlerinin** bağlantı noktaları, **A** ve **B** olarak adlandırılır.
    R-CPU veya H-CPU'da 1, bağlantı noktası 0'ı ifade eder ve R-CPU veya H-CPU'da 2, bağlantı noktası 1'i anlamına gelir.
  - Konfigürasyon: İki taraflı yedeklilikte, bağlantı noktası 0'dan iki bağlantı oluşturulur (S7 bağlantı noktası **0**'dan partner/düğüm noktası **A**'ya bağlantı => bağlantı **0A** ve S7 bağlantı noktası **0**'dan partner/düğüm noktası **B**'ye bağlantı => bağlantı **0B**) ve S7'nin bağlantı noktası 1'den bağlantı partnerinin A ve B düğüm noktalarına iki bağlantı oluşturulur (S7 bağlantı noktası **1**'den partner/düğüm noktası **A**'ya bağlantı => bağlantı **1A** ve S7 bağlantı noktası **1**'den partner/düğüm noktası **B**'ye bağlantı => bağlantı **1B**). Şekilde, bağlantı adları gösterilmektedir:



Resim 13-14 İki yönlü yedeklilik:

- Telegram işleme: Yapılandırılmış tüm bağlantılar üzerinden telegramlar alınabilir. İstemci, tek bir bağlantı üzerinden veya tüm bağlantılar üzerinden telegram gönderebilir. Bir bağlantıda bir telegram alındıysa, CPU, durumu ilgili DR_NDR_x veya ERROR_x çıkışında görüntüler. Her bağlantı bağımsız olarak çalışır ve diğer bağlantıların görüntülenmesi üzerinde bir etkisi yoktur.

- RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR yedeklilik çıkışları:
  – RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR yedeklilik Bit'leri durum çıkışlarının durumlarına bağlı olarak ayarlanır:

Number of faulty connections	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	okay	okay	okay	FALSE	FALSE	FALSE
1	okay	okay	okay	Error	FALSE	FALSE	FALSE
	okay	okay	Error	okay	FALSE	FALSE	FALSE
	okay	Error	okay	okay	FALSE	FALSE	FALSE
	Error	okay	okay	okay	FALSE	FALSE	FALSE
2	okay	okay	Error	Error	TRUE	FALSE	FALSE
	okay	Error	okay	Error	FALSE	TRUE	FALSE
	Error	okay	okay	Error	FALSE	FALSE	FALSE
	okay	Error	Error	okay	FALSE	FALSE	FALSE
	Error	okay	Error	okay	FALSE	TRUE	FALSE
	Error	Error	okay	okay	TRUE	FALSE	FALSE
3	Error	Error	Error	okay	TRUE	TRUE	FALSE
	Error	Error	okay	Error	TRUE	TRUE	FALSE
	Error	okay	Error	Error	TRUE	TRUE	FALSE
	okay	Error	Error	Error	TRUE	TRUE	FALSE
4	Error	Error	Error	Error	TRUE	TRUE	TRUE

Resim 13-15 Her iki tarafta yedeklilik kurulumu için alarm Bit'lerinin görüntülenmesi

Number of faulty connections	STATUS_0A	STATUS_0B	STATUS_1A	STATUS_1B	RED_ERR_S7	RED_ERR_DEV	TOT_COM_ERR
0	okay	0AFF	okay	0AFF	FALSE	FALSE	FALSE
1	okay	0AFF	Error	0AFF	TRUE	TRUE	FALSE
	Error	0AFF	okay	0AFF	TRUE	TRUE	FALSE
2	Error	0AFF	Error	0AFF	TRUE	TRUE	TRUE

Resim 13-16 Yedeklilik kurulumu için alarm Bit'lerinin bir sayfada görüntülenmesi

NOT

İstemci ve sunucu için port numaraları

Modbus istemcisi 2000'den başlayan bir port numarası kullanır. Modbus sunucusu genellikle 502 numaralı portu üzerinden adreslenir. CPU'ya bağlı olarak, 502 numaralı portu çoklu bağlantılar (çoklu port) için yapılandırmak mümkündür. Yerel port 502'nin iki veya daha fazla bağlantı için yapılandırılması halinde, belirtilmemiş bağlantılar için talepte bulunan istemciler, mevcut sunucu bağlantıları arasında rastlantısal olarak atanır. "MB_RED_SERVER" yordamına bağlanmak isteyen ilk istemciye otomatik olarak 0A bağlantısı atanmaz. İstemci taleplerinin sunucu bağlantılarına atanmasından sonra, ilgili atamalar, bağlantı sonlandırılana dek telegram alıp verme süresince kalır.

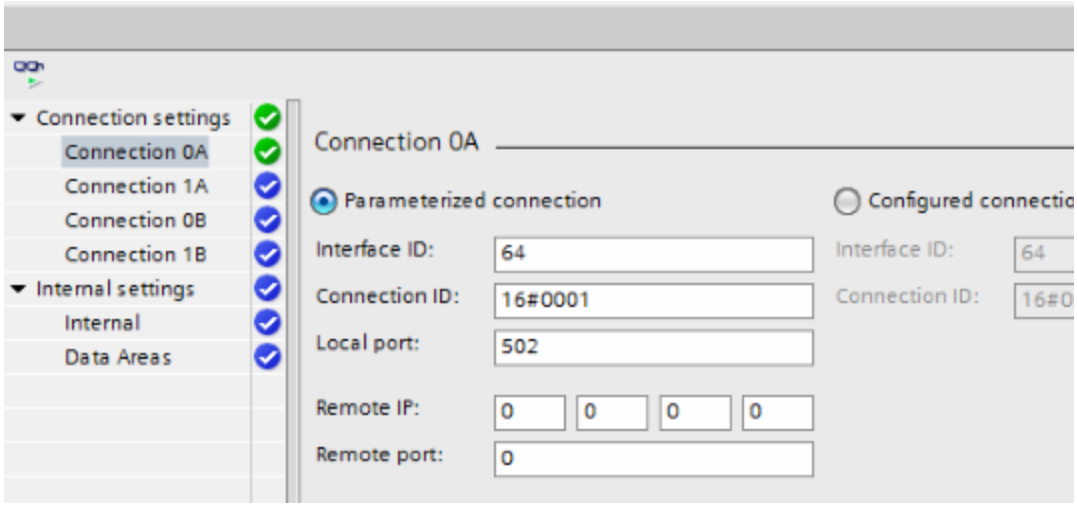
Parametrelendirme

S7-1200'de MB_RED_SERVER **V1.0** ve **V1.1** yordamını kullanabilirsiniz. CPU, bağlantıları CPU'nun yerel arayüzü veya CM/CP aracılığıyla uygular. CPU, bağlantıları TCON_IP_V4 yapısı üzerinden yapılandırır ve kurar.

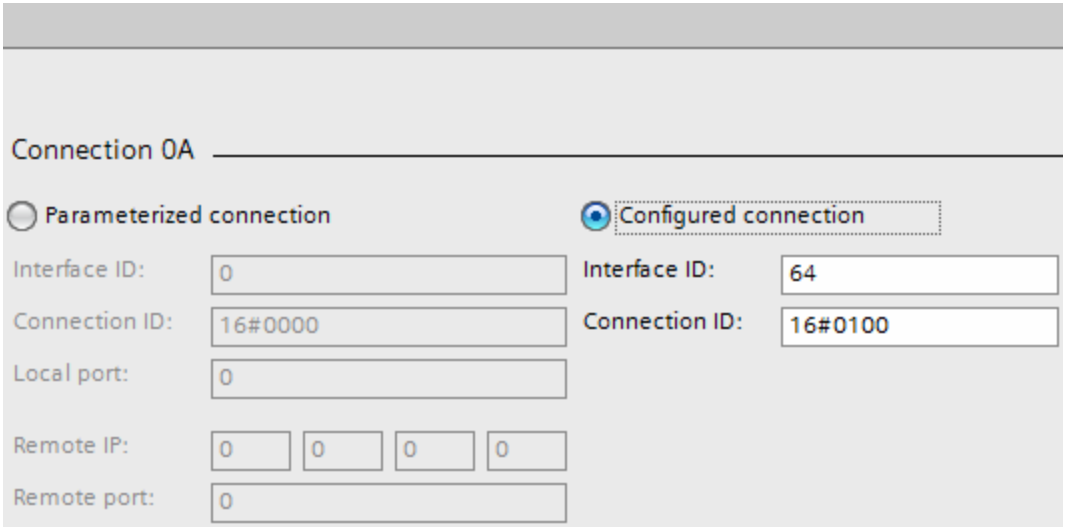
MB_RED_SERVER konfigürasyonu: MB_RED_SERVER yordamının konfigürasyon iletişim kutusunda aşağıdaki ayarları yapın:

- 0A, 1A, 0B ve 1B bağlantıları için bağlantı parametreleri (yedeklilik konfigürasyonu hakkında daha fazla bilgiyi, yukarıdaki "Çalışma ve Yedeklilik" bölümünde bulabilirsiniz)
- Dahili parametreler (opsiyonel)

Konfigürasyon iletişim kutusunu, MB_RED_SERVER yordamıyla veya teknoloji nesneleri üzerinden açabilirsiniz:

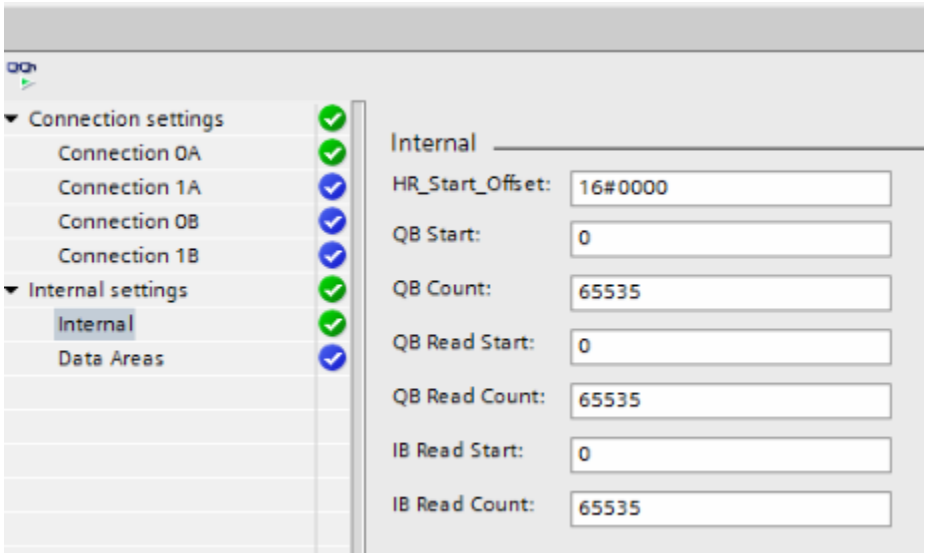


Resim 13-17 Parametrelendirilmiş sunucu bağlantısı



Resim 13-18 Yapılandırılmış sunucu bağlantısı

Değişken	Başlangıç değeri	Açıklama
Yapılandırılmış bağlantılar		
Interface ID	64	Kullanılan PN arayüzünün donanım kodu
Connection ID	16#0000	Kullanılan bağlantılar için bağlantı ID'leri. Bu bağlantı ID'leri, CPU genelinde benzersiz olmalıdır.
Local port	502	Sunucu bloğunun yerel port numarası. Modbus/TCP sunucu için standart port, 502'dir.
Remote IP	0.0.0.0	İstemcinin uzak IP adresi. Standart olarak, istemciye IP adresi girilmez.
Remote port	0	İstemcinin uzak port numarası. Standart olarak, istemciye port numarası girilmez.
Yapılandırılmış bağlantılar		
Interface ID	64	Kullanılan PN arayüzünün donanım kodu
Connection ID	16#0000	Kullanılan bağlantılar için bağlantı ID'leri. Bu bağlantılar, ağ görünümünde yapılandırılır.



Resim 13-19 Dahili parametreler (opsiyonel)

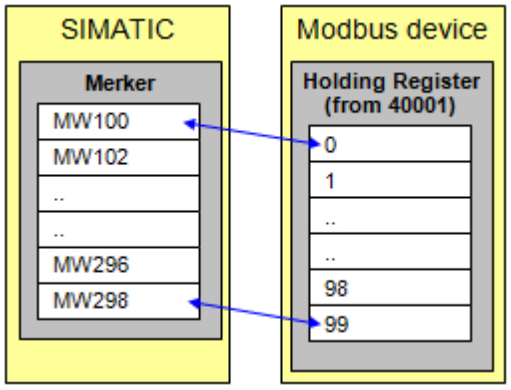
Değişken	Veri tipi	Başlangıç değeri	Açıklama
HR_Start_Offset	WORD	0	Modbus tutma kaydı başlangıç adresinin atanması
QB_Start	UINT	0	Modbus Master'ın yazabileceği çıkışların izin verilen adres aralığının başlangıç adresi (Bayt 0 - 65535)
QB_Count	UINT	0	Modbus Master'ın yazabileceği çıkış Bayt'ı sayısı *Örnek:* • QB_Start = 0 ve QB_Count = 10: Modbus Master, 0 ila 9 çıkış Bayt'larına yazabilir. • QB_Count = 0: Modbus Master herhangi bir çıkış Bayt'ına yazamaz.
QB_Read_Start	UINT	0	Modbus Master'ın okuyabileceği çıkışların izin verilen adres aralığının başlangıç adresi (Bayt 0 - 65535)
QB_Read_Count	UINT	0	Modbus Master'ın okuyabileceği çıkış Bayt'ı sayısı *Örnek:* • QB_Read_Start = 0 ve QB_Read_Count = 10: Modbus Master, 0 ila 9 çıkış Bayt'larını okuyabilir. • QB_Read_Count = 0: Modbus Master, çıkış Bayt'larını okuyabilir.
IB_Read_Start	UINT	0	Modbus Master'ın okuyabileceği girişlerin izin verilen adres aralığının başlangıç adresi (Bayt 0 - 65535)
IB_Read_Count	UINT	0	Modbus Master'ın okuyabileceği giriş Bayt'ı sayısı *Örnek:* IB_Read_Start = 0 ve IB_Read_Count = 10: Modbus Master, 0 ila 9 giriş Bayt'larını okuyabilir. IB_Read_Count = 0: Modbus Master, giriş Bayt'ını okuyamaz.
Data_Area_Array	ARRAY [1..8]		
data_type	UINT	0	Veri tipi: 0 ila 4

Değişken	Veri tipi	Başlangıç değeri	Açıklama
db	UINT	0	Veri bloğu numarası
start	UINT	0	Veri bloğundaki ilk Modbus adresi
length	UINT	0	Veri bloğundaki Modbus değerlerinin sayısı

Statik değişkenleri üzerinden adresleme HR_Start_Offset

Modbus tutma kaydı adresleri 0'dan başlar.

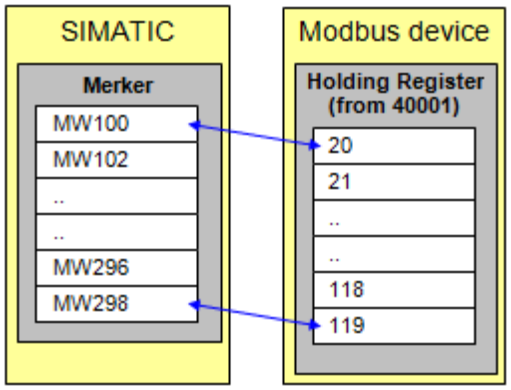
Örnek: MW100'de başlayan ve 100 sözcük uzunluğunda olan bir tutma kaydı.



Resim 13-20 HR_Start_Offset_0

HR_Start_Offset değişkenini, Modbus tutma kaydının 0'dan farklı bir başlangıç adresi olacak şekilde tanımlayabilirsiniz.

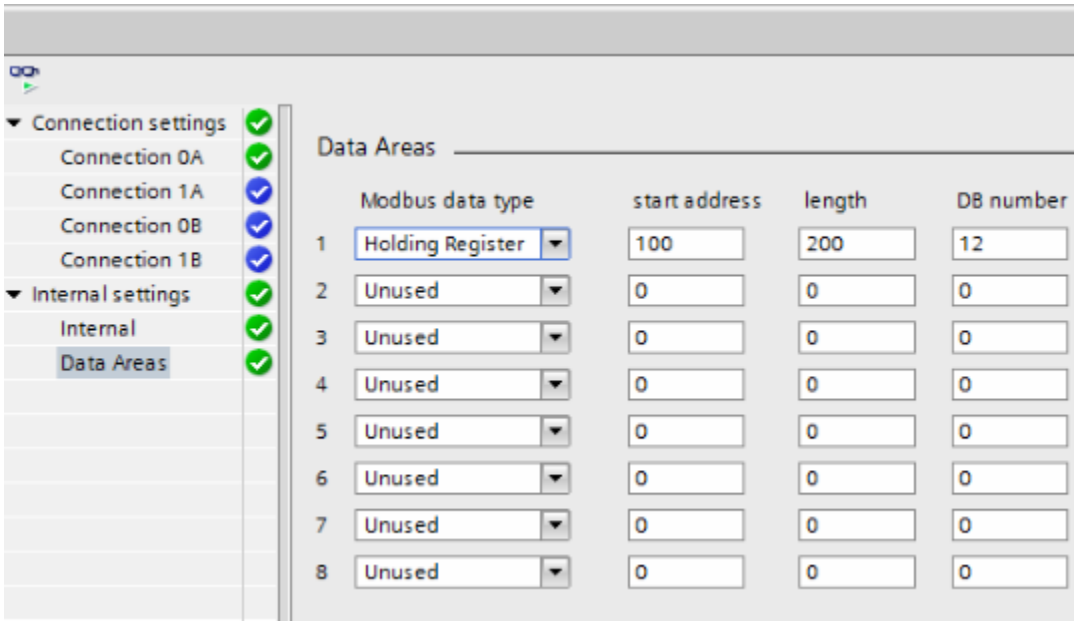
Örnek: HR_Start_Offset parametresindeki ofset değerinin 20 olmasının anlamı, tutma kaydının başlangıç adresinin 0'dan 20'ye kaydırılmış olmasıdır. Bu, tutma kaydını adres 20'nin altında ve adres 119'un üzerinde adreslediğinizde her zaman bir hataya yol açar.



Resim 13-21 HR_Start_Offset_20

Data_Area_Array [1..8]

SIMATIC S7 belleğinde MODBUS adreslerini atamak için sekiz veri alanı bulunmaktadır. Veri alanı "Tutma kaydı" veri tipi ile tanımlanırsa, MB_HOLD_REG parametresi değerlendirilmez. Bunun yerine Modbus Master, görevin türüne bağlı olarak Modbus kaydını ve veri bloklarındaki Bit'leri yazar veya okur. CPU, programın takip eden yürütülmesinde bu değerleri işlemeye devam edebilir.



Resim 13-22 Sunucu veri alanları

Her bir görev ile yalnızca bir DB'den okuyabilir veya bir DB'ye yazabilirsiniz. Farklı DB'lerde bulunan kayıtlara veya Bit değerlerine erişim, sayılar birbirini boşluksuz takip etse bile iki göreve bölüştürülmelidir. Konfigürasyon sırasında bunu aklınızda bulundurun. Bir veri bloğunda, Modbus Master'ın bir telegramda işleyebileceğinden daha fazla Modbus alanı (kayıtlar veya Bit değerleri) tahsis etmek mümkündür.

data_type

data_type parametresi, Modbus Master'ın bu veri bloğunda hangi MODBUS veri tiplerini atadığını gösterir. data_type'a "0" değeri girilirse, Modbus Master söz konusu veri alanını kullanmaz. Modbus Master birden çok Data_Area kullanacaksa, bunları art arda tanımlamanız gerekir. Modbus Master, data_type = 0'dan sonra herhangi bir girişi işlemez.

Kod	Veri tipi	Açıklama
0	Alan kullanılmıyor	
1	Çıkış Bit'leri (bobinler)	Bit
2	Giriş Bit'leri (girişler)	Bit
3	Tutma kaydı	Sözcük
4	Giriş sözcükleri (giriş kaydı)	Sözcük

db

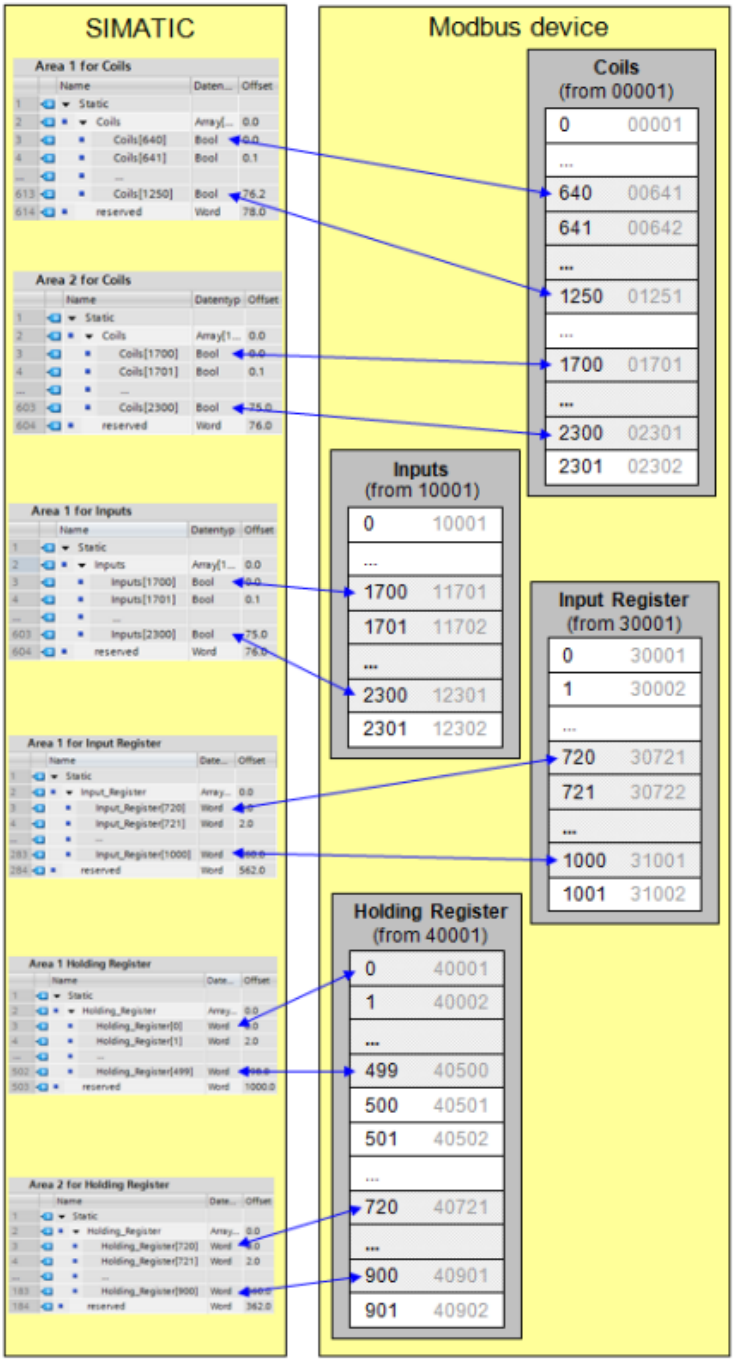
db parametresi aşağıda tanımlanan MODBUS kayıtlarını veya Bit değerlerini atayan veri bloğunu belirtir. CPU, 0 numaralı DB'ye izin vermez, çünkü bu, sistem için rezerve edilmiştir.

start, length

start parametresi, Modbus Master'ın DB'nin veri sözcüğü 0'a atadığı ilk Modbus adresini belirtir. length parametresi uzunluğu, Modbus Master'ın veri bloğunda atadığı MODBUS

adreslerinin sayısını tanımlar. Tanımlanan veri alanları çakışmasına izin verilmez. length parametresi, 0 olamaz.
Örnek: Data_Area_Array ile adres ataması

Veri alanı 1	data_type	3: Tutma kaydı
	db	11
	start	0
	length	500
Veri alanı 2	data_type	3: Tutma kaydı
	db	12
	start	720
	length	181
Veri alanı 3	data_type	4: Giriş sözcükleri
	db	13
	start	720
	length	281
Veri alanı 4	data_type	1: Çıkış Bit'leri
	db	14
	start	640
	length	611
Veri alanı 5	data_type	2: Giriş Bit'i
	db	15
	start	1700
	length	601
Veri alanı 6	data_type	1: Çıkış Bit'leri
	db	16
	start	1700
	length	601
Veri alanı 7	data_type	Kullanılmıyor
	db	0
	start	0
	length	0
Veri alanı 8	data_type	Kullanılmıyor
	db	0
	start	0
	length	0



Resim 13-23 Adres yapısı

Lisanslama

MB_RED_SERVER yordamı belli bir ücrete tabidir ve bu yordamı her bir CPU için ayrı ayrı lisanslamanız gerekmektedir. Lisanslama, iki adımda gerçekleşir:

- Lisansın IDENT_CODE'unun görüntülenmesi
- REG_KEY kayıt anahtarının girilmesi: REG_KEY kayıt anahtarını, her MB_RED_SERVER yordamında girmeniz gerekir. REG_KEY'i, tüm MB_RED_SERVER yordamlarının gereken kayıt anahtarını aldıkları bir global veri bloğunda saklayın.

Lisansın IDENT_CODE'unu görüntüleme prosedürü:

1. MB_RED_SERVER yordamını bir periyodik OB'de kendi gerekliliklerinize göre parametrelendirin. Programı CPU'ya yükleyin ve CPU'yu RUN işletim durumuna getirin.
2. Modbus yordamının örnek DB'sini açın ve "Tümünü izle" butonuna tıklayın.
3. Örnek DB, IDENT_CODE çıkışında 18 karakterlik bir karakter dizisi görüntüler.

		Name	Data type	Start value	Monitor value
1		Input			
2		REG_KEY	String[17]	''	''
3		USE_ALL_CONN	Bool	false	FALSE
4		REQ	Bool	false	FALSE
5		DISCONNECT	Bool	false	FALSE
6		MB_MODE	USInt	0	0
7		MB_DATA_ADDR	UDInt	0	0
8		MB_DATA_LEN	UInt	0	0
9		Output			
10		LICENSED	Bool	false	FALSE
11		IDENT_CODE	String[18]	''	'RTPCFIGDCDIIHJHAH4'
12		DONE	Bool	false	FALSE
13		BUSY	Bool	false	FALSE
14		ERROR	Bool	false	FALSE

Resim 13-24 Lisans

4. Kopyala/yapıştır fonksiyonunu kullanarak, bu karakter dizisini veri bloğundan kopyalayın ve forma yapıştırın (ürünü sipariş ettikten sonra e-posta ile aldığınız veya ayrıca CD'de bulunan).
5. Bu formu bir hizmet isteği üzerinden Müşteri desteği (https://support.industry.siemens.com/my/ww/en/requests/#createRequest)'a gönderin. Bunun ardından, size CPU'nuz için kayıt anahtarı gönderilecektir.

REG_KEY kayıt anahtarını girme prosedürü:

1. "Yeni blok ekle..." yoluyla benzersiz bir sembolik ada sahip yeni bir global veri bloğu ekleyin, örneğin "Lisans-DB".
2. Bu blokta, STRING[17] veri tipiyle bir REG_KEY parametresi oluşturun.

LICENSE_DB

		Name	Data type	Offset	Start value
1		Static			
2		REG_KEY	String[17]	0.0	''

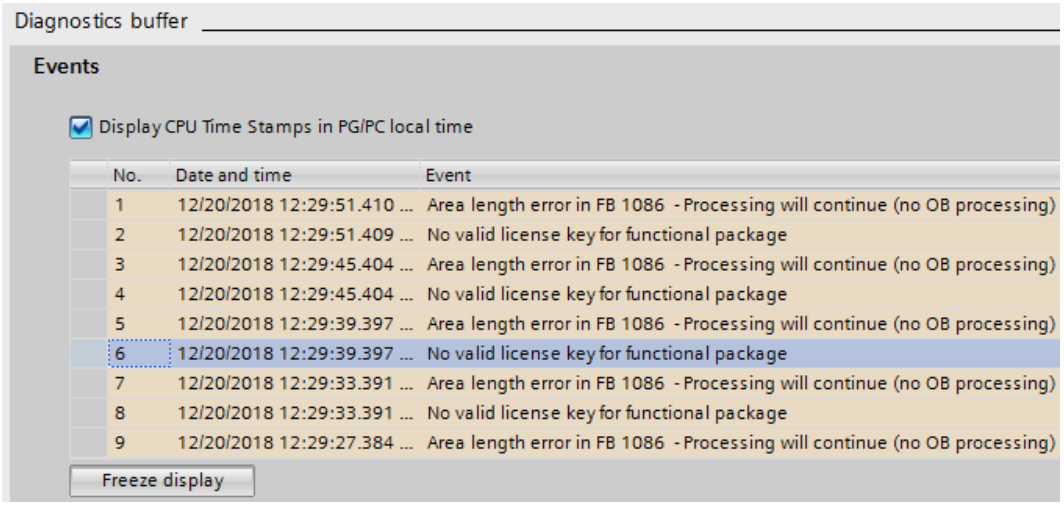
Resim 13-25 REG_KEY

3. Aktarılan 17 basamaklı kayıt anahtarını "Başlangıç değeri" sütununa kopyalamak için Kopyala/yapıştır fonksiyonunu kullanın.

4. MB_RED_SERVER yordamının REG_KEY parametresinde periyodik OB'de lisans DB'sinin adını ve karakter dizisinin adını (örn. License_DB.REG_KEY) girin.
5. Değiştirilmiş veri bloklarını CPU'ya yükleyin, çalışma zamanı boyunca kayıt anahtarını girebilirsiniz. Bunun için CPU'yu STOP'tan RUN işletim durumuna geçirmek gerekli değildir.
6. Artık Modbus/TCP iletişimi MB_RED_SERVER yordamı üzerinden bu CPU için lisanslanmıştır, LICENSED çıkış Bit'i TRUE'dur.

Eksik veya hatalı lisanslamayı düzeltme prosedürü:

- Yanlış bir kayıt anahtarı girerseniz veya herhangi bir kayıt anahtarı girmezseniz, CPU'nun ERROR LED'i yanıp söner. Buna ek olarak, S7-1200'ün CPU'su, eksik lisansla ilgili olarak teşhis tamponuna bir periyodik veri girişi yapar.



Resim 13-26 Teşhis tamponu

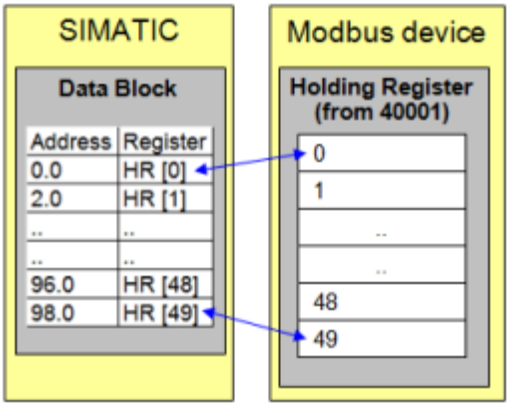
- Eksik veya hatalı kayıt anahtarında CPU, Modbus TCP iletişimini işler. Buna karşın CPU STATUS_x çıkışında her zaman "W#16#0A90" (fonksiyon paketi için geçerli lisans anahtarı yok) görüntüler. LICENSED çıkış Bit'i FALSE'tır.

MB_HOLD_REG giriş parametresi

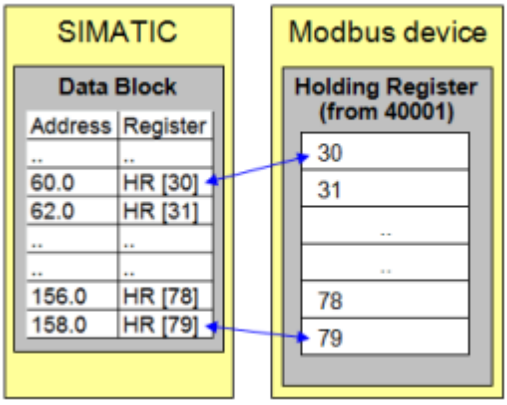
MB_HOLD_REG parametresi, bir Modbus istemcisinin okuma veya yazma erişimine sahip olduğu verileri saklamak için kullanılan bir veri tamponunu belirten işaretçidir. Bellek alanı olarak bir global veri bloğu (D) veya bir markör (M) kullanabilirsiniz:

- Veri bloğundaki (D) adres sayısının üst limiti, CPU'nuzdaki bir DB'nin maksimum boyutuna bağlıdır.
- Markör sayısı (M) için üst limit, CPU'nuzdaki maksimum markör alanına bağlıdır.

Aşağıdaki görseller, Modbus fonksiyonları 3 (birden çok WORD oku), 6 (bir WORD yaz), 16 (birden çok WORD yaz) ve 23 (birden çok WORD yaz ve oku) için Modbus adreslerinin tutma kaydına atanmasına ilişkin bazı örnekler içermektedir.



Resim 13-27 MB_HOLD_REG: Ofseti 0 olan veri bloğu



Resim 13-28 MB_HOLD_REG: Ofseti 60 olan veri bloğu

Data_Area_Array [1..8]: Opsiyonel Data_Area_Array [1..8], parametrelerinin kullanımı hakkında bilgi için yukarıdaki "Parametrelendirme" alt bölümüne bakın.

Çıkış parametreleri: ERROR_x, STATUS_x, RED_ERR_S7, RED_ERR_DEV ve TOT_COM_ERR

CPU, MB_RED_SERVER yordamının durum çıkışlarında hata mesajları görüntüler:

NOT

İlgili hata durum kodları, program editöründe tamsayı değerler veya onaltılık değerler olarak görüntülenebilir:

1. Programlama editöründe istenen bloğu açın.
2. "İzleme açık/kapalı" üzerine tıklayarak programlama durumunu açın. (Henüz bir çevrimiçi bağlantı kurmadıysanız, "Çevrimiçi ol" iletişim kutusu açılır. Bu iletişim kutusunda bir çevrimiçi bağlantı kurabilirsiniz.)
3. İzlemek istediğiniz değişkeni seçin ve bağlam menüsündeki "Görüntüleme formatı" altında istediğiniz görüntüleme formatını seçin.

STATUS_x parametresi (genel durum bilgileri)

STATUS (W#16#)	Açıklama
0000	Yordam hatasız bir şekilde yürütüldü.
0001	Bağlantı kuruldu.
0003	Bağlantı sonlandırıldı.
0A90	MB_RED_SERVER yordamı lisanslı değil. Daha fazla bilgiyi, yukarıdaki "Lisanslama" alt bölümünde bulabilirsiniz.
0AFF	Bağlantı yapılandırılmadı ve kullanılmıyor. 0A bağlantısının yapılandırılması gerekiyor.
7000	Aktif çağrı yok ve bağlantı kurulmadı (REQ = 0, DISCONNECT = 1).
7001	İlk çağrı: Bağlantı kurma başlatıldı.
7002	Ara çağrı. Bağlantı kuruluyor.
7003	Bağlantı sonlandırılır.
7005	Veriler gönderiliyor.
7006	Veriler alınıyor.

STATUS_x parametresi (protokol hatası)

STATUS (W#16#)	MB_RED_SERVER'in hata mesajındaki hata kodu (B#16#)	Açıklama
8380	-	Alınan Modbus telegramı yanlış formatta veya çok az Bayt alındı.
8381	01	Fonksiyon kodu desteklenmiyor.
8382	03	Veri uzunluğunda hata • Alınan Modbus telegramında geçersiz uzunluk belirtimi. • Modbus telegramının başlığındaki telegram uzunluğu, fiilen alınan Bayt sayısına karşılık gelmiyor. • Modbus telegramının başlığındaki Bayt sayısı, fiilen alınan Bayt sayısına karşılık gelmiyor (fonksiyon 15 ve 16).

STATUS (W#16#)	MB_RED_SERVER'in hata mesajındaki hata kodu (B#16#)	Açıklama
8383	02	Veri adresinde hata veya tutma kaydının adres aralığı dışında erişim (MB_HOLD_REG parametresi). Daha fazla bilgi için yukarıdaki "MB_HOLD_REG" alt bölüme bakın.
8384	03	Veri değerinde hata (fonksiyon 05)
8385	03	Teşhis kodu desteklenmiyor (fonksiyon kodu 08)

STATUS_x parametresi (parametre hatası)

STATUS (W#16#)	Açıklama
80BB	ActiveEstablished parametresi geçersiz bir değere sahip. Sunucu için yalnızca pasif bağlantı kurulmasına izin verilir (active_established = FALSE).
8187	MB_HOLD_REG parametresi geçersiz bir işaretçiye sahip. Veri alanı çok küçük.
8389	Geçersiz veri alanı tanımı: • Geçersiz data_type değeri • DB numarası geçersiz veya mevcut değil: – Geçersiz db değeri – DB numarası mevcut değil – DB numarası halen başka bir veri alanı tarafından kullanılıyor – Optimize edilmiş erişime sahip DB – DB, çalışma belleğinde değil • Geçersiz length değeri • Aynı MODBUS veri tipine ait çakışan MODBUS adres aralıkları

NOT

Dahili olarak kullanılan iletişim yordamlarından gelen hata kodları

MB_RED_SERVER yordamında, tablolarda listelenen hatalara ek olarak, yordam tarafından kullanılan iletişim yordamlarından ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" ve "T_RESET") kaynaklanan hatalar da ortaya çıkabilir.

Hata kodları, MB_RED_SERVER yordamının örnek veri bloğu üzerinden atanır. Hata kodları, STATUS altındaki münferit örneklerin "Static" alt bölümünde ilgili yordamlar için gösterilmektedir.

Hata kodlarının anlamı, ilgili iletişim yordamının dokümantasyonunda mevcuttur.

NOT

Veri gönderirken veya alırken iletişim hatası

Veri gönderirken veya alırken bir iletişim hatası ortaya çıkarsa, CPU mevcut bağlantıyı sonlandırır. Aşağıdaki hatalar ortaya çıkabilir:

- 80C4 - Geçici iletişim hatası; belirtilen bağlantı geçici olarak sonlandırılır.
- 80C5 - Uzak partner aktif olarak bağlantıyı sonlandırdı.
- 80A1 – Belirtilen bağlantı sonlandırıldı veya henüz kurulmadı.

Bunun anlamı, bağlantı kesildiğinde döndürülen herhangi bir STATUS değerini görebileceğiniz ve bağlantının kesilmesine neden olan STATUS kodunun yalnızca bağlantı kesildiğinde geri döndürüldüğüdür.

Örnek: Veri alırken geçici bir iletişim hatası oluşursa, önce STATUS 7003 (ERROR = False) ve bunun ardından 80C4 (ERROR = True) yayınlanır.

13.5.2.3 MODBUS TCP ile ilgili örnekler

Modbus TCP sunucusunda birden fazla bağlantınız olabilir. Bunun için MB_SERVER her bağlantı için bağımsız olarak çalıştırılmalıdır. Her bağlantı bağımsız bir örnek DB, bağlantı ID'si ve IP portu kullanmalıdır. S7-1200 için IP portu başına sadece bir bağlantıya izin verilir.

Optimum işletim performansı elde etmek için MB_SERVER her bir bağlantı için her program çevriminde yürütülmelidir.

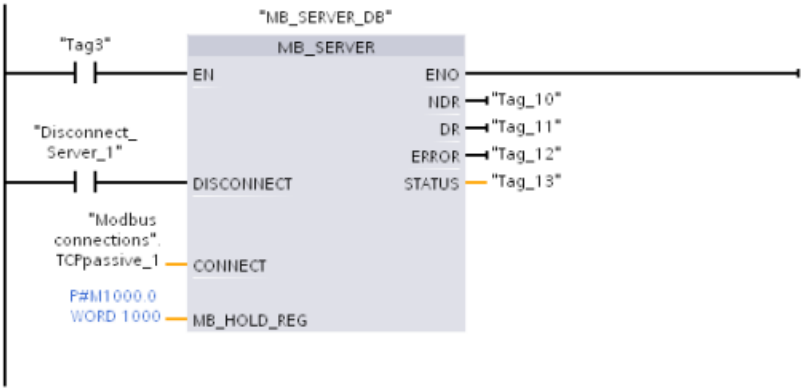
CONNECT parametresi, TCON_IP_V4 sistem veri tipini kullanır. Bu örnekte, ilgili veri yapıları "Modbus bağlantıları" DB'sinde bulunmaktadır. "Modbus bağlantıları" DB'si, iki TCP_IP_V4 yapısı içerir: "TCPpassiv_1" (bağlantı 1 için) ve "TCP_passiv_2" (bağlantı 2 için). Ağ yorumlarında sözü edilen bağlantı özelliği ID'si ve LocalPort, CONNECT veri yapısında saklanan veri elemanlarıdır.

TCP_IP_V4 CONNECT verileri de ADDR Array'in RemoteAddress'i içinde bir IP adresi de içermektedir. TCPpassiv_1 ve TCP_passiv_2'deki IP adresi atamalarının, TCP sunucu bağlantılarının kurulması üzerinde bir etkisi yoktur, ancak Modbus TCP istemcilerinin hangilerinin bağlantılar üzerinden her bir MB_SERVER ile iletişim kurabileceğini belirler. MB_SERVER bir Modbus istemci mesajını pasif olarak denetler ve mesajın gelen IP adresini ilgili RemoteAddress ADDR-Array'de saklanmakta olan IP adresiyle karşılaştırır.

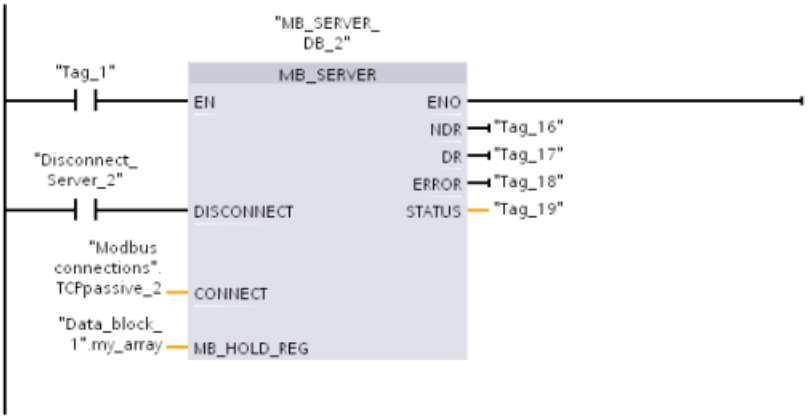
İki MB_SERVER yordamı için üç IP adresi varyantı mümkündür:

- **IP adresi = 0.0.0.0**
  Her bir MB_SERVER yordamı, herhangi bir IP adresine sahip tüm Modbus TCP istemcilerine yanıt verir.
- **IP adresi = TCPpassive_1 ve TCPpassive_2'deki aynı IP adresi**
  Her iki MB_SERVER bağlantısı da yalnızca bu IP adresinden gelen Modbus istemcilerine yanıt verir.
- **IP adresi = TCPpassive_1 ve TCPpassive_2'de farklı IP adresi**
  Her MB_SERVER yordamı yalnızca kendi TCP_IP_V4 verilerinde saklanan IP adresinden gelen Modbus istemcilerine yanıt verir.

Ağ 1: Bağlantı 1, örnek DB = "MB_SERVER_DB", "Modbus bağlantıları.TCPpassiv_1" içinde (ID = 1 ve LocalPort = 502)



Ağ 2: Bağlantı 2, örnek DB = "MB_SERVER_DB_1", "Modbus bağlantıları.TCPpassiv_2" içinde (ID = 2 ve LocalPort = 503)

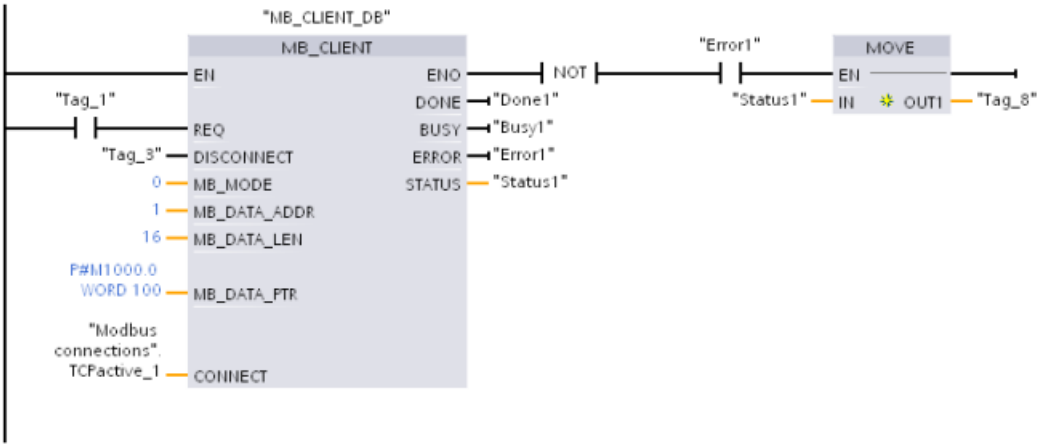


Aynı bağlantı üzerinden birden çok sayıda Modbus istemci talebi gönderilebilir. Bunun için aynı örnek DB, bağlantı ID'si ve port numarasını kullanın.

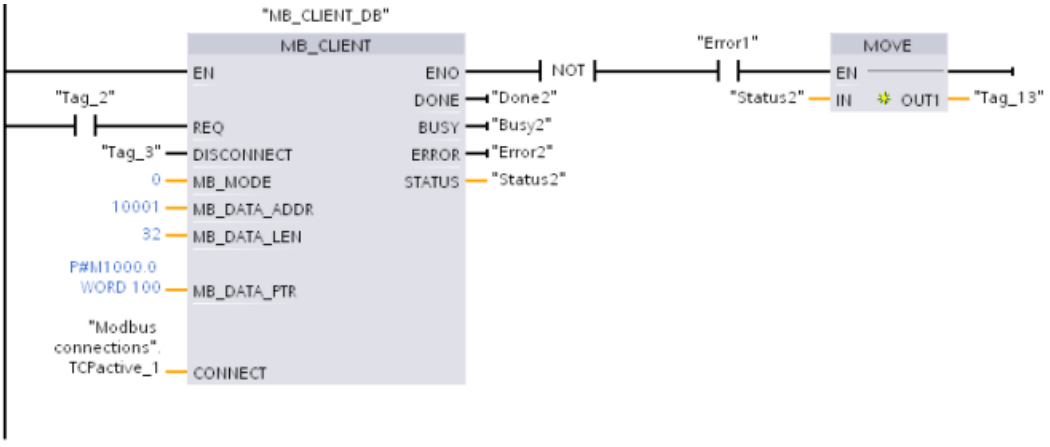
Her iki MB_CLIENT kutusunun CONNECT ("Modbus_bağlantıları".TCPactive_1) parametresinde aynı TCON_IP_v4 veri yapısını kullanması nedeniyle bağlantı ID'si, port numarası ve IP adresi aynıdır. CONNECT'in IP adresi verileri, hedef Modbus TCP sunucusunun IP adresini atar.

Bir seferde yalnızca bir MB_CLIENT yordamı aktif olabilir. Bir istemcinin yürütmeyi tamamlamasından sonra, bir sonraki istemci yürütmeye başlayabilir. Yürütme sıralamasından program lojiğiniz sorumludur. Bu örnekte, her iki istemcinin de tek bir Modbus istemcisinden desantral verileri okuduğunu ve verileri Modbus istemcisinin CPU'suna aktardığı gösterilmektedir (markörlerin bellek alanı M1000.0'dan başar). Yayınlanan bir hata algılanır (opsiyonel fonksiyon).

Ağ 1: Modbus fonksiyonu 1 - "Modbus bağlantıları".TCPactive_1'de atanan IP adresine sahip bir Modbus TCP sunucusundan 16 çıkış Bit'ini okur.



Ağ 2: Modbus fonksiyonu 2 - "Modbus bağlantıları".TCPactive_1'de atanan IP adresine sahip bir Modbus TCP sunucusundan 32 giriş Bit'ini okur.



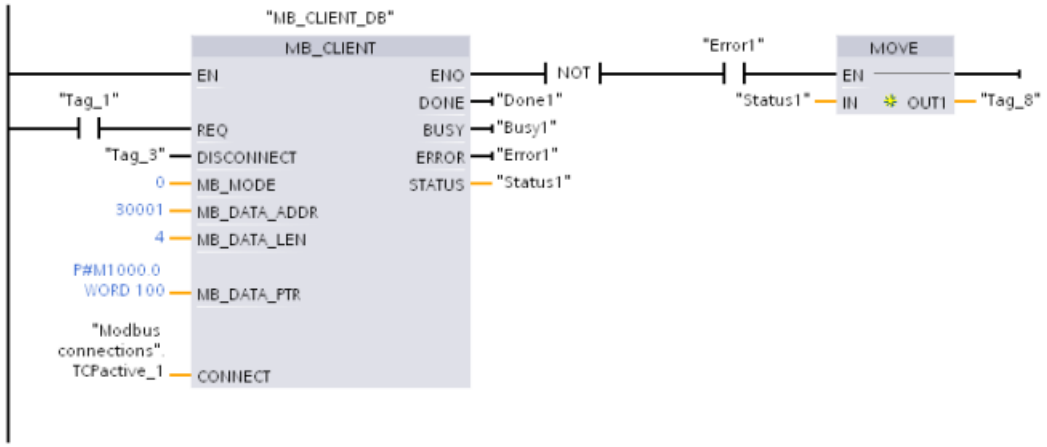
Birden çok Modbus TCP istemci talebi farklı bağlantılar üzerinden gönderilebilir. Bunun için farklı örnek DB'leri ve bağlantı ID'leri kullanmalısınız.

Aynı Modbus sunucusuna bağlantı kuruluyorsa, RemotePort numarası (IP portu) farklı olmalıdır. Farklı sunuculara bağlantılar için IP port numarasında herhangi bir kısıtlama yoktur.

Bu örnekte, iki Modbus TCP istemcisinin iki farklı Modbus TCP sunucusundan desantral verileri alarak aynı yerel CPU bellek alanında M1000.0 adresinden başlayarak aktardığı gösterilmektedir. Ayrıca, yayınlanan bir hata yakalanır (opsiyonel fonksiyon).

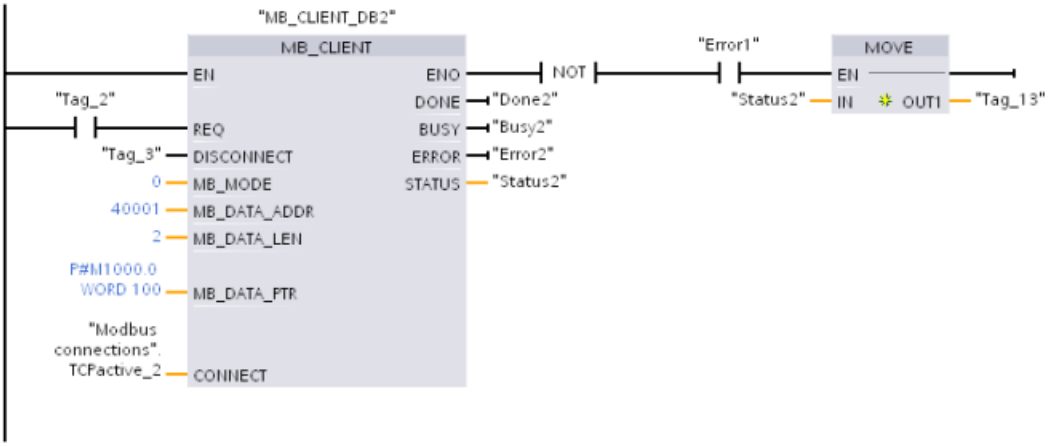
Ağ 1: Modbus fonksiyonu 4 - Bir Modbus TCP sunucusundan gelen girişlerin proses görünümünden sözcükleri okur.

Parametre CONNECT = "Modbus bağlantıları".TCPactive_1: Bağlantı **ID'si** = 1, **RemoteAddress** = 192.168.2.241, **RemotePort** = 502



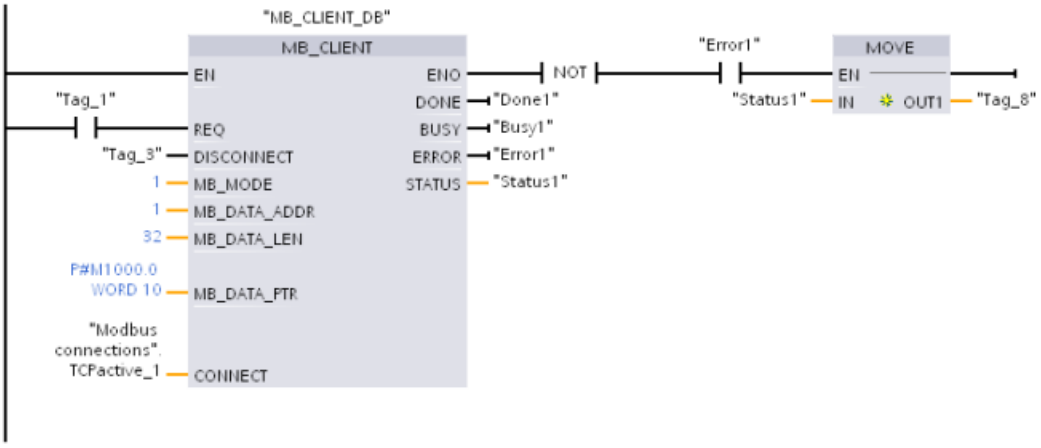
Ağ 2: Modbus fonksiyonu 3 - Bir Modbus TCP sunucusundan tutma kaydındaki sözcükleri okur.

CONNECT parametresi = "Modbus bağlantıları".TCPactive_2: Bağlantı **ID'si** = 2, **RemoteAddress** = 192.168.2.242, **RemotePort** = 502



Bu örnekte, Bit verilerini yerel CPU belleğinden (M1000.0'dan başlayarak) bir desantral Modbus TCP sunucusuna aktaran bir Modbus istemci talebi ele alınmaktadır.

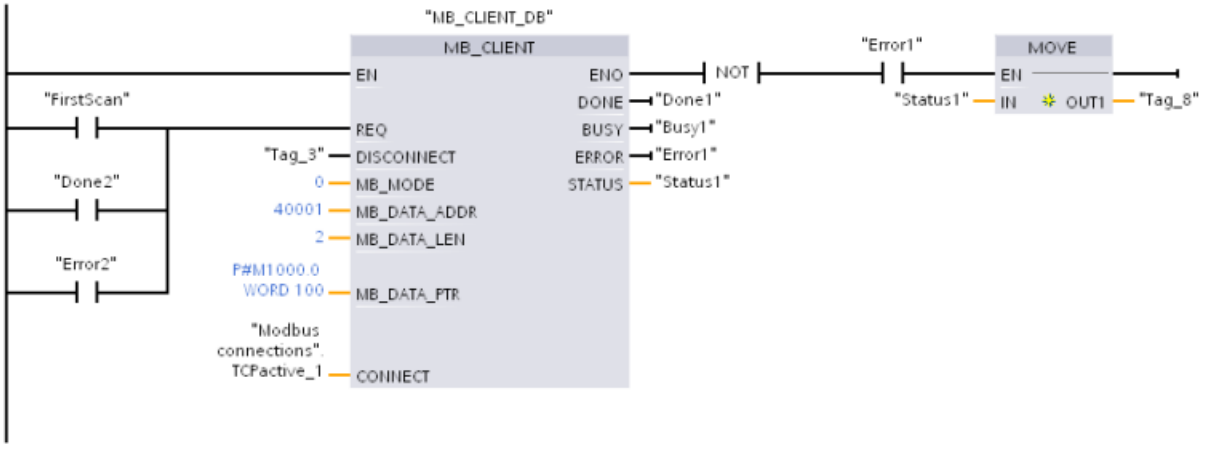
Ağ 1: Modbus fonksiyonu 15 - çıkış Bit'lerinin bir Modbus sunucusuna yazılması



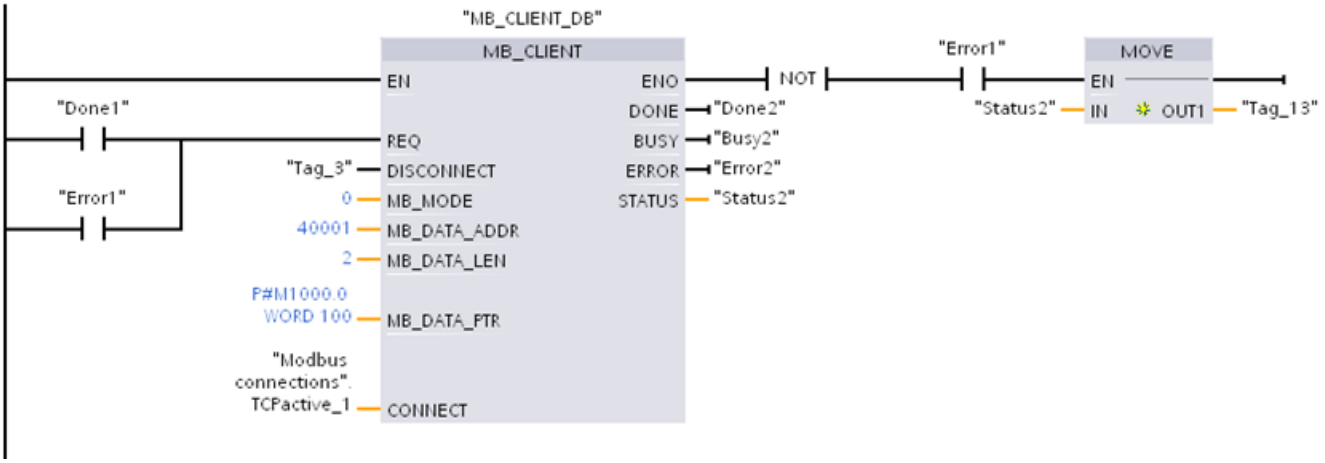
Her bir Modbus TCP talebinin yürütülmesinin tamamlandığından emin olmalısınız. Program lojiğiniz, yürütme sıralamasına kumanda etmelidir. Aşağıdaki örnekte, yürütme sıralamasına birinci ve ikinci istemci taleplerinin çıkışları tarafından nasıl kumanda edilebileceğini anlatılmaktadır.

Bu örnekte, her iki istemcinin de aynı CONNECT bağlantı verilerini (farklı zamanlarda) kullandığını gösterilmektedir. İstemciler, aynı desantral Modbus TCP sunucusundan tutma kaydı verilerini alıp, yerel CPU'nun markörlerinin bellek alanında bulunan aynı adrese aktarmaktadır. Ayrıca, yayınlanan bir hata yakalanır (opsiyonel fonksiyon).

Ağ 1: Modbus fonksiyonu 3 - Modbus TCP sunucusunun tutma kaydındaki sözcükleri okur.



Ağ 2: Modbus fonksiyonu 3 - Modbus TCP sunucusunun tutma kaydındaki sözcükleri okur.



13.5.3 Modbus RTU

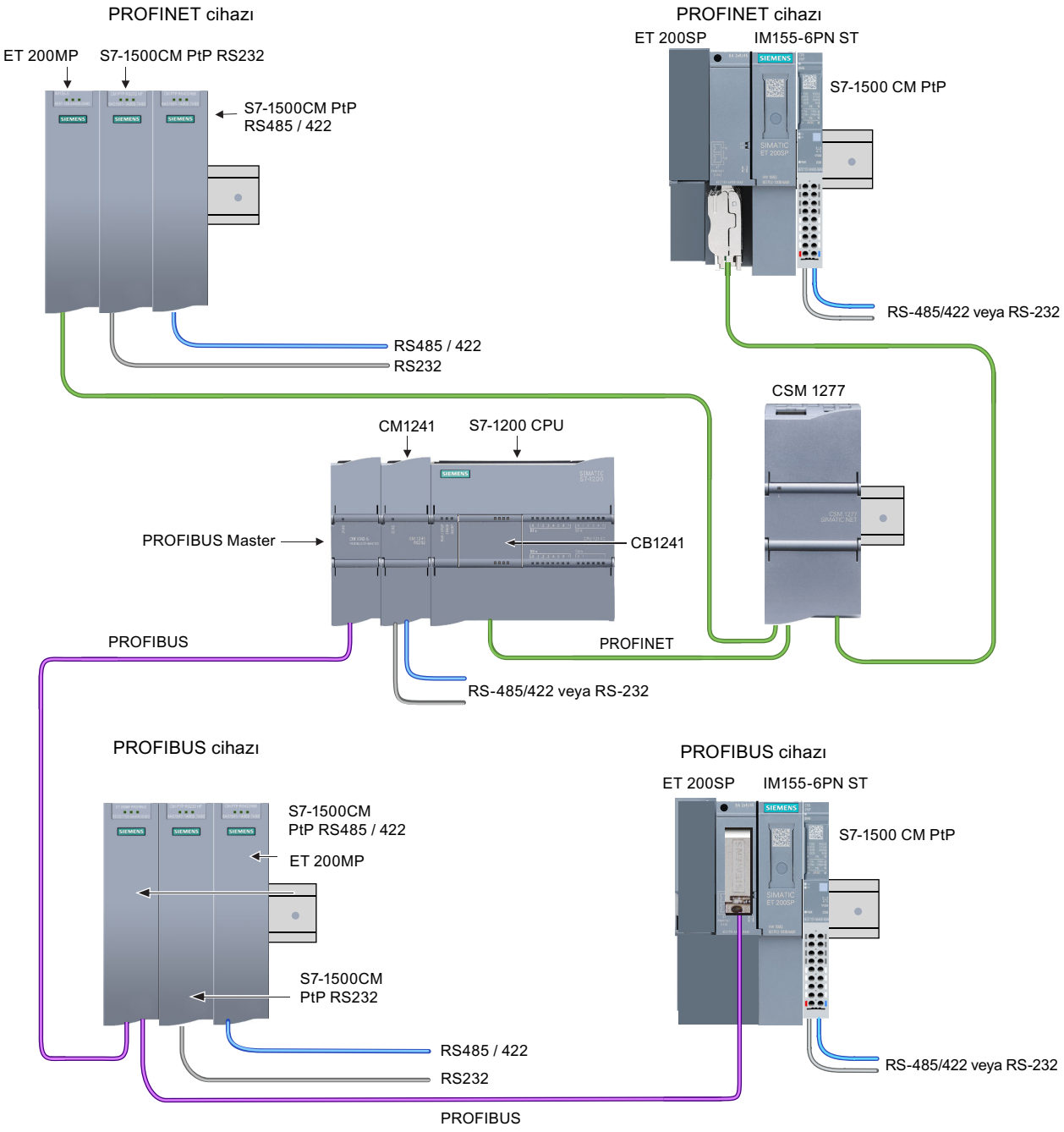
13.5.3.1 Genel bakış

Modbus RTU (Remote Terminal Unit) ağ iletişimi için standart bir protokoldür ve ağ üzerindeki Modbus cihazları arasında seri veri aktarımı için RS232 veya RS485 elektriksel bağlantı kullanır. RS232 veya RS485 CM veya RS485 CB ile bir CPU'ya PtP (noktadan noktaya) ağ portları ekleyebilirsiniz.

Modbus RTU, tüm iletişimin tek bir Master cihaz tarafından tetiklendiği, Slave'lerin ise yalnızca Master'ın talebine yanıt verebildiği bir Master/Slave ağı kullanır. Master bir Slave adresine bir talep gönderir ve yalnızca Slave adresi bu komuta yanıt verir.

Modbus RTU ile çeşitli cihazlarla (RFID okuyucular, GPS ve diğerleri) iletişim kurmak için desantral PROFINET veya PROFIBUS çevrebirimlerini kullanabilirsiniz:

- PROFINET (Sayfa 561): S7-1200 CPU'nun Ethernet arayüzü bir PROFINET arayüz modülüne bağlıdır. Harici cihazlarla seri iletişim, arayüz modülü ile modül taşıyıcısındaki PtP iletişim modülleri aracılığıyla mümkün olur.
- PROFIBUS (Sayfa 734): S7-1200 CPU ile modül taşıyıcısının sol tarafına bir PROFIBUS iletişim modülü takılır. PROFIBUS iletişim modülü, bir PROFIBUS arayüz modülü içeren bir modül taşıyıcısına bağlanır. Harici cihazlarla seri iletişim, arayüz modülü ile modül taşıyıcısındaki PtP iletişim modülleri aracılığıyla mümkün olur.


PROFINET cihazı
ET 200MP
S7-1500CM PtP RS232
S7-1500CM PtP
RS485 / 422
RS485 / 422
RS232
PROFINET cihazı
ET 200SP
IM155-6PN ST
S7-1500 CM PtP
RS-485/422 veya RS-232
CSM 1277
CM1241
S7-1200 CPU
PROFIBUS Master
CB1241
PROFIBUS
PROFINET
RS-485/422 veya RS-232
PROFIBUS cihazı
S7-1500CM
PtP RS485 / 422
ET 200MP
S7-1500CM
PtP RS232
RS485 / 422
RS232
PROFIBUS
PROFIBUS cihazı
ET 200SP
IM155-6PN ST
S7-1500 CM PtP
RS-485/422 veya RS-232

Modbus RTU, yordamlar ve sürümler

Modbus RTU yordamları (Sayfa 1022), desantral PROFINET ve PROFIBUS çevrebirimlerine bağlanmak için gereken olanakları sağlar. Bunlar, desantral çevrebirimler ve PtP cihazları ile modül taşıyıcısındaki PtP iletişim modülleri arasındaki iletişimin yapılandırılmasına izin verir. S7-1200 CM 1241 modülleri, bu Modbus RTU yordamlarını kullanmak için en azından Firmware sürümü V2.1'ye ihtiyaç duyar. Eski S7-1200 CPU programları, eski Modbus RTU yordamlarını (Sayfa 1085) kullanıyor olabilir.

STEP 7, Modbus RTU için yordamların çeşitli sürümlerini sunmaktadır. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

13.5.3.2 Desteklenen maksimum Modbus Slave sayısı

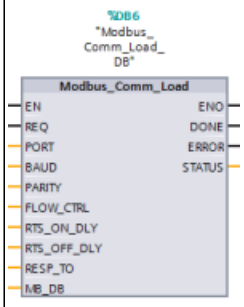
Modbus adreslemesi maksimum 247 Slave'i destekler (Slave numaraları 1 ila 247). Modbus ağındaki her segment, RS485 arayüzünün yükleme ve sürücü fonksiyonlarına bağlı olarak en fazla 32 cihaza sahip olabilir. 32 cihazlık limite ulaştığınızda, bir sonraki segmente ulaşmak için bir tekrarlayıcı kullanmanız gerekecektir. RS485 için bir Master'a bağlı olan 247 Slave'i desteklemek için yedi tekrarlayıcıya gerek duyarsınız.

Siemens'in tekrarlayıcıları yalnızca PROFIBUS ile çalışır. Bunların fonksiyonları, PROFIBUS token aktarımını denetlemektir. Siemens'in tekrarlayıcılarını, diğer protokollerle kullanamazsınız. Bu nedenle Modbus için diğer tedarikçilerin tekrarlayıcılarına gerek duyarsınız.

Modbus zaman aşımları için standart olarak yüksek değerler ayarlanmıştır. Dolayısıyla tekrarlayıcıların kullanımı zaman gecikmesi bakımından bir soruna neden olmaz. Modbus Master, bir Slave'in yavaş yanıt verip vermediğine veya birden çok tekrarlayıcının yanıtı geciktirip geciktirmediğine bakmaksızın çalışır.

13.5.3.3 Modbus RTU yordamları

Tablo 13-75 Modbus_Comm_Load yordamı

LAD/FBD	SCL	Açıklama
	**"Modbus_Comm_Load_DB"(** **REQ:=_bool_in,** **PORT:=_uint_in_,** **BAUD:=_udint_in_,** **PARITY:=_uint_in_,** **FLOW_CTRL:=_uint_in_,** **RTS_ON_DLY:=_uint_in_,** **RTS_OFF_DLY:=_uint_in_,** **RESP_TO:=_uint_in_,** **DONE=>_bool_out,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **MB_DB:=_fbtref_inout_);**	Modbus_Comm_Load yordamı, Modbus RTU protokolü üzerinden iletişim için SIPLUS I/O veya bir PtP portunu yapılandırır. Modbus RTU portunun donanım opsiyonları: Maksimum üç CM (RS485 veya RS232), artı bir CB (R4845) montajı Modbus RTU SIPLUS I/O için opsiyonlar: ET 200MP S7-1500CM PtP (RS485 / 422 veya RS232) veya ET 200SP S7-1500 CM PtP (RS485 / 422 veya RS232) kurulumunu yapın Modbus_Comm_Load yordamını programınıza eklerseniz, otomatik olarak bir örnek veri bloğu atanır.

Tablo 13-76 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
EN	IN	Bool	Not: Modbus RTU için Modbus_Comm_Load yordamı, PDREC ve WRREC yordamlarını kullanarak PTP modülünü başlatır. Ancak, RDREC/WRREC yordamı asenkron olarak çalışır, bu ise yordamın yürütülmesi için birden çok sayıda çevrime gerek olduğu anlamına gelir. Dolayısıyla Modbus_Comm_Load'ın EN parametresi RDREC/WRREC yordamı tamamlanana dek DOĞRU kalmalıdır.
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Yalnızca sürüm 2.0)
PORT	IN	Port	CM veya CB'yi kurup yapılandırdıktan sonra, port kodu, PORT kutu bağlantısındaki parametre açılır listesinde görüntülenir. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır.
BAUD	IN	UDInt	Baud hızının seçimi: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200. Diğer tüm veriler geçersizdir.
PARITY	IN	UInt	Parite seçimi: • 0 – yok • 1 – tek sayılı • 2 – çift sayılı
FLOW_CTRL [1]	IN	UInt	Akış kontrolü seçimi: • 0 – (standart) akış kontrolü yok • 1 – RTS her zaman AÇIK ile donanım akış kontrolü (RS485 portları için geçerli değildir) • 2 – RTS ile anahtarlanan donanım akış kontrolü

[1] Modbus_Comm_Load için opsiyonel parametreler (V 2.x veya üstü). LAD/FBD kutusunun altındaki oka tıklayarak bu parametreleri kapsayacak şekilde kutuyu genişletin.

Parametre ve veri tipi		Veri tipi	Açıklama
RTS_ON_DLY [1]	IN	UInt	RTS açma gecikmesinin seçimi: • 0 - (standart) "RTS aktif"ten mesajın ilk karakteri gönderilinceye dek gecikme yok. • 1 ila 65535 - Mesajın ilk karakteri gönderilinceye dek "RTS aktif "ten itibaren milisaniye cinsinden gecikme (RS485 portları için geçerli değildir). RTS gecikmeleri, FLOW_CTRL seçiminden bağımsız olarak uygulanmalıdır.
RTS_OFF_DLY [1]	IN	UInt	RTS kapatma gecikmesinin seçimi: • 0 – (standart) son karakterin aktarımından sonra "RTS aktif değil" durumuna kadar gecikme olmaz. • 1 ila 65535 - Son karakterin aktarımından "RTS aktif değil" durumuna kadar geçen milisaniye cinsinden gecikme (RS485 portları için geçerli değildir). RTS gecikmeleri, FLOW_CTRL seçiminden bağımsız olarak uygulanmalıdır.
RESP_TO [1]	IN	UInt	Yanıt zaman aşımı: Modbus_Master'ın Slave'den yanıt beklediği milisaniye cinsinden süre. Slave bu süre içinde yanıt vermezse, Modbus_Master talebi tekrarlar veya tekrar gönderim belirli bir sayıya ulaşmışsa, ilgili talebi bir hata ile sonlandırır. 5 msn ila 65535 msn (standart değer = 1000 msn).
MB_DB	IN	Variant	Modbus_Master veya Modbus_Slave yordamlarının örnek veri bloğuna bir referans. Modbus_Master veya Modbus_Slave'i programınıza ekledikten sonra, ilgili DB kodu, parametre açılır listesinde MB_DB bağlantısında görünür.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur. (yalnızca sürüm 2.0)
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

[1] Modbus_Comm_Load için opsiyonel parametreler (V 2.x veya üstü). LAD/FBD kutusunun altındaki oka tıklayarak bu parametreleri kapsayacak şekilde kutuyu genişletin.

Modbus_Comm_Load, Modbus RTU protokolü için bir portunu yapılandırmak üzere yürütülüyor. Bir port, Modbus RTU protokolü için yapılandırıldıktan sonra, yalnızca Modbus_Master veya Modbus_Slave yordamları tarafından kullanılabilir.

Modbus iletişimi için kullanılan her bir iletişim portunun konfigürasyonu için bir Modbus_Comm_Load yordamının yürütülmesi gerekir. Kullandığınız her porta benzersiz bir Modbus_Comm_Load örnek DB atamanız gerekir. CPU'ya en fazla üç iletişim modülü (RS232 veya RS485) ve bir iletişim kartı (RS485) monte edebilirsiniz. Modbus_Comm_Load yordamını yol alma OB'sinden çağırıp bir kez çalıştırabilirsiniz veya ilk çevrim için sistem markörünü (Sayfa 86) kullanarak bir seferlik yürütme için çağrıyı başlatabilirsiniz. Yalnızca Baud hızı veya parite gibi iletişim parametrelerinin değiştirilmesi gerekiyorsa Modbus_Comm_Load yordamını tekrar çalıştırın.

Modbus kütüphanesini desantral bir modül taşıyıcısındaki bir modülle kullanıyorsanız, Modbus_Comm_Load yordamını bir periyodik ara verme programında yürütülmelidir (örn. saniyede bir veya her 10 saniyede bir). Desantral modül taşıyıcısının gerilim beslemesi kesilirse veya modül çıkartılırsa, modül işletimi yeniden başlatıldığında PtP modülüne yalnızca belirtilen HWConfig parametresi gönderilir. Modbus_Master zaman aşımı ve Modbus_Slave tarafından tetiklenen tüm talepler sessiz moda geçer (mesajlara tepki verilmez). Modbus_Comm_Load yordamının periyodik olarak yürütülmesi bu sorunları giderir.

Programınıza Modbus_Master veya Modbus_Slave eklerseniz, yordama bir örnek veri bloğu atanır. Modbus_Comm_Load yordamında MB_DB parametresini belirtirseniz, bu örnek veri bloğuna referans verilir.

Modbus_Comm_Load örnek veri bloğunun (DB) değişkenleri

Aşağıdaki tabloda, Modbus_Comm_Load örnek DB'sinde programınızda kullanılabilecek genel statik değişkenler gösterilmektedir:

Tablo 13-77 Modbus_Comm_Load yordamının örnek DB'sindeki statik değişkenler

Değişken	Veri tipi	Standart değer	Açıklama
ICHAR_GAP	Word	0	Karakterler arasındaki maksimum karakter gecikme süresi. Bu parametre milisaniye cinsinden belirtilir ve alınan karakterler arasında beklenen süreyi artırır. Bu parametre için karşılık gelen Bit sürelerinin sayısı, Modbus standart değeri olan 35 Bit zamanına (3,5 karakter zamanı) eklenir.
RETRIES	Word	2	Master'ın "Yanıt yok" için 0x80C8 hata kodunu döndürmeden önce yürüttüğü yeniden denemelerin sayısı.
EN_SUPPLY_VOLT	Bool	0	L+ besleme gerilimi arızası için teşhisi etkinleştirir.
MODE	USInt	0	İşletim türü Aşağıdaki işletim türleri geçerlidir: • 0 = Tam dupleks (RS232) • 1 = Tam dupleks (RS422), dört telli işletim (noktadan noktaya) • 2 = Tam dupleks (RS422), dört telli işletim (çok noktalı Master, CM PtP, ET 200SP) • 3 = Tam dupleks (RS422), dört telli işletim (çok noktalı Slave, CM PtP, ET 200SP) • 4 = Yarı dupleks (RS485), iki telli işletim (aşağıdaki nota bakın)
LINE_PRE	USInt	0	Alım hattının çıkış durumu Aşağıdaki çıkış türleri geçerlidir: • 0 = Çıkış durumu "Yok" (aşağıdaki nota bakın) • 1 = Sinyal R(A) = 5 V DC, sinyal R(B) = 0 V DC (tel kopması algılaması): Bu çıkış durumunda tel kopması algılaması mümkündür. Yalnızca şu şekilde seçilebilir: "Tam dupleks (RS422), dört telli işletim (noktadan noktaya bağlantı)" ve "Tam dupleks (RS422), dört telli işletim (çok noktalı Slave)". • 2 = Sinyal R(A) = 0 V DC gerilimi, Sinyal R(B) = 5 V DC gerilimi: Bu standart ayar, boşta durumuna karşılık gelir (aktif gönderim işlemi yok). Bu çıkış durumunda tel kopması algılaması mümkün değildir.
BRK_DET	USInt	0	Tel kopması algılaması Şu ayarlar geçerlidir: • 0 = Tel kopması algılaması devre dışı bırakıldı • 1 = Tel kopması algılaması etkinleştirildi
EN_DIAG_ALARM	Bool	0	Teşhis alarmının etkinleştirilmesi: • 0 = Aktif değil • 1 = Aktif
STOP_BITS	USInt	1	Stop Bit'lerinin sayısı: • 1 = 1 Stop Bit'i • 2 = 2 Stop Bit'i • 0, 3 ila 255 = rezerve

NOT

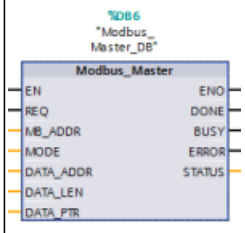
RS485 için CM 1241 ile PROFIBUS kablolarının kullanımı için gereken ayarlama

Tablo 13-78 Modbus_Comm_Load yürütmesinin koşul kodları [1]

STATUS (W#16#)	Açıklama
0000	Hata yok
8180	Geçersiz port ID değeri (iletişim modülünün yanlış portu/yanlış donanım kodu)
8181	Baud hızı için geçersiz değer.
8182	Parite için geçersiz değer.
8183	Akış kontrolü için geçersiz değer.
8184	Yanıt zaman aşımı süresi için geçersiz değer (yanıt zaman aşımı, minimum değer olan 5 msn'den küçük)
8185	MB_DB parametresi bir Modbus_Master veya Modbus_Slave yordamının örnek veri bloğu değildir.

[1] Yukarıda belirtilen Modbus_Comm_Load yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

Tablo 13-79 Modbus_Master yordamı

LAD/FBD	SCL	Açıklama
	**"Modbus_Master_DB"(** **REQ:=_bool_in_,** **MB_ADDR:=_uint_in_,** **MODE:=_usint_in_,** **DATA_ADDR:=_udint_in_,** **DATA_LEN:=_uint_in_,** **DONE=>_bool_out_,** **BUSY=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **DATA_PTR:=_variant_inout_);**	Modbus_Master yordamı, Modbus_Comm_Load yordamının önceki bir yürütmesi tarafından yapılandırılan bir port üzerinden bir Modbus Master olarak iletişim kurar. Modbus_Master yordamını programınıza eklerseniz, otomatik olarak bir örnek veri bloğu atanır. Bu Modbus_Master örnek veri bloğu, Modbus_Comm_Load yordamında MB_DB parametresini belirtirseniz kullanılır.

Tablo 13-80 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	0 = Talep yok 1 = Modbus Slave'e veri gönderme talebi
MB_ADDR	IN	V1.0: USInt V2.0: UInt	Modbus-RTU istasyon adresi: Standart adresleme aralığı (1 - 247) Genişletilmiş adresleme aralığı (1 - 65535) 0 değeri, bir mesajı tüm Modbus Slave'lere Broadcast vermek için rezerve edilmiştir. Broadcast için yalnızca Modbus fonksiyon kodları 05, 06, 15 ve 16 desteklenir.
MODE	IN	USInt	Seçim modu: Talep türünü belirtir (okuma, yazma veya teşhis). Daha fazla bilgiyi, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
DATA_ADDR	IN	UDInt	Slave'deki başlangıç adresi: Modbus Slave'de erişilecek verilerin başlangıç adresini belirtir. Geçerli adresleri, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.

Parametre ve veri tipi		Veri tipi	Açıklama
DATA_LEN	IN	UInt	Veri uzunluğu: Bu talep tarafından erişilecek Bit veya sözcük sayısını belirtir. Geçerli uzunlukları, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
DATA_PTR	IN_OUT	Variant	Veri işaretçisi: Yazılacak veya okunacak veri için M veya DB adresine (optimize edilmemiş DB) işaret eder.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - Devam eden Modbus_Master işlemi yok • 1 - Modbus_Master işlemi devam ediyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

Modbus_Master iletişimi kuralları

- Modbus_Master yordamının bu portla iletişim kurabilmesi için, bir portu yapılandırmak üzere Modbus_Comm_Load yordamının yürütülmesi gerekir.
- Bir port Modbus Master taleplerini başlatmak için kullanılacaksa, bu portun Modbus_Slave tarafından kullanılmasına izin verilmez. Bu port ile bir veya daha fazla Modbus_Master örneği kullanılabilir, ancak Modbus_Master'ın tüm yürütmeleri, bu port için aynı Modbus_Master örnek DB'sini kullanmalıdır.
- Modbus yordamları, iletişim sürecine kumanda etmek için iletişim alarm olaylarını kullanmaz. Programınız tamamlanan gönderim ve alım olayları için Modbus_Master yordamını sorgulamalıdır.
- Belirli bir port için Modbus_Master'ın yürütülmesini her zaman bir program çevrimi OB'sinden çağırın. Modbus_Master yordamları bir seferde yalnızca bir program çevriminde veya bir seferde bir periyodik/zaman gecikmeli yürütme katmanında yürütülebilir. Her iki yürütmenin öncelik katmanında bunların işlenmesine izin verilmez. Bir Modbus_Master yordamının daha yüksek önceliğe sahip bir yürütme katmanındaki başka bir Modbus_Master yordamı tarafından öncelikli olarak kesintiye uğratılması, usulüne uygun bir işletimle sonuçlanmaz. Modbus_Master yordamlarının yol alma, teşhis veya zaman hatası yürütme katmanlarında işlenmesine izin verilmez.
- Bir Modbus_Master yordamı bir aktarımı tetikledikten sonra, bu örnek, DONE = 1 veya ERROR = 1 durumu çıkana kadar etkinleştirilmiş EN girişi ile sürekli olarak yürütülmelidir. Belirli bir Modbus_Master örneği, bu iki olaydan biri gerçekleşene dek aktif olarak kabul edilir. Orijinal örnek aktifken, REQ girişi etkinleştirilmiş başka bir örneğe yapılan herhangi bir çağrı hata ile sonuçlanır. Orijinal örneğin devam eden yürütülmesi durursa, talep durumu "Blocked_Proc_Timeout" statik değişkeni tarafından belirtilen süre boyunca aktif kalır. Bu süre geçtikten sonra, REQ girişi etkinleştirilmiş bir sonraki çağrılan Modbus_Master yordamı aktif örnek haline gelir. Bu sayede, tek bir Modbus_Master yordamının monopolleşmesi veya porta erişimi engellemesi önlenir. Orijinal aktif örnek "Blocked_Proc_Timeout" statik değişkeni tarafından belirtilen süre boyunca etkinleştirilmezse, bu örnekten gelen bir sonraki yürütme (REQ girişi ayarlanmaksızın) aktif durumu siler. REQ ayarlanmışsa, bu yürütme, başka hiçbir örnek aktif değilmiş gibi yeni bir Modbus_Master talebini tetikler.

REQ parametresi

0 = Talep yok, 1 = Modbus Slave'e veri gönderme talebi

Bu girişe, seviye veya kenar kontrollü bir kontak aracılığıyla kumanda edebilirsiniz. Bu girişin her etkinleştirilmesinde, aynı örnek DB'yi kullanan başka bir Modbus_Master yordamının güncel talep işlenene dek bir talepte bulunamamasını sağlamak için bir durum makinesi başlatılır. Diğer tüm giriş durumları, yanıt alıncaya veya bir hata tespit edilinceye dek güncel talep için dahili olarak alınır ve saklanır.

Güncel talep işlenmeden önce aynı Modbus_Master örneği REQ = 1 girişi ile tekrar yürütülürse, bir sonraki aktarımlar gerçekleştirilmez. Ancak talep işlendiğinde, Modbus_Master REQ girişi = 1 ile tekrar yürütüldüğünde yeni bir talep yayınlanır.

Modbus fonksiyon tipini seçmek için DATA_ADDR ve MODE parametrelerini kullanın.

DATA_ADDR (Slave'deki Modbus başlangıç adresi): Modbus Slave'de erişilecek verilerin başlangıç adresini belirtir.

Modbus_Master yordamı, bir fonksiyon kodu girişi yerine bir MODE girişi kullanır. MODE ve Modbus adresi kombinasyonu, gerçek Modbus mesajında kullanılan fonksiyon kodunu belirler. Aşağıdaki tabloda, MODE parametresi, Modbus fonksiyon kodu ve Modbus adres aralığı arasındaki ilişki gösterilmektedir:

Tablo 13-81 Modbus fonksiyonları

MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	Modbus adresi
0	01	1 ila 2000 1 ila 1992 [1]	Çıkış Bit'ini oku: Talep başına 1 ila (1992 veya 2000) Bit	1 ila 9999
0	02	1 ila 2000 1 ila 1992 [1]	Giriş Bit'ini oku: Talep başına 1 ila (1992 veya 2000) Bit	10001 ila 19999
0	03	1 ila 125 1 ila 124 [1]	Tutma kaydını oku: Talep başına 1 ila (124 veya 125) sözcük	40001 ila 49999 veya 400001 ila 465535
0	04	1 ila 125 1 ila 124 [1]	Giriş sözcüklerini oku: Talep başına 1 ila (124 veya 125) Bit	30001 ila 39999
104	04	1 ila 125 1 ila 124 1	Giriş sözcüklerini oku: Talep başına 1 ila (124 veya 125) Bit	00000 ila 65535
1	05	1	Bir çıkış Bit'i yaz: Talep başına bir Bit	1 ila 9999
1	06	1	Bir tutma kaydı yaz: Talep başına 1 sözcük	40001 ila 49999 veya 400001 ila 465535
1	15	2 ila 1968 2 ila 1960 [1]	Birden fazla çıkış Bit'i yaz: Talep başına 2 ila (1960 veya 1968) Bit	1 ila 9999
1	16	2 ila 123 2 ila 122 [1]	Birden fazla tutma kaydı yaz: Talep başına 2 ila (122 veya 123) sözcük	40001 ila 49999 veya 400001 ila 465535
2	15	1 ila 1968 2 ila 1960 [1]	Bir veya birden fazla çıkış Bit'i yaz: Talep başına 1 ila (1960 veya 1968) Bit	1 ila 9999
2	16	1 ila 123 1 ila 122 [1]	Bir veya birden fazla tutma kaydı yaz: Talep başına 1 ila (122 veya 123) sözcük	40001 ila 49999 veya 400001 ila 465535

[1] Genişletilmiş adresleme ile maksimum veri uzunluğu, fonksiyonun veri tipine bağlı olarak 1 Bayt veya 1 sözcük daha azdır.

MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	Modbus adresi
11	11	0	Slave iletişim durumu sözcüğünü ve olay sayacını oku. Status sözcüğü meşgul olduğunu gösterir (0 = meşgul değil, 0xFFFF = meşgul). Olay sayacı, bir mesaj başarıyla işlendiğinde her defasında artırılır. Modbus_Master'ın DATA_ADDR ve DATA_LEN işlemcilerinde bu fonksiyon yok sayılır	
80	08	1	Teşhis kodu 0x0000 ile Slave durumunu kontrol et (kontrol döngüsü testi, Slave talebin bir yankısını geri döndürür) Talep başına 1 sözcük	
81	08	1	Teşhis kodu 0x000A ile Slave olay sayacını sıfırla Talep başına 1 sözcük	
3 ila 10, 12 ila 79, 82 ila 255			Rezerve	

[1] Genişletilmiş adresleme ile maksimum veri uzunluğu, fonksiyonun veri tipine bağlı olarak 1 Bayt veya 1 sözcük daha azdır.

DATA_PTR parametresi

DATA_PTR parametresi, yazılacak veya okunacak DB veya M adresini gösterir. Bir veri bloğu kullanırsanız, Modbus Slave'lerde okuma ve yazma işlemleri için veri belleği sağlayan bir global veri bloğu oluşturmanız gerekecektir.

NOT

DATA_PTR veri bloğu, direkt adreslemeye izin vermelidir.

Veri bloğu, doğrudan (mutlak) ve sembolik adreslemeye izin vermelidir. Veri bloğunu oluştururken "Standart" erişim özniteliğini seçmelisiniz.

Modbus_Master yordamının 4.0 sürümünden itibaren, "Optimize edilmiş blok erişimi" veri bloğu özniteliğini etkinleştirebilirsiniz. Optimize edilmiş bellekte yalnızca aşağıdaki veri tiplerine sahip tek bir eleman veya bir eleman Array'i kullanabilirsiniz: Bool, Byte, Char, Word, Int, DWord, Dint, Real, USInt, UInt, UDInt, SInt veya WChar.

DATA_PTR parametresi için veri bloğu yapıları:

- Bu veri tipleri, 30001 ila 39999, 40001 ila 49999 ve 400001 ila 465536 Modbus adreslerinden **sözcük okumak** ve 40001 ila 49999 ve 400001 ila 465536 Modbus adreslerine **sözcük yazmak** için geçerlidir.
  - WORD, UINT veya INT veri tiplerinin standart Array'i
  - WORD, UINT veya INT türünde, her bir elemanın benzersiz bir ada ve 16 Bit'lik bir veri tipine sahip olduğu adlandırılmış yapı.
  - Her elemanın benzersiz bir ada ve 16 veya 32 Bit'lik veri tipine sahip olduğu adlandırılmış karmaşık yapı.
- Modbus adresleri 00001 ila 09999 için Bit'leri **okumak** ve yazmak ve 10001 ila 19999 arasındaki Bit'leri okumak için kullanılır.
  - Boole veri tipleri için standart alan.
  - Benzersiz olarak adlandırılmış Boole değişkenlerinin adlandırılmış Boole yapısı.
- Gerekli olmamakla birlikte, her Modbus_Master yordamının kendisi için ayrı bir bellek alanına sahip olması önerilir. Bunun nedeni, birden fazla Modbus_Master yordamının aynı bellek alanında okuma ve yazması durumunda veri imhası olasılığının daha yüksek olmasıdır.
- DATA_PTR için veri alanlarının aynı global veri bloğunda olmaları zorunlu değildir. Modbus okuma işlemleri için birden çok veri bloğu, Modbus yazma işlemleri için bir veri bloğu veya her Slave istasyonu için bir veri bloğu oluşturabilirsiniz.

Modbus_Master yordamının veri bloğundaki (DB) değişkenler

Aşağıdaki tabloda, Modbus_Master örnek DB'sinde programınızda kullanabileceğiniz açık statik değişkenler gösterilmektedir:

Tablo 13-82 Modbus_Master yordamının örnek DB'sindeki statik değişkenler

Değişken	Veri tipi	Standart değer	Açıklama
Blocked_Proc_Timeout	Real	3.0	Bloke edilen bir Modbus_Master örneği beklenirken, bu örnek AKTİF olarak kaldırılmadan önceki süre (saniye cinsinden). Bu, örneğin, bir Modbus_Master talebi yayınlandığında ve program durduğunda ve talebi tamamen bitirmeden önce Modbus_Master fonksiyonunu çağırdığında oluşabilir. Zaman değeri 0'dan büyük ve 55 saniyeden küçük olmalıdır, aksi durumda bir hata oluşur.
Extended_Addressing	Bool	YANLIŞ	Slave istasyonun adreslemesini tek veya çift Bayt olarak yapılandırır: • YANLIŞ = 1 Bayt'lık adres, 0 ila 247 • DOĞRU = 2 Bayt'lık adres (genişletilmiş adreslemeye karşılık gelir), 0 ila 65535
MB_DB	MB_BASE	-	MB_DB yordamının Modbus_Comm_Load parametresi, MB_DB yordamının Modbus_Master parametresine bağlanmalıdır.

Programınız, Modbus_Master işlemlerine kumanda etmek için Blocked_Proc_Timeout ve Extended_Addressing değişkenlerine değerler yazabilir. Bu değişkenlerin programlama editöründe nasıl kullanılacağına dair bir örneği ve genişletilmiş Modbus adresleme hakkında daha fazla bilgiyi, Modbus_Slave alt bölümünde HR_Start_Offset (Sayfa 1032) ve Extended_Addressing (Sayfa 1032)'in açıklamasında bulabilirsiniz.

Koşul kodları

Tablo 13-83 Modbus_Master'ın yürütülmesi için durum kodları (iletişim ve konfigürasyon hataları) [1]

STATUS (W#16#)	Açıklama
0000	Hata yok
80C8	Slave zaman aşımı. Belirtilen Slave belirtilen süre boyunca yanıt vermedi. Slave cihazın Baud hızını, paritesini ve kablaj bağlantılarını kontrol edin. Bu hata yalnızca yapılandırılmış tüm yeniden denemeler gerçekleştirildikten sonra bildirilir.
80C9	Modbus_Master yordamının süresi aşağıdaki nedenlerden biriyle dolmuştur: • Yordam, iletişim için kullanılan modülden bir yanıt bekliyor. • Blocked_Proc_Timeout değeri çok küçük belirlenmiş. Desantral PROFIBUS veya PROFINET G/Ç cihazı aşağıdaki durumlardan birinden geri döndüğünde, bu hatanın bildirimi yapılır. • Bir güç veya iletişim kesintisi • İletişim modülünde bir takıp/çıkarma olayı. Bu durumlarda, donanım konfigürasyonu PLC'den yeniden yüklenir ve iletişim modülünü doğru yapılandırmak için Modbus_Comm_Load yordamını yeniden yürütülmelidir.
80D1	Alıcı cihaz devam etmekte olan bir aktarımı kesintiye uğratmak için akış kontrolü talep etmiş, ancak bekleme süresi içinde aktarımı yeniden etkinleştirmemiştir. Alıcı cihaz bekleme süresi içinde bir CTS algılamazsa, donanım akış kontrolü sırasında da bu hata üretilir.
80D2	DCE'den DSR sinyali alınmadığı için gönderim talebi iptal edildi.
80E0	Alım tamponu dolu olduğu için mesaj sonlandırıldı.
80E1	Mesaj, bir parite hatası nedeniyle sonlandırıldı.
80E2	Mesaj, bir telegram hatası nedeniyle sonlandırıldı.
80E3	Mesaj, bir taşma hatası nedeniyle sonlandırıldı.
80E4	Belirtilen uzunluk toplam tampon boyutunu aştığı için mesaj sonlandırıldı.
8180	Port ID için geçersiz değer veya Modbus_Comm_Load yordamında hata.
8186	Geçersiz Modbus istasyon adresi
8188	Broadcast talebi için geçersiz mod belirtildi
8189	Veri adresi için geçersiz değer.
818A	Veri uzunluğu için geçersiz değer.
818B	Yerel veri kaynağına veya yerel veri hedefi üzerinde geçersiz işaretçi: Boyut, doğru değil.
818C	DATA_PTR için geçersiz işaretçi veya Blocked_Proc_Timeout geçersiz. Veri alanı aşağıdakilerden biri olmalıdır: • Klasik DB • Sembolik veya bir kalıcı DB'deki temel veri tipleri Array'i • Markör
8200	Port, bir gönderme talebinin işlenmesiyle meşgul.
8280	Modülden okuma sırasında negatif onaylama PORT parametresindeki girişi kontrol edin. Bu hata, güç veya iletişim kaybı veya bir modülün çekip çıkarılması nedeniyle desantral PROFIBUS veya PROFINET G/Ç modülünün kaybından kaynaklanabilir.
8281	Modüle yazarken negatif onaylama. PORT parametresindeki girişi kontrol edin. Bu hata, güç veya iletişim kaybı veya bir modülün çekip çıkarılması nedeniyle desantral PROFIBUS veya PROFINET G/Ç modülünün kaybından kaynaklanabilir.

Tablo 13-84 Modbus_Master'in yürütülmesi için koşul kodları Modbus protokol hataları) [1]

STATUS (W#16#)	Slave'in yanıt kodu	Modbus protokol hatası
8380	-	CRC hatası
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluğunda hata
8383	02	Veri adresi hatası veya adres, DATA_PTR'nin geçerli aralığın dışında
8384	03'ten büyük	Veri değerinde hata
8385	03	Veri teşhis kodunun değeri desteklenmiyor (fonksiyon kodu 08)
8386	-	Yanıttaki fonksiyon kodu, talepteki koda karşılık gelmiyor.
8387	-	Yanlış Slave, tepki vermiş.
8388	-	Slave'in bir yazma talebine verdiği yanıt yanlış. Slave tarafından yayınlanan yazma talebi Master'ın gönderdiğine karşılık gelmiyor.

[1] Yukarıda belirtilen Modbus_Master yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

NOT

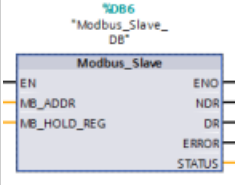
Profibus iletişimi için veri setinin maksimum uzunluğunun ayarlanması

CM1243-5 iletişim modülü, bir RS232, RS422 veya RS485 noktadan noktaya modül kullanan bir ET200SP veya ET200MP Profibus cihazına kumanda etmek için Profibus Master olarak kullanılıyorsa, bu durumda "max_record_len" veri bloğu değişkenini aşağıdaki gibi belirtik olarak 240 değerine ayarlamalısınız:

Modbus_Comm_Load yordamının yürütülmesinden çalıştırdıktan sonra, örnek DB'nin Send_P2P alt bölümündeki max_record_len parametresini (örneğin, "Modbus_Master_DB".Send_P2P.max_record_len) 240 olarak ayarlayın.

max_record_len değerinin belirtik ataması yalnızca Profibus bağlantıları için gereklidir. Profinet bağlantılarında ise max_record_len için geçerli bir değer zaten kullanılmaktadır.

Tablo 13-85 Modbus_Slave yordamı

LAD/FBD	SCL	Açıklama
	**"Modbus_Slave_DB"(MB_ADDR:=_uint_in_, NDR=>_bool_out_, DR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MB_HOLD_REG:=_variant_inout_);**	Modbus_Slave yordamı ile programınız iki şekilde iletişim kurabilir: • CM (RS485 veya RS232) ve CB'nin (RS485) bir PtP portu üzerinden Modbus RTU Slave olarak. • Modbus RTU SIPLUS I/O opsiyonları üzerinden Modbus RTU Slave olarak: – ET 200MP S7-1500CM PtP (RS485 / RS422 veya RS232) kurulumunu yapın. – ET 200SP S7-1500 CM PtP (RS485 / RS422 veya RS232) kurulumunu yapın. Bir desantral Modbus RTU Master bir talepte bulunduğunda, kullanıcı programınız, Modbus_Slave yordamını çalıştırarak talebe yanıt verir. Yordamı eklerseniz, STEP 7 örnek DB'yi otomatik olarak oluşturur. Bu Modbus_Slave_DB

LAD/FBD	SCL	Açıklama
		adını, MB_DB yordamı için Modbus_Comm_Load parametresini belirtmek için kullanırsınız.

Tablo 13-86 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
MB_ADDR	IN	V1.0: USInt V2.0: UInt	Modbus Slave'in istasyon adresi: Standart adresleme aralığı (1 - 247) Genişletilmiş adresleme aralığı (0 - 65535)
MB_HOLD_REG	IN_OUT	Variant	İşaretçi, Modbus tutma kaydı DB'si üzerinde: Modbus tutma kaydı, markörlerin bellek alanı veya bir veri bloğu olabilir.
NDR	OUT	Bool	Yeni veriler hazır: • 0 – Yeni veriler yok • 1 - Modbus Master tarafından yeni veri yazıldığını gösterir.
DR	OUT	Bool	Veriler oku: • 0 – Veri okunmadı • 1 - Modbus Master tarafından yeni veri okunduğunu gösterir.
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. Yürütme bir hata ile sona ererse, STATUS parametresindeki hata kodu, yalnızca ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütmenin hata kodu

Modbus iletişimi fonksiyon kodları (1, 2, 4, 5 ve 15) doğrudan girişlerin proses görünümündeki ve CPU'nun çıkışlarının proses görünümündeki Bit'leri ve sözcükleri okuyabilir ve yazabilir. Bu fonksiyon kodlarında, MB_HOLD_REG parametresi bir Bayt'tan büyük bir veri tipi olarak tanımlanmalıdır. Aşağıdaki tabloda, Modbus adreslerinin CPU'daki proses görünümüne örnek olarak atanmasını gösterilmektedir.

Tablo 13-87 Modbus adreslerinin proses görünümüne atanması

Modbus fonksiyonları						S7-1200	
Kodlar	Fonksiyon	Veri alanı	Adres aralığı			Veri alanı	CPU adresi
01	Bit'leri oku	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
02	Bit'leri oku	Giriş	10001	-	18192	Girişlerin proses görünümü	E0.0 ila E1023.7
04	Sözcükleri oku	Giriş	30001	-	30512	Girişlerin proses görünümü	EW0 ila EW1022
05	Bit yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
15	Bit'leri yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7

Modbus iletişiminin fonksiyon kodları (3, 6, 16), markörlerin bellek alanındaki bir adres aralığı veya bir veri bloğu olabilen bir Modbus tutma kaydı kullanır. Tutma kaydı türü, Modbus_Slave yordamının MB_HOLD_REG parametresi tarafından belirlenir.

NOT

MB_HOLD_REG veri bloğu tipi

Modbus tutma kaydına sahip bir veri bloğu doğrudan (mutlak) ve sembolik adreslemeye izin vermelidir. Veri bloğunu oluştururken "Standart" erişim özniteliğini seçmelisiniz.

Modbus_Slave yordamının 4.0 sürümünden itibaren, "Optimize edilmiş blok erişimi" veri bloğu öznitelliğini etkinleştirebilirsiniz. Optimize edilmiş bellekte yalnızca aşağıdaki veri tiplerine sahip tek bir eleman veya bir eleman Array'i kullanabilirsiniz: Bool, Byte, Char, Word, Int, DWord, Dint, Real, USInt, UInt, UDInt, SInt veya WChar.

Aşağıdaki tabloda, Modbus adreslerinin Modbus fonksiyon kodları 03 (sözcük oku), 06 (sözcük yaz) ve 16 (sözcük yaz) için kullanılan tutma kaydına nasıl eşleneceğine ilişkin örnekler gösterilmektedir. DB adreslerinin gerçek üst sınırı, ilgili CPU modelinin maksimum çalışma belleği ve maksimum M belleği tarafından belirlenir.

Tablo 13-88 Modbus adreslerinin CPU belleğine atanması

Modbus Master adresi	**MB_HOLD_REG parametrelerine örnekler**				
	MW100	**DB10.DBW0**	**MW120**	**DB10.DBW50**	**"Reçete".Bileşim maddesi**
40001	MW100	DB10.DBW0	MW120	DB10.DBW50	"Reçete".Bileşim maddesi[1]
40002	MW102	DB10.DBW2	MW122	DB10.DBW52	"Reçete".Bileşim maddesi[2]
40003	MW104	DB10.DBW4	MW124	DB10.DBW54	"Reçete".Bileşim maddesi[3]
40004	MW106	DB10.DBW6	MW126	DB10.DBW56	"Reçete".Bileşim maddesi[4]
40005	MW108	DB10.DBW8	MW128	DB10.DBW58	"Reçete".Bileşim maddesi[5]

Tablo 13-89 Teşhis fonksiyonları

S7-1200 Modbus_Slave'inin Modbus teşhis fonksiyonları		
Kodlar	Kısmi fonksiyon	Açıklama
08	0000H	Yankı testi sorgu verilerini yayınlar: • STEP 7 V15.1'den önce Modbus_Slave yordamı, alınan bir veri sözcüğünün yankısını bir Modbus Master'a geri döndürür. • STEP 7 V15.1'den itibaren, Modbus_Slave V4.1 veya daha yüksek yordamlar bir veya daha fazla alınan veri sözcüğünün yankılarını geri döndürür.
08	000AH	İletişim olayı sayacını sil: Modbus_Slave yordamı, Modbus fonksiyonu 11 için kullanılan iletişim olay sayacını siler.
11		İletişim olayı sayacını çağır: Modbus_Slave yordamı, Modbus_Slave'e gönderilen başarılı Modbus okuma ve yazma taleplerinin sayısını saymak için dahili bir iletişim olay sayacı kullanır. Fonksiyon 8, fonksiyon 11 ve Broadcast talepleri için sayaç artırılmaz. Bunun dışında, iletişim hatalarıyla (örn. parite veya CRC hataları) sonuçlanan talepler için de sayaç artırılmaz.

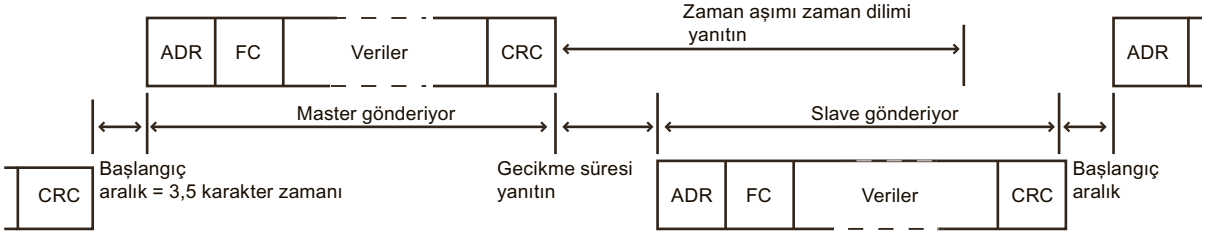
Modbus_Slave yordamı, talepler geçerli adreslere erişime yönelik olduğu sürece Modbus Master'larından gelen Broadcast yazma taleplerini destekler. Broadcast fonksiyonunda desteklenmeyen fonksiyon kodları için Modbus_Slave "0x8188" hata kodunu üretir.

Modbus_Slave iletişimi kuralları

- Modbus_Slave yordamının bu portla iletişim kurabilmesi için, bir portu yapılandırmak üzere Modbus_Comm_Load yordamının yürütülmesi gerekir.
- Bir portun bir Modbus_Master'a Slave olarak tepki vermesi isteniyorsa, bu port Modbus_Master yordamı ile programlanmamalıdır.
- Yalnızca bir Modbus_Slave örneği belirli bir port ile kullanılabilir, yoksa beklenmedik bir davranış ortaya çıkabilir.
- Modbus yordamları, iletişim sürecine kumanda etmek için iletişim alarm olaylarını kullanmaz. Programınız, tamamlanan gönderme ve alma işlemleri için Modbus_Slave yordamını sorgulayarak iletişim prosesine kumanda etmelidir.
- Modbus_Slave yordamı, bir Modbus_Master'dan gelen taleplere yanıt verilmesini sağlayacak bir sıklıkta düzenli olarak yürütülmelidir. Modbus_Slave yordamının bir OB program çevriminden itibaren her çevrimde yürütülmesi önerilir. Modbus_Slave, bir uyandırma alarmı OB'sinden yürütülebilir, ancak kesme programındaki zaman gecikmeleri aşırıya kaçarsa, diğer kesme programlarının yürütülmesini geçici olarak engelleyebilir, dolayısıyla bu yöntem önerilmez.

Modbus sinyalinin zaman kumandası

Modbus_Slave, Modbus_Master'dan herhangi bir talep almak ve buna yanıt vermek için düzenli olarak yürütülmelidir. Modbus_Slave'in yürütülme sıklığı Modbus_Master tarafından belirtilen yanıt zaman aşımı değerine göre değişir. Bu, aşağıdaki görselde sunulmaktadır.



RESP_TO yanıtı için zaman aşımı süresi, bir Modbus_Master'ın bir Modbus_Slave'den yanıtın başlamasını beklediği süredir. Bu zaman dilimi Modbus protokolü yerine ilgili Modbus_Master'ın bir parametresince belirlenir. Modbus_Slave'in yürütme sıklığı (bir yürütme ile bir sonraki yürütme arasında geçen süre) Modbus_Master'ın ilgili parametrelerine bağlı olmalıdır. Modbus_Master'ın yanıtı için zaman aşımı süresi boyunca Modbus_Slave'i en az iki kez yürütmelisiniz.

Modbus_Slave yordamının veri bloğundaki (DB) değişkenler

Aşağıdaki tabloda, Modbus_Slave örnek DB'sinde programınızda kullanabileceğiniz açık statik değişkenler gösterilmektedir:

Tablo 13-90 Modbus_Slave yordamının örnek DB'sindeki statik değişkenler

Değişken	Veri tipi	Standart değer	Açıklama
HR_Start_Offset	Word	0	Modbus tutma kaydının başlangıç adresini atar (standart değer = 0)
Extended_Addressing	Bool	YANLIŞ	Slave adreslemesini tek veya çift Bayt olarak yapılandırır: • DOĞRU = 1 Bayt'lık adres • DOĞRU = 2 Bayt'lık adres
Request_Count	Word	0	Bu Slave tarafından alınan tüm taleplerin sayısı
Slave_Message_Count	Word	0	Bu spesifik Slave için alınan taleplerin sayısı
Bad_CRC_Count	Word	0	CRC hatası içeren alınan taleplerin sayısı
Broadcast_Count	Word	0	Alınan Broadcast taleplerinin sayısı
Exception_Count	Word	0	Master'a geri döndürülen bir istisna ile onaylama gerektiren Modbus'a özgü hatalar.
Success_Count	Word	0	Bu spesifik Slave için alınan protokol hatasız taleplerin sayısı
MB_DB	MB_BASE	-	MB_DB yordamının Modbus_Comm_Load parametresi, MB_DB yordamının Modbus_Slave parametresine bağlanmalıdır.
QB_Start	UInt	0	CPU'nun yazabileceği çıkış Bayt'larının başlangıç adresi (AB0 ila AB65535)
QB_Count	UInt	65535	Bir uzak cihazın yazabileceği Bayt sayısı. QB_Count = 0 ise, bir uzak cihaz çıkışlara yazamaz. Örnek: Yalnızca AB10 ila AB17 yazılabilir olacaksa, QB_Start = 10 ve QB_Count = 8.
QB_Read_Start	UInt	0	CPU'nun okuyabileceği çıkış Bayt'larının başlangıç adresi (AB0 ila AB65535)
QB_Read_Count	UInt	65535	Uzak bir cihazın okuyabileceği çıkış Bayt'ı sayısı. QB_Count = 0 ise, bir uzak cihaz çıkışlardan okuyamaz. Örnek: Yalnızca AB10 ila AB17 okunabilirse, QB_Start = 10 ve QB_Count = 8.
IB_Read_Start	UInt	0	CPU'nun okuyabileceği giriş Bayt'larının başlangıç adresi (EB0 ila EB65535)
IB_Read_Count	UInt	65535	Uzak bir cihazın okuyabileceği giriş Bayt'ı sayısı. IB_Count = 0 ise, bir uzak cihaz girişlerden okuyamaz. Örnek: Yalnızca EB10 ila EB17 okunabilirse, IB_Start = 10 ve IB_Count = 8.

Programınız, kumanda eden Modbus sunucu işlemlerine ve aşağıdaki değişkenlere veri yazabilir:

- HR_Start_Offset
- Extended_Addressing
- QB_Start
- QB_Count
- QB_Read_Start
- QB_Read_Count
- IB_Read_Start
- IB_Read_Count

Aşağıdaki sürüm gereklilikleri, MB_Slave yordamının veri bloğundaki (DB) değişkenlerin kullanılabilirliği için geçerlidir:

Tablo 13-91 MB_Slave yordamının veri bloğundaki (DB) değişkenlerin kullanılabilirliği için sürüm gereklilikleri: Yordam, TIA Portal ve S7-1200 CPU

MB_Slave yordamının sürümü	TIA Portal sürümü	S7-1200 CPU Firmware'inin (FW) sürümü	Veri bloğu değişkenleri
3.0	V14 SP1	CPU FW V4.0 veya üstü	QB_Start
			QB_Count
4.0 veya üstü	V15 veya üstü	CPU FW V4.2 veya üstü	QB_Start
			QB_Count
			QB_Read_Start
			QB_Read_Count
			IB_Read_Start
			IB_Read_Count

HR_Start_Offset

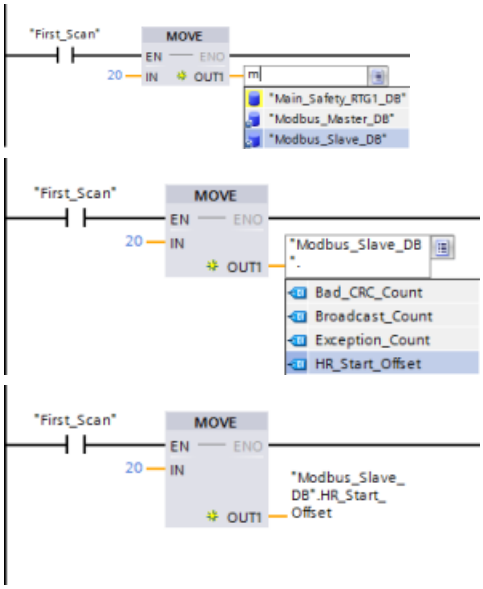
Modbus tutma kaydı adresleri, 40001 veya 400001'den başlar. Bu adresler, hedef sistem belleğindeki tutma kaydının başlangıç adresine karşılık gelir. Ancak Modbus tutma kaydı için 40001 veya 400001 dışında bir başlangıç adresi yapılandırmak isterseniz, HR_Start_Offset değişkenini yapılandırabilirsiniz.
Örneğin, MW100'de başlayan ve 100 sözcük uzunluğunda olan bir tutma kaydı yapılandırabilirsiniz. Tutma kaydı başlangıç adresini, 20'lik bir ofset ile 40001 yerine 40021 olarak belirtirsiniz. Bu durumda 40021'in altındaki ve 400119'un üzerindeki her adres, adresleme hatasına yol açacaktır.

Tablo 13-92 Modbus tutma kaydının adresleme örneği

HR_Start_Offset	Adres	Minimum	Maksimum
0	Modbus adresi (sözcük)	40001	40099
	S7-1200 adresi	MW100	MW298
20	Modbus adresi (sözcük)	40021	40119
	S7-1200 adresi	MW100	MW298

HR_Start_Offset, Modbus tutma kaydının başlangıç adresini belirten bir sözcük değeridir ve Modbus_Slave örnek veri bloğunda saklanır. Modbus_Slave'i programınıza eklenmesinin ardından, bu açık statik değişkeni parametre açılır listesinden seçebilirsiniz.

Örneğin bir LAD ağına Modbus_Slave eklediyseniz, bir önceki bir ağa gidip, HR_Start_Offset değerini atayabilirsiniz. Değer, Modbus_Slave yürütülmeden önce atanmalıdır.



Modbus Slave değişkenini, standart DB adı ile girin.

1. İmleci parametre alanına getirin ve m karakterini girin.
2. Açılır listeden "Modbus_Slave_DB" öğesini seçin.
3. İmleci DB adının sağına getirin (tırnak işaretinden sonra) ve bir nokta girin.
4. Açılır listeden "Modbus_Slave_DB.HR_Start_Offset" öğesini seçin.

Extended_Addressing

Yukarıda açıklanan HR_Start_Offset referansına benzer bir şekilde Extended_Addressing değişkenine erişilir, aradaki fark, Extended_Addressing değişkeninin bir Boole değeri olmasıdır. Boole değeri bir çıkış bobininden yazılmalıdır ve bir "Kaydır" kutusundan yazılamaz. Modbus Slave adreslemesi, tek Bayt (bu, Modbus standardıdır) veya çift Bayt olarak yapılandırılabilir. Genişletilmiş adresleme yöntemi, tek bir ağda 247'den fazla cihazı adreslemek için kullanılır. Genişletilmiş adreslemeyi tercih ederseniz, en fazla 64.000 adresi adresleyebilirsiniz. Aşağıda Modbus fonksiyon 1'in bir telegramı örnek olarak verilmektedir.

Tablo 13-93 Bir Bayt'lık Slave adresi (Bayt 0)

Fonksiyon 1	**Bayt 0**	**Bayt 1**	**Bayt 2**	**Bayt 3**	**Bayt 4**	**Bayt 5**	
Talep	Slave adresi	F kodu	Başlangıç adresi		Bobinlerin uzunluğu		
Geçerli yanıt	Slave adresi	F kodu	Uzunluk	Bobin verileri			
Hata yanıtı	Slave adresi	0x81	E kodu				

Tablo 13-94 İki Bayt'lık Slave adresi (Bayt 0 veya Bayt 1)

	Bayt 0	**Bayt 1**	**Bayt 2**	**Bayt 3**	**Bayt 4**	**Bayt 5**	**Bayt 6**
Talep	Slave adresi		F kodu	Başlangıç adresi		Bobinlerin uzunluğu	
Geçerli yanıt	Slave adresi		F kodu	Uzunluk	Bobin verileri		
Hata yanıtı	Slave adresi		0x81	E kodu			

Modbus adreslerine doğrudan erişim yerine veri bloklarındaki (DB) veri alanlarına erişim

DB'lerdeki veri alanlarına global DB'nin "Öznitelikler" özellik sayfasından erişebilirsiniz. "Yalnızca yükleme belleğinde sakla" ve "Optimize edilmiş blok erişimi" opsiyon kutularının seçimi kaldırılmalıdır.

Bir Modbus talebi geldiğinde ve karşılık gelen fonksiyon kodunun Modbus veri tipi için herhangi bir veri alanı ayarlanmadığında, Modbus_Slave yordamı bir talebi, bu yordamın önceki sürümlerinde olduğu gibi işler: Proses görünümlerine ve tutma kayıtlarına doğrudan erişebilirsiniz.

Fonksiyon kodunun Modbus veri tipi için bir veri alanı belirtilmişse, Modbus_Slave yordamı bu veri alanından okur veya bu veri alanına yazar. Okunması veya yazılması görevin türüne göre değişir.

NOT

Bir veri alanı yapılandırıldığında, Modbus_Slave yordamı, veri alanının veri tipine karşılık gelen örnek veri bloğundaki statik değişkenler tarafından yapılandırılan ofsetleri veya alanları yok sayar. Bu ofsetler ve alanlar, yalnızca MB_HOLD_REG tarafından referanslanan proses görünümü veya bellek için geçerlidir. Veri alanının başlangıç ve uzunluk parametreleri, ofsetleri ve alanları tanımlamada benzersiz bir yöntem sağlar.

Tek bir Modbus talebi, yalnızca bir veri alanına yazabilir veya buradan okuyabilir. Örneğin, birden fazla veri alanına yayılan tutma kayıtları okunacaksa, birkaç Modbus talebi gerekir.

Veri alanlarının belirlenmesinde aşağıdaki kurallar geçerlidir:

- Farklı DB'lerde en fazla sekiz veri alanı belirleyebilirsiniz, bu DB'lerden her birinin yalnızca bir veri alanı içermesine izin verilir. Bir MODBUS talebi, yalnızca tek bir veri alanından okuyabilir veya bu alana yazabilir. Her veri alanı bir MODBUS adres alanına karşılık gelir. Veri alanlarını, örnek DB'nin "Data_Area_Array" statik değişkeninde belirlersiniz.
- Sekizden az veri alanı kullanılacaksa, gerek duyduğunuz veri alanlarının aralıksız olarak birbiri ardına düzenlemelisiniz. Veri alanlarındaki ilk boş veri girişinde, işlem sırasında veri alanı araması durdurulur. Örneğin, saha elemanları 1, 2, 4 ve 5 veri alanları olarak belirtilmiştir, saha elemanı 3 boş olduğu için "Data_Area_Array", yalnızca saha elemanları 1 ve 2'yi algılar.
- Data_Area_Array, sekiz elemandan oluşur: Data_Area_Array[1] ila Data_Area_Array[8]
- Her Data_Area_Array[x], $1 \leq x \leq 8$ saha elemanı, MB_DataArea tipinde bir UDT'dir ve aşağıdaki şekilde yapılandırılmıştır:

Parametre	Veri tipi	Anlamı
data_ type	UInt	Bu veri alanına eşlenen MODBUS veri tipi için kod: – 0: Boş bir saha elemanı veya kullanılmayan bir veri alanı için kod. Bu durumda; db, start ve length değerleri önemsizdir. – 1: Proses görünümü çıkışı (fonksiyon kodları 1, 5 ve 15 ile ilişkili olarak) – 2: Proses görünümü girişi (fonksiyon kodu 2 ile ilişkili olarak) – 3: Tutma kayıtları (fonksiyon kodları 3, 6 ve 16 ile ilişkili olarak) – 4: Giriş kaydı (fonksiyon kodu 4 ile bağlantılı olarak) Not: Bir MODBUS veri tipi için bir veri alanı tanımladıysanız, Modbus_Slave yordamı artık bu MODBUS veri tipine doğrudan erişemez. Bu tür bir veri tipi için bir MODBUS talebinin adresi belirtilen bir veri alanına karşılık gelmiyorsa, STATUS'ta W#16#8383 değeri geri döndürülür.

Parametre	Veri tipi	Anlamı
db	UInt	MODBUS kaydının veya sonra belirtilen Bit'lerin eşlendiği veri bloğunun numarası DB numarası, veri alanlarında bir kez mevcut olmalıdır. DB numarası, birden fazla veri alanında mevcut olmamalıdır. Global DB'nin "Öznitelikler" özellikler sayfasında, "Yalnızca yükleme belleğinde sakla" ve "Optimize edilmiş modül erişimi" opsiyon kutularının seçimi kaldırılmalıdır. Veri alanları da DB'nin Bayt adresi 0 ile başlar. İzin verilen değerler: 1 ila 60999
start	UInt	0.0 adresinden başlamak üzere veri bloğuna eşlenen ilk MODBUS adresi İzin verilen değerler: 0 ila 65535
length	UInt	Bit sayısı (data_type değerleri 1 ve 2 için) veya kayıt sayısı (data_type değerleri 3 ve 4 için) Bir MODBUS veri tipinin MODBUS adres aralıkları çakışmamalıdır. İzin verilen değerler: 1 ila 65535

Veri alanlarını belirlenme örnekleri:

- Birinci örnek: data_type = 3, db = 1, start = 10, length = 6
  CPU, veri bloğu 1'deki (db = 1) tutma kayıtlarını (data_type = 3) eşler ve Modbus adresi 10'u (start = 10) veri sözcüğü 0'a ve son geçerli Modbus adresi 15'i (length = 6) veri sözcüğü 5'e saklar.
- İkinci örnek: data_type = 2, db = 15, start = 1700, length = 112
  CPU, girişleri (data_type = 2) veri bloğu 15'e (db = 15) eşler ve Modbus adresi 1700'ü (start = 1700) veri sözcüğü 0'a ve son geçerli Modbus adresi 1811'i (length = 112) veri sözcüğü 111'e saklar.

Koşul kodları

Tablo 13-95 Modbus_Slave'ın yürütülmesi için durum kodları (iletişim ve konfigürasyon hataları) [1]

STATUS (W#16#)	Açıklama
80D1	Alıcı cihaz devam etmekte olan bir aktarımı kesintiye uğratmak için akış kontrolü talep etmiş, ancak bekleme süresi içinde aktarımı yeniden etkinleştirmemiştir. Alıcı cihaz bekleme süresi içinde bir CTS algılamazsa, donanım akış kontrolü sırasında da bu hata üretilir.
80D2	DCE'den DSR sinyali alınmadığı için gönderim talebi iptal edildi.
80E0	Alım tamponu dolu olduğu için mesaj sonlandırıldı.
80E1	Mesaj, bir parite hatası nedeniyle sonlandırıldı.
80E2	Mesaj, bir telegram hatası nedeniyle sonlandırıldı.
80E3	Mesaj, bir taşma hatası nedeniyle sonlandırıldı.
80E4	Belirtilen uzunluk toplam tampon boyutunu aştığı için mesaj sonlandırıldı.
8180	Port ID için geçersiz değer veya Modbus_Comm_Load yordamında hata.
8186	Geçersiz Modbus istasyon adresi
8187	MB_HOLD_REG DB'si üzerinde geçersiz işaretçi. Aralık, çok küçük.

STATUS (W#16#)	Açıklama
818C	Geçersiz işaretçi MB_HOLD_REG Veri alanı aşağıdakilerden biri olmalıdır: • Klasik DB • Sembolik veya bir kalıcı DB'deki temel veri tipleri Array'i • Markör

Tablo 13-96 Modbus_Slave'in yürütülmesi için koşul kodları Modbus protokol hataları) [1]

STATUS (W#16#)	Slave'in yanıt kodu	Modbus protokol hatası
8380	Yanıt yok	CRC hatası
8381	01	Fonksiyon kodu desteklenmiyor veya Broadcast'larda desteklenmiyor
8382	03	Veri uzunluğunda hata
8383	02	Veri adresi hatası veya adres, DATA_PTR'nin geçerli aralığın dışında
8384	03	Veri değerinde hata
8385	03	Veri teşhis kodunun değeri desteklenmiyor (fonksiyon kodu 08)
8389		Geçersiz veri alanı tanımı: • Veri tipi için geçersiz değer • DB numarası geçersiz veya mevcut değil: – Geçersiz DB değeri – DB numarası mevcut değil – DB numarası halen başka bir veri alanı tarafından kullanılıyor – Optimize edilmiş erişime sahip DB – DB, çalışma belleğinde değil • Geçersiz uzunluk değeri • Aynı MODBUS veri tipine ait çakışan MODBUS adres aralıkları

[1] Yukarıda belirtilen Modbus_Slave yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

NOT

PROFIBUS iletişimi için veri setinin maksimum uzunluğunun ayarlanması

CM1243-5 iletişim modülü, bir RS232, RS422 veya RS485 noktadan noktaya modül kullanan bir ET200SP veya ET200MP PROFIBUS cihazına kumanda etmek için PROFIBUS Master olarak kullanılıyorsa, bu durumda "max_record_len" veri bloğu değişkenini aşağıdaki gibi belirtik olarak 240 değerine ayarlamalısınız:

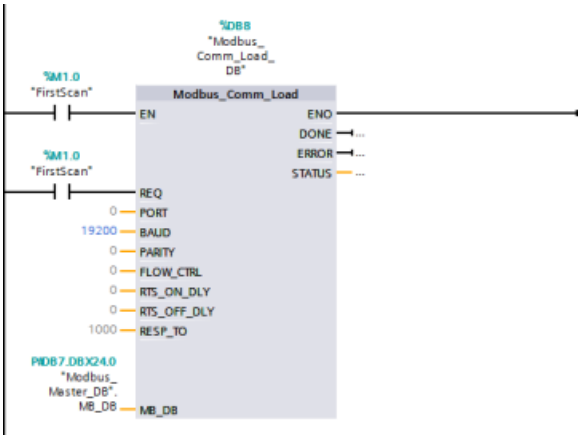
Modbus_Comm_Load yordamının yürütülmesinden çalıştırdıktan sonra, örnek DB'nin Send_P2P alt bölümündeki max_record_len parametresini (örneğin, "Modbus_Slave_DB".Send_P2P.max_record_len) 240 olarak ayarlayın.

max_record_len değerinin belirtik ataması yalnızca PROFIBUS bağlantıları için gereklidir. PROFINET bağlantılarında ise max_record_len için geçerli bir değer zaten kullanılmaktadır.

13.5.3.4 Modbus RTU örnekleri

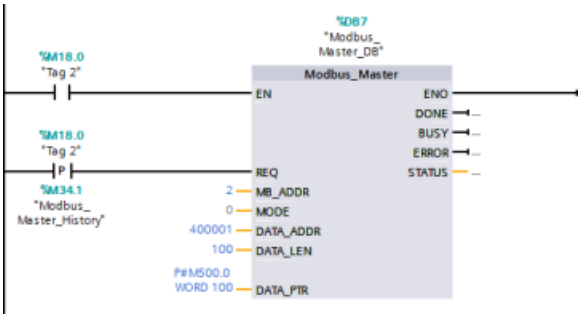
Modbus_Comm_Load, yol alma sırasında ilk çevrimin markörü üzerinden başlatılır. Modbus_Comm_Load sadece, seri portun konfigürasyonu çalışma zamanı sırasında değişmezse bu şekilde yürütülmelidir.

Ağ 1: RS485 modülünün iletişim portunu ilk çevrim sırasında yalnızca bir kez yapılandırın/başlatın.

