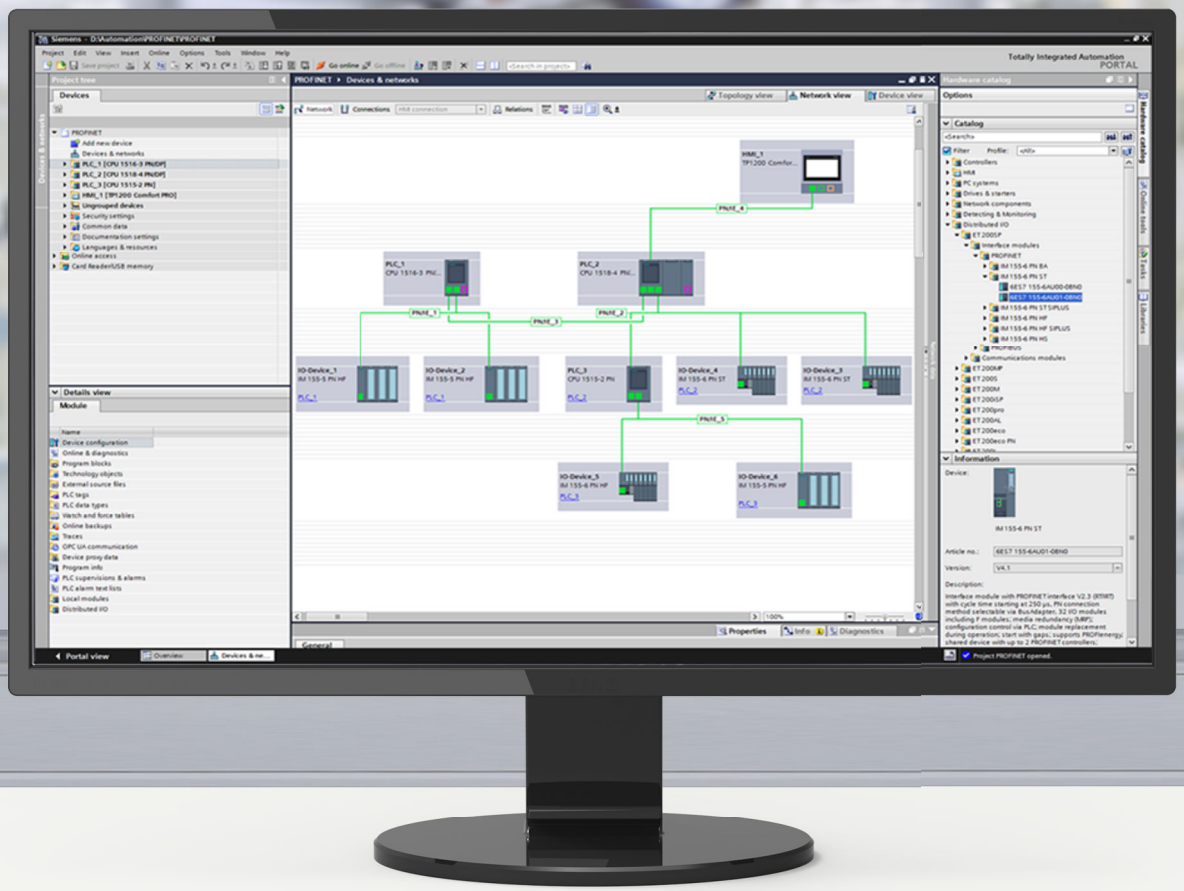
SIEMENS



Ausgabe 11/2023

FUNKTIONSHANDBUCH

SIMATIC

S7-1500, ET 200MP, ET 200SP, ET 200AL, ET 200pro, ET 200eco PN

Kommunikation

support.industry.siemens.com

SIEMENS

SIMATIC

S7-1500, ET 200MP, ET 200SP, ET 200AL, ET 200pro, ET 200eco PN Kommunikation

Funktionshandbuch

11/2023
A5E03735814-AL

Fortsetzung nächste Seite

Fortsetzung

S7-1500, ET 200MP, ET 200SP, ET 200AL, ET 200pro, ET 200eco PN Kommunikation

Funktionshandbuch

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR

bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG

bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT

bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG

bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG

Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

A5E03735814-AL
Ⓟ 10/2023 Änderungen vorbehalten

Inhaltsverzeichnis

Einleitung

1

Zweck der Dokumentation

Das vorliegende Funktionshandbuch vermittelt eine Übersicht über die Kommunikationsmöglichkeiten, die CPUs, Kommunikationsmodule und -prozessoren, und PC-Systeme der Systeme SIMATIC S7-1500, ET 200MP, ET 200SP, ET 200AL, ET 200pro und SIMATIC Drive Controller bieten. Im vorliegenden Funktionshandbuch wird die verbindungsorientierte, asynchrone Kommunikation beschrieben.

Die Dokumentation behandelt im Einzelnen:

- Überblick der Kommunikationsdienste
- Eigenschaften der Kommunikationsdienste
- Anwendertätigkeiten zum Einrichten der Kommunikationsdienste im Überblick

Erforderliche Grundkenntnisse

Zum Verständnis des Funktionshandbuchs sind folgende Kenntnisse erforderlich:

- allgemeine Kenntnisse auf dem Gebiet der Automatisierungstechnik
- Kenntnisse des Industrieautomatisierungssystems SIMATIC
- Kenntnisse im Umgang mit STEP 7 (TIA Portal)

Gültigkeitsbereich der Dokumentation

Die vorliegende Dokumentation gilt als Grundlagendokumentation für alle Produkte der Systeme SIMATIC S7-1500, ET 200MP, ET 200SP, ET 200AL und ET 200pro. Die Produkt-Dokumentationen bauen auf dieser Dokumentation auf.

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 11/2023 gegenüber Ausgabe 11/2022

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
Kapitel "Industrial Cybersecurity"	Durch die Digitalisierung und zunehmende Vernetzung von Maschinen und Industrieanlagen steigt auch die Gefahr von Cyberattacken. Insbesondere bei kritischen Infrastruktureinrichtungen sind entsprechende Schutzmaßnahmen daher Pflicht. Das Kapitel enthält die folgenden Informationen: • Grundlegende Informationen zum Thema Industrial Cybersecurity • Maßnahmen, wie Sie einzelne Komponenten und das gesamte System vor Manipulation und ungewünschten Zugriffen schützen.	Industrial Cybersecurity (Seite 26)

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
Überarbeitung der Tabellen für Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation.	Aktualisierte Informationen zu Protokollen und verwendeten Ports. Sie sehen auf den ersten Blick, welche Voreinstellungen gelten. Dadurch können Sie gezielt nur die Einstellungen anpassen, die für Ihren Anwendungsfall relevant sind.	Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation (Seite 36)
Aktualisierte Informationen zu den Verbindungsressourcen von CPU und HMI.	Aktualisierte Informationen zu den folgenden Verbindungsressourcen: • Maximal unterstützte Verbindungsressourcen für einige CPU-Typen • Maximal belegte Verbindungsressourcen für verschiedene HMI-Geräte	Verbindungsressourcen (Seite 397)
Erweiterung des redundanten Systems mit Kommunikationsprozessoren	Ab STEP 7 V19 können Sie ein redundantes System S7-1500R/H ab FW-Version V3.1 mit den Kommunikationsprozessoren CP 1543-1 erweitern.	System IP-Adressen bei Kommunikationsprozessoren (Seite 418).
Secure Open User Communication mit dem redundanten System	Ab STEP 7 V19 unterstützt ein redundantes System S7-1500R/H ab FW-Version V3.1 auch die Secure Open User Communication.	Open User Communication mit dem redundanten System S7-1500R/H (Seite 428)

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 11/2022 gegenüber Ausgabe 05/2021

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
Überarbeitung der Tabellen für Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation.	Aktualisierte Informationen zu Protokollen und verwendeten Ports. Sie sehen auf den ersten Blick, welche Voreinstellungen gelten. Dadurch können Sie gezielt nur die Einstellungen anpassen, die für Ihren Anwendungsfall relevant sind.	Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation (Seite 36)
Aktivieren/Deaktivieren des SNMP	Je nach FW-Version der S7-1500 CPUs ist das SNMP in den Voreinstellungen aktiviert bzw. deaktiviert. Die Voreinstellungen können Sie nach Bedarf ändern.	SNMP (Seite 114)
Überarbeitung der virtuellen Schnittstelle für IP-basierte Anwendungen	Bei einem CP 1543-1 ab einer Firmware Version V3.0 steht Ihnen die CP-interne Firewall zur Verfügung. Diese dient zur Absicherung des Datenverkehrs über die virtuelle Schnittstelle.	Virtuelle Schnittstelle für IP-basierte Anwendungen (Seite 393)
OPC UA Server: Lesen des Diagnosestatus des eigenen Adressraums	Mit der Nutzung der OPC UA Anweisung zum Lesen ("OPC_UA_ReadList") kann auf den eigenen Namensraum des OPC UA Servers zugegriffen werden. Damit ist es möglich, den Status des eigenen OPC UA Server sowie der Verbindungen von OPC UA Clients, der Session sowie auch von Subscriptions auszulesen und im Anwenderprogramm darauf zu reagieren. So können schnell z. B. Verbindungsprobleme erkannt und die Anlagenverfügbarkeit erhöht werden.	OPC UA-Server im Programm diagnostizieren (Seite 318)

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
OPC UA Server: Zeitstempelung der Source-Zeit von Knoten	Durch die Nutzung der OPC UA Anweisung zum Schreiben ("OPC_UA_WriteList") ist es möglich, sowohl den "SourceTimestamp" als auch den Statuscode einer OPC UA Variablen (Knoten) zu ändern. Damit kann ab der V18 zwischen der "Source"- und "Server"-Zeit unterschieden werden.	Client-Zugriffe und lokale Zugriffe auf den OPC UA-Server (Seite 217)
OPC UA GDS-Mechanismus: Jetzt auch für Webserver-Zertifikate nutzbar	Das Webserver-Zertifikat für die HTTPS-Kommunikation kann nun auch über den OPC UA GDS-Mechanismus, ohne separates Laden der Hardware-Konfiguration, verwaltet werden.	Wissenswertes zum Zertifikatsmanagement (Seite 62) Automatisiertes Zertifikatsmanagement mit GDS (Seite 193)

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 05/2021 gegenüber Ausgabe 11/2019

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
Verbesserte Security für SIMATIC PG/HMI-Kommunikation	• Erlaubt die eindeutige Identifikation jeder PLC basierend auf individuellen Zertifikaten • Bietet zusätzlichen Vertraulichkeitsschutz durch verschlüsselte Kommunikation • Schutz der Konfigurationsdaten durch individuelle Passwörter	Secure PG/HMI-Kommunikation (Seite 103)
Security Wizard für neue PLC Security-Mechanismen	• Schnelle und einfache Konfiguration der neuen Security-Mechanismen der PLC in einen Durchgang • Unterstützende Information, um passende Einstellungen für eigenen Anwendungsfall zu wählen	Schutz vertraulicher Konfigurationsdaten (Seite 74)
Zertifikatsverwaltung über OPC UA (Global Discovery Server, GDS)	• Zertifikatsupdate zur Laufzeit • Unterstützung von CRLs • Zugriffsschutz für Zertifikatsmanagement	Zertifikatsmanagement über Global Discovery Server (GDS) (Seite 193)
CPU-Meldungen an OPC UA Clients übermitteln	• Mit Subscriptions können Clients CPU-Meldungen als "Alarms and Conditions" vom OPC UA-Server der CPU abonnieren • Programmeldungen inkl. Begleitwerte werden vom OPC UA Server bereitgestellt • Quittierungspflichtige Alarme können vom OPC UA-Client quittiert werden (deaktivierbar) • Ein Meldeschwall wird als "Overload" angezeigt und mit der Refresh-Methode können Clients Meldungen nachladen	Meldungen auf dem OPC UA-Server bereitstellen (Seite 295)
Dynamische Zuweisung der Netzwerkkonfiguration mit DHCP	Einsatz der CPU in IT-verwalteten Netzwerken durch folgende Funktionen:	Adressvergabe über DHCP (Seite 366)

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
	• Anschluss der CPU an ein bestehendes Netzwerk ohne zusätzliche manuelle Konfiguration der Netzwerkschnittstelle • Anforderung von Netzwerkparametern für die CPU gemäß RFC 2131 von einem DHCPv4-Server (IP-Adresse und Subnetz-Maske, Default IP-Router-Adresse und weitere optionale Netzwerkparameter wie z. B. DNS- und NTP-Server-Adressen)	
Namensbasierte Adressierung mit DNS	• DNS-Server-Adressen können von der CPU über DHCP bezogen werden • Die CPU kann Host- und Domain-Namen von einem DHCP-Server beziehen für Applikationen, die mit OPC UA oder (Secure) OUC realisiert sind • Die CPU kann konfigurierten Host- oder Domain-Namen an DHCP-Server übertragen, die mit DNS-Server gekoppelt sind für dynamischen Abgleich (Dynamic DNS) • Der NTP-Client der CPU kann NTP-Server mit Namen ansprechen • Netzwerkparameter können geschrieben werden mit der neuen Anweisung "CommConfig", z. B. IP-Adressparameter, DNS-Server, Host and Domain Name	DHCP mit DNS (Seite 370)

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 11/2019 gegenüber Ausgabe 10/2018

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
IP-Forwarding	Einfacher Zugriff von der Leitebene auf die Feldebene für Konfiguration und Parametrierung von Devices, z. B. per PDM oder Webbrowser.	IP-Forwarding (Seite 384)
Erweiterung OPC UA-Server	Für S7-1500-CPUs ab Firmware V2.8 und TIA Portal Version 16 können Sie mit einer entsprechenden Runtime-Lizenz von folgenden Erweiterungen des integrierten OPC UA-Servers profitieren:	Kap. OPC UA-Kommunikation (Seite 163)

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
	• Verbesserte Diagnose: Über Meldungen im Diagnosepuffer, einer OPC UA Kategorie im Online & Diagnose Bereich im TIA Portal und auch einer verbesserten Verbindungsressourcen Anzeige erhält der OPC UA Anwender Informationen zum Status des OPC UA Servers. • Download Verhalten: Der OPC UA Server führt im Betriebszustand RUN bei einem Download aus dem TIA Portal nur einen Neustart aus, wenn die neu geladenen Daten Auswirkungen auf den Datenhaushalt des OPC UA Servers haben. • Server Interface Modellierung: Auch im TIA Portal können jetzt Server Interfaces modelliert werden oder auch OPC UA Companion Spezifikationen importiert und zum PLC Datenhaushalt gemapped werden.	

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 10/2018 gegenüber Ausgabe 12/2017

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
Beschreibung der Kommunikation mit dem redundanten System S7-1500R/H	Sie erhalten Informationen zu den Besonderheiten der Kommunikation mit dem redundanten System S7-1500R/H	Kap. Kommunikation mit dem redundanten System S7-1500R/H (Seite 411)
Erweiterung des Gültigkeitsbereichs des Funktionshandbuchs auf das redundante System S7-1500R/H	Funktionen, die Sie vom Automatisierungssystem SIMATIC S7-1500 her kennen, sind realisiert für das redundante System S7-1500R/H	Systemhandbuch Redundantes System S7-1500R/H (https://support.industry.siemens.com/cs/ww/de/view/109754833)

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 12/2017 gegenüber Ausgabe 09/2016

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
OPC UA Companion Specification	Über OPC UA Companion Specification lassen sich Methoden einheitlich und herstellerunabhängig spezifizieren. Über diese spezifizierten Methoden können Geräte der verschiedensten Hersteller einfacher in die Anlage und in Produktionsabläufe integriert werden.	Kap. OPC UA-Server-Schnittstelle projektieren (Seite 255)
Gesicherte Verbindung zu einem Mailserver über die Schnittstelle der CPU	Sie können eine gesicherte Verbindung zu einem Mailserver aufbauen ohne zusätzliche Hardware.	Kap. Secure OUC über E-Mail (Seite 100)
Gesicherte Kommunikation über Modbus TCP	Sie können gesicherte TCP-Verbindungen zwischen einem Modbus TCP-Client und einem Modbus TCP-Server aufbauen.	Kap. Secure OUC mit Modbus TCP (Seite 98)

Was ist neu im Funktionshandbuch Kommunikation, Ausgabe 09/2016 gegenüber Ausgabe 12/2014

Was ist neu?	Was ist der Kundennutzen?	Wo finden Sie die Informationen?
OPC UA-Server	OPC UA ist ein einheitlicher Standard zum Datenaustausch und ist unabhängig von bestimmten Betriebssystemplattformen. OPC UA nutzt integrierte Sicherheitsmechanismen auf verschiedenen Automatisierungssystemen, z. B. beim Datenaustausch, auf Anwendungsebene, zur Legitimation des Anwenders. Der OPC UA Server stellt zahlreiche Daten bereit: • Werte von PLC-Variablen, auf die Clients zugreifen dürfen • Datentypen dieser PLC-Variablen • Angaben zum OPC UA Server selbst und zur CPU Clients können sich dadurch einen Überblick über den Variablenhaushalt verschaffen und Werte einlesen und schreiben.	Kap. OPC UA-Kommunikation (Seite 163)
Secure Open User Communication	Sicherer Datenaustausch mit anderen Geräten.	Kap. Secure Open User Communication (Seite 86)
Zertifikate-Handling in STEP 7	Sie können in STEP 7 Zertifikate verwalten für die folgenden Anwendungen: • OPC UA Server • Secure Open User Communication • Webserver der CPU	Kap. Zertifikatsmanagement mit TIA Portal (Seite 63)
SNMP für die CPU deaktivieren	Sie können für die CPU SNMP deaktivieren. Das kann unter bestimmten Voraussetzungen sinnvoll sein, z. B. wenn die Sicherheitsrichtlinien in Ihrem Netzwerk kein SNMP zulassen.	Kap. SNMP (Seite 114)

Konventionen

STEP 7: Zur Bezeichnung der Projektier- und Programmiersoftware verwenden wir in der vorliegenden Dokumentation "STEP 7" als Synonym für "STEP 7 ab V12 (TIA Portal)".

Mit "S7-1500 CPUs" sind wegen durchgängiger Controller-Funktionen i. d. R. auch die CPU-Varianten S7-1500F, S7-1500T, S7-1500TF, S7-1500C, S7-1500R/H, S7-1500pro, ET200SP, S7-1500 Software Controller sowie SIMATIC Drive Controller gemeint. Unterschiede ergeben sich z. B. durch unterschiedliche Aufbautechnik und Anwendungen zwischen Advanced Controllern, Distributed Controllern und Software Controllern.

Die vorliegende Dokumentation enthält Abbildungen zu den beschriebenen Geräten. Die Abbildungen können vom gelieferten Gerät in Einzelheiten abweichen.

Beachten Sie außerdem die folgendermaßen gekennzeichneten Hinweise:

HINWEIS

Ein Hinweis enthält wichtige Informationen zum Produkt, zur Handhabung des Produkts oder zu dem Teil der Dokumentation, auf den besonders aufmerksam gemacht werden soll.

Industry Mall

Die Industry Mall ist das Katalog- und Bestellsystem der Siemens AG für Automatisierungs- und Antriebslösungen auf Basis von Totally Integrated Automation (TIA) und Totally Integrated Power (TIP).
Kataloge zu allen Produkten der Automatisierungs- und Antriebstechnik finden Sie im Internet (https://mall.industry.siemens.com).

1.1 Wegweiser Dokumentation Funktionshandbücher

1.1.1 Informationsklassen Funktionshandbücher

Die Dokumentation für das Automatisierungssystem SIMATIC S7-1500, für die auf SIMATIC S7-1500 basierenden CPUs 1513/1516pro-2 PN, SIMATIC Drive Controller und die Dezentralen Peripheriesysteme SIMATIC ET 200MP, ET 200SP, ET 200AL und ET 200eco PN gliedert sich in drei Bereiche.
Die Aufteilung bietet Ihnen die Möglichkeit, gezielt auf die gewünschten Inhalte zuzugreifen. Die Dokumentation finden Sie zum kostenlosen Download im Internet.
(https://support.industry.siemens.com/cs/ww/de/view/109742705)

Basisinformationen

Systemhandbücher und Getting Started beschreiben ausführlich die Projektierung, Montage, Verdrahtung und Inbetriebnahme der Systeme SIMATIC S7-1500, SIMATIC Drive Controller, ET 200MP, ET 200SP, ET 200AL und ET 200eco PN. Für die CPUs 1513/1516pro-2 PN nutzen Sie die entsprechenden Betriebsanleitungen.
Die Online-Hilfe von STEP 7 unterstützt Sie bei der Projektierung und Programmierung.
Beispiele:

- Getting Started S7-1500
- Systemhandbücher
- Betriebsanleitungen ET 200pro und CPU 1516pro-2 PN
- Online-Hilfe TIA Portal

Geräteinformationen

Gerätehandbücher enthalten eine kompakte Beschreibung der modulspezifischen Informationen wie Eigenschaften, Anschlussbilder, Kennlinien, technische Daten.
Beispiele:

- Gerätehandbücher zu CPUs
- Gerätehandbücher zu Interfacemodulen
- Gerätehandbücher zu Digitalmodulen
- Gerätehandbücher zu Analogmodulen
- Gerätehandbücher zu Kommunikationsmodulen
- Gerätehandbücher zu Technologiemodulen
- Gerätehandbücher zu Stromversorgungsmodulen
- Gerätehandbücher zu BaseUnits

Übergreifende Informationen

In den Funktionshandbüchern finden Sie ausführliche Beschreibungen zu übergreifenden Themen rund um den SIMATIC Drive Controller und das Automatisierungssystem S7-1500.
Beispiele:

- Funktionshandbuch Diagnose
- Funktionshandbuch Kommunikation
- Funktionshandbücher Motion Control
- Funktionshandbuch Webserver
- Funktionshandbuch Zyklus- und Reaktionszeiten
- Funktionshandbuch PROFINET
- Funktionshandbuch PROFIBUS

Produktinformation

Änderungen und Ergänzungen zu den Handbüchern werden in einer Produktinformation dokumentiert. Die Produktinformation hat in der Verbindlichkeit Vorrang gegenüber dem Geräte- und Systemhandbuch.
Sie finden die aktuellsten Produktinformationen im Internet:

- S7-1500/ET 200MP (https://support.industry.siemens.com/cs/de/de/view/68052815)
- SIMATIC Drive Controller (https://support.industry.siemens.com/cs/de/de/view/109772684)
- Motion Control (https://support.industry.siemens.com/cs/de/de/view/109794046)
- ET 200SP (https://support.industry.siemens.com/cs/de/de/view/73021864)
- ET 200eco PN (https://support.industry.siemens.com/cs/ww/de/view/109765611)

Manual Collections

Die Manual Collections beinhalten die vollständige Dokumentation zu den Systemen zusammengefasst in einer Datei.
Sie finden die Manual Collections im Internet:

- S7-1500/ET 200MP/SIMATIC Drive Controller (https://support.industry.siemens.com/cs/ww/de/view/86140384)
- ET 200SP (https://support.industry.siemens.com/cs/ww/de/view/84133942)
- ET 200AL (https://support.industry.siemens.com/cs/ww/de/view/95242965)
- ET 200eco PN (https://support.industry.siemens.com/cs/ww/de/view/109781058)

1.1.2 Basiswerkzeuge

Werkzeuge

Die nachfolgend beschriebenen Werkzeuge unterstützen Sie bei allen Schritten von der Planung, über die Inbetriebnahme bis zur Analyse Ihrer Anlage.

TIA Selection Tool

Das TIA Selection Tool unterstützt Sie bei der Auswahl, Konfiguration und Bestellung von Geräten für Totally Integrated Automation (TIA).

Als Nachfolger des SIMATIC Selection Tools fasst das TIA Selection Tool die bereits bekannten Konfiguratoren für die Automatisierungstechnik in einem Werkzeug zusammen.

Mit dem TIA Selection Tool erzeugen Sie aus Ihrer Produktauswahl oder Produktkonfiguration eine vollständige Bestell-Liste.
Sie finden das TIA Selection Tool im Internet.
(https://support.industry.siemens.com/cs/ww/de/view/109767888)

SIMATIC Automation Tool

Mit dem SIMATIC Automation Tool führen Sie - unabhängig vom TIA Portal - an verschiedenen SIMATIC S7-Stationen Massenoperationen für Inbetriebsetzungs- und Servicetätigkeiten aus.

Das SIMATIC Automation Tool bietet eine Vielzahl von Funktionen:

- Scannen eines PROFINET/Ethernet Anlagennetzes und Identifikation aller verbundenen CPUs
- Zuweisung von Adressen (IP, Subnetz, Gateway) und Gerätename (PROFINET Device) zu einer CPU
- Übertragung des Datums und der auf UTC-Zeit umgerechneten PG/PC-Zeit auf die Baugruppe
- Programm-Download auf CPU
- Betriebsartenumstellung RUN/STOP
- CPU-Lokalisierung durch LED-Blinken
- Auslesen von CPU-Fehlerinformation
- Lesen des CPU-Diagnosepuffers
- Rücksetzen auf Werkseinstellungen
- Firmwareaktualisierung der CPU und angeschlossener Module

Sie finden das SIMATIC Automation Tool im Internet.
(https://support.industry.siemens.com/cs/ww/de/view/98161300)

PRONETA

SIEMENS PRONETA (PROFINET Netzwerk-Analyse) ist ein Inbetriebnahme- und Diagnosetool für PROFINET-Netzwerke. PRONETA Basic verfügt über 2 Kernfunktionen:

- In der Netzwerkanalyse erhalten Sie eine Übersicht über die PROFINET-Topologie. Vergleichen Sie einen realen Ausbau mit einer Referenzanlage oder nehmen Sie einfache Parameteränderungen vor, z. B. an den Namen und IP-Adressen der Geräte.
- Der „IO Test" ermöglicht einen einfachen und schnellen Test der Verdrahtung und des Modulausbaus einer Anlage, inklusive einer Dokumentation der Testergebnisse.

Sie finden SIEMENS PRONETA Basic im Internet:
(https://support.industry.siemens.com/cs/ww/de/view/67460624)

SIEMENS PRONETA Professional bietet Ihnen als lizenziertes Produkt zusätzliche Funktionen. Es ermöglicht Ihnen das einfache Asset-Management in PROFINET-Netzwerken und unterstützt Betreiber von Automatisierungsanlagen in der automatisierten Datenerfassung der eingesetzten Komponenten durch eine Vielzahl an Funktionen:

- Die Anwenderschnittstelle (API) bietet einen Zugangspunkt in die Automatisierungszelle, um über MQTT oder eine Kommandozeile die Scan-Funktionen zu automatisieren.
- Mittels der PROFIenergy-Diagnose lässt sich für Geräte, die PROFIenergy unterstützen, sehr schnell der aktuelle Pausenmodus oder die Betriebsbereitschaft erkennen und bei Bedarf ändern.
- Der Datensatz-Assistent unterstützt PROFINET-Entwickler, azyklische PROFINET-Datensätze schnell und einfach lesen und schreiben zu können – und das ohne SPS und Engineering.

Sie finden SIEMENS PRONETA Professional im Internet. (https://www.siemens.com/proneta-professional)

SINETPLAN

SINETPLAN, der Siemens Network Planner, unterstützt Sie als Planer von Automatisierungssystemen und -netzwerken auf Basis von PROFINET. Das Tool erleichtert Ihnen bereits in der Planungsphase die professionelle und vorausschauende Dimensionierung Ihrer PROFINET-Installation. Weiterhin unterstützt Sie SINETPLAN bei der Netzwerkoptimierung und hilft Ihnen, Netzwerkressourcen bestmöglich auszuschöpfen und Reserven einzuplanen. So vermeiden Sie Probleme bei der Inbetriebnahme oder Ausfälle im Produktivbetrieb schon im Vorfeld eines geplanten Einsatzes. Dies erhöht die Verfügbarkeit der Produktion und trägt zur Verbesserung der Betriebssicherheit bei.

Die Vorteile auf einen Blick

- Netzwerkoptimierung durch portgranulare Berechnung der Netzwerklast
- höhere Produktionsverfügbarkeit durch Onlinescan und Verifizierung bestehender Anlagen
- Transparenz vor Inbetriebnahme durch Import und Simulierung vorhandener STEP 7 Projekte
- Effizienz durch langfristige Sicherung vorhandener Investitionen und optimale Ausschöpfung der Ressourcen

Sie finden SINETPLAN im Internet.
(https://new.siemens.com/de/de/produkte/automatisierung/industrielle-kommunikation/profinet/sinetplan.html)

1.1.3 S7 Port Configuration Tool (S7-PCT)

SIMATIC S7-PCT

Das Port Configuration Tool (PCT) ist eine PC basierte Parametriersoftware von Siemens IO-Link Master-Modulen und IO-Link Devices beliebiger Hersteller.

Sie binden IO-Link-Devices dabei über standardisierte Gerätebeschreibung "IODD" ein, die Sie vom jeweiligen Gerätehersteller beziehen. S7-PCT unterstützt sowohl die Version 1.0 als auch V1.1 der IODD.

S7-PCT wird über die Hardwarekonfiguration der IO-Link Master aus STEP 7 aufgerufen. Wenn STEP 7 nicht zum Einsatz kommt bzw. der IO-Link Master nicht an einer SIMATIC Steuerung betrieben wird, ist auch ein "standalone"-Betrieb möglich.

Weitere Informationen zu IO-Link finden Sie im Internet.
(https://new.siemens.com/de/de/produkte/automatisierung/industrielle-kommunikation/io-link.html)

1.1.4 S7 Failsafe Configuration Tool (S7-FCT)

SIMATIC S7-FCT

Das Failsafe Configuration Tool (FCT) ermöglicht Ihnen in Engineering-Systemen von Drittherstellern die GSD-Projektierung folgender Geräte:

- ausgewählte, funktional fehlersichere SIMATIC-Peripherie
- funktional fehlersichere SIRIUS ACT PROFINET Interfaces

Das Engineering-System muss dazu die folgenden Voraussetzungen erfüllen:

- Unterstützung der CPD-Systemintegration gemäß "PROFIsafe - Profile for Safety Technology on PROFIBUS DP and PROFINET IO"
- TCI-Implementierung nach Conformance Class C3

Weitere Informationen zum S7-FCT finden Sie im Internet.
(https://support.industry.siemens.com/cs/ww/de/view/109762827)

1.1.5 MultiFieldbus Configuration Tool (MFCT)

MultiFieldbus Configuration Tool

Das MultiFieldbus Configuration Tool (MFCT) ist eine PC-basierte Software und unterstützt bei der Konfiguration von MultiFieldbus- und DALI-Devices. Außerdem bietet das MFCT komfortable Möglichkeiten zum Massen-Firmwareupdate von ET 200-Geräten mit MultiFieldbus-Unterstützung und dem Lesen von Servicedaten für viele weitere Siemens Geräte.

Funktionsumfang des MFCT

- MultiFieldbus Konfiguration:
  Projektierung, Konfiguration und Diagnosen von MultiFieldbus-Devices, Bereitstellung der benötigten Projektdateien (Projekt-, UDT-, CSV- und EDS-Datei), Transfer/Export der Dateien auf Device und/oder Datenspeicher.
- DALI-Konfiguration:
  Geräteauswahl und Online-Konfiguration von DALI-Geräten.
- TM FAST:
  Generierung und Download von FPGA-UPD- und FPGA-DB-Dateien.

- Wartung:
  Topologiescan eines Ethernet Netzwerks, Servicedaten lesen, Parametern zuweisen und Firmware-Update.
- Einstellungen:
  Sprachumschaltung deutsch und englisch, Geschwindigkeit Netzwerk-Scanner, Einstellung des Netzwerk-Adapters, Installation von GSDML-und EDS-Dateien.

System-/Installationsvoraussetzungen für MFCT

Das MFCT läuft unter Microsoft Windows und erfordert keine Installation oder Administratorrechte.

Für MFCT müssen Sie zusätzlich folgende Software installieren:

- Microsoft .NET Framework 4.8 (Sie finden einen Offline-Installer im Internet. (https://support.microsoft.com/de-de/topic/microsoft-net-framework-4-8-offline-installer-f%C3%BCr-windows-9d23f658-3b97-68ab-d013-aa3c3e7495e0))
- NPcap aus dem Verzeichnis "Misc"
- PG/PC interface aus dem Verzeichnis "Misc"
- Microsoft C++ Redistributable für x86-Systeme (Sie finden die Installationsdaten zum Download im Internet. (https://aka.ms/vs/15/release/vc_redist.x86.exe))

Den Download des Tools und weitere Informationen sowie eine Dokumentation zu den einzelnen Funktionen des MFCT finden Sie im Internet.
(https://support.industry.siemens.com/cs/de/de/view/109773881)

1.1.6 Technische Dokumentation der SIMATIC

Weiterführende SIMATIC Dokumente ergänzen Ihre Informationen. Sie finden diese Dokumente und deren Nutzung über die nachfolgenden Links und QR-Codes.

Der Industry Online Support vervollständigt die Möglichkeiten, Informationen zu allen Themen zu erhalten. Und die Anwendungsbeispiele unterstützen Sie bei der Lösung Ihrer Automatisierungsaufgaben.

Überblick zur Technischen Dokumentation der SIMATIC

Hier finden Sie eine Übersicht der im Siemens Industry Online Support verfügbaren Dokumentation zur SIMATIC:

Industry Online Support International
(https://support.industry.siemens.com/cs/ww/de/view/109742705)

Wo Sie die Übersicht direkt im Siemens Industry Online Support finden und wie Sie den Siemens Industry Online Support auf Ihrem mobilen Endgerät nutzen, zeigen wir Ihnen in einem kurzen Video:

Schneller Einstieg in die technische Dokumentation von Automatisierungsprodukten per Video (https://support.industry.siemens.com/cs/ww/de/view/109780491)

YouTube-Video: Siemens Automation Products - Technical Documentation at a Glance (https://youtu.be/TwLSxxRQQsA)

Aufbewahren der Dokumentation

Bewahren Sie die Dokumentation zur späteren Verwendung auf.
Bei digital beigefügter Dokumentation:

1. Laden Sie nach dem Erhalt Ihres Produkts, spätestens vor der ersten Montage/Inbetriebnahme, die dazugehörende Dokumentation herunter. Nutzen Sie für den Download folgende Möglichkeiten:
   - Industry Online Support International: (https://support.industry.siemens.com)
     Dem Produkt ist über die Artikelnummer eine Dokumentation zugeordnet. Sie finden die Artikelnummer auf dem Produkt und auf dem Verpackungsetikett. Produkte mit neuen, nichtkompatiblen Funktionen erhalten eine neue Artikelnummer und Dokumentation.
   - ID Link:
     Wenn Ihr Produkt mit einem ID Link gekennzeichnet ist, erkennen Sie den ID Link als QR-Code mit einem Rahmen und schwarzer Rahmenecke rechts unten. Der ID Link führt Sie zum digitalen Typenschild Ihres Produkts. Scannen Sie den QR-Code auf dem Produkt oder auf dem Verpackungsetikett mit einer Smartphone-Kamera, einem Barcode-Scanner oder einer Lese-App. Rufen Sie den ID Link auf.
2. Bewahren Sie diese Version der Dokumentation auf.

Aktualisieren der Dokumentation

Die Dokumentation des Produkts wird in digitaler Form aktualisiert. Insbesondere bei Erweiterung der Funktionen werden neue Leistungsmerkmale in einer aktualisierten Version bereitgestellt.

1. Laden Sie die aktuelle Version wie oben beschrieben über Industry Online Support oder den ID Link.
2. Bewahren Sie auch diese Version der Dokumentation auf.

mySupport

Mit mySupport machen Sie das Beste aus Ihrem Industry Online Support.

Registrierung	Um die volle Funktionalität von mySupport zu nutzen, müssen Sie sich einmalig registrieren. Nach der Registrierung haben Sie die Möglichkeit, Filter, Favoriten und Tabs in Ihrem persönlichen Arbeitsbereich anzulegen.
Support-Anfragen	Ihre Daten sind in Support-Anfragen bereits vorausgefüllt und Sie können sich jederzeit einen Überblick über Ihre laufenden Anfragen verschaffen.
Dokumentation	Im Bereich Dokumentation stellen Sie sich Ihre persönliche Bibliothek zusammen.
Favoriten	Mit der Schaltfläche "Zu mySupport-Favoriten hinzufügen" merken Sie besonders interessante oder häufig benötigte Inhalte vor. Unter dem Punkt "Favoriten" finden Sie eine Liste Ihrer vorgemerkten Einträge.
Zuletzt gesehene Beiträge	Die zuletzt in mySupport aufgerufenen Seiten finden Sie unter "Zuletzt gesehene Beiträge".

CAx-Daten	Der Bereich CAx-Daten ermöglicht Ihnen den Zugriff auf aktuelle Produktdaten für Ihr CAx- oder CAe-System. Mit wenigen Klicks konfigurieren Sie Ihr eigenes Downloadpaket: • Produktbilder, 2D-Maßbilder, 3D-Modelle, Geräteschaltpläne, EPLAN-Makrodateien • Handbücher, Kennlinien, Bedienungsanleitungen, Zertifikate • Produktstammdaten

Sie finden mySupport im Internet. (https://support.industry.siemens.com/My/ww/de/)

Anwendungsbeispiele

Die Anwendungsbeispiele unterstützen Sie mit verschiedenen Tools und Beispielen bei der Lösung Ihrer Automatisierungsaufgaben. Dabei werden Lösungen im Zusammenspiel mehrerer Komponenten im System dargestellt - losgelöst von der Fokussierung auf einzelne Produkte.
Sie finden die Anwendungsbeispiele im Internet.
(https://support.industry.siemens.com/cs/ww/de/ps/ae)

Sicherheitshinweise

2

2.1 Allgemeine Sicherheitshinweise

Beachten Sie die sicherheitsrelevanten Hinweise im entsprechenden Systemhandbuch. Cybersecurity-relevante Hinweise finden Sie im Kapitel Industrial Cybersecurity (Seite 26).

Industrial Cybersecurity 3

Durch die Digitalisierung und zunehmende Vernetzung von Maschinen und Industrieanlagen steigt auch die Gefahr von Cyberattacken. Insbesondere bei kritischen Infrastruktureinrichtungen sind entsprechende Schutzmaßnahmen daher Pflicht.
Informieren Sie sich über allgemeine Informationen und Maßnahmen zum Thema Industrial Cybersecurity im Systemhandbuch und den Security Leitfaden für SIMATIC HMI Bediengeräte (https://support.industry.siemens.com/cs/de/de/view/109481300).
Dieses Kapitel gibt einen Überblick über securityrelevante Informationen, die die Kommunikation Ihres SIMATIC-Systems betreffen.

HINWEIS

Securityrelevante Änderungen an Software oder Geräten sind im Kapitel Einleitung (Seite 11) dokumentiert.

3.1 Cybersecurity-Hinweise

Siemens bietet Produkte und Lösungen mit Industrial Cybersecurity-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.
Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Cybersecurity-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen einen Bestandteil eines solchen Konzepts.
Die Kunden sind dafür verantwortlich, unbefugten Zugriff auf ihre Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Diese Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und nur wenn entsprechende Schutzmaßnahmen (z. B. Firewalls und/oder Netzwerksegmentierung) ergriffen wurden.
Weiterführende Informationen zu möglichen Schutzmaßnahmen im Bereich Industrial Cybersecurity finden Sie unter (https://www.siemens.com/global/en/products/automation/topic-areas/industrial-cybersecurity.html).
Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Produkt-Updates anzuwenden, sobald sie zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.
Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Cybersecurity RSS Feed unter (https://new.siemens.com/global/en/products/services/cert.html).

3.2 Cybersecurity-relevante Informationen in diesem Handbuch

Beachten Sie alle securityrelevanten Hinweise in diesem Kommunikationshandbuch.

Security-relevante Hinweise zu ...	Kapitel
Schnittstellen	Produktübersicht (Seite 28)
Ports und Protokolle	Kommunikationsmöglichkeiten im Überblick (Seite 33) Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation (Seite 36)
Secure Communication	Secure Communication (Seite 50)
Dienste aktivieren/deaktivieren	SNMP aktivieren und deaktivieren (Seite 114) Adressvergabe über DHCP (Seite 366) IP-Forwarding (Seite 384)
Security-Funktionen der OPC UA-Kommunikation (Authentifizierung, Zertifikate, Benutzer anlegen und Rollen, sichere Nachrichtenübertragung)	OPC UA-Kommunikation (Seite 163)
Aktivieren von Security-Funktionen in den Kommunikationsmodulen	Virtuelle Schnittstelle für IP-basierte Anwendungen (Seite 393)
Kommunikation mit dem redundanten System S7-1500R/H	Kommunikation mit dem redundanten System S7-1500R/H (Seite 411)
Schutz durch Industrial Ethernet Security mit CP 1543-1	Industrial Ethernet Security mit CP 1543-1 (Seite 436)
Secure PG/HMI-Kommunikation	Secure PG/HMI-Kommunikation (Seite 103)
Schutz vertraulicher Daten	Schutz vertraulicher Konfigurationsdaten (Seite 74)

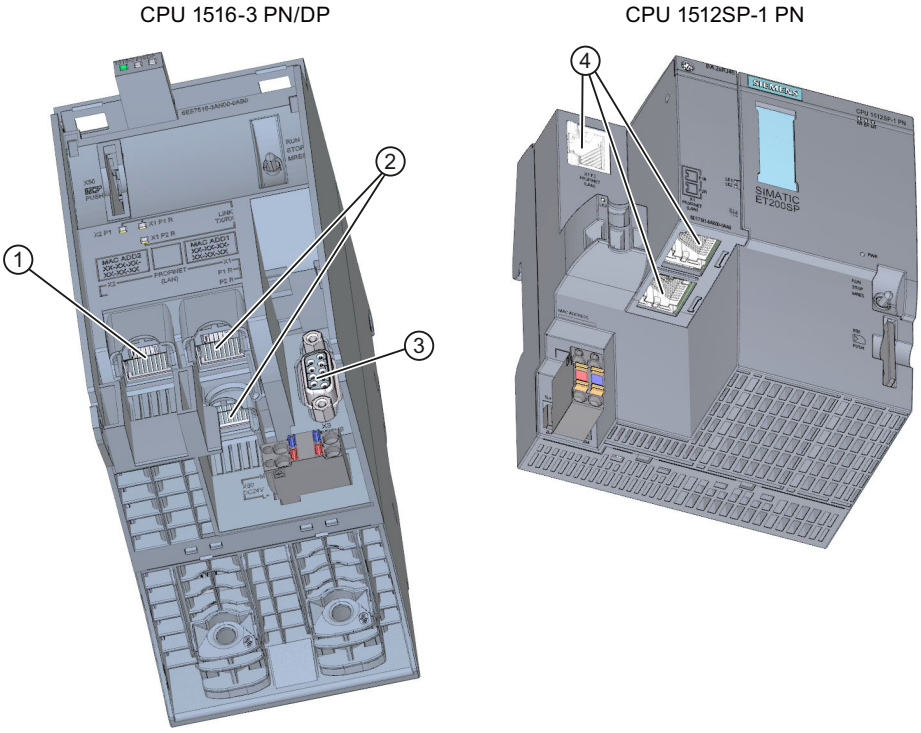
Produktübersicht

4

CPUs, Kommunikationsmodule und -prozessoren und PC-Systeme der Systeme S7-1500, ET 200MP, ET 200SP, ET 200pro und ET 200AL bieten Ihnen Schnittstellen für die Kommunikation über PROFINET, PROFIBUS und Punkt-zu-Punkt-Kopplung.

CPUs, Kommunikationsmodule und Kommunikationsprozessoren

PROFINET- und PROFIBUS DP-Schnittstellen sind in die CPUs S7-1500 integriert. Zum Beispiel verfügt die CPU 1516-3 PN/DP über 2 PROFINET- und 1 PROFIBUS DP-Schnittstelle. Weitere PROFINET- und PROFIBUS DP-Schnittstellen stehen über Kommunikationsmodule (CM) und Kommunikationsprozessoren (CP) zur Verfügung.

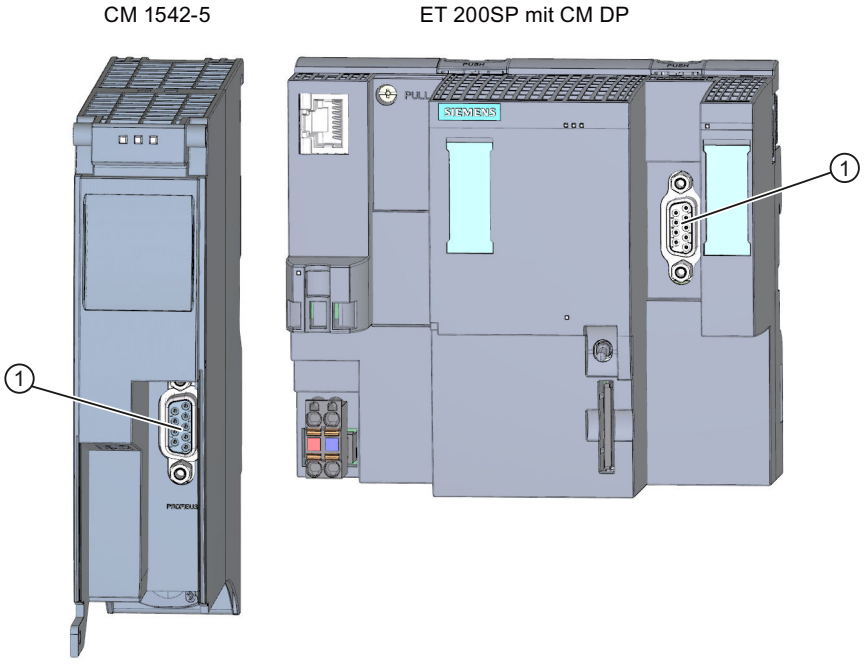


① PROFINET-Schnittstelle (X2) mit 1 Port
② PROFINET-Schnittstelle (X1) mit 2-Port-Switch
③ PROFIBUS DP-Schnittstelle (X3)
④ PROFINET-Schnittstelle (X1) mit 3-Port-Switch

Bild 4-1 Schnittstellen der CPU 1516-3 PN/DP und des CPU 1512SP-1 PN

Schnittstellen von Kommunikationsmodulen

Schnittstellen von Kommunikationsmodulen (CM) erweitern die Schnittstellen von CPUs (z. B. erweitert das Kommunikationsmodul CM 1542-5 das Automatisierungssystem S7-1500 um eine PROFIBUS-Schnittstelle).

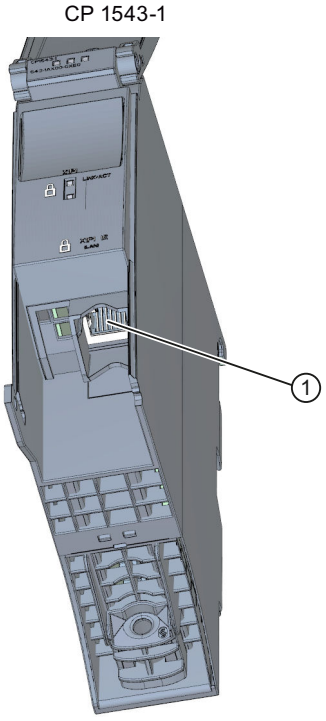


① PROFIBUS DP-Schnittstelle

Bild 4-2 PROFIBUS DP-Schnittstelle des CM 1542-5 und des CM DP (an einer ET 200SP CPU)

Schnittstellen von Kommunikationsprozessoren

Schnittstellen von Kommunikationsprozessoren (CP) bieten Ihnen zusätzliche Funktionalität zur Funktionalität der integrierten Schnittstellen der CPUs. CPs decken spezielle Anwendungsfälle ab, z. B. bietet der CP 1543-1 über seine Industrial Ethernet-Schnittstelle Security-Funktionen zur Absicherung von Industrial Ethernet-Netzwerken.