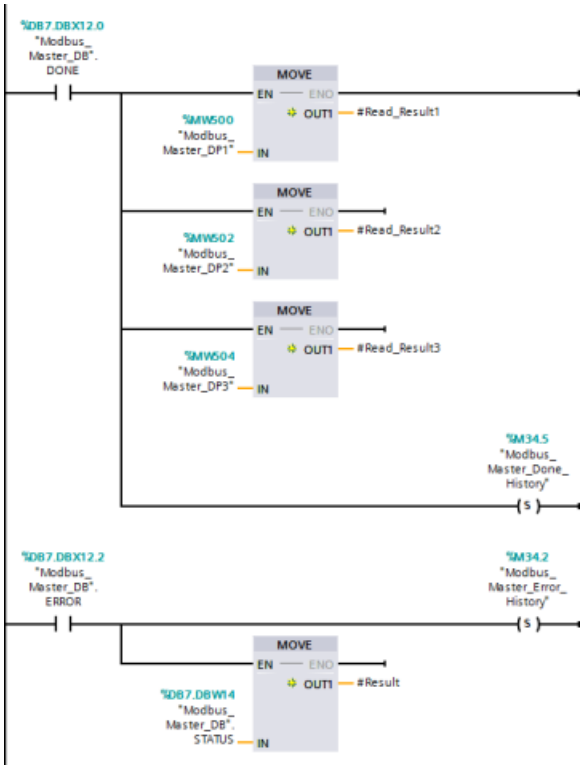


Bir Modbus_Master yordamını, bir Slave ile iletişim kurmak için OB program çevriminde kullanırsınız. Diğer Modbus_Master yordamları da OB program çevriminde diğer Slave'lerle iletişim kurmak veya bir Modbus_Master FB'si başka Slave'lerle iletişim için kullanılabilir.

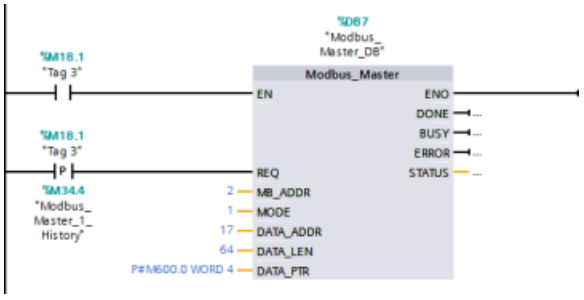
Ağ 2: MW500-MW698 adresindeki Slave No. 2'deki 400001 adresinden 100 sözcüklük tutma kaydı verisini okuyun.



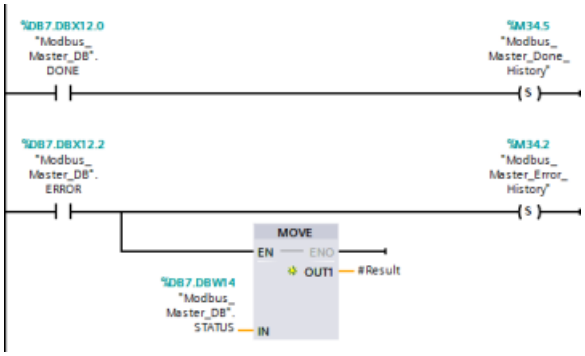
Ağ 3: Okunan tutma kaydı verilerinin ilk üç sözcüğünü başka bir adrese kaydırın ve bir DONE yapıldı Bit'i ayarlayın. Bu ağ ayrıca, bir hata oluştuğunda ERROR yapıldı Bit'ini ayarlar ve STATUS sözcüğünü başka bir adreste saklar.



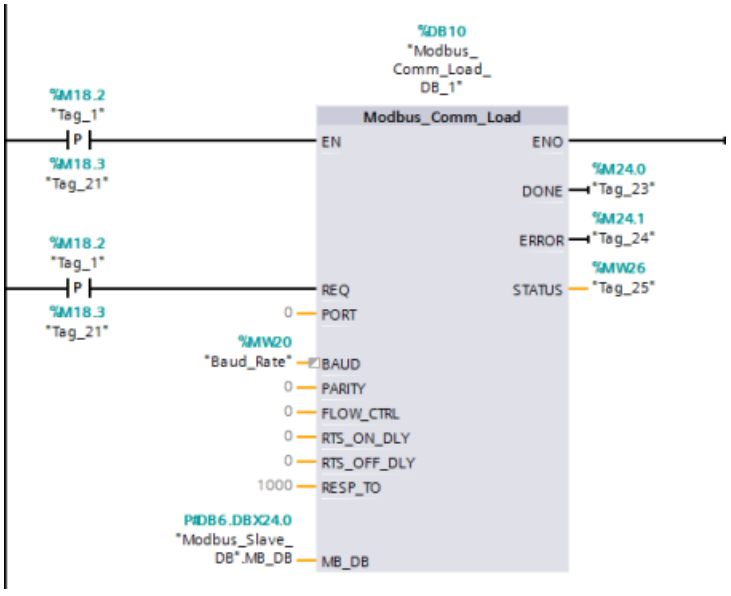
Ağ 4: Slave no. 2'de MW600-MW607'den 64 Bit'lik veriyi 00017 ila 00081 çıkış Bit'inin adreslerine yazın.



Ağ 5: Yazma işlemi bittiğinde yapıldı Bit'ini DONE olarak ayarlayın. Bir hata oluşursa, program, ERROR yapıldı Bit'ini ayarlar ve STATUS kodunu saklar.

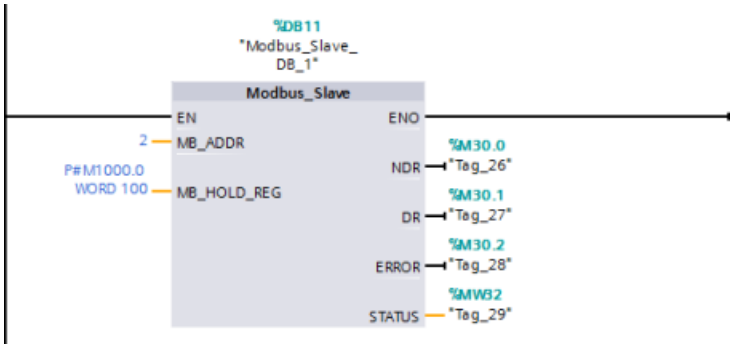


"Değişken_1"in her etkinleştirildiğinde, aşağıda gösterilen MB_COMM_LOAD yordamı başlatılır.
MB_COMM_LOAD yordamının bu şekilde yürütülmesi yalnızca, çalışma zamanı sırasında HMI konfigürasyonunun bir sonucu olarak seri port konfigürasyonu değişirse mümkündür.
Ağ 1: Bir HMI cihazı tarafından her değiştirildiklerinde RS485 modül parametrelerini başlatın.



Aşağıda gösterilen MB_SLAVE yordamı, her 10 msn'de bir yürütülen bir periyodik OB içindedir. Bu, Slave tarafından mutlak en hızlı yanıtı sağlamasa da kısa mesajlar için 9600 Baud hızında iyi bir işletim performansı sergiler (talepte 20 Bayt'a kadar).

Ağ 2: Her çevrimde Modbus Master'dan gelen talepleri kontrol edin. Modbus tutma kaydı için, MW1000'den başlamak üzere 100 sözcük yapılandırılır.



13.6 Eski sistemlerdeki PtP iletişimi yordamları (yalnızca CM/CB 1241)

STEP 7 V13 SP1 ve S7-1200 V4.1 CPU'lardan önce, PtP iletişim yordamları farklı adlara ve bazen de farklı arayüzlere sahipti. Noktadan noktaya iletişim (Sayfa 891) ve aynı şekilde Port (Sayfa 894) ve Parametre konfigürasyonu (Sayfa 908) için genel kavramlar her iki yordam türü için de geçerlidir. Programlamayla ilgili bilgileri, eski PtP yordamlarıyla ilgili açıklamalarda bulabilirsiniz.

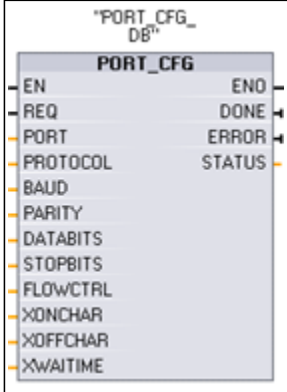
Tablo 13-97 Ortak hata sınıfları

Sınıfın açıklaması	Hata sınıfları	Açıklama
Port konfigürasyonu	80Ax	Arayüz konfigürasyonundaki yaygın hataların açıklaması için
Gönderim konfigürasyonu	80Bx	Gönderim konfigürasyonundaki yaygın hataların açıklaması için
Alım konfigürasyonu	80Cx	Alım konfigürasyonundaki yaygın hataların açıklaması için
Gönderim çalışma zamanı	80Dx	Gönderim sırasındaki yaygın çalışma zamanı hatalarının açıklaması için
Alım çalışma zamanı	80Ex	Alım sırasındaki yaygın çalışma zamanı hatalarının açıklaması için
Sinyal işleme:	80Fx	Sinyal işlemeye ilişkin yaygın hataların açıklaması için

13.6.1 Eski noktadan noktaya yordamlar

13.6.1.1 Port_Config (iletişim parametrelerinin dinamik olarak yapılandırılması)

Tablo 13-98 PORT_CFG yordamı (port konfigürasyonu)

LAD/FBD	SCL	Açıklama
	**"PORT_CFG_DB"(** **REQ:=_bool_in_,** **PORT:=_uint_in_,** **PROTOCOL:=_uint_in_,** **BAUD:=_uint_in_,** **PARITY:=_uint_in_,** **DATABITS:=_uint_in_,** **STOPBITS:=_uint_in_,** **FLOWCTRL:=_uint_in_,** **XONCHAR:=_char_in_,** **XOFFCHAR:=_char_in_,** **WAITTIME:=_uint_in_,** **DONE=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_);**	PORT_CFG ile Baud hızı gibi port parametrelerini programınız aracılığıyla değiştirebilirsiniz. Cihaz konfigürasyonu özelliklerinde portun statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. Bunun ardından PORT_CFG ile programınızdaki konfigürasyonu değiştirebilirsiniz.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

PORT_CFG'nin konfigürasyon değişiklikleri CPU'da kalıcı olarak saklanmaz. Cihaz konfigürasyonunda saklanan parametreler, CPU'nun RUN'dan STOP işletim durumuna geçmesinde veya tekrar açıldığında geri yüklenir. Daha fazla bilgi için, İletişim portlarının yapılandırılması (Sayfa 894) ve Akış kontrolünü yönetimi (Sayfa 896) bölümlerine bakın.

Tablo 13-99 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişin yükselen kenarında konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
PROTOCOL	IN	UInt	0 - Noktadan noktaya iletişim protokolü (standart değer). 1..n - belirli protokollerin gelecekteki tanımı
BAUD	IN	UInt	Portun Baud hızı (standart değer: 6): 1 = 300 Baud, 2 = 600 Baud, 3 = 1200 Baud, 4 = 2400 Baud, 5 = 4800 Baud, 6 = 9600 Baud, 7 = 19200 Baud, 8 = 38400 Baud, 9 = 57600 Baud, 10 = 76800 Baud, 11 = 115200 Baud
PARITY	IN	UInt	Portun paritesi (standart değer: 1): 1 = Parite yok, 2 = Çift sayılı parite, 3 = Tek sayılı parite, 4 = Markör paritesi, 5 = Boşluk paritesi
DATABITS	IN	UInt	Karakter başına Bit (standart değer): 1): 1 = 8 veri Bit'i, 2 = 7 veri Bit'i
STOPBITS	IN	UInt	Stop Bit'leri (standart değer: 1): 1 = 1 Stop Bit'i, 2 = 2 Stop Bit'i

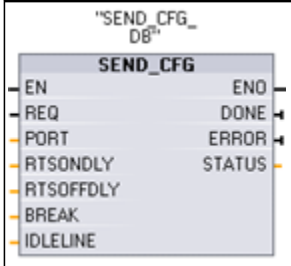
Parametre ve veri tipi		Veri tipi	Açıklama
FLOWCTRL	IN	UInt	Akış kontrolü (standart değer: 1): 1 = Akış kontrolü yok, 2 = XON/XOFF, 3 = Donanım RTS her zaman AÇIK, 4 = Donanım RTS anahtarlanmış
XONCHAR	IN	Char	XON karakteri olarak çalışan karakteri belirtir. Bu, tipik olarak bir DC1 karakteridir (16#11). Bu parametre, yalnızca akış kontrolü etkinleştirildiğinde değerlendirilir. (Standart değer: 16#11)
XOFFCHAR	IN	Char	XOFF karakteri olarak çalışan karakteri belirtir. Bu, tipik olarak bir DC3 karakteridir (116#3). Bu parametre, yalnızca akış kontrolü etkinleştirildiğinde değerlendirilir. (Standart değer: 16#13)
XWAITIME	IN	UInt	Bir XON karakterini aldıktan sonra bir XOFF karakteri için ne kadar süre bekleneceğini veya RTS'yi etkinleştirdikten sonra CTS sinyali için ne kadar süre bekleneceğini belirtir (0 ila 65535 msn). Bu parametre, yalnızca akış kontrolü etkinleştirildiğinde değerlendirilir. (Standart değer: 2000)
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Tablo 13-100 Koşul kodları

STATUS (W#16#....)	Açıklama
80A0	Belirtilen protokol mevcut değil.
80A1	Belirtilen Baud hızı mevcut değil.
80A2	Belirtilen parite opsiyonu mevcut değil.
80A3	Belirtilen sayıda veri Bit'i mevcut değil.
80A4	Belirtilen sayıda Stop Bit'i mevcut değil.
80A5	Belirtilen akış kontrol tipi mevcut değil.
80A6	Bekleme süresi 0 ve akış kontrolü aktif
80A7	XON ve XOFF, geçersiz değerde (örn. aynı değer)

13.6.1.2 SEND_CFG (gönderim konfigürasyonu)

Tablo 13-101 SEND_CFG yordamı (gönderim konfigürasyonu)

LAD/FBD	SCL	Açıklama
	**"SEND_CFG_DB"(REQ:=_bool_in_, PORT:=_uint_in_, RTSONDLY:=_uint_in_, RTSOFFDLY:=_uint_in_, BREAK:=_uint_in_, IDLELINE:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	SEND_CFG, bir PtP iletişim arayüzü için seri aktarım parametrelerinin dinamik konfigürasyonuna olanak tanır. Bir CM veya CB'de mevcut mesajlar, SEND_CFG yürütüldüğünde silinir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Cihaz konfigürasyonu özelliklerinde portun statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. Bunun ardından SEND_CFG ile programınızdaki konfigürasyonu değiştirebilirsiniz.

SEND_CFG'nin konfigürasyon değişiklikleri CPU'da kalıcı olarak saklanmaz. Cihaz konfigürasyonunda saklanan parametreler, CPU'nun RUN'dan STOP işletim durumuna geçmesinde veya tekrar açıldığında geri yüklenir. Bkz. Gönderim parametrelerinin yapılandırılması

Tablo 13-102 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişte yükselen kenardaki konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	Bir CM veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
RTSONDLY	IN	UInt	RTS'yi etkinleştirdikten sonra Tx verilerinin aktarımını gerçekleştirmeden önce beklenecek milisaniye sayısı. Bu parametre, yalnızca donanım akış kontrolü etkinleştirildiğinde geçerlidir. Geçerli aralık, 0 ila 65535 msn'dir. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 0)
RTSOFFDLY	IN	UInt	RTS'yi devre dışı bırakmadan önce, Tx verilerinin aktarımından sonra beklenecek milisaniye sayısı: Bu parametre, yalnızca donanım akış kontrolü etkinleştirildiğinde geçerlidir. Geçerli aralık, 0 ila 65535 msn'dir. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 0)
BREAK	IN	UInt	Bu parametre, Bit sürelerinin belirtilen sayısı için her mesajın başlangıcında bir duraklama gönderildiğini belirtir. En fazla 8 saniyeye kadar maksimum 65535 Bit süresi mümkündür. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 12)
IDLELINE	IN	UInt	Bu parametre, her mesajın başlangıcından önce belirtilen sayıda Bit süresi boyunca hattın boşta çalışacağını belirtir. En fazla 8 saniyeye kadar maksimum 65535 Bit süresi mümkündür. 0 değeri, fonksiyonu devre dışı bırakır. (Standart değer: 12)

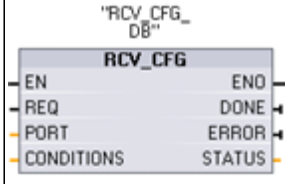
Parametre ve veri tipi		Veri tipi	Açıklama
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

Tablo 13-103 Koşul kodları

STATUS (W#16#....)	Açıklama
80B0	Bir gönderim alarmının konfigürasyonuna izin verilmez.
80B1	Duraklama süresi, izin verilen maksimum değerden büyüktür.
80B2	Boşta çalışma süresi, izin verilen maksimum değerden büyüktür.

13.6.1.3 RCV_CFG (alım konfigürasyonu)

Tablo 13-104 RCV_CFG yordamı (alım konfigürasyonu)

LAD/FBD	SCL	Açıklama
	**"RCV_CFG_DB"(REQ:=_bool_in_, PORT:=_uint_in_, CONDITIONS:=_struct_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	RCV_CFG bir PtP iletişim arayüzü için seri alım parametrelerinin dinamik konfigürasyonunu gerçekleştirir. Bu yordam, alınan bir mesajın başlangıcını ve sonunu işaretleyen koşulları yapılandırır. Bir CM veya CB'de mevcut mesajlar, RCV_CFG yürütüldüğünde silinir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Cihaz konfigürasyonu özelliklerinde iletişim bağlantısının statik başlangıç konfigürasyonunu ayarlayabilir veya yalnızca standart değerleri kullanabilirsiniz. Bunun ardından RCV_CFG ile programınızdaki konfigürasyonu değiştirebilirsiniz.

RCV_CFG'nin konfigürasyon değişiklikleri CPU'da kalıcı olarak saklanmaz. Cihaz konfigürasyonunda saklanan parametreler, CPU'nun RUN'dan STOP işletim durumuna geçmesinde veya tekrar açıldığında geri yüklenir. Daha fazla bilgiyi, Alım parametrelerinin yapılandırılması (Sayfa 899) altında bulabilirsiniz.

Tablo 13-105 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişte yükselen kenardaki konfigürasyon değişikliğini etkinleştirir. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
CONDITIONS	IN	CONDITIONS	CONDITIONS veri yapısı, aşağıda açıklandığı gibi mesajın başlangıç ve bitiş koşullarını belirtir.
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir çevrim boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatalı tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

RCV_PTP yordamı için başlangıç koşulları

RCV_PTP yordamı, RCV_CFG yordamı tarafından belirtilen konfigürasyonu kullanarak, noktadan noktaya iletişim mesajlarının başlangıcını ve sonunu belirler. Mesajın başlangıcı, başlangıç koşulları tarafından belirlenir. Mesajın başlangıcı, bir veya daha fazla başlangıç koşuluna göre belirlenebilir. Birden çok başlangıç koşulu belirtilmişse, mesajın başlaması için tüm koşulların karşılanması gerekir.

Mesaj başlangıcı için koşullara ilişkin bir açıklamayı, "Alım parametrelerinin yapılandırılması (Sayfa 899)" altında bulabilirsiniz.

CONDITIONS parametresinin veri tipi yapısı, bölüm 1 (başlangıç koşulları)

Tablo 13-106 START koşulları için CONDITIONS yapısı

Parametre ve veri tipi		Veri tipi	Açıklama
STARTCOND	IN	UInt	Başlangıç koşulunun belirtilmesi (standart değer: 1) • 01H - Başlangıç karakteri • 02H - Herhangi bir karakter • 04H - Hat duraklaması • 08H - Hat boşta • 10H - Karakter dizisi 1 • 20H - Karakter dizisi 2 • 40H - Karakter dizisi 3 • 80H - Karakter dizisi 4
IDLETIME	IN	UInt	Hat boşta çalışmanın zaman aşımı için gereken Bit süresi. (Standart değer: 40). Sadece "Hat boşta" durumuyla bağlantılı olarak. 0 ila 65535
STARTCHAR	IN	Byte	"Başlangıç karakteri" koşulu için başlangıç karakteri. (Standart değer: B#16#2)
SEQ[1].CTL	IN	Byte	Karakter dizisi 1, her karakter için kumandayı yok say/karşılaştır: (Standart değer: B#16#0) Bunlar, başlangıç karakter dizisindeki her karakter için etkinleştirme Bit'leridir. • 01H - Karakter 1 • 02H - Karakter 2 • 04H - Karakter 3 • 08H - Karakter 4 • 10H - Karakter 5 Bir Bit'in belirli bir karakter için devre dışı bırakılması, karakter dizisinin o pozisyonunda her bir karakterin bir eşleşme olduğu anlamına gelir.
SEQ[1].STR	IN	Char[5]	Karakter dizisi 1, başlangıç karakteri (5 karakter). Standart değer: 0
SEQ[2].CTL	IN	Byte	Karakter dizisi 2, her karakter için kumandayı yok say/karşılaştır. Standart değer: B#16#0)
SEQ[2].STR	IN	Char[5]	Karakter dizisi 2, başlangıç karakteri (5 karakter). Standart değer: 0
SEQ[3].CTL	IN	Byte	Karakter dizisi 3, her karakter için kumandayı yok say/karşılaştır. Standart değer: B#16#0
SEQ[3].STR	IN	Char[5]	Karakter dizisi 3, başlangıç karakteri (5 karakter). Standart değer: 0
SEQ[4].CTL	IN	Byte	Karakter dizisi 4, her karakter için kumandayı yok say/karşılaştır. Standart değer: B#16#0
SEQ[4].STR	IN	Char[5]	Karakter dizisi 4, başlangıç karakteri (5 karakter), standart değer: 0

Örnek

Aşağıda gösterilen alınmış onaltılık kodlanmış mesajı göz önünde bulundurun: "**68** 10 aa **68** bb 10 aa 16". Yapılandırılmış başlangıç karakter dizileri aşağıdaki tabloda gösterilmektedir. İlk karakter 68H başarıyla alındıktan sonra, başlangıç karakter dizileri değerlendirilir. Dördüncü karakterin (ikinci 68H) başarıyla alımından sonra, başlangıç koşulu 1 karşılanmış olur. Başlangıç koşulları karşılandığında, bitiş koşulları değerlendirilir.

Parite, çerçeveleme veya karakterler arasındaki zaman aralıkları gibi çeşitli hatalar nedeniyle başlangıç karakter dizisinin işlenmesi durabilir. Burada başlangıç koşulu karşılanmadığından, bu hatalar mesajın alınmamasına neden olur.

Tablo 13-107 Başlangıç koşulları

Başlangıç koşulu	İlk karakter	İlk karakter +1	İlk karakter +2	İlk karakter +3	İlk karakter +4
1	**68**H	xx	xx	**68**H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

RCV_PTP yordamı için bitiş koşulları

Bir mesajın sonu, belirtilen bitiş koşulları tarafından belirlenir. Bir mesajın sonu, yapılandırılmış bir veya daha fazla bitiş koşulunun ilk kez ortaya çıkışıyla belirlenir. "Alım parametrelerinin yapılandırılması (Sayfa 899)" altındaki "Mesaj başlangıç koşulları" alt bölümü, RCV_CFG yordamı için yapılandırabileceğiniz bitiş koşullarını açıklanmaktadır.

Bitiş koşullarını, cihaz konfigürasyonunda iletişim arayüzünün özelliklerinde veya RCV_CFG yordamını kullanarak yapılandırabilirsiniz. CPU'nun STOP'tan RUN işletim durumuna her geçişinde, alım parametreleri (başlangıç ve bitiş koşulları) cihaz konfigürasyonu ayarlarına sıfırlanır. STEP 7 kullanıcı programı RCV_CFG'i yürüttüğünde, ayarlar RCV_CFG'in koşullarına getirilir.

CONDITIONS parametresinin veri tipi yapısı, bölüm 2 (bitiş koşulları)

Tablo 13-108 END koşulları için CONDITIONS yapısı

Parametre	Parametre tipi	Veri tipi	Açıklama
ENDCOND	IN	UInt 0	Bu parametre, mesaj bitişi koşulunu belirtir: • 01H - Yanıt süresi • 02H - Mesaj süresi • 04H - Karakter aralığı • 08H - Maksimum uzunluk • 10H - N + LEN + M • 20H - Karakter dizisi
MAXLEN	IN	UInt 1	Maksimum mesaj uzunluğu: Yalnızca "Maksimum uzunluk" bitiş koşulu seçildiğinde, kullanılır. 1 ila 1024 Bayt
N	IN	UInt 0	Mesajdaki uzunluk alanının Bayt pozisyonu. Yalnızca N + LEN + M bitiş koşulunda kullanılır. 1 ila 1022 Bayt
LENGTHSIZE	IN	UInt 0	Uzunluk alanının boyutu (1, 2 veya 4 Bayt). Yalnızca N + LEN + M bitiş koşulunda kullanılır.

Parametre	Parametre tipi	Veri tipi	Açıklama
LENGTHM	IN	UInt 0	Uzunluk alanından sonra, uzunluk alanı değerinin içermediği karakter sayısını belirtin. Bu belirtim, yalnızca N + LEN + M bitiş koşuluyla kullanılır. 0 ila 255 Bayt
RCVTIME	IN	UInt 200	Alınan ilk karakter için ne kadar bekleneceğini belirtirler. Belirtilen süre içinde bir karakter başarıyla alınmadığı takdirde, alım yordamı bir hatayla sonlandırılır. Bu belirtim, yalnızca "Yanıt süresi" koşulunda kullanılır. (0 - 65535 Bit süresi, maksimum 8 saniye) Değerlendirme bir yanıtın ilk karakteri alındığında sona erdiğinden, bu parametre bir bitiş koşulu değildir. Bu yalnızca, bir yanıt beklendiği halde herhangi bir yanıt alınmadığı için alım yordamının sonlandırılması anlamında bir bitiş koşuludur. Burada ayrı bir bitiş koşulu tanımlamanız gerekir.
MSGTIME	IN	UInt 200	Tüm mesajın eksiksiz olarak alınması için, ilk karakteri aldıktan sonra ne kadar süre bekleneceğini belirtin. Bu parametre, sadece "Mesajın zaman aşımı" koşulu seçildiğinde kullanılır. (0 ila 65535 msn)
CHARGAP	IN	UInt 12	Karakterler arasındaki Bit sürelerinin sayısını belirtin. Karakterler arasındaki Bit süresi sayısı belirtilen değeri aşarsa, bitiş koşulu karşılanmış olur. Bu belirtim, yalnızca "Karakter aralığı" koşulunda kullanılır. (0 - 65535 Bit süresi, maksimum 8 saniye)
SEQ.CTL	IN	Byte B#16#0	Karakter dizisi 1, her karakter için kumandayı yok say/karşılaştır: Bunlar, bitiş karakteri dizisindeki her karakter için etkinleştirme Bit'leridir. Karakter 1, Bit 0'dır, karakter 2, Bit 1'dir, ..., karakter 5, Bit 4'tür. Bir Bit'in belirli bir karakter için devre dışı bırakılması, karakter dizisinin o pozisyonunda her bir karakterin bir eşleşme olduğu anlamına gelir.
SEQ.STR	IN	Char[5] 0	Karakter dizisi 1, başlangıç karakteri (5 karakter).

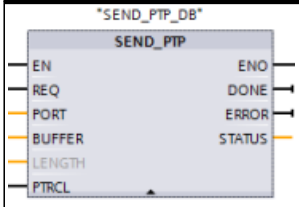
Tablo 13-109 Koşul kodları

STATUS (W#16#....)	Açıklama
80C0	Geçersiz başlangıç koşulu seçildi
80C1	Geçersiz bitiş koşulu seçildi, bitiş koşulu seçilmedi
80C2	Alım alarmı etkinleştirildi ve bu mümkün değil.
80C3	"Maksimum uzunluk" bitiş koşulu etkinleştirildi ve maksimum uzunluk 0 veya > 1024.
80C4	Hesaplanan uzunluk etkinleştirildi ve N, >= 1023'tür.
80C5	Hesaplanan uzunluk etkinleştirildi ve uzunluk 1, 2 veya 4 değil.
80C6	Hesaplanan uzunluk etkinleştirildi ve M değeri > 255'tir.
80C7	Hesaplanan uzunluk etkinleştirildi ve hesaplanan uzunluk > 1024'tür.
80C8	Yanıt zaman aşımı etkinleştirildi ve yanıt zaman aşımı sıfırdır.
80C9	Karakter aralığı için zaman aşımı etkinleştirildi ve zaman aşımı sıfırdır.

STATUS (W#16#....)	Açıklama
80CA	Hat boşta için zaman aşımı etkinleştirildi ve zaman aşımı sıfırdır.
80CB	Bitiş karakteri dizisi etkinleştirildi, ancak tüm karakterler "ilgili değil".
80CC	Başlangıç karakter dizisi (4'ten biri) etkinleştirildi, ancak tüm karakterler "ilgili değil".

13.6.1.4 SEND_PTP (gönderim tamponu aktarımı)

Tablo 13-110 SEND_PTP yordamı (noktadan noktaya verilerin gönderimi)

LAD/FBD	SCL	Açıklama
	**"SEND_PTP_DB"(REQ:=_bool_in_, PORT:=_uint_in_, BUFFER:=_variant_in_, LENGTH:=_uint_in_, PTRCL:=_bool_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	SEND_PTP veri aktarımını başlatır ve atanmış olan tamponu iletişim arayüzüne aktarır. CM veya CB atanmış olan Baud hızında verileri gönderirken, CPU'nun programı çalışmaya devam eder. Bir seferde yalnızca bir gönder yordamının beklemede olmasına izin verilir. CM veya CB bir mesaj gönderirken, ikinci bir SEND_PTP yordamı yürütülürse CM veya CB bir hata verir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-111 Parametreler için kullanılan veri tipleri

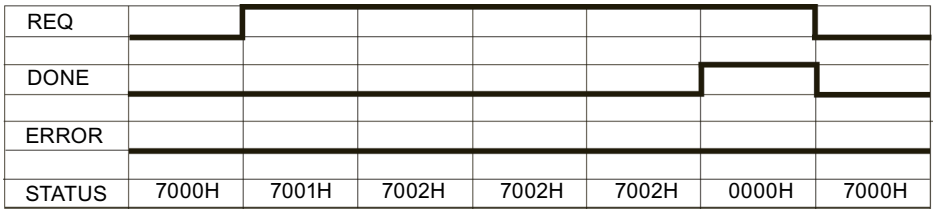
Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu serbest bırakma girişinin yükselen kenarında talep edilen aktarımın serbest bırakılması. Böylece tamponun içeriği noktadan noktaya iletişim arayüzüne aktarılır. (Standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
BUFFER	IN	Varyant	Bu parametre, gönderim tamponunun başlangıç adresini gösterir. (Standart değer: 0) **Not:** Boole verileri veya Boole alanları desteklenmez.
LENGTH [1]	IN	UInt	Bayt cinsinden iletilen çerçeve uzunluğu (standart değer: 0) Karmaşık bir yapı gönderiminde her zaman uzunluk 0'ı kullanın.
PTRCL	IN	Bool	Gelecekteki kullanım için rezerve edilmiştir
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir çevrim boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatalı tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

[1] Opsiyonel parametreler: LAD/FBD kutusunun altındaki oka tıklayarak bu parametreyi kapsayacak şekilde kutuyu genişletin.

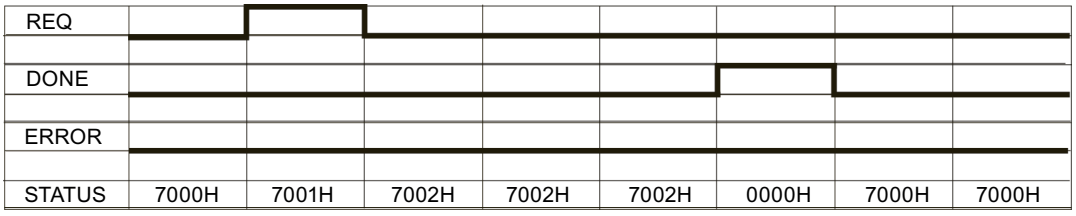
Bir gönder yordamı devam ederken, DONE ve ERROR çıkışları YANLIŞ durumundadır. Bir gönder yordamının sona ermesinden sonra, gönder yordamı durumunu bildirmek için DONE veya ERROR çıkışlarından biri DOĞRU olarak ayarlanır. DONE veya ERROR, DOĞRU durumdayken STATUS çıkışı geçerlidir.
İletişim arayüzü gönderilen verileri kabul ederse, bu yordam, 16#7001 durumunu yayınlar. Takip eden SEND_PTP yürütmeleri, CM veya CB hâlâ gönderiyorsa 16#7002 değerini geri döndürür. Gönder yordamının sona ermesinden sonra CM veya CB, gönder yordamı için

16#0000 durumunu yayınlar (herhangi bir hata oluşmadıysa). REQ = 0 içeren SEND_PTP takip eden yürütmeleri 16#7000 (serbest) durumunu geri döndürür.

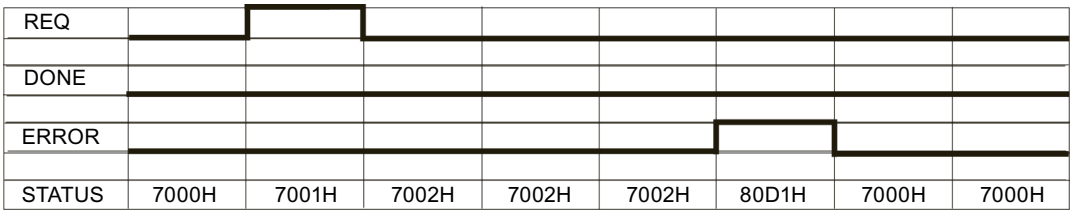
Aşağıdaki diyagramda, çıkış değerleri ile REQ arasındaki ilişki gösterilmektedir. Burada, gönderim işleminin durumunu kontrol etmek için yordamın düzenli olarak çağrıldığını varsayılmaktadır. Aşağıdaki şekilde ise, yordamın her çevrimde çağrıldığını varsayılmaktadır (DURUM değerleri ile temsil ediliyor).



Aşağıdaki şekilde, gönder yordamını başlatmak için REQ hattı darbeli (bir çevrim için) olduğunda DONE ve STATUS parametrelerinin nasıl yalnızca bir çevrim için geçerli olduğu gösterilmektedir.



Aşağıdaki şekilde, bir hata durumunda DONE, ERROR ve STATUS parametrelerinin ilişkisini gösterilmektedir.



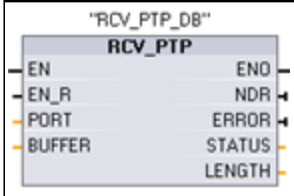
DONE, ERROR ve STATUS değerleri, yalnızca SEND_PTP'nin aynı örnek DB ile tekrar yürütülene kadar geçerli kalır.

Tablo 13-112 Koşul kodları

STATUS (W#16#....)	Açıklama
80D0	Aktif göndericide yeni talep
80D1	Bekleme süresi içinde CTS olmadığı için gönderim iptal edildi
80D2	DCE cihazından DSR gelmediği için gönderim iptal edildi
80D3	Bekleme kuyruğu taşması nedeniyle gönderim iptal edildi (1024 Bayt'tan fazla gönderim)
80D5	Ters ön gerilim sinyali (tel kopması)
833A	BUFFER parametresi için DB mevcut değil.

13.6.1.5 RCV_PTP (alım mesajlarını etkinleştirme)

Tablo 13-113 RCV_PTP yordamı (noktadan noktaya verilerin alımı)

LAD/FBD	SCL	Açıklama
	**"RCV_PTP_DB"(EN_R:=_bool_in_, PORT:=_uint_in_, BUFFER:=_variant_in_, NDR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, LENGTH=>_uint_out_);**	RCV_PTP, CM veya CB'de alınan mesajları denetler. Bir mesaj mevcutsa, CM veya CB'den CPU'ya aktarılır. Bir hata, uygun STATU değerini geri döndürür.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-114 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
EN_R	IN	Bool	Bu giriş DOĞRU ve bir mesaj mevcut ise, mesaj CM veya DB'den BUFFER'a aktarılır. EN_R = YANLIŞ ise, CM veya CB'deki mesajlar denetlenir ve NDR, ERROR ve STATUS çıkışları güncellenir, ancak mesaj BUFFER'a aktarılmaz. (Standart değer: 0)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
BUFFER	IN	Varyant	Bu parametre, alım tamponunun başlangıç adresini gösterir. Bu tampon, maksimum mesaj uzunluğunu alacak kadar büyük olmalıdır. Boole verileri veya Boole alanları desteklenmez. (Standart değer: 0)
NDR	OUT	Bool	Yeni veriler hazırsa ve yordam hatasız tamamlandıysa, bir yürütme boyunca DOĞRU.
ERROR	OUT	Bool	Yordam bir hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU.
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)
LENGTH	OUT	UInt	Çıkış mesajının Bayt cinsinden uzunluğu (standart değer: 0)

EN_R girişi ile RCV_PTP yordamının mesaj tamponu arasındaki şu ilişkiye dikkat edin:

EN_R girişi, alınan bir mesajın BUFFER'daki kopyasına kumanda eder.

EN_R girişi TRUE ve bir mesaj mevcut ise, CPU, mesajı CM veya CB'den BUFFER'a aktarır ve NDR, ERROR, STATUS, ve LENGTH çıkışlarını günceller.

EN_R, FALSE ise, CPU, CM veya CB'yi mesajlar açısından denetler ve NDR, ERROR, ve STATUS çıkışlarını günceller, ancak mesajı BUFFER'a aktarmaz EN_R'nin standart değerinin FALSE olduğunu unutmayın).

EN_R'nin TRUE olarak ayarlanması ve RCV_PTP yordamının yürütülmesinin EN girişi ile denetlemenizi öneririz.

NDR veya ERROR DOĞRU durumundayken, STATUS değeri geçerlidir. STATUS değeri, CM veya CB'deki alım işleminin sonlandırılmasının nedenini getirir. Bu, tipik olarak alım işleminin başarılı olduğunu ve alım işleminin normal şekilde sona erdiğini gösteren pozitif bir değerdir. STATUS değeri negatifse (onaltılık değerin en yüksek değerli Bit'i ayarlanmışsa), parite, çerçeveleme veya taşma hatası gibi bir hata durumu nedeniyle alım işlemi sonlandırılmıştır.

Her bir PtP iletişim arayüzü, maksimum 1024 Bayt'ı tampona alabilir. Bu, bir büyük mesaj veya birden çok sayıda küçük mesaj olabilir. CM veya CB'de birden çok mesaj mevcutsa, RCV_PTP

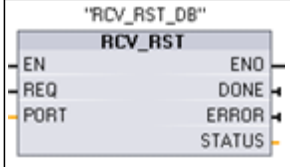
yordamı mevcut en eski mesajı yayınlar. Bunun ardından yürütülen bir RCV_PTP yordamı en eski ikinci mesajı yayınlar.

Tablo 13-115 Koşul kodları

STATUS (W#16#...)	Açıklama
0000	Tampon yok
0094	Maksimum karakter uzunluğu alındığı için, mesaj sonlandırıldı
0095	Mesaj zaman aşımı nedeniyle, mesaj sonlandırıldı
0096	Karakter aralığı zaman aşımı nedeniyle mesaj sonlandırıldı
0097	Yanıt zaman aşımı nedeniyle mesaj sonlandırıldı
0098	"N+LEN+M" uzunluk koşulu karşılandığı için, mesaj sonlandırıldı
0099	Bitiş karakteri dizisi karşılandığı için, mesaj sonlandırıldı
80E0	Alım tamponu dolu olduğundan, mesaj sonlandırıldı
80E1	Parite hatası nedeniyle, mesaj sonlandırıldı
80E2	Çerçeveleme hatası nedeniyle mesaj sonlandırıldı
80E3	Taşma hatası nedeniyle, mesaj sonlandırıldı
80E4	Hesaplanan uzunluk tampon boyutunu aştığından, mesaj sonlandırıldı
80E5	Ters ön gerilim sinyali (tel kopması)
833A	BUFFER parametresi için DB mevcut değil.

13.6.1.6 RCV_RST (alım tamponunun silinmesi)

Tablo 13-116 RCV_RST yordamı (alıcıyı sıfırla)

LAD/FBD	SCL	Açıklama
	**"RCV_RST_DB"(REQ:=_bool_in_, PORT:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	RCV_RST, CM veya CB'deki alım tamponunu siler.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-117 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu serbest bırakma girişinin yükselen kenarında alım tamponunun silinmesini etkinleştirir (standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)

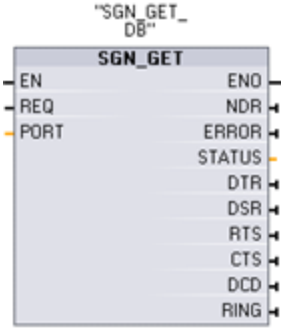
Parametre ve veri tipi		Veri tipi	Açıklama
DONE	OUT	Bool	Bir çevrim boyunca DOĞRU, son talebin hatasız yürütüldüğü anlamına gelir.
ERROR	OUT	Bool	DOĞRU, son talebin hatalarla yürütüldüğü anlamına gelir. Bu çıkış DOĞRU ise STATUS çıkışı ilgili hata kodlarını içerir.
STATUS	OUT	Word	Hata kodu (standart değer: 0) İletişim durum kodları için Noktadan noktaya yordamlar için ortak parametreler (Sayfa 908)'ye bakın.

NOT

Bir iletişim hatasından veya Baud hızı gibi bir iletişim parametresini değiştirdikten sonra mesaj tamponlarının silinmesini sağlamak için RCV_RST yordamını kullanabilirsiniz. RCV_RST'nin yürütülmesi, modülün tüm dahili mesaj tamponlarının silinmesine neden olur. Mesaj tamponlarını sildikten sonra, programınız bir sonraki alım yordamını yürüttüğünde, RCV_RST çağrılmadan önceki döneme ait eski mesajlar yerine yalnızca yeni mesajları geri döndüreceğinden emin olabilirsiniz.

13.6.1.7 SGN_GET (RS232 sinyallerinin sorgulanması)

Tablo 13-118 SGN_GET yordamı (RS232 sinyallerinin çağrılması)

LAD/FBD	SCL	Açıklama
	**"SGN_GET_DB"(REQ:=_bool_in_, PORT:=_uint_in_, NDR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, DTR=>_bool_out_, DSR=>_bool_out_, RTS=>_bool_out_, CTS=>_bool_out_, DCD=>_bool_out_, RING=>_bool_out_);**	SGN_GET, RS232 iletişim sinyallerinin güncel durumlarını okur. Bu fonksiyon, yalnızca RS232-CM için geçerlidir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-119 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişteki yükselen kenarda RS232 sinyal durumları çağrılır (standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır.
NDR	OUT	Bool	Yeni veriler hazırsa ve yordam hatasız tamamlandıysa, bir çevrim boyunca DOĞRU
ERROR	OUT	Bool	Yordam bir hatayla tamamlandıktan sonra bir çevrim boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)
DTR	OUT	Bool	Veri terminali hazır, modül hazır (çıkış). Standart değer: False

Parametre ve veri tipi		Veri tipi	Açıklama
DSR	OUT	Bool	Veri seti hazır, iletişim katılımcısı hazır (giriş). Standart değer: False
RTS	OUT	Bool	Gönderim talebi, modül gönderime hazır (çıkış). Standart değer: False
CTS	OUT	Bool	Gönderime hazır, iletişim katılımcısı verileri alabilir (giriş). Standart değer: False
DCD	OUT	Bool	Veri taşıyıcı algılanır, sinyal seviyesi alınır (her zaman False, desteklenmez)
RING	OUT	Bool	Çağrı göstergesi, gelen çağrı mesajı (her zaman False, desteklenmez)

Tablo 13-120 Koşul kodları

STATUS (W#16#....)	Açıklama
80F0	CM veya CB, RS-485'tir ve herhangi bir sinyal mevcut değil

13.6.1.8 SGN_SET (RS-232 sinyallerinin ayarlanması)

Tablo 13-121 SGN_SET yordamı (RS232 sinyallerini ayarla)

LAD/FBD	SCL	Açıklama
	**"SGN_SET_DB"(REQ:=_bool_in_, PORT:=_uint_in_, SIGNAL:=_byte_in_, RTS:=_bool_in_, DTR:=_bool_in_, DSR:=_bool_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);**	SGN_SET, RS232 iletişim sinyallerinin durumlarını okur. Bu fonksiyon, yalnızca RS232-CM için geçerlidir.

[1] Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

Tablo 13-122 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	RS232 sinyallerini ayarlama yordamı, bu girişteki yükselen kenar ile başlatılır (standart değer: Yanlış)
PORT	IN	PORT	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır. (Standart değer: 0)
SIGNAL	IN	Byte	Sabit ayarlanacak sinyalleri belirtir: (birden çok sayıda izin verilir) Standart değer: 0 • 01H = RTS • 02H = DTR • 04H = DSR
RTS	IN	Bool	Gönderim talebi, modül ayarlanacak değeri göndermeye hazır (doğru veya yanlış), standart değer: Yanlış
DTR	IN	Bool	Veri terminali, modül ayarlanacak değeri göndermeye hazır (doğru veya yanlış). Standart değer: Yanlış
DSR	IN	Bool	Veri seti hazır (DCE arayüzleri için geçerlidir), kullanılmıyor.

Parametre ve veri tipi		Veri tipi	Açıklama
DONE	OUT	Bool	Son talep hatasız tamamlandıktan sonra bir yürütme boyunca DOĞRU
ERROR	OUT	Bool	Son talep hatayla tamamlandıktan sonra bir yürütme boyunca DOĞRU
STATUS	OUT	Word	Yürütme koşulu (standart değer: 0)

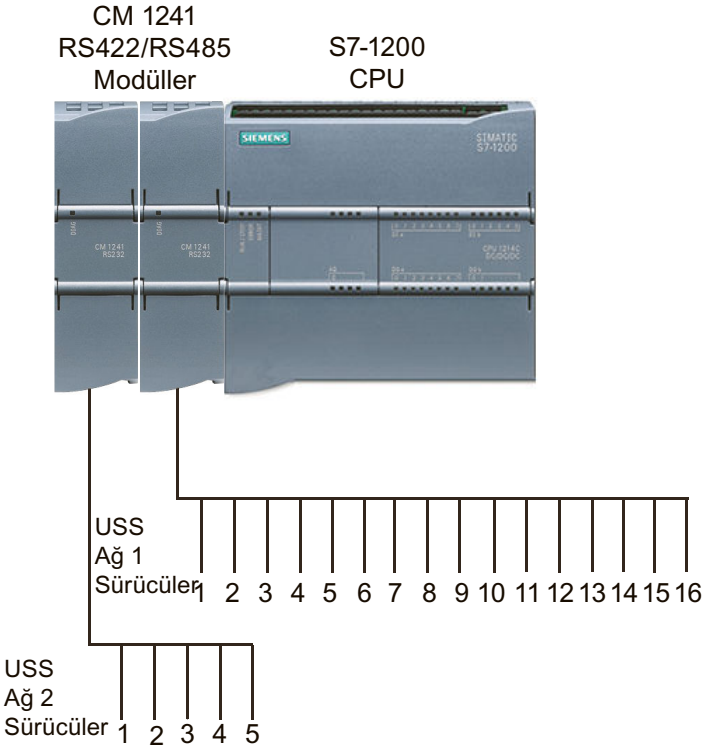
Tablo 13-123 Koşul kodları

STATUS (W#16#....)	Açıklama
80F0	CM veya CB, RS485'tir ve herhangi bir sinyal ayarlanamaz
80F1	Donanım akış kontrolü nedeniyle sinyaller ayarlanamıyor
80F2	Modül bir DTE cihazı olduğu için, DSR ayarlanamıyor
80F3	Modül bir DCE cihazı olduğu için, DTR ayarlanamıyor

13.7 Daha eski sistemlerdeki USS iletişim yordamları (yalnızca CM/CB 1241)

USS yordamları, üniversal seri arayüz (USS) protokolünü destekleyen motor sürücülerinin işleyişine kumanda eder. USS yordamları ile CM 1241 RS485 iletişim modüllerini veya bir CB 1241 RS485 iletişim kartını kullanarak, RS485 bağlantıları üzerinden birden çok sürücüyle iletişim kurabilirsiniz. Bir S7-1200 CPU'ya üç adede kadar CM 1241 RS422/RS485 modülü ve bir CB 1241 RS485 kartı takılabilir. Her bir RS485 portu, on altı adede kadar sürücüyü çalıştırabilir.

USS protokolü, bir seri veriyolu üzerinden iletişim kurmak için bir Master/Slave ağı kullanır. Master, seçili bir Slave'e mesaj göndermek için bir adres parametresi kullanır. Bir Slave'in kendisi, bir talep almadan önce asla gönderimde bulunmaz. Münferit Slave'ler arasında doğrudan mesaj aktarımı mümkün değildir. USS iletişimi yarı dupleks işletimde çalışır. Aşağıdaki USS görselinde, bir sürücünün örnek uygulamasına yönelik bir ağ diyagramı gösterilmektedir.



STEP 7, USS yordamlarının farklı sürümlerini sağlar. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

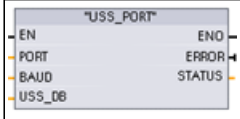
USS protokollerinin kullanım gereklilikleri

Dört USS yordamı, USS protokolünü desteklemek için 2 FB ve 2 FC kullanır. Her bir USS ağı için, USS_Port_Scan için bir örnek veri bloğu (DB) ve tüm USS_Drive_Control çağrıları için bir örnek veri bloğu kullanılır. STEP 7 bilgi sisteminde veya Siemens Industry Online Support'ta "USS protokolü kullanım gereklilikleri (https://support.industry.siemens.com/cs/ww/en/view/109798671/140076559243)" başlığı altında diğer gereklilikleri bulabilirsiniz.

13.7.1 Eski USS yordamları

13.7.1.1 USS_PORT yordamı (USS ağı üzerinden iletişim işleme)

Tablo 13-124 USS_PORT yordamı

LAD/FBD	SCL	Açıklama
	**USS_PORT(** **PORT:=_uint_in_,** **BAUD:=_dint_in_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **USS_DB:=_fbtref_inout_);**	USS_PORT yordamı, bir USS ağı üzerinden iletişimi yönetir.

Tablo 13-125 Parametreler için kullanılan veri tipleri

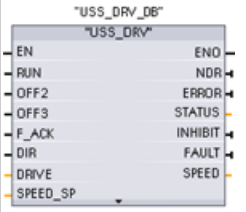
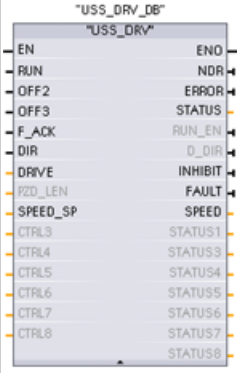
Parametre ve veri tipi		Veri tipi	Açıklama
PORT	IN	Port	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır.
BAUD	IN	DInt	USS iletişimi için Baud hızı.
USS_DB	INOUT	USS_BASE	Programınıza bir USS_DRV yordamı eklendiğinde, oluşturulan ve başlatılan örnek DB'nin adı.
ERROR	OUT	Bool	DOĞRU ise, bu çıkış bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir.
STATUS	OUT	Word	Talebin durum değeri, çevrimin veya başlatmanın sonucunu gösterir. "USS_Extended_Error" değişkenindeki bazı durum kodları için daha fazla bilgi mevcuttur.

Normalde, programdaki PtP iletişim portu başına yalnızca bir USS_PORT yordamı mevcuttur ve bu yordama yapılan her çağrı, tek bir sürücüye veya bir sürücüden yapılan bir aktarıma kumanda eder. Bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları, aynı örnek DB'yi kullanmalıdır.

Programınız USS_PORT yordamını, sürücüde zaman aşımı oluşturmayacak sıklıkta yürütmelidir. Sürücü zaman aşımlarını önlemek ve USS_DRV çağrıları için en son USS veri güncellemelerine sahip olmak için, USS_PORT yordamı genellikle bir uyandırma alarmı OB'sinden çağrılır.

13.7.1.2 USS_DRV yordamı (sürücü ile veri alışverişi)

Tablo 13-126 USS_DRV yordamı

LAD/FBD	SCL	Açıklama
Standart görünüm  Genişletilmiş görünüm  	**"USS_DRV_DB"(** **RUN:=_bool_in_,** **OFF2:=_bool_in_,** **OFF3:=_bool_in_,** **F_ACK:=_bool_in_,** **DIR:=_bool_in_,** **DRIVE:=_usint_in_,** **PZD_LEN:=_usint_in_,** **SPEED_SP:=_real_in_,** **CTRL3:=_word_in_,** **CTRL4:=_word_in_,** **CTRL5:=_word_in_,** **CTRL6:=_word_in_,** **CTRL7:=_word_in_,** **CTRL8:=_word_in_,** **NDR=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **RUN_EN=>_bool_out_,** **D_DIR=>_bool_out_,** **INHIBIT=>_bool_out_,** **FAULT=>_bool_out_,** **SPEED=>_real_out_,** **STATUS1=>_word_out_,** **STATUS3=>_word_out_,** **STATUS4=>_word_out_,** **STATUS5=>_word_out_,** **STATUS6=>_word_out_,** **STATUS7=>_word_out_,** **STATUS8=>_word_out_);**	USS_DRV yordamı, talep mesajları üreterek ve sürücünün yanıt mesajlarını değerlendirerek bir sürücü ile veri alışverişinde bulunur. Her bir sürücü için ayrı bir fonksiyon bloğu kullanılmalıdır, ancak bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları aynı örnek veri bloğunu kullanmalıdır. İlk USS_DRV yordamını eklediğinizde, DB adını girmelisiniz. Bunun ardından, ilk yordamı eklediğinizde oluşturulan bu DB'ye bakın. Yordamı eklerseniz, STEP 7, DB'yi otomatik olarak oluşturur.

[1] LAD ve FBD: Tüm parametreleri görmek için kutuyu genişletin. Bunun için kutunun alt kısmına tıklayın. Gri olarak görüntülenen parametre bağlantıları opsiyoneldir, parametrelendirme gerekmez.

Tablo 13-127 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
RUN	IN	Bool	Sürücünün başlatma Bit'i: Bu parametre DOĞRU ise, bu giriş, sürücünün önceden ayarlanmış devir sayısında çalışmasını sağlar. Sürücü, RUN işletim durumundayken YANLIŞ olarak değişirse, motor hareketsiz durum aşamasına kadar yavaşlar. Bu davranış, gerilim beslemesini kapatmaktan (OFF2) ve motoru frenlemekten (OFF3) farklıdır.
OFF2	IN	Bool	"Hareketsiz duruma yavaşla" Bit'i: Eğer bu parametre YANLIŞ ise, bu Bit frenlemeksizin sürücünün yavaşlayarak durma aşamasına gelmesine neden olur.
OFF3	IN	Bool	Hızlı Stop Bit'i: Bu parametre YANLIŞ ise, bu Bit, sürücüyü frenleyerek hızlı durmaya neden olur.
F_ACK	IN	Bool	Hata onaylama Bit'i: Bu Bit, bir sürücünün hata Bit'ini sıfırlamak için kullanılır. Bu Bit, hata silindikten sonra ayarlanır ve böylece sürücü önceki hatanın artık bildirilmesine gerek olmadığını anlar.

Parametre ve veri tipi		Veri tipi	Açıklama
DIR	IN	Bool	Sürücü yönünün kumandası: Bu Bit, sürücü ileri yönde çalışacağı zaman ayarlanır (SPEED_SP pozitif olduğunda).
DRIVE	IN	USInt	Sürücünün adresi: Bu giriş, USS sürücüsünün adresidir. Geçerli aralık, sürücü 1 ile sürücü 16 arasındadır.
PZD_LEN	IN	USInt	Sözcük uzunluğu: Bu, Proses Durumu Değerleri (PZD) veri sözcüklerinin sayısıdır. Geçerli değerler 2, 4, 6 veya 8 sözcüktür. Standart değer, 2'dir.
SPEED_SP	IN	Real	Devir sayısı hedef değeri: Bu, yapılandırılan frekansın yüzdesi cinsinden sürücünün devir sayısıdır. Bir pozitif değer, sürücünün ileriye doğru çalıştığı anlamına gelir (DIR doğru ise). Geçerli aralık, 200,00 ile -200,00 arasındadır.
CTRL3	IN	Word	Kumanda sözcüğü 3: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL4	IN	Word	Kumanda sözcüğü 4: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL5	IN	Word	Kumanda sözcüğü 5: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL6	IN	Word	Kumanda sözcüğü 6: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL7	IN	Word	Kumanda sözcüğü 7: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
CTRL8	IN	Word	Kumanda sözcüğü 8: Sürücünün kullanıcı yapılandırmalı bir parametresine yazılan değer. Bunu sürücüde yapılandırmanız gerekir (opsiyonel parametre).
NDR	OUT	Bool	Yeni veriler hazır: Bu parametre DOĞRU ise, Bit, yeni bir iletişim talebinden gelen verilerin çıkışta mevcut olduğunu bildirir.
ERROR	OUT	Bool	Hata oluştu: DOĞRU ise, bu bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir. Bir hata durumunda diğer tüm çıkışlar sıfıra ayarlanır. İletişim hataları, yalnızca USS_PORT yordamının ERROR ve STATUS çıkışlarında bildirilir.
STATUS	OUT	Word	Talebin durum değeri, çevrimin sonucunu gösterir. Bu, sürücü tarafından yayınlanan bir durum sözcüğü değildir.
RUN_EN	OUT	Bool	İşletim serbest bırakılır: Bu Bit, sürücünün çalışıp çalışmadığını bildirir.
D_DIR	OUT	Bool	Sürücü yönü: Bu Bit, sürücünün ileriye doğru çalışıp çalışmadığını bildirir.
INHIBIT	OUT	Bool	Sürücü bloke edilmiş: Bu Bit, sürücü blokaj Bit'inin durumunu bildirir.
FAULT	OUT	Bool	Sürücü hatası: Bu Bit, sürücüde bir hata oluştuğunu bildirir. Bu Bit'i silmek için arızayı gidermeli ve F_ACK Bit'ini ayarlamalısınız.
SPEED	OUT	Real	Sürücü devir sayısının gerçek değeri (sürücü durum sözcüğü 2'nin ölçeklendirilmiş değeri): Bu, yapılandırılan devir sayısının yüzdesi cinsinden sürücünün devir sayısıdır.
STATUS1	OUT	Word	Sürücünün durum sözcüğü 1: Bu değer, bir sürücünün sabit durum Bit'lerini içerir.

Parametre ve veri tipi		Veri tipi	Açıklama
STATUS3	OUT	Word	Sürücünün durum sözcüğü 3: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS4	OUT	Word	Sürücünün durum sözcüğü 4: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS5	OUT	Word	Sürücünün durum sözcüğü 5: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS6	OUT	Word	Sürücünün durum sözcüğü 6: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS7	OUT	Word	Sürücünün durum sözcüğü 7: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.
STATUS8	OUT	Word	Sürücünün durum sözcüğü 8: Bu değer, sürücünün kullanıcı tanımlı bir durum sözcüğünü içerir.

USS_DRV ilk kez yürütüldüğünde, USS adresi (DRIVE parametresi) tarafından belirtilen sürücü, örnek DB'de başlatılır. Bu başlatmanın ardından, takip eden USS_PORT yordamları bu sürücü numarası üzerinden o sürücü ile iletişime başlayabilir.

Sürücü numarasını değiştirirseniz, örnek DB'nin başlatılması için CPU'nun önce STOP ve ardından tekrar RUN işletim durumuna alınması gerekir. Giriş parametreleri USS gönderim tamponunda yapılandırılır ve varsa çıkışlar "önceki" geçerli bir yanıt tamponundan okunur. USS_DRV yordamının yürütülmesi sırasında herhangi bir veri aktarımı gerçekleşmez. USS_PORT'in yürütülmesinden sonra sürücüler iletişim kurar. USS_DRV yalnızca gönderilecek mesajları yapılandırır ve önceki bir talep ile alınmış olabilecek verileri değerlendirir.

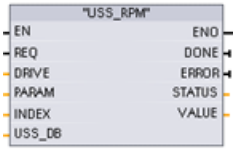
Sürücünün dönüş yönüne, DIR (Bool) girişi veya SPEED_SP girişindeki (Real) ön işaret (pozitif veya negatif) üzerinden kumanda edebilirsiniz. Aşağıdaki tabloda, motorun ileri doğru çalıştığını varsayarak sürücünün dönüş yönünü belirlemek için bu girişlerin birlikte çalışması açıklanmaktadır.

Tablo 13-128 SPEED_SP ve DIR parametreleri arasındaki etkileşim

SPEED_SP	DIR	Sürücünün dönüş yönü:
Değer > 0	0	Geriye
Değer > 0	1	İleri doğru
Değer < 0	0	İleri doğru
Değer < 0	1	Geriye

13.7.1.3 USS_Read_Param yordamı (sürücüden parametre okunması)

Tablo 13-129 USS_RPM yordamı

LAD/FBD	SCL	Açıklama
	**USS_RPM(REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, VALUE=>_variant_out_, USS_DB:=_fbtref_inout_);**	USS_RPM yordamı, bir sürücüden bir parametre okur. Bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları aynı veri bloğunu kullanmalıdır. USS_RPM, ana programdaki bir çevrim OB'sinden çağrılmalıdır.

Tablo 13-130 Parametreler için kullanılan veri tipleri

Parametre tipi		Veri tipi	Açıklama
REQ	IN	Bool	Gönderim talebi: REQ DOĞRU ise, yeni bir okuma talebi gerekli olur. Bu parametre için talep halen beklemedeyse, bu işlem yok sayılır.
DRIVE	IN	USInt	Sürücünün adresi: DRIVE, USS sürücünün adresidir. Geçerli aralık, sürücü 1 ile sürücü 16 arasındadır.
PARAM	IN	UInt	Parametre numarası: PARAM, hangi sürücü parametresine yazılacağını belirtir. Bu parametre için aralık, 0 ile 2047 arasındadır. Bazı sürücülerde en yüksek değerli Bayt, 2047'den büyük PARAM değerlerine erişebilir. Genişletilmiş bir aralığa erişim hakkında daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
INDEX	IN	UInt	Parametre indeksi: INDEX, hangi sürücü parametresi indeksine yazılacağını belirtir. Bu, en düşük Bayt'ın gerçek indeks değeri olduğu 16 Bit'lik bir değerdir ve (0 ila 255) aralığındadır. En yüksek değerli Bayt sürücü tarafından da kullanılabilir ve sürücüye özgüdür. Daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
USS_DB	INOUT	USS_BASE	Programınıza bir USS_DRV yordamı eklendiğinde, oluşturulan ve başlatılan örnek DB'nin adı.
VALUE	IN	Word, Int, UInt, DWord, DInt, UDInt, Real	Bu, okunan parametrenin değeridir ve yalnızca DONE Bit'i DOĞRU ise geçerlidir.
DONE[1]	OUT	Bool	Bu parametre DOĞRU ise, VALUE çıkışında daha önce talep edilen okuma parametresinin değeri bulunur. Bu Bit, USS_DRV yordamı sürücüden okuma yanıtı algıladığında ayarlanır. Bu Bit, şu durumlarda sıfırlanır: Yanıt verilerini başka bir USS_RPM sorgusu aracılığıyla veya bir sonraki iki USS_DRV çağrısının ikincisinde talep edersiniz.
ERROR	OUT	Bool	Hata oluştu: DOĞRU ise, ERROR bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir. Bir hata durumunda diğer tüm çıkışlar sıfıra ayarlanır. İletişim hataları yalnızca USS_PORT yordamının ERROR ve STATUS çıkışlarında bildirilir.
STATUS	OUT	Word	STATUS, okuma talebinin sonucunu belirtir. "USS_Extended_Error" değişkenindeki bazı durum kodları için daha fazla bilgi mevcuttur.

[1] DONE Bit'i, referans verilen motor sürücüsünden geçerli verilerin okunduğunu ve CPU'ya iletildiğini gösterir. USS kütüphanesinin başka bir parametreyi hemen okuyabildiğini belirtmez. Parametre kanalı ilgili sürücü tarafından kullanılmak üzere serbest bırakılmadan önce, motor sürücüsüne boş bir parametre karakteristik değeri (PKD) talebi gönderilmeli ve ayrıca yordam tarafından onaylanmalıdır. Belirli motor sürücüsü için hemen bir USS_RPM veya USS_WPM FC'nin çağrılması, 0x818A hatasıyla sonuçlanır.

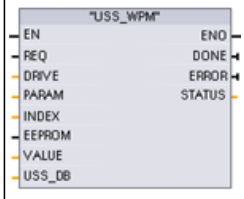
13.7.1.4 USS_WPM yordamı (sürücüdeki parametreyi değiştirme)

NOT

EEPROM yazma yordamları (bir USS sürücüsündeki EEPROM için)

EEPROM yazma işleminin kullanımında aşırıya kaçmayın. EEPROM'un kullanım ömrünü uzatmak için, EEPROM yazma işlemlerinin sayısını mümkün olduğu kadar düşük tutun.

Tablo 13-131 USS_WPM yordamı

LAD/FBD	SCL	Açıklama
	**USS_WPM(REQ:=_bool_in_,** **DRIVE:=_usint_in_,** **PARAM:=_uint_in_,** **INDEX:=_uint_in_,** **EEPROM:=_bool_in_,** **VALUE:=_variant_in_,** **DONE=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **USS_DB:=_fbtref_inout_);**	USS_WPM yordamı, sürücüdeki bir parametreyi değiştirir. Bir USS ağına ve bir PtP iletişim portuna atanan tüm USS fonksiyonları aynı veri bloğunu kullanmalıdır. USS_WPM, bir ana programın çevrim OB'sinden çağrılmalıdır.

Tablo 13-132 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Gönderim talebi: REQ DOĞRU ise, yeni bir yazma talebi gerekli olur. Bu parametre için talep halen beklemedeyse, bu işlem yok sayılır.
DRIVE	IN	USInt	Sürücünün adresi: DRIVE, USS sürücünün adresidir. Geçerli aralık, sürücü 1 ile sürücü 16 arasındadır.
PARAM	IN	UInt	Parametre numarası: PARAM, hangi sürücü parametresine yazılacağını belirtir. Bu parametre için aralık, 0 ile 2047 arasındadır. Bazı sürücülerde en yüksek değerli Bayt, 2047'den büyük PARAM değerlerine erişebilir. Genişletilmiş bir aralığa erişim hakkında daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
INDEX	IN	UInt	Parametre indeksi: INDEX, hangi sürücü parametresi indeksine yazılacağını belirtir. Bu, en düşük Bayt'ın gerçek indeks değeri olduğu 16 Bit'lik bir değerdir ve (0 ila 255) aralığındadır. En yüksek değerli Bayt sürücü tarafından da kullanılabilir ve sürücüye özgüdür. Daha fazla bilgiyi, sürücünüzün işletme kitabında bulabilirsiniz.
EEPROM	IN	Bool	Sürücünün EEPROM'unda saklama: DOĞRU ise, sürücü parametresine yazma işlemi, sürücü EEPROM'unda saklanır. YANLIŞ ise, yazılan değer yalnızca geçici olarak saklanır ve sürücüye bir sonraki güç açılmasında silinir.
VALUE	IN	Word, Int, UInt, DWord, DInt, UDInt, Real	Yazılacak parametrenin değeri. REQ'nun durum değişiminde geçerli olmalıdır.
USS_DB	INOUT	USS_BASE	Programınıza bir USS_DRV yordamı eklendiğinde, oluşturulan ve başlatılan örnek DB'nin adı.

[1] DONE Bit'i, referans verilen motor sürücüsünden geçerli verilerin okunduğunu ve CPU'ya iletildiğini gösterir. USS kütüphanesinin başka bir parametreyi hemen okuyabildiğini belirtmez. Parametre kanalı ilgili sürücü tarafından kullanılmak üzere serbest bırakılmadan önce, motor sürücüsüne boş bir parametre karakteristik değeri (PKD) talebi gönderilmeli ve ayrıca yordam tarafından onaylanmalıdır. Belirli motor sürücüsü için hemen bir USS_RPM veya USS_WPM FC'nin çağrılması, 0x818A hatasıyla sonuçlanır.

Parametre ve veri tipi		Veri tipi	Açıklama
DONE[1]	OUT	Bool	DOĞRU ise, DONE, VALUE girişinin sürücüye yazıldığını gösterir. Bu Bit, USS_DRV yordamının sürücüden yazma yanıtı algıladığında ayarlanır. Bu Bit, yanıt verilerini başka bir USS_WPM sorgusu aracılığıyla talep ettiğinizde veya sonraki iki USS_DRV çağrısının ikincisinde sıfırlanır.
ERROR	OUT	Bool	DOĞRU ise, ERROR bir hatanın oluştuğunu ve STATUS çıkışının geçerli olduğunu gösterir. Bir hata durumunda diğer tüm çıkışlar sıfıra ayarlanır. İletişim hataları yalnızca USS_PORT yordamının ERROR ve STATUS çıkışlarında bildirilir.
STATUS	OUT	Word	STATUS, yazma talebinin sonucunu belirtir. "USS_Extended_Error" değişkenindeki bazı durum kodları için daha fazla bilgi mevcuttur.

[1] DONE Bit'i, referans verilen motor sürücüsünden geçerli verilerin okunduğunu ve CPU'ya iletildiğini gösterir. USS kütüphanesinin başka bir parametreyi hemen okuyabildiğini belirtmez. Parametre kanalı ilgili sürücü tarafından kullanılmak üzere serbest bırakılmadan önce, motor sürücüsüne boş bir parametre karakteristik değeri (PKD) talebi gönderilmeli ve ayrıca yordam tarafından onaylanmalıdır. Belirli motor sürücüsü için hemen bir USS_RPM veya USS_WPM FC'nin çağrılması, 0x818A hatasıyla sonuçlanır.

13.7.2 Eski USS durum kodları

USS yordamının durum kodları, USS fonksiyonlarının STATUS çıkışında yayınlanır.

Tablo 13-133 STATUS kodları [1]

STATUS (W#16#....)	Açıklama
0000	Hata yok
8180	Sürücüden gelen yanıtın uzunluğu, sürücüden alınan karakterler ile aynı değil. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
8181	VALUE parametresi Word, Real veya Double Word veri tipine sahip değil.
8182	Kullanıcı, sözcük tipinde bir parametre değeri girdi ve sürücüden çift sözcük veya gerçek formatta yanıt aldı.
8183	Kullanıcı, bir çift sözcük veya gerçek tipinde parametre değeri girdi ve sürücüden sözcük formatında yanıt aldı.
8184	Sürücüden gelen yanıt telegramında yanlış bir sağlama toplamı vardı. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
8185	Geçersiz sürücü adresi (sürücüler için geçerli adres aralığı: 1 ila 16)
8186	Devir sayısı hedef değeri, referansı aralığın dışında (geçerli hız hedef değer aralığı: -%200 ila %200).
8187	Gönderilen talebe, yanlış sürücü numarası yanıt verdi. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
8188	Geçersiz PZD sözcük uzunluğu belirtildi (geçerli aralık = 2, 4, 6 veya 8 sözcük)
8189	Geçersiz Baud hızı belirtildi.
818A	Parametre talep kanalı, bu sürücü için olan başka bir talep tarafından kullanılıyor.
818B	Sürücü, taleplere ve yeniden denemelere yanıt vermedi. Hatanın ortaya çıktığı sürücünün numarası "USS_Extended_Error" değişkeninde yayınlanır. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.

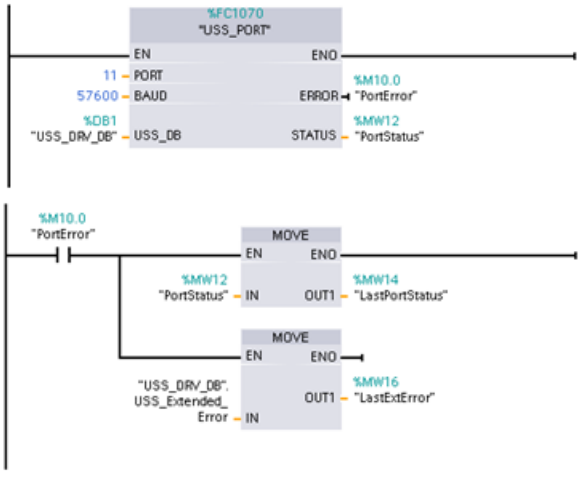
STATUS (W#16#....)	Açıklama
818C	Sürücü, bir parametre talebi için genişletilmiş bir hata yayınladı. Genişletilmiş hataların açıklamasını, bu tablonun altında bulabilirsiniz.
818D	Sürücü, bir parametre talebi için bir "Geçersiz erişim" hatası yayınladı. Parametre erişiminin neden sınırlandırılmış olabileceği hakkında daha fazla bilgi için, sürücü işletme kitabına bakın.
818E	Sürücü başlatılmadı. USS_DRV yordamı bu sürücü için en azından bir kez çağrılmadıysa, bu hata kodu USS_RPM veya USS_WPM'ye yayınlanır. Bu işlem, USS_DRV'nin ilk çevrimindeki başlatmanın, bekleyen bir parametre okuma veya yazma talebinin üzerine yazmasını engeller, çünkü sürücü daha sonra yeni bir giriş olarak başlatılır. Bu hatayı düzeltmek için, bu sürücü için USS_DRV yordamını çağırın.
80Ax-80Fx	USS kütüphanesi tarafından PtP iletişimi için çağrılan FB'ler tarafından geri döndürülen spesifik hatalar - Bu hata kodları, USS kütüphanesi tarafından değiştirilmez ve PtP yordamının açıklamalarında tanımlanır.

[1] Yukarıda sıralanan USS yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

Çeşitli STATUS kodları için daha fazla bilgi, USS_DRV örnek DB'sinin "USS_Extended_Error" değişkeninde sağlanmaktadır. STATUS kodları onaltılık 8180, 8184, 8187 ve 818B için, USS_Extended_Error iletişim hatasının ortaya çıktığı sürücünün sürücü numarasını içerir. USS_RPM veya USS_WPM yordamı kullanıldığında, STATUS kodu onaltılık 818C için USS_Extended_Error sürücü tarafından yayınlanan bir sürücü hata kodu içerir.

Örnek: İletişim hataları mesajı

İletişim hataları (STATUS = 16#818B) yalnızca USS_PORT yordamıyla bildirilir, USS_DRV yordamıyla bildirilmez. Örnek: Ağ düzgün bir şekilde sonlandırılmamış, bir sürücü, RUN işletim durumuna geçebiliyor, ancak USS_DRV yordamı tüm çıkış parametrelerinde 0 gösteriyor. Bu durumda, iletişim hatasını yalnızca USS_PORT yordamı üzerinden saptayabilirsiniz. Bu hatanın yalnızca bir çevrim boyunca görülebilmesi nedeniyle burada ilgili edinim mantığını girmelisiniz. Bu, aşağıdaki örnekte açıklanmaktadır. Bu örnekte USS_PORT yordamının hata Bit'i DOĞRU ise, markörlerin bellek alanına STATUS ve USS_Extended_Error değerleri konur. STATUS kodunun değeri onaltılık 8180, 8184, 8187 veya 818B ise, sürücü numarası, USS_Extended_Error değişkenine konur.



Ağ 1 Port durumu "PortStatus" ve genişletilmiş hata kodu değerleri "USS_DRV_DB".USS_Extended_Error yalnızca bir program çevrimi boyunca geçerli kalır. Bu değerler daha sonraki işleme için toplanır.

Ağ 2 "PortError" kontağı, "PortStatus" değerinin "LastPortStatus" içinde ve "USS_DRV_DB".USS_Extended_Error değerinin ise "LastExtError" içinde saklanmasını tetikler.

Bir sürücünün dahili parametrelerine okuma ve yazma erişimi

USS sürücüleri, bir sürücünün dahili parametrelerine okuma ve yazma erişimlerini destekler. Bu fonksiyon, sürücüye desantral olarak kumanda edilmesine ve konfigürasyonuna olanak tanır. Sürücü parametrelerine erişimler, örneğin aralık dışında değer veya sürücünün güncel işletim türünde izin verilmeyen talepler gibi hatalardan dolayı başarısız olabilir. Sürücü, "USS_Extended_Error" değişkeninde yayınlanan bir hata kodu üretir. Bu hata kodu yalnızca USS_RPM veya USS_WPM yordamının son yürütülmesi için geçerlidir. STATUS code değeri onaltılık 818C ise, sürücünün hata kodu "USS_Extended_Error" değişkeninde saklanır. "USS_Extended_Error"un hata kodunun değeri, sürücü varyantına göre değişir. Parametreler için okuma ve yazma fonksiyonlarının genişletilmiş hata kodlarına yönelik açıklama için sürücünün işletme kitabına başvurun.

13.7.3 Sürücü ünitesi için olan eski USS yordamı ile ilgili genel gereklilikler

Sürücü ünitesi için olan eski USS yordamı ile ilgili genel gereklilikler

- Sürücüler için, 4 parametre karakteristik değer sözcüğünün kullanımı ayarlanmalıdır.
- Sürücüler; 2, 4, 6 veya 8 PZD sözcüğü için yapılandırılabilir.
- Sürücüdeki PZD sözcüklerinin sayısı, sürücünün USS_DRV yordamının PZD_LEN girişine karşılık gelmelidir.
- Tüm sürücülerin Baud hızı, USS_PORT yordamının giriş BAUD'una karşılık gelmelidir.
- Sürücü, uzaktan kumanda için ayarlanmalıdır.
- Sürücü COM bağlantısındaki nominal frekans değeri, USS olarak ayarlanmalıdır.
- Sürücü adresi için, 1 ila 16 belirlenmelidir. Bu adres, sürücünün USS_DRV bloğundaki DRIVE girişine karşılık gelmelidir.
- Sürücü referansının polaritesini kullanmak için sürücünün yön kumandası ayarlanmalıdır.
- RS485 ağı, usulüne uygun tamamlanmış olmalıdır.

Genel USS sürücü bağlantısı ve kurulumu USS talimatları (V4.1) ve eski USS yordamları (V4.0'a kadar) için aynıdır. Daha fazla bilgi için bkz. Örnek: Genel USS sürücü bağlantısı ve kurulumu (Sayfa 952).

13.8 Daha eski sistemlerdeki Modbus TCP iletişimi

13.8.1 Genel bakış

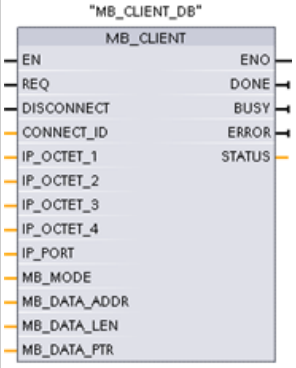
STEP 7 V13 SP1 ve S7-1200 V4.1 CPU'lardan önce, Modbus TCP iletişim yordamları farklı adlara ve bazen de farklı arayüzlere sahipti. Her iki yordam türü için de genel konseptler geçerlidir. Programlamayla ilgili bilgileri münferit eski Modbus TCP yordamlarıyla ilgili açıklamalarda bulabilirsiniz.

STEP 7, Modbus TCP için yordamların çeşitli sürümlerini sunmaktadır. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

13.8.2 Eski Modbus TCP yordamları

13.8.2.1 MB_CLIENT yordamları (PROFINET üzerinden Modbus TCP istemcisi olarak iletişim)

Tablo 13-134 MB_CLIENT yordamı

LAD/FBD	SCL	Açıklama
	**"MB_CLIENT_DB"(** **REQ:=_bool_in_,** **DISCONNECT:=_bool_in_,** **CONNECT_ID=_uint_in_,** **IP_OCTET_1:=_byte_in_,** **IP_OCTET_2:=_byte_in_,** **IP_OCTET_3:=_byte_in_,** **IP_OCTET_4:=_byte_in_,** **IP_PORT:=_uint_in_,** **MB_MODE:=_usint_in_,** **MB_DATA_ADDR:=_udint_in_,** **MB_DATA_LEN:=_uint_in_,** **DONE=>_bool_out_,** **BUSY=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **MB_DATA_PTR:=_variant_inout_);**	MB_CLIENT, Modbus TCP istemcisi olarak S7-1200 CPU üzerindeki PROFINET portu aracılığıyla iletişim kurar. İletişim için ek bir donanım modülüne gerek yoktur. MB_CLIENT; bir istemci-sunucu bağlantısı kurabilir, bir Modbus fonksiyon talebi gönderebilir, bir yanıt alabilir ve bir Modbus TCP sunucusundan bağlantıyı kesmeye kumanda edebilir.

Tablo 13-135 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	In	Bool	YANLIŞ = Modbus iletişim talebi yok DOĞRU = Modbus TCP sunucusuyla iletişim talebi
DISCONNECT	IN	Bool	Programınız, bir Modbus sunucu cihazına bağlantının kurulmasına ve sonlandırılmasına kumanda etmek için DISCONNECT parametresini kullanabilir. DISCONNECT = 0 ise ve bağlantı mevcut değilse, MB_CLIENT atanan IP adresi ve port numarası ile bağlantı kurmaya çabalar. DISCONNECT = 1 ise ve bir bağlantı mevcutsa, bağlantıyı kesmeyi dener. Bu girişin etkinleştirildiği her zaman başka hiçbir işlem başlatılmaz.
CONNECT_ID	IN	UInt	CONNECT_ID parametresi, PLC cihazındaki her bağlantıyı benzersiz olarak tanımlar. MB_CLIENT veya MB_SERVER yordamının her benzersiz örneği, benzersiz bir CONNECT_ID parametresi içermelidir.
IP_OCTET_1	IN	USInt	Modbus TCP sunucusunun IP adresi: Sekizli 1 İstemcinin Modbus TCP protokolü üzerinden bağlandığı ve iletişim kurduğu Modbus TCP sunucusunun 32 Bit'lik IP adresinin (IPv4) 8 Bit'lik kısmı.
IP_OCTET_2	IN	USInt	Modbus TCP sunucusunun IP adresi: Sekizli 2
IP_OCTET_3	IN	USInt	Modbus TCP sunucusunun IP adresi: Sekizli 3
IP_OCTET_4	IN	USInt	Modbus TCP sunucusunun IP adresi: Sekizli 4
IP_PORT	IN	UInt	Standart değer = 502: İstemcinin bağlanmaya çalıştığı ve daha sonra TCP/IP protokolü üzerinden iletişim kurduğu sunucunun IP port numarası.
MB_MODE	IN	USInt	Seçim modu: Talep türünü atar (okuma, yazma veya teşhis). Daha fazla bilgiyi, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
MB_DATA_ADDR	IN	UDInt	Modül başlangıç adresi: MB_CLIENT tarafından erişilecek verinin başlangıç adresini atar. Geçerli adresleri, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.

Parametre ve veri tipi		Veri tipi	Açıklama
MB_DATA_LEN	IN	UInt	Modbus veri uzunluğu: Bu talep tarafından erişilecek Bit veya sözcük sayısını atar. Geçerli uzunlukları, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
MB_DATA_PTR	IN_OUT	Variant	İşaretçi, Modbus veri kaydı üzerinde: Kayıt, bir Modbus sunucusuna gönderilen veya bir Modbus sunucusundan alınan verileri tampona alır. İşaretçi, optimize edilmemiş bir global DB veya markörlerin bellek alanında bir adres atamalıdır.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - İşlenen MB_CLIENT yordamı yok • 1 - MB_CLIENT yordamı işleniyor
ERROR	OUT	Bool	ERROR Bit'i, MB_CLIENT yürütmesi hatalı bir şekilde sona erdikten sonra bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

REQ parametresi

YANLIŞ = Modbus iletişim talebi yok
DOĞRU = Modbus TCP sunucusuyla iletişim talebi

Hiçbir MB_CLIENT örneği aktif değilse ve DISCONNECT = 0 parametresi REQ = 1 ise, yeni bir Modbus talebi başlatılır. Bağlantı henüz oluşturulmamışsa, yeni bir bağlantı kurulacaktır.

Mevcut talep tamamlanmadan önce aynı MB_CLIENT örneği DISCONNECT = 0 ve REQ = 1 ile tekrar çalıştırılırsa, bir sonraki Modbus aktarımı yürütülmeyecektir. Ancak mevcut talep tamamlandıktan sonra, MB_CLIENT'in REQ=1 ile yürütülmesi koşuluyla yeni bir talep işlenebilir.

Mevcut MB_CLIENT iletişim talebi tamamlandığında, DONE Bit'i bir çevrim boyunca DOĞRU'dur. DONE Bit'i, birden çok MB_CLIENT talebini sıralamak için bir zaman aralığı olarak kullanılabilir.

NOT

MB_CLIENT 'in işlenmesi sırasında giriş verilerinin tutarlılığı

Bir Modbus istemcisi bir Modbus işlemi başlattıktan sonra, tüm giriş durumları dahili olarak saklanır ve bir sonraki her çağrıda karşılaştırılır. Karşılaştırmanın amacı, belirli bir çağrının aktif istemci talebinin orijinal tetikleyicisi olup olmadığını belirlemektir. MB_CLIENT 'in birden çok çağrısı, paylaşılan bir örnek DB kullanılarak gerçekleştirilebilir.

Bir MB_CLIENT işleminin aktif olarak işlendiği süre boyunca girişlerin değişmemesi önemlidir. Bu kural çiğnenirse, MB_CLIENT aktif örneği belirleyemez.

Modbus iletişim fonksiyonunu seçmek için MB_MODE ve MB_DATA_ADDR parametrelerini kullanın.

MB_DATA_ADDR, erişilecek verilerin Modbus başlangıç adresi atar. MB_MODE yordamı, bir fonksiyon kodu girişi yerine bir MB_CLIENT girişi kullanır.

MB_MODE ve MB_DATA_ADDR değerlerinin kombinasyonu, gerçek Modbus mesajında kullanılan fonksiyon kodunu belirler. Aşağıdaki tabloda, MB_MODE parametresi, Modbus fonksiyon kodu ve Modbus fonksiyonu arasındaki ilişki gösterilmektedir:

Tablo 13-136 Modbus fonksiyonları

MB_MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	MB_DATA_ADDR
0	01	1 ila 2000	Çıkış Bit'ini oku: Talep başına 1 ila 2000 Bit	1 ila 9999
0	02	1 ila 2000	Giriş Bit'ini oku: Talep başına 1 ila 2000 Bit	10001 ila 19999
0	03	1 ila 125	Tutma kaydını oku: Talep başına 1 ila 125 sözcük	40001 ila 49999 veya 400001 ila 465535
0	04	1 ila 125	Giriş sözcüklerini oku: Talep başına 1 ila 125 sözcük	30001 ila 39999
1	05	1	Bir çıkış Bit'i yaz: Talep başına bir Bit	1 ila 9999
1	06	1	Bir tutma kaydı yaz: Talep başına 1 sözcük	40001 ila 49999 veya 400001 ila 465535
1	15	2 ila 1968	Birden fazla çıkış Bit'i yaz: Talep başına 2 ila 1968 Bit	1 ila 9999
1	16	2 ila 123	Birden fazla tutma kaydı yaz: Talep başına 2 ila 123 sözcük	40001 ila 49999 veya 400001 ila 465535
2	15	1 ila 1968	Bir veya birden fazla çıkış Bit'i yaz: Talep başına 1 ila 1968 Bit	1 ila 9999
2	16	1 ila 123	Bir veya birden fazla tutma kaydı yaz: Talep başına 1 ila 123 sözcük	40001 ila 49999 veya 400001 ila 465535
11	11	0	Sunucu iletişim durumu sözcüğünü ve olay sayacını oku. Status sözcüğü meşgul olduğunu gösterir (0 = meşgul değil, 0xFFFF = meşgul). Olay sayacı, bir mesaj başarıyla işlendiğinde her defasında artırılır. MB_CLIENT'in hem MB_DATA_ADDR parametresi hem de MB_DATA_LEN parametresi bu fonksiyonda dikkate alınmaz.	

MB_MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	MB_DATA_ADDR
80	08	1	Teşhis kodu 0x0000 ile sunucu durumunu kontrol et (kontrol döngüsü testi, sunucu talebin bir yankısını geri döndürür) Talep başına 1 sözcük	
81	08	1	Teşhis kodu 0x000A ile sunucu olay sayacını sıfırla Talep başına 1 sözcük	
3 ila 10, 12 ila 79, 82 ila 255			Rezerve	

NOT

MB_DATA_PTR , bir Modbus TCP sunucusundan okunan veya sunucuya yazılan verileri saklamak için bir tampon atar

Veri tamponu, optimize edilmemiş bir global DB'de veya markör bellek alanındaki bir adreste olabilir.

Markörlerin bellek alanındaki bir tampon için standart ANY işaretçi formatını kullanın. Bu, P#"Bit adresi" "Veri tipi" "Uzunluk" formatıdır. Bir örnek: P#M1000.0 WORD 500.

MB_DATA_PTR parametresi tarafından bir iletişim tamponunun atanması

- MB_CLIENT'ın iletişim fonksiyonları:
  - Modbus sunucu adreslerinden (00001 ila 09999) 1 Bit'lik verileri okur ve yazar
  - Modbus sunucu adreslerinden (10001 ila 19999) 1 Bit'lik verileri okur
  - Modbus sunucu adreslerinden (30001 - 39999) ve (40001 - 49999) 16 Bit sözcük verilerini okur
  - Modbus sunucu adreslerine (40001 ila 49999) 16 Bit sözcük verilerini yazar
- Sözcük veya Bit boyutu verileri, DB veya markör alanında MB_DATA_PTR tarafından atanan tampona veya tampondan aktarılır.
- Bir DB, MB_DATA_PTR tarafından tampon olarak atanırsa, tüm DB veri elemanlarına veri tipleri atamanız gerekir.
  - 1 Bit'lik Bool veri tipi, bir Modbus Bit adresini temsil eder.
  - WORD, UInt ve Int gibi münferit 16 Bit'lik sözcüklere sahip veri tipleri, bir Modbus sözcük adresini temsil eder.
  - DWORD, DInt ve Real gibi 32 Bit çift sözcük veri tipleri, iki Modbus sözcük adresini temsil eder.

- Karmaşık DB elemanları MB_DATA_PTR tarafından atanabilir, örn.
  - Standart Array'ler
  - Her elemanın benzersiz olduğu adlandırılmış yapılardır.
  - Her elemanın benzersiz bir adı ve 16 veya 32 Bit veri tipi olduğu adlandırılmış karmaşık yapılardır.
- MB_DATA_PTR için veri alanlarının aynı global veri bloğunda (veya markörlerin bellek alanında) olmaları zorunlu değildir. Modbus okuma işlemleri için bir veri bloğu, Modbus yazma işlemleri için diğer bir veri bloğu veya her MB_CLIENT istasyonu için bir veri bloğu oluşturabilirsiniz.

Birden çok istemci bağlantısı

Bir Modbus TCP istemcisi, PLC cihazı için izin verilen maksimum açık kullanıcı iletişim bağlantısı sayısı kadar eş zamanlı bağlantıları destekler. Modbus TCP istemcileri ve sunucuları da dahil olmak üzere bir PLC cihazı için toplam bağlantı sayısı, Açık Kullanıcı İletişim tipinin desteklenen maksimum bağlantı sayısını aşmasına izin verilmez. Modbus TCP bağlantıları, istemci ve/veya sunucu bağlantıları tarafından ortaklaşa kullanılabilir.

Münferit istemci bağlantıları aşağıdaki kurallara uymalıdır:

- Her MB_CLIENT bağlantısı, farklı bir örnek DB kullanmalıdır.
- Her MB_CLIENT bağlantısı, bir sunucunun benzersiz bir IP adresini belirtmelidir.
- Her MB_CLIENT bağlantısı, benzersiz bir bağlantı ID'si belirtmelidir.
- Sunucu konfigürasyonuna bağlı olarak benzersiz IP portu numaraları gerekebilir.

Bağlantı ID'si, her bir bağlantı için benzersiz olmalıdır. Bunun anlamı, her örnek DB için yalnızca tek bir benzersiz bağlantı ID'si kullanılabileceğidir. Özetle bunun anlamı, örnek DB ve bağlantı ID'sinin eşleşmiş ve her bağlantı için benzersiz olması gerektiğidir.

Tablo 13-137 MB_CLIENT örnek veri bloğunun kullanıcı tarafından erişilebilen statik değişkenleri

Değişken	Veri tipi	Ön ayar	Açıklama
Blocked_Proc_Timeout	Real	3.0	Bloke edilen bir Modbus istemci örneğini AKTİF olarak kaldırmadan önce bekleme süresi (saniye olarak). Bu, bir istemci talebi gönderildiğinde ve ardından bu talep tamamen işlenmeden önce uygulama istemci fonksiyonunu gerçekleştirmeyi durdurduğunda ortaya çıkabilir. S7-1200'de maksimum değer, 55 saniyedir.
MB_Unit_ID	Sözcük	255	Modbus cihaz kodu: Bir Modbus TCP sunucusu, IP adresi aracılığıyla adreslenir. Dolayısıyla MB_UNIT_ID parametresi Modbus TCP adreslemesi için kullanılmaz. MB_UNIT_ID parametresi, Modbus RTU protokolündeki Slave adres alanına karşılık gelir. Bir Modbus TCP sunucusu, bir Modbus RTU protokolüne Gateway olarak kullanıldığında, Slave cihaz, seri ağda MB_UNIT_ID kullanılarak tanımlanabilir. Bu durumda, MB_UNIT_ID parametresi bu talebi, doğru Modbus RTU Slave adresine iletir. Bazı Modbus TCP cihazları, başlatma için MB_UNIT_ID parametresinin sınırlı bir değer aralığında olmasını gerektirebilir.
RCV_TIMEOUT	Real	2,0	MB_CLIENT'in bir sunucunun bir talebe yanıt vermesini beklediği saniye cinsinden süre.
Bağlı	Bool	0	Atanan sunucuya bağlantının bağlı olup olmadığını belirtir: 1 = Bağlı, 0 = Bağlı değil

Tablo 13-138 MB_CLIENT protokol hatası

STATUS (W#16#)	Modbus istemcisine yanıt kodu (B#16#)	Modbus protokol hatası
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluğunda hata
8383	02	Veri adresi hatası veya MB_HOLD_REG adres aralığının sınırları dışında erişim
8384	03	Veri değerinde hata
8385	03	Veri teşhis kodunun değeri desteklenmiyor (fonksiyon kodu 08)

Tablo 13-139 MB_CLIENT yürütmesinin koşul kodları [1]

STATUS (W#16#)	MB_CLIENT parametre hatası
7001	MB_CLIENT, bir Modbus sunucusunun atanan TCP portunda bir bağlantı kurma veya sonlandırma talebine yanıt vermesini bekliyor. Bu, yalnızca bir bağlantı kurmanın veya sonlandırmanın ilk kez yürütülmesi için bildirilir.
7002	MB_CLIENT, bir Modbus sunucusunun atanan TCP portunda bir bağlantı kurma veya sonlandırma talebine yanıt vermesini bekliyor. Bu, bir bağlantı kurulumunun veya sonlandırılmasının tamamlanmasını beklerken takip eden tüm yürütmeler için bildirilir.
7003	Bir bağlantı başarıyla sonlandırıldı (yalnızca bir PLC çevrimi için geçerlidir).
80C8	Sunucu belirlenen süre içinde yanıt vermedi. MB_CLIENT, ayrılan süre içinde orijinal olarak gönderilen işlem ID'si aracılığıyla bir yanıt almalıdır, aksi takdirde bu hata yayınlanır. Modbus sunucu cihazına bağlantıyı kontrol edin. Bu hata yalnızca yapılandırılmış tüm yeniden denemeler (uygulanabilirse) gerçekleştirildikten sonra bildirilir.
8188	Mod değeri, geçersiz
8189	Veri adresi için geçersiz değer
818A	Veri uzunluğu için geçersiz değer.
818B	DATA_PTR alanının üzerinde geçersiz işaretçi. Bu, MB_DATA_ADDRESS + MB_DATA_LEN kombinasyonu olabilir.
818C	DATA_PTR işaretçisi, optimize edilmiş bir DB aralığını işaret ediyor (bu, optimize edilmemiş bir DB aralığı veya markörlerin bellek alanındaki bir alan olmalıdır)
8200	Port, mevcut bir Modbus talebini işlemekle meşgul.
8380	Alınan Modbus çerçevesinin formatı yanlış veya çok az sayıda Bayt alındı.
8387	Atanan bağlantı ID parametresi, önceki talepler için kullanılan ID'den farklı. MB_CLIENT'in her örnek DB'sinde yalnızca bir bağlantı ID'si kullanılabilir. Bir sunucudan alınan Modbus TCP protokol ID'si 0 değilse, bu aynı zamanda dahili bir hatadır.
8388	Bir Modbus sunucusu, talep edilen miktardan farklı bir miktarda veri geri döndürdü. Bu, sadece Modbus fonksiyonları 15 ve 16 için geçerlidir.

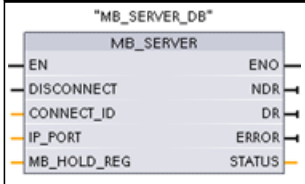
[1] Yukarıda listelenen MB_CLIENT hatalarının yanı sıra iletişim için temel T blok yordamları (TCON, TDISCON, TSEND ve TRCV (Sayfa 616)) tarafından da hatalar yayınlanır.

13.8.2.2 MB_SERVER (Modbus TCP sunucusu olarak PROFINET üzerinden iletişim)

MB_SERVER yordamı, S7-1200 CPU üzerindeki PROFINET bağlantısı üzerinden bir Modbus TCP sunucusu olarak iletişim kurar. MB_SERVER, bir Modbus TCP istemcisinden gelen bağlantı taleplerini işler, Modbus taleplerini alır ve işler ve yanıtları gönderir.
Yordamı kullanmak için ek bir donanım modülüne gerek yoktur.

DİKKAT
Güvenlik uyarıları Ağdaki her istemci, proses ekranındaki giriş ve çıkışlara ve Modbus tutma kaydında belirtilmiş olan veri bloğu veya Bit bellek alanlarına okuma ve yazma erişimine sahiptir. Bu opsiyonun amacı, bir IP adresine erişimi kısıtlamak ve yetkisiz okuma ve yazmayı önlemektir. Yine de paylaşılan adresin yetkisiz erişim için de kullanılabileceğini dikkate alınmalıdır.

Tablo 13-140 MB_SERVER yordamı

LAD/FBD	SCL	Açıklama
	**"MB_SERVER_DB"(** **DISCONNECT:=_bool_in_,** **CONNECT_ID:=_uint_in_,** **IP_PORT:=_uint_in_,** **NDR=>_bool_out_,** **DR=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,** **MB_HOLD_REG:=_variant_inout_);**	MB_SERVER, S7-1200 CPU üzerindeki PROFINET bağlantısı üzerinden bir Modbus TCP sunucusu olarak iletişim kurar. İletişim için ek bir donanım modülüne gerek yoktur. MB_SERVER, bir Modbus TCP istemcisi ile bağlantı kurma talebini kabul edebilir, bir Modbus fonksiyon talebi alabilir ve bir yanıt mesajı gönderebilir.

Tablo 13-141 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
DISCONNECT	IN	Bool	MB_SERVER, bir partner cihazla "pasif" bir bağlantı kurmaya çalışıyor. Bunun anlamı, sunucunun talep gönderen IP adreslerinden TCP bağlantı talepleri olup olmadığını pasif olarak izlemekte olduğudur. DISCONNECT = 0 ise ve bağlantı yoksa, pasif bir bağlantı kurulabilir. DISCONNECT = 1 ise ve bir bağlantı varsa, bağlantıyı kesmek için bir girişimde bulunulur. Bu şekilde, programınız bir bağlantının ne zaman kabul edildiğine kumanda eder. Bu girişin etkinleştirildiği her zaman başka hiçbir işlem başlatılmaz.
CONNECT_ID	IN	UInt	CONNECT_ID, PLC cihazındaki her bağlantıyı benzersiz olarak tanımlar. MB_CLIENT veya MB_SERVER yordamının her benzersiz örneği, benzersiz bir CONNECT_ID parametresi içermelidir.
IP_PORT	IN	UInt	Standart değer = 502: Bir Modbus istemcisi tarafından bir bağlantı talebine yanıt olarak denetlenen portu tanımlayan IP port numarası.
MB_HOLD_REG	IN_OUT	Variant	İşaretçi, MB_SERVER'ın Modbus tutma kaydı üzerinde: Tutma kaydı, optimize edilmemiş bir global DB veya markör bellek alanındaki bir adres olmalıdır. Bu bellek alanı, Modbus kayıt fonksiyonları 3 (okuma), 6 (yazma) ve 16 (yazma) kullanılarak bir Modbus istemcisi tarafından erişilebilen verileri saklamak için kullanılır.
NDR	OUT	Bool	Yeni veriler hazır: 0 = Yeni veri yok, 1 = Yeni verinin bir Modbus istemcisinden yazıldığını belirtir

Parametre ve veri tipi		Veri tipi	Açıklama
DR	OUT	Bool	Veriler oku: 0 = Yeni veri yok, 1 = Verilerin bir Modbus istemcisi tarafından okunduğunu belirtir
ERROR	OUT	Bool	ERROR Bit'i, MB_SERVER yürütmesi hatalı bir şekilde sona erdikten sonra bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

MB_SERVER'de, gelen Modbus fonksiyon kodları (1, 2, 4, 5 ve 15), S7-1200 CPU'nun giriş ve çıkışların proses görünümündeki Bit'leri ve sözcükleri doğrudan okuyabilir ve yazabilir. Veri aktarımı için kullanılan fonksiyon kodlarıyla (3, 6 ve 16), MB_HOLD_REG parametresi bir Bayt'tan büyük bir veri tipi olarak tanımlanmalıdır. Aşağıdaki tabloda, Modbus adreslerinin CPU'daki proses görünümüne atanması gösterilmektedir.

Tablo 13-142 Modbus adreslerinin proses görünümüne atanması

Modbus fonksiyonları						S7-1200	
Kodlar	Fonksiyon	Veri alanı	Adres aralığı			Veri alanı	CPU adresi
01	Bit'leri oku	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
02	Bit'leri oku	Giriş	10001	-	18192	Girişlerin proses görünümü	E0.0 ila E1023.7
04	Sözcükleri oku	Giriş	30001	-	30512	Girişlerin proses görünümü	EW0 ila EW1022
05	Bit yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
15	Bit'leri yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7

Gelen Modbus mesaj fonksiyon kodları (3, 6 ve 16), bir markörün bellek alanındaki bir adres aralığı veya veri bloğu olabilen bir Modbus tutma kaydındaki sözcükleri okur veya yazar. Tutma kaydı türü, MB_HOLD_REG parametresi tarafından belirlenir.

NOT

MB_HOLD_REG parametresinin atanması

Modbus tutma kaydı, optimize edilmemiş bir global DB veya markör bellek alanındaki bir adres olmalıdır.

Markörlerin bellek alanındaki bir Modbus tutma kaydı için standart Any işaretçisi formatını kullanın. Bu, P#"Bit adresi" "Veri tipi" "Uzunluk" formatıdır. Örnek: P#M1000.0 WORD 500.

Aşağıdaki tabloda, Modbus adreslerinin Modbus fonksiyon kodları 03 (sözcük oku), 06 (sözcük yaz) ve 16 (sözcük yaz) için kullanılan tutma kaydına nasıl eşleneceğine ilişkin örnekler gösterilmektedir. DB adreslerinin gerçek üst sınırı, ilgili CPU modelinin maksimum çalışma belleği ve maksimum M belleği tarafından belirlenir.

Tablo 13-143 Modbus adreslerinin CPU belleğindeki adreslerle nasıl atandığına ilişkin örnekler

Modbus adresi	MB_HOLD_REG parametrelerine örnekler		
	P#M100.0 Word 5	P#DB10.DBx0.0 Word 5	"Reçete".Bileşim maddesi
40001	MW100	DB10.DBW0	"Reçete".Bileşim maddesi[1]
40002	MW102	DB10.DBW2	"Reçete".Bileşim maddesi[2]
40003	MW104	DB10.DBW4	"Reçete".Bileşim maddesi[3]
40004	MW106	DB10.DBW6	"Reçete".Bileşim maddesi[4]
40005	MW108	DB10.DBW8	"Reçete".Bileşim maddesi[5]

Birden çok sunucu bağlantısı

Birden çok sunucu bağlantısı kurulabilir. Bu sayede, tek bir PLC cihazının aynı anda birden fazla Modbus TCP istemcisine bağlanması mümkündür.

Bir Modbus TCP sunucusu, PLC cihazı için izin verilen maksimum açık kullanıcı iletişim (Open User Communication) bağlantısı sayısına kadar eş zamanlı bağlantıları destekler. Modbus TCP istemcileri ve sunucuları da dahil olmak üzere bir PLC cihazı için toplam bağlantı sayısının, desteklenen maksimum açık kullanıcı iletişimi bağlantı sayısını aşmasına izin verilmez. Modbus TCP bağlantıları, istemci ve/veya sunucu bağlantıları tarafından ortaklaşa kullanılabilir.

Münferit sunucu bağlantıları aşağıdaki kurallara uymalıdır:

- Her MB_SERVER bağlantısı, farklı bir örnek DB kullanmalıdır.
- Her MB_SERVER bağlantısı, benzersiz bir IP portu numarası ile oluşturulmalıdır. Port başına yalnızca bir bağlantı desteklenir.
- Her MB_SERVER bağlantısı, benzersiz bir bağlantı ID'si kullanmalıdır.
- MB_SERVER, her bağlantı için ayrı olarak çağrılmalıdır (karşılık gelen örnek DB ile birlikte).

Bağlantı ID'si, her bir bağlantı için benzersiz olmalıdır. Bunun anlamı, her örnek DB için yalnızca tek bir benzersiz bağlantı ID'si kullanılabileceğidir. Özetle bunun anlamı, örnek DB ve bağlantı ID'sinin eşleşmiş ve her bağlantı için benzersiz olması gerektiğidir.

Tablo 13-144 Modbus teşhis fonksiyon kodları

MB_SERVER'ın Modbus teşhis fonksiyonu		
Kodlar	Kısmi fonksiyon	Açıklama
08	0x0000	Yankı testi sorgu verilerini yayınlar: MB_SERVER yordamı, alınan bir veri sözcüğünün yankısını bir Modbus istemcisine geri döndürür.
08	0x000A	İletişim olayı sayacını sil: MB_SERVER yordamı, Modbus fonksiyonu 11 için kullanılan iletişim olay sayacını siler.
11		İletişim olayı sayacını çağır: MB_SERVER yordamı, Modbus sunucusuna gönderilen başarılı Modbus okuma ve yazma taleplerinin sayısını saymak için dahili bir iletişim olay sayacı kullanır. Fonksiyon 8 veya fonksiyon 11'de sayaç artırılmaz. Sayacın sayma değeri, iletişim hatasıyla sonuçlanan talepler için de artırılmaz. Aynı anda yalnızca bir istemci-sunucu bağlantısı olduğundan, Broadcast fonksiyonu Modbus TCP ile kullanılamaz.

MB_SERVER değişkenleri

Bu tabloda, programınızda kullanılabilen MB_SERVER örnek veri bloğundaki genel statik değişkenler gösterilmektedir.

Tablo 13-145 MB_SERVER'ın genel statik değişkenleri

Değişken	Veri tipi	Standart değer	Açıklama
HR_Start_Offset	Sözcük	0	Modbus tutma kaydının başlangıç adresini atar
Request_Count	Sözcük	0	Bu sunucu tarafından alınan tüm taleplerin sayısı
Server_Message_Count	Sözcük	0	Belirli sunucu için alınan taleplerin sayısı
Xmt_Rcv_Count	Sözcük	0	Bir hata oluşan gönderimlerin veya veri alımlarının sayısı. Geçersiz bir Modbus mesajı olan bir mesaj alındığında da sayma değeri arttırılır.
Exception_Count	Sözcük	0	İade edilen bir istisna gerektiren Modbus'a özgü hatalar geri döndürüldü
Success_Count	Sözcük	0	Bu spesifik sunucu için protokol hatası olmaksızın alınan taleplerin sayısı.
Bağlı	Bool	0	Atanan istemciye bağlantının bağlı olup olmadığını belirtir: 1 = Bağlı, 0 = Bağlı değil

Programınız, HR_Start_Offset 'e değerler yazabilir ve Modbus sunucu işlemlerine kumanda edebilir. Diğer değişkenler ise Modbus durumunu izlemek için okunabilir.

HR_Start_Offset

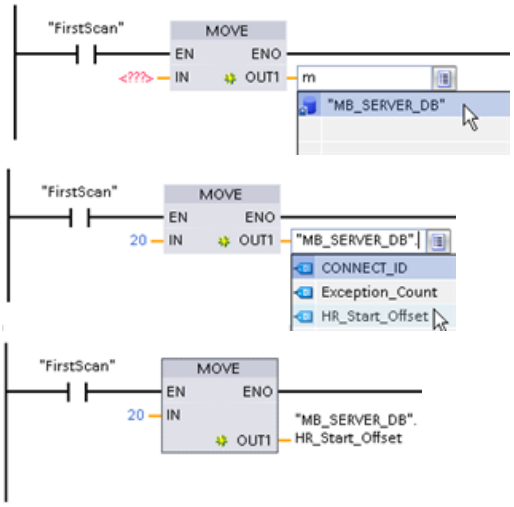
Modbus tutma kaydı adresleri 40001'den başlar. Bu adresler, hedef sistem belleğindeki tutma kaydının başlangıç adresine karşılık gelir. Ancak Modbus tutma kaydı için 40001 dışında bir başlangıç adresi yapılandırmak isterseniz, HR_Start_Offset değişkenini yapılandırabilirsiniz. Örneğin, MW100'de başlayan ve 100 sözcük uzunluğunda olan bir tutma kaydı yapılandırabilirsiniz. Tutma kaydı başlangıç adresini, 20'lik bir ofset ile 40001 yerine 40021 olarak belirtirsiniz. Bu durumda 40021'in altındaki ve 40119'un üzerindeki her adres, adresleme hatasına yol açacaktır.

Tablo 13-146 Modbus tutma kaydının adresleme örneği

HR_Start_Offset	Adres	Minimum	Maksimum
0	Modbus adresi (sözcük)	40001	40099
	S7-1200 adresi	MW100	MW298
20	Modbus adresi (sözcük)	40021	40119
	S7-1200 adresi	MW100	MW298

HR_Start_Offset, Modbus tutma kaydının başlangıç adresini belirten bir sözcük değeridir ve MB_SERVER örnek veri bloğunda saklanır. MB_SERVER'ı programınıza eklenmesinin ardından, bu açık statik değişkeni parametre açılır listesinden seçebilirsiniz.

Örneğin bir LAD ağına MB_SERVER eklediyseniz, bir önceki bir ağa gidip, HR_Start_Offset değerini atayabilirsiniz. Değer, MB_SERVER yürütülmeden önce atanmalıdır.



Modbus sunucusu için bir değişken girilmesi Standart DB adı üzerinden:

1. İmleci parametre alanına getirin ve m karakterini girin.
2. DB adlarının açılır listesinden "MB_SERVER_DB" öğesini seçin.
3. DB değişkenleri açılır listesinden "MB_SERVER_DB.HR_Start_Offset" öğesini seçin.

Tablo 13-147 MB_SERVER yürütmesinin koşul kodları [1]

STATUS (W#16#)	Modbus sunucusuna yanıt kodu(B#16#)	Modbus protokol hatası
7001		MB_SERVER, atanan TCP portuna bağlanmak için bir Modbus istemcisini bekliyor. Bu kod, bir bağlantı kurma işlemi ilk kez yürütüldüğünde veya sonlandırıldığında bildirilir.
7002		MB_SERVER, atanan TCP portuna bağlanmak için bir Modbus istemcisini bekliyor. Bu kod, bir bağlantı kurulumunun veya sonlandırılmasının tamamlanmasını beklerken sonraki tüm yürütmeler için bildirilir.
7003		Bir bağlantı başarıyla sonlandırıldı (yalnızca bir PLC çevrimi için geçerlidir).
8187		MB_HOLD_REG üzerinde geçersiz işaretçi: Aralık, çok küçük.
818C		Optimize edilmiş MB_HOLD_REG alanı üzerinde işaretçi, (bu, optimize edilmemiş bir DB alanı veya markörlerin bellek alanındaki bir alan olmalıdır) veya bloke edilmiş işlem için zaman aşımı 55 saniye sınırını aşıyor. (S7-1200'e özgü)
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluğunda hata
8383	02	Veri adresi hatası veya MB_HOLD_REG adres aralığının sınırları dışında erişim
8384	03	Veri değerinde hata
8385	03	Veri teşhis kodunun değeri desteklenmiyor (fonksiyon kodu 08)

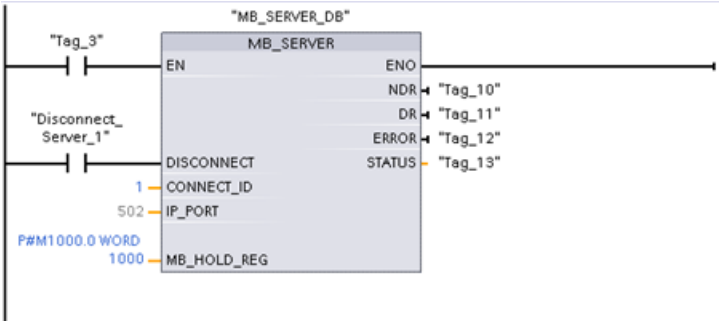
[1] Yukarıda listelenen MB_SERVER hatalarının yanı sıra iletişim için temel T blok yordamları (TCON, TDISCON, TSEND ve TRCV (Sayfa 616)) tarafından da hatalar bildirilir.

13.8.3 Daha eski Modbus TCP örnekleri

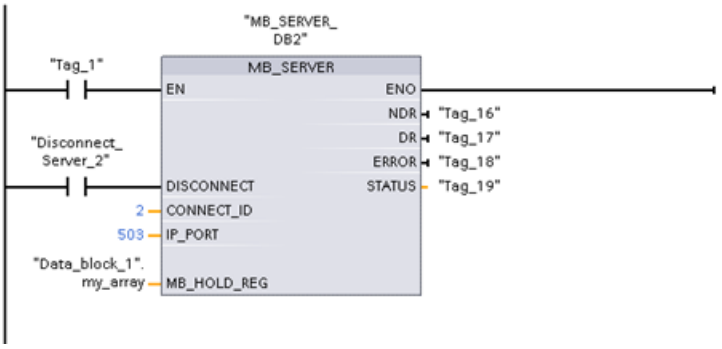
13.8.3.1 Örnek: Eski sistemlerdeki çoklu TCP-bağlantıları için MB_SERVER

Modbus TCP sunucusunda birden fazla bağlantınız olabilir. Bunun için MB_SERVER her bağlantı için bağımsız olarak çalıştırılmalıdır. Her bağlantı bağımsız bir örnek DB, bağlantı ID'si ve IP portu kullanmalıdır. S7-1200 için IP portu başına sadece bir bağlantıya izin verilir. Optimum işletim performansı elde etmek için MB_SERVER her bir bağlantı için her program çevriminde yürütülmelidir.

Ağ 1: Bağlantı no. 1, bağımsız IP_PORT, bağlantı ID'si ve örnek DB ile



Ağ 2: Bağlantı no. 2, bağımsız IP_PORT, bağlantı ID'si ve örnek DB ile



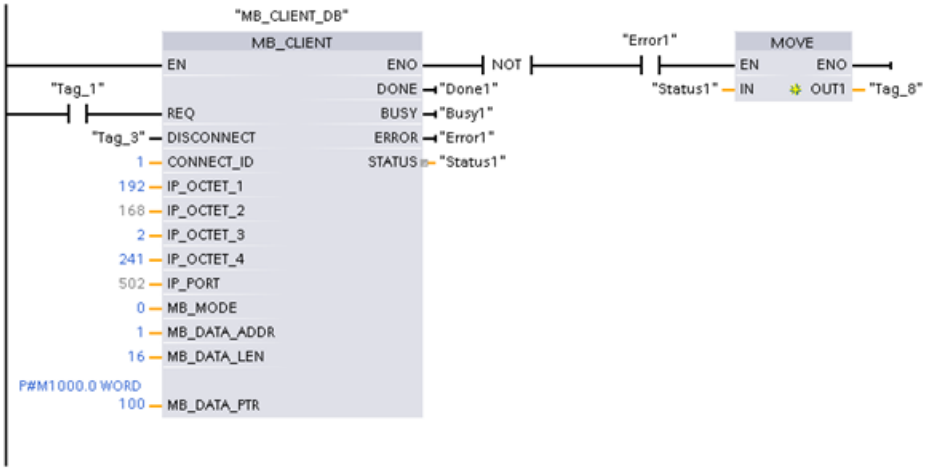
13.8.3.2 Örnek: Eski sistemlerdeki MB_CLIENT 1: Ortak TCP bağlantısında çoklu talepler

Aynı bağlantı üzerinden birden çok sayıda Modbus istemci talebi gönderilebilir. Bunun için aynı örnek DB, bağlantı ID'si ve port numarasını kullanın.

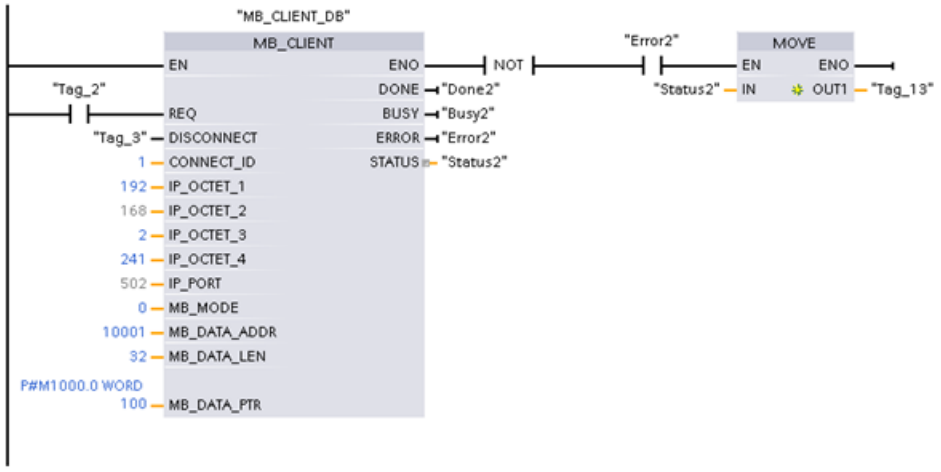
Bir seferde yalnızca 1 istemci aktif olabilir. Bir istemcinin yürütmeyi tamamlamasından sonra, bir sonraki istemci yürütmeye başlar. Yürütme sıralamasından programınız sorumludur.

Bu örnekte, her iki istemcinin de aynı bellek alanına yazdığı gösterilmektedir. Ayrıca, yayınlanan bir hata yakalanır (opsiyonel fonksiyon).

Ağ 1: Modbus fonksiyonu 1 - Çıkışların proses görünümünde 16 Bit okur



Ağ 2: Modbus fonksiyonu 2 - Girişlerin proses görünümünde 32 Bit okur



13.8.3.3 Örnek: Eski sistemlerdeki MB_CLIENT 2: Farklı TCP bağlantılarıyla çoklu talepler

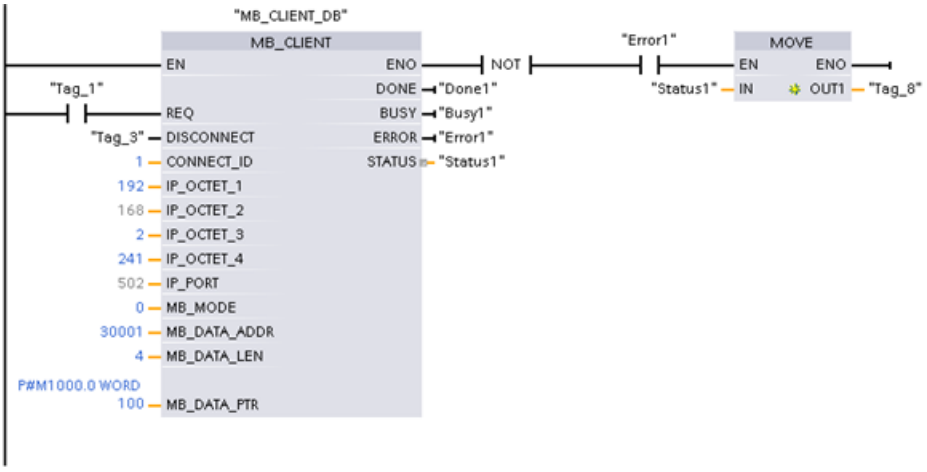
Birden çok Modbus istemci talebi, farklı bağlantılar üzerinden gönderilebilir. Bunun için farklı örnek DB'leri, IP adresleri ve bağlantı ID'leri kullanmalısınız.

Aynı Modbus sunucusuna bağlantı kuruluyorsa, Port numaraları farklı olmalıdır. Farklı sunuculara bağlantılar için port numarasında herhangi bir kısıtlama yoktur.

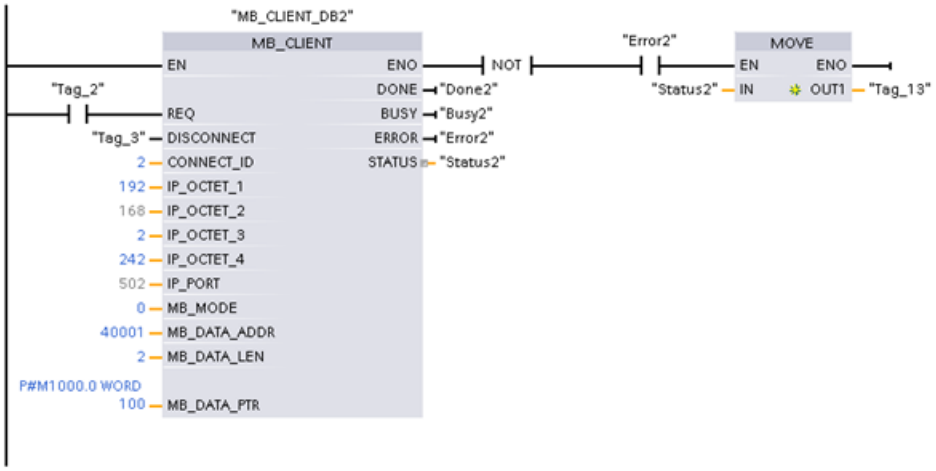
Bu örnekte, her iki istemcinin de aynı bellek alanına yazdığı gösterilmektedir. Ayrıca, yayınlanan bir hata yakalanır (opsiyonel fonksiyon).

Ağ 1:

Modbus fonksiyonu 4 - Giriş sözcüklerini okur (S7-1200 belleğinde)



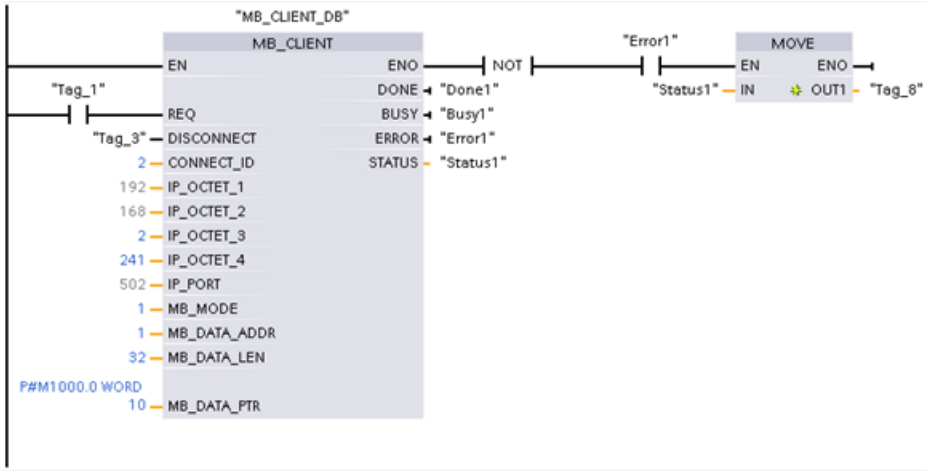
Ağ 2: Modbus fonksiyonu 3 - Bir Modbus TCP sunucusunun tutma kaydındaki sözcükleri okur



13.8.3.4 Örnek: Eski sistemlerdeki MB_CLIENT 3: Çıkışların proses görünümü için yazma talebi

Bu örnekte, bir Modbus istemcisinin S7-1200 çıkışlarının proses görünümüne yazma talebi gösterilmektedir.

Ağ 1: Modbus fonksiyonu 15 - S7-1200 çıkışlarının proses görünümüne Bit yazma

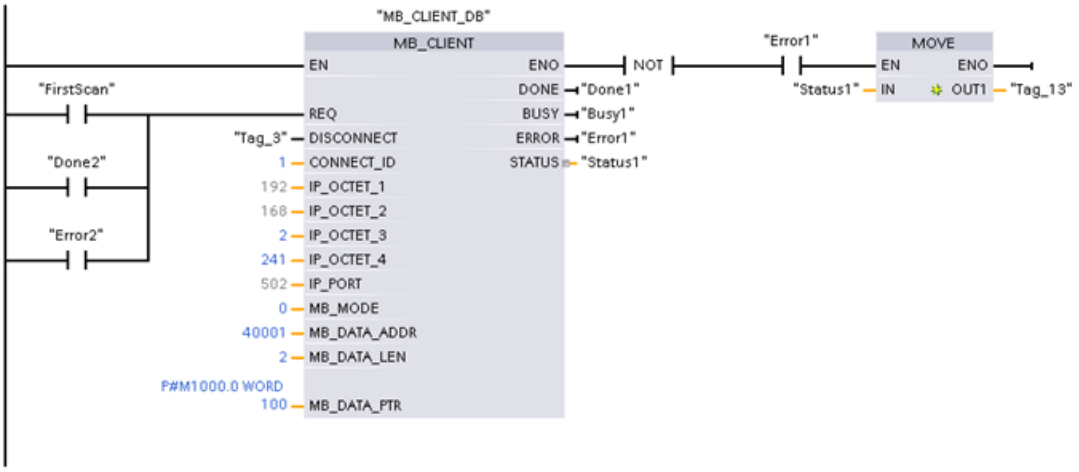


13.8.3.5 Örnek: Eski sistemlerdeki MB_CLIENT 4: Birden fazla talebin koordine edilmesi

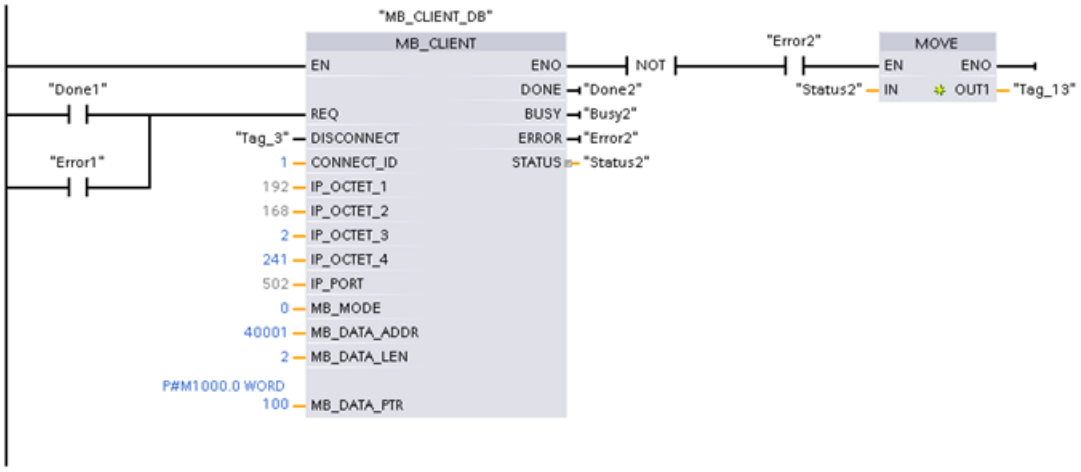
Her bir Modbus TCP talebinin yürütülmesinin tamamlandığından emin olmalısınız. Bu koordinasyon programınız tarafından gerçekleştirilmelidir. Aşağıdaki örnekte, birinci ve ikinci istemci taleplerinin çıkışları kullanılarak yürütmenin nasıl koordine edilebileceği gösterilmektedir.

Bu örnekte, her iki istemcinin de aynı bellek alanına yazdığı gösterilmektedir. Ayrıca, yayınlanan bir hata yakalanır (opsiyonel fonksiyon).

Ağ 1: Modbus fonksiyonu 3 - Tutma kaydındaki sözcükleri okur



Ağ 2: Modbus fonksiyonu 3 - Tutma kaydındaki sözcükleri okur



13.9 Eski sistemlerdeki Modbus RTU iletişimi (yalnızca CM/CB 1241)

13.9.1 Genel bakış

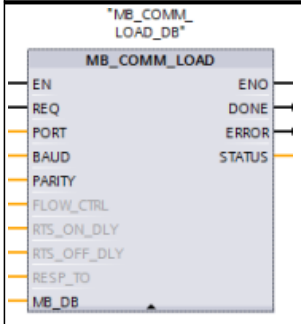
STEP 7 V13 SP1 ve S7-1200 V4.1 CPU'lardan önce, Modbus RTU iletişim yordamları farklı adlara ve bazen de farklı arayüzlere sahipti. Her iki yordam türü için de genel konseptler geçerlidir. Programlamayla ilgili bilgileri, münferit eski Modbus RTU yordamlarıyla ilgili açıklamalarda bulabilirsiniz.