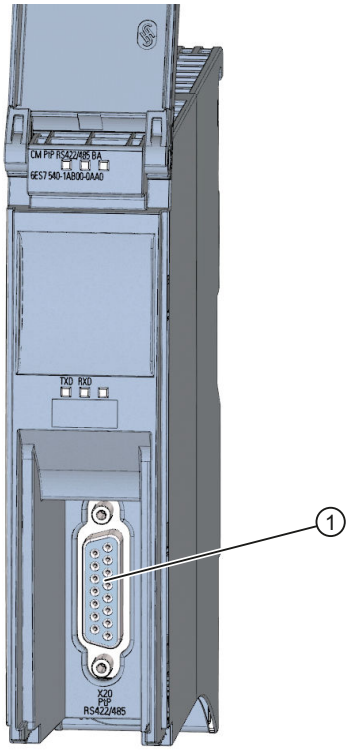
① Industrial Ethernet-Schnittstelle

Bild 4-3 Industrial Ethernet-Schnittstelle des CP 1543-1

Schnittstellen von Kommunikationsmodulen für Punkt-zu-Punkt-Kopplung

Die Kommunikationsmodule für Punkt-zu-Punkt-Kopplung bieten Ihnen Kommunikation über ihre RS232-, RS422- und RS485-Schnittstelle, z. B. Freeport- oder Modbus-Kommunikation.

CM PtP RS422/485 BA

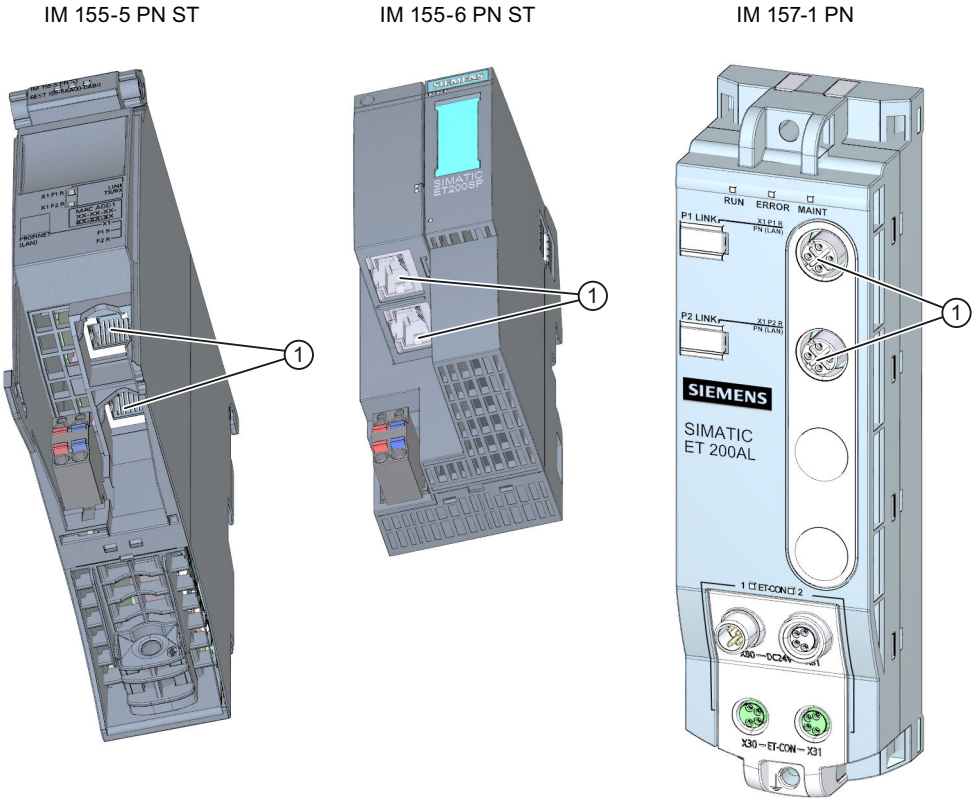


① Schnittstelle für Punkt-zu-Punkt-Kopplung

Bild 4-4 Beispiel für Schnittstelle für Punkt-zu-Punkt-Kopplung am CM PtP RS422/485 BA

Schnittstellen von Interfacemodulen

PROFINET- und PROFIBUS DP-Schnittstellen von Interfacemodulen (IM) in ET 200MP, ET 200SP und ET 200AL dienen der Anbindung der Dezentralen Peripherie ET 200MP, ET 200SP und ET 200AL an PROFINET bzw. PROFIBUS des überlagerten IO-Controllers bzw. DP-Masters.



① PROFINET-Schnittstelle mit 2-Port-Swich

Bild 4-5 PROFINET-Schnittstellen IM 155-5 PN ST (ET 200MP), IM 155-6 PN ST (ET 200SP), und IM 157-1 PN (ET 200AL)

Kommunikationsdienste

Die nachfolgend beschriebenen Kommunikationsdienste nutzen die Schnittstellen und Kommunikationsmechanismen, die Ihnen das System über CPUs, Kommunikationsmodule und -prozessoren bietet.

Kommunikationsdienste

5

5.1 Kommunikationsmöglichkeiten im Überblick

Übersicht über die Kommunikationsmöglichkeiten

Für Ihre Automatisierungsaufgabe stehen Ihnen folgende Kommunikationsmöglichkeiten zur Verfügung.

Tabelle 5-1 Möglichkeiten der Kommunikation

Möglichkeiten der Kommunikation	Funktionalität	Über Schnittstelle:		
		PN/IE[1]	DP	serielle
PG-Kommunikation[2]	Zur Inbetriebnahme, Test, Diagnose	X	X	-
HMI-Kommunikation[2]	Zum Bedienen und Beobachten	X	X	-
Offene Kommunikation über TCP/IP[2]	Datenaustausch über PROFINET/Industrial Ethernet mit TCP/IP Anweisungen: • TSEND_C/TRCV_C • TSEND/TRCV • TCON • T_DISCON	X	-	-
Offene Kommunikation über ISO-on-TCP[2]	Datenaustausch über PROFINET/Industrial Ethernet mit ISO-on-TCP Anweisungen: • TSEND_C/TRCV_C • TSEND/TRCV • TCON • T_DISCON	X	-	-
Offene Kommunikation über UDP[2]	Datenaustausch über PROFINET/Industrial Ethernet mit UDP Anweisungen: • TSEND_C/TRCV_C • TUSEND/TURCV • TCON • T_DISCON	X	-	-
Offene Kommunikation über ISO (nur CPs mit PROFINET/Industrial Ethernet-Schnittstelle)	Datenaustausch über PROFINET/Industrial Ethernet mit ISO-Protokoll Anweisungen: • TSEND_C/TRCV_C • TSEND/TRCV • TCON • T_DISCON	X	-	-

1 IE - Industrial Ethernet

2 Beachten Sie die Besonderheiten für S7-1500R/H

3 Nur über interne PROFINET-Schnittstelle der CPU und über Ethernet-Schnittstelle CP 1543-1 mit aktivierter Funktion "Zugriff auf PLC über Kommunikationsmodul"

Möglichkeiten der Kommunikation	Funktionalität	Über Schnittstelle:		
		PN/IE[1]	DP	serielle
Offene Kommunikation über FDL (nur CM 1542-5 ab Firmwarestand V2.0)	Datenaustausch über PROFIBUS mit Protokoll FDL Anweisungen: • TSEND_C/TRCV_C • TSEND/TRCV • TUSEND/TURCV • TCON • T_DISCON	-	X	-
OPC UA-Server[3]	Datenaustausch mit OPC UA-Clients	X	-	-
Kommunikation über Modbus TCP	Datenaustausch über PROFINET mit Protokoll Modbus TCP Anweisungen: • MB_CLIENT • MB_SERVER	X	-	-
E-Mail	Prozessmeldungen über E-Mail versenden Anweisung: • TMAIL_C	X	-	-
FTP (nur CPs mit PROFINET/Industrial Ethernet-Schnittstelle)	Dateiverwaltung und Dateizugriff über FTP (File Transfer Protocol); CP kann FTP-Client und FTP-Server sein Anweisung: • FTP_CMD	X	-	-
Fetch/Write (nur CPs mit PROFINET/Industrial Ethernet-Schnittstelle)	Serverdienste über TCP/IP, ISO-on-TCP und ISO Über spezielle Anweisungen für Fetch/Write	X	-	-
S7-Kommunikation	Datenaustausch über PROFINET/PROFIBUS mit S7-Protokoll. Anweisungen: • PUT/GET • BSEND/BRCV • USEND/URCV	X	X	-
Serielle Punkt-zu-Punkt-Kopplung	Datenaustausch über Punkt-zu-Punkt mit Freeport-, 3964(R)-, USS- oder Modbus-Protokoll Über spezielle Anweisungen für PtP, USS bzw. Modbus RTU	-	-	X
Webserver	Datenaustausch über HTTP(S), z. B. zur Diagnose	X	-	-
SNMP (Simple Network Management Protocol)	Zur Überwachung und Fehlererkennung von IP-Netzwerken, ggf. Parametrierung der IP-Netzkomponenten über Standardprotokoll SNMP	X	-	-
Uhrzeitsynchronisation	Über PN/IE-Schnittstelle: CPU ist NTP-Client (Network Time Protocol)	X	-	-
	Über DP-Schnittstelle: CPU/CM/CP ist Uhrzeitmaster oder Uhrzeit-Slave	-	X	-

1 IE - Industrial Ethernet

2 Beachten Sie die Besonderheiten für S7-1500R/H

3 Nur über interne PROFINET-Schnittstelle der CPU und über Ethernet-Schnittstelle CP 1543-1 mit aktivierter Funktion "Zugriff auf PLC über Kommunikationsmodul"

Informationen zu S7-1500R/H

Informationen zu den Kommunikationsmöglichkeiten mit dem redundanten System S7-1500R/H finden Sie im Kapitel Kommunikation mit dem redundanten System S7-1500R/H (Seite 411).

Weitere Informationen

- Ein Anwendungsbeispiel für die Konfiguration der TLS-basierten PG/HMI-Kommunikation und Schutz vertraulicher Konfigurationsdaten der CPU finden Sie in diesem Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/109798583).
- Ein Anwendungsbeispiel zur CPU-CPU Kommunikation mit SIMATIC Controllern (Kompendium) allgemein finden Sie in diesem Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/20982954).
- Eine TIA-Bibliothek "LOpcUa", die Ihnen Funktionsbausteine für die Implementierung von OPC UA PubSub für SIMATIC S7-1500 bereitstellt, finden Sie in diesem Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/109782455).

- Wie Sie bei der S7-1500 die Fetch/Write-Kommunikation über einen CP1543-1 konfigurieren, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/102420020).
- Weitere Informationen zu den Fetch/Write-Diensten finden Sie in der Online-Hilfe STEP 7.

- Informationen zur PtP-Kopplung finden Sie im Funktionshandbuch CM PtP - Konfigurationen für Punkt-zu-Punkt-Kopplungen (https://support.industry.siemens.com/cs/de/de/view/59057093).
- Die Beschreibung der Webserverfunktionalität finden Sie im Funktionshandbuch Webserver (https://support.industry.siemens.com/cs/ww/de/view/59193560).

- Allgemeine Informationen zum Standardprotokoll SNMP finden Sie auf den Service & Support Seiten im Internet (https://support.industry.siemens.com/cs/ww/de/view/15166742). Antworten auf die Frage, welche SNMP-Anfragen S7-1500 CPUs und S7-1200 CPUs unterstützen, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/at/de/view/79993228).
- Informationen zur Uhrzeitsynchronisation finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/86535497).

5.2 Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation

Dieser Abschnitt gibt eine Übersicht über die unterstützten Protokolle und die verwendeten Portnummern bei Kommunikation über PN/IE-Schnittstellen. Für jedes Protokoll sind die Adressparameter, die betroffene Kommunikationsschicht sowie die Kommunikationsrolle und Kommunikationsrichtung angegeben.

Diese Informationen ermöglichen Ihnen, Security-Maßnahmen zum Schutz des Automatisierungssystems auf die verwendeten Protokolle abzustimmen (z. B. Firewall). Da sich die Security-Maßnahmen auf Ethernet- bzw. PROFINET-Netze beschränken, sind in den Tabellen keine PROFIBUS-Protokolle aufgeführt.

HINWEIS

Verwendete Portnummern

Die angegebenen Portnummern sind die von der S7-1500 CPU standardmäßig verwendeten Portnummern. Viele Kommunikationsprotokolle bzw. Implementierungen erlauben Ihnen, andere Portnummern zu verwenden.

Die folgenden Tabellen zeigen die verschiedenen Schichten und Protokolle, die in den S7-1500 CPUs und in den S7-1500 Kommunikationsmodulen Einsatz finden.

Schichten und Protokolle der S7-1500 CPUs und Software Controller (über PROFINET-Schnittstelle der CPU)

Die folgende Tabelle zeigt die von S7-1500 CPUs, ET 200SP CPUs und der CPUs 1513/1516pro-2 PN unterstützten Protokolle. Die S7-1500 Software Controller unterstützen ebenfalls die in der folgenden Tabelle aufgeführten Protokolle für die Ethernet-Schnittstellen, die dem Software Controller zugewiesen sind.

Tabelle 5-2 Schichten und Protokolle der S7-1500 CPUs und Software Controller (über PROFINET-Schnittstelle der CPU)

Protokoll / Rolle	Portnummer	(2) Link-Layer-Schicht (4 Transportschicht	Beschreibung / Funktion	Voreinstellung / Hinweise
PROFINET-Protokolle				
DCP	Nicht relevant	(2) Ethertype 0x8892 (PROFINET)	PROFINET Discovery and Basic Configuration Protocol. DCP ermittelt PROFINET-Geräte und ermöglicht die Grundeinstellungen.	Voreinstellung: bis FW-Version V3.0 aktiviert, ab FW-Ver-sion V3.1 schreibgeschützt aktiviert. DCP lässt im schreibgeschützten Modus während einer aktiven Kommunikationsbeziehung keine DCP-Set Befehle von außen zu. Funktion in den CPU-Eigenschaften durch Boundary "Ende der Erfassung erreichbarer Teilnehmer" der Schnittstelle deaktivierbar.

[1] Hinweis: OUC (offene Kommunikation) liefert einen direkten Zugang zu den Protokollen UDP und TCP. Sie müssen die Port-Einschränkungen und Definitionen der IANA (Internet Assigned Numbers Authority) berücksichtigen.

[2] Verwenden Sie keine Ports für OUC, die bereits durch andere Protokolle belegt sind.

Protokoll / Rolle	Portnummer	(2) Link-Layer-Schicht (4 Transportschicht	Beschreibung / Funktion	Voreinstellung / Hinweise
DHCP Client	68	(4) UDP	Dynamic Host Configuration Protocol. Im Hochlauf der PROFINET-Schnittstelle wird die IP-Adress-Suite von einem DHCP-Server bezogen.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften (ab FW-Version 2.9) änderbar.
LLDP	Nicht relevant	(2) Ethertype 0x88CC (LLDP)	PROFINET Link Layer Discovery Protocol. LLDP ermittelt und verwaltet Nachbarschaftsbeziehungen zwischen PROFINET-Geräten.	Voreinstellung: Aktiviert. Sendefunktion durch Boundary "Ende der Topologieerkennung" in den CPU-Eigenschaften deaktivierbar, Empfangsbereitschaft bleibt aktiv. LLDP verwendet die spezielle Multicast-MAC-Adresse 01-80-C2-00-00-0E.
MRP	Nicht relevant	(2) Ethertype 0x88E3 (IEC 62493-2-2010)	Media Redundancy Protocol. MRP ermöglicht die Steuerung von redundanten Übertragungswegen in einer Ringtopologie.	Voreinstellung: "Manager (Auto)". In den CPU-Eigenschaften änderbar. Wenn Sie die CPU projektieren und die PN-Schnittstelle mit einem Subnetz verbinden ist die Voreinstellung im TIA Portal "nicht Teilnehmer des Rings". MRP verwendet normkonforme Multicast-MAC-Adressen.
PROFINET IO Data	Nicht relevant	(2) Ethertype 0x8892 (PROFINET)	PROFINET Cyclic IO Data Transfer Mit PROFINET IO Telegrammen werden zyklisch IO-Daten zwischen PROFINET IO-Controller und IO-Devices über Ethernet übertragen.	Voreinstellung: Deaktiviert. Das Protokoll ist nur bei PROFINET IO Datenverkehr aktiviert.
PROFINET Context Manager	34964	(4) UDP	PROFINET-Verbindung ohne RPC. Management von Applikations- und Kommunikationsbeziehungen zwischen IO-Controller und IO-Devices.	Voreinstellung: Aktiviert (UDP-Port geöffnet). Diese Funktion ist nicht deaktivierbar.
PTCP	Nicht relevant	(2) Ethertype 0x8892 (PROFINET)	PROFINET Precision Transparent Clock Protocol, basierend auf IEEE 1588. PTCP ermöglicht eine Zeitverzögerungsmessung zwischen RJ45 Ports und damit die Sendetakt- und Zeitsynchronisation.	Voreinstellung: Deaktiviert. Aktivierbar durch folgende Projektierungen: • IRT mit einer Sync-Domain. • Portverschaltung mit einer spezifizierten Leitungslänge. Funktion in den CPU-Eigenschaften durch Boundary "Ende der Sync-Domain" der Schnittstelle deaktivierbar. PTCP verwendet normkonforme Multicast-MAC-Adressen.
Verbindungsorientierte Kommunikationsprotokolle				

[1] Hinweis: OUC (offene Kommunikation) liefert einen direkten Zugang zu den Protokollen UDP und TCP. Sie müssen die Port-Einschränkungen und Definitionen der IANA (Internet Assigned Numbers Authority) berücksichtigen.

[2] Verwenden Sie keine Ports für OUC, die bereits durch andere Protokolle belegt sind.

Protokoll / Rolle	**Portnummer**	**(2) Link-Layer-Schicht (4 Transportschicht**	**Beschreibung / Funktion**	**Voreinstellung / Hinweise**
HTTP Server	80	(4) TCP	Hypertext Transfer Protocol. HTTP wird zur Kommunikation mit dem CPU-internen Webserver verwendet.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften aktivierbar. Voraussetzung: Webserver in den CPU-Eigenschaften ist aktiviert.
HTTPS Server	443	(4) TCP	Hypertext Transfer Protocol Secure. HTTPS wird zur Kommunikation mit dem CPU-internen Webserver über Secure Socket Layer (SSL) verwendet.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften aktivierbar. Voraussetzung: Webserver in den CPU-Eigenschaften ist aktiviert.
IGMPv2	Nicht relevant	(3) Vermittlungsschicht	Internet Group Management Protocol. IGMPv2 ist ein Netzwerkprotokoll zur Organisation von Multicast-Kreisen (nur UDP-Mulicast).	IGMPv2 ist eine Funktionalität des IP-Stacks. Diese Systemfunktion wird durch die Multicast-Funktion aktiviert.
ISO-on-TCP Server	102	(4) TCP	ISO-on-TCP Protocol (gemäß RFC 1006). Das S7-Protokoll nutzt ISO-on-TCP gemäß RFC 1006 für PG/HMI-Kommunikation mit dem Engineering System (TIA Portal).	Voreinstellung: Aktiviert. Diese Funktion ist nicht deaktivierbar.
MODBUS TCP Server / Client	502	(4) TCP	MODBUS Transmission Control Protocol. MODBUS/TCP wird durch MB_CLIENT/MB_SERVER Anweisungen im Anwenderprogramm verwendet.	Voreinstellung: Deaktiviert. Über Modbus-Anweisungen im Anwenderprogramm aktivierbar.
NTP Client	123	(4) UDP	Network Time Protocol. NTP wird zur Synchronisation der Systemzeit der CPU mit der Uhrzeit eines NTP-Servers verwendet.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften aktivierbar.
OPC UA Server / Client	4840	(4) TCP	Open Platform Communications Unified Architecture (basierend auf TCP/IP Protokoll). Ein Kommunikationsstandard mit Reichweite von der Enterprise-Ebene bis auf die Feldebene.	Voreinstellung: Deaktiviert. Server- und Client-Funktion in den CPU-Eigenschaften aktivierbar. Client-Zugriff im Anwenderprogramm konfigurierbar.
OUC[1] Secure OUC Server / Client	1 ... 1999 bedingt nutzbar[2] 2000 ... 5000 empfohlen	(4) TCP (4) UDP (4) ISO-on-TCP (Port: 102)	Open User Communication (TCP/UDP). Secure Open User Communication (TLS). OUC-Anweisungen ermöglichen den Verbindungsaufbau, Verbindungsab-	Voreinstellung: Deaktiviert. Sie aktivieren das jeweilige Protokoll durch die entsprechende Open-User-Communication-Anweisung im Anwenderprogramm bzw.

[1] Hinweis: OUC (offene Kommunikation) liefert einen direkten Zugang zu den Protokollen UDP und TCP. Sie müssen die Port-Einschränkungen und Definitionen der IANA (Internet Assigned Numbers Authority) berücksichtigen.

[2] Verwenden Sie keine Ports für OUC, die bereits durch andere Protokolle belegt sind.

Protokoll / Rolle	Portnummer	(2) Link-Layer-Schicht (4 Transportschicht	Beschreibung / Funktion	Voreinstellung / Hinweise
OUC[1] Secure OUC Server / Client	Ab einer FW-Version V3.0 gilt für programmierte und konfigurierte Verbindungen: 5001 ... 65535 bedingt nutzbar[2]	(4) TCP (4) UDP (4) ISO-on-TCP (Port: 102)	bau und Datentransfer über das Anwenderprogramm.	durch die Konfiguration von Verbindungen in der Netzwerkansicht. Für FW-Versionen kleiner V3.0 gilt: • Programmierte Verbindungen: 5001 ... 49152 • Konfigurierte Verbindungen: 5001 ... 65535
SMTP Client	25	(4) TCP	Simple Mail Transfer Protocol. SMTP wird zum Senden von E-Mails verwendet	Voreinstellung: Deaktiviert. Über TMAIL_C-Anweisung im Anwenderprogramm aktivierbar.
SMTPS (SMTP over TLS) Client	465	(4) TCP	Simple Mail Transfer Protocol Secure. SMTPS wird zum Senden von E-Mails über eine gesicherte Verbindung verwendet.	Voreinstellung: Deaktiviert. Über TMAIL_C-Anweisung im Anwenderprogramm aktivierbar.
SMTP mit STARTTLS Client	25 587	(4) TCP	Simple Mail Transfer Protocol mit dem SMTP-Befehl "STARTTLS" SMTP wird zum Senden von E-Mails verwendet.	Voreinstellung: Deaktiviert. Über TMAIL_C-Anweisung im Anwenderprogramm aktivierbar.
SNMP Agent	161 162 (trap)	(4) UDP	Simple Network Management Protocol. SNMP ermöglicht das Auslesen und Setzen von Netzwerk-Management-Daten (SNMP managed Objects) durch den SNMP-Manager	Voreinstellung: bis FW-Version V2.9 aktiviert, ab FW-Version V3.0 deaktiviert. Über Datensatz im Anwenderprogramm aktivierbar. Ab FW-Version V3.0 in den CPU-Eigenschaften aktivierbar. Ab FW-Version V3.1 in den CPU-Eigenschaften zusätzlich als schreibgeschützt aktivierbar.
Syslog (System Logging)	6514 514	(4) TCP (4) UDP	Syslog ist ein IETF-Standardprotokoll (RFC 5424) für die Übertragung von Ereignissen, die eine CPU erfasst.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften aktivierbar. In den CPU-Eigenschaften können Sie die Weiterleitung von Syslog-Meldungen an einen Syslog-Server konfigurieren. Die Erfassung von System Logging Ereignissen innerhalb einer CPU ab FW-Version V3.1 ist nicht deaktivierbar.
Reserved	49152 ... 65535	(4) TCP (4) UDP	Wenn eine Applikation keinen lokalen Port adressiert, dann verwendet die CPU diesen Portbereich für den aktiven Verbindungspunkt.	-

[1] Hinweis: OUC (offene Kommunikation) liefert einen direkten Zugang zu den Protokollen UDP und TCP. Sie müssen die Port-Einschränkungen und Definitionen der IANA (Internet Assigned Numbers Authority) berücksichtigen.

[2] Verwenden Sie keine Ports für OUC, die bereits durch andere Protokolle belegt sind.

Schichten und Protokolle der S7-1500 Software Controller (über Ethernet-Schnittstelle der Windows-Seite)

Die folgende Tabelle zeigt die Protokolle, die vom S7-1500 Software Controller über die Windows zugewiesenen Ethernet-Schnittstellen unterstützt werden.

Tabelle 5-3 Schichten und Protokolle der S7-1500 Software Controller (über Ethernet-Schnittstelle der Windows-Seite)

Protokoll / Rolle	**Portnummer**	**(2) Link-Layer-Schicht (4 Transportschicht**	**Beschreibung / Funktion**	**Hinweise / Voreinstellung**
PROFINET-Protokolle				
DCP	Nicht relevant	(2) Ethertype 0x8892 (PROFINET)	PROFINET Discovery and Basic Configuration Protocol. DCP ermittelt PROFINET-Geräte und ermöglicht die Grundeinstellungen.	Voreinstellung: bis Version V30.0 aktiviert. Ab Version V30.1 schreibgeschützt aktiviert. DCP lässt im schreibgeschützten Modus während einer aktiven Kommunikationsbeziehung keine DCP-Set Befehle von außen zu. Funktion durch Boundary "Ende der Erfassung erreichbarer Teilnehmer" in den CPU-Eigenschaften deaktivierbar.
DHCP Client	68	(4) UDP	Dynamic Host Configuration Protocol. Im Hochlauf der PROFINET-Schnittstelle wird die IP-Adress-Suite von einem DHCP-Server bezogen.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften (ab FW-Version 2.9) änderbar.
Verbindungsorientierte Kommunikationsprotokolle				
HTTP Server	Einstellbar[1]	(4) TCP	Hypertext Transfer Protocol. HTTP wird zur Kommunikation mit dem CPU-internen Webserver verwendet.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften änderbar. Die Portnummer passen Sie zur Vermeidung von Konflikten mit anderen Webservern unter Windows an. Wenn Sie den Webserverzugang des S7-1500 Software Controllers nutzen, müssen Sie den zugeordneten Port in der Windows Firewall freischalten.
IGMPv2	Nicht relevant	(3) Vermittlungsschicht	Internet Group Management Protocol. IGMPv2 ist ein Netzwerkprotokoll zur Organisation von Multicast-Kreisen (nur UDP-Mulicast).	IGMPv2 ist eine Funktionalität des IP-Stacks. Diese Systemfunktion wird durch die Multicast-Funktion aktiviert.
ISO-on-TCP Server	102	(4) TCP	ISO-on-TCP Protocol (gemäß RFC 1006). Das S7-Protokoll nutzt ISO-on-TCP gemäß RFC 1006 für PG/HMI-Kommunikation mit dem Engineering System (TIA Portal).	Voreinstellung: Deaktiviert.

[1] Voreinstellung bei Windows zugewiesenen Schnittstellen: 81

[2] Hinweis: OUC (offene Kommunikation) liefert einen direkten Zugang zu den Protokollen UDP und TCP. Sie müssen die Port-Einschränkungen und Definitionen der IANA (Internet Assigned Numbers Authority) berücksichtigen.

[3] Verwenden Sie keine Ports für OUC, die bereits durch andere Protokolle belegt sind.

[4] Verwenden Sie keine Ports für OUC, die bereits durch andere Windows-Anwendungen belegt sind.

Protokoll / Rolle	Portnummer	(2) Link-Layer-Schicht (4 Transportschicht	Beschreibung / Funktion	Hinweise / Voreinstellung
OUC[2] und Secure OUC	1 ... 1999 bedingt nutzbar[3,4] 2000 ... 5000 empfohlen[4] 5001 ... 49151 bedingt nutzbar[3,4]	(4) TCP (4) UDP (4) ISO-on-TCP (Port: 102)	Open User Communication (TCP/UDP). Secure Open User Communication (TLS). OUC-Anweisungen ermöglichen den Verbindungsaufbau, Verbindungsabbau und den Datentransfer, basierend auf dem Socket Layer.	Voreinstellung: Deaktiviert. Über Datensatz im Anwenderprogramm aktivierbar. Wenn Sie OUC nutzen, müssen Sie die Ports in der Windows Firewall freischalten.
SMTP Client	25	(4) TCP	Simple Mail Transfer Protocol. SMTP wird zum Senden von E-Mails verwendet.	Voreinstellung: Deaktiviert. Über Bausteinaufruf im Anwenderprogramm bzw. ab Version V3.0 über CPU-Einstellungen aktivierbar.
Syslog (System Logging)	6514 514	(4) TCP (4) UDP	Syslog ist ein IETF-Standardprotokoll (RFC 5424) für die Übertragung von Ereignissen, die eine CPU erfasst.	Voreinstellung: Deaktiviert. In den CPU-Eigenschaften aktivierbar. In den CPU-Eigenschaften können Sie die Weiterleitung von Syslog-Meldungen an einen Syslog-Server konfigurieren. Die Erfassung von System Logging Ereignissen innerhalb einer CPU ab FW-Version V3.1 ist nicht deaktivierbar.
Reserved	49152 ... 65535	(4) TCP (4) UDP	Wenn die Applikation nicht die lokale Portnummer bestimmt, wird dieser dynamische Port-Bereich für den aktiven Verbindungsendpunkt verwendet.	Wenn Sie diese Kommunikation nutzen, müssen Sie die Ports in der Windows Firewall freischalten.

1 Voreinstellung bei Windows zugewiesenen Schnittstellen: 81

2 Hinweis: OUC (offene Kommunikation) liefert einen direkten Zugang zu den Protokollen UDP und TCP. Sie müssen die Port-Einschränkungen und Definitionen der IANA (Internet Assigned Numbers Authority) berücksichtigen.

3 Verwenden Sie keine Ports für OUC, die bereits durch andere Protokolle belegt sind.

4 Verwenden Sie keine Ports für OUC, die bereits durch andere Windows-Anwendungen belegt sind.

Schichten und Protokolle von S7-1500 Kommunikationsmodulen

Die Dokumentation zu den Protokollen von S7-1500 Kommunikationsmodulen (z. B. CP 1543-1) finden Sie hier (https://support.industry.siemens.com/cs/ww/de/view/67700710).

5.3 Verbindungsressourcen im Überblick

Verbindungsressourcen

Einige Kommunikationsdienste benötigen Verbindungen. Verbindungen belegen Ressourcen in den beteiligten CPUs, CPs und CMs (z. B. Speicherbereiche im Betriebssystem der CPU). In den meisten Fällen wird für eine Verbindung eine Ressource pro CPU/CP/CM belegt. Bei HMI-Kommunikation werden pro HMI-Verbindung bis zu 3 Verbindungsressourcen benötigt.

Die zur Verfügung stehenden Verbindungsressourcen sind abhängig von der eingesetzten CPU, den CPs und CMs und dürfen eine definierte Obergrenze für das Automatisierungssystem nicht überschreiten.

Verfügbare Verbindungsressourcen in einer Station

Die maximale Anzahl der Ressourcen einer Station wird durch die CPU bestimmt.

Jede CPU bringt reservierte Verbindungsressourcen für PG-, HMI- und Webserver-Kommunikation mit. Daneben gibt es verfügbare Ressourcen für andere Kommunikationsdienste, z. B. für SNMP, E-Mail-Verbindungen, HMI- und S7-Kommunikation sowie für offene Kommunikation.

Wann werden Verbindungsressourcen belegt?

Der Zeitpunkt für die Belegung der Verbindungsressourcen hängt davon ab, wie die Verbindung eingerichtet wird, automatisch, programmiert oder projektiert (siehe Kapitel Einrichten einer Verbindung (Seite 43)).

Weitere Informationen

Nähere Informationen zur Belegung von Verbindungsressourcen und zur Anzeige von Verbindungsressourcen in STEP 7 finden Sie im Kapitel Verbindungsressourcen (Seite 397).

5.4 Einrichten einer Verbindung

Automatische Verbindung

STEP 7 richtet eine Verbindung automatisch ein (z. B. PG- oder HMI-Verbindung), sofern Sie die PG/PC-Schnittstelle physikalisch mit einer Schnittstelle der CPU verbunden haben und in STEP 7, im Dialog "Online verbinden" die Schnittstellen-Zuordnung vorgenommen haben.

Programmiertes Einrichten der Verbindung

Die programmierte Verbindung richten Sie im Programmeditor von STEP 7 im Kontext einer CPU durch Parametrierung von Anweisungen für Kommunikation, z. B. TSEND_C ein.
Bei der Festlegung der Verbindungsparameter (im Inspektorfenster, in den Eigenschaften der Anweisung) werden Sie durch die komfortable Bedienoberfläche unterstützt.

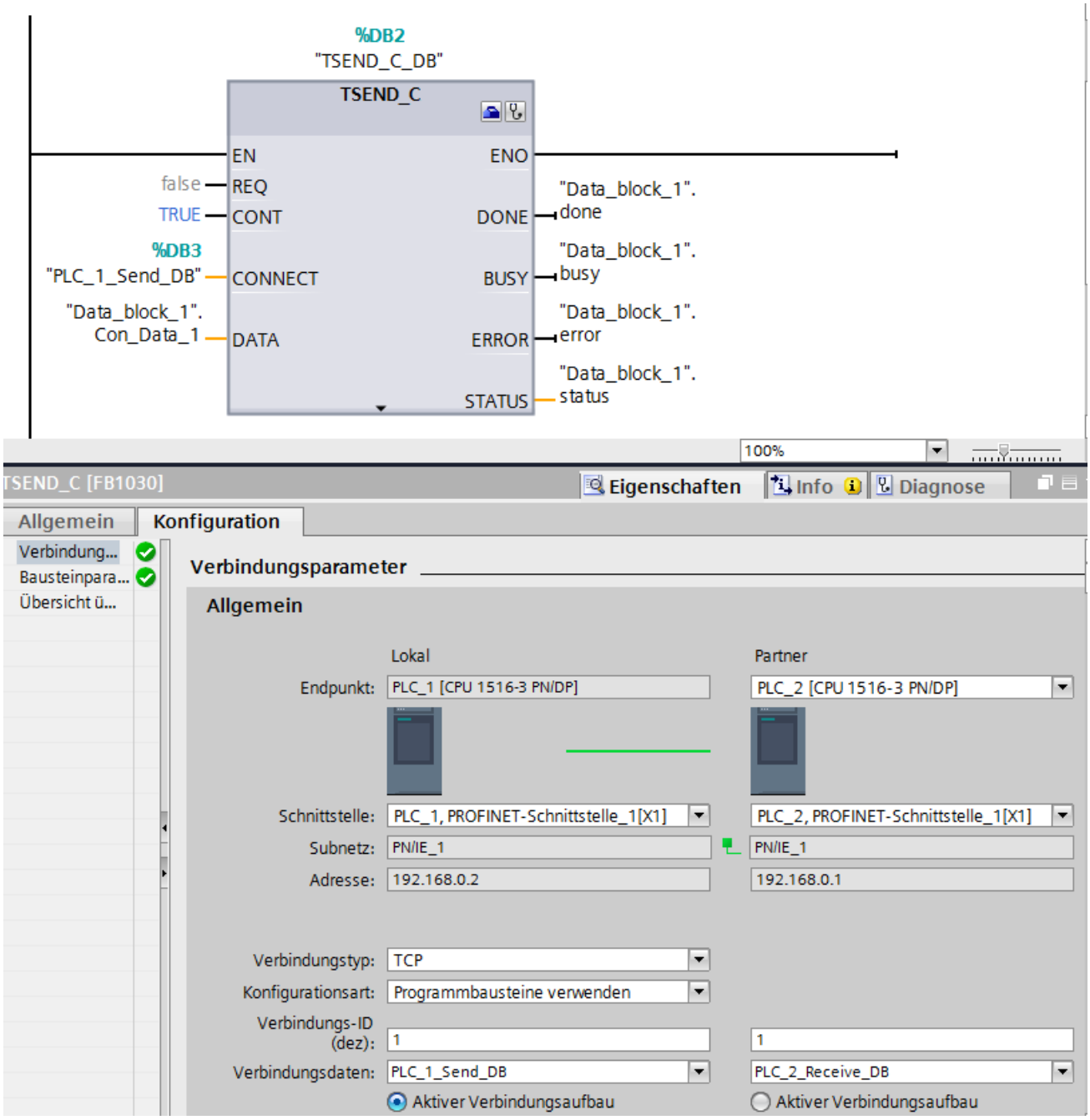


Bild 5-1 Programmiertes Einrichten

Projektiertes Einrichten der Verbindung

Die projektierte Verbindung richten Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 im Kontext einer CPU oder eines Software-Controllers ein.

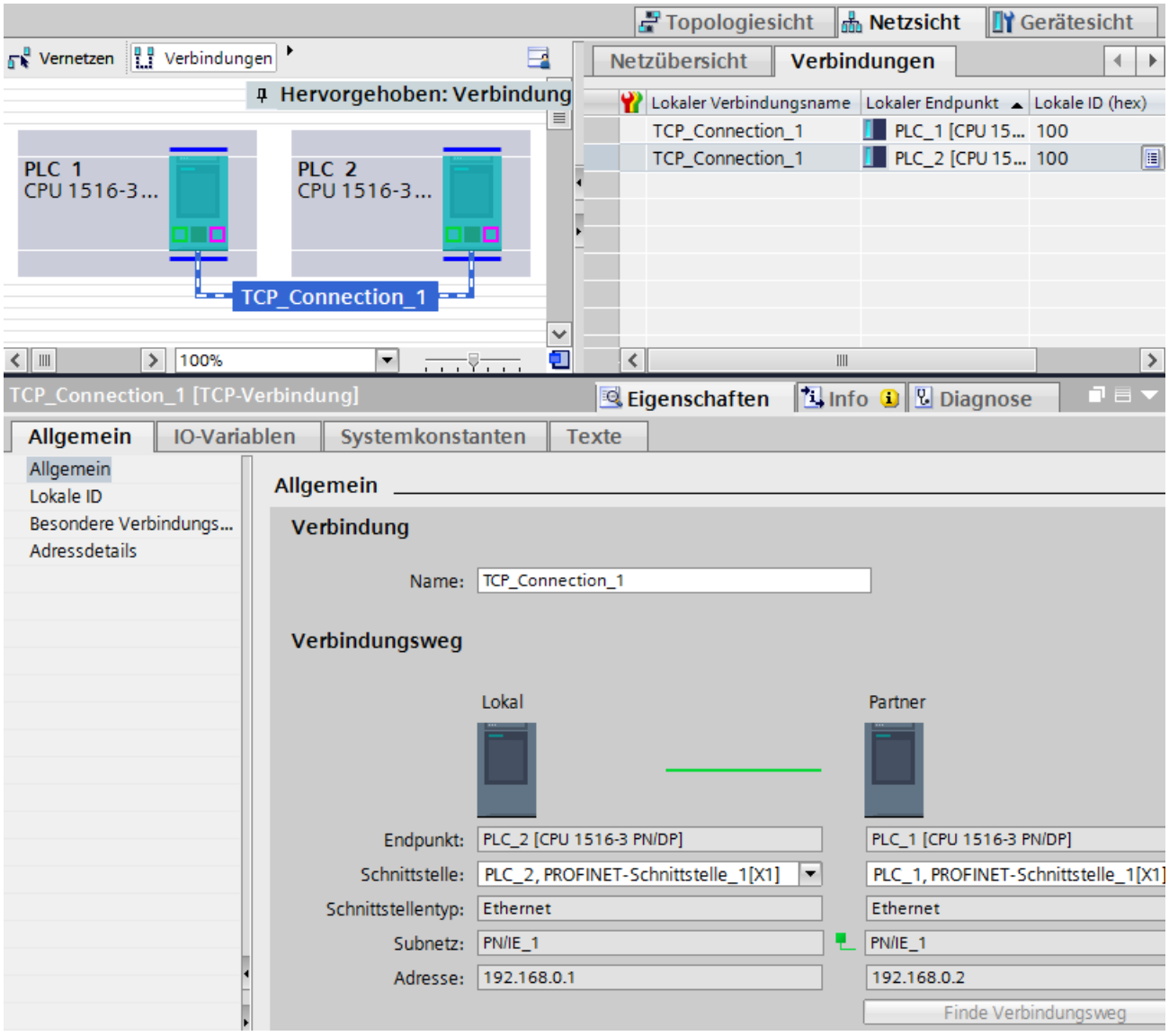


Bild 5-2 Projektiertes Einrichten

Auswirkungen auf die Verbindungsressourcen der CPU

Sie können sich oftmals alternativ für eine projektierte oder programmierte Verbindung entscheiden. Das programmierte Einrichten ermöglicht die Freigabe von Verbindungsressourcen nach der Datenübertragung. Programmierte Verbindungen sind wie geroutete Verbindungen nicht garantiert, d. h. sie werden nur aufgebaut, wenn Ressourcen frei sind. Beim projektierten Einrichten steht die Ressource nach dem Download der Konfiguration bis zur erneuten Änderung der Konfiguration zur Verfügung. Für den Verbindungsaufbau über projektierte Verbindungen sind daher entsprechende Ressourcen reserviert. Die Tabelle "Verbindungsressourcen" im Inspektorfenster der CPU zeigt eine Übersicht der bereits belegten und noch verfügbaren Verbindungsressourcen an.

Wie richte ich welche Verbindung ein?

Tabelle 5-4 Einrichten der Verbindung

Verbindung	Automatisch	Programmiertes Einrichten	Projektiertes Einrichten
PG-Verbindung	X	-	-
HMI-Verbindung	X	-	X
Web-Kommunikation	X	-	-
OPC UA-Server-Kommunikation	X	-	-
OPC UA-Client-Kommunikation	-	X	-
Offene Kommunikation über TCP/IP-Verbindung	-	X	X
Offene Kommunikation über ISO-on-TCP-Verbindung	-	X	X
Offene Kommunikation über UDP-Verbindung	-	X	X
Offene Kommunikation über ISO-Verbindung	-	X	X
Offene Kommunikation über FDL-Verbindung	-	X	X
Kommunikation über Modbus TCP-Verbindung	-	X	-
E-Mail-Verbindung	-	X	-
FTP-Verbindung	-	X	-
S7-Verbindung*	-	-	X

* Beachten Sie, dass Sie bei einer S7-1500 CPU die Nutzung von PUT/GET-Kommunikation in den Eigenschaften der CPU freigeben müssen. Weitere Informationen dazu finden Sie in der Online-Hilfe von STEP 7.

Weitere Informationen

Weitere Informationen zur Belegung von Verbindungsressourcen und zur Anzeige von Verbindungsressourcen in STEP 7 finden Sie im Kapitel Verbindungsressourcen (Seite 397).

5.5 Datenkonsistenz

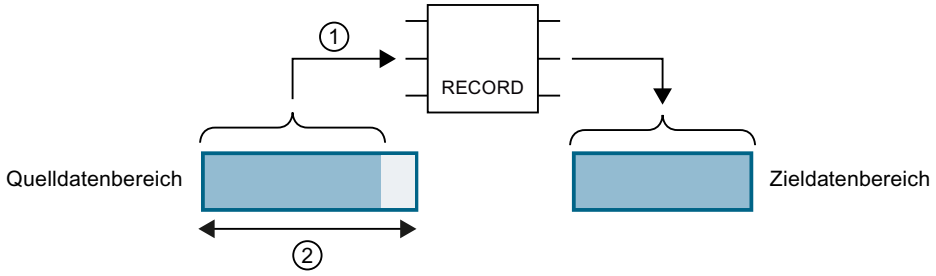
Definition

Für die Übertragung von Daten ist die Datenkonsistenz eine wichtige Eigenschaft, die Sie bei der Projektierung einer Kommunikationsaufgabe berücksichtigen müssen. Geschieht das nicht, kann es zu Fehlfunktionen kommen.

Ein Datenbereich, der nicht durch konkurrierende Prozesse verändert werden kann, wird als konsistenter Datenbereich bezeichnet. Das heißt, ein in sich zusammengehöriger Datenbereich, der größer ist als die Maximalgröße des konsistenten Datenbereichs, kann zu einem Zeitpunkt teilweise aus neuen und aus alten Daten bestehen.

Eine Inkonsistenz kann entstehen, wenn eine Anweisung für Kommunikation z. B. durch einen Prozessalarm-OB mit höherer Priorität unterbrochen wird. Dadurch wird auch die Übertragung des Datenbereichs unterbrochen. Verändert das Anwenderprogramm in diesem OB jetzt die Daten, die noch nicht von der Kommunikationsanweisung verarbeitet wurden, stammen die übertragenen Daten aus unterschiedlichen Zeitpunkten.

Das folgende Bild zeigt einen Datenbereich, der kleiner ist als die Maximalgröße des konsistenten Datenbereichs. In diesem Fall wird bei der Übertragung des Datenbereichs sichergestellt, dass während des Datenzugriffs keine Unterbrechung durch das Anwenderprogramm erfolgt und damit die Daten nicht geändert werden.

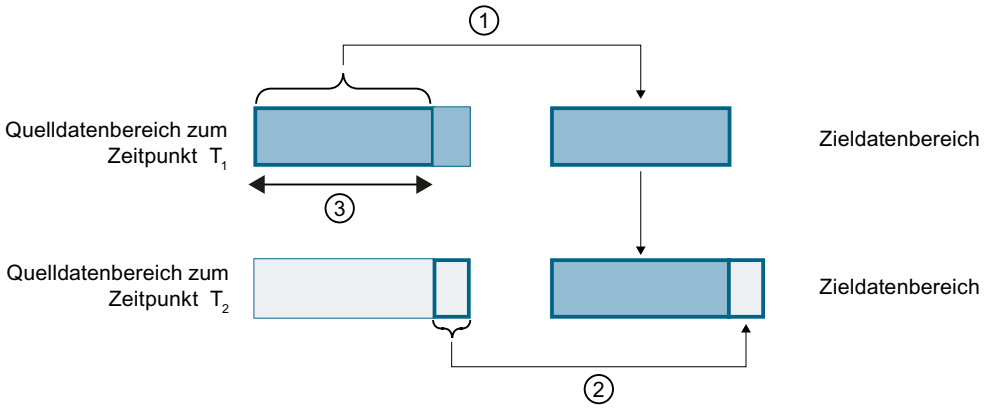


① Der Quelldatenbereich ist kleiner als die Maximalgröße des konsistenten Datenbereichs (②). Die Anweisung überträgt die Daten zusammenhängend in den Zieldatenbereich.

② Maximalgröße konsistenter Datenbereich

Bild 5-3 Konsistente Übertragung von Daten

Das folgende Bild zeigt einen Datenbereich, der größer ist als die Maximalgröße des konsistenten Datenbereichs. In diesem Fall können die Daten während einer Unterbrechung der Datenübertragung verändert werden. Eine Unterbrechung entsteht z.B. auch, wenn der Datenbereich in mehreren Teilen übertragen werden muss. Werden die Daten während der Unterbrechung verändert, dann stammen die übertragenen Daten aus unterschiedlichen Zeitpunkten.