STEP 7, Modbus RTU için yordamların çeşitli sürümlerini sunmaktadır. Yordamların sürümleriyle ilgili bilgileri, STEP 7 bilgi sistemindeki Yordam sürümlerinin kullanımı (https://support.industry.siemens.com/cs/ww/en/view/109798671/113722878475) bölümünde bulabilirsiniz.

13.9.2 Eski Modbus RTU yordamları

13.9.2.1 MB_COMM_LOAD (Modbus RTU için PtP modülü üzerindeki portun yapılandırılması)

Tablo 13-148 MB_COMM_LOAD yordamı

LAD/FBD	SCL	Açıklama
	**"MB_COMM_LOAD_DB"(REQ:=_bool_in, PORT:=_uint_in_, BAUD:=_udint_in_, PARITY:=_uint_in_, FLOW_CTRL:=_uint_in_, RTS_ON_DLY:=_uint_in_, RTS_OFF_DLY:=_uint_in_, RESP_TO:=_uint_in_, DONE=>_bool_out, ERROR=>_bool_out_, STATUS=>_word_out_, MB_DB:=_fbtref_inout_);**	MB_COMM_LOAD yordamı, Modbus RTU protokolü üzerinden iletişim için bir PtP portunu yapılandırır. Modbus portunun donanım opsiyonları: Maks. üç CM (RS485 veya RS232), artı bir CB (R4845) montajı MB_COMM_LOAD yordamını programınıza eklerseniz, otomatik olarak bir örnek veri bloğu atanır.

Tablo 13-149 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır. (Yalnızca sürüm 2.0)
PORT	IN	Port	CM'yi veya CB'yi kurduktan ve yapılandırdıktan sonra, PORT kutusu bağlantısındaki parametre açılır listesinde port kodu görünür. Atanan CM veya CB port değeri, cihaz konfigürasyonunun "Donanım kodu" özelliğidir. Portun sembolik adı, PLC değişkenler tablosunun "Sistem sabitleri" sekmesinde atanır.
BAUD	IN	UDInt	Baud hızının seçimi: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200. Diğer tüm veriler geçersizdir.
PARITY	IN	UInt	Parite seçimi: • 0 – yok • 1 – tek sayılı • 2 – çift sayılı
FLOW_CTRL [1]	IN	UInt	Akış kontrolü seçimi: • 0 – (standart) akış kontrolü yok • 1 – RTS her zaman AÇIK ile donanım akış kontrolü (RS485 portları için geçerli değildir) • 2 – RTS ile anahtarlanan donanım akış kontrolü
RTS_ON_DLY [1]	IN	UInt	RTS açma gecikmesinin seçimi: • 0 - (standart) "RTS aktif"ten mesajın ilk karakteri gönderilinceye dek gecikme yok. • 1 ila 65535 - Mesajın ilk karakteri gönderilinceye dek "RTS aktif "ten itibaren milisaniye cinsinden gecikme (RS485 portları için geçerli değildir). RTS gecikmeleri, FLOW_CTRL seçiminden bağımsız olarak uygulanmalıdır.

[1] MB_COMM_LOAD için opsiyonel parametreler (V 2.x veya üstü). LAD/FBD kutusunun altındaki oka tıklayarak bu parametreleri kapsayacak şekilde kutuyu genişletin.

Parametre ve veri tipi		Veri tipi	Açıklama
RTS_OFF_DLY [1]	IN	UInt	RTS kapatma gecikmesinin seçimi: • 0 – (standart) son karakterin aktarımından sonra "RTS aktif değil" durumuna kadar gecikme olmaz. • 1 ila 65535 - Son karakterin aktarımından "RTS aktif değil" durumuna kadar geçen milisaniye cinsinden gecikme (RS485 portları için geçerli değildir). RTS gecikmeleri, FLOW_CTRL seçiminden bağımsız olarak uygulanmalıdır.
RESP_TO [1]	IN	UInt	Yanıt zaman aşımı: MB_MASTER'ın Slave'den yanıt beklediği milisaniye cinsinden süre. Slave bu süre içinde yanıt vermezse, MB_MASTER talebi tekrarlar veya tekrar gönderim belirli bir sayıya ulaşmışsa, ilgili talebi bir hata ile sonlandırır. 5 msn ila 65535 msn (standart değer = 1000 msn).
MB_DB	IN	Variant	MB_MASTER veya MB_SLAVE yordamlarının örnek veri bloğuna bir referans. MB_SLAVE veya MB_MASTER'ı programınıza ekledikten sonra, ilgili DB kodu, MB_DB-Box bağlantısındaki parametre açılır listesinde görünür.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur. (yalnızca sürüm 2.0)
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

[1] MB_COMM_LOAD için opsiyonel parametreler (V 2.x veya üstü). LAD/FBD kutusunun altındaki oka tıklayarak bu parametreleri kapsayacak şekilde kutuyu genişletin.

MB_COMM_LOAD, Modbus RTU protokolü için bir portunu yapılandırmak üzere yürütülüyor. Bir port, Modbus RTU protokolü için yapılandırıldıktan sonra, yalnızca MB_MASTER veya MB_SLAVE yordamları tarafından kullanılabilir.

Modbus iletişimi için kullanılan her bir iletişim portunun konfigürasyonu için bir MB_COMM_LOAD yordamının yürütülmesi gerekir. Kullandığınız her porta benzersiz bir MB_COMM_LOAD örnek DB atamanız gerekir. CPU'ya en fazla üç iletişim modülü (RS232 veya RS485) ve bir iletişim kartı (RS485) monte edebilirsiniz. MB_COMM_LOAD yordamını yol alma OB'sinden çağırıp bir kez çalıştırabilirsiniz veya ilk çevrim için sistem markörünü (Sayfa 86) kullanarak bir seferlik yürütme için çağrıyı başlatabilirsiniz. Yalnızca Baud hızı veya parite gibi iletişim parametrelerinin değiştirilmesi gerekiyorsa MB_COMM_LOAD yordamını tekrar çalıştırın.

Programınıza MB_MASTER veya MB_SLAVE eklerseniz, yordama bir örnek veri bloğu atanır. MB_COMM_LOAD yordamında MB_DB parametresini belirtirseniz, bu örnek veri bloğuna referans verilir.

MB_COMM_LOAD veri bloğu değişkenleri

Aşağıdaki tabloda, MB_COMM_LOAD örnek DB'sinde programınızda kullanılabilecek genel statik değişkenler gösterilmektedir:

Tablo 13-150 Örnek DB'deki statik değişkenler

Değişken	Veri tipi	Açıklama
ICHAR_GAP	UInt	Karakterler arasındaki karakter aralığı için gecikme. Bu parametre milisaniye cinsinden belirtilir ve alınan karakterler arasında beklenen süreyi artırmak için kullanılır. Bu parametre için karşılık gelen Bit sürelerinin sayısı, Modbus standart değeri olan 35 Bit zamanına (3,5 karakter zamanı) eklenir.
RETRIES	UInt	Master'ın "Yanıt yok" için 0x80C8 hata kodunu döndürmeden önce yürüttüğü yeniden denemelerin sayısı.
STOP_BITS	USInt	Bir karakterin her karesi için Stop Bit'lerinin sayısı. Geçerli değerler 1 ve 2'dir.

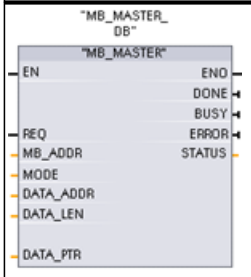
Tablo 13-151 MB_COMM_LOAD'un yürütme koşulu kodları [1]

STATUS (W#16#)	Açıklama
0000	Hata yok
8180	Geçersiz port ID değeri (iletişim modülünün yanlış portu/yanlış donanım kodu)
8181	Baud hızı için geçersiz değer.
8182	Parite için geçersiz değer.
8183	Akış kontrolü için geçersiz değer.
8184	Yanıt zaman aşımı süresi için geçersiz değer (yanıt zaman aşımı, minimum değer olan 5 msn'den küçük)
8185	MB_DB parametresi bir MB_MASTER veya MB_SLAVE yordamının örnek veri bloğu değildir.

[1] Yukarıda belirtilen MB_COMM_LOAD yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

13.9.2.2 MB_MASTER (Modbus RTU Master olarak PtP portu üzerinden iletişim)

Tablo 13-152 MB_MASTER yordamı

LAD/FBD	SCL	Açıklama
	**"MB_MASTER_DB"(REQ:=_bool_in_, MB_ADDR:=_uint_in_, MODE:=_usint_in_, DATA_ADDR:=_udint_in_, DATA_LEN:=_uint_in_, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, DATA_PTR:=_variant_inout_);**	MB_MASTER yordamı, MB_COMM_LOAD yordamının önceki bir yürütmesi tarafından yapılandırılan bir port üzerinden bir Modbus Master olarak iletişim kurar. MB_MASTER yordamını programınıza eklerseniz, otomatik olarak bir örnek veri bloğu atanır. Bu MB_MASTER örnek veri bloğu, MB_COMM_LOAD yordamında MB_DB parametresini belirtirseniz kullanılır.

Tablo 13-153 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
REQ	IN	Bool	0 = Talep yok 1 = Modbus Slave'e veri gönderme talebi

Parametre ve veri tipi		Veri tipi	Açıklama
MB_ADDR	IN	V1.0: USInt V2.0: UInt	Modbus-RTU istasyon adresi: Standart adresleme aralığı (1 - 247) Genişletilmiş adresleme aralığı (1 - 65535) 0 değeri, bir mesajı tüm Modbus Slave'lere Broadcast vermek için rezerve edilmiştir. Broadcast için yalnızca Modbus fonksiyon kodları 05, 06, 15 ve 16 desteklenir.
MODE	IN	USInt	Seçim modu: Talep türünü belirtir (okuma, yazma veya teşhis). Daha fazla bilgiyi, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
DATA_ADDR	IN	UDInt	Slave'deki başlangıç adresi: Modbus Slave'de erişilecek verilerin başlangıç adresini belirtir. Geçerli adresleri, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
DATA_LEN	IN	UInt	Veri uzunluğu: Bu talep tarafından erişilecek Bit veya sözcük sayısını belirtir. Geçerli uzunlukları, aşağıdaki Modbus fonksiyonları tablosunda bulabilirsiniz.
DATA_PTR	IN	Variant	Veri işaretçisi: Yazılacak veya okunacak veri için M veya DB adresine (optimize edilmemiş DB) işaret eder.
DONE	OUT	Bool	DONE Bit'i, kendisinden sonra son talebin hatasız olarak yürütüldüğü bir çevrim boyunca DOĞRU'dur.
BUSY	OUT	Bool	• 0 - İşlenen MB_MASTER yordamı yok • 1 - MB_MASTER yordamı işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. STATUS parametresindeki hata kodu, sadece ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütme koşulu

Modbus Master iletişimi için kurallar

- MB_MASTER yordamının bu portla iletişim kurabilmesi için, bir portu yapılandırmak üzere MB_COMM_LOAD yordamının yürütülmesi gerekir.
- Bir port Modbus Master taleplerini başlatmak için kullanılacaksa, bu portun MB_SLAVE tarafından kullanılmasına izin verilmez. Bu port ile bir veya daha fazla MB_MASTER örneği kullanılabilir. Ancak MB_MASTER'ın tüm yürütmeleri bu port için aynı MB_MASTER örnek DB'sini kullanmalıdır.
- Modbus yordamları, iletişim sürecine kumanda etmek için iletişim alarm olaylarını kullanmaz. Programınız tamamlanan gönderim ve alım olayları için MB_MASTER yordamını sorgulamalıdır.
- Bir program çevrimi OB'sinden belirli bir port için MB_MASTER'ın tüm yürütmelerini çağırırsınız. Modbus Master yordamları bir seferde yalnızca bir program çevriminde veya bir seferde bir periyodik/zaman gecikmeli yürütme katmanında yürütülebilir. Her iki yürütmenin öncelik katmanında bunların işlenmesine izin verilmez. Bir Modbus Master yordamının daha yüksek önceliğe sahip bir yürütme katmanındaki başka bir Modbus Master yordamı tarafından öncelikli olarak kesintiye uğratılması, usulüne uygun bir işletimle sonuçlanmaz. Modbus Master yordamlarının yol alma, teşhis veya zaman hatası yürütme katmanlarında işlenmesine izin verilmez.

- Bir Master yordamı bir aktarımı tetikledikten sonra, bu örnek, DONE = 1 veya ERROR = 1 durumu çıkana kadar etkinleştirilmiş EN girişi ile sürekli olarak yürütülmelidir. Belirli bir MB_MASTER örneği, bu iki olaydan biri gerçekleşene dek aktif olarak kabul edilir. Orijinal örnek aktifken, REQ girişi etkinleştirilmiş başka bir örneğe yapılan herhangi bir çağrı hata ile sonuçlanır. Orijinal örneğin devam eden yürütülmesi durursa, talep durumu Blocked_Proc_Timeout statik değişkeni tarafından belirtilen süre boyunca aktif kalır. Bu süre geçtikten sonra, REQ girişi etkinleştirilmiş bir sonraki çağrılan Master yordamı aktif örnek haline gelir. Bu sayede, tek bir Modbus Master yordamının monopolleşmesi veya porta erişimi engellemesi önlenir. Orijinal aktif örnek "Blocked_Proc_Timeout" statik değişkeni tarafından belirtilen süre boyunca etkinleştirilmezse, bu örnekten gelen bir sonraki yürütme (REQ girişi ayarlanmaksızın) aktif durumu siler. Eğer (REQ ayarlanmışsa), bu yürütme, başka hiçbir örnek aktif değilmiş gibi yeni bir Master talebini tetikler.

REQ parametresi

0 = Talep yok, 1 = Modbus Slave'e veri gönderme talebi

Bu girişe, seviye veya kenar kontrollü bir kontak aracılığıyla kumanda edebilirsiniz. Bu girişin her etkinleştirilmesinde, aynı örnek DB'yi kullanan başka bir MB_MASTER yordamının güncel talep işlenene dek bir talepte bulunamamasını sağlamak için bir durum makinesi başlatılır. Diğer tüm giriş durumları, yanıt alıncaya veya bir hata tespit edilinceye dek güncel talep için dahili olarak alınır ve saklanır.

Güncel talep tamamlanmadan önce aynı MB_MASTER örneği REQ girişi = 1 ile tekrar çalıştırılırsa, sonraki aktarımlar yürütülmeyecektir. Ancak talep işlendiğinde, MB_MASTER, REQ girişi = 1 ile tekrar yürütüldüğünde yeni bir talep yayınlanır.

Modbus fonksiyon tipini seçmek için DATA_ADDR ve MODE parametrelerini kullanın.

DATA_ADDR (Slave'deki Modbus başlangıç adresi): Modbus Slave'de erişilecek verilerin başlangıç adresini belirtir.

MB_MASTER yordamı, bir fonksiyon kodu girişi yerine bir MODE girişi kullanır. MODE ve Modbus adresi kombinasyonu, gerçek Modbus mesajında kullanılan fonksiyon kodunu belirler. Aşağıdaki tabloda, MODE parametresi, Modbus fonksiyon kodu ve Modbus adres aralığı arasındaki ilişki gösterilmektedir:

Tablo 13-154 Modbus fonksiyonları

MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	Modbus adresi
0	01	1 ila 2000 1 ila 1992 [1]	Çıkış Bit'ini oku: Talep başına 1 ila (1992 veya 2000) Bit	1 ila 9999
0	02	1 ila 2000 1 ila 1992 [1]	Giriş Bit'ini oku: Talep başına 1 ila (1992 veya 2000) Bit	10001 ila 19999
0	03	1 ila 125 1 ila 124 [1]	Tutma kaydını oku: Talep başına 1 ila (124 veya 125) sözcük	40001 ila 49999 veya 400001 ila 465535
0	04	1 ila 125 1 ila 124 [1]	Giriş sözcüklerini oku: Talep başına 1 ila (124 veya 125) Bit	30001 ila 39999
1	05	1	Bir çıkış Bit'i yaz: Talep başına bir Bit	1 ila 9999

[1] Genişletilmiş adresleme ile maksimum veri uzunluğu, fonksiyonun veri tipine bağlı olarak 1 Bayt veya 1 sözcük daha azdır.

MODE	Modbus fonksiyonu	Verilerin uzunluğu	İşlem ve veriler	Modbus adresi
1	06	1	Bir tutma kaydı yaz: Talep başına 1 sözcük	40001 ila 49999 veya 400001 ila 465535
1	15	2 ila 1968 2 ila 1960 [1]	Birden fazla çıkış Bit'i yaz: Talep başına 2 ila (1960 veya 1968) Bit	1 ila 9999
1	16	2 ila 123 2 ila 122 [1]	Birden fazla tutma kaydı yaz: Talep başına 2 ila (122 veya 123) sözcük	40001 ila 49999 veya 400001 ila 465535
2	15	1 ila 1968 2 ila 1960 [1]	Bir veya birden fazla çıkış Bit'i yaz: Talep başına 1 ila (1960 veya 1968) Bit	1 ila 9999
2	16	1 ila 123 1 ila 122 [1]	Bir veya birden fazla tutma kaydı yaz: Talep başına 1 ila (122 veya 123) sözcük	40001 ila 49999 veya 400001 ila 465535
11	11	0	Slave iletişim durumu sözcüğünü ve olay sayacını oku. Status sözcüğü meşgul olduğunu gösterir (0 = meşgul değil, 0xFFFF = meşgul). Olay sayacı, bir mesaj başarıyla işlendiğinde her defasında artırılır. MB_MASTER'ın DATA_ADDR ve DATA_LEN işlemcilerinde bu fonksiyon yok sayılır.	
80	08	1	Teşhis kodu 0x0000 ile Slave durumunu kontrol et (kontrol döngüsü testi, Slave talebin bir yankısını geri döndürür) Talep başına 1 sözcük	
81	08	1	Teşhis kodu 0x000A ile Slave olay sayacını sıfırla Talep başına 1 sözcük	
3 ila 10, 12 ila 79, 82 ila 255			Rezerve	

[1] Genişletilmiş adresleme ile maksimum veri uzunluğu, fonksiyonun veri tipine bağlı olarak 1 Bayt veya 1 sözcük daha azdır.

DATA_PTR parametresi

DATA_PTR parametresi, yazılacak veya okunacak DB veya M adresini gösterir. Bir veri bloğu kullanırsanız, Modbus Slave'lerde okuma ve yazma işlemleri için veri belleği sağlayan bir global veri bloğu oluşturmanız gerekecektir.

NOT

DATA_PTR veri bloğu, direkt adreslemeye izin vermelidir.

Veri bloğu, doğrudan (mutlak) ve sembolik adreslemeye izin vermelidir. Veri bloğunu oluştururken "Standart" erişim özniteliğini seçmelisiniz.

DATA_PTR parametresi için veri bloğu yapıları:

- Bu veri tipleri, 30001 ila 39999, 40001 ila 49999 ve 400001 ila 465536 Modbus adreslerinden **sözcük okumak** ve 40001 ila 49999 ve 400001 ila 465536 Modbus adreslerine **sözcük yazmak** için geçerlidir.
  - WORD, UINT veya INT veri tiplerinin standart Array'i
  - WORD, UINT veya INT türünde, her bir elemanın benzersiz bir ada ve 16 Bit'lik bir veri tipine sahip olduğu adlandırılmış yapı.
  - Her elemanın benzersiz bir ada ve 16 veya 32 Bit'lik veri tipine sahip olduğu adlandırılmış karmaşık yapı.
- Modbus adresleri 00001 ila 09999 için Bit'leri **okumak** ve yazmak ve 10001 ila 19999 arasındaki Bit'leri okumak için kullanılır.
  - Boole veri tipleri için standart alan.
  - Benzersiz olarak adlandırılmış Boole değişkenlerinin adlandırılmış Boole yapısı.
- Gerekli olmamakla birlikte, her MB_MASTER yordamının kendisi için ayrı bir bellek alanına sahip olması önerilir. Bunun nedeni, birden fazla MB_MASTER yordamının aynı bellek alanında okuma ve yazması durumunda veri imhası olasılığının daha yüksek olmasıdır.
- DATA_PTR için veri alanlarının aynı global veri bloğunda olmaları zorunlu değildir. Modbus okuma işlemleri için birden çok veri bloğu, Modbus yazma işlemleri için bir veri bloğu veya her Slave istasyonu için bir veri bloğu oluşturabilirsiniz.

Modbus Master veri bloğundaki değişkenler

Aşağıdaki tabloda, MB_MASTER örnek DB'sinde programınızda kullanılabilecek genel statik değişkenler gösterilmektedir:

Tablo 13-155 Örnek DB'deki statik değişkenler

Değişken	Veri tipi	Başlangıç değeri	Açıklama
Blocked_Proc_Timeout	Real	3.0	Bloke edilen bir Modbus Master örneğini AKTİF olarak kaldırmadan önce bekleme süresi (saniye olarak). Bu, örneğin bir Master talebi yayınlandığında ve program durduğunda ve talebi tamamen bitirmeden önce Master fonksiyonunu çağırdığında oluşabilir. Zaman değeri 0'dan büyük ve 55 saniyeden küçük olmalıdır, aksi durumda bir hata oluşur. Standart değer, 0,5 saniyedir.
Extended_Addressing	Bool	Yanlış	Slave adreslemesini tek veya çift Bayt olarak yapılandırır. Standart değer, 0'dır. (0 = Bir Bayt'lık adres, 1 = iki Bayt'lık adres)

Programınız, Modbus Master işlemlerine kumanda etmek için Blocked_Proc_Timeout ve Extended_Addressing değişkenlerine değerler yazabilir. Bu değişkenlerin programlama editöründe nasıl kullanılacağına dair bir örneği ve genişletilmiş Modbus adresleme hakkında daha fazla bilgiyi, MB_SLAVE yordamı alt bölümündeki HR_Start_Offset ve Extended_Addressing değişkenlerinin açıklamasında bulabilirsiniz. (Sayfa 1093)

Koşul kodları

Tablo 13-156 MB_MASTER'ın yürütülmesi için durum kodları (iletişim ve konfigürasyon hataları) [1]

STATUS (W#16#)	Açıklama
0000	Hata yok
80C8	Slave zaman aşımı. Slave'in Baud hızını, paritesini ve kablajını kontrol edin.
80D1	Alıcı cihaz devam etmekte olan bir aktarımı kesintiye uğratmak için akış kontrolü talep etmiş, ancak bekleme süresi içinde aktarımı yeniden etkinleştirmemiştir. Alıcı cihaz bekleme süresi içinde bir CTS algılamazsa, donanım akış kontrolü sırasında da bu hata üretilir.
80D2	DCE'den DSR sinyali alınmadığı için gönderim talebi iptal edildi.
80E0	Alım tamponu dolu olduğu için mesaj sonlandırıldı.
80E1	Mesaj, bir parite hatası nedeniyle sonlandırıldı.
80E2	Mesaj, bir telegram hatası nedeniyle sonlandırıldı.
80E3	Mesaj, bir taşma hatası nedeniyle sonlandırıldı.
80E4	Belirtilen uzunluk toplam tampon boyutunu aştığı için mesaj sonlandırıldı.
8180	Port ID için geçersiz değer veya MB_COMM_LOAD yordamında hata
8186	Geçersiz Modbus istasyon adresi
8188	Broadcast talebi için geçersiz mod belirtildi
8189	Veri adresi için geçersiz değer.
818A	Veri uzunluğu için geçersiz değer.
818B	Yerel veri kaynağına veya yerel veri hedefi üzerinde geçersiz işaretçi: Boyut, doğru değil.
818C	DATA_PTR veya Blocked_Proc_Timeout için geçersiz işaretçi: Veri alanı, bir DB (sembolik ve direkt adreslemeye izin veren) veya markörlerin bellek alanı olmalıdır.
8200	Port, bir gönderme talebinin işlenmesiyle meşgul.

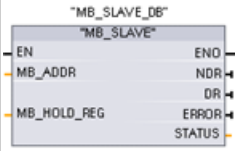
Tablo 13-157 MB_MASTER'ın yürütülmesi için durum kodları (Modbus protokol hataları) [1]

STATUS (W#16#)	Slave'in yanıt kodu	Modbus protokol hatası
8380	-	CRC hatası
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluğunda hata
8383	02	Veri adresi hatası veya adres, DATA_PTR'nin geçerli aralığın dışında
8384	03'ten büyük	Veri değerinde hata
8385	03	Veri teşhis kodunun değeri desteklenmiyor (fonksiyon kodu 08)
8386	-	Yanıttaki fonksiyon kodu, talepteki koda karşılık gelmiyor.
8387	-	Yanlış Slave, tepki vermiş.
8388	-	Slave'in bir yazma talebine verdiği yanıt yanlış. Slave tarafından yayınlanan yazma talebi Master'ın gönderdiğine karşılık gelmiyor.

[1] Yukarıda belirtilen MB_MASTER yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

13.9.2.3 MB_SLAVE (Modbus RTU Slave olarak PtP portu üzerinden iletişim)

Tablo 13-158 MB_SLAVE yordamı

LAD/FBD	SCL	Açıklama
	**"MB_SLAVE_DB"(MB_ADDR:=_uint_in_, NDR=>_bool_out_, DR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MB_HOLD_REG:=_variant_inout_);**	MB_SLAVE yordamı ile programınız bir CM'nin (RS485 veya RS232) veya bir CB'nin (RS485) PtP portu üzerinden Modbus Slave olarak iletişim kurabilir. Bir desantral Modbus RTU Master bir talepte bulunduğunda, kullanıcı programınız, MB_SLAVE yordamını çalıştırarak talebe yanıt verir. Yordamı eklerseniz, STEP 7 örnek DB'yi otomatik olarak oluşturur. Bu MB_SLAVE_DB adını, MB_COMM_LOAD yordamı için MB_DB parametresini belirtmek için kullanırsınız.

Tablo 13-159 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipi	Açıklama
MB_ADDR	IN	V1.0: USInt V2.0: UInt	Modbus Slave'in istasyon adresi: Standart adresleme aralığı (1 - 247) Genişletilmiş adresleme aralığı (0 - 65535)
MB_HOLD_REG	IN	Variant	İşaretçi, Modbus tutma kaydı DB'si üzerinde: Modbus tutma kaydı, markörlerin bellek alanı veya bir veri bloğu olabilir.
NDR	OUT	Bool	Yeni veriler hazır: • 0 – Yeni veriler yok • 1 - Modbus Master tarafından yeni veri yazıldığını gösterir.
DR	OUT	Bool	Veriler oku: • 0 – Veri okunmadı • 1 - Modbus Master tarafından yeni veri okunduğunu gösterir.
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca DOĞRU'dur. Yürütme bir hata ile sona ererse, STATUS parametresindeki hata kodu, yalnızca ERROR = DOĞRU olan bir çevrimde geçerlidir.
STATUS	OUT	Word	Yürütmenin hata kodu

Modbus iletişimi fonksiyon kodları (1, 2, 4, 5 ve 15) doğrudan girişlerin proses görünümündeki ve CPU'nun çıkışlarının proses görünümündeki Bit'leri ve sözcükleri okuyabilir ve yazabilir. Bu fonksiyon kodlarında MB_HOLD_REG parametresi bir Bayt'tan büyük bir veri tipi olarak tanımlanmalıdır. Aşağıdaki tabloda, Modbus adreslerinin CPU'daki proses görünümüne örnek olarak atanmasını gösterilmektedir.

Tablo 13-160 Modbus adreslerinin proses görünümüne atanması

Modbus fonksiyonları						S7-1200	
Kodlar	Fonksiyon	Veri alanı	Adres aralığı			Veri alanı	CPU adresi
01	Bit'leri oku	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
02	Bit'leri oku	Giriş	10001	-	18192	Girişlerin proses görünümü	E0.0 ila E1023.7
04	Sözcükleri oku	Giriş	30001	-	30512	Girişlerin proses görünümü	EW0 ila EW1022
05	Bit yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7
15	Bit'leri yaz	Çıkış	1	-	8192	Çıkışların proses görünümü	A0.0 ila A1023.7

Modbus iletişiminin fonksiyon kodları (3, 6, 16), markörlerin bellek alanındaki bir adres aralığı veya bir veri bloğu olabilen bir Modbus tutma kaydı kullanır. Tutma kaydı türü, MB_SLAVE yordamının MB_HOLD_REG parametresi tarafından belirlenir.

NOT

MB_HOLD_REG veri bloğu tipi

Modbus tutma kaydına sahip bir veri bloğu doğrudan (mutlak) ve sembolik adreslemeye izin vermelidir. Veri bloğunu oluştururken "Standart" erişim özniteliğini seçmelisiniz.

Aşağıdaki tabloda, Modbus adreslerinin Modbus fonksiyon kodları 03 (sözcük oku), 06 (sözcük yaz) ve 16 (sözcük yaz) için kullanılan tutma kaydına nasıl eşleneceğine ilişkin örnekler gösterilmektedir. DB adreslerinin gerçek üst sınırı, ilgili CPU modelinin maksimum çalışma belleği ve maksimum M belleği tarafından belirlenir.

Tablo 13-161 Modbus adreslerinin CPU belleğine atanması

Modbus Master adresi	**MB_HOLD_REG parametrelerine örnekler**				
	MW100	**DB10.DBW0**	**MW120**	**DB10.DBW50**	**"Reçete".Bileşim maddesi**
40001	MW100	DB10.DBW0	MW120	DB10.DBW50	"Reçete".Bileşim maddesi[1]
40002	MW102	DB10.DBW2	MW122	DB10.DBW52	"Reçete".Bileşim maddesi[2]
40003	MW104	DB10.DBW4	MW124	DB10.DBW54	"Reçete".Bileşim maddesi[3]
40004	MW106	DB10.DBW6	MW126	DB10.DBW56	"Reçete".Bileşim maddesi[4]
40005	MW108	DB10.DBW8	MW128	DB10.DBW58	"Reçete".Bileşim maddesi[5]

Tablo 13-162 Teşhis fonksiyonları

S7-1200 MB_SLAVE'inin Modbus teşhis fonksiyonları		
Kodlar	Kısmi fonksiyon	Açıklama
08	0000H	Yankı testi sorgu verilerini yayınlar: MB_SLAVE yordamı, alınan bir veri sözcüğünün yankısını bir Modbus Master'a geri döndürür.
08	000AH	İletişim olayı sayacını sil: MB_SLAVE yordamı, Modbus fonksiyonu 11 için kullanılan iletişim olay sayacını siler.
11		İletişim olayı sayacını çağır: MB_SLAVE yordamı, Modbus Slave'e gönderilen başarılı Modbus okuma ve yazma taleplerinin sayısını saymak için dahili bir iletişim olay sayacı kullanır. Fonksiyon 8, fonksiyon 11 ve Broadcast talepleri için sayaç artırılmaz. Bunun dışında, iletişim hatalarıyla (örn. parite veya CRC hataları) sonuçlanan talepler için de sayaç artırılmaz.

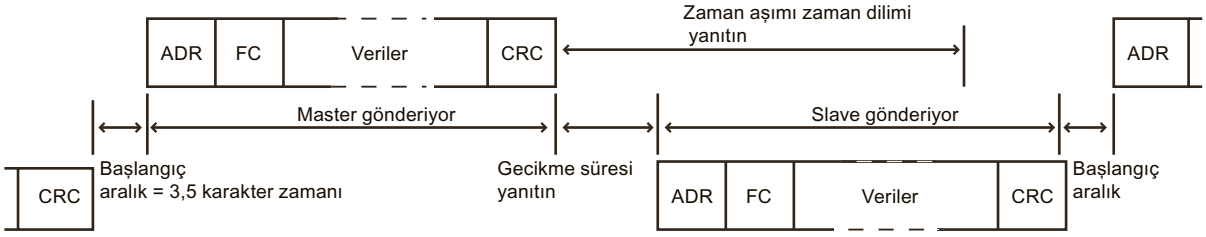
MB_SLAVE yordamı, talepler geçerli adreslere erişime yönelik olduğu sürece Modbus Master'larından gelen Broadcast yazma taleplerini destekler. Broadcast fonksiyonunda desteklenmeyen fonksiyon kodları için, MB_SLAVE, 0x8188 hata kodunu üretir.

Modbus Slave iletişimi kuralları

- MB_SLAVE yordamının bu portla iletişim kurabilmesi için, bir portu yapılandırmak üzere MB_COMM_LOAD yordamının yürütülmesi gerekir.
- Bir portun Slave olarak bir Modbus Master'a tepki vermesi isteniyorsa, bu port MB_MASTER yordamı ile programlanmamalıdır.
- Yalnızca bir MB_SLAVE örneği belirli bir port ile kullanılabilir, yoksa beklenmedik bir davranış ortaya çıkabilir.
- Modbus yordamları, iletişim sürecine kumanda etmek için iletişim alarm olaylarını kullanmaz. Programınız, tamamlanan gönderme ve alma işlemleri için MB_SLAVE yordamını sorgulayarak iletişim prosesine kumanda etmelidir.
- MB_SLAVE yordamı, bir Modbus Master'dan gelen taleplere zamanında yanıt verilmesini sağlayacak sıklıkta düzenli olarak yürütülmelidir. Bir program çevrimi OB'sinden MB_SLAVE'i her çevrimi yürütmenizi öneririz. MB_SLAVE bir uyandırma alarmı OB'sinden yürütülebilir, ancak kesme programındaki zaman gecikmeleri aşırıya kaçarsa, diğer kesme programlarının yürütülmesini geçici olarak engelleyebilir, dolayısıyla bu yöntem önerilmez.

Modbus sinyalinin zaman kumandası

Modbus Master'dan her talebi almak ve buna yanıt vermek için, MB_SLAVE düzenli olarak yürütülmelidir. MB_SLAVE'in yürütülme sıklığı, Modbus Master tarafından belirtilen yanıt zaman aşımı değerine göre değişir. Bu, aşağıdaki görselde sunulmaktadır.



RESP_TO yanıtı için zaman aşımı süresi, bir Modbus Master'ın bir Modbus Slave'den yanıtın başlamasını beklediği süredir. Bu zaman dilimi Modbus protokolü tarafından değil, ilgili Modbus Master'ın bir parametresince belirlenir. MB_SLAVE'in yürütme sıklığı (bir yürütme ile bir sonraki yürütme arasında geçen süre) Modbus Master'ın ilgili parametrelerine bağlı olmalıdır. Modbus Master'ın yanıtı için zaman aşımı süresi boyunca MB_SLAVE'i en az iki kez yürütmelisiniz.

Modbus Slave değişkenleri

Bu tabloda, programınızda kullanılabilen MB_SLAVE örnek veri bloğundaki genel statik değişkenler gösterilmektedir.

Tablo 13-163 Modbus Slave değişkenleri

Değişken	Veri tipi	Açıklama
Request_Count	Word	Slave tarafından alınan tüm taleplerin sayısı
Slave_Message_Count	Word	Belirli bir Slave için alınan taleplerin sayısı
Bad_CRC_Count	Word	CRC hatası içeren alınan taleplerin sayısı
Broadcast_Count	Word	Alınan Broadcast taleplerinin sayısı
Exception_Count	Word	İade edilen bir istisna gerektiren Modbus'a özgü hatalar geri döndürüldü
Success_Count	Word	Belirli bir Slave için alınan protokol hatasız taleplerin sayısı
HR_Start_Offset	Word	Modbus tutma kaydının başlangıç adresini belirtir (standart = 0)
Extended_Addressing	Bool	Slave adreslemesini tek veya çift Bayt olarak yapılandırır. (0 = tek Bayt'lık adres, 1 = çift Bayt'lık adres, standart = 0)

Programınız, HR_Start_Offset ve Extended_Addressing değişkenlerine değerler yazabilir ve Modbus Slave işlemlerine kumanda edebilir. Diğer değişkenler ise Modbus durumunu izlemek için okunabilir.

HR_Start_Offset

Modbus tutma kaydı adresleri, 40001 veya 400001'den başlar. Bu adresler, hedef sistem belleğindeki tutma kaydının başlangıç adresine karşılık gelir. Ancak Modbus tutma kaydı için 40001 veya 400001 dışında bir başlangıç adresi yapılandırmak isterseniz, HR_Start_Offset değişkenini yapılandırabilirsiniz.

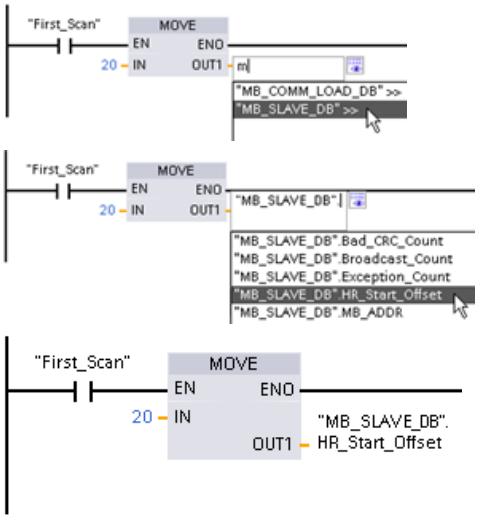
Örneğin, MW100'de başlayan ve 100 sözcük uzunluğunda olan bir tutma kaydı yapılandırabilirsiniz. Tutma kaydı başlangıç adresini, 20'lik bir ofset ile 40001 yerine 40021 olarak belirtirsiniz. Bu durumda 40021'in altındaki ve 400119'un üzerindeki her adres, adresleme hatasına yol açacaktır.

Tablo 13-164 Modbus tutma kaydının adresleme örneği

HR_Start_Offset	Adres	Minimum	Maksimum
0	Modbus adresi (sözcük)	40001	40099
	S7-1200 adresi	MW100	MW298
20	Modbus adresi (sözcük)	40021	40119
	S7-1200 adresi	MW100	MW298

HR_Start_Offset, Modbus tutma kaydının başlangıç adresini belirten bir sözcük değeridir ve MB_SLAVE örnek veri bloğunda saklanır. MB_SLAVE'i programınıza ekledikten sonra, bu açık statik değişkeni parametre açılır listesinden seçebilirsiniz.

Örneğin bir LAD ağına MB_SLAVE eklediyseniz, bir önceki bir ağa gidip, HR_Start_Offset değerini atayabilirsiniz. Değer, MB_SLAVE yürütülmeden önce atanmalıdır.



Modbus Slave değişkenini, standart DB adı ile girin.

1. İmleci parametre alanına getirin ve m karakterini girin.
2. Açılır listeden "MB_SLAVE_DB" öğesini seçin.
3. İmleci DB adının sağına getirin (tırnak işaretinden sonra) ve bir nokta girin.
4. Açılır listeden "MB_SLAVE_DB.HR_Start_Offset" öğesini seçin.

Extended_Addressing

Yukarıda açıklanan HR_Start_Offset referansına benzer bir şekilde Extended_Addressing değişkenine erişilir, aradaki fark, Extended_Addressing değişkeninin bir Boole değeri olmasıdır. Boole değeri bir çıkış bobininden yazılmalıdır ve bir "Kaydır" kutusundan yazılamaz.

Modbus Slave adreslemesi, tek Bayt (bu, Modbus standardıdır) veya çift Bayt olarak yapılandırılabilir. Genişletilmiş adresleme yöntemi, tek bir ağda 247'den fazla cihazı adreslemek için kullanılır. Genişletilmiş adreslemeyi tercih ederseniz, en fazla 64.000 adresi adresleyebilirsiniz. Aşağıda Modbus fonksiyon 1'in bir telegramı örnek olarak verilmektedir.

Tablo 13-165 Bir Bayt'lık Slave adresi (Bayt 0)

Fonksiyon 1	**Bayt 0**	**Bayt 1**	**Bayt 2**	**Bayt 3**	**Bayt 4**	**Bayt 5**	
Talep	Slave adresi	F kodu	Başlangıç adresi		Bobinlerin uzunluğu		
Geçerli yanıt	Slave adresi	F kodu	Uzunluk	Bobin verileri			
Hata yanıtı	Slave adresi	0x81	E kodu				

Tablo 13-166 İki Bayt'lık Slave adresi (Bayt 0 veya Bayt 1)

	Bayt 0	**Bayt 1**	**Bayt 2**	**Bayt 3**	**Bayt 4**	**Bayt 5**	**Bayt 6**
Talep	Slave adresi		F kodu	Başlangıç adresi		Bobinlerin uzunluğu	
Geçerli yanıt	Slave adresi		F kodu	Uzunluk	Bobin verileri		
Hata yanıtı	Slave adresi		0x81	E kodu			

Koşul kodları

Tablo 13-167 MB_SLAVE'in yürütülmesi için durum kodları (iletişim ve konfigürasyon hataları) [1]

STATUS (W#16#)	Açıklama
80D1	Alıcı cihaz devam etmekte olan bir aktarımı kesintiye uğratmak için akış kontrolü talep etmiş, ancak bekleme süresi içinde aktarımı yeniden etkinleştirmemiştir. Alıcı cihaz bekleme süresi içinde bir CTS algılamazsa, donanım akış kontrolü sırasında da bu hata üretilir.
80D2	DCE'den DSR sinyali alınmadığı için gönderim talebi iptal edildi.
80E0	Alım tamponu dolu olduğu için mesaj sonlandırıldı.
80E1	Mesaj, bir parite hatası nedeniyle sonlandırıldı.
80E2	Mesaj, bir telegram hatası nedeniyle sonlandırıldı.
80E3	Mesaj, bir taşma hatası nedeniyle sonlandırıldı.
80E4	Belirtilen uzunluk toplam tampon boyutunu aştığı için mesaj sonlandırıldı.
8180	Port ID için geçersiz değer veya MB_COMM_LOAD yordamında hata
8186	Geçersiz Modbus istasyon adresi
8187	MB_HOLD_REG DB'si üzerinde geçersiz işaretçi: Aralık, çok küçük.
818C	M bellek veya DB üzerinde geçersiz işaretçi MB_HOLD_REG (DB aralığı sembolik ve doğrudan adreslere izin vermelidir)

Tablo 13-168 MB_SLAVE'in yürütülmesi için durum kodları (Modbus protokol hataları) [1]

STATUS (W#16#)	Slave'in yanıt kodu	Modbus protokol hatası
8380	Yanıt yok	CRC hatası
8381	01	Fonksiyon kodu desteklenmiyor veya Broadcast'larda desteklenmiyor
8382	03	Veri uzunluğunda hata
8383	02	Veri adresi hatası veya adres, DATA_PTR'nin geçerli aralığın dışında
8384	03	Veri değerinde hata
8385	03	Veri teşhis kodunun değeri desteklenmiyor (fonksiyon kodu 08)

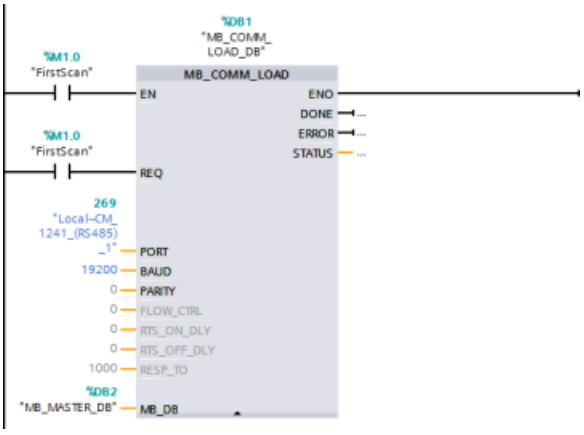
[1] Yukarıda belirtilen MB_SLAVE yordamlarının hatalarına ek olarak, temeldeki PtP iletişim yordamları da hata geri döndürebilir.

13.9.3 Eski sistemlerdeki Modbus RTU ile ilgili örnek

13.9.3.1 Örnek: Daha eski sistemlerdeki bir Modbus RTU-Master ile ilgili örnek program

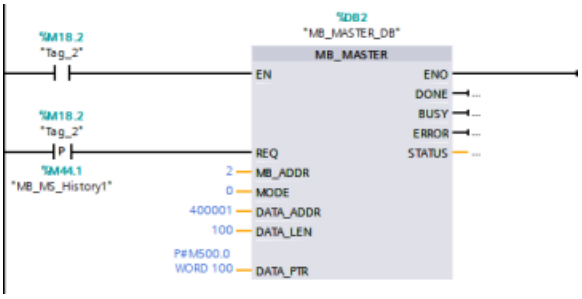
MB_COMM_LOAD, yol alma sırasında ilk çevrimin markörü üzerinden başlatılır. MB_COMM_LOAD sadece, seri portun konfigürasyonu çalışma zamanı sırasında değişmezse bu şekilde yürütülmelidir.

Ağ 1: RS485 modülünün iletişim portunu ilk çevrim sırasında yalnızca bir kez yapılandırın/başlatın.

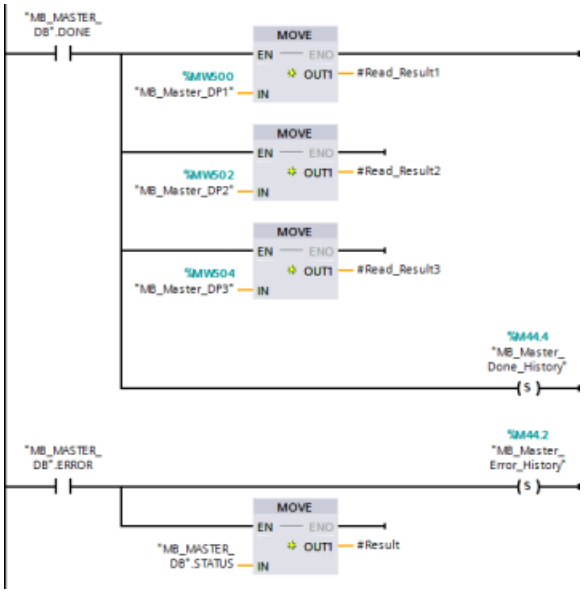


Bir MB_MASTER yordamını, bir Slave ile iletişim kurmak için OB program çevriminde kullanırsınız. Diğer MB_MASTER yordamları da OB program çevriminde diğer Slave'lerle iletişim kurmak veya bir MB_MASTER FB'si başka Slave'lerle iletişim için kullanılabilir.

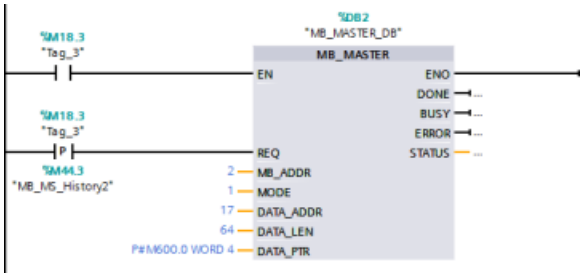
Ağ 2: MW500-MW698 adresindeki Slave No. 2'deki 400001 adresinden 100 sözcüklük tutma kaydı verisini okuyun.



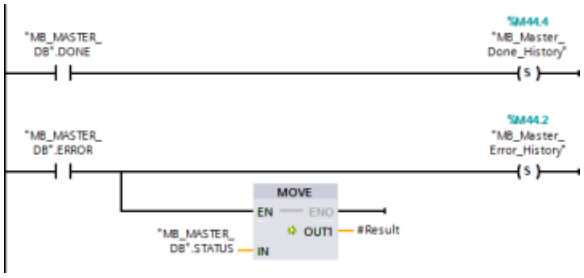
Ağ 3: Okunan tutma kaydı verilerinin ilk üç sözcüğünü başka bir adrese kaydırın ve bir DONE yapıldı Bit'i ayarlayın. Bu ağ ayrıca, bir hata oluştuğunda ERROR yapıldı Bit'ini ayarlar ve STATUS sözcüğünü başka bir adreste saklar.



Ağ 4: Slave No. 2'de MW600-MW607'den 64 Bit'lik veriyi 00017 ila 00081 çıkış Bit'inin adreslerine yazın.



Ağ 5: Yazma işlemi bittiğinde yapıldı Bit'ini DONE olarak ayarlayın. Bir hata oluşursa, program, ERROR yapıldı Bit'ini ayarlar ve STATUS kodunu saklar.

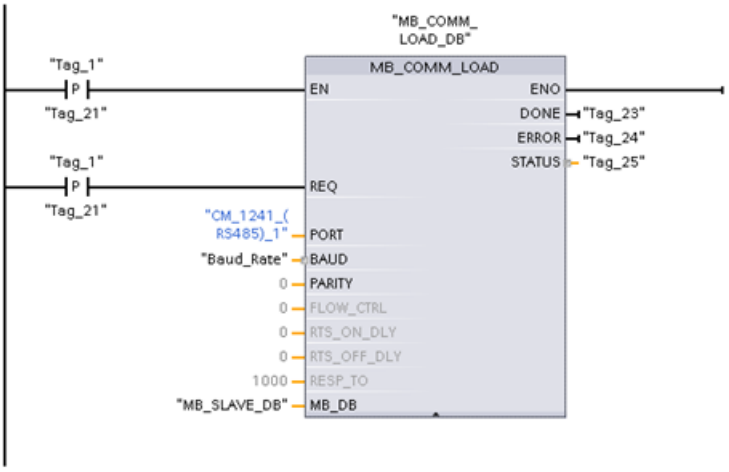


13.9.3.2 Örnek: Daha eski sistemlerdeki bir Modbus RTU Slave için örnek program

"Değişken_1"in her etkinleştirildiğinde, aşağıda gösterilen MB_COMM_LOAD yordamı başlatılır.

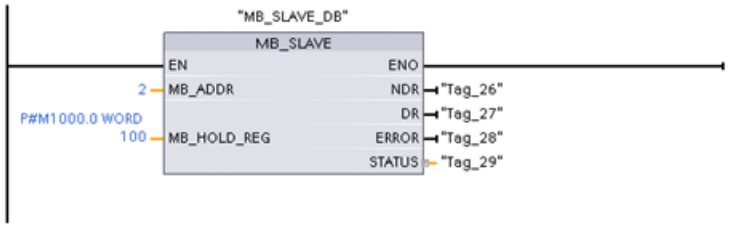
MB_COMM_LOAD yordamının bu şekilde yürütülmesi yalnızca, çalışma zamanı sırasında HMI konfigürasyonunun bir sonucu olarak seri port konfigürasyonu değişirse mümkündür.

Ağ 1: Bir HMI cihazı tarafından her değiştirildiklerinde RS485 modül parametrelerini başlatın.



Aşağıda gösterilen MB_SLAVE yordamı, her 10 msn'de bir yürütülen bir periyodik OB içindedir. Bu, Slave tarafından mutlak en hızlı yanıtı sağlamasa da, kısa mesajlar için 9600 Baud hızında iyi bir işletim performansı sergiler (talepte 20 Bayt'a kadar).

Ağ 2: Her çevrimde Modbus Master'dan gelen talepleri kontrol edin. Modbus tutma kaydı için, MW1000'den başlamak üzere 100 sözcük yapılandırılır.



13.10 Industrial Remote Communication (IRC)

13.10.1 Telecontrol CP'lerine genel bakış

Industrial Remote Communication, geniş şekilde dağıtılmış farklı boyutlardaki makinelere, sistemlere ve uygulamalara güvenli ve ekonomik erişim olanağı sunar. Industrial Remote Communication, CP modülleri üzerinden şu iletişim türlerini kapsar:

- TeleControl: Telecontrol, büyük bir coğrafi bölgeye dağıtılmış olan proses istasyonlarının (Remote Terminal Unit'ler/RTU'lar), kullanım ve izleme amacıyla bir veya daha fazla proses kontrol sistemiyle gerçekleştirilen bağlantısıdır. Uzak ağların ürün yelpazesindeki çok sayıda farklı aktarma bileşeni, açık ve özel ağlarda bir bölge boyunca uzaktan iletişimi desteklemektedir. Özel Telecontrol protokolleri, proses verilerinin olay kontrollü veya periyodik alışverişini gerçekleştirir, bu da tüm prosese etkin şekilde kumanda etmeyi sağlar.
- TeleService: Teleservice; hata algılama, teşhis, bakım, onarım veya optimizasyon amacıyla uzak teknik sistemlerle (makineler, sistemler, bilgisayarlar, vb.) veri alışverişini kapsar.
- Uzaktan iletişim için ek uygulamalar, örn. denetim, Smart-Grid uygulamaları ve makine denetimi.

S71200 için TeleControl CP'leri

TeleControl uygulamaları için şu iletişim işlemcileri mevcuttur, ayrıca bunların birçoğu S7-1200 Web sunucusuna (Sayfa 805) erişim de sağlar:

- **CP 1243-1:**
  - Sipariş numarası: 6GK7 243-1BX30-0XE0
  - SIMATIC S7-1200'ü Industrial Ethernet'e veya internet üzerinden şu kontrol sistemlerine bağlamak için kullanılan iletişim işlemcisi:
    - Telecontrol Server (OPC sunucu uygulaması TCSB V3)
    - DNP3 Masterstation
    - IEC 60870-5 Masterstation
  - CP, VPN teknolojisi ve güvenlik duvarının yardımıyla S7-1200'e korumalı erişim olanağı sağlar.
  - CP, CPU'nun arayüz genişletmesi olarak kullanılabilir. Bu rolde, ağ ayırma amacıyla kullanılır.
  - CP ile CPU arasındaki iletişim, PLC değişkenlerine erişen yapılandırılabilir veri noktaları üzerinden gerçekleştirilir.
- **CP 1243-7 LTE-xx:**
  - SIMATIC S7-1200'ü bir kablosuz ağ üzerinden şu kontrol sistemlerine bağlamak için kullanılan iletişim işlemcisi:
    - Telecontrol Server (OPC sunucu uygulaması TCSB V3)
    - DNP3 Masterstation
    - IEC 60870-5 Masterstation
  - SIMATIC S7-1200'ü bir kablosuz ağ üzerinden şu kontrol sistemlerine bağlamak için kullanılan iletişim işlemcisi:
  - Aşağıdaki mobil iletişim spesifikasyonlarının desteklenmesi: GSM/GPRS, UMTS (3G), LTE

  - CP, farklı mobil iletişim spesifikasyonlarına sahip ülkelerde de kullanılabilmesi için iki varyant şekilde kullanıma sunulmuştur:
    - CP 1243-7 LTE-US:
    - Kuzey Amerika Standardı
    - Sipariş numarası: 6GK7 243-7SX30-0XE0
    - CP 1243-7 LTE-EU:
    - Batı Avrupa Standardı
    - Sipariş numarası: 6GK7 243-7KX30-0XE0
  - CP, VPN teknolojisi ve güvenlik duvarının yardımıyla S7-1200'e korumalı erişim olanağı sağlar.
  - CP, S7 iletişimi için CPU'nun ek Ethernet arayüzü olarak kullanılabilir.
  - CP ile CPU arasındaki iletişim, PLC değişkenlerine erişen yapılandırılabilir veri noktaları üzerinden gerçekleştirilir.
- **CP 1243-8 IRC:**
  - Sipariş numarası: 6GK7 242-8RX30-0XE0
  - SIMATIC S7-1200'ün bir ST7 ağına, bir veri noktası konfigürasyonuna ve bir VPN'e bağlanması için kullanılan iletişim işlemcisi

NOT

CP 1243-1 dışındaki diğer CP'lere yönelik TeleControl uygulamaları için TeleControl Server Basic yazılımına ihtiyacınız bulunmaktadır.

Güvenli iletişim

Kendini kanıtlamış SINAUT ST7 protokolü veya standartlaştırılmış DNP3 protokolü veya IEC 60870-5, Industrial Remote Communication'da güvenliği sağlar (https://new.siemens.com/global/en/products/automation/industrial-communication/industrial-remote-communication.html). TeleControl çözümü, veri sahtekârlığını ve veri kaybını önlemek için kapsamlı önlemler sunar. Her aktarma modülü, binlerce veri telegramı için büyük bir belleğe sahiptir, bu sayede aktarım bağlantısındaki devre dışı kalma süreleri önlenebilir. Özel VPN içeren çözümler, özel IP tabanlı ağları korur.

CP 1243-1 iletişim işlemcisi, SIMATIC S7-1200 kumanda ünitesini Ethernet ağlarına güvenli bir şekilde bağlar. İletişim işlemcisi, entegre güvenlik duvarı (Stateful Inspection) ve VPN protokolünün (IPsec) güvenlik fonksiyonları ile S7-1200 istasyonlarının ve alt seviyedeki ağların yetkisiz kişilerin erişimine karşı korunmasını destekler ve şifreleme aracılığıyla veri aktarımını manipülasyona ve casusluğa karşı korumaya yardımcı olur. Ayrıca CP, S7-1200 istasyonunun TeleControl Server Basic kontrol sistemi yazılımına IP tabanlı uzak ağlar aracılığıyla entegrasyonu için de kullanılabilir.

13.10.2 Daha fazla bilgi

Aşağıdaki CP işletme kitapları; bağlantı, ön koşullar, uygulamalar, blok yordamları, aksesuarlar ve konfigürasyon örnekleriyle ilgili ayrıntılı bilgiler sunar:

- CP 1243-1 işletme kılavuzu (https://support.industry.siemens.com/cs/ww/en/view/103948898)
- CP 1243-7 LTE işletme kılavuzu (https://support.industry.siemens.com/cs/ww/en/view/109476704)
- CP 1243-8 IRC işletme kılavuzu (https://support.industry.siemens.com/cs/ww/en/view/109777054)

Bu S7-1200 SIMATIC Telecontrol CP modülleriyle ilgili daha fazla bilgiyi ve dokümantasyonu şu linklerde bulabilirsiniz:

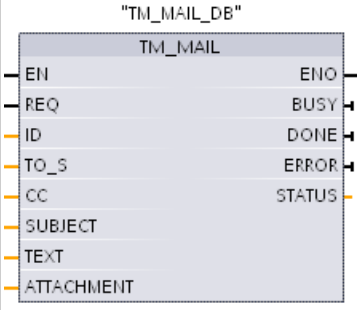
- CP 1243-1 (https://support.industry.siemens.com/cs/ww/en/ps/15922)
- CP 1243-7 LTE (https://support.industry.siemens.com/cs/ww/en/ps/15924)
- CP 1243-8 IRC (https://support.industry.siemens.com/cs/ww/en/ps/21162)
- Kullanılabilirliğe göre Firmware güncellemeleri (https://support.industry.siemens.com/cs/ww/en/ps/21770/dl)

14

TeleService iletişimi (SMTP ile e-posta iletişimi)

14.1 TM_Mail (e-posta gönder) yordamı

Tablo 14-1 TM_MAIL yordamı

LAD/FBD	SCL	Açıklama
	**"TM_MAIL_DB"(** **REQ:=_bool_in_,** **ID:=_int_in_,** **TO_S:=_string_in_,** **CC:=_string_in_,** **SUBJECT:=_string_in_,** **TEXT:= _string_in_,** **ATTACHMENT:=_variant_in_,** **BUSY=>_bool_out_,** **DONE=>_bool_out_,** **ERROR=>_bool_out_,** **STATUS=>_word_out_,);**	TM_MAIL yordamı, SMTP (Simple Mail Transfer Protocol) aracılığıyla, TCP/IP üzerinden, CPU'nun Industrial Ethernet bağlantısı ile bir e-posta gönderir. Ethernet tabanlı internet bağlantısı olmayan yerlerde, sabit telefon şebekesi hatlarıyla bağlantı için opsiyonel bir Teleservice adaptörü kullanılabilir. TM_MAIL asenkron olarak yürütülür ve görev, birden çok TM_MAIL çağrısını kapsar. TM_MAIL'i çağırırsanız bir örnek DB'si atamanız gerekir. **Örnek DB'si kalıcı olarak tanımlanmamış olmalıdır**. Bu sayede, CPU, STOP işletim durumundan RUN işletim durumuna geçerken örnek DB'sinin başlatılması ve TM_MAIL'in yeniden yürütülmesinin tetiklenebilmesi sağlanır.

[1] STEP 7, yordamı eklediğiniz takdirde örnek DB'sini otomatik olarak oluşturur.

Bir e-posta gönderme işlemini, REQ giriş parametresinde 0'dan 1'e giden bir pozitif kenarla başlatabilirsiniz. Aşağıdaki tabloda, BUSY, DONE ile ERROR arasındaki ilişkiler gösterilmektedir. Bu parametreleri sonraki üst üste çağrılarda değerlendirerek TM_MAIL'in yürütülmesinin ilerleyişini denetleyebilir ve tamamlanmasını algılayabilirsiniz.

BUSY çıkış parametresinin durumu 1 iken 0 olarak değişirse, DONE, ERROR, STATUS ve SFC_STATUS çıkış parametreleri sadece birer çevrim için geçerli olur. Sonraki program yürütme çevrimlerinde durum değişikliklerini tanımlayabilmeniz için, program lojiğiniz geçici çıkış durum değerlerini saklamalıdır.

NOT

TM_MAIL, TCP/IP üzerinden, CPU'nun Ethernet arayüzünü kullanarak bir posta gönderir. Bir postayı bir CP arayüzü üzerinden (SSL ile veya olmadan) göndermek için TMAIL_C (e-postayı CPU'nun Ethernet arayüzü üzerinden gönder) yordamı (Sayfa 644) yordamını kullanın.

Tablo 14-2 Done, Busy ve Error parametreleri arasındaki etkileşim

DONE	BUSY	ERROR	Açıklama
İlgili değil	1	İlgili değil	Görev düzenleniyor.
1	0	0	Görev, başarıyla gerçekleştirildi.
0	0	1	Görev hatayla sonlandı. Hatanın sebebini STATUS parametresinde bulabilirsiniz.
0	0	0	Düzenlenen görev yok.

CPU, TM_MAIL yürütülürken STOP işletim durumuna ayarlanırsa, e-posta sunucusuyla kurulan iletişim bağlantısı sonlandırılır. E-posta sunucusuyla kurulan iletişim bağlantısı, CPU'nun Industrial Ethernet veriyolu üzerindeki iletişimi sırasında sorunlar oluşursa da kaybolur. Bu tür durumlarda gönderme işlemi kesilir ve e-posta alıcıya ulaşamaz.

DİKKAT

Kullanıcı programının değiştirilmesi

Program bloklarını silme ve değiştirme işlemleri, TM_MAIL çağrıları veya TM_MAIL örnek DB'lerinin çağrıları, program bloklarının bağlantısını kesebilir. Bağlı program bloklarını kaydetmezseniz, TPC/IP iletişim fonksiyonları, maddi hasarlara yol açabilen tanımsız bir duruma gelebilir. Değiştirilen bir program bloğu aktarıldıktan sonra bir CPU yeniden başlatma (sıcak başlatma) veya bir soğuk başlatma işlemi gerçekleştirmeniz gerekir.

Program bloklarının bağlantısını kesmemek için, kullanıcı programının doğrudan TM_MAIL çağrılarını etkileyen kısımlarını sadece aşağıdaki durumlarda değiştirin:

- CPU STOP işletim durumunda ise
- E-posta gönderilemiyorsa (REQ ve BUSY = 0)

Veri tutarlılığı

İşlem başlatılırsa ADDR_MAIL_SERVER giriş parametresi okunur. Yeni bir değer, ancak o anki işlem tamamlandıysa ve yeni bir TM_MAIL yordamı başlatılırsa etkili olur.

Bunun tersine, WATCH_DOG_TIME, TO_S, CC, FROM, SUBJECT, TEXT, ATTACHMENT, USERNAME ve PASSWORD parametreleri, TM_MAIL'in yürütülmesi sırasında okunur ve ancak görev tamamlandıysa değiştirilebilir (BUSY = 0)

Çevirmeli bağlantı: TS adaptörünün IE parametrelerinin yapılandırılması

İnternet servis sağlayıcınızın sunucusuyla bir çevirmeli bağlantı kurmak için, Teleservice adaptörünün IE parametrelerini, giden çağrılar için yapılandırmanız gerekir. Çağrı için kullanılan özniteliği talebe ayarlarsanız, bağlantı sadece bir e-posta gönderildiğinde kurulur. Analog bir modem bağlantısında, bağlantı işlemi için daha fazla zaman gerekir (yaklaşık fazladan bir dakika). Bu fazlalık süreyi WATCH_DOG_TIME değerine dahil etmeniz gerekir.

Tablo 14-3 Parametreler için kullanılan veri tipleri

Parametre ve veri tipi		Veri tipleri	Açıklama
REQ	IN	Bool	Yordam, bir yükselen kenar (0'dan 1'e) aracılığıyla başlatılır.
ID	IN	Int	Bağlantı kodu: TCON, TDISCON, TSEND ve TRCV yordamlarının ID parametrelerine bakın. Kullanıcı programında bu yordamın diğer örnekleri için kullanılmayan bir sayı kullanılmalıdır.
TO_S	IN	String	Alıcı adresleri: Maksimum 240 karakterlik bir uzunluğa sahip STRING verileri.
CC	IN	String	CC kopyası (opsiyonel) için kullanılan alıcı adresleri: Maksimum 240 karakterlik bir uzunluğa sahip STRING verileri.
SUBJECT	IN	String	E-postanın konusu: Maksimum 240 karakterlik bir uzunluğa sahip STRING verileri.
TEXT	IN	String	E-postanın metni (opsiyonel): Maksimum 240 karakterlik (Bayt) bir uzunluğa sahip STRING. Bu parametre bir boş karakter dizisi içeriyorsa, e-posta, metin mesajı olmadan gönderilir.
ATTACHMENT	IN	Variant	İşaretçi, e-posta ekinin üzerinde 65.534 Bayt'lık bir maksimum uzunluğa sahip sözcük veya çift sözcük verileri. Hiçbir değer atanmazsa, e-posta ek olmadan gönderilir.
DONE	OUT	Bool	• 0 - Görev henüz başlatılmadı veya halen gerçekleştiriliyor. • 1 - Görev hatasız bir şekilde yürütüldü.
BUSY	OUT	Bool	• 0 - Devam eden bir işlem yok • 1 - İşlem işleniyor
ERROR	OUT	Bool	ERROR Bit'i, kendisinden sonra son talebin hatayla sonlandığı bir çevrim boyunca = 1'dir. STATUS çıkışındaki hata kodu, sadece ERROR = 1 olan bir çevrimde geçerlidir.
STATUS	OUT	Word	TM_MAIL yordamının geri dönüş değeri veya hata bilgileri.
ADDR_MAIL_SERVER	[1] Static	DWord	Posta sunucusunun IP adresi: IP adresinin her fragmentini iki 4-Bit onaltılık karakterden oluşan sekizli olarak atamanız gerekir. IP adresinin fragmenti = onaltılık değer A'ya karşılık gelen ondalık değer 10 ise, bu sekizli için "0A" ifadesini girmeniz gerekir. Örnek: IP adresi = 192.168.0.10 ADDR_MAIL_SERVER = DW#16#C0A8000A, burada aşağıdakiler geçerlidir: • 192 = 16#C0, • 168 =16#A8 • 0 = 16#00 • 10 = 16#0A

[1] Bu parametrelerin değerleri her TM_MAIL çağrısında değiştirilmez. Değerler, TM_MAIL örnek veri bloğunda atanır ve sadece bir kez, TM_MAIL'in ilk çağrısında referanslanır.

Parametre ve veri tipi		Veri tipleri	Açıklama
WATCH_DOG_TIME	[1] Static	Time	SMTP ile bağlantının başlangıcından SMTP aktarımının sonuna kadarTM_MAIL'in tüm SMTP prosesini sonlandırması gereken maksimum süre. Bu süre aşılırsa, TM_MAIL'in yürütülmesi bir hatayla sona erer. TM_MAIL tamamlanana ve hata yayınlanana kadar gerçek zaman gecikmesi, bağlantının kesilmesi için ek süre gerekirken WATCH_DOG_TIME değerini aşabilir. Önce 2 dakikalık bir süre ayarlamanız gerekir. Bu süre, bir ISDN telefon bağlantısında çok daha kısa olabilir.
USERNAME	[1] Static	String	Posta hesabının kullanıcı adı: Maksimum 180 karakterlik bir uzunluğa sahip STRING verileri.
PASSWORD	[1] Static	String	Posta sunucusu için kullanılan şifre: Maksimum 180 karakterlik bir uzunluğa sahip STRING verileri.
FROM	[1] Static	String	Gönderenin adresi: Maksimum 240 karakterlik bir uzunluğa sahip STRING
SFC_STATUS	[1] Static	Word	Çağrılan iletişim bloklarının yürütülmesi için gereken koşul kodu

[1] Bu parametrelerin değerleri her TM_MAIL çağrısında değiştirilmez. Değerler, TM_MAIL örnek veri bloğunda atanır ve sadece bir kez, TM_MAIL'in ilk çağrısında referanslanır.

SMTP kimlik doğrulama

TM_MAIL, SMTP-AUTH-LOGIN kimlik doğrulama yöntemini destekler. Bu kimlik doğrulama yöntemiyle ilgili bilgileri, posta sunucusunun işletme kitabında veya internet servis sağlayıcınızın web sitesinde bulabilirsiniz.

AUTH LOGIN kimlik doğrulama yöntemi, posta sunucusuyla bağlantı kurmak için TM_MAIL yordamının USERNAME ve PASSWORD parametrelerini kullanır. Kullanıcı adı ve şifre, önceden bir e-posta sunucusundaki bir e-posta hesabında ayarlanmış olmalıdır.

USERNAME parametresine hiçbir değer atanmadıysa, AUTH LOGIN kimlik doğrulama yöntemi kullanılmaz ve e-posta, kimlik doğrulama işlemi gerçekleştirilmeden gönderilir.

TO_S:, CC: ve FROM: Parametre

TO_S:, CC: ve FROM: parametreleri, aşağıdaki örneklerde gösterildiği gibi karakter dizileridir:
TO: <wenna@mydomain.com>, <ruby@mydomain.com>,
CC: <admin@mydomain.com>, <judy@mydomain.com>,
FROM: <admin@mydomain.com>
Bu karakter dizileri girilirken şu kurallara uyulmalıdır:

- "TO:", "CC:" ve "FROM:" karakterleri, iki nokta üst üste işareti dahil olmak üzere girilmelidir.
- Her adresin önünde bir boşluk karakteri ve bir küçüktür işareti "<" bulunmalıdır. Örneğin "TO:" ile <e-posta adresi> arasında bir boşluk karakteri olmalıdır.
- Her adresin arkasına bir büyüktür işareti ">" girilmelidir.
- TO_S: ve CC: adreslerinde, her adresten sonra bir virgül işareti "," girilmelidir. Örneğin "TO: <email address>,"deki tek e-posta adresinden sonra virgül gereklidir.
- FROM: girişi için, sonda virgül olmadan sadece bir e-posta adresi belirtilmelidir.

Çalışma zamanı modu ve bellek kullanımı nedeniyle, TM_MAIL yordamının TO_S:, CC: ve FROM: verileri için sözdizimi kontrolü gerçekleştirilmez. Yukarıdaki format kurallarına tam olarak uyulmazsa, SMTP-E posta sunucusunun aktarımı başarısız olur.

STATUS ve SFC_STATUS parametreleri

Yürütme işleminin TM_MAIL tarafından geri döndürülen koşul kodları aşağıdaki şekilde sınıflandırılabilir:

- W#16#0000: TM_MAIL'in yürütülmesi başarıyla tamamlandı.
- W#16#7xxx: TM_MAIL'in yürütülme durumu.
- W#16#8xxx: Bir iletişim cihazının veya posta sunucusunun dahili bir çağrısında hata oluştu.

Aşağıdaki tabloda, dahili olarak çağrılan iletişim modüllerinin koşul kodları istisna olmak kaydıyla, TM_MAIL'i yürütme koşul kodları gösterilmektedir.

NOT

E-posta sunucusuna yönelik talepler

TM_MAIL sadece SMTP aracılığıyla Port 25 üzerinden bir e-posta sunucusuyla iletişim kurabilir. Atanan port numarası değiştirilemez.

Bu arada çoğu IT departmanı ve harici e-posta sunucusu, virüs bulaşmış bir PC'nin tehlikeli bir e-posta kaynağına dönüşmesini önlemek için Port 25'i bloke eder.

SMTP üzerinden dahili bir e-posta sunucusuyla bağlantı kurabilir ve dahili sunucunun, e-postaları internet üzerinden harici bir e-posta sunucusuna iletmek için gerekli olan güncel güvenlik eklentilerini yönetmesini sağlayabilirsiniz.

Örnek: Dahili bir e-posta sunucusunun konfigürasyonu

Microsoft Exchange'i dahili e-posta sunucusu olarak kullanıyorsanız, sunucuyu, S7-1200 hedef sisteme atanmış olan IP adresinin SMTP erişimine izin verilecek şekilde yapılandırabilirsiniz. Exchange Management konsolunu yapılandırın: Sunucu konfigürasyonu > Hub taşıma > Alıcı konektörü > IP rölesi. "Ağlar" sekmesinde "Şu IP adreslerine sahip uzak sunuculardan e-postalar alındı" alanı bulunur. Buraya, TM_MAIL yordamını yürüten PLC cihazının IP adresini girebilirsiniz. Bu bağlantı tipi için dahili bir Microsoft Exchange sunucusunda kimlik doğrulama gerekmez.

E posta sunucusunun konfigürasyonu

TM_MAIL, Port 25, SMTP ve AUTH-LOGIN kimlik doğrulama (opsiyonel) üzerinden iletişime izin veren sadece bir e-posta sunucusu kullanabilir.

Uzak SMTP oturum açma işlemini kabul etmek için e-posta sunucusunda uyumlu bir hesap kurun. Ardından TM_MAIL için kullanılan örnek DB'sini düzenleyin ve e-posta hesabınızla gerçekleştirilen bağlantının kimlik doğrulamasını yapmayı sağlayan TM_MAILUSERNAME ile PASSWORD karakter dizilerini girin.

Tablo 14-4 Koşul kodları

STATUS (W#16#...):	SFC_STATUS (W#16#...):	Açıklama
0000	-	TM_MAIL'in yürütülmesi hatasız bir şekilde tamamlandı. Bu STATUS kodu 0, gerçekten bir e-postanın gönderilmesini sağlamaz (bu tabloyu takip eden nottaki 1. maddeye bakın).
7001	-	TM_MAIL aktiftir (BUSY = 1).
7002	7002	TM_MAIL aktiftir (BUSY = 1).
8xxx	xxxx	TM_MAIL'in yürütülmesi, iletişim yordamının dahili çağrısında bir hatayla sonlandı. SFC_STATUS parametresiyle ilgili daha fazla bilgiyi, temel açık PROFINET kullanıcı iletişimi yordamlarının STATUS parametresinin açıklamalarında bulabilirsiniz.
8010	xxxx	Bağlantı kurulamadı: SFC_STATUS parametresiyle ilgili daha fazla bilgiyi, TCON yordamının STATUS parametresinin açıklamasında bulabilirsiniz.
8011	xxxx	Veriler gönderilirken hata: SFC_STATUS parametresiyle ilgili daha fazla bilgiyi, TSEND yordamının STATUS parametresinin açıklamasında bulabilirsiniz.
8012	xxxx	Veriler alınırken hata: SFC_STATUS parametresiyle ilgili daha fazla bilgiyi, TRCV yordamının STATUS parametresinin açıklamasında bulabilirsiniz.
8013	xxxx	Bağlantı kurulamadı: SFC_STATUS parametresinin değerlendirilmesiyle ilgili daha fazla bilgiyi, TCON ve TDISCON yordamlarının STATUS parametresinin açıklamalarında bulabilirsiniz.
8014	-	Bağlantı kurulamadı: Muhtemelen posta sunucusu için yanlış bir IP adresi (ADDR_MAIL_SERVER) veya bağlantı için çok kısa bir süre (WATCH_DOG_TIME) belirttiniz. CPU'nun ağla bağlantısının olmaması veya CPU konfigürasyonunun yanlış olması da mümkündür.
8015	-	ATTACHMENT parametresi için geçersiz işaretçi: Veri tipi ve uzunluk atamasına sahip bir varyant işaretçi kullanın. Örnek: "P#DB.DBX0.0" yanlıştır ve "P#DB.DBX0.0 Byte 256" doğrudur.

STATUS (W#16#...):	SFC_STATUS (W#16#...):	Açıklama
82xx, 84xx, 85xx	-	Hata mesajı, posta sunucusundan geliyor ve SMTP protokolüne ait hata numarası "8"e karşılık geliyor. Bu tabloyu takip eden nottaki 2. maddeye bakın.
8450	-	İşlem yürütülmüyor: Posta kutusu kullanılamıyor, işlemi daha sonra tekrarlayın.
8451	-	İşlem iptal edildi: Düzenleme sırasında yerel hata. İşlemi daha sonra tekrarlayın.
8500	-	Komut sözdiziminde hata: Sebep, e-posta sunucusunun LOGIN kimlik doğrulama işlemini desteklememesi olabilir. TM_MAIL'nin parametrelerini kontrol edin. Bir e-postayı kimlik doğrulaması olmadan göndermeyi deneyin. USERNAME parametresini boş bir karakter dizisiyle değiştirmeyi deneyin.
8501	-	Sözdizimi hatası: Yanlış parametre veya yanlış argüman; muhtemelen TO_S veya CC parametresine yanlış bir adres girdiniz.
8502	-	Komut bilinmiyor veya uygulanmadı: Başta FROM parametresi olmak üzere yaptığınız girişleri kontrol edin. Muhtemelen bu parametre eksik ve "@" veya "." karakterlerinden birini unuttunuz.
8535	-	SMTP kimlik doğrulama eksik. Muhtemelen yanlış bir kullanıcı adı veya yanlış bir şifre girdiniz.
8550	-	Posta sunucusuna erişilemiyor veya erişim yetkileriniz yok. Muhtemelen yanlış bir kullanıcı adı veya yanlış bir şifre girdiniz veya posta sunucunuz oturum açma yoluyla erişimi desteklemiyor. Bu hatanın başka bir sebebi, TO_S veya CC parametrelerine "@" karakterinden sonra alan adını yanlış girmek olabilir.
8552	-	İşlem iptal edildi: Atanan hafıza boyutu aşıldı; işlemi daha sonra tekrarlayın.
8554	-	Aktarma başarısız oldu: İşlemi daha sonra tekrarlayın.

NOT

Muhtemelen bildirilmeyen bir e-posta aktarım hatası mevcut

- Bir alıcı adresinin hatalı şekilde belirtilmesi, TM_MAIL'de STATUS hatası oluşturmaz. Bu gibi bir durumda, ek alıcıların (e-posta adresleri doğru olan alıcılar) e-postayı alması garanti edilemez.
- SMTP hata kodlarıyla ilgili daha fazla bilgiyi, internette veya posta sunucusunun hata dokümantasyonunda bulabilirsiniz. Posta sunucusunun son hata mesajını okumanız da mümkündür. Hata mesajı, TM_MAIL'in örnek DB'sine ait buffer1 parametresine kaydedilir.

15

Çevrimiçi araçlar ve teşhis araçları

15.1 Durum LED'leri

CPU ve G/Ç modülleri, modülün veya G/Ç'lerin işletim durumunu bildirmek için LED'lerden faydalanır.

Bir CPU'daki durum LED'leri

CPU şu durum göstergelerini sunar:

- STOP/RUN
  - Sarı renkli sürekli ışık STOP işletim durumunu gösterir
  - Yeşil renkli sürekli ışık, RUN işletim durumunu gösterir
  - Yanıp sönme (sırayla yeşil ve sarı), CPU'nun STARTUP işletim türünde olduğunu gösterir
- ERROR
  - Yanıp sönen bir LED, örn. bir dahili CPU hatası, memory card'daki bir hata veya bir konfigürasyon hatası (uygun olmayan modüller) gibi bir hatayı belirtir.
  - Üç saniye süreyle yanıp sönen kırmızı flaşör, artık mevcut olmayan bir hatayı gösterir. Bununla ilgili bir örnek, gerilim kesintisinde gerçek zaman saatinin (RTC) sıfırlanarak standart ayara döndürülmesidir.
  - Hata durumu:
    - Kırmızı sürekli ışık, arızalı bir donanım olduğunu belirtir
    - Firmware bir hata algılarsa tüm LED'ler yanıp söner
- Bir memory card takarsanız, LED MAINT (bakım) LED'i yanıp söner. CPU'yu kapatın ve tekrar açın. Bu durumda CPU, STOP işletim durumuna geçer. CPU STOP işletim durumuna geçtikten sonra, memory card'ın değerlendirilmesi işlemini başlatmak için şu fonksiyonlardan birini yürütün:
  - CPU'nun RUN işletim durumuna ayarlanması
  - Bir tam silme işlemi gerçekleştirin (MRES)
  - CPU'yu kapatın ve tekrar açın

LED'Lerin durumunu LED yordamı (Sayfa 405) ile de belirleyebilirsiniz.

Tablo 15-1 Bir CPU'nun durum LED'leri

Açıklama	STOP/RUN Sarı / yeşil	ERROR Kırmızı	MAINT Sarı
Şebeke kapalı	Kapalı	Kapalı	Kapalı
Yol alma, kendi kendine test veya Firmware güncellemesi	Yanıp sönme (değişimli olarak sarı ve yeşil)	-	Kapalı
STOP işletim durumu	Açık (sarı)	-	-

Açıklama	STOP/RUN Sarı / yeşil	ERROR Kırmızı	MAINT Sarı
RUN işletim durumu	Açık (yeşil)	-	-
Memory card'ı çıkartın	Açık (sarı)	-	Yanıp sönen
Hata	Açık (sarı veya yeşil)	Yanıp sönen	-
Bakım gerekiyor • Zorlanmış G/Ç • Batarya değişimi gerekli (batarya kartı takılıyken)	Açık (sarı veya yeşil)	-	Açık
Donanım arızalı	Açık (sarı)	Açık	Kapalı
LED testi veya CPU Firmware'i arızalı	Yanıp sönme (değişimli olarak sarı ve yeşil)	Yanıp sönen	Yanıp sönen
CPU konfigürasyonun sürümü bilinmiyor veya uyumsuz	Açık (sarı)	Yanıp sönen	Yanıp sönen

CPU da PROFINET iletişiminin durumunu gösteren iki LED sunar. PROFINET LED'lerini görmek için Klemens şeridinin alt koruyucu kapağını açın.

- Başarılı bir bağlantıyı göstermek için Link (yeşil) açılır
- Aktarma faaliyetini göstermek için Rx/Tx (sarı) açılır

CPU ve her dijital sinyal modülü (SM), her dijital giriş ve çıkış için bir I/O Channel LED'i sunar. İlgili girişin veya çıkışın durumunu göstermek için I/O Channel (yeşil) açılır veya kapatılır.

"CPU konfigürasyonun sürümü bilinmiyor veya uyumsuz" hatası

Teşhis tamponu, şu durumlarda ortaya çıkabilen "CPU konfigürasyonun sürümü bilinmiyor veya uyumsuz" hatasını bildirebilir:

- Örneğin bir S7-1200 V3.0 programı gibi geçersiz bir projeyi bir S7-1200 V4.x CPU'suna yüklemeyi denerseniz
- Gizli PLC konfigürasyon verilerini koruma (Sayfa 154) özelliği CPU ile projede aynı olmayan bir projeyi yüklemeye çalışırsanız

Bu duruma, geçersiz bir sürüme sahip bir aktarım kartı (Sayfa 115) kullanarak neden olduysanız, hatayı gidermek için aşağıdaki yöntemi izleyin:

1. Aktarım kartını çıkartın.
2. STOP işletim durumundan RUN işletim durumuna geçin.
3. Bir tam silme (MRES) işlemi gerçekleştirin veya cihazı kapatın ve yeniden açın.

Bu duruma, bir program kartında (Sayfa 118) geçersiz bir program kullanarak neden olduysanız, CPU'yu fabrika ayarlarına döndürün (Sayfa 1122). Bunun için "Memory card'ı biçimlendir" opsiyonunu kullanabilirsiniz.

Bu duruma, gizli PLC konfigürasyon verilerinin korunması özelliğinin CPU'da ve projede farklı olması nedeniyle sebep olduysanız, gizli PLC konfigürasyon verilerini korumak için kullanılan çevrimiçi CPU'nun şifresini ve projenin şifresini belirlemek için Çevrimiçi araçları ve teşhis araçlarını (Sayfa 1121) kullanın veya çevrimiçi CPU'dan şifreyi silin.

CPU'nun hata durumunu ortadan kaldırdıktan sonra geçerli bir program yükleyebilirsiniz.

Ciddi bir hatadan sonra S7-1200'ün davranışı

CPU Firmware'i ciddi bir hata algılarsa, hata modunda cihazın yeniden başlatılması denenir ve hata modu, STOP/RUN, ERROR ve MAINT LED'lerinin sürekli yanıp sönmesiyle gösterilir. Kullanıcı programı ve donanım konfigürasyonu, yeniden hata modunda başlatma işleminden sonra hata modunda yüklenmez.

CPU, hata modunda yeniden başlatma işlemini başarıyla gerçekleştirirse, CPU şu eylemleri gerçekleştirir:

- CPU'nun ve sinyal kartı çıkışlarının 0'a ayarlanması
- Merkezi modül taşıyıcısındaki sinyal modüllerinin çıkışlarının ve desantral çevrebirimlerinin, modülün dijital çıkışlarının cihaz konfigürasyonunda "CPU STOP durumuna tepki" seçilerek ayarlanması

Hata modunda yeniden başlatma işlemi başarısız olursa (örn. bir donanım hatası nedeniyle), STOP ve ERROR LED'leri açılır ve MAINT LED'i kapatılır.

⚠ İKAZ

Hata durumunda işletim gerçekleştirilemez

Kumanda üniteleri, güvenli olmayan işletim durumlarında devre dışı kalabilir ve bu nedenle, kumanda edilen cihazların kontrolsüz şekilde davranmasına neden olabilir. Otomasyon sisteminin bu şekilde beklenmeyen işletim davranışı, ölümcül veya ağır yaralanmalara ve/veya maddi hasarlara neden olabilir.

Bu nedenle, PLC'nize bağlı bir ACİL KAPATMA fonksiyonu, elektro mekanik veya başka yedekli güvenlik tertibatları kullanın.

Bir sinyal modülündeki (SM) durum LED'leri

Ayrıca her dijital SM, modülün durumunu gösteren bir DIAG LED'i sunar:

- Yeşil renk, modülün çalışmaya hazır olduğunu gösterir
- Kırmızı renk, modülün arızalı olduğunu veya çalışmaya hazır olmadığını gösterir

Her analog SM, analog girişler ve çıkışların her biri için bir I/O Channel LED'i sunar.

- Yeşil renk, kanalın yapılandırıldığını ve aktif olduğunu gösterir
- Kırmızı renk, ilgili analog girişin veya çıkışın bir hata durumuna sahip olduğunu gösterir

Ayrıca her analog SM, modülün durumunu gösteren bir DIAG LED'i sunar:

- Yeşil renk, modülün çalışmaya hazır olduğunu gösterir
- Kırmızı renk, modülün arızalı olduğunu veya çalışmaya hazır olmadığını gösterir

SM, modül geriliminin (alan tarafındaki gerilim, gerekli ise) mevcut olup olmadığını algılar.

Tablo 15-2 Bir sinyal modülünün (SM) durum LED'leri

Açıklama	DIAG (sarı/yeşil)	I/O Channel (sarı/yeşil)
Alan tarafındaki gerilim kapalı*	Kırmızı renkte yanıp sönüyor	Kırmızı renkte yanıp sönüyor
Yapılandırılmamış veya güncelleme devam ediyor	Yeşil renkte yanıp sönüyor	Kapalı
Modül hatasız şekilde yapılandırılmış	Açık (yeşil)	Açık (yeşil)

* Durum, sadece analog sinyal modüllerinde desteklenir.

Açıklama	DIAG (sarı/yeşil)	I/O Channel (sarı/yeşil)
Hata koşulu	Kırmızı renkte yanıp sönüyor	-
G/Ç hatası (teşhis etkinleştirilmişken)	-	Kırmızı renkte yanıp sönüyor
G/Ç hatası (teşhis devre dışı bırakılmışken)	-	Açık (yeşil)

* Durum, sadece analog sinyal modüllerinde desteklenir.

Analog modül teşhisi

Analog modüller, modül ve kanal tipine bağlı olarak birçok teşhis opsiyonuna sahiptir. Bu teşhis opsiyonlarını, TIA Portal'da projenin cihaz konfigürasyonunda veya modülün genel özelliklerinde her modül ve her kanal için münferit olarak etkinleştirebilir veya devre dışı bırakabilirsiniz.

Modül hatası

Akım beslemesi hataları aşağıdaki şekilde bildirilir:

Akım beslemesi hatası	Bildirilen hata
Bir güç kaynağı teşhis hatasına sahip analog modül şunları bildirir:	Taşma: 32767 tüm giriş kanalları için
	Çıkış modülleri için etkinleştirildiyse eksik gerilim beslemesi teşhisi

Kanal tipi hatası

Bu teşhisi her kanal ve her kanal tipi için münferit olarak etkinleştirebilirsiniz (bkz. aşağıdaki tablo).

Kanal tipi	Bildirilen hata
Gerilim girişi	Taşma: 32767
	Aşağı taşma: -32768
Akım girişi (0 ila 20 mA)	Taşma: 32767
	Aşağı taşma: -32768
Akım girişi (4 ila 20 mA) (giriş < 1,185 mA iken)	Tel kopması: 32767
	Taşma: 32767
Gerilim çıkışı (çıkış > 0,5 V iken)	Kısa devre teşhisi, etkinleştirildiyse
Akım çıkışı (çıkış > 1,0 mA iken)	Etkinleştirildiyse elektrik devresinde kesinti teşhisi
RTD girişi	Tel kopması: 32767
	Taşma: 32767
	Aşağı taşma: -32768
Direnç girişi	Tel kopması: 32767
	Taşma: 32767
Termo eleman girişi	Tel kopması: 32767
	Taşma: 32767
	Aşağı taşma: -32768

Herhangi bir kanaldaki teşhis hatası içeren bir analog giriş modülü, teşhis etkinleştirilmemiş olsa bile kanalda 32767 veya -32768 kodunu bildirir. Devre dışı bırakıldılarsa, analog giriş kanalları 32767 kodunu bildirir.

Analog giriş modülleri, birden fazla kanalda eş zamanlı olarak teşhis hatalarına sahip olabilir (birden fazla hata). Bu durum gerçekleşirse, CPU'nun sadece ilk hatası bildirilir. İlk hata bildirildikten sonra, modülde ilk hatanın nedeni ortadan kaldırılana kadar başka hata bildirilmez. İlk hata giderildikten sonra, hata durumu hâlâ mevcutsa ikinci hata bildirilir.

Bir sinyal kartındaki (SB) durum LED'leri

Her analog SB, analog girişlerin ve çıkışların her biri için bir I/O Channel LED'i sunar.

Tablo 15-3 Bir sinyal kartının (SB) durum LED'leri

Açıklama	I/O Channel (Kırmızı/Yeşil)
Yapılandırılmamış veya güncelleme devam ediyor	Kapalı
Kart hatasız şekilde yapılandırılmış	Açık (yeşil)
G/Ç hatası (teşhis etkinleştirilmişken)	Kırmızı renkte yanıp sönüyor
G/Ç hatası (teşhis devre dışı bırakılmışken)	Açık (yeşil)

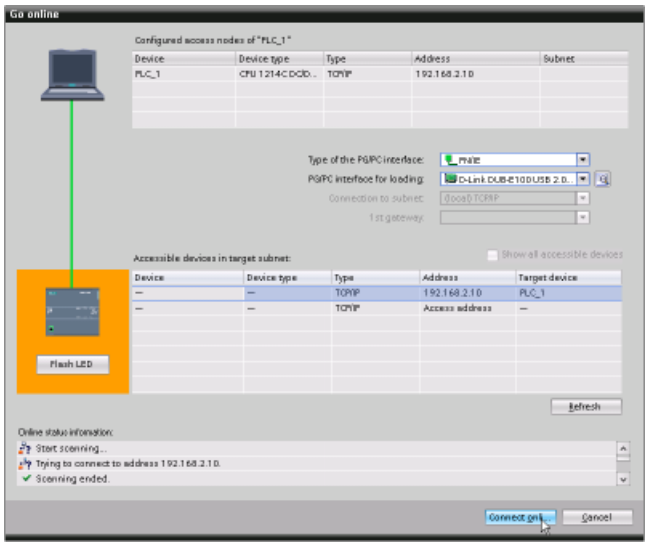
15.2 Bir CPU ile çevrimiçi bağlantı kurma

Programların ve proje verilerinin yüklenmesi ve aşağıdaki işlemler için programlama cihazı ile CPU arasındaki bir çevrimiçi bağlantı kurabilirsiniz:

- Kullanıcı programının test edilmesi
- CPU'nun işletim durumunun görüntülenmesi ve değiştirilmesi (Sayfa 1126)
- CPU'nun tarih ve saatinin görüntülenmesi ve ayarlanması (Sayfa 1119)
- Modül bilgilerinin görüntülenmesi
- Çevrimdışı ve çevrimiçi program bloklarının karşılaştırılması ve senkronize edilmesi (Sayfa 1128)
- Program bloklarının CPU'ya ve CPU'dan yüklenmesi
- Teşhisin ve teşhis tamponunun (Sayfa 1128) görüntülenmesi
- Bir izleme tablosu (Sayfa 1134) ile kullanıcı programını, değerleri izleyerek ve bu değerlere kumanda ederek test etme
- Bir Force tablosu ile CPU'daki değerlerin zorlanması (Sayfa 1137)
- CPU'nun fabrika ayarlarına sıfırlanması (Sayfa 1122)

Yapılandırılmış bir CPU ile bir çevrimiçi bağlantı kurmak için, proje gezgininde CPU'ya ve proje görünümünde "Çevrimiçi bağlan" butonuna tıklayın: 

Bu CPU ile ilk kez çevrimiçi oluyorsanız, bu arayüzde bir CPU ile bir çevrimiçi bağlantı kurmadan önce, "Çevrimiçi bağlan" iletişim kutusunda PG/PC arayüzünün tipini ve özel PG/PC arayüzünü seçmeniz gerekir.



CPU, Gizli PLC konfigürasyon verilerini koruma (Sayfa 154) özelliğine sahipse, muhtemelen sizden CPU'ya güvenmeniz istenecektir. CPU'nun sertifikasını görüntüleyebilir ve doğrulayabilir ve CPU çevrimiçi bağlantısına güvenmek mi yoksa bağlantıyı reddetmek mi istediğinize karar verebilirsiniz.

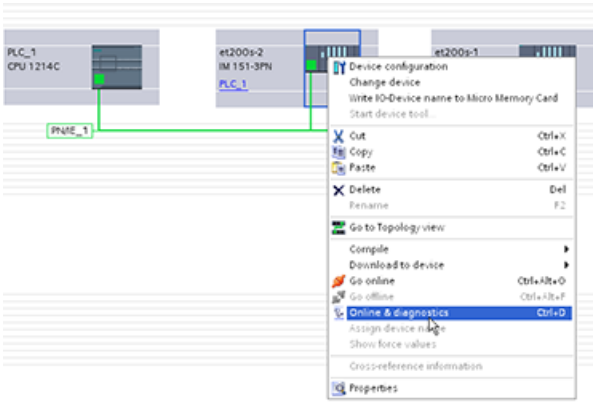
Bağlantı kurulduktan sonra turuncu renkli çerçeve bir çevrimiçi bağlantıyı gösterir. Şimdi, proje gezginindeki "Çevrimiçi ve teşhis" bölümünde yer alan araçları ve "Çevrimiçi araçlar" görev kartını kullanabilirsiniz.

15.3 Bir PROFINET IO cihazına çevrimiçi olarak bir ad atanması

CPU'ya bağlanmadan önce, PROFINET ağınızdaki cihazlara birer ad atanmalıdır. Cihazların henüz adı yoksa veya bir cihazın adının değiştirilmesi gerekiyorsa, PROFINET cihazlarınıza, "Cihazlar ve ağlar" editöründe adlar atayabilirsiniz.

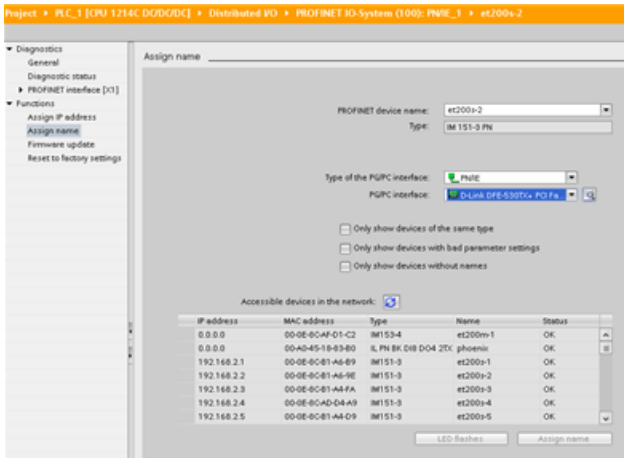
Bir PROFINET IO cihazının adını hem STEP 7 projesinde hem de "Çevrimiçi ve teşhis" aracı üzerinden PROFINET IO cihazının konfigürasyon belleğinde atamanız gerekir (örn. bir ET200 S arayüz modülünün konfigürasyon belleğinde). Bir ad eksik kalmışsa veya iki saklama konumundaki adlar eşleşmiyorsa, PROFINET IO veri alışverişi modu işlemez.

1. "Cihazlar ve ağlar" editöründe farenin sağ tuşuyla istediğiniz PROFINET IO cihazına tıklayın ve "Çevrimiçi ve teşhis" öğesini seçin.

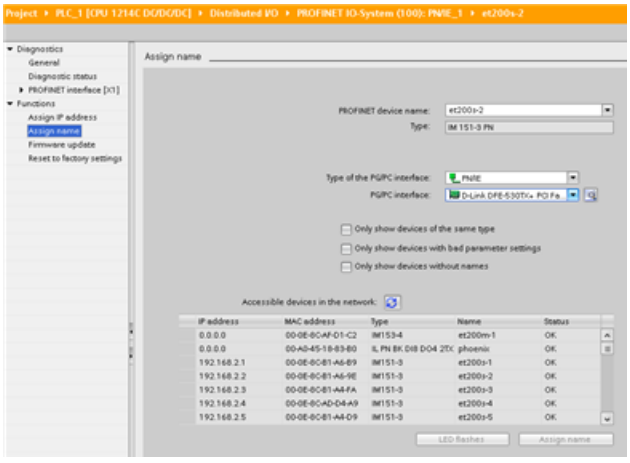


2. "Çevrimiçi ve teşhis" iletişim kutusunda aşağıdaki menü komutlarını seçin:
   - "Fonksiyonlar"
   - "PROFINET cihaz adı ata"

   Ağdaki tüm PROFINET IO cihazlarını görüntülemek için "Listeyi güncelle" butonuna tıklayın.

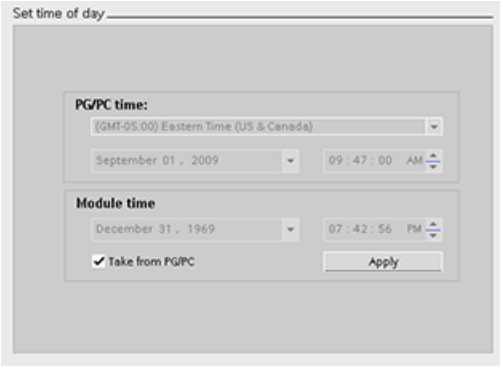


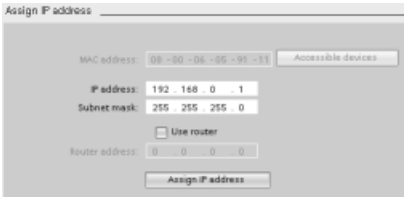
3. Adı PROFINET IO cihazının konfigürasyon belleğine yazmak için, bu işlemden sonra görüntülenen listede istediğiniz PROFINET IO cihazına ve ardından "Ad ata" butonuna tıklayın.



15.4 IP adresi ve saatin ayarlanması

Çevrimiçi CPU'nun IP adresini (Sayfa 569) ve saatini ayarlayabilirsiniz. Bir çevrimiçi CPU'nun proje gezginindeki "Çevrimiçi ve teşhis" öğesine eriştikten sonra IP adresini görüntüleyebilir veya değiştirebilirsiniz. Aynı şekilde CPU'nun tarih ve saatini de çevrimiçi olarak çağırabilir ve değiştirebilirsiniz.





NOT

Çevrimiçi ve teşhis arayüzü ile saati ayarlamak için IP adresi cihazda atanmış olmalıdır.

15.5 Firmware güncellemesi

Bağlı bir CPU'nun Firmware'ini STEP 7'deki çevrimiçi araçlar ve teşhis araçları aracılığıyla aşağıdaki yöntemlerden biriyle güncelleyebilirsiniz:

- Projede CPU üzerinden güncelleme
- Proje gezgininde erişilebilir katılımcılar üzerinde güncelleme

Projenizdeki bir CPU'nun Firmware'inin güncellenmesi

Bir Firmware güncellemesi için aşağıdaki yöntemi izleyin:

1. Proje gezgininde, bağlı CPU'ya karşılık gelen CPU'yu açın.
2. Bağlı CPU'nun çevrimiçi ve teşhis görünümünü açın.
3. "Fonksiyonlar" klasöründe "Firmware güncellemesi" girişini seçin.
4. "Firmware yükleyici" alanındaki "Gözat" butonuna tıklayın ve Firmware güncelleme dosyasının saklama konumuna gidin. Bu sırada, S7-1200 (http://support.automation.siemens.com/WW/view/en/34612486/133100) Firmware güncellemesini Siemens Industry Online-Support Web sitesinden (http://support.industry.siemens.com) indirdiğiniz ve kaydettiğiniz sabit diskinizdeki bir saklama konumu söz konusu olabilir.
5. Modülünüz için uygun bir dosyayı seçin. Tabloda, seçilen dosya için uyumlu modüller görüntülenir.
6. "Güncellemeyi yürüt" butonuna tıklayın. Gerekirse, CPU'nuzun işletim durumunu değiştirmek için iletişim kutularındaki talimatları uygulayın.

Firmware güncellemesi yüklenirken, ilerleme durumu STEP 7'de görüntülenir. Güncelleme tamamlandıktan sonra modülü yeni Firmware ile başlatmanız istenir.

NOT

Modülü yeni Firmware ile başlatmak istemiyorsanız, eski Firmware, modül örneğin kapatma ve açma yoluyla sıfırlanana kadar aktif durumda kalır. Yeni Firmware ancak modül sıfırlandıktan sonra aktif hale gelir.

Firmware'in erişilebilir katılımcılar üzerinden güncellenmesi

Bir veya daha fazla erişilebilir katılımcı için bir Firmware güncellemesi gerçekleştirmek için aşağıdaki yöntemi izleyin:

1. Proje gezgininde "Çevrimiçi erişim" öğesini açın.
2. CPU'nuzun bağlı olduğu iletişim arayüzünü açın.
3. "Erişilebilir katılımcıları güncelle" öğesine çift tıklayın ve STEP 7 çevrimiçi cihazları görüntüleyene kadar bekleyin.
4. Güncellemek istediğiniz CPU'yu genişletin ve "Çevrimiçi ve teşhis" öğesine çift tıklayın.
5. "Fonksiyonlar" klasöründe "Firmware güncellemesi" girişini genişletin. PLC'yi ve PLC için kullanılan yerel modülleri göreceksiniz. "PLC" bölümü veya "Yerel modüller" bölümü aracılığıyla Firmware güncellemesini yukarıdaki "Firmware yükleyici" bölümünde açıklandığı şekilde devam ettirebilirsiniz.

NOT

"Çevrimiçi ve teşhis" öğesi aracılığıyla Firmware'i güncellemek için IP adresi cihazda atanmış olmalıdır.

Bir Firmware güncellemesini aşağıdaki yöntemlerden birini kullanarak da gerçekleştirebilirsiniz:

- Bir SIMATIC memory card (Sayfa 124) ile
- Web sunucusunun "Modül bilgileri" standart web sitesi üzerinden (Sayfa 819)
- SIMATIC Automation Tool (https://support.industry.siemens.com/cs/ww/en/view/98161300) ile

15.6 Gizli PLC konfigürasyon verilerini koruma şifresinin belirlenmesi veya silinmesi

"Gizli PLC konfigürasyon verilerini koruma" fonksiyonuyla, projenizdeki her CPU'yu münferit olarak koruyabilirsiniz. Cihaz konfigürasyonunda (Sayfa 154) bu korumayı etkinleştirebilir ve bir "Gizli PLC konfigürasyon verilerini koruma" şifresi belirleyebilirsiniz. TIA Portal ve SIMATIC Automation Tool gibi istemciler, PLC'deki gizli verilere ancak şifre girildikten sonra erişebilir.

Bu fonksiyonu, Dijital güvenlik asistanında (Sayfa 153) da etkinleştirebilir ve "Gizli PLC konfigürasyon verilerini koruma" şifresini belirleyebilirsiniz.

Çevrimiçi araçların ve teşhis araçların kullanılması

CPU'daki gizli PLC konfigürasyon verilerini koruma şifresi projedeki şifreyle aynı değilse, CPU, RUN işletim durumuna geçemez. CPU'nun RUN işletim durumuna geçebilmesi için, gizli PLC konfigürasyon verilerini korumak için kullanılan doğru şifreyi belirlemeniz veya bu şifreyi silmeniz gerekir.

CPU çevrimiçi (Sayfa 1117) ise, Gizli PLC konfigürasyon verilerini koruma şifresini çevrimiçi CPU'da aşağıdaki şekilde belirleyebilir veya silebilirsiniz.

1. CPU'yu STOP işletim durumuna ayarlayın.
2. CPU'nuz için çevrimiçi araçları ve teşhis araçlarını açın.
3. "Fonksiyonlar" menüsünde "Gizli PLC konfigürasyon verilerini koruma şifresini belirle" opsiyonunu seçin.
4. Bir şifre belirlemek için "Belirle" butonuna veya gizli PLC konfigürasyon verilerini korumak için kullanılan mevcut şifreyi çevrimiçi CPU'da silmek için "Sil" butonuna tıklayın. "Sil" butonu kullanılamıyorsa, çevrimiçi CPU, gizli PLC konfigürasyon verilerini koruma şifresine sahip değildir.

NOT

"Çevrimiçi ve teşhis" öğesi aracılığıyla gizli PLC konfigürasyon verilerini koruma şifresini belirlemek veya silmek için IP adresi cihazda atanmış olmalıdır.

15.7 Fabrika ayarlarına sıfırlama

Aşağıdaki durumlarda, bir S7-1200'ü sıfırlayarak başlangıçtaki fabrika ayarlarına döndürebilirsiniz:

- CPU, bir çevrimiçi bağlantıya sahiptir.
- CPU, STOP işletim durumundadır.

NOT

CPU, RUN işletim durumundaysa ve sıfırlama işlemini başlatıyorsanız, bir komut istemini onayladıktan sonra CPU'yu STOP işletim durumuna ayarlayabilirsiniz.

Prosedür

Bir CPU'yu sıfırlayarak fabrika ayarlarına döndürmek için aşağıdaki yöntemi izleyin:

1. CPU'nun "Çevrimiçi ve teşhis" görünümünü açın.
2. "Fonksiyonlar" klasöründe "Fabrika ayarlarına sıfırla" girişini seçin.
3. IP adresini silmek istiyorsanız "IP adresini sil" opsiyon kutusunu etkinleştirin.
4. Şifreyi silmek istiyorsanız "Gizli PLC konfigürasyon verilerini koruma şifresini sil" opsiyon kutusunu etkinleştirin. Örneğin CPU'ya yeni bir projeyi yüklemek istiyorsanız veya CPU'yu yeni bir cihazla değiştiriyorsanız (Sayfa 1365) bu şifreyi silmeniz gerekir.
5. Mevcut durumda çevrimiçi CPU'ya takılı olan memory card'ı biçimlendirmek istiyorsanız "Memory card'ı biçimlendir" opsiyon kutusunu etkinleştirin. CPU, programınızı memory card'dan yürütüyorsa ve programı biçimlendirmek istiyorsanız bu opsiyon kutusunu etkinleştirin.
6. "PLC'yi sıfırla" butonuna tıklayın.
7. Modülü kendi ayarlarınızla sıfırlamak istediğinizi onaylamak için onay mesajını "Evet" ile onaylayın.

Sonuç

Modül, gerekirse STOP işletim durumuna ayarlanır ve sıfırlanarak fabrika ayarlarına döndürülür: Çevrimiçi CPU, aşağıdaki işlemleri gerçekleştirir:

- Çalışma belleği ve kalıcı veri alanları silinir
- Dahili bir yükleme belleği söz konusuysa bu yükleme belleği silinir; SIMATIC memory card'daki yükleme belleği SADECE "Memory card'ı biçimlendir" öğesini de seçtiyseniz silinir
- Tüm parametreler ve işlemci alanları, yapılandırılan değerlerine ayarlanır
- Teşhis tamponu silinir
- Saat sıfırlanır
- I&M0 dışındaki tanımlama ve bakım verileri (I&M) silinir
- İşletim saati sayaçları sıfırlanır
- IP adresi, yaptığınız seçime bağlı olarak korunur veya silinir. (MAC adresi sabit şekilde atanmıştır ve asla değiştirilemez.)
  "IP adresini sil" öğesini seçmediyseniz, CPU, donanım konfigürasyonunuzdaki ayarlarda yer alan IP adresini, alt ağ maskesini ve yönlendirici adresini (kullanılıyorsa), bu değerleri kullanıcı programı veya başka bir araç aracılığıyla değiştirmediyseniz korur. Bu gibi bir durumda CPU, değiştirilen değerleri geri yükler.

- Mevcutsa kumanda veri seti (Sayfa 134) silinir
- Gizli PLC konfigürasyon verilerini koruma şifresi, yaptığınız ayara bağlı olarak silinir veya korunur
- Çevrimiçi CPU'ya bir memory card takılıysa ve memory card'ın biçimlendirilmesiyle ilgili opsiyonu seçtiyseniz memory card biçimlendirilir

15.8 Bir modülün arızalar açısından kontrol edilmesi (servis verilerinin kaydedilmesi)

Servis verilerinin kaydedilmesi

S7-1200 CPU'larında bir modülün servis verilerini kaydedebilirsiniz.

SIEMENS Müşteri servisi, bakım durumunda, teşhis amacıyla sisteminize ait bir modülün durumuyla ilgili bilgilere ihtiyaç duyar. Sisteminizde bu tür durum oluşursa, muhtemelen müşteri servisi tarafından modülün servis verilerini kaydetmeniz ve sonuç dosyasını müşteri servisine göndermeniz rica edilir.

Bir modülün servis verilerini şu şekilde kaydedebilirsiniz

Bir modülün servis verilerini çevrimiçi görünümde veya teşhis görünümünde şu konumlarda kaydedebilirsiniz:

"Servis verilerini kaydet" grubunun "Fonksiyonlar" klasöründeki "Servis verilerini kaydet" grubu şu bölümlerden oluşur:

- Çevrimiçi veriler
- Servis verilerinin kaydedilmesi

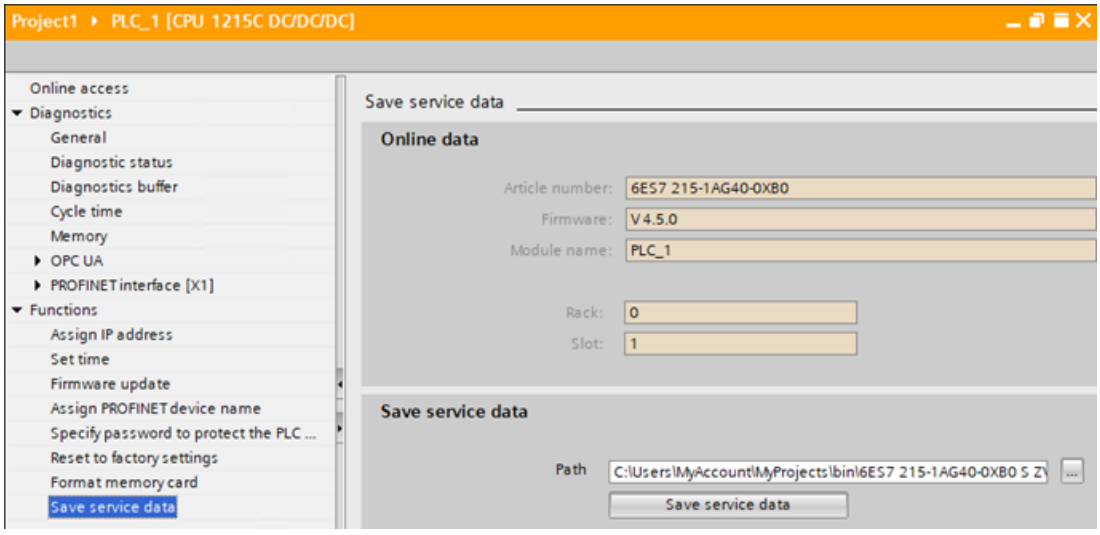
NOT

Bir modülün servis verilerini kaydetmek için IP adresi cihazda atanmış olmalıdır.

"Çevrimiçi veriler"

"Çevrimiçi veriler" bölümünde, modülün şu verileri görüntülenir:

- Sipariş numarası (MLFB)
- Firmware sürümü
- Modül adı (bu adı donanımı yapılandırırken belirlediniz)
- Modül taşıyıcısı
- Yuva



"Servis verilerini kaydet"

Özel servis verileri içeren bir dosya oluşturmak ve bu dosyayı kaydetmek için aşağıdaki yöntemi izleyin:

1. Dosya sisteminde, dosyayı kaydetmek istediğiniz konumu seçin:
   - "Yol" alanındaki ön ayarlı yolu kullanın.
   - Üç noktalı butona (Gözat) tıklayın. Bu işlemin ardından görüntülenen iletişim kutusunda istediğiniz yolu ve dosya adını belirtin.
2. "Verileri kaydet" butonuna tıklayın.
3. Servis verilerinin okunması için kullanılan bir durum göstergesi görüntülenir. Servis verileri başarılı bir şekilde ayıklandıysa "Servis verilerinin kaydetme işlemi tamamlandı" mesajı görüntülenir.

Notlar:

- Servis verileri, S7-1200'den RUN veya STOP işletim durumunda ayıklanabilir. Arıza/kritik hata durumunda servis verileri okunamaz.
- CPU'da bir şifre koruma kademesi programladıysanız, servis verilerini ayıklayabilmeniz için bu şifreyi girmeniz gerekir. Servis verileri ayıklanırken bir veri seti yazıldığı için, şifre aracılığıyla kimlik doğrulama, tüm koruma kademelerinde gereklidir.
- TIA Portal, bir servis verileri dosyasının kaydedilmesini destekler.
- Servis verileri dosyasındaki S7-1200 servis verileri elemanları şifrelenmiştir.

15.9 Bir SIMATIC memory card'ın STEP 7 aracılığıyla biçimlendirilmesi

Memory card'ı, bağlı bir CPU'da STEP 7'deki çevrimiçi araçlar ve teşhis araçları aracılığıyla biçimlendirebilirsiniz. Bunun için aşağıdaki yöntemi izleyin:

1. CPU'nun STOP işletim durumunda olduğundan emin olun. CPU, RUN işletim durumundaysa ve bir biçimlendirme işlemi başlatırsanız, STEP 7, sizden STEP 7'nin CPU'yu STOP işletim durumuna ayarlayabileceğini onaylamanızı ister.
2. Bağlı CPU'ya bir memory card takın.
3. Bağlı CPU için Çevrimiçi ve teşhis opsiyonunu, projedeki CPU üzerinden veya proje gezgininde yer alan Çevrimiçi erişim öğesinin altındaki erişilebilir katılımcılar üzerinden açın.
4. CPU çevrimiçi değilse, bağlı CPU için "Çevrimiçi bağlan" komutunu seçin.
5. "Fonksiyonlar" menüsünde "Memory card'ı biçimlendir" opsiyonunu seçin.
6. "Biçimlendir" butonuna tıklayın.
7. Görüntülenen mesajı "Evet" ile onaylayın.

Ardından STEP 7, memory card'ı biçimlendirir ve işlem tamamlandığında bilgi penceresinde bir mesaj görüntüler. CPU, biçimlendirme işlemi tamamlandığında STOP işletim durumunda bulunur ve STOP ve MAINT lambaları yanıp söner. Bu aşamada RUN işletim durumuna geçemezsiniz. Aşağıdaki önlemlerden birini uygulamanız gerekir:

- Memory card'ı çıkartın ve CPU'yu yenden başlatın: CPU'nun dahili yükleme belleği bir program içeriyorsa, CPU bu programla başlatılır.
- CPU'yu, memory card'ı çıkartmadan yeniden başlatın: CPU'nun dahili yükleme belleği bir program içeriyorsa, CPU, bu programı memory card'a kopyalar ve bu programla başlatılır. Dahili yükleme belleği hiçbir program içermiyorsa, CPU, memory card'ı bir program kartına (Sayfa 118) dönüştürür ve bir indirme işlemini bekler.

NOT

STEP 7'de bir SIMATIC memory card'ı biçimlendirmek için IP adresi cihazda atanmış olmalıdır.

Devre dışı bırakmayla bağlantılı riskler

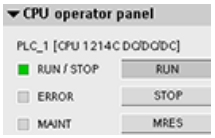
S7-1200 CPU'ları, memory card'ın ve dahili Flash belleğin güvenli şekilde silinmesini desteklemez. Bu nedenle, özel ve gizli bilgilerin kaybolmasını önlemek için, devre dışı bırakma işlemi sırasında CPU ve memory card'ı güvenli bir şekilde tasfiye etmeniz gerekir.

NOT

Bir memory card'ın biçimlendirilmesi, dahili yükleme belleğinin içeriğini etkilemez.

CPU, memory card'ı taktığınız anda dahili yükleme belleğini kullanıyorsa ve memory card'ın takılması ile biçimlendirme işleminin gerçekleştirilmesi arasında CPU'yu yeniden başlatmadıysanız, CPU, dahili yükleme belleğinin içeriğini kaydetmeye devam eder.

15.10 Çevrimiçi CPU'nun kontrol paneli



CPU kontrol panelinde, çevrimiçi CPU'nun işletim durumu (STOP veya RUN) görüntülenir. Burada, CPU'da bir hatanın oluşup oluşmadığı veya değerlerin zorlanıp zorlanmadığı da gösterilir.

"Çevrimiçi araçlar" görev kartının CPU kontrol panelinde bir çevrimiçi CPU'nun işletim durumunu değiştirebilirsiniz. "Çevrimiçi araçlar" görev kartı, CPU çevrimiçi ise kullanılabilir.

NOT

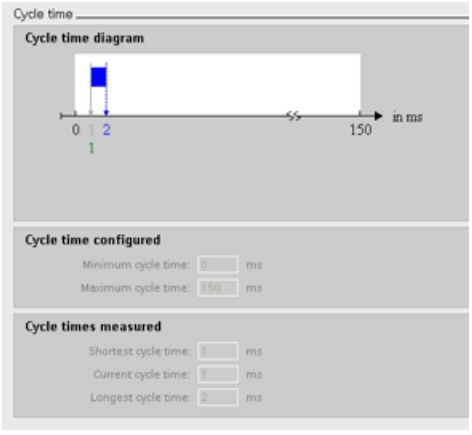
CPU kontrol panelini kullanmak için IP adresi cihazda atanmış olmalıdır.

15.11 Çevrim süresinin ve bellek kullanımının denetlenmesi

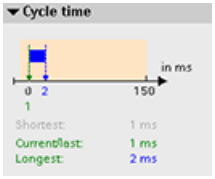
Bir çevrimiçi CPU'nun çevrim süresini ve bellek kullanımını denetleyebilirsiniz?

Çevrimiçi CPU ile bağlantı kurulduktan sonra, şu ölçüm değerlerini görüntülemek için "Çevrimiçi araçlar" görev kartını açın:

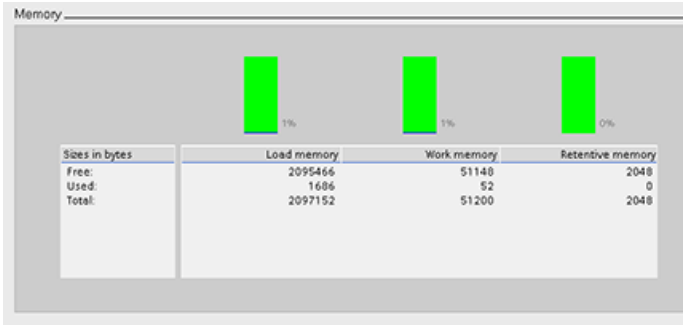
- Çevrim süresi
- Bellek kullanımı

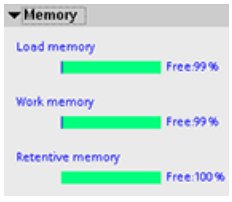


Resim



Resim





Resim

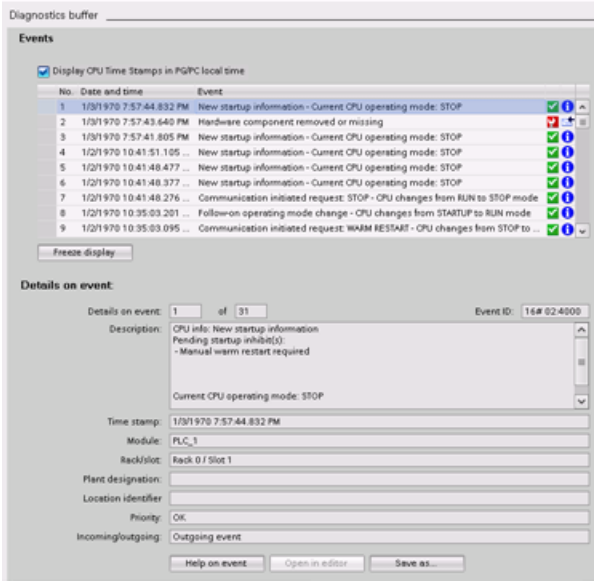
NOT

Çevrim süresini ve bellek kullanımını denetlemek için IP adresi cihazda atanmış olmalıdır.

15.12 Teşhis olaylarının CPU'da görüntülenmesi

Teşhis tamponunda, CPU'daki son olayları görüntüleyebilirsiniz. Teşhis tamponu, proje gezgininde "Çevrimiçi ve teşhis" öğesi üzerinden bir çevrimiçi CPU için kullanıma sunulmaktadır. Bu bileşen şu girişleri içerir:

- Teşhis olayları
- CPU'nun işletim durumundaki değişiklikler (STOP ile RUN işletim durumu arasında geçiş)



İlk giriş, en yeni olaya karşılık gelir. Teşhis tamponundaki her giriş, olayın protokollendirildiği tarih ve saati ve bir açıklamayı içerir.

Maksimum 50 giriş desteklenir.

CPU, sıfırlanarak fabrika ayarlarına döndürülürken, teşhis tamponu, girişlerin silinmesi yoluyla sıfırlanır.

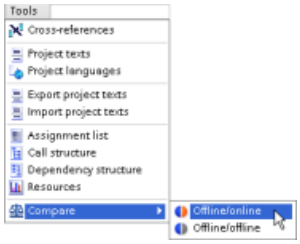
Teşhis bilgilerini kaydetmek için GET DIAG (Sayfa 430) yordamını da kullanabilirsiniz.

NOT

Teşhis olaylarının CPU'da görüntülenmesi için IP adresi cihazda atanmış olmalıdır.

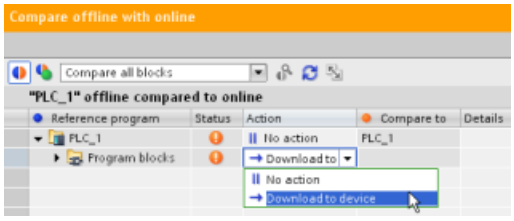
15.13 Çevrimdışı ve çevrimiçi CPU'ların karşılaştırılması

Bir çevrimiçi CPU'daki kod bloklarını projenizin kod bloklarıyla karşılaştırabilirsiniz. Projenizin kod blokları çevrimiçi CPU'nun kod bloklarına karşılık gelmiyorsa, "Karşılaştır" editöründe projenizi çevrimiçi CPU ile eşitleme olanağından faydalanabilirsiniz. Bunun için projenizin kod bloklarını CPU'ya yükleyin veya projede bulunan ve çevrimiçi CPU'da mevcut olmayan blokları silin.



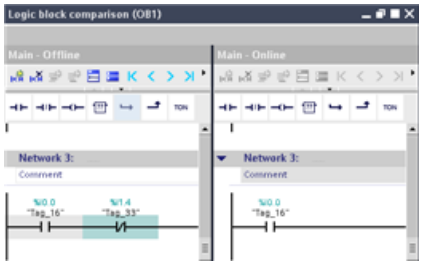
Projenizde CPU'yu seçin.

"Karşılaştır" editöründe "Çevrimdışı/çevrimiçi karşılaştır" komutunu seçin. (Komutu, "Araçlar" menüsü aracılığıyla veya projenizde CPU'ya sağ tıklayarak çağırın.)



Bir nesnenin silinmesi, hiçbir önlemin uygulanmaması veya nesnenin cihaza yüklenmesi arasından seçim yapmak için bu nesnenin "Eylem" sütununa tıklayın.

"Senkronize et" butonuna tıklayarak kod bloklarını yükleyebilirsiniz.



"Şununla karşılaştır" sütununda farenin sağ tuşuyla bir nesneye tıklayın ve kod bloklarını yan yana görüntülemek için "Ayrıntılı karşılaştırmayı başlat" öğesini seçin.

Bu ayrıntılı karşılaştırmada, çevrimiçi CPU'nun kod blokları ile projenizdeki CPU'nun kod blokları arasındaki farklar vurgulanır.

NOT

Çevrimdışı/çevrimiçi karşılaştırması için korunan CPU'ya okuma erişimi gerekir

STEP 7 açısından, çevrimdışı/çevrimiçi karşılaştırması için "HMI erişimi" güvenlik kademesi yeterli değildir. Çevrimdışı/çevrimiçi karşılaştırması için "Okuma erişimi"ne veya "Tam erişim"e ihtiyacınız bulunmaktadır.

Ayrıca bkz. CPU için erişim koruması (Sayfa 156)

15.14 Bir çevrimiçi/çevrimdışı topoloji karşılaştırmasının gerçekleştirilmesi

STEP 7'deki topolojiye genel bakışta, yapılandırılmış çevrimdışı topolojiyi gerçek çevrimiçi topolojiyle karşılaştırabilirsiniz.

İzlenecek prosedür

Yapılandırılmış ve gerçek topoloji arasındaki farkları belirlemek için aşağıdaki yöntemi izleyin:

1. Topoloji görünümünün topolojiye genel bakış tablosunu çağırın.
2. Topolojiye genel bakışın sembol çubuğundaki "Çevrimdışı/çevrimiçi karşılaştırması" butonuna tıklayın: 

Sonuç

STEP 7, topolojiye genel bakış tablosundaki "Partner istasyon", "Partner arayüz" ve "Kablo verileri" sütunlarını kaldırır ve "Durum" ve "Eylem" için karşılaştırma sütunlarını ekler. Durum sütunu, topolojiye genel bakıştaki her cihaz veya her port için karşılaştırma durumunu aşağıdaki gibi gösterir:

Sembol	Anlamı
	Alt seviyedeki en az bir bileşende farklı topoloji
	Aynı topoloji
	Topoloji bilgileri sadece çevrimdışı olarak mevcuttur veya cihaz devre dışı bırakılmıştır
	Topoloji bilgileri sadece çevrimiçi olarak mevcuttur
	Farklı topoloji
	Cihaz hiçbir topoloji fonksiyonunu desteklemiyor

Eylem sütunu, karşılaştırılan her port veya karşılaştırılan her cihaz için aşağıdaki olası opsiyonları sunar:

Sembol	Anlamı
	Hiçbir eylem uygulanamıyor
	Çevrimiçi bağlantının kabul edilmesi

Karşılaştırmayı tekrarlamak için, topolojiye genel bakışın sembol çubuğunda  butonuna tıklayın.

Topoloji görünümü, topolojiye genel bakış ve çevrimiçi/çevrimdışı topoloji karşılaştırmasıyla ilgili daha fazla bilgiyi, STEP 7 bilgi sisteminde bulabilirsiniz. Ayrıca STEP 7 ile PROFINET işletme kitabında (https://support.industry.siemens.com/cs/ww/en/view/49948856) daha fazla bilgi bulabilirsiniz.