① Der Quelldatenbereich ist größer als die Maximalgröße des konsistenten Datenbereichs (③). Zum Zeitpunkt T1 überträgt die Anweisung nur so viel Daten vom Quelldatenbereich in den Zieldatenbereich, wie in den konsistenten Datenbereich reinpassen.

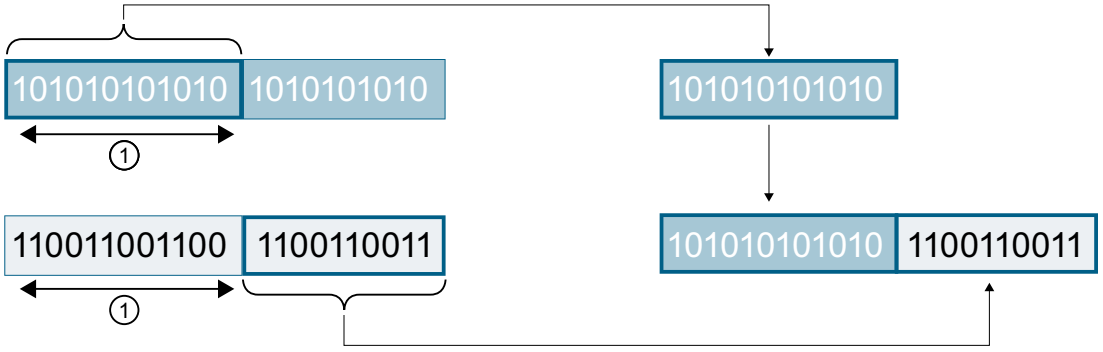
② Zum Zeitpunkt T2 überträgt die Anweisung den Rest des Quelldatenbereichs in den Zieldatenbereich. Nach der Übertragung liegen im Zieldatenbereich Daten aus unterschiedlichen Zeitpunkten. Wenn sich die Daten im Quelldatenbereich in der Zwischenzeit geändert haben, kann eine Inkonsistenz entstehen.

③ Maximalgröße konsistenter Datenbereich

Bild 5-4 Übertragung von Daten größer dem maximalen Konsistenzbereich

Beispiel für eine Inkonsistenz

Das folgende Bild zeigt ein Beispiel für Veränderung von Daten während der Übertragung. Im Zieldatenbereich liegen Daten aus unterschiedlichen Zeitpunkten.



① Maximalgröße konsistenter Datenbereich

Bild 5-5 Beispiel: Veränderung von Daten während der Übertragung

Systemspezifische maximale Datenkonsistenz für S7-1500:

Eine Inkonsistenz tritt nicht auf, wenn die systemspezifische Maximalgröße der konsistenten Daten eingehalten wird. Bei S7-1500 werden die Kommunikationsdaten in Blöcken bis maximal 512 Byte während des Programmzyklus konsistent in/aus dem Anwenderspeicher kopiert. Für alle größeren Datenbereiche wird keine Datenkonsistenz garantiert. Ist eine definierte Datenkonsistenz gefordert, so dürfen die Kommunikationsdaten im Anwenderprogramm der CPU nicht größer als 512 Byte sein. Auf diese Datenbereiche können Sie dann z. B. von einem HMI-Gerät mit Lesen/Schreiben von Variablen konsistent zugreifen.

Wenn mehr Daten als die systemspezifische Maximalgröße konsistent übertragen werden sollen, dann müssen Sie selbst durch entsprechende Maßnahmen im Anwenderprogramm die Datenkonsistenz sicherstellen.

Datenkonsistenz sicherstellen

Einsatz von Anweisungen für Zugriff auf gemeinsame Daten:

Existieren im Anwenderprogramm Kommunikationsanweisungen, welche auf gemeinsame Daten zugreifen, z. B. TSEND/TRCV, können Sie den Zugriff auf diesen Datenbereich z. B. über den Parameter "DONE" selbst koordinieren. Die Datenkonsistenz der Datenbereiche, die mit einer Anweisung für Kommunikation übertragen werden, kann deshalb im Anwenderprogramm sichergestellt werden.

HINWEIS

Maßnahmen im Anwenderprogramm

Um Datenkonsistenz zu erreichen, können Sie die zu übertragenden Daten auf einen separaten Datenbereich (z. B. globaler Datenbaustein) umkopieren. Während das Anwenderprogramm weiterhin mit den Originaldaten arbeitet, können Sie die im separaten Datenbereich gespeicherten Daten konsistent mit der Kommunikationsanweisung übertragen.

Verwenden Sie für das Umkopieren nicht unterbrechbare Anweisungen, wie UMOVE_BLK oder UFILL_BLK. Diese Anweisungen gewährleisten eine Datenkonsistenz bis 16 KByte.

Einsatz von Anweisungen PUT/GET bzw. Schreiben/Lesen über HMI-Kommunikation:

Bei S7-Kommunikation mit den Anweisungen PUT/GET bzw. Schreiben/Lesen über HMI-Kommunikation müssen Sie bereits bei der Programmierung bzw. Projektierung die Größe der konsistenten Datenbereiche berücksichtigen. Im Anwenderprogramm einer S7-1500 als Server ist keine Anweisung vorhanden, die die Datenübertragung im Anwenderprogramm koordinieren kann. Die über PUT/GET-Anweisungen ausgetauschten Daten aktualisiert die S7-1500 während der Laufzeit des Anwenderprogramms. Es gibt keinen Zeitpunkt innerhalb der Bearbeitung des zyklischen Anwenderprogramms, an dem die Daten konsistent ausgetauscht werden. Die Länge des zu übertragenden Datenbereichs sollte kleiner sein als 512 Bytes.

Weitere Informationen

- Die max. Anzahl konsistenter Daten finden Sie auch in den Gerätehandbüchern der Kommunikationsmodule in den Technischen Daten.
- Weitere Informationen zur Datenkonsistenz finden Sie in der Beschreibung der Anweisungen in der Online-Hilfe STEP 7.

5.6 Secure Communication

5.6.1 Grundlagen zu Secure Communication

5.6.1.1 Wissenswertes zu Secure Communication

Für STEP 7 (TIA Portal) ab V14 und für S7-1500 CPUs ab Firmware V2.0 sind die Möglichkeiten zur sicheren Kommunikation, im Folgenden als "Secure Communication" bezeichnet, erheblich erweitert worden.

Mit "S7-1500 CPUs" sind auch die CPU-Varianten S7-1500F, S7-1500T, S7-1500C sowie S7-1500pro CPUs und ET200SP CPUs gemeint.

In nachfolgenden Versionen unterstützen weitere Komponenten Secure Communication (z. B. Secure OUC), siehe nächsten Abschnitt.

Ab Firmware Version V4.4 unterstützen auch S7-1200 CPUs Secure Communication.

Voraussetzung

- CPUs, die Verbindungsbeschreibungs-DBs mit der Struktur des SDT TCON_IP_V4_SEC bzw. SDT TCON_QDN_SEC unterstützen. Das sind folgende CPUs:
  - S7-1200 ab Firmware V4.4
  - S7-1500 ab Firmware V2.0
- zusätzlich optional über folgende CPs:
  - CP 1243-1 ab Firmware V3.2
  - CP 1243-8 IRC ab Firmware V3.2
  - CP 1543-1 ab Firmware V2.0
  - CP 1545-1
  - CP 1543SP-1

  Nicht möglich ist Secure Communication über CP 1242-7 GPRS V2.

Public Key Infrastructure (PKI)

Das Attribut "secure" wird für die Kennzeichnung von Kommunikationsmechanismen verwendet, die auf einer Public Key Infrastructure (PKI) aufbauen (z. B. RFC 5280 für Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List Profile). Mit Public Key Infrastructure (PKI) ist ein System gemeint, das digitale Zertifikate ausstellen, verteilen und prüfen kann. Die ausgestellten digitalen Zertifikate werden innerhalb der PKI zur Absicherung von rechnergestützter Kommunikation verwendet. Wenn eine PKI ein asymmetrisches Schlüsselverfahren nutzt, dann können die Nachrichten in einem Netzwerk digital signiert und verschlüsselt werden.

Komponenten, die Sie in STEP 7 (TIA Portal) für secure Communication projektiert haben, verwenden ein asymmetrisches Schlüsselverfahren mit öffentlichem Schlüssel (Public Key) und privatem Schlüssel (Private Key). Als Verschlüsselungsprotokoll wird TLS (Transport Layer Security) eingesetzt. TLS ist Nachfolger für das Protokoll SSL (Secure Sockets Layer).

Ziele für Secure Communication

Secure Communication wird angewendet, um folgende Ziele zu erreichen:

- Vertraulichkeit
  d. h. die Daten sind für nicht autorisierten Lauscher geheim bzw. nicht lesbar.
- Integrität
  d. h. die Nachricht, die beim Empfänger eintrifft ist dieselbe, unveränderte Nachricht, die der Sender geschickt hat. Die Nachricht wurde auf dem Transportweg nicht verändert.
- Endpunkt Authentifizierung
  d. h. der Kommunikationspartner als Endpunkt ist genau derjenige, der er vorgibt zu sein und der erreicht werden soll. Die Identität des Partners ist geprüft.

Waren diese Ziele in der Vergangenheit hauptsächlich für die IT-Welt und für die vernetzten Computer von Belang, so sind heutzutage auch im industriellen Umfeld Maschinen und Steuerungen mit schützenswerten Daten durch ihre Vernetzung genauso gefährdet und stellen hohe Anforderungen an einen sicheren Datenaustausch.

Bisher und weiterhin gängig ist der Schutz der Automatisierungszelle mit Hilfe des Zellenschutzkonzepts per Firewall, oder per Verbindung über VPN, z. B. mit dem Security Modul.

Zunehmend besteht aber der Kommunikationsbedarf, auch Daten an externe Rechner in verschlüsselter Form über Intranet oder öffentliche Netze zu übertragen

Gemeinsame Prinzipien von Secure Communication

Unabhängig vom Kontext basiert Secure Communication auf dem Konzept der Public Key Infrastructure (PKI) und beinhaltet folgende Komponenten:

- Ein asymmetrisches Verschlüsselungsverfahren. Dieses Verfahren ermöglicht Folgendes:
  - Verschlüsselung oder Entschlüsselung der Nachrichten mit Hilfe von öffentlichen oder privaten Schlüsseln.
  - Überprüfung von Signaturen an Nachrichten und Zertifikaten.
    Die Nachrichten/Zertifikate werden vom Sender/Zertifikatsinhaber mit ihrem privaten Schlüssel signiert. Der Empfänger/Prüfer überprüft die Signatur mit dem öffentlichen Schlüssel des Senders/Zertifikatsinhabers.
- Transport und Speicherung der öffentlichen Schlüssel mit Hilfe von X.509-Zertifikaten:
  - X.509-Zertifikate sind digital signierte Daten, mit der die Echtheit von öffentlichen Schlüsseln in Bezug auf die gebundene Identität überprüft werden kann.
  - X.509-Zertifikate können Informationen enthalten, die die Benutzung der öffentlichen Schlüssel genauer charakterisieren bzw. einschränken. Zum Beispiel ab wann ein öffentlicher Schlüssel in einem Zertifikat gültig ist und ab wann er ungültig wird.
  - X.509-Zertifikate enthalten abgesichert die Informationen über den Herausgeber des Zertifikats.

Die folgenden Ausführungen geben einen Überblick über diese Grundkonzepte, die z.B. für den Umgang mit Zertifikaten in STEP 7 (TIA Portal) oder für die Programmierung der Kommunikationsanweisungen für secure Open User Communication (sOUC) erforderlich sind.

Secure Communication bei STEP 7

STEP 7 ab V14 stellt die jeweils notwendige PKI zur Verfügung, die für die Projektierung und den Betrieb einer Secure Communication erforderlich ist.
Beispiele:

- Das Hypertext Transfer Protokoll (HTTP) wird mit Hilfe des Protokolls TLS (Transport Layer Security) zum Hypertext Transfer Protokoll Secure (HTTPS). Weil HTTPS eine Kombination aus HTTP und TLS ist, wird es im entsprechenden RFC "HTTP over TLS" genannt. Die Verwendung von HTTPS erkennt man im Browser daran, dass in der Aufrufzeile des Browsers das URL-Schema "https://" statt "http://" verwendet wird. Die meisten Browser heben eine solchermaßen abgesicherte Verbindung zusätzlich optisch hervor.
- Open User Communication wird zur secure Open User Communication. Das zugrunde liegende Protokoll ist ebenfalls TLS.
- E-Mail Provider bieten ebenfalls Zugang über das Protokoll "Secure SMTP over TLS" an, um die Sicherheit des E-Mail-Verkehrs zu erhöhen.

Das folgende Bild zeigt das Protokoll TLS im Kontext der Kommunikationsschichten.

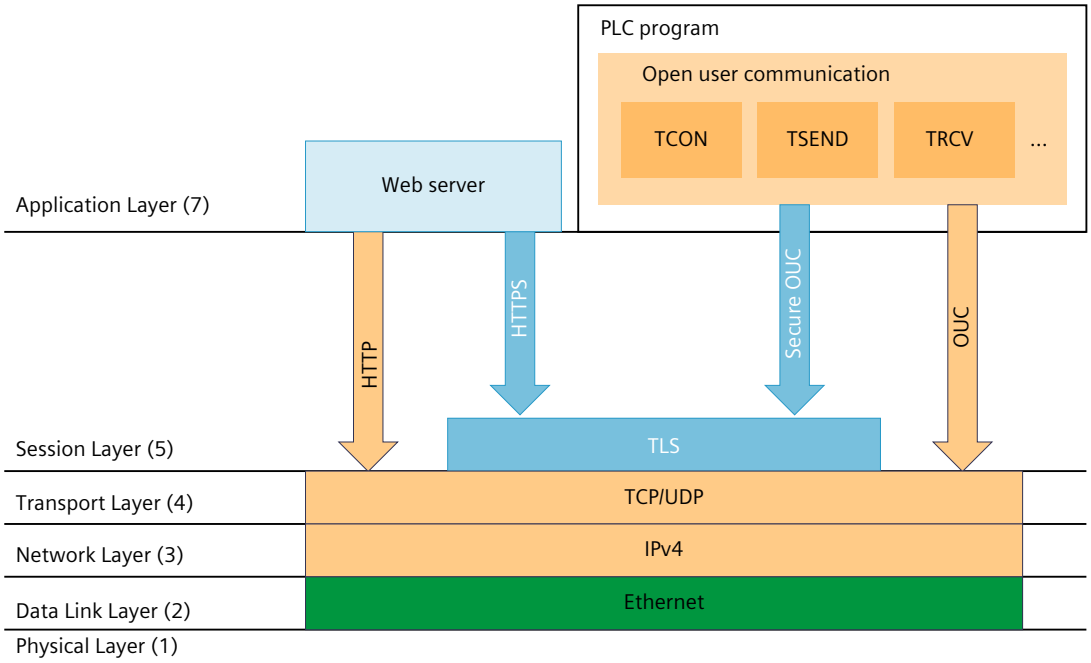


Bild 5-6 Protokoll TLS im Kontext der Kommunikationsschichten

Secure Communication bei OPC UA

In den S7-1500 CPUs ist ab Firmware V2.0 ein OPC UA-Server implementiert. OPC UA Security umfasst ebenfalls Authentifizierung, Verschlüsselung und Datenintegrität über digitale X.509-Zertifikate und nutzt ebenfalls eine Public Key Infrastructure (PKI). Entsprechend den Anforderungen der Anwendung können Sie verschiedene Security-Stufen für die Endpunkt-Security wählen. Der Beschreibung der OPC UA Server-Funktionalität ist ein eigenes Kapitel gewidmet.

Secure Communication für PG/HMI-Kommunikation

Mit den zentralen Komponenten des TIA Portals, STEP 7 und WinCC ist ab Version V17 zusammen mit den aktuellen Steuerungen und aktuellen HMI-Geräten eine innovierte und standardisierte sichere (secure) PG/PC- und HMI-Kommunikation - kurz PG/HMI-Kommunikation - implementiert.

Weitere Informationen

Weitere Informationen zu OPC UA finden Sie im Kap. S7-1500 CPU als OPC UA-Server nutzen (Seite 211).
Weitere Informationen zur sicheren PG/HMI-Kommunikation finden Sie im Kap. Secure PG/HMI-Kommunikation (Seite 103).

5.6.1.2 Geräteabhängige Security-Merkmale

Transport Layer Security (TLS) ist ein weit verbreitetes Sicherheitsprotokoll, das die Datensicherheit für die Kommunikation verbessert. Für das Automatisierungssystem S7-1500 wird TLS für Secure Communication verwendet für folgende zertifikatsbasierte Anwendungen:

- Webserver (Protokollvariante HTTPS)
- Secure Open User Communication (OUC) incl. Secure E-Mail (TMAIL_C-Anweisung)
- Secure PG/HMI-Kommunikation

TLS sorgt für die Authentizität, Vertraulichkeit und Integrität der Kommunikation zwischen Client und Server bei den aufgeführten Anwendungen, z. B. zwischen dem Webserver der CPU und Webbrowsern, die z. B. eine Diagnose-Webseite der CPU anzeigen sollen.
Die Anwendungen OPC UA-Server und OPC UA-Client nutzen zwar nicht TLS direkt, aber die verwendeten kryptografischen Verfahren sind vergleichbar.
TLS wird ständig weiterentwickelt, was dazu führt, dass es verschiedene TLS-Versionen gibt, die sich hinsichtlich der unterstützten Cypher-Suiten (standardisierte Sammlung kryptographischer Methoden) und Performance unterscheiden.
Verantwortlich für die Beschreibung des TLS-Protokolls ist die Internet Engineering Task Force (IETF). Es gilt folgende Zuordnung:

- TLS 1.3 entspricht RFC 8446
- TLS 1.2 entspricht RFC 5246

Dazu kommt, dass nicht jedes Gerät alle in den RFCs definierten kryptografischen Methoden unterstützt; daher handeln Client und Server nach dem Verbindungsaufbau eine Methode aus, die beide unterstützen (Handshake) sowie die zu verwendenden Parameter.

Unterstützte TLS-Versionen S7-1500

Die folgende Tabelle zeigt, welche TLS-Versionen in welcher CPU-Firmware-Version unterstützt werden.

CPU Firmware Version	Unterstützte TLS-Version
V3.0	TLS 1.2, TLS 1.3
V2.9	TLS 1.2, TLS 1.3
V2.8 ... V2.0	TLS 1.2

Unterstützte Verschlüsselungsmethoden und -parameter für die Zertifikatserstellung

Zum Erzeugen des öffentlichen Schlüssels für ein neues Zertifikat stellen Sie im TIA Portal Verschlüsselungsmethode und Verschlüsselungsparameter ein. Diese Zertifikatsparameter sind geräteabhängig und abhängig von der verwendeten Anwendung.

Eine Möglichkeit: Sie wählen in den CPU-Eigenschaften den Bereich "Schutz & Security > Zertifikatsmanager" und erstellen ein neues Gerätezertifikat. Im Dialog "Zertifikate erstellen" finden Sie im Bereich "Zertifikatsparameter" Einstellungen für die Verschlüsselungsmethode und den Verschlüsselungsparameter.

Beispiel: RSA 2048 steht für die asymmetrische RSA-Verschlüsselungsmethode mit Schlüssellänge 2048 bit.

Die folgende Tabelle zeigt in Abhängigkeit von der CPU-Anwendung bzw. Services die unterstützten Verschlüsselungsmethoden und Verschlüsselungsparameter.

Verschlüsselungsmethode/-Parameter S7-1500 (Firmware V3.0)	**Webserver (HTTPS) Secure PG/HMI-Kommunikation Secure OUC**	**OPC UA**
EC prime256v1	ja	nein
EC secp384r1	ja	nein
EC secp256k1	nein	nein
RSA 1024	ja	ja
RSA 2048	ja	ja
RSA 4096	ja	ja
RSA 8192	nein	nein

5.6.1.3 Vertraulichkeit durch Verschlüsselung

Ein wichtiger Beitrag zur Datensicherheit ist die Verschlüsselung der Nachrichten. Wenn verschlüsselte Nachrichten auf dem Transportweg von einem Dritten abgefangen werden, kann dieser potentielle Lauscher nichts mit diesen Nachrichten anfangen.

Es gibt eine Vielzahl mathematischer Verfahren (Algorithmen) zur Verschlüsselung von Nachrichten.

Allen Algorithmen gemeinsam ist, dass sie einen Parameter "Schlüssel" verarbeiten, um Nachrichten zu verschlüsseln bzw. zu entschlüsseln.

- Algorithmus + Schlüssel + Nachricht => verschlüsselte Nachricht
- Verschlüsselte Nachricht + Schlüssel + Algorithmus => (entschlüsselte) Nachricht

Symmetrische Verschlüsselung

Wesentlich beim symmetrischen Verschlüsselungsverfahren ist, dass beide Kommunikationspartner zur Verschlüsselung und Entschlüsselung von Nachrichten denselben Schlüssel anwenden, wie in folgendem Bild dargestellt ist: Bob verwendet denselben Schlüssel zum Verschlüsseln wie Alice zum Entschlüsseln. Allgemein sagt man auch, dass beide Seiten als Geheimnis den geheimen Schlüssel teilen, mit dem sie eine Nachricht verschlüsseln oder entschlüsseln können.

① Bob verschlüsselt seine Nachricht mit dem symmetrischen Schlüssel

② Alice entschlüsselt die verschlüsselte Nachricht mit dem symmetrischen Schlüssel

Bild 5-7 Symmetrische Verschlüsselung

Anschaulich ist das Verfahren mit einem Aktenkoffer vergleichbar, für den Absender und Empfänger jeweils den gleichen Schlüssel haben, um ihn verschließen bzw. öffnen zu können.

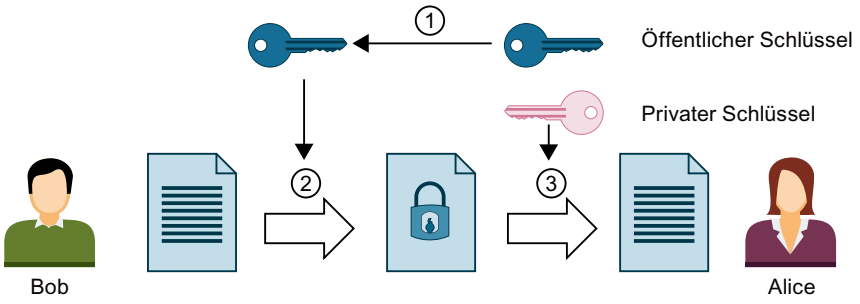
- Vorteil: Symmetrische Verschlüsselungsalgorithmen (z. B. AES, Advanced Encryption Algorithm) arbeiten schnell.
- Nachteile: Wie kommt der Schlüssel zu einem Empfänger, ohne dass er in falsche Hände gerät? Dies ist ein Schlüssel-Verteilungsproblem. Außerdem lässt sich bei einer genügend großer Anzahl abgefangener Nachrichten der Schlüssel erraten, daher muss der Schlüssel ausreichend oft neu vereinbart werden.

Bei einer Vielzahl von Kommunikationspartnern ist außerdem auch eine Vielzahl von Schlüsseln zu verteilen.

Asymmetrische Verschlüsselung

Das asymmetrische Schlüsselverfahren arbeitet mit einem Schlüsselpaar, das aus einem öffentlichen Schlüssel und einem privaten Schlüssel besteht. Im Zusammenhang mit einer PKI wird es auch Public-Key-Verfahren oder nur PKI-Verfahren genannt: Ein Kommunikationspartner, im Bild unten Alice, besitzt einen privaten Schlüssel und einen öffentlichen Schlüssel. Der öffentliche Schlüssel wird der Öffentlichkeit, also jedem potenziellen Kommunikationspartner, zur Verfügung gestellt. Jeder, der den öffentlichen Schlüssel hat, kann Nachrichten für Alice verschlüsseln. Im Bild unten ist das Bob.

Der private Schlüssel von Alice, der von ihr geheim gehalten werden muss, wird von Alice dazu benutzt, um eine an sie adressierte verschlüsselte Nachricht zu entschlüsseln.



① Alice stellt Bob ihren öffentlichen Schlüssel zur Verfügung. Dazu sind keine Vorsichtsmaßnahmen erforderlich: Jeder darf den öffentlichen Schlüssel für Nachrichten an Alice nutzen, wenn er sich sicher ist, dass es tatsächlich der öffentliche Schlüssel von Alice ist.

② Bob verschlüsselt seine Nachricht mit dem öffentlichen Schlüssel von Alice.

③ Alice entschlüsselt die verschlüsselte Nachricht von Bob mit ihrem privaten Schlüssel. Da nur Alice den privaten Schlüssel besitzt und ihn niemals aus der Hand gibt, kann auch nur sie diese Nachricht entschlüsseln. Mit dem privaten Schlüssel kann sie jede Nachricht entschlüsseln, die mit ihrem öffentlichen Schlüssel verschlüsselt wurde - nicht nur die von Bob!

Bild 5-8 Asymmetrische Verschlüsselung

Anschaulich ist das Verfahren mit einem Briefkasten vergleichbar, in den jeder eine Nachricht hineinwerfen kann, aber nur derjenige, der den Schlüssel für den Briefkasten besitzt, kann die Nachricht wieder herausholen.

- Vorteile: Eine mit öffentlichem Schlüssel verschlüsselte Nachricht kann nur vom Inhaber des privaten Schlüssels entschlüsselt werden. Da ein anderer (privater) Schlüssel zum Entschlüsseln benutzt werden muss, ist es auch wesentlich schwerer, den Schlüssel zur Entschlüsselung aus der Menge der verschlüsselten Nachrichten zu erraten. Dadurch müssen die öffentlichen Schlüssel nicht streng geheim aufbewahrt werden, wie es bei symmetrischen Schlüsseln nötig ist.
  Ein weiterer Vorteil besteht in der einfacheren Verteilung von öffentlichen Schlüsseln. Beim asymmetrischen Verfahren ist kein besonders gesicherter Kanal erforderlich zur Übertragung der öffentlichen Schlüssel vom Empfänger zum Sender, der die Nachrichten verschlüsselt. Bei der Verwaltung der Schlüssel gibt es also weniger Aufwand als im symmetrischen Verschlüsselungsverfahren.
- Nachteile: Rechenintensiver Algorithmus (z. B. RSA, benannt nach den drei Mathematikern Rivest, Shamir und Adleman), daher geringere Performance im Vergleich zur symmetrischen Verschlüsselung.

Verschlüsselungsverfahren in der Praxis

In der Praxis wie z. B. beim Webserver der CPU und bei der Secure Open User Communication wird das TLS-Protokoll unterhalb der jeweiligen Anwendungsschicht genutzt. Anwendungsschichten sind z. B. HTTP oder SMTP wie im vorhergehenden Abschnitt gezeigt.

TLS (Transport Layer Security) nutzt eine Kombination aus asymmetrischer Verschlüsselung und symmetrischer Verschlüsselung (hybrides Verschlüsselungsverfahren) zur sicheren Datenübertragung z. B. im Internet und nutzt folgende Unterprotokolle:

- TLS Handshake Protocol, zuständig für Authentifizierung der Kommunikationspartner sowie Aushandeln der später für die Datenübertragung zu nutzenden Algorithmen und Schlüssel auf Basis asymmetrischer Verschlüsselungsverfahren.
- TLS Record Protocol, zuständig für Verschlüsselung der Nutzdaten mittels symmetrischer Verschlüsselungsverfahren und Datenaustausch.

Sowohl die asymmetrische als auch die symmetrische Verschlüsselung gelten als sichere Verschlüsselungsverfahren - es gibt bezüglich der Verfahren keinen prinzipiellen Unterschied hinsichtlich der Sicherheit. Der Grad der Sicherheit hängt ab von den Parametern wie z. B. von der gewählten Schlüssellänge.

Missbrauch von Verschlüsselung

Man sieht einem öffentlichen Schlüssel als eine Bitfolge nicht an, welcher Identität dieser öffentliche Schlüssel zugeordnet ist. Ein Betrüger könnte seinen öffentlichen Schlüssel zur Verfügung stellen und behaupten, er sei eine ganz andere Person. Wenn ein Dritter diesen Schlüssel im Glauben nutzt, er hätte den gewünschten Kommunikationspartner adressiert, landen möglicherweise vertrauliche Informationen bei dem Betrüger. Der Betrüger entschlüsselt dann mit seinem privaten Schlüssel die gar nicht für ihn bestimmte Nachricht und vertrauliche Informationen geraten dann in die falschen Hände.

Um solchen Missbrauch zu verhindern, muss bei den Kommunikationspartnern das Vertrauen geschaffen werden, dass sie es mit dem gewünschten Kommunikationspartner zu tun haben. Um dieses Vertrauen herzustellen, nutzt man in einer PKI digitale Zertifikate.

5.6.1.4 Authentizität und Integrität durch Signaturen

Angriffe von Programmen, welche die Kommunikation zwischen Server und Client abfangen und agieren, als wären sie selbst Client oder Server, nennt man "Man-In-The-Middle Attacks". Wenn die falsche Identität dieser Programme nicht erkannt wird, können sie z. B. wichtige Informationen über das S7-Programm erhalten oder Werte in der CPU setzen und damit eine Maschine oder Anlagen angreifen. Zur Vermeidung solcher Angriffe werden digitale Zertifikate verwendet.

Secure Communication verwendet digitale Zertifikate, die dem Standard X.509 der International Telecommunication Union (ITU) entsprechen. Damit lässt sich die Identität eines Programms, eines Rechners oder einer Organisation prüfen (authentifizieren).

Wie Zertifikate Vertrauen schaffen

Die wesentliche Aufgabe von X.509-Zertifikaten ist es, eine Identität mit den Daten eines Zertifikatsinhabers (z. B. E-Mail-Adresse, Rechnername) an den öffentlichen Schlüssel der Identität zu binden. Identitäten können Personen, Rechner oder Maschinen sein.

Zertifikate werden von Zertifizierungsstellen (Certificate Authority, CA) oder dem Inhaber des Zertifikates selber ausgegeben. PKI-Systeme legen fest, wie die Nutzer den Zertifizierungsstellen und den von ihnen herausgegebenen Zertifikaten trauen können.

Der Weg zum Zertifikat:

1. Wer ein Zertifikat haben möchte, reicht über eine bei der Zertifizierungsstelle angeschlossene Registrierungsstelle einen Zertifikatsantrag ein.
2. Die Zertifizierungsstelle bewertet Antrag und Antragsteller anhand festgelegter Kriterien.
3. Wenn sich die Identität des Antragstellers eindeutig feststellen lässt, beglaubigt die Zertifizierungsstelle diese Identität durch Ausstellen eines signierten Zertifikats. Aus dem Antragsteller ist nun der Zertifikatsinhaber geworden.

Im folgenden Bild ist der Sachverhalt vereinfacht dargestellt. Nicht gezeigt ist die Möglichkeit, wie Alice die digitale Signatur prüfen kann.

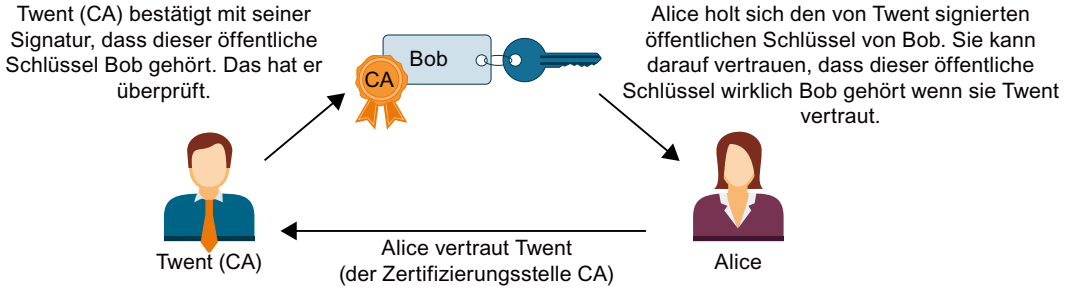


Bild 5-9 Signieren eines Zertifikates durch eine Zertifizierungsstelle

Selbstsignierte Zertifikate

Selbstsignierte Zertifikate sind Zertifikate, dessen Signatur vom Zertifikateinhaber stammt und nicht von einer unabhängigen Zertifizierungsstelle.
Beispiele:

- Sie können ein Zertifikat erstellen und selbst signieren, um zum Beispiel Nachrichten zu einem Kommunikationspartner zu verschlüsseln. Im Beispiel oben könnte Bob selber (statt Twent) sein Zertifikat mit seinem privaten Schlüssel signieren. Alice kann mit Hilfe von Bobs öffentlichem Schlüssel prüfen, dass Signatur und öffentlicher Schlüssel von Bob zusammenpassen. Für eine einfache Anlagen-interne Kommunikation, die verschlüsselt ablaufen soll, ist das ausreichend.
- Bei einem Stammzertifikat handelt es sich z. B. um ein von der Zertifizierungsstelle (Aussteller) selbstsigniertes Zertifikat, welches den öffentlichen Schlüssel der Zertifizierungsstelle enthält.

Besonderheiten von selbstsignierten Zertifikaten

Die Attribute "CN" (Common Name of Subject) für den Zertifikatsinhaber und "Issuer" (Aussteller) von selbstsignierten Zertifikaten sind identisch: Sie haben Ihr Zertifikat ja selbst signiert. Das Feld "CA" (Certificate Autority) für die Zertifizierungsstelle muss auf "False" stehen; das selbstsignierte Zertifikat soll ja nicht dazu benutzt werden, andere Zertifikate zu signieren.
Selbstsignierte Zertifikate sind nicht in eine PKI-Hierarchie eingebettet.

Inhalte von Zertifikaten

Ein Zertifikat nach dem Standard X.509 V3, der Standard, der auch von STEP 7 bzw. den S7-1500 CPUs genutzt wird, besteht im Wesentlichen aus folgenden Teilen:

- Öffentlicher Schlüssel
- Angaben über den Zertifikatsinhaber (d. h. den Schlüsselinhaber); das ist z. B. der Common Name (CN) of Subject
- Attribute wie Seriennummer und Gültigkeitsdauer
- Digitale Signatur (Beglaubigung) der Zertifizierungsstelle (CA), dass die Angaben stimmen.

Daneben gibt es Erweiterungen, z. B.

- Angabe, für welchen Verwendungszweck der öffentliche Schlüssel verwendet werden darf (Key Usage), z. B. zum Signieren oder zur Schlüssel-Verschlüsselung.
  Wenn Sie mit STEP 7 ein neues Zertifikat erstellen, z. B. im Kontext Secure Open User Communication, wählen Sie aus der Liste der möglichen Verwendungszwecke den treffenden Eintrag aus der Liste aus, z. B. "TLS".
- Angabe eines "Alternativen Namens des Zertifikatsinhabers" ("SAN", Subject Alternative Name), der z. B. bei der sicheren Kommunikation mit Webservern (HTTP over TLS) dazu genutzt wird, um sicherzustellen, dass das Zertifikat auch dem Webserver gehört, der im URL der Adresszeile des Web-Browsers angegeben wurde.

Wie Signaturen erzeugt und verifiziert werden

Die technische Voraussetzung, dass Zertifikate geprüft werden können, liefert die asymmetrische Schlüsselverwendung: Am Beispiel des Zertifikats "MyCert" werden die Prozesse "Signieren" und "Signatur prüfen" gezeigt.

Signatur erzeugen:

1. Der Aussteller des Zertifikates "MyCert" erzeugt aus den Daten des Zertifikats mit einer bestimmten Hash-Funktion (z.B. SHA-1, Secure Hash Algorithm) einen Hash-Wert.
   Der Hash-Wert ist eine Bitfolge mit konstanter Länge. Die stets gleiche Länge des Hash-Wertes bietet den Vorteil, dass das Signieren des Hash-Wertes immer die gleiche Zeit beansprucht.
2. Aus dem so erzeugten Hash-Wert erzeugt der Aussteller des Zertifikats mit Hilfe des private Schlüssels eine digitale Signatur. Häufig wird dazu das RSA-Signaturverfahren benutzt.
3. Die digitale Signatur wird im Zertifikat gespeichert. Dadurch ist das Zertifikat signiert.

Signatur verifizieren:

1. Der Prüfer des Zertifikats "MyCert" besorgt sich das Zertifikat des Ausstellers und damit den öffentlichen Schlüssel.
2. Mit demselben Hash-Algorithmus, der bei der Signierung verwendet wurde (z.B. SHA-1), wird aus den Daten des Zertifikats erneut ein Hash-Wert gebildet.
3. Dieser Hash-Wert wird dann verglichen mit dem Hash-Wert, der mit Hilfe des öffentlichen Schlüssel des Zertifikate-Ausstellers und dem Signaturalgorithmus zur Prüfung der Signatur ermittelt wird.
4. Wenn die Prüfung der Signatur ein positives Ergebnis liefert, ist sowohl die Identität des Zertifikatsinhabers als auch die Integrität, d. h. die Echtheit und Unverfälschtheit des Zertifikate-Inhalts nachgewiesen. Jeder, der den öffentlichen Schlüssel, d. h. das Zertifikat der Zertifizierungsstelle hat, kann die Signatur prüfen und so erkennen, dass das Zertifikat tatsächlich von der Zertifizierungsstelle signiert wurde.

Im folgenden Bild ist gezeigt, wie Alice mit Hilfe des öffentlichen Schlüssels des Zertifikats von Twent (verkörpert die Zertifizierungsstelle CA) die Signatur an Bobs öffentlichem Schlüssel verifiziert. Voraussetzung für die Prüfung ist also die Verfügbarkeit des Zertifikats der Zertifizierungsstelle zum Prüfungszeitpunkt. Die Validierung selbst läuft automatisch in der TLS-Session ab.

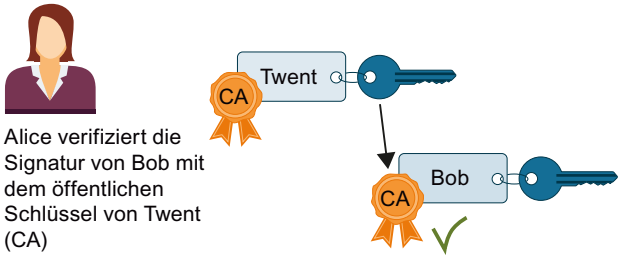


Bild 5-10 Verifizierung eines Zertifikats über den öffentichen Schlüssel des Zertifikates einer Zertifizierungsstelle

Signatur von Nachrichten

Die oben beschriebene Methode zum Signieren und Verifizieren von Zertifikaten nutzt die TLS-Session auch zum Signieren und Verifizieren von Nachrichten:

Wenn von einer Nachricht ein Hash-Wert erzeugt wird und dieser Hash-Wert mit dem privaten Schlüssel des Senders verschlüsselt und an die originale Nachricht angehängt wird, ist der Empfänger der Nachricht in der Lage, die Integrität (Unversehrtheit) der Nachricht zu erkennen. Der Empfänger entschlüsselt den Hash-Wert mit dem öffentlichen Schlüssel des Senders, bildet aus der empfangenen Nachricht selbst den Hash-Wert und vergleicht beide Werte. Wenn die Werte unterschiedlich sind, wurde die Nachricht oder der verschlüsselte Hash-Wertauf dem Transportweg verfälscht.

Kette von Zertifikaten bis zum Stammzertifikat

Die Zertifikate einer PKI sind häufig hierarchisch organisiert: An der Spitze der Hierarchie stehen Stammzertifikate, auch Wurzelzertifikate oder Root-Zertifikate genannt. Das sind Zertifikate, die nicht durch eine übergeordnete Zertifizierungsstelle beglaubigt werden. Zertifikatsinhaber und Zertifikatsaussteller von Stammzertifikaten sind identisch. Stammzertifikate genießen absolutes Vertrauen, sie sind der "Anker" des Vertrauens und müssen deshalb beim Empfänger als vertrauenswürdige Zertifikate bekannt sein. Sie werden in einem für vertrauenswürdige Zertifikate vorgesehenen Bereich gespeichert.

Die Funktion von Stammzertifikaten kann je nach PKI z. B. darin bestehen, Zertifikate von untergeordneten Zertifizierungsstellen, sogenannte Zwischenzertifikate, zu signieren. Damit überträgt sich das Vertrauen vom Stammzertifikat auf das Zwischenzertifikat. Ein Zwischenzertifikat kann genauso gut ein Zertifikat signieren wie ein Stammzertifikat, daher werden beide auch "CA-Zertifikate" genannt.

Diese Hierarchie lässt sich über mehrere Zwischenzertifikate weiterführen bis zum End-Entity Zertifikat. Das End-Entity Zertifikat ist das Zertifikat des Benutzers, der identifiziert werden soll.

Bei der Validierung wird die Hierarchie in umgekehrter Richtung durchlaufen: Wie oben beschrieben wird der Zertifikatsaussteller ermittelt, mit seinem öffentlichen Schlüssel die Signatur geprüft, dann das Zertifikat des übergeordneten Zertifikatsausstellers ermittelt bis die Vertrauenskette bis zum Stammzertifikat durchlaufen ist.

Fazit: Die Kette von Zwischenzertifikaten bis zum Stammzertifikat, der Zertifikate-Pfad, muss in jedem Gerät vorhanden sein, das ein End-Entity-Zertifikat vom Kommunikationspartner validieren soll, unabhängig davon, welche Art von Secure Communication Sie projektieren.

5.6.2 Verwalten von Zertifikaten

5.6.2.1 Wissenswertes zum Zertifikatsmanagement

Dieser Abschnitt zeigt die verfügbaren Zertifikatsmanagement-Möglichkeiten einer S7-1500 CPU in Abhängigkeit vom genutzten Service (CPU-Applikation) und von den Versionen des TIA Portals / der CPU-Firmware.

Überblick Zertifikatsmanagement-Möglichkeiten

Zertifikate für die gesicherte Kommunikation können Sie seit TIA Portal Version V14 und CPU-FW-Version V2.0 für verschiedene Services der S7-1500 CPU im TIA Portal verwalten und in die CPU laden.

Das TIA Portal ab Version V17 zusammen mit den S7-1500-CPUs ab FW Version ab V2.9 unterstützen eine weitere Möglichkeit des Zertifikatsmanagements: Sie können mit GDS-Push-Methoden zur Laufzeit Zertifikate in die CPU übertragen bzw. erneuern, ohne die CPU neu laden zu müssen.

Über denselben Weg können Sie ab TIA Portal Version V18 auch Webserver-Zertifikate für eine S7-1500 CPU (ab Firmware V3.0) übermitteln.

Die folgende Tabelle gibt einen Überblick über die Möglichkeiten des Zertifikatsmanagements in Abhängigkeit vom genutzten Service und der TIA Portal- bzw. CPU-Firmware-Version.

Service	Zertifikatsmanagment mit TIA Portal (TIA Portal Version / S7-1500 CPU-FW-Version)	Zertifikatsmanagement mit OPC UA GDS Push Methoden (TIA Portal Version / S7-1500 CPU-FW- Version)
Webserver	ab V14 / ab V2.0	ab V18 / ab V3.0
Secure OUC-Kommunikation	ab V14 / ab V2.0	-
OPC UA-Server	ab V14 / ab V2.0	ab V17 / ab V2.9
OPC UA-Client	ab V15.1 / ab V2.6	-
Secure PG/HMI Communication	ab V17 / ab V2.9	-

Weitere Informationen

Eine Beschreibung des Zertifikatsmanagement mit GDS-Push Methoden finden Sie hier: Zertifikatsmanagement über Global Discovery Server (GDS) (Seite 193)

5.6.2.2 Zertifikatsmanagement mit TIA Portal

STEP 7 ab Version V14 zusammen mit den S7-1500-CPUs ab FW Version 2.0 unterstützen die Internet-PKI (RFC 5280) soweit, dass eine S7-1500-CPU in der Lage ist, mit Geräten zu kommunizieren, die ebenfalls die Internet PKI unterstützen.

Die Nutzung von X.509-Zertifikaten z. B. zur Prüfung von Zertifikaten wie in den vorangegangenen Abschnitten beschrieben ist eine Konsequenz hieraus.

STEP 7 ab V14 nutzt eine PKI ähnlich der Internet PKI. Nicht unterstützt werden z. B. Certificate Revocation Lists (CRLs).

Zertifikate erstellen oder zuweisen im TIA Portal

Für Geräte mit Security-Eigenschaften wie z. B. eine S7-1500-CPU ab Firmware V2.0 erstellen Sie in STEP 7-Zertifikate für verschiedene Anwendungen.

Folgende Bereiche im Inspektorfenster der CPU erlauben das Erstellen neuer oder das Auswählen vorhandener Zertifikate:

- "Webserver > Security" - für die Erzeugung bzw. Zuweisung von Webserver-Zertifikaten.
- "Schutz & Security > Verbindungsmechanismen" - für die Erzeugung bzw. Zuweisung von PLC-Kommunikationszertifikaten (Secure PG/HMI-Kommunikation, ab TIA Portal V17).
- "Schutz & Security > Zertifikatsmanager" - für die Erzeugung bzw. Zuweisung aller Arten von Zertifikaten; voreingestellt für das Erstellen von Zertifikaten sind TLS-Zertifikate für Secure Open User Communication.
- "OPC UA > Server > Security" - für die Erzeugung bzw. Zuweisung von OPC UA Server-Zertifikaten.

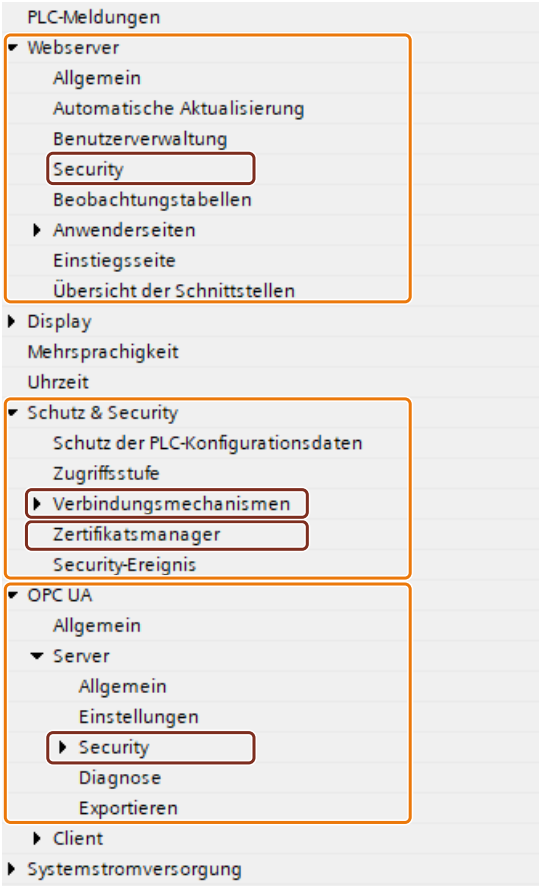


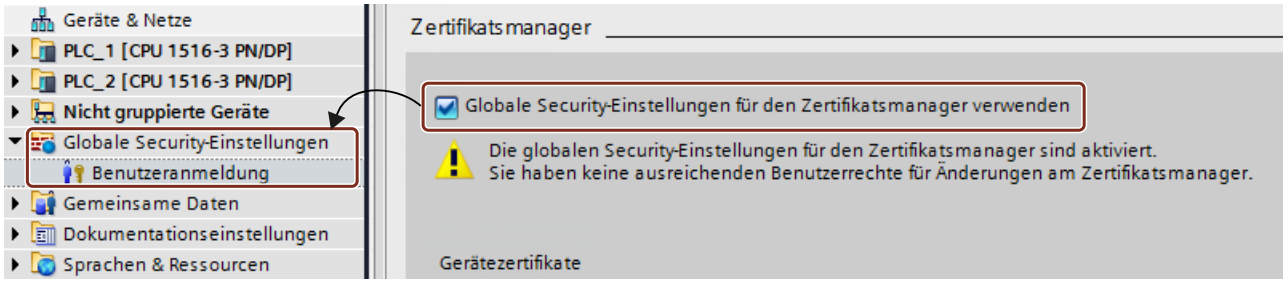
Bild 5-11 Security-Einstellungen für eine S7-1500 CPU in STEP 7

Besonderheit des Bereichs "Schutz & Security > Zertifikatsmanager"

Nur in diesem Bereich des Inspektorfensters schalten Sie zwischen dem globalen, d. h. projektweitem und lokalen, d. h. gerätespezifischem Zertifikatsmanager um (Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden"). Die Option entscheidet darüber, ob Sie Zugriff auf alle Zertifikate im Projekt haben oder nicht.

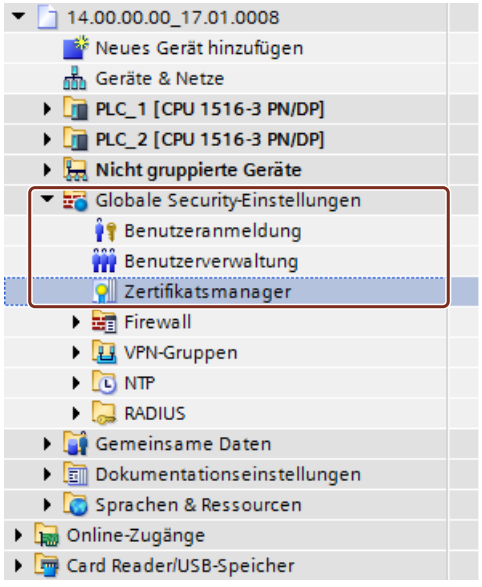
- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen **nicht** verwenden, haben Sie nur Zugriff auf den lokalen Zertifikatsspeicher der CPU. Sie haben keinen Zugriff z. B. auf importierte Zertifikate oder Stammzertifikate. Ohne diese Zertifikate ist nur eine eingeschränkte Funktionalität verfügbar; Sie können z. B. nur selbstsignierte Zertifikate erzeugen.
- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen verwenden und Sie z. B. als Administrator angemeldet sind, haben Sie Zugriff auf den globalen, projektweiten Zertifikatsspeicher. Sie können z. B. der CPU importierte Zertifikate zuweisen oder Zertifikate erstellen, die von der Projekt-CA (Zertifizierungsstelle des Projekts) ausgestellt und signiert sind.

Das folgende Bild zeigt, wie nach dem Aktivieren der Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" im Inspektorfenster der CPU die "Globalen Security-Einstellungen" in der Projektnavigation erscheinen.



Wenn Sie in der Projektnavigation auf "Benutzeranmeldung" unterhalb der Globalen Security-Einstellungen doppelklicken und sich anmelden, erscheint dort auch unter anderem eine Zeile "Zertifikatsmanager".

Mit einem Doppelklick auf die Zeile "Zertifikatsmanager" erhalten Sie Zugang zu allen Zertifikaten im Projekt, aufgeteilt in die Register "CA" (Zertifizierungsstellen), "Gerätezertifikate" und "Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen".



Private Schlüssel

STEP 7 erzeugt private Schlüssel beim Erzeugen von Gerätezertifikaten bzw. Server-Zertifikaten (End-Entity-Zertifikate). Wo der private Schlüssel verschlüsselt gespeichert wird, hängt von der Verwendung der globalen Security-Einstellungen für den Zertifikatsmanager ab:

- Wenn Sie die globalen Security-Einstellungen verwenden, dann wird der private Schlüssel im globalen (projektweiten) Zertifikatsspeicher verschlüsselt gespeichert.
- Wenn Sie die globalen Security-Einstellungen nicht verwenden, dann wird der private Schlüssel im lokalen (CPU-spezifischen) Zertifikatsspeicher verschlüsselt gespeichert.

Angezeigt wird das Vorhandensein des privaten Schlüssels, der z. B. für die Entschlüsselung von Daten notwendig ist, in der Spalte "Privater Schlüssel" im Register "Gerätezertifikate" des Zertifikatsmanagers in den globalen Security-Einstellungen.

Beim Laden der Hardware-Konfiguration wird das Gerätezertifikat, der öffentliche Schlüssel sowie der private Schlüssel in die CPU geladen.

ACHTUNG
Aktivierung der Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" - Konsequenzen Die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" beeinflusst die bisher verwendeten privaten Schlüssel: Wenn Sie bereits Zertifikate erstellt haben ohne Verwendung des Zertifikatsmanagers in den globalen Security-Einstellungen und dann die Option zur Verwendung des Zertifikatsmanagers umstellen, dann gehen die privaten Schlüssel verloren und die Zertifikats-ID kann sich ändern! Eine Warnung macht Sie auf diesen Sachverhalt aufmerksam. Legen Sie daher zu Beginn der Projektierung fest, welche Option zum Zertifikatsmanagement erforderlich ist.

5.6.2.3 Beispiele zum Verwalten von Zertifikaten

Wie in den vorangegangenen Abschnitten erläutert, sind Zertifikate für jede Art von Secure Communication erforderlich. Im Folgenden wird beispielhaft gezeigt, wie Sie mit STEP 7 die Zertifikate handhaben, damit die Voraussetzungen für Secure Open User Communication gegeben sind.

Dabei wird im Folgenden unterschieden, um welche Geräte es sich bei den beteiligten Kommunikationspartnern handelt. Die jeweiligen Schritte zum Versorgen der Kommunikationsteilnehmer mit den benötigten Zertifikaten sind jeweils beschrieben. Vorausgesetzt wird immer eine S7-1500 CPU bzw. ein S7-1500 Software Controller ab Firmware Version 2.0.

Allgemein gilt:

Während des Aufbaus einer sicheren Verbindung ("Handshake") übermitteln die Kommunikationspartner in der Regel nur ihre End-Entity-Zertifikate (Gerätezertifikate).

Daher müssen sich die zur Prüfung des übermittelten Gerätezertifikats notwendigen CA-Zertifikate im Zertifikatsspeicher des jeweiligen Kommunikationspartners befinden.

HINWEIS

In der CPU muss das aktuelle Datum / die aktuelle Uhrzeit eingestellt sein.

Wenn Sie Secure Communication nutzen (z. B. HTTPS, Secure OUC, OPC UA), dann achten Sie darauf, dass die betroffenen Baugruppen über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.

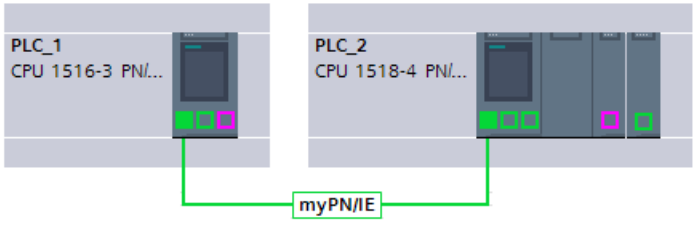
Secure Open User Communication zwischen 2 S7-1500 CPUs

Zwei S7-1500-CPUs PLC_1 und PLC_2 sollen miteinander Daten austauschen über Secure Open User Communication.

Die erforderlichen Gerätezertifikate erzeugen Sie mit STEP 7 und weisen Sie den CPUs zu wie im Folgenden beschrieben.

Es werden STEP 7-Projekt-Zertifizierungsstellen (CA des Projekts) verwendet, um die Gerätezertifikate zu signieren.

Die Zertifikate sind im Anwenderprogramm (Kommunikationsanweisung TCON in Verbindung mit dem zugehörigen Systemdatentyp, z. B. TCON_IPV4_SEC) über ihre Zertifikats-ID zu referenzieren. Die Zertifikats-ID vergibt STEP 7 automatisch beim Erzeugen oder beim Anlegen von Zertifikaten.



Vorgehen

STEP 7 lädt automatisch die erforderlichen CA-Zertifikate zusammen mit der Hardware-Konfiguration in die beteiligten CPUs, so dass die Voraussetzungen für die Zertifikatsprüfung für beide CPUs gegeben sind. Sie müssen also nur die Gerätezertifikate für die jeweilige CPU erzeugen - alles Übrige erledigt STEP 7 für Sie.

1. Markieren Sie PLC_1 und aktivieren Sie im Bereich "Schutz & Security" die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden".
2. Melden Sie sich in der Projektnavigation im Bereich "Globale Security-Einstellungen" als Benutzer an. Bei einem neuen Projekt ist bei der erstmaligen Anmeldung die Rolle "Administrator" vorgesehen.
3. Kehren Sie zurück zur PLC-1 in den Bereich "Schutz & Security". Klicken Sie in der Tabelle "Gerätezertifikate" in eine leere Zeile der Spalte "Zertifikatsinhaber", um ein neues Zertifikat hinzuzufügen.
4. Klicken Sie in der Klappliste zur Auswahl eines Zertifikats auf die Schaltfläche "Hinzufügen".
   Der Dialog "Neues Zertifikat erzeugen" wird geöffnet.
5. Belassen Sie die Voreinstellungen in diesem Dialog; sie sind auf die Verwendung für Secure Open User Communication zugeschnitten (Verwendung: TLS).
   Tipp: Ergänzen Sie den voreingestellten Namen des Zertifikatsinhabers, in diesem Fall den CPU-Namen. Belassen Sie zur besseren Unterscheidung den voreingestellten CPU-Namen für den Fall, dass Sie viele Gerätezertifikate verwalten müssen.
   Beispiel: PLC_1/TLS wird zu PLC_1-SecOUC-Chassis17FactoryState.
6. Übersetzen Sie die Konfiguration.
   Das Gerätezertifikat und das CA-Zertifikat sind Bestandteil der Konfiguration.
7. Wiederholen Sie die beschriebenen Schritte für PLC_2.

Im nächsten Schritt müssen Sie die Anwenderprogramme für den Datenaustausch erstellen und die Konfigurationen zusammen mit dem Programm laden.

Selbstsignierte Zertifikate statt CA-Zertifikate verwenden

Beim Anlegen von Gerätezertifikaten können Sie die Option "Selbstsigniert" wählen. Selbstsignierte Zertifikate können Sie erstellen, ohne für die globalen Security-Einstellungen angemeldet zu sein. Diese Vorgehensweise wird nicht empfohlen, da die so erstellten Zertifikate nicht im globalen Zertifikatsspeicher vorhanden sind und daher nicht direkt einer Partner-CPU zugewiesen werden können.

Wie oben beschrieben sollten Sie sorgfältig den Namen des Zertifikatsinhabers wählen, um das richtige Zertifikat zweifelsfrei einem Gerät zuordnen zu können.

Für selbstsignierte Zertifikate ist keine Prüfung mit den CA-Zertifikaten des STEP 7-Projekts möglich. Um selbstsignierte Zertifikate prüfen zu können, müssen Sie für jede CPU das selbstsignierte Zertifikat des Kommunikationspartners in die Liste der vertrauenswürdigen Partnergeräte aufnehmen. Dazu müssen Sie die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden" aktiviert haben und in den globalen Security-Einstellungen als Benutzer angemeldet sein.

Um das selbstsignierte Zertifikat vom Kommunikationspartner der CPU hinzuzufügen, gehen sie folgendermaßen vor:

1. Markieren Sie PLC_1 und navigieren Sie zur Tabelle "Zertifikate von Partnergeräten" im Bereich "Schutz & Security".
2. Klicken Sie in eine leere Zeile der Spalte "Zertifikatsinhaber", um die Klappliste zum Hinzufügen oder Auswählen von Zertifikaten zu öffnen.
3. Wählen Sie aus der Klappliste das selbstsignierte Zertifikat des Kommunikationspartners und bestätigen Sie die Auswahl.

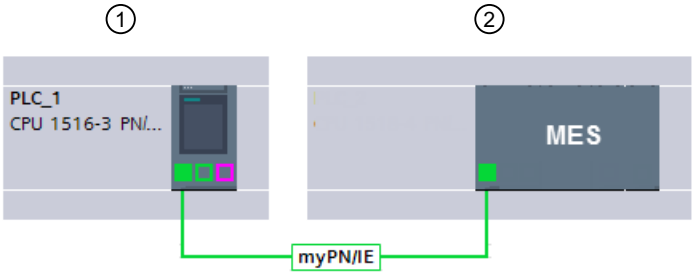
Im nächsten Schritt müssen Sie die Anwenderprogramme für den Datenaustausch erstellen und die Konfigurationen zusammen mit dem Programm laden.

Secure Open User Communication zwischen S7-1500 CPU als TLS-Client und Fremdgerät als TLS-Server

Zwei Geräte sollen miteinander Daten austauschen über eine TLS-Verbindung bzw. TLS-Sitzung, z. B. zum Austausch von Rezepturen, Produktionsdaten oder Qualitätsdaten:

- Eine S7-1500-CPU (PLC_1) als TLS-Client; die CPU nutzt Secure Open User Communication
- Ein Fremdgerät (z. B. ein Manufacturing Execution System (MES)) als TLS-Server

Die S7-1500 CPU baut als TLS-Client die TLS-Verbindung/Sitzung zum MES-System auf.



① TLS-Client

② TLS-Server

Zur Authentifizierung des TLS-Servers benötigt die S7-1500 CPU die CA-Zertifikate des MES-Systems: Das Stammzertifikat und ggf. die Zwischenzertifikate zur Prüfung des Zertifikatspfads.

Diese Zertifikate müssen Sie in den globalen Zertifikatspeicher der S7-1500 CPU importieren.

Um Zertifikate des Kommunikationspartners zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu importierende Zertifikat die passende Tabelle (Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen).
3. Öffnen Sie in der Tabelle mit Rechtsklick das Kontextmenü. Klicken Sie auf "Importieren" und importieren Sie das benötigte Zertifikat bzw. die benötigten CA-Zertifikate.
   Das Zertifikat erhält durch den Import eine Zertifikats-ID und kann im nächsten Schritt einer Baugruppe zugewiesen werden.
4. Markieren Sie PLC_1 und navigieren Sie zur Tabelle "Zertifikate von Partnergeräten" im Bereich "Schutz & Security".
5. Klicken Sie in eine leere Zeile der Spalte "Zertifikatsinhaber", um die importierten Zertifikate hinzuzufügen.
6. Wählen Sie aus der Klappliste die benötigten CA-Zertifikate des Kommunikationspartners und bestätigen Sie die Auswahl.

Optional kann das MES-System zur Authentifizierung der CPU (d. h. des TLS-Clients) ebenfalls ein Gerätezertifikat der CPU anfordern. Dem MES-System müssen in diesem Fall die CA-Zertifikate der CPU zur Verfügung gestellt werden. Voraussetzung für den Import der Zertifikate ins MES-System ist ein vorhergehender Export der CA-Zertifikate aus dem STEP 7-Projekt der CPU. Gehen sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu exportierende Zertifikat die passende Tabelle (CA-Zertifikate).
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Exportieren".
5. Wählen Sie das Exportformat des Zertifikats.

Im nächsten Schritt müssen Sie die Anwenderprogramme für den Datenaustausch erstellen und die Konfigurationen zusammen mit dem Programm laden.

Secure Open User Communication zwischen S7-1500 CPU als TLS-Server und Fremdgerät als TLS-Client

Wenn die S7-1500 CPU als TLS-Server agiert und das Fremdgerät, z. B. ein ERP-System (Enterprise Resource Planning System) die TLS-Verbindung/Sitzung aufbaut, benötigen Sie folgende Zertifikate:

- Für die S7-1500 CPU erzeugen Sie ein Gerätezertifikat (Server-Zertifikat) mit privatem Schlüssel und laden es mit der Hardware-Konfiguration in die S7-1500 CPU. Sie verwenden beim Erzeugen des Server-Zertifikats die Option "Von Zertifizierungsstelle signiert".
  Der private Schlüssel wird für den Schlüsselaustausch benötigt, wie im Bild für das Beispiel "HTTP over TLS" erläutert.
- Für das ERP-System müssen Sie das CA-Zertifikat des STEP 7-Projekts exportieren und in das ERP-System importieren/laden. Mit dem CA-Zertifikat prüft das ERP-System das Serverzertifikat der S7-1500, das von der CPU an das ERP-System während des Aufbaus der TLS-Verbindung/Sitzung übermittelt wird.

① TLS-Server
② TLS-Client

Bild 5-12 Secure OUC zwischen einer S7-1500 CPU und einem ERP-System

Die Beschreibung der erforderlichen Handlungsschritte entnehmen Sie dem vorangegangenen Abschnitten.

Secure Open User Communication zu einem Mailserver (SMTP over TLS)

Eine S7-1500 CPU kann mit der Kommunikationsanweisung TMAIL-C eine sichere Verbindung zu einem E-Mail-Server aufbauen.
Die Systemdatentypen TMail_V4_SEC und TMail_QDN_SEC ermöglichen Ihnen, den Partnerport des E-Mail-Servers zu bestimmen und so den E-Mail-Server über "SMTP over TLS" zu erreichen.

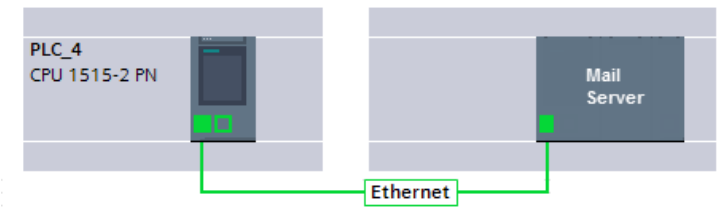


Bild 5-13 Secure OUC zwischen einer S7-1500 CPU und einem Mail Server

Notwendige Voraussetzung für eine sichere E-Mail-Verbindung ist ein Import des Stammzertifikats und der Zwischenzertifikate vom Mailserver (Provider) in den globalen Zertifikatsspeicher der S7-1500 CPU. Mithilfe dieser Zertifikate kann die CPU das Serverzertifikat prüfen, das beim Aufbau der TLS-Verbindung/Sitzung vom Mail-Server gesendet wird.
Um Zertifikate des Mailservers zu importieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu importierende Zertifikat die passende Tabelle (Vertrauenswürdige Zertifikate und Stammzertifizierungsstellen).
3. Öffnen Sie in der Tabelle mit Rechtsklick das Kontextmenü. Klicken Sie auf "Importieren" und importieren Sie das benötigte Zertifikat bzw. die benötigten CA-Zertifikate.
   Das Zertifikat erhält durch den Import eine Zertifikats-ID und kann im nächsten Schritt einer Baugruppe zugewiesen werden.
4. Markieren Sie PLC_1 und navigieren Sie zur Tabelle "Zertifikate von Partnergeräten" im Bereich "Schutz & Security".

5. Klicken Sie in eine leere Zeile der Spalte "Zertifikatsinhaber", um die importierten Zertifikate hinzuzufügen.
6. Wählen Sie aus der Klappliste die benötigten CA-Zertifikate des Kommunikationspartners und bestätigen Sie die Auswahl.

Im nächsten Schritt müssen Sie die Anwenderprogramme für die E-Mail-Client-Funktion der CPU erstellen und die Konfigurationen zusammen mit dem Programm laden.

5.6.2.4 Wie Kommunikation mit Zertifikaten funktioniert: HTTP over TLS

Im Folgenden wird gezeigt, wie die beschriebenen Mechanismen genutzt werden, um eine Secure Communication zwischen einem Webbrowser und dem Webserver einer S7-1500 CPU aufzubauen.

Zunächst sind die Änderungen für die Option "Zugriff nur über HTTPS" in STEP 7 beschrieben. Ab STEP 7 V14 haben Sie die Möglichkeit, Einfluss auf das Server-Zertifikat des Webservers einer S7-1500-CPU ab Firmware V2.0 zu nehmen: Das Server-Zertifikat wird ab diesen Versionen mit STEP 7 erzeugt.

Außerdem wird gezeigt, welche Prozesse beim Aufruf einer Webseite des Webservers der CPU mit einem Webbrowser eines PCs über eine verschlüsselte HTTPS-Verbindung ablaufen.

Umgang mit Webserver-Zertifikaten für S7-1500 CPUs ab FW V2.0

Für S7-1500 CPUs mit einem Firmware-Stand kleiner V2.0 haben Sie beim Einstellen der Webserver-Eigenschaften ohne Voraussetzungen die Option "Zugriff nur über HTTPS zulassen" wählen können.

Mit der Hantierung von Zertifikaten sind Sie bei diesen CPUs nicht in Berührung gekommen; die erforderlichen Zertifikate für den Webserver erzeugt die CPU automatisch.

Bei S7-1500 CPUs ab Firmware V2.0 erzeugt STEP 7 das Server-Zertifikat (End-Entity-Zertifikat) für die CPU. In den Eigenschaften der CPU (Webserver > Security) weisen Sie dem Webserver ein Server-Zertifikat zu.

Da immer ein Server-Zertifikatsname voreingestellt ist, ändert sich an der einfachen Projektierung vom Webserver nichts: Sie aktivieren den Webserver, die Option "Zugriff nur über HTTPS zulassen" ist voreingestellt - STEP 7 generiert beim Übersetzen ein Server-Zertifikat mit dem voreingestellten Namen.

Unabhängig davon, ob Sie den Zertifikatsmanager in den globalen Security-Einstellungen verwenden oder nicht: STEP 7 hat alle Informationen, um das Server-Zertifikat erzeugen zu können.

Zusätzlich haben Sie die Möglichkeit, die Eigenschaften des Server-Zertifikats zu bestimmen, z. B. den Namen oder die Gültigkeitsdauer.

HINWEIS

In der CPU muss das aktuelle Datum / die aktuelle Uhrzeit eingestellt sein.

Wenn Sie Secure Communication nutzen (z. B. HTTPS, Secure OUC, OPC UA), dann achten Sie darauf, dass die betroffenen Baugruppen über die aktuelle Uhrzeit und das aktuelle Datum verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.

Laden des Webserver-Zertifikats

Mit dem Laden der Hardware-Konfiguration in die CPU wird das von STEP 7 erzeugte Server-Zertifikat automatisch mitgeladen.

- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen verwenden, signiert die Zertifizierungsstelle des Projekts (CA-Zertifikat) das Server-Zertifikat des Webservers. Beim Laden wird das CA-Zertifikat des Projekts automatisch mitgeladen.
- Wenn Sie den Zertifikatsmanager in den globalen Security-Einstellungen nicht verwenden, erzeugt STEP 7 das Server-Zertifikat als selbst signiertes Zertifikat.

Wenn Sie den Webserver der CPU über die IP-Adresse der CPU adressieren, dann ist mit jeder Änderung der IP-Adresse einer Ethernet-Schnittstelle der CPU ein neues Serverzertifikat (End-Entity-Zertifikat) zu erzeugen und zu laden. Der Grund ist, dass sich mit der IP-Adresse die Identität der CPU ändert - und die muss nach den Regeln der PKI beglaubigt (signiert) werden.
Sie können dieses Problem vermeiden, indem Sie die CPU über einen Domainnamen adressieren statt mit ihrer IP-Adresse, z. B. "myconveyer-cpu.room13.myfactory.com". Dazu müssen Sie die Domainnamen der CPUs über einen DNS-Server verwalten.

Webbrowser mit CA-Zertifikat des Webservers versorgen

Im Webbrowser sollte der Anwender, der per HTTPS auf die Webseiten der CPU zugreift, das CA-Zertifikat der CPU installieren. Wenn kein Zertifikat installiert ist, wird nämlich eine Warnung angezeigt, mit der Empfehlung, die Seite nicht zu benutzen. Um die Seite zu sehen, muss der Anwender dann explizit eine "Ausnahme hinzufügen".
Das gültige Wurzelzertifikat (Root Certificate) erhält der Anwender als Download auf der Webseite "Intro" des CPU-Webservers unter "Zertifikat herunterladen".
Eine andere Möglichkeit bietet STEP 7: Exportieren Sie das CA-Zertifikat des Projekts mit dem Zertifikatsmanager in den globalen Security-Einstellungen in STEP 7. Anschließend importieren Sie das CA-Zertifikat in den Browser.

Ablauf der Secure Communication

Das folgende Bild zeigt vereinfacht den prinzipiellen Ablauf des Aufbaus der Kommunikation ("Handshake") mit dem Schwerpunkt auf dem Aushandeln der Schlüssel, die zum Datenaustausch (hier über HTTP over TLS) verwendet werden.
Der Ablauf ist aber prinzipiell übertragbar auf alle Kommunikationsmöglichkeiten, die auf der Nutzung von TLS basieren, also auch für Secure Open User Communication (siehe Grundlagen zur Secure Communication).

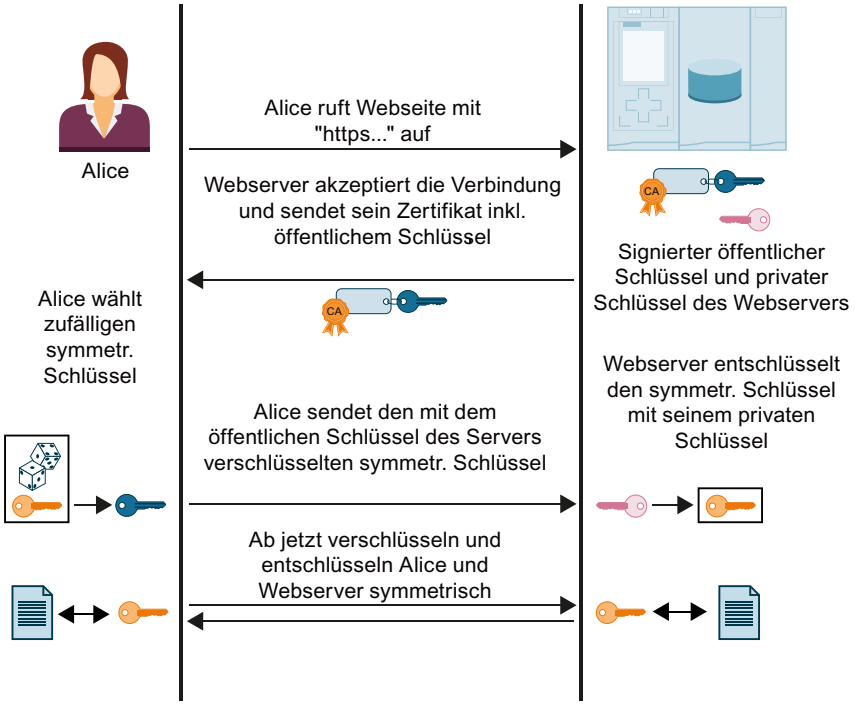


Bild 5-14 Handshake bei https

Nicht dargestellt im Bild sind die Aktionen, die auf Seiten von Alice (Browser) ablaufen, um das vom Webserver gesendete Zertifikat zu prüfen. Vom positiven Ausgang der Prüfung hängt es ab, ob Alice dem übermittelten Webserver-Zertifikat und damit der Identität des Webservers trauen kann und sich auf den Datenaustausch einlassen darf.
Die Schritte zum Prüfen der Authentizität des Webservers im Einzelnen:

1. Alice muss die öffentlichen Schlüssel aller beteiligten Zertifizierungsstellen kennen, d. h., Alice benötigt die gesamte Zertifikate-Kette zum Prüfen des Webserver-Zertifikats (d. h. des End-Entity-Zertifikats des Webservers):
   Üblicherweise hat Alice in ihrem Zertifikatspeicher das benötigte Stammzertifikat. Mit der Installation eines Webbrowsers werden z. B. eine Reihe von vertrauenswürdigen Stammzertifikaten mitinstalliert. Wenn sie das Stammzertifikat nicht hat, muss sie es von der Zertifizierungsstelle herunterladen und im Zertifikatespeicher des Browsers installieren. Die Zertifizierungsstelle kann auch das Gerät sein, auf dem sich der Webserver befindet.
   Um die Zwischenzertifikate zu erhalten, gibt es folgende Möglichkeiten:
   - Der Server selbst sendet zusammen mit seinem End-Entity-Zertifikat die erforderlichen Zwischenzertifikate an Alice, und zwar als signierte Nachricht, damit Alice die Integrität der Zertifikate-Kette prüfen kann.
   - In den Zertifikaten befinden sich oft die URLs vom jeweiligen Zertifikate-Aussteller. Über diese URLs kann Alice die erforderlichen Zwischenzertifikate laden.

Wenn Sie Zertifikate in STEP 7 hantieren, wird immer davon ausgegangen, dass Sie die erforderlichen Zwischenzertifikate und das Stammzertifikat in das Projekt importiert und der Baugruppe zugeordnet haben.

2. Mit den öffentlichen Schlüsseln der Zertifikate validiert Alice die Signaturen der Zertifikate-Kette.
3. Der symmetrische Schlüssel muss erzeugt und an den Webserver übermittelt werden.
4. Wenn der Webserver mit seinem Domainnamen adressiert wird, verifiziert Alice gemäß den in RFC 2818 festgelegten PKI-Regeln die Identität des Webservers: Das kann sie, weil die URL des Webservers, in diesem Fall der "Fully Qualified Domain Name" (FQDN), im End-Entity-Zertifikat des Webservers hinterlegt ist. Wenn der Zertifikatseintrag im Feld "Subject Alternative Name" mit dem Eintrag in der Adresszeile des Browsers übereinstimmt, ist alles in Ordnung.

Weiter geht es mit dem Datenaustausch mit dem symmetrischen Schlüssel, wie im Bild oben gezeigt.

5.6.3 Voraussetzungen für Secure Communication

5.6.3.1 Schutz vertraulicher Konfigurationsdaten

Wie in den Grundlagen zur Secure Communication beschrieben ist, sind für die ordnungsgemäße Funktion zertifikatsbasierter Protokolle private Schlüssel erforderlich, die bestmöglich zu schützen sind.

Diese Schlüssel und andere schützenswerte Daten können Sie ab STEP 7 V17 mit einem Passwort schützen: Dem Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten.

Ein Verzicht auf das Passwort ist möglich, wenn Sie Maßnahmen umgesetzt haben, die einen unberechtigten Zugriff auf das TIA Portal Projekt und die Projektierung der CPU unterbinden.

Unabhängig davon, ob Sie ein Passwort vergeben oder nicht: Das TIA Portal erzeugt eine Schlüsselinformation, die für den Schutz der vertraulichen PLC-Konfigurationsdaten sorgt. Auf den Ablauf der Secure Communication hat das Passwort keinen Einfluss. Die Komplexität des Passworts zum Schutz vertraulicher PLC-Konfigurationsdaten bestimmt aber, wie gut z. B. die privaten Schlüssel geschützt sind.

Das Vorhandensein einer Schlüsselinformation ist Voraussetzung für Secure Communication wie z. B. die TLS-basierte Secure PG/HMI-Kommunikation: Nur wenn diese Schlüsselinformation vorliegt, kann die CPU Zertifikate hantieren, die für Secure Communication erforderlich sind.

Das folgende Bild zeigt die beschriebenen Zusammenhänge.

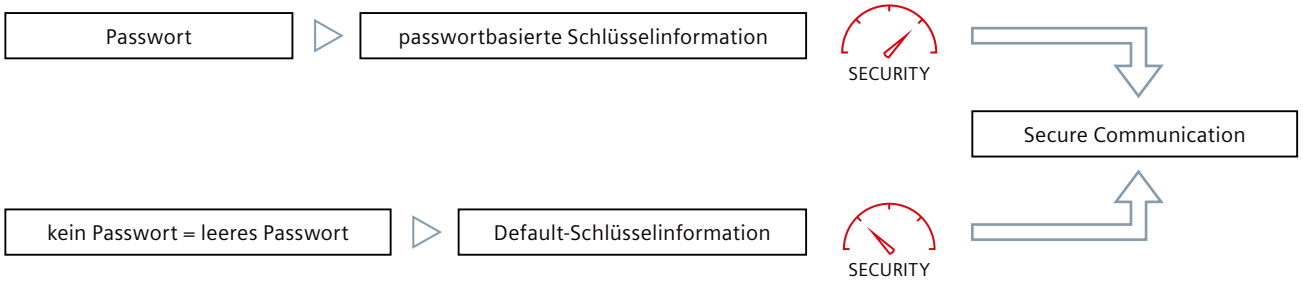


Bild 5-15 Zusammenhänge zum Schutz vertraulicher Konfigurationsdaten

Assistent für Security-Einstellungen

Wenn Sie eine CPU, die Secure PG/HMI-Kommunikation unterstützt, im TIA Portal aus dem Hardware-Katalog zum Projekt hinzufügen, dann startet ein Assistent für die Security-Einstellungen der CPU.
Der Assistent führt Sie schrittweise durch folgende CPU-Einstellungen:

- Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten
- Modus der PG/PC- und HMI-Kommunikation
- Zugriffsstufe

Jede dieser Einstellungen ist im Assistenten ausführlich erläutert. Am Schluss werden alle Einstellungen noch einmal übersichtlich zusammengefasst.
Der Assistent startet auch, wenn Sie z. B. in der Netzsicht vom TIA Portal einen Baugruppentausch durchführen und die neue CPU unterstützt im Gegensatz zur ausgetauschten CPU die Secure PG/HMI-Kommunikation.
Alle Einstellungen im Assistenten werden im Inspektorfenster (CPU-Eigenschaften) übernommen.
Sie können den Assistenten jederzeit über eine Start-Schaltfläche im Bereich "Schutz & Security" der CPU-Eigenschaften starten.

Voraussetzung

- TIA Portal ab Version V17
- CPU unterstützt Secure PG/HMI-Kommunikation (für S7-1500 CPUs z. B. ab Firmware-Version 2.9)
- Die CPU ist noch nicht geladen bzw. die CPU ist zurückgesetzt auf Werkseinstellungen mit der Option "Passwort für den Schutz vertraulicher PLC-Konfigurationsdaten löschen"

Vorgehen

1. Öffnen Sie die CPU-Eigenschaften in der Netzsicht oder in der Gerätesicht.
2. Navigieren Sie zum Bereich "Schutz & Security > Schutz der PLC-Konfigurationsdaten".
   Ergebnis: Die Option "Vertrauliche PLC-Konfigurationsdaten schützen" ist zunächst aktiviert und das leere Feld für die Passworteingabe ist rot hinterlegt.
3. Konfigurieren Sie ein Passwort (Empfehlung) über die Schaltfläche "Festlegen" oder deaktivieren Sie die Option "Vertrauliche PLC-Konfigurationsdaten schützen".
4. Vervollständigen Sie die Konfiguration und erstellen Sie das Anwenderprogramm.
5. Laden Sie die CPU.
   Beim Laden der Hardware-Konfiguration werden Sie einmalig aufgefordert, das Passwort erneut einzugeben.
   Hintergrund: Das konfigurierte Passwort wird zwar im TIA Portal genutzt, um die Schlüsselinformation zum Schutz vertraulicher Konfigurationsdaten zu erzeugen und diese Daten damit zu schützen. Aber weder das Passwort noch die Schlüsselinformation wird aus Security-Gründen im Projekt gespeichert. Damit die Schlüsselinformation in die CPU gelangt, wird sie beim Laden der Hardware-Konfiguration neu erzeugt, sodass an dieser Stelle auch eine einmalige Eingabe des Passworts erforderlich ist.

Zertifikatsbasierte Kommunikation auch zwischen PG/HMI und CPU

Weil ab TIA Portal Version V17 und CPU Firmware-Version V2.9 (S7-1500) bzw. V4.5 (S7-1200) auch die PG/HMI-Kommunikation zertifikatsbasiert abläuft, werden Sie im Verlauf der Inbetriebnahme zum Akzeptieren des Server-Zertifikats aufgefordert.

Tipps und Regeln für das Passwort-Management

- Verwalten Sie Ihre Passwörter in einem Passwortmanager.
- Nutzen Sie die Einstellungen zum Überprüfen der Passwort-Richtlinien im TIA Portal, um neu eingegebene Passwörter auf Einhaltung der Richtlinien checken zu lassen und damit z. B. triviale Passwörter zu verhindern:
  - Navigieren Sie in der Projektnavigation zum Bereich "<Projektname> > Security-Einstellungen > Einstellungen" und wählen den Bereich "Passwort-Richtlinien".
  - Legen Sie z. B. fest, wie viele Zeichen das Passwort mindestens haben muss oder wie viele Sonderzeichen mindestens enthalten sein müssen.
- Sie müssen nicht für jede CPU in einer Anlage oder Maschine verschiedene Passwörter vergeben. Bei entsprechenden Voraussetzungen können Sie auch für eine Gruppe von CPUs dasselbe Passwort festlegen. Diese Strategie hat auch Vorteile für den Ersatzteilfall: Wenn Sie der Ersatz-CPU ebenfalls das Gruppen-Passwort zuweisen, dann reduziert sich der Aufwand für den Austausch der CPU.

  Beachten Sie hierbei das Risiko, dass bei einer Kompromittierung des Passworts einer dieser CPUs alle CPUs mit demselben Passwort angreifbar sind.
- Das Festlegen von Passwörtern hat auch Auswirkungen auf den Ersatzteilfall, da das Passwort für vertrauliche PLC Konfigurationsdaten zusätzlich zur Projektierung auf die neue (Ersatz-) CPU übertragen werden muss (siehe Regeln für den Ersatzteilfall (Seite 85)).
- Bei **S7-1500R/H CPUs** wird beim Laden das Passwort für vertrauliche PLC-Konfigurationsdaten nur auf eine der beiden CPUs geladen. Damit der Sync-Up-Prozess funktioniert und die Partner-CPU ebenfalls ordnungsgemäß arbeitet, muss das Passwort vor dem Sync-Up auf die Partner-CPU übertragen werden, und zwar mit dem Online- und Diagnose-Editor:
  - Wählen Sie in der Online- und Diagnosesicht den Bereich "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten festlegen".
  - Geben Sie das erforderliche Passwort ein und klicken auf die Schaltfläche "Festlegen".

    Sofern das korrekte Passwort eingegeben wurde, kann die Partner-CPU die geschützten PLC-Konfigurationsdaten nutzen und der Sync-Up-Prozess kann beginnen.

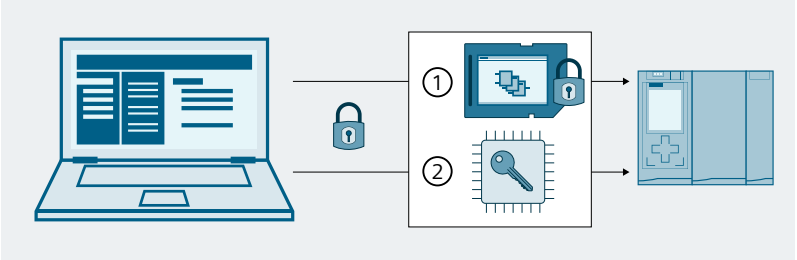
5.6.3.2 Wissenswertes zum Schutz vertraulicher PLC-Konfigurationsdaten

Das Konzept für die über Security-Standards abgesicherte Secure Communication umfasst folgende Komponenten:

- Eine passwortbasierte Schlüsselinformation, die für den Schutz vertraulicher Konfigurationsdaten (z. B. private Schlüssel für Zertifikate, Passwörter) genutzt wird.
- Ein standardisiertes Protokoll (TLS), das die Kommunikation zwischen den Teilnehmern (z. B. Programmiergerät und CPU) absichert.

Prinzip "Schutz vertraulicher Konfigurationsdaten"

Das folgende Bild zeigt vereinfacht, wie vertrauliche Konfigurationsdaten z. B. einer Standard-S7-1500 CPU geschützt werden: Die beiden Bestandteile Projekt und Schlüsselinformation werden beim erstmaligen Laden in verschiedenen Speicherbereichen platziert. Das Projekt im Ladespeicher (Memory Card), die Schlüsselinformation in einem CPU-intenen Speicherbereich.
Bei anderen Zielsystemen (z. B. S7-1200 CPU, Software Controller) mit abweichenden Speicherkonzepten ist die Umsetzung an die entsprechenden Speicherkonzepte angepasst, das Prinzip ist aber dasselbe.



① Projekt mit passwortgeschützten vertraulichen Konfigurationsdaten (hier: im Ladespeicher = Memory Card)

② Schlüsselinformation (aus Passwort erzeugt), um die geschützten vertraulichen Konfigurationsdaten nutzen zu können (hier: im internen CPU-Speicherbereich)

Bild 5-16 Prinzip zum Schutz vertraulicher Konfigurationsdaten

Zwei Speicherbereiche für mehr Sicherheit

Die geladenen Bestandteile sind aufeinander bezogen wie zwei zusammenpassende Puzzleteile: Das Projekt ist an die geladene Schlüsselinformation gebunden, die geladene Schlüsselinformation ist an das Passwort gebunden, das während der Projektierung vergeben wurde.

Projekt und Schlüsselinformation müssen zusammenpassen, sonst läuft die CPU nicht an.

Das Prinzip der zwei getrennten Speicherbereiche gilt auch für die S7-1200 CPUs und für S7-1500-CPU-Varianten ohne Memory Card wie z. B. beim SW-Controller oder bei PLCSim/ PLCSim Advanced. Bei den Varianten ohne Memory Card werden zwei getrennte Partitionen genutzt, um die beiden Informationen unabhängig voneinander verwalten zu können.

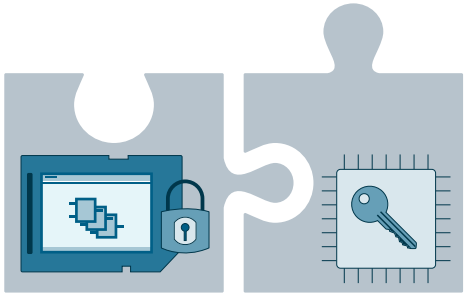


Bild 5-17 Prinzip zwei getrennter Speicherbereiche

5.6.3.3 Passwort ändern

Es macht einen Unterschied, ob die CPU bereits geladen wurde oder nicht. Wenn die CPU bereits geladen wurde, verfügt sie über die Schlüsselinformation, mit der die passwort-geschützten PLC-Konfigurationsdaten nutzbar sind.

Passwort ändern - Konfiguration noch nicht geladen

Solange Sie noch keine Konfiguration in die CPU geladen haben, können Sie ein eingegebenes Passwort ändern oder die Aktivierung des Passwort-Schutzes zurücknehmen.

Voraussetzung

- Die CPU ist noch nicht geladen

Vorgehen

1. Öffnen Sie die CPU-Eigenschaften in der Netzsicht oder in der Gerätesicht.
2. Navigieren Sie zum Bereich "Schutz & Security > Schutz der PLC-Konfigurationsdaten".
3. Klicken Sie auf die Schaltfläche "Ändern" oder deaktivieren Sie die Option "Vertrauliche PLC-Konfigurationsdaten schützen".
4. Geben Sie im Dialog das bisher gültige Passwort ein. Bei einer Passwortänderung geben Sie außerdem das neue Passwort ein und bestätigen Sie das neue Passwort.

Solange Sie noch keine Konfiguration in die CPU geladen haben, befindet sich die CPU in einer Bereitstellungsphase (siehe CPU-Verhalten vom Laden bis zur Betriebsbereitschaft (Seite 108)) und Sie können eine beliebige gültige Konfiguration mit Ihrem konfigurierten Passwort laden.

Passwort ändern - Konfiguration ist bereits geladen

Wenn die CPU bereits mit einer Konfiguration geladen wurde und die Konfiguration ist mit einem Passwort für vertrauliche PLC-Konfigurationsdaten geschützt ist, müssen Sie zunächst die CPU entweder auf Werkseinstellungen zurücksetzen und dabei das Passwort für vertrauliche PLC-Konfigurationsdaten in der CPU löschen oder direkt online löschen und anschließend neu festlegen.

Voraussetzungen

- Sie haben Schreibzugriff auf die CPU
- Die CPU ist im Betriebszustand STOP

Vorgehen

1. Markieren Sie die CPU in der Netzsicht.
2. Wählen Sie im Kontextmenü den Befehl "Online & Diagnose".
3. Wenn Sie das Projekt auf der Memory Card ebenfalls ändern, d. h. anschließend die Konfiguration erneut laden wollen:
   - In der geöffneten Online- und Diagnosesicht wählen Sie den Bereich "Rücksetzen auf Werkseinstellungen".
   - Aktivieren Sie die Option "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten löschen". Um ein wiederholtes Anlaufen der CPU zu vermeiden, aktivieren Sie zusätzlich die Option "Memory Card formatieren".
   - Laden Sie anschließend das Projekt mit der geänderten Konfiguration und dem gewünschten Passwort.
4. Wenn Sie das Projekt auf der Memory Card nicht ändern müssen, d. h. es ist nur das falsche Passwort gesetzt:
   - Wählen Sie in der Online- und Diagnosesicht den Bereich "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten festlegen".
   - Klicken Sie auf die Schaltfäche "Löschen". Wenn die Schaltfläche "Löschen" nicht bedienbar ist, wurde in der CPU noch kein Passwort gesetzt.
   - Geben Sie das erforderliche Passwort ein und klicken auf die Schaltfläche "Festlegen".

   Sofern das korrekte Passwort eingegeben wurde, kann die CPU die geschützten PLC-Konfigurationsdaten nutzen.

Kein Schreibzugriff auf die CPU

Falls Sie keinen Schreibzugriff auf den Ladespeicher haben (Zugriffsstufe Lesezugriff), dann entfernen Sie die Memory Card aus der CPU bzw. löschen Sie die Memory Card extern z. B. in Ihrem Computer bevor Sie Rücksetzen auf Werkseinstellungen durchführen mit der Option "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten löschen".

HINWEIS

Rücksetzen auf Werkseinstellungen mit Betriebsartenschalter der CPU

Durch das Zurücksetzen der CPU auf Werkseinstellungen über den Betriebsartenschalter wird zwar auch die IP-Adresse der CPU gelöscht, nicht aber das Passwort für den Schutz vertraulicher PLC-Konfigurationsdaten.

Weitere Informationen

Informationen zum Vorgehen im Ersatzteilfall finden Sie in Kap. Regeln für den Ersatzteilfall (Seite 85).

5.6.3.4 Passwort rücksetzen

Der Schutz der vertraulichen PLC-Konfigurationsdaten kann zurückgesetzt werden. Dies kann beispielsweise notwendig sein, wenn Sie das Passwort ändern möchten, aber das aktuelle Passwort nicht mehr wissen.