15.15 CPU'daki değerlerin izlenmesi ve bu değerlere kumanda edilmesi

STEP 7, CPU'nun izlenmesine yönelik çevrimiçi fonksiyonlar sunar:

- Değişkenlerin güncel değerlerini görüntüleyebilir ve izleyebilirsiniz. İzleme fonksiyonu, program akışını değiştirmez. Program akışı ve CPU'daki programın verileriyle ilgili bilgileri sağlar.
- Kullanıcı programının akışına ve verilerine başka fonksiyonlarla da kumanda edebilirsiniz:
  - Kullanıcı programının nasıl tepki göstereceğini belirlemek için çevrimiçi CPU'da değişkenlerin değerine kumanda edebilirsiniz.
  - Bir çevrebirim çıkışını (A0.1:P veya "Başlat":P gibi) belirli bir değer almaya zorlayabilirsiniz.
  - Çıkışları STOP işletim durumunda etkinleştirebilirsiniz.

NOT

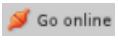
Kontrol fonksiyonlarını her zaman dikkatli bir şekilde kullanın. Bu fonksiyonlar, kullanıcı/sistem programının yürütülmesini ciddi şekilde etkileyebilir.

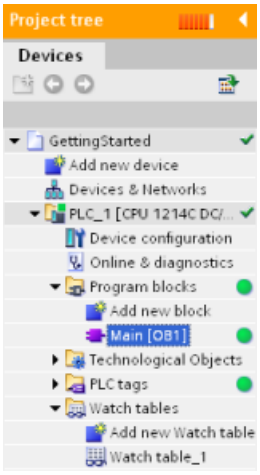
Tablo 15-4 STEP 7 editörlerinin çevrimiçi fonksiyonları

Editör	İzleme	Kumanda	Force fonksiyonu
İzleme tablosu	Evet	Evet	Hayır
Force tablosu	Evet	Hayır	Evet
Programlama editörü	Evet	Evet	Hayır
Değişken tablosu	Evet	Hayır	Hayır
DB editörü	Evet	Evet	Hayır

15.15.1 CPU'nun değerlerini izlemek için çevrimiçi olma

Değişkenlerin izlenmesi, CPU ile çevrimiçi bir bağlantının kurulmasını gerektirir. Bunun için fonksiyon çubuğunda "Çevrimiçi bağlan" butonuna tıklamanız yeterli olacaktır.





CPU ile bir bağlantı kurduysanız, STEP 7, çalışma alanlarının başlıklarını turuncu renkle gösterir.

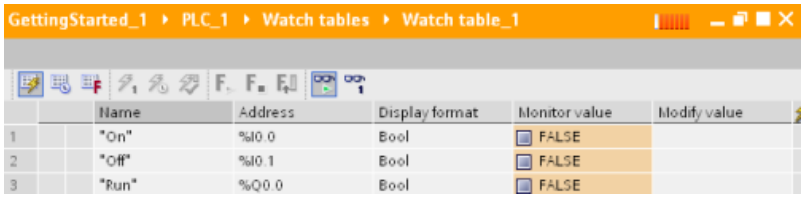
Proje gezgininde, çevrimdışı proje ile çevrimiçi CPU'nun bir karşılaştırması görüntülenir. Yeşil bir yuvarlak, CPU ile projenin senkronize edildiği, yani her ikisinin de aynı konfigürasyona ve aynı kullanıcı programına sahip olduğu anlamına gelir.

Değişkenler tabloları, değişkenleri gösterir. İzleme tablolarında, doğrudan adresler ve değişkenler de gösterilebilir.





Kullanıcı programının yürütülmesini izlemek ve değişkenlerin değerlerini görüntülemek için fonksiyon çubuğundaki "Tümünü izle" butonuna tıklayın.



"İzleme değeri" alanında, her değişken için kullanılan değer görüntülenir.

15.15.2 Durumu programlama editöründe görüntüleme

LAD ve FBD editöründe 50 adede kadar değişkenin durumunu izleyebilirsiniz. LAD editörünü editör çubuğu üzerinden çağırın. Editör çubuğu aracılığıyla, editörleri açmak veya kapatmak zorunda kalmadan açık editörler arasında geçiş yapabilirsiniz.

Kullanıcı programınızın durumunu görüntülemek için programlama editöründeki fonksiyon çubuğunda yer alan "İzleme açık/kapalı" butonuna tıklayın.



Programlama editöründe, ağdaki sinyal akışı yeşil renkle gösterilir.

Yordamın değerini değiştirmek için farenin sağ tuşuyla yordama veya parametreye tıklamanız da mümkündür.

15.15.3 Değerleri geri yüklemek için bir DB'nin çevrimiçi değerlerinin bir anlık kaydının alınması

Değerleri daha sonra kullanmak için, bir çevrimiçi CPU'dan veri bloğu değişkenlerinin gerçek değerlerinin bir anlık kaydını alabilirsiniz.

Aşağıdaki ön koşullar geçerlidir:

- CPU ile bir çevrimiçi bağlantıya ihtiyaç duyuyorsunuz.
- STEP 7'de DB'yi açmış olmanız gerekir.

Bir anlık kaydın alınması

Bir anlık kayıt almak için aşağıdaki yöntemi izleyin:

1. DB editöründeki "Tüm değişkenleri izle" butonuna tıklayın: "İzleme değeri" sütununda gerçek veri değerleri görüntülenir.
2. Gerçek değerlerin bir anlık kaydını almak ve "Anlık kayıt" sütununda görüntülemek için butonuna tıklayın.

Bu anlık kaydı, CPU'nun gerçek değerlerini güncellemek veya başlangıç değerlerini değiştirmek için daha sonra kullanabilirsiniz.

Anlık kaydın değerlerinin CPU'ya kopyalanması

Anlık kaydın değerlerini, CPU'daki veri bloğu değişkenlerinin gerçek değerlerine kopyalamak için şu butona tıklayın: 

Çevrimiçi CPU, anlık kaydın değerlerini gerçek değerlere yükler. "İzleme değeri" sütununda CPU'nun gerçek değerleri görüntülenir. Çevrim, CPU'daki değerleri muhtemelen daha sonra değiştirir, ancak kopyalama anında CPU, anlık kaydın değerlerini tutarlı bir yükleme işleminde yükler.

NOT

Anlık kaydınız durum bilgileri, zaman değerleri veya hesaplanmış bilgiler içeriyorsa, CPU bu değerleri, anlık kayıt anında oldukları şekilde geri yükler.

Anlık kaydın değerlerinin başlangıç değerlerine kopyalanması

Anlık kaydın değerlerini CPU'daki veri bloğu değişkenlerinin başlangıç değerlerine kopyalamak için şu butona tıklayın: 

CPU, DB'yi derlemeniz ve CPU'ya yüklemenizin ardından RUN işletim durumuna geçerse, DB yeni başlangıç değerlerini kullanır.

Münferit anlık kayıt veya izleme değerlerinin başlangıç değerlerine kopyalanması

Veri bloğu editöründe münferit değerleri de kopyalayabilir ve başlangıç değerlerine ekleyebilirsiniz. Farenin sağ tuşuyla herhangi bir sütundaki bir değere tıklayın ve bu değeri panoya yerleştirmek için "Kopyala" komutunu seçin. Ardından farenin sağ tuşuyla herhangi bir başlangıç değerine tıklayabilir ve bu değeri panodaki değerle değiştirmek için "Yapıştır" komutunu seçebilirsiniz.

CPU, DB'yi derlemeniz ve CPU'ya yüklemenizin ardından RUN işletim durumuna geçerse, DB yeni başlangıç değerlerini kullanır.

15.15.4 İzleme tablosu aracılığıyla CPU'daki değerlerin izlenmesi ve bu değerlere kumanda edilmesi

CPU, programınızı yürütürken, bir izleme tablosunun yardımıyla veri noktalarını izleyebilir ve bu noktalara kumanda edebilirsiniz. Bu veri noktalarında, izleme veya kontrol fonksiyonuna bağlı olarak proses görünümü (E veya A), M, DB veya fiziksel girişler (E_:P) söz konusu olabilir. İzleme fonksiyonu bellek alanı A'daki sadece son yazılan değeri görüntüleyebildiği ve fiziksel çıkışlardan gerçek değeri okumadığı için, fiziksel çıkışları (A_:P) doğru şekilde izleyemezsiniz.

İzleme fonksiyonu, program akışını değiştirmez. Program akışı ve CPU'daki programın verileriyle ilgili bilgileri sağlar.

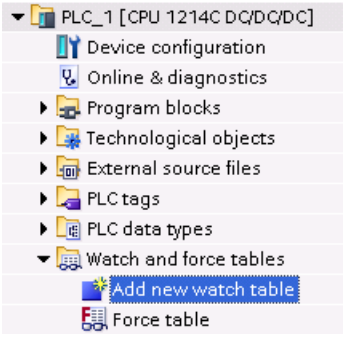
Kontrol fonksiyonları, programın akışına ve verilerine kumanda etmenize olanak sağlar. Kontrol fonksiyonlarını her zaman dikkatli bir şekilde kullanın. Bu fonksiyonlar, kullanıcı/sistem programının yürütülmesini ciddi şekilde etkileyebilir. Kumanda, Force fonksiyonu ve çıkışları STOP işletim durumunda etkinleştirme şeklinde üç kontrol fonksiyonu bulunmaktadır.

İzleme tablosuyla şu çevrimiçi fonksiyonları yürütebilirsiniz:

- Değişkenlerin durumunu izleme
- Münferit değişkenlerin değerlerini düzenleme

Bir değişkenin ne zaman izlenmesi veya değiştirilmesi gerektiği arasında seçim yapabilirsiniz:

- Çevrim başlangıcı: Değer, çevrim başlangıcı sırasında okunur veya yazılır
- Çevrim sonu: Değer, çevrim sonu sırasında okunur veya yazılır
- Stop işletim durumuna geçiş



Bir izleme tablosunu şu şekilde oluşturabilirsiniz:

1. "Yeni izleme tablosu ekle" öğesine çift tıklayarak yeni bir izleme tablosu açın.
2. İzleme tablosuna eklenmesi gereken bir değişkenin adını girin.

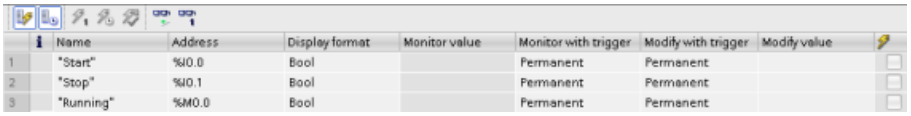
Değişkenlerin izlenmesi için şu olanaklar mevcuttur:

- Tümünü izle: Bu komutla, aktif izleme tablosundaki görünür değişkenlerin izlenmesi işlemi başlatılır.
- Şimdi izle: Bu komutla, aktif izleme tablosundaki görünür değişkenlerin izlenmesi işlemi başlatılır. İzleme tablosu, değişkenleri hemen ve sadece bir kez izler.

Değişkenlerin değiştirilmesi için şu olanaklar mevcuttur:

- "0'a ayarla" öğesi, seçilen adresin değerini "0"a ayarlar.
- "1'e ayarla" öğesi, seçilen adresin değerini "1"e ayarlar.
- "Şimdi kumanda et" öğesi, seçilen adresin değerin hemen ve sadece bir çevrim için değiştirir.
- "Tetikle kumanda et" öğesi, seçilen adresler için kullanılan değerleri değiştirir.
  Bu fonksiyon, seçilen adresin değiştirildiğini onaylamak için kullanılan bir geri bildirim oluşturmaz. Değişikliğin onaylanması gerekiyorsa "Şimdi kumanda et" fonksiyonu kullanılmalıdır.
- "PA'yı etkinleştir" öğesi, çıkışların kilitlenmesini sağlayan komutu devre dışı bırakır ve CPU, STOP işletim durumunda iken kullanılabilir.

Ancak değişkenlerin bu şekilde izlenmesi, CPU ile çevrimiçi bir bağlantının kurulmasını gerektirir.



	Name	Address	Display format	Monitor value	Monitor with trigger	Modify with trigger	Modify value
1	"Start"	%I0.0	Bool		Permanent	Permanent	
2	"Stop"	%I0.1	Bool		Permanent	Permanent	
3	"Running"	%M0.0	Bool		Permanent	Permanent	

İzleme tablosunun üst kenarındaki butonlarla çeşitli fonksiyonları seçebilirsiniz.
İzlenecek değişkenlerin adını girin ve açılır listeden bir görüntüleme formatı seçin. CPU ile bir çevrimiçi bağlantı mevcutsa, "İzle" butonuna tıklandığında veri noktasının gerçek değeri "İzleme değeri" alanında görüntülenir.

15.15.4.1 Değişkenleri tetikle izleme veya değişkenlere tetikle kumanda etme

Tetik, seçilen adresin, çevrimde hangi noktada izleneceğini veya kumanda edileceğini belirler.

Tablo 15-5 Tetikleme türleri

Tetik	Açıklama
Sürekli	Veriler sürekli kaydedilir
Çevrimin başlangıcında	Sürekli: Veriler, çevrimin başlangıcında, CPU girişleri okuduktan sonra sürekli olarak kaydedilir
	Bir kez: Veriler, çevrimin başlangıcında, CPU girişleri okuduktan sonra kaydedilir
Çevrimin sonunda	Sürekli: Veriler, çevrimin sonunda, CPU çıkışlara yazmadan önce sürekli olarak kaydedilir
	Bir kez: Veriler, çevrimin sonunda, CPU çıkışlara yazmadan önce kaydedilir
STOP işletim durumunda geçilirken	Sürekli: Veriler, CPU, STOP işletim durumuna geçerken sürekli olarak kaydedilir
	Bir kez: Veriler, CPU, STOP işletim durumuna geçtikten sonra bir kez kaydedilir

Bir PLC değişkenine belirli bir tetikle kumanda etmek için çevrimin başlangıcını veya sonunu seçin.

- Bir çıkışa kumanda etme: Bir çıkışa kumanda etmek için en iyi tetik olayı, çevrimin sonunda, CPU çıkışlara yazmadan hemen öncedir.
  Fiziksel çıkışlara hangi değerin yazılacağını belirlemek için çevrimin başlangıcında çıkışların değerini izleyin. Program lojiğini kontrol etmek ve gerçek G/Ç davranışıyla karşılaştırmak için, CPU değerleri fiziksel çıkışlara yazmadan önce de çıkışları izleyin.
- Bir girişe kumanda etme: Bir girişe kumanda etmek için en iyi tetik olayı, çevrimin başlangıcında, CPU girişleri okuduktan hemen sonra ve kullanıcı programı girişleri kullanmadan öncedir.
  Çevrim sırasında değerlerin değiştiğinden şüpheleniyorsanız, çevrim devam ederken değerlerin değişmediğinden emin olmak için çevrim sonunda girişlerin değerini denetlemek mantıklı olabilir. Değerler farklılık gösteriyorsa, kullanıcı programınız muhtemelen yanlış bir girişe yazıyordur.

CPU'nun neden STOP işletim durumuna geçtiğini belirlemek için "STOP işletim durumuna geçiş" tetiğiyle son proses değerlerini kaydedin.

15.15.4.2 Çıkışların STOP işletim durumunda etkinleştirilmesi

CPU, STOP işletim durumunda ise, izleme tablosuyla çıkışlara yazabilirsiniz. Bu fonksiyonun yardımıyla çıkışların kablajını kontrol edebilir ve bir çıkış klemensine bağlı telin, 1 veya 0 sinyalini bağlı proses cihazının klemensine ilettiğinden emin olabilirsiniz.

⚠ İKAZ

STOP işletim durumunda fiziksel çıkışlara yazarken oluşan tehlikeler

Bir fiziksel çıkışın etkinleştirilmesi, CPU, STOP işletim durumundayken de bağlı proses noktasını etkinleştirebilir ve cihazların istenmeyen davranışlar sergilemesine neden olabilir. Bu, ölümcül veya ağır yaralanmalara yol açabilir.

Bu nedenle, izleme tablosunda bir çıkışa yazma işleminden önce, fiziksel çıkış değerindeki bir değişikliğin cihazlarda istenmeyen davranışlara yol açamayacağından emin olunmalıdır. Proses sistemleriniz için geçerli güvenlik önlemlerine her zaman dikkat edin.

Çıkışlar etkinleştirildiyse, STOP işletim durumunda çıkışların durumunu değiştirebilirsiniz. Çıkışlar devre dışı bırakıldıysa, STOP işletim durumunda çıkışlara kumanda edemezsiniz. STOP işletim durumunda izleme tablosundan çıkış değerlerinde değişiklik yapmaya olanak sağlamak için aşağıdaki yöntemi izleyin:

1. "Çevrimiçi" menüsünden "Genişletilmiş mod" menü komutunu seçin.
2. "Çevrimiçi" menüsünden "Kumanda et" komutunu ve ardından "PA'yı etkinleştir" opsiyonunu seçin veya bağlam menüsünde farenin sağ tuşuyla izleme tablosunun ilgili satırına tıklayın.

Desantral çevrebirim projelendirildiyse, çıkışları STOP işletim durumunda etkinleştiremezsiniz. Yine de bu işlem gerçekleştirilmeye çalışılırsa bir hata mesajı yayınlanır.

CPU, RUN işletim durumuna geçerse, "PA'yı etkinleştir" opsiyonu devre dışı bırakılır.

Girişler veya çıkışlar zorlandıysa, CPU, STOP işletim durumunda hiçbir çıkışı etkinleştiremez. Force fonksiyonu önce sonlandırılmalıdır.

15.15.5 CPU'daki değerlerin zorlanması

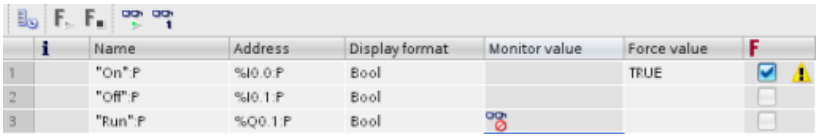
15.15.5.1 Force tablosuyla çalışma

Bir Force tablosu, bir girişin veya çıkışın değerini zorla, çevrebirim girişinin veya çıkışının adresi için önceden belirtilen bir değere ayarlayan "Force" fonksiyonunu sunar. Force fonksiyonu, girişlerin proses görünümünde kullanıcı programı yürütülmeden ve çıkışların proses görünümünde ise çıkışlar modüllere yazılmadan önce yürütülür.

NOT

Force değerleri CPU'ya kaydedilir ve Force tablosuna kaydedilmez.

Hiçbir girişi ("E" adresi) veya çıkışı ("A" adresi) zorlayamazsınız. Ancak bir çevrebirim girişini veya çevrebirim çıkışını zorlayabilirsiniz. Force tablosu adrese otomatik olarak bir ":P" ekler (örnek: "On":P veya "Run":P).



	i	Name	Address	Display format	Monitor value	Force value	F
1		"On":P	%I0.0:P	Bool		TRUE	☑ ⚠
2		"Off":P	%I0.1:P	Bool			☐
3		"Run":P	%Q0.1:P	Bool			☐

Zorlanacak giriş veya çıkış için kullanılacak değeri "Force değeri" hücresine girin. Daha sonra, girişin veya çıkışın Force fonksiyonunu etkinleştirmek için "Force" sütunundaki opsiyon kutusunu etkinleştirebilirsiniz.



Değişkenlerin değerlerini Force tablosundaki değerlere ayarlamaya zorlamak için "Force fonksiyonunu başlat veya değiştir" butonuna tıklayın. Değişkenlerin değerlerini sıfırlamak için "Force fonksiyonunu sonlandır" butonuna tıklayın.

Force tablosunda, bir girişin zorlanan değerinin durumunu izleyebilirsiniz. Ancak bir çıkışın zorlanan değerini izleyemezsiniz.

Zorlanan değerlerin durumunu programlama editöründe de görüntüleyebilirsiniz.



NOT

Bir giriş veya çıkış bir Force tablosunda zorlanırsa, Force eylemleri, proje konfigürasyonunun bir parçası hâline gelir. STEP 7 kapatılırken, CPU programındaki zorlanan elemanlar silinene kadar aktif durumda kalır. Bu zorlanan elemanları silmek için STEP 7 üzerinden çevrimiçi CPU ile bir bağlantı kurmanız ve ardından Force tablosunun yardımıyla bu elemanlar için Force fonksiyonunu devre dışı bırakmanız veya durdurmanız gerekir.

15.15.5.2 Force fonksiyonunun çalışma şekli

CPU, Force tablosunda fiziksel girişlerin ve çıkışların adreslerini (E_:P veya A_:P) belirterek ve ardından Force fonksiyonunu başlatarak, girişleri ve çıkışları zorlamanıza izin verir.

Programda, fiziksel girişlerin okunan değerlerinin üzerine Force değeri yazılır. Program, zorlanan değeri düzenleme sırasında kullanır. Program bir fiziksel çıkışa yazarsa, çıkış değerinin üzerine Force değeri yazılır. Zorlanan değer, fiziksel çıkışta görüntülenir ve proseste kullanılır.

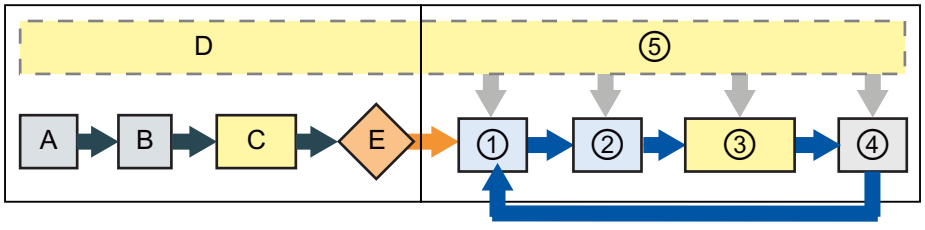
Bir giriş veya çıkış Force tablosunda zorlanırsa, Force eylemleri, kullanıcı programının bir parçası hâline gelir. Programlama yazılımını kapatsanız da, zorlanan değerler, yürütülen CPU programında, siz tekrar programlama yazılımında çevrimiçi olana ve Force fonksiyonunu sonlandırana kadar aktif durumda kalır. Bir memory card'dan başka bir CPU'ya yüklenen zorlanmış girişlere ve çıkışlara sahip programlar, programda seçilen giriş ve çıkışları da zorlar.

CPU, yazma korumalı bir memory card'daki bir kullanıcı programını yürütürken, yazma korumalı kullanıcı programındaki değerlerin üzerine yazamadığınız için G/Ç'nin zorlanmasını bir izleme tablosu üzerinden tetikleyemez veya değiştiremezsiniz. Yazma korumalı değerleri zorlamaya yönelik her deneme bir hataya neden olur. Bir kullanıcı programını aktarmak için bir memory card kullanıldığında, bu memory card'a kaydedilen zorlanmış elemanlar, birlikte CPU'ya aktarılır.

NOT

HSC, PWM ve PTO'ya atanan dijital G/Ç'ler zorlanamaz

Hızlı sayaç (HSC), darbe süresi modülasyonu (PWM) ve darbe dizisi (PTO) tarafından kullanılan G/Ç'ler cihaz konfigürasyonu sırasında atanır. Bu fonksiyonlara dijital G/Ç'ler atanırsa, atanan G/Ç'lerin adreslerinin değerleri Force tablosunun Force fonksiyonu aracılığıyla değiştirilemez.



Yol alma		RUN	
A	E bellek alanının silinmesi, Force fonksiyonu tarafından etkilenmez.	①	CPU, A belleği fiziksel çıkışlara yazılırken, çıkışların güncellenmesi sırasında Force değerini sunar.
B	Çıkış değerlerinin başlatılması, Force fonksiyonu tarafından etkilenmez.	②	Fiziksel girişler okunurken CPU, girişler E bellek alanına kopyalanmadan hemen önce Force değerlerini uygular.
C	CPU, yol alma OB'leri yürütülürken, kullanıcı programının fiziksel girişe erişmesi durumunda Force değerini sunar.	③	CPU, kullanıcı programı (program çevrimi OB'leri) yürütülürken, kullanıcı programının fiziksel girişe erişmesi veya fiziksel çıkışa yazması durumunda Force değerini sunar.
D	Bekleme kuyruğundaki alarm olaylarının kaydedilmesi etkilenmez.	④	İletişim taleplerinin işlenmesi ve kendi kendine test teşhisi, Force fonksiyonu tarafından etkilenmez.
E	Çıkışlara yazma özelliğinin serbest bırakılması etkilenmez.	⑤	Çevrimin herhangi bir kısmı sırasında alarmların işlenmesi etkilenmez.

15.16 RUN işletim durumunda yükleme

CPU, "RUN işletim durumunda yükleme"yi destekler Bu fonksiyon, program tarafından kumanda edilen prosese zarar vermeden kullanıcı programında daha küçük değişiklikler yapmanıza olanak sağlar. Ancak bu fonksiyon, prosesi olumsuz etkileyebilen ve hatta tehlikeli olabilen daha büyük program değişikliklerine de olanak sağlar.

⚠ İKAZ

RUN işletim durumunda yükleme sırasında oluşan riskler

RUN işletim durumunda değişiklikleri CPU'ya yüklerseniz, değişiklikler prosesi hemen etkiler. Programı RUN işletim durumunda değiştirirseniz, bu durum proseste beklenmeyen davranışlara neden olabilir ve bu da ölüm, ağır yaralanmalar ve/veya maddi hasarlarla sonuçlanabilir.

RUN işletim durumunda bir yükleme işlemi sadece, programın RUN işletim durumunda düzenlenmesinin proses davranışları üzerindeki etkileri konusunda bilgi sahibi olan yetkili personel tarafından gerçekleştirilmelidir.

"RUN işletim durumunda yükleme" fonksiyonu, STOP işletim durumuna geçmek zorunda kalmadan bir programda değişiklikler yapmanıza ve bu değişiklikleri CPU'ya yüklemenize olanak sağlar.

- Prosesi kapatmak zorunda kalmadan güncel proseste daha küçük değişiklikler yapabilirsiniz (örn. bir parametre değerinin değiştirilmesi).
- Ayrıca bu fonksiyon, program hatalarının daha hızlı bir şekilde giderilmesine olanak sağlar (örn. bir NO kontak veya NC kontak için kullanılan lojiği ters çevirmek).

Program blokları ve değişkenler üzerinde aşağıdaki değişiklikleri yapabilir ve bu değişiklikleri RUN işletim durumunda yükleyebilirsiniz:

- Fonksiyonlar (FC'ler), fonksiyon blokları (FB'ler) ve değişken tablolarının oluşturulması, bunların üzerine yazılması ve silinmesi
- Veri blokları (DB'ler) ve fonksiyon blokları (FB'ler) için kullanılan örnek veri bloklarının oluşturulması, silinmesi ve bunların üzerine yazılması. DB yapılarını ekleyebilir ve RUN işletim durumunda yükleyebilirsiniz. CPU, konfigürasyon ayarlarınıza (Sayfa 1144) bağlı olarak mevcut blok değişkenlerinin değerlerini kaydedebilir ve yeni veri bloğu değişkenlerini başlangıç değerleriyle başlatabilir veya tüm veri bloğu değişkenlerini sıfırlayarak başlangıç değerlerine ayarlayabilir. RUN işletim durumunda hiçbir Web sunucusu DB'sini (kumanda ünitesi veya fragment) yükleyemezsiniz.
- Organizasyon bloklarının (OB'ler) üzerine yazma; ancak hiçbir OB oluşturulamaz veya silinemez.

RUN işletim durumunda, aynı anda en fazla yirmi blok yükleyebilirsiniz. Yirmiden fazla bloğun yüklenmesi gerekiyorsa, CPU, STOP işletim durumuna ayarlanmalıdır.

Değişiklikleri gerçek bir prosese yüklerseniz (örn. bir programda hata giderme sırasında olduğu gibi simülasyonu yapılmış bir prosesten farklı olarak), yükleme işleminden önce makineler ve makine kullanıcıların güvenliği için olası sonuçları mutlaka düşünmeniz gerekir.

NOT

CPU, RUN işletim durumundaysa ve program değişiklikleri yapıldıysa, STEP 7 önce her zaman değişiklikleri RUN işletim durumunda yüklemeye çalışır. Bunun engellemek istiyorsanız, CPU'yu STOP işletim durumuna ayarlamanız gerekir.

Yapılan değişiklikler "RUN işletim durumunda yükleme" fonksiyonu tarafından desteklenmiyorsa, STEP 7, CPU'yu STOP işletim durumuna ayarlamanızı talep eder.

15.16.1 "RUN işletim durumunda yükleme" fonksiyonuyla ilgili ön koşullar

Program değişikliklerinizi ancak aşağıdaki koşulların sağlanmış olması durumunda RUN işletim durumunda CPU'ya yükleyebilirsiniz:

- Programınız başarıyla derlenebilmelidir.
- STEP 7'nin çalıştığı programlama cihazı ile CPU arasındaki iletişimi başarılı bir şekilde kurmuş olmanız gerekir.

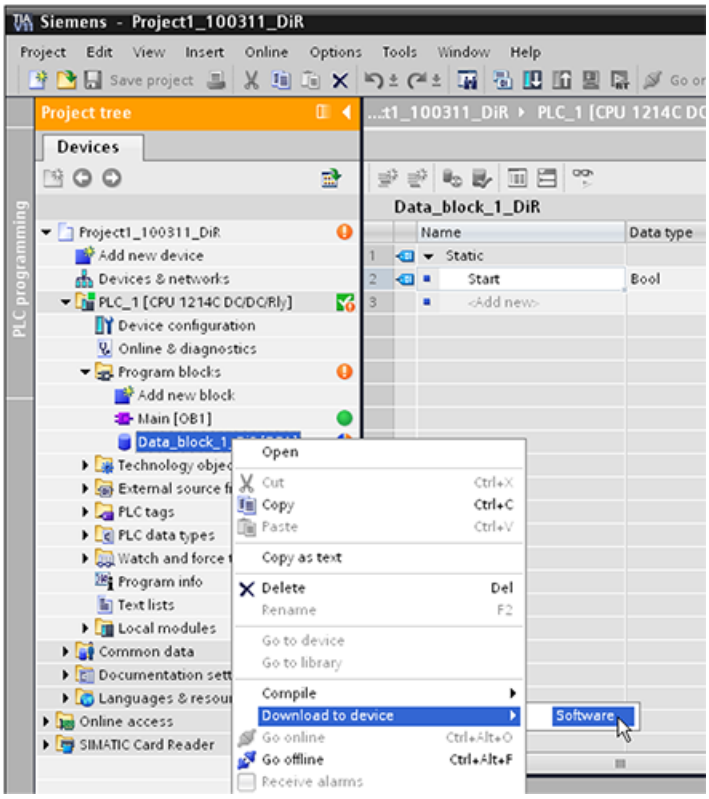
Ayrıca bakınız

Mevcut blokların RUN işletim durumunda değiştirilmesi ve hedef sisteme yüklenmesi (Sayfa 1144)

15.16.2 Programın RUN işletim durumunda değiştirilmesi

Programı RUN işletim durumunda değiştirmek için, önce CPU'nun ve programın ön koşulları (Sayfa 1141) sağlaması gerekir; bu durumda aşağıdaki yöntem izlenmelidir:

1. Programınızı RUN işletim durumunda yüklemek için aşağıdaki yöntemlerden birini izleyin:
   - "Çevrimiçi" menüsünde "Cihaza yükle" komutunu seçin.
   - Sembol çubuğundaki "Cihaza yükle" butonuna tıklayın.
   - "Proje gezgininde" farenin sağ tuşuyla "Program blokları"na tıklayın ve "Cihaza yükle > Yazılım" komutunu seçin.



   Program başarıyla derlenirse, STEP 7, programı CPU'ya yükler.
2. STEP 7, programı yüklemenizi veya işlemi iptal etmenizi talep ederse, programı CPU'ya yüklemek için "Yükle" öğesine tıklayın.

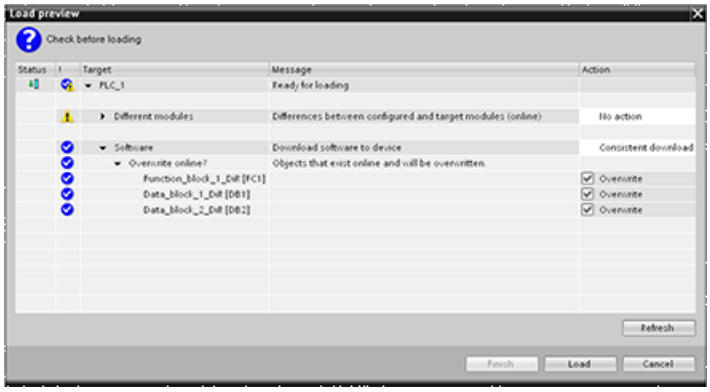
15.16.3 Seçilen blokların yüklenmesi

"Program blokları" klasöründe münferit blokları veya birden fazla bloğu yüklemek üzere seçebilirsiniz.

Yüklemek üzere tek bir bloğu seçerseniz, "Eylem" sütununda tek opsiyon olarak "Tutarlı yükleme" görüntülenir. Hangi blokların yüklenmesi gerektiğini kontrol etmek için kategori hücresini genişletebilirsiniz. Bu örnekte, çevrimdışı blokta küçük bir değişiklik yapılmıştır ve başka blokların yüklenmesi gerekmez.

Bu örnekte birden fazla bloğun yüklenmesi gerekmektedir.

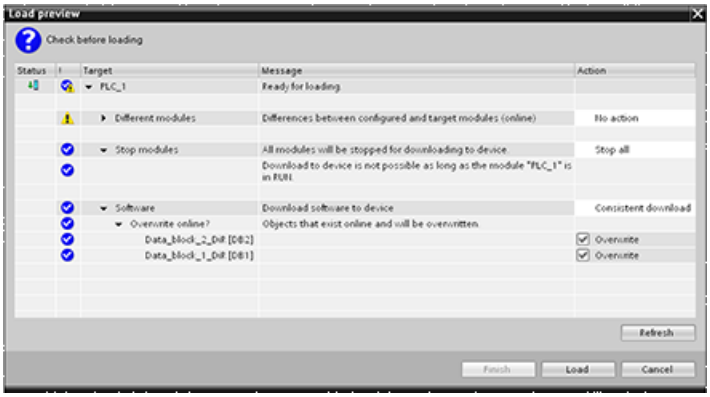




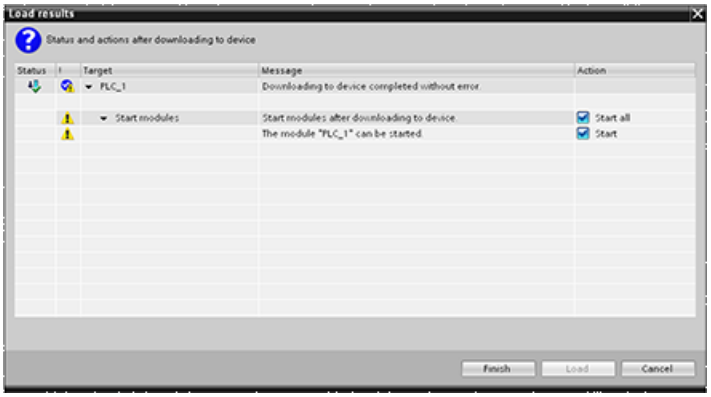
NOT

RUN işletim durumunda, aynı anda en fazla yirmi blok yükleyebilirsiniz. Yirmiden fazla bloğun yüklenmesi gerekiyorsa, CPU, STOP işletim durumuna ayarlanmalıdır.

Blokları RUN işletim durumunda yüklemeyi denerseniz, ancak sistem asıl yükleme işleminden önce bunun mümkün olmadığını algılarsa, iletişim kutusunda modüllerin durdurulması gerektiğini belirten bir satır görüntülenir.

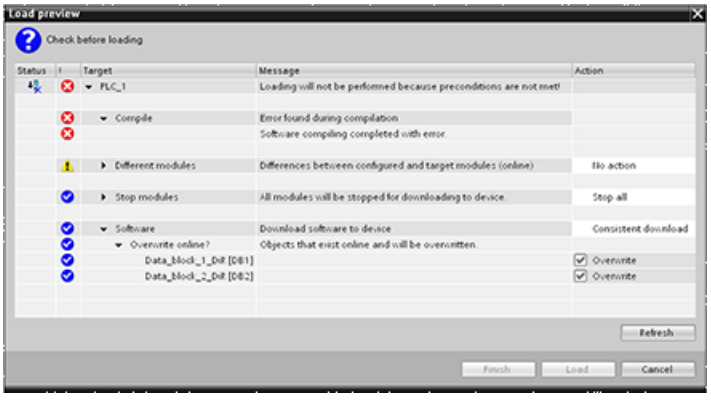


"Yükle" butonuna tıklayın. Bunun üzerine "Yükleme sonuçları" iletişim kutusu görüntülenir. Yükleme işlemini tamamlamak için "Tamamla" butonuna tıklayın.



15.16.4 Bir derleme hatası içeren seçilmiş tek bir bloğun başka bir blokta yüklenmesi

Bir derleme hatası içeren tutarlı bir yükleme işlemini başka bir blokta gerçekleştirmeye çalışırsanız, iletişim kutusunda bir hata görüntülenir ve yükleme butonu devre dışı bırakılır.



Önce, diğer blokta derleme hatasını düzeltmeniz gerekir. Ardından, "Yükle" butonu tekrar etkinleştirilir.



15.16.5 Mevcut blokların RUN işletim durumunda değiştirilmesi ve hedef sisteme yüklenmesi

RUN işletim durumunda yükleme fonksiyonuyla, veri bloklarına ve fonksiyon bloklarına değişkenler ekleyebilir ve bu bileşenleri değiştirebilir ve ardından, değiştirilen bloğu RUN işletim durumunda CPU'ya yükleyebilirsiniz.

Yeniden başlangıç olmadan yükleme

Her DB ve FB, bloğa değişkenler eklemek ve daha sonra bu bileşeni RUN işletim durumunda CPU'ya yüklemek için kullanabileceğiniz bir bellek rezervine sahiptir. Bu bellek rezervinin başlangıç boyutu, standart olarak 100 Bayt'tır. Bellek rezervinin boyutu çerçevesinde verilerinize ek değişkenler ekleyebilir ve genişletilmiş bloğu RUN işletim durumunda CPU'ya yükleyebilirsiniz. Bloğunuzda ek değişkenler için daha fazla belleğe ihtiyacınız varsa bellek rezervini büyütmeniz de mümkündür. Belleğin alabileceğinden daha fazla değişken eklerseniz, genişletilmiş bloğu RUN işletim durumunda CPU'ya yükleyemezsiniz.



"Yeniden başlangıç olmadan yükleme" fonksiyonuyla, daha fazla veri bloğu değişkeni ekleyerek ve genişletilmiş veri bloğunu RUN işletim durumunda CPU'ya yükleyerek bir veri bloğunu genişletebilirsiniz. Bu şekilde bir veri bloğuna değişkenler ekleyebilir ve DB'yi yeniden başlangıç olmadan programınıza yükleyebilirsiniz. CPU, mevcut veri bloğu değişkenlerinin değerlerini kaydeder ve yeni eklenen değişkenleri başlangıç değerleriyle birlikte başlatır.

Bu fonksiyonu bir çevrimiçi proje için CPU ile RUN işletim durumunda etkinleştirmek için aşağıdaki yöntemi izleyin:

1. STEP 7 proje gezginindeki program bloklarının klasöründe bloğu açın.
2. Fonksiyonu etkinleştirmek için blok editöründe "Yeniden başlangıç olmadan yükleme" butonuna tıklayın. (Sembol, etkinleştirilmiş durumdayken çerçeve içine alınır:)
3. Yaptığınız seçimi "TAMAM" ile onaylayın.
4. Blok arayüzüne değişkenler ekleyin ve bloğu RUN işletim durumunda yükleyin. Bu sayede, bellek rezerviniz izin verdiği ölçüde yeni değişkenler ekleyebilir ve yükleyebilirsiniz.

Bloğunuza, bellek rezervi için yapılandırdığınızdan daha fazla Bayt eklediyseniz, bloğu RUN işletim durumunda yüklemeyi denediğiniz anda STEP 7'de bir hata görüntülenir. Blok özelliklerini düzenlemeniz ve bellek rezervini büyütmeniz gerekir. "Yeniden başlangıç olmadan yükleme" fonksiyonu etkinleştirilmiş durumda olduğu sürece, mevcut hiçbir girişi silemez ve bloğun bellek rezervini değiştiremezsiniz. "Yeniden başlangıç olmadan yükleme" fonksiyonunu devre dışı bırakmak için aşağıdaki yöntemi izleyin:

1. Fonksiyonu devre dışı bırakmak için blok editöründe "Yeniden başlangıç olmadan yükleme" butonuna tıklayın. (Sembol, devre dışı bırakılmış durumdayken çerçeve içine alınmaz:)
2. Yaptığınız seçimi "TAMAM" ile onaylayın.
3. Bloğu CPU'ya yükleyin. Genişletilmiş bloğu yüklemek için yükleme iletişim alanında "Yeniden başlangıç" öğesini seçmeniz gerekir.

Bu durumda, yükleme sırasında mevcut ve yeni tüm blok değişkenleri başlangıç değerleriyle yeniden başlatılır.

Kalıcı blok değişkenlerinin yüklenmesi

Kalıcı blok değişkenlerini RUN işletim durumunda yükleyebilmek için kalıcı bir bellek rezervinin atanması gerekir. Bu kalıcı bellek rezervini yapılandırmak için aşağıdaki yöntemi izleyin:

1. STEP 7 proje gezginindeki program blokları klasöründe bloğa farenin sağ tuşuyla tıklayın ve bağlam menüsünden "Özellikler" öğesini seçin.
2. "Yeniden başlangıç olmadan yükleme" özelliğini seçin.
3. "Kalıcı değişkenler için yeniden başlangıç olmadan yükleme" opsiyon kutusunu etkinleştirin.
4. Kalıcı bellek rezervi için kullanılabilen Bayt sayısını yapılandırın.
5. Yaptığınız değişiklikleri "TAMAM" ile onaylayın.
6. Veri bloğuna kalıcı veri bloğu değişkenleri ekleyin ve veri bloğunu RUN işletim durumunda yükleyin. Bu sayede, kalıcı bellek rezerviniz izin verdiği ölçüde yeni kalıcı veri bloğu değişkenleri ekleyebilir ve yükleyebilirsiniz.

Veri bloğunuza, bellek rezervi için yapılandırdığınızdan daha fazla kalıcı Bayt eklediyseniz, bloğu RUN işletim durumunda yüklemeyi denediğiniz anda STEP 7'de bir hata görüntülenir. Değişkenleri RUN işletim durumunda yükleyebilmek için kalıcı blok değişkenlerini sadece kalıcı bellek rezervinin boyutu çerçevesinde ekleyebilirsiniz.

Genişletilmiş bir kalıcı blok değişkenini yüklüyorsanız, değişkenler kendi güncel değerlerini içerir.

Yeni bloklar için bellek rezervinin boyutunun yapılandırılması

Yeni veri blokları için bellek rezervinin standart boyutu 100 Bayt'tır. Yeni bir blok oluşturuyorsanız, bu blok 100 Bayt'lık bir rezervi kapsar. Yeni blokları başka bir boyuttaki bir bellek rezerviyle oluşturmak istiyorsanız, PLC programlama ayarlarında ilgili ayarı değiştirebilirsiniz:

1. STEP 7'de **Opsiyonlar > Ayarlar** menü komutunu seçin.
2. "Ayarlar" iletişim alanında "PLC programlama" alanını genişletin ve "Genel" öğesini seçin.
3. "Yeniden başlangıç olmadan yükleme" alt bölümünde bellek rezervi için kullanılacak Bayt sayısını girin.

STEP 7, yeni bloklar oluşturulurken, yeni bloklar için yapılandırdığınız bellek rezervini kullanır.

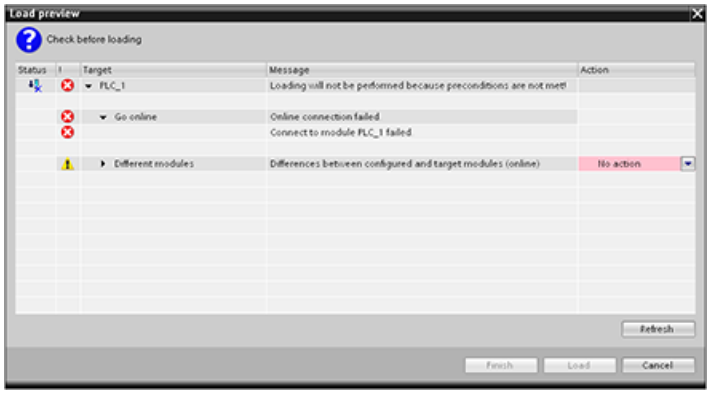
Sınırlamalar

Blokların RUN işletim durumunda düzenlenmesi ve yüklenmesiyle ilgili olarak aşağıdaki kısıtlamalar geçerlidir:

- Blok arayüzünü genişletme ve yeni değişkenler ekleme ve RUN işletim durumunda yükleme işlemleri, sadece optimize edilen bloklarda (Sayfa 176) mümkündür.
- Yeniden başlangıç olmadan bir bloğun yapısını değiştiremez ve değiştirilen bloğu RUN işletim durumunda yükleyemezsiniz. Yeni elemanların bir Struct (Sayfa 106) değişkenine eklenmesi, değişken adlarının, Array boyutlarının, veri tiplerinin veya kalıcı durumun değiştirilmesi için, RUN işletim durumunda yüklemek için bloğu yeniden başlatmak gerekir. Mevcut blok değişkenlerinde gerçekleştirebileceğiniz ve yeniden başlatma olmadan RUN işletim durumunda yükleyebileceğiniz değişiklikler; sadece başlangıç değerlerinde (veri blokları), standart değerlerde (fonksiyon blokları) veya yorumlardaki değişikliklerdir.
- RUN işletim durumunda, en fazla bellek rezervinin alabileceği sayıda yeni blok değişkenini yükleyebilirsiniz.
- RUN işletim durumunda, en fazla kalıcı bellek rezervinin alabileceği sayıda yeni kalıcı blok değişkenini yükleyebilirsiniz.

15.16.6 Yükleme işlemi başarısız olduğunda sistem tepkisi

RUN işletim durumunda ilk yükleme işlemi sırasında bir ağ bağlantısı hatası oluşursa, STEP 7, aşağıda gösterilen "Önizlemeyi yükle" iletişim kutusunu görüntüler:



15.16.7 RUN işletim durumunda yükleme sırasındaki güvenlikle ilgili hususlar

Programı RUN işletim durumunda yüklemeden önce, RUN işletim durumunda yaptığınız değişikliklerin, aşağıdaki durumlarda CPU'nun işletimi üzerinde oluşturacağı etkileri düşünün:

- Bir çıkış için kullanılan kumanda lojiğini sildiyseniz, CPU, CPU kapatılana veya STOP işletim durumuna ayarlanana kadar çıkışın son durumunu korur.
- Bir hızlı sayacı veya bir darbe çıkışını sildiyseniz ve iki fonksiyondan biri çalışıyorsa, hızlı sayaç veya darbe çıkışı bir sonraki kapatma işlemine veya bir sonraki STOP işletim durumuna geçişe kadar çalışmaya devam eder.
- İlk çevrimin markörü aracılığıyla etkinleştirilen lojik, ancak bir sonraki açma işleminden veya STOP işletim durumundan RUN işletim durumuna bir sonraki geçişten sonra yürütülür. İlk çevrimin markörü, sadece RUN işletim durumuna geçiş aracılığıyla ayarlanır ve RUN işletim durumunda yükleme işleminden etkilenmez.
- Veri bloklarının (DB'ler) güncel değerleri ve/veya değişkenlerin üzerine yazılabilir.

NOT

Programınızı RUN işletim durumunda yükleyebilmeniz için, CPU, RUN işletim durumunda yükleme özelliğini desteklemelidir. Program hatasız bir şekilde derlenmeli ve STEP 7 ile CPU arasındaki iletişim hatasız olmalıdır.

Program blokları ve değişkenler üzerinde aşağıdaki değişiklikleri yapabilir ve bu değişiklikleri RUN işletim durumunda yükleyebilirsiniz:

- Fonksiyonlar (FC'ler), fonksiyon blokları (FB'ler) ve değişken tablolarının oluşturulması, bunların üzerine yazılması ve silinmesi
- Veri blokları (DB'ler) oluşturma ve silme; ancak DB yapı değişikliklerinin üzerine yazılamaz. DB'nin çıkış değerlerinin üzerine yazılabilir. RUN işletim durumunda hiçbir Web sunucusu DB'sini (kumanda ünitesi veya fragment) yükleyemezsiniz.
- Organizasyon bloklarının (OB'ler) üzerine yazma; ancak hiçbir OB oluşturulamaz veya silinemez.

RUN işletim durumunda, aynı anda en fazla yirmi blok yükleyebilirsiniz. Yirmiden fazla bloğun yüklenmesi gerekiyorsa, CPU, STOP işletim durumuna ayarlanmalıdır.

Bir yükleme işlemini başlattığınızda, STEP 7'de başka işlemleri ancak yükleme işlemi tamamlandığı takdirde gerçekleştirebilirsiniz.

"RUN işletim durumunda yükleme" sırasında başarısız olabilen yordamlar

Aşağıdaki yordamlar, CPU'da RUN işletim durumunda yükleme fonksiyonundan kaynaklanan değişiklikler etkinleştirildiği takdirde kısa süreli hatalar bildirebilir. Hata, CPU yüklenen değişiklikleri etkinleştirmeye hazırlanırken yordam başlatıldığında ortaya çıkar. CPU, bu süre zarfında, kullanıcı programının henüz düzenleme aşamasında bulunan yükleme belleğine erişim işlemlerini işlediği sürece kullanıcı programının yükleme belleğine sonraki erişimleri durdurur. Bu durum, yüklenen değişikliklerin sürekli olarak etkinleştirilebilmesi için gerçekleşir.

Yordam	Etkinleştirme devam ettiği sürece verilen tepki
DataLogCreate	DURUM = W#16#80C0, ERROR = DOĞRU
DataLogOpen	DURUM = W#16#80C0, ERROR = DOĞRU
DataLogWrite	DURUM = W#16#80C0, ERROR = DOĞRU
DataLogClose	DURUM = W#16#80C0, ERROR = DOĞRU
DataLogNewFile	DURUM = W#16#80C0, ERROR = DOĞRU
DataLogClear	DURUM = W#16#80C0, ERROR = DOĞRU
DataLogDelete	DURUM = W#16#80C0, ERROR = DOĞRU
READ_DBL	RET_VAL = W#16#82C0
WRIT_DBL	RET_VAL = W#16#82C0
Create_DB	RET_VAL = W#16#80C0
Delete_DB	RET_VAL = W#16#80C0
RTM	RET_VAL = 0x80C0

Hata oluştuğu takdirde tüm durumlarda yordamın VKE'si yanlıştır. Hata geçicidir. Bu hata oluşursa, yordamı daha sonra yürütün.

NOT

Yordamı, OB'nin güncel düzenlenme işlemi sırasında yürütmeye çalışmayın.

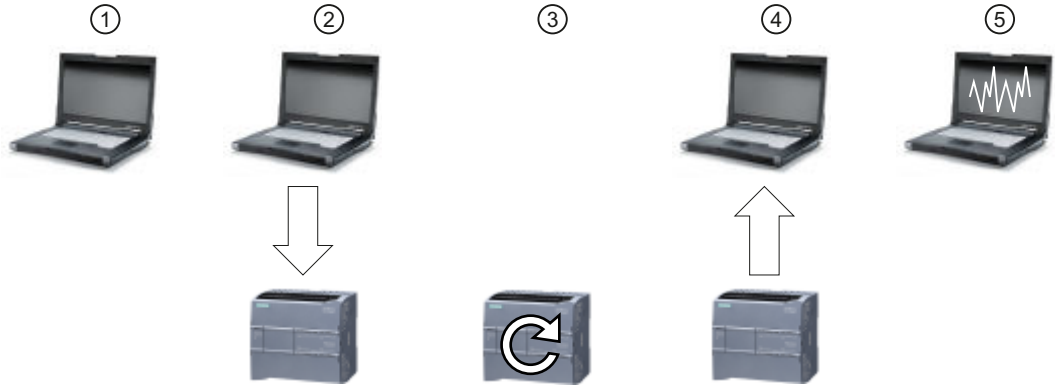
15.17 Tetikleme koşullarında CPU verilerinin takip edilmesi ve kaydedilmesi

STEP 7; takip etmek ve kaydetmek istediğiniz PLC değişkenlerini yapılandırmak için kullanabileceğiniz Takip ve Lojik analizörü fonksiyonları sunar. Takip ölçümlerinin kaydedilen verilerini daha sonra programlama cihazınıza yükleyebilir ve STEP 7 araçlarıyla analiz edebilir, yönetebilir ve grafiksel olarak hazırlayabilirsiniz. STEP 7 proje gezginindeki "Traces" klasöründe, takip öğeleri oluşturabilir ve bunları yönetebilirsiniz.

NOT

Takip ölçümlerinin verileri sadece STEP 7 projesinde kullanılabilir ve başka araçlar aracılığıyla işlenmek üzere kullanıma sunulmaz.

Aşağıdaki şekilde, takip fonksiyonunun işleyişi gösterilmektedir:



① Takip fonksiyonunu STEP 7'nin Takip editöründe ayarlayın. Şu opsiyonları belirleyebilirsiniz:
- Kaydedilecek veri değerleri
- Kaydetme süresi
- Kaydetme sıklığı
- Tetikleme koşulu

② Takip konfigürasyonunu STEP 7'den PLC'ye aktarın.

③ PLC, tetikleme koşulu sağlandığında programı yürütür ve takip verilerinin kaydedilmesi işlemini başlatır.

④ Kaydedilen değerleri PLC'den STEP 7'ye aktarın.

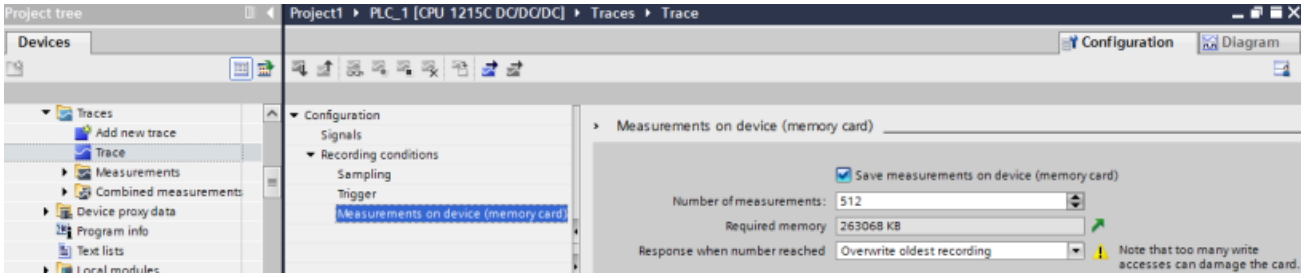
⑤ Verileri değerlendirmek, grafiksel olarak görüntülemek ve yedeklemek için STEP 7'deki araçları kullanın.

S7-1200, tetik olayı başına kaydedilen maksimum 16 değişkenle iki takip görevini destekler. Her takip görevi, takip değerlerinin kaydedilmesi ve örneğin değişken adresleri ve zaman notu gibi bu değerlerle bağlantılı yönetim değerleri için RAM bellekte 524.288 Bayt sağlar.

Takip ölçümlerinin memory card'a kaydedilmesi

S7-1200 CPU'su, takip ölçümlerini sadece SIMATIC memory card'a kaydedebilir. CPU'nuzda memory card yoksa, program, takip ölçümlerini kaydetmeyi denediğinde CPU, teşhis tamponunda bir giriş oluşturur. CPU, takip ölçümleri için bellek alanını, her zaman harici yükleme belleğinin 1 MB'ı kullanılabilecek şekilde sınırlar. Bir takip ölçümü, izin verilen maksimum değerden daha fazla bellek gerektiriyorsa, CPU, ölçümü kaydetmez ve teşhis tamponunda bir giriş oluşturur.

Ayrıca, STEP 7'de "En eski kaydın üzerine yaz" öğesi seçildiğinde, veri taşıyıcıya sürekli yazılması nedeniyle yükleme belleğinin kullanım ömrü kısalabilir. CPU, "En eski kaydın üzerine yaz" öğesi seçildiğinde, yapılandırılan sayıda takip ölçümünü kaydettikten sonra en eski mesajı en yeni mesajla değiştirir. Ardından, diğer takip ölçümleri belirlenir ve kaydedilir. En eski ölçümlerin üzerine yazma özelliği, ara sıra ortaya çıkan sorunların teşhisinde faydalıdır.



CPU, maksimum 999 takip ölçüm sonucunu destekler. CPU'nun takip ölçümlerini harici yükleme belleğine kaydettiği süre zarfında CPU, takip görevi için tanımlanan tetikleme koşulunu kontrol etmez. CPU, takip ölçümlerini kaydettikten sonra tetikleme koşullarını tekrar kontrol eder.

Örneklere erişim

Takip programlama, konfigürasyonu yükleme, takip verilerini karşıya yükleme ve verilerin Logic Analyzer'da gösterimiyle ilgili ayrıntıları, STEP 7 bilgi sisteminde bulabilirsiniz. "Çevrimiçi fonksiyonların ve teşhis fonksiyonlarının kullanımı" > "Takip ve lojik analizörü fonksiyonunu kullanma" bölümü, ayrıntılı örnekler içermektedir.

Ayrıca "Industry Automation SINAMICS/SIMATIC Takip ve lojik analizörü fonksiyonunu kullanma" (https://support.industry.siemens.com/cs/ww/en/view/64897128) çevrimiçi işletme kitabında ayrıntılı bir yardım bölümüdür.

15.18 Bir tel kopmasının bir SM 1231 modülü aracılığıyla belirlenmesi

Gerilim ve akım için analog girişlerin ölçüm aralıkları (SB ve SM) (Sayfa 1271) başlığında açıklandığı gibi, SM 1231 modülü, bir tel kopması ve aynı zamanda bir taşma durumunda 32767 (16#7FFF) analog giriş değerini yayınlar. Bu iki durumun hangisinin oluştuğunu belirlemek isterseniz, belirleme amacıyla STEP 7 programınıza lojiği dahil edebilirsiniz. Durumu belirleme yöntemi şu görevlerden oluşur:

- Gelen veya giden bir teşhis olayı mevcutsa çağrılan bir teşhis hatası OB'si oluşturun.
- RALRM yordamının bir çağrısını kaydedin.
- Durumla ilgili bilgileri içeren AINFO parametresi için Bayt'lardan oluşan bir Array ayarlayın.
- CPU, teşhis alarmı OB'sini tetiklerse, RALRM_DB'nin AINFO yapısına ait 32 ve 33 Bayt'larını değerlendirin.

Bir teşhis hatası OB'sinin oluşturulması

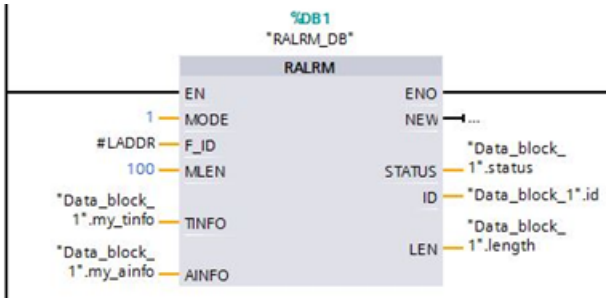
Bir tel kopması durumunun ne zaman ortaya çıktığını belirlemek için bir teşhis hatası OB'si oluşturabilirsiniz. CPU, bu OB'yi her zaman, bir gelen veya giden teşhis olayı oluştuğunda çağırır.

CPU teşhis hatası OB'sini çağırırsa, LADDR giriş parametresi, hatalı modülün donanım kodunu içerir. SM 1231 modülünün donanım kodunu, SM 1231 modülünün STEP 7 cihaz konfigürasyonunda bulabilirsiniz.

RALRM yordamının çağrılması

RALRM yordamının çağrısını programlamak için aşağıdaki yöntemi izleyin:

1. STEP 7 programınıza bir RALRM çağrısı ekleyin.
2. F_ID giriş parametresi için teşhis hatası OB'sinin LADDR parametresinin donanım kodunu ayarlayın.
3. TINFO ve AINFO giriş parametreleri için Bayt'lardan oluşan bir Array kullanın. En az 34 Bayt'lık bir Array boyutu kullanın.



Bir teşhis alarmı oluştuktan sonra AINFO'nun değerlendirilmesi

Hata alarmı OB'si yürütüldükten sonra Bayt Array'i AINFO modül teşhisiyle ilgili bilgileri içerir. 0 - 25 aralığındaki Bayt'lar başlık bilgileridir. Sonraki Bayt'lar modül teşhisine aittir:

Bayt	Açıklama	
26 ve 27	16#8000 word değeri, PROFINET-Stil'de bir teşhisin söz konusu olduğunu belirtir	
28 ve 29	Bu teşhisten sorumlu kanal numarasını içeren sözcük	
30	aaabb000 Bit örüntüsü, kanalın türünü (aaa) ve hatanın türünü (bb) belirtir	
	aaa	bb
	000: Rezerve	00: Rezerve
	001: Giriş kanalı	01: gelen hata
	010: Çıkış kanalı	10: giden hata
	011: Giriş/çıkış kanalı	11: giden hata, başka hatalar mevcut
31	Veri formatının görüntülenmesi 0: Serbest veri formatı 1: Bit 2: İki Bit 3: Dört Bit 4: Bayt 5: Sözcük (iki Bayt) 6: Çift sözcük (dört Bayt) 7: İki çift sözcük (sekiz Bayt)	
32 ve 33	Hata türünü tanımlayan sözcük: 16#0000 rezerve 16#0001: Kısa devre 16#0002: Düşük gerilim 16#0003: Aşırı gerilim 16#0004: Aşırı yük 16#0005: Aşırı sıcaklık 16#0006: Tel kopması 16#0007: üst limit aşıldı 16#0008: alt limit aşıldı 16#0009: Hata	

Örneğin bu AINFO yapısının 26 - 33 aralığındaki Bayt'larına dikkat edin:

29	my_ainfo[26]	Byte	16#0	16#80
30	my_ainfo[27]	Byte	16#0	16#00
31	my_ainfo[28]	Byte	16#0	16#00
32	my_ainfo[29]	Byte	16#0	16#00
33	my_ainfo[30]	Byte	16#0	16#28
34	my_ainfo[31]	Byte	16#0	16#05
35	my_ainfo[32]	Byte	16#0	16#00
36	my_ainfo[33]	Byte	16#0	16#07

- 26 ve 27 Bayt'larındaki sözcük, PROFINET-Stil'de bir teşhisin söz konusu olduğunu belirten 16#8000'dir.
- 28 ve 29 Bayt'larındaki sözcük, bu teşhisin Kanal 0 veya modül için geçerli olduğunu gösterir.
- Bayt 30, Bit örüntüsü aaa bb 00 olarak = 001 01 000 olan 16#28'dir. Bu değer, bu teşhisin bir giriş kanalı için geçerli olduğunu ve bir gelen hatanın söz konusu olduğu gösterir.

- Bayt 31, bir sözcük değerini belirten 5'tir.
- 32 ve 33 Bayt'larındaki sözcük değeri, üst sınır değerin aşıldığını gösteren 16#0007'dir.

Bir teşhis alarmı olayının AINFO bilgilerini kaydederek bu şekilde teşhis olayının nedenini belirleyebilirsiniz.

15.19 Bir CPU'nun yedeklenmesi ve geri yüklenmesi

15.19.1 Yedekleme ve geri yüklemeyle ilgili opsiyonlar

Zaman içinde otomasyon sisteminizde çok sayıda değişiklik yapabilir, örneğin yeni cihazlar ekleyebilir, mevcut cihazları değiştirebilir veya kullanıcı programını uyarlayabilirsiniz. Bir yedek kopyası oluşturduysanız, bu değişikliklerin istenmeyen davranışlara yol açması halinde, otomasyon sistemi için daha önceki bir sürümü geri yükleyebilirsiniz. STEP 7 ve S7-1200 CPU'su, donanım konfigürasyonu ve yazılımın yedeklenmesi ve geri yüklenmesiyle ilgili çeşitli opsiyonlar sunar.

Yedeklemeyle ilgili opsiyonlar

Aşağıdaki tabloda, S7 CPU'larının yedeklenmesi ve geri yüklenmesiyle ilgili opsiyonlara bir genel bakış sunulmaktadır:

	İzleme değerlerinin anlık kaydı	**Cihazdan yükle (yazılım)**	**Cihazı yeni istasyon olarak yükle (donanım ve yazılım)**	**Yedek kopyasını çevrimiçi cihazdan yükle**
Uygulama durumu	Veri bloğunun özel bir durumunun geri yüklenmesi. Zaman notu da dahil olmak üzere, veri bloklarının gerçek değerleri projede kabul edilir.	Blokların bir CPU'dan projeye yüklenmesi.	Donanım konfigürasyonunun ve yazılımın bir cihazdan projeye yüklenmesi.	Bir CPU'nun tam bir yedek kopyasının geri yükleme noktası olarak oluşturulması. Yedek kopyası tutarlıdır ve değiştirilemez veya açılamaz.
Ön koşul	CPU bir projede mevcuttur. Veri blokları çevrimiçi ve çevrimdışı olarak aynı olmalıdır.	CPU projede mevcuttur.	Cihaz, TIA Portal'ın donanım kataloğunda mevcuttur. Gereken tüm HSP'ler veya GSD dosyaları kuruldu.	-
Mümkün olduğu işletim durumu	RUN, STOP	RUN, STOP	RUN, STOP	STOP
F-CPU'lar için mümkün	Evet	Evet	Hayır	Evet
Yedek kopyası düzenlenebilir	Evet	Evet	Evet	Hayır

Yedek kopyasının içerikleri

Aşağıdaki tabloda, hangi verileri hangi opsiyonlarla yükleyebileceğiniz ve yedekleyebileceğiniz gösterilmektedir:

	İzleme değerlerinin anlık kaydı	**Cihazdan yükle (yazılım)**	**Cihazı yeni istasyon olarak yükle (donanım ve yazılım)**	**Yedek kopyasını çevrimiçi cihazdan yükle**
Veri bloklarının gerçek değerleri	Anlık kayıt mümkündür	Yükleme mümkündür	Yükleme mümkündür	Yedekleme mümkündür
Yazılım modülleri	-	Yükleme mümkündür	Yükleme mümkündür	Yedekleme mümkündür
PLC değişkenleri (değişkenlerin ve sabitlerin adları)	-	Yükleme mümkündür	Yükleme mümkündür	Yedekleme mümkündür
Teknoloji nesneleri	-	Yükleme mümkündür	Yükleme mümkündür	Yedekleme mümkündür
Donanım konfigürasyonu	-	-	Yükleme mümkündür	Yedekleme mümkündür
İzleme tabloları (Web sunucusu)	-	-	Yükleme mümkün değildir	Yedekleme mümkündür
Yerel veriler, markörler, zamanlar, sayaçlar ve proses görünümü	Anlık kayıt mümkün değildir	Yükleme mümkün değildir	Yükleme mümkün değildir	Yedekleme mümkündür
Arşiv ve reçeteler (PLC)	-	-	-	Yedekleme mümkündür
SIMATIC memory card'daki genel bilgiler, örneğin program blokları veya GSD dosyaları ile ilgili yardım	-	-	-	Yedekleme mümkündür

Gerçek değerlerin yedeklenmesiyle ilgili özel notlar

"Çevrimiçi cihazdan yedeklemeyi yükle" tipinde yedekleme, kalıcı olarak tanımlanmış değişkenlerin gerçek değerlerini yedekler. Verilerin tutarlılığını sağlamak için, yedekleme işlemi sırasında kalıcı verilere tüm yazma erişimlerini devre dışı bırakın.

STOP işletim durumundan RUN işletim durumuna geçiş, kalıcı olmayan verilerin gerçek değerlerini başlangıç değerlerine ayarlar. Bir CPU yedek kopyası sadece kalıcı olmayan verilerin başlangıç değerlerini içerir.

15.19.2 Bir çevrimiçi CPU'nun yedeklemesi

Konfigürasyonunuzun bir yedek kopyasını oluşturmak, daha ileride belirli bir konfigürasyona dönmek isteyebileceğiniz durumlar için mantıklı olabilir. Bu durumda güncel konfigürasyonu geri yükleyebilirsiniz.

Ön koşul

Bu sayede, istediğiniz kadar yedek kopyası oluşturabilir ve bir CPU için çok sayıda konfigürasyonu kaydedebilirsiniz. Bir yedek kopyası oluşturmak için şu ön koşullar sağlanmış olmalıdır:

- CPU'yu STEP 7 projesinde oluşturdunuz.
- CPU'yu, CPU'nun PROFINET arayüzü üzerinden doğrudan programlama cihazına/PC'ye bağladınız. Yedekleme ve geri yükleme işlemleri, CM'lerin PROFIBUS arayüzlerini desteklemez.
- CPU çevrimiçidir. (Çevrimiçi bağlantı yoksa, yedekleme işlemi tarafından bir çevrimiçi bağlantı kurulur.)
- CPU, STOP işletim durumundadır. (CPU, STOP işletim durumunda değilse, yedekleme işlemi, CPU'nun STOP işletim durumuna ayarlanabileceğini onaylamanızı talep eder.)

Prosedür

Bir CPU'nun güncel konfigürasyonundan bir yedek kopyası oluşturmak için aşağıdaki yöntemi izleyin:

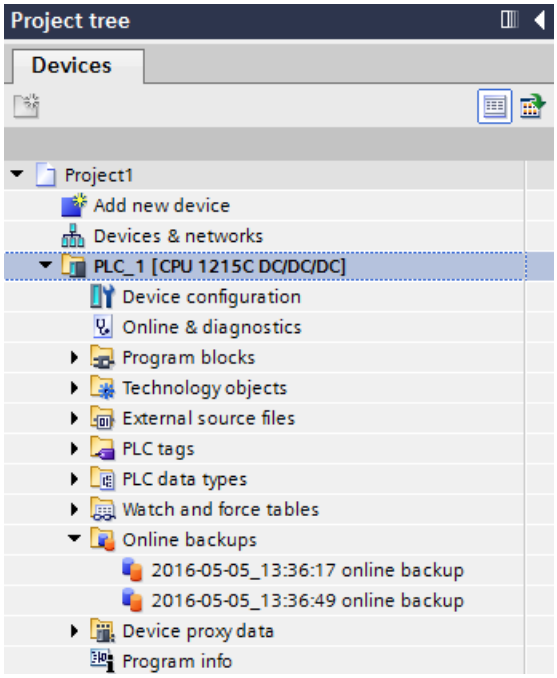
1. Proje gezgininde CPU'yu seçin.
2. "Çevrimiçi" menüsünde "Çevrimiçi cihazdan yedeklemeyi yükle" komutunu seçin.
   CPU'ya okuma erişimi için tanımlanan şifreyi girmeniz ve CPU'nun STOP işletim durumuna geçmesi gerektiğini onaylamanız gerekebilir.

Sonuç

Yedek kopyaları, CPU'nun adı ve yedek kopyasının saati ve tarihi aracılığıyla adlandırılır. Yedek kopyası, bir CPU'nun belirli bir konfigürasyonunu geri yüklemek için gereken tüm verileri kapsar. CPU şu verileri yedekler:

- Mevcutsa memory card'ın içerikleri, aksi takdirde dahili yükleme belleğinin içerikleri
- Veri blokları, sayaçlar ve markörlerin kalıcı bellek alanları
- IP adres parametreleri gibi diğer kalıcı bellek içerikleri

Yedek, CPU'nun gerçek değerlerini içerir, ancak teşhis tamponunu içermez.
Yedek, gizli PLC konfigürasyon verilerini koruma şifresini içermez.
Yedek kopyasını proje gezgininde, CPU'nun altındaki "Çevrimiçi yedeklemeler" klasöründe bulabilirsiniz. Aşağıdaki şekilde, kendisi için iki yedek kopyası oluşturulan bir S7-1200 CPU'su gösterilmektedir:



NOT

Çevrimiçi CPU'yu, SIMATIC Automation Tool (SAT) veya Web sunucusunun "Çevrimiçi yedekleme" standart web sitesi (Sayfa 830) aracılığıyla da yedekleyebileceğinize dikkat edin.

Dosyaları STEP 7 üzerinden yedeklemeniz durumunda, STEP 7, bu dosyaları STEP 7 projesine saklar. Web sunucusu üzerinden dosyaları yedeklediğinizde, PC'niz veya cihazınız yedekleme dosyalarını standart olarak İndirilenler klasörüne saklar. STEP 7 yedekleme dosyalarını Web sunucusu üzerinden ve Web sunucusu yedekleme dosyalarını STEP 7 üzerinden geri yükleyemezsiniz. STEP 7 yedekleme dosyalarını, doğrudan PC'nizin veya cihazınızın indirme klasörüne saklayabilirsiniz. Bu yöntemi kullandığınızda, bu dosyaları Web sunucusundan geri yükleyebilirsiniz.

Yedekleme dosyalarının PC'nize veya cihazınıza kaydedilmesi

Bir yedekleme dosyasını PC'nize veya cihazınıza kaydetmek için aşağıdaki yöntemi izleyin:

1. Proje gezgininde farenin sağ tuşuyla Çevrimiçi yedeklemeler klasöründeki bir dosyaya tıklayın.
2. Bağlam menüsünde "Farklı kaydet" komutunu seçin.
3. Dosyayı kaydetmek istediğiniz klasöre, örneğin PC'nizdeki veya cihazınızdaki indirmeler için kullanılan standart klasöre gidin.
4. "Kaydet" butonuna tıklayın.

15.19.3 Bir CPU'nun geri yüklenmesi

Bir CPU'nun konfigürasyonunu daha önce yedeklediyseniz, bu yedek kopyasını CPU'ya aktarabilirsiniz. CPU, bir yedek geri yüklenirken STOP işletim durumuna geçer. CPU için bir erişim kademesi yapılandırıldıysa, CPU'ya okuma erişimi için kullanılan şifreyi girmeniz gerekir.
Yedek, gizli PLC konfigürasyon verilerini koruma şifresini içermez.

⚠ İKAZ
İçeriği bilinmeyen yedekleme kopyalarının geri yüklenmesi İçeriği bilinmeyen bir yedekleme kopyasını geri yüklemeniz, hatalı fonksiyonlarda veya program hatalarında ciddi maddi hasarlara veya kişisel yaralanmalara neden olabilir. Yedekleme kopyasının, bilinen bir konfigürasyon içerdiğinden emin olun.

Bir yedeğin, gizli PLC konfigürasyon verilerinin korunduğu bir CPU'da geri yüklenmesi

CPU'nuz gizli PLC konfigürasyon verilerini koruma (Sayfa 154) özelliğine sahipse, gizli PLC konfigürasyon verilerini korumak için yedek dosyasındaki yapılandırılmış şifrenin CPU'daki şifreyle aynı olduğundan emin olmanız gerekir.
Şifreler aynı değilse, CPU, RUN işletim durumuna geçemez.
Gizli PLC konfigürasyon verilerini koruma şifresi CPU'daki şifreden farklı olan bir yedek dosyasını geri yüklemeye çalışırsanız, geri yükleme işlemi başarısız olur. Fakat CPU'daki gizli PLC konfigürasyon verilerinin koruması ile CPU'da geri yüklediğiniz projenin koruması aynı olmadığı için, hata durumunda CPU yeniden başlatılır.
Bu durumda, CPU'daki gizli PLC konfigürasyon verilerinin korumasını, geri yüklenen projedekiyle aynı olacak şekilde ayarlamanız gerekir. CPU'daki gizli PLC konfigürasyon verilerini koruma şifresini burada belirleyebilir veya silebilirsiniz:

- SIMATIC Automation Tool, V4.0 SP3'ten itibaren
- TIA Portal, Çevrimiçi ve teşhis (Sayfa 1121)
- SIMATIC memory card (Sayfa 122)

Ön koşul

Bir yedek kopyasını geri yüklemek için şu ön koşullar sağlanmış olmalıdır:

- STEP 7 projesi, CPU ve daha önce oluşturulan bir yedek kopyası için kullanılan bir konfigürasyonu kapsıyor.
- CPU, CPU'nun PROFINET arayüzü üzerinden doğrudan programlama cihazına bağlıdır.
- CPU, STOP işletim durumundadır.
- Bir erişim kademesi (Sayfa 156) yapılandırıldıysa, CPU'ya tam erişim için kullanılan şifreyi biliyorsunuz.

Prosedür

Bir yedek kopyasını geri yüklemek için aşağıdaki yöntemi izleyin:

1. Alt seviyedeki nesneleri görüntülemek için CPU'yu proje gezgininde açın.
2. "Çevrimiçi yedeklemeler" klasöründe, geri yüklemek istediğiniz yedek kopyasını seçin.
3. "Çevrimiçi" menüsünde "Cihaza yükle" komutunu seçin.
   - Önceden bir çevrimiçi bağlantı oluşturduysanız (Sayfa 1117), "Önizlemeyi yükle" iletişim alanı görüntülenir. Bu iletişim alanında alarmlar görüntülenir ve yükleme işlemiyle ilgili önlemler önerilir.
   - Önceden bir çevrimiçi bağlantı oluşturmadıysanız, "Cihaza gelişmiş yükleme" iletişim alanı görüntülenir ve önce CPU ile çevrimiçi bağlantı kurmak istediğiniz arayüzü seçmeniz gerekir.
4. "Önizlemeyi yükle" iletişim alanındaki alarmları kontrol edin ve gerekirse "Eylem" sütunundaki önlemleri seçin.
5. "Yükle" butonuna tıklayın. ("Yükle" butonu, yükleme işlemi mümkün hale geldiğinde seçilebilir.)
6. STEP 7, yedeği CPU'ya geri yükler. "Yükleme işleminin sonuçları" iletişim alanında, yükleme işleminin başarılı olup olmadığını kontrol edebilir ve gerekiyorsa başka önlemler alabilirsiniz.
7. "Yükleme işleminin sonuçları" iletişim alanında kontrolleri gerçekleştirdikten sonra "Tamamla" butonuna tıklayın.
   İstenirse, CPU'ya tam erişim için gereken şifreyi girin ve CPU'nun STOP işletim durumuna geçmesi gerektiğini onaylayın.
   STEP 7, yedek kopyasının içeriğini CPU'ya geri yükler ve CPU'yu yeniden başlatır.

NOT

Bir CPU yedek kopyasını, Web sunucusunun "Çevrimiçi yedekleme" standart web sitesi (Sayfa 830) aracılığıyla da geri yükleyebileceğinize dikkat edin.

Teknik veriler

A

A.1 Çevrimiçi destek için Siemens web sitesi

Bu ürünlerle ilgili teknik bilgileri, Siemens Industry Online Support (https://support.industry.siemens.com/cs/us/en/) web sitesinde bulabilirsiniz.

A.2 Genel teknik veriler

Karşılanan standartlar

S7-1200 otomasyon sisteminin yapısı, aşağıdaki standartlara ve kontrol yönetmeliklerine uygundur. S7-1200 için kontrol kriterleri, bu standartlara ve kontrol yönetmeliklerine dayanmaktadır.

S7-1200 varyantlarının tamamının bu standartlar uyarınca sertifikalı olmayabileceğini ve sertifika durumunun önceden bildirilmeksizin değiştirilebileceğini dikkate alın. Ürüne iliştirilmiş ruhsatlara bakarak geçerli sertifikaları öğrenmek, kullanıcının sorumluluğundadır. Belli bir sipariş numarası için geçerli ruhsatların bir listesine gerek duyuyorsanız, Siemens temsilcinizle iletişime geçebilirsiniz.

CE ruhsatı

CE

S7-1200 otomasyon sistemi, aşağıdaki AB direktiflerinin gerekliliklerini ve güvenlikle ilgili hedeflerini karşılar ve AB Resmi Gazetelerinde yer alan programlanabilir kontrol sistemleri için uyumlaştırılmış Avrupa standartlarına (EN) uygundur.

- 2014/35/AB "Belirli gerilim sınırları dahilinde kullanım için elektrikli donanımlar" (Alçak gerilim direktifi)
- 2014/30/AB "Elektromanyetik uyumluluk" (EMU direktifi)
- 2014/34/AB "Patlama tehlikesi bulunan ortamlarda kullanım amacına yönelik cihazlar ve koruyucu sistemler" (Patlamaya karşı koruma direktifi)
- 2011/65/AT "Belirli tehlikeli maddelerin elektrikli ve elektronik cihazlarda kullanımının sınırlanması" (RoHS direktifi)

İlgili resmi makamlar için gerekli olan AB Uygunluk Beyanlarına şuradan ulaşılabilir:
Siemens AG
Digital Industries
Factory Automation
Postfach 1963
D-92209 Amberg

Ayrıca AB Uygunluk Beyanlarını, Siemens Industry Online Support'un web sitesinde "Uygunluk beyanı" anahtar sözcüğünü aratarak da indirebilirsiniz.

UKCA Uygunluk İşareti

S7-1200 otomasyon sistemi, Britanya Hükümetinin resmi konsolide listesinde yayınlanan programlanabilir kontrol sistemleri için belirlenmiş Britanya Standartlarına (BS) uygundur. S7-1200 otomasyon sistemi, aşağıdaki yönetmeliklerin ve ilişkili eklerin gerekliliklerini ve koruma hedeflerini karşılar:

- Elektrikli ekipmanlar yönetmeliği (Güvenlik) 2016 (alçak gerilim)
- Elektromanyetik uyumluluk yönetmelikleri (EMU) 2016
- Patlama tehlikesi bulunan ortamlarda kullanım amacına yönelik cihazlar ve koruyucu sistemler 2016 (Patlamaya karşı koruma yönetmelikleri)
- Belirli tehlikeli maddelerin elektrikli ve elektronik cihazlarda kullanımının sınırlanması yönetmeliği 2012 (RoHS)

İlgili resmi makamlar için gerekli olan Birleşik Krallık Uygunluk Beyanlarına şuradan ulaşılabilir:
Siemens AG
Digital Industries
Factory Automation
Postfach 1963
D-92209 Amberg
Ayrıca Birleşik Krallık Uygunluk Beyanlarını, Siemens Industry Online Support'un web sitesinde "Uygunluk beyanı" anahtar sözcüğünü aratarak da indirebilirsiniz.

cULus ruhsatı

Underwriters Laboratories Inc.; şunlarla uyumludur:

- Underwriters Laboratories, Inc.: UL 508 Listed (Endüstriyel kontrol cihazları)
- Canadian Standards Association: CSA C22.2 Sayı 142 (Proses kontrol cihazları)

NOT

SIMATIC S7-1200 ürün serisi, CSA standardı ile uyumludur.

cULus işareti, S7-1200'ün Underwriters Laboratories (UL) tarafından UL 508 ve CSA 22.2 No. 142 standartları uyarınca göre denetlendiğini ve ruhsatlandırıldığını belirtir.

cULus HAZ. LOC. Ruhsat



Underwriters Laboratories Inc.; uyumlu olduğu standartlar:

- ANSI/ISA 12.12.01
- CAN/CSA C22.2 No. 213 (patlama tehlikesi bulunan ortamlar) Class I, Division 2, Group A, B, C, D T3C'de kullanım için RUHSATLI; Class I, Zone 2, Group IIC T3 cULus haz.loc için montaj talimatları.
- UYARI - Patlama tehlikesi - Ortamın tehlikesiz olduğu bilinmediği sürece elektrik devresinde akım varsa ilgili bağlantıyı kesmeyin.
- UYARI - Patlama tehlikesi - Bileşenlerin değiştirilmesi Class I, Division 2 ve Zone 2 için uygunluğu olumsuz etkileyebilir.
- Bu cihaz; Class I, Division 2, Group A, B, C, D; Class I, Zone 2, Group IIC veya Patlama tehlikesi bulunmayan ortamlarda kullanım için uygundur.

ÖNEMLİ İSTİSNA: Aynı anda açılabilecek giriş ve çıkışların sayısını, teknik veriler bölümünde bulabilirsiniz. Bazı modeller, Ta = 60 °C'ye düşürülmüştür.

FM sertifikalandırması



FM ruhsatları
Sertifikasyon standardı Sınıf numarası 3600, 3611 (ANSI/UL 121201), 3810 (ANSI/UL 61010-1),
CSA standardı C22.2 No. 0-10, C22.2 No. 213, C22.2 No. 61010-1
Şu ortamlarda kullanım için ruhsatlandırılmıştır:
Class I, Division 2, Gas Group A, B, C, D, Temperature Class T3C Ta = 60 °C [CA, US]
Class I, Zone 2, Group IIC, Temperature Class T3 Ta = 60 °C [US]
Canadian Class I, Zone 2; CEC 18-150 [CA] uyarınca kurulum
ÖNEMLİ İSTİSNA: Aynı anda açılabilecek giriş ve çıkışların sayısını, teknik veriler bölümünde bulabilirsiniz. Bazı modeller, Ta = 60 °C'ye düşürülmüştür.

⚠ İKAZ

Bileşenlerin değiştirilmesi Class I, Division 2 ve Zone 2 için uygunluğu olumsuz etkileyebilir.

Cihazların onarımının yalnızca yetkili bir Siemens Service Center tarafından gerçekleştirilmesine izin verilmektedir.

ATEX ruhsatı

EN 60079-7 uyarınca (Patlama tehlikesi bulunan ortamlar için elektrikli cihazlar - Bölüm 7: Artırılmış güvenlik "e") ve EN IEC 60079-0 (gaz patlama tehlikesi bulunan ortamlar için elektrikli cihazlar - Bölüm 0: Genel gereklilikler)

II 3 G Ex ec IIC T3 Gc

UL 21 ATEX 2622X

Özel kullanım koşulları:

1. IEC 60664-1 uyarınca kirlenme derecesi 2, cihazın kullanım alanında izin verilen maksimum değerdir.
2. Cihaz, EN 60079-7 uyarınca en azından IP54 koruma türü sağlayan bir muhafaza içine kurulmalıdır.
3. Gerilim pik değerlerine karşı koruma, cihazın besleme klemenslerindeki ölçüm pik geriliminin % 140'ını aşmayan bir seviyede sağlanmalıdır.

IECEx ruhsatı

IEC 60079-7 uyarınca (Patlama tehlikesi bulunan ortamlar - Bölüm 7: Artırılmış güvenlik "e" ile cihaz koruması) ve EN IEC 60079-0 (patlama tehlikesi bulunan ortamlar - Bölüm 0: Donanım - Genel gereklilikler)

Ex ec IIC T3 Gc

IECEx UL 21.0107X

Özel kullanım koşulları:

1. IEC 60664-1 uyarınca cihazın yalnızca maksimum kirlenme derecesi 2 olan bir alanda kullanılmasına izin verilir.
2. Cihaz, IEC 60079-7 uyarınca en azından IP54 koruma türü sağlayan bir muhafaza içine kurulmalıdır.
3. Gerilim pik değerlerine karşı koruma, cihazın besleme klemenslerindeki ölçüm pik geriliminin % 140'ını aşmayan bir seviyede sağlanmalıdır.

UKEx ruhsatı

EN 60079-7 uyarınca (Patlama tehlikesi bulunan ortam - Bölüm 7: Artırılmış güvenlik "e" ile cihaz koruması) ve EN IEC 60079-0 (patlama tehlikesi bulunan ortamlar - Bölüm 0: Donanım - Genel gereklilikler)

II 3 G Ex ec IIC T3 Gc

UL 21UKEX2277X

Özel kullanım koşulları:

1. IEC 60664-1 uyarınca kirlenme derecesi 2, cihazın kullanım alanında izin verilen maksimum değerdir.
2. Cihaz, EN 60079-7 uyarınca en azından IP54 koruma türü sağlayan bir muhafaza içine kurulmalıdır.
3. Gerilim pik değerlerine karşı koruma, cihazın besleme klemenslerindeki ölçüm pik geriliminin % 140'ını aşmayan bir seviyede sağlanmalıdır.

CCCEx ruhsatı



GB 3836.3 uyarınca (Patlayıcı ortam - Bölüm 3: Artırılmış güvenlik "e" ile cihaz koruması)
GB 3836.1 uyarınca (Patlayıcı ortam - Bölüm 1: Cihazlar - Genel gereklilikler)
Ex ec IIC T3 Gc
Güvenli kullanım için özel koşullar:

- GB/T 16935.1 uyarınca cihazın yalnızca maksimum kirlenme derecesi 2 olan bir alanda kullanılmasına izin verilir.
- Cihaz, GB 3836.3 uyarınca en azından IP54 koruma türü sağlayan bir muhafaza içine kurulmalıdır.
- Gerilim pik değerlerine karşı koruma, cihazın besleme klemenslerindeki ölçüm pik geriliminin % 140'ını aşmayan bir seviyede sağlanmalıdır.

Avustralya ve Yeni Zelanda - RCM Mark (Regulatory Compliance Mark)



S7-1200 otomasyon sistemi, AS/NZS 61000.6.4 ve IEC 610000-6-4 (Sınıf A) uyarınca standartların gerekliliklerini karşılar.

Kore sertifikalandırması



S7-1200 otomasyon sistemi, Kore Sertifikasyonu (KC işareti) gerekliliklerini karşılamaktadır. A Sınıfı bir cihaz olarak sınıflandırılmış olup, konutta kullanım yerine endüstriyel uygulamalar için tasarlanmıştır.

Avrasya Gümrük Birliği (Belarus, Kazakistan, Rusya Federasyonu) için ruhsat



EAC (Avrasya Uygunluğu): Gümrük Birliği'nin teknik yönetmeliklerine göre uygunluk beyanı (Technical Regulation of Customs Union, TR CU)

Denizcilik sektörü için ruhsat

S7-1200 ürünleri için, belirli pazarlara ilişkin olarak ruhsatlar ile ilgili ve belirli resmi makamlara yönelik düzenli olarak başvurular yapılmaktadır. Belli bir sipariş numarası için geçerli ruhsatların bir listesine gerek duyuyorsanız, Siemens temsilcinizle iletişime geçebilirsiniz.

Sınıflandırma kuruluşları:

- American Bureau of Shipping (ABS): ABD
- Bureau Veritas (BV): Fransa
- Det Norske Veritas (DNV): Norveç
- Germanischer Lloyd (GL): Almanya
- Lloyds Register of Shipping (LRS): İngiltere
- Nippon Kaiji Kyokai (ClassNK): Japonya
- Korean Register of Shipping: Kore
- China Classification Society (CSS): Çin

Endüstriyel ortamlar

S7-1200 otomasyon sistemi, endüstriyel ortamlarda kullanılmak üzere tasarlanmıştır.

Tablo A-1 Endüstriyel ortamlar

Kullanım alanı	Parazit yayımına yönelik gereklilikler	Parazit gidermeye yönelik gereklilikler
Endüstri	EN 61000-6-4	EN 61000-6-2

Elektromanyetik uyumluluk (EMU)

Bir elektrikli cihazın elektromanyetik uyumluluğu (EMU), bu cihazın elektromanyetik bir ortamda kullanım amacına uygun olarak işletimi ve çevredeki diğer elektrikli cihazların çalışmasını olumsuz etkileyebilecek elektromanyetik parazitler yaymama yeteneğidir.

Tablo A-2 Parazite dayanıklılık EN 61000-6-2

Elektromanyetik uyumluluk - EN 61000-6-2 uyarınca parazit giderme	
EN 61000-4-2 Elektrostatik deşarj	Tüm yüzeylerde hava üzerinden 8 kV deşarj Açıktaki iletken yüzeylerle temas üzerinden 6 kV deşarj
EN 61000-4-3 Yüksek frekanslı elektromanyetik alanlara karşı parazite dayanıklılığın test edilmesi	80 ila 1000 MHz, 10 V/m, % 80 AM (1 kHz) 1,4 ila 6 GHz, 3 V/m, % 80 AM (1 kHz)
EN 61000-4-4 Hızlı geçici parazitler	2 kV, 5 kHz AC ve DC sistem gerilimine bağlantı arayüzü ağında 2 kV, 5 kHz giriş/çıkışlara bağlantı arayüzü klemensinde
EN 6100-4-5 Dalgalanma dayanımı	AC sistemleri - 2 kV ortak mod, 1 kV diferansiyel mod DC sistemleri - 2 kV ortak mod, 1 kV diferansiyel mod DC sistemleri için aşağıdaki "Dalgalanma dayanımı" bölümüne bakın.

Elektromanyetik uyumluluk - EN 61000-6-2 uyarınca parazit giderme	
EN 61000-4-6 Hat üzerinde taşınan parazitler	150 kHz ila 80 MHz, 10 V efektif, 1 kHz'te % 80 AM
EN 61000-4-11 Gerilim düşümleri	AC sistemleri 60 Hz'de 1 çevrim için % 0, 12 çevrim için % 40 ve 30 çevrim için % 70

Tablo A-3 EN 61000-6-4 uyarınca hat üzerinde taşınan parazit yayımları

Elektromanyetik uyumluluk - EN 61000-6-4 uyarınca hat üzerinde taşınan parazit yayımları		
Hat üzerinde taşınan parazit yayımları EN 55016, Sınıf A, Grup 1	0,15 ila 0,5 MHz	<79 dB (µV) sözde pik; <66 dB (µV) ortalama değer
	0,5 MHz ila 5 MHz	<73 dB (µV) sözde pik; <60 dB (µV) ortalama değer
	5 MHz ila 30 MHz	<73 dB (µV) sözde pik; <60 dB (µV) ortalama değer
Parazit yayımı EN 55016, Sınıf A, Grup 1	30 MHz ila 230 MHz	< 40dB (µV/m) sözde pik; 10 m'lik bir mesafede ölçülmüş
	230 MHz ila 1 GHz	< 47dB (µV/m) sözde pik; 10 m'lik bir mesafede ölçülmüş
	1 GHz ila 3 GHz	< 76dB (µV/m) sözde pik; 10 m'lik bir mesafede ölçülmüş

Dalgalanma dayanımı

Yıldırımdan etkilenen kablaj sistemleri harici koruma ile donatılmalıdır. Yıldırım darbe gerilimlerine karşı korumaya yönelik değerlendirme için ilgili spesifikasyonu EN 61000-4-5 standardında bulabilirsiniz. İşletim sınırları, EN 61000-6-2'de tanımlanmıştır. S7-1200 doğru akım CPU'lar ve sinyal modülleri, bu standarda göre şok gerilimlere maruz kaldıklarında, güvenli işletimi sürdürmek için harici korumaya gerek duyarlar.

Aşağıda gerekli dalgalanma dayanımını sağlayan bazı cihazlar listelenmiştir. Bu cihazlar, yalnızca üreticinin önerilerini uygun olarak göre doğru şekilde kurulduğunda koruma sağlar. Bu bağlamda aynı veya daha iyi teknik verilere sahip diğer üreticilerin cihazları da kullanılabilir:

Tablo A-4 Dalgalanma dayanımını destekleyen cihazlar

Alt sistem	**Koruyucu cihaz**
Güç kaynağı +24 V DC	BLITZDUCTOR VT, BVT AVD 24, Parça numarası 918 422
Industrial Ethernet	DEHNpatch DPA M CLE RJ45B 48, Parça numarası 929 121
RS-485	BLITZDUCTOR XT, Temel ünite BXT BAS, Parça numarası 920 300
	BLITZDUCTOR XT, Modül BXT ML2 BD HFS 5, Parça numarası 920 271
RS-232	BLITZDUCTOR XT, Temel ünite BXT BAS, Parça numarası 920 300
	BLITZDUCTOR XT, Modül BXT ML2 BE S 12, Parça numarası 920 222
Dijital girişler +24 V DC	DEHN, Inc., Tip DCO SD2 E 24, Parça numarası 917 988

Alt sistem	Koruyucu cihaz
Dijital çıkışlar ve enkoder beslemesi +24 V DC	DEHN, Inc., Tip DCO SD2 E 24, Parça numarası 917 988
Analog G/Ç	DEHN, Inc., Tip DCO SD2 E 12, Parça numarası 917 987
Röle çıkışları	Koruma gerekmez

Çevresel koşullar

Tablo A-5 Taşıma ve depolama

Çevresel koşullar - taşıma ve depolama	
EN 6006822, Test Bb, kuru ısı ve EN 6006821 Test Ab, soğuk	-40 °C ila +70 °C
EN 60068230, Test Db, nemli ısı	25 °C ila 55 °C, % 95 nem
EN 60068-2-14, Test Na, termal şok	-40 °C ila +70 °C, durma süresi 3 saat, 2 çevrim
EN 60068-2-32 Serbest düşüş	0,3 m, 5 kez, nakliye ambalajında
Atmosferik basınç	1140 ila 660 hPa (-1000 ile 3500 m arası yüksekliğe karşılık gelir)

Tablo A-6 İklimsel ortam koşulları

Ortam koşulları - İklimsel ortam koşulları	
S7-1200 otomasyon sistemi, hava koşullarına dayanıklı olan sabit yerlerde kullanım için uygundur. İşletim koşulları, DIN IEC 60721-3-3 standardı gerekliliklerini temel alır: • Sınıf 3M3 (mekanik gereklilikler) • Sınıf 3K3 (iklimsel gereklilikler)	
Ortam sıcaklıkları (cihazın 25 mm altındaki hava girişi)	-20 °C ila 60 °C C yatay montaj -20 °C ila 50 °C dikey montaj % 95 nem, yoğuşmasız Aksi belirtilmediği sürece
Atmosferik basınç	1140 ila 795 hPa (-1000 ile 2000 m arası yüksekliğe karşılık gelir)
Kirletici madde konsantrasyonu	SO_2: < 0,5 ppm; H_2S: < 0,1 ppm; ilgili LF < %60 yoğuşmasız
	ISA-S71.04 reaktivite derecesi G1, G2, G3
EN 60068-2-14, Test Nb, sıcaklık değişimi	0 °C ila 60 °C
EN 60068227 Mekanik darbe dayanımı	15 G, darbe 11 msn, 3 eksenin her birinde 6 şok
EN 6006826 Sinüs dalgası	Montaj rayına montaj: 3,5 mm / 5-9 Hz, 1 G / 8,4 - 150 Hz Panoya montaj: 5 ila 8,4 Hz arasında 7,0 mm; 8,4 ila 150 Hz arasında 2 G Eksen başına 10 sapma, 1 oktav/dakika

IEC 61131-2 uyarınca kirlenme derecesi ve aşırı gerilim kategorisi

- Kirlenme derecesi 2
- Aşırı gerilim kategorisi: II

Koruma türü

- EN 61131-2 uyarınca Koruma sınıfı II (Toprak hattı gerekmez)

Koruma derecesi

- IP20 mekanik koruma, EN 60529
- Standartlaş bir prob ile saptanan yüksek gerilim ile doğrudan temasa karşı koruma. Toz, kir, su ve çapı < 12,5 mm olan yabancı nesnelere karşı harici koruma gerekir.

Ölçüm gerilimleri

Tablo A-7 Ölçüm gerilimleri

Nominal gerilim	Tolerans
24 V DC	20,4 V DC ila 28,8 V DC
120/230 V AC	85 V AC ila 264 V AC, 47 ila 63 Hz

NOT

Mekanik bir kontak, S7-1200 CPU'ya veya bir dijital genişletme modülüne giden çıkış gerilimini devreye aldığında, dijital çıkışlara yaklaşık 50 mikrosaniye boyunca "1" sinyali gönderilir. Bu, makinenin veya prosesin beklenmedik şekilde çalışmasına neden olarak hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilir. Özellikle kısa darbelere yanıt veren cihazlar ile çalışırken bu olguyu dikkate almalısınız.

Yanlış kutup bağlama koruması

CPU'lar, sinyal modülleri (SM'ler) ve sinyal kartları (SB'ler) için, +24 V DC gerilim beslemesi veya kullanıcı tarafı giriş gerilimi olan tüm klemens çiftlerinde yanlış kutup bağlama koruması mevcuttur. Bununla birlikte, farklı klemens çiftleri ters polarite ile bağlandığı takdirde sistemde hasar oluşması yine de mümkün olmaktadır.

S7-1200 sisteminin 24 V DC giriş portlarından bazıları birbirine bağlıdır; bu sırada, mantıksal bir referans iletken çok sayıda M klemensini bağlar. Örneğin aşağıdaki elektrik devreleri, bilgi formlarında "potansiyeli ayrı değil" olarak belirtilmişlerse birbirlerine bağlıdır: CPU'nun 24 V DC beslemesi, CPU'nun sensör beslemesi, bir SM'nin röle bobini için kullanılan güç girişi ve potansiyeli ayrı olmayan bir analog girişin beslemesi. Potansiyeli ayrı olmayan tüm M klemensleri aynı harici referans potansiyeline bağlanmalıdır.

⚠ İKAZ

Potansiyeli ayrı olmayan M klemenslerini farklı referans potansiyellerine bağlarsanız, bu durum istenmeyen akım akışına neden olur ve bu da hedef sistemin ve bağlı cihazların hasar görmesine veya öngörülemeyen işletim performansına yol açabilir.

Bu direktiflere uyulmaması, hasara veya öngörülemeyen işletime ve bununla birlikte ağır veya hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Bir S7-1200 sisteminde, potansiyeli ayrı olmayan tüm M klemenslerini her zaman aynı referans potansiyeline bağlayın.

DC çıkışları

CPU'lar, sinyal modülleri (SM'ler) ve sinyal kartları (SB'ler) üzerindeki DC gerilimi çıkışları için kısa devreye dayanıklı devreler mevcut değildir.

Bir rölenin kullanım ömrü

Yürütülen örnek testler üzerinden tahmin edilen tipik performans verileri aşağıda verilmiştir. Gerçek performans, ilgili uygulamaya göre şekillenecektir. Yüke uygun harici bir koruma devresi, kontakların kullanım ömrünü uzatır. NC kontaklar, endüktif yük veya lamba yükü altında tipik olarak bir NO kontağın kullanım ömrünün yaklaşık üçte biri kadar bir ömre sahiptir.

Bir harici koruma devresi, kontakların kullanım ömrünü uzatır.

Tablo A-8 Tipik performans verileri

Aktüatör seçimi için veriler				
Termik daimi akım		maks. 2 A		
Kontakların anahtarlama kapasitesi ve kullanım ömrü				
	Ohmik yükte	Gerilim	Akım	Anahtarlama çevrimlerinin sayısı (tipik)
		24 V DC	2,0 A	0,1 milyon
		24 V DC	1,0 A	0,2 milyon
		24 V DC	0,5 A	1,0 milyon
		48 V AC	1,5 A	1,5 milyon

Aktüatör seçimi için veriler				
	Ohmik yükte	60 V AC	1,5 A	1,5 milyon
		120 V AC	2,0 A	1,0 milyon
		120 V AC	1,0 A	1,5 milyon
		120 V AC	0,5 A	2,0 milyon
		230 V AC	2,0 A	1,0 milyon
		230 V AC	1,0 A	1,5 milyon
		230 V AC	0,5 A	2,0 milyon
	Endüktif yükte (şuna göre IEC 947-5-1 DC13/AC15)	Gerilim	Akım	Anahtarlama çevrimlerinin sayısı (tipik)
		24 V DC	2,0 A	0,05 milyon
		24 V DC	1,0 A	0,1 milyon
		24 V DC	0,5 A	0,5 milyon
		24 V AC	1,5 A	1,0 milyon
		48 V AC	1,5 A	1,0 milyon
		60 V AC	1,5 A	1,0 milyon
		120 V AC	2,0 A	0,7 milyon
		120 V AC	1,0 A	1,0 milyon
		120 V AC	0,5 A	1,5 milyon
		230 V AC	2,0 A	0,7 milyon
		230 V AC	1,0 A	1,0 milyon
		230 V AC	0,5 A	1,5 milyon
Bir dijital girişin etkinleştirilmesi		Mümkün		
Devre frekansı				
	Mekanik	Maks. 10 Hz		
	Ohmik yükte	Maks. 1 Hz		
	Endüktif yükte (IEC 947-5-1 DC13/AC15 uyarınca)	Maks. 0,5 Hz		
	Lamba yükünde	Maks. 1 Hz		

Dahili CPU belleğinde saklama

- Kalıcı verilerin ve veri protokolü verilerinin kullanım ömrü: 10 yıl
- Gerilim kesintisi durumunda kalıcı veriler, yazma çevrimi dayanıklılığı: 2 milyon çevrim
- Veri protokolü verileri: Yazma çevrimi dayanıklılığı: 500 milyon veri protokolü girişi

NOT

Veri protokollerinin dahili CPU belleği üzerindeki etkisi

Bir veri protokolünün her yazma işlemi en az 2 KB bellek kullanır. Programınız sıkça daha küçük miktarlarda verileri yazıyorsa, her yazma işleminde en az 2 KB bellek kullanılır. Daha iyi bir uygulama, küçük veri elemanlarının, sonra daha az sıklıkla veri protokolüne yazılacak olan bir veri bloğunda (DB) toplanması şeklindedir.

Programınız çok sayıda veri protokolü girişini çok sık yazıyorsa, bu durumda değiştirilebilir bir SD memory card kullanımını düşünebilirsiniz.

A.3 Bağlantı yerleşimi - PROFINET arayüz portu X1

S7-1200 CPU, PROFINET ağına bir standart RJ45 soketi ile bağlanır. Soketin bağlantı yerleşimi, CPU tipine bağlıdır.

Tek portlu CPU'lar

Tek portlu CPU'lar (CPU 1211C, CPU 1212C ve CPU 1214C) aşağıdaki standart Ethernet MDI bağlantı yerleşimine sahiptir:

Pin	Sinyal adı	Açıklama	RJ45 soketinin bağlantı yerleşimi
1	TD+	Veri gönderimi	 87654321 X1P1
2	TD-		
3	RD+	Veri alımı	
4	GND	Şasi	
5	GND		
6	RD-	Veri alımı	
7	GND	Şasi	
8	GND		

Çift portlu CPU'lar

Çift portlu CPU'lar (CPU 1215C ve CPU1217C) aşağıdaki standart Ethernet MDI-X pin yerleşimine sahiptir:

Pin	Sinyal adı	Açıklama	RJ45 soketinin bağlantı yerleşimi
1	TD+	Veri gönderimi	
2	TD-		
3	RD+	Veri alımı	
4	GND	Şasi	
5	GND		
6	RD-	Veri alımı	
7	GND	Şasi	
8	GND		

NOT

Çift portlu CPU

Pinin altında görüntülenen sayı, çift portlu CPU'larda pinler arasında çapraz geçiş (Crossover) olmadığını gösterir. Cihazlar, dahili bir Ethernet Switch'e sahiptir: TD+/- ve RD+/- çiftleri için dahili olarak çapraz geçiş yoktur.

Otomatik anlaşma

Bağlantı konfigürasyonu otomatik anlaşmaya olanak tanırsa, S7-1200 CPU, kablo tipini otomatik olarak algılar ve gerekirse gönderim/alım hatlarını birbirleriyle değiştirir. Bağlantı konfigürasyonunda otomatik anlaşma aktif değilse, CPU, otomatik değiştirmeyi de devre dışı bırakır. Bir portun otomatik anlaşma ayarı TIA Portal'da bağlantı opsiyonları diyaloğunda yapılandırılır. Bu, CPU özelliklerindeki PROFINET arayüzü (X1) için porta özgü genişletilmiş bir opsiyondur. Bkz. "PROFINET portunun yapılandırılması", alt bölüm 11.2.3.4: Ayrıntılı bilgilerle birlikte "Projenizdeki bir CPU için IP adresinin yapılandırılması" (Sayfa 571)*.

A.4 CPU 1211C

A.4.1 Genel teknik veriler ve performans özellikleri

Tablo A-9 Genel teknik veriler

Teknik veriler	CPU 1211C AC/DC/röle	CPU 1211C DC/DC/röle	CPU 1211C DC/DC/DC
Sipariş numarası	6ES7211-1BE40-0XB0	6ES7211-1HE40-0XB0	6ES7211-1AE40-0XB0
Boyutlar G x Y x D (mm)	90 x 100 x 75		
Nakliye ağırlığı	420 gram	380 gram	370 gram
Performans kaybı	10 W	8 W	
Kullanılabilir elektrik akımı (CM veriyolu)	maks. 750 mA (5 V DC)		
Kullanılabilir elektrik akımı (24 V DC)	maks. 300 mA (enkoder beslemesi)		
Dijital giriş akım tüketimi (24 V DC)	4 mA/giriş		

Tablo A-10 CPU özellikleri

Teknik veriler		Açıklama
Kullanıcı belleği (Bkz. "Genel teknik veriler" (Sayfa 1160), "Dahili CPU belleğinde saklama").	Çalışma belleği	75 KB
	Yükleme belleği	1 MB, dahili, SD kart boyutuna kadar genişletilebilir
	Kalıcı	14 kBayt
Entegre dijital G/Ç		6 giriş/4 çıkış
Entegre analog G/Ç		2 giriş
Proses görünümünün boyutu		1024 Bayt girişler (E)/1024 Bayt çıkışlar (A)
Markör (M)		4096 Bayt
Geçici (yerel) bellek		• Yol alma ve program çevrimi için 16 KB (ilgili FB'ler ve FC'ler de dahil) • Diğer her alarm öncelik seviyesi için 6 KB (FB'ler ve FC'ler dahil)
Ek sinyal modülleri		Yok
Genişletme SB, CB, BB		maks. 1
Ek iletişim modülleri		maks. 3 CM
Hızlı sayaçlar		Herhangi bir entegre veya SB girişiyle kullanım için 6 adede kadar yapılandırılmış. Bkz. CPU 1211C için "Donanım girişinin bağlantı yerleşimi" (Sayfa 533): HSC standart adreslerinin atanması 100/[1]80 kHz (Ea.0 ila Ea.5)
Darbe çıkışları [2]		Herhangi bir entegre veya SB çıkışıyla kullanım için 4 adede kadar yapılandırılmış. 100 kHz (Aa.0 ila Aa.3)
Darbe toplama için girişler		6

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Teknik veriler	Açıklama
Gecikme alarmları	1 msn çözünürlük ile 4
Uyandırma alarmları	1 msn çözünürlük ile 4
Kenar alarmları	6 yükselen ve 6 alçalan (opsiyonel sinyal kartı ile 10 ve 10)
Memory card	SIMATIC memory card (opsiyonel)
Gerçek zaman saati hassasiyeti	+/- 60 saniye/ay
Gerçek zaman saati tamponlama	Tipik 20 gün/min. 12 gün, 40 °C'de (bakım gerektirmeyen yüksek performanslı kondensatör)

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Tablo A-11 Enerji

Yordam türü		Yürütme hızı	
		Direkt adresleme (E, A ve Mi)	DB erişimi
Boole değeri		0,08 µs/yordam	
Kaydırma	Move_Bool	0,3 µs/yordam	1,17 µs/yordam
	Move_Word	0,137 µs/yordam	1,0 µs/yordam
	Move_Real	0,72 µs/yordam	1,0 µs/yordam
Gerçek sayılar aritmetiği	Add Real	1,48 µs/yordam	1,78 µs/yordam

NOT

Ölçülen süreleri etkileyen birçok değişken mevcuttur. Yukarıda belirtilen performans süreleri, bu kategorideki en hızlı yordamlar ve hatasız programlar içindir.

A.4.2 CPU 1211C tarafından desteklenen süreler, sayaçlar ve kod blokları

Tablo A-12 CPU 1211C tarafından desteklenen bloklar, süreler ve sayaçlar

Eleman		Açıklama
Bloklar	Tip	OB, FB, FC, DB
	Boyut	Çalışma belleğinin boyutuna kadar
	FB'ler, FC'ler ve DB'ler için adres aralıkları	FB ve FC: 1 ila 65535 (örn. FB 1 ila FB 65535) DB: 1 ila 59999
	Gömme derinliği	Çevrim veya yol alma OB'lerinden 16 adet Rastgele alarm olayı OB'lerinden 6 adet[1]
	Denetim	2 kod bloğunun durumu eş zamanlı olarak denetlenebilir.
OB'ler	Program çevrimi	Birden çok
	Yol alma	Birden çok
	Gecikme alarmı	4 (olay başına 1)

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Eleman		Açıklama
OB'ler	Uyandırma alarmları	4 (olay başına 1)
	Proses alarmları	50 (olay başına 1)
	Zaman hatası alarmları	1
	Teşhis hatası alarmları	1
	Modüllerin çıkartılması veya takılması	1
	Modül taşıyıcısında veya istasyonda hata	1
	Saat alarmı	Birden çok
	Durum	1
	Güncelleme	1
	Profil	1
	MC-Interpolator	1
	MC-Servo	1
	MC-PreServo	1
	MC-PostServo	1
Zamanlar	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, süre başına 16 Bayt
Sayaç	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, boyut, sayma türüne bağlıdır • SInt, USInt: 3 Bayt • Int, UInt: 6 Bayt • DInt, UDInt: 12 Bayt

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Tablo A-13 İletişim

Teknik veriler	Açıklama
Arayüzler	1
Tip	Ethernet
HMI cihazı	4
Programlama cihazı (PG)	1
Bağlantılar	• Açık kullanıcı iletişimi için 8 bağlantı (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV • GET/PUT verileri için 8 CPU/CPU bağlantısı (istemci veya sunucu) • GET/PUT veya açık kullanıcı iletişimine dinamik atama için 6 bağlantı • Güvenlik sertifikaları için maks. 64 bağlantı

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler	**Açıklama**
Veri hızları	10/100 Mbit/sn
Elektriksel ayırma (lojiğe harici sinyal)	Potansiyeli ayrı dönüştürücü, 1500 V AC (tip kontrolü)[1]
Kablo türü	CAT5e blendajlı
Arayüzler	
PROFINET arayüzü sayısı	1
PROFIBUS arayüzü sayısı	0
Arayüz	
Arayüz için donanım	
Port sayısı	1
Entegre Switch	Hayır
RJ-45 (Ethernet)	Evet; X1
Protokoller	
PROFINET IO kontrolörü	Evet
PROFINET IO cihazı	Evet
SIMATIC iletişimi	Evet
Açık IE iletişimi	Evet
Web sunucusu	Evet
Medya yedekliliği	Hayır
PROFINET IO kontrolörü	
Hizmetler	
PG/OP iletişimi	Evet
S7-Routing	Evet
Adım senkronizasyonlu işletim	Hayır
Açık IE iletişimi	Evet
IRT	Hayır
MRP	Hayır
PROFIenergy	Evet. S7-1200 CPU yalnızca PROFIenergy ünitesini destekler (I-Device fonksiyonlarıyla).
Öncelikli yol alma	Evet (maks. 16 PROFINET cihazı)
Bağlanabilir IO cihazlarının maksimum sayısı	16
RT için maks. bağlanabilir IO cihazların sayısı	16
Bunlardan seri olarak maks. sayı	16
Aynı anda etkinleştirilebilir/devre dışı bırakılabilir IO cihazlarının sayısı	8
Güncelleme süreleri	Güncelleme süresinin minimum değeri; PROFINET IO için ayarlanan iletişim bileşenine, IO cihazlarının sayısına ve yapılandırılmış kullanıcı verilerinin miktarına da bağlıdır.
RT ile	

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler		Açıklama
Gönderim adımı 1 msn		1 msn ila 512 msn
PROFINET IO cihazı		
Hizmetler		
PG/OP iletişimi		Evet
S7-Routing		Evet
Adım senkronizasyonlu işletim		Hayır
Açık IE iletişimi		Evet
IRT, destekleniyor		Hayır
MRP, destekleniyor		Hayır
PROFIenergy		Evet
Shared Device		Evet
Shared Device'lı IO kontrolörlerinin maksimum sayısı		2
SIMATIC iletişimi		
S7 iletişimi, sunucu olarak		Evet
S7 iletişimi, istemci olarak		Evet
Görev başına maks. kullanıcı verisi		Bkz. online yardım (S7 iletişimi, kullanıcı verilerinin boyutu)
Açık IE iletişimi		
TCP/IP:		Evet
	Verilerin maks. uzunluğu	8 KB
	Port başına çoklu pasif bağlantı desteği	Evet
ISO on TCP (RFC1006):		Evet
	Verilerin maks. uzunluğu	8 KB
UDP		Evet
	Verilerin maks. uzunluğu	1472 Bayt
DHCP		Hayır
SNMP		Evet
DCP		Evet
LLDP		Evet

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Tablo A-14 Güç kaynağı

Teknik veriler		**CPU 1211C AC/DC/röle**	**CPU 1211C DC/DC/röle**	**CPU 1211C DC/DC/DC**
Gerilim aralığı		85 ila 264 V AC	20,4 V DC ila 28,8 V DC	
Şebeke frekansı		47 ila 63 Hz	--	
Giriş akımı	yalnızca CPU maksimum yükte	120 V AC'de 60 mA 240 V AC'de 30 mA	24 V DC'de 300 mA	24 V DC'de 300 mA
	Maksimum yükte tüm genişletme modülleriyle birlikte CPU	120 V AC'de 180 mA 240 V AC'de 90 mA	24 V DC'de 900 mA	
Ani akım (maks.)		264 V AC'de 20 A	28,8 V DC'de 12 A	
$I^2 t$		0,8 A^2 sn	0,5 A^2 sn	
Elektriksel ayırma (lojiğe giriş gücü)		1500 V AC	Elektriksel olarak ayrılmamış	
Toprağa kaçak akım, AC hattından fonksiyonel topraklamaya		maks. 0,5 mA	--	
Gecikme süresi (gerilim kaybı)		120 V AC'de 20 msn 240 V AC'de 80 msn	24 V DC'de 10 msn	
Dahili sigorta, kullanıcı tarafından değiştirilemez		3 A, 250 V, yavaş		

Tablo A-15 Enkoder beslemesi

Teknik veriler	**CPU 1211C AC/DC/röle**	**CPU 1211C DC/DC/röle**	**CPU 1211C DC/DC/DC**
Gerilim aralığı	20,4 ila 28,8 V DC	L+ eksi 4 V DC (min.)	
Nominal çıkış akımı (maks.)	300 mA (kısa devreye dayanıklı)		
Maks. dalgalılık/parazit akımları (<10 MHz)	< 1 V pik-pik	Giriş hattı gibi	
Elektriksel ayırma (CPU lojiğinden enkoder beslemesine)	Elektriksel olarak ayrılmamış		

A.4.3 Dijital girişler ve çıkışlar

Tablo A-16 Dijital girişler

Teknik veriler	**CPU 1211C AC/DC/röle, CPU 1211C DC/DC/röle ve CPU 1211C DC/DC/DC**
Girişlerin sayısı	6
Tip	Akım çeken/akım besleyen (IEC Tip 1, akım çeken ise)
Nominal gerilim	24 V DC'de 4 mA, nominal
İzin verilen sürekli gerilim	maks. 30 V DC
Şok gerilim	0,5 sn için 35 V DC
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
Filtre süreleri	µsn ayarları: 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0 msn ayarları: 0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0
HSC giriş adım frekansları (maks.) (Mantıksal seviye 1 = 15 ila 26 V DC)	100/80 kHz (Ea.0 ila Ea.5)
Eş zamanlı devreye alınmış girişlerin sayısı	60 °C yatayda 6, 50 °C dikey
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız, HSC girişleri için 50 m

Tablo A-17 Dijital çıkışlar

Teknik veriler	**CPU 1211C AC/DC/röle ve CPU 1211C DC/DC/röle**	**CPU 1211C DC/DC/DC**
Çıkışlar	4	
Tip	Röle, mekanik	MOSFET, elektronik (akım besleyen)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	--	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	--	maks. 0,1 V DC
Akım (maks.)	2,0 A	0,5 A
Lamba yükü	30 W DC/200 W AC	5 W
AÇIK'taki direnç	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω
Çıkış başına kaçak akım	--	maks. 10 µA
Ani akım	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır	
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobinden lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1	
Endüktif klemens gerilimi	--	L+ eksi 48 V DC, 1 W kayıp güç
Rölenin maksimum devre frekansı	1 Hz	--

[1] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

[2] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Teknik veriler	CPU 1211C AC/DC/röle ve CPU 1211C DC/DC/röle	CPU 1211C DC/DC/DC
Anahtarlama gecikmesi (Aa.0 ila Aa.3)	maks. 10 msn	maks. 1,0 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 3,0 µsn
Darbe jeneratörü frekansı	Önerilmiyor [1]	100 kHz (Aa.0 ila Aa.3)[2], min. 2 Hz
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	--
Nominal yükte kontakların kullanım ömrü	100.000 anahtarlama sayısı açık/kapalı	--
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	
Bir dijital girişin kumandası	Evet	
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)	
Artırılmış yük için paralel çıkışlar	Hayır	
Eş zamanlı devreye alınmış çıkışların sayısı	60 °C yatayda 4, 50 °C dikey	
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız	

[1] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

[2] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

A.4.4 Analog girişler

Tablo A-18 Analog girişler

Teknik veriler	Açıklama
Girişlerin sayısı	2
Tip	Gerilim (tek adımlı giriş)
Tam sapma aralığı	0 ila 10 V
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	10,001 ila 11,759 V
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	11,760 ila 11,852 V
Taşma aralığı (veri sözcüğü)	32512 ila 32767
Çözünürlük	10 Bit
Maks. dayanım gerilimi	35 V DC
Düzleştirme	Yok, zayıf, orta veya güçlü Bkz. CPU'nun analog girişleri için adım yanıtı (msn) (Sayfa 1181) tablosu
Gürültü azaltma	10, 50 veya 60 Hz
Empedans	≥100 kΩ

Teknik veriler	Açıklama
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % 3,0 / % 3,5'i
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

A.4.4.1 CPU'nun entegre analog girişlerinin adım yanıtı

Tablo A-19 Adım yanıtı (msn), 0 V ila 10 V %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	Bastırma frekansı (entegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Yok (1 çevrim): Ortalama alma yok	50 msn	50 msn	100 msn
Zayıf (4 çevrim): 4 tarama değeri	60 msn	70 msn	200 msn
Orta (16 çevrim): 16 tarama değeri	200 msn	240 msn	1150 msn
Güçlü (32 çevrim): 32 tarama değeri	400 msn	480 msn	2300 msn
Tarama süresi	**4,17 msn**	**5 msn**	**25 msn**

A.4.4.2 CPU'nun entegre analog portlarının tarama aralığı

Tablo A-20 CPU'nun entegre analog girişlerinin tarama aralığı

Bastırma frekansı (entegrasyon süresi seçimi)	Tarama süresi
60 Hz (16,6 msn)	4,17 msn
50 Hz (20 msn)	5 msn
10 Hz (100 msn)	25 msn

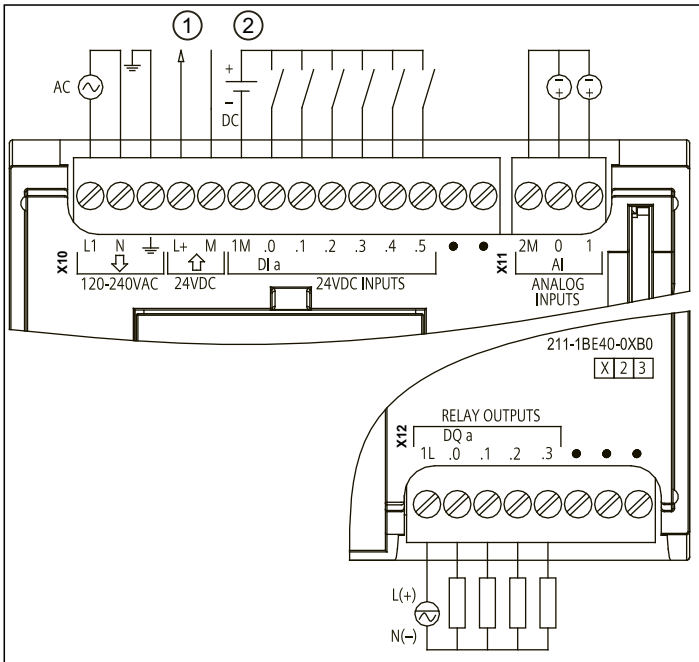
A.4.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar)

Tablo A-21 Gerilim için analog girişi gösterimi (CPU'lar)

Sistem		Gerilim ölçüm aralığı	
Desimal	**Onaltılı**	**0 ila 10 V**	
32767	7FFF	11,852 V	Taşma
32512	7F00		
32511	7EFF	11,759 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

A.4.5 CPU 1211C'nin devre şemaları

Tablo A-22 CPU 1211C AC/DC/röle (6ES7211-1BE40-0XB0)



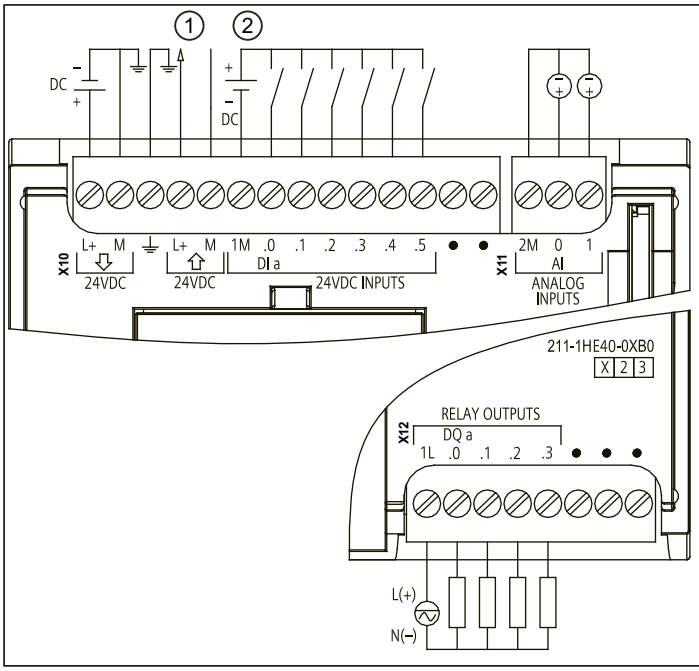
①	24-V-DC enkoder gerilimi Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: İki klemensten biri, L1 veya N (L2) 240 V AC'ye kadar bir gerilim kaynağına bağlanabilir. N klemensi L2 olarak değerlendirilebilir ve topraklanmasına gerek yoktur. L1 ve N (L2) klemensleri için yanlış kutup bağlama koruması gerekli değildir.	
Not 3: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-23 CPU 1211C AC/DC/röle için bağlantı yerleşimi (6ES7211-1BE40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L1 / 120-240 V AC	2 M	1L
2	N / 120-240 V AC	AI 0	DO a.0
3	Fonksiyonel topraklama	AI 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	--	DO a.2
5	M / enkoder beslemesi 24 V DC	--	DO a.3
6	1M	--	Bağlantı yok
7	DI a.0	--	Bağlantı yok
8	DI a.1	--	Bağlantı yok
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--

Pin	X10	X11 (altın kaplamalı)	X12
12	DI a.5	--	--
13	Bağlantı yok	--	--
14	Bağlantı yok	--	--

Tablo A-24 CPU 1211C DC/DC/röle (6ES7211-1HE40-0XB0)

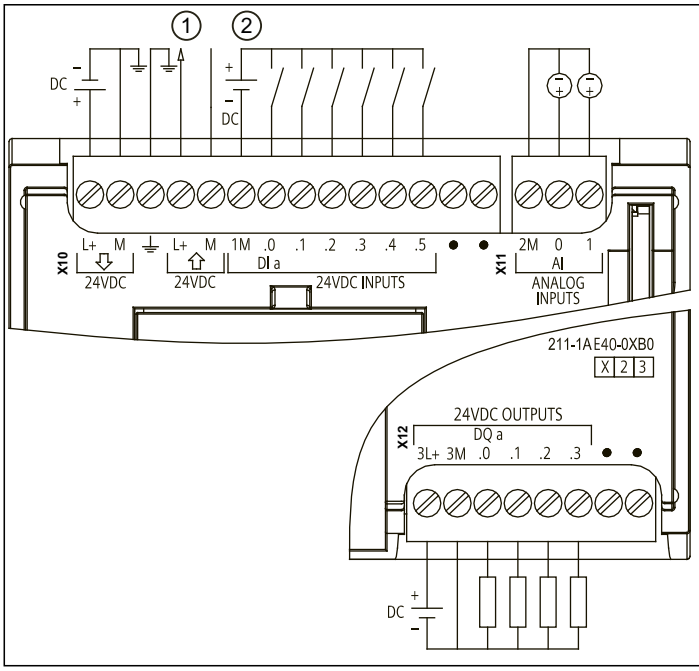


①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-25 CPU 1211C DC/DC/röle için bağlantı yerleşimi (6ES7211-1HE40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	1L
2	M / 24 V DC	AI 0	DO a.0
3	Fonksiyonel topraklama	AI 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	--	DO a.2
5	M / enkoder beslemesi 24 V DC	--	DO a.3
6	1M	--	Bağlantı yok
7	DI a.0	--	Bağlantı yok
8	DI a.1	--	Bağlantı yok
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Bağlantı yok	--	--
14	Bağlantı yok	--	--

Tablo A-26 CPU 1211C DC/DC/DC (6ES7211-1AE40-0XB0)



①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-27 CPU 1211C DC/DC/DC için bağlantı yerleşimi (6ES7211-1AE40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	3L+
2	M / 24 V DC	AI 0	3M
3	Fonksiyonel topraklama	AI 1	DO a.0
4	L+ / enkoder beslemesi 24 V DC	--	DO a.1
5	M / enkoder beslemesi 24 V DC	--	DO a.2
6	1M	--	DO a.3
7	DI a.0	--	Bağlantı yok
8	DI a.1	--	Bağlantı yok
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Bağlantı yok	--	--
14	Bağlantı yok	--	--

NOT

Kullanılmayan analog girişlere kısa devre yapın.