Passwort vergessen - Konfiguration noch nicht geladen

Da Sie beim erstmaligen Laden der CPU über das TIA Portal das Passwort eingeben müssen, kann die CPU-Projektierung für diese CPU nicht weiter verwendet werden. Für eine Passwort-Änderung in den CPU-Eigenschaften ist das bisher gültige Passwort ebenfalls einzugeben. Wenn Sie das Passwort vergessen haben, gehen Sie folgendermaßen vor:

Voraussetzung

- Die CPU ist noch nicht geladen

Vorgehen

1. Öffnen Sie die CPU-Eigenschaften in der Netzsicht oder in der Gerätesicht.
2. Navigieren Sie zum Bereich "Schutz & Security > Schutz der PLC-Konfigurationsdaten".
3. Klicken Sie auf die Schaltfläche "Rücksetzen".
   Beachten Sie, dass die Zertifikate der CPU (z. B. Zertifikate für Webserver, für OPC UA-Server, für PG/PC- und HMI-Kommunikation) nach dem Rücksetzen nicht mehr verwendet werden können und ggf. neu erstellt und neu zugewiesen werden müssen.
   - Wenn Sie die globalen Security-Einstellungen für den Zertifikatsmanager nutzen, dann müssen Sie die Zertifikate aus dem Zertifikatsmanager erneut zuweisen.
   - Wenn Sie die globalen Security-Einstellungen für den Zertifikatsmanager nicht nutzen, dann müssen Sie die Zertifikate neu erstellen und zuweisen.
4. Bestätigen Sie das Zurücksetzen des Passwortes.

Die Option zum Schutz vertraulicher PLC-Konfigurationsdaten ist weiterhin aktiviert.

Passwort löschen - Konfiguration ist bereits geladen

Wenn die CPU bereits mit einer Konfiguration geladen wurde und die Konfiguration ist mit einem Passwort für vertrauliche PLC-Konfigurationsdaten geschützt, können Sie für das Laden eines neuen Projekts das Passwort für vertrauliche PLC-Konfigurationsdaten in der CPU online löschen und anschließend neu festlegen.

Voraussetzungen

- Sie haben Schreibzugriff auf die CPU
- Die CPU ist im Betriebszustand STOP

Vorgehen

1. Markieren Sie die CPU in der Netzsicht.
2. Wählen Sie im Kontextmenü den Befehl "Online & Diagnose".

3. Klicken Sie im Bereich "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten festlegen" auf die Schaltfäche "Löschen".
   Wenn die Schaltfläche "Löschen" nicht bedienbar ist, wurde in der CPU noch kein Passwort gesetzt.

ACHTUNG
Löschen des Passworts für vertrauliche Konfigurationsdaten Wenn das Passwort gelöscht wird und ein geladenes Projekt erfordert aber ein entsprechendes Passwort, wird dieses Projekt möglicherweise ohne Passwort nicht mehr funktionieren.

4. Geben Sie nun bei Bedarf über die Schaltfläche "Festlegen" ein neues Passwort ein.
5. Führen Sie einen Neustart der CPU durch.

Weitere Informationen

Informationen zum Ändern des Passworts finden Sie in Kap. Passwort ändern (Seite 78).

5.6.3.5 Passwort über SIMATIC Memory Card zuweisen

Wenn Sie einer CPU das Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten übermitteln wollen, ohne das TIA Portal zu nutzen, können Sie für diese Funktion auch eine SIMATIC Memory Card nutzen.
Die Nutzung einer SIMATIC Memory Card bietet sich für die folgenden Zwecke an:

- Vorbereiten einer neuen CPU
  Wenn eine CPU neu aufgesetzt wird, sollte diese mit einem Passwort zum Schutz der vertraulichen PLC-Konfigurationsdaten konfiguriert sein. Nachdem diese Konfiguration abgeschlossen ist, ist es möglich, eine weitere SIMATIC Memory Card mit dem gewünschten Projekt zu nutzen.
  (S7-1200 CPU: Es kann auch eine "Transfer"-Karte mit Transfer-Job genutzt werden, um das Programm auf der CPU zu installieren.)
- CPU hat ein Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten, aber das Passwort passt nicht zum Projekt
  Sie können im Fall ungleicher Passwörter das korrekte Passwort mit der Memory Card in der CPU setzten.
  (S7-1200 CPU: bestückt entweder mit SIMATIC "Transfer"-Karte oder mit SIMATIC "Programm"-Karte.)
- Rücksetzen des Passworts zum Schutz vertraulicher PLC-Konfigurationsdaten in der CPU
  Als Vorbereitung einer Entsorgung der CPU oder als Vorbereitung für ein neues Projekts für die CPU.

Voraussetzung

- TIA Portal ab Version V17

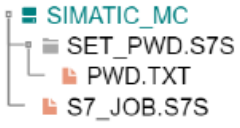
Prinzipielles Vorgehen

1. Erstellen einer SIMATIC Memory Card mit "SET PASSWORD"-Job
   Mit dieser Aktion wird eine einem speziellen Muster folgende Ordner- und Dateistruktur erstellt und ein Passwort zum Schutz der vertraulichen PLC-Konfigurationsdaten als Nur-Text in eine spezielle Datei auf der SIMATIC Memory Card geschrieben. Siehe Beschreibung unten.
2. Einsetzen einer vorbereiteten SIMATIC Memory Card in die CPU und CPU einschalten.
   Die PLC liest das Passwort, verarbeitet es und speichert das Ergebnis im internen Speicher. Ein eventuell bestehender Eintrag wird überschrieben.
3. SIMATIC Memory Card ziehen und CPU neu starten.
   Ergebnisse (S7-1500): Während die CPU die SIMATIC Memory Card liest, zeigt die LED das gleiche Verhalten wie bei einem Firmware-Update. Während die CPU das Passworts setzt, blinkt die RUN/STOP-LED. Nachdem der Vorgang erfolgreich abgeschlossen wurde, leuchtet die RUN/STOP-LED gelb und die MAINT-LED blinkt gelb.

Das Ergebnis des Vorgangs wird im Diagnosepuffer als Erfolgs- oder Fehlermeldung angezeigt. Falls das Passwort nicht gesetzt werden konnte, blinkt die Error-LED zusammen mit den anderen LEDs.

SIMATIC Memory Card mit "SET PASSWORD"-Job erstellen

1. Erstellen Sie einen Ordner im Root-Verzeichnis mit der Bezeichnung "SET_PWD.S7S".
2. Erstellen Sie eine Textdatei mit der Bezeichnung "PWD.TXT" mit dem Passwort als Nur-Text im soeben angelgten Ordner auf der Memory Card.
3. Erstellen Sie eine Textdatei mit der Bezeichnung "S7_JOB.S7S" im Root-Verzeichnis der Memory Card mit dem Inhalt "SET_PWD".
   Diese Datei ist das "Job file", um ein Passwort zum Schutz der vertraulichen PLC-Konfigurationsdaten der PLC zuzuweisen.
4. Die Dateistruktur auf der SIMATIC Memory Card sieht dann aus wie folgt:



HINWEIS

Sichere Aufbewahrung der SIMATIC Memory Card

Bewahren Sie die SIMATIC Memory Card an einem sicheren Ort auf, zu dem nur autorisierte Personen Zugang haben.

Regeln und Empfehlungen

- Das Setzen des Passworts muss in einer sicheren Umgebung erfolgen.
- Der Inhalt der Textdatei "PWD.TXT" definiert das Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten. Es muss dem Passwort entsprechen, dass Sie auch in der CPU-Konfiguration vergeben haben.
- Um ein existierendes Passwort einer PLC zurückzusetzen, muss die Textdatei "PWD.TXT" leer sein, d. h. die Dateigröße ist 0 Byte.
- Verwenden Sie einen beliebigen Texteditor, um die Textdatei zu erstellen. Das empfohlene Textformat ist "UTF-8".
- Bei den Ordner- und Dateinamen spielt die Groß- und Kleinschreibung keine Rolle. Das Passwort selbst unterscheidet aber Groß- und Kleinschreibung.
- Fügen Sie kein CR/LF Zeichen am Ende ein (PWD.TXT oder S7_JOB.S7S).

5.6.3.6 Besonderheiten beim Sichern und Wiederherstellen einer CPU

Sie können im TIA Portal eine funktionsfähige Projektierung einer CPU sichern, um zu einem späteren Zeitpunkt darauf zurückgreifen zu können, d. h. Sie können dann die ursprünglich gesicherte Projektierung wiederherstellen. Das ermöglicht Ihnen, eine geänderte Projektierung zu laden, um z. B. Funktionserweiterungen zu testen, Programme zur Fehlersuche in der Anlage zu ändern oder Sie können Komponenten testweise ersetzen. Anschließend können Sie die ursprüngliche gesicherte Projektierung der CPU wiederherstellen.

Projektierung sichern

Bei der Sicherung einer CPU (Menü "Online", Befehl "Sicherung von Online-Gerät laden" im TIA Portal) wird das Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten nicht mitgesichert.

Wiederherstellen der Sicherung

Beim Wiederherstellen der Sicherung einer CPU (Menü "Online", Befehl "Laden in Gerät" bei markierter Sicherung im TIA Portal) kann die CPU nur dann mit einem PG/PC oder HMI kommunizieren, wenn folgende Bedingung erfüllt ist:

- Nach dem Wiederherstellen einer Projektierung, die mit einem Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten geschützt wurde, muss in der CPU genau dieses Passwort vorliegen.

  Andernfalls kann die CPU nicht auf die Konfigurationsdaten zugreifen und läuft daher nicht an.

Abhilfe

Wenn der oben genannte Fehlerfall eintritt, d. h. das Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten passt nicht zur Sicherung, müssen Sie das Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten in der CPU löschen und anschließend das korrekte Passwort setzen. Nach einem Neustart der CPU ist die Sicherung funktionsfähig.

5.6.3.7 Tipps zur Fehlervermeidung und Fehlerbehandlung

Die folgende Beschreibung listet einige Anwendungsfälle auf, die möglicherweise zu CPU-Fehlermeldungen führen.

Diagnosepuffer gibt Auskunft

Die CPU erkennt, wenn das Passwort zum Schutz vertraulicher Konfigurationsdaten und die geladene Konfiguration nicht zusammenpassen. Eine Meldung im Diagnosepuffer weist auf mögliche Ursachen und Abhilfemaßnahmen hin und führt in der Regel zur Lösung des Problems.

Typische "Fallstricke"

Auf folgende Umstände sollten Sie achten, um Fehler zu vermeiden bzw. zu beheben:

- Konfiguration geladen?
  Unabhängig davon, ob Sie Ihre vertraulichen Konfigurationsdaten mit einem Passwort schützen oder nicht: Ohne geladene Konfiguration verlässt die CPU die Bereitstellungsphase nicht.
- Sie versuchen die CPU mit einem konfigurierten Passwort zu laden und die CPU hat bereits vorher ein anderes Passwort erhalten.
  Beispiel: CPU wird gegen eine andere CPU aus dem Lager getauscht, die Austausch-CPU wurde nicht vollständig zurückgesetzt (Rücksetzen auf Werkseinstellungen mit Option "Lösche Passwort für den Schutz vertraulicher PLC-Konfigurationsdaten").
  Abhilfe:
  - Austausch-CPUs immer mit der entsprechenden Option vorbereiten (Passwort gelöscht).
  - Verwenden Sie für die zu ladende Konfiguration dasselbe Passwort, das schon für die bereits geladene Konfiguration verwendet wurde.
  - Möglicherweise wurde auch das falsche Projekt / die falsche CPU-Konfiguration geladen. Kontrollieren Sie, ob die passende CPU-Konfiguration vorliegt.
  - Nutzen Sie die Online-Funktion "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten setzen", um das Passwort zu löschen bzw. um das dasselbe Passwort zu setzen wie in der CPU-Konfiguration. Führen Sie anschließend einen Neustart durch.
- Ein gleiches Fehlerbild entsteht, wenn Ihre CPU-Konfiguration kein Passwort verwendet und die bereits geladene Konfiguration ein anwenderdefiniertes Passwort erfordert.
  Abhilfe:
  - Nutzen Sie die Online-Funktion "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten setzen", um das Passwort zu löschen bzw. um das dasselbe Passwort zu setzen wie in der CPU-Konfiguration. Führen Sie anschließend einen Neustart durch.

5.6.3.8 Regeln für den Ersatzteilfall

Die Vergabe von Passwörtern zum Schutz vertraulicher PLC-Konfigurationsdaten hat auch Auswirkungen auf den Ersatzteilfall.

Regeln für den Ersatzteilfall

Berücksichtigen Sie folgende Regeln für den Ersatzteilfall:

Projektierung der Ersatz-CPU über TIA Portal

- Eine CPU als Ersatz für eine vorhandene CPU sollte keine Projektierung und kein konfiguriertes Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten haben. Vorteil: Sie können ohne weitere Vorbereitung das Projekt in die Ersatz-CPU laden - unabhängig davon, ob ein Passwort konfiguriert ist oder nicht.
- Falls die Ersatz-CPU bereits projektiert ist, müssen Sie die CPU auf Werkseinstellungen zurücksetzen mit folgenden gesetzten Optionen:
  - "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten löschen"
  - "Memory Card formatieren"

Ersatz-CPU wird über Memory Card mit Projektierungsdaten versorgt

- Falls Sie in Ihrem Projekt einer CPU **kein** Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten zugewiesen haben, können Sie ohne weitere Hantierung die Memory Card der auszutauschenden CPU in eine fabrikneue, unbenutzte CPU stecken. Falls die Ersatz-CPU bereits mit einem Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten projektiert wurde, müssen Sie zunächst diese CPU auf Werkseinstellungen zurücksetzen mit der Option "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten löschen".
- Falls Sie einer Gruppe von CPUs dasselbe Passwort zugewiesen haben, können Sie über das TIA Portal oder über eine entsprechend vorbereitete Memory Card der Ersatz-CPU ebenfalls das Gruppen-Passwort zuweisen (siehe Schutz vertraulicher Konfigurationsdaten (Seite 74)).
  In diesem Fall können Sie z. B. eine Memory Card mit dem aktuellen Projekt ohne weitere Passwort-Hantierung in die CPU stecken und in Betrieb nehmen.
- Falls Sie jeder CPU in Ihrem Projekt unterschiedliche Passwörter vergeben, müssen Sie beim Einsatz der Ersatz-CPU zunächst mit dem Online- und Diagnose-Editor das für die jeweilige CPU gültige Passwort setzen (Bereich "Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten festlegen", siehe Passwort ändern (Seite 78)).

Weitere Informationen

Wie Sie mithilfe der SIMATIC Memory Card das Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten zuweisen lesen Sie in Kap. Passwort über SIMATIC Memory Card zuweisen (Seite 81).

5.6.4 Secure Open User Communication

5.6.4.1 Secure OUC von einer S7-1500 CPU als TLS-Client zu einem Fremd-PLC (TLS-Server)

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über TCP von einer S7-1500 CPU als TLS-Client zu einem TLS-Server einrichten.

Gesicherte TCP-Verbindung von einer S7-1500 CPU als TLS-Client zu einem TLS-Server einrichten

S7-1500 CPUs ab Firmwarestand V2.0 unterstützen Secure Communication mit Adressierung über ein Domain Name System (DNS).

Für die gesicherte TCP-Kommunikation über den Domainnamen müssen Sie selbst einen Datenbaustein mit dem Systemdatentyp TCON_QDN_SEC erstellen, parametrieren und direkt an einer der Anweisungen TSEND_C, TRCV_C oder TCON aufrufen.

Voraussetzungen:

- Aktuelles Datum und Uhrzeit sind in der CPU eingestellt.
- In Ihrem Netz befindet sich mindestens ein DNS-Server.
- Sie haben für die S7-1500 CPU mindestens einen DNS-Server konfiguriert.
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.

Um eine gesicherte TCP-Verbindung zu einem TLS-Server einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_QDN_SEC.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "DNS ConnectionSEC" vom Datentyp TCON_QDN_SEC definiert ist.

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	DNS Connection SEC	TCON_QDN_SEC		
3	ConnPara	TCON_QDN		parameter of the TCP connection
4	InterfaceId	HW_ANY	0	not relevant
5	ID	CONN_OUC	5	connection reference / identifier
6	ConnectionType	Byte	11	type of connection: 16#0B=11=TCP/IP, 16#13
7	ActiveEstablished	Bool	true	active/passive connection establishment
8	RemoteQDN	String[254]	'plc_1.factory127.'	fully or partially qualified domain name of rem
9	RemotePort	UInt	4000	remote UDP / TCP port number
10	LocalPort	UInt	0	local UDP / TCP port number
11	ActivateSecureConn	Bool	true	activate the security functionality of that conn
12	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a
13	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 a
14	TLSServerCertRef	UDInt	7	for Server side: Reference to own X.509 V3 se
15	TLSClientCertRef	UDInt	0	for Client side: add id of own X.509 V3 client c

Bild 5-18 Datentyp TCON_QDN_SEC

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteQDN" den vollqualifizierten Domainnamen (FQDN) des TLS-Servers ein.

4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "ExtTLSCapabilities": Wenn Sie den Wert 1 eintragen, dann validiert der Client den subjectAlternateName im X.509-V3-Zertifikat des Servers, um die Identität des Servers zu überprüfen. Diese Validierung erfolgt im Kontext der Anweisung.
   - "TLSServerCertRef": ID des X.509-V3-Zertifikats (gewöhnlich ein CA-Zertifikat), das vom TLS-Client benutzt wird, um die Authentifizierung des TLS-Servers zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Client zur Validierung der Server-Authentifizierung alle (CA-) Zertifikate, die aktuell im Certificate Store des Clients geladen sind.

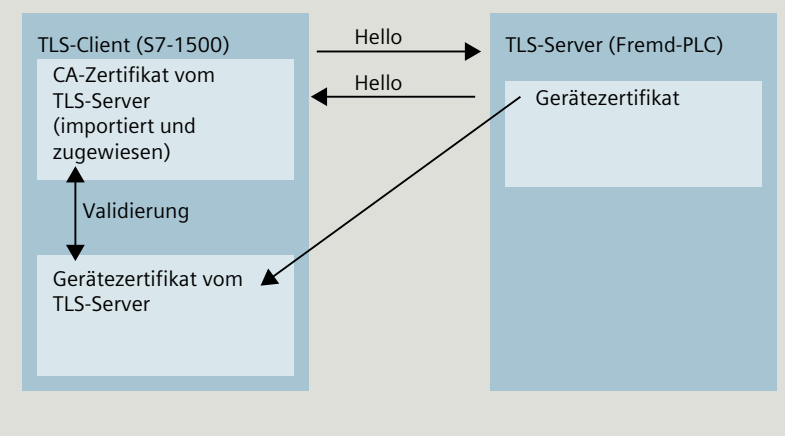


Bild 5-19 Zertifikate-Handling aus Perspektive der S7-1500 als TLS-Client

   - "TLSClientCertRef": ID des eigenen X.509-V3-Zertifikats.
5. Legen Sie im Programmeditor eine der Anweisungen TSEND_C, TRCV_C oder TCON an.
6. Verschalten Sie den Parameter CONNECT einer der Anweisungen TSEND_C, TRCV_C oder TCON mit der Variablen vom Datentyp TCON_QDN_SEC.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "DNS connectionSEC" (Datentyp TCON_QDN_SEC) verschaltet.

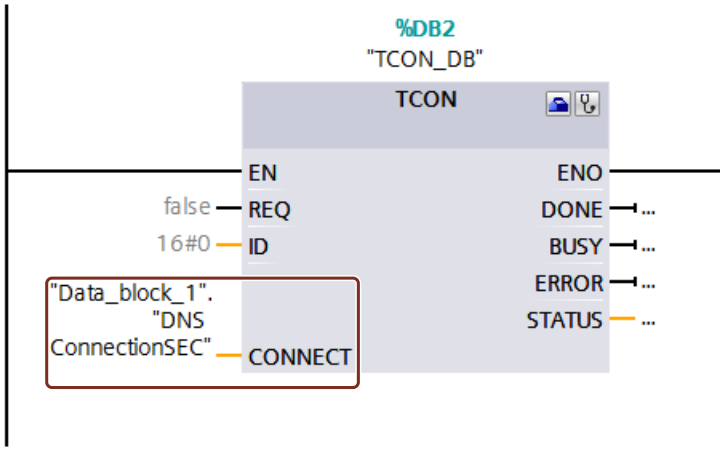


Bild 5-20 Anweisung TCON

Weitere Informationen

Weitere Informationen zum Systemdatentyp TCON_QDN_SEC finden Sie in der Onlinehilfe zu STEP 7.
Weitere Informationen zur sicheren Kommunikation finden Sie im Kapitel Secure Communication (Seite 50).

5.6.4.2 Secure OUC von einer S7-1500 CPU als TLS-Server zu einem Fremd-PLC (TLS-Client)

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über TCP von einer S7-1500 CPU als TLS-Server zu einem TLS-Client einrichten.

Gesicherte TCP-Verbindung über den Domainnamen des Kommunikationspartners einrichten

S7-1500 CPUs ab Firmwarestand V2.0 unterstützen Secure Communication mit Adressierung über ein Domain Name System (DNS).
Für die gesicherte TCP-Kommunikation über den Domainnamen müssen Sie selbst einen Datenbaustein mit dem Systemdatentyp TCON_QDN_SEC erstellen, parametrieren und direkt an einer der Anweisungen TSEND_C, TRCV_C oder TCON aufrufen.
Voraussetzungen:

- Aktuelles Datum und Uhrzeit sind in der CPU eingestellt.
- In Ihrem Netz befindet sich mindestens ein DNS-Server.
- Sie haben für die S7-1500 CPU mindestens einen DNS-Server konfiguriert.
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.

Um eine gesicherte TCP-Verbindung zu einem TLS-Client einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_QDN_SEC.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "DNS ConnectionSEC" vom Datentyp TCON_QDN_SEC definiert ist.

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	DNS Connection SEC2	TCON_QDN_SEC		
3	ConnPara	TCON_QDN		parameter of the TCP connection
4	InterfaceId	HW_ANY	0	not relevant
5	ID	CONN_OUC	8	connection reference / identifier
6	ConnectionType	Byte	11	type of connection: 16#0B=11=TCP/IP, 16#13
7	ActiveEstablished	Bool	false	active/passive connection establishment
8	RemoteQDN	String[254]	''	fully or partially qualified domain name of rem
9	RemotePort	UInt	0	remote UDP / TCP port number
10	LocalPort	UInt	2010	local UDP / TCP port number
11	ActivateSecureConn	Bool	true	activate the security functionality of that conn
12	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a
13	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 a
14	TLSServerCertRef	UDInt	5	for Server side: Reference to own X.509 V3 se
15	TLSClientCertRef	UDInt	0	for Client side: add id of own X.509 V3 client c

Bild 5-21 TCON_QDN_SEC_Server

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein.
   Tragen Sie z. B. bei "ID" die lokale ID der TCP-Verbindung ein.

4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "TLSServerReqClientCert": Anforderung eines X.509-V3-Zertifikats vom TLS-Client.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.

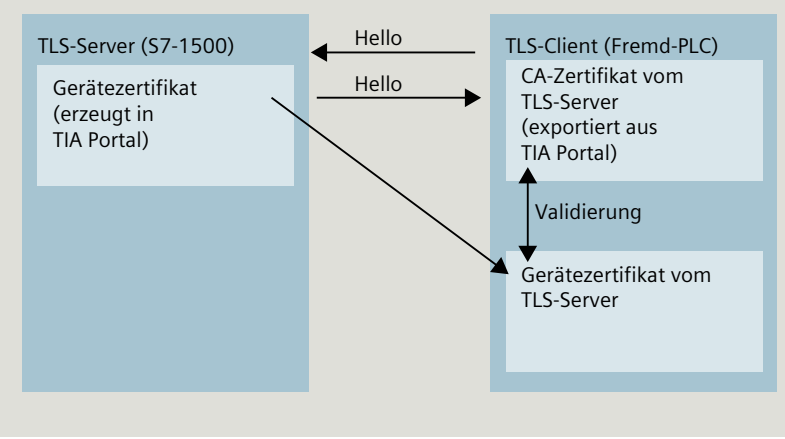


Bild 5-22 Zertifikate-Handling aus Perspektive der S7-1500 als TLS-Server

   - "TLSClientCertRef": ID des X.509-V3-Zertifikats (oder einer Gruppe von X.509-V3-Zertifikaten), das vom TLS-Server benutzt wird, um die Authentifizierung des TLS-Clients zu validieren. Wenn dieser Parameter 0 ist, benutzt der TLS-Server zur Validierung der Client-Authentifizierung alle (CA-) Zertifikate, die aktuell im Certificate Store des Servers geladen sind.

5. Legen Sie im Programmeditor eine der Anweisungen TSEND_C, TRCV_C oder TCON an.
6. Verschalten Sie den Parameter CONNECT einer der Anweisungen TSEND_C, TRCV_C oder TCON mit der Variablen vom Datentyp TCON_QDN_SEC.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "DNS connectionSEC" (Datentyp TCON_QDN_SEC) verschaltet.

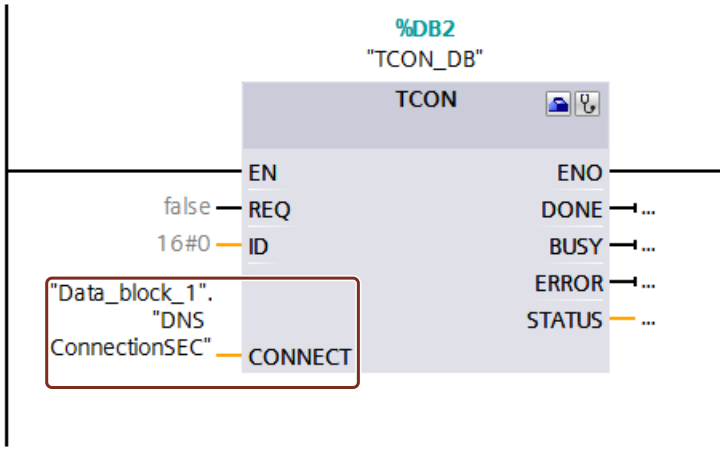


Bild 5-23 Anweisung TCON

Weitere Informationen

Weitere Informationen zu den Systemdatentypen TCON_QDN_SEC finden Sie in der Onlinehilfe zu STEP 7.
Weitere Informationen zur sicheren Kommunikation finden Sie im Kapitel Secure Communication (Seite 50).

5.6.4.3 Secure OUC zwischen zwei S7-1500 CPUs

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über TCP zwischen zwei S7-1500 CPUs einrichten. Dabei agiert eine S7-1500 CPU als TLS-Client (aktiver Verbindungsaufbau) und die andere S7-1500 CPU als TLS-Server (passiver Verbindungsaufbau).

Gesicherte TCP-Verbindung zwischen zwei S7-1500 CPUs einrichten

Für die gesicherte TCP-Kommunikation zwischen zwei S7-1500 CPUs müssen Sie in jeder CPU einen Datenbaustein mit dem Systemdatentyp TCON_IP_V4_SEC erstellen, parametrieren und direkt an einer der Anweisungen TSEND_C, TRCV_C oder TCON aufrufen.
Voraussetzungen:

- Aktuelles Datum und Uhrzeit sind in der CPU eingestellt.
- Beide S7-1500 CPU haben mindestens Firmwarestand V2.0
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.

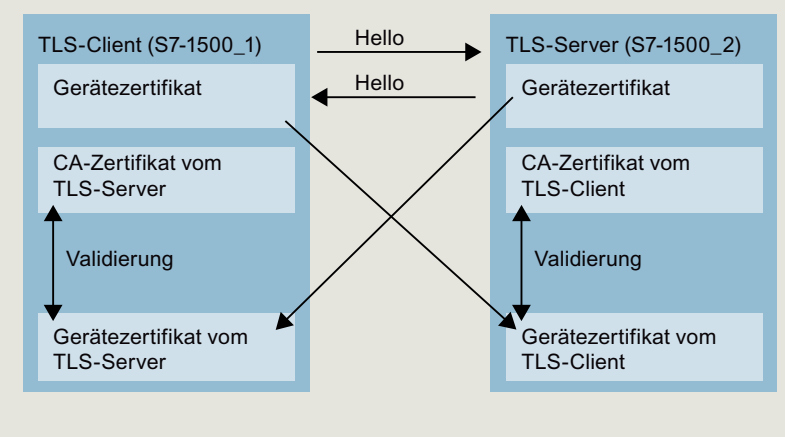


Bild 5-24 Zertifikate-Handling bei Secure OUC zwischen zwei S7-1500 CPUs

Einstellungen am TLS-Client

Um eine gesicherte TCP-Verbindung im TLS-Client einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Client" vom Datentyp TCON_IP_V4_SEC definiert ist.

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	SEC connection 1 TLS-Client	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	72	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	true	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	100	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against th
19	TLSServerCertRef	UDInt	1	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	5	for Client side: add id of own X.509 V3 client certificate; for Ser

Bild 5-25 IP_V4_SEC_Client

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAddress" die IPv4-Adresse des TLS-Servers ein.

HINWEIS

Verbindungsparameter InterfaceId

Beachten Sie, dass Sie im Datentyp TCON_IP_V4_SEC den Wert "**0**" für die InterfaceId eintragen können. In diesem Fall sucht die CPU selbst nach einer passenden lokalen Schnittstelle der CPU.

4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "TLSServerCertRef": Tragen Sie den Wert 2 (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA256) bzw. den Wert 1 ein (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA1)). Wenn Sie ein anderes CA-Zertifikat nutzen, tragen Sie die entsprechende ID aus dem Zertifikatsmanager der Globalen Security-Einstellungen ein.
   - "TLSClientCertRef": ID des eigenen X.509-V3-Zertifikats.

5. Legen Sie im Programmeditor eine der Anweisungen TSEND_C, TRCV_C oder TCON an.
6. Verschalten Sie den Parameter CONNECT einer der Anweisungen TSEND_C, TRCV_C oder TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Einstellungen am TLS-Server

Um eine gesicherte TCP-Verbindung im TLS-Server einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC. Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Server" vom Datentyp TCON_IP_V4_SEC definiert ist.

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	SEC connection 1 TLS-Server	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	120	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	false	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	10	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	true	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against the
19	TLSServerCertRef	UDInt	6	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	1	for Client side: add id of own X.509 V3 client certificate; for Serv

Bild 5-26 IP_V4_SEC_Server

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAddress" die IPv4-Adresse des TLS-Clients ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "TLSServerReqClientCert ": Anforderung eines X.509-V3-Zertifikats vom TLS-Client. Tragen Sie den Wert "true" ein.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.
   - "TLSClientCertRef": Tragen Sie den Wert 2 (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA256) bzw. den Wert 1 ein (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA1)). Wenn Sie ein anderes CA-Zertifikat nutzen, tragen Sie die entsprechende ID aus dem Zertifikatsmanager der Globalen Security-Einstellungen ein.
5. Legen Sie im Programmeditor einer der Anweisungen TSEND_C, TRCV_C oder TCON an.

6. Verschalten Sie den Parameter CONNECT einer der Anweisungen TSEND_C, TRCV_C oder TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TSEND_C mit der Variablen "SEC connection 1 TLS-Client" (Datentyp TCON_IP_V4_SEC) verschaltet.

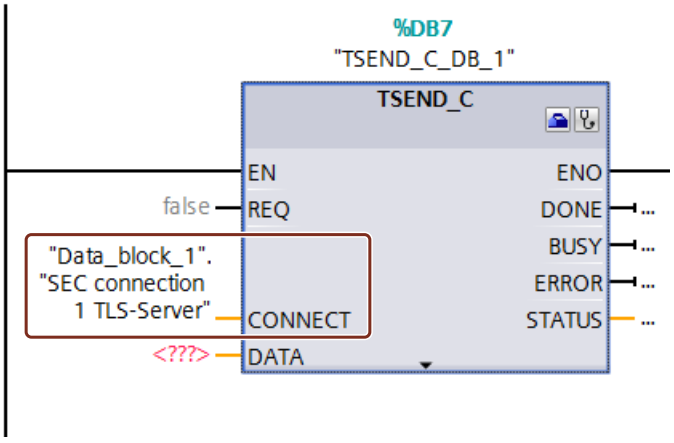


Bild 5-27 TSEND_C

Weitere Informationen

Weitere Informationen zu den Systemdatentypen TCON_IP_V4_SEC finden Sie in der Onlinehilfe zu STEP 7.
Weitere Informationen zur sicheren Kommunikation finden Sie im Kapitel Secure Communication (Seite 50).

5.6.4.4 Secure OUC über CP-Schnittstelle

Im Folgenden sind die Besonderheiten beschrieben, die bei Secure Open User Communication über eine CP-Schnittstelle zu berücksichtigen sind. Mindestens eine Station ist eine S7-1500 Station mit folgenden Baugruppen:

- S7-1500 CPU ab Firmwarestand V2.0 (ausgenommen S7-1500 Software Controller)
- CP 1543-1 ab Firmwarestand V2.0 bzw. CP 1543SP-1 ab Firmwarestand V1.0

Der CP agiert in einer S7-1500 Station als TLS-Client (aktiver Verbindungsaufbau) oder als TLS-Server (passiver Verbindungsaufbau).
Die grundsätzliche Vorgehensweise und das Konzept für die Nutzung von Secure Communication über eine CP-Schnittstelle ist ähnlich wie Secure Communication über die Schnittstellen der S7-1500 CPUs. Im Wesentlichen müssen Sie dem CP in der Rolle als TLS-Server oder TLS-Client die Zertifikate zuordnen und nicht der CPU. Daher gelten andere Regeln und Vorgehensweisen, die im Folgenden beschrieben sind.

Hantierung von Zertifikaten für CPs

Generell gilt: Sie müssen beim Zertifikatsmanager in den Globalen Security-Einstellungen angemeldet sein. Auch das Erstellen von selbst signierten Zertifikaten ist nicht ohne Anmeldung für die Globalen Security-Einstellungen möglich. Sie müssen als Benutzer mit ausreichenden Rechten ausgestattet sein (Administrator oder Benutzer mit der Rolle "Standard" mit dem Recht "Security konfigurieren").

Ausgangspunkt für die Erstellung oder Zuweisung von Zertifikaten beim CP ist der Bereich "Security > Security-Eigenschaften". In diesem Bereich melden Sie sich für die Globalen Security-Einstellungen an.

Vorgehensweise:

1. Markieren Sie in der Netzsicht von STEP 7 den CP und wählen im Inspektorfenster den Bereich "Security > Security-Eigenschaften".
2. Klicken Sie auf die Schaltfläche "Benutzeranmeldung".
3. Melden Sie sich mit Benutzernamen und Passwort an.
4. Aktivieren Sie die Option "Aktiviere Security-Funktionen".
   Die Security-Eigenschaften werden initialisiert.
5. Klicken Sie in die erste Zeile der Tabelle "Gerätezertifikate", um ein neues Gerätezertifikat zu erzeugen oder ein bestehendes Gerätezertifikat auszuwählen.
6. Falls der Kommunikationspartner ebenfalls eine S7-1500 Station ist, müssen Sie dem Kommunikationspartner ebenfalls mit STEP 7 ein Gerätezertifikat zuweisen wie hier bzw. wie bei der S7-1500 CPU beschrieben.

Beispiel: Gesicherte TCP-Verbindung zwischen zwei S7-1500 CPUs über CP-Schnittstellen einrichten

Für die gesicherte TCP-Kommunikation zwischen zwei S7-1500 CPs müssen Sie in jeder CPU einen Datenbaustein mit dem Systemdatentyp TCON_IP_V4_SEC erstellen, parametrieren und direkt an einer der Anweisungen TSEND_C, TRCV_C oder TCON aufrufen.

Voraussetzungen:

- Beide S7-1500 CPUs haben mindestens Firmwarestand V2.0; wenn Sie den CP 1543SP-1 verwenden: Firmwarestand ab V1.0.
- Beide CPs (z. B. CP 1543-1) haben mindestens Firmwarestand V2.0
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate.
  - Ein Gerätezertifikat (End-Entity-Zertifikat) für den CP muss erzeugt sein und sich im Zertifikatsspeicher des CP befinden. Wenn ein Kommunikationspartner ein Fremdgerät ist (z. B. ein MES oder ERP-System), muss für dieses Gerät ebenfalls ein Gerätezertifikat vorhanden sein.
  - Das Stammzertifikat (CA-Zertifikat), mit dem das Gerätezertifikat des Kommunikationspartners signiert ist, muss sich im Zertifikatsspeicher des CP bzw. im Zertifikatsspeicher des Fremdgeräts befinden. Falls Sie Zwischenzertifikate nutzen, müssen Sie sicherstellen, dass der gesamte Zertifikatepfad im validierenden Gerät vorhanden ist. Diese Zertifikate nutzt ein Gerät zur Validierung des Gerätezertifikats des Kommunikationspartners.
- Den Kommunikationspartner müssen Sie grundsätzlich über seine IPv4-Adresse adressieren, nicht über seinen Domänennamen.

Das folgende Bild zeigt die verschiedenen Zertifikate in den Geräten für den Fall, dass beide Kommunikationspartner über einen CP 1543-1 kommunizieren. Außerdem zeigt das Bild die Übertragung der Gerätezertifikate beim Verbindungsaufbau ("Hello").

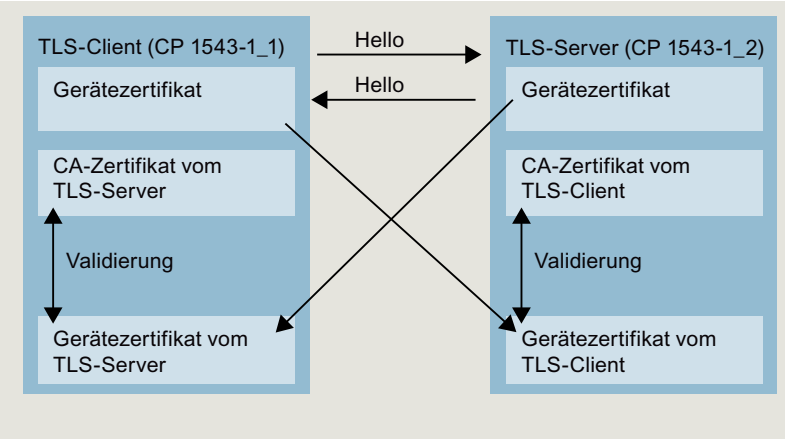


Bild 5-28 Zertifikate-Handling bei Secure OUC zwischen zwei S7-1500 CPUs über CP-Schnittstellen

Einstellungen am TLS-Client

Um eine gesicherte TCP-Verbindung im TLS-Client einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.

2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC. Geben Sie dazu im Feld "Datentyp" die Zeichenfolge "TCON_IP_V4_SEC" ein.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Client" vom Datentyp TCON_IP_V4_SEC definiert ist.
   Die InterfaceId hat den Wert der HW-Kennung der IE-Schnittstelle des lokalen CP (TLS-Client).

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	SEC connection 1 TLS-Client	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	258	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	true	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	100	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against th
19	TLSServerCertRef	UDInt	1	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	5	for Client side: add id of own X.509 V3 client certificate; for Ser

Bild 5-29 IP_V4_SEC_Client

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAddress" die IPv4-Adresse des TLS-Servers ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine ungesicherte TCP- oder UDP-Verbindung einrichten.
   - "TLSServerCertRef": Tragen Sie den Wert 2 (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA256) bzw. den Wert 1 ein (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA1)). Wenn Sie ein anderes CA-Zertifikat nutzen, tragen Sie die entsprechende ID aus dem Zertifikatsmanager der Globalen Security-Einstellungen ein.
   - "TLSClientCertRef": ID des eigenen X.509-V3-Zertifikats.
5. Legen Sie im Programmeditor eine der Anweisungen TSEND_C, TRCV_C oder TCON an.
6. Verschalten Sie den Parameter CONNECT einer der Anweisungen TSEND_C, TRCV_C oder TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Einstellungen am TLS-Server

Um eine gesicherte TCP-Verbindung im TLS-Server einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4_SEC.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "SEC connection 1 TLS-Server" vom Datentyp TCON_IP_V4_SEC definiert ist.
   Die InterfaceId hat den Wert der HW-Kennung der IE-Schnittstelle des lokalen CP (TLS-Server).

Data_block_1

	Name	Data type	Start value	Comment
1	Static			
2	SEC connection 1 TLS-Server	TCON_IP_V4_SEC		
3	ConnPara	TCON_IP_v4		parameter of the TCP connection
4	InterfaceId	HW_ANY	260	HW-identifier of IE-interface submodule
5	ID	CONN_OUC	10	connection reference / identifier
6	ConnectionType	Byte	11	type of connetion: 11=TCP/IP, 19=UDP (17=TCP/IP)
7	ActiveEstablished	Bool	false	active/passive connection establishment
8	RemoteAddress	IP_V4		remote IP address (IPv4)
9	ADDR	Array[1..4] of Byte		IPv4 address
10	ADDR[1]	Byte	192	IPv4 address
11	ADDR[2]	Byte	168	IPv4 address
12	ADDR[3]	Byte	1	IPv4 address
13	ADDR[4]	Byte	10	IPv4 address
14	RemotePort	UInt	4711	remote UDP/TCP port number
15	LocalPort	UInt	4711	local UDP/TCP port number
16	ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17	TLSServerReqClientCert	Bool	true	Just for server side: The TLS server requests a client certificate
18	ExtTLSCapabilities	Word	16#0	Bit 0: Just for client side: validate given IPv4 address against the
19	TLSServerCertRef	UDInt	6	for Server side: Reference to own X.509 V3 server certificate; fo
20	TLSClientCertRef	UDInt	1	for Client side: add id of own X.509 V3 client certificate; for Serv

Bild 5-30 IP_V4_SEC_Server

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "RemoteAddress" die IPv4-Adresse des TLS-Clients ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine unsichere TCP- oder UDP-Verbindung einrichten.
   - "TLSServerReqClientCert ": Anforderung eines X.509-V3-Zertifikats vom TLS-Client. Tragen Sie den Wert "true" ein.
   - "TLSServerCertRef": ID des eigenen X.509-V3-Zertifikats.
   - "TLSClientCertRef": Tragen Sie den Wert 2 (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA256) bzw. den Wert 1 ein (Referenz auf das CA-Zertifikat des TIA Portal-Projekts (SHA1)). Wenn Sie ein anderes CA-Zertifikat nutzen, tragen Sie die entsprechende ID aus dem Zertifikatsmanager der Globalen Security-Einstellungen ein.
5. Legen Sie im Programmeditor eine Anweisung TSEND_C, TRCV_C oder TCON an.
6. Verschalten Sie den Parameter CONNECT der Anweisung TSEND_C, TRCV_C oder TCON mit der Variablen vom Datentyp TCON_IP_V4_SEC.

Laden des Geräts als neue Station

Wenn Sie eine Konfiguration mit Zertifikaten und projektierter Secure Open User Communication als neue Station in Ihr STEP 7-Projekt hochladen, dann werden die Zertifikate des CP im Gegensatz zu den Zertifikaten der CPU nicht mit hochgeladen. Nach dem Laden des Geräts als neue Station sind keine Zertifikate in den entsprechenden Tabellen der CPs für die Gerätezertifikate mehr enthalten.

Sie müssen die Projektierung der Zertifikate nach dem Hochladen erneut durchführen. Andernfalls führt ein erneutes Laden der Konfiguration dazu, dass die ursprünglich im CP vorhandenen Zertifikate gelöscht werden und die Secure Communication nicht funktioniert.

Secure OUC-Verbindungen über CPU- und CP-Schnittstellen - Gemeinsamkeiten

- Verbindungsressourcen:
  Keine Unterschiede zwischen OUC und Secure OUC. Eine programmierte Secure OUC-Verbindung verbraucht ebenso eine Verbindungsressource wie eine OUC-Verbindung, unabhängig davon, über welche IE-/PROFINET-Schnittstelle die Station kommuniziert.
- Verbindungsdiagnose:
  Keine Unterschiede zwischen OUC und Secure OUC-Verbindungsdiagnose.
- Laden von Projekten mit Secure OUC-Verbindungen in die CPU:
  Nur im STOP der CPU möglich, falls Zertifikate mitgeladen werden.
  Empfehlung: Laden in Gerät > Hardware und Software. Grund: Sicherstellen der Konsistenz zwischen Programm mit Secure OUC, Hardware-Konfiguration und Zertifikaten.
  Zertifikate werden mit der Hardware-Konfiguration geladen - daher erfordert das Laden ein Stoppen der CPU. Das Nachladen von Bausteinen, die weitere Secure OUC-Verbindungen nutzen, ist nur dann im RUN möglich, wenn sich die dafür erforderlichen Zertifikate bereits auf der Baugruppe befinden.

5.6.4.5 Secure OUC mit Modbus TCP

Für die gesicherte Modbus TCP-Verbindung müssen Sie selbst einen Datenbaustein mit einem der Systemdatentypen TCON_IP_V4_SEC oder TCON_QDN_SEC erstellen, parametrieren und direkt an der Anweisung MB_Server bzw. MB_CLIENT aufrufen.

Voraussetzungen

- S7-1500 CPU ab Firmware Version V2.5
- Der Modbus-Client (TLS-Client) kann den Modbus-Server (TLS-Server) über IP-Kommunikation im Netzwerk erreichen.
- TLS-Client und TLS-Server besitzen alle notwendigen Zertifikate

Beispiel Gesicherte Modbus TCP-Verbindung zu einem Modbus TCP-Server einrichten

Im Folgenden ist beschrieben, wie Sie eine Secure Open User Communication über Modbus TCP von einem Modbus TCP-Client zu einem Modbus TCP-Server einrichten.
Um eine gesicherte Verbindung von einem Modus TCP-Client (TLS-Client) zu einem Modbus TCP-Server (TLS-Server) einzurichten IPv4-Adresse des Mailservers einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_IP_V4 SEC.

Data_block_1

		Name	Data type	Start value	Comment
1		Static			
2		SEC_ModbusTCP_1	TCON_IP_V4...		
3		ConnPara	TCON_IP_v4		parameter of the TCP connection
4		InterfaceId	HW_ANY	64	HW-identifier of IE-interface submodule
5		ID	CONN_OUC	15	connection reference / identifier
6		ConnectionType	Byte	11	type of connection: 11=TCP/IP, 19=UDP (17=TCP/IP)
7		ActiveEstablished	Bool	true	active/passive connection establishment
8		RemoteAddress	IP_V4		remote IP address (IPv4)
9		ADDR	Array[1..4] of B...		IPv4 address
10		ADDR[1]	Byte	192	IPv4 address
11		ADDR[2]	Byte	168	IPv4 address
12		ADDR[3]	Byte	10	IPv4 address
13		ADDR[4]	Byte	100	IPv4 address
14		RemotePort	UInt	502	remote UDP/TCP port number
15		LocalPort	UInt	502	local UDP/TCP port number
16		ActivateSecureConn	Bool	true	activate the security functionality of that connection in general
17		TLSServerReqClientCert	Bool	false	Just for server side: The TLS server requests a client certificate
18		ExtTLSCapabilities	Word	0	Bit 0: Just for client side: validate given IPv4 address against the subjectAlt
19		TLSServerCertRef	UDInt	2	for Server side: Reference to own X.509 V3 server certificate; for Client side:
20		TLSClientCertRef	UDInt	7	for Client side: add id of own X.509 V3 client certificate; for Server side: add

Bild 5-31 TCON_IP_V4_SEC

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "MailServerAdress" die IPv4-Adresse des Mailservers ein.
4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein. Tragen Sie z. B. bei "TLSServerCertRef" die Zertifikat-ID vom CA-Zertifikat des Kommunikationspartners ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure Modbus TCP-Verbindung einrichten.
   - "TLSServerCertRef": Referenz auf das X.509 V3 (CA)-Zertifikat des Modbus TCP-Servers, welches vom TLS Client benutzt wird, um die Authentifizierung des Modbus TCP-Servers zu validieren.
5. Legen Sie im Programmeditor eine Anweisung MB_Client an.
6. Verschalten Sie den Parameter CONNECT der Anweisung MB_Client mit der Variablen vom Datentyp TCON_IP_V4_SEC.