A.5 CPU 1212C

A.5.1 Genel teknik veriler ve performans özellikleri

Tablo A-28 Genel

Teknik veriler	CPU 1212C AC/DC/röle	CPU 1212C DC/DC/röle	CPU 1212C DC/DC/DC
Sipariş numarası	6ES7212-1BE40-0XB0	6ES7212-1HE40-0XB0	6ES7212-1AE40-0XB0
Boyutlar G x Y x D (mm)	90 x 100 x 75		
Nakliye ağırlığı	425 gram	385 gram	370 gram
Performans kaybı	11 W	9 W	
Kullanılabilir elektrik akımı (SM ve CM veriyolu)	maks. 1000 mA (5 V DC)		
Kullanılabilir elektrik akımı (24 V DC)	maks. 300 mA (enkoder beslemesi)		
Dijital giriş akım tüketimi (24 V DC)	4 mA/giriş		

Tablo A-29 CPU özellikleri

Teknik veriler		Açıklama
Kullanıcı belleği (Bkz. "Genel teknik veriler" (Sayfa 1160), "Dahili CPU belleğinde saklama").	Çalışma belleği	100 KB
	Yükleme belleği	2 MB dahili, SD kart boyutuna kadar genişletilebilir
	Kalıcı	14 kBayt
Entegre dijital G/Ç		8 giriş/6 çıkış
Entegre analog G/Ç		2 giriş
Proses görünümünün boyutu		1024 Bayt girişler (E)/1024 Bayt çıkışlar (A)
Markör (M)		4096 Bayt
Geçici (yerel) bellek		• Yol alma ve program çevrimi için 16 KB (ilgili FB'ler ve FC'ler de dahil) • Diğer her alarm öncelik seviyesi için 6 KB (FB'ler ve FC'ler dahil)
Ek sinyal modülleri		maks. 2 SM
Genişletme SB, CB, BB		maks. 1
Ek iletişim modülleri		maks. 3 CM
Hızlı sayaçlar		Herhangi bir entegre veya SB girişiyle kullanım için 6 adede kadar yapılandırılmış. Bkz. CPU 1212C için "Donanım girişinin bağlantı yerleşimi" (Sayfa 533): HSC standart adreslerinin atanması • 100/[1]80 kHz (Ea.0 ila Ea.5) • 30/[1]20 kHz (Ea.6 ila Ea.7)
Darbe çıkışları [2]		Herhangi bir entegre veya SB çıkışıyla kullanım için 4 adede kadar yapılandırılmış. • 100 kHz (Aa.0 ila Aa.3) • 20 kHz (Aa.4 ila Aa.5)
Darbe toplama için girişler		8
Gecikme alarmları		1 msn çözünürlük ile 4

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Teknik veriler	Açıklama
Uyandırma alarmları	1 msn çözünürlük ile 4
Kenar alarmları	8 yükselen ve 8 alçalan (opsiyonel sinyal kartı ile 12 ve 12)
Memory card	SIMATIC memory card (opsiyonel)
Gerçek zaman saati hassasiyeti	+/- 60 saniye/ay
Gerçek zaman saati tamponlama	Tipik 20 gün/min. 12 gün, 40 °C'de (bakım gerektirmeyen yüksek performanslı kondensatör)

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Tablo A-30 Enerji

Yordam türü		Yürütme hızı	
		Direkt adresleme (E, A ve Mi)	DB erişimi
Boole değeri		0,08 µs/yordam	
Kaydırma	Move_Bool	0,3 µs/yordam	1,17 µs/yordam
	Move_Word	0,137 µs/yordam	1,0 µs/yordam
	Move_Real	0,72 µs/yordam	1,0 µs/yordam
Gerçek sayılar aritmetiği	Add Real	1,48 µs/yordam	1,78 µs/yordam

NOT

Ölçülen süreleri etkileyen birçok değişken mevcuttur. Yukarıda belirtilen performans süreleri, bu kategorideki en hızlı yordamlar ve hatasız programlar içindir.

A.5.2 CPU 1212C tarafından desteklenen süreler, sayaçlar ve kod blokları

Tablo A-31 CPU 1212C tarafından desteklenen bloklar, süreler ve sayaçlar

Eleman		Açıklama
Bloklar	Tip	OB, FB, FC, DB
	Boyut	Çalışma belleğinin boyutuna kadar
	FB'ler, FC'ler ve DB'ler için adres aralıkları	FB ve FC: 1 ila 65535 (örn. FB 1 ila FB 65535) DB: 1 ila 59999
	Gömme derinliği	Çevrim veya yol alma OB'lerinden 16 adet Rastgele alarm olayı OB'lerinden 6 adet[1]
	Denetim	2 kod bloğunun durumu eş zamanlı olarak denetlenebilir.
OB'ler	Program çevrimi	Birden çok
	Yol alma	Birden çok
	Gecikme alarmı	4 (olay başına 1)
	Uyandırma alarmları	4 (olay başına 1)

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Eleman		Açıklama
OB'ler	Proses alarmları	50 (olay başına 1)
	Zaman hatası alarmları	1
	Teşhis hatası alarmları	1
	Modüllerin çıkartılması veya takılması	1
	Modül taşıyıcısında veya istasyonda hata	1
	Saat alarmı	Birden çok
	Durum	1
	Güncelleme	1
	Profil	1
	MC-Interpolator	1
	MC-Servo	1
	MC-PreServo	1
	MC-PostServo	1
Zamanlar	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, süre başına 16 Bayt
Sayaç	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, boyut, sayma türüne bağlıdır • SInt, USInt: 3 Bayt • Int, UInt: 6 Bayt • DInt, UDInt: 12 Bayt

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Tablo A-32 İletişim

Teknik veriler	Açıklama
Arayüzler	1
Tip	Ethernet
HMI cihazı	4
Programlama cihazı (PG)	1
Bağlantılar	• Açık kullanıcı iletişimi için 8 bağlantı (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV • GET/PUT verileri için 8 CPU/CPU bağlantısı (istemci veya sunucu) • GET/PUT veya açık kullanıcı iletişimine dinamik atama için 6 bağlantı • Güvenlik sertifikaları için maks. 64 bağlantı
Veri hızları	10/100 Mbit/sn

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler	Açıklama
Elektriksel ayırma (lojiğe harici sinyal)	Potansiyeli ayrı dönüştürücü, 1500 V AC (tip kontrolü)[1]
Kablo türü	CAT5e blendajlı
Arayüzler	
PROFINET arayüzü sayısı	1
PROFIBUS arayüzü sayısı	0
Arayüz	
Arayüz için donanım	
Port sayısı	1
Entegre Switch	Hayır
RJ-45 (Ethernet)	Evet; X1
Protokoller	
PROFINET IO kontrolörü	Evet
PROFINET IO cihazı	Evet
SIMATIC iletişimi	Evet
Açık IE iletişimi	Evet
Web sunucusu	Evet
Medya yedekliliği	Hayır
PROFINET IO kontrolörü	
Hizmetler	
PG/OP iletişimi	Evet
S7-Routing	Evet
Adım senkronizasyonlu işletim	Hayır
Açık IE iletişimi	Evet
IRT	Hayır
MRP	Hayır
PROFIenergy	Evet. S7-1200 CPU yalnızca PROFIenergy ünitesini destekler (I-Device fonksiyonlarıyla).
Öncelikli yol alma	Evet (maks. 16 PROFINET cihazı)
Bağlanabilir IO cihazlarının maksimum sayısı	16
RT için maks. bağlanabilir IO cihazların sayısı	16
Bunlardan seri olarak maks. sayı	16
Aynı anda etkinleştirilebilir/devre dışı bırakılabilir IO cihazlarının sayısı	8
Güncelleme süreleri	Güncelleme süresinin minimum değeri; PROFINET IO için ayarlanan iletişim bileşenine, IO cihazlarının sayısına ve yapılandırılmış kullanıcı verilerinin miktarına da bağlıdır.
RT ile	
Gönderim adımı 1 msn	1 msn ila 512 msn

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler		Açıklama
PROFINET IO cihazı		
Hizmetler		
PG/OP iletişimi		Evet
S7-Routing		Evet
Adım senkronizasyonlu işletim		Hayır
Açık IE iletişimi		Evet
IRT, destekleniyor		Hayır
MRP, destekleniyor		Hayır
PROFIenergy		Evet
Shared Device		Evet
Shared Device'lı IO kontrolörlerinin maksimum sayısı		2
SIMATIC iletişimi		
S7 iletişimi, sunucu olarak		Evet
S7 iletişimi, istemci olarak		Evet
Görev başına maks. kullanıcı verisi		Bkz. online yardım (S7 iletişimi, kullanıcı verilerinin boyutu)
Açık IE iletişimi		
TCP/IP:		Evet
	Verilerin maks. uzunluğu	8 KB
	Port başına çoklu pasif bağlantı desteği	Evet
ISO on TCP (RFC1006):		Evet
	Verilerin maks. uzunluğu	8 KB
UDP		Evet
	Verilerin maks. uzunluğu	1472 Bayt
DHCP		Hayır
SNMP		Evet
DCP		Evet
LLDP		Evet

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Tablo A-33 Güç kaynağı

Teknik veriler		**CPU 1212C AC/DC/röle**	**CPU 1212C DC/DC/röle**	**CPU 1212C DC/DC/DC**
Gerilim aralığı		85 ila 264 V AC	20,4 V DC ila 28,8 V DC	
Şebeke frekansı		47 ila 63 Hz	--	
Ani akım (maks. yük)	Yalnızca CPU	120 V AC'de 80 mA 240 V AC'de 40 mA	24 V DC'de 400 mA	
	Tüm genişletme modülleriyle birlikte CPU	120 V AC'de 240 mA 240 V AC'de 120 mA	24 V DC'de 1200 mA	
Ani akım (maks.)		264 V AC'de 20 A	28,8 V DC'de 12 A	
I^2t		0,8 A^2 sn	0,5 A^2 sn	
Elektriksel ayırma (lojiğe giriş gücü)		1500 V AC	Elektriksel olarak ayrılmamış	
Toprağa kaçak akım, AC hattından fonksiyonel topraklamaya		maks. 0,5 mA	--	
Gecikme süresi (gerilim kaybı)		120 V AC'de 20 msn 240 V AC'de 80 msn	24 V DC'de 10 msn	
Dahili sigorta, kullanıcı tarafından değiştirilemez		3 A, 250 V, yavaş		

Tablo A-34 Enkoder beslemesi

Teknik veriler	**CPU 1212C AC/DC/röle**	**CPU 1212C DC/DC/röle**	**CPU 1212C DC/DC/DC**
Gerilim aralığı	20,4 ila 28,8 V DC	L+ eksi 4 V Min. DC	
Nominal çıkış akımı (maks.)	300 mA (kısa devreye dayanıklı)		
Maks. dalgalılık/parazit akımları (<10 MHz)	< 1 V pik-pik	Giriş hattı gibi	
Elektriksel ayırma (CPU lojiğinden enkoder beslemesine)	Elektriksel olarak ayrılmamış		

A.5.3 Dijital girişler ve çıkışlar

Tablo A-35 Dijital girişler

Teknik veriler	**CPU 1212C AC/DC/röle, DC/DC/röle ve DC/DC/DC**
Girişlerin sayısı	8
Tip	Akım çeken/akım besleyen (IEC Tip 1, akım çeken ise)
Nominal gerilim	24 V DC'de 4 mA, nominal
İzin verilen sürekli gerilim	maks. 30 V DC
Şok gerilim	0,5 sn için 35 V DC
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
Filtre süreleri	µsn ayarları: 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0 msn ayarları: 0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0
HSC giriş adım frekansları (maks.) (Mantıksal seviye 1 = 15 ila 26 V DC)	100/80 kHz (Ea.0 ila Ea.5) 30 /20 kHz (Ea.6 ila Ea.7)
Eş zamanlı devreye alınmış girişlerin sayısı	60 °C yatayda veya 50 °C dikeyde 4 (bitişik nokta yok) 55 °C yatayda veya 45 °C dikeyde 8
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız, HSC girişleri için 50 m

Tablo A-36 Dijital çıkışlar

Teknik veriler	**CPU 1212C AC/DC/röle ve DC/DC/röle**	**CPU 1212C DC/DC/DC**
Çıkışlar	6	
Tip	Röle, mekanik	MOSFET, elektronik (akım besleyen)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	--	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	--	maks. 0,1 V DC
Akım (maks.)	2,0 A	0,5 A
Lamba yükü	30 W DC/200 W AC	5 W
AÇIK'taki direnç	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω
Çıkış başına kaçak akım	--	maks. 10 µA
Ani akım	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır	
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobinden lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	2	1

[1] Röle grupları arasındaki elektriksel ayırma, şebeke geriliminin SELV/PELV'den ayrılmasını ve 250 V AC'ye kadar farklı fazların hattan toprağa ayrılmasını sağlar.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından % 10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Teknik veriler	**CPU 1212C AC/DC/röle ve DC/DC/röle**	**CPU 1212C DC/DC/DC**
Elektriksel ayırma (gruptan gruba)	1500 V AC [1]	--
Endüktif klemens gerilimi	--	L+ eksi 48 V DC, 1 W kayıp güç
Anahtarlama gecikmesi (Aa.0 ila Aa.3)	maks. 10 msn	maks. 1,0 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 3,0 µsn
Anahtarlama gecikmesi (Aa.4 ila Aa.5)	maks. 10 msn	maks. 5 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 20 µsn
Rölenin maksimum devre frekansı	1 Hz	--
Darbe jeneratörü frekansı	Önerilmiyor[2]	100 kHz (Aa.0 ila Aa.3)[3], 2 Hz min. 20 kHz (Aa.4 ila Aa.5)[3]
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	--
Nominal yükte kontakların kullanım ömrü	100.000 anahtarlama sayısı açık/kapalı	--
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet	
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)	
Artırılmış yük için paralel çıkışlar	Hayır	
Eş zamanlı devreye alınmış çıkışların sayısı	60 °C yatayda veya 50 °C dikeyde 3 (bitişik nokta yok) 55 °C yatayda veya 45 °C dikeyde 6	
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız	

[1] Röle grupları arasındaki elektriksel ayırma, şebeke geriliminin SELV/PELV'den ayrılmasını ve 250 V AC'ye kadar farklı fazların hattan toprağa ayrılmasını sağlar.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından % 10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

A.5.4 Analog girişler

Tablo A-37 Analog girişler

Teknik veriler	Açıklama
Girişlerin sayısı	2
Tip	Gerilim (tek adımlı giriş)
Tam sapma aralığı	0 ila 10 V
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	10,001 ila 11,759 V
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	11,760 ila 11,852 V
Taşma aralığı (veri sözcüğü)	32512 ila 32767
Çözünürlük	10 Bit
Maks. dayanım gerilimi	35 V DC
Düzleştirme	Yok, zayıf, orta veya güçlü Bkz. CPU'nun analog girişleri için adım yanıtı (msn) (Sayfa 1193) tablosu
Gürültü azaltma	10, 50 veya 60 Hz
Empedans	≥100 kΩ
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % 3,0 / % 3,5'i
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

A.5.4.1 CPU'nun entegre analog girişlerinin adım yanıtı

Tablo A-38 Adım yanıtı (msn), 0 V ila 10 V %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	Bastırma frekansı (entegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Yok (1 çevrim): Ortalama alma yok	50 msn	50 msn	100 msn
Zayıf (4 çevrim): 4 tarama değeri	60 msn	70 msn	200 msn
Orta (16 çevrim): 16 tarama değeri	200 msn	240 msn	1150 msn
Güçlü (32 çevrim): 32 tarama değeri	400 msn	480 msn	2300 msn
Tarama süresi	**4,17 msn**	**5 msn**	**25 msn**

A.5.4.2 CPU'nun entegre analog portlarının tarama aralığı

Tablo A-39 CPU'nun entegre analog girişlerinin tarama aralığı

Bastırma frekansı (entegrasyon süresi seçimi)	Tarama süresi
60 Hz (16,6 msn)	4,17 msn
50 Hz (20 msn)	5 msn
10 Hz (100 msn)	25 msn

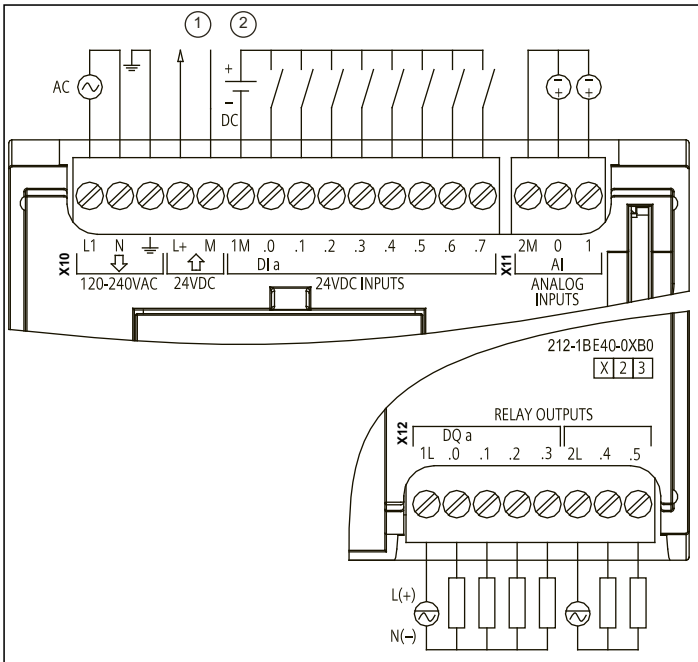
A.5.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar)

Tablo A-40 Gerilim için analog girişi gösterimi (CPU'lar)

Sistem		**Gerilim ölçüm aralığı**	
Desimal	**Onaltılı**	**0 ila 10 V**	
32767	7FFF	11,852 V	Taşma
32512	7F00		
32511	7EFF	11,759 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

A.5.5 CPU 1212C'nin devre şemaları

Tablo A-41 CPU 1212C AC/DC/röle (6ES7212-1BE40-0XB0)



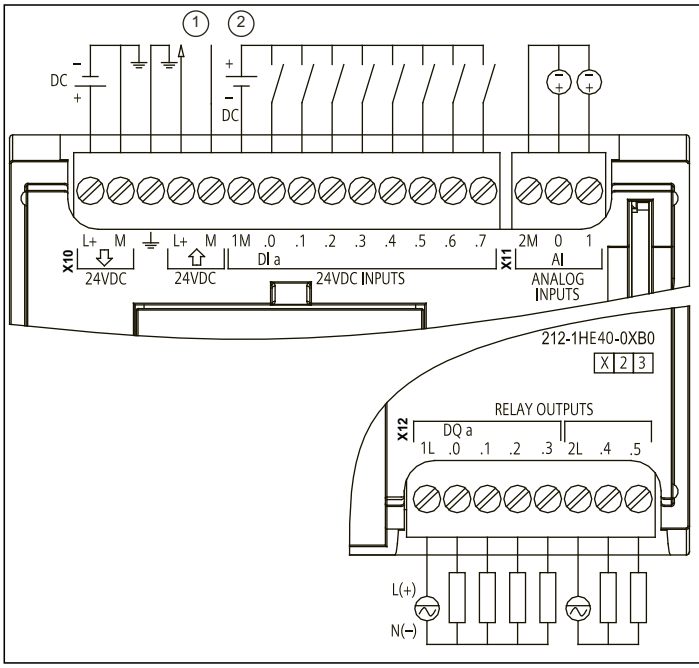
①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: İki klemensten biri, L1 veya N (L2) 240 V AC'ye kadar bir gerilim kaynağına bağlanabilir. N klemensi L2 olarak değerlendirilebilir ve topraklanmasına gerek yoktur. L1 ve N (L2) klemensleri için yanlış kutup bağlama koruması gerekli değildir.	
Not 3: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-42 CPU 1212C AC/DC/röle için bağlantı yerleşimi (6ES7212-1BE40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L1 / 120-240 V AC	2 M	1L
2	N / 120-240 V AC	AI 0	DO a.0
3	Fonksiyonel topraklama	AI 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	--	DO a.2
5	M / enkoder beslemesi 24 V DC	--	DO a.3
6	1M	--	2L
7	DI a.0	--	DO a.4
8	DI a.1	--	DO a.5
9	DI a.2	--	--
10	DI a.3	--	--

Pin	X10	X11 (altın kaplamalı)	X12
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

Tablo A-43 CPU 1212C DC/DC/röle (6ES7212-1HE40-0XB0)



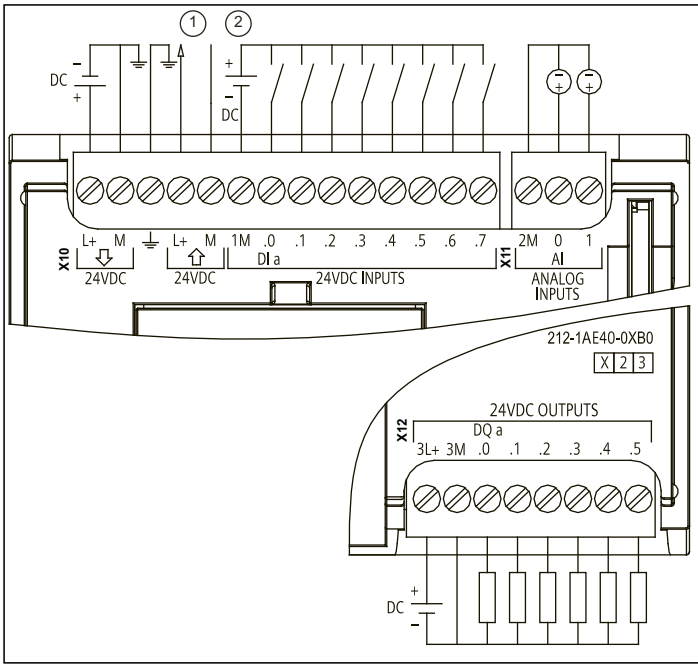
①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-44 CPU 1212C DC/DC/röle için bağlantı yerleşimi (6ES7212-1HE40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	1L
2	M / 24 V DC	AI 0	DO a.0
3	Fonksiyonel topraklama	AI 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	--	DO a.2
5	M / enkoder beslemesi 24 V DC	--	DO a.3
6	1M	--	2L
7	DI a.0	--	DO a.4
8	DI a.1	--	DO a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--

Pin	X10	X11 (altın kaplamalı)	X12
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

Tablo A-45 CPU 1212C DC/DC/DC (6ES7212-1AE40-0XB0)



①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-46 CPU 1212C DC/DC/DC için bağlantı yerleşimi (6ES7212-1AE40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	3L+
2	M / 24 V DC	AI 0	3M
3	Fonksiyonel topraklama	AI 1	DO a.0
4	L+ / enkoder beslemesi 24 V DC	--	DO a.1
5	M / enkoder beslemesi 24 V DC	--	DO a.2
6	1M	--	DO a.3
7	DI a.0	--	DO a.4
8	DI a.1	--	DO a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

NOT

Kullanılmayan analog girişlere kısa devre yapın.

A.6 CPU 1214C

A.6.1 Genel teknik veriler ve performans özellikleri

Tablo A-47 Genel

Teknik veriler	CPU 1214C AC/DC/röle	CPU 1214C DC/DC/röle	CPU 1214C DC/DC/DC
Sipariş numarası	6ES7214-1BG40-0XB0	6ES7214-1HG40-0XB0	6ES7214-1AG40-0XB0
Boyutlar (G x Y x D) (mm)	110 x 100 x 75		
Nakliye ağırlığı	475 gram	435 gram	415 gram
Performans kaybı	14 W	12 W	
Kullanılabilir elektrik akımı (SM ve CM veriyolu)	maks. 1.600 mA (5 V DC)		
Kullanılabilir elektrik akımı (24 V DC)	maks. 400 mA (enkoder beslemesi)		
Dijital giriş akım tüketimi (24 V DC)	4 mA/giriş		

Tablo A-48 CPU özellikleri

Teknik veriler		Açıklama
Kullanıcı belleği (Bkz. "Genel teknik veriler" (Sayfa 1160), "Dahili CPU belleğinde saklama").	Çalışma belleği	150 KB
	Yükleme belleği	4 MB dahili, SD kart boyutuna kadar genişletilebilir
	Kalıcı	14 kBayt
Entegre dijital G/Ç		14 giriş/10 çıkış
Entegre analog G/Ç		2 giriş
Proses görünümünün boyutu		1024 Bayt girişler (E)/1024 Bayt çıkışlar (A)
Markör (M)		8192 Bayt
Geçici (yerel) bellek		• Yol alma ve program çevrimi için 16 KB (ilgili FB'ler ve FC'ler de dahil) • Diğer her alarm öncelik seviyesi için 6 KB (FB'ler ve FC'ler dahil)
Ek sinyal modülleri		maks. 8 SM
Genişletme SB, CB, BB		maks. 1
Ek iletişim modülleri		maks. 3 CM
Hızlı sayaçlar		Herhangi bir entegre veya SB girişiyle kullanım için 6 adede kadar yapılandırılmış. Bkz. CPU 1214C için "Donanım girişinin bağlantı yerleşimi" (Sayfa 533): HSC standart adreslerinin atanması • 100/[1]80 kHz (Ea.0 ila Ea.5) • 30/[1]20 kHz (Ea.6 ila Eb.5)
Darbe çıkışları [2]		Herhangi bir entegre veya SB çıkışıyla kullanım için 4 adede kadar yapılandırılmış. • 100 kHz (Aa.0 ila Aa.3) • 20 kHz (Aa.4 ila Ab.1)
Darbe toplama için girişler		14

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Teknik veriler	Açıklama
Gecikme alarmları	1 msn çözünürlük ile 4
Uyandırma alarmları	1 msn çözünürlük ile 4
Kenar alarmları	12 yükselen ve 12 alçalan (opsiyonel sinyal kartı ile 16 ve 16)
Memory card	SIMATIC memory card (opsiyonel)
Gerçek zaman saati hassasiyeti	+/- 60 saniye/ay
Gerçek zaman saati tamponlama	Tipik 20 gün/min. 12 gün, 40 °C'de (bakım gerektirmeyen yüksek performanslı kondensatör)

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Tablo A-49 Enerji

Yordam türü		Yürütme hızı	
		Direkt adresleme (E, A ve Mi)	DB erişimi
Boole değeri		0,08 µs/yordam	
Kaydırma	Move_Bool	0,3 µs/yordam	1,17 µs/yordam
	Move_Word	0,137 µs/yordam	1,0 µs/yordam
	Move_Real	0,72 µs/yordam	1,0 µs/yordam
Gerçek sayılar aritmetiği	Add Real	1,48 µs/yordam	1,78 µs/yordam

NOT

Ölçülen süreleri etkileyen birçok değişken mevcuttur. Yukarıda belirtilen performans süreleri, bu kategorideki en hızlı yordamlar ve hatasız programlar içindir.

A.6.2 CPU 1214C tarafından desteklenen süreler, sayaçlar ve kod blokları

Tablo A-50 CPU 1214C tarafından desteklenen bloklar, süreler ve sayaçlar

Eleman		Açıklama
Bloklar	Tip	OB, FB, FC, DB
	Boyut	OB, FB, FC: 64 KB DB: çalışma belleğinin boyutuna kadar
	FB'ler, FC'ler ve DB'ler için adres aralıkları	FB ve FC: 1 ila 65535 (örn. FB 1 ila FB 65535) DB: 1 ila 59999
	Gömme derinliği	Çevrim veya yol alma OB'lerinden 16 adet Rastgele alarm olayı OB'lerinden 6 adet[1]
	Denetim	2 kod bloğunun durumu eş zamanlı olarak denetlenebilir.
OB'ler	Program çevrimi	Birden çok
	Yol alma	Birden çok

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Eleman		**Açıklama**
OB'ler	Gecikme alarmları	4 (olay başına 1)
	Uyandırma alarmları	4 (olay başına 1)
	Proses alarmları	50 (olay başına 1)
	Zaman hatası alarmları	1
	Teşhis hatası alarmları	1
	Modüllerin çıkartılması veya takılması	1
	Modül taşıyıcısında veya istasyonda hata	1
	Saat alarmı	Birden çok
	Durum	1
	Güncelleme	1
	Profil	1
	MC-Interpolator	1
	MC-Servo	1
	MC-PreServo	1
	MC-PostServo	1
Zamanlar	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, süre başına 16 Bayt
Sayaç	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, boyut, sayma türüne bağlıdır • SInt, USInt: 3 Bayt • Int, UInt: 6 Bayt • DInt, UDInt: 12 Bayt

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Tablo A-51 İletişim

Teknik veriler	Açıklama
Arayüzler	1
Tip	Ethernet
HMI cihazı	4
Programlama cihazı (PG)	1
Bağlantılar	• Açık kullanıcı iletişimi için 8 bağlantı (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV • GET/PUT verileri için 8 CPU/CPU bağlantısı (istemci veya sunucu) • GET/PUT veya açık kullanıcı iletişimine dinamik atama için 6 bağlantı • Güvenlik sertifikaları için maks. 64 bağlantı
Veri hızları	10/100 Mbit/sn
Elektriksel ayırma (lojiğe harici sinyal)	Potansiyeli ayrı dönüştürücü, 1500 V AC (tip kontrolü)[1]
Kablo türü	CAT5e blendajlı
Arayüzler	
PROFINET arayüzü sayısı	1
PROFIBUS arayüzü sayısı	0
Arayüz	
Arayüz için donanım	
Port sayısı	1
Entegre Switch	Hayır
RJ-45 (Ethernet)	Evet; X1
Protokoller	
PROFINET IO kontrolörü	Evet
PROFINET IO cihazı	Evet
SIMATIC iletişimi	Evet
Açık IE iletişimi	Evet
Web sunucusu	Evet
Medya yedekliliği	Hayır
PROFINET IO kontrolörü	
Hizmetler	
PG/OP iletişimi	Evet
S7-Routing	Evet
Adım senkronizasyonlu işletim	Hayır
Açık IE iletişimi	Evet
IRT	Hayır
MRP	Hayır
PROFIenergy	Evet. S7-1200 CPU yalnızca PROFIenergy ünitesini destekler (I-Device fonksiyonlarıyla).

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler		**Açıklama**
Öncelikli yol alma		Evet (maks. 16 PROFINET cihazı)
Bağlanabilir IO cihazlarının maksimum sayısı		16
RT için maks. bağlanabilir IO cihazların sayısı		16
Bunlardan seri olarak maks. sayı		16
Aynı anda etkinleştirilebilir/devre dışı bırakılabilir IO cihazlarının sayısı		8
Güncelleme süreleri		Güncelleme süresinin minimum değeri; PROFINET IO için ayarlanan iletişim bileşenine, IO cihazlarının sayısına ve yapılandırılmış kullanıcı verilerinin miktarına da bağlıdır.
RT ile		
Gönderim adımı 1 msn		1 msn ila 512 msn
PROFINET IO cihazı		
Hizmetler		
PG/OP iletişimi		Evet
S7-Routing		Evet
Adım senkronizasyonlu işletim		Hayır
Açık IE iletişimi		Evet
IRT, destekleniyor		Hayır
MRP, destekleniyor		Hayır
PROFIenergy		Evet
Shared Device		Evet
Shared Device'lı IO kontrolörlerinin maksimum sayısı		2
SIMATIC iletişimi		
S7 iletişimi, sunucu olarak		Evet
S7 iletişimi, istemci olarak		Evet
Görev başına maks. kullanıcı verisi		Bkz. online yardım (S7 iletişimi, kullanıcı verilerinin boyutu)
Açık IE iletişimi		
TCP/IP:		Evet
	Verilerin maks. uzunluğu	8 KB
	Port başına çoklu pasif bağlantı desteği	Evet
ISO on TCP (RFC1006):		Evet
	Verilerin maks. uzunluğu	8 KB
UDP		Evet
	Verilerin maks. uzunluğu	1472 Bayt
DHCP		Hayır

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler	Açıklama
SNMP	Evet
DCP	Evet
LLDP	Evet

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Tablo A-52 Güç kaynağı

Teknik veriler		CPU 1214C AC/DC/röle	CPU 1214C DC/DC/röle	CPU 1214C DC/DC/DC
Gerilim aralığı		85 ila 264 V AC	20,4 V DC ila 28,8 V DC	
Şebeke frekansı		47 ... 63 Hz	--	
Ani akım (maks. yük)	Yalnızca CPU	120 V AC'de 100 mA 240 V AC'de 50 mA	24 V DC'de 500 mA	
	Tüm genişletme modülleriyle birlikte CPU	120 V AC'de 300 mA 240 V AC'de 150 mA	24 V DC'de 1500 mA	
Ani akım (maks.)		264 V AC'de 20 A	28,8 V DC'de 12 A	
I^2t		0,8 A^2 sn	0,5 A^2 sn	
Elektriksel ayırma (lojiğe giriş gücü)		1500 V AC	Elektriksel olarak ayrılmamış	
Toprağa kaçak akım, AC hattından fonksiyonel topraklamaya		maks. 0,5 mA	-	
Gecikme süresi (gerilim kaybı)		120 V AC'de 20 msn 240 V AC'de 80 msn	24 V DC'de 10 msn	
Dahili sigorta, kullanıcı tarafından değiştirilemez		3 A, 250 V, yavaş		

Tablo A-53 Enkoder beslemesi

Teknik veriler	CPU 1214C AC/DC/röle	CPU 1214C DC/DC/röle	CPU 1214C DC/DC/DC
Gerilim aralığı	20,4 ila 28,8 V DC	L+ eksi 4 V DC (min.)	
Nominal çıkış akımı (maks.)	400 mA (kısa devreye dayanıklı)		
Maks. dalgalılık/parazit akımları (<10 MHz)	< 1 V pik-pik	Giriş hattı gibi	
Elektriksel ayırma (CPU lojiğinden sensör beslemesine)	Elektriksel olarak ayrılmamış		

A.6.3 Dijital girişler ve çıkışlar

Tablo A-54 Dijital girişler

Teknik veriler	**CPU 1214C AC/DC/röle**	**CPU 1214C DC/DC/röle**	**CPU 1214C DC/DC/DC**
Girişlerin sayısı	14		
Tip	Akım çeken/akım besleyen (IEC Tip 1, akım çeken ise)		
Nominal gerilim	4 mA'da 24 V DC, nominal		
İzin verilen sürekli gerilim	maks. 30 V DC		
Şok gerilim	0,5 sn için 35 V DC		
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC		
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC		
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)		
Potansiyeli ayrılmış gruplar	1		
Filtre süreleri	µsn ayarları: 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0 msn ayarları: 0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0		
HSC giriş adım frekansları (maks.) (Mantıksal seviye 1 = 15 ila 26 V DC)	100/80 kHz (Ea.0 ila Ea.5) 30/20 kHz (Ea.6 ila Eb.5)		
Eş zamanlı devreye alınmış girişlerin sayısı	• 60 °C yatayda veya 50 °C dikeyde 7 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 14		
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız, HSC girişleri için 50 m		

Tablo A-55 Dijital çıkışlar

Teknik veriler	**CPU 1214C AC/DC/röle ve DC/DC/röle**	**CPU 1214C DC/DC/DC**
Çıkışlar	10	
Tip	Röle, mekanik	MOSFET, elektronik (akım besleyen)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	--	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	--	maks. 0,1 V DC
Akım (maks.)	2,0 A	0,5 A
Lamba yükü	30 W DC/200 W AC	5 W
AÇIK'taki direnç	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω
Çıkış başına kaçak akım	--	maks. 10 µA
Ani akım	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır	
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobinden lojiğe)	707 V DC (tip kontrolü)

[1] Röle grupları arasındaki elektriksel ayırma, şebeke geriliminin SELV/PELV'den ayrılmasını ve 250 V AC'ye kadar farklı fazların hattan toprağa ayrılmasını sağlar.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanabilmek için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Teknik veriler	**CPU 1214C AC/DC/röle ve DC/DC/röle**	**CPU 1214C DC/DC/DC**
Potansiyeli ayrılmış gruplar	2	1
Elektriksel ayırma (gruptan gruba)	1500 V AC [1]	--
Endüktif klemens gerilimi	--	L+ eksi 48 V DC, 1 W kayıp güç
Anahtarlama gecikmesi (Aa.0 ila Aa.3)	maks. 10 msn	maks. 1,0 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 3,0 µsn
Anahtarlama gecikmesi (Aa.4 ila Ab.1)	maks. 10 msn	maks. 5 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 20 µsn
Rölenin maksimum devre frekansı	1 Hz	--
Darbe jeneratörü frekansı	Önerilmiyor [2]	100 kHz (Aa.0 ila Aa.3)[3], 2 Hz min. 20 kHz (Aa.4 ila Ab.1)[3]
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	--
Nominal yükte kontakların kullanım ömrü	100.000 anahtarlama sayısı açık/kapalı	--
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	
Bir dijital girişin kumandası	Evet	
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)	
Artırılmış yük için paralel çıkışlar	Hayır	
Eş zamanlı devreye alınmış çıkışların sayısı	• 60 °C yatayda veya 50 °C dikeyde 5 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 10	
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız	

[1] Röle grupları arasındaki elektriksel ayırma, şebeke geriliminin SELV/PELV'den ayrılmasını ve 250 V AC'ye kadar farklı fazların hattan toprağa ayrılmasını sağlar.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanabilmek için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

A.6.4 Analog girişler

Tablo A-56 Analog girişler

Teknik veriler	Açıklama
Girişlerin sayısı	2
Tip	Gerilim (tek adımlı giriş)
Tam sapma aralığı	0 ila 10 V
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	10,001 ila 11,759 V
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	11,760 ila 11,852 V
Taşma aralığı (veri sözcüğü)	32512 ila 32767
Çözünürlük	10 Bit
Maks. dayanım gerilimi	35 V DC
Düzleştirme	Yok, zayıf, orta veya güçlü Bkz. CPU'nun analog girişleri için adım yanıtı (msn) (Sayfa 1207) tablosu
Gürültü azaltma	10, 50 veya 60 Hz
Empedans	≥100 kΩ
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % 3,0 / % 3,5'i
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

A.6.4.1 CPU'nun entegre analog girişlerinin adım yanıtı

Tablo A-57 Adım yanıtı (msn), 0 V ila 10 V %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	Bastırma frekansı (entegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Yok (1 çevrim): Ortalama alma yok	50 msn	50 msn	100 msn
Zayıf (4 çevrim): 4 tarama değeri	60 msn	70 msn	200 msn
Orta (16 çevrim): 16 tarama değeri	200 msn	240 msn	1150 msn
Güçlü (32 çevrim): 32 tarama değeri	400 msn	480 msn	2300 msn
Tarama süresi	**4,17 msn**	**5 msn**	**25 msn**

A.6.4.2 CPU'nun entegre analog portlarının tarama aralığı

Tablo A-58 CPU'nun entegre analog girişlerinin tarama aralığı

Bastırma frekansı (entegrasyon süresi seçimi)	Tarama süresi
60 Hz (16,6 msn)	4,17 msn
50 Hz (20 msn)	5 msn
10 Hz (100 msn)	25 msn

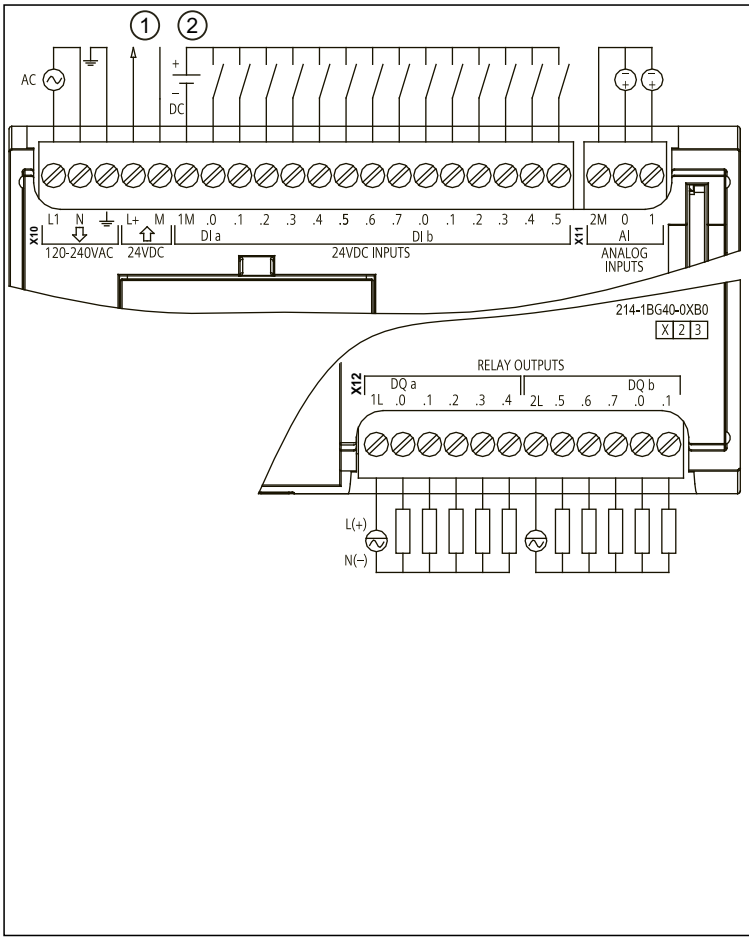
A.6.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar)

Tablo A-59 Gerilim için analog girişi gösterimi (CPU'lar)

Sistem		**Gerilim ölçüm aralığı**	
Desimal	**Onaltılı**	**0 ila 10 V**	
32767	7FFF	11,852 V	Taşma
32512	7F00		
32511	7EFF	11,759 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

A.6.5 CPU 1214C'nin devre şemaları

Tablo A-60 CPU 1214C AC/DC/röle (6ES7214-1BG40-0XB0)



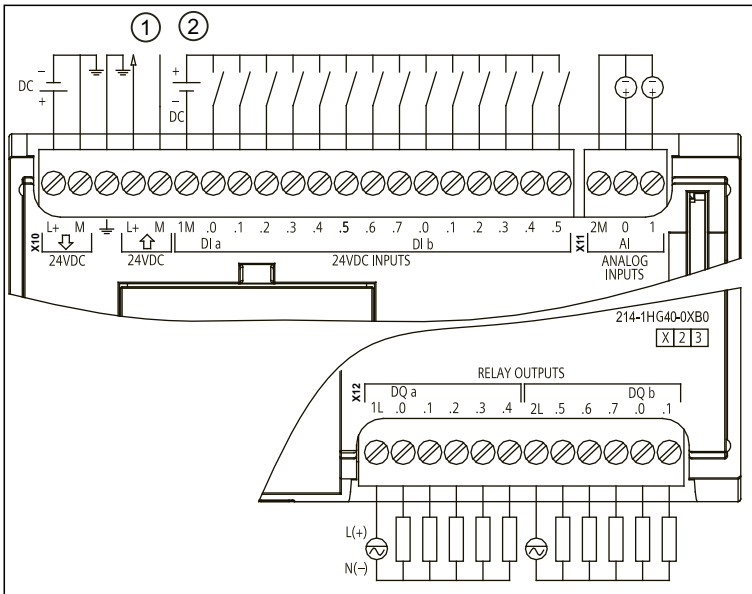
①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: İki klemensten biri, L1 veya N (L2) 240 V AC'ye kadar bir gerilim kaynağına bağlanabilir. N klemensi L2 olarak değerlendirilebilir ve topraklanmasına gerek yoktur. L1 ve N (L2) klemensleri için yanlış kutup bağlama koruması gerekli değildir.	
Not 3: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-61 CPU 1214C AC/DC/röle için bağlantı yerleşimi (6ES7214-1BG40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L1 / 120-240 V AC	2 M	1L
2	N / 120-240 V AC	AI 0	DO a.0
3	Fonksiyonel topraklama	AI 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	--	DO a.2
5	M / enkoder beslemesi 24 V DC	--	DO a.3
6	1M	--	DO a.4
7	DI a.0	--	2L
8	DI a.1	--	DO a.5
9	DI a.2	--	DO a.6
10	DI a.3	--	DO a.7

Pin	X10	X11 (altın kaplamalı)	X12
11	DI a.4	--	DO b.0
12	DI a.5	--	DO b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tablo A-62 CPU 1214C DC/DC/röle (6ES7214-1HG40-0XB0)



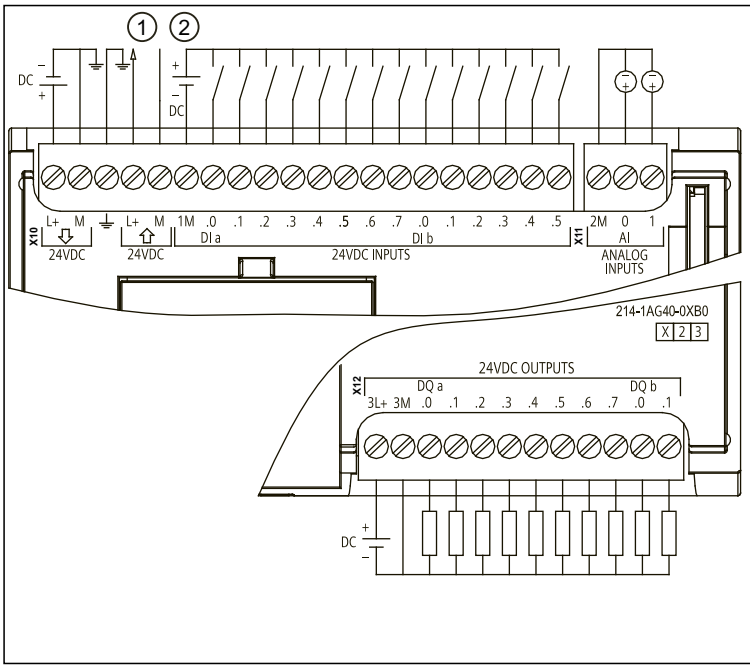
1	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
2	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-63 CPU 1214C DC/DC/röle için bağlantı yerleşimi (6ES7214-1HG40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	1L
2	M / 24 V DC	AI 0	DO a.0
3	Fonksiyonel topraklama	AI 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	--	DO a.2
5	M / enkoder beslemesi 24 V DC	--	DO a.3
6	1M	--	DO a.4
7	DI a.0	--	2L

Pin	X10	X11 (altın kaplamalı)	X12
8	DI a.1	--	DO a.5
9	DI a.2	--	DO a.6
10	DI a.3	--	DO a.7
11	DI a.4	--	DO b.0
12	DI a.5	--	DO b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tablo A-64 CPU 1214C DC/DC/DC (6ES7214-1AG40-0XB0)



①	Enkoder beslemesi 24 V DC Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-65 CPU 1214C DC/DC/DC için bağlantı yerleşimi (6ES7214-1AG40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	3L+
2	M / 24 V DC	AI 0	3M
3	Fonksiyonel topraklama	AI 1	DO a.0
4	L+ / enkoder beslemesi 24 V DC	--	DO a.1

Pin	X10	X11 (altın kaplamalı)	X12
5	M / enkoder beslemesi 24 V DC	--	DO a.2
6	1M	--	DO a.3
7	DI a.0	--	DO a.4
8	DI a.1	--	DO a.5
9	DI a.2	--	DO a.6
10	DI a.3	--	DO a.7
11	DI a.4	--	DO b.0
12	DI a.5	--	DO b.1
13	DI a.6	--	--
14	DI a.7	--	-
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

NOT

Kullanılmayan analog girişlere kısa devre yapın.

A.7 CPU 1215C

A.7.1 Genel teknik veriler ve performans özellikleri

Tablo A-66 Genel

Teknik veriler	CPU 1215C AC/DC/röle	CPU 1215C DC/DC/röle	CPU 1215C DC/DC/DC
Sipariş numarası	6ES7215-1BG40-0XB0	6ES7215-1HG40-0XB0	6ES7215-1AG40-0XB0
Boyutlar (G x Y x D) (mm)	130 x 100 x 75		
Nakliye ağırlığı	585 gram	550 gram	520 gram
Performans kaybı	14 W	12 W	
Kullanılabilir elektrik akımı (SM ve CM veriyolu)	maks. 1600 mA (5 V DC)		
Kullanılabilir elektrik akımı (24 V DC)	maks. 400 mA (enkoder beslemesi)		
Dijital giriş akım tüketimi (24 V DC)	4 mA/giriş		

Tablo A-67 CPU özellikleri

Teknik veriler		Açıklama
Kullanıcı belleği (Bkz. "Genel teknik veriler" (Sayfa 1160), "Dahili CPU belleğinde saklama").	Çalışma belleği	200 kB
	Yükleme belleği	4 MB dahili, SD kart boyutuna kadar genişletilebilir
	Kalıcı	14 kBayt
Entegre dijital G/Ç		14 giriş/10 çıkış
Entegre analog G/Ç		2 giriş/2 çıkış
Proses görünümünün boyutu		1024 Bayt girişler (E)/1024 Bayt çıkışlar (A)
Markör (M)		8192 Bayt
Geçici (yerel) bellek		• Yol alma ve program çevrimi için 16 KB (ilgili FB'ler ve FC'ler de dahil) • Diğer her alarm öncelik seviyesi için 6 KB (FB'ler ve FC'ler dahil)
Ek sinyal modülleri		maks. 8 SM
Genişletme SB, CB, BB		maks. 1
Ek iletişim modülleri		maks. 3 CM
Hızlı sayaçlar		Herhangi bir entegre veya SB girişiyle kullanım için 6 adede kadar yapılandırılmış. Bkz. CPU 1215C için "Donanım girişinin bağlantı yerleşimi" (Sayfa 533): HSC standart adreslerinin atanması • 100/[1]80 kHz (Ea.0 ila Ea.5) • 30/[1]20 kHz (Ea.6 ila Eb.5)
Darbe çıkışları [2]		Herhangi bir entegre veya SB çıkışıyla kullanım için 4 adede kadar yapılandırılmış. • 100 kHz (Aa.0 ila Aa.3) • 20 kHz (Aa.4 ila Ab.1)
Darbe toplama için girişler		14
Gecikme alarmları		1 msn çözünürlük ile 4

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Teknik veriler	Açıklama
Uyandırma alarmları	1 msn çözünürlük ile 4
Kenar alarmları	12 yükselen ve 12 alçalan (opsiyonel sinyal kartı ile 16 ve 16)
Memory card	SIMATIC memory card (opsiyonel)
Gerçek zaman saati hassasiyeti	+/- 60 saniye/ay
Gerçek zaman saati tamponlama	Tipik 20 gün/min. 12 gün, 40 °C'de (bakım gerektirmeyen yüksek performanslı kondensatör)

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanmak için dijital bir sinyal kartı (SB) takmanız gerekir.

Tablo A-68 Enerji

Yordam türü		Yürütme hızı	
		Direkt adresleme (E, A ve Mi)	DB erişimi
Boole değeri		0,08 µs/yordam	
Kaydırma	Move_Bool	0,3 µs/yordam	1,17 µs/yordam
	Move_Word	0,137 µs/yordam	1,0 µs/yordam
	Move_Real	0,72 µs/yordam	1,0 µs/yordam
Gerçek sayılar aritmetiği	Add Real	1,48 µs/yordam	1,78 µs/yordam

NOT

Ölçülen süreleri etkileyen birçok değişken mevcuttur. Yukarıda belirtilen performans süreleri, bu kategorideki en hızlı yordamlar ve hatasız programlar içindir.

A.7.2 CPU 1215C tarafından desteklenen süreler, sayaçlar ve kod blokları

Tablo A-69 CPU 1215C tarafından desteklenen bloklar, süreler ve sayaçlar

Eleman		Açıklama
Bloklar	Tip	OB, FB, FC, DB
	Boyut	OB, FB, FC: 64 kBayt DB: çalışma belleğinin boyutuna kadar
	FB'ler, FC'ler ve DB'ler için adres aralıkları	FB ve FC: 1 ila 65535 (örn. FB 1 ila FB 65535) DB: 1 ila 59999
	Gömme derinliği	Çevrim veya yol alma OB'lerinden 16 adet Rastgele alarm olayı OB'lerinden 6 adet[1]
	Denetim	2 kod bloğunun durumu eş zamanlı olarak denetlenebilir.
OB'ler	Program çevrimi	Birden çok
	Yol alma	Birden çok
	Gecikme alarmları	4 (olay başına 1)

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Eleman		Açıklama
OB'ler	Uyandırma alarmları	4 (olay başına 1)
	Proses alarmları	50 (olay başına 1)
	Zaman hatası alarmları	1
	Teşhis hatası alarmları	1
	Modüllerin çıkartılması veya takılması	1
	Modül taşıyıcısında veya istasyonda hata	1
	Saat alarmı	Birden çok
	Durum	1
	Güncelleme	1
	Profil	1
	MC-Interpolator	1
	MC-Servo	1
	MC-PreServo	1
	MC-PostServo	1
Zamanlar	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, süre başına 16 Bayt
Sayaç	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, boyut, sayma türüne bağlıdır • SInt, USInt: 3 Bayt • Int, UInt: 6 Bayt • DInt, UDInt: 12 Bayt

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Tablo A-70 İletişim

Teknik veriler	Açıklama
Arayüzler	2
Tip	Ethernet
HMI cihazı	4
Programlama cihazı (PG)	1
Bağlantılar	• Açık kullanıcı iletişimi için 8 bağlantı (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV • GET/PUT verileri için 8 CPU/CPU bağlantısı (istemci veya sunucu) • GET/PUT veya açık kullanıcı iletişimine dinamik atama için 6 bağlantı • Güvenlik sertifikaları için maks. 64 bağlantı

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler	**Açıklama**
Veri hızları	10/100 Mbit/sn
Elektriksel ayırma (lojiğe harici sinyal)	Potansiyeli ayrı dönüştürücü, 1500 V AC (tip kontrolü)[1]
Kablo türü	CAT5e blendajlı
Arayüzler	
PROFINET arayüzü sayısı	1
PROFIBUS arayüzü sayısı	0
Arayüz	
Arayüz için donanım	
Port sayısı	2
Entegre Switch	Evet
RJ-45 (Ethernet)	Evet; X1
Protokoller	
PROFINET IO kontrolörü	Evet
PROFINET IO cihazı	Evet
SIMATIC iletişimi	Evet
Açık IE iletişimi	Evet
Web sunucusu	Evet
Medya yedekliliği	Evet
PROFINET IO kontrolörü	
Hizmetler	
PG/OP iletişimi	Evet
S7-Routing	Evet
Adım senkronizasyonlu işletim	Hayır
Açık IE iletişimi	Evet
IRT	Hayır
MRP	Evet; MRP istemcisi olarak
PROFIenergy	Evet. S7-1200 CPU yalnızca PROFIenergy ünitesini destekler (I-Device fonksiyonlarıyla).
Öncelikli yol alma	Evet (maks. 16 PROFINET cihazı)
Bağlanabilir IO cihazlarının maksimum sayısı	16
RT için maks. bağlanabilir IO cihazların sayısı	16
Bunlardan seri olarak maks. sayı	16
Aynı anda etkinleştirilebilir/devre dışı bırakılabilir IO cihazlarının sayısı	8
Güncelleme süreleri	Güncelleme süresinin minimum değeri; PROFINET IO için ayarlanan iletişim bileşenine, IO cihazlarının sayısına ve yapılandırılmış kullanıcı verilerinin miktarına da bağlıdır.
RT ile	

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler		**Açıklama**
Gönderim adımı 1 msn		1 msn ila 512 msn
PROFINET IO cihazı		
Hizmetler		
PG/OP iletişimi		Evet
S7-Routing		Evet
Adım senkronizasyonlu işletim		Hayır
Açık IE iletişimi		Evet
IRT, destekleniyor		Hayır
MRP, destekleniyor		Evet
PROFIenergy		Evet
Shared Device		Evet
Shared Device'lı IO kontrolörlerinin maksimum sayısı		2
SIMATIC iletişimi		
S7 iletişimi, sunucu olarak		Evet
S7 iletişimi, istemci olarak		Evet
Görev başına maks. kullanıcı verisi		Bkz. online yardım (S7 iletişimi, kullanıcı verilerinin boyutu)
Açık IE iletişimi		
TCP/IP:		Evet
	Verilerin maks. uzunluğu	8 KB
	Port başına çoklu pasif bağlantı desteği	Evet
ISO on TCP (RFC1006):		Evet
	Verilerin maks. uzunluğu	8 KB
UDP:		Evet
	Verilerin maks. uzunluğu	1472 Bayt
DHCP		Hayır
SNMP		Evet
DCP		Evet
LLDP		Evet

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Tablo A-71 Güç kaynağı

Teknik veriler		**CPU 1215C AC/DC/röle**	**CPU 1215C DC/DC/röle**	**CPU 1215C DC/DC/DC**
Gerilim aralığı		85 ila 264 V AC	20,4 V DC ila 28,8 V DC	
Şebeke frekansı		47 ila 63 Hz	--	
Ani akım (maks. yük)	Yalnızca CPU	120 V AC'de 100 mA 240 V AC'de 50 mA	24 V DC'de 500 mA	
	Tüm genişletme modülleriyle birlikte CPU	120 V AC'de 300 mA 240 V AC'de 150 mA	24 V DC'de 1500 mA	
Ani akım (maks.)		264 V AC'de 20 A	28,8 V DC'de 12 A	
I^2t		0,8 A^2 sn	0,5 A^2 sn	
Elektriksel ayırma (lojiğe giriş gücü)		1500 V AC	Elektriksel olarak ayrılmamış	
Toprağa kaçak akım, AC hattından fonksiyonel topraklamaya		maks. 0,5 mA	-	
Gecikme süresi (gerilim kaybı)		120 V AC'de 20 msn 240 V AC'de 80 msn	24 V DC'de 10 msn	
Dahili sigorta, kullanıcı tarafından değiştirilemez		3 A, 250 V, yavaş		

Tablo A-72 Enkoder beslemesi

Teknik veriler	**CPU 1215C AC/DC/röle**	**CPU 1215C DC/DC/röle**	**CPU 1215C DC/DC/DC**
Gerilim aralığı	20,4 ila 28,8 V DC	L+ eksi 4 V DC (min.)	
Nominal çıkış akımı (maks.)	400 mA (kısa devreye dayanıklı)		
Maks. dalgalılık/parazit akımları (<10 MHz)	< 1 V pik-pik	Giriş hattı gibi	
Elektriksel ayırma (CPU lojiğinden enkoder beslemesine)	Elektriksel olarak ayrılmamış		

A.7.3 Dijital girişler ve çıkışlar

Tablo A-73 Dijital girişler

Teknik veriler	**CPU 1215C AC/DC/röle**	**CPU 1215C DC/DC/röle**	**CPU 1215C DC/DC/DC**
Girişlerin sayısı	14		
Tip	Akım çeken/akım besleyen (IEC Tip 1, akım çeken ise)		
Nominal gerilim	24 V DC'de 4 mA, nominal		
İzin verilen sürekli gerilim	maks. 30 V DC		
Şok gerilim	0,5 sn için 35 V DC		
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC		
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC		
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)		
Potansiyeli ayrılmış gruplar	1		
Filtre süreleri	us ayarları: 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0 msn ayarları: 0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0		
HSC giriş adım frekansları (maks.) (Mantıksal seviye 1 = 15 ila 26 V DC)	100/80 kHz (Ea.0 ila Ea.5) 30/20 kHz (Ea.6 ila Eb.5)		
Eş zamanlı devreye alınmış girişlerin sayısı	• 60 °C yatayda veya 50 °C dikeyde 7 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 14		
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız, HSC girişleri için 50 m		

Tablo A-74 Dijital çıkışlar

Teknik veriler	**CPU 1215C AC/DC/röle ve CPU 1215C DC/DC/röle**	**CPU 1215C DC/DC/DC**
Çıkışlar	10	
Tip	Röle, mekanik	MOSFET, elektronik (akım besleyen)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	--	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	--	maks. 0,1 V DC
Akım (maks.)	2,0 A	0,5 A
Lamba yükü	30 W DC/200 W AC	5 W
AÇIK'taki direnç	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω
Çıkış başına kaçak akım	--	maks. 10 µA
Ani akım	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır	
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobinden lojiğe)	707 V DC (tip kontrolü)

[1] Röle grupları arasındaki elektriksel ayırma, şebeke geriliminin SELV/PELV'den ayrılmasını ve 250 V AC'ye kadar farklı fazların hattan toprağa ayrılmasını sağlar.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanabilmek için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Teknik veriler	**CPU 1215C AC/DC/röle ve CPU 1215C DC/DC/röle**	**CPU 1215C DC/DC/DC**
Potansiyeli ayrılmış gruplar	2	1
Elektriksel ayırma (gruptan gruba)	1500 V AC [1]	--
Endüktif klemens gerilimi	--	L+ eksi 48 V DC, 1 W kayıp güç
Anahtarlama gecikmesi (Aa.0 ila Aa.3)	maks. 10 msn	maks. 1,0 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 3,0 µsn
Anahtarlama gecikmesi (Aa.4 ila Ab.1)	maks. 10 msn	maks. 5 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 20 µsn
Rölenin maksimum devre frekansı	1 Hz	--
Darbe jeneratörü frekansı	Önerilmiyor [2]	100 kHz (Aa.0 ila Aa.3)[3], 2 Hz min. 20 kHz (Aa.4 ila Ab.1)[3]
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	--
Nominal yükte kontakların kullanım ömrü	100.000 anahtarlama sayısı açık/kapalı	--
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	
Bir dijital girişin kumandası	Evet	
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)	
Artırılmış yük için paralel çıkışlar	Hayır	
Eş zamanlı devreye alınmış çıkışların sayısı	• 60 °C yatayda veya 50 °C dikeyde 5 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 10	
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız	

[1] Röle grupları arasındaki elektriksel ayırma, şebeke geriliminin SELV/PELV'den ayrılmasını ve 250 V AC'ye kadar farklı fazların hattan toprağa ayrılmasını sağlar.

[2] Röle çıkışlarına sahip CPU varyantlarında, darbe çıkışlarını kullanabilmek için dijital bir sinyal kartı (SB) takmanız gerekir.

[3] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

A.7.4 Analog girişler ve çıkışlar

Tablo A-75 Analog girişler

Teknik veriler	Açıklama
Girişlerin sayısı	2
Tip	Gerilim (tek adımlı giriş)
Tam sapma aralığı	0 ila 10 V
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	10,001 ila 11,759 V
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	11,760 ila 11,852 V
Taşma aralığı (veri sözcüğü)	32512 ila 32767
Çözünürlük	10 Bit
Maks. dayanım gerilimi	35 V DC
Düzleştirme	Yok, zayıf, orta veya güçlü Bkz. CPU'nun analog girişleri için adım yanıtı (msn) (Sayfa 1221) tablosu
Gürültü azaltma	10, 50 veya 60 Hz
Empedans	≥100 kΩ
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % 3,0 / % 3,5'i
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

A.7.4.1 CPU'nun entegre analog girişlerinin adım yanıtı

Tablo A-76 Adım yanıtı (msn), 0 V ila 10 V %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	Bastırma frekansı (entegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Yok (1 çevrim): Ortalama alma yok	50 msn	50 msn	100 msn
Zayıf (4 çevrim): 4 tarama değeri	60 msn	70 msn	200 msn
Orta (16 çevrim): 16 tarama değeri	200 msn	240 msn	1150 msn
Güçlü (32 çevrim): 32 tarama değeri	400 msn	480 msn	2300 msn
Tarama süresi	**4,17 msn**	**5 msn**	**25 msn**

A.7.4.2 CPU'nun entegre analog portlarının tarama aralığı

Tablo A-77 CPU'nun entegre analog girişlerinin tarama aralığı

Bastırma frekansı (entegrasyon süresi seçimi)	Tarama süresi
60 Hz (16,6 msn)	4,17 msn
50 Hz (20 msn)	5 msn
10 Hz (100 msn)	25 msn

A.7.4.3 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar)

Tablo A-78 Gerilim için analog girişi gösterimi (CPU'lar)

Sistem		**Gerilim ölçüm aralığı**	
Desimal	**Onaltılı**	**0 ila 10 V**	
32767	7FFF	11,852 V	Taşma
32512	7F00		
32511	7EFF	11,759 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

A.7.4.4 Analog çıkışların teknik verileri

Tablo A-79 Analog çıkışlar

Teknik veriler	Açıklama
Çıkışlar	2
Tip	Akım
Tam sapma aralığı	0 ... 20 mA
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	20,01 ila 23,52 mA
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	Bkz alt bilgi[1]
Veri sözcüğü taşma aralığı	32512 ila 32767
Çözünürlük	10 Bit
Çıkış empedansı	≤ 500 Ω maks.
Elektriksel ayırma (alandan lojiğe)	Yok

[1] Taşma durumunda, analog çıkışlar cihaz konfigürasyonunun özelliklerine göre davranır. "CPU STOP'a tepki" parametresi için aşağıdakilerden birini seçin: "Yedek değeri aç" veya "Son değeri tut".

Teknik veriler	Açıklama
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın %3,0 / %3,5'i
Ayarlama süresi	2 msn
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

[1] Taşma durumunda, analog çıkışlar cihaz konfigürasyonunun özelliklerine göre davranır. "CPU STOP'a tepki" parametresi için aşağıdakilerden birini seçin: "Yedek değeri aç" veya "Son değeri tut".

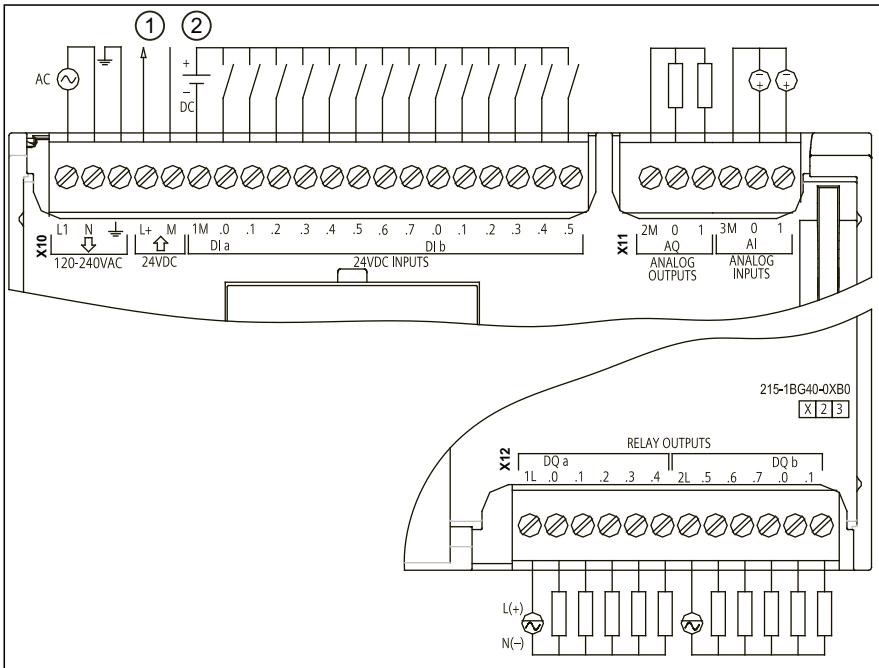
Tablo A-80 Akım için analog çıkışların gösterimi (CPU 1215C ve CPU 1217C)

Sistem		Akım çıkışı aralığı	
Desimal	**Onaltılı**	**0 mA ila 20 mA**	
32767	7FFF	Bkz. Not 1	Taşma
32512	7F00	Bkz. Not 1	
32511	7EFF	23,52 mA	Aşırı salınım aralığı
27649	6C01		
27648	6C00	20 mA	Ölçüm aralığı
20736	5100	15 mA	
34	22	0,0247 mA	
0	0	0 mA	
Negatif değerler		Negatif değerler desteklen-mez	

[1] Taşma durumunda, analog çıkışlar, ayarlanan cihaz konfigürasyonunun özelliklerine göre davranır. "CPU STOP'a tepki" parametresi için aşağıdakilerden birini seçin: "Yedek değeri aç" veya "Son değeri tut".

A.7.5 CPU 1215C'nin devre şemaları

Tablo A-81 CPU 1215C AC/DC/röle (6ES7215-1BG40-0XB0)



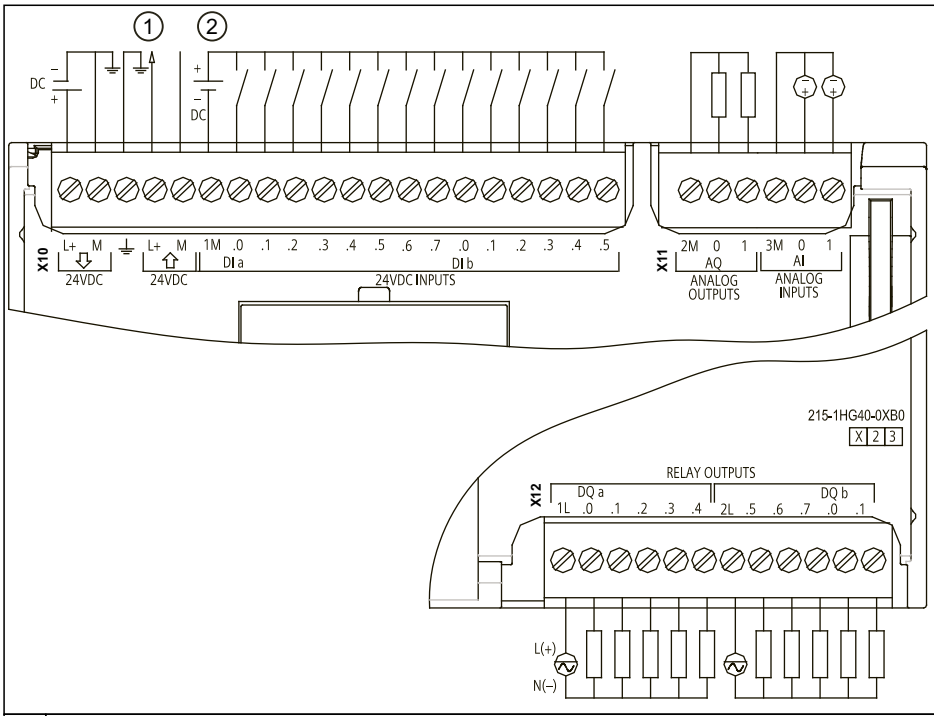
①	24-V-DC enkoder gerilimi Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: İki klemensten biri, L1 veya N (L2) 240 V AC'ye kadar bir gerilim kaynağına bağlanabilir. N klemensi L2 olarak değerlendirilebilir ve topraklanmasına gerek yoktur. L1 ve N (L2) klemensleri için yanlış kutup bağlama koruması gerekli değildir.	
Not 3: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-82 CPU 1215C AC/DC/röle için bağlantı yerleşimi (6ES7215-1BG40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L1 /120-240 V AC	2 M	1L
2	N / 120 - 240 V AC	AO 0	DO a.0
3	Fonksiyonel topraklama	AO 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	3M	DO a.2
5	M / enkoder beslemesi 24 V DC	AI 0	DO a.3
6	1M	AI 1	DO a.4
7	DI a.0	--	2L
8	DI a.1	--	DO a.5

Pin	X10	X11 (altın kaplamalı)	X12
9	DI a.2	--	DO a.6
10	DI a.3	--	DO a.7
11	DI a.4	--	DO b.0
12	DI a.5	--	DO b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tablo A-83 CPU 1215C DC/DC/röle (6ES7215-1HG40-0XB0)

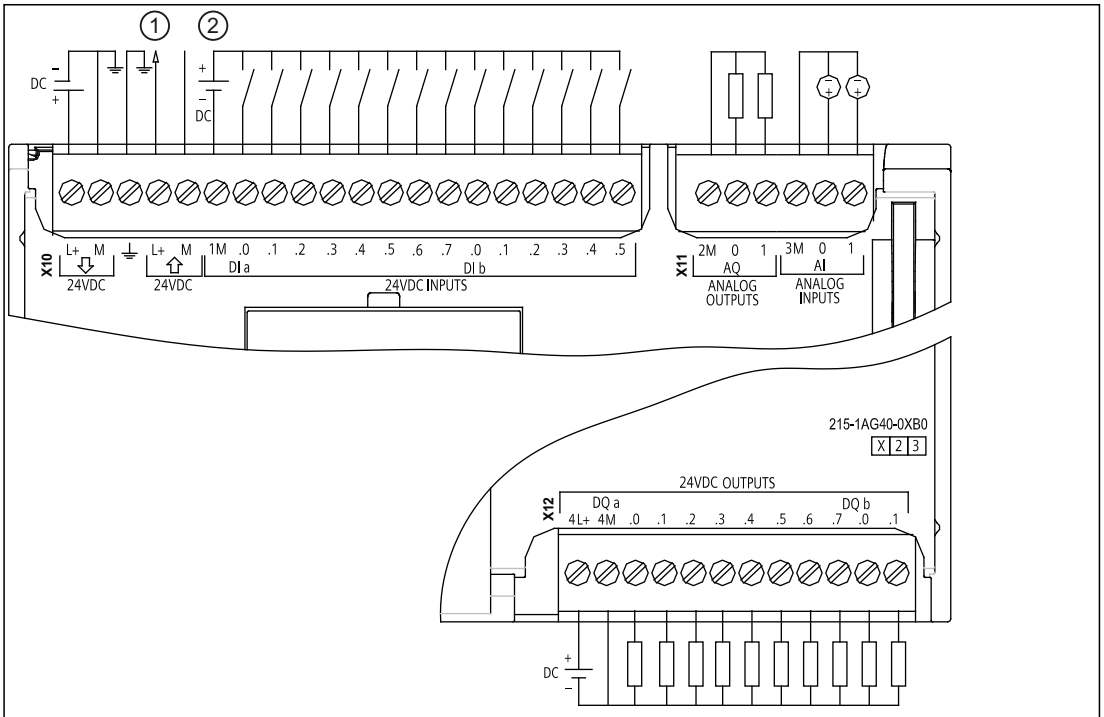


①	24-V-DC enkoder gerilimi Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-84 CPU 1215C DC/DC/röle için bağlantı yerleşimi (6ES7215-1HG40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L+ / 24 V DC	2 M	1L
2	M / 24 V DC	AO 0	DO a.0
3	Fonksiyonel topraklama	AO 1	DO a.1
4	L+ / enkoder beslemesi 24 V DC	3M	DO a.2
5	M / enkoder beslemesi 24 V DC	AI 0	DO a.3
6	1M	AI 1	DO a.4
7	DI a.0	--	2L
8	DI a.1	--	DO a.5
9	DI a.2	--	DO a.6
10	DI a.3	--	DO a.7
11	DI a.4	--	DO b.0
12	DI a.5	--	DO b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tablo A-85 CPU 1215C DC/DC/DC (6ES7215-1AG40-0XB0)



①	24-V-DC enkoder gerilimi Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
Not 1: X11 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-86 CPU 1215C DC/DC/DC için bağlantı yerleşimi (6ES7215-1AG40-0XB0)

Pin	X10	X11 (altın kaplamalı)	X12
1	L1 / 24 V DC	2 M	4L+
2	M / 24 V DC	AO 0	4M
3	Fonksiyonel topraklama	AO 1	DO a.0
4	L+ / enkoder beslemesi 24 V DC	3M	DO a.1
5	M / enkoder beslemesi 24 V DC	AI 0	DO a.2
6	1M	AI 1	DO a.3
7	DI a.0	--	DO a.4
8	DI a.1	--	DO a.5
9	DI a.2	--	DO a.6
10	DI a.3	--	DO a.7
11	DI a.4	--	DO b.0
12	DI a.5	--	DO b.1

Pin	X10	X11 (altın kaplamalı)	X12
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

NOT

Kullanılmayan analog girişlere kısa devre yapın.

A.8 CPU 1217C

A.8.1 Genel teknik veriler ve performans özellikleri

Tablo A-87 Genel

Teknik veriler	CPU 1217C DC/DC/DC
Sipariş numarası	6ES7217-1AG40-0XB0
Boyutlar (G x Y x D) (mm)	150 x 100 x 75
Nakliye ağırlığı	530 gram
Performans kaybı	12 W
Kullanılabilir elektrik akımı (SM ve CM veriyolu)	maks. 1.600 mA (5 V DC)
Kullanılabilir elektrik akımı (24 V DC)	maks. 400 mA (enkoder beslemesi)
Dijital giriş akım tüketimi (24 V DC)	4 mA/giriş

Tablo A-88 CPU özellikleri

Teknik veriler		Açıklama
Kullanıcı belleği (Bkz. "Genel teknik veriler" (Sayfa 1160)", "Dahili CPU belleğinde saklama").	Çalışma belleği	250 kB
	Yükleme belleği	4 MB, dahili, SD kart boyutuna kadar genişletilebilir
	Kalıcı	14 kBayt
Entegre dijital G/Ç		14 giriş/10 çıkış
Entegre analog G/Ç		2 giriş/2 çıkış
Proses görünümünün boyutu		1024 Bayt girişler (E)/1024 Bayt çıkışlar (A)

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

Teknik veriler	Açıklama
Markör (M)	8192 Bayt
Geçici (yerel) bellek	• Yol alma ve program çevrimi için 16 KB (ilgili FB'ler ve FC'ler de dahil) • Diğer her alarm öncelik seviyesi için 6 KB (FB'ler ve FC'ler dahil)
Ek sinyal modülleri	maks. 8 SM
Genişletme SB, CB, BB	maks. 1
Ek iletişim modülleri	maks. 3 CM
Hızlı sayaçlar	Herhangi bir entegre veya SB girişiyle kullanım için 6 adede kadar yapılandırılmış (bkz. Donanım konfigürasyonu tablosu CPU 1217C dijital giriş (DI)) (Sayfa 1233) • 1 MHz (Eb.2 ila Eb.5) • 100/[1]80 kHz (Ea.0 ila Ea.5) • 30/[1]20 kHz (Ea.6 ila Eb.1)
Darbe çıkışları	Herhangi bir entegre veya SB çıkışıyla kullanım için 4 adede kadar yapılandırılmış (bkz. Donanım konfigürasyonu tablosu CPU 1217C Dijital çıkış (DO)) (Sayfa 1233) • 1 MHz (Aa.0 ila Aa.3) • 100 kHz (Aa.4 ila Ab.1)
Darbe toplama için girişler	14
Gecikme alarmları	1 msn çözünürlük ile 4
Uyandırma alarmları	1 msn çözünürlük ile 4
Kenar alarmları	12 yükselen ve 12 alçalan (opsiyonel sinyal kartı ile 16 ve 16)
Memory card	SIMATIC memory card (opsiyonel)
Gerçek zaman saati hassasiyeti	+/- 60 saniye/ay
Gerçek zaman saati tamponlama	Tipik 20 gün/min. 12 gün, 40 °C'de (bakım gerektirmeyen yüksek performanslı kondensatör)

[1] Daha yavaş hız, HSC bir A/B sayacı olarak yapılandırılmışsa geçerlidir.

Tablo A-89 Enerji

Yordam türü		Yürütme hızı	
		Direkt adresleme (E, A ve Mi)	DB erişimi
Boole değeri		0,08 µs/yordam	
Kaydırma	Move_Bool	0,3 µs/yordam	1,17 µs/yordam
	Move_Word	0,137 µs/yordam	1,0 µs/yordam
	Move_Real	0,72 µs/yordam	1,0 µs/yordam
Gerçek sayılar aritmetiği	Add Real	1,48 µs/yordam	1,78 µs/yordam

NOT

Ölçülen süreleri etkileyen birçok değişken mevcuttur. Yukarıda belirtilen performans süreleri, bu kategorideki en hızlı yordamlar ve hatasız programlar içindir.

A.8.2 CPU 1217C tarafından desteklenen süreler, sayaçlar ve kod blokları

Tablo A-90 CPU 1217C tarafından desteklenen bloklar, süreler ve sayaçlar

Eleman		Açıklama
Bloklar	Tip	OB, FB, FC, DB
	Boyut	OB, FB, FC: 64 kBayt DB: çalışma belleğinin boyutuna kadar
	FB'ler, FC'ler ve DB'ler için adres aralıkları	FB ve FC: 1 ila 65535 (örn. FB 1 ila FB 65535) DB: 1 ila 59999
	Gömme derinliği	Çevrim veya yol alma OB'lerinden 16 adet Rastgele alarm olayı OB'lerinden 6 adet[1]
	Denetim	2 kod bloğunun durumu eş zamanlı olarak denetlenebilir.
OB'ler	Program çevrimi	Birden çok
	Yol alma	Birden çok
	Gecikme alarmları	4 (olay başına 1)
	Uyandırma alarmları	4 (olay başına 1)
	Proses alarmları	50 (olay başına 1)
	Zaman hatası alarmları	1
	Teşhis hatası alarmları	1
	Modüllerin çıkartılması veya takılması	1
	Modül taşıyıcısında veya istasyonda hata	1
	Saat alarmı	Birden çok
	Durum	1
	Güncelleme	1
	Profil	1
	MC-Interpolator	1
	MC-Servo	1
	MC-PreServo	1
	MC-PostServo	1
Zamanlar	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, süre başına 16 Bayt
Sayaç	Tip	IEC
	Adet	Sadece bellek kapasitesiyle sınırlıdır
	Kaydetme	DB'deki yapı, boyut, sayma türüne bağlıdır • SInt, USInt: 3 Bayt • Int, UInt: 6 Bayt • DInt, UDInt: 12 Bayt

[1] Güvenlik programları iki gömme seviyesi kullanır. Bununla birlikte, kullanıcı programı, güvenlik programlarında dört seviyeli bir gömme derinliğine sahiptir.

Tablo A-91 İletişim

Teknik veriler	**Açıklama**
Arayüzler	2
Tip	Ethernet
HMI cihazı	4
Programlama cihazı (PG)	1
Bağlantılar	• Açık kullanıcı iletişimi için 8 bağlantı (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRC • GET/PUT verileri için 8 CPU/CPU bağlantısı (istemci veya sunucu) • GET/PUT veya açık kullanıcı iletişimine dinamik atama için 6 bağlantı • Güvenlik sertifikaları için maks. 64 bağlantı
Veri hızları	10/100 Mbit/sn
Elektriksel ayırma (lojiğe harici sinyal)	Potansiyeli ayrı dönüştürücü, 1500 V AC (tip kontrolü)[1]
Kablo türü	CAT5e blendajlı
Arayüzler	
PROFINET arayüzü sayısı	1
PROFIBUS arayüzü sayısı	0
Arayüz	
Arayüz için donanım	
Port sayısı	2
Entegre Switch	Evet
RJ-45 (Ethernet)	Evet; X1
Protokoller	
PROFINET IO kontrolörü	Evet
PROFINET IO cihazı	Evet
SIMATIC iletişimi	Evet
Açık IE iletişimi	Evet
Web sunucusu	Evet
Medya yedekliliği	Evet
PROFINET IO kontrolörü	
Hizmetler	
PG/OP iletişimi	Evet
S7-Routing	Evet
Adım senkronizasyonlu işletim	Hayır
Açık IE iletişimi	Evet
IRT	Hayır
MRP	Evet; MRP istemcisi olarak
PROFIenergy	Evet. S7-1200 CPU yalnızca PROFIenergy ünitesini destekler (I-Device fonksiyonlarıyla).

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler		Açıklama
Öncelikli yol alma		Evet (maks. 16 PROFINET cihazı)
Bağlanabilir IO cihazlarının maksimum sayısı		16
RT için maks. bağlanabilir IO cihazların sayısı		16
Bunlardan seri olarak maks. sayı		16
Aynı anda etkinleştirilebilir/devre dışı bırakılabilir IO cihazlarının sayısı		8
Güncelleme süreleri		Güncelleme süresinin minimum değeri; PROFINET IO için ayarlanan iletişim bileşenine, IO cihazlarının sayısına ve yapılandırılmış kullanıcı verilerinin miktarına da bağlıdır.
RT ile		
Gönderim adımı 1 msn		1 msn ila 512 msn
PROFINET IO cihazı		
Hizmetler		
PG/OP iletişimi		Evet
S7-Routing		Evet
Adım senkronizasyonlu işletim		Hayır
Açık IE iletişimi		Evet
IRT, destekleniyor		Hayır
MRP, destekleniyor		Evet
PROFIenergy		Evet
Shared Device		Evet
Shared Device'lı IO kontrolörlerinin maksimum sayısı		2
SIMATIC iletişimi		
S7 iletişimi, sunucu olarak		Evet
S7 iletişimi, istemci olarak		Evet
Görev başına maks. kullanıcı verisi		Bkz. online yardım (S7 iletişimi, kullanıcı verilerinin boyutu)
Açık IE iletişimi		
TCP/IP:		Evet
	Verilerin maks. uzunluğu	8 KB
	Port başına çoklu pasif bağlantı desteği	Evet
ISO on TCP (RFC1006):		Evet
	Verilerin maks. uzunluğu	8 KB
UDP:		Evet
	Verilerin maks. uzunluğu	1472 Bayt
DHCP		Hayır

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Teknik veriler	Açıklama
SNMP	Evet
DCP	Evet
LLDP	Evet

[1] Ethernet portunun potansiyel ayrımı, kısa süreli ağ bozuklukları durumunda tehlikeli gerilimler nedeniyle ortaya çıkabilecek tehlikeyi sınırlamak için tasarlanmıştır. AC şebeke gerilimlerinde rutin potansiyel ayrımının güvenlik gerekliliklerini karşılamaz.

Tablo A-92 Güç kaynağı

Teknik veriler		CPU 1217C DC/DC/DC
Gerilim aralığı		20,4 V DC ila 28,8 V DC
Şebeke frekansı		--
Giriş akımı (maks. yük)	Yalnızca CPU	24 V DC'de 600 mA
	Tüm genişletme modülleriyle birlikte CPU	24 V DC'de 1600 mA
Ani akım (maks.)		28,8 V DC'de 12 A
$I^2 t$		0,5 A^2 sn
Elektriksel ayırma (lojiğe giriş gücü)		Elektriksel olarak ayrılmamış
Gecikme süresi (gerilim kaybından sonra)		24 V DC'de 10 msn
Dahili sigorta, kullanıcı tarafından değiştirilemez		3 A, 250 V, yavaş

Tablo A-93 Enkoder beslemesi

Teknik veriler	CPU 1217C DC/DC/DC
Gerilim aralığı	L+ eksi 4 V DC (min.)
Nominal çıkış akımı (maks.)	400 mA (kısa devreye dayanıklı)
Maks. dalgalılık/parazit akımları (<10 MHz)	Giriş hattı gibi
Elektriksel ayırma (CPU lojiğinden enkoder beslemesine)	Elektriksel olarak ayrılmamış

A.8.3 Dijital girişler ve çıkışlar

Tablo A-94 Dijital girişler

Teknik veriler	**CPU 1217C DC/DC/DC**
Girişlerin sayısı	14: Toplam: 10: Akım çeken/akım besleyen (IEC Tip 1, akım çeken ise) 4: Diferansiyel (RS422/RS485)
Tip: Akım çeken/akım besleyen (IEC Tip 1 akım çeken)	Ea.0 ila Ea.7, Eb.0 ila Eb.1
Nominal gerilim	24 V DC'de 4 mA, nominal
İzin verilen sürekli gerilim	maks. 30 V DC
Şok gerilim	0,5 sn için 35 V DC
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
Filtre süreleri	us ayarları: 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0 msn ayarları: 0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0
HSC giriş adım frekansları (maks.) (Mantıksal seviye 1 = 15 ila 26 V DC)	100/80 kHz (Ea.0 ila Ea.5) 30/20 kHz (Ea.6 ila Eb.1)
Tip: Diferansiyel giriş (RS422/RS485)	Eb.2 ila Eb.5 (0,2+ 0,2- ila 0,5+ 0,5-)
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 sn, 3 V devam eden efektif değer (RS422/RS485 karakteristik)
Entegre sonlandırma ve ön gerilim	390 Ω ila 2M Eb'-', 390 Ω ila +5 V Eb'-', (T/B gerilimsiz ise, ön gerilim KAPALI) 220 Ω, Eb'+' ila Eb'-'
Alıcı çıkış empedansı	100 Ω ön gerilim ve sonlandırma dahil
Diferansiyel alıcı Yanıt limiti/hassasiyet	min. +/- 0,2 V, 60 mV tipik histerizis (RS422/RS485 karakteristik)
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
Filtre süreleri	us ayarları: 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0 msn ayarları: 0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0
HSC giriş adım frekansları (maks.)	Tek faz sayacı: 1 MHz (Eb.2 ila Eb.5) A/B sayacı: 1 MHz (Eb.2 ila Eb.5)
Ofset Kanal-Kanal diferansiyel girişi	maks. 40 ns
Genel teknik veriler (tüm dijital girişler)	
Eş zamanlı devreye alınmış girişlerin sayısı	60 °C yatayda veya 50 °C dikeyde 5 akım çeken/akım besleyen giriş (bitişik nokta yok) ve 4 diferansiyel giriş 55 °C yatayda veya 45 °C dikeyde 14
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız HSC girişlerde 50 m blendajlı (akım çeken/akım besleyen) 50 m blendajlı, tüm diferansiyel girişler için bükülmüş

Tablo A-95 Donanım konfigürasyonu tablosu CPU 1217C dijital giriş (DI)

Giriş	**Tip ve frekans**
DIa.0	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 100 kHz
DIa.1	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 100 kHz
DIa.2	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 100 kHz
DIa.3	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 100 kHz
DIa.4	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 100 kHz
DIa.5	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 100 kHz
DIa.6	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 30 kHz
DIa.7	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 30 kHz
DIb.0	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 30 kHz
DIb.1	Tip: Giriş tip 1, 24 V, akım çeken/akım besleyen Hızlı sayacın giriş frekansı: maks. 30 kHz
DIb.2+ .2-	Tip: Diferansiyel giriş RS422/RS485 Hızlı sayacın giriş frekansı: maks. 1 MHz
DIb.3+ .3-	Tip: Diferansiyel giriş RS422/RS485 Hızlı sayacın giriş frekansı: maks. 1 MHz
DIb.4+ .4-	Tip: Diferansiyel giriş RS422/RS485 Hızlı sayacın giriş frekansı: maks. 1 MHz
DIb.5+ .5-	Tip: Diferansiyel giriş RS422/RS485 Hızlı sayacın giriş frekansı: maks. 1 MHz

Tablo A-96 Dijital çıkışlar

Teknik veriler	**CPU 1217C DC/DC/DC**
Çıkışlar	10 toplam 6: MOSFET, elektronik (akım besleyen) 4: Diferansiyel (RS422/RS485)
Tip: MOSFET, elektronik (akım besleyen çıkış)	Aa.4 - Ab.1
Gerilim aralığı	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	maks. 0,1 V DC
Akım (maks.)	0,5 A
Lamba yükü	5 W

[1] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Teknik veriler	**CPU 1217C DC/DC/DC**
AÇIK'taki direnç	maks. 0,6 Ω
Çıkış başına kaçak akım	maks. 10 µA
Ani akım	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
Endüktif klemens gerilimi	L+ eksi 48 V DC, 1 W kayıp güç
Anahtarlama gecikmesi (Aa.4 ila Ab.1)	maks. 1,0 µsn, Kapalı'dan Açık'a maks. Açık'tan Kapalı'ya 3,0 µsn
Rölenin maksimum devre frekansı	--
Darbe jeneratörü frekansı	maks. 100 kHz (Aa.4 ila Ab.1)[1], min. 2 Hz
Tip: Diferansiyel çıkış (RS422/RS485)	Aa.0 ila Aa.3 (.0+ 0- ila .3+ .3-)
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 sn, 3 V devam eden efektif değer (RS422/RS485 karakteristik)
Gönderici diferansiyel çıkış gerilimi	RL = 100 Ω'de min. 2 V, RL = 54 Ω'de min. 1,5 V (RS422/RS485 karakteristiği)
Entegre sonlandırma	100 Ω, Aa'+' ila Aa'-'
Sürücü çıkış empedansı	100 Ω, sonlandırma dahil
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
Anahtarlama gecikmesi (DAa.0 ila DAa.3)	maks. 100 nsn
Ofset Kanal-Kanal diferansiyel çıkışı	maks. 40 ns
Darbe jeneratörü frekansı	1 MHz (Aa.0 ila Aa.3), min. 2 Hz
Genel teknik veriler (tüm dijital çıkışlar)	
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet
Yedekli yük kumandası için paralel çıkışlar	Evet (yalnızca Aa.4 ila Ab.1; aynı referans iletken ile)
Artırılmış yük için paralel çıkışlar	Hayır
Eş zamanlı devreye alınmış çıkışların sayısı	3 çıkış MOSFET, elektronik (akım çeken) (bitişik nokta yok) ve 60 °C yatay veya 50 °C dikeyde 4 diferansiyel çıkış 55 °C yatayda veya 45 °C dikeyde 10
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız

[1] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Tablo A-97 Donanım konfigürasyonu tablosu CPU 1217C dijital çıkış (DO)

Çıkış	Tip ve frekans
DQa.0+ .0-	Tip: Diferansiyel çıkış RS422/RS485 Darbe jeneratörü çıkış frekansı: maks. 1 MHz, min. 2 Hz
DQa.1+ .1-	Tip: Diferansiyel çıkış RS422/RS485 Darbe jeneratörü çıkış frekansı: maks. 1 MHz, min. 2 Hz
DQa.2+ .2-	Tip: Diferansiyel çıkış RS422/RS485 Darbe jeneratörü çıkış frekansı: maks. 1 MHz, min. 2 Hz
DQa.3+ .3-	Tip: Diferansiyel çıkış RS422/RS485 Darbe jeneratörü çıkış frekansı: maks. 1 MHz, min. 2 Hz
DQa.4	Tip: 24 V akım besleyen çıkış Darbe jeneratörü çıkış frekansı: maks. 100 kHz, min. 2 Hz
DQa.5	Tip: 24 V akım besleyen çıkış Darbe jeneratörü çıkış frekansı: maks. 100 kHz, min. 2 Hz
DQa.6	Tip: 24 V akım besleyen çıkış Darbe jeneratörü çıkış frekansı: maks. 100 kHz, min. 2 Hz
DQa.7	Tip: 24 V akım besleyen çıkış Darbe jeneratörü çıkış frekansı: maks. 100 kHz, min. 2 Hz
DQb.0	Tip: 24 V akım besleyen çıkış Darbe jeneratörü çıkış frekansı: maks. 100 kHz, min. 2 Hz
DQb.1	Tip: 24 V akım besleyen çıkış Darbe jeneratörü çıkış frekansı: maks. 100 kHz, min. 2 Hz

A.8.4 Analog girişler ve çıkışlar

A.8.4.1 Analog girişlerin teknik verileri

Tablo A-98 Analog girişler

Teknik veriler	Açıklama
Girişlerin sayısı	2
Tip	Gerilim (tek adımlı giriş)
Tam sapma aralığı	0 ila 10 V
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	10,001 ila 11,759 V
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	11,760 ila 11,852 V
Taşma aralığı (veri sözcüğü)	32512 ila 32767
Çözünürlük	10 Bit
Maks. dayanım gerilimi	35 V DC
Düzleştirme	Yok, zayıf, orta veya güçlü Bkz. CPU'nun analog girişleri için adım yanıtı (msn) (Sayfa 1238) tablosu
Gürültü azaltma	10, 50 veya 60 Hz
Empedans	≥100 kΩ

Teknik veriler	Açıklama
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % 3,0 / % 3,5'i
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

A.8.4.2 CPU'nun entegre analog girişlerinin adım yanıtı

Tablo A-99 Adım yanıtı (msn), 0 V ila 10 V %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	Bastırma frekansı (entegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Yok (1 çevrim): Ortalama alma yok	50 msn	50 msn	100 msn
Zayıf (4 çevrim): 4 tarama değeri	60 msn	70 msn	200 msn
Orta (16 çevrim): 16 tarama değeri	200 msn	240 msn	1150 msn
Güçlü (32 çevrim): 32 tarama değeri	400 msn	480 msn	2300 msn
Tarama süresi	**4,17 msn**	**5 msn**	**25 msn**

A.8.4.3 CPU'nun entegre analog portlarının tarama aralığı

Tablo A-100 CPU'nun entegre analog girişlerinin tarama aralığı

Bastırma frekansı (entegrasyon süresi seçimi)	Tarama süresi
60 Hz (16,6 msn)	4,17 msn
50 Hz (20 msn)	5 msn
10 Hz (100 msn)	25 msn

A.8.4.4 Gerilim için analog girişlerin ölçüm aralıkları (CPU'lar)

Tablo A-101 Gerilim için analog girişi gösterimi (CPU'lar)

Sistem		Gerilim ölçüm aralığı	
Desimal	**Onaltılı**	**0 ila 10 V**	
32767	7FFF	11,852 V	Taşma
32512	7F00		
32511	7EFF	11,759 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

A.8.4.5 Analog çıkışların teknik verileri

Tablo A-102 Analog çıkışlar

Teknik veriler	**Açıklama**
Çıkışlar	2
Tip	Akım
Tam sapma aralığı	0 ... 20 mA
Tam sapma (veri sözcüğü)	0 ila 27648
Aşırı salınım aralığı	20,01 ila 23,52 mA
Aşırı salınım aralığı (veri sözcüğü)	27649 ila 32511
Taşma aralığı	Bkz alt bilgi[1]
Veri sözcüğü taşma aralığı	32512 ila 32767
Çözünürlük	10 Bit
Çıkış empedansı	≤ 500 Ω maks.
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın %3,0 / %3,5'i
Ayarlama süresi	2 msn
Hat uzunluğu (metre)	100 m, blendajlı, bükülmüş iletken çifti

[1] Taşma durumunda, analog çıkışlar cihaz konfigürasyonunun özelliklerine göre davranır. "CPU STOP'a tepki" parametresi için aşağıdakilerden birini seçin: "Yedek değeri aç" veya "Son değeri tut".

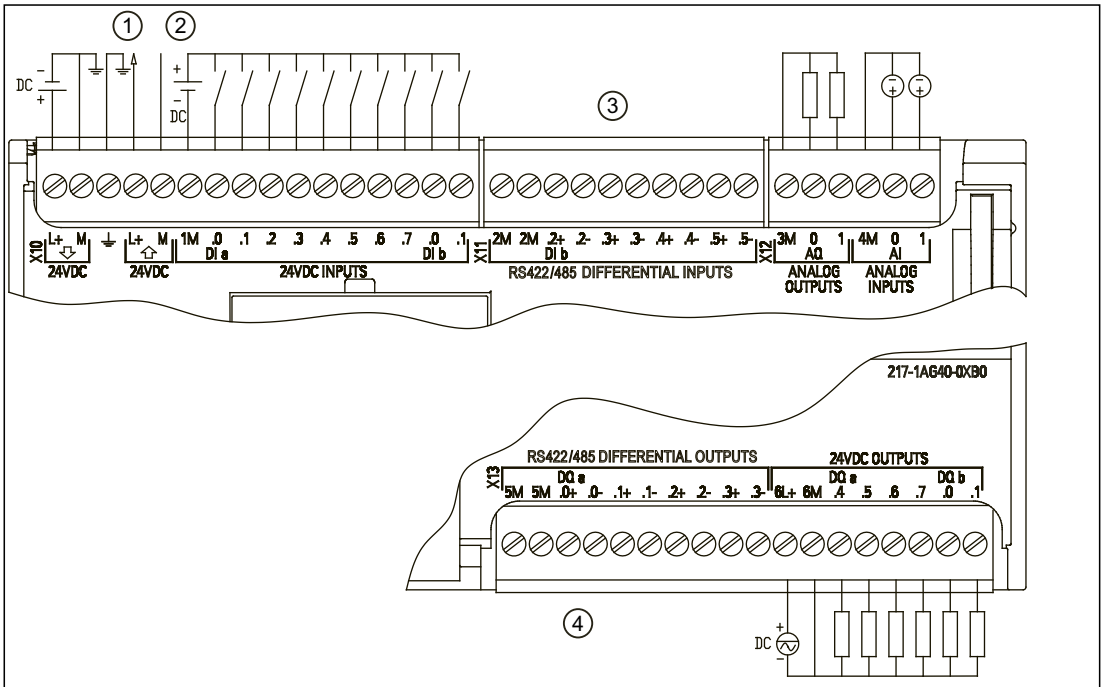
Tablo A-103 Akım için analog çıkışların gösterimi (CPU 1215C ve CPU 1217C)

Sistem		**Akım çıkışı aralığı**	
Desimal	**Onaltılı**	**0 mA ila 20 mA**	
32767	7FFF	Bkz. Not 1	Taşma
32512	7F00	Bkz. Not 1	
32511	7EFF	23,52 mA	Aşırı salınım aralığı
27649	6C01		
27648	6C00	20 mA	Ölçüm aralığı
20736	5100	15 mA	
34	22	0,0247 mA	
0	0	0 mA	
Negatif değerler		Negatif değerler desteklen-mez	

[1] Taşma durumunda, analog çıkışlar, ayarlanan cihaz konfigürasyonunun özelliklerine göre davranır. "CPU STOP'a tepki" parametresi için aşağıdakilerden birini seçin: "Yedek değeri aç" veya "Son değeri tut".

A.8.5 CPU 1217C'nin devre şemaları

Tablo A-104 CPU 1217C DC/DC/DC (6ES7217-1AG40-0XB0)



①	24-V-DC enkoder gerilimi Ek parazite dayanıklılık için, bir enkoder beslemesi kullanmasanız bile "M "yi toprağa bağlayın.
②	Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.
③	Bkz. CPU 1217C diferansiyel giriş (DI) ayrıntıları ve uygulama örneği (Sayfa 1241).
④	Bkz. CPU 1217C diferansiyel çıkış (DO) ayrıntıları ve uygulama örneği (Sayfa 1242).
Not 1: X12 soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar (Sayfa 1357) bölümünde bulabilirsiniz.	
Not 2: CPU'nun Ethernet portu hakkında daha fazla bilgiyi, Cihaz konfigürasyonu (Sayfa 128) altında bulabilirsiniz.	

Tablo A-105 CPU 1217C DC/DC/DC için bağlantı yerleşimi (6ES7217-1AG40-0XB0)

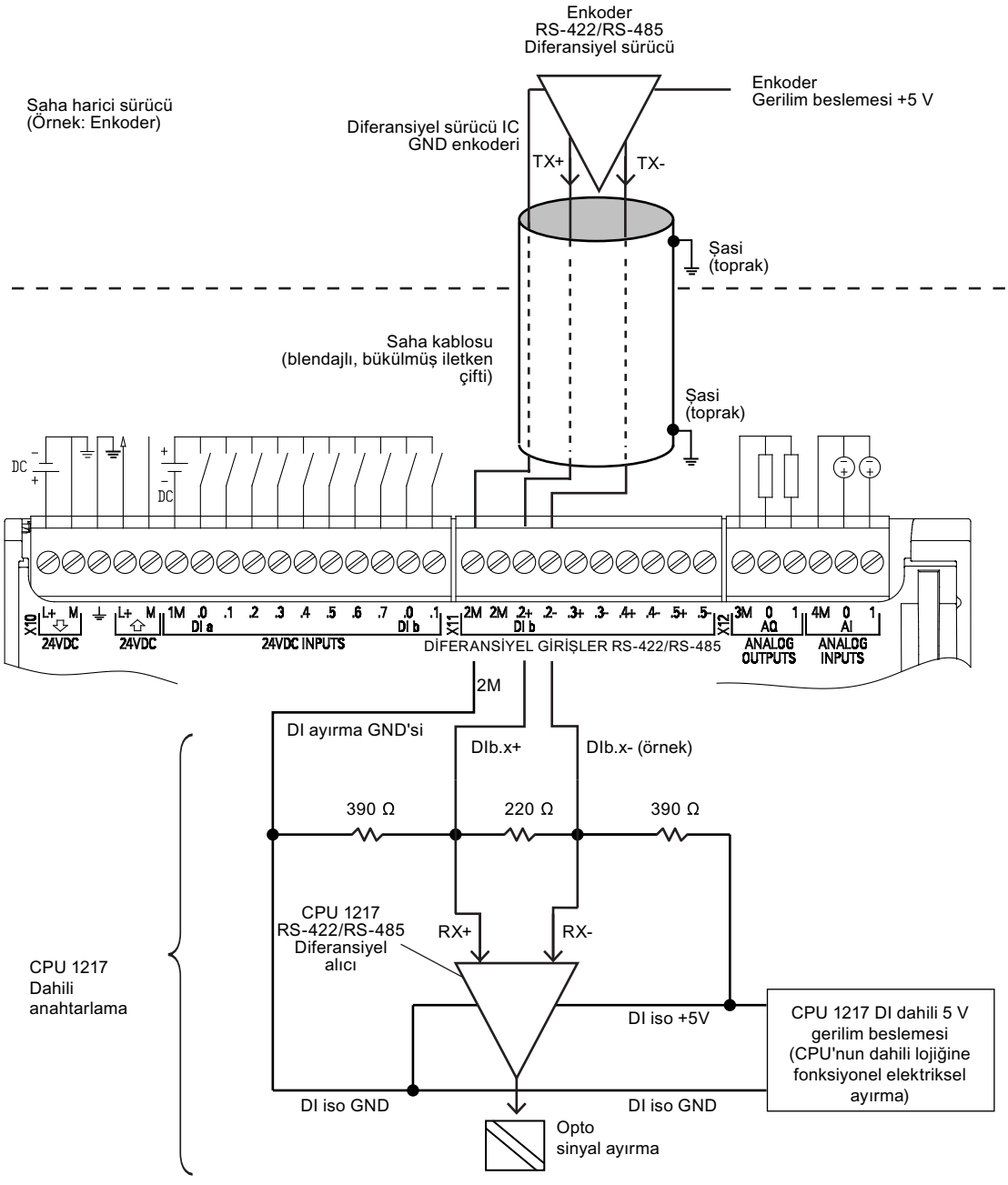
Pin	X10	X11	X12 (altın kaplamalı)	X13
1	L+ / 24 V DC	2M	3M	5M
2	M / 24 V DC	2M	AO 0	5M
3	Fonksiyonel topraklama	DI b.2+	AO 1	DO a.0+
4	L+ / enkoder beslemesi 24 V DC	DI b.2-	4M	DO a.0-
5	M / enkoder beslemesi 24 V DC	DI b.3+	AI 0	DO a.1+
6	1M	DI b.3-	AI 1	DO a.1-
7	DI a.0	DI b.4+	--	DO a.2+
8	DI a.1	DI b.4-	--	DO a.2-

Pin	X10	X11	X12 (altın kaplamalı)	X13
9	DI a.2	DI b.5+	--	DO a.3+
10	DI a.3	DI b.5-	--	DO a.3-
11	DI a.4	--	--	6L+
12	DI a.5	--	--	6M
13	DI a.6	--	--	DO a.4
14	DI a.7	--	--	DO a.5
15	DI b.0	--	--	DO a.6
16	DI b.1	--	--	DO a.7
17	--	--	--	DO b.0
18	--	--	--	DO b.1

NOT

Kullanılmayan analog girişlere kısa devre yapın.

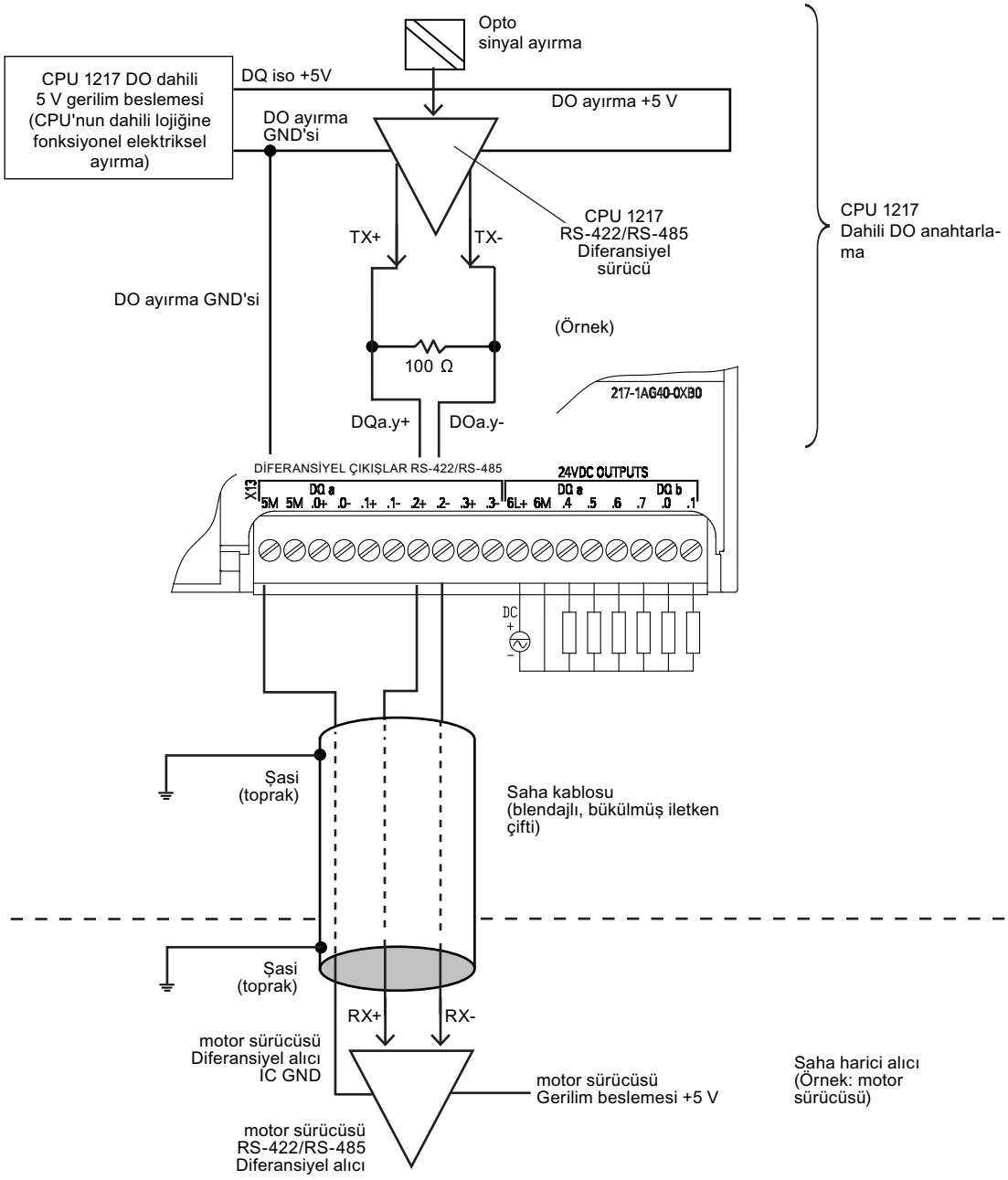
A.8.6 CPU 1217C diferansiyel giriş (DI) ayrıntıları ve uygulama örneği



Notlar

- Her bir dijital diferansiyel girişte, klemens bloğundaki vidalı klemensler açık olduğunda ön gerilim KAPALI durumdadır.
- Entegre DI sonlandırma ve ön gerilim = 100 Ω eşdeğer empedans.
- Ortak mod gerilim aralığı, entegre DI sonlandırma ve ön gerilim dirençleri ile sınırlıdır. Ayrıntılı bilgileri, teknik veriler bölümünde bulabilirsiniz.

A.8.7 CPU 1217C diferansiyel çıkış (DO) ayrıntıları ve uygulama örneği



Not

- Ortak mod gerilim aralığı, entegre DO sonlandırma direnci ile sınırlıdır. Ayrıntılı bilgileri, teknik veriler bölümünde bulabilirsiniz.

A.9 Dijital sinyal modülleri (SM'ler)

A.9.1 SM 1221 dijital giriş modülü için teknik veriler

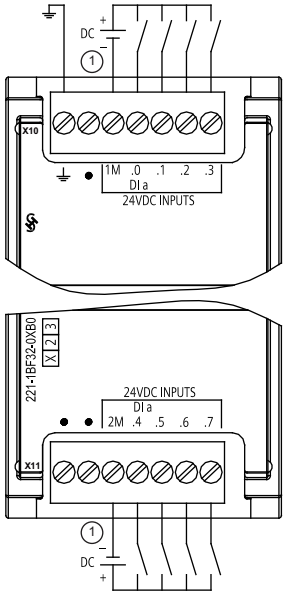
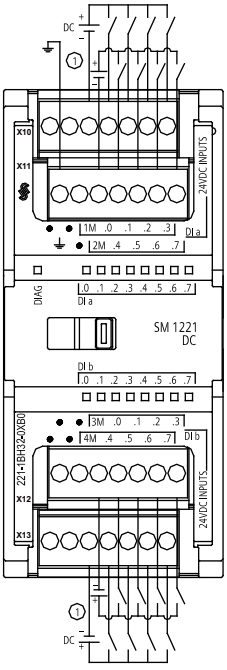
Tablo A-106 Genel teknik veriler

Model	**SM 1221 DI 8 x 24 V DC**	**SM 1221 DI 16 x 24 V DC**
Sipariş numarası	6ES7221-1BF32-0XB0	6ES7221-1BH32-0XB0
Boyutlar (G x Y x D) (mm)	45 x 100 x 75	
Ağırlık	170 gram	210 gram
Performans kaybı	1,5 W	2,5 W
Akım tüketimi (SM veriyolu)	105 mA	130 mA
Akım tüketimi (24 V DC)	4 mA / giriş	

Tablo A-107 Dijital girişler

Model	**SM 1221 DI 8 x 24 V DC**	**SM 1221 DI 16 x 24 V DC**
Girişlerin sayısı	8	16
Tip	Akım çeken/akım besleyen (IEC Tip 1, akım çeken ise)	
Nominal gerilim	4 mA'da 24 V DC, nominal	
İzin verilen sürekli gerilim	maks. 30 V DC	
Şok gerilim	0,5 sn için 35 V DC	
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC	
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC	
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)	
Potansiyeli ayrılmış gruplar	2	4
Filtre süreleri	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn (4'lü gruplar halinde seçilebilir)	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn (4'lü gruplar halinde seçilebilir)
Eş zamanlı devreye alınmış girişlerin sayısı	8	16
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız	

Tablo A-108 Dijital girişi SM'lerinin devre şemaları

SM 1221 DI 8 x 24 V DC (6ES7221-1BF32-0XB0)	SM 1221 DI 16 x 24 V DC (6ES7221-1BH32-0XB0)
	

① Akım çeken girişlerde "-"yi "M "ye bağlayın (gösterilmiştir). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.

Tablo A-109 SM 1221 DI 8 x 24 V DC için bağlantı yerleşimi (6ES7221-1BF32-0XB0)

Pin	X10	X11
1	Fonksiyonel topraklama	Bağlantı yok
2	Bağlantı yok	Bağlantı yok
3	1M	2M
4	DI a.0	DI a.4
5	DI a.1	DI a.5
6	DI a.2	DI a.6
7	DI a.3	DI a.7

Tablo A-110 SM 1221 DI 16 x 24 V DC için bağlantı yerleşimi (6ES7221-1BH32-0XB0)

Pin	X10	X11	X12	X13
1	Bağlantı yok	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	3 M	4 M
4	DI a.0	DI a.4	DI b.0	DI b.4

Pin	X10	X11	X12	X13
5	DI a.1	DI a.5	DI b.1	DI b.5
6	DI a.2	DI a.6	DI b.2	DI b.6
7	DI a.3	DI a.7	DI b.3	DI b.7

A.9.2 8 çıkışlı SM 1222 dijital çıkış modülü için teknik veriler

Tablo A-111 Genel teknik veriler

Model	SM 1222 DO 8 × röle	SM 1222 DO 8 × röle (geçiş)	SM 1222 DQ 8 × 24 V DC
Sipariş numarası	6ES7222-1HF32-0XB0	6ES7222-1XF32-0XB0	6ES7222-1BF32-0XB0
Boyutlar G × Y × D (mm)	45 × 100 × 75	70 × 100 × 75	45 × 100 × 75
Ağırlık	190 gram	310 gram	180 gram
Performans kaybı	4,5 W	5 W	1,5 W
Akım tüketimi (SM veriyolu)	120 mA	140 mA	120 mA
Akım tüketimi (24 V DC)	11 mA / röle bobini	16,7 mA / röle bobini	50 mA

Tablo A-112 Dijital çıkışlar

Model	SM 1222 DO 8 × röle	SM 1222 DO 8 × röle (geçiş)	SM 1222 DQ 8 × 24 V DC
Çıkışlar	8	8	8
Tip	Röle, mekanik	Röle CO kontak	MOSFET, elektronik (akım besleyen)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	--	--	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	--	--	maks. 0,1 V DC
Akım (maks.)	2,0 A	2,0 A	0,5 A
Lamba yükü	30 W DC/200 W AC	30 W DC/200 W AC	5 W
AÇIK'taki kontak direnci	maks. 0,2 Ω, yeni ise	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω
Çıkış başına kaçak akım	--	--	maks. 10 µA
Ani akım	Kapalı kontaklarda 7 A	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır	Hayır	
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobin lojiği)	1500 V AC (bobinden kontağa)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	2	8	1
İletken başına akım (maks.)	10 A	2 A	4 A
Endüktif klemens gerilimi	--	--	L+ eksi 48 V, 1 W kayıp güç
Anahtarlama gecikmesi	maks. 10 msn	maks. 10 msn	Kapalı'dan Açık'a maks. 50 µsn Açık'tan Kapalı'ya 200 µsn
Rölenin maksimum devre frekansı	1 Hz	1 Hz	--

Model	SM 1222 DO 8 × röle	SM 1222 DO 8 × röle (geçiş)	SM 1222 DQ 8 × 24 V DC
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	10.000.000 anahtarlama sayısı açık/kapalı	--
Nominal yükte kontakların kullanım ömrü (NO kontak)	100.000 anahtarlama sayısı açık/kapalı	100.000 anahtarlama sayısı açık/kapalı	--
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet		
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)		
Artırılmış yük için paralel çıkışlar	Hayır		
Eş zamanlı devreye alınmış çıkışların sayısı	8	• 60 °C yatayda veya 50 °C dikeyde 4 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 8	8
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız	500 m blendajlı, 150 m blendajsız	500 m blendajlı, 150 m blendajsız

A.9.3 16 çıkışlı SM 1222 dijital çıkış modülü için teknik veriler

Tablo A-113 Genel teknik veriler

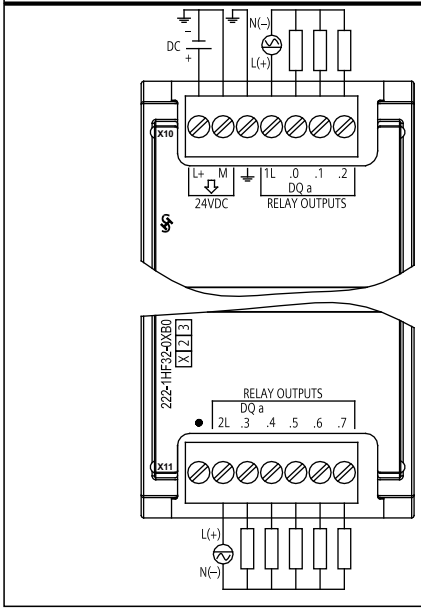
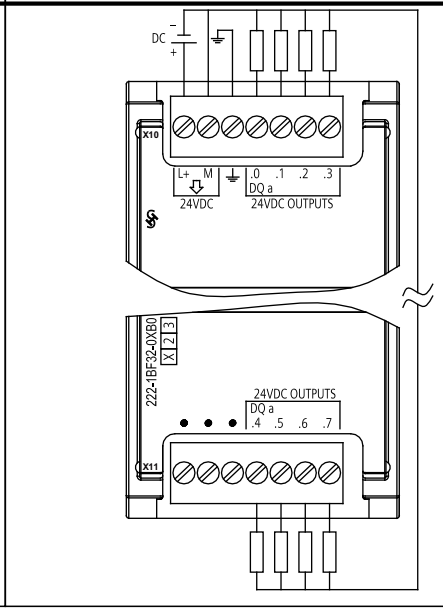
Model	SM 1222 DO 16 x röle	SM 1222 DQ 16 x 24 V DC	SM 1222 DQ 16 x 24 V DC akım çeken
Sipariş numarası	6ES7222-1HH32-0XB0	6ES7222-1BH32-0XB0	6ES7222-1BH32-1XB0
Boyutlar (G x Y x D) (mm)	45 x 100 x 75	45 x 100 x 75	45 x 100 x 75
Ağırlık	260 gram	220 gram	220 gram
Performans kaybı	8,5 W	2,5 W	2,5 W
Akım tüketimi (SM veriyolu)	135 mA	140 mA	140 mA
Akım tüketimi (24 V DC)	11 mA / röle bobini	100 mA	40 mA

Tablo A-114 Dijital çıkışlar

Model	SM 1222 DQ 16 x röle	SM 1222 DQ 16 x 24 V DC	SM 1222 DQ 16 x 24 V DC akım çeken
Çıkışlar	16	16	16
Tip	Röle, mekanik	MOSFET, elektronik (akım besleyen)	MOSFET, elektronik (akım çeken)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	-	min. 20 V DC	0,5 V DC
10 kΩ yükteki mantıksal sinyal 0	-	maks. 0,1 V DC	24 V (tipik) eksi 0,75 V DC
Akım (maks.)	2,0 A	0,5 A	0,5 A
Lamba yükü	30 W DC/200 W AC	5 W	5 W
AÇIK'taki kontak direnci	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω	maks. 0,5 Ω
Çıkış başına kaçak akım	--	maks. 10 µA	maks. 75 µA

Model	**SM 1222 DQ 16 x röle**	**SM 1222 DQ 16 x 24 V DC**	**SM 1222 DQ 16 x 24 V DC akım çeken**
Ani akım	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A	Maks. 100 msn için 8 A
Aşırı yük koruması	Hayır		Evet, akım sınırlaması ile korumalı alan 1 A ila 3,5 A
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobin lojiği)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	4	1	1
İletken başına akım (maks.)	10 A	8 A	Akım sınırlama
Endüktif klemens gerilimi	-	L+ eksi 48 V, 1 W kayıp güç	45 V
Anahtarlama gecikmesi	maks. 10 msn	Kapalı'dan Açık'a maks. 50 µsn Açık'tan Kapalı'ya 200 µsn	Kapalı'dan Açık'a maks. 20 µsn Açık'tan Kapalı'ya 350 µsn
Rölenin maksimum devre frekansı	1 Hz	-	-
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	-	-
Nominal yükte kontakların kullanım ömrü (NO kontak)	100.000 anahtarlama sayısı açık/kapalı	-	-
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet	Evet	Evet (M okuyan)
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)		
Artırılmış yük için paralel çıkışlar	Hayır		
Eş zamanlı devreye alınmış çıkışların sayısı	• 60 °C yatayda veya 50 °C dikeyde 8 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 16	16	16
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız		

Tablo A-115 8 çıkışlı dijital çıkış SM'lerinin devre şemaları

SM 1222 DQ 8 x röle (6ES7222-1HF32-0XB0)	**SM 1222 DQ 8 x 24 V DC (6ES7222-1BF32-0XB0)**
	

Tablo A-116 SM 1222 DQ 8 x röle için bağlantı yerleşimi (6ES7222-1HF32-0XB0)

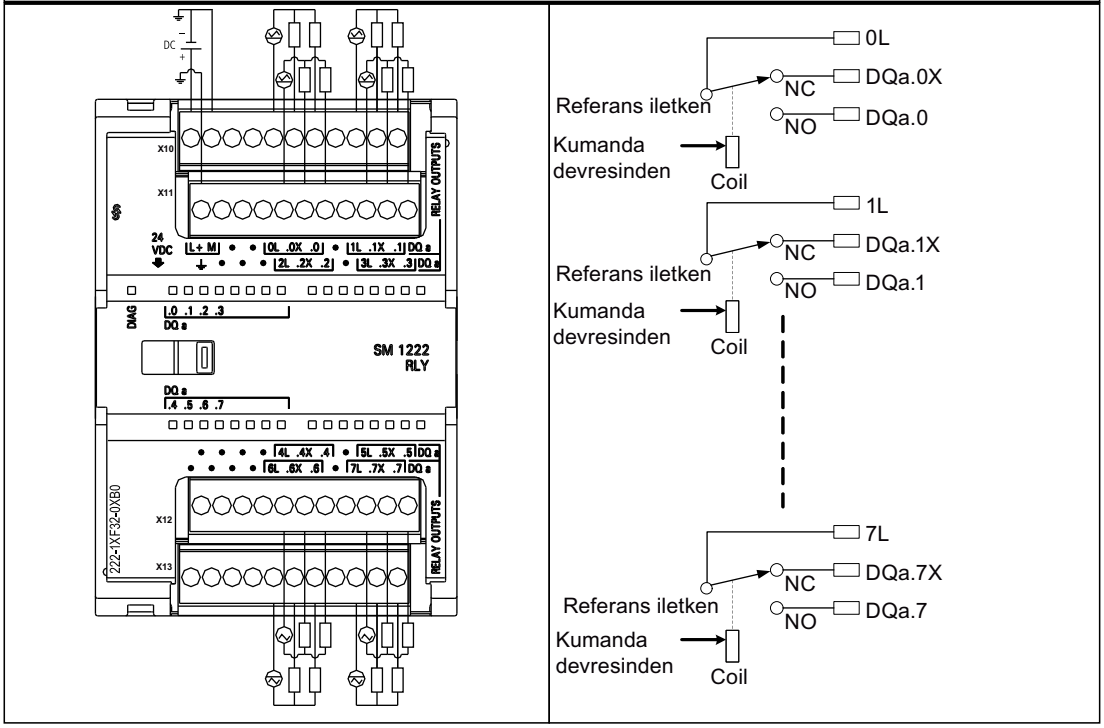
Pin	X10	X11
1	L+ / 24 V DC	Bağlantı yok
2	M / 24 V DC	2L
3	Fonksiyonel topraklama	DO a.3
4	1L	DO a.4
5	DO a.0	DO a.5
6	DO a.1	DO a.6
7	DO a.2	DO a.7

Tablo A-117 SM 1222 DQ 8 x 24 V DC için bağlantı yerleşimi (6ES7222-1BF32-0XB0)

Pin	X10	X11
1	L+ / 24 V DC	Bağlantı yok
2	M / 24 V DC	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok
4	DO a.0	DO a.4
5	DO a.1	DO a.5
6	DO a.2	DO a.6
7	DO a.2	DO a.7

Tablo A-118 Anahtarlamalı dijital çıkış SM için 8 röle çıkışlı devre şeması

SM 1222 DQ 8 x röle, geçiş (6ES7222-1XF32-0XB0)

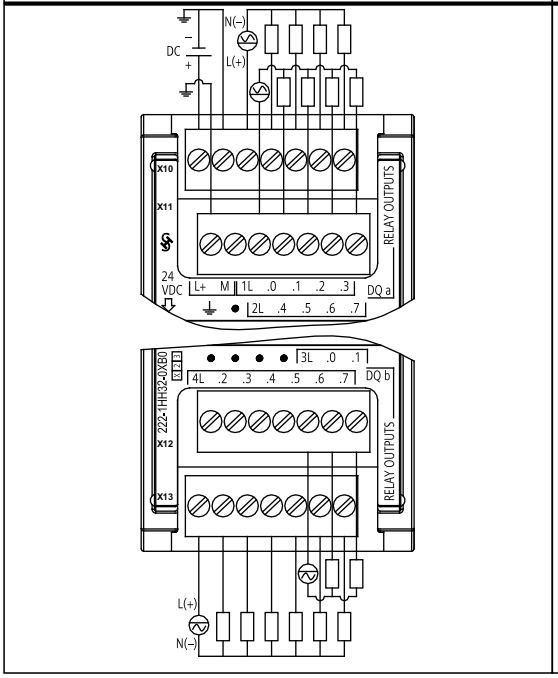
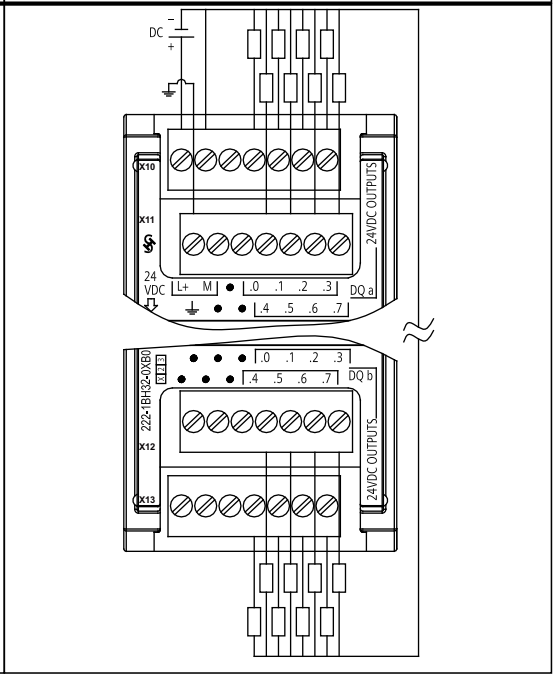


Bir geçiş rölesi çıkışı, ortak bir klemens üzerinden iki elektrik devresine kumanda eder: Bir NC kontak ve bir NO kontak. Çıkış "0" örneğinde gösterildiği gibi devre dışı bırakılmış çıkışta, ortak gerilim (0L) NC kontağa (.0X) bağlanır ve NO kontaktan (.0) ayrılır. Devreye alınmış bir çıkış, gerilim (0L) NC kontağından (.0X) ayrılır ve NO kontağına (.0) bağlanır.

Tablo A-119 SM 1222 DQ 8 x röle, geçiş için bağlantı yerleşimi (6ES7222-1XF32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel toprakla-ma	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
5	0L	2L	4L	6L
6	DO a.0X	DO a.2X	DO a.4X	DO a.6X
7	DO a.0	DO a.2	DO a.4	DO a.6
8	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
9	1L	3L	5L	7L
10	DO a.1X	DO a.3X	DO a.5X	DO a.7X
11	DO a.1	DO a.3	DO a.5	DO a.7

Tablo A-120 16 çıkışlı dijital çıkış SM'lerinin devre şemaları

SM 1222 DQ 16 x röle (6ES7222-1HH32-0XB0)	**SM 1222 DQ 16 x 24 V DC (6ES7222-1BH32-0XB0)**
	

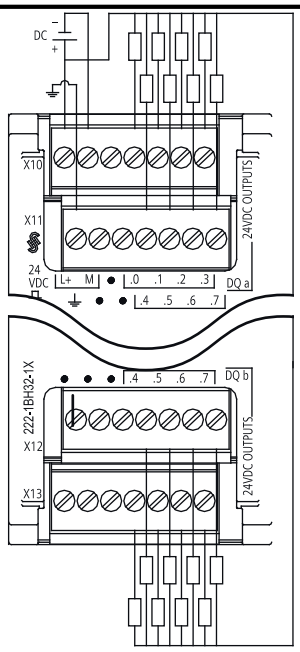
Tablo A-121 SM 1222 DQ 16 x röle için bağlantı yerleşimi (6ES7222-1HH32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	4L
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	DO b.2
3	1L	2L	Bağlantı yok	DO b.3
4	DO a.0	DO a.4	Bağlantı yok	DO b.4
5	DO a.1	DO a.5	3L	DO b.5
6	DO a.2	DO a.6	DO b.0	DO b.6
7	DO a.3	DO a.7	DO b.1	DO b.7

Tablo A-122 SM 1222 DQ 16 x 24 V DC için bağlantı yerleşimi (6ES7222-1BH32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	DO a.0	DO a.4	DO b.0	DO b.4
5	DO a.1	DO a.5	DO b.1	DO b.5
6	DO a.2	DO a.6	DO b.2	DO b.6
7	DO a.3	DO a.7	DO b.3	DO b.7

Tablo A-123 SM dijital çıkışının devre şeması, 16 çıkış, 24 V DC akım çeken

SM 1222 DQ 16 x 24 V DC akım çeken (6ES7222-1BH32-1XB0)	
	

Tablo A-124 SM 1222 DQ 16 x 24 V DC akım çeken için bağlantı yerleşimi (6ES7222-1BH32-1XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	DO a.0	DO a.4	DO b.0	DO b.4
5	DO a.1	DO a.5	DO b.1	DO b.5
6	DO a.2	DO a.6	DO b.2	DO b.6
7	DO a.3	DO a.7	DO b.3	DQ b.7

A.9.4 SM 1223 (V DC) dijital giriş/çıkış modülünün teknik verileri

Tablo A-125 Genel teknik veriler

Model	**SM 1223 DI 8 × 24 V DC, DQ 8 × röle**	**SM 1223 DI 16 × 24 V DC, DQ 16 × röle**	**SM 1223 DI 8 × 24 V DC, DQ 8 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC akım çeken**
Sipariş numarası	6ES7223-1PH32-0-XB0	6ES7223-1PL32-0-XB0	6ES7223-1BH32-0-XB0	6ES7223-1BL32-0-XB0	6ES7223-1BL32-1-XB0
Boyutlar G × Y × D (mm)	45 × 100 × 75	70 × 100 × 75	45 × 100 × 75	70 × 100 × 75	70 × 100 × 75
Ağırlık	230 gram	350 gram	210 gram	310 gram	310 gram
Performans kaybı	5,5 W	10 W	2,5 W	4,5 W	4,5 W
Akım tüketimi (SM veri-yolu)	145 mA	180 mA	145 mA	185 mA	185 mA
Akım tüketimi (24 V DC)	4 mA / giriş 11 mA / röle bobini		150 mA	200 mA	40 mA

Tablo A-126 Dijital girişler

Model	**SM 1223 DI 8 × 24 V DC, DQ 8 × röle**	**SM 1223 DI 16 × 24 V DC, DQ 16 × röle**	**SM 1223 DI 8 × 24 V DC, DQ 8 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC akım çeken**
Girişlerin sayısı	8	16	8	16	16
Tip	Akım çeken/akım besleyen (IEC Tip 1 akım çeken)	Akım çeken/akım besleyen (IEC Tip 1 akım çeken)	Akım çeken/akım besleyen (IEC Tip 1 akım çeken)	Akım çeken/akım besleyen (IEC Tip 1 akım çeken)	Akım çeken/akım besleyen (IEC Tip 1 akım çeken)
Nominal gerilim	4 mA'da 24 V DC, nominal	4 mA'da 24 V DC, nominal	4 mA'da 24 V DC, nominal	4 mA'da 24 V DC, nominal	4 mA'da 24 V DC, nominal
İzin verilen sürekli gerilim	maks. 30 V DC	maks. 30 V DC	maks. 30 V DC	maks. 30 V DC	maks. 30 V DC
Şok gerilim	0,5 sn için 35 V DC	0,5 sn için 35 V DC	0,5 sn için 35 V DC	0,5 sn için 35 V DC	0,5 sn için 35 V DC
Mantıksal sinyal 1 (min.)	2,5 mA'da 15 V DC	2,5 mA'da 15 V DC	2,5 mA'da 15 V DC	2,5 mA'da 15 V DC	2,5 mA'da 15 V DC
Mantıksal sinyal 0 (maks.)	1 mA'da 5 V DC	1 mA'da 5 V DC	1 mA'da 5 V DC	1 mA'da 5 V DC	1 mA'da 5 V DC
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	2	2	2	2	2

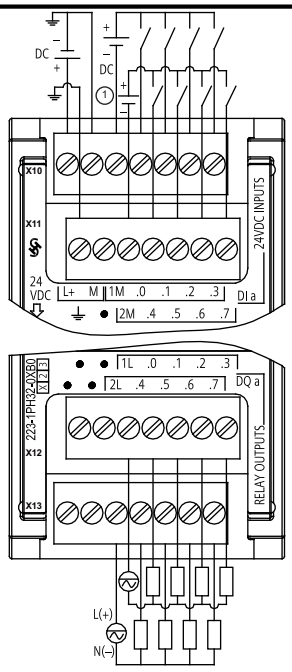
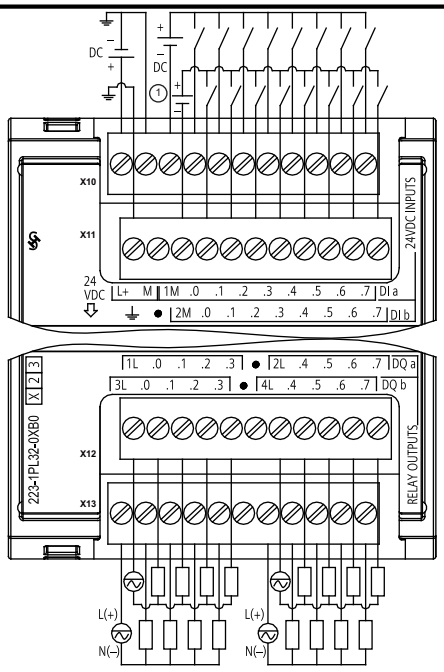
Model	**SM 1223 DI 8 × 24 V DC, DQ 8 × röle**	**SM 1223 DI 16 × 24 V DC, DQ 16 × röle**	**SM 1223 DI 8 × 24 V DC, DQ 8 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC akım çeken**
Filtre süreleri	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn 4'lü gruplar halinde seçilebilir	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn 4'lü gruplar halinde seçilebilir	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn 4'lü gruplar halinde seçilebilir	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn 4'lü gruplar halinde seçilebilir	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 ve 12,8 msn 4'lü gruplar halinde seçilebilir
Eş zamanlı devreye alınmış girişlerin sayısı	8	• 60 °C yatayda veya 50 °C dikeyde 8 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 16	8	16	16
Hat uzunluğu (metre)	500 m blendajlı, 300 m blendajsız	500 m blendajlı, 300 m blendajsız	500 m blendajlı, 300 m blendajsız	500 m blendajlı, 300 m blendajsız	500 m blendajlı, 300 m blendajsız

Tablo A-127 Dijital çıkışlar

Model	**SM 1223 DI 8 × 24 V DC, DQ 8 × röle**	**SM 1223 DI 16 × 24 V DC, DQ 16 × röle**	**SM 1223 DI 8 × 24 V DC, DQ 8 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC akım çeken**
Çıkışlar	8	16	8	16	16
Tip	Röle, mekanik	Röle, mekanik	MOSFET, elektronik (akım besleyen)	MOSFET, elektronik (akım besleyen)	MOSFET, elektronik (akım çeken)
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC	5 ila 30 V DC veya 5 ila 250 V AC	20,4 ila 28,8 V DC	20,4 ila 28,8 V DC	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	--	--	min. 20 V DC	min. 20 V DC	0,5 V DC
10 kΩ yükteki mantıksal sinyal 0	--	--	maks. 0,1 V DC	maks. 0,1 V DC	24 V (tipik) eksi 0,75 V DC
Akım (maks.)	2,0 A	2,0 A	0,5 A	0,5 A	0,5 A
Lamba yükü	30 W DC/200 W AC	30 W DC/200 W AC	5 W	5 W	5 W
AÇIK'taki kontak direnci	maks. 0,2 Ω, yeni ise	maks. 0,2 Ω, yeni ise	maks. 0,6 Ω	maks. 0,6 Ω	maks. 0,5 Ω
Çıkış başına kaçak akım	--	--	maks. 10 µA	maks. 10 µA	maks. 75 µA
Ani akım	Kapalı kontaklarda 7 A	Kapalı kontaklarda 7 A	Maks. 100 msn için 8 A	Maks. 100 msn için 8 A	Akım sınırlama
Aşırı yük koruması	Hayır	Hayır	Hayır	Hayır	Evet, akım sınırlaması ile korumalı alan 1 A ila 3,5 A
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobin lojiği)	1500 V AC (bobinden kontağa) Yok (bobin lojiği)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	2	4	1	1	1

Model	**SM 1223 DI 8 × 24 V DC, DQ 8 × röle**	**SM 1223 DI 16 × 24 V DC, DQ 16 × röle**	**SM 1223 DI 8 × 24 V DC, DQ 8 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC**	**SM 1223 DI 16 × 24 V DC, DQ 16 × 24 V DC akım çeken**
İletken başına akım	10 A	8 A	4 A	8 A	8 A
Endüktif klemens gerilimi	--	--	L+ eksi 48 V, 1 W kayıp güç	L+ eksi 48 V, 1 W kayıp güç	45 V
Anahtarlama gecikmesi	maks. 10 msn	maks. 10 msn	Kapalı'dan Açık'a maks. 50 µsn maks. Açık'tan Kapalı'ya 200 µsn	Kapalı'dan Açık'a maks. 50 µsn maks. Açık'tan Kapalı'ya 200 µsn	Kapalı'dan Açık'a maks. 20 µsn maks. Açık'tan Kapalı'ya 350 µsn
Rölenin maksimum devre frekansı	1 Hz	1 Hz	--	--	--
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı	10.000.000 anahtarlama sayısı açık/kapalı	--	--	--
Nominal yükte kontakların kullanım ömrü (NO kontak)	100.000 anahtarlama sayısı açık/kapalı	100.000 anahtarlama sayısı açık/kapalı	--	--	--
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet	Evet	Evet	Evet	Evet (M okuyan)
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)	Evet (aynı referans iletken ile)	Evet (aynı referans iletken ile)	Evet (aynı referans iletken ile)	Evet (aynı referans iletken ile)
Artırılmış yük için paralel çıkışlar	Hayır	Hayır	Hayır	Hayır	Hayır
Eş zamanlı devreye alınmış çıkışların sayısı	8	• 60 °C yatayda veya 50 °C dikeyde 8 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 16	8	16	16
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız	500 m blendajlı, 150 m blendajsız	500 m blendajlı, 150 m blendajsız	500 m blendajlı, 150 m blendajsız	500 m blendajlı, 150 m blendajsız

Tablo A-128 Dijital giriş/çıkış modüllerinin devre şemaları (V DC/röle)

SM 1223 DI 8 x 24 V DC, DQ 8 x röle (6ES7223-1PH32-0XB0)	SM 1223 DI 16 x 24 V DC, DQ 16 x röle (6ES7223-1PL32-0XB0)	Notlar
		① Akım çeken girişlerde "-"yi "M"ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.

Tablo A-129 SM 1223 DI 8 x 24 V DC, DQ 8 x röle için bağlantı yerleşimi (6ES7223-1PH32-0XB0)

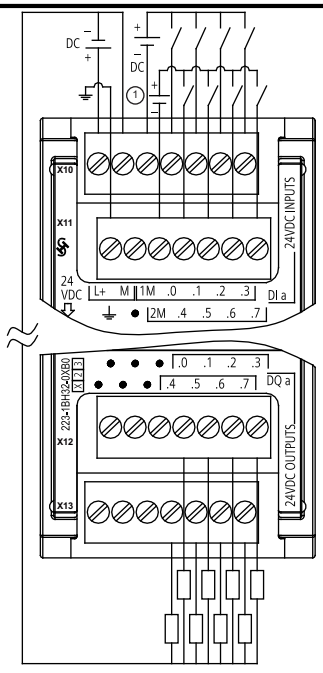
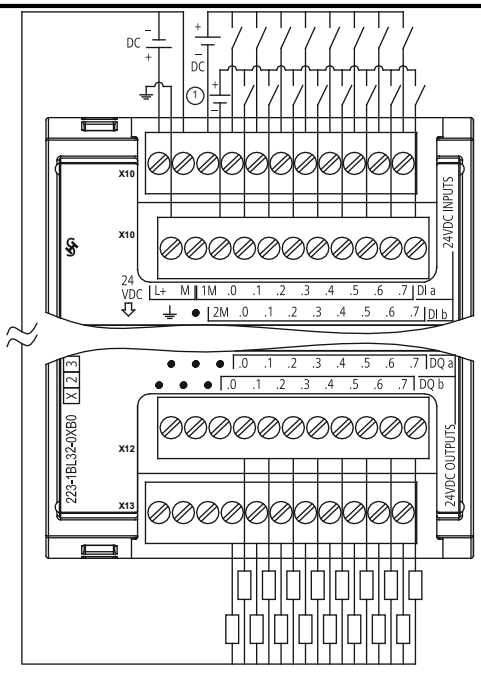
Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	1L	2L
4	DI a.0	DI a.4	DO a.0	DO a.4
5	DI a.1	DI a.5	DO a.1	DO a.5
6	DI a.2	DI a.6	DO a.2	DO a.6
7	DI a.3	DI a.7	DO a.3	DO a.7

Tablo A-130 SM 1223 DI 16 x 24 V DC, DQ 16 x röle için bağlantı yerleşimi (6ES7223-1PL32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	1L	3L
2	M / 24 V DC	Bağlantı yok	DO a.0	DO b.0
3	1M	2M	DO a.1	DO b.1
4	DI a.0	DI b.0	DO a.2	DO b.2
5	DI a.1	DI b.1	DO a.3	DO b.3
6	DI a.2	DI b.2	Bağlantı yok	Bağlantı yok
7	DI a.3	DI b.3	2L	4L

Pin	X10	X11	X12	X13
8	DI a.4	DI b.4	DO a.4	DO b.4
9	DI a.5	DI b.5	DO a.5	DO b.5
10	DI a.6	DI b.6	DO a.6	DO b.6
11	DI a.7	DI b.7	DO a.7	DO b.7

Tablo A-131 Dijital giriş/çıkış modüllerinin devre şemaları (V DC)

SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC (6ES7223-1BH32-0XB0)	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC (6ES7223-1BL32-0XB0)	Notlar
		① Akım çeken girişlerde "-"yi "M"ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.

Tablo A-132 SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC için bağlantı yerleşimi (6ES7223-1BH32-0XB0)

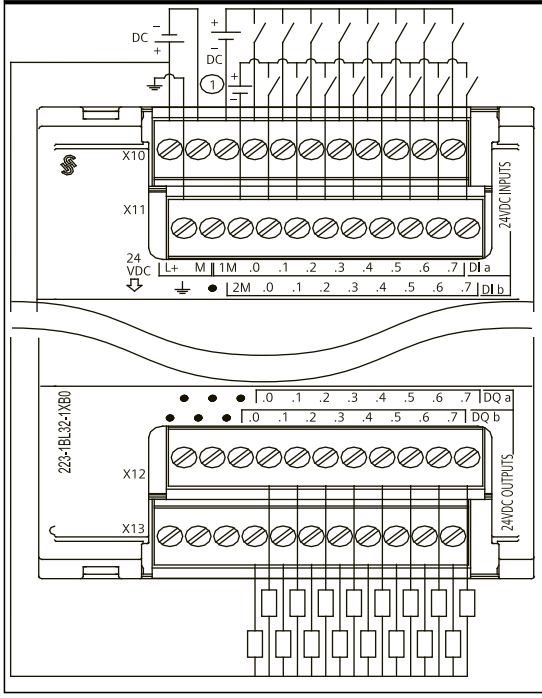
Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	Bağlantı yok	Bağlantı yok
4	DI a.0	DI a.4	DO a.0	DO a.4
5	DI a.1	DI a.5	DO a.1	DO a.5
6	DI a.2	DI a.6	DO a.2	DO a.6
7	DI a.3	DI a.7	DO a.3	DO a.7

Tablo A-133 SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC için bağlantı yerleşimi (6ES7223-1BL32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok

Pin	X10	X11	X12	X13
3	1M	2M	Bağlantı yok	Bağlantı yok
4	DI a.0	DI b.0	DO a.0	DO b.0
5	DI a.1	DI b.1	DO a.1	DO b.1
6	DI a.2	DI b.2	DO a.2	DO b.2
7	DI a.3	DI b.3	DO a.3	DO b.3
8	DI a.4	DI b.4	DO a.4	DO b.4
9	DI a.5	DI b.5	DO a.5	DO b.5
10	DI a.6	DI b.6	DO a.6	DO b.6
11	DI a.7	DI b.7	DO a.7	DO b.7

Tablo A-134 Dijital giriş/çıkış sinyal modüllerinin devre şeması (V DC)

SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC akım çeken (6ES7223-1BL32-1XB0)	Notlar
	① Akım çeken girişlerde "-"yi "M"ye bağlayın (gösterilen). Akım besleyen girişlerde "+" ile "M" arasında bağlantı yapın.

Tablo A-135 SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC akım çeken için bağlantı yerleşimi (6ES7223-1BL32-1XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	Bağlantı yok	Bağlantı yok
4	DI a.0	DI b.0	DO a.0	DO b.0
5	DI a.1	DI b.1	DO a.1	DO b.1
6	DI a.2	DI b.2	DO a.2	DO b.2

Pin	X10	X11	X12	X13
7	DI a.3	DI b.3	DO a.3	DO b.3
8	DI a.4	DI b.4	DO a.4	DO b.4
9	DI a.5	DI b.5	DO a.5	DO b.5
10	DI a.6	DI b.6	DO a.6	DO b.6
11	DI a.7	DI b.7	DO a.7	DQ b.7

A.9.5 SM 1223 (V AC) dijital giriş/çıkış modülünün teknik verileri

Tablo A-136 Genel teknik veriler

Model	**SM 1223 DI 8 x120/230 V AC / DQ 8 x röle**
Sipariş numarası	6ES7223-1QH32-0XB0
Boyutlar G x Y x D (mm)	45 x 100 x 75 mm
Ağırlık	190 gram
Performans kaybı	7,5 W
Akım tüketimi (SM veriyolu)	120 mA
Akım tüketimi (24 V DC)	Devreye alındığında, çıkış başına 11 mA

Tablo A-137 Dijital girişler

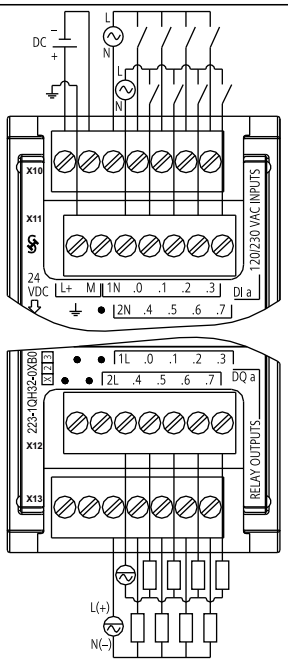
Model	**SM 1223 DI 8 x 120/230 V AC / DQ 8 x röle**
Girişlerin sayısı	8
Tip	IEC Tip 1
Nominal gerilim	6 mA'da 120 V AC, 9 mA'da 230 V AC
İzin verilen sürekli gerilim	264 V AC
Şok gerilim	--
Mantıksal sinyal 1 (min.)	2,5 mA'da 79 V AC
Mantıksal sinyal 0 (maks.)	1 mA'da 20 V AC
Kaçak akım (maks.)	1 mA
Elektriksel ayırma (alandan lojiğe)	1500 V AC
Potansiyeli ayrılmış gruplar[1]	4
Giriş gecikmesi süreleri	Tipik: 0,2 ila 12,8 msn, uygulayıcı tarafından ayarlanabilir Maksimum: -
Bağlantı 2 telli yakınlık sensörü (Bero) (maks.)	1 mA
Hat uzunluğu	Blendajsız: 300 metre Blendajlı: 500 metre
Eş zamanlı devreye alınmış girişlerin sayısı	8

[1] Bir grubun kanalları aynı faza sahip olmalıdır.

Tablo A-138 Dijital çıkışlar

Model	**SM 1223 DI 8 x 120/230 V AC / DQ 8 x röle**
Çıkışların sayısı	8
Tip	Röle, mekanik
Gerilim aralığı	5 ila 30 V DC veya 5 ila 250 V AC
Maks. akımdaki mantıksal sinyal 1	--
10 kΩ yükteki mantıksal sinyal 0	--
Akım (maks.)	2,0 A
Lamba yükü	30 W DC/200 W AC
AÇIK'taki kontak direnci	maks. 0,2 Ω, yeni ise
Çıkış başına kaçak akım	--
Ani akım	Kapalı kontaklarda 7 A
Aşırı yük koruması	Hayır
Elektriksel ayırma (alandan lojiğe)	1500 V AC (bobinden kontağa) Yok (bobin lojiği)
Potansiyeli ayrılmış gruplar	2
İletken başına akım (maks.)	10 A
Endüktif klemens gerilimi	--
Anahtarlama gecikmesi (maks.)	10 msn
Rölenin maksimum devre frekansı	1 Hz
Mekanik kullanım ömrü (yüksüz)	10.000.000 anahtarlama sayısı açık/kapalı
Nominal yükte kontakların kullanım ömrü	100.000 anahtarlama sayısı açık/kapalı
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet
Yedekli yük kumandası için paralel çıkışlar	Evet (aynı referans iletken ile)
Artırılmış yük için paralel çıkışlar	Hayır
Eş zamanlı devreye alınmış çıkışların sayısı	• 60 °C yatayda veya 50 °C dikeyde 4 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 8
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız

Tablo A-139 SM 1223 DI 8 x 120/230 V AC, DQ 8 x röle (6ES7223-1QH32-0XB0)



Tablo A-140 SM 1223 DI 8 x 120/240 V AC, DQ 8 x röle için bağlantı yerleşimi (6ES7223-1QH32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1N	2N	1L	2L
4	DI a.0	DI a.4	DO a.0	DO a.4
5	DI a.1	DI a.5	DO a.1	DO a.5
6	DI a.2	DI a.6	DO a.2	DO a.6
7	DI a.3	DI a.7	DO a.3	DO a.7

A.10 Analog sinyal modülleri (SM'ler)

A.10.1 SM 1231 analog giriş modülünün teknik verileri

Tablo A-141 Genel teknik veriler

Model	SM 1231 AI 4 x 13 Bit	SM 1231 AI 8 x 13 Bit	SM 1231 AI 4 x 16 Bit
Sipariş numarası	6ES7231-4HD32-0XB0	6ES7231-4HF32-0XB0	6ES7231-5ND32-0XB0
Boyutlar G x Y x D (mm)	45 x 100 x 75		
Ağırlık	180 gram		
Performans kaybı	2,2 W	2,3 W	2,0 W
Akım tüketimi (SM veriyolu)	80 mA	90 mA	80 mA
Akım tüketimi (24 V DC)	45 mA		65 mA

Tablo A-142 Analog girişler

Model	SM 1231 AI 4 x 13 Bit	SM 1231 AI 8 x 13 Bit	SM 1231 AI 4 x 16 Bit
Girişlerin sayısı	4	8	4
Tip	Gerilim veya akım (diferansiyel): 2'li gruplar halinde seçilebilir		Gerilim veya akım (diferansiyel)
Aralık	±10 V, ±5 V, ±2,5 V, 0 ila 20 mA veya 4 mA ila 20 mA		±10 V, ±5 V, ±2,5 V, ±1,25 V, 0 ila 20 mA veya 4 mA ila 20 mA
Tam sapma (veri sözcüğü)	-27648 ila 27648 Gerilim / 0 ila 27648 Akım		
Aşırı/düşük salınım aralığı (veri sözcüğü) Gerilim ve akım (Sayfa 1271) için analog giriş aralıkları alt bölümüne bakın.	Gerilim: 32511 ila 27649 / -27649 ila -32512 Akım: 32511 ila 27649 / 0 ila -4864		
Taşma/aşağı taşma (veri sözcüğü) Gerilim ve akım (Sayfa 1271) için giriş aralıkları alt bölümüne bakın.	Gerilim: 32767 ila 32512 / -32513 ila -32768 Akım 0 ila 20 mA: 32767 ila 32512 / -4865 ila -32768 Akım 4 ila 20 mA: 32767 ila 32512 (-4864'ün altındaki değerler tel kopmasını gösterir)		
Çözünürlük1	12 Bit + ön işaret Bit'i		15 Bit + ön işaret Bit'i
Maksimum dayanım gerilimi/akımı	±35 V / ±40 mA		
Düzleştirme	Yok, zayıf, orta veya güçlü Adım yanıt süreleri (Sayfa 1271) alt bölümüne bakın.		
Gürültü azaltma	400, 60, 50 veya 10 Hz Tarama hızları (Sayfa 1271) alt bölümüne bakın.		
Giriş empedansı Parametrelendirmeden önce Gerilim Akım	 ≥ 1 MΩ ≥ 9 MΩ, FS 06 ve üstü ≥ 1 MΩ ≥ 270 Ω, < 290 Ω		 ≥ 1 MΩ ≥ 1 MΩ < 315 Ω, > 280 Ω

[1] Bir kanala, işletim aralığının dışında gerilimler uygulanırsa, bu, diğer kanallarda parazite yol açabilir.

Model	SM 1231 AI 4 x 13 Bit	SM 1231 AI 8 x 13 Bit	SM 1231 AI 4 x 16 Bit
Elektriksel ayırma Alandan lojiğe Lojikten 24 V DC'ye Alandan 24 V DC'ye Kanaldan kanala	Yok		 707 V DC (tip kontrolü) 707 V DC (tip kontrolü) 500 V DC (tip kontrolü) Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın ±% 0,1 / ±% 0,2'si		Tam sapmanın ±% 0,1 / ±% 0,3'ü
Ölçme prensibi	Gerçek değer dönüştürme		
Ortak mod bastırma	40 dB, 60 Hz'teki nominal değer		
İşletim sinyal aralığı[1]	Sinyal artı ortak mod gerilimi, +12 V'tan küçük ve -12 V'tan büyük olmalıdır		
Hat uzunluğu (metre)	100 m, bükülmüş ve blendajlı		

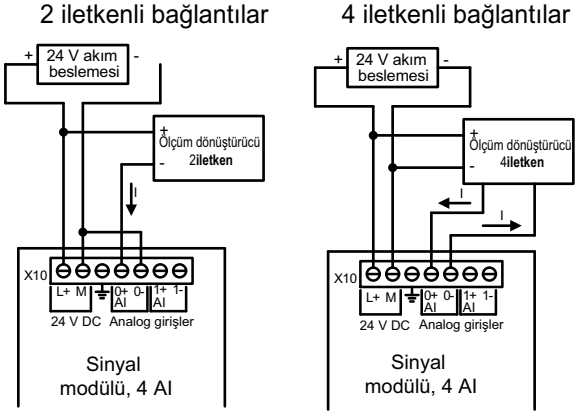
[1] Bir kanala, işletim aralığının dışında gerilimler uygulanırsa, bu, diğer kanallarda parazite yol açabilir.

Tablo A-143 Teşhis

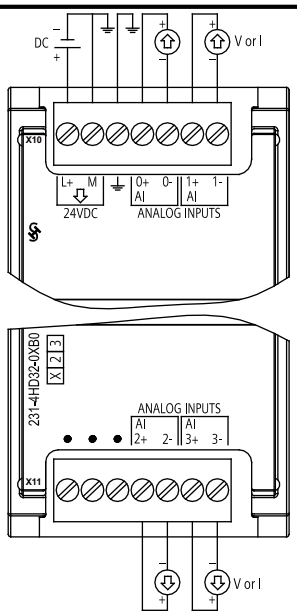
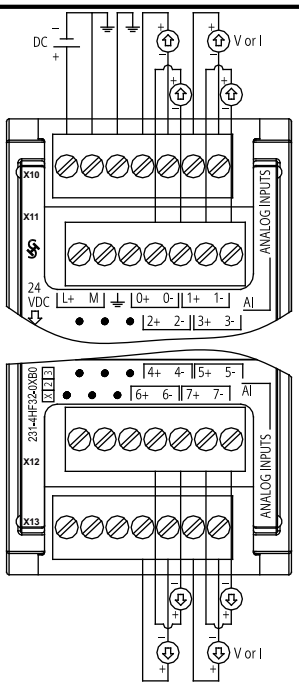
Model	SM 1231 AI 4 x 13 Bit	SM 1231 AI 8 x 13 Bit	SM 1231 AI 4 x 16 Bit
Taşma/aşağı taşma	Evet		
24 Volt DC alçak gerilim	Evet		
Tel kopması	Yalnızca 4 ila 20 mA aralığı (giriş -4864'ün altında olduğunda; 1,185 mA)		

SM 1231 akım ölçümü

Aşağıdaki görseldeki gibi akım ölçümünü, bir 2 veya 4 telli ölçüm dönüştürücü ile düzenleyebilirsiniz:



Tablo A-144 Analog giriş SM'lerinin devre şemaları

SM 1231 AI 4 x 13 Bit (6ES7231-4HD32-0XB0)	**SM 1231 AI 8 x 13 Bit (6ES7231-4HF32-0XB0)**
	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

Tablo A-145 SM 1231 AI 4 x 13 Bit için bağlantı yerleşimi (6ES7231-4HD32-0XB0)

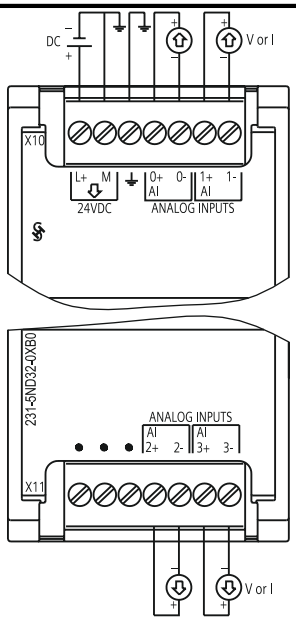
Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok
2	M / 24 V DC	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok
4	AI 0+	AI 2+
5	AI 0-	AI 2-
6	AI 1+	AI 3+
7	AI 1-	AI 3-

Tablo A-146 SM 1231 AI 8 x 13 Bit için bağlantı yerleşimi (6ES7231-4HF32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0+	AI 2+	AI 4+	AI 6+

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
5	AI 0-	AI 2-	AI 4-	AI 6-
6	AI 1+	AI 3+	AI 5+	AI 7+
7	AI 1-	AI 3-	AI 5-	AI 7-

Tablo A-147 Analog giriş SM'inin devre şeması

SM 1231 AI 4 x 16 Bit (6ES7231-5ND32-0XB0)	
	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

Tablo A-148 SM 1231 AI 4 x 16 Bit için bağlantı yerleşimi (6ES7231-5ND32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok
2	M / 24 V DC	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok
4	AI 0+	AI 2+
5	AI 0-	AI 2-
6	AI 1+	AI 3+
7	AI 1-	AI 3-

NOT

Kullanılmayan gerilim giriş kanallarına kısa devre yapın.

Kullanılmayan akım giriş kanalları 0 ila 20 mA aralığına ayarlanmalı ve/veya tel kopması bildirimi devre dışı bırakılmalıdır.

Akım modu için yapılandırılan girişler, yalnızca modül devreye alındığında ve yapılandırıldığında döngü akımını taşır.

Akım giriş kanalları, yalnızca göndericiye harici akım sağlandığında çalıştırılabilir.

A.10.2 SM 1232 analog çıkış modülünün teknik verileri

Tablo A-149 Genel teknik veriler

Teknik veriler	SM 1232 AO 2 x 14 Bit	SM 1232 AO 4 x 14 Bit
Sipariş numarası	6ES7232-4HB32-0XB0	6ES7232-4HD32-0XB0
Boyutlar (G x Y x D) (mm)	45 x 100 x 75	
Ağırlık	180 gram	
Performans kaybı	1,8 W	2,0 W
Akım tüketimi (SM veriyolu)	80 mA	
Akım tüketimi (24 V DC)	45 mA (yük olmadan)	

Tablo A-150 Analog çıkışlar

Teknik veriler	SM 1232 AO 2 x 14 Bit	SM 1232 AO 4 x 14 Bit
Çıkışlar	2	4
Tip	Gerilim veya akım	
Aralık	±10 V, 0 ila 20 mA veya 4 mA ila 20 mA	
Çözünürlük	Gerilim: 14 Bit Akım: 13 Bit	
Tam sapma (veri sözcüğü)	Gerilim: -27.648 ila 27.648; Akım: 0 ila 27.648 Gerilim ve akım (Sayfa 1272) için çıkış aralıkları alt bölümüne bakın.	
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % ±0,3 / % ±0,6'sı	
Yatışma süresi (değerin % 95'i)	Gerilim: 300 µsn (R), 750 µsn (1 uF) Akım: 600 µsn (1 mH), 2 msn (10 mH)	

Teknik veriler	SM 1232 AO 2 x 14 Bit	SM 1232 AO 4 x 14 Bit
Yük empedansı	Gerilim: ≥ 1000 Ω Akım: ≤ 600 Ω	
Maksimum çıkış kısa devre akımı	Gerilim modu: ≤ 24 mA Akım modu: ≥ 38,5 mA	
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	
Elektriksel ayırma (alandan lojiğe)	Yok	
Elektriksel ayırma (çıkışa 24 V)	Yok	
Hat uzunluğu (metre)	100 m, bükülmüş ve blendajlı	

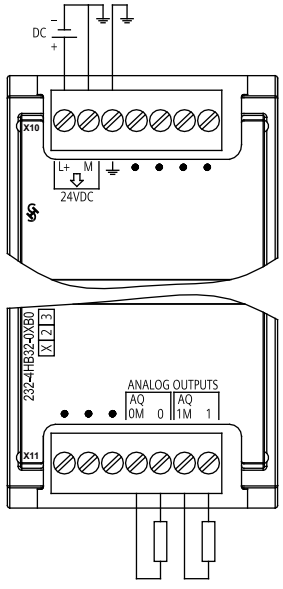
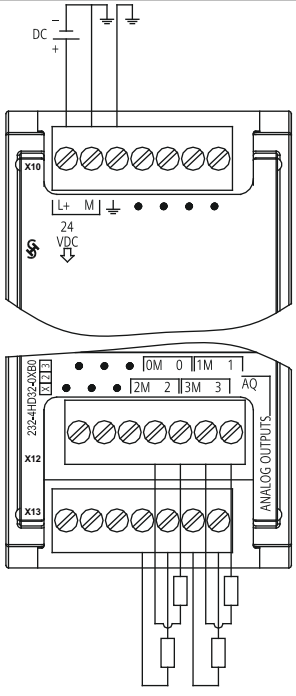
Tablo A-151 Teşhis

Teknik veriler	SM 1232 AO 2 x 14 Bit	SM 1232 AO 4 x 14 Bit
Taşma/aşağı taşma	Evet	
Toprakla kısa devre (yalnızca gerilim modu)	Evet	
Tel kopması (yalnızca akım modu)[1]	Evet	
24 V DC alçak gerilim[2]	Evet	

[1] Kısa devre tespiti ancak çıkış gerilimi -0,5 V'tan düşük veya +0,5 V'tan yüksek olduğunda mümkündür.

[2] Tel kopması algılaması yalnızca, çıkış akımı 1 mA'dan büyükse mümkündür.

Tablo A-152 Analog çıkış SM'lerinin devre şemaları

SM 1232 AQ 2 x 14 Bit (6ES7232-4HB32-0XB0)	SM 1232 AQ 4 x 14 Bit (6ES7232-4HD32-0XB0)
	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

Tablo A-153 SM 1232 AQ 2 x 14 Bit için bağlantı yerleşimi (6ES7232-4HB32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok
2	M / 24 V DC	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok
4	Bağlantı yok	AO 0M
5	Bağlantı yok	AO 0
6	Bağlantı yok	AO 1M
7	Bağlantı yok	AO 1

Tablo A-154 SM 1232 AQ 4 x 14 Bit için bağlantı yerleşimi (6ES7232-4HD32-0XB0)

Pin	X10 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
4	Bağlantı yok	AO 0M	AO 2M
5	Bağlantı yok	AO 0	AO 2
6	Bağlantı yok	AO 1M	AO 3M
7	Bağlantı yok	AO 1	AIO 3

A.10.3 SM 1234 analog giriş/analog çıkış modülünün teknik verileri

Tablo A-155 Genel teknik veriler

Teknik veriler	SM 1234 AI 4 x 13 Bit / AO 2 x 14 Bit
Sipariş numarası	6ES7234-4HE32-0XB0
Boyutlar (G x Y x D) (mm)	45 x 100 x 75
Ağırlık	220 gram
Performans kaybı	2,4 W
Akım tüketimi (SM veriyolu)	80 mA
Akım tüketimi (24 V DC)	60 mA (yük olmadan)

Tablo A-156 Analog girişler

Model	SM 1234 AI 4 x 13 Bit / AO 2 x 14 Bit
Girişlerin sayısı	4
Tip	Gerilim veya akım (diferansiyel): 2'li gruplar halinde seçilebilir
Aralık	±10 V, ±5 V, ±2,5 V, 0 ila 20 mA veya 4 mA ila 20 mA
Tam sapma (veri sözcüğü)	-27648 ila 27648
Aşırı/düşük salınım aralığı (veri sözcüğü)	Gerilim: 32511 ila 27649 / -27649 ila -32512 Akım: 32511 ila 27649 / 0 ila -4864 Gerilim ve akım (Sayfa 1271) için giriş aralıkları alt bölümüne bakın.

[1] Bir kanala, işletim aralığının dışında gerilimler uygulanırsa, bu, diğer kanallarda parazite yol açabilir.

Model	SM 1234 AI 4 x 13 Bit / AO 2 x 14 Bit
Taşma/aşağı taşma (veri sözcüğü)	Gerilim: 32767 ila 32512 / -32513 ila -32768 Akım: 32767 ila 32512 / -4865 ila -32768 Gerilim ve akım (Sayfa 1271) için giriş aralıkları alt bölümüne bakın.
Çözünürlük	12 Bit + ön işaret Bit'i
Maksimum dayanım gerilimi/akımı	±35 V / ±40 mA
Düzleştirme	Yok, zayıf, orta veya güçlü Adım yanıt süreleri (Sayfa 1271) alt bölümüne bakın.
Gürültü azaltma	400, 60, 50 veya 10 Hz Tarama hızları (Sayfa 1271) alt bölümüne bakın.
Giriş empedansı	≥ 9 MΩ, FS 07 ve üstü ≥ 1 MΩ (gerilim) / ≥ 270 Ω, < 290 Ω (akım)
Elektriksel ayırma (alandan lojiğe)	Yok
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın ±% 0,1 / ±% 0,2'si
Analog dijital dönüştürme süresi	625 µsn (400 Hz bastırma)
Ortak mod bastırma	40 dB, 60 Hz'teki nominal değer
İşletim sinyal aralığı[1]	Sinyal artı ortak mod gerilimi, +12 V'tan küçük ve -12 V'tan büyük olmalıdır
Hat uzunluğu (metre)	100 m, bükülmüş ve blendajlı

[1] Bir kanala, işletim aralığının dışında gerilimler uygulanırsa, bu, diğer kanallarda parazite yol açabilir.

Tablo A-157 Analog çıkışlar

Teknik veriler	SM 1234 AI 4 x 13 Bit / AO 2 x 14 Bit
Çıkışlar	2
Tip	Gerilim veya akım
Aralık	±10 V veya 0 ila 20 mA veya 4 mA ila 20 mA
Çözünürlük	Gerilim: 14 Bit, Akım: 13 Bit
Tam sapma (veri sözcüğü)	Gerilim: -27.648 ila 27.648; Akım: 0 ila 27648 Gerilim ve akım (Sayfa 1272) için çıkış aralıkları alt bölümüne bakın.
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın % ±0,3 / % ±0,6'sı
Yatışma süresi (değerin % 95'i)	Gerilim: 300 µsn (R), 750 µsn (1 uF) Akım: 600 µsn (1 mH), 2 msn (10 mH)
Yük empedansı	Gerilim: ≥ 1000 Ω Akım: ≤ 600 Ω
Maksimum çıkış kısa devre akımı	Gerilim modu: ≤ 24 mA Akım modu: ≥ 38,5 mA
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)
Elektriksel ayırma (alandan lojiğe)	Yok
Elektriksel ayırma (çıkışa 24 V)	Yok
Hat uzunluğu (metre)	100 m, bükülmüş ve blendajlı

Tablo A-158 Teşhis

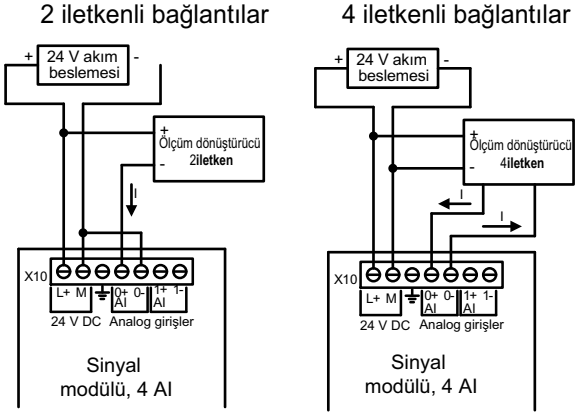
Model	**SM 1234 AI 4 x 13 Bit / AO 2 x 14 Bit**
Taşma/aşağı taşma	Evet
Toprakla kısa devre (yalnızca gerilim modu)[1]	Evet, çıkışlarda
Tel kopması (yalnızca akım modu)[2]	Evet, çıkışlarda
24 Volt DC alçak gerilim	Evet

1 Kısa devre tespiti ancak çıkış gerilimi -0,5 V'tan düşük veya +0,5 V'tan yüksek olduğunda mümkündür.

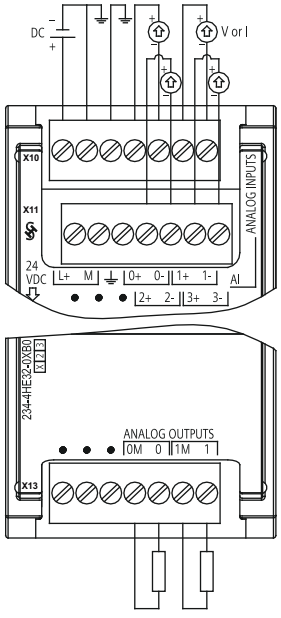
2 Tel kopması algılaması yalnızca, çıkış akımı 1 mA'dan büyükse mümkündür.

SM 1234 akım ölçümü

Aşağıdaki görseldeki gibi akım ölçümünü, bir 2 veya 4 telli ölçüm dönüştürücü ile düzenleyebilirsiniz:



Tablo A-159 Analog giriş/çıkış SM'inin devre şemaları

SM 1234 AI 4 x 13 Bit / AQ 2 x 14 Bit (6ES7234-4HE32-0XB0)	
	

SM 1234 AI 4 x 13 Bit / AQ 2 x 14 Bit (6ES7234-4HE32-0XB0)	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

Tablo A-160 SM 1234 AI 4 x 13 Bit / AQ 2 x 14 Bit için bağlantı yerleşimi (6ES7234-4HE32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X13 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok
4	AI 0+	AI 2+	AO 0M
5	AI 0-	AI 2-	AO 0
6	AI 1+	AI 3+	AO 1M
7	AI 1-	AI 3-	AO 1

NOT

Kullanılmayan gerilim giriş kanallarına kısa devre yapın.

Kullanılmayan akım giriş kanalları 0 ila 20 mA aralığına ayarlanmalı ve/veya tel kopması bildirimi devre dışı bırakılmalıdır.

Akım modu için yapılandırılan girişler, yalnızca modül devreye alındığında ve yapılandırıldığında döngü akımını taşır.

Akım giriş kanalları, yalnızca göndericiye harici akım sağlandığında çalıştırılabilir.

A.10.4 Analog girişlerin adım yanıtı

Tablo A-161 Adım yanıtı (msn), 0 ila tam sapmaya kadar, %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	**Gürültü azaltma/bastırma frekansı (entegrasyon süresi seçimi)**			
	400 Hz (2,5 msn)	**60 Hz (16,6 msn)**	**50 Hz (20 msn)**	**10 Hz (100 msn)**
Yok (1 çevrim): Ortalama alma yok	4 msn	18 msn	22 msn	100 msn
Zayıf (4 çevrim): 4 tarama değeri	9 msn	52 msn	63 msn	320 msn
Orta (16 çevrim): 16 tarama değeri	32 msn	203 msn	241 msn	1200 msn
Güçlü (32 çevrim): 32 tarama değeri	61 msn	400 msn	483 msn	2410 msn

A.10.5 Analog girişlerin tarama süresi ve güncelleme süreleri

Tablo A-162 Tüm kanallar için tarama süresi ve modül güncelleme süreleri

Bastırma frekansı (entegrasyon süresi)	**Tüm kanallar için tarama süresi ve modül güncelleme süreleri**			
	400 Hz (2,5 msn)	**60 Hz (16,6 msn)**	**50 Hz (20 msn)**	**10 Hz (100 msn)**
SM 4 kanal × 13 Bit	0,625 msn	4,17 msn	5 msn	25 msn
SM 8 kanal × 13 Bit	1,25 msn	4,17 msn	5 msn	25 msn
SM 4 kanal × 16 Bit	0,417 msn	0,397 msn	0,400 msn	0,400 msn

A.10.6 Gerilim ve akım için analog girişlerin ölçüm aralıkları (SB ve SM)

Tablo A-163 Gerilim için analog girişi gösterimi (SB ve SM)

Sistem		**Gerilim ölçüm aralığı**				
Desimal	**Onaltılı**	**±10 V**	**±5 V**	**±2,5 V**	**±1,25 V**	
32767	7FFF[1]	11,851 V	5,926 V	2,963 V	1,481 V	Taşma
32512	7F00					
32511	7EFF	11,759 V	5,879 V	2,940 V	1,470 V	Aşırı salınım aralığı
27649	6C01					
27648	6C00	10 V	5 V	2,5 V	1,250 V	Ölçüm aralığı
20736	5100	7,5 V	3,75 V	1,875 V	0,938 V	
1	1	361,7 µV	180,8 µV	90,4 µV	45,2 µV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-20736	AF00	-7,5 V	-3,75 V	-1,875 V	-0,938 V	
-27648	9400	-10 V	-5 V	-2,5 V	-1,250 V	
-27649	93FF					Düşük salınım aralığı
-32512	8100	-11,759 V	-5,879 V	-2,940 V	-1,470 V	
-32513	80FF					Aşağı taşma
-32768	8000	-11,851 V	-5,926 V	-2,963 V	-1,481 V	

[1] 7FFF aşağıdaki nedenlerden biri için geri döndürülebilir: Geçerli değerler mevcut olmadan önce (örneğin yol alma sırasında hemen) veya bir tel kopması algılandığında taşma (tabloya bakın).

Tablo A-164 Akım için analog girişlerin gösterimi (SB ve SM)

Sistem		**Akım ölçüm aralığı**		
Desimal	**Onaltılı**	**0 mA ila 20 mA**	**4 mA ila 20 mA**	
32767	7FFF	> 23,52 mA	> 22,81 mA	Taşma
32511	7EFF	23,52 mA	22,81 mA	Aşırı salınım aralığı
27649	6C01			
27648	6C00	20 mA	20 mA	Nominal aralık
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF			Düşük salınım aralığı
-4864	ED00	-3,52 mA	1,185 mA	
32767[1]	7FFF		< 1,185 mA	Tel kopması (4 ila 20 mA) (Sayfa 1151)
-32768	8000	< -3,52 mA		Aşağı taşma (0 ila 20 mA)

[1] Tel kopması alarmının durumundan bağımsız olarak 32767 (16#7FFF) tel kopma değeri her zaman geri döndürülür.

A.10.7 Gerilim ve akım için analog çıkışların ölçüm aralıkları (SB ve SM)

Tablo A-165 Gerilim için analog çıkışı gösterimi (SB ve SM)

Sistem		Gerilim çıkış aralığı	
Desimal	**Onaltılı**	**±10 V**	
32767	7FFF	Bkz. Not 1	Taşma
32512	7F00	Bkz. Not 1	
32511	7EFF	11,76 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
1	1	361,7 µV	
0	0	0 V	
-1	FFFF	-361,7 µV	
-20736	AF00	-7,5 V	
-27648	9400	-10 V	
-27649	93FF		Düşük salınım aralığı
-32512	8100	-11,76 V	
-32513	80FF	Bkz. Not 1	Aşağı taşma
-32768	8000	Bkz. Not 1	

[1] Taşma veya aşağı taşma durumunda, analog çıkışlar STOP işletim durumu için yedek değeri alır.

Tablo A-166 Akım için analog çıkışı gösterimi (SB ve SM)

Sistem		Akım çıkışı aralığı		
Desimal	**Onaltılı**	**0 mA ila 20 mA**	**4 mA ila 20 mA**	
32767	7FFF	Bkz. Not 1	Bkz. Not 1	Taşma
32512	7F00	Bkz. Not 1	Bkz. Not 1	
32511	7EFF	23,52 mA	22,81 mA	Aşırı salınım aralığı
27649	6C01			
27648	6C00	20 mA	20 mA	Ölçüm aralığı
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF		4 mA ila 578,7 nA	Düşük salınım aralığı
-6912	E500		0 mA	
-6913	E4FF			Mümkün değil. Çıkış değeri 0 mA ile sınırlıdır.
-32512	8100			
-32513	80FF	Bkz. Not 1	Bkz. Not 1	Aşağı taşma
-32768	8000	Bkz. Not 1	Bkz. Not 1	

[1] Taşma veya aşağı taşma durumunda, analog çıkışlar STOP işletim durumu için yedek değeri alır.

A.11 Termo eleman ve RTD sinyal modülleri (SM'ler)

A.11.1 SM 1231 termo eleman

Tablo A-167 Genel teknik veriler

Model	**SM 1231 AI 4 x 16 Bit TC**	**SM 1231 AI 8 x 16 Bit TC**
Sipariş numarası	6ES7231-5QD32-0XB0	6ES7231-5QF32-0XB0
Boyutlar G x Y x D (mm)	45 x 100 x 75	
Ağırlık	180 gram	190 gram
Performans kaybı	1,5 W	
Akım tüketimi (SM veriyolu)	80 mA	
Akım tüketimi (24 V DC) [1]	40 mA	

[1] 20,4 ila 28,8 V DC (sınıf 2, güç sınırlı veya PLC'den enkoder gerilimi)

Tablo A-168 Analog girişler

Model		**SM 1231 AI 4 x 16 Bit TC**	**SM 1231 AI 8 x 16 Bit TC**
Girişlerin sayısı		4	8
Aralık Nominal aralık (veri sözcüğü) Taşma/aşağı taşma (veri sözcüğü) Taşma/aşağı taşma (veri sözcüğü)		Bkz. Termo eleman seçenekler tablosu (Sayfa 1277).	
Çözünürlük	Sıcaklık aralığı	0,1 °C / 0,1 °F	
	Gerilim	15 Bit artı ön işaret	
Maks. dayanım gerilimi		±35 V	
Gürültü azaltma		Seçili filtre ayarı için 85 dB (10 Hz, 50 Hz, 60 Hz veya 400 Hz)	
Ortak mod bastırma		120 V AC'de > 120 dB	
Empedans		≥ 10 MΩ	
Elektriksel ayırma	Alandan lojiğe	707 V DC (tip kontrolü)	
	Alandan 24 V DC'ye	707 V DC (tip kontrolü)	
	24 V DC'den lojiğe	707 V DC (tip kontrolü)	
Kanaldan kanala		120 V AC	
Hassasiyet		Bkz. Termo eleman seçenekler tablosu (Sayfa 1277).	
Tekrarlama hassasiyeti		± % 0,05 tam sapma	
Ölçme prensibi		Entegre edici	
Modül güncelleme süresi		Bkz. Gürültü azaltma seçenekler tablosu (Sayfa 1277).	
Hata, soğuk bağlantı noktası		±1,5 °C	
Hat uzunluğu (metre)		Enkodere maks. 100 m	
Hat direnci		Maks. 100 Ω	

Tablo A-169 Teşhis

Model	SM 1231 AI 4 x 16 Bit TC	SM 1231 AI 8 x 16 Bit TC
Taşma/aşağı taşma[1]	Evet	
Tel kopması[2, 3]	Evet	
24 V DC alçak gerilim [1]	Evet	

[1] Modül projelendirmesi sırasında ilgili alarmlar devre dışı bırakılırsa, "Taşma", "Aşağı taşma" ve "Alçak gerilim" teşhis alarmlarından gelen bilgiler de analog veri değerleriyle bildirilir.

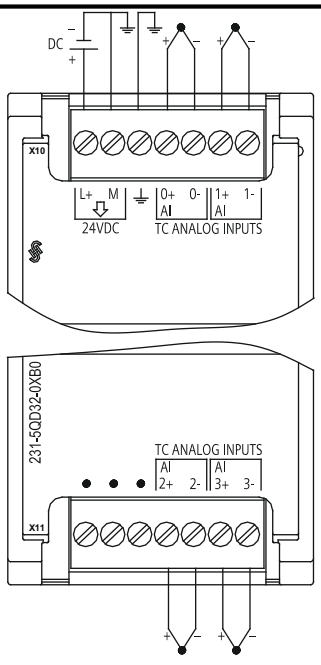
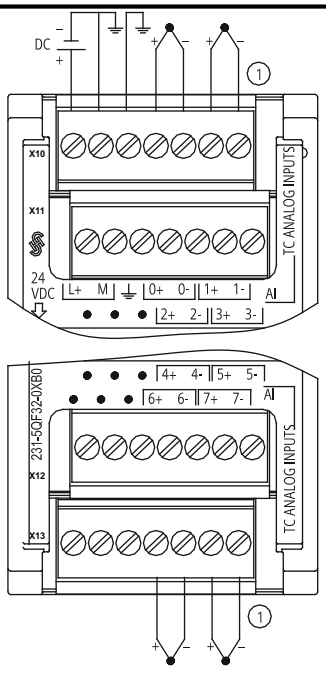
[2] Tel kopması alarmı devre dışı bırakılmışsa ve enkoder kablajında açıkta bir tel mevcutsa, modül rastlantısal değerler bildirebilir.

[3] Modül, tel kopma kontrolünü her 6 saniyede bir yürütür; bunun sonucunda, etkinleştirilmiş her kanal için güncelleme süresini her 6 saniyede bir 9 msn uzatır.

SM 1231 TC analog termo eleman sinyal modülü, modül girişlerine bağlı gerilimin değerini ölçer. Sıcaklık ölçüm tipi olarak "termo eleman" veya "gerilim" mümkündür.

- "Termo eleman": Ölçüm değeri, derece cinsinden on ile çarpılarak yayınlanır (örneğin: 25,3 derece, ondalık değer 253 olarak gösterilir).
- "Gerilim": Nominal aralıktaki ölçüm aralığı değeri, 27648 ondalıktır.

Tablo A-170 Termo eleman SM'lerinin devre şemaları

SM 1231 AI 4 x TC 16 Bit (6ES7231-5QD32-0XB0)	SM 1231 AI 8 x TC-Bit (6ES7231-5QF32-0XB0)
	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

① TC 2, 3, 4 ve 5'in bağlantıları, anlaşılırlık açısından gösterilmemiştir.

Tablo A-171 SM 1231 AI 4 x TC 16 Bit için bağlantı yerleşimi (6ES7231-5QD32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok
2	M / 24 V DC	Bağlantı yok

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)
3	Fonksiyonel topraklama	Bağlantı yok
4	AI 0+ /TC	AI 2+ /TC
5	AI 0- /TC	AI 2- /TC
6	AI 1+ /TC	AI 3+ /TC
7	AI 1- /TC	AI 3- /TC

Tablo A-172 SM 1231 AI 8 x TC Bit için bağlantı yerleşimi (6ES7231-5QF32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0+ /TC	AI 2+ /TC	AI 4+ /TC	AI 6+ /TC
5	AI 0- /TC	AI 2- /TC	AI 4- /TC	AI 6- /TC
6	AI 1+ /TC	AI 3+ /TC	AI 5+ /TC	AI 7+ /TC
7	AI 1- /TC	AI 3- /TC	AI 5- /TC	AI 7- /TC

NOT

Kullanılmayan analog girişlere kısa devre yapın.

Termo eleman tarafından kullanılmayan kanallar, devre dışı bırakılabilir. Kullanılmayan bir kanal devre dışı bırakılırsa, hata oluşmaz.

A.11.1.1 Bir termo elemanın temel çalışma şekli

İki farklı metalin elektriksel olarak birbirine bağlanmasıyla termo elemanlar oluşur. Burada bağlantı noktasının sıcaklığıyla orantılı bir gerilim üretilir. Bu düşük bir gerilimdir. Bir mikrovolt, çok sayıda dereceyi temsil edebilir. Termo elemanlarla sıcaklık ölçüm yönteminin temeli, bir termo elemanın gerilimini ölçmek, ek bağlantı noktalarını kompanse etmek ve sonuçları doğrusallaştırmaktır.

SM 1231 termo eleman modülüne bir termo eleman bağladığınızda, farklı metallerin iki ucu modülün sinyal bağlantısına bağlanır. İki farklı ucun birbirine bağlandığı noktada termo elemanın sensörü bulunur.

Farklı uçların sinyal bağlantısına bağlandığı noktada iki termo eleman daha oluşturulur. Bağlantının sıcaklığı bir gerilim oluşturur ve bu da termo eleman sensörünün gerilimine eklenir. Bu gerilim kompanse edilmezse, yayınlanan sıcaklık, sensörün sıcaklığından sapacaktır.

Soğuk bağlantının kompanzasyonu sayesinde, termo eleman bağlantı noktasında kompanse edilir. Termo elemanlar için tablolar, bağlantı noktasında genellikle sıfır Santigrat derece olan bir referans sıcaklığını baz almaktadır. Soğuk bağlantı kompanzasyonu sayesinde, bağlantı, sıfır Santigrat dereceye kompanse edilir. Bağlantının termo elemanı ile eklenen gerilim, soğuk bağlantı noktasının kompanzasyonu ile düzeltilir. Modül sıcaklığı, dahili olarak ölçülür ve

bunun ardından sensör dönüşümüne eklenecek bir değere dönüştürülür. Düzeltilmiş sensör dönüşümü, daha sonra termo eleman tabloları aracılığıyla doğrusallaştırılır.

Soğuk bağlantı noktasının kompanzasyonunun optimum seviyede gerçekleşmesi için termo eleman modülünün termik stabil bir çevrede olması gerekir. Modül çevresindeki sıcaklıkta yavaş değişim (0,1 °C/dakikadan az), modül spesifikasyonu dahilinde doğru kompanse edilecektir. Modül üzerindeki hava hareketleri de soğuk bağlantı noktasının kompanzasyonunda hatalara yol açar.

Soğuk bağlantı noktası hatalarının daha iyi kompanzasyonu gerekli ise, harici bir izotermal klemens şeridi kullanabilirsiniz. 0 °C'ye referanslı bir klemens şeridi veya 50 °C'ye referanslı bir klemens şeridi, termo eleman modülü ile kullanılabilir.

A.11.1.2 SM 1231 termo eleman için seçenekler tablosu

SM 1231 termo eleman sinyal modülü tarafından desteklenen çeşitli termo eleman tiplerinin aralıklarını ve hassasiyetini aşağıdaki tabloda bulabilirsiniz.

Tablo A-173 Termo eleman seçenekler tablosu

Tip	Minimum alt aralık[1]	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[2]	Hassasiyet normal aralığı [3, 4] ,25 °C'de	Normal aralık [1, 2, 6] Hassasiyet -20 °C ila 60 °C
J	-210,0 °C	-150,0 °C	1.200,0 °C	1.450,0 °C	±0,3 °C	±0,6 °C
	-346,0 °F	-238,0 °F	2192,0 °F	2642,0 °F	±0,5 °F	±1,1 °F
K	-270,0 °C	-200,0 °C	1.372,0 °C	1.622,0 °C	±0,4 °C	±1,0 °C
	-454,0 °F	-328,0 °F	2501,6 °F	2951,6 °F	±0,7 °F	±1,8 °F
T	-270,0 °C	-200,0 °C	400,0 °C	540,0 °C	±0,5 °C	±1,0 °C
	-454,0 °F	-328,0 °F	752,0 °F	1004,0 °F	±0,9 °F	±1,8 °F
E	-270,0 °C	-200,0 °C	1.000,0 °C	1.200,0 °C	±0,3 °C	±0,6 °C
	-454,0 °F	-328,0 °F	1832,0 °F	2192,0 °F	±0,5 °F	±1,1 °F
R & S	-50,0 °C	100,0 °C	1.768,0 °C	2.019,0 °C	±1,0 °C	±2,5 °C
	-58,0 °C	212,0 °F	3214,4 °F	3276,6 °F [5]	±1,8 °F	±4,5 °F
B	0,0 °C	200,0 °C	800,0 °C	--	±2,0 °C	±2,5 °C
	32,0 °F	392,0 °F	1472,0 °F	--	±3,6 °F	±4,5 °F
	--	800,0 °C	1.820,0 °C	1.820,0 °C	±1,0 °C	±2,3 °C
	--	1472,0 °F	3276,6 °F [5]	3276,6 °F [5]	±1,8 °F	±4,1 °F
N	-270,0 °C	-200,0 °C	1.300,0 °C	1.550,0 °C	±1,0 °C	±1,6 °C
	-454,0 °F	-328,0 °F	2372,0 °F	2822,0 °F	±1,8 °F	±2,9 °F
C	0,0 °C	100,0 °C	2.315,0 °C	2.500,0 °C	±0,7 °C	±2,7 °C
	32,0 °F	212,0 °F	3276,6 °F [5]	3276,6 °F [5]	±1,3 °F	±4,9 °F

1 Alt aralık için minimumun altındaki termo eleman değerleri -32768 olarak belirtilir.

2 Üst aralık için maksimumun üzerindeki termo eleman değerleri +32767 olarak yayınlanır.

3 Soğuk bağlantı noktasındaki dahili hata, tüm aralıklar için ±1,5 °C'dir. Bu değeri, bu tabloda sıralanan hatalara eklemelisiniz. Modül, burada belirtilen spesifikasyonlar karşılanana dek, en azından 30 dakikalık bir ısınma süresine ihtiyaç duyar. -10 °C'nin altındaki ortam sıcaklıklarında soğuk bağlantı noktasındaki dahili hata, 1,5°C'den fazla olabilir.

4 Yayılan 970 MHz ila 990 MHz arası bir radyo frekansında, SM 1231 AI 4 x 16 Bit TC'nin hassasiyeti bozulabilir.

Tip	Minimum alt aralık[1]	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[2]	Hassasiyet normal aralığı [3, 4] ,25 °C'de	Normal aralık [1, 2, 6] Hassasiyet -20 °C ila 60 °C
TXK/XK(L)	-200,0 °C	-150,0 °C	800,0 °C	1.050,0 °C	±0,6 °C	±1,2 °C
	-328,0 °F	302,0 °F	1472,0 °F	1922,0 °F	±1,1 °F	±2,2 °F
Gerilim	-32512	-27648 -80 mV	27648 80 mV	32511	± % 0,05	± % 0,1

1 Alt aralık için minimumun altındaki termo eleman değerleri -32768 olarak belirtilir.

2 Üst aralık için maksimumun üzerindeki termo eleman değerleri +32767 olarak yayınlanır.

3 Soğuk bağlantı noktasındaki dahili hata, tüm aralıklar için ±1,5 °C'dir. Bu değeri, bu tabloda sıralanan hatalara eklemelisiniz. Modül, burada belirtilen spesifikasyonlar karşılanana dek, en azından 30 dakikalık bir ısınma süresine ihtiyaç duyar. -10 °C'nin altındaki ortam sıcaklıklarında soğuk bağlantı noktasındaki dahili hata, 1,5°C'den fazla olabilir.

4 Yayılan 970 MHz ila 990 MHz arası bir radyo frekansında, SM 1231 AI 4 x 16 Bit TC'nin hassasiyeti bozulabilir.

5 °F cinsinden mesajla birlikte alt limit 3276,6

6 Soğuk bağlantıdaki kompanzasyon hatası, 0 °C'nin altındaki ortam sıcaklığı için karakterize edilmemiştir ve belirtilen değeri aşabilir.

NOT

Termo eleman kanalı

Termo eleman sinyal modülündeki her kanal için farklı bir termo eleman tipi yapılandırılabilir (ayarlama işlemi, modül yapılandırması sırasında yazılımda gerçekleştirilebilir).

Tablo A-174 SM 1231 termo eleman için gürültü azaltma ve güncelleme süreleri

Bastırma frekansı seçimi	Entegrasyon süresi	4 kanallı modül güncelleme süresi (saniye)	8 kanallı modül güncelleme süresi (saniye)
400 Hz (2,5 msn)	10 msn [1]	0,143	0,285
60 Hz (16,6 msn)	16,67 msn	0,223	0,445
50 Hz (20 msn)	20 msn	0,263	0,525
10 Hz (100 msn)	100 msn	1,225	2,450

1 400 Hz filtre seçiliyken, modülün çözünürlüğünü ve hassasiyetini korumak için gereken entegrasyon süresi 10 msn'dir. Bu seçim, 100 Hz ve 200 Hz'deki gürültüyü de bastırır.

Termo elemanların ölçümü için 100 msn'lik bir entegrasyon süresi önerilir. Daha düşük ayarlanmış entegrasyon süreleri sıcaklığın daha yüksek tekrarlama hassasiyetine neden olur.

NOT

Güç açıldıktan sonra modül, A/D dönüştürücü için dahili kalibrasyonu yürütür. Bu zaman diliminde modül, ilgili kanal için geçerli veriler mevcut olana dek her kanalda 32767 değerini bildirir. Bu başlatma süresinin kullanıcı programında dikkate alınması gerekli olabilir. Modülün konfigürasyonu başlatma işleminin süresini etkileyebileceğinden, konfigürasyonunuzda modülün davranışını kontrol edin. Gerekli olursa, modülün başlatma süresine uyum sağlamak için kullanıcı programınıza lojik ekleyebilirsiniz.
Bu lojiği, başlatma tamamlanana dek işletimi bloke edecek olan "yol alma OB'sindeki" bir okuma talebi ile uygulayabilirsiniz. Okuma talebini, anında erişim ile uygulamalısınız. Termo elemanın okuma talebinin değeri 32767 ise, bu değer değişene dek okuma işlemi tekrarlanmalıdır. Her modül için, bu sorgulamanın yalnızca modülün en büyük numaralı girişi için yürütülmesi gerekir (modül girişleri, 0 ile 7 arasındaki sırayla başlatılır).

J tipi bir termo eleman için analog değerlerin gösterimi

J tipi bir termo eleman için analog değerleri, aşağıdaki tabloda bulabilirsiniz.

Tablo A-175 J tipi termo elemanlar için analog değerlerin gösterimi

°C cinsinden Tip J	Birimler		°F cinsinden Tip J	Birimler		Aralık
	Desimal	Onaltılı		Desimal	Onaltılı	
> 1450,0	32767	7FFF	> 2642,0	32767	7FFF	Taşma
1450,0 : 1200,1	14500 : 12001	38A4 : 2EE1	2642,0 : 2192,2	26420 : 21922	6734 : 55A2	Aşım
1200,0 : -150	12000 : 1500	2EE0 : FA24	2192,0 : -238,0	21920 : -2380	55A0 : F6B4	Ölçüm aralığı
-150,1 : -210	-1501 : -2100	FA23 : F7CC	-238,1 : -346,0	-2381 : -3460	F6B3 : F27C	Aşağı aşım
< -210,0	-32768	8000	< -346,0	-32768	8000	Aşağı taşma[1]

[1] Hatalı kablaj (örn. yanlış kutup bağlama veya açık girişler) veya negatif aralıktaki bir sensör hatası (örn. yanlış termo eleman tipi) termo eleman modülünün aşağı taşma bildirmesine yol açabilir.

A.11.2 SM 1231 RTD

SM 1231 RTD'nin teknik verileri

Tablo A-176 Genel teknik veriler

Teknik veriler	SM 1231 AI 4 x RTD x 16 Bit	SM 1231 AI 8 x RTD x 16 Bit
Sipariş numarası	6ES7231-5PD32-0XB0	6ES7231-5PF32-0XB0
Boyutlar G x Y x D (mm)	45 x 100 x 75	70 x 100 x 75
Ağırlık	220 gram	270 gram
Performans kaybı	1,5 W	
Akım tüketimi (SM veriyolu)	80 mA	90 mA
Akım tüketimi (24 V DC) [1]	40 mA	

[1] 20,4 ila 28,8 V DC (sınıf 2, güç sınırlı veya CPU'dan enkoder gerilimi)

Tablo A-177 Analog girişler

Teknik veriler		SM 1231 AI 4 x RTD x 16 Bit	SM 1231 AI 8 x RTD x16 Bit
Girişlerin sayısı		4	8
Tip		Modül referansı RTD ve Ω	
Aralık Nominal aralık (veri sözcüğü) Aşırı/düşük salınım aralığı (veri sözcüğü) Taşma/aşağı taşma (veri sözcüğü)		Bkz. RTD enkoder seçenekler tablosu (Sayfa 1283).	
Çözünürlük	Sıcaklık	0,1 °C / 0,1 °F	
	Direnç	15 Bit artı ön işaret	
Maks. dayanım gerilimi		±35 V	
Gürültü azaltma		Seçili gürültü azaltma için 85 dB (10 Hz, 50 Hz, 60 Hz veya 400 Hz)	
Ortak mod bastırma		> 120 dB	
Empedans		≥ 10 MΩ	
Elektriksel ayırma	Alan tarafından lojiğe	707 V DC (tip kontrolü)	
	Alandan 24 V DC'ye	707 V DC (tip kontrolü)	
	24 V DC'den lojiğe	707 V DC (tip kontrolü)	
Kanalları kendi içinde ayırma		Yok	
Hassasiyet		Bkz. RTD enkoder seçenekler tablosu (Sayfa 1283).	
Tekrarlama hassasiyeti		± % 0,05 tam sapma	
Enkoderin maksimum kayıp gücü		0,5 mW	
Ölçme prensibi		Entegre edici	
Modül güncelleme süresi		Bkz. Gürültü azaltma seçenekler tablosu (Sayfa 1283).	
Hat uzunluğu (metre)		Enkodere maks. 100 m	
Hat direnci		20 Ω, maks. 10 Ω RTD için 2,7 Ω	

Tablo A-178 Teşhis

Teknik veriler	SM 1231 AI 4 x RTD x 16 Bit	SM 1231 AI 8 x RTD x16 Bit
Taşma/aşağı taşma [1,2]	Evet	
Tel kopması[3]	Evet	
24 V DC alçak gerilim [1]	Evet	

[1] Modül projelendirmesi sırasında ilgili alarmlar devre dışı bırakılırsa, "Taşma", "Aşağı taşma" ve "Alçak gerilim" teşhis alarmlarından gelen bilgiler de analog veri değerleriyle bildirilir.

[2] Direnç ölçüm aralıkları için aşağı taşma denetimi genellikle etkinleştirilmez.

[3] Tel kopması alarmı devre dışı bırakılmışsa ve enkoder kablajında açıkta bir tel mevcutsa, modül rastlantısal değerler bildirebilir.

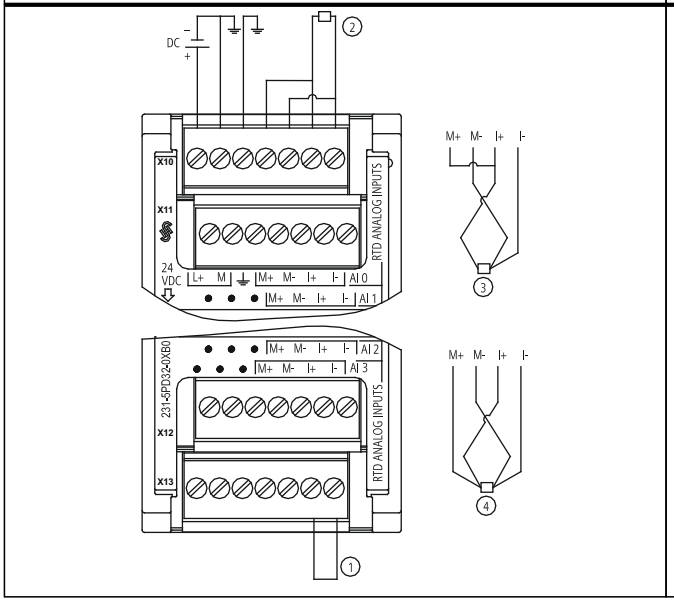
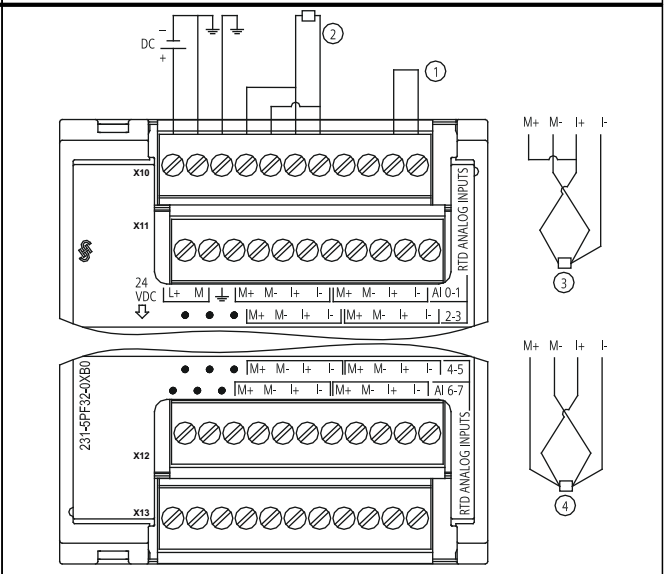
SM 1231 RTD analog sinyal modülü, modül girişlerine bağlı gerilimin değerini ölçer. Ölçüm türü olarak "direnç" veya "termik direnç" seçilebilir.

- "Direnç": Nominal aralıktaki ölçüm aralığı değeri, 27648 ondalıktır.
- "Termik direnç": Ölçüm değeri, derece cinsinden on ile çarpılarak yayınlanır (örneğin: 25,3 derece, ondalık değer 253 olarak gösterilir). İklimsel aralığın değerleri, yüz ile çarpılarak derece cinsinden verilir (örnek: 25,34 derece, ondalık değer 2534 olarak gösterilir).

SM 1231 RTD modülü sensör direncini ölçmek için bir akım kullanır. Bu akımın genliği ölçülen dirence göre değişir. Akımın maksimum değeri, sıfır Ohm direncinde 0,48 mA'dır. Akımın minimum değeri 4400 Ohm'luk bir RTD sensöründe 0,34 mA'dır. Akım darbeleri, modüldeki aktif kanal sayısına bağlı olarak değişir. Eğer iki kanal aktifse, bir kanal için devreye alma süresi %50'dir. Sekiz kanal aktifse, bir kanal için devreye alma süresi %12,5'tir. Kesintiye uğramış elektrik devreli sensördeki maksimum çıkış gerilimi, 5 V DC'dir (I+ ve I- klemensleri arasında ölçülür).

SM 1231 RTD sinyal modülü, enkoder direncine olan 2 telli, 3 telli ve 4 telli bağlantılar üzerinden ölçümleri destekler.

Tablo A-179 RTD-SM'lerinin devre şemaları

SM 1231 RTD 4 x 16 Bit (6ES7231-5PD32-0XB0)	SM 1231 RTD 8 x 16 Bit (6ES7231-5PF32-0XB0)
	

① Kullanılmayan RTD girişlerini geri döngüye alın

② 2 telli RTD ③ 3 telli RTD ④ 4 telli RTD

NOT: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.

Tablo A-180 SM SM 1231 RTD 4 x 16 Bit için bağlantı yerleşimi (6ES7231-5PD32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0 M+ /RTD	AI 1 M+ /RTD	AI 2 M+ /RTD	AI 3 M+ /RTD
5	AI 0 M- /RTD	AI 1 M- /RTD	AI 2 M- /RTD	AI 3 M- /RTD
6	AI 0 I+ /RTD	AI 1 I+ /RTD	AI 2 I+ /RTD	AI 3 I+ /RTD
7	AI 0 I- /RTD	AI 1 I- /RTD	AI 2 I- /RTD	AI 3 I- /RTD

Tablo A-181 SM 1231 RTD 8 x 16 Bit için bağlantı yerleşimi (6ES7231-5PF32-0XB0)

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Fonksiyonel topraklama	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0 M+ /RTD	AI 2 M+ /RTD	AI 4 M+ /RTD	AI 6 M+ /RTD
5	AI 0 M- /RTD	AI 2 M- /RTD	AI 4 M- /RTD	AI 6 M- /RTD

Pin	X10 (altın kaplamalı)	X11 (altın kaplamalı)	X12 (altın kaplamalı)	X13 (altın kaplamalı)
6	AI 0 I+ /RTD	AI 2 I+ /RTD	AI 4 I+ /RTD	AI 6 I+ /RTD
7	AI 0 I- /RTD	AI 2 I- /RTD	AI 4 I- /RTD	AI 6 I- /RTD
8	AI 1 M+ /RTD	AI 3 M+ /RTD	AI 5 M+ /RTD	A7 M+ /RTD
9	AI 1 M- /RTD	AI 3 M- /RTD	AI 5 M- /RTD	AI 7 M- /RTD
10	AI 1 I+ /RTD	AI 3 I+ /RTD	AI 5 I+ /RTD	AI 7 I+ /RTD
11	AI 1 I- /RTD	AI 3 I- /RTD	AI 5 I- /RTD	AI 7 I- /RTD

NOT

RTD tarafından kullanılmayan kanallar, devre dışı bırakılabilir. Kullanılmayan bir kanal devre dışı bırakılırsa, hata oluşmaz.

RTD modülü, bir kanal kullanılmıyorsa ve devre dışı bırakılmamışsa, otomatik olarak ortaya çıkan ek stabilizasyon süresini gidermek için sürekli bir akım döngüsüne gerek duyar. Tutarlılık nedenlerinden dolayı, RTD modülüne bir direnç bağlanmalıdır (örn. 2 telli RTD bağlantısı).

A.11.2.1 SM 1231 RTD için seçenekler tablosu

Tablo A-182 RTD modülleri tarafından desteklenen çeşitli enkoderler için aralıklar ve hassasiyet

Sıcaklık katsayısı	RTD tipi	Minimum alt aralık[1]	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[2]	Normal aralık hassasiyeti, 25 °C'de	Normal aralık hassasiyeti, -20 °C ila 60 °C
Pt 0,003850 ITS90 DIN EN 60751	PT 100 Klima	-145,00 °C	-120,00 °C	145,00 °C	155,00 °C	±0,20 °C	±0,40 °C
	Pt 10	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±1,0 °C	±2,0 °C
	Pt 50	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 100						
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0,003902 Pt 0,003916 Pt 0,003920	Pt 100	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0,003910	Pt 10	-273,2 °C	-240,0 °C	1.100,0 °C	1.295 °C	±1,0 °C	±2,0 °C
	Pt 50	-273,2 °C	-240,0 °C	1.100,0 °C	1.295 °C	±0,8 °C	±1,6 °C
	Pt 100						
	Pt 500						

[1] Alt aralık için minimumun altındaki RTD değerleri -32768 olarak yayınlanır.

[2] Üst aralık için maksimumun üzerindeki RTD değerleri +32767 olarak yayınlanır.

Sıcaklık katsayısı	RTD tipi	Minimum alt aralık[1]	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[2]	Normal aralık hassasiyeti, 25 °C'de	Normal aralık hassasiyeti, -20 °C ila 60 °C
Ni 0,006720 Ni 0,006180	Ni 100	-105,0 °C	-60,0 °C	250,0 °C	295,0 °C	±0,5 °C	±1,0 °C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						
LG-Ni 0,005000	LG-Ni 1000	-105,0 °C	-60,0 °C	250,0 °C	295,0 °C	±0,5 °C	±1,0 °C
Ni 0,006170	Ni 100	-105,0 °C	-60,0 °C	180,0 °C	212,4 °C	±0,5 °C	±1,0 °C
Cu 0,004270	Cu 10	-240,0 °C	-200,0 °C	260,0 °C	312,0 °C	±1,0 °C	±2,0 °C
Cu 0,004260	Cu 10	-60,0 °C	-50,0 °C	200,0 °C	240,0 °C	±1,0 °C	±2,0 °C
	Cu 50	-60,0 °C	-50,0 °C	200,0 °C	240,0 °C	±0,6 °C	±1,2 °C
	Cu 100						
Cu 0,004280	Cu 10	-240,0 °C	-200,0 °C	200,0 °C	240,0 °C	±1,0 °C	±2,0 °C
	Cu 50	-240,0 °C	-200,0 °C	200,0 °C	240,0 °C	±0,7 °C	±1,4 °C
	Cu 100						

[1] Alt aralık için minimumun altındaki RTD değerleri -32768 olarak yayınlanır.

[2] Üst aralık için maksimumun üzerindeki RTD değerleri +32767 olarak yayınlanır.

Tablo A-183 Direnç

Aralık	Minimum alt aralık	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[1]	Normal aralık 25 °C'de hassasiyet	Normal aralık hassasiyeti, -20 °C ila 60 °C
150 Ω	Geçerli değil	0 (0 Ω)	27648 (150 Ω)	176.383 Ω	± % 0,05	± % 0,1
300 Ω	Geçerli değil	0 (0 Ω)	27648 (300 Ω)	352.767 Ω	± % 0,05	± % 0,1
600 Ω	Geçerli değil	0 (0 Ω)	27648 (600 Ω)	705.534 Ω	± % 0,05	± % 0,1

[1] Üst aralık için minimumun üzerindeki direnç değerleri +32767 olarak yayınlanır.

NOT

Bağlı bir enkoderi olmayan etkinleştirilmiş tüm kanallar için modül 32767 değerini bildirir. Bunun dışında açık devreli hat denetimi etkinleştirilirse, modül üzerinde ilgili kırmızı LED'ler yanıp söner.

500 Ω ve 1000 Ω RTD aralıklarının diğer düşük değerli dirençlerle kullanılması durumunda, hata, belirtilen hatanın iki katına artabilir.

10 Ω RTD aralıkları için optimum seviyede hassasiyet, 4 telli bağlantılar ile mümkündür.

Bağlantı hatlarının 2 telli moddaki direnci, enkoder ölçümünde bir hataya yol açar. Dolayısıyla ölçüm hassasiyeti artık güvence altına değildir.

Tablo A-184 RTD modülleri için gürültü azaltma ve güncelleme süreleri

Bastırma frekansı seçimi	Entegrasyon süresi	Güncelleme süresi (saniye)	
		4 kanallı modül	8 kanallı modül
400 Hz (2,5 msn)	10 msn [1]	4/2 telli: 0,142 3 telli: 0,285	4/2 telli: 0,285 3 telli: 0,525
60 Hz (16,6 msn)	16,67 msn	4/2 telli: 0,222 3 telli: 0,445	4/2 telli: 0,445 3 telli: 0,845
50 Hz (20 msn)	20 msn	4/2 telli: 0,262 3 telli: 0,505	4/2 telli: 0,524 3 telli: 1,015
10 Hz (100 msn)	100 msn	4/2 telli: 1,222 3 telli: 2,445	4/2 telli: 2,425 3 telli: 4,845

[1] Modülün çözünürlüğünü ve hassasiyetini, 400 Hz filtre seçiliyken korumak için entegrasyon süresi 10 msn'dir. Bu seçim, 100 Hz ve 200 Hz'deki gürültüyü de bastırır.

NOT

Güç açıldıktan sonra modül, A/D dönüştürücü için dahili kalibrasyonu yürütür. Bu zaman diliminde modül, ilgili kanal için geçerli veriler mevcut olana dek her kanalda 32767 değerini bildirir. Bu başlatma süresinin kullanıcı programında dikkate alınması gerekebilir. Modülün konfigürasyonu başlatma işleminin süresini etkileyebileceğinden, konfigürasyonunuzda modülün davranışını kontrol edin. Gerekli olursa, modülün başlatma süresine uyum sağlamak için kullanıcı programınıza lojik ekleyebilirsiniz.
Bu lojiği, başlatma tamamlanana dek işletimi bloke edecek olan "yol alma OB'sindeki" bir okuma talebi ile uygulayabilirsiniz. Okuma talebini, anında erişim ile uygulamalısınız. Direnç sıcaklığı sensörünün okuma talebinin değeri 32767 ise, bu değer değişene dek okuma işlemi tekrarlanmalıdır. Her modül için, bu sorgulamanın yalnızca modülün en büyük numaralı girişi için yürütülmesi gerekir (modül girişleri, 0 ile 7 arasındaki sırayla başlatılır).

RTD'ler için analog değerlerin gösterimi

Aşağıdaki tablolarda RTD standart sıcaklık aralığı olan sensörler için dijitalleştirilmiş ölçüm değeri sunulmaktadır.

Tablo A-185 Direnç termometreleri için analog değerlerin gösterimi PT 100, 200, 500, 1000 ve PT 10, 50, 100, 500 GOST (0,003850) standart

°C cinsinden Pt x00 standardı (1 basamak = 0,1 °C)	Birimler		°F cinsinden Pt x00 standardı (1 basamak = 0,1 °F)	Birimler		Aralık
	Desimal	Onaltılı		Desimal	Onaltılı	
> 1000.0	32767	7FFF	> 1832.0	32767	7FFF	Taşma
1000.0 : 850.1	10000 : 8501	2710 : 2135	1832.0 : 1562.1	18320 : 15621	4790 : 3D05	Aralığın üzerinde

°C cinsinden Pt x00 standardı (1 basamak = 0,1 °C)	Birimler		°F cinsinden Pt x00 standardı (1 basamak = 0,1 °F)	Birimler		Aralık
	Desimal	Onaltılı		Desimal	Onaltılı	
850.0 : -200.0	8500 : -2000	2134 : F830	1562.0 : -328.0	15620 : -3280	3D04 : F330	Ölçüm aralığı
-200.1 : -243.0	-2001 : -2430	F82F : F682	-328.1 : -405.4	-3281 : -4054	F32F : F02A	Aralığın altında
< -243.0	-32768	8000	< -405.4	-32768	8000	Aşağı taşma

A.12 Teknoloji modülleri

A.12.1 SM 1278 4xIO-Link-Master SM

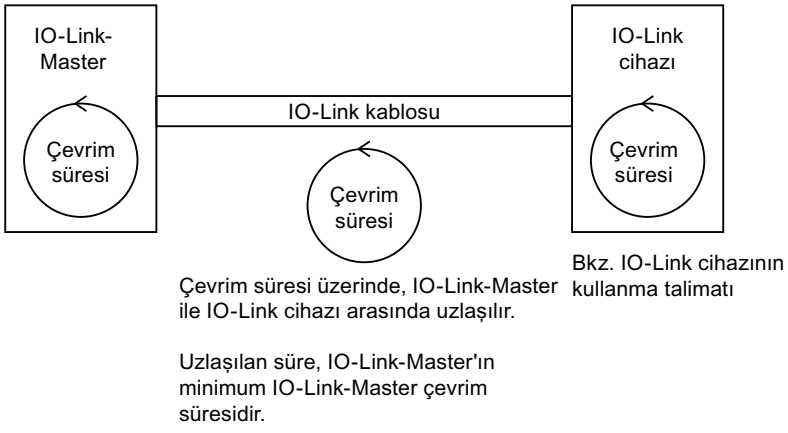
Tablo A-186 Genel teknik veriler

Teknik veriler		Sinyal modülü SM 1278 4xIO-Link-Master
Sipariş numarası		6ES7278-4BD32-0XB0
Boyutlar G x Y x D (mm)		45 x 100 x 75
Ağırlık		150 gram
Genel bilgiler		
	I&M verileri	Evet, IM0 ila IM3
Besleme gerilimi		
	Nominal gerilim (DC)	24 V DC
	Geçerli aralık alt limiti (DC)	19,2 V; IO-Link kullanırken 20,5 V (çünkü Master üzerindeki IO-Link cihazları için besleme gerilimi en azından 20 V olmalıdır)
	Üst limit geçerli aralığı (DC)	28,8 V DC
	Yanlış kutup bağlama koruması	Evet
Giriş akımı		
	Akım tüketimi	65 mA, yüksüz
Döner enkoder beslemesi		
	Çıkışların sayısı	4
	Çıkış akımı, nominal değer	Kanal başına 200 mA
Kayıp güç		
	Kayıp güç, tipik	1 W, port yükü olmadan
Dijital girişler / çıkışlar		
	Hat uzunluğu (metre)	Maks. 20 m, blendajsız
SDLC		

Teknik veriler			**Sinyal modülü SM 1278 4xIO-Link-Master**
	Hat uzunluğu (metre)		Maks. 20 m, blendajsız
IO-Link			
	Port sayısı		4
	Aynı anda kumanda edilebilen port sayısı		4
	IO-Link protokolü 1.0		Evet
	IO-Link protokolü 1.1		Evet
İşletim durumu			
	IO-Link		Evet
	DI		Evet
	DO		Evet, maks. 100 mA
IO-Link cihazlarının bağlantısı			
	Port tipi A		Evet
		Aktarım hızı	4,8 kBd (COM1)
			38,4 kBd (COM2)
			230,4 kBd (COM3)
	Min. çevrim süresi		2 msn, dinamik, kullanıcı verilerinin uzunluğuna bağlı olarak
	Proses verilerinin boyutu, port başına giriş		Maks. 32 Bayt
	Proses verilerinin boyutu, modül başına giriş		32 Bayt
	Proses verilerinin boyutu, port başına çıkış		Maks. 32 Bayt
	Proses verilerinin boyutu, modül başına çıkış		32 Bayt
	Cihaz parametresi için bellek boyutu		2 KB
	Maks. blendajsız kablo uzunluğu (metre)		20 m
Alarmlar/teşhis/durum bilgileri			
	Durum göstergesi		Evet
Alarmlar			
	Teşhis alarmı		Evet, port teşhisi yalnızca IO-Link modunda mevcuttur
Teşhis alarmları			
	Teşhis		
		Besleme gerilimi denetimi	Evet
		Kısa devre	Evet
Teşhis göstergesi LED'i			
	Besleme gerilimi denetimi		Evet, yanıp söner kırmızı DIAG-LED'i
	Kanal durum göstergesi		Evet, kanal durumu Qn (SIO modu) ve PORT durumu Cn (IO-Link modu) için kanal başına bir yeşil LED
	Kanal teşhisi için		Evet, kırmızı Fn-LED'i
	Modül teşhisi		Evet, yeşil/kırmızı DIAG-LED'i
Potansiyel ayırma			
	Kanallarda potansiyel ayırma		
		Kanallar arasında	Hayır

Teknik veriler			**Sinyal modülü SM 1278 4xIO-Link-Master**
		Kanallar ve arka panel veriyolu arasında	Evet
İzolasyon			
	İzolasyon şununla kontrol edildi		707 V DC (tip kontrolü)
Çevresel koşullar			
	Çalışma sıcaklığı		
		Min.	-20 °C
		Maks.	60 °C
		Yatay montaj, min.	-20 °C
		Yatay montaj, maks.	60 °C
		Dikey montaj, min.	-20 °C
		Dikey montaj, maks.	50 °C

Tepki süresine genel bakış

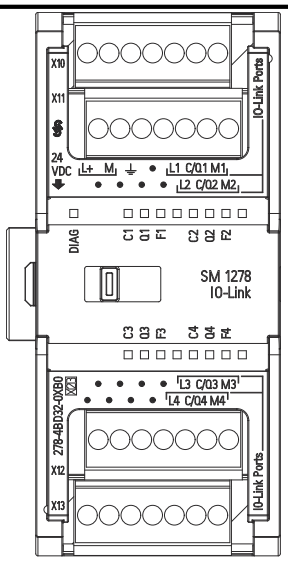


Bkz. IO-Link cihazının kullanma talimatı

Çevrim süresi üzerinde, IO-Link-Master ile IO-Link cihazı arasında uzlaşılır.

Uzlaşılan süre, IO-Link-Master'ın minimum IO-Link-Master çevrim süresidir.

Tablo A-187 SM 1278 IO-Link-Master için devre şeması

SM 1278 IO-Link-Master (6ES7278-4BD32-0XB0)	
	

Tablo A-188 SM 1278 IO-Link-Master için bağlantı yerleşimi (6ES7278-4BD32-0XB0)

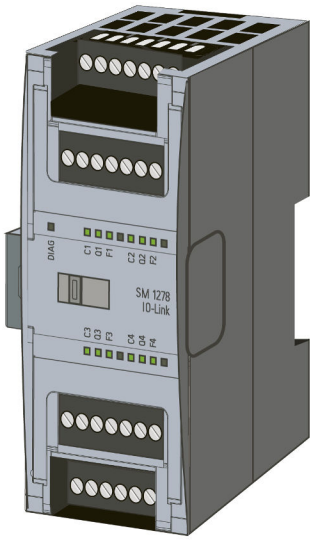
Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 V DC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Fonksiyonel toprakla-ma	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
5	L_1	L_2	L_3	L_4
6	C/O_1	C/O_2	C/O_3	C/O_4
7	M_1	M_2	M_3	M_4

A.12.1.1 SM 1278 4xIO-Link-Master - genel bakış

SM 1278 4xIO-Link-Master, bir sinyal modülü ve bir iletişim modülü olarak çalışan 4 portlu bir modüldür. IO-Link modülündeki her port, tek bir 24 V DC dijital giriş olarak veya 24 V DC dijital çıkış olarak işletilebilir.

IO-Link-Master, STEP 7 S7-1200 kumanda programınızdaki fonksiyon bloğu (FB) IO_LINK_DEVICE'ı kullanarak bir IO-Link cihazı ile aperiyodik iletişimi programlar. IO_LINK_DEVICE FB'si, programınızda hangi IO-Link-Master'ın kullanıldığını ve bu Master'ın veri alıp vermesi için hangi portları kullandığını belirtir.
Siemens Industry Online Support web sitesi (http://support.industry.siemens.com), IO-Link kütüphanesi ile çalışmaya yönelik ayrıntılı bilgiler sağlar. Web sitesi arama alanına "IO-Link" yazarak, IO-Link ürünleri ve kullanımlarına yönelik bilgilere erişebilirsiniz.

Modül görünümü



Özellikler

Teknik özellikler

- IO-Link spesifikasyonu V1.1'e göre IO-Link-Master (daha fazla bilgi için IO-Link konsorsiyumunun web sitesine (http://io-link.com/en/index.php) bakın).
- Dört arayüzlü (kanallı) seri iletişim modülü
- Veri aktarımı hızı COM1 (4.8 kbaud), COM2 (38.4 kbaud), COM3 (230.4 kbaud)
- SIO modu (standart IO modu)
- Dört adede kadar IO-Link cihazı (3 iletkenli bağlantı) veya dört standart aktüatör veya standart enkoder bağlantısı
- Her arayüz için programlanabilir teşhis fonksiyonu

Desteklenen fonksiyonlar

- I&M (kurulum ve bakım) için tanımlama verileri
- Firmware güncellemesi
- S7-PCT port konfigürasyonu aracı, STEP 7 Professional ve bir S7-1200 V4.x veya daha yüksek CPU. aracılığıyla IO-Link parametrelendirmesi; STEP 7 Professional'da, TIA Portal üzerinden IO-Link parametrelendirmesi, sınırlı bir fonksiyon kapsamı ile gerçekleştirilebilir.
- PQI (Port Qualifier Information) Bit'leri
- IO-Link kütüphane FB'leri ile yedekleme ve geri yükleme

IO-Link, bir Master ve bir cihaz arasındaki bir noktadan noktaya bağlantıdır. Cihazlar olarak bunlar, hem geleneksel hem de akıllı sensörler/aktüatörler, kalitesi kanıtlanmış 3 iletkenli teknolojide blendajsız standart kablolar üzerinden IO-Link'e bağlanabilir. IO-Link, geleneksel dijital sensörler ve aktüatörler ile geriye dönük uyumludur. Elektrik devresi ve veri kanalı, kalitesi kanıtlanmış 24 V DC teknolojisinde oluşturulmuştur.
SIMATIC-IO-Link teknolojisi ile ilgili daha fazla bilgiyi, Siemens Industry Online-Support (http://support.automation.siemens.com) web sitesindeki "IO-Link sistem kitabında" bulabilirsiniz.

NOT

IO-Link parametre verileri

SM 4xIO-Link Master modülünün değişimi sırasında, parametre verileri otomatik olarak atanmaz.

⚠ DİKKAT

Çekme ve takma

Sisteminizde yük açıkken SM 4xIO-Link Master modülün takılması, tesisinizde tehlikeli durumlara yol açabilir.

S7-1200 otomasyon sistemi bundan zarar görebilir.

SM 4xIO-Link-Master modülü yalnızca yük kapalıyken çıkarın ve takın.

Fabrika ayarlarına sıfırlamanın etkileri

S7-PCT ile yapılan parametrelendirmeyi, "Fabrika ayarlarına sıfırlama" fonksiyonu ile teslimat durumuna geri yükleyebilirsiniz.

"Fabrika ayarlarına sıfırlama" işleminden sonra SM 1278 4xIO-Link modülü aşağıdaki şekilde parametrelendirilir:

- Portlar DE modundadır
- Portlar, 0.0 ila 0.3 göreli adreslerine atanır
- PortQualifier, devre dışı bırakılmıştır
- 1 ile 3 arasındaki bakım verileri silinir

NOT

Fabrika ayarlarına sıfırlamada, cihaz parametreleri silinir ve teslimat durumu geri yüklenir.

Bir SM 1278 4xIO-Link sinyal modülünü çıkardığınızda, depoya almadan önce fabrika ayarlarına sıfırlayın.

İzlenecek prosedür

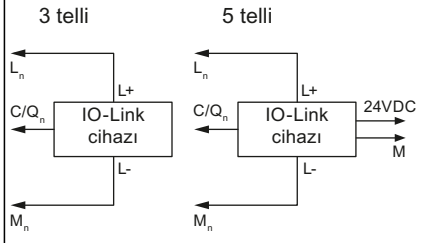
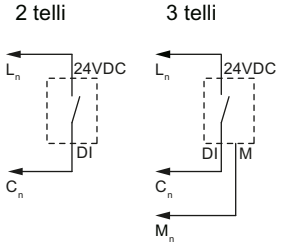
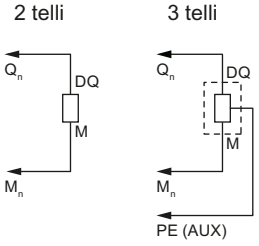
"Fabrika ayarlarına sıfırlama" işleminde, S7-PCT online yardımda "Master konfigürasyonu > 'Komutlar' sekmesi altında açıklandığı şekilde ilerleyin.

A.12.1.2 Bağlantı

Bağlantı yerleşimi ile ilgili daha fazla bilgi için, SM 1278 I/O-Link-Master (6ES 278-4BD32-0XB0) (Sayfa 1286) için bağlantı yerleşimi tablosuna bakın.
Aşağıdaki tabloda SM 1278 4xIO-Link-Master için klemens ataması gösterilmektedir:

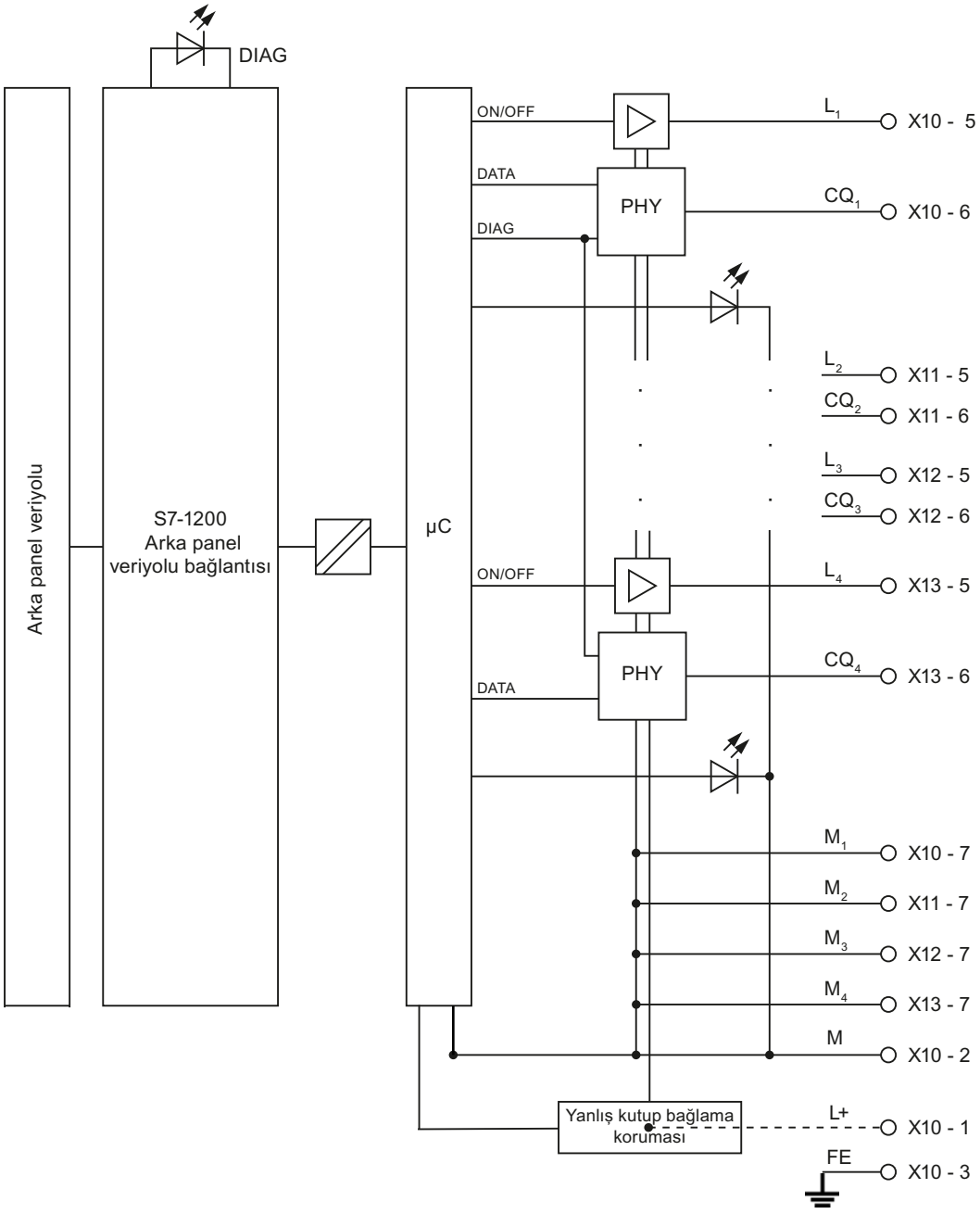
Pin	X10	X11	X12	X13	Notlar	BaseUnit'ler
7	M_1	M_2	M_3	M_4	• M_n: Şasi, Slave'e • C/O_n: SDLC, DI veya DO • L_n: Slave'den 24 V DC'ye • M: Şasi • L+: Master'dan 24 V DC'ye • RES: rezerve edilmiştir; atanamaz	A1
6	C/O_1	C/O_2	C/O_3	C/O_4		
5	L_1	L_2	L_3	L_4		
4	RES	RES	RES	RES		
3	⏚ (fonksiyonel topraklama)	RES	RES	RES		
2	M	RES	RES	RES		
1	L+	RES	RES	RES		

Aşağıdaki tabloda bağlantı örnekleri gösterilmektedir; burada n = arayüz numarasıdır.

IO-Link işletimi	DI işletim türü	DO işletim türü
		

NOT

Bağlanan sensörler, Master modülün L_n bağlantısı üzerinden sağlanan gerilim beslemesini kullanmalıdır.


DIAG
Arka panel veriyolu
S7-1200
Arka panel
veriyolu bağlantısı
µC
ON/OFF
DATA
DIAG
PHY
L1
X10 - 5
CQ1
X10 - 6
L2
X11 - 5
CQ2
X11 - 6
L3
X12 - 5
CQ3
X12 - 6
ON/OFF
DATA
PHY
L4
X13 - 5
CQ4
X13 - 6
M1
X10 - 7
M2
X11 - 7
M3
X12 - 7
M4
X13 - 7
M
X10 - 2
Yanlış kutup bağlama koruması
L+
X10 - 1
FE
X10 - 3

A.12.1.3 Parametre/adres aralığı

SM 1278 4xIO-Link-Master yapılandırması

Modül entegrasyonu, parametrelendirme ve işletime alma, STEP 7'yi gerektirir. Belirli fonksiyonlar için ayrıca S7-PCT'ye de gerek duyarsınız (Port Configuration Tool).
Daha fazla bilgiyi, SIMATIC IO-Link (https://support.industry.siemens.com/cs/ww/en/view/65949252) sistem kitabında bulabilirsiniz.

Aşağıdaki tabloda SM 1278 4xIO-Link-Master için parametreler gösterilmektedir:

Parametre	Değer aralığı	Ön ayar	RUN'daki konfigürasyon	Verimlilik alanı
Teşhis arayüzü 1	• Devre dışı bırakma • Etkinleştirme	Devre dışı bırakma	Evet	Arayüz (kanal)
Teşhis arayüzü 2	• Devre dışı bırakma • Etkinleştirme	Devre dışı bırakma	Evet	Arayüz (kanal)
Teşhis arayüzü 3	• Devre dışı bırakma • Etkinleştirme	Devre dışı bırakma	Evet	Arayüz (kanal)
Teşhis arayüzü 4	• Devre dışı bırakma • Etkinleştirme	Devre dışı bırakma	Evet	Arayüz (kanal)

1 ila 4 arayüzleri için teşhis etkinleştirme parametresi

Bu parametreyi kullanarak, dört IO-Link arayüzünün belirli arayüzleri için teşhis etkinleştirilebilir.
Arayüz atamaları, aşağıdaki gibidir:
Arayüz 1 → Kanal 1
Arayüz 2 → Kanal 2
Arayüz 3 → Kanal 3
Arayüz 4 → Kanal 4

SM 4xIO-Link Master'ın giriş ve çıkış adreslerinin maksimum boyutu her biri için 32 Bayt'tır. TIA Portal'ın donanım konfigürasyonunda S7-PCT port konfigürasyon aracını kullanarak adres aralıklarını atayabilirsiniz.

Kullanıcı programında parametrelendirme

Cihazı, çalışma zamanı içinde yapılandırılabilirsiniz.

Çalışma zamanı içinde parametreleri değiştirme

Modül parametreleri, 128 numaralı veri setinde bulunur. Düzenlenebilir parametreleri, WRREC yordamı ile modüle aktarabilirsiniz.
CPU'yu sıfırladıktan sonra (kapatıp açma), CPU, WRREC yordamı ile parametrelendirme sırasında modüle aktarılan parametrelerin üzerine yazar.

Parametrelendirme için yordam

Kullanıcı programındaki G/Ç modülünü parametrelendirmek için aşağıdaki yordamı kullanabilirsiniz:

Yordam	Uygulama
SFB 53 WRREC	Değiştirilebilir parametrelerin modüle aktarılması.

Hata mesajı

Bir hata oluşursa, aşağıdaki geri dönüş değeri bildirilir:

Hata kodu	Anlamı
$80B1_H$	Hatalı veri uzunluğu
$80E0_H$	Hatalı Header bilgisi
$80E1_H$	Parametre hatası

Veri seti yapısı

Aşağıdaki tabloda, IO-Link parametreleri gösterilmektedir:

Ofset	Adı	Tip	Ön ayar	Açıklama
0	Sürüm	1 Bayt	0x02	IO-Link V1.1'e göre IO-Link Master için 0x02 veri setinin yapısını gösterir
1	Parametre uzunluğu	1 Bayt	0x02	Parametre uzunluğu (2 Bayt + 2 Header)
IO-Link başlangıç parametresi				
2	Arayüz teşhisi (arayüz1 1 ila n)	1 Bayt	0x00	1'den n'ye kadar arayüz için teşhisinin etkinleştirilmesi
3	IOL özellikleri	1 Bayt	0x00	Modül özellikleri

Aşağıdaki tabloda sürüm veri seti gösterilmektedir:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rezerve		Ana sürüm (00)		Alt sürüm (0010)			

Aşağıdaki tabloda, arayüz teşhisinin veri seti gösterilmektedir:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rezerve				EN_Port4	EN_Port3	EN_Port2	EN_Port1

EN_Portx:

0 = Teşhis devre dışı bırakıldı

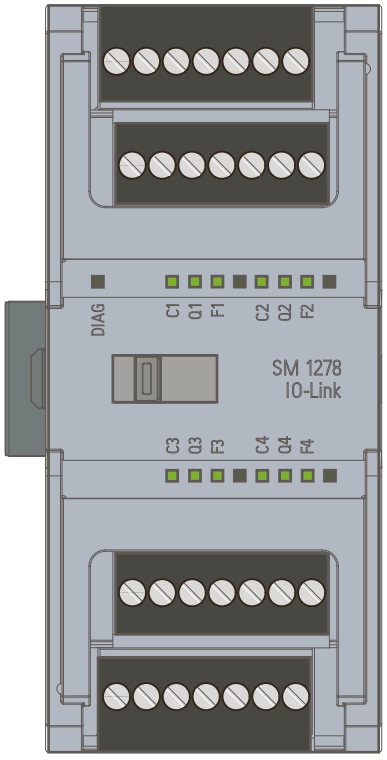
1 = Teşhis etkinleştirildi

Aşağıdaki tabloda IOL özelliklerine sahip veri seti gösterilmektedir:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rezerve							

A.12.1.4 Alarm, hata ve sistem bildirimleri

LED göstergesi



LED göstergelerinin anlamı

Aşağıdaki tabloda durum ve hata göstergelerinin anlamları verilmiştir. Teşhis alarmları için düzeltici önlemler "Teşhis mesajları" bölümünde bulabilirsiniz.

LED DIAG

DIAG	Anlamı
▫ Kapalı	S7-1200'ün arka panel veriyolu beslemesi TAMAM değil
☼ Yanıp sönen	Modül yapılandırılmamış
■ Açık	Modül parametrelenmiş ve modül teşhisi yok
☼ Yanıp sönen	Modül parametrelenmiş ve modül teşhisi VEYA L+ bağlı değil

Port durumu LED'i

IO-Link port modundaki bir IO-Link portu için geçerlidir.

COM/1 ... COM/4	Anlamı
▫ Kapalı	Port, devre dışı bırakıldı
☼ Yanıp sönen	Port, etkinleştirildi, cihaz bağlı değil veya port, yapılandırılmış cihaza bağlı değil
■ Açık	Port, etkinleştirildi, cihaz bağlı

Kanal durumu LED'i

DI/O modundaki bir IO-Link portu için geçerlidir.

DI/O1 ... DI/O4	Anlamı
▫ Kapalı	Proses sinyali = 0
■ Açık	Proses sinyali = 1

Port hatası LED'i

F1...F4	Anlamı
▫ Kapalı	Hata yok
■ Açık	Hata

Modül hataları, yalnızca IO-Link modunda teşhis (modül durumu) olarak görüntülenir.

Teşhis alarmı	Hata kodu (desimal)	STATUS (W#16#...)	Anlamı (IO-Link hata kodu)	IO-Link Master	IO-Link cihazı
Kısa devre	1	1804	IO-Link cihazındaki proses kablolarında kısa devre	X	
		7710	IO cihazında kısa devre		X
Düşük gerilim	2	5111 5112	Besleme gerilimi çok düşük		X
Aşırı gerilim	3	5110	Besleme gerilimi çok yüksek		X
Aşırı ısınma	5	1805	Master'da sıcaklık aşıldı	X	
		4000 4210	Cihazda sıcaklık aşıldı		X
Tel kopması	6	1800	• Hiçbir IO-Link cihazı bağlı değil • IO-Link cihazına giden sinyal hattında tel kopması • IO-Link cihazı, diğer bir hata nedeniyle iletişim kuramıyor	X	
Taşma	7	8C10 8C20	Proses değişkeni aralığı aşıldı		X
		8C20	Ölçüm aralığı aşıldı		
Aşağı taşma	8	8C30	Proses değişkeni aralığı çok küçük		X
Hata	9	---	Burada listelenmeyen tüm IO-Link hata kodları, bu PROFIBUS DP hatasına atanır.		X
Parametrelendirme hatası	16	1882 1883	IO-Link-Master yapılandırılamadı	X	
		1802	Yanlış cihaz		
		1886	Bellek hatası		
		6320 6321 6350	Cihaz yapılandırılamadı		X
Besleme gerilimi eksik	17	1806	Cihaz için L+ besleme gerilimi eksik	X	
		1807	Cihaz için L+ besleme gerilimi çok düşük (< 20 V)		
Arızalı sigorta	18	5101	Cihazdaki sigorta arızalı		X
Güvenlik kapatması	25	1880	Ciddi hata (Master değiştirilmeli)	X	
Harici arıza	26	1809 180A 180B 180C 180D	Veri belleğinde hata	X	
		1808	IO-Link cihazında aynı anda 6'dan fazla hata beklemede		

A.13 Dijital sinyal kartları (SB'ler)

A.13.1 SB 1221 200 kHz dijital girişin teknik verileri

Tablo A-189 Genel teknik veriler

Teknik veriler	**SB 1221 DI 4 x 24 V DC, 200 kHz**	**SB 1221 DI 4 x 5 V DC, 200 kHz**
Sipariş numarası	6ES7221-3BD30-0XB0	6ES7221-3AD30-0XB0
Boyutlar (G x Y x D) (mm)	38 x 62 x 21	
Ağırlık	35 gram	
Performans kaybı	1,5 W	1,0 W
Akım tüketimi (SM veriyolu)	40 mA	
Akım tüketimi (24 V DC)	7 mA / giriş + 20 mA	15 mA / giriş + 15 mA

Tablo A-190 Dijital girişler

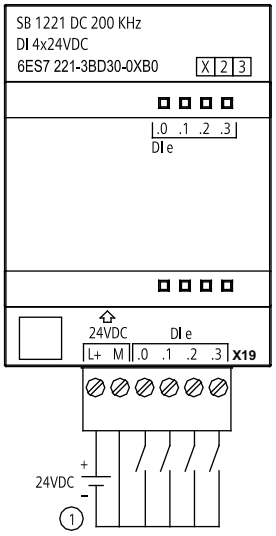
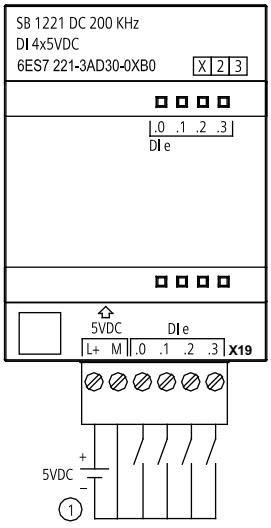
Teknik veriler		**SB 1221 DI 4 x 24 V DC, 200 kHz**	**SB 1221 DI 4 x 5 V DC, 200 kHz**
Girişlerin sayısı		4	
Tip		Akım besleyen	
Nominal gerilim		7 mA'da 24 V DC, nominal	15 mA'da 5 V DC, nominal
İzin verilen sürekli gerilim		28,8 V DC	6 V DC
Şok gerilim		0,5 sn için 35 V DC	6 V
Mantıksal sinyal 1		0 V (10 mA) L+ eksi 10 V (2,9 mA)	0 V (20 mA) L+ eksi 2,0 V (5,1 mA)
Mantıksal sinyal 0		L+ eksi 5 V (1,4 mA) L+ (0 mA)	L+ eksi 1,0 V (2,2 mA) L + (0 mA)
HSC giriş adım frekansları (maks.)		Tek faz sayacı: 200 kHz A/B sayacı: 160 kHz	
Elektriksel ayırma (alandan lojiğe)		707 V DC (tip kontrolü)	
Potansiyeli ayrılmış gruplar		1	
Filtre süreleri	µsn ayarları:	0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0	
	msn ayarları:	0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0	
Eş zamanlı devreye alınmış girişlerin sayısı		• 60 °C yatayda veya 50 °C dikeyde 2 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 4	4
Hat uzunluğu (metre)		50 m, blendajlı, bükülmüş iletken çifti	

NOT

20 kHz'in üzerinde frekansları değiştirirken, dijital girişlerin kare dalga sinyalleri alması önemlidir. Giriş sinyalinin kalitesini iyileştirmek için aşağıdaki olanaklara sahipsiniz:

- Hattı, minimum uzunluğa kısaltın.
- Yalnızca akım çeken bir sürücü yerine akım çeken/akım besleyen (P-M anahtarlaması) bir sürücü kullanın.
- Hat kablosunu daha kaliteli bir kablo ile değiştirin.
- Anahtarlama devrelerinin/bileşenlerin gerilimini 24 V'tan 5 V'a düşürün.
- Girişe, harici bir yük ekleyin.

Tablo A-191 Dijital giriş SB'lerinin devre şemaları (200 kHz)

SB 1221 DI 4 x 24 V DC, 200 kHz (6ES7221-3BD30-0XB0)	**SB 1221 DI 4 x 5 V DC, 200 kHz (6ES7221-3AD30-0XB0)**
	

① Yalnızca akım besleyen girişleri destekler

Tablo A-192 SB 1221 DI 4 x 24 V DC, 200 kHz için bağlantı yerleşimi (6ES7221-3BD30-0XB0)

Pin	X19
1	L+ / 24 V DC
2	M / 24 V DC
3	DI e.0
4	DI e.1
5	DI e.2
6	DI e.3

Tablo A-193 SB 1221 DI 4 x 5 V DC, 200 kHz için bağlantı yerleşimi (6ES7221-3AD30-0XB0)

Pin	X19
1	L+ / 5 V DC
2	M / 5 V DC
3	DI e.0
4	DI e.1
5	DI e.2
6	DI e.3

A.13.2 SB 1222 200 kHz dijital çıkışın teknik verileri

Tablo A-194 Genel teknik veriler

Teknik veriler	SB 1222 DQ 4 x 24 V DC, 200 kHz	SB 1222 DQ 4 x 5 V DC, 200 kHz
Sipariş numarası	6ES7222-1BD30-0XB0	6ES7222-1AD30-0XB0
Boyutlar (G x Y x D) (mm)	38 x 62 x 21	
Ağırlık	35 gram	
Performans kaybı	0,5 W	
Akım tüketimi (SM veriyolu)	35 mA	
Akım tüketimi (24 V DC)	15 mA	

Tablo A-195 Dijital çıkışlar

Teknik veriler	SB 1222 DQ 4 x 24 V DC, 200 kHz	SB 1222 DQ 4 x 5 V DC, 200 kHz
Çıkışlar	4	
Çıkış tipi	MOSFET, elektronik (akım çeken/akım besleyen)[1]	
Gerilim aralığı	20,4 ila 28,8 V DC	4,25 ila 6,0 V DC
Maks. akımdaki mantıksal sinyal 1	L+ eksi 1,5 V	L+ eksi 0,7 V
Maks. akımdaki mantıksal sinyal 0	maks. 1,0 V DC	maks. 0,2 V DC
Akım (maks.)	0,1 A	
Lamba yükü	--	
AÇIK'taki kontak direnci	maks. 11 Ω	maks. 7 Ω
KAPALI'da direnç	maks. 6 Ω	maks. 0,2 Ω
Çıkış başına kaçak akım	--	
Darbe jeneratörü frekansı	maks. 200 kHz, min. 2 Hz	
Ani akım	0,11 A	
Aşırı yük koruması	Hayır	
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)	
Potansiyeli ayrılmış gruplar	1	
İletken başına akım	0,4 A	
Endüktif klemens gerilimi	Hayır	
Anahtarlama gecikmesi	1,5 µsn + 300 nsn yükselen 1,5 µsn + 300 nsn alçalan	200 ns + 300 nsn yükselen 200 nsn + 300 nsn alçalan
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	
Bir dijital girişin kumandası	Evet	
Yedekli yük kumandası için paralel çıkışlar	Hayır	

[1] Hem akım besleyen hem de akım çeken konfigürasyonların aynı devreler tarafından desteklenmesi nedeniyle, akım besleyen bir yükün aktif durumu, akım çeken bir yükün durumunun tersidir. Akım besleyen bir çıkış pozitif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, AÇIK), akım çeken bir çıkış ise negatif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, KAPALI). Bir kullanıcı programı olmadan modülün takılması durumunda, bu modül için standart değer 0 V'tur; bunun anlamı, akım çeken bir yükün açık olduğudur.

Teknik veriler	SB 1222 DQ 4 x 24 V DC, 200 kHz	SB 1222 DQ 4 x 5 V DC, 200 kHz
Artırılmış yük için paralel çıkışlar	Hayır	
Eş zamanlı devreye alınmış çıkışların sayısı	• 60 °C yatayda veya 50 °C dikeyde 2 (bitişik nokta yok) • 55 °C yatayda veya 45 °C dikeyde 4	4
Hat uzunluğu (metre)	50 m, blendajlı, bükülmüş iletken çifti	

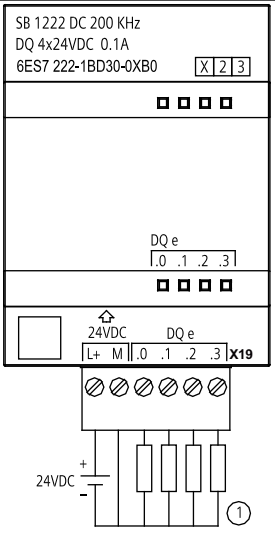
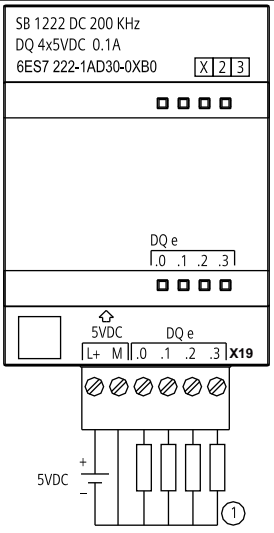
[1] Hem akım besleyen hem de akım çeken konfigürasyonların aynı devreler tarafından desteklenmesi nedeniyle, akım besleyen bir yükün aktif durumu, akım çeken bir yükün durumunun tersidir. Akım besleyen bir çıkış pozitif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, AÇIK), akım çeken bir çıkış ise negatif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, KAPALI). Bir kullanıcı programı olmadan modülün takılması durumunda, bu modül için standart değer 0 V'tur; bunun anlamı, akım çeken bir yükün açık olduğudur.

NOT

20 kHz'in üzerinde frekansları değiştirirken, dijital girişlerin kare dalga sinyalleri alması önemlidir. Giriş sinyalinin kalitesini iyileştirmek için aşağıdaki olanaklara sahipsiniz:

- Hattı, minimum uzunluğa kısaltın.
- Yalnızca akım çeken bir sürücü yerine akım çeken/akım besleyen (P-M anahtarlaması) bir sürücü kullanın.
- Hat kablosunu daha kaliteli bir kablo ile değiştirin.
- Anahtarlama devrelerinin/bileşenlerin gerilimini 24 V'tan 5 V'a düşürün.
- Girişe, harici bir yük ekleyin.

Tablo A-196 Dijital çıkış SB'lerinin devre şemaları (200 kHz)

SB 1222 DQ 4 x 24 V DC, 200 kHz (6ES7222-1BD30-0XB0)	**SB 1222 DQ 4 x 5 V DC, 200 kHz (6ES7222-1AD30-0XB0)**
	

① Akım besleyen çıkışlar için "Load"u, "-"ye bağlayın (resme bakın). Akım çeken çıkışlarda, "Load"u, "+"ya bağlayın. Hem akım besleyen hem de akım çeken konfigürasyonların aynı devreler tarafından desteklenmesi nedeniyle, akım besleyen bir yükün aktif durumu, akım çeken bir yükün durumunun tersidir. Akım besleyen bir çıkış pozitif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, AÇIK), akım çeken bir çıkış ise negatif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, KAPALI). Bir kullanıcı programı olmadan modülün takılması durumunda, bu modül için standart değer 0 V'tur; bunun anlamı, akım çeken bir yükün açık olduğudur.

NOT

Şasi bağlantı telinin güvenli bir şekilde topraklandığından emin olun. Yüksek hızlı DO SB'lerinin toprak bağlantısının kaybı, muhtemelen bir DC yükünü etkinleştirmek için yeterli kaçak akıma yol açabilir. Çıkışlar önemli DC yük uygulamaları için kullanılıyorsa, ek bir güvenlik önlemi olarak SB'ye yedekli bir topraklama hattı bağlanmalıdır.