5.6.4.6 Secure OUC über E-Mail

Gesicherte Verbindung zu einem Mailserver über die Schnittstelle der CPU

Für die gesicherte Verbindung zu einem Mailserver müssen Sie selbst einen Datenbaustein mit einem der Systemdatentypen TMAIL_V4_SEC, TMAIL_QDN_SEC erstellen, parametrieren und direkt an der Anweisung TMAIL_C aufrufen.

Voraussetzungen

- Anweisung TMAIL_C ab Anweisungsversion V5.0
- STEP 7 ab V15
- CPU S7-1500 ab V2.5
- Sie haben der CPU (TLS-Client) alle CA-Zertifikate des Mailservers (TLS-Servers) zugewiesen und die Konfiguration in die CPU geladen.
- Aktuelles Datum und Uhrzeit sind in der CPU eingestellt.

Verfahren zum Aufbau der gesicherten Verbindung zum Mailserver

Sie haben zwei Verfahren zur Auswahl, wie die gesicherte Verbindung zum Mailserver aufgebaut wird:

- SMTPS: Der Client versucht sofort eine TLS-Verbindung zum Mailserver herzustellen ("Handshake"-Verfahren). Wenn der Mailserver TLS nicht unterstützt, dann kommt keine Verbindung zustande.
- STARTTLS: Client baut eine TCP-Verbindung zum Mailserver auf. Über die TCP-Verbindung sendet der Client eine Anfrage zum "Upgrade" der bestehenden Verbindung zu einer gesicherten TLS-Verbindung. Unterstützt der Mailserver TLS, dann sendet er dem Client den Befehl zum Aufbau einer gesicherten Verbindung. Der Mailserver nutzt dazu den SMTP-Befehl "STARTTLS". Der Client baut daraufhin eine gesicherte Verbindung zum Mailserver auf. Vorteil: Wenn der Mailserver kein TLS unterstützt, dann können Client und Mailserver ungesichert miteinander kommunizieren.

Welches Verfahren Sie für die Kommunikation verwenden, legen Sie über die die Einstellung "Remote Port" im Datentypen am Bausteinparameter "MAIL_ADDR_PARAM" fest.

Tabelle 5-5 Portnummern für die Verfahren SMTPS und STARTTLS

Verfahren	Port
SMTPS	465[1]
STARTTLS	beliebig (≠465)[2]

[1] Die Anweisung TMAIL_C verwendet nur für Port 465 SMTPS. Für alle anderen Ports wird STARTTLS verwendet.

[2] gemäß RFC nutzen Mailserver die Ports 25 und 587 für gesicherte Verbindungen mit STARTTLS. Die Verwendung anderer Portnummern für SMTP ist nicht RFC-konform, erfolgreiche Kommunikation mit einem solchen Mailserver ist nicht garantiert.

Beispiel: Gesicherte Verbindung zu einem Mailserver einrichten über IPv4

Im Folgenden ist beschrieben, wie Sie mit der Kommunikationsanweisung TMAIL_C eine gesicherte Verbindung zu einem IPv4-Mailserver einrichten.
Um eine gesicherte Verbindung über die IPv4-Adresse des Mailservers einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TMAIL_V4_SEC.
   Das folgende Beispiel zeigt den globalen Datenbaustein "MailConnDB", in dem die Variable "MailConnectionSEC" vom Datentyp TMAIL_V4_SEC definiert ist.

MailConnDB

Name	Datentyp	Startwert	Kommentar
Static			
MailConnectionSEC	TMail_V4_SEC		
InterfaceId	HW_ANY	"Local~CP_1543-1_1~Ethernet-Schnittstelle_1"	Use HW-identifier of the IE-interface to specify th
ID	CONN_OUC	100	connection reference / identifier
ConnectionType	Byte	16#20	type of connection 16#20=32=TMail_V4 or TMai
ActiveEstablished	Bool	true	active / passive connection establishment
WatchDogTime	Time	T#5000ms	watchdog time to monitor SMTP server associat
MailServerAddress	IP_V4		IPv4 address of mail server
ADDR	Array[1..4] of Byte		IPv4 address
ADDR[1]	Byte	144	IPv4 address
ADDR[2]	Byte	145	IPv4 address
ADDR[3]	Byte	2	IPv4 address
ADDR[4]	Byte	20	IPv4 address
UserName	String[254]	'myName'	user name which is necessary to login into the
PassWord	String[254]	'myPW'	user password which is necessary to login into
From	EMAIL_ADDR		source mail address
LocalPartPlusAt...	String[64]	'Mustermann@'	local part of e-mail address plus "@" sign
FullQualifiedD...	String[254]	'siemens.com'	full qualified domain name part of e-mail addre
RemotePort	UInt	587	remote TCP port number
ActivateSecureConn	Bool	TRUE	activate the security functionality of that conne
ExtTLSCapabilities	Byte	16#0	for further capability extensions of the TLS hand
TLSServerCertRef	UDInt	7	Reference to the X.509 V3 (CA-) certificate of the

Bild 5-32 Datentyp TMAIL_V4_SEC

3. Stellen Sie die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "MailServerAdress" die IPv4-Adresse des Mailservers ein.

HINWEIS

Verbindungsparameter InterfaceId

Beachten Sie, dass Sie ab Anweisungsversion V5.0 der Anweisung TMAIL_C im Datentyp TMAIL_V4_SEC den Wert "**0**" für die InterfaceId. In diesem Fall sucht die CPU selbst nach einer passenden lokalen Schnittstelle der CPU.

4. Stellen Sie die Parameter für Secure Communication in der Spalte "Startwert" ein. Tragen Sie z. B. bei "TLSServerCertRef" die Zertifikat-ID vom CA-Zertifikat des Kommunikationspartners ein.
   - "ActivateSecureConn": Aktivierung von Secure Communication für diese Verbindung. Falls dieser Parameter den Wert FALSE hat, sind die nachfolgenden Sicherheitsparameter irrelevant. Sie können in diesem Fall eine non-secure TCP- oder UDP-Verbindung einrichten.
   - "TLSServerCertRef": Referenz auf das X.509 V3 (CA)-Zertifikat des Mail Servers, welches vom TLS Client benutzt wird, um die Authentifizierung des Mailservers zu validieren.
5. Legen Sie im Programmeditor eine Anweisung TMAIL_C an.
6. Verschalten Sie den Parameter MAIL_ADDR_PARAM der Anweisung TMAIL_C mit der Variablen vom Datentyp TMAIl_V4_SEC.
   Im folgenden Beispiel ist der Parameter Mail_ADDR_PARAM der Anweisung TMAIL_C mit der Variablen "MailConnectionSEC" (Datentyp TMAIL_V4_SEC) verschaltet.

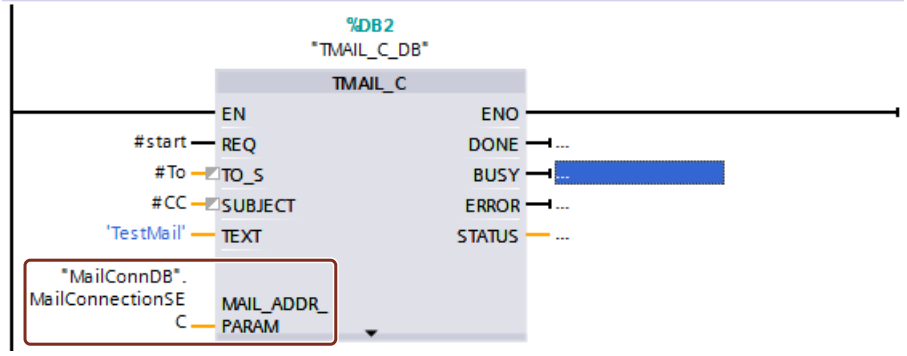


Bild 5-33 Anweisung TMAIL_C

Gesicherte Verbindung zu einem Mailserver über die Schnittstelle eines Kommunikationsmoduls

Für die gesicherte Verbindung zu einem Mailserver über ein Kommunikationsmodul müssen Sie selbst einen Datenbaustein mit einem der Systemdatentypen TMAIL_V4_SEC, TMAIL_QDN_SEC oder TMAIL_V6_SEC (nur CP) erstellen, parametrieren und direkt an der Anweisung TMAIL_C aufrufen.

Voraussetzungen:

- Anweisung TMAIL_C mit Anweisungsversion **V4.0**
- S7-1500 CPU ab Firmwarestand V2.0 mit Kommunikationsmodul CP 1543-1 ab Firmwarestand V2.0
- ET 200SP CPU ab Firmwarestand V2.0 mit Kommunikationsmodul CP 1542SP-1 (IRC) ab Firmwarestand V1.0
- Sie haben dem CP (TLS-Client) alle CA-Zertifikate des Mailservers (TLS-Servers) zugewiesen und die Konfiguration in die CPU geladen.
- Aktuelles Datum und Uhrzeit sind in der CPU eingestellt.

Wie Sie die Verbindung gesicherte Verbindung zu einem Mailserver über die Schnittstelle eines Kommunikationsmoduls festlegen, finden Sie in der Onlinehilfe zu STEP 7 beschrieben.

Anwendungsbeispiel

Wie Sie über den CP einer S7-1500 oder S7-1200 Station eine gesicherte Verbindung zu einem E-Mail-Server einrichten und mit der Standard-Anweisung "TMAIL_C" aus der S7-CPU eine E-Mail verschicken, finden Sie in diesem Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/46817803).

Weitere Informationen

Weitere Informationen zu den Systemdatentypen TMail_V4_SEC und TMAIL_QDN_SEC finden Sie in der Onlinehilfe zu STEP 7.
Weitere Informationen zur sicheren Kommunikation finden Sie im Kapitel Secure Communication (Seite 50).

5.6.5 Secure PG/HMI-Kommunikation

5.6.5.1 PG/HMI-Kommunikation auf Basis standardisierter Security-Mechanismen

Mit den zentralen Komponenten des TIA Portals, STEP 7 und WinCC ist ab Version V17 zusammen mit den aktuellen Steuerungen und aktuellen HMI-Geräten eine innovierte und standardisierte sichere (secure) PG/PC- und HMI-Kommunikation - kurz PG/HMI-Kommunikation - implementiert.
Im Einzelnen sind folgende CPU-Familien gemeint:

- S7-1500 Steuerungsfamilie ab Firmware-Version V2.9
- S7-1200 Steuerungsfamilie ab Firmware-Version V4.5
- Software-Controller ab Firmware-Version V21.9
- SIMATIC Drive Controller ab Firmware-Version V2.9
- PLCSim und PLCSim Advanced Version V4.0

Des weiteren wurden HMI-Komponenten aktualisiert, um Secure PG/HMI-Kommunikation unterstützen zu können:

- Panels bzw. PCs, die mit WinCC Basic, Comfort und Advanced projektiert werden
- PCs mit WinCC RT Professional
- WinCC Unified PCs und Comfort Panels

Ebenfalls aktualisiert werden SINAMICS RT SW ab Version V6.1 und STARTDRIVE ab Version V17.

Eigenschaften der PG/HMI-Kommunikation

Merkmal der PG-Kommunikation und der HMI-Kommunikation ist vor allem ihre Einfachheit: Eine Online-Verbindung von einem Programmiergerät mit installiertem TIA Portal zu einer CPU herzustellen, um z. B. ein Programm zu laden, erfordert wenig Aufwand. Dabei erfüllt diese Online-Verbindung auch Kriterien wie z. B. Vertraulichkeit und Integrität - basierend auf einem bewährten SIMATIC-Kommunkationsstandard.
Im Zuge der Einbindung von Maschinen und Anlagen in eine offene IT-Umgebung müssen allerdings die Weichen gestellt werden für eine Kommunikation zwischen Programmiergerät/HMI-Gerät und CPU, die nicht nur sicher ist im Sinne der Bewahrung von Vertraulichkeit für schützenswerte Daten und Integrität; diese Sicherheit soll auch allgemein anerkannten Standards genügen und damit den Herausforderungen der Zukunft gerecht werden.

Mit der TIA Portal-Version V14 wurde bereits das Verfahren "Open User Communication" für die Anwenderprogramm-basierte Kommunikation um die Variante **"Secure** Open User Communication" erweitert. Weitere zertifikatsbasierte Kommunikationsmechanismen haben sich etabliert (HTTPS, Secure SMTP over TLS oder OPC UA). Ab TIA Portal Version V17 erfährt die PG/HMI-Kommunikation ebenso eine Aufwertung: Auch hier hält das Protokoll TLS (Transport Layer Security) Einzug, um die PG/HMI-Kommunikation mittels standardisierter Security-Mechanismen abzusichern.

Das ändert sich

Zusätzliches optionales Passwort für mehr Security

Auffälligste Änderung im Rahmen der Projektierung der oben genannten Geräte ist die Möglichkeit, ein Passwort zum Schutz vertrauenswürdiger Konfigurationsdaten der jeweiligen CPU zu vergeben. Damit sind Daten gemeint wie z. B. private Schlüssel, die für die ordnungsgemäße Funktion zertifikatsbasierter Protokolle notwendig sind (Secure Communication) - Ab TIA Portal V17 auch für die PG/HMI-Kommunikation. Sie haben die Möglichkeit, die vergebenen Passwörter über eine Richtlinien-Einstellung bereits während der Eingabe im TIA Portal zu prüfen und damit die Einhaltung vorgegebener Passwort-Richtlinien in Ihrem Unternehmen sicherzustellen.

Wenn Ihre Maschine oder Anlage nach ganzheitlicher Betrachtung basierend auf dem Siemens Industrial-Defense-in-Depth-Konzept diesen Schutz nicht erfordert, z. B. weil ein anderer, gleichwertiger Schutz vorhanden ist, können Sie auf die Passwort-Vergabe verzichten. Ein Verzicht auf das Passwort ist möglich, wenn Sie Maßnahmen umgesetzt haben, die einen unberechtigten Zugriff auf das TIA Portal Projekt und die Projektierung der CPU unterbinden.

⚠ WARNUNG

Ohne Passwort schwacher Schutz privater Schlüssel

Beachten Sie, dass ohne ein Passwort zum Schutz vertrauenswürdiger Konfigurationsdaten die privaten Schlüssel für Zertifikate, die für Secure Communication benötigt werden, nur schwach geschützt sind.

Zertifikatsbasierte Kommunikation zwischen PG/HMI und CPU

Weil die PG/HMI-Kommunikation zertifikatsbasiert abläuft, werden Sie im Verlauf der Inbetriebnahme zum Akzeptieren des Server-Zertifikats aufgefordert.

Weitere Parametriermöglichkeiten erlauben Ihnen, das Verhalten der CPU im Betrieb zu bestimmen: Sie können z. B. einstellen, dass die CPU auch Verbindungen mit Geräten zulässt, wenn diese die Secure PG/HMI-Kommunikation nicht unterstützen.

Wartung / Ersatzteilfall

Für den problemlosen Austausch der CPU im Ersatzteilfall müssen Sie bestimmte Regeln berücksichtigen (siehe Regeln für den Ersatzteilfall (Seite 85)).

Weitere Informationen

Eine Übersicht zum Schutz vertraulicher Konfigurationsdaten finden Sie in Kap. Schutz vertraulicher Konfigurationsdaten (Seite 74).

5.6.5.2 Weitere Einstellungen für die Secure PG/HMI-Kommunikation

Neben der Vergabe eines Passworts zum Schutz vertraulicher PLC-Konfigurationsdaten haben Sie noch weitere Einstellungsmöglichkeiten für das Verhalten der CPU im Betrieb.

Modus der PG/PC- und HMI-Kommunikation

Sie können einstellen, wie die CPU mit Programmiergeräten und HMI-Geräten kommunizieren darf:

- Nur über Secure PG/HMI-Kommunikation
- Sowohl über Secure PG/HMI-Kommunikation als auch über die bisher verwendete PG/HMI-Kommunikation, kurz "Legacy-PG/HMI-Kommunikation" genannt.

Vorgehen

1. Navigieren Sie in den CPU-Eigenschaften zum Bereich "Schutz & Security > Verbindungsmechanismen".
2. Wählen Sie die gewünschte Option.

Zertifikat auswählen oder neu erzeugen

Wenn Sie den Verbindungsmechanismus für PG/HMI-Kommunikation auswählen, können Sie im selben Kontext ein PLC-Kommunikationszertifikat für die Absicherung der Verbindung auswählen bzw. vom TIA Portal neu erzeugen lassen. Wenn Sie ein Passwort vergeben haben bzw. wenn Sie die Option zum Schutz vertraulicher PLC-Konfigurationsdaten deaktiviert haben (d. h. kein Passwort gesetzt), dann ist im Bereich "Schutz & Security > Verbindungsmechanismen" bereits ein Zertifikat mit passenden Einstellungen und einem gültigen Default-Namen voreingestellt.

Vorgehen

Falls Sie vom TIA Portal ein neues Zertifikat erzeugen lassen wollen oder ein anderes bereits bestehende Zertifikat auswählen wollen:

1. Klicken Sie im Feld "PLC-Kommunikationszertifikat" auf die die drei Punkte zum Erweitern des Felds.
2. Wählen Sie das gewünschte Zertifikat aus oder klicken Sie auf die Schaltfläche "Hinzufügen".
3. Beim Hinzufügen erscheint ein Dialog mit Einstellungsmöglichkeiten für das Zertifikat. Der Verwendungszweck ist auf "TLS-Server" festgelegt, andere Parameter (z. B. Name, Hash-Algorithmus, ...) können Sie ändern.

Es gelten die allgemeinen Regeln für das Zertifikatsmanagement; d. h. wenn Sie ein CA-Zertifikat erzeugen wollen, dann muss die Option "Globale Einstellungen für den Zertifikatsmanager" aktiviert sein. Sie haben aber auch die Möglichkeit, ein selbstsigniertes PLC-Zertifikat zu erzeugen.

Weitere Informationen

Grundlegendes zum Thema Zertifikatsmanagement finden Sie in Kap. Zertifikatsmanagement mit TIA Portal (Seite 63).

5.6.5.3 Tipp zur zertifikatsbasierten Kommunikation zwischen PG und CPU

Die zertifikatsbasierte PG/PC-Kommunikation (Secure PG/PC-Kommunikation) bringt es mit sich, dass der Kommunikationspartner der CPU - das Programmiergerät mit installiertem TIA Portal - dem Gerätezertifikat der CPU vertrauen muss, damit eine Verbindung aufgebaut werden kann.

Vereinfacht beschrieben gibt es aus TIA Portal-Sicht folgende Möglichkeiten, dem Zertifikat einer CPU zu vertrauen:

- Das PG mit TIA Portal ist im Besitz des Gerätezertifikats der CPU, weil es z. B. im Projekt erzeugt oder importiert wurde. In diesem Fall läuft die Zertifikatsprüfung automatisch und ohne Rückfrage ab.
- Das PG mit TIA Portal ist nicht im Besitz des Gerätezertifikats der CPU; z. B. weil die CPU über "Erreichbare Teilnehmer" ermittelt wurde und nicht im Projekt vorhanden ist. In dem Fall fragt das TIA Portal den Benutzer des TIA Portals, ob dem Zertifikat vertraut werden kann. Möglicherweise ist das nur mit großem Aufwand möglich, weil die CPU z. B. nicht in Sichtweite ist und daher die Authentizität nicht unmittelbar geprüft werden kann.
- Das PG mit TIA Portal ist im Besitz des CA-Zertifikat (Zertifizierungsstelle) und alle CPUs, die im Netz vom TIA Portal erreicht werden können, haben Gerätezertifikate, die von diesem CA-Zertifikat ausgestellt wurden.

  Vorteil dieser Lösung: Das TIA Portal kann Gerätezertifikate automatisch prüfen, auch wenn die Gerätezertifikate der Kommunikationspartner nicht im TIA Portal vorhanden sind.

Im Folgenden ist die Lösung mit einem CA-Zertifikat (Zertifizierungsstelle) näher erläutert.

Voraussetzung

Mit der Zertifizierungsstelle des TIA Portals können Sie für eine CPU Gerätezertifikate erzeugen und dafür die vorhandenen CA-Zertifikate zum Signieren der Gerätezertifikate nutzen. Sie können aber ebenso eine andere Zertifizierungsstelle ins TIA Portal importieren und verwenden.

Voraussetzung ist die Aktivierung der Globalen Security-Einstellungen für den Zertifikatsmanager. Nur mit dieser Einstellung können Sie CA-signierte Zertifikate erzeugen.

Siehe auch hier: Zertifikatsmanagement mit TIA Portal (Seite 63)

CA-Zertifikat für Programmiergeräte exportieren

Um nach dem Anlegen und Zuweisen eines Zertifikats das entsprechende CA-Zertifikat zu exportieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Projektnavigation in den globalen Security-Einstellungen den Zertifikatsmanager.
2. Wählen Sie für das zu exportierende Zertifikat die Tabelle "CA-Zertifikate".
3. Öffnen Sie bei selektiertem Zertifikat mit Rechtsklick das Kontextmenü.
4. Klicken Sie auf "Exportieren".
5. Wählen Sie das Exportformat des Zertifikats und den Ablageort.

CA-Zertifikat im TIA Portal ablegen

Um das exportierte Zertifikat einem PG mit TIA Portal bekannt zu machen und damit die automatische Zertifikatsprüfung zu ermöglichen, gehen Sie folgendermaßen vor:

1. Kopieren Sie das im vorhergehenden Schritt exportierte CA-Zertifikat in das folgende Verzeichnis:
   C:\ProgramData\Siemens\Automation\Certstore\Trusted
2. Starten Sie das TIA Portal.
   Im Register "Info" des Inspektorfensters erscheint für jedes CA-Zertifikat eine Meldung, die Auskunft darüber gibt, ob das CA-Zertifikat erfolgreich in den CA-Store des TIA Portals übernommen werden konnte.
   Bei Misserfolg werden allerdings keine detaillierten Ursachen ausgegeben.

Gerätezertifikate in die TIA Portal Zertifikatssperrliste (CRL) aufnehmen

Sie haben die Möglichkeit, einzelne Gerätezertifikate zu einer Zertifikatssperrliste (CRL) hinzuzufügen, z. B. weil der zugehörige Schlüssel als nicht mehr sicher angesehen wird.

Wenn das TIA Portal eine Verbindung mit einer CPU aufnimmt, deren Gerätezertifikat sich in der Zertifikatssperrliste befindet, dann erscheint ein Dialog im TIA Portal, ob Sie dem Zertifikat dennoch trauen wollen. Wenn Sie ablehnen, wird die Verbindung nicht aufgebaut.

Um ein Gerätezertifikat in die Zertifikatssperrliste aufzunehmen, gehen Sie folgendermaßen vor:

1. Kopieren Sie das Gerätezertifikat in das folgende Verzeichnis:
   C:\ProgramData\Siemens\Automation\Certstore\CRL
2. Starten Sie das TIA Portal.
   Im Register "Info" des Inspektorfensters erscheint für jedes Zertifikat eine Meldung, die Auskunft darüber gibt, ob das Zertifikat erfolgreich in den CRL-Store des TIA Portals übernommen werden konnte.
   Bei Misserfolg werden allerdings keine detaillierten Ursachen ausgegeben.

5.6.5.4 CPU-Verhalten vom Laden bis zur Betriebsbereitschaft

Damit die Kommunikation zwischen CPU und einem Programmiergerät oder HMI-Gerät abgesichert ist, muss sie zunächst ein Zertifikat haben. Das Zertifikat für den produktiven Betrieb bekommt sie aber erst mit dem Laden des Projekts in die CPU.

Damit das initiale Laden ebenfalls abgesichert ist, erstellt die CPU zunächste ein selbst signiertes Zertifikat. Die folgende Beschreibung erläutert die verschiedenen Phasen des Verbindungsaufbaus.

Voraussetzung für den erstmaligen Verbindungsaufbau zum Laden der CPU

- Kein Passwort für vertrauliche PLC-Konfigurationsdaten in der CPU vorhanden.
  Falls die CPU bereits geladen wurde und daher schon ein Passwort für vertrauliche PLC-Konfigurationsdaten hat, dann muss dieses Passwort zum Projekt passen, das geladen werden soll.
- Projekt mit CPU-Konfiguration (inklusive Passwort für vertrauliche PLC-Konfigurationsdaten) und Anwenderprogramm liegt vor.
- Die CPU befindet sich im Betriebszustand STOP.
- Programmiergerät und CPU sind direkt mitinander verbunden und befinden sich in einer geschützten Umgebung, d. h. Sie können die CPU identifizieren, die geladen werden soll und die Verbindung zwischen CPU und Programmiergerät kontrollieren.

Erstmaliger Verbindungsaufbau zur CPU - Bereitstellungsphase

Schon der erste Verbindungsaufbau zum Laden der CPU ist über das TLS-Verfahren abgesichert im Sinne einer Secure PG/HMI-Kommunikation.

Allerdings nutzt die CPU für diesen Verbindungsaufbau ihr Hersteller-Gerätezertifikat (falls vorhanden) oder ein selbstsigniertes Zertifikat. Die CPU ist in dieser Phase nur eingeschränkt nutzbar. In dieser Phase wartet die CPU auf die Bereitstellung der Passwort-basierten Schlüsselinformation - oder einfacher: sie erwartet das Passwort für vertrauliche PLC-Konfigurationsdaten. Im Folgenden wird diese Phase Bereitstellungsphase genannt. Dass sich die CPU in der Bereitstellungsphase befindet, signalisiert sie durch eine entsprechende Meldung im Diagnosepuffer.

Mit dem Laden eines Projekts in die CPU erhält die CPU die Projektdaten:

- Hardware-Konfiguration inklusive projektierter Zertifikate für Secure Kommunikation (OPC UA, HTTPS, Secure OUC, Secure PG/HMI-Kommunikation)
- Anwenderprogramm

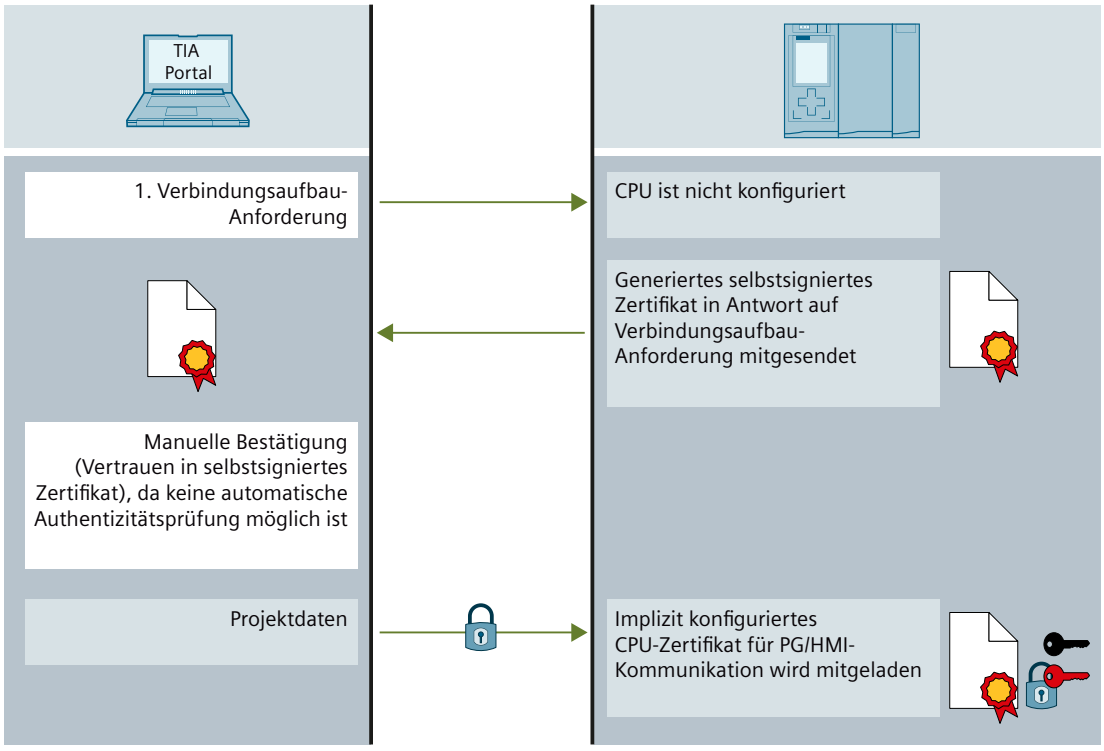


Bild 5-34 Verbindungsaufbau Bereitstellungsphase

⚠ WARNUNG

Potentielle Sicherheitsrisiken bei der Inbetriebnahme

Bei der Inbetriebnahme stellt die CPU ein Hersteller-Gerätezertifikat (falls vorhanden) oder ein selbstsigniertes Zertifikat zur Verfügung, dem Sie vertrauen müssen, damit ein Verbindungsaufbau zustande kommt. Vertrauen Sie diesem Zertifikat nur, wenn sich das Programmiergerät und die CPU in einem geschützten Netz befinden und direkt miteinander verbunden sind. In ungeschützter Umgebung können diese Zertifikate manipuliert werden und Angreifern den Zugriff auf die Kommunikation zwischen Programmiergerät/HMI und CPU ermöglichen (z. B. durch Man-in-the-Middle-Angriffe).

Bereitstellungsphase beenden

Das Passwort für vertrauliche PLC-Konfigurationsdaten selbst bzw. die Schlüsselinformation, die aus dem Passwort erzeugt wird, speichert das TIA Portal nicht im Projekt.

Das Passwort wird daher beim ersten Laden bzw. beim Laden eines neuen Projekts dialoggeführt abgefragt und als Schlüsselinformation zur CPU transferiert. Erst durch diesen Schritt ist die CPU in der Lage, die geschützten PLC-Konfigurationsdaten zu nutzen - damit ist die Bereitstellungsphase beendet und die CPU kann in Betrieb gehen.

Wenn Sie die vertraulichen PLC-Konfigurationsdaten nicht durch ein Passwort schützen, entfällt auch die Eingabe des Passworts beim ersten Laden. Auf den Ablauf der PG/HMI-Kommunikation hat das keinen Einfluss, allerdings müssen Sie in diesem Fall bedenken, dass die vertraulichen PLC-Konfigurationsdaten (z. B. private Schlüssel) kaum vor unbefugtem Zugriff geschützt sind.

Hochlauf der PG/HMI-Kommunikation

Wenn die CPU geladen ist und das CPU-Zertifikat für secure PG/HMI-Kommunikation erhalten hat, verbindet sich das Programmiergerät erneut - diesmal auf Grundlage des geladenen CPU-Zertifikats.

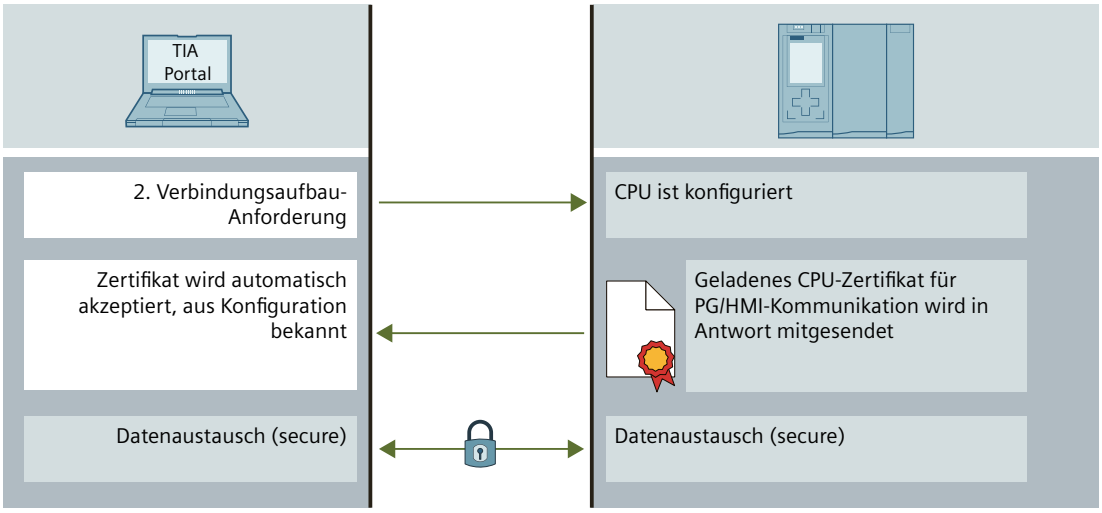


Bild 5-35 Hochlauf der PG/HMI-Kommunikation

5.6.5.5 Secure HMI-Kommunikation verwenden

Ab TIA Portal Version V17 kommunizieren CPU und HMI-Gerät über Secure HMI-Kommunikation, sofern beide Geräte die Voraussetzungen dafür erfüllen.

Grundlage der Secure HMI-Kommunikation ist, dass das HMI-Gerät die Authentizität der CPU über ihr PLC-Kommunikationszertifikat, das die CPU beim Verbindungsaufbau sendet, prüfen und als "vertrauenswürdig" einstufen kann. Nur in diesem Fall ist die Secure HMI-Kommunikation möglich.

Dieses Kapitel beschreibt, welche Maßnahmen Sie für die verschiedenen HMI-Geräte ergreifen müssen, um das PLC-Kommunikationszertifikat manuell als "vertrauenswürdig" zu kennzeichnen.

Voraussetzung

- CPU und HMI-Gerät unterstützen Secure HMI-Kommunikation.
- Auf der CPU befindet sich ein aktuelles Projekt (TIA Portal ab V17)

Secure HMI-Kommunikation projektieren

1. Projektieren Sie das HMI-Gerät mit einer Meldeanzeige.

   HINWEIS

   Ohne die Meldeanzeige können Sie Fehler beim Verbindungsaufbau nicht erkennen.

2. Projektieren Sie die CPU mit den erforderlichen Security-Einstellungen. Wählen Sie ein PLC-Kommunikationszertifikat zur Absicherung der HMI-Verbindung aus oder lassen Sie ein PLC-Kommunikationszertifikat vom TIA Portal erzeugen.
3. Projektieren Sie die HMI-Verbindung zwischen CPU und HMI-Gerät.
4. Laden Sie das Projekt in die CPU und in das HMI-Gerät. Beim Projekt-Transfer wird das PLC-Kommunikationszertifikat und ggf. auch ein erforderliches CA-Zertifikat (Zertifizierungsstelle) auf die CPU und auf das HMI-Gerät übertragen.

Dem PLC-Kommunikationszertifikat vertrauen

Während des Verbindungsaufbaus überträgt die CPU das PLC-Kommunikationszertifikat an das HMI-Gerät.

- Wenn das PLC-Kommunikationszertifikat bereits im Status "vertrauenswürdig" auf dem HMI-Gerät vorliegt, dann wird automatisch eine Secure HMI-Kommunikation zwischen CPU und HMI-Gerät aufgebaut.
- Wenn das PLC-Kommunikationszertifikat noch nicht im Status "vertrauenswürdig" auf dem HMI-Gerät vorliegt, dann erscheint eine Meldung in der Meldeanzeige des HMI-Geräts, dass der CPU nicht vertraut wird sowie ein Fehlercode.

  Sie müssen in diesem Fall das PLC-Kommunikationszertifikat auf dem HMI-Gerät als "vertrauenswürdig" kennzeichnen.

Gehen Sie abhängig vom Typ Ihres HMI-Geräts wie folgt vor.

Basic Panels 2nd Generation

1. Wählen Sie im Start Center "Settings > Internet Settings > Certificate store".
2. Selektieren Sie in der Liste "Available certificates in Device" das PLC-Kommunikationszertifikat der CPU.
3. Betätigen Sie die Schaltfläche "Trust".
4. Starten Sie die HMI Runtime-Software neu.

Unified Comfort Panels

1. Öffnen Sie das Control Panel.
2. Wählen Sie "Security > Certificates".
3. Wählen Sie in der Auswahlliste "Certificate store" den Eintrag "Other Certificates".
4. Selektieren Sie in der Liste "Other certificates" das PLC-Kommunikationszertifikat der CPU.
5. Betätigen Sie die Schaltfläche "Trust".
6. Starten Sie die HMI Runtime-Software neu.

Comfort Panels, Mobile Panels 2nd Generation

1. Öffnen Sie den Dateimanager über das Windows CE Desktop-Symbol "My Device".
2. Navigieren Sie in das Verzeichnis "\flash\simatic\SystemRoot\OMS\Untrusted". Dort finden Sie das PLC-Kommunikationszertifikat der CPU.
3. Kopieren Sie das PLC-Kommunikationszertifikat der CPU in das Verzeichnis "\flash\simatic\SystemRoot\OMS\Trusted".
4. Starten Sie die HMI Runtime-Software neu.

Wenn das PLC-Kommunikationszertifikat im Status "vertrauenswürdig" auf dem HMI-Gerät vorliegt, dann kann die Secure HMI-Kommunikation aufgebaut werden. Weitere Informationen finden Sie in der Betriebsanleitung des HMI-Geräts.

5.6.5.6 Legacy-PG/PC-Kommunikation für TIA Portal verwenden

Ab TIA Portal Version V17 kommunizieren TIA Portal und S7-1200/S7-1500 CPU ab Firmwarestand V4.5/V2.9 automatisch "secure" - die Verbindungspartner stellen ihre Verbindungsmechanismen automatisch auf das höchstmögliche Security-Verfahren ein.
Nur besondere Umstände (siehe Informationen zur Kompatibilität (Seite 113)) bewirken ein Zurückfallen auf die bisherige PG/PC-Kommunikation, kurz "Legacy PG/PC-Kommunikation" genannt.
Es kann aber Einsatzfälle geben, bei denen die höhere Security nicht gewünscht ist, da sie die Übertragungsgeschwindigkeit bei CPUs mit geringerer Kommunikations-Performance beeinträchtigen kann.

Voraussetzung

- Es dürfen keine Online-Verbindungen zu CPUs aufgebaut sein.
- Für CPUs, die online erreicht werden sollen, darf die Option "Nur Secure PG/PC- und HMI-Kommunikation zulassen" nicht aktiviert sein (CPU-Parameter im Bereich "Verbindungsmechanismen").
- Die Kommunikationspartner befinden sich in einer geschützten Umgebung z. B. während der Inbetriebnahmephase.

Legacy PG/PC-Kommunikation einstellen

1. Wählen Sie den Befehl "Nur Legacy PG/PC-Kommunikation verwenden" im Menü "Online".
2. Aktivieren Sie das Optionskästchen vor dem Menübefehl.

Ergebnis: Alle Online-Verbindungen werden wie bei TIA Portal-Versionen < V17 aufgebaut.
Die Einstellung bleibt für die Dauer der Sitzung bestehen. Wenn Sie ein Projekt öffnen, ist die Option "Nur Legacy-PG/PC-Kommunikation verwenden" nicht gesetzt.

Verhalten bei aktivierter Option "Nur Legacy PG/PC-Kommunikation verwenden"

- Ein Passwort zum Schutz vertraulicher PLC-Konfigurationsdaten kann online für CPUs weder festgelegt, geändert oder gelöscht werden. Diese Funktionen erfordern ein Deaktivieren der Option "Nur Legacy PG/PC-Kommunikation verwenden".
- Eine CPU, die so eingestellt ist, dass sie nur Secure PG/PC- und HMI-Kommunikation zulässt, kann nicht mehr online erreicht werden.

5.6.5.7 Informationen zur Kompatibilität

Die folgende Beschreibung gibt Auskunft über das Zusammenspiel zwischen verschiedenen TIA Portal Versionen mit den unterschiedlichen CPU-Firmware-Versionen und die Auswirkungen auf die Art der PG/HMI-Verbindung.

Projekte erstellt mit TIA Portal < V17

Wenn Sie ein Projekt z. B. mit TIA Portal Version V16 für eine S7-1500 CPU (z. B. Version V2.8) erstellt haben, dann ist die entsprechende Konfiguration mit TIA Portal V17 auch auf eine S7-1500 CPU V2.9 z. B. im Ersatzteil-Fall ladbar - und zwar mit demselben Verhalten wie ein auf einer S7-1500 CPU V2.8.

Auch Projekte, die mit TIA Portal < V17 erstellt und auf eine Memory Card übertragen wurden, funktionieren problemlos in einer S7-1500 CPU V2.9.

Sobald Sie allerdings das Projekt mit TIA Portal ≥ V17 öffnen, die Firmware-Version der CPU über einen Gerätetausch aktualisieren und damit als CPU mit einer Firmware-Version ≥ V2.9 abspeichern, gilt das Konzept zum Schutz vertraulicher PLC-Konfigurationsdaten (siehe Wissenswertes zum Schutz vertraulicher PLC-Konfigurationsdaten (Seite 77)). Das Projekt ist nicht mehr mit Vorgänger-Versionen von TIA Portal V17 bearbeitbar.

PG/HMI und CPU unterschiedlich verbunden

Wie in den vorangegangenen Kapiteln erläutert, zeichnet sich die Secure PG/HMI-Verbindung ab V17 zwischen PG/HMI-Gerät und CPU (aktuelle Version) durch das verwendete standardisierte Kommunikationsverfahren TLS (Transport Layer Security) aus.

Sie haben die Möglichkeit, eine V2.9-CPU mit einem aktuellen Programmiergerät mit TIA Portal ab V17 zu verbinden und zusätzlich z. B. mit einem HMI-Gerät mit einer Runtime aus der Vorgänger-Version: Die Geräte stellen ihre Verbindungsmechanismen automatisch darauf ein. Um die beiden Verbindungsmechanismen besser unterscheiden zu können, nennen wir das bisherige Verfahren, das auf einer Variante der S7-Kommunikation basiert, "Legacy-Verfahren".

Zusammengefasst ("PG" steht hier für ein Programmiergerät mit TIA Portal):

- PG/HMI und CPU kommen mit der V17 (oder Folgeversion): TLS-Verfahren wird verwendet.
- PG/HMI kommt aus einer Vorgänger-Version (< V17): Legacy-Verfahren wird verwendet - vorausgesetzt, Sie haben in den CPU-Eigenschaften die Option "Nur secure PG/PC- und HMI-Kommunikation zulassen" deaktiviert.
- CPU kommt mit der V17 (oder größer), mehrere PGs/HMIs sind angeschlossen und kommen sowohl aus V17 (oder größer) und Vorgänger-Versionen: TLS+Legacy-Verfahren werden verwendet - vorausgesetzt, Sie haben in den CPU-Eigenschaften die Option "Nur secure PG/PC- und HMI-Kommunikation zulassen" deaktiviert.

Wenn sich der CPU-Zustand ändert

Der Diagnosepuffer gibt Ihnen Auskunft, wenn sich ihr Zustand aufgrund von Ereignissen im Zusammenhang mit der Secure PG/HMI-Kommunikation ändert.

Beispiele:

- Nach erfolgreichem Laden einer Konfiguration mit projektiertem Passwort meldet der Diagnosepuffer, dass die CPU von der Bereitstellungsphase in den Secure Mode (TLS-Verfahren) wechselt.
- Sie haben ein PG mit TIA Portal V17 an eine CPU V2.9 angeschlossen. Automatisch ergibt sich die Secure PG/HMI-Kommunikation (TLS-Verfahren), sofern "nur Legacy PG/PC-Kommunikation verwenden" im Online-Menü deaktiviert ist.

Weitere Informationen

Informationen zu Geräte- bzw. Firmware-spezifischen Merkmalen wie z. B. die genutzte TLS-Version finden Sie im Kap. Geräteabhängige Security-Merkmale (Seite 53).

5.7 SNMP

5.7.1 SNMP aktivieren und deaktivieren

Das Netzwerk-Management-Protokoll SNMP (Simple Network Management Protocol) wird zur Überwachung und Diagnose der Netzwerktopologie genutzt. SNMP nutzt das Transportprotokoll UDP und kennt 2 Rollen: Den SNMP-Manager (Client) und den SNMP-Agent (Server).

- Der SNMP-Manager überwacht die Netzwerkknoten.
- Die SNMP-Agents sammeln in den einzelnen Netzwerkknoten verschiedene netzwerkspezifische Informationen und legen Sie in strukturierter Form in der MIB (Management Information Base) ab. Mit Hilfe dieser Daten können verschiedene Dienste und Tools (als SNMP-Manager) eine ausführliche Netzwerkdiagnose durchführen.

SNMP wird auch in einem PROFINET IO-System zur Verwaltung der Netzwerkinfrastruktur und der IO-Controller/IO-Devices verwendet.

HINWEIS

Wenn SNMP für ein Gerät deaktiviert ist, dann stehen Ihnen verschiedene Möglichkeiten zur Diagnose der Netzwerktopologie (z.B. über das PRONETA-Tool) nicht mehr zur Verfügung.

Beispiel: Das TIA Portal ermittelt für den Topologievergleich Online-Offline, welche Ports tatsächlich verschaltet sind und nutzt für diese Funktion SNMP.

Voreinstellungen abhängig von der Firmware-Version

S7-1500 CPUs haben einen integrierten SNMP-Agent. Je nach Firmware-Version gilt für SNMP eine unterschiedliche Voreinstellung (SNMP aktiviert oder deaktiviert).

Bei S7-1500 CPUs mit einer FW-Version < V3.0 ist der SNMP-Agent in der Voreinstellung **aktiviert** und kann nur per Datensatz im Anwenderprogramm deaktiviert werden.

Unter bestimmten Voraussetzungen kann es sinnvoll sein, SNMP zu deaktivieren. Beispiele:

- Die Sicherheitsrichtlinien in Ihrem Netzwerk lassen den Einsatz von SNMP nicht zu.
- Sie verwenden eine eigene SNMP-Lösung, z. B. über eigene Kommunikationsanweisungen.

Bei S7-1500 CPUs ab einer FW-Version V3.0 ist der SNMP-Agent in der Voreinstellung **deaktiviert**. Die Voreinstellung "deaktiviert" ist auch dann wirksam, wenn keine Projektierung geladen wurde oder wenn keine Memory Card gesteckt ist. Um bei S7-1500 CPUs ab einer FW-Version V3.0 die SNMP-Einstellungen zu ändern, bietet STEP 7 V18 die folgenden Möglichkeiten:

- Konfigurieren Sie SNMP in den CPU-Eigenschaften im TIA Portal.
- Aktivieren/Deaktivieren Sie SNMP im Anwenderprogramm durch die Übertragung eines Datensatzes an eine PROFINET-Schnittstelle.

HINWEIS

Ersatzteilfall

Aus Kompatibilitätsgründen verhält sich eine S7-1500 CPU ab Firmware Version V3.0 mit einem geladenen Vorgänger-Projekt (CPU-Firmware < V3.0) wie die CPU im Vorgänger-Projekt:

SNMP ist aktiviert und Community-Strings "public" und "private" sind wirksam.

SNMP konfigurieren

Ab CPU-Firmware-Version V3.0 und TIA Portal Version V18 haben Sie die Möglichkeit, folgende Einstellungen für SNMP in den CPU-Eigenschaften zu ändern:

- SNMP aktivieren (Voreinstellung: deaktiviert)
- Read-only Community-String (Voreinstellung: "public")
- Read-write Community-String (Voreinstellung: "private")

Die Einstellungen finden Sie im Bereich "Erweiterte Konfiguration > SNMP".

Ab CPU-Firmware-Version V3.1 und TIA Portal Version V19 haben Sie bei aktiviertem SNMP zusätzlich die Möglichkeit, den schreibgeschützten Zugriff für SNMP zu aktivieren.