Tablo A-197 SB 1222 DQ 4 x 24 V DC, 200 kHz için bağlantı yerleşimi (6ES7222-1BD30-0XB0)

Pin	X19
1	L+ / 24 V DC
2	M / 24 V DC
3	DO e.0
4	DO e.1
5	DO e.2
6	DO e.3

Tablo A-198 SB 1222 DQ 4 x 5 V DC, 200 kHz için bağlantı yerleşimi (6ES7222-1AD30-0XB0)

Pin	X19
1	L+ / 5 V DC
2	M / 5 V DC

Pin	X19
3	DO e.0
4	DO e.1
5	DO e.2
6	DO e.3

A.13.3 SB 1223 200 kHz dijital giriş/çıkışın teknik verileri

Tablo A-199 Genel teknik veriler

Teknik veriler	SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz	SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz
Sipariş numarası	6ES7223-3BD30-0XB0	6ES7223-3AD30-0XB0
Boyutlar (G x Y x D) (mm)	38 x 62 x 21	
Ağırlık	35 gram	
Performans kaybı	1,0 W	0,5 W
Akım tüketimi (SM veriyolu)	35 mA	
Akım tüketimi (24 V DC)	7 mA / giriş + 30 mA	15 mA / giriş + 15 mA

Tablo A-200 Dijital girişler

Teknik veriler		SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz	SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz
Girişlerin sayısı		2	
Tip		Akım besleyen	
Nominal gerilim		7 mA'da 24 V DC, nominal	15 mA'da 5 V DC, nominal
İzin verilen sürekli gerilim		28,8 V DC	6 V DC
Şok gerilim		0,5 sn için 35 V DC	6 V
Mantıksal sinyal 1		0 V (10 mA) L+ eksi 10 V (2,9 mA)	0 V (20 mA) L+ eksi 2,0 V (5,1 mA)
Mantıksal sinyal 0		L+ eksi 5 V (1,4 mA) L+ (0 mA)	L+ eksi 1,0 V (2,2 mA) L + (0 mA)
HSC giriş adım frekansları (maks.)		Tek faz sayacı: 200 kHz A/B sayacı: 160 kHz	
Elektriksel ayırma (alandan lojiğe)		707 V DC (tip kontrolü)	
Potansiyeli ayrılmış gruplar		1 (çıkış yönünde ayırma yok)	
Filtre süreleri	µsn ayarları:	0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0	
	msn ayarları:	0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0	
Eş zamanlı devreye alınmış girişlerin sayısı		2	
Hat uzunluğu (metre)		50 m, blendajlı, bükülmüş iletken çifti	

Tablo A-201 Dijital çıkışlar

Teknik veriler	**SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz**	**SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz**
Çıkışlar	2	
Çıkış tipi	MOSFET, elektronik (akım çeken/akım besleyen)[1]	
Gerilim aralığı	20,4 ila 28,8 V DC	4,25 ila 6,0 V DC
Nominal değer	24 V DC	5 V DC
Maks. akımdaki mantıksal sinyal 1	L+ eksi 1,5 V	L+ eksi 0,7 V
Maks. akımdaki mantıksal sinyal 0	maks. 1,0 V DC	maks. 0,2 V DC
Akım (maks.)	0,1 A	
Lamba yükü	--	
AÇIK'taki kontak direnci	maks. 11 Ω	maks. 7 Ω
KAPALI'da direnç	maks. 6 Ω	maks. 0,2 Ω
Çıkış başına kaçak akım	--	
Darbe jeneratörü frekansı	maks. 200 kHz, min. 2 Hz	
Ani akım	0,11 A	
Aşırı yük koruması	Hayır	
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)	
Potansiyeli ayrılmış gruplar	1 (giriş yönünde ayırma yok)	
İletken başına akım	0,2 A	
Endüktif klemens gerilimi	Hayır	
Anahtarlama gecikmesi	1,5 µsn + 300 nsn yükselen 1,5 µsn + 300 nsn alçalan	200 ns + 300 nsn yükselen 200 nsn + 300 nsn alçalan
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)	
Bir dijital girişin kumandası	Evet	
Yedekli yük kumandası için paralel çıkışlar	Hayır	
Artırılmış yük için paralel çıkışlar	Hayır	
Eş zamanlı devreye alınmış çıkışların sayısı	2	
Hat uzunluğu (metre)	50 m, blendajlı, bükülmüş iletken çifti	

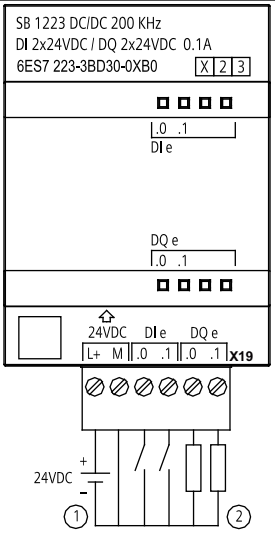
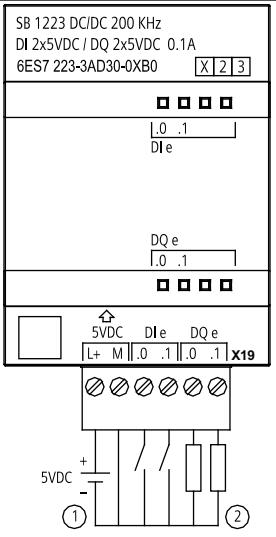
[1] Hem akım besleyen hem de akım çeken konfigürasyonların aynı devreler tarafından desteklenmesi nedeniyle, akım besleyen bir yükün aktif durumu, akım çeken bir yükün durumunun tersidir. Akım besleyen bir çıkış pozitif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, AÇIK), akım çeken bir çıkış ise negatif lojik gösterir (yükte akım akışı olduğunda Q Bit'i ve LED, KAPALI). Bir kullanıcı programı olmadan modülün takılması durumunda, bu modül için standart değer 0 V'tur; bunun anlamı, akım çeken bir yükün açık olduğudur.

NOT

20 kHz'in üzerinde frekansları değiştirirken, dijital girişlerin kare dalga sinyalleri alması önemlidir. Giriş sinyalinin kalitesini iyileştirmek için aşağıdaki olanaklara sahipsiniz:

- Hattı, minimum uzunluğa kısaltın.
- Yalnızca akım çeken bir sürücü yerine akım çeken/akım besleyen (P-M anahtarlaması) bir sürücü kullanın.
- Hat kablosunu daha kaliteli bir kablo ile değiştirin.
- Anahtarlama devrelerinin/bileşenlerin gerilimini 24 V'tan 5 V'a düşürün.
- Girişe, harici bir yük ekleyin.

Tablo A-202 Dijital giriş/çıkış SB'lerinin devre şemaları (200 kHz)

SB 1223 DI 2 x 24 V DC/DQ 2 x 24 V DC, 200 kHz (6ES7223-3BD30-0XB0)	**SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz (6ES7223-3AD30-0XB0)**
	

(1) Yalnızca akım besleyen girişleri destekler

1 Akım besleyen çıkışlar için "Load"u, "-"ye bağlayın (resme bakın). Akım çeken çıkışlarda, "Load"u, "+"ya bağlayın.

NOT

Şasi bağlantı telinin güvenli bir şekilde topraklandığından emin olun. Yüksek hızlı DO SB'lerinin toprak bağlantısının kaybı, muhtemelen bir DC yükünü etkinleştirmek için yeterli kaçak akıma yol açabilir. Çıkışlar önemli DC yük uygulamaları için kullanılıyorsa, ek bir güvenlik önlemi olarak SB'ye yedekli bir topraklama hattı bağlanmalıdır.

Tablo A-203 SB 1223 DI 2 x 24 V DC/DQ 2 x 24 V DC, 200 kHz için bağlantı yerleşimi (6ES7223-3BD30-0XB0)

Pin	X19
1	L+ / 24 V DC
2	M / 24 V DC

Pin	X19
3	DI e.0
4	DI e.1
5	DO e.0
6	DO e.1

Tablo A-204 SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz için bağlantı yerleşimi (6ES7223-3AD30-0XB0)

Pin	X19
1	L+ / 5 V DC
2	M / 5 V DC
3	DI e.0
4	DI e.1
5	DO e.0
6	DO e.1

A.13.4 2 x 24 V DC girişli / 2 x 24 V DC çıkışlı SB 1223 için teknik veriler

Tablo A-205 Genel teknik veriler

Teknik veriler	SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC
Sipariş numarası	6ES7223-0BD30-0XB0
Boyutlar G x Y x D (mm)	38 x 62 x 21
Ağırlık	40 gram
Performans kaybı	1,0 W
Akım tüketimi (SM veriyolu)	50 mA
Akım tüketimi (24 V DC)	4 mA / giriş

Tablo A-206 Dijital girişler

Teknik veriler		SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC
Girişlerin sayısı		2
Tip		IEC Tip 1 akım çeken
Nominal gerilim		4 mA'da 24 V DC, nominal
İzin verilen sürekli gerilim		maks. 30 V DC
Şok gerilim		0,5 sn için 35 V DC
Mantıksal sinyal 1 (min.)		2,5 mA'da 15 V DC
Mantıksal sinyal 0 (maks.)		1 mA'da 5 V DC
HSC giriş adım frekansları (maks.)		Tek faz sayacı: 30 kHz (15 ila 26 V DC) A/B sayacı: 20 kHz (15 ila 26 V DC)
Elektriksel ayırma (alandan lojiğe)		707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar		1
Filtre süreleri	µsn ayarları:	0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0

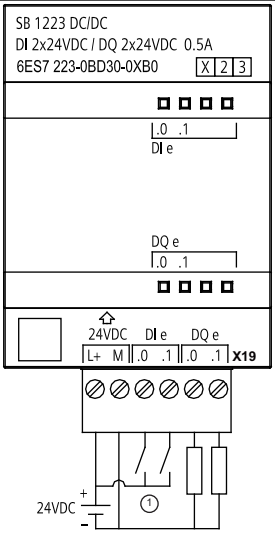
Teknik veriler		**SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC**
	msn ayarları:	0,05 / 0,1 / 0,2 / 0,4 / 0,8 / 1,6 / 3,2 / 6,4 / 10,0 / 12,8 / 20,0
Eş zamanlı devreye alınmış girişlerin sayısı		2
Hat uzunluğu (metre)		500 blendajlı, 300 blendajsız

Tablo A-207 Dijital çıkışlar

Teknik veriler	**SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC**
Çıkışlar	2
Çıkış tipi	MOSFET, elektronik (akım besleyen)
Gerilim aralığı	20,4 ila 28,8 V DC
Maks. akımdaki mantıksal sinyal 1	min. 20 V DC
10 kΩ yükteki mantıksal sinyal 0	maks. 0,1 V DC
Akım (maks.)	0,5 A
Lamba yükü	5 W
AÇIK'taki kontak direnci	maks. 0,6 Ω
Çıkış başına kaçak akım	maks. 10 µA
Darbe dizisi çıkış frekansı (PTO)	maks. 20 kHz, min. 2 Hz [1]
Ani akım	Maks. 100 msn için 5 A
Aşırı yük koruması	Hayır
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Potansiyeli ayrılmış gruplar	1
İletken başına akım	1 A
Endüktif klemens gerilimi	L+ eksi 48 V, 1 W kayıp güç
Anahtarlama gecikmesi	Kapalı'dan Açık'a maks. 2 µsn maks. Açık'tan Kapalı'ya 10 µsn
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)
Bir dijital girişin kumandası	Evet
Yedekli yük kumandası için paralel çıkışlar	Hayır
Artırılmış yük için paralel çıkışlar	Hayır
Eş zamanlı devreye alınmış çıkışların sayısı	2
Hat uzunluğu (metre)	500 m blendajlı, 150 m blendajsız

[1] Darbe alıcısına ve kabloya bağlı olarak, ek bir yük direnci (nominal akımın en azından %10'u) darbe sinyallerinin kalitesini ve parazite dayanıklılığını artırabilir.

Tablo A-208 Dijital giriş/çıkış SM'sinin devre şeması

SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC (6ES7223-0BD30-0XB0)	
	

① Yalnızca akım çeken girişleri destekler

Tablo A-209 2 x 24 V DC, DQ 2 x 24 V DC'li SB 1223 DI için bağlantı yerleşimi (6ES7223-0BD30-0XB0)

Pin	X19
1	L+ / 24 V DC
2	M / 24 V DC
3	DI e.0
4	DI e.1
5	DO e.0
6	DO e.1

A.14 Analog sinyal kartları (SB'ler)

A.14.1 SB 1231 1 analog girişin teknik verileri

NOT

Bu SB'yi kullanabilmek için V2.0 veya üstü Firmware'e sahip bir CPU'ya ihtiyacınız vardır.

Tablo A-210 Genel teknik veriler

Teknik veriler	**SB 1231 AI 1 x 12 Bit**
Sipariş numarası	6ES7231-4HA30-0XB0
Boyutlar G x Y x D (mm)	38 x 62 x 21
Ağırlık	35 gram
Performans kaybı	0,4 W
Akım tüketimi (SM veriyolu)	55 mA
Akım tüketimi (24 V DC)	Yok

Tablo A-211 Analog girişler

Teknik veriler	**SB 1231 AI 1x12 Bit**
Girişlerin sayısı	1
Tip	Gerilim veya akım (diferansiyel)
Aralık	±10 V, ±5 V, ±2,5 veya 0 ila 20 mA
Çözünürlük	11 Bit + ön işaret Bit'i
Tam sapma (veri sözcüğü)	-27648 ila 27648
Aralık üstü/aralık altı (veri sözcüğü)	Gerilim: 32511 ila 27649 / -27649 ila -32512 Akım: 32511 ila 27649 / 0 ila -4864 (Bkz. Gerilim için analog giriş gösterimi ve akım için analog giriş gösterimi (Sayfa 1316)).
Taşma/aşağı taşma (veri sözcüğü)	Gerilim: 32767 ila 32512 / -32513 ila -32768 Akım: 32767 ila 32512 / -4865 ila -32768 (Bkz. Gerilim için analog giriş gösterimi ve akım için analog giriş gösterimi (Sayfa 1316)).
Maksimum dayanım gerilimi/akımı	±35 V / ±40 mA
Düzleştirme	Yok, zayıf, orta veya güçlü (Bkz. Adım yanıt süresi için analog giriş yanıt süreleri (Sayfa 1315-1316)).
Gürültü azaltma	400, 60, 50 veya 10 Hz (Bkz. Tarama hızları için analog giriş tepki verme süresi (Sayfa 1316)).
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın ±%0,3 / ± %0,6'sı
Giriş empedansı	Gerilim: 150 kΩ; akım: 250 Ω
Ölçme prensibi	Gerçek değer dönüştürme
Ortak mod bastırma	40 dB, 60 Hz'teki nominal değer

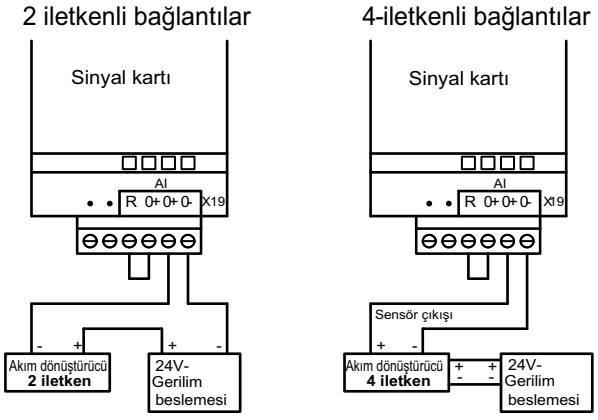
Teknik veriler	SB 1231 AI 1x12 Bit
İşletim sinyal aralığı	Sinyal artı ortak mod gerilimi, +35 V'tan küçük ve -35 V'tan büyük olmalıdır
Elektriksel ayırma (alandan lojiğe)	Yok
Hat uzunluğu (metre)	100 m, bükülmüş ve blendajlı

Tablo A-212 Teşhis

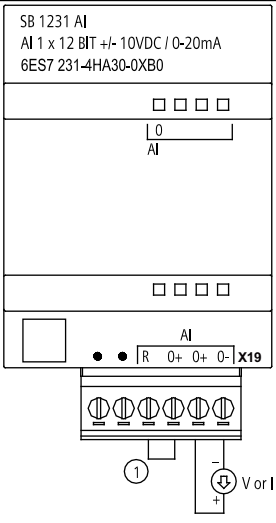
Teknik veriler	SB 1231 AI 1 x 12 Bit
Taşma/aşağı taşma	Evet
24 Volt DC alçak gerilim	Hayır

SB 1231 için ölçüm dönüştürücü

Ölçüm dönüştürücüler, aşağıda gösterildiği gibi 2 telli ve 4 telli ölçüm dönüştürücü olarak sunulmaktadır.



Tablo A-213 Analog giriş SB'sinin devre şeması

SB 1231 AI x 12 Bit (6ES7231-4HA30-0XB0)	
	① Elektrik akımı için "R" ve "0+" bağlayın. Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.

Tablo A-214 SB 1231 AI x 12 Bit için bağlantı yerleşimi (6ES7231-4HA30-0XB0)

Pin	X19 (altın kaplamalı)
1	Bağlantı yok
2	Bağlantı yok
3	AI R
4	AI 0+
5	AI 0+
6	AI 0-

A.14.2 SB 1232 1 analog çıkışın teknik verileri

Tablo A-215 Genel teknik veriler

Teknik veriler	SB 1232 AO 1 x 12 Bit
Sipariş numarası	6ES7232-4HA30-0XB0
Boyutlar (G x Y x D) (mm)	38 x 62 x 21
Ağırlık	40 gram
Performans kaybı	1,5 W
Akım tüketimi (SM veriyolu)	15 mA
Akım tüketimi (24 V DC)	40 mA (yük olmadan)

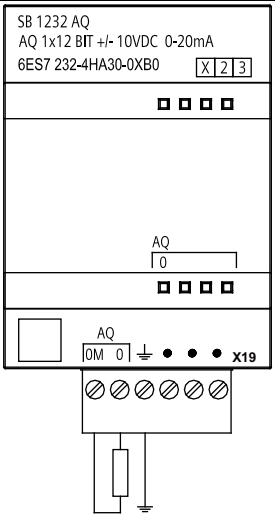
Tablo A-216 Analog çıkışlar

Teknik veriler	SB 1232 AO 1 x 12 Bit
Çıkışlar	1
Tip	Gerilim veya akım
Aralık	±10 V veya 0 ila 20 mA
Çözünürlük	Gerilim: 12 Bit Akım: 11 Bit
Tam sapma (veri sözcüğü) Gerilim ve akım (Sayfa 1317) için çıkış aralıkları alt bölümüne bakın.	Gerilim: -27.648 ila 27.648 Akım: 0 ila 27648
Hassasiyet (25 °C / -20 ila 60 °C)	Tam sapmanın ±%0,5 / ±%1'i
Yatışma süresi (değerin % 95'i)	Gerilim: 300 µsn (R), 750 µsn (1 uF) Akım: 600 µsn (1 mH), 2 msn (10 mH)
Yük empedansı	Gerilim: ≥ 1000 Ω Akım: ≤ 600 Ω
RUN'dan STOP'a geçişteki davranış	Son değer veya yedek değer (ön ayar 0)
Elektriksel ayırma (alandan lojiğe)	Yok
Hat uzunluğu (metre)	100 m, bükülmüş ve blendajlı

Tablo A-217 Teşhis

Teknik veriler	**SB 1232 AO 1 x 12 Bit**
Taşma/aşağı taşma	Evet
Toprakla kısa devre (yalnızca gerilim modu)	Evet
Tel kopması (yalnızca akım modunda)	Evet

Tablo A-218 SB 1232 AO 1 x 12 Bit için devre şeması

SB 1232 AQ 1 x 12 Bit (6ES7232-4HA30-0XB0)	
	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

Tablo A-219 SB 1232 AQ 1 x 12 Bit için bağlantı yerleşimi (6ES7232-4HA30-0XB0)

Pin	**X19 (altın kaplamalı)**
1	AO 0M
2	AO 0
3	Fonksiyonel topraklama
4	Bağlantı yok
5	Bağlantı yok
6	Bağlantı yok

A.14.3 Analog girişler ve çıkışların ölçüm aralıkları

A.14.3.1 Analog girişlerin adım yanıtı

Tablo A-220 Adım yanıtı (msn), 0 V ila 10 V %95'te ölçülmüştür

Düzleştirme seçimi (tarama değerlerinin ortalamasını alma)	**Entegrasyon süresi seçimi**			
	400 Hz (2,5 msn)	**60 Hz (16,6 msn)**	**50 Hz (20 msn)**	**10 Hz (100 msn)**
Yok (1 çevrim): Ortalama alma yok	4,5 msn	18,7 msn	22,0 msn	102 msn
Zayıf (4 çevrim): 4 tarama değeri	10,6 msn	59,3 msn	70,8 msn	346 msn
Orta (16 çevrim): 16 tarama değeri	33,0 msn	208 msn	250 msn	1240 msn
Güçlü (32 çevrim): 32 tarama değeri	63,0 msn	408 msn	490 msn	2440 msn
Tarama süresi	**0,156 msn**	**1,042 msn**	**1,250 msn**	**6,250 msn**

A.14.3.2 Analog girişlerin tarama süresi ve güncelleme süreleri

Tablo A-221 Tarama süresi ve güncelleme süresi

Seçim	**Tarama süresi**	**SB güncelleme süresi**
400 Hz (2,5 msn)	0,156 msn	0,156 msn
60 Hz (16,6 msn)	1,042 msn	1,042 msn
50 Hz (20 msn)	1,250 msn	1,25 msn
10 Hz (100 msn)	6,250 msn	6,25 msn

A.14.3.3 Gerilim ve akım için analog girişlerin ölçüm aralıkları (SB ve SM)

Tablo A-222 Gerilim için analog girişi gösterimi (SB ve SM)

Sistem		**Gerilim ölçüm aralığı**				
Desimal	**Onaltılı**	**±10 V**	**±5 V**	**±2,5 V**	**±1,25 V**	
32767	7FFF[1]	11,851 V	5,926 V	2,963 V	1,481 V	Taşma
32512	7F00					
32511	7EFF	11,759 V	5,879 V	2,940 V	1,470 V	Aşırı salınım aralığı
27649	6C01					
27648	6C00	10 V	5 V	2,5 V	1,250 V	Ölçüm aralığı
20736	5100	7,5 V	3,75 V	1,875 V	0,938 V	
1	1	361,7 µV	180,8 µV	90,4 µV	45,2 µV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-20736	AF00	-7,5 V	-3,75 V	-1,875 V	-0,938 V	
-27648	9400	-10 V	-5 V	-2,5 V	-1,250 V	

[1] 7FFF aşağıdaki nedenlerden biri için geri döndürülebilir: Geçerli değerler mevcut olmadan önce (örneğin yol alma sırasında hemen) veya bir tel kopması algılandığında taşma (tabloya bakın).

Sistem		Gerilim ölçüm aralığı				
Desimal	**Onaltılı**	**±10 V**	**±5 V**	**±2,5 V**	**±1,25 V**	
-27649	93FF					Düşük salınım aralığı
-32512	8100	-11,759 V	-5,879 V	-2,940 V	-1,470 V	
-32513	80FF					Aşağı taşma
-32768	8000	-11,851 V	-5,926 V	-2,963 V	-1,481 V	

[1] 7FFF aşağıdaki nedenlerden biri için geri döndürülebilir: Geçerli değerler mevcut olmadan önce (örneğin yol alma sırasında hemen) veya bir tel kopması algılandığında taşma (tabloya bakın).

Tablo A-223 Akım için analog girişlerin gösterimi (SB ve SM)

Sistem		Akım ölçüm aralığı		
Desimal	**Onaltılı**	**0 mA ila 20 mA**	**4 mA ila 20 mA**	
32767	7FFF	> 23,52 mA	> 22,81 mA	Taşma
32511	7EFF	23,52 mA	22,81 mA	Aşırı salınım aralığı
27649	6C01			
27648	6C00	20 mA	20 mA	Nominal aralık
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF			Düşük salınım aralığı
-4864	ED00	-3,52 mA	1,185 mA	
32767[1]	7FFF		< 1,185 mA	Tel kopması (4 ila 20 mA) (Sayfa 1151)
-32768	8000	< -3,52 mA		Aşağı taşma (0 ila 20 mA)

[1] Tel kopması alarmının durumundan bağımsız olarak 32767 (16#7FFF) tel kopma değeri her zaman geri döndürülür.

A.14.3.4 Gerilim ve akım için analog çıkışların ölçüm aralıkları (SB ve SM)

Tablo A-224 Gerilim için analog çıkışı gösterimi (SB ve SM)

Sistem		Gerilim çıkış aralığı	
Desimal	**Onaltılı**	**±10 V**	
32767	7FFF	Bkz. Not 1	Taşma
32512	7F00	Bkz. Not 1	
32511	7EFF	11,76 V	Aşırı salınım aralığı
27649	6C01		
27648	6C00	10 V	Ölçüm aralığı
20736	5100	7,5 V	
1	1	361,7 µV	
0	0	0 V	
-1	FFFF	-361,7 µV	
-20736	AF00	-7,5 V	
-27648	9400	-10 V	
-27649	93FF		Düşük salınım aralığı
-32512	8100	-11,76 V	
-32513	80FF	Bkz. Not 1	Aşağı taşma
-32768	8000	Bkz. Not 1	

[1] Taşma veya aşağı taşma durumunda, analog çıkışlar STOP işletim durumu için yedek değeri alır.

Tablo A-225 Akım için analog çıkışı gösterimi (SB ve SM)

Sistem		Akım çıkışı aralığı		
Desimal	**Onaltılı**	**0 mA ila 20 mA**	**4 mA ila 20 mA**	
32767	7FFF	Bkz. Not 1	Bkz. Not 1	Taşma
32512	7F00	Bkz. Not 1	Bkz. Not 1	
32511	7EFF	23,52 mA	22,81 mA	Aşırı salınım aralığı
27649	6C01			
27648	6C00	20 mA	20 mA	Ölçüm aralığı
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF		4 mA ila 578,7 nA	Düşük salınım aralığı
-6912	E500		0 mA	
-6913	E4FF			Mümkün değil. Çıkış değeri 0 mA ile sınırlıdır.
-32512	8100			
-32513	80FF	Bkz. Not 1	Bkz. Not 1	Aşağı taşma
-32768	8000	Bkz. Not 1	Bkz. Not 1	

[1] Taşma veya aşağı taşma durumunda, analog çıkışlar STOP işletim durumu için yedek değeri alır.

A.14.4 Termo eleman sinyal kartı (SB'ler)

A.14.4.1 SB 1231 1 termo eleman analog girişin teknik verileri

NOT

Bu SB'yi kullanabilmek için V2.0 veya üstü Firmware'e sahip bir CPU'ya ihtiyacınız vardır.

Tablo A-226 Genel teknik veriler

Teknik veriler	SB 1231 AI 1 x 16 Bit termo eleman
Sipariş numarası	6ES7231-5QA30-0XB0
Boyutlar G x Y x D (mm)	38 x 62 x 21
Ağırlık	35 gram
Performans kaybı	0,5 W
Akım tüketimi (SM veriyolu)	5 mA
Akım tüketimi (24 V DC)	20 mA

Tablo A-227 Analog girişler

Teknik veriler		SB 1231 AI 1x16 Bit termo eleman
Girişlerin sayısı		1
Tip		Potansiyelsiz, TC ve mV
Aralık • Nominal aralık (veri sözcüğü) • Taşma/aşağı taşma (veri sözcüğü) • Taşma/aşağı taşma (veri sözcüğü)		Bkz. Termo eleman filtre seçenekleri tablosu (Sayfa 1320).
Çözünürlük	Sıcaklık aralığı	0,1 °C/0,1 °F
	Gerilim	15 Bit artı ön işaret
Maks. dayanım gerilimi		±35 V
Gürültü azaltma		Seçili filtre ayarı için 85 dB (10 Hz, 50 Hz, 60 Hz, 400 Hz)
Ortak mod bastırma		120 V AC'de > 120 dB
Empedans		≥ 10 MΩ
Hassasiyet		Bkz. Termo eleman seçenekler tablosu (Sayfa 1320).
Tekrarlama hassasiyeti		± % 0,05 tam sapma
Ölçme prensibi		Entegre edici
Modül güncelleme süresi		Bkz. Termo eleman filtre seçenekleri tablosu (Sayfa 1320).
Hata, soğuk bağlantı noktası		±1,5 °C
Elektriksel ayırma (alandan lojiğe)		707 V DC (tip kontrolü)
Hat uzunluğu (metre)		Enkodere maks. 100 m
Hat direnci		Maks. 100 Ω

Tablo A-228 Teşhis

Teknik veriler	SB 1231 AI 1 x 16 Bit termo eleman
Taşma/aşağı taşma [1]	Evet
Tel kopması[2, 3]	Evet

[1] Modül projelendirmesi sırasında ilgili alarmlar devre dışı bırakılırsa, "Taşma" ve "Aşağı taşma" teşhis alarmlarından gelen bilgiler de analog veri değerleriyle bildirilir.

[2] Tel kopması alarmı devre dışı bırakılmışsa ve enkoder kablajında açıkta bir tel mevcutsa, modül rastlantısal değerler bildirebilir.

[3] Modül, tel kopma kontrolünü her 6 saniyede bir yürütür; bunun sonucunda, etkinleştirilmiş her kanal için güncelleme süresini her 6 saniyede bir 9 msn uzatır.

SM 1231 TC analog termo eleman sinyal modülü, modül girişlerine bağlı gerilimin değerini ölçer.

SM 1231 analog termo eleman sinyal kartı, sinyal kartı girişlerine bağlı gerilimin değerini ölçer. Sıcaklık ölçüm tipi olarak "termo eleman" veya "gerilim" mümkündür.

- "Termo eleman": Ölçüm değeri, derece cinsinden on ile çarpılarak yayınlanır (örneğin: 25,3 derece, ondalık değer 253 olarak gösterilir).
- "Gerilim": Nominal aralıktaki ölçüm aralığı değeri, 27648 ondalıktır.

A.14.4.2 Bir termo elemanın temel çalışma şekli

İki farklı metalin elektriksel olarak birbirine bağlanmasıyla termo elemanlar oluşur. Burada bağlantı noktasının sıcaklığıyla orantılı bir gerilim üretilir. Bu düşük bir gerilimdir. Bir mikrovolt, çok sayıda dereceyi temsil edebilir. Termo elemanlarla sıcaklık ölçüm yönteminin temeli, bir termo elemanın gerilimini ölçmek, ek bağlantı noktalarını kompanse etmek ve sonuçları doğrusallaştırmaktır.

SM 1231 termo eleman modülüne bir termo eleman bağladığınızda, farklı metallerin iki ucu modülün sinyal bağlantısına bağlanır. İki farklı ucun birbirine bağlandığı noktada termo elemanın sensörü bulunur.

Farklı uçların sinyal bağlantısına bağlandığı noktada iki termo eleman daha oluşturulur. Bağlantının sıcaklığı bir gerilim oluşturur ve bu da termo eleman sensörünün gerilimine eklenir. Bu gerilim kompanse edilmezse, yayınlanan sıcaklık, sensörün sıcaklığından sapacaktır.

Soğuk bağlantının kompanzasyonu sayesinde, termo eleman bağlantı noktasında kompanse edilir. Termo elemanlar için tablolar, bağlantı noktasında genellikle sıfır Santigrat derece olan bir referans sıcaklığını baz almaktadır. Soğuk bağlantı kompanzasyonu sayesinde, bağlantı, sıfır Santigrat dereceye kompanse edilir. Bağlantının termo elemanı ile eklenen gerilim, soğuk bağlantı noktasının kompanzasyonu ile düzeltilir. Modül sıcaklığı, dahili olarak ölçülür ve bunun ardından sensör dönüşümüne eklenecek bir değere dönüştürülür. Düzeltilmiş sensör dönüşümü, daha sonra termo eleman tabloları aracılığıyla doğrusallaştırılır.

Soğuk bağlantı noktasının kompanzasyonunun optimum seviyede gerçekleşmesi için termo eleman modülünün termik stabil bir çevrede olması gerekir. Modül çevresindeki sıcaklıkta yavaş değişim (0,1 °C/dakikadan az), modül spesifikasyonu dahilinde doğru kompanse edilecektir. Modül üzerindeki hava hareketleri de soğuk bağlantı noktasının kompanzasyonunda hatalara yol açar.

Soğuk bağlantı noktası hatalarının daha iyi kompanzasyonu gerekli ise, harici bir izotermal klemens şeridi kullanabilirsiniz. 0 °C'ye referanslı bir klemens şeridi veya 50 °C'ye referanslı bir klemens şeridi, termo eleman modülü ile kullanılabilir.

SM 1231 termo eleman için seçenekler tablosu

SB 1231 termo eleman sinyal kartı tarafından desteklenen çeşitli termo eleman tiplerinin aralıklarını ve hassasiyetini aşağıdaki tabloda bulabilirsiniz.

Tablo A-229 Termo eleman seçenekler tablosu

Tip	Alt aralık için minimum[1]	Nominal aralık alt limiti	Nominal aralık üst limiti	Üst aralık için maksimum[2]	Normal aralık [3, 4] Hassasiyet 25 °C	Normal aralık [1, 2, 6] Hassasiyet -20 °C ila 60 °C
J	-210,0 °C	-150,0 °C	1.200,0 °C	1.450,0 °C	±0,3 °C	±0,6 °C
	-346,0 °F	-238,0 °F	2192,0 °F	2642,0 °F	±0,5 °F	±1,1 °F
K	-270,0 °C	-200,0 °C	1.372,0 °C	1.622,0 °C	±0,4 °C	±1,0 °C
	-454,0 °F	-328,0 °F	2501,6 °F	2951,6 °F	±0,7 °F	±1,8 °F
T	-270,0 °C	-200,0 °C	400,0 °C	540,0 °C	±0,5 °C	±1,0 °C
	-454,0 °F	-328,0 °F	752,0 °F	1004,0 °F	±0,9 °F	±1,8 °F
E	-270,0 °C	-200,0 °C	1.000,0 °C	1.200,0 °C	±0,3 °C	±0,6 °C
	-454,0 °F	-328,0 °F	1832,0 °F	2192,0 °F	±0,5 °F	±1,1 °F
R & S	-50,0 °C	100,0 °C	1.768,0 °C	2.019,0 °C	±1,0 °C	±2,5 °C
	-58,0 °C	212,0 °F	3214,4 °F	3276,6 °F [5]	±1,8 °F	±4,5 °F
B	0,0 °C	200,0 °C	800,0 °C	--	±2,0 °C	±2,5 °C
	32,0 °F	392,0 °F	1472,0 °F	--	±3,6 °F	±4,5 °F
	--	800,0 °C	1.820,0 °C	1.820,0 °C	±1,0 °C	±2,3 °C
	--	1472,0 °F	3276,6 °F [5]	3276,6 °F [5]	±1,8 °F	±4,1 °F
N	-270,0 °C	-200,0 °C	1.300,0 °C	1.550,0 °C	±1,0 °C	±1,6 °C
	-454,0 °F	-328,0 °F	2372,0 °F	2822,0 °F	±1,8 °F	±2,9 °F
C	0,0 °C	100,0 °C	2.315,0 °C	2.500,0 °C	±0,7 °C	±2,7 °C
	32,0 °F	212,0 °F	3276,6 °F [5]	3276,6 °F [5]	±1,3 °F	±4,9 °F
TXK/XK (L)	-200,0 °C	-150,0 °C	800,0 °C	1.050,0 °C	±0,6 °C	±1,2 °C
	-328,0 °F	302,0 °F	1472,0 °F	1922,0 °F	±1,1 °F	±2,2 °F
Gerilim	-32512	-27648 -80 mV	27648 80 mV	32511	± % 0,05	± % 0,1

[1] Alt aralık için minimumun altındaki termo eleman değerleri -32768 olarak yayınlanır.

[2] Üst aralık için maksimumun üzerindeki termo eleman değerleri +32767 olarak yayınlanır.

[3] Soğuk bağlantı noktasındaki dahili hata, tüm aralıklar için ±1,5 °C'dir. Bu değeri, bu tabloda sıralanan hataya eklemelisiniz. Modül, belirtilen spesifikasyonlar ile uyumlu çalışana dek, en azından 30 dakikalık bir ısınma süresine gerek duyar.

[4] Yayılan 970 MHz ile 990 MHz arasındaki bir radyo frekansında, SM 1231 AI 4 × 16 Bit TC'nin hassasiyeti bozulabilir.

[5] °F cinsinden mesajla birlikte alt limit 3276,6 °F

[6] Soğuk bağlantıdaki kompanzasyon hatası, 0 °C'nin altındaki ortam sıcaklığı için karakterize edilmemiştir ve belirtilen değeri aşabilir.

Tablo A-230 SM 1231 termo eleman için filtre seçenekleri tablosu

Bastırma frekansı (Hz)	Entegrasyon süresi (msn)	Sinyal kartı güncelleme süresi (saniye)
10	100	0,306
50	20	0,066
60	16,67	0,056
400[1]	10	0,036

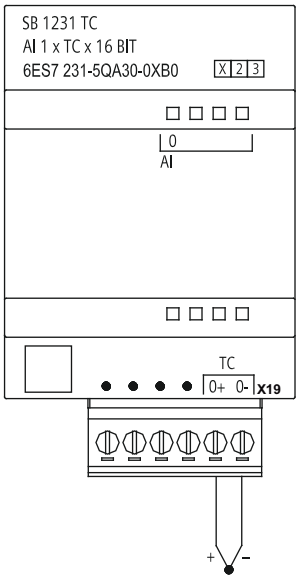
[1] 400 Hz filtre seçiliyken, modülün çözünürlüğünü ve hassasiyetini korumak için gereken entegrasyon süresi 10 msn'dir. Bu seçim, 100 Hz ve 200 Hz'deki gürültüyü de bastırır.

Termo elemanların ölçümü için 100 msn'lik bir entegrasyon süresi önerilir. Daha düşük ayarlanmış entegrasyon süreleri sıcaklığın daha yüksek tekrarlama hassasiyetine neden olur.

NOT

Güç açıldıktan sonra modül, A/D dönüştürücü için dahili kalibrasyonu yürütür. Bu zaman diliminde modül, ilgili kanal için geçerli veriler mevcut olana dek her kanalda 32767 değerini bildirir. Bu başlatma süresinin kullanıcı programında dikkate alınması gerekebilir. Modülün konfigürasyonu başlatma işleminin süresini etkileyebileceğinden, konfigürasyonunuzda modülün davranışını kontrol edin. Gerekli olursa, modülün başlatma süresine uyum sağlamak için kullanıcı programınıza lojik ekleyebilirsiniz.
Bu lojiği, başlatma tamamlanana dek işletimi bloke edecek olan "yol alma OB'sindeki" bir okuma talebi ile uygulayabilirsiniz. Okuma talebini, anında erişim ile uygulamalısınız. Termo elemanın okuma talebinin değeri 32767 ise, bu değer değişene dek okuma işlemi tekrarlanmalıdır. Her modül için, bu sorgulamanın yalnızca modülün en büyük numaralı girişi için yürütülmesi gerekir (modül girişleri, 0 ile 7 arasındaki sırayla başlatılır).

Tablo A-231 SB 1231 AI 1 x 16 termo elemanın devre şeması

SB 1231 AI 1 x 16 Bit termo eleman (6ES7231-5QA30-0XB0)	
	
Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.	

Tablo A-232 SB 1231 AI 1 x 16 Bit termo eleman için bağlantı yerleşimi (6ES7231-5QA30-0XB0)

Pin	X19 (altın kaplamalı)
1	Bağlantı yok
2	Bağlantı yok
3	Bağlantı yok
4	Bağlantı yok
5	AI 0- /TC
6	AI 0+ /TC

A.14.5 RTD sinyal kartları (SB'ler)

A.14.5.1 SB 1231 1 analog girişin teknik verileri nelerdir?

NOT

Bu SB'yi kullanabilmek için V2.0 veya üstü Firmware'e sahip bir CPU'ya ihtiyacınız vardır.

Tablo A-233 Genel teknik veriler

Teknik veriler	SB 1231 AI 1 x 16 Bit RTD
Sipariş numarası	6ES7231-5PA30-0XB0
Boyutlar G x Y x D (mm)	38 x 62 x 2
Ağırlık	35 gram
Performans kaybı	0,7 W
Akım tüketimi (SM veriyolu)	5 mA
Akım tüketimi (24 V DC)	25 mA

Tablo A-234 Analog girişler

Teknik veriler		SB 1231 AI 1 x 16 Bit RTD
Girişlerin sayısı		1
Tip		Modül referansı RTD ve Ohm
Aralık • Nominal aralık (veri sözcüğü) • Taşma/aşağı taşma (veri sözcüğü) • Taşma/aşağı taşma (veri sözcüğü)		Bkz. Seçenekler tabloları (Sayfa 1325).
Çözünürlük	Sıcaklık	0,1 °C / 0,1 °F
	Gerilim	15 Bit artı ön işaret
Maks. dayanım gerilimi		±35 V
Gürültü azaltma		85 dB (10 Hz, 50 Hz, 60 Hz, 400 Hz)
Ortak mod bastırma		> 120 dB

Teknik veriler	SB 1231 AI 1 x 16 Bit RTD
Empedans	≥ 10 MΩ
Hassasiyet	Bkz. Seçenekler tabloları (Sayfa 1325).
Tekrarlama hassasiyeti	± % 0,05 tam sapma
Enkoderin maksimum kayıp gücü	0,5 m W
Ölçme prensibi	Entegre edici
Modül güncelleme süresi	Bkz. Seçenekler tablosu (Sayfa 1325).
Elektriksel ayırma (alandan lojiğe)	707 V DC (tip kontrolü)
Hat uzunluğu (metre)	Enkodere maks. 100 m
Hat direnci	maks. 10 Ω RTD için 20 Ω, 2,7

Tablo A-235 Teşhis

Teknik veriler	SB 1231 AI 1 x 16 Bit RTD
Taşma/aşağı taşma[1, 2]	Evet
Tel kopması[3]	Evet

[1] Modül projelendirmesi sırasında ilgili alarmlar devre dışı bırakılırsa, "Taşma" ve "Aşağı taşma" teşhis alarmlarından gelen bilgiler de analog veri değerleriyle bildirilir.

[2] Direnç ölçüm aralıkları için aşağı taşma denetimi genellikle etkinleştirilmez.

[3] Tel kopması alarmı devre dışı bırakılmışsa ve enkoder kablajında açıkta bir tel mevcutsa, modül rastlantısal değerler bildirebilir.

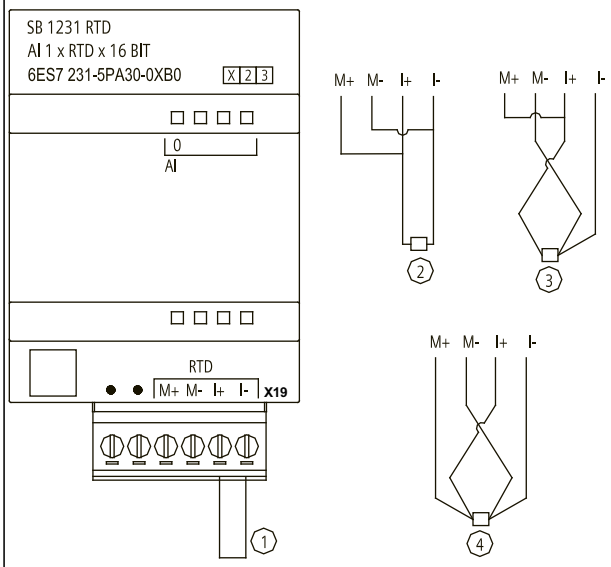
SM 1231 RTD analog termo eleman sinyal kartı, sinyal kartı girişlerine bağlı direncin değerini ölçer. Ölçüm türü olarak "direnç" veya "termik direnç" seçilebilir.

- "Direnç": Nominal aralıktaki ölçüm aralığı değeri, 27648 ondalıktır.
- "Termik direnç": Ölçüm değeri, derece cinsinden on ile çarpılarak yayınlanır (örneğin: 25,3 derece, ondalık değer 253 olarak gösterilir). İklimsel aralığın değerleri, yüz ile çarpılarak derece cinsinden verilir (örnek: 25,34 derece, ondalık değer 2534 olarak gösterilir).

SB 1231 RTD sinyal modülü, enkoder direncine olan 2 telli, 3 telli ve 4 telli bağlantılar üzerinden ölçümleri destekler.

Tablo A-236 SB 1231 AI 1 x 16 Bit RTD'nin devre şeması

SB 1231 AI 1 x 16 Bit RTD (6ES7231-5PA30-0XB0)



① Kullanılmayan RTD girişini geri döngüye alın

② 2 iletkenli RTD

③ 3 iletkenli RTD

④ 4 iletkenli RTD

Not: Soket bağlayıcılar, altın kaplamalı olmalıdır. Sipariş numaralarını, Ek C'deki yedek parçalar bölümünde bulabilirsiniz.

Tablo A-237 SB 1231 AI 1 x 16 Bit RTD için bağlantı yerleşimi (6ES7231-5PA30-0XB0)

Pin	X19 (altın kaplamalı)
1	Bağlantı yok
2	Bağlantı yok
3	AI 0 M+ /RTD
4	AI 0 M- /RTD
5	AI 0 I+ /RTD
6	AI 0 I- /RTD

A.14.5.2 SB 1231 RTD için seçenekler tabloları

Tablo A-238 RTD modülleri tarafından desteklenen çeşitli enkoderler için aralıklar ve hassasiyet

Sıcaklık katsayı-sı	RTD tipi	Minimum alt aralık[1]	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[2]	Normal aralık hassasiyeti, 25 °C'de	Normal aralık hassasiyeti, -20 °C ila 60 °C
Pt 0,003850 ITS90 DIN EN 60751	PT 100 Klima	-145,00 °C	-120,00 °C	-145,00 °C	-155,00 °C	±0,20 °C	±0,40 °C
	Pt 10	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±1,0 °C	±2,0 °C
	Pt 50	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 100						
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0,003902 Pt 0,003916 Pt 0,003920	Pt 100	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0,003910	Pt 10	-273,2 °C	-240,0 °C	1.100,0 °C	1.295 °C	±1,0 °C	±2,0 °C
	Pt 50	-273,2 °C	-240,0 °C	1.100,0 °C	1.295 °C	±0,8 °C	±1,6 °C
	Pt 100						
	Pt 500						
Ni 0,006720 Ni 0,006180	Ni 100	-105,0 °C	-60,0 °C	250,0 °C	295,0 °C	±0,5 °C	±1,0 °C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						
LG-Ni 0,005000	LG-Ni 1000	-105,0 °C	-60,0 °C	250,0 °C	295,0 °C	±0,5 °C	±1,0 °C
Ni 0,006170	Ni 100	-105,0 °C	-60,0 °C	180,0 °C	212,4 °C	±0,5 °C	±1,0 °C
Cu 0,004270	Cu 10	-240,0 °C	-200,0 °C	260,0 °C	312,0 °C	±1,0 °C	±2,0 °C
Cu 0,004260	Cu 10	-60,0 °C	-50,0 °C	200,0 °C	240,0 °C	±1,0 °C	±2,0 °C
	Cu 50	-60,0 °C	-50,0 °C	200,0 °C	240,0 °C	±0,6 °C	±1,2 °C
	Cu 100						
Cu 0,004280	Cu 10	-240,0 °C	-200,0 °C	200,0 °C	240,0 °C	±1,0 °C	±2,0 °C
	Cu 50	-240,0 °C	-200,0 °C	200,0 °C	240,0 °C	±0,7 °C	±1,4 °C
	Cu 100						

[1] Alt aralık için minimumun altındaki RTD değerleri -32768 olarak yayınlanır.

[2] Üst aralık için maksimumun üzerindeki RTD değerleri +32768 olarak yayınlanır.

Tablo A-239 Dayanıklık kapasitesi:

Aralık	Minimum alt aralık	Nominal aralık alt limiti	Nominal aralık üst limiti	Maksimum üst aralık[1]	Normal aralık hassasiyeti, 25 °C'de	Normal aralık hassasiyeti, -20 °C ila 60 °C
150 Ω	Geçerli değil	0 (0 Ω)	27648 (150 Ω)	176,383 Ω	± % 0,05	± % 0,1
300 Ω	Geçerli değil	0 (0 Ω)	27648 (300 Ω)	352,767 Ω	± % 0,05	± % 0,1
600 Ω	Geçerli değil	0 (0 Ω)	27648 (600 Ω)	705,534 Ω	± % 0,05	± % 0,1

[1] Üst aralık için maksimumun üzerindeki direnç değerleri +32767 olarak yayınlanır.

NOT

Bağlı bir enkoderi olmayan etkinleştirilmiş tüm kanallar için modül 32767 değerini bildirir. Bunun dışında açık devreli hat denetimi etkinleştirilirse, modül üzerinde ilgili kırmızı LED'ler yanıp söner.

10 Ω RTD aralıkları için optimum seviyede hassasiyet, 4 telli bağlantılar ile mümkündür.

Bağlantı hatlarının 2 telli moddaki direnci, enkoder ölçümünde bir hataya yol açar. Dolayısıyla ölçüm hassasiyeti artık güvence altına değildir.

Tablo A-240 RTD modülleri için gürültü azaltma ve güncelleme süreleri

Bastırma frekansı seçimi	Entegrasyon süresi	1 kanallı 4/2 telli modül Güncelleme süresi (saniye)	1 kanallı 3 telli modül Güncelleme süresi (saniye)
400 Hz (2,5 msn)	10 msn [1]	0,036	0,071
60 Hz (16,6 msn)	16,67 msn	0,056	0,111
50 Hz (20 msn)	20 msn	0,066	1,086
10 Hz (100 msn)	100 msn	0,306	0,611

[1] Modülün çözünürlüğünü ve hassasiyetini, 400 Hz filtre seçiliyken korumak için entegrasyon süresi 10 msn'dir. Bu seçim, 100 Hz ve 200 Hz'deki gürültüyü de bastırır.

NOT

Güç açıldıktan sonra modül, A/D dönüştürücü için dahili kalibrasyonu yürütür. Bu zaman diliminde modül, ilgili kanal için geçerli veriler mevcut olana dek her kanalda 32767 değerini bildirir. Bu başlatma süresinin kullanıcı programında dikkate alınması gerekebilir. Modülün konfigürasyonu başlatma işleminin süresini etkileyebileceğinden, konfigürasyonunuzda modülün davranışını kontrol edin. Gerekli olursa, modülün başlatma süresine uyum sağlamak için kullanıcı programınıza lojik ekleyebilirsiniz.
Bu lojiği, başlatma tamamlanana dek işletimi bloke edecek olan "yol alma OB'sindeki" bir okuma talebi ile uygulayabilirsiniz. Okuma talebini, anında erişim ile uygulamalısınız. Direnç sıcaklığı sensörünün okuma talebinin değeri 32767 ise, bu değer değişene dek okuma işlemi tekrarlanmalıdır. Her modül için, bu sorgulamanın yalnızca modülün en büyük numaralı girişi için yürütülmesi gerekir (modül girişleri, 0 ile 7 arasındaki sırayla başlatılır).

A.15 BB 1297 batarya kartı

BB 1297 batarya kartı

S7-1200 BB 1297 batarya kartı, gerçek zaman saatinin uzun süreli tamponlanması için kullanılır. S7-1200 CPU'nun sinyal kartının yuvasına takılabilir (Firmware 3.0'dan itibaren). BB'nin işleyebilmesi için, BB 1297'yi cihaz konfigürasyonuna dahil etmeli ve donanım konfigürasyonunu CPU'ya yüklemelisiniz.

Batarya (CR1025 tipi), BB 1297'nin teslim kapsamına dahil değildir ve ayrıca satın alınmalıdır.

NOT

BB 1297 mekanik açısından, 3.0 sürümünden itibaren Firmware'e sahip CPU'lar için tasarlanmıştır.

BB 1297'nin konnektörünün bağlayıcısının CPU'ya takılamaması nedeniyle, BB 1297'yi daha önceki sürümlere sahip CPU'lar ile kullanmayın.

⚠ İKAZ

Belirtilmeyen bir bataryanın BB 1297'ye takılması veya belirtilmeyen bir bataryanın anahtarlama devresine başka bir şekilde bağlanması, yangına veya bileşenlerin hasar görmesine ve makinelerin öngörülemeyen şekilde çalışmasına neden olabilir.

Yangın veya makinelerin öngörülemeyen bir şekilde çalışması, hayati tehlikeye yol açabilecek yaralanmalara veya maddi hasarlara neden olabilir.

Gerçek zaman saatini tamponlamak için sadece belirtilen CR1025 bataryaları kullanın.

Tablo A-241 Genel teknik veriler

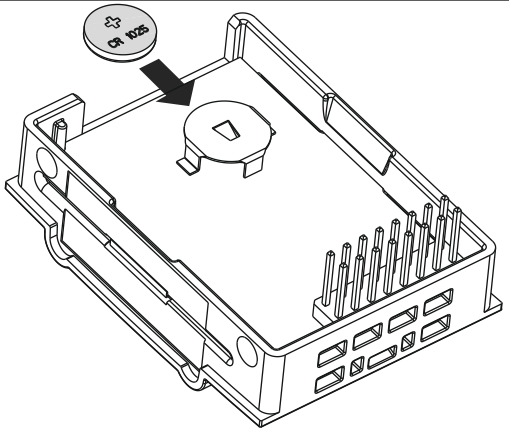
Teknik veriler	**BB 1297 batarya kartı**
Sipariş numarası	6ES7297-0AX30-0XA0
Boyutlar G x Y x D (mm)	38 x 62 x 21
Ağırlık	28 gram
Performans kaybı	0,5 W
Akım tüketimi (SM veriyolu)	11 mA
Akım tüketimi (24 V DC)	Yok

Batarya (dahil değildir)	**BB 1297 batarya kartı**
Şebeke kesintisini köprüleme	yaklaşık 1 yıl
Batarya tipi	CR1025 Bkz. BB 1297 batarya kartının bataryasının takılması veya değiştirilmesi (Sayfa 46)
Nominal gerilim	3 V
Nominal kapasite	En azından 30 mAH

Teşhis	**BB 1297 batarya kartı**
Kritik batarya durumu	< 2,5 V

Teşhis	**BB 1297 batarya kartı**
Batarya teşhisi	Düşük gerilim göstergesi: • Düşük batarya gerilimi, CPU'nun MAINT-LED'inin sürekli sarı yanmasına yol açar. • Teşhis tamponu olayı: 16#06:2700 "alt modül bakımı gerekli: En az bir batarya tükenmiş (BATTF)"
Batarya durumu	Batarya durumu Bit'i 0 = Batarya TAMAM 1 = Batarya zayıf
Batarya durumunun güncellenmesi	Batarya durumu yol almada ve daha sonra CPU RUN işletim durumundayken günde bir kez güncellenir.

Tablo A-242 Batarya kartı BB 1297 için montaj görseli

BB 1297 batarya kartı (6ES7297-0AX30-0XA0)	
	

A.16 İletişim arayüzleri

A.16.1 PROFIBUS

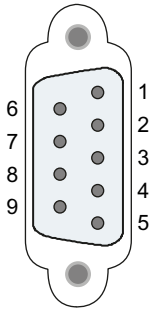
A.16.1.1 CM 1242-5 PROFIBUS DP-SLAVE

Tablo A-243 CM 1242-5'in teknik verileri

Teknik veriler	
Sipariş numarası	6GK7242-5DX30-0XE0
Arayüzler	
PROFIBUS'a bağlantı	9 pinli Sub-D soket
Ağ bileşenleri (örn. optik ağ bileşenleri) bağlanırken PROFIBUS arayüzündeki maksimum akım tüketimi	5 V'ta 15 mA (yalnızca veriyolu sonlandırması için) *)
İzin verilen ortam koşulları	
Ortam sıcaklığı • Depolama sırasında • Taşıma sırasında • dikey montajda işletim sırasında (yatay montaj rayı) • yatay montajda işletim sırasında (yatay montaj rayı)	 • -40 °C ila 70 °C • -40 °C ila 70 °C • 0 °C ila 55 °C • 0 °C ila 45 °C
İşletim sırasında 25 °C'de bağıl nem, yoğuşma olmadan, maks.	% 95
Koruma türü	IP20
Gerilim beslemesi, akım tüketimi ve kayıp güç	
Gerilim beslemesinin türü	DC
Arka panel veriyolundan gerilim beslemesi	5 V
Akım tüketimi (tipik)	150 mA
Efektif kayıp güç (tipik)	0,75 W
Potansiyel ayırma • PROFIBUS arayüzü, şasiye • PROFIBUS arayüzü, dahili anahtarlamaya	1 dakika için 710 V AC
Boyutlar ve ağırlıklar	
• Genişlik • Yükseklik • Derinlik	• 30 mm • 100 mm • 75 mm
Ağırlık • Net ağırlık • Ambalaj dahil ağırlık	 • 115 g • 152 g

A.16.1.2 CM 1242-5'in Sub-D soketinin bağlantı yerleşimi

PROFIBUS arayüzü



Tablo A-244 Sub-D soketin bağlantı yerleşimi

Pin	Açıklama	Pin	Açıklama
1	- dolu değil -	6	P5V2: Gerilim beslemesi +5V
2	- dolu değil -	7	- dolu değil -
3	RxD/TxD-P: Veri kablosu B:	8	RxD/TxD-N: Veri kablosu A
4	RTS	9	- dolu değil -
5	M5V2: Veri referans potansiyeli (şasi DGND)	Muhafaza	Şasi bağlantısı

A.16.1.3 CM 1243-5 PROFIBUS DP-Master

Tablo A-245 CM 1243-5'in teknik verileri

Teknik veriler	
Sipariş numarası	6GK7243-5DX30-0XE0
Arayüzler	
PROFIBUS'a bağlantı	9 pinli Sub-D soket
Ağ bileşenleri (örn. optik ağ bileşenleri) bağlanırken PROFIBUS arayüzündeki maksimum akım tüketimi	5 V'ta 15 mA (yalnızca veriyolu sonlandırması için) *)
İzin verilen ortam koşulları	
Ortam sıcaklığı • Depolama sırasında • Taşıma sırasında • dikey montajda işletim sırasında (yatay montaj rayı) • yatay montajda işletim sırasında (yatay montaj rayı)	 • -40 °C ila 70 °C • -40 °C ila 70 °C • 0 °C ila 55 °C • 0 °C ila 45 °C

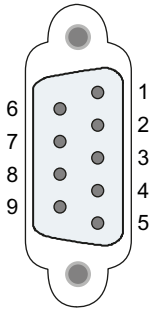
Teknik veriler	
İşletim sırasında 25 °C'de bağıl nem, yoğuşma olmadan, maks.	% 95
Koruma türü	IP20
Gerilim beslemesi, akım tüketimi ve kayıp güç	
Gerilim beslemesinin türü	DC
Gerilim beslemesi / harici • minimum • maksimum	24 V • 19,2 V • 28,8 V
Akım tüketimi (tipik) • +24 V DC'den • S7-1200 arka panel veriyolundan	 • 100 mA • 0 mA
Efektif kayıp güç (tipik) • +24 V DC'den • S7-1200 arka panel veriyolundan	 • 2,4 W • 0 W
Harici besleme 24 V DC • Min. hat kesiti • Maks. hat kesiti • Vidalı klemenslerin sıkma torku	 • min.: 0,14 mm^2 (AWG 25) • maks.: 1,5 mm^2 (AWG 15) • 0,45 Nm (4 lb-in)
Potansiyel ayırma • PROFIBUS arayüzü, şasiye • PROFIBUS arayüzü, dahili anahtarlamaya	1 dakika için 710 V AC
Boyutlar ve ağırlıklar	
• Genişlik • Yükseklik • Derinlik	• 30 mm • 100 mm • 75 mm
Ağırlık • Net ağırlık • Ambalaj dahil ağırlık	 • 134 g • 171 g

NOT

CM 1243-5 (PROFIBUS Master modülü), CPU'nun 24 V DC enkoder beslemesi üzerinden beslenmelidir.

A.16.1.4 CM 1243-5'in Sub-D soketinin bağlantı yerleşimi

PROFIBUS arayüzü



Tablo A-246 Sub-D soketin bağlantı yerleşimi

Pin	Açıklama	Pin	Açıklama
1	- dolu değil -	6	VP: +5 V gerilim beslemesi, yalnızca veriyolu sonlandırma dirençleri içindir; harici cihazların beslenmesi için değildir.
2	- dolu değil -	7	- dolu değil -
3	RxD/TxD-P: Veri kablosu B:	8	RxD/TxD-N: Veri kablosu A
4	CNTR-P: RTS	9	- dolu değil -
5	DGND: Veri sinyalleri ve VP için şasi	Muhafaza	Şasi bağlantısı

PROFIBUS kablosu

NOT

PROFIBUS kablosu blendajlamasının döşenmesi

PROFIBUS kablosu blendajlaması döşenmelidir.

Bunun için PROFIBUS kablosunun ucundan bir parça izolasyonu sıyırın ve blendajı fonksiyonel topraklamaya bağlayın.

A.16.2 CP 1242-7

NOT
CP 1242-7 GPRS denizcilik sektöründeki uygulamalar için ruhsatı yoktur
CP 1242-7, denizcilik sektörü için ruhsatı yoktur.

NOT
Bu modülü kullanabilmek için V2.0 veya üstü Firmware'e sahip bir CPU'ya ihtiyacınız vardır.

A.16.2.1 CP 1242-7 GPRS

Tablo A-247 CP 1242-7 GPRS V2'nin teknik verileri

Teknik veriler	
Sipariş numarası	6GK7242-7KX3-0XE0
Kablosuz bağlantı arayüzü	
Anten bağlantısı	SMA soketi
Nominal empedans	50 Ohm
Kablosuz bağlantı	
Maksimum gönderim gücü	• GSM 850, Class 4: +33 dBm ±2dBm • GSM 900, Class 4: +33 dBm ±2dBm • GSM 1800, Class 1: +30 dBm ±2dBm • GSM 1900, Class 1: +30 dBm ±2dBm
GPRS	Çoklu yuva Sınıf 10 Terminal cihaz Sınıf B Kodlama şeması 1...4 (GMSK)
SMS	İşletim türü, giden: MO Hizmeti: Noktadan noktaya
İzin verilen ortam koşulları	
Ortam sıcaklığı • Depolama sırasında • Taşıma sırasında • dikey montajda işletim sırasında (yatay montaj rayı) • yatay montajda işletim sırasında (yatay montaj rayı)	 • -40 ℃ ila 70 ℃ • -40 ℃ ila 70 ℃ • 0 °C ila 55 °C • 0 ℃ ila 45 ℃
İşletim sırasında 25 °C'de bağıl nem, yoğuşma olmadan, maks.	% 95
Koruma türü	IP20
Gerilim beslemesi, akım tüketimi ve kayıp güç	
Gerilim beslemesinin türü	DC
Gerilim beslemesi / harici • minimum • maksimum	24 V • 19,2 V • 28,8 V

Teknik veriler	
Akım tüketimi (tipik) • +24 V DC'den • S7-1200 arka panel veriyolundan	 • 100 mA • 0 mA
Efektif kayıp güç (tipik) • +24 V DC'den • S7-1200 arka panel veriyolundan	 • 2,4 W • 0 W
+24 V DC güç kaynağı • Min. hat kesiti • Maks. hat kesiti • Vidalı klemenslerin sıkma torku	 • min.: 0,14 mm^2 (AWG 25) • maks.: 1,5 mm^2 (AWG 15) • 0,45 Nm (4 lb-in)
Potansiyel ayırma Dahili devreye karşı güç kaynağı	1 dakika için 710 V AC
Boyutlar ve ağırlıklar	
• Genişlik • Yükseklik • Derinlik	• 30 mm • 100 mm • 75 mm
Ağırlık • Net ağırlık • Ambalaj dahil ağırlık	 • 133 g • 170 g

NOT

Antenler nedeniyle CPU'da parazit oluşmasının önlenmesi

Bir anten çok yakın yerleştirilmişse veya önerilmeyen antenler kullanırsanız, bu durum CPU'nun arızalanmasına yol açabilir. Önerilen antenler için, LTE/UMTS/GSM için ANT794-4MR anten kompakt işletim kılavuzuna () (yalnızca İngilizce ve Almanca dillerinde mevcuttur) bakın.

A.16.2.2 GSM/GPRS anteni ANT794-4MR

GSM/GPRS anteni ANT794-4MR'nin teknik verileri

ANT794-4MR	
Sipariş numarası	6NH9860-1AA00
Mobil iletişim ağları	GSM/GPRS
Frekans aralıkları	• 824 ila 960 MHz (GSM 850, 900) • 1710 ila 1880 MHz (GSM 1800, 900) • 1900 ila 2200 MHz (GSM / UMTS)
Karakteristik	yönsüz
Anten kazancı	0 dB
Empedans	50 Ohm
Durağan dalga oranı (SWR)	< 2,0
Maks. güç	20 W

ANT794-4MR	
Polarite	Doğrusal yatay
Konnektör	SMA
Anten kablosu uzunluğu	5 m
Dış malzeme	Sert PVC, UV ışınlarına dayanıklı
Koruma türü	IP20
İzin verilen ortam koşulları • Çalışma sıcaklığı • Taşıma/depolama sıcaklığı • Bağıl nem	 • -40 °C ila +70 °C • -40 °C ila +70 °C • % 100
Dış malzeme	Sert PVC, UV ışınlarına dayanıklı
Yapısal tasarım	5 m sabit bağlı kablo ve SMA fişli anten
Ölçüler (D x U) mm cinsinden	25 x 193
Ağırlık • Kablolu anten • Montaj parçaları	 • 310 g • 54 g
Montaj	Birlikte gelen tutma açıları

A.16.2.3 Yassı anten ANT794-3M

Yassı anten ANT794-3M'nin teknik verileri

ANT794-3M		
Sipariş numarası	6NH9870-1AA00	
Mobil iletişim ağları	**GSM 900**	**GSM 1800/1900**
Frekans aralıkları	890 - 960 MHz	1710 - 1990 MHz
Durağan dalga oranı (VSWR)	≤ 2:1	≤ 1,5:1
Geri dönüş kaybı (Tx)	≈ 10 dB	≈ 14 dB
Anten kazancı	0 dB	
Empedans	50 Ohm	
Maks. güç	10 W	
Anten kablosu	SMA fişli HF kablo RG 174 (kalıcı olarak bağlı)	
Kablo uzunluğu	1,2 m	
Koruma türü	IP64	
İzin verilen sıcaklık aralığı	-40 °C ila +75 °C	
Alev alabilirlik	UL 94 V2	
Dış malzeme	ABS Polylac PA-765, açık gri (RAL 7035)	
Ölçüler (G x U x Y) mm cinsinden	70,5 x 146,5 x 20,5	
Ağırlık	130 g	

A.16.3 CM 1243-2 AS-i Master

A.16.3.1 AS-i Master CM 1243-2'nin teknik verileri

Tablo A-248 AS-i Master CM 1243-2'nin teknik verileri

Teknik veriler	
Sipariş numarası	3RK7243-2AA30-0XB0
Firmware sürümü	V1.0
Tarih	01.12.2011
Arayüzler	
Maksimum akım tüketimi S7-1200 arka panel veriyolundan AS-i hattından	 maks. 250 mA, Besleme gerilimi S7-1200 iletişim veriyolu 5 V DC maks. 100 mA
Şunlar arasındaki maksimum akım dayanıklılığı ASI+/ASI klemensleri	8 A
Fiş yerleşimi	Bkz. Bölüm AS-i Master'ın elektrik bağlantısı (Sayfa 1337)
İletken kesiti	0,2 mm² (AWG 24) ... 3,3 mm² (AWG 12)
ASI fişinin sıkma torku	0,56 Nm
İzin verilen ortam koşulları	
Ortam sıcaklığı Depolama sırasında Taşıma sırasında dikey montajda işletim sırasında (yatay standart montaj rayı) yatay montajda işletim sırasında (dikey standart montaj rayı)	 -40 °C ... 70 °C -40 °C ... 70 °C 0 °C ... 55 °C 0 °C ... 45 °C
İşletim sırasında 25 °C'de bağıl nem, yoğuşma olmadan, maks.	% 95
Koruma türü	IP20
Gerilim beslemesi, akım tüketimi, kayıp güç	
Gerilim beslemesinin türü	DC
Akım tüketimi (tipik) S7-1200 arka panel veriyolundan	 200 mA
Toplam kayıp güç (tipik): • S7-1200 arka panel veriyolundan • AS-i hattından	 1 W 2,4 W
Boyutlar ve ağırlıklar	
Genişlik Yükseklik Derinlik	30 mm 100 mm 75 mm
Ağırlık Net ağırlık Ambalaj dahil ağırlık	 122 g 159 g

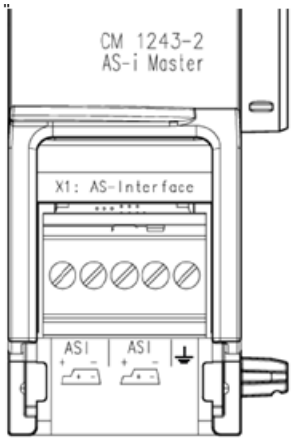
A.16.3.2 AS-i Master'ın elektrik bağlantısı

AS-i Master CM 1243-2 için gerilim beslemesi

AS-i Master CM 1243-2, S7-1200'ün iletişim veriyolu üzerinden gerilimle beslenir. Bu sayede, AS-i besleme gerilimi kesilse bile S7-1200'e bir teşhis mesajı gönderilebilir. İletişim veriyoluna olan bağlantı, AS-i Master CM 1243-2'nin sağ tarafındadır.

AS-Interface bağlantı klemensleri

AS-i kablosunu bağlamak için çıkarılabilir klemens, AS-i Master CM 1243-2'nin ön tarafındaki alt koruyucu kapağın arkasında bulunur.



AS-i profilli kablo kullanıldığında, bağlantı kablosunun doğru polaritesi ilgili sembolden görülebilir.



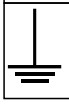
Klemens bloğunun sökülmesi ve yeniden takılması hakkında bilgi için "Montaj" bölümüne (Sayfa 50) bakın.

NOT

Bağlantı kontaklarının maksimum dayanıklılık kapasitesi

Bağlantı kontaklarının dayanım kapasitesi maks. 8 A'dir. AS-i hattında bu değer aşılırsa, AS-i Master CM 1243-2 AS-i hattına "döngüye alınmamalı", bunun yerine tek yönlü aktarım hattı üzerinden bağlanmalıdır (AS-i Master CM 1243-2'nin yalnızca bir bağlantı çifti doludur).
AS-i Master üzerinden akım geçişi ve 4 amperin üzerindeki akımlarda, kullanılan kabloların en az 75 °C çalışma sıcaklığına uygun olması gerektiğini de dikkate alın.
AS-i kablosunun bağlanması ile ilgili daha fazla bilgiyi, " SIMATIC S7-1200 için AS-i Master CM 1243-2 ve AS-i veri ayırma ünitesi DCM 1271" İşletme kitabının "Modüllerin kurulumu, bağlantısı ve işletime alınması" alt bölümünde bulabilirsiniz.

Fiş yerleşimi

Yazı	Anlamı
ASI+	AS-i bağlantısı – pozitif polarite
ASI–	AS-i bağlantısı – negatif polarite
	Fonksiyonel topraklama

A.16.4 RS232, RS422 ve RS485

A.16.4.1 CB 1241 RS485'in teknik verileri

NOT

Bu CB'yi kullanabilmek için V2.0 veya üstü Firmware'e sahip bir CPU'ya ihtiyacınız vardır.

Tablo A-249 Genel teknik veriler

Teknik veriler	CB 1241 RS485
Sipariş numarası	6ES7241-1CH30-1XB0
Boyutlar (G x Y x D) (mm)	38 x 62 x 21
Ağırlık	40 gram

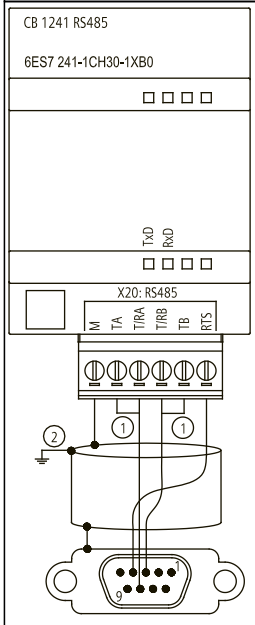
Tablo A-250 Gönderici ve alıcı

Teknik veriler	CB 1241 RS485
Tip	RS485 (2 telli yarı dubleks)
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 sn, 3 V devam eden efektif değer
Gönderici diferansiyel çıkış gerilimi	R_L= 100 Ω'da min. 2 V min. R_L = 54 Ω'da 1,5 V
Sonlandırma ve ön gerilim	10 k +5 V'luk B'de, RS485 Pin 3 10 k GND'ye A'da, RS485 Pin 4
Opsiyonel sonlandırma	TB pinini T/RB pinine kısa devre yapın, efektif sonlandırma empedansı 127 Ω'dur, RS485 pin 3'e bağlayın TA pinini T/RA pinine kısa devre yapın, efektif sonlandırma empedansı 127 Ω'dur, RS485 Pin 4'e bağlayın
Alıcı çıkış empedansı	min. 5,4 kΩ sonlandırma dahil
Alıcının yanıt limiti/hassasiyeti	min. +/- 0,2 V, 60 mV tip. histerizis
Potansiyel ayırma Şasiye RS485 sinyali CPU lojiğine RS485 sinyali	707 V DC (tip kontrolü)
Hat uzunluğu (blendajlı)	maks. 1000 m
Baud hızı	300 Baud, 600 Baud, 1,2 kBit/sn, 2,4 kBit/sn, 4,8 kBit/sn, 9,6 kBit/sn (standart), 19,2 kBit/sn, 38,4 kBit/sn, 57,6 kBit/sn, 76,8 kBit/sn, 115,2 kBit/sn

Teknik veriler	**CB 1241 RS485**
Parite	Parite yok (standart), çift sayılı, tek sayılı, işaret (parite Bit'i her zaman 1'de) ve boşluk (parite Bit'i her zaman 0'da).
Stop Bit'lerinin sayısı	1 (standart), 2
Akış kontrolü	Desteklenmiyor
Bekleme süresi	0 ila 65535 msn

Tablo A-251 Güç kaynağı

Teknik veriler	**CB 1241 RS485**
Kayıp güç	1,5 W
Akım tüketimi (SM veriyolu), maks.	50 mA
Maks. akım tüketimi (24 V DC)	80 mA

CB 1241 RS485 (6ES7241-1CH30-1XB0)	
	
① Ağı tamamlamak için "TA" ve TB"yi gösterildiği gibi bağlayın. (Yalnızca RS485 ağındaki terminal cihazları sonlandırın.)	
② Blendajlı, bükümlü iletken çiftleri kullanın ve kablo blendajını toprağa bağlayın.	

RS485 ağının yalnızca iki ucunu sonlandırırsınız. İki uç cihaz arasındaki cihazlar sonlandırılmaz. Daha fazla bilgi için, "Bir RS485 veriyolu bağlantı fişinin sonlandırılması" (Sayfa 889) bölümünde bulabilirsiniz.

Tablo A-252 CB 1241 RS485 için bağlantı yerleşimi (6ES7241-1CH30-1XB0)

Pin	**9 kutuplu soket bağlayıcı**	**X20**
1	RS485 / mantıksal toprak	--
2	RS485 / serbest	--
3	RS485 / TxD+	4 - T/RB

Pin	9 kutuplu soket bağlayıcı	X20
4	RS485 / RTS	6 - RTS
5	RS485 / mantıksal toprak	--
6	RS485 / 5 V güç	--
7	RS485 / serbest	--
8	RS485 / TxD-	3 - T/RA
9	RS485 / serbest	--
Manşon		1 - M

A.16.4.2 CM 1241 RS232'nin teknik verileri

Tablo A-253 Genel teknik veriler

Teknik veriler	CM 1241 RS232
Sipariş numarası	6ES7241-1AH32-0XB0
Boyutlar (mm)	30 x 100 x 75
Ağırlık	150 gram

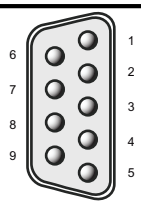
Tablo A-254 Gönderici ve alıcı

Teknik veriler	CM 1241 RS232
Tip	RS232 (tam dupleks)
Göndericinin çıkış gerilimi	R_L = 3 kΩ'da min. +/- 5 V
Göndericinin çıkış gerilimi	maks. +/- 15 V DC
Alıcı çıkış empedansı	min. 3 kΩ
Alıcının yanıt limiti/hassasiyeti	min. 0,8 V Low, maks. 2,4 V High, tip. histerizis 0,5 V
Alıcı giriş gerilimi	maks. +/- 30 V DC
Potansiyel ayırma Şasiye RS-232 sinyali CPU lojiğine RS-232 sinyali	707 V DC (tip kontrolü)
Hat uzunluğu (blendajlı)	maks. 10 m
Baud hızı	300 Baud, 600 Baud, 1,2 kBit/sn, 2,4 kBit/sn, 4,8 kBit/sn, 9,6 kBit/sn (standart), 19,2 kBit/sn, 38,4 kBit/sn, 57,6 kBit/sn, 76,8 kBit/sn, 115,2 kBit/sn
Parite	Parite yok (standart), çift sayılı, tek sayılı, işaret (parite Bit'i her zaman 1'de) ve boşluk (parite Bit'i her zaman 0'da).
Stop Bit'lerinin sayısı	1 (standart), 2
Akış kontrolü	Donanım, yazılım
Bekleme süresi	0 ila 65535 msn

Tablo A-255 Güç kaynağı

Teknik veriler	CM 1241 RS232
Kayıp güç	1 W
+5 V DC'den	200 mA

Tablo A-256 RS232 soket bağlayıcı (fiş)

Pin	Açıklama	Soket bağlayıcı (fiş)	Pin	Açıklama
1 DCD	Veri taşıyıcı algılama: Giriş		6 DSR	Veri seti hazır: Giriş
2 RxD	DCE'den veri alımı: Giriş		7 RTS	Gönderim talebi: Çıkış
3 TxD	DCE'ye veri gönderimi: Çıkış		8 CTS	Gönderim için hazır: Giriş
4 DTR	Veri terminali hazır: Çıkış		9 RI	Çağrı göstergesi (kullanılmıyor)
5 GND	Lojik şasisi		SHELL	Topraklama bağlantısı

A.16.4.3 CM 1241 RS422/485'in teknik verileri

CM 1241 RS422/485'in teknik verileri

Tablo A-257 Genel teknik veriler

Teknik veriler	CM 1241 RS422/485
Sipariş numarası	6ES7241-1CH32-0XB0
Boyutlar G x Y x D (mm)	30 x 100 x 75
Ağırlık	155 gram

Tablo A-258 Gönderici ve alıcı

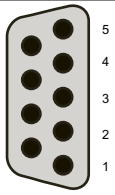
Teknik veriler	CM 1241 RS422/485
Tip	RS422 veya RS485, 9 pinli Sub-D soket
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 sn, 3 V devam eden efektif değer
Gönderici diferansiyel çıkış gerilimi	R_L= 100 Ω'da min. 2 V min. R_L = 54 Ω'da 1,5 V
Sonlandırma ve ön gerilim	10 kΩ +5 V B'de, PROFIBUS Pin 3 10 kΩ GND A'da, PROFIBUS Pin 8 Dahili ön gerilim opsiyonları mevcut veya dahili ön gerilim yok Tüm durumlarda harici sonlandırma gereklidir, bilgi için S7-1200 otomasyon sisteminin sistem kitabındaki RS485 veriyolu bağlantı fişinin sonlandırılması (Sayfa 889) ve RS422 ve RS485'in yapılandırılması (Sayfa 936) konularına bakın.
Alıcı çıkış empedansı	min. 5,4 kΩ sonlandırma dahil
Alıcının yanıt limiti/hassasiyeti	min. +/- 0,2 V, 60 mV tip. histerizis
Potansiyel ayırma Şasiye RS485 sinyali CPU lojiğine RS485 sinyali	707 V DC (tip kontrolü)
Hat uzunluğu (blendajlı)	maks. 1000 m (baud hızına bağlıdır)

Teknik veriler	CM 1241 RS422/485
Baud hızı	300 Baud, 600 Baud, 1,2 kBit/sn, 2,4 kBit/sn, 4,8 kBit/sn, 9,6 kBit/sn (standart), 19,2 kBit/sn, 38,4 kBit/sn, 57,6 kBit/sn, 76,8 kBit/sn, 115,2 kBit/sn
Parite	Parite yok (standart), çift sayılı, tek sayılı, işaret (parite Bit'i her zaman 1'de) ve boşluk (parite Bit'i her zaman 0'da).
Stop Bit'lerinin sayısı	1 (standart), 2
Akış kontrolü	XON/XOFF, RS422 modda desteklenir
Bekleme süresi	0 ila 65535 msn

Tablo A-259 Güç kaynağı

Teknik veriler	CM 1241 RS422/485
Kayıp güç	1,1 W
+5 V DC'den	220 mA

Tablo A-260 RS485 veya RS422 soket bağlayıcı (soket)

Pin	Açıklama	Soket bağlayıcı (soket)	Pin	Açıklama
1	Lojik veya iletişim şasisi		6 PWR	100 Ohm seri dirençli +5 V: Çıkış
2 TxD+ [1]	RS422 için bağlıdır RS485 için dolu değil: Çıkış		7	Bağlı değil
3 TxD+ [2]	Sinyal B (RxD/TxD+): Giriş/çıkış		8 TXD- [2]	Sinyal A (RxD/TxD-): Giriş/çıkış
4 RTS [3]	Çıkış gönderim talebi/RTS (TTL seviyesi)		9 TXD- [1]	RS422 için bağlıdır RS485 için dolu değil: Çıkış
5 GND	Lojik veya iletişim şasisi		SHELL	Topraklama bağlantısı

[1] Pin 2 (TxD+) ve pin 9 (TxD-), RS422 gönderim sinyalleri

[2] Pin 3 (RxD/Tx+) ve pin 8 (RxD/TxD-), RS485 gönderim ve alım sinyalleri. RS422'de, pin 3 RxD+ sinyaline ve pin 8 RxD- sinyaline karşılık gelir.

[3] RTS, TTL seviyesinde bir sinyaldir ve bu sinyali kullanarak başka bir yarı dupleks cihaza kumanda etmek için kullanılabilir. Gönderim e sırasında aktif, aksi halde pasiftir.

A.17 TeleService (TS adaptörü ve TS adaptör modülü)

Aşağıdaki işletme kitapları, TS adaptörü IE Basic ve TS adaptörü modülünün teknik verilerini içerir:

- Endüstriyel yazılım mühendisliği araçları
  Modüler TS adaptörü
- Endüstriyel yazılım mühendisliği araçları
  TS adaptörü IE Basic

Bu ürünle ilgili daha fazla bilgiyi ve ürün dokümantasyonunu, TS adaptörü ürün kataloğunu içeren web sitesinde (https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=TS%20Adapter%20IE%20basic&tab=) bulabilirsiniz.

A.18 SIMATIC memory card'lar

Kapasite	Sipariş numarası
32 GB	6ES7954-8LT02-0AA0
2 GB	6ES7954-8LP01-0AA0
256 MB	6ES7954-8LL02-0AA0
24 MB	6ES7954-8LF02-0AA0
12 MB	6ES7954-8LE02-0AA0
4 MB	6ES7954-8LC02-0AA0

A.19 Giriş simülatörleri

Tablo A-261 Genel teknik veriler

Teknik veriler	8 bağlantı klemensli simülatör	14 bağlantı klemensli simülatör	CPU 1217C simülatörü
Sipariş numarası	6ES7274-1XF30-0XA0	6ES7274-1XH30-0XA0	6ES7274-1XK30-0XA0
Boyutlar (G x Y x D) (mm)	43 x 35 x 23	67 x 35 x 23	93 x 40 x 23
Ağırlık	20 gram	30 gram	43 gram
Giriş ve çıkışlar	8	14	14
Şunlarla kullanım:	CPU 1211C, CPU 1212C	CPU 1214C, CPU 1215C	CPU 1217C

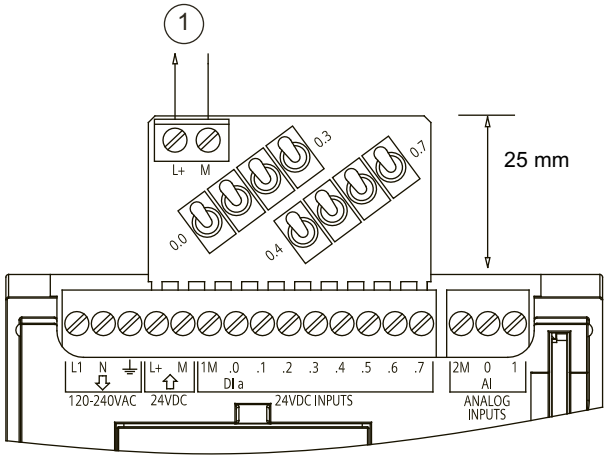
⚠ İKAZ

Giriş simülatörlerinin güvenli kullanımı

Bu giriş simülatörleri Sınıf I DIV 2 ve Sınıf I Bölge 2 patlama tehlikesi bulunan işletme tesislerinde kullanım için uygun değildir. Şalterler, Sınıf I DIV2 veya Sınıf I Bölge 2 işletmelerde kullanıldığında ateşlenebilir/patlayabilir. İzinsiz kullanım, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilir.

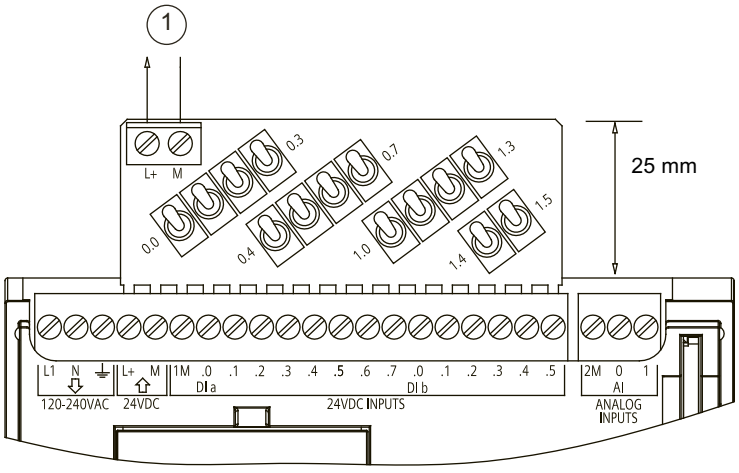
Bu giriş simülatörlerini yalnızca Ex olmayan alanlarda kullanın. Bunları, Sınıf I DIV 2 ve Sınıf I Bölge 2 tehlike gruplarının patlama tehlikesi bulunan işletme sahalarında kullanmayın.

8 bağlantı klemensli simülatör (6ES7274-1XF30-0XA0)



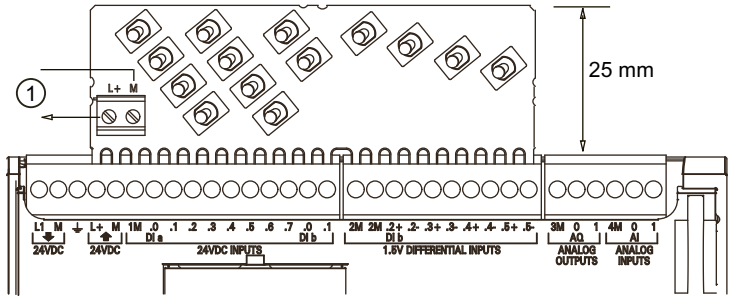
① Enkoder beslemesi 24 V DC

14 bağlantı klemensli simülatör (6ES7274-1XF30-0XA0)



① Enkoder beslemesi 24 V DC

CPU 1217C simülatörü (6ES7274-1XK30-0XA0)



① Enkoder beslemesi 24 V DC

A.20 S7-1200 potansiyometre modülü

S7-1200 potansiyometre modülü, S7-1200 CPU için aksesuardır. Potansiyometrenin konumuyla orantılı olarak her potansiyometre, CPU'nun iki analog girişinin her birine, 0 V DC ila 10 V DC beslemek için bir çıkış gerilimi üretir. Potansiyometrenin kurulumunda aşağıdaki yöntemi izleyin:

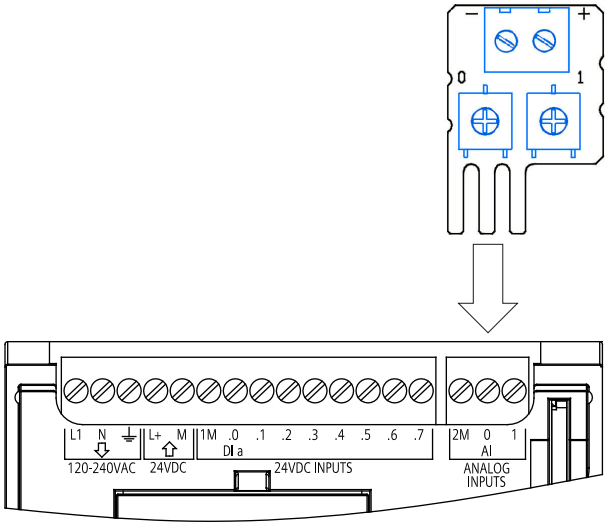
1. Devre kartının "parmaklarını" bir S7-1200 CPU'nun herhangi bir analog giriş klemens bloğuna takın ve potansiyometre modülü üzerindeki 2 yollu bağlantısına harici bir DC gerilim beslemesine bağlayın.
2. Küçük bir tornavida kullanarak potansiyometreyi ayarlayın: Gerilim çıkışını artırmak istiyorsanız sağa çevirin, gerilim çıkışını azaltmak için sola çevirin.

NOT

S7-1200 potansiyometre modülü ile çalışırken elektrostatik tehlike altında olan parçalar koruma önlemlerine uyun.

Teknik veriler	**S7-1200 potansiyometre modülü**
Sipariş numarası	6ES7274-1XA30-0XA0
Şunlarla kullanım:	Tüm S71200 CPU'lar
Potansiyometre sayısı	2
Boyutlar (G x Y x D) (mm)	20 x 33 x 14
Ağırlık	26 gram
2 yollu bağlantı üzerinde kullanıcı tarafından sağlanan gerilim girişi[1] (Sınıf 2, güç sınırlı veya PLC'den enkoder gerilimi)	16,4 V DC ila 28,8 V DC
Hat uzunluğu (metre)/tip	< 30 m, blendajlı, bükülmüş iletken çifti
Giriş akımı tüketimi	maks. 10 mA
S7-1200 CPU analog girişlerine potansiyometre gerilim çıkışı[1]	Min. 0 V DC ila 10,5 V DC
Potansiyel ayırma	Elektriksel olarak ayrılmamış
Ortam sıcaklığı aralığı	-20 °C ila 60 °C

[1] Potansiyometre modülünün çıkış geriliminin kararlılığı, kullanıcı tarafından 2 katlı bağlantıda sağlanan gerilim girişinin kalitesine bağlıdır ve analog giriş gerilimi olarak değerlendirilebilir.

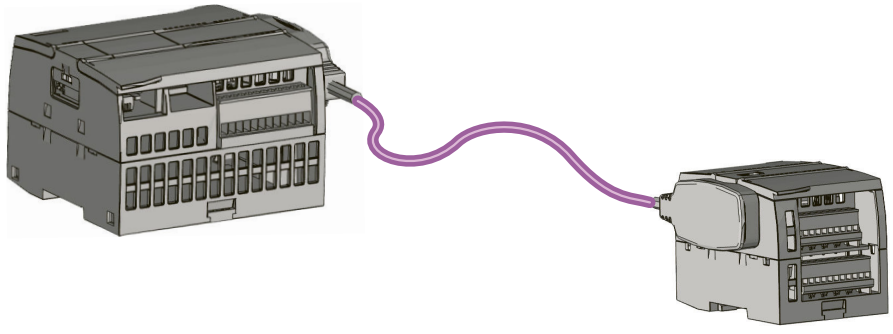


A.21 Genişletme modülleri için bağlantı kablosu

Tablo A-262 Uzatma kablosu

Teknik veriler	
Sipariş numarası	6ES7290-6AA30-0XA0
Hat uzunluğu	2 m
Ağırlık	200 g

S7-1200 uzatma kablosunun takılması ve çıkarılması ile ilgili daha fazla bilgi için "Montaj" (Sayfa 51) alt bölümüne bakın.



A.22 İlişkili ürünler

A.22.1 PM 1207 akım besleme modülü

PM 1207, SIMATIC S7-1200 için bir akım besleme modülüdür. Modül aşağıdaki performans özelliklerini sunar:

- Giriş 120/230 V AC, çıkış 24 V DC/2,5 A

Bu ürünle ilgili daha fazla bilgiyi ve ürün dokümantasyonunu, PM 1207 ürün kataloğunu içeren web sitesinde () bulabilirsiniz.

A.22.2 CSM 1277 Compact Switch modül

CSM1277, Industrial Ethernet için bir Compact Switch modüldür. Kumanda panelleri, programlama cihazları veya diğer kumanda üniteleriyle eşzamanlı iletişim sağlamak üzere S7-1200'ün Ethernet arayüzünü çoğaltmak için kullanılabilir. Modül aşağıdaki performans özelliklerini sunar:

- Industrial Ethernet'e bağlantı için 4 x RJ45 soketi
- Harici 24 V DC beslemeyi yukarıdan bağlamak için 3 kutuplu klemens şeridi
- Industrial Ethernet bağlantılarının teşhisi ve durum göstergesi için LED'ler
- Sipariş numarası 6GK7277-1AA00-0AA0

Bu ürünle ilgili daha fazla bilgiyi ve ürün dokümantasyonunu, CSM 1277 ürün kataloğunu içeren web sitesinde () bulabilirsiniz.

A.22.3 CM CANopen modülü

CM CANopen modülü, SIMATIC S7-1200 PLC ile CANopen çalıştıran bir cihaz arasında takılabilir bir modüldür. CM CANopen modülü, Master veya Slave olarak yapılandırılabilir. İki CM CANopen modules modülü vardır: CANopen modülü (sipariş numarası 021620-B) ve daha yüksek yükler için tasarlanmış CANopen (Ruggedized) modülü (sipariş numarası 021730-B).

CANopen modülü aşağıdaki performans özelliklerini sunar:

- CPU başına 3 modül bağlantı olanağı
- Maks. 16 CANopen Slave düğümü bağlantısı
- Modül başına 256 Bayt giriş ve 256 Bayt çıkış
- 3 LED, modüle yönelik ağ ve G/Ç durumu hakkında teşhis bilgileri sağlar
- CANopen ağ konfigürasyonunun PLC'de saklanmasını destekler
- Modül, TIA Portal'daki konfigürasyon paketinin donanım kataloğuna entegre edilebilir
- CANopen Configuration Studio (teslim kapsamına dahildir) veya başka bir harici CANopen konfigürasyon aracı ile CANopen konfigürasyonu
- CANopen iletişim profilleri CiA 301 Rev. 4.2 ve CiA 302 Rev. 4.1 gerekliliklerini karşılar
- Kullanıcı tanımlı protokolleri işlemek için şeffaf CAN 2.0A'yı destekler
- TIA Portal'da herhangi bir PLC programlaması için önceden hazırlanmış fonksiyon blokları mevcuttur
- CM CANopen modüllerinin teslim kapsamına şunlar dahildir: Alt ağlar için vidalı klemensler ile donatılmış Sub-D soket bağlayıcılar, CM CANopen Configuration Studio CD'si ve USB konfigürasyon kablosu.

Bu ürünle ilgili daha fazla bilgiyi ve ürün dokümantasyonunu, ürün kataloğunu içeren web sitesinde CM CANopen bulabilirsiniz.

A.22.4 İletişim modülü RF120C

RF10C iletişim modülü, Siemens tarafından sunulan RFID sistemlerinin ve kod okuyucuların doğrudan ve basitçe bir S7-1200'e bağlanmasına olanak tanır. Okuyucu, RF120C'ye noktadan noktaya bağlantı üzerinden bağlanır. CPU'nun solundaki bir S7-1200'e üç adede kadar iletişim modülü bağlanabilir. RF120C iletişim modülü TIA Portal üzerinden yapılandırılır. RF120C iletişim modülünün sipariş numarası 6GT2002-0LA00'dır.
Bu ürünle ilgili daha fazla bilgiyi ve ürün dokümantasyonunu, RF120C ürün kataloğunu içeren web sitesinde bulabilirsiniz.

A.22.5 SM 1238 enerji ölçer

SM 1238 enerji ölçer 480 V AC, bir S7-1200 sisteminde kullanım için tasarlanmıştır. Modül, 200 farklı elektriksel ölçüm ve enerji değerini kaydeder. Bu sayede, bir üretim tesisinin münferit bileşenlerinin enerji gereksinimleri hakkında makineye yakın şeffaflık elde edersiniz. Enerji tüketimini ve güç tüketimini saptamak için, SM 1238 enerji ölçer tarafından sağlanan ölçüm değerlerini kullanabilirsiniz.
Bu ürünle ilgili daha fazla bilgiyi, ürün dokümantasyonunu ve teknik verileri, SM 1238 enerji ölçer () ürün kataloğunu içeren web sitesinde bulabilirsiniz.

A.22.6 SIWAREX elektronik tartım sistemi

SIWAREX WP231, WP241 ve WP251

S7-1200'de SIWAREX WP231, WP241 ve WP251 elektronik tartım sistemleri kullanılabilir. Bu modüller, entegre iletişim, kontrol ve gözetleme, teşhis sistemi ve TIA Portal'daki projelendirme araçları gibi modern bir otomasyon sisteminin tüm fonksiyonlarını kullanır.

- SIWAREX WP231 () kalibrasyonlu elektronik tartım sistemi (1 kanal), DMS yük hücreleri/tam köprüleri için (1-4 MV/V), SIMATIC S7-1200 için, RS-485 ve Ethernet arayüzü, entegre G/Ç: 4 DI / 4 DO, 1 AO (0/4...20 MA)
- SIWAREX WP241 () elektronik tartım sistemi, DMS yük hücreleri/tam köprüleri (1-4 M/V) için konveyör bant terazisi (1 kanal) olarak işletim, SIMATIC S7-1200 için, RS-485 ve Ethernet arayüzü, entegre G/Ç: 4 DI / 4 DO, 1 AO (0/4...20 MA)
- SIWAREX WP251 () parti ve dolum prosesleri için elektronik tartım sistemi (1 kanal), DMS yük hücreleri/tam köprüleri için (1-4 MV/V), SIMATIC S7-1200 için, RS-485 ve Ethernet arayüzü, entegre G/Ç: 4 DI / 4 DO, 1 AO (0/4...20 MA),

Güç bilançosunun hesaplanması

B

Tüm S7-1200 CPU'lar, CPU'nun kendisinin yanı sıra genişletme modüllerini ve 24 V DC sensör güç çıkışını besleyen dahili bir gerilim beslemesine sahiptir.

Beş tip genişletme modülü vardır: Sinyal modülleri, iletişim modülleri, sinyal kartları, iletişim kartları ve batarya kartları.

Sinyal modülleri (SM), CPU'nun sağ tarafına bağlanır. CPU'nuza maksimum sayıda sinyal modülü bağlayabilirsiniz.

CPU'nun modeline göre bağlayabileceğiniz maksimum sinyal modülü sayısı için aşağıdaki tabloya bakın:

CPU varyantı	İzin verilen maksimum sinyal modülü sayısı
1211C	0
1212C	2
1214C	8
1215C	8
1217C	8

İletişim modülleri (CM), CPU'nun sol tarafına bağlanır. Her CPU'ya maksimum üç iletişim modülü bağlayabilirsiniz.

Sinyal kartları (SB), iletişim kartları (CB) ve batarya kartları (BB) CPU'nun üst kısmına bağlanır. Her CPU'ya maksimum 1 sinyal kartı, iletişim kartı veya batarya kartı bağlayabilirsiniz.

CPU'nun konfigürasyonunuz için ne kadar güç sağlayabileceğini hesaplamak üzere aşağıdaki bilgileri kullanın.

Her CPU, 5 V ve 24 V DC gerilimi sağlar:

- CPU, bağlanması mümkün olabilecek tüm genişletme modülleri için 5 V DC gerilimi sağlar. 5 V DC güç bilançosu değerini aşarsanız, muhtemelen CPU'nuza maksimum sayıda genişletme modülünü bağlayamazsınız. Güç bilançosu değerine uyulana kadar genişletme modüllerini çıkartmanız gerekir.

NOT

Güç bilançosu değerinin aşılması, sistemin devre dışı kalmasına yol açabilir.

- Her CPU, entegre girişleri ve genişletme modüllerinin röle bobinlerini veya diğer gerekli bileşenleri 24 V DC gerilimle besleyen bir 24 V DC enkoder beslemesine sahiptir. 24 V DC güç ihtiyacı, enkoder beslemesinin gücünü aşarsa, sisteminize harici bir 24 V DC gerilim beslemesi bağlayabilirsiniz. Harici 24 V DC beslemeyi manuel olarak girişlere ve genişletme modüllerindeki röle bobinlerine bağlamanız gerekir.

  ⚠ İKAZ

  Paralel devrelerde ortaya çıkan riskler

  24 V DC enkoder beslemesine harici bir 24 V DC gerilim kaynağını paralel olarak bağlarsanız, uygun çıkış gerilimi oluşturulurken iki gerilim kaynağı birbiriyle etkileşime girebilir.

  Bunun sonucunda, kullanım ömrü kısalabilir veya iki gerilim kaynağından biri anında devre dışı kalabilir. Öngörülemeyen işletim, hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara yol açabilir.

  CPU'nun DC enkoder beslemesi ve harici bir gerilim kaynağı, gerilimi farklı noktalara iletmek zorundadır.

PLC sisteminin 24 V giriş portlarından bazıları birbirine bağlıdır; bu sırada, bir mantıksal referans iletken çok sayıda M klemensini bağlar. CPU'nun 24 V DC besleme girişi, SM'nin röle bobini girişi ve potansiyeli ayrı olmayan bir analog besleme girişi, veri formlarında potansiyeli ayrı olmayan olarak belirtilmişlerse birbirine bağlanan elektrik devrelerine örnektir. Potansiyeli ayrı olmayan tüm M klemensleri aynı harici referans potansiyeline bağlanmalıdır.

⚠ İKAZ

Potansiyeli ayrı olmayan M klemenslerinin farklı referans potansiyellerine bağlanması durumunda ortaya çıkan riskler

Potansiyeli ayrı olmayan M klemenslerini farklı referans potansiyellerine bağlarsanız, bu durum öngörülemeyen bir akım akışına neden olur. Öngörülemeyen akım akışı, hasarlara veya PLC veya cihazın hatalı şekilde çalışmasına ve bu da hayati tehlikeye yol açabilecek yaralanmalara ve/veya maddi hasarlara neden olabilir.

Bir PLC sisteminde, potansiyeli ayrı olmayan tüm M klemenslerini her zaman aynı referans potansiyeline bağlayın.

CPU'ların güç bilançoları ve sinyal modüllerinin güç gereksinimleri ile ilgili bilgileri, teknik verilerde (Sayfa 1160) bulabilirsiniz.

Bir güç bilançosu örneği

Aşağıdaki örnekte; bir CPU 1214C AC/DC/röle, bir SB 1223 2 x 24 V DC giriş / 2 x 24 V DC çıkış, bir CM 1241, üç SM 1223 8 DC giriş / 8 röle çıkışı ve bir SM 1221 8 DC giriş içeren bir konfigürasyona yönelik güç gereksiniminin örnek bir hesaplaması gösterilmektedir. Bu örnekte toplam 48 giriş ve 36 çıkış bulunmaktadır.

NOT

CPU, dahili röle bobinleri için gereken güç gereksinimini halen atamıştır. Dahili röle bobinlerinin güç gereksinimini, güç bilançosu hesaplamasına dahil etmeniz gerekmez.

Bu örnekteki CPU, sinyal modülleri için yeterli 5 V DC gerilim sağlar, ancak enkoder beslemesi tüm girişler ve genişletme röleleri için yeterli 24 V DC gerilimi sağlamaz. Girişler ve çıkışlar 456 mA gerektirir ve CPU yalnızca 400 mA sağlar. Bu yapı, tüm 24 V DC giriş ve çıkışları beslemek için ek olarak minimum 56 mA 24 V DC gerilime gerek duyar.

Tablo B-1 Bir güç bilançosu örneği

CPU'nun güç bilançosu	**5 V DC**	**24 V DC**
CPU 1214C AC/DC/röle	1600 mA	400 mA
Eksi		
Minimum sistem gereksinimleri	**5 V DC**	**24 V DC**
CPU 1214C, 14 giriş	-	14 * 4 mA = 56 mA
1 SB 1223 2 x 24 V DC girişi / 2 x 24 V DC çıkışı	50 mA	2 * 4 mA = 8 mA
1 CM 1241 RS422/485, 5 V	220 mA	
3 SM 1223, 5 V gerilim	3 * 145 mA = 435 mA	-
1 SM 1221, 5 V gerilimi	1 * 105 mA = 105 mA	-
3 SM 1223, her biri 8 giriş	-	3 * 8 * 4 mA = 96 mA
3 SM 1223, her biri 8 röle çıkışı	-	3 * 8 * 11 mA = 264 mA
1 SM 1221, her biri 8 giriş	-	8 * 4 mA = 32 mA
Toplam gereksinim	810 mA	456 mA
Eşittir		
Güç bilançosu	**5 V DC**	**24 V DC**
Toplam güç bilançosu	790 mA	(56 mA)

Güç bilançosu hesaplaması için formül

Bu tablo yardımıyla S7-1200 CPU'nun konfigürasyonunuz için ne kadar güç sağlayabileceğini hesaplayabilirsiniz. Teknik veriler (Sayfa 1160), CPU'ların münferit modellerinin mevcut gücü ve sinyal modüllerinizin güç gereksinimleri hakkında bilgi sağlar.

Tablo B-2 Güç bilançosu hesaplaması

CPU'nun güç bilançosu	**5 V DC**	**24 V DC**
Eksi		
Minimum sistem gereksinimleri	**5 V DC**	**24 V DC**
Toplam gereksinim		
Eşittir		
Güç bilançosu	**5 V DC**	**24 V DC**
Toplam güç bilançosu		

Sipariş bilgileri

C

C.1 CPU modülleri

Tablo C-1 S71200 CPU'lar

CPU varyantları		Sipariş numarası
CPU 1211C	CPU 1211C DC/DC/DC	6ES7211-1AE40-0XB0
	CPU 1211C AC/DC/röle	6ES7211-1BE40-0XB0
	CPU 1211C DC/DC/röle	6ES7211-1HE40-0XB0
CPU 1212C	CPU 1212C DC/DC/DC	6ES7212-1AE40-0XB0
	CPU 1212C AC/DC/röle	6ES7212-1BE40-0XB0
	CPU 1212C DC/DC/röle	6ES7212-1HE40-0XB0
CPU 1214C	CPU 1214C DC/DC/DC	6ES7214-1AG40-0XB0
	CPU 1214C AC/DC/röle	6ES7214-1BG40-0XB0
	CPU 1214C DC/DC/röle	6ES7214-1HG40-0XB0
CPU 1215C	CPU 1215C DC/DC/DC	6ES7215-1AG40-0XB0
	CPU 1215C AC/DC/röle	6ES7215-1BG40-0XB0
	CPU 1215C DC/DC/röle	6ES7215-1HG40-0XB0
CPU 1217C	CPU 1217C DC/DC/DC	6ES7217-1AG40-0XB0

C.2 Sinyal modülleri (SM'ler), sinyal kartları (SB'ler) ve batarya kartları (BB'ler)

Tablo C-2 Sinyal modülü (SM'ler)

Sinyal modülleri		Sipariş numarası
Dijital giriş	SM 1221 8 x 24 V DC girişi (akım çeken/akım besleyen)	6ES7221-1BF32-0XB0
	SM 1221 16 x 24 V DC girişi (akım çeken/akım besleyen)	6ES7221-1BH32-0XB0
Dijital çıkış	SM 1222 8 x 24 V DC çıkışı (kaynak)	6ES7222-1BF32-0XB0
	SM 1222 16 x 24 V DC çıkışı (kaynak)	6ES7222-1BH32-0XB0
	SM 1222 16 x 24 V DC çıkışı (akım çeken)	6ES7222-1BH32-1XB0
	SM 1222 8 x röle çıkışı	6ES7222-1HF32-0XB0
	SM 1222 8 x röle çıkışı (anahtarlamalı)	6ES7222-1XF32-0XB0
	SM 1222 16 x röle çıkışı	6ES7222-1HH32-0XB0
Dijital girişler / çıkış-lar	SB 1223 8 x 24 V DC girişi (akım çeken/akım besleyen) / 8 x 24 V DC çıkışı (akım besleyen)	6ES7223-1BH32-0XB0
	SM 1223 16 x 24 V DC girişi (akım çeken/akım besleyen) / 16 x 24 V DC çıkışı (akım besleyen)	6ES7223-1BL32-0XB0
	SM 1223 16 x 24 V DC girişi / 16 x 24 V DC çıkışı (akım çeken)	6ES7223-1BL32-1XB0

Sinyal modülleri		Sipariş numarası
Dijital girişler / çıkışlar	SM 1223 8 x 24 V DC girişi (akım çeken/akım besleyen) / 8 x röle çıkışı	6ES7223-1PH32-0XB0
	SM 1223 16 x 24 V DC girişi (akım çeken/akım besleyen) / 16 x röle çıkışı	6ES7223-1PL32-0XB0
	SM 1223 8 x 120/230 V AC girişi (akım çeken/akım besleyen) / 8 x röle çıkışı	6ES7223-1QH32-0XB0
Analog giriş	SM 1231 4 x analog giriş	6ES7231-4HD32-0XB0
	SM 1231 8 x analog giriş	6ES7231-4HF32-0XB0
	SM 1231 4 x analog giriş x 16 Bit (son derece fonksiyonel)	6ES7231-5ND32-0XB0
	SM 1238 enerji ölçer 480 V AC	6ES7238-5XA32-0XB0
Analog çıkış	SM 1232 2 x analog çıkış	6ES7232-4HB32-0XB0
	SM 1232 4 x analog çıkış	6ES7232-4HD32-0XB0
Analog girişler/çıkışlar	SM 1234 4 x analog giriş / 2 x analog çıkış	6ES7234-4HE32-0XB0
RTD ve termo eleman	SM 1231 TC 4 x 16 Bit	6ES7231-5QD32-0XB0
	SM 1231 TC 8 x 16 Bit	6ES7231-5QF32-0XB0
	SM 1231 RTD 4 x 16 Bit	6ES7231-5PD32-0XB0
	SM 1231 RTD 8 x 16 Bit	6ES7231-5PF32-0XB0
Teknoloji modülleri	SM 1278 4xIO-Link-Master	6ES7278-4BD32-0XB0
	SIWAREX WP231 kalibrasyonlu elektronik tartım sistemi (1 kanal), DMS yük hücreleri/tam köprüleri için (1-4 MV/V), SIMATIC S7-1200 için, RS-485 ve Ethernet arayüzü, entegre G/Ç: 4 DI / 4 DO, 1 AO (0/4...20 MA)	7MH4960-2AA01
	SIWAREX WP241 elektronik tartım sistemi, DMS yük hücreleri/tam köprüleri (1-4 M/V) için konveyör bant terazisi (1 kanal) olarak işletim, SIMATIC S7-1200 için, RS-485 ve Ethernet arayüzü, entegre G/Ç: 4 DI / 4 DO, 1 AO (0/4...20 MA)	7MH4960-4AA01
	SIWAREX WP251 parti ve dolum prosesleri için elektronik tartım sistemi (1 kanal), DMS yük hücreleri/tam köprüleri için (1-4 MV/V), SIMATIC S7-1200 için, RS-485 ve Ethernet arayüzü, entegre G/Ç: 4 DI / 4 DO, 1 AO (0/4...20 MA),	7MH4960-6AA01

Tablo C-3 Sinyal kartları (SB'ler) ve batarya kartları (BB'ler)

Sinyal ve batarya kartları		Sipariş numarası
Dijital giriş	SB 1221 200 kHz, 4 x 24-V-DC girişi (akım besleyen)	6ES7221-3BD30-0XB0
	SB 1221 200 kHz, 4 x 5 V DC girişi (akım besleyen)	6ES7221-3AD30-0XB0
Dijital çıkış	SB 1222 200 kHz, 4 x 24-V-DC çıkışı (akım çeken/akım besleyen)	6ES7222-1BD30-0XB0
	SB 1222 200 kHz, 4 x 5-V-DC çıkışı (akım çeken/akım besleyen)	6ES7222-1AD30-0XB0
Dijital girişler / çıkışlar	SB 1223 2 x 24 V DC girişi (akım çeken) / 2 x 24 V DC çıkışı (akım besleyen)	6ES7223-0BD30-0XB0
	SB 1223 200 kHz 2 x 24 V DC girişi (akım besleyen) / 2 x 24 V DC çıkışı (akım çeken/akım besleyen)	6ES7223-3BD30-0XB0
	SB 1223 200 kHz 2 x 5 V DC girişi (akım besleyen) / 2 x 5 V DC çıkışı (akım çeken/akım besleyen)	6ES7223-3AD30-0XB0
Analog	SB 1232 1 analog çıkış	6ES7232-4HA30-0XB0
	SB 1231 1 analog giriş	6ES7231-4HA30-0XB0

Sinyal ve batarya kartları		Sipariş numarası
Analog	SB 1231 1 analog giriş termo eleman	6ES7231-5QA30-0XB0
	SB 1231 1 analog giriş RTD	6ES7231-5PA30-0XB0
Batarya	BB 1297 batarya kartı (batarya tipi CR1025 dahil değildir)	6ES7297-0AX30-0XA0

C.3 İletişim

Tablo C-4 İletişim modülü (CM)

İletişim modülü (CM)			Sipariş numarası
RS232, RS422 ve RS485	CM 1241 RS232	RS232	6ES7241-1AH32-0XB0
	CM 1241 RS422/485	RS422/485	6ES7241-1CH32-0XB0
PROFIBUS	CM 1243-5	PROFIBUS Master	6GK7243-5DX30-0XE0
	CM 1242-5	PROFIBUS Slave	6GK7242-5DX30-0XE0
AS-i Master	CM 1243-2	AS-i Master	3RK7243-2AA30-0XB0

Tablo C-5 İletişim kartı (CB)

İletişim kartı (CB)			Sipariş numarası
RS485	CB 1241 RS485	RS485	6ES7241-1CH30-1XB0

Tablo C-6 İletişim işlemcisi (CP)

CP	Arayüz	Sipariş numarası
CP 1242-7 GPRS V2	GPRS	6GK7242-7KX31-0XE0
CP 1243-7 LTE-US	LTE	6GK7243-7SX30-0XE0
CP 1243-7 LTE-EU	LTE	6GK7243-7KX30-0XE0
CP 1243-1	IE arayüz	6GK7243-1BX30-0XE0
CP 1243-8 IRC	IE ve seri arayüz	6GK7243-8RX30-0XE0

Tablo C-7 TeleService

TS adaptörü	Sipariş numarası
TS modülü RS232	6ES7972-0MS00-0XA0
TS modülü modem	6ES7972-0MM00-0XA0

Tablo C-8 Aksesuar

Aksesuar			Sipariş numarası
Anten	ANT794-4MR	GSM/GPRS anteni	6NH9860-1AA00
	ANT794-3M	Yassı anten	6NH9870-1AA00

Tablo C-9 Soket bağlayıcı

Soket bağlayıcının türü		Sipariş numarası
RS485	Vidalı klemens bağlantısı, 35 derecelik kablo çıkışı	6ES7972-0BA42-0XA0
	FastConnect bağlantısı, 35 derecelik kablo çıkışı	6ES7972-0BA60-0XA0

C.4 Hata güvenli CPU'lar ve sinyal modülleri

Tablo C-10 Hata güvenli CPU'lar

Hata güvenli CPU modelleri		Sipariş numarası
CPU 1212FC	CPU 1212FC DC/DC/DC	6ES7212-1AF40-0XB0
	CPU 1212FC DC/DC/röle	6ES7212-1HF40-0XB0
CPU 1214FC	CPU 1214FC DC/DC/DC	6ES7214-1AF40-0XB0
	CPU 1214FC DC/DC/röle	6ES7214-1HF40-0XB0
CPU 1215FC	CPU 1215FC DC/DC/DC	6ES7215-1AF40-0XB0
	CPU 1215FC DC/DC/röle	6ES7215-1HF40-0XB0

Tablo C-11 Hata güvenli sinyal modülleri

Fonksiyonel güvenlikli sinyal modülleri		Sipariş numarası
Dijital giriş	SM 1226 F-DI 16 x 24 V DC	6ES7226-6BA32-0XB0
Dijital çıkış	SM 1226 F-DQ 4 x 24 V DC	6ES7226-6DA32-0XB0
	SM 1226 F-DQ 2 x Relay	6ES7226-6RA32-0XB0

C.5 Diğer modüller

Tablo C-12 İlişkili ürünler

Açıklama		Sipariş numarası
Güç kaynağı	PM 1207 güç kaynağı	6EP1332-1SH71
Ethernet Switch	CSM 1277 Ethernet Switch - 4 port	6GK7277-1AA10-0AA0
CM CANopen	SIMATIC S7-1200 için CANopen	021620-B
	SIMATIC S7-1200 için CANopen (Ruggedized)	021730-B
RF120C	İletişim modülü RF120C	6GT2002-0LA00

C.6 Memory card'lar

Tablo C-13 Memory card'lar

SIMATIC memory card'lar	Sipariş numarası
SIMATIC MC 32 GB	6ES7954-8LT03-0AA0
SIMATIC MC 2 GB	6ES7954-8LP02-0AA0
SIMATIC MC 256 MB	6ES7954-8LL03-0AA0
SIMATIC MC 24 MB	6ES7954-8LF03-0AA0
SIMATIC MC 12 MB	6ES7954-8LE03-0AA0
SIMATIC MC 4 MB	6ES7954-8LC03-0AA0

C.7 Yedek parçalar ve diğer donanımlar

Tablo C-14 Uzatma kablosu, simülatör ve uç tutucular

Açıklama		**Sipariş numarası**
Genişletme modülleri için bağlantı kablosu	Genişletme modülleri için bağlantı kablosu, 2 m	6ES7290-6AA30-0XA0
G/Ç simülatörü	Simülatör (1211C/1212C - 8 G/Ç)	6ES7274-1XF30-0XA0
	Simülatör (1214C/1215C - 14 G/Ç)	6ES7274-1XH30-0XA0
	Simülatör CPU 1217C	6ES7274-1XK30-0XA0
Potansiyometre modülü	S7-1200 potansiyometre modülü	6ES7274-1XA30-0XA0
Ethernet çekme koruması	Çekme koruması bir RJ45 bağlantısı, 10/100 Mbit/sn	6ES7290-3AA30-0XA0
	Çekme koruması iki RJ45 bağlantısı, 10/100 Mbit/sn	6ES7290-3AB30-0XA0
Yedek kapatma klapesi	CPU 1211C/1212C	6ES7291-1AA30-0XA0
	CPU 1214C	6ES7291-1AB30-0XA0
	CPU 1215C	6ES7291-1AC30-0XA0
	CPU 1217C	6ES7291-1AD30-0XA0
	Sinyal modülü (SM), 45 mm	6ES7291-1BA30-0XA0
	Sinyal modülü (SM), 70 mm	6ES7291-1BB30-0XA0
	İletişim modülü (6ES72xx-xxx32-0XB0 ve 6ES72xx-xxx30-0XB0 modülleri ile kullanım için)	6ES7291-1CC30-0XA0
Uç tutucu	Uç tutucu termoplast, 10 mm	8WA1808
	Uç tutucu çelik, 10,3 mm	8WA1805

Klemens bloğunun değişimi

Modülünüz için doğru klemens bloğunu kullanmanız önemlidir. Değiştirmek istediğiniz doğru klemens bloğunu belirlemek için aşağıdaki tablolara ve modülünüzün teknik verilerine bakın.

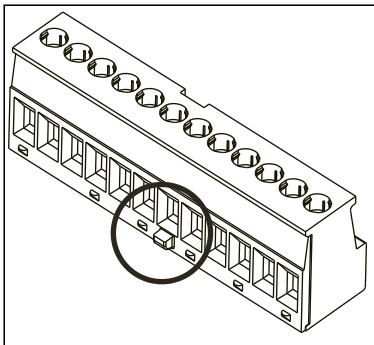
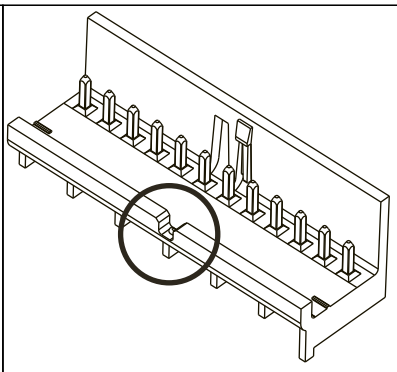
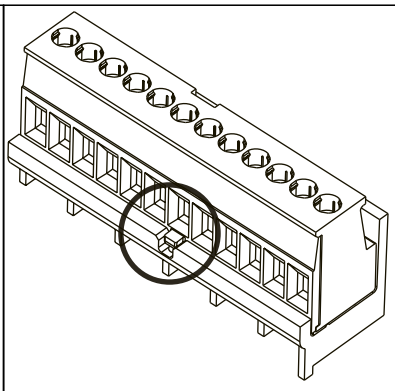
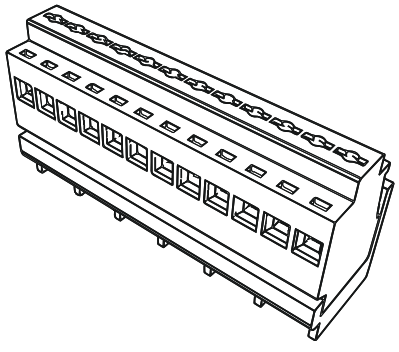
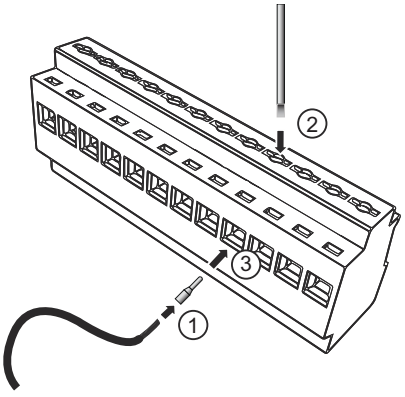
NOT

Çıkarılabilir kodlanmış klemens blokları

Güvenliği ve usulüne uygun şekilde işletimi sağlamak için PLC'ler her zaman doğru şekilde kablolanmalıdır.

CPU veya SM'nizdeki bir klemens bloğunu değiştirirken, modülünüz için doğru klemens bloğunu ve kablaj kaynağını kullanmanız önemlidir.

Kodlama sayesinde, yüksek gerilime bağlı bir klemens bloğunu yanlışlıkla alçak gerilimli bir modüle veya bir özel gerilime bağlı bir klemens bloğunu normal gerilimli bir modüle takmanız önlenir. Bazı klemens blokları sol, sağ veya orta olarak kodlanmıştır.

		
çıkarılabilir klemens bloğu (kodlanmış örnek gösterilmiştir)	Cihazda kodlama	Cihaz üzerindeki kodlama, yalnızca ilgili çıkarılabilir klemens bloğuyla eşleşmektedir
		
Push-in klemens bloğu	Aşağıdaki yöntemi izleyin: 1. Teli ve krimp bağlantısını yerleştirin. 2. Yuvayı açmak için küçük bir tornavida kullanarak dili aşağı bastırın. 3. Krimp bağlantısını, klemens bloğundaki açılmış yuvaya yerleştirin.	

Tablo C-15 S7-1200 CPU V4'ten itibaren - yedek klemens blokları

V4.0 üstü bir S7-1200 CPU'nuz varsa (sipariş numarası)	**Bu durumda bu yedek klemens bloklarını kullanın (4/paket)**		
	Klemens bloğunun (vidalanabilir) sipariş numarası	**Sipariş numarası eşdeğer Push-in klemens bloğu**	**Klemens bloklarının açıklaması**
CPU 1211C DC/DC/DC (6ES7211-1AE40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH30-0XA0	6ES7292-2AH30-0XA0	8 kutuplu, kalaylanmış
	6ES7292-1AP30-0XA0	6ES7292-2AP30-0XA0	14 kutuplu, kalaylanmış
CPU 1211C DC/DC/röle (6ES7211-1HE40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH40-0XA0	6ES7292-2AH40-0XA0	8 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AP30-0XA0	6ES7292-2AP30-0XA0	14 kutuplu, kalaylanmış
CPU 1211C AC/DC/röle (6ES7211-1BE40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH40-0XA0	6ES7292-2AH40-0XA0	8 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AP40-0XA0	6ES7292-2AP40-0XA0	14 kutuplu, kalaylanmış, kodlanmış
CPU 1212C DC/DC/DC (6ES7212-1AE40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH30-0XA0	6ES7292-2AH30-0XA0	8 kutuplu, kalaylanmış
	6ES7292-1AP30-0XA0	6ES7292-2AP30-0XA0	14 kutuplu, kalaylanmış
CPU 1212C DC/DC/röle (6ES7212-1HE40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH40-0XA0	6ES7292-2AH40-0XA0	8 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AP30-0XA0	6ES7292-2AP30-0XA0	14 kutuplu, kalaylanmış
CPU 1212C AC/DC/röle (6ES7212-1BE40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH40-0XA0	6ES7292-2AH40-0XA0	8 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AP40-0XA0	6ES7292-2AP40-0XA0	14 kutuplu, kalaylanmış, kodlanmış
CPU 1214C DC/DC/DC (6ES7214-1AG40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AM30-0XA0	6ES7292-2AM30-0XA0	12 kutuplu, kalaylanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1214C DC/DC/röle (6ES7214-1HG40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AM40-0XA0	6ES7292-2AM40-0XA0	12 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1214C AC/DC/röle (6ES7214-1BG40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AM40-0XA0	6ES7292-2AM40-0XA0	12 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AV40-0XA0	6ES7292-2AV40-0XA0	20 kutuplu, kalaylanmış, kodlanmış
CPU 1215C DC/DC/DC (6ES7215-1AG40-0XB0)	6ES7292-1BF30-0XB0	6ES7292-2BF30-0XB0	6 kutuplu, altın kaplamalı
	6ES7292-1AM30-0XA0	6ES7292-2AM30-0XA0	12 kutuplu, kalaylanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1215C DC/DC/röle (6ES7215-1HG40-0XB0)	6ES7292-1BF30-0XB0	6ES7292-2BF30-0XB0	6 kutuplu, altın kaplamalı
	6ES7292-1AM40-0XA0	6ES7292-2AM40-0XA0	12 kutuplu, kalaylanmış, kodlanmış

V4.0 üstü bir S7-1200 CPU'nuz varsa (sipariş numarası)	Bu durumda bu yedek klemens bloklarını kullanın (4/paket)		
	Klemens bloğunun (vidalanabilir) sipariş numarası	Sipariş numarası eşdeğer Push-in klemens bloğu	Klemens bloklarının açıklaması
CPU 1215C DC/DC/röle (6ES7215-1HG40-0XB0)	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1215C AC/DC/röle (6ES7215-1BG40-0XB0)	6ES7292-1BF30-0XB0	6ES7292-2BF30-0XB0	6 kutuplu, altın kaplamalı
	6ES7292-1AM40-0XA0	6ES7292-2AM40-0XA0	12 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AV40-0XA0	6ES7292-2AV40-0XA0	20 kutuplu, kalaylanmış, kodlanmış
CPU 1217C DC/DC/DC (6ES7217-1AG40-0XB0)	6ES7292-1BF30-0XB0	6ES7292-2BF30-0XB0	6 kutuplu, altın kaplamalı
	6ES7292-1AK30-0XA0	6ES7292-2AK30-0XA0	10 kutuplu, kalaylanmış
	6ES7292-1AR30-0XA0	6ES7292-2AR30-0XA0	16 kutuplu, kalaylanmış
	6ES7292-1AT30-0XA0	6ES7292-2AT30-0XA0	18 kutuplu, kalaylanmış

Tablo C-16 S7-1200 SM'ler V3.2'den itibaren - yedek klemens blokları

S7-1200 SM'leriniz V3.2 üstü ise (sipariş numarası)	Bu durumda bu yedek klemens bloklarını kullanın (4/paket)		
	Klemens bloğunun (vidalanabilir) sipariş numarası	Sipariş numarası eşdeğer Push-in klemens bloğu	Klemens bloklarının açıklaması
SM 1221 DI 8 × DC (6ES7221-1BF32-0XB0)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 8 × DC (6ES7222-1BF32-0XB0)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 8 × röle (6ES7222-1HF32-0XB0)	6ES7292-1AG40-0XA1	6ES7292-2AG40-0XA1	7 kutuplu, kalaylanmış, solda kodlanmış
SM 1238 enerji ölçer 480 V AC (6ES7238-5XA32-0XB0), gerilim girişi için (üstte)	6ES7292-1AG40-0XA2	6ES7292-2AG40-0XA2	7 kutuplu, kalaylanmış, ortada kodlanmış
SM 1238 enerji ölçer 480 V AC (6ES7238-5XA32-0XB0) akım girişi için (altta)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1231 AI 4 × 13 Bit (6ES7231-4HD32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1232 AQ 2 × 14 Bit (6ES7232-4HB32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1231 AI 4 × TC (6ES7231-5QD32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1231 AI 4 × 16 Bit (6ES7231-5ND32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1221 DI 16 × DC (6ES7221-1BH32-0XB0)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 16 × DC (6ES7222-1BH32-0XB0)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 16 × röle (6ES7222-1HH32-0XB0)	6ES7292-1AG40-0XA0	6ES7292-2AG40-0XA0	7 kutuplu, kalaylanmış, sağda kodlanmış
SM 1223 DI 8 × DC/DQ 8 × DC (6ES7223-1BH32-0XB0)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1223 8 × DC/8 × röle	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış

S7-1200 SM'leriniz V3.2 üstü ise (sipariş numarası)	**Bu durumda bu yedek klemens bloklarını kullanın (4/paket)**		
	Klemens bloğunun (vidalanabilir) sipariş numarası	**Sipariş numarası eşdeğer Push-in klemens bloğu**	**Klemens bloklarının açıklaması**
(6ES7223-1PH32-0XB0)	6ES7292-1AG40-0XA0	6ES7292-2AG40-0XA0	7 kutuplu, kalaylanmış, sağda kodlanmış
SM 1223 8 × AC/8 × röle (6ES7223-1QH32-0XB0)	6ES7292-1AG40-0XA0	6ES7292-2AG40-0XA0	7 kutuplu, kalaylanmış, sağda kodlanmış
SM 1234 AI 4 / AQ 2 (6ES7234-4HE32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1231 AI 8 × 13 Bit (6ES7231-4HF32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1232 AQ 4 × 14 Bit (6ES7232-4HD32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1231 AI 4 × RTD (6ES7231-5PD32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1231 AI 8 × TC (6ES7231-5QF32-0XB0)	6ES7292-1BG30-0XA0	6ES7292-2BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1278 IO LINK (6ES7278-4BD32 0XB0)	6ES7292-1AG30-0XA0	6ES7292-2AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 8 × röle (anahtarlamalı) (6ES7222-1XF32-0XB0)	6ES7292-1AL40-0XA0	6ES7292-2AL40-0XA0	11 kutuplu, kalaylanmış, kodlanmış
SM 1223 DI 16 × DC/DQ 16 × DC (6ES7223-1BL32-0XB0)	6ES7292-1AL30-0XA0	6ES7292-2AL30-0XA0	11 kutuplu, kalaylanmış
SM 1223 DI 16 × DC/DQ 16 × röle (6ES7223-1PL32-0XB0)	6ES7292-1AL30-0XA0	6ES7292-2AL30-0XA0	11 kutuplu, kalaylanmış
	6ES7292-1AL40-0XA0	6ES7292-2AL40-0XA0	11 kutuplu, kalaylanmış, kodlanmış
SM 1231 AI 8 × RTD (6ES7231-5PF32-0XB0)	6ES7292-1BL30-0XA0	6ES7292-2BL30-0XA0	11 kutuplu, altın kaplamalı

Tablo C-17 S7-1200 SB'ler, CB'ler ve BB'ler - yedek klemens blokları

Bir S7-1200 SB, CB veya BB'niz varsa (sipariş numarası)	**Bu durumda bu yedek klemens bloklarını kullanın (4/paket)**	
	Klemens bloğunun (vidalanabilir) sipariş numarası	**Klemens bloklarının açıklaması**
SB 1221 DI 4 × 5 V DC (6ES7221-3AD30-0XB0)	6ES7292-1BF30-0XA0	6 kutuplu
SB 1221 DI 4 × 5 V DC (6ES7221-3AD30-0XB0)		
SB 1221 DI 4 × 24 V DC (6ES7221-3BD30-0XB0)		
SB 1222 DO 4 × 5 V DC (6ES7222-1AD30-0XB0)		
SB 1222 DO 4 × 24 V DC (6ES7222-1BD30-0XB0)		
SB 1223 DI 2 × 24 V DC / DQ 2 × 24 V DC (6ES7223-0BD30-0XB0)		
SB 1223 DI 2 × 5 V DC / DQ 2 × 5 V DC (6ES7223-3AD30-0XB0)		
SB 1223 DI 2 × 24 V DC / DQ 2 × 24 V DC (6ES7223-3BD30-0XB0)		
SB 1231 AI 1 × 12 Bit (6ES7231-4HA30-0XB0)		

Bir S7-1200 SB, CB veya BB'niz varsa (sipariş numarası)	Bu durumda bu yedek klemens bloklarını kullanın (4/paket)	
	Klemens bloğunun (vidalanabilir) sipariş numarası	Klemens bloklarının açıklaması
SB 1231 AI 1 × RTD (6ES7231-5PA30-0XB0)	6ES7292-1BF30-0XA0	6 kutuplu
SB 1231 AI 1 × TC (6ES7231-5QA30-0XB0)		
SB 1232 AO 1 × 12 Bit (6ES7232-4HA30-0XB0)		
CB 1231 RS-485 (6ES7241-1CH30-1XB0)		
BB 1297 batarya (6ES7297-0AX30-0XA0)		

Tablo C-18 Hata güvenli CPU'lar - yedek klemens blokları

Bir hata güvenli CPU'nuz varsa (sipariş numarası)	Bu durumda bu yedek klemens bloklarını kullanın (4/paket)		
	Sipariş numarası klemens bloğu	Eşdeğer Push-in klemens bloğu sipariş numarası	Klemens bloklarının açıklaması
CPU 1212FC DC/DC/DC (6ES7212-1AF40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH30-0XA0	6ES7292-2AH30-0XA0	12 kutuplu, kalaylanmış
	6ES7292-1AP30-0XA0	6ES7292-2AP30-0XA0	20 kutuplu, kalaylanmış
CPU 1212FC DC/DC/röle (6ES7212-1HF40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AH40-0XA0	6ES7292-2AH40-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AP30-0XA0	6ES7292-2AP30-0XA0	12 kutuplu, kalaylanmış, kodlanmış
CPU 1214FC DC/DC/DC (6ES7214-1AF40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AM30-0XA0	6ES7292-2AM30-0XA0	12 kutuplu, kalaylanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1214FC DC/DC/röle (6ES7214-1HF40-0XB0)	6ES7292-1BC30-0XA0	6ES7292-2BC30-0XA0	3 kutuplu, altın kaplamalı
	6ES7292-1AM40-0XA0	6ES7292-2AM40-0XA0	12 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1215FC DC/DC/DC (6ES7215-1AF40 0XB0)	6ES7292-1BF30-0XB0	6ES7292-2BF30-0XB0	6 kutuplu, altın kaplamalı
	6ES7292-1AM30-0XA0	6ES7292-2AM30-0XA0	12 kutuplu, kalaylanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış
CPU 1215FC DC/DC/röle (6ES7215-1HF40 0XB0)	6ES7292-1BF30-0XB0	6ES7292-2BF30-0XB0	6 kutuplu, altın kaplamalı
	6ES7292-1AM40-0XA0	6ES7292-2AM40-0XA0	2 kutuplu, kalaylanmış, kodlanmış
	6ES7292-1AV30-0XA0	6ES7292-2AV30-0XA0	20 kutuplu, kalaylanmış

Tablo C-19 Hata güvenli sinyal modülleri - yedek klemens blokları

Bir hata güvenli sinyal modülünüz varsa (sipariş numarası)	**Bu durumda bu yedek klemens bloklarını kullanın (4/paket)**		
	Sipariş numarası klemens bloğu	**Eşdeğer Push-in klemens bloğu sipariş numarası**	**Klemens bloklarının açıklaması**
SM 1226 F-DI (6ES7226-6BA32-0XB0)	6ES7292-1AL30-0XA0	6ES7292-2AL30-0XA0	11 kutuplu, kalaylanmış
SM 1226 F-DQ (6ES7226-6DA32-0XB0)	6ES7292-1AL30-0XA0	6ES7292-2AL30-0XA0	11 kutuplu, kalaylanmış
SM 1226 F-röle (6ES7226-6RA32-0XB0)	6ES7292-1AL40-0XA0	6ES7292-2AL40-0XA0	11 kutuplu, kalaylanmış, kodlanmış

C.8 Programlama yazılımı

Tablo C-20 Programlama yazılımı

SIMATIC yazılımı		**Sipariş numarası**
Programlama yazılımı	STEP 7 Basic V18	6ES7822-0AA08-0YA5
	STEP 7 Professional V18	6ES7822-1AA08-0YA5

C.9 OPC UA lisansları

Tablo C-21 S7-1200 için OPC UA lisansları

OPC UA lisansları		**Sipariş numarası**
SIMATIC lisansları	SIMATIC OPC UA S7-1200 Basic DVD	6ES7823-0BA00-2BA0
	SIMATIC OPC UA S7-1200 Basic DL	6ES7823-0BE00-2BA0

Cihaz değişimi ve yedek parça uyumluluğu

D

D.1 Gizli PLC konfigürasyon verilerini koruyan bir CPU'nun değişimi

Gizli PLC konfigürasyon verilerini korumaya yönelik şifrelerin atanması, parça değişimini de etkiler.

Parça değişimi kuralları

Parçaları değiştirirken aşağıdaki kuralları dikkate alın:

TIA Portal'da yedek CPU'nun yapılandırılması

- Mümkünse, yedek CPU'da olduğu gibi, gizli PLC konfigürasyon verilerini korumaya yönelik proje konfigürasyonu ve yapılandırılmış şifresi olmayan bir CPU kullanın.
  Bunun avantajı: Böylece daha fazla hazırlık yapmaksızın projeyi yedek CPU'ya yükleyebilirsiniz.
- Yedek CPU halen yapılandırılmışsa, CPU'yu fabrika ayarlarına sıfırlamanız (Sayfa 1122) ve şu opsiyonları seçmeniz gerekir:
  - Gizli PLC konfigürasyon verilerini koruma şifresini silin
  - CPU'nun, bir memory card'ı varsa, Memory card biçimlendirin

Konfigürasyon verileri yüklü olan SIMATIC memory card ile bir CPU'nun değişimi

- Projenizdeki bir CPU'ya gizli PLC konfigürasyon verilerini korumak için bir **şifre atamadıysanız**, değiştirilecek CPU'nun memory card'ını başka bir işlem yapmaksızın henüz kullanılmayan yeni bir CPU'ya takabilirsiniz.
- Değiştirilecek CPU halen bir şifre ile yapılandırılmış ise önce "Gizli PLC konfigürasyon verilerini koruma şifresini sil (Sayfa 1122)" opsiyonunu kullanarak bu CPU'yu fabrika ayarlarına sıfırlamanız gerekir.
- Gizli PLC konfigürasyon verilerini korumak için aynı şifreyi bir CPU grubuna atadıysanız, grup şifresini TIA Portal'daki cihaz konfigürasyonunda (Sayfa 154) yedek CPU'ya da atayabilirsiniz. Bu durumda, örneğin güncel projeyi içeren bir memory card'ı CPU'ya takabilir ve başka şifre düzenlemesi yapmaksızın CPU'yu işletime alabilirsiniz.
- Projenizdeki her CPU'ya farklı bir şifre atarsanız, çevrimiçi olun (Sayfa 1117) ve çevrimiçi ve teşhis araçlarını kullanarak yedek CPU için gizli verileri korumaya yönelik şifreyi belirleyin.Çevrimiçi fonksiyonlarda, "Gizli PLC konfigürasyon verilerinin korunması için şifre belirle (Sayfa 1121)" opsiyonunu seçin.

D.2 Bir CPU V3.0'ın CPU V4.x ile değişimi

Bir CPU V3.0'ı CPU V4.x'e yükseltmek için CPU donanımını değiştirmeniz gerekir. Firmware güncellemesi ile bir CPU V3.0, CPU V4.x'e yükseltilemez.

Dolayısıyla, STEP 7 projenizde CPU V3.0'ınızı bir CPU V4.x ile değiştirebilir (Sayfa 144) ve CPU V3.0 için geliştirilmiş halen mevcut olan STEP 7 projenizde kullanabilirsiniz.

Bir CPU V3.0'ı bir CPU V4.x ile değiştirirseniz, bu sırada sinyal ve iletişim modülleriniz için Firmware güncellemelerini (Sayfa 124) arayabilir ve bunları yükleyebilirsiniz.

NOT

STEP 7'de V4.x'ten V3.0'a cihaz değişimi mümkün değildir

Bir CPU V3.0'ı bir CPU V4.x ile değiştirebilirsiniz, ancak konfigürasyonu yükledikten sonra bir CPU V4.x'i bir CPU V3.0 ile değiştiremezsiniz. Mevcut STEP 7 projenizi bir CPU V3.0 için görüntülemek veya başka amaçlarla kullanmak istiyorsanız, cihaz değişiminden önce projenizin bir arşivini oluşturun.

Değiştirilen cihaz konfigürasyonunu henüz yüklemediyseniz, geri alabilirsiniz. Yükleme işleminden sonra, V3.0'dan V4.x'e cihaz değişimini artık geri alamazsınız.

İki CPU sürümü arasındaki bazı konfigürasyon ve işletim değişikliklerini dikkate alın.

Bir STEP 7 projesinin yükseltilmesi

V13'e kadar olan bir STEP 7 projesi ile çalışmak için, önce bu projeyi STEP 7 V13 SP1 veya SP2'ye yükseltmelisiniz. Bunun ardından STEP 7 V13 SPx projesini güncel STEP 7 sürümüne yükseltebilirsiniz.

⚠ İKAZ

STEP 7'nin eski sürümlerinden program lojiğini düzenlerken ve yürütürken karşılaşılan riskler

Program lojiğini kopyalayarak sürümü V13'e kadar olan STEP 7'deki bir projeyi güncel sürüme yükseltemezsiniz. STEP 7 projenizin yükseltme işlemini, yukarıda açıklandığı şekilde gerçekleştirmeniz gerekir. Daha eski bir sürümden daha yeni bir sürüme kopyaladığınız STEP 7 program lojiğini yürütmek, öngörülemeyen program davranışlarına neden olabilir ve ölümcül veya ciddi yaralanmalara yol açabilir.

Organizasyon blokları

OB yürütmesinin kesintiye uğrayabilir olup olmamasını (Sayfa 80) yapılandırabilirsiniz. V3.0 sürümü CPU'ların projelerinde STEP 7, standart olarak tüm OB'ler için kesintisiz yürütmeyi ayarlar.

STEP 7, tüm OB öncelikleri (Sayfa 80) için V3.0 CPU'nun STEP 7 projesindeki değerlerini ayarlar.

Aşağıda gösterildiği gibi, gerektiğinde kesintiye uğrama ve öncelik ayarlarını değiştirebilirsiniz.

Teşhis hatası OB'sinin (Sayfa 72) yol alma bilgisi, hiçbir teşhis olayı beklemediğinde bir bütün olarak alt modülü temel alır.

CPU şifre koruması

STEP 7, CPU V4.x için olan şifre koruma kademesini (Sayfa 156) CPU V3.0'da ayarlanan aynı seviyeye getirir ve sürüm 3.0'ın şifresini CPU V4.x'in "Tam erişim (koruma yok)" şifresine atar:

Sürüm 3.0'daki koruma kademesi	Sürüm V4.x'deki koruma kademesi
Koruma yok	Tam erişim (koruma yok)
Yazma koruması	Okuma erişimi
Okuma/yazma koruması	HMI erişimi

V4.x koruma kademesi "Erişim yok (Tam koruma)" kademesinin sürüm 3.0'da mevcut olmadığını dikkate alın.

Web sunucusu

V3.0 projenizde kullanıcı tanımlı web siteleri kullanıyorsanız, projenizi yükseltmeye başlamadan önce, bu siteleri projenizin kurulum dizinindeki "UserFiles\Webserver" alt dizinine saklayın. Kullanıcı tanımlı web siteleri bu dizinde saklanıyorsa, STEP 7 projesini sakladığınızda kullanıcı tanımlı web siteleri de birlikte saklanacaktır.

Bir CPU V3.0'ı bir CPU V4.x ile değiştirirseniz, Web sunucusunu ve HTTPS ayarlarını etkinleştirmek için olan Web sunucusu proje ayarlarınız (Sayfa 798), sürüm 3.0'dakiler ile aynı olur. Bunun ardından, Web sunucusuyla çalışmak için gereken kullanıcıları, yetkileri, şifreleri (Sayfa 801) ve dilleri (Sayfa 798) değiştirebilirsiniz. Kullanıcıları ek yetkiler ile yapılandırmazsanız, standart web sitelerinde (Sayfa 808) görüntülenenler ile sınırlı kalırsınız. S7-1200 CPU V4.x, o ana kadar yapılandırılmış "admin" kullanıcısını ve ilişkili şifreyi desteklemez.

S7-1200 V3.0 veri protokolü için Web sunucusu sayfası daha önce "İndir ve Sil" fonksiyonlarını sunuyordu. Veri protokollerine eriştiğiniz sürüm 4.x Web sunucusunun dosya tarayıcısı sayfası (Sayfa 838) artık bu fonksiyonu sunmamaktadır. Bunun yerine, Web sunucusu veri protokolü dosyalarını indirme, yeniden adlandırma ve silme olanaklarını sunmaktadır.

Aktarım kartının uyumsuzluğu

Bir sürüm 3.0 programını, bir sürüm 4.x CPU'ya aktarmak için bir sürüm 3.0 aktarım kartını (Sayfa 112) kullanamazsınız. Bunun için, V3.0 projesini STEP 7'de açıp, cihazı V4.x CPU olarak değiştirmeli (Sayfa 144) ve bunun ardından STEP 7 projesini V4.x CPU'nuza yüklemelisiniz. Projenizi bir sürüm 4.x projesine değiştirdikten sonra, takip eden program aktarımları için bir V4.x aktarım kartı kullanabilirsiniz.

GET/PUT iletişimi

S7-1200 V3.0 CPU'larda, GET/PUT üzerinden iletişim standart olarak etkinleştirilmiştir. CPU V3.0'ınızı bir CPU V4.x (Sayfa 144) ile değiştirirseniz, uyumluluk bilgi alanında GET/PUT yordamının etkinleştirildiğine ilişkin bir mesaj görüntülenir.

Hareket kumandası desteği

S7-1200 CPU V4.x, sürüm V1.0 ve V2.0 hareket kumandası kütüphanelerini desteklemez. V1.0 veya V2.0 hareket kumandası kütüphaneleri olan bir STEP 7 projesi için bir cihaz değişimi gerçekleştirirseniz, bu cihaz değişiminin derlenmesi sırasında, V1.0 ve V2.0 hareket kumanda kütüphanelerini uyumlu sürüm 3.0 hareket kütüphanesi yordamları ile değiştirilir.
İki farklı hareket kumandası yordamı sürümüne (V3.0 ve V5.0) sahip bir STEP 7 projesi için V3.0 CPU'dan V4.x CPU'ya bir cihaz değişimi gerçekleştirirseniz, cihaz değişiminin derlenmesi sırasında uyumlu sürüm 5.0 hareket kumanda yordamlarının yerini alır.
CPU V3.0'dan CPU V4.x'e bir cihaz değişimi gerçekleştirirken, hareket kumandası için olan teknoloji nesnesinin (TO) sürümü, otomatik olarak V3.0'dan V5.0'a değiştirilmez. Daha yeni sürümlere yükseltmek istiyorsanız, yordam dizininde projeniz için gerekli S7-1200 hareket kumandası sürümünü seçmeniz gerekir.
S7-1200 V4.x CPU, aşağıdaki hareket kumandası sürümlerine izin verir:

- V3.0
- V4.0
- V5.0
- V6.0

V3.0 ve V5.0 hareket kumandası sürümlerinde TO yapısı farklıdır. Aynı şekilde tüm ilişkili bloklar da değişir. Blok arayüzleri, izleme tabloları ve takip fonksiyonu, yeni V5.0 hareket kumandası yapısına güncellenir. CPU V3.0 ve CPU V4.x arasındaki hareket kumandası eksen parametrelerindeki farklılıklar aşağıdaki iki tabloda listelenmiştir:

CPU V3.0 (Hareket kumandası V3.0)	CPU V4.x (Hareket kumandası V5.0)
Config.General.LengthUnit	Units.LengthUnit
Config.Mechanics.PulsesPerDriveRevolution	Actor.DriveParameter.PulsesPerDriveRevolution
Config.Mechanics.LeadScrew	Mechanics.LeadScrew
Config.Mechanics.InverseDirection	Actor.InverseDirection
Config.DynamicLimits.MinVelocity	DynamicLimits.MinVelocity
Config.DynamicLimits.MaxVelocity	DynamicLimits.MaxVelocity
Config.DynamicDefaults.Acceleration	DynamicDefaults.Acceleration
Config.DynamicDefaults.Deceleration	DynamicDefaults.Deceleration
Config.DynamicDefaults.EmergencyDeceleration	DynamicDefaults.EmergencyDeceleration
Config.DynamicDefaults.Jerk	DynamicDefaults.Jerk
Config.PositionLimits_SW.Active	PositionLimitsSW.Active
Config.PositionLimits_SW.MinPosition	PositionLimitsSW.MinPosition
Config.PositionLimits_SW.MaxPosition	PositionLimitsSW.MaxPosition
Config.PositionLimits_HW.Active	PositionLimitsHW.Active
Config.PositionLimits_HW.MinSwitchedLevel	PositionLimitsHW.MinSwitchLevel
Config.PositionLimits_HW.MaxSwitchedLevel	PositionLimitsHW.MaxSwitchLevel
Config.Homing.AutoReversal	Homing.AutoReversal
Config.Homing.Direction	Homing.ApproachDirection
Config.Homing.SideActiveHoming	Sensor[1].ActiveHoming.SideInput
Config.Homing.SidePassiveHoming	Sensor[1].PassiveHoming.SideInput

CPU V3.0 (Hareket kumandası V3.0)	CPU V4.x (Hareket kumandası V5.0)
Config.Homing.Offset	Sensor[1].ActiveHoming.HomePositionOffset
Config.Homing.FastVelocity	Homing.ApproachVelocity
Config.Homing.SlowVelocity	Homing.ReferencingVelocity
MotionStatus.Position	Position
MotionStatus.Velocity	Velocity
MotionStatus.Distance	StatusPositioning.Distance
MotionStatus.TargetPosition	StatusPositioning.TargetPosition
StatusBits.SpeedCommand	StatusBits.VelocityCommand
StatusBits.Homing	StatusBits.HomingCommand

Yeniden adlandırılan tek "commandtable" parametresi aşağıdaki komutları içeren Array'dir:

V3.0	V4.x
Config.Command[]	Command[]

Not: "Command[]" Array'i, sürüm 3.0'da "TO_CmdTab_Config_Command" tipinde bir UDT ve sürüm 4.x'de "TO_Struct_Command" tipinde bir UDT'dir.

Değiştirilen yordamlar

Aşağıdaki yordamlar, parametreleri veya davranışı değiştirir:

- RDREC ve WRREC (Sayfa 352)
- CONV (Sayfa 271-272)

HMI panel iletişimi

HMI panellerinizden (Sayfa 32) biri veya daha fazlası S7-1200 CPU sürüm 3.0'a bağlanmışsa, S7-1200 CPU sürüm 4.x ile gerçekleşen iletişim, bu süreçte kullanılan iletişim türüne ve HMI panelinin Firmware sürümüne bağlıdır. Projenizi yeniden derleyin ve CPU ve HMI'ya yükleyin ve/veya HMI Firmware'inizi güncelleyin.

Program bloklarını yeniden derlemek için ön koşul

Bir CPU V3.0'ı bir CPU V4.x ile değiştirdikten sonra, CPU sürüm 4.x'e yükleyebilmek için tüm program bloklarını yeniden derlemeniz gerekir. Bunun dışında, bir PLC seri numarasına bağlı teknik bilgi koruması (Sayfa 159) veya kopyalama koruması (Sayfa 161) olan bloklarda, blokları derlemek ve yükleyebilmek için korumayı kaldırmanız gerekir. (Ancak, bir memory card'a bağlı olan kopya korumasını kaldırmanız gerekli değildir). Başarılı bir derlemeden sonra, teknik bilgi korumasını ve/veya PLC seri numarasına bağlı kopyalama korumasını yeniden yapılandırabilirsiniz. Projenizin bir OEM (Orijinal Ekipman Üreticisi) tarafından sağlanan teknik bilgi korumalı bloklar içermesi durumunda, bu blokları sürüm 4.x için kullanılabilir kılmak üzere OEM ile iletişime geçmelisiniz.

Genel olarak Siemens, cihaz değişiminden sonra, donanım konfigürasyonunu ve yazılımı STEP 7'de yeniden derlemenizi ve projenizdeki tüm cihazlara yüklemenizi önerir. Projenin derlenmesi sırasında ortaya çıkan hataları düzeltin ve herhangi bir hata oluşmayana dek derlemeyi tekrarlayın. Bunun ardından projeyi CPU V4.x'e yükleyebilirsiniz.

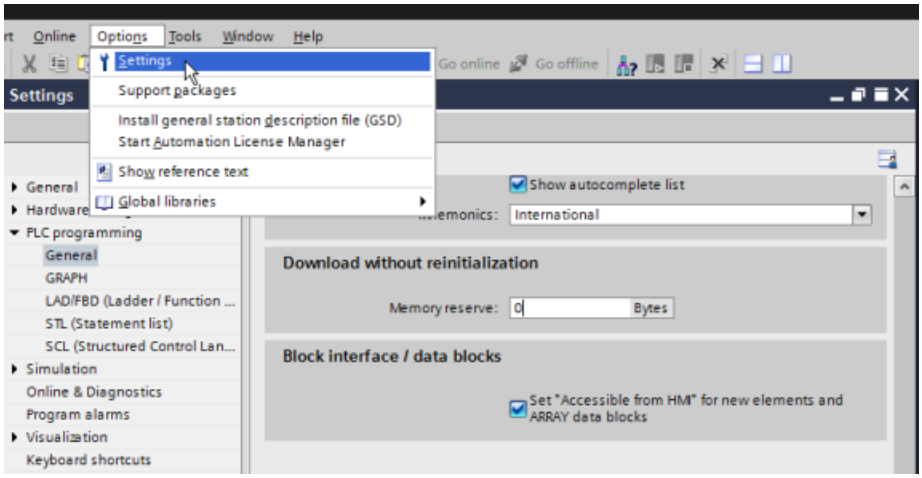
S7-1200 V3.0 projeleri, S7-1200 CPU V4.x ile uyumlu olmayabilir.

V4.0 ve üstü S7-1200, yeniden başlatma olmadan yüklemeyi desteklemek için her DB'ye 100 Bayt'lık bir rezerv alanı ekler.

Bir V3.0 projesini bir V4.x CPU'ya yüklemeden önce, bu 100 Bayt'lık rezerv alanını DB'lerden silebilirsiniz.

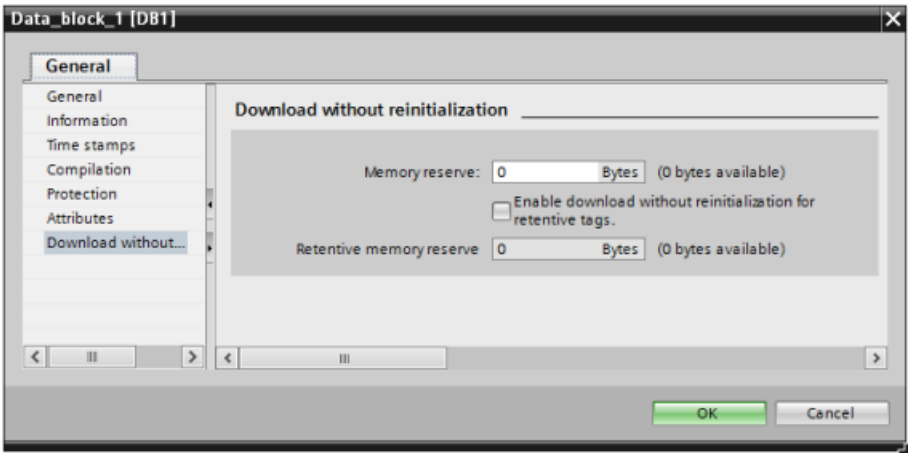
100 Bayt'lık rezerv alanını silmek için cihaz değişiminden önce aşağıdaki yöntemi izleyin:

1. TIA Portal'ın ana menüsünde "Opsiyonlar" > "Ayarlar" menü komutunu seçin.
2. Proje gezgininde "PLC programlama" > "Genel" düğümünü açın.
3. "Yeniden başlatma olmadan yükle" alanında 0 Bayt'lık bir bellek rezervi belirleyin.



Halihazırda cihaz değişimini gerçekleştirdiyseniz, her modüldeki 100 Bayt'lık rezervi ayrı ayrı silmeniz gerekir:

1. Proje gezgininde program blokları klasöründeki bir veri bloğuna farenin sağ tuşuyla tıklayın ve bağlam menüsünden "Özellikler " öğesini seçin.
2. Veri bloğunun özellikler iletişim kutusunda "Yeniden başlatma olmadan yükle" düğümünü seçin.
3. Bellek rezervini 0 Bayt olarak ayarlayın.
4. Bu işlemi, projenizdeki her veri bloğu için tekrarlayın.



D.3 S7-1200 ila V3.0 - yedek klemens blokları

Tablo D-1 S7-1200 CPU ila V3.0 - yedek klemens blokları

V3.0 üstü bir S7-1200 CPU'nuz varsa (sipariş numarası)	Bu durumda bu yedek klemens bloklarını kullanın (4/paket)	
	Klemens bloklarının sipariş numaraları	Klemens bloklarının açıklaması
CPU 1211C DC/DC/DC (6ES7211-1AE31-0XB0)	6ES7292-1BC30-0XA0 6ES7292-1AH30-0XA0 6ES7292-1AP30-0XA0	3 kutuplu, altın kaplamalı 8 kutuplu, altın kaplamalı 14 kutuplu, kalaylanmış
CPU 1211C AC/DC/röle (6ES7211-1BE31-0XB0)		
CPU 1211C DC/DC/röle (6ES7211-1HE31-0XB0)		
CPU 1212C DC/DC/DC (6ES7212-1AE31-0XB0)		
CPU 1212C AC/DC/röle (6ES7212-1BE31-0XB0)		
CPU 1212C DC/DC/röle (6ES7212-1HE31-0XB0)		
CPU 1214C DC/DC/DC (6ES7214-1AG31-0XB0)	6ES7292-1BC30-0XA0 6ES7292-1AM30-0XA0 6ES7292-1AV30-0XA0	3 kutuplu, altın kaplamalı 12 kutuplu, kalaylanmış 20 kutuplu, kalaylanmış
CPU 1214C AC/DC/röle (6ES7214-1BG31-0XB0)		
CPU 1214C DC/DC/röle (6ES7214-1HG31-0XB0)		
CPU 1215C DC/DC/DC (6ES7215-1AG31-0XB0)	6ES7292-1BF30-0XB0 6ES7292-1AM30-0XA0 6ES7292-1AV30-0XA0	6 kutuplu, altın kaplamalı 12 kutuplu, kalaylanmış 20 kutuplu, kalaylanmış
CPU 1215C AC/DC/röle (6ES7215-1BG31-0XB0)		
CPU 1215C DC/DC/röle (6ES7215-1HG31-0XB0)		

Tablo D-2 S7-1200 SM'ler V3.0'a kadar - yedek klemens blokları

V3.0 üstü bir S7-1200 SM'niz varsa (sipariş numarası)	Bu durumda bu yedek klemens bloklarını kullanın (4/paket)	
	Klemens bloklarının sipariş numaraları	Klemens bloklarının açıklaması
SM 1221 DI 8 × DC (6ES7221-1BF30-0XB0)	6ES7292-1AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 8 × DC (6ES7222-1BF30-0XB0)		
SM 1222 DQ 8 × röle (6ES7222-1HF30-0XB0)		
SM 1231 AI 4 × 13 Bit (6ES7231-4HD30-0XB0)	6ES7292-1BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1232 AQ 2 × 14 Bit (6ES7232-4HB30-0XB0)		
SM 1231 AI 4 × TC (6ES7231-5QD30-0XB0)		

V3.0 üstü bir S7-1200 SM'niz varsa (sipariş numarası)	**Bu durumda bu yedek klemens bloklarını kullanın (4/paket)**	
	Klemens bloklarının sipariş numaraları	**Klemens bloklarının açıklaması**
SM 1231 AI 4 × 16 Bit (6ES7231-5ND30-0XB0)	6ES7292-1BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1221 DI 16 × DC (6ES7221-1BH30-0XB0)	6ES7292-1AG30-0XA0	7 kutuplu, kalaylanmış
SM 1222 DQ 16 × DC (6ES7222-1BH30-0XB0)		
SM 1222 DQ 16 × röle (6ES7222-1HH30-0XB0)		
SM 1223 DI 8 × DC / DQ 8 × DC (6ES7223-1BH30-0XB0)		
SM 1223 8 × DC / 8 × röle (6ES7223-1PH30-0XB0)		
SM 1223 8 × AC / 8 × röle (6ES7223-1QH30-0XB0)		
SM 1234 AI 4 / AQ 2 (6ES7234-4HE30-0XB0)	6ES7292-1BG30-0XA0	7 kutuplu, altın kaplamalı
SM 1231 AI 8 × 13 Bit (6ES7231-4HF30-0XB0)		
SM 1232 AQ 4 × 14 Bit (6ES7232-4HD30-0XB0)		
SM 1231 AI 4 × RTD (6ES7231-5PD30-0XB0)		
SM 1231 AI 8 × TC (6ES7231-5QF30-0XB0)		
SM 1222 DQ 8 × röle (anahtarlama) (6ES7222-1XF30-0XB0)	6ES7292-1AL30-0XA0	11 kutuplu, kalaylanmış
SM 1223 DI 16 × DC/DQ 16 × DC (6ES7223-1BL30-0XB0)		
SM 1223 16 × DC/16 × röle (6ES7223-1PL30-0XB0)		
SM 1231 AI 8 × RTD (6ES7231-5PF30-0XB0)	6ES7292-1BL30-0XA0	11 kutuplu, altın kaplamalı

Endeks

"

"Değişken durumu" web sitesi, 828

"Modül bilgileri" standart web sitesi, 819

"Modüllerin çıkartılması veya takılması" OB'si, 75

/

/=-Kutu (FBD'de atamayı iptal etme), 206

=

== Kutu (FBD karşılaştırma işlemi), 224

=-Kutu (FBD'de atama yapma), 206

>

>=1 kutusu (FBD'deki mantıksal VEYA bağlantısı), 205

A

A bellek alanı
  Darbe çıkışları, 444
  Darbe kanallarını yapılandırma, 446

ABS (mutlak değer oluştur), 232

AC
  Kablaj ve topraklama direktifler, 54
  Potansiyel ayırmayla ilgili direktifler, 55
  Topraklama direktifleri, 55
  Kablaj direktifleri, 56

Açma ve kapatma gecikmesi süreleri, 211

ACOS (ark kosinüs değerini oluştur), 234

ACT_TINT (saat alarmını etkinleştir), 389

Açık kullanıcı iletişimi
  TRCV_C ile bir bağlantı kurma ve verileri okuma, 601
  TSEND_C ile bir bağlantı kurma ve veri gönderimi, 601
  Eski TRCV_C yordamıyla bir bağlantı kurma ve verileri okuma, 610
  Eski TSEND_C yordamı ile bir bağlantı kurma ve veri gönderimi, 610

Açık kullanıcı iletişimi için yordamların geri dönüş değerleri, 680

ADD (Topla), 229

Adresleme
  PLC'deki bellek alanları ve G/Ç, 93

Adresleri, durum bilgilerini sorgula, 414

Adım
  Markör Bit'leri, 90

Ağ bağlantısı
  Cihazları kullanma, 563
  iki cihaz arasında, 685
  iki cihaz arasında, 686
  Birden çok CPU, 689
  Birden çok CPU, 737
  Birden çok CPU, 741

Ağ bağlantısı oluşturma
  PLC cihazlarının arasında, 563

Ağ iletişimi, 681
  Kablo sonlandırması, 890

Aktarım kartı, 112
  Genel bakış, 112
  CPU'ya ekleme, 113
  Yol alma parametrelerinin yapılandırılması, 115
  Kaybedilen bir şifre için kullanılan boş aktarım kartı, 127

Aktarım kartı (program kartı), 1344

Aktarım modülü (T blok), 686

Aktif/pasif iletişim
  Partnerlerin yapılandırılması, 564
  Bağlantı ID'leri, 586
  Parametre, 589
  Partnerlerin yapılandırılması, 755

Akıllı telefon, Web sunucusuna erişim, 805

Akım tüketimi, 40, 1350

Akış kontrolü, 896
  Konfigürasyon, 896
  idare, 897

Alarmlar
Genel bakış, 68
Alarm gecikmesi, 81
ATTACH (OB'yi bir alarm olayına ata), 381
DETACH (bir OB'nin bir alarm olayına atamasını kaldır), 381
Gecikme alarmı yordamları, 391
Alt ağ maskesi, 572
Alım
Parametrelerin konfigürasyonu, 688
Konfigürasyon hatası, 918
Çalışma zamanı geri dönüş değerleri, 924
Konfigürasyon hatası, 1052
Çalışma zamanı geri dönüş değerleri, 1055
Analog değerlerin ölçeklenmesi, 278
Analog değerlerin standartlaştırılması, 278
Analog G/Ç
Fiziksel birimlere dönüştürme, 98
Konfigürasyon, 147
Fiziksel birimlere dönüştürme, 278
Durum göstergeleri, 1115
Adım yanıt süreleri (CPU), 1181
Adım yanıt süreleri (CPU), 1193
Adım yanıt süreleri (CPU), 1207
Adım yanıt süreleri (CPU), 1221
Adım yanıt süreleri (CPU), 1238
Adım yanıt süreleri (SM), 1271
Gerilim (Girişler), 1272
Akım (girişler), 1272
Akım ve gerilim (çıkışlar), 1273
Adım yanıtı süreleri (SB), 1316
Gerilim (Girişler), 1316
Akım (girişler), 1317
Akım ve gerilim (çıkışlar), 1318
Analog sinyal kartları
SB 1231, 1312
SB 1232, 1314
SB 1231 termo eleman, 1319
SB 1231 RTD, 1323
Analog sinyal modülleri
SM 1231, 1262
SM 1232, 1266
SM 1234, 1268
SM 1231 termo eleman, 1274
SM 1231 RTD, 1280
Arayüz konfigürasyonu
Yordamlar, 931
Aritmetik, 228, 229
Array'ler, elemanlara erişim, 271
Array'lerin değişkenlerle indekslenmesi, 271
ASi
RDREC ve WRREC (veri setlerini oku/yaz) yordamları, 352
AS-i Master CM 1243-2, 739
AS-i Master modülü CM1243-2 eklenmesi, 741
AS-i Slave eklenmesi, 741
Ağ bağlantısı, 741
AS-i Master cihaz konfigürasyonu, 742
AS-i Slave adresi atanması, 744
STEP 7 olmadan AS-i Slave konfigürasyonu, 746
AS-i Slave adreslerine sistem ataması, 746
Sistem ataması, 746
STEP 7 ile AS-i Slave konfigürasyonu, 747
Dijital değerlerin aktarılması, 747
Analog değerlerin aktarılması, 747
ASIN (ark sinüs değerini oluştur), 234
Aşırı gerilim kategorisi, 1167
AT değişken bindirmesi, 110
ATEX ruhsatı, 1163
ATH (ASCII karakter dizisini onaltılık sayıya dönüştür), 331
ATTACH (OB'yi bir alarm olayına ata), 381
ATTR_DB (bir veri bloğunun özniteliğini oku), 493
Avustralya ve Yeni Zelanda - RCM Mark ruhsatı, 1164
AWP_Enum_Def, 851
AWP_Import_Fragment, 855
AWP_In_Variable, 845, 849
AWP_Out_Variable, 848
AWP_Start_Fragment, 854
AWP komutları, 843
Değişkenleri yaz, 845
Özel değişkenleri oku, 848
Özel değişkenlerin yazılması, 849
Takma adların kullanımı, 850
Enum tiplerinin tanımlanması, 851
Enum tipinin referanslanması, 852
Fragmentlerin oluşturulması, 854
Fragmentlerin içe aktarımı, 855
Tanımların bağlanması, 855

B

Bağlantı kontağı
Maksimum dayanıklılık, 1338
Bağlantılar
İletişim türleri, 553
Bağlantı sayısı (PROFINET/PROFIBUS), 557
Partner, 564
Bağlantı ID'leri , 586
Konfigürasyon, 589
Ethernet protokolleri, 754
S7 bağlantısı, 754
Tipler, birden çok katılımcıyla bağlantılar, 754
Partner, 755
Web sunucusu, 886
Bağlantının kontrol edilmesi, 641
Başlangıç koşulları, mesaj, 900
Batarya kartı (BB)
BB 1297, 1328
Batarya yerleştirme, 1329
Baud hızı, 896
BB 1297, 1328
Bekleme kuyruğu, 81
Bekleme süresi, 896
Belirtilmemiş CPU, 131
Bellek
Yükleme belleği, çalışma belleği ve kalıcı bellek, 86
Sistem ve zamanlama markörü, 88
L (yerel veri belleği), 93
E (girişlerin proses görünümü), 95
A (Çıkışların proses görünümü), 95
M (Bit markörü), 96
geçici bellek, 96
Kullanımın denetlenmesi, 1128
Bellek adreslerine erişim, 93
Bellek alanları
Proses görünümü ve fiziksel erişim, 93
Boole değerlerinin veya Bit değerlerinin adreslenmesi, 94
Bellek alanları, Web sunucusu ile görüntüleme, 815
Bellek alanı E
İzleme tablosunun izlenmesi, 1132
Bellekteki adresler, 95
Bilgi kaynakları, 5
Bilinmeyen CPU sürümü nedeniyle hata, 1114
Bir Array için kullanılan değişken indeksi, 271
Bir çevrimiçi DB'nin değerlerini kaydetme, 1133
Bir cihazın eklenmesi
Belirtilmemiş CPU, 131
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Bir CPU V3.0'ın CPU V4.x ile değişimi, 1366
Bir CPU V3.0'ı V4.x'e yükseltme, 1366
Bir DB'nin değerlerini sıfırlama, 1133
Birden çok katılımcıyla bağlantılar, 754
Birden çok sayıda AWP değişkeni tanımı, 855
Birden fazla örnek DB'ye sahip bir FB, 176
Bir memory card'ın biçimlendirilmesi, 1126
Bir mesajın başlangıcı için karakter, 900
Bir PLC cihazıyla bir otomasyon çözümünün tasarlanması, 169, 170
Bir rölenin kullanım ömrü, 1169
Bir yedekleme kopyasının geri yüklenmesi, 1159
Bit bağlantısı
NO kontak ve NC kontaklar, 204
AND (VE), OR (VEYA) ve XOR (ÖZEL VEYA) işlemleri, 205
NOT işlemi (VKE'yi ters çevir), 205
NO kontak ve NC kontak bobinleri, 206
Ayarlama ve sıfırlama işlemleri, 207
Yükselen kenar ve alçalan kenar işlemleri, 209
Bitiş koşulları, mesaj, 902
Blendajlı bağlantılar, topraklama direktifleri, 55

Bloklar
desteklenen, 29
Tipler, 60
Tipler, 60
Fonksiyon (FC), 60
Fonksiyon bloğu (FB), 60
Organizasyon blokları (OB'ler), 68
Organizasyon blokları (OB'ler), 81
Teknik bilgi koruması (şifre koruması), 160
Kopyalama koruması, 161
Doğrusal ve yapılandırılmış programlar, 170
çağırma, 172
Organizasyon blokları (OB'ler), 173
Fonksiyon (FC), 174
Fonksiyon bloğu (FB), 175
Veri bloğu (DB), 175
Veri bloğu (DB), 176
çağırma, 186
cihaza yükleme, 194
Blokların bir çevrimiçi CPU'dan kopyalanması, 199
Tutarlılık kontrolü, 203
CPU 1211C Teknik veriler, 1174
CPU 1212C teknik veriler, 1186
CPU 1214C teknik veriler, 1200
CPU 1215C teknik veriler, 1214
CPU 1217C teknik veriler, 1230
Blokların bir çevrimiçi CPU'dan kopyalanması, 199
Bobinler, 204
Boole değerlerine veya Bit değerlerine erişim, 94
Boş satır, 900

C

CALCULATE (hesapla), 228
CAN_DINT (gecikme alarmını iptal et), 391
CAN_TINT (saat alarmını sil), 389
CB 1241
Sonlandırma, 891
RS485, 1340
CEIL (kayar noktalı sayıdan bir sonraki daha yüksek tamsayıyı oluştur), 276
CE ruhsatı, 1160
Char (Character veri tipi), 104
Chars_TO_Strg (CHAR elemanlarından oluşan Array'i karakter dizisine dönüştür), 329
Cihaz
PROFINET IO, 689
PROFINET IO cihaz adları, 690
Shared, 705
Cihaz değişimi
İzlenecek prosedür, 144
CPU V3.0'ın CPU V4.x ile değişimi, 1366
Cihaz konfigürasyonu, 128
Yeni cihaz ekleme, 130
Algılama, 131
Modüller ekleme, 133
Bir cihaz tipinin değiştirilmesi, 144
CPU'nun yapılandırılması, 145
Modüllerin yapılandırılması, 147
Modüllerin yapılandırılması, 162
Ağ bağlantısı, 563
PROFINET portu, 571
Ethernet portu, 571
Saat senkronizasyonu özelliği (PROFINET), 580
PROFIBUS, 738
AS-i Slave, 744
Cihazın değiştirilmesi, 144
CM CANopen modülü
021620-B, 021630-B, 1348
Compact Switch modül, CSM 1277, 1348
CONCAT (karakter dizilerini birleştir), 334
CONTINUE, SCL, 301
CONV (değeri dönüştür), 272
COS (kosinüs değerini oluştur), 234
CountOfElements (ARRAY elemanlarının sayısını sorgula), 265
CP modülü
Web sunucusuna erişim, 806
Web sunucusu başlangıç sayfası, 814
CPU'dan yükleme
Blokların bir çevrimiçi CPU'dan kopyalanması, 199
CPU'lar
CPU 1211C, 1173
CPU 1212C, 1185
CPU 1214C, 1199
CPU 1215C, 1213
CPU 1217C, 1228

CPU'ya yükle
Projeyi cihaza, 194
MAC ve IP adreslerini görüntüleme, 576
Kullanıcı tanımlı web siteleri için DB'ler, 861
CPU konfigürasyonu
Çevrim süresi denetlemesi, 84
İşletim parametreleri, 145
Darbe kanalları, 446
HMI ile iletişim, 684
Birden çok CPU, 685
CPU memory card
Ekleme, 113
Aktarım kartı, 116
Program kartı, 120
CPU özellikleri, kullanıcı tanımlı web siteleri
STEP 7 konfigürasyonu, 858
Birden çok dil kurulumu, 879
CPU tanımlama, Web sunucusu ile görüntüleme, 815
CREATE_DB (veri bloğu oluştur), 487
CSM 1277 Compact Switch modül, 1348
CTD (geriye doğru sayım), 218-219
CTRL_HSC_EXT (hızlı sayaca kumanda et, geliştirilmiş)
Yordama genel bakış, 510
Örnek, 511
CTRL_HSC (hızlı sayaca kumanda et), 537
CTS (donanım akış kontrolü, PtP), 897
CTU (ileriye doğru sayım), 218-219
CTUD (ileriye ve geriye doğru sayım), 218-219
cULus ruhsatı, 1161

D

D_ACT_DP, 362
Dalgalanma dayanımı, 1166
Darbe
Toplama Bit'leri, dijital giriş konfigürasyonu, 150
Çıkışlar, 444
Darbe
Gecikme, 211
Darbe toplama, 147
DataLogClear, 470
DataLogDelete, 472
Date
Date veri tipi, 102
DTL (Date and Time long veri tipi), 103
DB_ANY_TO_VARIANT (DB_ANY'yi VARIANT'a dönüştür), 281
DB (veri bloğu), 176
DB değerlerinin anlık kaydı, 1133
DB fragmentleri (kullanıcı tanımlı web siteleri)
AWP komutu ile oluşturma, 854
AWP komutu ile içe aktarım, 855
oluşturma, 858
DB fragmentlerinin manuel kumandası, 880
DC
Kablaj ve topraklama direktifler, 54
Potansiyel ayırmayla ilgili direktifler, 55
Topraklama direktifleri, 55
Kablaj direktifleri, 56
Endüktif yüklerle ilgili direktifler, 58
Çıkışlar, 1169
DEC (azalt), 231
DECO (şifreyi çöz), 305
Değerlerin karşılaştırılması, 224
Değerleri zorlama, 1138, 1139
değişken
Denetim durumu veya değeri, 1132
Force fonksiyonu, 1139
Değişken
Slice, 108
Bindirme, 110
Değiştirme
Web sunucusu üzerinden değişkenler, 828
İzleme tablosu, 1135
DELETE_DB (veri bloğunu sil), 495
DELETE (bir karakter dizisindeki karakteri sil), 336
DEMUX (çoklamayı çöz), 307
Denizcilik sektörü için ruhsat, 1165
Deserialize, 239
Destek, 4
DETACH (bir OB'nin bir alarm olayına atamasını kaldır), 381
DeviceStates, örnekler, 422
DeviceStates (bir G/Ç sisteminin modül durumunu oku) yordamı, 420

Devre şemaları
CPU 1211C, 1182
CPU 1212C, 1195
CPU 1214C, 1209
CPU 1215C, 1224
CPU 1217C, 1240
SM 1221, 1245
SM 1222, 1249
SM 1223, 1256
SM 1223, 1261
SM 1231, 1264
SM 1232, 1267
SM 1234, 1271
SM 1231 termo eleman, 1275
SM 1231 RTD, 1282
SM 1278 IO-Link-Master, 1289
SB 1221, 1302
SB 1222, 1305
SB 1223, 1308
SB 1223, 1311
SB 1231, 1313
SB 1232, 1315
SB 1231 termo eleman, 1322
SB 1231 RTD, 1325
CB 1241 RS 485, 1340
Dijital G/Ç
Konfigürasyon, 147
Darbe toplama, 147
Durum göstergeleri, 1114
Dijital giriş filtresi süresi, 148
Dijital sinyal kartları
SB 1221, 1301
SB 1222, 1303
SB 1223, 1306
SB 1223, 1309
Dijital sinyal modülleri
SM 1221, 1244
SM 1222 DQ 8, 1246
SM 1222 DQ 16, 1247
SM 1223 DI ve DQ, 1253
SM 1223, 1259
Dil değiştirme, kullanıcı tanımlı web siteleri, 876
Diller, kullanıcı tanımlı web siteleri, 876
Dinamik bağlantı, 161
Direktifler
Kurulum, 38
Montaj için izlenecek prosedür, 44
CPU'nun takılması, 45
Kablaj ve topraklama direktifler, 54
Ayırma, 55
Topraklama, 55
Kablaj, 56
Lamba yükleri, 57
endüktif yükler, 58
DIS_AIRT (daha yüksek öncelikli alarmların ve asenkron hata olaylarının yürütülmesini devre dışı bırak), 393
DIV (böl), 229
Dizinler, kullanıcı tanımlı web siteleri için diller, 876
Doğrusal programlama, 170
Dokümantasyon, 5
Donanım akış kontrolü, 897
Donanım konfigürasyonu, 128
Yeni cihaz ekleme, 130
Algılama, 131
Modüller ekleme, 133
CPU'nun yapılandırılması, 145
Modüllerin yapılandırılması, 147
Modüllerin yapılandırılması, 162
Ağ bağlantısı, 563
PROFINET portu, 571
Ethernet portu, 571
PROFIBUS, 738
AS-i Slave cihazı, 744
Dönüştürme (SCL yordamları), 273
DPNRM_DG, 378
DPRD_DAT (bir DP Standart Slave'in tutarlı verilerini oku), 371
DP Standart Slave'ler
Tüm girişleri GETIO ile oku, 354
Tüm çıkışları SETIO ile yaz, 355
Girişlerin parçasını GETIO_PART ile oku, 356
Çıkışların bölümünü SETIO_PART ile yaz, 358
DPWR_DAT (bir DP Standart Slave'in tutarlı verilerini yaz), 371
Duraklama, 899, 900
Durum LED'leri
İletişim arayüzleri, 889
CPU, 1113

Durum OB'si, 77

E

Eksen kumanda komutlarının hareket dizisi olarak yürütülmesi (MC_CommandTable), 546
Elektromanyetik uyumluluk (EMU), 1165, 1166
EN_AIRT (daha yüksek öncelikli alarmların ve asenkron hata olaylarının yürütülmesini etkinleştir), 393
ENCO (şifrele), 305
Endian formatı, 262
ENDIS_PW (şifre onayını kısıtla ve serbest bırak), 286
Endüktif yükler için koruma devreleri, 58
Endüktif yükler için şok giderici koruma devreleri, 58
Endüktif yüklerle ilgili direktifler, 58
Endüstriyel ortamlar, ruhsatlar, 1165
Enum tiplerinin tanımlanması, kullanıcı tanımlı web siteleri, 851, 852
EN ve ENO (sinyal akışı), 192
E-posta, TMAIL_C ile gönderme, 645
EQ_ElemType (Bir ARRAY elemanının veri tipini, bir değişkenin veri tipiyle EŞİTLİK açısından karşılaştır), 226
EQ_Type (veri tipini, bir değişkenin veri tipiyle EŞİTLİK açısından karşılaştır), 226
Erişilebilir katılımcılar
  Firmware güncellemesi, 1121
  Bir memory card'ın biçimlendirilmesi, 1126
Erişim
  Kullanıcı tanımlı web siteleri, 862
Erişim koruması, CPU, 156
Eski TCON, TDISCON, TSEND ve TRCV yordamları, 631
Eski TRCV_C ve TSEND_C yordamları, 610
Eski USS protokolü için kütüphane
  Genel bakış, 1060
  Yordam sürümleri, 1060
  Kullanım gereklilikleri, 1061
  USS_PORT (USS ağı üzerinden iletişim düzenlenmesi), 1061
  USS_DRV (sürücü ile veri alışverişi), 1062
  USS_RPM (sürücüden parametre okunması), 1065
  USS_WPM (sürücüdeki parametreyi değiştirme), 1066
  Durum kodları, 1067
Esnek makineler (konfigürasyon kumandası), 134
Ethernet
  İletişim türleri, 553
  İletişim bağlantısı sayısı, 557
  anahtarlamalı, 562
  Ağ bağlantısı, 563
  IP adresi, 571
  MAC adresi, 571
  Genel bakış, 582
  Geçici mod, 584
  Bağlantı ID'leri, 586
  CSM 1277 Compact Switch modül, 1348
Ethernet, eski yordamlar
  TSEND_C ve TRCV_C, 610
  TCON, TDISCON, TSEND ve TRCV, 631
Ethernet, yordamlar
  RDREC ve WRREC (veri setlerini oku/yaz), 352
  RALRM (alarmı al), 359
  DPWR_DAT (bir DP Standart Slave'in tutarlı verilerini yaz) ve DPRD_DAT (Bir DP Standart Slave'in tutarlı verilerini oku), 371
  RCVREC (veri setini al), 373
  PRVREC (veri setini hazırla), 376
  DPNRM_DG (bir DP-Slave'in teşhis verilerini oku), 378
  TSEND_C ve TRCV_C, 601
  TCON, TDISCON, TSEND ve TRCV, 616
  TUSEND ve TURCV, 665
  T_CONFIG (arayüzü yapılandır), 670
  GET (uzak CPU'dan veri oku) ve PUT (uzak CPU'ya veri yaz), 750
Ethernet protokolleri, 582
  Birden çok katılımcıyla bağlantılar, 754
EXIT, SCL, 302
EXP (üs değerini al), 234
EXPT (üssünü al), 234

F

F_TRIG (Değişkenin negatif sinyal kenarında ayarlanması), 210
Fabrika ayarlarına sıfırlama, 1123, 1123
Faz kayması, uyandırma alarmı OB'leri, 70
FB (fonksiyon bloğu), 60
FBD (fonksiyon planı), 184
FC (fonksiyon), 60, 174
FieldRead (alanı oku), 270
FieldWrite (alana yaz), 270
FileDelete (memory card'daki dosyayı sil), 507
FileReadC (dosyayı memory card'dan oku), 502
FileWriteC (dosyayı memory card'a yaz), 504
FILL_BLK (belleği Bit örüntüsüyle doldur), 244
Filtre süresi, 148
FIND (Bir karakter dizisindeki karakterleri bul), 339
Firmware güncellemesi
  bir memory card ile, 124
  bir memory card ile, 124
  Web sunucusu üzerinden, 822
  Web sunucusu üzerinden, 822
  SIMATIC Automation Tool ile, 822
  SIMATIC Automation Tool ile, 822
  STEP 7'den, 1120
  STEP 7'den, 1120
FLOOR (kayar noktalı sayıdan bir sonraki daha düşük tamsayıyı oluştur), 276
FM sertifikalandırması, 1162
Fonksiyon, I-Device, 692
Fonksiyon (FC)
  Genel bakış, 60
  Teknik bilgi koruması, 160
  Doğrusal ve yapılandırılmış programlar, 170
  Kullanıcı programındaki kod bloklarının çağrılması, 172
  Genel bakış, 174
Fonksiyon bloğu (FB)
  Genel bakış, 60
  Teknik bilgi koruması, 160
  Doğrusal ve yapılandırılmış programlar, 170
  Kullanıcı programındaki kod bloklarının çağrılması, 172
  Genel bakış, 175
Fonksiyonlar, yeni, 33
FOR, SCL, 299
FRAC (ondalık basamakları belirle), 234
Frekans, adım Bit'leri, 90

G

G/Ç
  Endüktif yüklerle ilgili direktifler, 58
  Adresleme, 97
  Dijital durum göstergeleri, 1114
  Analog durum göstergeleri, 1115
  Durumu programlama editöründe izleme, 1133
  Bir izleme tablosunda izleme, 1135
  Değerleri zorlama, 1138
  Değerleri zorlama işlemi, 1139
  Adım yanıt süreleri (CPU), 1181
  Adım yanıt süreleri (CPU), 1193
  Adım yanıt süreleri (CPU), 1207
  Adım yanıt süreleri (CPU), 1221
  Adım yanıt süreleri (CPU), 1238
  Adım yanıt süreleri (SM), 1271
  Analog giriş gerilimi, 1272
  Analog giriş akımı, 1272
  Analog çıkış akımı ve gerilimi, 1273
  Adım yanıtı süreleri (SB), 1316
  Analog giriş gerilimi, 1316
  Analog giriş akımı, 1317
  Analog çıkış akımı ve gerilimi, 1318
G/Ç sistemi, veri alışverişi, 697
G/Ç sistemleri arasında veri alışverişi, 697
GATHER, 252
GATHER_BLK, 255
Geçici bellek
  OB öncelik kademesi başına maksimum, 96
  Bloklarla kullanım, 96-97
Geçici mod, TCP ve ISO on TCP, 584
Gecikme, 81
Gecikme alarmları (programlama), 391
Gecikme alarmı OB'leri, 69
Gen_UsrMsg (kullanıcı teşhis mesajları oluştur), 394
Genişletilmiş blok arayüzü
  RUN işletim durumunda yükleme, 1145
GEO2LOG (donanım kodunu yuva aracılığıyla belirle) yordamı, 496
GEOADDR, 501

Geri dönüş değerleri
Açık kullanıcı iletişimi için yordamlar, 680
PtP yordamları, 909
Gerilim beslemesi modülü
PM1207, 1348
GET_DIAG (teşhis bilgilerini oku), 430
GET_ERROR_ID (hata ID'sini yerel olarak çağır), 292
GET_ERROR (hatayı yerel olarak çağır), 291
Get_Features (genişletilmiş fonksiyonların çağrılması), 929
Get_IM_Data (tanımlama ve bakım verilerini oku) yordamı, 407
Get_name, 408
GET (verileri çıkartılmış CPU'dan oku), 565
Bağlantının yapılandırılması, 565
GetBlockName (bloğun adını oku), 350
GetInstanceName (blok örneğinin adını oku), 346
GetInstancePath (blok örneğinin birleşik global adını sorgula), 348
GETIO, 354
GETIO_PART, 356
GETSMCInfo, 435
GetSMCInfo (memory card ile ilgili bilgileri oku), 435
GetStationInfo, 414
GetSymbolName (giriş parametresindeki bir değişkenin adını oku), 340
GetSymbolPath (giriş parametresi kaynağının birleşik global adını sorgula), 343
Giriş filtresi süresi, 148
Girişler ve çıkışlar
Darbe toplama Bit'leri, 150
Giriş simülatörleri, 1344
Giriş ve çıkışlar
İzleme, 1132
Gizli PLC konfigürasyon verilerinin korunması
SIMATIC memory card, 122
Dijital güvenlik asistanı, 153
Fonksiyonlar, 154
Global kütüphane
USS protokolüne genel bakış, 942
USS protokolüne genel bakış, 1060
Global veri bloğu, 93, 176
Gömme derinliği, 60
Gönderilen mesaj konfigürasyonu, 898
Gönderim
Konfigürasyon hatası, 914
Çalışma zamanı hatası, 922
Konfigürasyon hatası, 1049
Çalışma zamanı hatası, 1054
Gönderim parametrelerinin konfigürasyonu, 564, 687, 755
Görselleştirme, HMI cihazları, 32
GOTO, SCL, 303
GSD dosyası, 701
GÜÇ AÇIK'tan sonra yol alma, 64
Güç bilançosu, 40
Genel bakış, 1350
Örnek, 1352
Hesaplamalar için formül, 1353
Güncelleme OB'si, 77
Güvenli iletişim, 560
Güvenlik
Teşhis tamponundaki olayların azaltılması, 91
Kaybedilen şifre, 127
Dijital güvenlik asistanının kullanılması, 153
Gizli PLC konfigürasyon verilerinin korunması, 154
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Kopyalama koruması, 161

H

Hafıza kartı
Yol alma parametrelerinin yapılandırılması, 115
Halka
Topoloji, 721
port, 728
Hareket dizisi (MC_CommandTable), 546
Hareket kumandası
Genel bakış, 539
Yordamlar, 541
Hata
Zaman hatası, 72
Teşhis hatası, 73
Gelişmiş yordamlardaki sık görülen hatalar, 502
Hata giderme
LED göstergeler, 1113
Teşhis tamponu, 1128
Hat boşta, 899

Hava akışı, 39
HMI cihazları
Genel bakış, 32
Ağ bağlantısı, 563
PROFINET iletişiminin yapılandırılması, 684
HSC_Count (sistem veri tipi), 515
HSC_Frequency (sistem veri tipi), 518
HSC_Period (sistem veri tipi), 516
HSC (hızlı sayaç)
Konfigürasyon, 525
Sayma türleri, 527
İşletim aşaması, 528
HTA (onaltılık sayıyı ASCII karakter dizisine dönüştür), 331
HTML sayfaları, kullanıcı tanımlı, 840
Geliştirilmesi, 841
yenilenme, 842
S7-1200 verilerine erişim, 843
Sayfa düzeni, 858
Web sitesi örneği, 869
Diller için saklama konumu, 879
HTTP bağlantıları, Web sunucusu, 886
Hızlı sayaç
CTRL_HSC_EXT (hızlı sayaca kumanda et, geliştirilmiş), 510
Senkronizasyon, 519
Geçit (gate) fonksiyonu, 520
Kaydetme fonksiyonu, 522
Karşılaştırma fonksiyonu, 523
Uygulama, 524
Konfigürasyon, 525
Sayma türleri, 527
İşletim aşaması, 528
CTRL_HSC_EXT (hızlı sayaca kumanda et, geliştirilmiş), 537
Zorlanamaz, 1139

I

I-Device (akıllı IO cihazı)
Fonksiyonlar, 692
Özellikler, 693
alt seviyedeki PN IO sistemi, 694
Yapılandırma, 699
GSD dosyası ile yapılandırma, 701
Shared, 711
IF-THEN, SCL, 297
IN_Range (değer, aralık dahilinde), 225
INC (arttır), 231
INSERT (bir karakter dizisine karakter ekle), 337
INV (tekli tümleyen oluştur), 304
IO2MOD (donanım kodunu bir G/Ç adresi aracılığıyla belirle), 499
IO-Link
Sinyal modülü (Link Master), 1286
Cihaz profili, 1289
Özellikler, 1290
Fonksiyonlar, 1290
Değişim, 1291
Fabrika ayarlarına sıfırlama, 1291
Fiş yerleşimi, 1292
Diyagram, 1293
Yapılandırma, 1294
Parametre, 1294
Adres aralığı, 1294
Parametre, 1295
LED durumu, 1297
Teşhis alarmları, 1299
IP adresi, 145
Cihaz konfigürasyonu, 145
Atama, 567
çevrimiçi CPU'ya atama, 570
Yapılandırma, 571
MAC adresi, 571
Atama, 575
görüntüleme ve yükleme, 576
CPU'nun çevrimiçi olarak yapılandırılması, 1120
IP adresi, acil durum adresi (geçici), 763
IP yönlendiricisi, 571
IS_ARRAY (ARRAY açısından kontrol et), 227
IS_NULL (işaretçi EŞİTTİR sıfır sorgusu), 227
ISO on TCP
Bağlantı konfigürasyonu, 564
Geçici mod, 584
Bağlantı ID'leri, 586
Parametre, 589
ISO on TCP protokolü, 582

J

JavaScript, standart web siteleri, 886
JMP_LIST (atlama listesini tanımla), 283

JMP (VKE = 1'de atla), 282
JMPN (VKE = 0'da atla), 282

K

Kablaj direktifleri, 39
  Hava akışı ve soğutma için bırakılması gereken mesafe, 39
  Ön koşullar, 54
  Topraklama, 55
Kablolar
  Ağ iletişimi, 890
  Genişletme, 1347
Kalıcı bellek, 28
  CPU 1211C, 1173
  CPU 1212C, 1185
  CPU 1214C, 1199
  CPU 1215C, 1213
  CPU 1217C, 1228
Kalıcı blok değişkenleri
  RUN işletim durumunda yükleme, 1146
Karakter aralığı, 903
Karakter dizisi
  Mesaj başlangıcı , 900
  Mesaj bitişi, 904
Karakter pozisyonu, mesaj uzunluğu, 905
Karşılaştırma tablosu
  CPU varyantları, 28
  Modüller, 31
Kaybedilen şifre, 127
Kenar işlemleri, yükselen ve alçalan, 209
Kirlenme derecesi, 1167
Kirlenme derecesi/aşırı gerilim kategorisi, 1167
Klasör, kullanıcı tanımlı web siteleri için diller, 876
Klemens bloğu, 51, 1359
Klemens bloğu, takma ve sökme, 51
Kod blokları, genel bakış, 29
Konfigürasyon
  Çevrim süresi , 84
  İletişim yükü, 86
  Yol alma parametreleri, 115
  Algılama, 131
  Modüller ekleme, 133
  CPU parametreleri, 145
  Modüller, 147
  Modüller, 162
  HSC (hızlı sayaç), 525
  Ağ bağlantısı, 563
  PROFINET portu, 571
  Ethernet portu, 571
  IP adresi, 571
  MAC adresi, 571
  PLC/PLC iletişimi, 685
  PROFIBUS adresinin atanması, 738
  AS-i Master cihaz, 742
  AS-i Slave cihazı, 744
  Arayüzler, 895
  İletişim arayüzleri, 895
  Mesaj alımı, 899
  3964(R), iletişim arayüzleri/portlar, 906
  RS422, işletim türleri, 937
  RS485, işletim türleri, 938
Konfigürasyon, 3964(R)
  Öncelik ve protokol parametreleri, 907
Konfigürasyon, kullanıcı tanımlı web siteleri
  STEP 7 konfigürasyonu, 858
  Birden çok dil kurulumu, 879
Konfigürasyon kumandası (opsiyon uygulaması)
  Kumanda veri seti, 138
Konfigürasyon kumandası (opsiyon yönetimi), 134
  Örnek, 141
Kontaklar, 204
Kontrol paneli, CPU, 1127
Kopyalama koruması
  Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Kore sertifikalandırması ruhsatı, 1164
Koruma
  CPU erişim koruması, 156
  Bir kod bloğu için teknik bilgi koruması, 160
Koruma derecesi, 1168

Koruma kademesi
Kaybedilen şifre, 127
CPU, 156
Kod yapı elemanı, 160
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Koruma türü, 1168
Kullanıcı konfigürasyonu, Web sunucusu, 801
Kullanıcı programı
Program kartı, 112
Aktarım kartı, 112
Memory card, 112
Şifre koruması, 160
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Doğrusal ve yapılandırılmış programlar, 170
Kullanıcı programındaki kod bloklarının çağrılması, 172
Organizasyon blokları (OB'ler), 173
cihaza yükleme , 194
Blokların bir çevrimiçi CPU'dan kopyalanması, 199
Kullanıcı programındaki kod bloklarının çağrılması, 172
Kullanıcı tanımlı web siteleri
Genel bakış, 840
HTML editör ile oluşturma, 841
yenilenme, 842
S7-1200 verilerine erişim için AWP komutları, 843
Değişkenleri oku, 844
Değişkenleri yaz, 845
Özel değişkenleri oku, 848
Özel değişkenlerin yazılması, 849
Fragmentlerin oluşturulması, 854
Fragmentlerin içe aktarımı, 855
Özel karakterlerin işlenmesi, 856
Yapılandırma, 858
Program bloklarının oluşturulması, 858
Program bloklarının silinmesi, 859
WWW yordamı ile etkinleştirme, 860
STEP 7'de programlama, 860
İlgili DB'lerin CPU'ya yüklenmesi, 861
PC üzerinden erişim, 862
Yükleme belleği sınırlamaları, 863
Örnek, 864
HTML listesi, 869
Çoklu dil, 876
Birden çok dil için konfigürasyon, 879
DB fragmentlerinin manuel kumandası, 880
Kumanda DB'si üzerinden etkinleştirme ve devre dışı bırakma, 880
Kullanıcı tanımlı web siteleri için DB'ler oluşturma, 858
Kullanıcı tanımlı web siteleri için kumanda DB'si
WWW yordamı parametreleri, 860
Talep komutları ve durumları, 880
Global komutlar, 880
Kullanıcı tanımlı web sitelerinde Enum tipleri, 851, 852
Kullanıcı tanımlı web sitelerinde Enum tiplerinin atanması, 852
Kullanıcı tanımlı web sitelerindeki takma adlar, 850
Kullanıcı tanımlı web sitelerinde program dilinin değiştirilmesi, 877
Kullanıcı tanımlı web sitelerinin güncellenmesi, 842
Kullanım modülü, 32
Kullanımı görüntülemek için çapraz referans, 202
Kurulum
Güç bilançosu, 40
Kısıtlı TSAP'ler ve port numaraları, 681

L

Label (atlama işareti), 282
LAD (kontak planı)
Genel bakış, 183
Programlama editörü ve çevrimiçi durum, 1133
Çevrimiçi durumu denetleme, 1133
Durum, 1138
Lamba yükleriyle ilgili direktifler, 57
LED (LED durumunu okuma), 406
LED göstergeler
İletişim arayüzleri, 889
CPU durumu, 1113
İletişim arayüzü, 1113
LEFT (bir karakter dizisinin sol karakterlerini oku), 335
LEN (bir karakter dizisinin uzunluğunu belirle), 334
LIMIT (limiti ayarla), 233
Lisanslar, OPC UA, 1364
LN (Doğal logaritmasını al), 234
LOG2GEO (yuvayı donanım kodu aracılığıyla belirle), 498
Lojik analizörü, 1150
LOWER_BOUND (alt ARRAY sınırını oku), 266

M

MAC adresi
Konfigürasyon, 571
görüntüleme ve yükleme, 576
belirleme, 577
MAC ve IP adreslerini görüntüleme, 576
MAX_LEN (bir karakter dizisinin maksimum uzunluğu), 333
MAX (maksimumu çağır), 232
MB_CLIENT, 958
MB_CLIENT yordamları (PROFINET üzerinden Modbus TCP istemcisi olarak iletişim), kullanımdan kaldırılmıştır, 1070
MB_COMM_LOAD (Modbus RTU için PtP modülü üzerindeki portun yapılandırılması), kullanımdan kaldırılmıştır, 1086
MB_MASTER (Modbus Master olarak PtP portu üzerinden iletişim), kullanımdan kaldırılmıştır, 1088
MB_RED_CLIENT, 977
MB_RED_SERVER, 995
MB_SERVER, 968
MB_SERVER (Modbus TCP sunucusu olarak PROFINET üzerinden iletişim), kullanımdan kaldırılmıştır, 1076
MB_SLAVE (Modbus Slave olarak PtP portu üzerinden iletişim), kullanımdan kaldırılmıştır, 1094
MB_Slave (SIPLUS I/O veya PtP portu üzerinden Modbus RTU Slave olarak iletişim), 1032-1033
MC_ChangeDynamic (eksenin dinamik ayarlarını değiştir), 548
MC_CommandTable, 546
MC_Halt (ekseni duraklat), 544
MC_Home (eksenin referans nokta hareketini gerçekleştir), 543
MC_MoveAbsolute (ekseni mutlak şekilde konumla), 544
MC_MoveJog (Ekseni JOG modunda hareket ettir), 546
MC_MoveRelative (ekseni bağıl şekilde konumla), 545
MC_MoveVelocity (ekseni önceden tanımlanmış hızla hareket ettir), 545
MC_Power (ekseni serbest bırak/bloke et), 542
MC_ReadParam (bir teknoloji nesnesinin parametrelerini oku), 547
MC_Reset (hatayı onayla), 543
MC_WriteParam (teknoloji nesnesinin parametrelerini yaz), 547
MC-PostServo OB, 80
MC-PreServo OB, 79
Medya yedekliliği
Bir halka topolojisindeki çalışma şekli, 722
Yapılandırma, 726
Medya yedeklilik protokolü (MRP), 719

Memory card, 112
Genel bakış, 112
CPU'ya ekleme, 113
Aktarım kartı, 116
Program kartı, 120
Gizli PLC konfigürasyon verilerinin korunması, 122
Firmware güncellemesi, 124
Kaybedilen bir şifre için kullanılan boş aktarım kartı, 127
Bilgileri oku, 435
Uyumsuzluk hatası, 1114
Memory card'ı CPU'ya takma, 113
Memory card kullanım ömrü, 122
Mesafe, hava akışı ve soğutma, 39
Mesaj alımının yapılandırılması
PtP cihazı, konfigürasyon, 899
PtP örnek programı, 935
Mesaj gönderimi için konfigürasyon, 898
Mesaj gönderimi yapılandırılması
PtP cihazı, konfigürasyon, 898
PtP örnek programı, 934
Mesaj konfigürasyonu
Gönderim parametrelerinin yapılandırılması, 898
Alım parametreleri, 899
Yordamlar, 931
Mesaj koşulları
Başlama, 900
Son, 902
Mesaj uzunluğu, maksimum, 903
Mesajın sonu için karakter, 903
MicroMaster sürücü, bağlanması, 952
MID (bir karakter dizisinin orta karakterlerini oku), 335
MIN (minimumu çağır), 232
Mobil cihaz
Web sunucusuna erişim, 805
Web sitesi yapısı, 809
MOD (bir bölme işleminin kalanı), 230
Modbus
Fonksiyon kodları, 955
Ağdaki istasyon adresleri, 956
Bellekteki adresler, 956
Modbus_Comm_Load yordamı (Modbus RTU için SIPLUS I/O veya PtP modülü üzerindeki portun yapılandırılması), 1023
Modbus_Master (SIPLUS I/O veya PtP portu üzerinden Modbus RTU Master olarak iletişim), 1026
Modbus örnek programları
Modbus RTU Master, 1042
Modbus RTU Slave, 1044
Modbus RTU
Genel bakış, 1020
Maksimum Modbus Slave sayısı, 1022
Modbus RTU, eski yordamlar
MB_COMM_LOAD (Modbus RTU için PtP modülü üzerindeki portun yapılandırılması), 1086
MB_MASTER (Modbus Master olarak PtP portu üzerinden iletişim), 1088
MB_SLAVE (Modbus Slave olarak PtP portu üzerinden iletişim), 1094
Modbus RTU, yordamlar
Sürümler, 1022
Modbus_Comm_Load, 1023
Modbus_Master, 1026
Modbus_Slave, 1032-1033
MODBUS TCP
Genel bakış, 957
Maksimum bağlantı sayısı, 957
Modbus TCP, eski yordamlar
MB_CLIENT yordamları (PROFINET üzerinden Modbus TCP istemcisi olarak iletişim), 1070
MB_SERVER (Modbus TCP sunucusu olarak PROFINET üzerinden iletişim), 1076
Modbus TCP, yordamlar
Sürümler, 958
MB_CLIENT, 958
MB_SERVER, 968
MB_RED_CLIENT, 977
MB_RED_SERVER, 995
ModuleStates, 425
ModuleStates örneği, 426
Modüller
Karşılaştırma tablosu, 31
Sinyal kartı (SB), 31
İletişim kartları (CB), 31
Sinyal modülleri (SM), 32
İletişim işlemcisi (CP), 32
İletişim modülü (CM), 32
Termik alan, 39
Termik alan, 42
Parametrelerin yapılandırılması, 147
Parametrelerin yapılandırılması, 162
Modüllerde CPU iletişiminin kaybı, 75

Modül taşıyıcısı veya istasyon hatası OB'si ("Rack or station failure OB"), 76
Montaj
Sinyal modülleri (SM), 32
Direktifler, 38
Genel bakış, 38
Direktifler, 38
Hava akışı, mesafe, soğutma ve termik alan, 39
Soğutma, 39
Hava akışı, 39
Mesafe, 39
Termik alan, 39
Termik alan, 42
Montaj boyutları, 42
Hava akışı, mesafe, soğutma ve termik alan, 42
Boyutlar, 42
Genel bakış, 44
Genel bakış, 44
CPU, 45
CPU, 45
Kart, iletişim (CB) ve sinyal (SB), 47
Sinyal kartı (SB), 47
İletişim kartı (CB), 47
Modül, iletişim (CM) ve sinyal modülü (SM), 48
Sinyal modülü (SM), 48
Modül, iletişim (CM) ve sinyal modülü (SM), 50
İletişim modülü (CM), 50
Klemens bloğu, 51
Klemens bloğu, 51
Uzatma kablosu, 52
Kablaj ve topraklama direktifler, 54
Kablaj ve topraklama direktifler, 54
Potansiyel ayırmayla ilgili direktifler, 55
Potansiyel ayırmayla ilgili direktifler, 55
Topraklama direktifleri, 55
Topraklama direktifleri, 55
Kablaj direktifleri, 56
Kablaj direktifleri, 56
Lamba yükleriyle ilgili direktifler, 57
Lamba yükleriyle ilgili direktifler, 57
Endüktif yüklerle ilgili direktifler, 58
Endüktif yüklerle ilgili direktifler, 58
MOVE_BLK (aralığı kopyala), 236
MOVE (değeri kopyala), 236
MUL (çarp), 229
Müşteri desteği, 4
MUX (çokla), 306

N

N_TRIG (VKE"nin negatif sinyal kenarı için sorgulanması), 210
N (İşlemcinin negatif sinyal kenarı için sorgulanması), 209
N=-Kutu ve N bobini (İşlemcinin negatif sinyal kenarında ayarlanması), 209
NE_ElemType (Bir ARRAY elemanının veri tipini, bir değişkenin veri tipiyle EŞİTSİZLİK açısından karşılaştır), 226
NE_Type (veri tipini, bir değişkenin veri tipiyle EŞİTSİZLİK açısından karşılaştır), 226
NEG (ikili tümleyen oluştur), 230
Network Time Protocol (NTP), 579
NO kontak/NC kontak"; "Kontak, 204
NO kontak ve NC kontak"; "Bobin, 206
Noktadan noktaya iletişim
Serbestçe programlanabilir iletişim, 892
Programlama, 931
NORM_X (standartlaştır), 277
NOT_NULL (işaretçi EŞİT DEĞİLDİR sıfır sorgusu), 227
NOT_OK (geçersizliği kontrol et), 225
NOT (VKE'yi ters çevir), 205

O

OB'lerin öncelik sınıfı
Genel bakış, 68
Olay yürütme ve bekleme kuyruğu, 82
OB'ler, 68
OK (geçerliliği kontrol et), 225
Okuma
DB'lerden, G/Ç'lerden veya bellekten, 190
DB'lerden, G/Ç'lerden veya bellekten, 260
HTTP değişkenleri, 848
Olay yürütme ve bekleme kuyruğu, 81
Ölçüm gerilimleri, 1168, 1168
Ölçümler, takip görevleri, 1151
OPC UA
Sertifikalar (önerilen), 560
Yöntem çağrıları, 791
OPC UA lisansları, 1364

OPC UA sunucusu
Etkinleştirme, 765
Konfigürasyon ayarları, 768
Güvenliğe genel bakış, 770
Desteklenen güvenlik direktifleri, 771
Güvenilir istemciler, 773
Kullanıcı kimlik doğrulaması, 774
Sunucu arayüzünün tanımlanması, 776
Veri tipleri ve katılımcı tipleri, 776
PLC katılımcıları, 777
Arayüzler, 778
Yardımcı spesifikasyonlar, 778
OPC UA yöntem çağrıları, 791
Open User Communication
Sertifikalar (önerilen), 560
Opsiyon yönetimi (konfigürasyon kumandası), 134
Optimize edilmiş veri blokları, 177
Organizasyon bloğu
Genel bakış, 60
Yol almayı işleme, 67
Öncelik sınıfları, 68
Çağırma ve fonksiyonlar, 68
Uyandırma alarmı, 70
Geçici belleğe atama, 96
Teknik bilgi koruması, 160
Doğrusal ve yapılandırılmış programlama, 170
Kullanıcı programındaki kod bloklarının çağrılması, 172
Düzenleme, 173
oluşturma, 174
Birden fazla çevrim OB, 174
Çalışma şeklinin yapılandırılması, 174
Başlangıç bilgisini RD_SINFO ile oku, 397
Örnekler, çalışma zamanı esnasında kullanılan karakter dizisi yordamları
GetSymbolName, 341
GetSymbolPath, 345
GetInstanceName, 347
GetInstancePath, 349
GetBlockName, 351
Örnekler, çeşitli
Analog değer işleme, 98
Bir değişken veri tipinin Slice'ı, 109
AT değişken bindirmesi, 110
Konfigürasyon kumandası (opsiyon yönetimi), 141
SCL'de ENO değerlendirmesi, 193
Array elemanlarına erişim, 271
Analog değer işleme, 278
İç içe geçirilmiş CASE yordamları, SCL, 298
Reçete, 451
Reçete, 458
Veri protokolü programı, 481
hızlı sayaç, 524
Seçilen blokların RUN işletim durumunda yüklenmesi, 1143
Takip ve lojik analizörü fonksiyonu, 1151
CPU 1217C diferansiyel giriş ve uygulama, 1242
CPU 1217C diferansiyel çıkış ve uygulama, 1243
S7-1200 IO-Link-Master bağlantısı, 1292
Güç bilançosunun hesaplanması, 1352
Örnekler, daha eski sistemlerdeki Modbus RTU
Master programı, 1100
Slave programı, 1102
Örnekler, eski sistemlerdeki Modbus TCP yordamları
MB_HOLD_REG parametresi, 1077
Tutma kaydı adresleme, 1079
Çoklu TCP bağlantıları için MB_SERVER, 1081
MB_CLIENT: Ortak Modbus TCP bağlantısında çoklu talepler, 1082
Farklı TCP bağlantılarında çoklu talepler için MB_CLIENT, 1083
MB_CLIENT Çıkışların proses görünümü için yazma talebi, 1084
Çoklu Modbus TCP talebini koordine etmek MB_CLIENT, 1084

Örnekler, iletişim
PROFINET iletişim protokolleri, 582
Verilerin gönderilmesi ve alınması için kullanılan iki ayrı bağlantılı CPU iletişimi, 586
Verilerin gönderilmesi ve alınması için kullanılan ortak bir bağlantılı CPU iletişimi, 587
TSEND_C veya TRCV_C bağlantıları üzerinden CPU iletişimi, 588
T_CONFIG, IP parametrelerinin değiştirilmesi, 676
T_CONFIG, IP parametreleri ve PROFINET IO cihaz adının değiştirilmesi, 677
T_CONFIG, NTP sunucularının IP parametrelerinin değiştirilmesi, 679
IO cihazı ve IO kontrolörü olarak I-Device, 696
Shared Device, 705
Shared I-Device, 711
ASi Slave adresleme, 744
PROFINET SIMATIC S7 bağlantısının yapılandırılması, 759
PROFIBUS SIMATIC S7 bağlantısının yapılandırılması, 760
Örnekler, Modbus
Modbus TCP, MB_CLIENT bağlantı parametresi, 963
Modbus TCP, MB_SERVER bağlantı parametresi, 969
Modbus TCP, MB_HOLD_REG parametrelerine örnekler, 970
Modbus TCP, tutucu sekme adresi, 973
Çoklu TCP bağlantıları için MB_SERVER, 1015
Ortak TCP bağlantısında çoklu talepler için MB_CLIENT, 1017
Farklı TCP bağlantılarında çoklu talepler için MB_CLIENT, 1018
Modbus TCP MB_CLIENT Çıkışların proses görünümüne yazma talebi, 1019
Birden fazla talebi koordine etmek için Modbus TCP MB_CLIENT , 1019
Bir Modbus RTU Master için örnek program, 1042
Modbus RTU Slave için örnek program, 1044
Örnekler, Modbus RTU için eski yordamlar
MB_HOLD_REG parametresi, 1095
Tutma kaydı adresleme, 1097
Örnekler, PtP iletişimi
Mesaj için başlama koşulu, 901
Mesaj bitişi için koşul, 904
Mesaj içindeki mesaj uzunluğu, 905
Receive_Config, 917
Terminal emülatörü, 933
Konfigürasyon, 934
STEP 7 programlama, 940
Terminal emülatörü, 941
Terminal emülatörü için örneğin yürütülmesi, 942
Eski sistemlerdeki PtP iletişimi, RCV_CFG, 1051
Örnekler, USS iletişimi
USS iletişim hataları mesajı, 951
Eski sistemlerde USS iletişim hataları mesajı, 1068
Örnekler, Web sunucusu
Mobil cihaz üzerinden erişim, 805
Değişkenleri oku, 845
Takma adlar, 845
Değişkenleri yaz, 846
Özel değişkenleri oku, 848
Özel değişkenlerin yazılması, 850
Takma adlar, 851
Enum tipleri, 851
Enum tipleri, 852
DB fragmentleri, 855
AWP deklarasyonlarının bağlanması, 855
AWP komutlarındaki özel karakterler, 856
kullanıcı tanımlı web sitesi, 864
Değişkenleri oku, 866
Enum tipleri, 867
Değişkenleri yaz, 868
Özel değişkenlerin yazılması, 869
kullanıcı tanımlı web sitesi, 869
Dil ayarı için kullanıcı tanımlı web sitesi, 877
Fragment denetimi için STEP 7 programı, 882

Örnekler, yordamlar
PEEK ve POKE varyantları, 190
Zaman bobinleri, 213
LIMIT (limiti ayarla), 234
Deserialize, 240
Serialize, 242
SWAP (sırayı değiştir), 259
PEEK ve POKE varyantları, 260
RUNTIME (program çalışma zamanını ölç), 295
CONTINUE, SCL, 302
EXIT, SCL, 302
GOTO (SCL), 303
RETURN, SCL, 303
DECO (şifreyi çöz), 306
SHL (sola kaydır), SCL, 309
ROR (sağa döndür), SCL, 310
S_CONV (karakter dizisini dönüştür), 326
STRG_VAL (karakter dizisini sayı değerine dönüştür), 327
VAL_STRG (sayı değerini karakter dizisine dönüştür), 328
ATH (ASCII karakter dizisini onaltılık sayıya dönüştür), 331
HTA (onaltılık sayıyı ASCII karakter dizisine dönüştür), 332
SET_CINT uyandırma alarmının yürütülmesi ve zaman parametreleri, 385
DeviceStates, PROFIBUS ve PROFINET, 422
ModuleStates, PROFIBUS ve PROFINET, 426
GET_DIAG ve modlar, 434
CTRL_PWM, 448
CTRL_HSC_EXT, 511
TM_MAIL, 1111
Örnek veri bloğu, 93, 175
Otomatik anlaşma, 574
Oturum açma/oturumu kapatma, standart web siteleri, 811
OUT_Range (değer, aralık dışında), 225
Özel karakterler
Kullanıcı tanımlı web siteleri, 856

P

P_TRIG (VKE'nin yükselen sinyal kenarı için sorgulanması), 210
P (İşlemcinin yükselen sinyal kenarı için sorgulanması) , 209
P=-Kutu ve P bobini (işlemcinin yükselen sinyal kenarında ayarlanması), 209
P3964_Config (3964(R) protokolü yapılandırması), 919
Error, 920
Paneller (HMI), 32
Parametreler BUFFER, SEND_P2P, 923
Parametrelerin konfigürasyonu
Gönderim, 687
Alım, 688
SEND_P2P için LENGH ve BUFFER, 923
Parametreler LENGTH, SEND_P2P, 923
Parite, 896
Pasif/aktif iletişim
Partnerlerin yapılandırılması, 564
Bağlantı ID'leri, 586
Parametre, 589
Partnerlerin yapılandırılması, 755
PEEK, PEEK_WORD, PEEK_BOOL, PEEK_DWORD, PEEK_BLK, 190, 260
Performans zamanları, 1174, 1186, 1200, 1214, 1229
PG/PC ve HMI iletişimi için kullanılan işletim türü, 153
PID
PID_Compact (entegre ayarlı üniversal PID kontrolör), 550
PID_3STEP (valflar için ayarlı PID kontrolör), 551
PID_Temp (sıcaklık ayarı için üniversal PID kontrolör), 552
PID valf ayarı, 551
PLC
Fonksiyonlara genel bakış, 27
Çevrim süresi , 84
Çevrim süresi konfigürasyonu, 86
İletişim yükü, 86
Değişken tablosu, 93
Modüller ekleme, 133
Bir otomasyon çözümünün tasarlanması, 169
Bir IP adresinin bir çevrimiçi CPU'ya atanması, 570
Çevrimiçi olma, 1117
PLC cihazı
Blokların kullanılması, 170
PM 1207 akım besleme modülü, 1348
PN-Slave
D_ACT_DP ile etkinleştirme veya devre dışı bırakma, 362

POKE, POKE_BOOL, POKE_BLK, 190, 260
Port_Config (iletişim parametrelerinin dinamik olarak yapılandırılması), kullanımdan kaldırılmıştır, 1046
Port_Config (port yapılandırması), 910
Portal görünümü, 35
Port konfigürasyonu, 895
  3964(R), 906
  Error, 912
  PtP örnek programı, 934
  Error, 1047
Port numaraları
  İletişim partnerlerine atama, 582
  kısıtlı, 681
Potansiyel ayırmayla ilgili direktifler, 55
Potansiyometre modülü, 1346
PROFIBUS
  DPNRM_DG (bir DP-Slave'in teşhis verilerini oku), 378
  DeviceStates örneği, 422
  ModuleStates örneği, 426
  İletişim bağlantısı sayısı, 557
  Ağ bağlantısı, 563
  Master ve Slave, 734
  CM 1242-5, DP-Slave ve DP-Master olarak kullanım, 735
  CM 1243-5 (DP-Master) eklenmesi, 737
  DP-Slave eklenmesi, 737
  Ağ bağlantısı, 737
  Adres ataması, 738
  S7 bağlantısı, 754
PROFIBUS, yordamlar
  RDREC ve WRREC (veri setlerini oku/yaz), 352
  RALRM (alarmı al), 359
  DPWR_DAT (bir DP Standart Slave'in tutarlı verilerini yaz) ve DPRD_DAT (Bir DP Standart Slave'in tutarlı verilerini oku) , 371
  GET (uzak CPU'dan veri oku) ve PUT (uzak CPU'ya veri yaz), 750
PROFIenergy, 380
Profil OB'si, 78
PROFINET
  IP adresinin yapılandırılması, 145
  Saat senkronizasyonu, 145
  DeviceStates, örnek, 423
  ModuleStates, örnek, 428
  İletişim türleri, 553
  İletişim bağlantısı sayısı, 557
  Ağ bağlantısı, 563
  IP adresi, 571
  MAC adresi, 571
  Ethernet adresinin özellikleri, 572
  Bir ağın test edilmesi, 575
  Saat senkronizasyonu özelliği, 580
  Sistemin başlatma saati, 581
  Cihaz adları ve adresleme, 581
  IP adresinin atanması, 581
  Genel bakış, 582
  Geçici mod, 584
  Bağlantı ID'leri, 586
  Bağlantıyı sonlandırma ve yeni bağlantı kurma, 639
  CPU ve HMI cihazı arasındaki iletişimin yapılandırılması, 684
  İki cihaz arasındaki ağ bağlantısı, 685
  PLC/PLC iletişimi, 685
  CPU-CPU iletişimi, 685
  İki cihaz arasındaki ağ bağlantısı, 686
  Ağ bağlantısı, 689
  S7 bağlantısı, 754
PROFINET, eski yordamlar
  TSEND_C ve TRCV_C, 610
  Eski TCON, TDISCON, TSEND ve TRCV yordamları, 631

PROFINET, yordamlar
RDREC ve WRREC (veri setlerini oku/yaz), 352
RALRM (alarmı al), 359
DPWR_DAT (bir DP Standart Slave'in tutarlı verilerini yaz) ve DPRD_DAT (Bir DP Standart Slave'in tutarlı verilerini oku), 371
RCVREC (veri setini al), 373
PRVREC (veri setini hazırla), 376
TSEND_C ve TRCV_C (verileri Ethernet üzerinden gönder ve al), 601
TCON, TDISCON, TSEND ve TRCV, 616
T_RESET, 639
T_DIAG, 641
TUSEND ve TURCV, 665
T_CONFIG (arayüzü yapılandır), 670
TRCV_C, 688
GET (uzak CPU'dan veri oku) ve PUT (uzak CPU'ya veri yaz), 750
PROFINET IO
bir cihazın eklenmesi, 689
bir CPU ve cihaz adlarının atanması, 690
Cihaz adlarını çevrimiçi olarak atama, 1119
PROFINET IO cihazları
Tüm çıkışları SETIO ile yaz, 355
Girişlerin parçasını GETIO_PART ile oku, 356
Çıkışların bölümünü SETIO_PART ile yaz, 358
PROFINET portu
Otomatik anlaşma, 574
PROFINET RT, 582
Program
Öncelik sınıfları, 68
Memory card, 112
Şifre koruması, 160
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Doğrusal ve yapılandırılmış programlar, 170
Kullanıcı programındaki kod bloklarının çağrılması, 172
Organizasyon blokları (OB'ler), 173
cihaza yükleme, 194
Blokların bir çevrimiçi CPU'dan kopyalanması, 199
Program çevrimi OB'si, 69
Program kartı
Genel bakış, 112
CPU'ya ekleme, 113
Yol alma parametrelerinin yapılandırılması, 115
Oluşturma, 120
Program kumandası (SCL), 296
IF-THEN, 297
CASE, 298
FOR, 299
WHILE, 300
REPEAT, 301
CONTINUE, 301
EXIT, 302
GO TO, 303
RETURN, 303
Programlama
CPU'nun işletim durumları, 64
Öncelik sınıfları, 68
Belirtilmemiş CPU, 131
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
Doğrusal program, 170
Yapılandırılmış program, 170
Kullanıcı programındaki kod bloklarının çağrılması, 172
Fonksiyon (FC), 174
LAD (kontak planı), 183
FBD (fonksiyon planı), 184
SCL (Structured Control Language), 184
SCL (Structured Control Language), 185
SCL (Structured Control Language), 186
Sinyal akışı (EN ve ENO), 192
Sistem saati, 314
RTM (işletim saati sayacı), 318
PID_Compact (entegre ayarlı üniversal PID kontrolör), 550
PID_3STEP (valflar için ayarlı PID kontrolör), 551
PID_Temp (sıcaklık ayarı için üniversal PID kontrolör), 552
PtP yordamları, 931
Kod bloklarının karşılaştırılması ve senkronize edilmesi, 1129
Programlama bilgileri
Çağrı yapısında, 203
Programlama editörü
Durumu denetleme, 1133
Program yapısı, 172
Programın izlenmesi, 201
Programın test edilmesi, 201
Programın yürütülmesi, 60

Proje
Aktarım kartı, 116
Program kartı, 120
Erişim koruması, 156
Kod bloğunun korunması, 160
Bir CPU, bir memory card veya bir şifre ile bağlantı, 161
cihaza yükleme, 194
Kod bloklarının karşılaştırılması ve senkronize edilmesi, 1129
Proje görünümü, 35, 35
Proses alarmı OB'si, 71
Proses görünümü
Girişleri GETIO ile oku, 354
Çıkışları SETIO ile yaz, 355
Proses görünümü aralığını GETIO_PART ile oku, 356
Proses görünümü aralığını SETIO_PART ile aktar, 358
Değerleri zorlama, 1138
Force fonksiyonu, 1139
Protokol
PROFINET RT, 582
TCP ve ISO on TCP, 582
UDP, 582
İletişim, 892
Serbestçe programlanabilir iletişim, 892
Modbus, 892
USS, 892
Protokoller, iletişim, 585
PRVREC (veri setini hazırla), 376
PTO (darbe dizisi)
CTRL_PWM (darbe süresi modülasyonu), 439
CTRL_PTO (darbe dizisi), 441
Çalışma şekli, 444
Darbe kanallarını yapılandırma, 446
Zorlanamaz, 1139
PtP iletişimi, 892
Bağlantıların yapılandırılması, 895
Parametrelerin yapılandırılması, 898
Yordam, geri dönüş değerleri, 909
Hata sınıfları, 910
Programlama, 931
eski koşul/hata sınıfları, 1045
PtP iletişimi, 3864(R)
Bağlantıların yapılandırılması, 906
PtP iletişimi, 3964(R)
Öncelik ve protokol parametrelerinin yapılandırılması, 907
PtP iletişimi, örnek program, 933
Konfigürasyon, 934
STEP 7 programlama, 940
Terminal emülatörü, 941
yürütülme, 942
PtP örnek programı için terminal emülatörü, 941
PUT (verileri çıkartılmış CPU'ya yaz), 565
Bağlantının yapılandırılması, 565
PWM (darbe süresi modülasyonu)
CTRL_PWM (darbe süresi modülasyonu), 439
CTRL_PTO (darbe dizisi), 441
Çalışma şekli, 444
Darbe kanallarını yapılandırma, 446
Çevrim süresi ve darbe süresi, 447
G/Ç adresleri, 449
Darbe süresini değiştirme, 449
Çevrim süresini değiştirme, 449
Zorlanamaz, 1139

Q

QRY_CINT (uyandırma alarmı parametrelerini sorgula), 386
QRY_DINT (gecikme alarmının durumunu sorgula), 391
QRY_TINT (saat alarmının durumunu sorgula), 390

R

R_TRIG (değişkenin yükselen sinyal kenarında ayarlanması), 210
R (Çıkışı sıfırla), 207
RALRM (alarmı al), 359
RCV_CFG (seri alım parametreleri dinamik yapılandırma), kullanımdan kaldırılmıştır, 1049
RCV_PTP (alım mesajlarını etkinleştirme), eskimiş, 1055
RCV_RST (alım tamponunun silinmesi), kullanımdan kaldırılmıştır, 1056
RCVREC (veri setini al), 373
RD_ADDR (G/Ç adreslerini donanım kodu aracılığıyla belirle), 500
RD_LOC_T (yerel saati oku), 314
RD_SINFO (güncel OB'nin başlangıç bilgisini oku), 397
RD_SYS_T (saati oku), 314
RDREC (veri setini oku), 352, 367

RE_TRIGR (çevrim denetleme süresini yeniden başlat), 289
READ_BIG (verileri Big Endian formatında oku), 262
READ_DBL (yükleme belleğindeki veri bloğundan oku), 490
READ_LITTLE (verileri Little Endian formatında oku), 262
Receive_Config (alım konfigürasyonu), 915
Receive_P2P (noktadan noktaya verilerin alımı), 924
Receive_Reset (alım tamponunun silinmesi), 926
Reçete
  Genel bakış, 450
  DB yapısı, 451
  RecipeExport (reçeteyi dışa aktarma), 454
  RecipeImport (reçeteyi içe aktarma), 456
  Örnek program, 458
REPEAT, SCL, 301
REPLACE (bir karakter dizisindeki karakteri değiştir), 338
RESET_BF (Bit alanını sıfırla), 207
RET (geri dönüş değeri), 285
RETURN, SCL, 303
RIGHT (bir karakter dizisinin sağ karakterlerini oku), 335
ROL (sola döndür) ve ROR (sağa döndür), 310
ROUND (sayıyı yuvarla), 275
RS (Flipflop 'un sıfırlanması/ayarlanması), 208
RS232- ve RS485 iletişim modülleri, 889
RS485 fişi
  Sonlandırma, 890
RT (zamanı sıfırlama), 211
RTS (donanım akış kontrolü, PtP), 897
RTS açma gecikmesi, kapatma gecikmesi, 898
Ruhsatlar
  CE, 1160
  cULus, 1161
  FM, 1162
  ATEX, 1163
  CCCEx ruhsatı, 1164
  Avustralya ve Yeni Zelanda - RCM Mark, 1164
  Kore sertifikalandırması, 1164
  Denizcilik sektörü, 1165
RUN işletim durumu, 64
  STOP işletim durumuna geçiş, 92
  Force fonksiyonu, 1139
RUN işletim durumunda düzenleme, 1140
RUN işletim durumunda test etme, 1140, 1148
RUN işletim durumunda yükleme
  Genel bakış, 1140
  Ön koşullar, 1141
  STEP 7'den, 1142
  Seçilen blokların yüklenmesi, 1143
  Derleme hataları, 1144
  Genişletilmiş blok arayüzü, 1145
  Yeniden başlangıç olmadan yükleme, 1145
  Bellek rezervi ve kalıcı bellek rezervi, 1145
  Bellek rezervi için global ayarlar, 1146
  Sınırlamalar, 1147
  Yükleme başarısız oldu, 1147
  Güvenlik hususları, 1148
RUNTIME (program çalışma zamanını ölç), 294

S

S_CONV (karakter dizisini dönüştür), 321
S_MOV (karakter dizisini kaydır), 320
S (Çıkışı ayarla), 207
S7-1200'ün performans özelliklerinin genişletilmesi, 31
S7 iletişimi (bağlantının yapılandırılması), 565
S7-Routing, 729
Saat
  CPU gerçek zaman saati, 92
  T_COMBINE (zamanları bağla), 313
  Okuma ve ayarlama yordamları, 314
Saat OB'si, 76
Saat senkronizasyonu
  Gerçek zaman saatinin ayarlanması, 167
  Network Time Protocol (NTP), 580
Sabit uzunluk, 903

Sayaç
Adet, sadece bellek kapasitesiyle sınırlıdır, 30
Bellek boyutu, 30
CTU (ileriye doğru sayım), 218-219
CTD (geriye doğru sayım), 218-219
CTUD (ileriye ve geriye doğru sayım), 218-219
İşlem (standart sayaç), 220
CTRL_HSC_EXT (hızlı sayaca kumanda et, geliştirilmiş), 510
HSC konfigürasyonu, 525
CTRL_HSC (hızlı sayaca kumanda et), 537
Adet, 1175
Boyut, 1175
Sayma türleri, hızlı sayaç, 527
Sayılar
İkili, 100
Tamsayı, 101
Real, 101
SCALE_X (ölçekle), 277
SCATTER, 245
SCATTER_BLK, 248
SCL (Structured Control Language)
Blokların çağrılması, 172
Genel bakış, 184
Programlama editörü, 185
Var-Değişken bölümü, 185
FB veya FC'nin çağrılması, 186
Operatörlerin önceliği, 186
Adresleme, 186
Kumanda yordamları, 186
Koşullar, 186
Operatörler, 186
İfadeler, 186
EN ve ENO (sinyal akışı), 192
Bit bağlantısı, 204
Zamanlar, 211
Değerlerin karşılaştırılması, 224
Dönüştürme yordamları, 273
Program kumandası, 296
Kumanda yordamları, 296
Secure PG/PC ve HMI iletişimi, sertifikalar (önerilen), 560
SEL (seç), 306
SEND_CFG (seri aktarım dinamik yapılandırma), kullanımdan kaldırılmıştır, 1048
Send_Config (gönderim konfigürasyonu), 913
SEND_P2P (noktadan noktaya verilerin gönderimi), 921
Parametreler LENGH ve BUFFER, 923
SEND_PTP (gönderim tamponu aktarımı), 1053
Senkronizasyon
Gerçek zaman saati, 167
Çevrimiçi ve çevrimdışı CPU, 197
hızlı sayaç (HSC), 519
Zaman, Network Time Protocol (NTP), 580
Serbestçe programlanabilir iletişim için protokol, 892
Serialize, 241
Seri iletişim, 892
Sertifikalar, TLS, 595
Sertifika parametreleri, 560
Servis ve destek, 4
SET_BF (Bit alanını ayarla), 207
SET_CINT (uyandırma alarmını parametrelendir), 384
Set_Features (genişletilmiş fonksiyonların ayarlanması), 930
SET_TIMEZONE (zaman dilimini ayarla), 316
SET_TINTL (bir saat alarmının tarih ve saatini belirle), 387
SETIO, 355
SETIO_PART, 358
SGN_GET (RS232 sinyallerinin çağrılması), 927
SGN_GET (RS232 sinyallerini sorgula), kullanımdan kaldırılmıştır, 1057
SGN_SET (RS-232 sinyallerinin ayarlanması), kullanımdan kaldırılmıştır, 1058
Shared Device
Konsept, 702
Konfigürasyon, 705
Shared I-Device, konfigürasyon, 711
SHL (sola kaydır) ve SHR (sağa kaydır) yordamları, 309
Siemens güvenlik sertifikası, web siteleri, 813
Siemens teknik desteği, 4
Signal_Set (RS232 sinyallerinin ayarlanması), 928
Simülatörler, 1344
SIN (sinüs değerini oluştur), 234
Sinyal işleme hatası, 927, 928, 1058, 1059

Sinyal kartları (SB'ler) ve RTD sinyal modülleri (SM'ler)
Analog giriş gerilimi, 1272
Analog giriş akımı, 1272
Analog çıkış akımı ve gerilimi, 1273
Analog giriş gerilimi, 1316
Analog giriş akımı, 1317
Analog çıkış akımı ve gerilimi, 1318
Sinyal kartları (SB)
Genel bakış, 31
Sökme, 47
Montaj, 47
Modüller ekleme, 133
Parametrelerin konfigürasyonu, 147
Parametrelerin konfigürasyonu, 162
SB 1221 DI 4 x 24 V DC, 200 kHz, 1301
SB 1221 DI 4 x 5 V DC, 200 kHz, 1301
SB 1222 DQ 4 x 24 V DC, 200 kHz, 1303
SB 1222 DQ 4 x 5 V DC, 200 kHz, 1303
SB 1223 DI 2 x 24 V DC/DQ 2 x 24 V DC, 200 kHz, 1306
SB 1223 DI 2 x 5 V DC/DQ 2 x 5 V DC, 200 kHz, 1306
SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC, 1309
SB 1231 AI 1 x 12 Bit, 1312
SB 1232 AO 1 x 12 Bit, 1314
SB 1231 AI 1 x 16 Bit termo eleman, 1319
SB 1231 AI 1 x 16 Bit RTD, 1323
Güç gereksinimi, 1350
Sinyal modülleri (SM), genel
CPU genişletmesi, 32
Takma ve sökme, 48
Uzatma kablosu, 52
Modüller ekleme, 133
Parametrelerin konfigürasyonu, 147
Parametrelerin konfigürasyonu, 162
Adım yanıt süreleri, 1271
Güç gereksinimi, 1350
Sinyal modülü (SM), tipler
SM 1221, 1244
SM 1222, 1246
SM 1222, 1247
SM 1223, 1253
SM 1223, 1259
SM 1231, 1262
SM 1232, 1266
SM 1234, 1268
SM 1231, 1274
SM 1231, 1280
SM 1278, 1286
Sipariş numaraları
CPU'lar, 1354
Sinyal modülleri, 1354
Sinyal kartları, batarya kartları, 1355
İletişim arayüzleri (CM, CB ve CP), 1356
Soket bağlayıcı ve klemens bağlantıları, 1356
CPU 1214FC, CPU 1215FC, 1357
FS sinyal modülleri, 1357
CSM 1277 Ethernet Switch, 1357
PM 1207 güç kaynağı, 1357
Memory card'lar, 1357
Uzatma kablosu, 1358
Simülatörler, 1358
Klemens blokları, 1358
Uç tutucu, 1358
Programlama yazılımı, 1364
STEP 7, 1364
Sistem belleği Bayt'ı, 89
Sistem saati
Okuma ve ayarlama yordamları, 314
Slice (bir değişken veri tipinin), 108
SM 1231 RTD
Seçenekler tabloları, 1283
Seçenekler tabloları, 1326
SM ve SB
Karşılaştırma tablosu, 31
Cihaz konfigürasyonu, 128
SNMP, etkinleştirilmesi ve devre dışı bırakılması, 731
Soğuk bağlantı noktasının kompanzasyonu, termo eleman, 1277, 1320
Soğutma, 39
Sorgulama mimarisi, 932
Sorgulama mimarisi Master, 932

Sorgulama mimarisi Slave, 932
SQR (karesini al), 234
SQRT (karekökünü al), 234
SR (Flipflop 'un ayarlanması/sıfırlanması), 208
SRT_DINT (gecikme alarmını başlat), 391
SSS, 5
Standart makine projeleri (konfigürasyon kumandası), 134
Standart montaj rayı, 44
Standart veri blokları, 177
Standart web siteleri, 796
  PC üzerinden erişim, 803
  Güvenli erişim, 804
  Yapı, 808
  Oturum açma ve oturumu kapatma, 811
  Giriş, 813
  Start, 814
  İşletim durumu değiştir, 814
  Teşhis, 815
  Teşhis tamponu, 818
  Modül bilgileri, 819
  İletişim, 824
  Değişken durumu, 828
  JavaScript, 886
  Çerez sınırlamaları, 886
Standart web sitesi "Değişken durumu", 828
Standart web sitesi "Giriş", 813
Standart web sitesi "İletişim", 824
Standart web sitesi "Start", 814
STATUS, RDREC, WRREC, RALARM, 367
STEP 7
  Görünüm (portal ve proje), 35
  Sürüm uyumluluğu, 36
  Kod blokları, 60
  CPU'nun işletim durumları, 64
  Öncelik sınıfı, 68
  Çevrim süresi , 84
  Çevrim süresi konfigürasyonu, 86
  İletişim yükü, 86
  Cihazları yapılandırılması, 128
  Cihazların eklenmesi, 130
  Cihazların eklenmesi, 133
  Cihazları yapılandırılması, 145
  Cihazları yapılandırılması, 147
  Şifre koruması, 160
  Cihazları yapılandırılması, 162
  Doğrusal ve yapılandırılmış programlar, 170
  Kod blokları, 172
  Fonksiyon (FC), 174
  Projeyi cihaza yükleme , 194
  Blokların bir çevrimiçi CPU'dan kopyalanması, 199
  Ağ bağlantısı, 563
  Bir IP adresinin bir çevrimiçi CPU'ya atanması, 570
  PROFINET portu, 571
  Ethernet portu, 571
  Cihazların eklenmesi, 689
  PROFIBUS konfigürasyonu, 738
  AS-i Master cihaz konfigürasyonu, 742
  AS-i Slave adresi girilmesi, 745
  PLC ile çevrimiçi olma, 1117
  Kod bloklarının karşılaştırılması ve senkronize edilmesi, 1129
  İzleme tablosu, 1132
  Durumu programlama editöründe izleme, 1133
  Değerleri zorlama, 1138
STEP 7'de kopyalama, kesme ve yapıştırma, 36
STEP 7'nin kullanıcı arayüzü, 35
STEP 7 programlama
  Kullanıcı tanımlı web siteleri, 860
  PtP örnek programı, 940
Stop Bit'leri, 896
STOP işletim durumu, 64
  Çıkışları etkinleştirme, 1137
  Force fonksiyonu, 1139
STP (Programı sonlandır), 290

Strg_TO_Chars (Karakter dizisini CHAR elemanlarından oluşan Array'e dönüştür), 329
STRG_VAL (karakter dizisini sayı değerine dönüştür), 321
String
  Veri tipi, 104
  Verilere genel bakış, 320
  S_MOV (karakter dizisini kaydır), 320
SUB (Çıkar), 229
Sürücüler, MM4 sürücünün kurulumu, 954
SWAP (sırayı değiştir), 259
SWITCH (atlama dağıtma), 284
Sıcak başlatma, 64
Sınırlamalar
  Kullanıcı tanımlı web siteleri, 863
  Web sunucusu, 885

T

T_ADD (zamanları topla), 312
T_COMBINE (zamanları bağla), 313
T_CONFIG (arayüzü yapılandır), 670
T_CONV (zamanları dönüştür ve ayrıştır), 311
T_DIAG, 641
T_DIFF (zaman farkı), 313
T_RESET, 639
T_SUB (zamanları çıkar), 312
Tablet, Web sunucusuna erişim, 805
Takip fonksiyonu, 1150
Takvim, 311
TAN (tanjant değerini oluştur), 234
Tarih
  T_CONV (zamanları dönüştür ve ayrıştır), 311
  T_ADD (zamanları topla), 312
  T_SUB (zamanları çıkar), 312
  T_DIFF (zaman farkı), 313
  T_COMBINE (zamanları bağla), 313
  SET_TIMEZONE (zaman dilimini ayarla), 316
TCON
  Konfigürasyon, 564
  Bağlantı ID'leri, 586
  Bağlantı parametreleri, 589
TCON_Param, 589
TCON, TDISCON, TSEND ve TRCV, 616
TCP
  Bağlantı konfigürasyonu, 564
  Bağlantı konfigürasyonu, 564
  Protokoller, 582
  Geçici mod, 584
  Bağlantı ID'leri, 586
  Parametre, 589
TCP/IP iletişimi, 582
Teknik bilgi koruması
  Şifre koruması, 160
  Web sunucusunun görüntülenmesi, 815
Teknik destek, 4

Teknik veriler, 1160
Genel teknik veriler, 1160
Ruhsatlar, 1160
Endüstriyel ortamlar, 1165
Elektromanyetik uyumluluk (EMU), 1165
Çevresel koşullar, 1167
Ölçüm gerilimleri, 1168
CPU 1211C, 1173
Adım yanıt süreleri (CPU), 1181
CPU 1212C, 1185
Adım yanıt süreleri (CPU), 1193
CPU 1214C, 1199
Adım yanıt süreleri (CPU), 1207
CPU 1215C, 1213
Adım yanıt süreleri (CPU), 1221
CPU 1217C, 1228
Adım yanıt süreleri (CPU), 1238
SM 1221 DI 8 x 24 V DC, 1244
SM 1221 DI 16 x 24 V DC, 1244
SM 1222 DQ 8, 1246
SM 1222 DQ 16, 1247
SM 1223 DI ve DQ, 1253
SM 1223 DI 8 x 120/230 V AC/DQ 8 x röle, 1259
SM 1231 AI 4 x 13 Bit, 1262
SM 1231 AI 8 x 13 Bit, 1262
SM 1231 AI 4 x 16 Bit, 1262
SM 1232 AO 2 x 14 Bit, 1266
SM 1232 AO 4 x 14 Bit, 1266
SM 1234 AI 4 x 13 Bit / AO 2 x 14 Bit, 1268
Adım yanıt süreleri (SM), 1271
Analog giriş gerilimi, 1272
Analog giriş akımı, 1272
Analog çıkış akımı ve gerilimi, 1273
SM 1231 AI 4 x 16 Bit TC, 1274
SM 1231 AI 8 x 16 Bit TC, 1274
Sinyal modülü SM 1231 AI 4 x RTD x 16 Bit, 1280
Sinyal modülü SM 1231 AI 8 x RTD x 16 Bit, 1280
SM 1278 4xIO-Link-Master, 1286
SB 1221 DI 4 x 24 V DC, 200 kHz, 1301
SB 1221 DI 4 x 5 V DC, 200 kHz, 1301
SB 1222 DQ 4 x 24 V DC, 200 kHz, 1303
SB 1222 DQ 4 x 5 V DC, 200 kHz, 1303
SB 1223 DI 2 x 24 V DC/DQ 2 x 24 V DC, 200 kHz, 1306
SB 1223 DI 2 x 5 V DC/DQ 2 x 5 V DC, 200 kHz, 1306
SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC, 1309
SB 1231 AI 1 x 12 Bit, 1312
SB 1232 AO 1 x 12 Bit, 1314
Adım yanıtı süreleri (SB), 1316
Analog giriş gerilimi, 1316
Analog giriş akımı, 1317
Analog çıkış akımı ve gerilimi, 1318
SB 1231 AI 1 x 16 Bit termo eleman, 1319
SB 1231 AI 1 x 16 Bit RTD, 1323
BB 1297, 1328
CB 1241 RS485, 1340
CM 1241 RS232, 1341
CM 1241 RS422/485, 1342
Memory card'lar, 1344
Giriş simülatörleri, 1344
Potansiyometre modülü, 1346
Teknoloji modülü, SM 1278 4xIO-Link-Master, 1286
Teknoloji yordamları, 510
TeleControl
İletişim işlemcileri, 1103
TeleService iletişimi
TM_MAIL (e-posta gönder), 1106
Temel paneller (HMI), 32
Termik alan, 39, 42
Termo eleman
Temel çalışma şekli, 1277
Soğuk bağlantı noktasının kompanzasyonu, 1277
SM 1231 için seçenekler tablosu, 1277
SB 1231 AI 1 x 16 Bit, 1319
Temel çalışma şekli, 1320
Soğuk bağlantı noktasının kompanzasyonu, 1320
SM 1231 için seçenekler tablosu, 1321
SB 1231 filtre seçenekleri tablosu, 1322
Teşhis
Güvenlik olaylarının azaltılması, 91
Standart web siteleri ile görüntüleme, 815
Durum LED göstergeleri, 1113
Çevrim süresi ve bellek kullanımının denetlenmesi, 1128
Teşhis, yordamlar
LED (LED durumunu okuma), 406
Get_IM_Data (tanımlama ve bakım verilerini oku) yordamı, 407
Get_name, 408
DeviceStates (bir G/Ç sisteminin modül durumunu oku) yordamı, 420
ModuleStates (bir modülün modül durumu bilgilerini oku) yordamı, 425
GET_DIAG (teşhis bilgilerini oku), 430
Teşhis hatası OB'si, 73

Teşhis tamponu
Genel bakış, 91
standart web sitelerinde görüntüleme, 818
Olayların görüntülenmesi, 1128
Tetikleme
İzleme tablosundaki değerler, 1136
Takip, 1150
TIA Portal, portal görünümü ve proje görünümü, 35
Time
Time veri tipi, 102
TOD (Time of Day veri tipi), 102
DTL (Date and Time long veri tipi), 103
T_DIFF (zaman farkı), 313
SET_TIMEZONE (zaman dilimini ayarla), 316
TLS sertifikaları, 595
TM_MAIL (e-posta gönder), 1106
TMAIL_C, 645
Topoloji
Görünüm, 35
Halka, 721
TRCV_C (verileri Ethernet üzerinden al, TCP), 584
Geçici mod, 584
Bağlantı ID'leri, 586
Bağlantı parametreleri, 589
TRCV_C (verileri Ethernet üzerinden al, TCP)
Konfigürasyon, 564
TRCV (verileri Ethernet üzerinden al, TCP)
Geçici mod, 584
Bağlantı ID'leri, 586
Parametrelerin konfigürasyonu, 688
Trigonometri yordamları, 234
TRUNC (tamsayı oluştur) , 275
TS adaptörü, 31
TSAP (Transport Service Access Point), 566
Cihazlara atamayla ilgili yordamlar, 582
Tanım, 583
Kısıtlı TSAP'ler ve port numaraları, 681
Genel parametrelerin yapılandırılması, 686
Genel parametrelerin yapılandırılması, 754
TSEND_C (verileri Ethernet üzerinden gönder, TCP), 564
Konfigürasyon, 564
Bağlantı ID'leri, 586
Bağlantı parametreleri, 589
Yordam konfigürasyonu, 687
TSEND (verileri Ethernet üzerinden gönder, TCP)
Bağlantı ID'leri, 586
TURCV (verileri Ethernet üzerinden al, UDP), 564
Konfigürasyon, 564
Bağlantı parametreleri, 589
TUSEND (verileri Ethernet üzerinden gönder, UDP), 564
Konfigürasyon, 564
Parametre, 589
Tutarlılık kontrolü, 203
Tırnak işareti kullanım kuralları, Web sunucusu, 856

U

UDP
Bağlantı konfigürasyonu, 564
Protokol, 582
Parametre, 589
UFILL_BLK (belleği kesintisiz şekilde Bit örüntüsüyle doldur), 244
UMOVE_BLK (aralığı kesintisiz olarak kopyala), 236
UPPER_BOUND (üst ARRAY sınırını oku), 268
USS protokolü için kütüphane
Genel bakış, 942
Yordam sürümleri, 944
Kullanım gereklilikleri, 944
USS_Port_Scan (USS ağı üzerinden iletişim düzenlenmesi), 944
USS_Drive_Control (sürücü ile veri alışverişi), 945
USS_Read_Param (sürücüden parametre okuma), 948
USS_Write_Param (sürücüdeki parametreyi değiştirme), 949
Durum kodları, 950
Uyandırma alarmı OB'si, 70
Uyumluluk, 36
Uyumsuz CPU sürümü nedeniyle hata, 1114
Uzatma kablosu, 52
Montaj, 52
Sökme, 52
Uzunluk, PtP mesajı, 905

V

VAL_STRG (sayı değerini karakter dizisine dönüştür), 321
VARIANT_TO_DB_ANY (VARIANT'ı DB_ANY'ye dönüştür), 280
VariantGet (VARIANT değişken değerini oku), 263
VariantPut (VARIANT değişken değerini yaz), 264

VE (mantıksal bağlantı), 304
ve kutusu (FBD'deki mantıksal VE bağlantısı), 205
Veri aktarımı, başlat, 921, 1053
Veri bloğu
  Genel bakış, 60
  Global DB ile örnek DB arasındaki fark, 93
  Organizasyon blokları (OB'ler), 173
  Genel bakış, 176
  Global veri bloğu, 176
  Optimize edilmiş erişim, 177
  Standart erişim, 177
  CREATE_DB ile oluşturma, 487
  Veri bloklarını yükleme belleğinden oku/yükleme belleğine yaz, 490
  Özniteliği ATTR_DB ile okuma, 493
  DELETE_DB ile silme, 495
  CONF_DATA, 673
  Fragmentlerin kullanıcı tanımlı web sitelerine içe aktarımı, 855
  Değerleri kaydetme ve sıfırlama, 1133
Veri bloğu
  Çevrimiçi ve çevrimdışı CPU'ların başlangıç değerlerinin senkronize edilmesi, 197
Veri kullanımı blokları (DHB'ler), 176
Veri protokolü
  Veri protokollerine genel bakış, 460
  Veri seti yapısı, 461
  oluşturma, 462
  açma, 466
  yazma, 468
  boş, 470
  kapatma, 471
  silme, 472
  yeni dosya oluşturma, 474
  Veri protokollerini görüntüleme, 477
  Boyut sınırlaması ve boyut hesaplama, 479
  Örnek program, 481
Veri tipleri
  Genel bakış, 99
  Bool, Bayt, Word ve DWord, 100
  USInt, SInt, UInt, Int, UDInt, Dint (tamsayı), 101
  Real, LReal (kayar noktalı), 101
  Time, Date, TOD (saat), DTL (Date and Time long), 102
  Karakterler ve karakter dizileri, 104
  Array'ler, 106
  Struct, 106
  PLC veri tipi editörü, 107
  Variant (işaretçi), 107
Veriyolu konektörü, 32
VEYA (mantıksal bağlantı), 304

W

WChar (Word Character veri tipi), 104
Web siteleri
  STEP 7 ile ilgili servis, destek ve dokümantasyon, 5
Web sunucusu
  Sertifikalar (önerilen), 560
  Desteklenen tarayıcılar, 798
  Güncelleme hızı, 799
  Etkinleştirme, 799
  Kullanıcı konfigürasyonu, 801
  Standart web siteleri, 803
  Mobil cihaz erişimi, 805
  CP modülü üzerinden erişim, 806
  Bir mobil cihazda görüntüleme, 809
  Kullanıcı tanımlı web siteleri, 840
  Tırnak işareti kullanım kuralları, 856
  Sınırlamalar, 885
  Maksimum HTTP bağlantısı sayısı, 886
Web sunucusu bağlantılarının maksimum sayısı, 886
Web sunucusu için desteklenen tarayıcılar, 798
Web sunucusuna kablosuz bağlantı, 805
Web sunucusu üzerinden değişkenlerin denetlenmesi, 828
WHILE, SCL, 300
WR_LOC_T (yerel saati ayarla), 314
WR_SYS_T (saati ayarla), 314
WRIT_DBL (yükleme belleğindeki veri bloğuna yaz), 490

WRITE_BIG (verileri Big Endian formatında yaz), 262
WRITE_LITTLE (verileri Little Endian formatında yaz), 262
WRREC (veri setini yaz), 352, 367
WString (Word String veri tipi), 104
WWW (kullanıcı tanımlı web sitelerinin senkronizasyonu), 860

X

x kutusu (FBD'deki mantıksal ÖZEL VEYA bağlantısı), 205
XON / XOFF, 898
XOR (mantıksal bağlantı), 304

Y

Yanlış kutup bağlama koruması, 1169
Yapılandırılmış programlama, blok yapısı, 170
Yardım hattı, 4
Yazma
  DB'lere, G/Ç'lere veya belleğe, 190
  DB'lere, G/Ç'lere veya belleğe, 260
  HTTP değişkenleri, 849
Yaz saati için kullanılan TimeTransformationRule, 317
Yaz saati TimeTransformationRule, 317
Yazılım akış kontrolü, 898
yedekleme
  Opsiyonlar, içerikler ve özel hususlar, 1154
  bir çevrimiçi CPU, 1156
  Dosyaların kaydedilmesi, 1158
  Bir yedekleme kopyasının geri yüklenmesi, 1159
Yedekleme dosyalarının kaydedilmesi, 1158
Yedeklilik
  MRP, 719
  İstemciler, 721
  Etki alanları, 723
Yeni cihaz ekleme
  CPU, 130
  Belirtilmemiş CPU, 131
  Mevcut donanımı algılama, 131
Yeniden başlatma yok, 64
Yeni fonksiyonlar, 33
Yerel ile partner CPU arasındaki bağlantı, 564
Yerel saat
  Okuma ve ayarlama yordamları, 314
Yerel veriler
  OB öncelik kademesi başına maksimum, 96
  Bloklarla kullanım, 96-97
Yol alma
  Yol almayı işleme, 67
  Force fonksiyonu, 1139
Yol alma OB, 69
Yol alma parametreleri, 115
Yönlendiricinin IP adresi, 572
Yordamlar
  Bit bağlantısı, 205
  Ayarlama ve sıfırlama yordamları, 207
  Zaman, 211
  Sayaç, 218-219
  Değerlerin karşılaştırılması, 224
  Aralık değerlendirmesi, 225
  geçerli veya geçersiz gerçek sayılar açısından kontrol et, 225
  Eşittir ve Eşit değildir karşılaştırması, 226
  Sıfır karşılaştırması, 227
  Array kontrolü, 227
  Move yordamları, 236
  Dönüştürme, 272
  Gerçek sayıları yuvarla ve kısalt, 275
  Atlama yordamları, 282
  Program kumandası (SCL), 296
  mantıksal bağlantılar, 304
  Kaydırma ve döndürme yordamları, 309
  Tarih, saat ve saat fonksiyonları, 311
  Karakter dizisi ve karakterler, 320
  Ortak parametreler, 680
  Eski USS durum kodları, 1067

Yordamlar, eskimiş
Port_Config (iletişim parametrelerinin dinamik olarak yapılandırılması), 1046
SEND_CFG (seri aktarım dinamik yapılandırma), 1048
RCV_CFG (seri alım parametreleri dinamik yapılandırma), 1049
SEND_PTP (gönderim tamponu aktarımı), 1053
RCV_PTP (alım mesajlarını etkinleştirme), 1055
RCV_RST (alım tamponunun silinmesi), 1056
SGN_GET (RS232 sinyallerinin sorgulanması), 1057
SGN_SET (RS232 sinyallerinin belirlenmesi), 1058
USS_PORT (USS ağı üzerinden iletişim düzenlenmesi), 1061
USS_DRV (sürücü ile veri alışverişi), 1062
USS_RPM (sürücüden parametre okunması), 1065
USS_WPM (sürücüdeki parametreyi değiştirme), 1066
MB_CLIENT yordamları (PROFINET üzerinden Modbus TCP istemcisi olarak iletişim), 1070
MB_SERVER (Modbus TCP sunucusu olarak PROFINET üzerinden iletişim), 1076
MB_COMM_LOAD (Modbus RTU için PtP modülü üzerindeki portun yapılandırılması), 1086
MB_MASTER (Modbus Master olarak PtP portu üzerinden iletişim), 1088
MB_SLAVE (Modbus Slave olarak PtP portu üzerinden iletişim), 1094

Yordamlar, yürütme hızları, 1174, 1186, 1200, 1214, 1229

Yordamların yürütme hızları, 1174, 1186, 1200, 1214, 1229

Yükleme belleği, 28
Kullanıcı tanımlı web siteleri, 863
CPU 1211C, 1173
CPU 1212C, 1185
CPU 1214C, 1199
CPU 1215C, 1213
CPU 1217C, 1228

Yüklemek için bir çevrimiçi CPU'nun algılanması, 131

Z

Zaman
T_CONV (zamanları dönüştür ve ayrıştır), 311
T_ADD (zamanları topla), 312
T_SUB (zamanları çıkar), 312
Okuma ve ayarlama yordamları, 314
CPU'nun çevrimiçi olarak yapılandırılması, 1120

Zaman hatası OB'si, 72

Zamanlar
Adet, sadece bellek kapasitesiyle sınırlıdır, 30
Bellek boyutu, 30
Yordamlar, 211
Çalışma şekli, 214
Boyut, 1175
Adet, 1175

Zamanı sıfırlama (RT), 211