Bedeutung und Eigenschaften von Community-Strings

Ein SNMP Community-String, auch "Community-Name" genannt, ist wie eine Kennung oder ein Passwort, das den Zugriff auf Informationen/Statistiken eines Geräts, z. B. eines Routers, ermöglicht.
Insofern sollten Sie zur Erhöhung der Sicherheit die voreingestellten Community-Strings in den CPU-Eigenschaften ändern. Ein SNMP-Manager authentisiert sich beim SNMP-Agent durch die Übertragung des Community-Strings bei einem Request.

Die Community-Strings werden als Klartext übertragen.

- Der voreingestellte Community-String für SNMP-Read-only-Operationen (GET) ist "public".
- Der voreingestellte Community-String für SNMP-Read-Write-Operationen (SET) ist "private".

Anzahl der Zeichen für den Community-String: 1-240.

Folgende Zeichen können Sie für den Community-String verwenden:

- a-z
- A-Z
- 0-9
- -
- .

SNMP im Anwenderprogramm aktivieren/deaktivieren

Zusätzlich zur Konfiguration in den CPU-Eigenschaften können Sie SNMP auch im Anwenderprogramm aktivieren bzw. deaktivieren. Übertragen Sie dazu einen Datensatz 0xB071 zu einer PROFINET-Schnittstelle der CPU. In diesem Datensatz ist kodiert, ob SNMP aktivert/deaktiviert werden soll. Egal an welche PROFINET-Schnittstelle Sie den Datensatz übertragen, der Datensatz wirkt an allen Schnittstellen der CPU.

Eine Möglichkeit, den Datensatz 0xB071 zu übertragen: Definieren Sie die Datensatzstruktur in einem Datenbaustein und übertragen Sie ihn im Programmzyklus-OB (z. B. OB1) mit der Anweisung "WRREC" (Datensatz Schreiben) an eine PROFINET-Schnittstelle der CPU.

Dazu gehen Sie folgendermaßen vor:

1. Legen Sie in STEP 7 einen Datenbaustein an, der die Struktur des Datensatzes 0xB071 enthält.
   Die folgende Tabelle zeigt die Struktur des Datensatzes 0xB071.

Byte	Element	Kodierung	Erläuterung
0-1	BlockID	0xF003	Header Die Datensatzlänge wird ab dem Byte 4 "Version" gezählt.
2-3	BlockLength	8	
4	Version	0x01	
5	Subversion	0x00	
6-7	Reserviert	-	-
8-11	SNMP-Steuerung	0,1	0: SNMP deaktivieren. 1: SNMP aktivieren.

2. Übertragen Sie den Datensatz 0xB071 z. B. im Programmzyklus-OB (OB1) mit der Anweisung "WRREC" (Datensatz Schreiben) an die CPU. Nutzen Sie als Hardware-Kennung eine integrierte PROFINET-Schnittstelle der CPU.

Zusammenspiel von SNMP-Konfiguration und Anwenderprogramm

- Die SNMP-Einstellung "aktiviert/deaktiviert" über das Anwenderprogramm ist nicht dauerhaft in der CPU gespeichert. Z. B. nach jedem NETZ-AUS/NETZ-EIN-Übergang, Laden einer neuen Hardware-Konfiguration oder Rücksetzen auf Werkseinstellungen ist die konfigurierte Einstellung wieder wirksam.
- Beim Laden der Konfiguration aus der CPU ("Laden des Geräts als neue Station") wird die konfigurierte SNMP-Einstellung (aktiviert/deaktiviert) berücksichtigt. Eine zuvor per Datensatz im Anwenderprogramm gesetzte SNMP-Einstellung wird nicht berücksichtigt.
- Die Community-Strings können nur in der Konfiguration geändert werden; die Community-Strings sind nicht per Datensatz im Anwenderprogramm einstellbar.
  Sie können aber die projektierten Community-Strings per Datensatz aktivieren.
  Beispiel:
  Sie haben in der Konfiguration einer S7-1500 CPU eingestellt, dass SNMP deaktiviert ist. Sie ändern die voreingestellten Community-Strings in den CPU-Eigenschaften und laden anschließend die Konfiguration in die CPU.
  Per Datensatzübertragung aktivieren Sie anschließend SNMP.
  Ergebnis: Die geänderten Community-Strings sind wirksam.
- Bei S7-1500 CPUs mit einer Firmware-Version < V3.0 sind bei aktiviertem SNMP immer die voreingestellten Community-Strings ("public" und "private") wirksam.

5.7.2 SNMP durch Datensatzübertragung aktivieren/deaktivieren: Beispiel für eine CPU 1516-3 PN/DP

Einleitung

Weil Sie Ihre Netzwerk-Infrastruktur, CPUs und IO-Devices mit SNMP verwalten möchten, wollen Sie SNMP für eine CPU 1516-3 PN/DP aktivieren. Im folgenden Beispiel wird dazu der Datensatz 0xB071 an eine PROFINET-Schnittstelle übertragen.

Voraussetzung

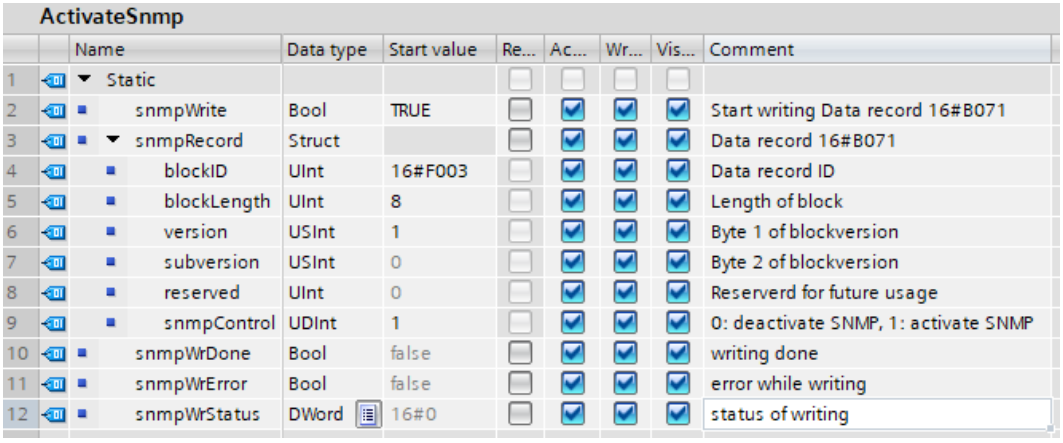
- CPU 1516-3 PN/DP ab FW-Stand V2.0
- STEP 7 ab Version V14

Lösung

Übertragen Sie den Datensatz 0xB071 zu einer PROFINET-Schnittstelle der CPU. Dadurch wird SNMP an allen PROFINET-Schnittstellen der CPU aktiviert.
Das folgende Beispiel zeigt Ihnen, wie Sie den Datensatz in einem Global-Datenbaustein anlegen und in einem Programmzyklus-OB (z. B. OB1) an die PROFINET-Schnittstelle (Local~PROFINET_interface_1) übertragen.
Um SNMP für die adressierte PROFINET-Schnittstelle der CPU 1516-3 PN/DP zu aktivieren, gehen Sie folgendermaßen vor:

1. Legen Sie einen Global-Datenbausein an.
2. Vergeben Sie einen Namen, z. B. "ActivateSnmp".

3. Legen Sie unter "Static" die Struktur des Datensatzes 0xB071 (im Bild: "snmpRecord") und weitere Variablen zur Übertragung des Datensatzes an. Das folgende Bild zeigt den Aufbau des Datenbausteins "ActivateSnmp".



ActivateSnmp

	Name	Data type	Start value	Re...	Ac...	Wr...	Vis...	Comment
1	Static			☐	☐	☐	☐	
2	snmpWrite	Bool	TRUE	☐	☑	☑	☑	Start writing Data record 16#B071
3	snmpRecord	Struct		☐	☑	☑	☑	Data record 16#B071
4	blockID	UInt	16#F003	☐	☑	☑	☑	Data record ID
5	blockLength	UInt	8	☐	☑	☑	☑	Length of block
6	version	USInt	1	☐	☑	☑	☑	Byte 1 of blockversion
7	subversion	USInt	0	☐	☑	☑	☑	Byte 2 of blockversion
8	reserved	UInt	0	☐	☑	☑	☑	Reserverd for future usage
9	snmpControl	UDInt	1	☐	☑	☑	☑	0: deactivate SNMP, 1: activate SNMP
10	snmpWrDone	Bool	false	☐	☑	☑	☑	writing done
11	snmpWrError	Bool	false	☐	☑	☑	☑	error while writing
12	snmpWrStatus	DWord	16#0	☐	☑	☑	☑	status of writing

Bild 5-36 Struktur des Global-Datenbausteins "ActivateSnmp"

4. Übertragen Sie den Datensatz 0xB071 in einem Programmzyklus-OB (z. B. OB1) mit der Anweisung "WRREC" (Datensatz Schreiben) an die CPU 1516-3 PN/DP. Ein Programmbeispiel finden Sie im nächsten Abschnitt.

Programmierbeispiel für die Datensatzübertragung im OB1

Im folgenden Programmcode wird der Datensatz 0xB071 übertragen:

```
//----------------------------------------
// Start writing SNMP settings
//----------------------------------------
IF "ActivateSnmp".snmpWrite THEN
 IF (NOT "ActivateSnmp".snmpWriteDone)
 AND (NOT "ActivateSnmp".snmpWriteError) THEN
 "instWrrec_1"(REQ := "ActivateSnmp".snmpWrite,
 ID := "Local~PROFINET-Schnittstelle_1",
 INDEX := 16#B071,
 DONE => "ActivateSnmp".snmpWriteDone,
 ERROR => "ActivateSnmp".snmpWriteError,
 STATUS => "ActivateSnmp".snmpWriteStatus,
 RECORD := "ActivateSnmp".snmpRecord);
 END_IF;
 IF "ActivateSnmp".snmpWriteError THEN
 ; // add error handling
 END_IF;
 IF "ActivateSnmp".snmpWriteDone THEN
 "ActivateSnmp".snmpWrite := FALSE;
 END_IF;
END_IF;
```

SNMP wieder deaktivieren

Mit kleinen Änderungen können Sie den oben verwendeten Programmcode zum Deaktivieren von SNMP verwenden. Weisen Sie im Anwenderprogramm der Variablen "ActivateSnmp".snmpRecord.snmpControl den Wert "0" zu:

```
"ActivateSnmp".snmpRecord.snmpControl := 0;
```

Beim nächsten Aufruf der Anweisung "WRREC" wird SNMP wieder deaktiviert.

5.7.3 SNMP durch Datensatzübertragung aktivieren/deaktivieren bei S7-1500R/H CPUs

Bei S7-1500R/H-Systemen aktivieren/deaktivieren Sie SNMP im Anwenderprogramm wie bei Standard-CPUs. Jedoch besitzen die PROFINET-Schnittstellen (X1, X2,...) beider CPUs unterschiedliche Hardware-Kennungen. Die PROFINET-Schnittstelle X1 der linken CPU z. B. hat eine andere Hardware-Kennung als die PROFINET-Schnittstelle X1 der rechten CPU. Das S7-1500R/H-System synchronisiert den SNMP-Status (aktiviert/deaktiviert) nicht automatisch für beide CPUs. Ein per Anweisung "WRREC" gesetzter SNMP-Status (aktiviert/deaktiviert) wirkt nur auf die CPU, deren PROFINET-Schnittstelle die Anweisung "WRREC" adressiert.

Beispiel:

Das S7-1500R/H System befindet sich im Systemzustand "RUN-Redundant". Wenn mit der Anweisung „WRREC“ z. B. eine PROFINET-Schnittstelle der linken CPU adressiert ist, wird der SNMP-Status der linken CPU geändert. Der SNMP-Status der rechten CPU bleibt unverändert. Wenn die linke CPU ausfällt oder getauscht wird, dann gilt nach dem SYNCUP wieder der ungeänderte SNMP-Status.

Abhilfe:

Rufen Sie die Anweisung "WRREC" 2 Mal auf. Adressieren Sie im ersten Aufruf von "WRREC" die Hardware-Kennung einer PROFINET-Schnittstelle der linken CPU. Führen Sie einen weiteren Aufruf der Anweisung "WRREC" aus. Adressieren Sie diesmal die Hardware-Kennung einer PROFINET-Schnittstelle der rechten CPU.

Hardware-Kennungen für PROFINET-Schnittstelle X1:

- Die PROFINET-Schnittstelle X1 der linken CPU hat die Hardware-Kennung 65164 (voreingestellter Name: Local1~PROFINET-interface_1).
- Die PROFINET-Schnittstelle X1 der rechten CPU hat die Hardware-Kennung 65364 (voreingestellter Name: Local2~PROFINET-interface_1).

Die Adressierung über die jeweiligen Hardware-Kennungen der PROFINET-Schnittstellen X1 wird auch in dem folgenden Beispiel zum Aufruf der Anweisung "WRREC" für beide R/H-CPUs verwendet.

HINWEIS

Übertragung des Datensatzes an die Backup-CPU

Übertragen Sie den Datensatz an die adressierte PROFINET-Schnittstelle der Backup-CPU erst dann, wenn das S7-1500R/H System den Systemzustand "Run-REDUNDANT" erreicht hat. Ansonsten kann der Datensatz an die adressierte PROFINET-Schnittstelle der Backup-CPU nicht übertragen werden.

Wenn das S7-1500R/H-System den Systemzustand "Run-REDUNDANT" erreicht hat, wird der CPU-Redundanzfehler-OB (OB72) gestartet. Die Variable "Fault_ID" des OB72 erhält den Fehlercode "B#16#03" oder "B#16#06".

Beispiel: WRREC-Aufrufe für beide R/H-CPUs

Um SNMP für die adressierte PROFINET-Schnittstelle beider CPUs durch das Übertragen von Datensätzen zu aktivieren/deaktivieren, gehen Sie folgendermaßen vor:

1. Legen Sie einen Global-Datenbaustein an.
2. Vergeben Sie einem Namen, z. B. "ActivateSnmp".
3. Legen Sie unter "Static" die Struktur des Datensatzes 0xB071 (im Bild: "snmpRecord") und weitere Variablen zur Übertragung des Datensatzes an. Das folgende Bild zeigt den Aufbau des Datenbausteins "ActivateSnmp".

ActivateSnmp

	Name	Data type	Start value	Re...	Ac...	Wr..	Vis...	Comment
1	Static			☐	☐	☐	☐	
2	snmpWrite	Bool	TRUE	☐	☑	☑	☑	Start writing Data record 16#B071
3	snmpRecord	Struct		☐	☑	☑	☑	Data record 16#B071
4	blockID	UInt	16#F003	☐	☑	☑	☑	Data record ID
5	blockLength	UInt	8	☐	☑	☑	☑	Length of block
6	version	USInt	1	☐	☑	☑	☑	Byte 1 of blockversion
7	subversion	USInt	0	☐	☑	☑	☑	Byte 2 of blockversion
8	reserved	UInt	0	☐	☑	☑	☑	Reserverd for future usage
9	snmpControl	UDInt	1	☐	☑	☑	☑	0: deactivate SNMP, 1: activate SNMP
10	plcLeft	Struct		☐	☑	☑	☑	Writing status of left plc
11	snmpWrDone	Bool	false	☐	☑	☑	☑	writing done
12	snmpWrError	Bool	false	☐	☑	☑	☑	error while writing
13	snmpWrStatus	DWord	16#0	☐	☑	☑	☑	status of writing
14	plcRight	Struct		☐	☑	☑	☑	Writing status of right plc
15	snmpWrDone	Bool	false	☐	☑	☑	☑	writing done
16	snmpWrError	Bool	false	☐	☑	☑	☑	error while writing
17	snmpWrStatus	DWord	16#0	☐	☑	☑	☑	status of writing

Bild 5-37 Struktur des Global-Datenbausteins "ActivateSnmp"

4. Fügen Sie Ihrem Anwenderprogramm den Organisationsbaustein "CPU redundancy error" (OB72) hinzu. Ein Programmbeispiel für den OB72 finden Sie im nächsten Abschnitt.
5. Öffnen Sie den Programmzyklus-OB (OB1).
6. Führen Sie im OB1 zwei "WRREC"-Anweisungen durch zur Übertragung des Datensatzes an die jeweils adressierte PROFINET-Schnittstelle beider CPUs. Ein Programmbeispiel für den OB1 finden Sie im nächsten Abschnitt.

   Ergebnis: Der Datensatz 0xB071 wurde an die jeweils adressierte PROFINET-Schnittstelle beider CPUs übertragen.

Programmierbeispiel für die Organisationsbausteine OB72 und OB1

Öffnen Sie den hinzugefügten OB72. Mit dem folgenden Programmcode ermitteln Sie, ob das R/H-System den Zustand "Run-REDUNDANT" eingenommen hat und setzen den Startbefehl für die "WRREC"-Anweisungen:

```
//-----------------------------------------
// Check redundancy state and set "snmpWrite"
//-----------------------------------------
IF #Fault_ID = B#16#03 OR #Fault_ID = B#16#06 THEN
 "ActivateSnmp".snmpWrite := TRUE;
END_IF;
```

Öffnen Sie den Programmzyklus-OB (OB1). Mit dem folgenden Programmcode führen Sie 2 "WRREC"-Anweisungen zur Übertragung des Datensatzes an die jeweils adressierte PROFINET-Schnittstelle beider CPUs durch:

```
//----------------------------------------
// Start writing SNMP settings
//----------------------------------------
IF "ActivateSnmp".snmpWrite THEN
 IF (NOT "ActivateSnmp".plcLeft.snmpWrDone)
 AND (NOT "ActivateSnmp".plcLeft.snmpWrError) THEN
 // write SNMP settings for the left PLC
 "instWrrec_1"(REQ := "ActivateSnmp".snmpWrite,
 ID := "Local1~PROFINET_interface_1",
 INDEX := 16#B071,
 DONE => "ActivateSnmp".plcLeft.snmpWrDone,
 ERROR => "ActivateSnmp".plcLeft.snmpWrError,
 STATUS => "ActivateSnmp".plcLeft.snmpWrStatus,
 RECORD := "ActivateSnmp".snmpRecord);
 END_IF;
 IF "ActivateSnmp".plcLeft.snmpWrError THEN
 ; // add error handling for left plc
 END_IF;
 IF (NOT "ActivateSnmp".plcRight.snmpWrDone)
 AND (NOT "ActivateSnmp".plcRight.snmpWrError) THEN
 // write SNMP settings for the right PLC
 "instWrrec_2"(REQ := "ActivateSnmp".snmpWrite,
 ID := "Local2~PROFINET_interface_1",
 INDEX := 16#B071,
 DONE => "ActivateSnmp".plcRight.snmpWrDone,
 ERROR => "ActivateSnmp".plcRight.snmpWrError,
 STATUS =>
 "ActivateSnmp".plcRight.snmpWrStatus,
 RECORD := "ActivateSnmp".snmpRecord);
 END_IF;
 IF "ActivateSnmp".plcRight.snmpWrError THEN
 ; // add error handling for right plc
 END_IF;
 IF "ActivateSnmp".plcLeft.snmpWrDone
 AND "ActivateSnmp".plcRight.snmpWrDone THEN
 "ActivateSnmp".snmpWrite := FALSE;
 END_IF;
END_IF;
```

SNMP wieder deaktivieren

Mit kleinen Änderungen können Sie den oben verwendeten Programmcode zum Deaktivieren von SNMP verwenden. Weisen Sie im Anwenderprogramm der Variablen "ActivateSnmp".snmpRecord.snmpControl den Wert "0" zu:

```
"ActivateSnmp".snmpRecord.snmpControl := 0;
```

Beim nächsten Aufruf der "WRREC"-Anweisungen wird SNMP wieder deaktiviert.

PG-Kommunikation

6

Eigenschaften

Über die PG-Kommunikation tauscht die CPU oder ein anderes kommunikationsfähiges Modul Daten mit einer Engineering Station (z. B. PG, PC) aus. Der Datenaustausch ist über PROFIBUS- und PROFINET-Subnetze möglich. Der Übergang zwischen S7-Subnetzen wird ebenfalls unterstützt.

Mit der PG-Kommunikation stehen Ihnen Funktionen zur Verfügung, die Sie zum Laden von Programmen und Konfigurationsdaten, zum Durchführen von Tests und zum Auswerten von Diagnoseinformationen benötigen. Diese Funktionen sind im Betriebssystem des kommunikationsfähigen Moduls integriert.

HINWEIS

Ab TIA Portal Version V17 wird das Protokoll TLS (Transport Layer Security) für die PG/HMI-Kommunikation unterstützt, um mittels standardisierter Security-Mechanismen den Datenaustausch zwischen PG/PC und CPU abzusichern.

Weitere Informationen finden Sie in folgenden Kapiteln:

- Voraussetzungen für Secure Communication (Seite 74)
- Secure PG/HMI-Kommunikation (Seite 103)

Voraussetzungen

- Das PG/PC ist physikalisch mit dem kommunikationsfähigen Modul verbunden.
- Wenn das kommunikationsfähige Modul über S7-Routing erreicht werden soll, muss die Hardware-Konfiguration in die beteiligten Stationen (S7-Router und Endpunkt) geladen sein.

Vorgehen zum online Verbinden

Für die PG-Kommunikation müssen Sie eine Online-Verbindung mit der CPU herstellen:

1. Markieren Sie in STEP 7 in der Projektnavigation die CPU.
2. Wählen Sie den Menübefehl "Online > Online verbinden".
3. Nehmen Sie im Dialog "Online verbinden" die folgenden Einstellungen für Ihre Online-Verbindung vor:
   - Wählen Sie in der Klappliste "Typ der PG/PC-Schnittstelle" den Schnittstellentyp (z. B. PN/IE)
   - Wählen Sie in der Klappliste "PG/PC-Schnittstelle" diejenige PG/PC-Schnittstelle (z. B. Ind. Ethernet-Karte), über die Sie die Online-Verbindung herstellen wollen.

- Wählen Sie in der Klappliste "Verbindung mit Schnittstelle/Subnetz" die Schnittstelle oder das S7-Subnetz aus, mit dem das Programmiergerät physikalisch verbunden ist.
- Falls das kommunikationsfähige Modul über einen S7-Router (Gateway) erreichbar ist, wählen Sie in der Klappliste "1. Gateway" denjenigen S7-Router aus, der die betroffenen Subnetze miteinander verbindet.

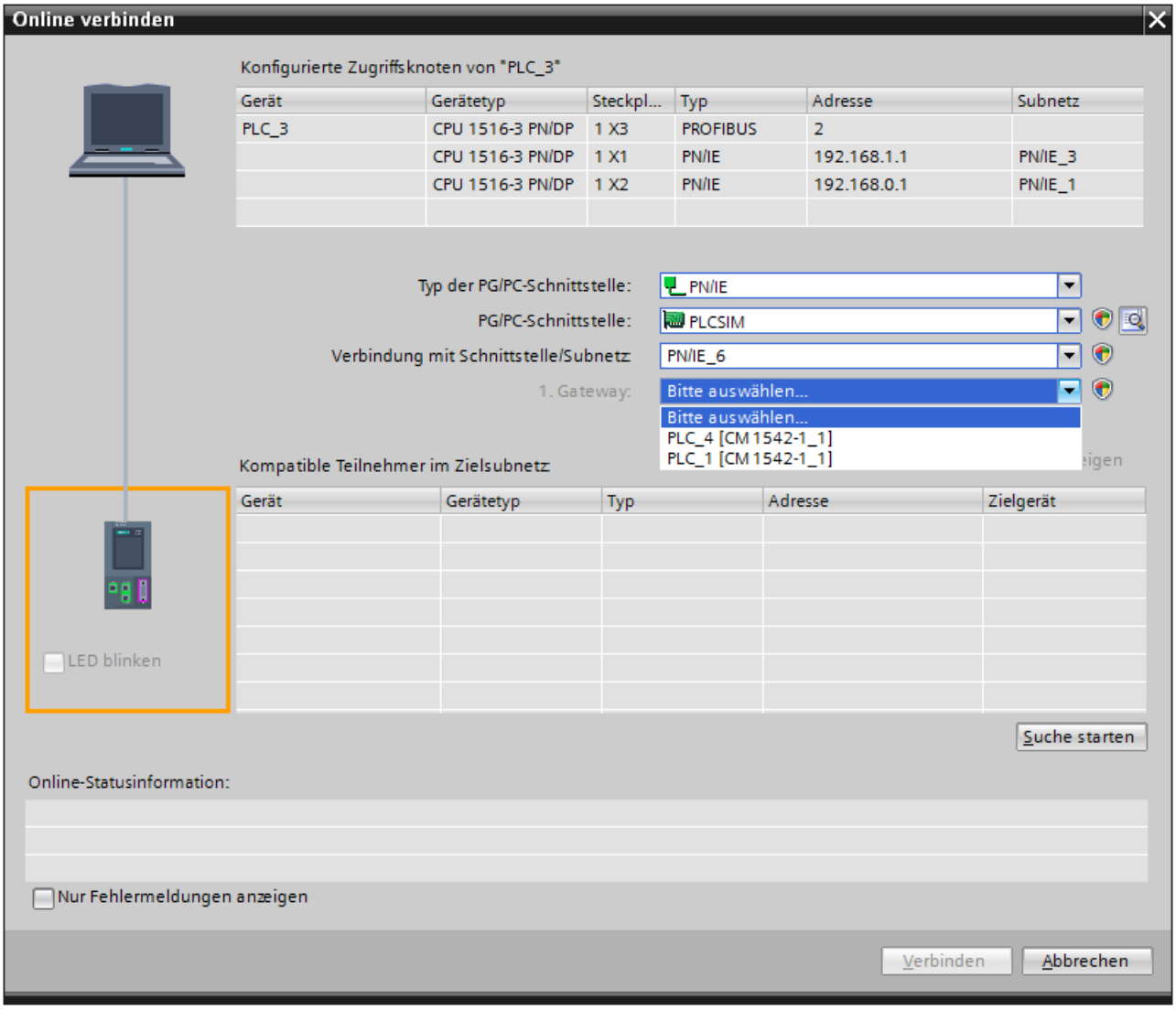


Bild 6-1 PG-Kommunikation einrichten

4. Klicken Sie auf "Suche starten".
   Nach kurzer Zeit erscheinen in der Tabelle "Kompatible Teilnehmer im Zielsubnetz" alle Geräte, die Sie mit PG-Kommunikation ansprechen können.
5. Wählen Sie in der Tabelle "Kompatible Teilnehmer im Zielsubnetz" die entsprechende CPU aus und bestätigen Sie mit "Verbinden".

Weitere Informationen

Weitere Informationen zum "Online verbinden" finden Sie in der Online-Hilfe STEP 7.

HMI-Kommunikation

7

Eigenschaften

Über die HMI-Kommunikation tauschen ein oder mehrere HMI-Geräte (z. B. HMI Basic/Comfort/Mobile Panel) Daten zum Bedienen und Beobachten mit einer CPU über die PROFINET- oder PROFIBUS DP-Schnittstelle aus. Der Datenaustausch erfolgt über HMI-Verbindungen.

Wenn Sie mehrere HMI-Verbindungen zu einer CPU einrichten möchten, verwenden Sie z. B.:

- die PROFINET- und PROFIBUS DP-Schnittstellen der CPU
- CPs und CMs mit den entsprechenden Schnittstellen

HINWEIS

Ab TIA Portal Version V17 wird das Protokoll TLS (Transport Layer Security) für die PG/HMI-Kommunikation unterstützt, um mittels standardisierter Security-Mechanismen den Datenaustausch zwischen PG/PC und CPU abzusichern.

Weitere Informationen finden Sie in folgenden Kapiteln:

- Voraussetzungen für Secure Communication (Seite 74)
- Secure PG/HMI-Kommunikation (Seite 103)

Vorgehen zum Einrichten von HMI-Kommunikation

Sobald Sie eine Variable, z. B. eine Variable aus einem globalen Datenbaustein, per Drag & Drop in ein HMI-Bild oder in die HMI-Variablentabelle hineinziehen, richtet STEP 7 automatisch eine HMI-Verbindung ein. Alternativ können Sie die HMI-Verbindung auch selbst einrichten.

Um eine HMI-Verbindung einzurichten, gehen Sie folgendermaßen vor.

1. Konfigurieren Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 das HMI-Gerät in einer vorhandenen Konfiguration mit der CPU.
2. Wählen Sie die Schaltfläche "Verbindungen" und aus der Klappliste "HMI-Verbindung".
3. Ziehen Sie per Drag & Drop eine Linie zwischen den Endpunkten der Verbindung (HMI-Gerät und CPU). Die Endpunkte sind farblich hervorgehoben. Falls noch kein zugehöriges S7-Subnetz vorhanden ist, wird dieses automatisch angelegt.

4. Wählen Sie im Register "Verbindungen" die Zeile der HMI-Verbindung.
   Im Bereich "Allgemein", im Register "Eigenschaften" sehen Sie die Eigenschaften der HMI-Verbindung, die Sie z. T. ändern können.

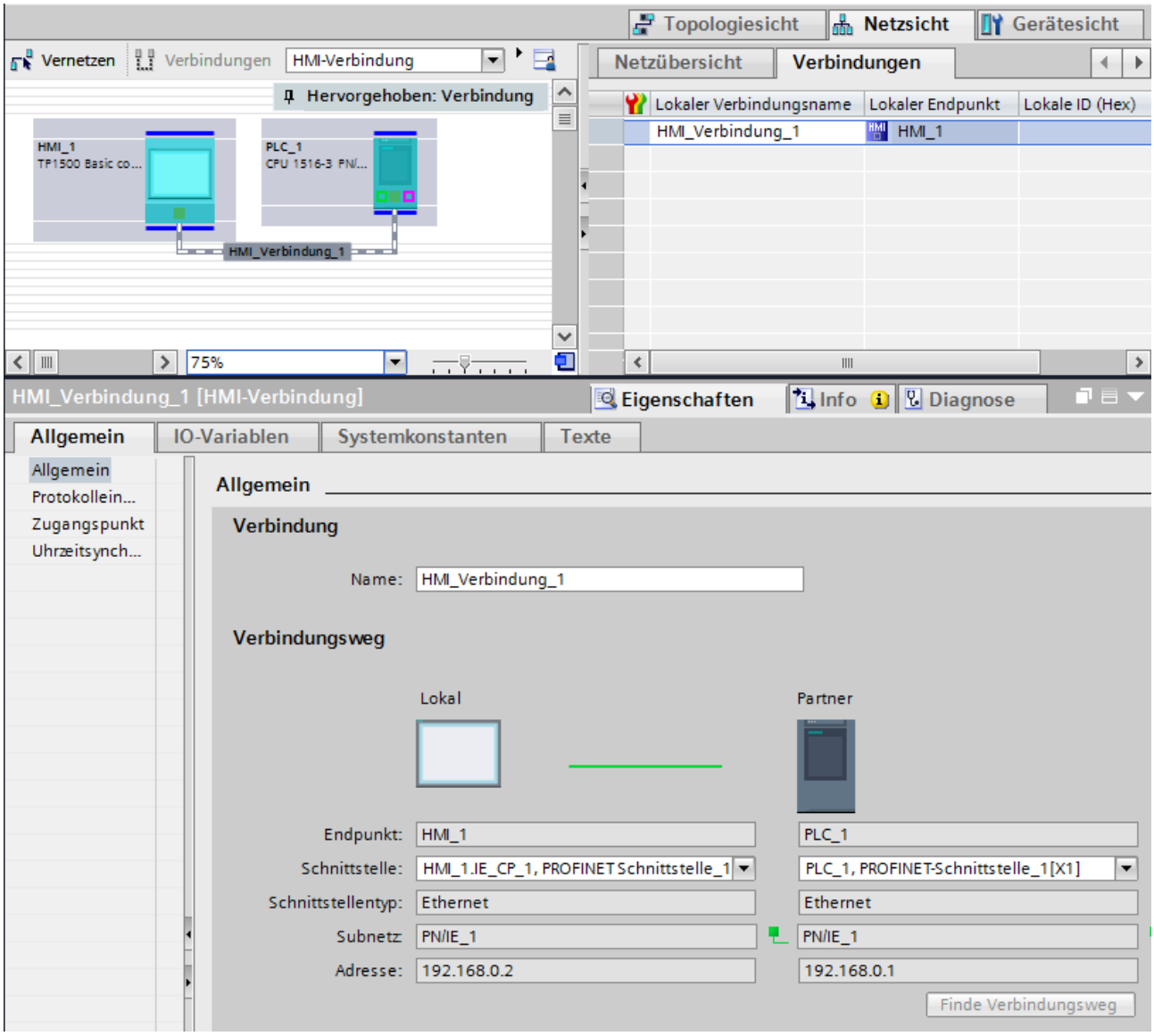


Bild 7-1 HMI-Kommunikation einrichten

5. Laden Sie die Hardware-Konfiguration in die CPU.
6. Laden Sie die Hardware-Konfiguration in das HMI-Gerät.

Weitere Informationen

Informationen zum S7-Routing für HMI-Verbindungen finden Sie im Kapitel S7-Routing (Seite 380).
Weitere Informationen zum Einrichten von HMI-Verbindungen finden Sie in der Online-Hilfe STEP 7.

Open User Communication 8

8.1 Open User Communication im Überblick

Merkmale von Open User Communication

Über Open User Communication, auch "Offene Kommunikation" genannt, tauscht die CPU Daten mit einem weiteren, kommunikationsfähigen Gerät aus. Die Open User Communication zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikationspartner können zwei SIMATIC CPUs oder eine SIMATIC CPU und ein geeignetes Fremdgerät sein)
- Kommunikation über unterschiedliche Protokolle (in STEP 7 als "Verbindungstypen" bezeichnet)
- Hohe Flexibilität hinsichtlich der zu übertragenden Datenstrukturen; ermöglicht damit offenen Datenaustausch mit beliebigen Kommunikationsteilnehmern, sofern diese die zur Verfügung gestellten Verbindungstypen unterstützen
- Secure Communication: Zum Schutz Ihres Automatisierungssystems können Sie Daten über Open User Communication gesichert austauschen. Bei der Secure Open User Communication werden die Daten signiert und verschlüsselt gesendet, siehe auch Secure Open User Communication (Seite 86).
- Open User Communication ist in verschiedenen Automatisierungssystemen möglich, siehe technische Daten der jeweiligen Gerätehandbücher.
  Beispiele:
  - Integrierte PROFINET/Ind. Ethernet-Schnittstellen von CPUs (S7-1500, ET 200SP CPU, S7-1500 Software Controller, CPUs 1513/1516pro-2 PN)
  - PROFINET/Ind. Ethernet-Schnittstellen von Kommunikationsmodulen (z. B. CP 1543-1, CM 1542-1, CP 1543SP-1)

Informationen zu Secure Communication finden Sie im Kapitel Secure Communication (Seite 50).

Informationen zu S7-1500R/H

Informationen zur Open User Communication mit dem redundanten System S7-1500R/H finden Sie im Kapitel Kommunikation mit dem redundanten System S7-1500R/H (Seite 411).

8.2 Protokolle für Open User Communication

Protokolle für Open User Communication

Für die offene Kommunikation stehen folgende Protokolle zur Verfügung:

Tabelle 8-1 Transportprotokolle für offene Kommunikation

Transportprotokoll	Über Schnittstelle
TCP gemäß RFC 793	PROFINET/Industrial Ethernet
ISO-on-TCP gemäß RFC 1006 (Class 4)	PROFINET/Industrial Ethernet
ISO gemäß ISO/IEC 8073	Industrial Ethernet (nur CP 1543-1)
UDP gemäß RFC 768	PROFINET/Industrial Ethernet
FDL	PROFIBUS

Tabelle 8-2 Applikationsprotokolle für offene Kommunikation

Applikationsprotokoll	Genutztes Transportprotokoll
Modbus TCP	**TCP** gemäß RFC 793
E-Mail	**TCP** gemäß RFC 793
FTP	**TCP** gemäß RFC 793

TCP, ISO-on-TCP, ISO, UDP

Diese Protokolle (außer UDP) bauen vor der Datenübertragung eine Transportverbindung zum Kommunikationspartner auf. Verbindungsorientierte Protokolle werden eingesetzt, wenn es bei der Datenübertragung besonders auf Sicherheit vor Datenverlust ankommt.

Bei UDP ist möglich:

- Unicast an einen oder Broadcast an alle Teilnehmer am PROFINET über die PROFINET-Schnittstelle der CPU oder die Industrial Ethernet-Schnittstelle des CP 1543-1
- Multicast an alle Empfänger eines Multicast-Kreises über die PROFINET-Schnittstelle der CPU oder die PROFINET/Industrial Ethernet-Schnittstelle des CP 1543-1

Maximale Anzahl unterstützter Multicast-Kreise und maximale Nutzdatenlängen: Siehe Technische Daten der jeweiligen Gerätehandbücher.

Protokoll zur Kommunikation via PROFIBUS: FDL

Die Datenübertragung über eine FDL-Verbindung (Fieldbus Data Link) ist geeignet für die Übertragung zusammenhängender Datenblöcke zu einem Kommunikationspartner am PROFIBUS, der das Senden bzw. Empfangen entsprechend des FDL-Dienstes SDA (Send Data with Acknowledge) nach EN 50170, Vol 2. unterstützt. Beide Partner sind gleichberechtigt, d. h. jeder Partner kann ereignisabhängig den Sende- und Empfangsvorgang anstoßen.

Entsprechend des FDL-Dienstes SDN (Send Data with No Acknowledge) nach EN 50170, Vol 2. sind bei FDL möglich:

- Broadcast an alle Teilnehmer am PROFIBUS über die PROFIBUS-Schnittstelle des CM 1542-5
- Multicast an alle Empfänger eines Multicast-Kreises über die PROFIBUS-Schnittstelle des CM 1542-5

Modbus TCP

Das Modbus-Protokoll ist ein Kommunikationsprotokoll mit Linientopologie auf Basis einer Master/Slave-Architektur. In der Übertragungsart Modbus TCP (Transmission Control Protocol) werden die Daten als TCP/IP-Pakete übertragen.

Die Kommunikation wird ausschließlich über entsprechende Anweisungen im Anwenderprogramm gesteuert.

E-Mail und FTP

Über E-Mail ist z. B. das Versenden von Datenbausteininhalten (z. B. Prozessdaten) als Anhang möglich.

Die FTP-Verbindung (FTP = File-Transfer-Funktionen) verwenden Sie für die Übertragung von Dateien zu und von den S7-Geräten.

Client-seitig wird die Kommunikation über entsprechende Anweisungen im Anwenderprogramm gesteuert.

Anwendungsbeispiel: MQTT Publisher für die SIMATIC S7-1500 CPU

Das "Message Queue Telemetry Transport" (MQTT) ist ein einfaches Protokoll auf TCP/IP-Ebene. Es eignet sich für den Nachrichtenaustausch zwischen Geräten mit geringer Funktionalität und für die Übertragung über unzuverlässige Netze.

Das Anwendungsbeispiel stellt Ihnen einen Funktionsbaustein zur Verfügung, mit dem Sie das MQTT-Protokoll in die SIMATIC S7-1500 implementieren können.

Das Anwendungsbeispiel finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/109772284).

Bausteinbibliothek für SYSLOG-Meldungen

Syslog ist ein einfach aufgebautes binäres Protokoll auf UDP/IP-Ebene. Es ermöglicht Anwendungen Meldungen, Warnhinweise oder Fehlerzustände an einen Syslog-Server zu senden. Syslog wird typischerweise für Computersystem-Management und Sicherheits-Überwachung benutzt und hat sich inzwischen als ein Standard im Bereich der Protokollierung etabliert.

Um das Syslog-Protokoll in eine S7-1500 zu implementieren, wird Ihnen mit der Bibliothek "LSyslog" eine Lösung angeboten. Zusätzlich zur Bibliothek wird Ihnen ein Anwendungsbeispiel zur Verfügung gestellt, das Ihnen zeigt, wie Sie Syslog-Meldungen in Ihrer Steuerung generieren und an den Syslog-Server senden können.

Die Bausteinbibliothek "LSyslog" und das dazugehörige Anwendungsbeispiel finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/51929235).

8.3 Anweisungen für Open User Communication

Einleitung

Sie richten die Open User Communication über die entsprechende Verbindung (z. B. TCP-Verbindung) wie folgt ein:

- durch Programmieren in den Anwenderprogrammen der Kommunikationspartner oder
- durch Projektieren der Verbindung in STEP 7 im Hardware- und Netzwerkeditor

Unabhängig vom Einrichten der Verbindung durch Programmierung oder Projektierung sind in den Anwenderprogrammen beider Kommunikationspartner immer Anweisungen zum Senden und Empfangen der Daten notwendig.

Einrichten der Verbindung über das Anwenderprogramm

Beim programmierten Einrichten der Verbindung wird der Verbindungsauf- und -abbau über Anweisungen im Anwenderprogramm realisiert.

In bestimmten Anwendungsbereichen ist es vorteilhaft, die Kommunikationsverbindungen nicht statisch mittels Projektierung in der Hardware-Konfiguration einzurichten, sondern über das Anwenderprogramm. Sie können die Verbindungen über eine spezifische Applikation programmgesteuert und damit bei Bedarf einrichten. Das programmierte Einrichten ermöglicht außerdem die Freigabe von Verbindungsressourcen nach der Datenübertragung.

Für jede Kommunikationsverbindung ist eine Datenstruktur notwendig, die die Parameter für den Aufbau der Verbindung enthält (z. B. Systemdatentyp "TCON_IP_v4" für TCP).

Die Systemdatentypen (SDT) werden vom System zur Verfügung gestellt und haben eine vordefinierte Struktur, die nicht änderbar ist.

Die verschiedenen Protokolle haben jeweils eigene Datenstrukturen (siehe folgende Tabelle). Die Parameter werden in einem Datenbaustein ("Verbindungsbeschreibungs-DB") z. B. des Systemdatentyps TCON_IP_v4 gespeichert.

Sie haben zwei Möglichkeiten, den DB mit der Datenstruktur vorzugeben:

- Empfehlung: Datenbaustein bei der Parametrierung der Verbindung in den Eigenschaften im Programmeditor automatisch anlegen lassen, mit Hilfe der Verbindungsparametrierung bei den Anweisungen TSEND_C, TRCV_C und TCON
- Datenbaustein manuell erstellen, parametrieren und direkt an die Anweisung schreiben Notwendig für:
  - Secure OUC
  - Verbindung über DNS
  - E-Mail
  - FTP

Im "Verbindungsbeschreibungs-DB" können Sie die Verbindungsparameter modifizieren.
Wie Sie die Anweisung TCON programmieren, um zwischen zwei S7-1500 CPUs eine Verbindung für die Open User Communication einzurichten, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/58875807).

Protokolle, Systemdatentypen und einsetzbare Anweisungen für programmiertes Einrichten

Die folgende Tabelle zeigt Ihnen die Protokolle der Open User Communication und die dazu passenden Systemdatentypen und Anweisungen.

Tabelle 8-3 Anweisungen bei programmiertem Einrichten der Verbindung

Protokoll	Systemdatentyp	Anweisungen
TCP	• TCON_QDN • TCON_IP_v4	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TCON, TSEND/TRCV oder • TCON, TUSEND/TURCV (Abbau der Verbindung über TDISCON möglich)
ISO-on-TCP	• TCON_IP_RFC	
ISO gemäß ISO/IEC 8073 (Class 4)	• TCON_ISOnative[1] • TCON_Configured	
UDP	• TCON_IP_v4 • TADDR_Param • TADDR_SEND_QDN • TADDR_RCV_IP	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C • TUSEND/TURCV/TRCV (Abbau der Verbindung über TDISCON möglich)
FDL[1]	• TCON_FDL	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TCON, TSEND/TRCV oder • TCON, TUSEND/TURCV (Abbau der Verbindung über TDISCON möglich)

[1] dieses Protokoll ist nur über das CM 1542-5 verwendbar

[2] dieses Protokoll ist nur über den CP 1543-1 verwendbar

[3] anwenderdefinierter Datentyp

Protokoll	Systemdatentyp	Anweisungen
Modbus TCP	• TCON_IP_v4 • TCON_QDN • TCON_Configured	• MB_CLIENT • MB_SERVER
E-Mail	• TMAIL_v4 • TMAIL_v6 • TMAIL_FQDN	• TMAIL_C
FTP[2]	• FTP_CONNECT_IPV4[3] • FTP_CONNECT_IPV6[3] • FTP_CONNECT_NAME[3]	• FTP_CMD

[1] dieses Protokoll ist nur über das CM 1542-5 verwendbar

[2] dieses Protokoll ist nur über den CP 1543-1 verwendbar

[3] anwenderdefinierter Datentyp

Die folgende Tabelle zeigt Ihnen die verschiedenen Verbindungen der Secure Open User Communication und die dazu passenden Systemdatentypen und Anweisungen.

Secure OUC-Verbindung	Systemdatentyp	Anweisungen
Gesicherte TCP-Verbindung von einer S7-1500 CPU als TLS-Client zu einem Fremd-PLC (TLS-Server) Gesicherte TCP-Verbindung von einer S7-1500 CPU als TLS-Server zu einem Fremd-PLC (TLS-Client)	• TCON_QDN_SEC	• TSEND_C/TRCV_C • TCON, TSEND/TRCV
Gesicherte TCP-Verbindung zwischen zwei S7-1500 Stationen	• TCON_IP_V4_SEC[1]	
Gesicherte Verbindung zu einem Mailserver[2]	• TMAIL_V4_SEC • TMAIL_QDN_SEC	• TMAIL_C (ab V5.0)
Gesicherte Modbus TCP-Verbindung	• TCON_IP_V4_SEC[1]	• MB_Client • MB_Server
	• TCON_QDN_SEC	

[1] Auch über CP 1543-1 möglich

[2] Gesicherte Verbindung zu einem Mailserver auch möglich mit CP1543-1 und TMAIL_C (V4.0)

Einrichten der Verbindung über Verbindungsprojektierung

Beim Einrichten über die Verbindungsprojektierung werden die Adressparameter der Verbindung im Hardware- und Netzwerkeditor von STEP 7 festgelegt.

Für das Senden und Empfangen der Daten nutzen Sie die gleichen Anweisungen, wie beim programmierten Einrichten von Verbindungen:

Tabelle 8-4 Anweisungen zum Senden/Empfangen bei projektierten Verbindungen

Protokoll	Senden/Empfangen bei projektierten Verbindungen
Einsetzbare Anweisungen:	
TCP	Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TSEND/TRCV oder • TUSEND/TURCV
ISO-on-TCP	
ISO gemäß ISO/IEC 8073 (Class 4)	

Protokoll	Senden/Empfangen bei projektierten Verbindungen
UDP	Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TUSEND/TURCV
FDL	Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TSEND/TRCV oder • TUSEND/TURCV
Modbus TCP	Nicht unterstützt
E-Mail	Nicht unterstützt
FTP	Nicht unterstützt

Weitere Anweisungen für offene Kommunikation

Die folgenden Anweisungen können Sie sowohl bei im Anwenderprogramm eingerichteten Verbindungen, als auch bei projektierten Verbindungen einsetzen:

- T_RESET: Verbindung abbauen und aufbauen
- T_DIAG: Verbindung überprüfen

Basisbeispiele für Open User Communication

Der Siemens Online Support bietet Ihnen Funktionsbausteine (FBs) an, die Ihnen die Handhabung der Anweisungen der Open User Communication erleichtern. Die Funktionsbausteine mit zugehörigen Beispielen finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/109747710).

Weitere Informationen

In der Online-Hilfe STEP 7 finden Sie beschrieben:

- die Anwender- und Systemdatentypen
- die Anweisungen für offene Kommunikation
- die Verbindungsparameter

Informationen zur Belegung und Freigabe von Verbindungsressourcen finden Sie im Kapitel Belegung von Verbindungsressourcen (Seite 400).

Informationen zu Secure Open User Communication finden Sie im Kapitel Secure Open User Communication (Seite 86).

8.4 Open User Communication mit Adressierung über Domainnamen

S7-1500 CPUs, ET 200SP CPUs und die CPUs 1513/1516pro-2 PN unterstützen ab Firmwarestand V2.0 die Open User Communication mit Adressierung über ein Domain Name System (DNS). In der CPU ist ein DNS-Client integriert. Bei der Kommunikation über DNS verwenden Sie Domainnamen als Alias für IP-Adressen zur Adressierung von Kommunikationspartnern. Die Adressierung der Kommunikationspartner über Domainnamen ist für offene Kommunikation über TCP und UDP möglich.

Als Voraussetzung für die Kommunikation über DNS muss sich in Ihrem Netz mindestens ein DNS-Server befinden.

Der S7-1500 Software Controller unterstützt Kommunikation über DNS für alle Schnittstellen, die dem Software-Controller zugeordnet sind.

Kommunikation über DNS einrichten

Damit eine CPU eine Verbindung zu einem Kommunikationspartner über dessen Domainnamen aufbauen kann, muss der DNS-Client der CPU die IPv4-Adresse von mindestens einem DNS-Server kennen. Die CPU unterstützt bis zu 4 verschiedene DNS-Server.

Um die Kommunikation über Domainnamen für eine S7-1500 CPU einzurichten, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU in der Netzsicht von STEP 7.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "Erweiterte Konfiguration" > "DNS-Konfiguration".
3. Tragen Sie in der Tabelle "Serverliste" in der Spalte "DNS-Serveradressen" die IPv4-Adresse von einem DNS-Server ein.
   Sie können bis zu 4 IPv4-Adressen von DNS-Servern eintragen.

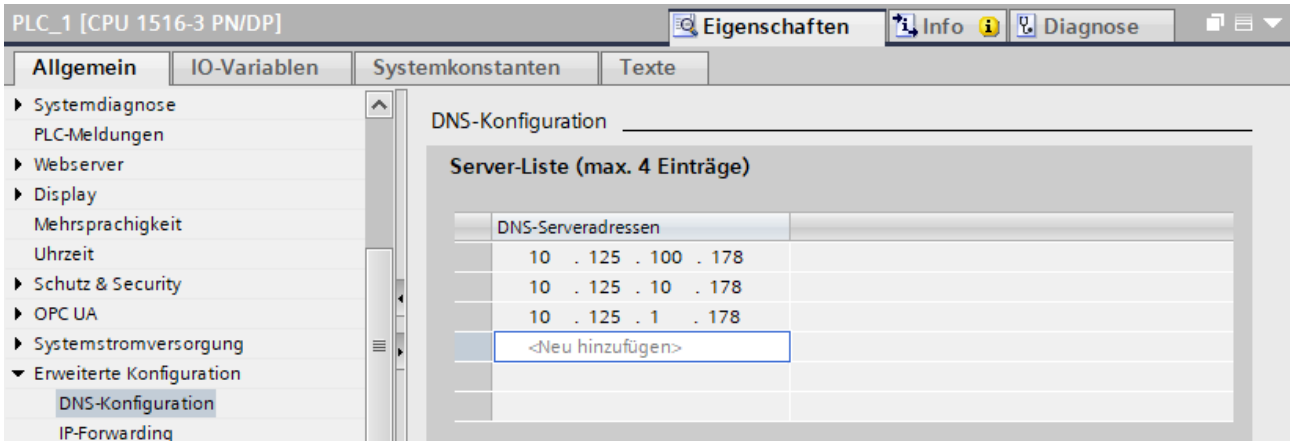


Bild 8-1 DNS-Serveradressen eintragen am Beispiel einer CPU 1516-3 PN/DP

TCP-Verbindung über den Domainnamen des Kommunikationspartners einrichten

Für die TCP-Kommunikation über den Domainnamen müssen Sie selbst einen Datenbaustein mit dem Systemdatentyp TCON_QDN erstellen, parametrieren und direkt an der Anweisung aufrufen. Die Anweisungen TCON, TSEND_C und TRCV_C unterstützen den Systemdatentyp TCON_QDN:
Um eine TCP-Verbindung über den Domainnamen des Kommunikationspartners einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_QDN.
   Das folgende Beispiel zeigt den globalen Datenbaustein "Data_block_1", in dem die Variable "DNS Connection1" vom Datentyp TCON_QDN definiert ist.

Data_block_1

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	DNS Connection1	TCON_QDN		
3	InterfaceId	HW_ANY	0	not relevant
4	ID	CONN_OUC	16#0	connection reference / identifier
5	ConnectionType	Byte	16#0B	type of connection: 16#0B=11=TCP/IP, 16#13=19=UDP
6	ActiveEstablished	Bool	false	active/passive connection establishment
7	RemoteQDN	String[254]	''	fully or partially qualified domain name of remote partner
8	RemotePort	UInt	0	remote UDP / TCP port number
9	LocalPort	UInt	0	local UDP / TCP port number

Bild 8-2 Datentyp TCON_QDN

3. Programmieren Sie die Parameter der TCP-Verbindung (z. B. den vollqualifizierten Domainnamen (FQDN)) in der Variablen vom Datentyp TCON_QDN.
4. Legen Sie im Programmeditor eine Anweisung TCON an.
5. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variable vom Datentyp TCON_QDN.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "DNS connection1" (Datentyp TCON_QDN) verschaltet.

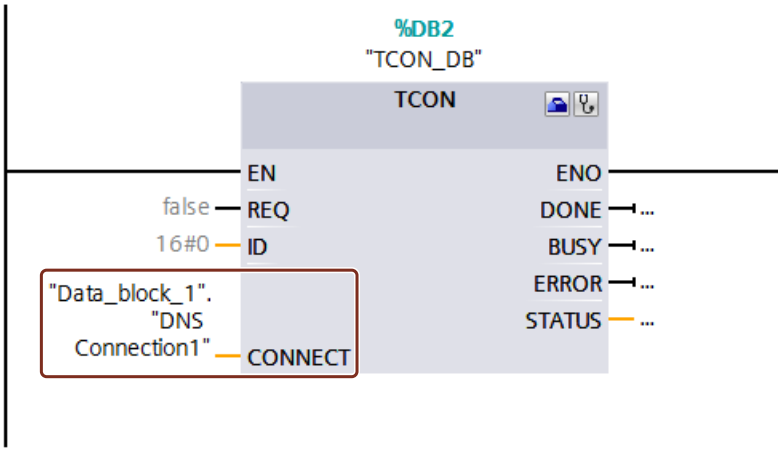


Bild 8-3 Anweisung TCON

UDP-Verbindung über den Domainnamen des Kommunikationspartners adressieren

Beim Senden von Daten über UDP können Sie für S7-1500-CPUs ab Firmware-Version V2.0 den Empfänger mit seinem voll qualifizierten Domainnamen (FQDN) adressieren. Dabei verweisen Sie bei der Anweisung TUSEND am Parameter ADDR auf eine Struktur vom Typ TADDR_SEND_QDN.
Der Empfänger kann eine IPv4- oder eine IPv6-Adresse zurückliefern. Verweisen Sie bei der Anweisung TURCV am Parameter ADDR daher auf eine Struktur vom Typ TADDR_RCV_IP. Nur diese kann beide IP-Adresstypen aufnehmen.

HINWEIS

Netzlast

Im Gegensatz zu TCP arbeitet das Protokoll UDP nicht verbindungsorientiert. Bei jeder Flanke am Bausteinparameter REQ führt die Anweisung TUSEND bzw. TURCV Abfrage des DNS-Servers durch. Dies kann zu hoher Netzwerklast bzw. Last auf dem DNS-Server führen.

Weitere Information

Weitere Informationen zu den Systemdatentypen TCON_QDN, TADDR_SEND_QDN und TADDR_RCV_IP finden Sie in der Onlinehilfe zu STEP 7.
Wie Sie eine gesicherte TCP-Verbindung über den Domainnamen des Kommunikationspartners einrichten, finden Sie im Kapitel Secure Open User Communication (Seite 86).

8.5 Open User Communication über TCP, ISO-on-TCP, UDP und ISO einrichten

Verbindung für die Anweisungen TSEND_C, TRCV_C oder TCON parametrieren

Voraussetzung: Im Programmiereditor ist eine Anweisung TSEND_C, TRCV_C oder TCON angelegt.

1. Selektieren Sie im Programmiereditor einen Baustein der Open User Communication TCON, TSEND_C oder TRCV_C.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften > Konfiguration".
3. Selektieren Sie die Gruppe "Verbindungsparameter". Solange Sie noch keinen Verbindungspartner selektiert haben, ist nur die leere Klappliste für den Partner-Endpunkt aktiv. Alle anderen Eingabemöglichkeiten sind deaktiviert.
   Es werden die bereits bekannten Verbindungsparameter angezeigt:
   - Name des lokalen Endpunkts
   - Schnittstelle des lokalen Endpunkts

– IPv4-Adresse des lokalen Endpunkts

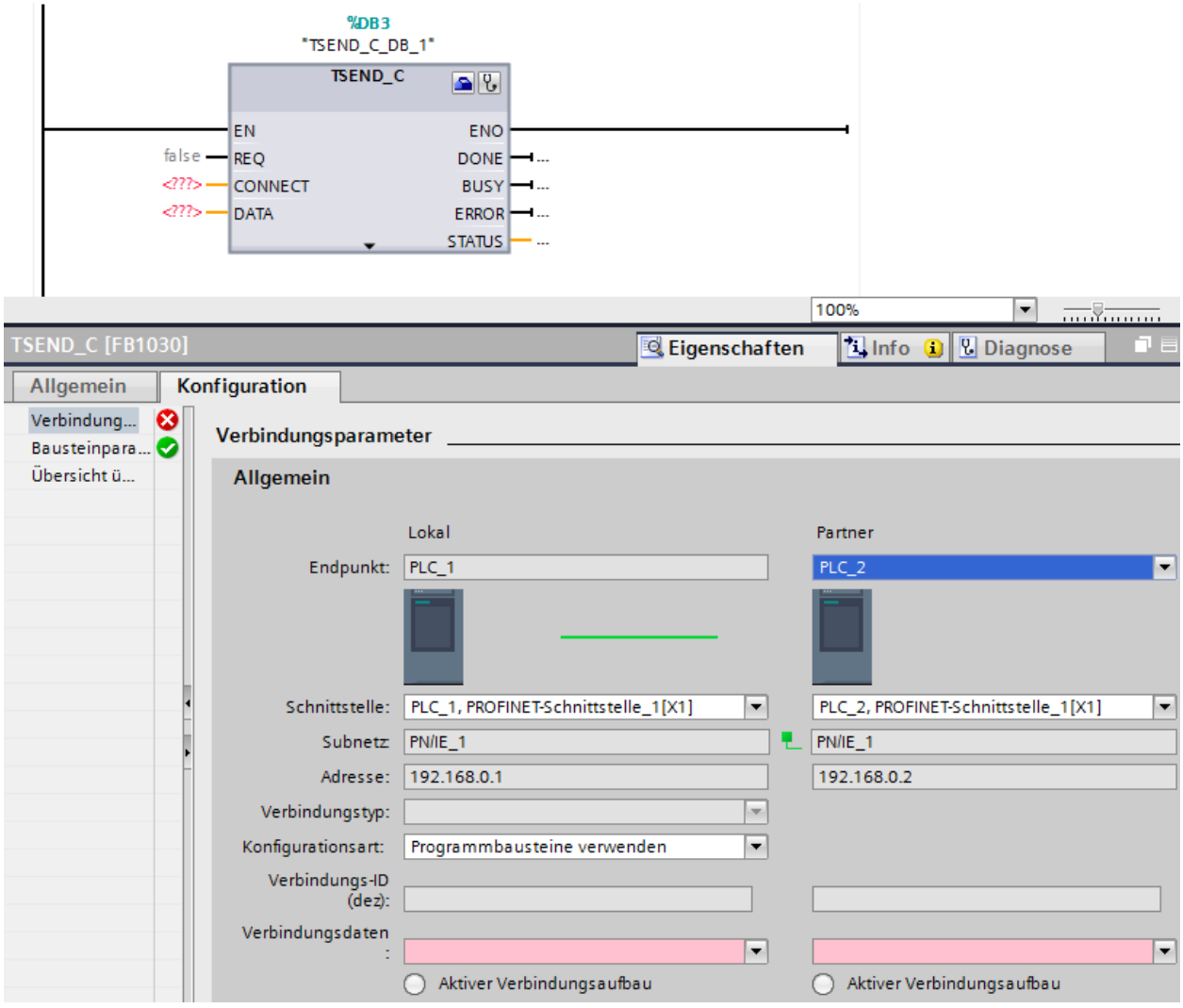


Bild 8-4 Verbindungsparametrierung für TSEND_C

4. Wählen Sie in der Klappliste des Partner-Endpunkts einen Verbindungspartner. Als Kommunikationspartner kommt ein unspezifiziertes Gerät oder eine im Projekt vorhandene CPU in Frage. Bestimmte Verbindungsparameter werden danach als Vorgabe automatisch eingetragen.

   Die folgenden Parameter werden eingestellt:

   – Name des Partner-Endpunkts
   – Schnittstelle des Partner-Endpunkts
   – IPv4-Adresse des Partner-Endpunkts

   Wenn die Verbindungspartner vernetzt sind, wird der Name des Subnetzes angezeigt.
5. Wählen Sie in der Klappliste "Konfigurationsart" zwischen der Verwendung von Programmbausteinen oder konfigurierten Verbindungen.

6. Wählen Sie in der Klappliste "Verbindungsdaten" einen vorhandene Verbindungsbeschreibungs-DBs oder bei konfigurierten Verbindungen unter "Verbindungsname" eine vorhandene Verbindung. Sie können auch einen neue Verbindungsbeschreibungs-DBs oder eine neue konfigurierte Verbindung anlegen. Sie können später noch andere Verbindungsbeschreibungs-DBs oder konfigurierte Verbindungen wählen oder die Namen der Verbindungsbeschreibungs-DBs ändern, um neue Datenbausteine zu erstellen:
   - Den ausgewählten Datenbaustein sehen Sie auch an der Beschaltung des Eingangsparameters CONNECT der ausgewählten Anweisung TCON, TSEND_C oder TRCV_C.
   - Wenn Sie für den Verbindungspartner bereits einen Verbindungsbeschreibungs-DB über den Parameter CONNECT der Anweisung TCON, TSEND_C oder TRCV_C angegeben haben, können Sie entweder diesen DB verwenden oder einen neuen DB anlegen.
   - Wenn Sie den Namen des angezeigten Datenbausteins in der Klappliste bearbeiten, wird automatisch ein neuer Datenbaustein mit dem geänderten Namen, aber derselben Struktur und demselben Inhalt generiert und für die Verbindung verwendet.
   - Geänderte Namen eines Datenbausteins müssen im Kontext des Kommunikationspartners eindeutig sein.
   - Ein Verbindungsbeschreibungs-DB muss je nach CPU-Typ und Verbindung die Struktur TCON_Param, TCON_IP_v4 oder TCON_IP_RFC haben.
   - Ein Datenbaustein kann nicht für einen unspezifizierten Partner ausgewählt werden.

   Nach Auswahl oder Anlegen des Verbindungsbeschreibungs-DBs oder der konfigurierten Verbindung werden weitere Werte ermittelt und eingetragen.

   Für spezifizierte Verbindungspartner gilt:
   - Verbindungstyp ISO-on-TCP
   - Verbindungs-ID mit dem Vorgabewert 1
   - Aktiver Verbindungaufbau vom lokalen Partner
   - TSAP-ID
     für S7-1200/1500: E0.01.49.53.4F.6F.6E.54.43.50.2D.31

   Für unspezifizierte Verbindungspartner gilt:
   - Verbindungstyp TCP
   - Partnerport 2000

   Bei konfigurierter Verbindung mit spezifiziertem Verbindungspartner gilt:
   - Verbindungstyp TCP
   - Verbindungs-ID mit dem Vorgabewert 257
   - Aktiver Verbindungaufbau vom lokalen Partner
   - Partnerport 2000

   Bei konfigurierter Verbindung mit unspezifiziertem Verbindungspartner gilt:
   - Verbindungstyp TCP
   - Lokaler Port 2000

7. Geben Sie ggf. eine Verbindungs-ID für den Verbindungspartner an. Für einen unspezifizierten Partner kann keine Verbindungs-ID vergeben werden.

   HINWEIS

   Sie müssen bei einem bekannten Verbindungspartner einen eindeutigen Wert für die Verbindungs-ID eingeben. Die Eindeutigkeit der Verbindungs-ID wird nicht durch die Verbindungsparametrierung geprüft und es wird bei Anlegen einer neuen Verbindung kein Vorgabewert für die Verbindungs-ID eingetragen.

8. Wählen Sie den gewünschten Verbindungstyp aus der entsprechenden Klappliste. Die Adressdetails werden abhängig vom Verbindungstyp mit Werten vorbelegt. Sie haben die Wahl zwischen:
   - TCP
   - ISO-on-TCP
   - UDP
   - ISO (nur bei Konfigurationsart "Konfigurierte Verbindung verwenden")

   Sie können die Eingabefelder in den Adressdetails bearbeiten. Je nach eingestelltem Protokoll können Sie die Ports (für TCP und UDP) oder die TSAPs (für ISO-on-TCP und ISO) bearbeiten.
9. Stellen Sie bei TCP, ISO und ISO-on-TCP das Verhalten für den Verbindungsaufbau über die Optionsfelder "Aktiver Verbindungsaufbau" ein. Sie können auswählen, welcher Kommunikationspartner die Verbindung aktiv aufbauen soll.

Geänderte Werte werden von der Verbindungsparametrierung sofort auf Eingabefehler geprüft und in den Datenbaustein für die Verbindungsbeschreibung eingetragen.

HINWEIS

Die Open User Communication zwischen zwei Kommunikationspartnern ist erst dann lauffähig, wenn auch der Programmteil für den Partner-Endpunkt in die Hardware geladen wurde. Achten Sie darauf, dass Sie für eine funktionierende Kommunikation nicht nur die Verbindungsbeschreibung der lokalen CPU in das Gerät laden, sondern auch die der Partner-CPU.

Verbindungen, z. B. für TSEND/TRCV, projektieren

Wenn Sie z. B. die Anweisungen für TSEND/TRCV für offene Kommunikation nutzen wollen, müssen Sie zunächst eine Verbindung (z. B. TCP-Verbindung) projektieren.

Um eine TCP-Verbindung zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 die Kommunikationspartner.
2. Wählen Sie die Schaltfläche "Verbindungen" und aus der Klappliste den Verbindungstyp "TCP-Verbindung".
3. Verbinden Sie per Drag & Drop die Kommunikationspartner miteinander (über Schnittstelle oder lokalen Endpunkt). Falls noch kein zugehöriges S7-Subnetz vorhanden ist, wird dieses automatisch angelegt.
   Alternativ können Sie auch eine Verbindung zu unspezifizierten Partnern einrichten.
4. Selektieren Sie die angelegte Verbindung in der Netzsicht.

5. Stellen Sie im Bereich "Allgemein", im Register "Eigenschaften" ggf. die Eigenschaften der Verbindung ein, z. B. den Namen der Verbindung und die verwendeten Schnittstellen der Kommunikationspartner.
   Für Verbindungen zu einem unspezifizierten Partner stellen Sie die Adresse des Partners ein.
   Im Bereich "Lokale ID" finden Sie die lokale ID (Referenz der Verbindung im Anwenderprogramm).
6. Wählen Sie in der Projektnavigation für eine der beiden CPUs den Ordner "Programmbausteine" und öffnen Sie im Ordner den OB 1 durch Doppelklick. Der Programmeditor öffnet sich.
7. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation", "Open User Communication" die gewünschte Anweisung, z. B. TSEND und ziehen Sie sie per Drag & Drop in ein Netzwerk des OB 1.
8. Vergeben Sie am Parameter ID der Anweisung die lokale ID der projektierten Verbindung, die für die Übertragung der Daten verwendet werden soll.
9. Verschalten Sie den Parameter "DATA" an der Anweisung TSEND mit den Anwenderdaten, z. B. in einem Datenbaustein.
10. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Richten Sie nach der oben beschriebenen Vorgehensweise die Verbindung in der Partner-CPU mit der Anweisung zum Empfangen, TRCV ein und laden Sie sie in die CPU.

Besonderheit bei ISO-Verbindungen mit CP 1543-1

Wenn Sie den Verbindungstyp "ISO-Verbindung" nutzen, müssen Sie das Kontrollkästchen "ISO-Protokoll verwenden" in den Eigenschaften des CP aktivieren, damit die Adressierung über MAC-Adressen funktioniert.

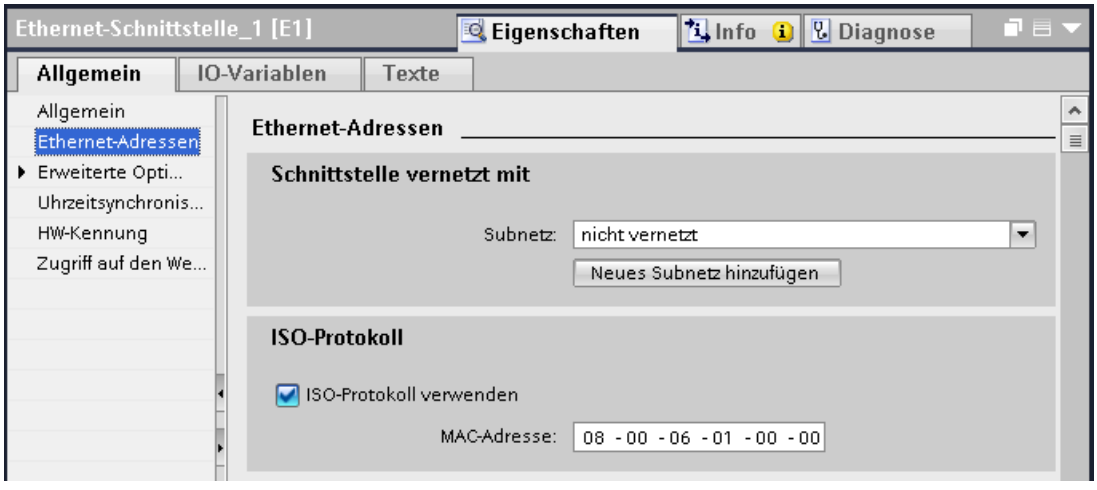


Bild 8-5 CP 1543-1 ISO-Protokoll wählen

Weitere Informationen

In der Online-Hilfe STEP 7 finden Sie beschrieben:

- Die Anweisungen für offene Kommunikation
- Die Verbindungsparameter

Wie sich die Anweisungen TSEND_C und TRCV_C in der S7-1500 verhalten, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/109479564).

8.6 Kommunikation über FDL einrichten

Voraussetzung

- Projektierungs-Software: STEP 7 Professional V14
- Endpunkt der Verbindung: CPU S7-1500 ab Firmware-Version V2.0 mit dem Kommunikationsmodul CM 1542-5 mit Firmware-Version V2.0

Einrichten einer konfigurierten FDL-Verbindung

Um in STEP 7 eine konfigurierte FDL-Verbindung einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie im Programmeditor eine Anweisung TSEND_C an.
2. Selektieren Sie die Anweisung TSEND_C und navigieren Sie im Inspektorfenster zu "Eigenschaften" > "Allgemein" > "Verbindungsparameter".
3. Wählen Sie unter Endpunkt den Partner-Endpunkt aus. Nutzen Sie einen der beiden folgenden Partner-Endpunkten:
   - CPU S7-1500 mit CM 1542-5
   - Unspezifiziert
4. Wählen Sie unter Konfigurationsart "Konfigurierte Verbindung verwenden aus".
5. Wählen Sie unter Verbindungstyp "FDL" aus.
6. Wählen Sie unter Schnittstelle die folgenden Schnittstellen aus:
   - Lokal: PROFIBUS-Schnittstelle des CM 1542-5
   - Spezifizierter Partner: PROFIBUS-Schnittstelle des CM 1542-5
7. Wählen Sie bei Verbindungsdaten die Einstellung <neu> aus.

Das folgende Bild zeigt eine vollständig konfigurierte FDL-Verbindung in STEP 7.

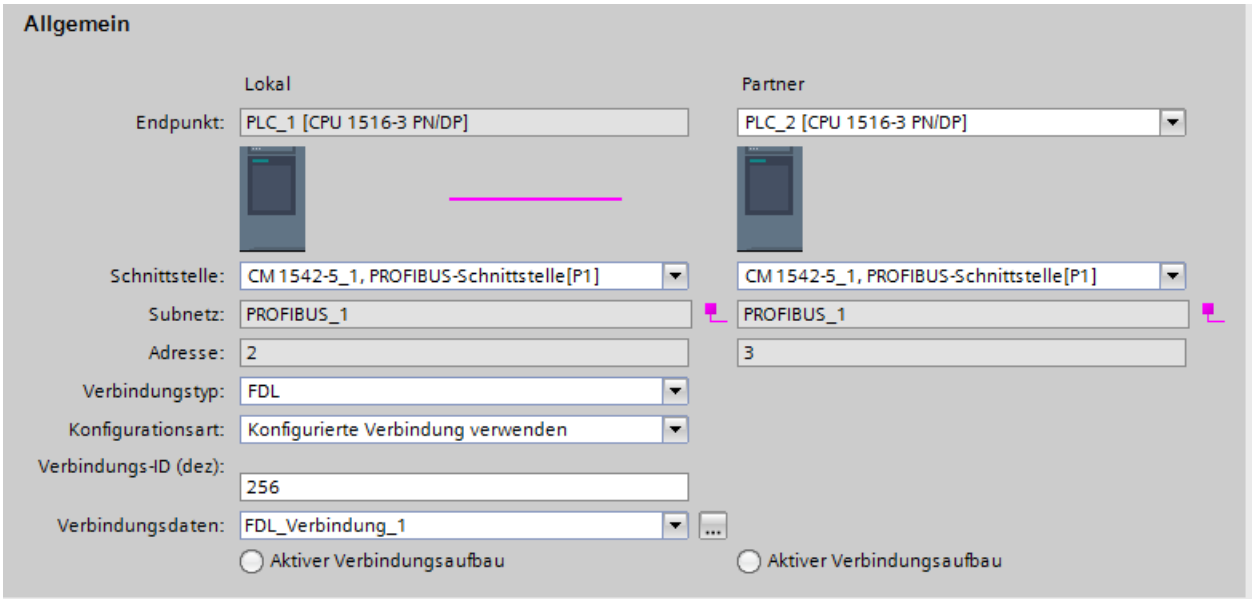


Bild 8-6 FDL-Verbindung konfigurieren

Einrichten einer FDL-Verbindung im Anwenderprogramm

Für die Kommunikation über FDL müssen Sie jeweils den Datenbaustein des Systemdatentyps TCON_FDL selbst erstellen, parametrieren und direkt an der Anweisung aufrufen. Gehen Sie folgendermaßen vor:

1. Legen Sie in der Projektnavigation einen globalen Datenbaustein an.
2. Definieren Sie im globalen Datenbaustein eine Variable vom Datentyp TCON_FDL.
   Das folgende Beispiel zeigt den globalen Datenbaustein "FDL_connection", in dem die Variable "FDL_connection" vom Datentyp TCON_FDL definiert ist.

FDL_connection

	Name	Datentyp	Startwert	R...	E...	S...	S...	E...	Überw..	Kommentar
1	Static									
2	FDL_connection	TCON_FDL			☑	☑	☑			
3	InterfaceId	HW_ANY	0		☑	☑	☑			HW identifier of PB interface submodule
4	ID	CONN_OUC	16#0		☑	☑	☑			connection reference / identifier
5	ConnectionType	Byte	16#15		☑	☑	☑			type of connection: 21= FDL connection
6	ActiveEstablished	Bool	false		☑	☑	☑			active/passive connection establishment
7	ServiceId	Byte	16#0		☑	☑	☑			service id: 0 – default, 1 – SDA, 2 – SDN
8	RemotePBAddress	Byte	16#0		☑	☑	☑			remote ProfiBus partner address
9	LocalPBAddress	Byte	16#0		☑	☑	☑			local ProfiBus partner address
10	RemoteLSAP	Byte	16#0		☑	☑	☑			remote PB link-layer service access point
11	LocalLSAP	Byte	16#0		☑	☑	☑			local PB link-layer service access point

Bild 8-7 FDL-Verbindung programmieren

3. Programmieren Sie die Parameter der FDL-Verbindung (z. B. die PROFIBUS-Adressen) in der Variablen vom Datentyp TCON_FDL.
4. Legen Sie im Programmeditor eine Anweisung TCON an.

5. Verschalten Sie den Parameter CONNECT der Anweisung TCON mit der Variable vom Datentyp TCON_FDL.
   Im folgenden Beispiel ist der Parameter CONNECT der Anweisung TCON mit der Variablen "FDL_Connection" (Datentyp TCON_FDL) verschaltet.

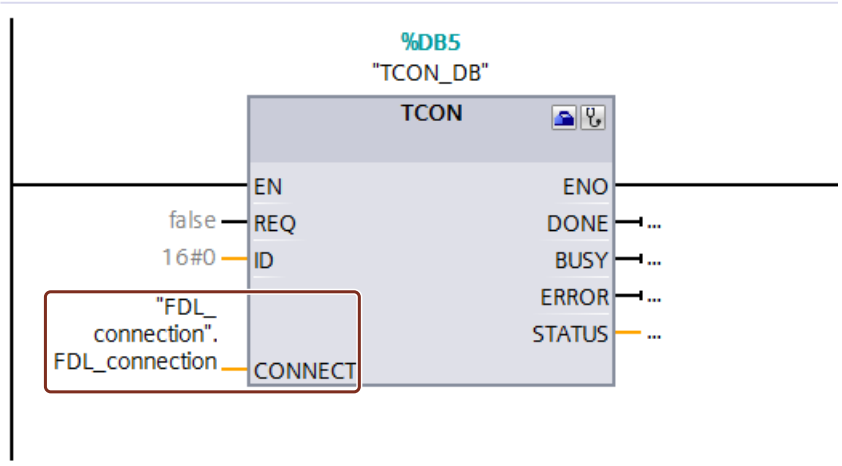


Bild 8-8 Beispiel: Anweisung TCON für FDL-Verbindung

8.7 Kommunikation über Modbus TCP einrichten

Einrichten einer Verbindung über das Anwenderprogramm für Modbus TCP

Die Parametrierung erfolgt im Programmeditor an der Anweisung MB_CLIENT bzw. MB_SERVER.

Vorgehen zum Einrichten der Kommunikation über Modbus TCP

Die Anweisung MB_CLIENT kommuniziert als Modbus TCP-Client über die TCP-Verbindung. Mit der Anweisung bauen Sie eine Verbindung zwischen dem Client und dem Server auf, senden Modbus-Anfragen zum Server und empfangen die entsprechenden Modbus-Antworten. Weiterhin steuern Sie mit dieser Anweisung den Abbau der TCP-Verbindung.
Die Anweisung MB_SERVER kommuniziert als Modbus TCP-Server über eine TCP-Verbindung. Die Anweisung verarbeitet Verbindungsanfragen eines Modbus-Clients, empfängt und bearbeitet Modbus-Anfragen und sendet Antwort-Meldungen. Weiterhin können Sie den Abbau der TCP-Verbindung steuern.
Voraussetzung: Der Client kann den Server über IP-Kommunikation im Netzwerk erreichen.

1. Konfigurieren Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 ein Automatisierungssystem S7-1500 mit CPU.
2. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie im Ordner den OB 1 durch Doppelklick. Der Programmeditor öffnet sich.
3. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation", "Weitere", "MODBUS TCP" die gewünschte Anweisung, z. B. MB_CLIENT und ziehen Sie sie per Drag & Drop in ein Netzwerk des OB 1.

4. Parametrieren Sie die Anweisung MB_CLIENT bzw. MB_SERVER. Halten Sie dabei folgende Regeln ein:
   Für jede MB_CLIENT-Verbindung muss eine IPv4-Adresse des Servers spezifiziert sein.
   Jede Verbindung MB_CLIENT oder MB_SERVER muss einen eindeutigen Instanz-DB mit einer der Datenstrukturen TCON_IP_v4, TCON_QDN oder TCON_Configured verwenden.
   Jede Verbindung benötigt eine eindeutige Verbindungs-ID. Verbindungs-ID und Instanz-DB gehören jeweils paarweise zusammen und müssen für jede Verbindung eindeutig sein.

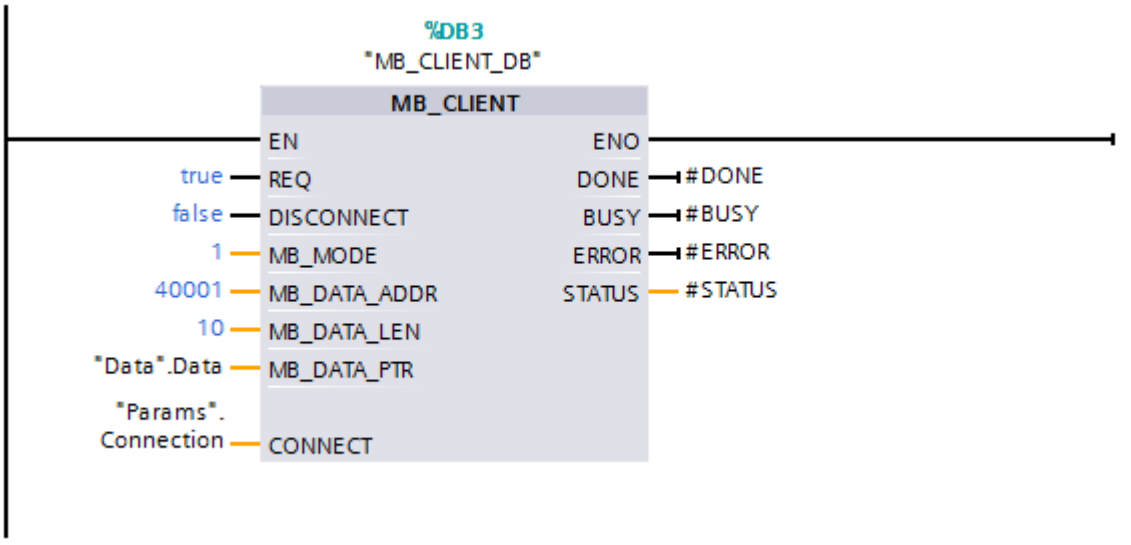


Bild 8-9 MB_CLIENT

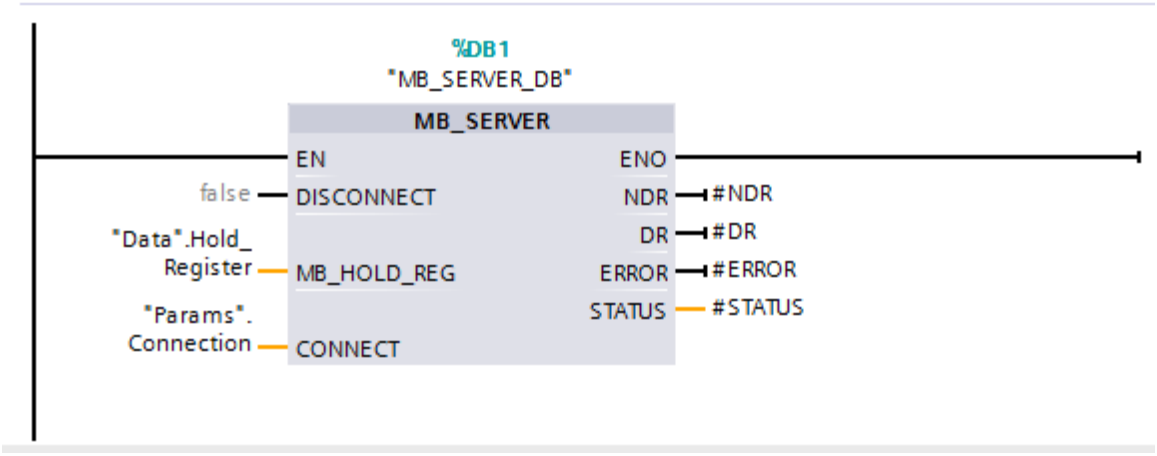


Bild 8-10 MB_SERVER

5. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Redundante Kommunikation über Modbus TCP

Redundante Kommunikation über Modbus TCP parametrieren Sie mit den Anweisungen MB_RED_CLIENT bzw. MB_RED_SERVER:

Anweisung MB_RED_CLIENT: Über die Anweisung "MB_RED_CLIENT" bauen Sie eine redundante Verbindung zwischen dem Client und dem Server auf, senden Modbus-Anfragen, empfangen Antworten und steuern den Verbindungsabbau des Modbus TCP-Clients.

Anweisung MB_RED_SERVER: Die Anweisung "MB_RED_SERVER" verarbeitet Verbindungsanfragen eines Modbus TCP-Clients, empfängt und bearbeitet Modbus-Anfragen und sendet Antwort-Meldungen. Die CPUs sind in der Lage:

- mehrere Serververbindungen zu bearbeiten und
- auf einem Serverport gleichzeitig mehrere Verbindungen von verschiedenen Clients anzunehmen.

Weitere Informationen zu den Anweisungen MB_RED_CLIENT bzw. MB_RED_SERVER finden Sie in der Online-Hilfe zu STEP 7.

Modbus TCP-Server als Gateway zu Modbus RTU

Wenn Sie einen Modbus TCP-Server als Gateway zu einem Modbus RTU-Protokoll verwenden, dann adressieren Sie das Slavegerät im seriellen Netzwerk über den statischen Parameter MB_UNIT_ID. Der Parameter MB_UNIT_ID entspricht dem Feld der Slaveadresse beim Modbus RTU-Protokoll. Der Parameter MB_UNIT_ID würde in diesem Fall die Anforderung an die richtige Modbus RTU-Slaveadresse weiterleiten.

Die Gateway Funktion müssen Sie selbst programmieren.

Den Parameter MB_UNIT_ID finden Sie in dem der Anweisung MB_CLIENT zugehörigen Instanzdatenbaustein.

Weitere Informationen zum Parameter MB_UNIT_ID finden Sie in der Online-Hilfe zu STEP 7.

Verweis

- Wie Sie die Modbus TCP-Kommunikation zwischen zwei S7-1500 CPUs programmieren und parametrieren, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/94766380).
- Wie Sie eine Modbus TCP-Kommunikation zwischen einer S7-1500 CPU und einer S7-1200 CPU programmieren und parametrieren, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/102020340).

8.8 Kommunikation über E-Mail einrichten

Einrichten einer Verbindung über das Anwenderprogramm für E-Mail

Für die Kommunikation über E-Mail müssen Sie jeweils den Datenbaustein des entsprechenden Systemdatentyps selbst erstellen, parametrieren und direkt an der Anweisung aufrufen. Das Vorgehen ist nachfolgend dargestellt.

Vorgehen zum Einrichten der Kommunikation über E-Mail

E-Mails können von einer CPU gesendet werden. Für das Senden von E-Mails aus dem Anwenderprogramm der CPU setzen Sie die Anweisung TMAIL_C ein.

Voraussetzung: Der SMTP-Server ist über das IPv4-Netzwerk erreichbar.

1. Konfigurieren Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 ein Automatisierungssystem S7-1500 mit CPU.
2. Parametrieren Sie die Anweisung TMAIL_C, geben Sie z. B. bei Subject den Betreff der Mail ein.
3. Erzeugen Sie in einem globalen Datenbaustein eine Variable vom Typ TMAIL_v4, TMAIL_v6 (nur CP 1543-1) oder TMAIL_FQDN (nur CP 1543-1).
4. Stellen Sie in der Variable die Verbindungsparameter der TCP-Verbindung in der Spalte "Startwert" ein. Tragen Sie z. B. bei "MailServerAdress" die IPv4-Adresse des Mailservers ein (für TMAIL_v4)

   HINWEIS

   Verbindungsparameter InterfaceId

   Beachten Sie, dass Sie ab Anweisungsversion V5.0 der Anweisung TMAIL_C im Datentyp TMAIL_V4_SEC den Wert "**0**" für die InterfaceId eintragen können. In diesem Fall sucht die CPU selbst nach einer passenden lokalen Schnittstelle der CPU.

   Verschalten Sie die Variable mit dem Parameter MAIL_ADDR_PARAM der Anweisung TMAIL_C.
5. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Weitere Informationen

In der Online-Hilfe STEP 7 finden Sie beschrieben:

- die Systemdatentypen
- die Anweisungen für offene Kommunikation
- die Verbindungsparameter

8.9 Kommunikation über FTP einrichten

Einrichten einer Verbindung über das Anwenderprogramm für FTP

Für die Kommunikation über FTP müssen Sie jeweils den Datenbaustein des entsprechenden Systemdatentyps selbst erstellen, parametrieren und direkt an der Anweisung aufrufen. Das Vorgehen ist nachfolgend dargestellt.

FTP-Client und -Server-Funktionalität

Dateien können von einer CPU an einen FTP-Server gesendet und von diesem empfangen werden. Die Kommunikation über FTP ist für S7-1500 nur über CP 1543-1 möglich. Der CP kann FTP-Server, FTP-Client oder beides sein. FTP-Clients können auch Fremdsysteme/PCs sein.

Für die FTP-Server-Funktionalität projektieren Sie den CP entsprechend in STEP 7.

Mit der FTP-Client-Funktionalität realisieren Sie z. B. den Aufbau und Abbau einer FTP-Verbindung, das Übertragen und Löschen von Dateien auf dem Server. Für die FTP-Client-Funktionalität setzen Sie die Anweisung FTP_CMD ein.

Vorgehen zum Einrichten der FTP-Server-Funktionalität

Voraussetzung: Der FTP-Server ist über das IPv4-Netzwerk erreichbar.

1. Konfigurieren Sie in der Gerätesicht des Hardware- und Netzwerkeditors von STEP 7 ein Automatisierungssystem S7-1500 mit CPU und CP 1543-1.
   Zugleich müssen Sie in der HW-Konfiguration der CPU S7-1500 unter der Bereichsnavigation "Schutz" im Abschnitt "Verbindungsmechanismen" die Option "Zugriff über PUT/GET Kommunikation durch entfernten Partner (PLC, HMI, OPC, ...) erlauben" aktivieren.
2. Nehmen Sie in den Eigenschaften des CP unter "FTP-Konfiguration" folgende Einstellungen vor:
   - Wählen Sie das Auswahlkästchen "FTP-Server für S7-CPU-Daten verwenden" an.
   - Ordnen Sie die CPU, einen Datenbaustein und einen Dateinamen, unter dem der DB für FTP abgelegt wird zu.

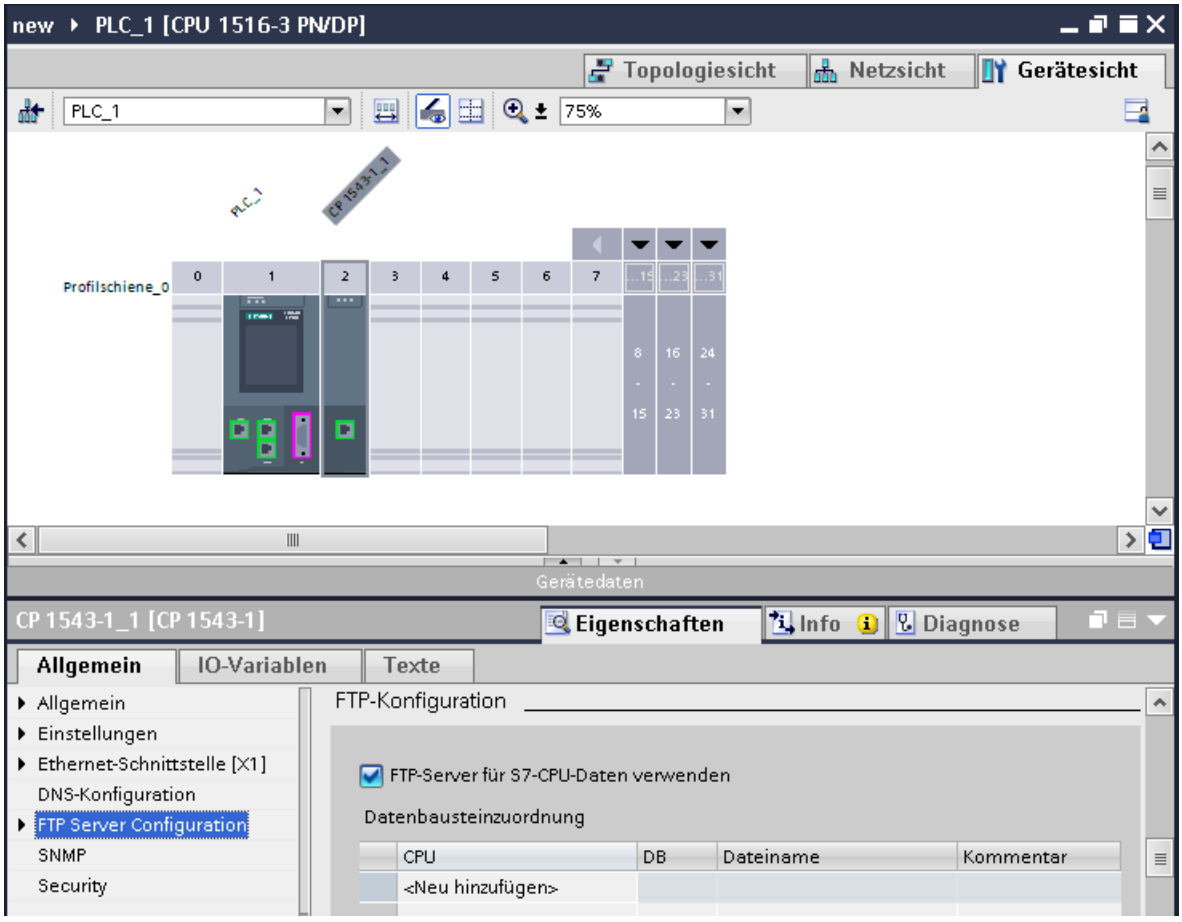


Bild 8-11 FTP-Konfiguration einrichten

3. Laden Sie die Hardware-Konfiguration in die CPU.

Vorgehen zum Einrichten der FTP-Client-Funktionalität

Voraussetzung: Der FTP-Server ist über das IPv4-Netzwerk erreichbar**.**

1. Konfigurieren Sie in der Gerätesicht des Hardware- und Netzwerkeditors von STEP 7 ein Automatisierungssystem S7-1500 mit CPU und CP 1543-1.
   Zugleich müssen Sie in der HW-Konfiguration der CPU S7-1500 unter der Bereichsnavigation "Schutz" im Abschnitt "Verbindungsmechanismen" die CheckBox "Zugriff über PUT/GET Kommunikation durch entfernten Partner (PLC, HMI, OPC, ...) erlauben" aktivieren.
2. Rufen Sie die Anweisung FTP_CMD im Anwenderprogramm der CPU auf.
3. Parametrieren Sie an der Anweisung FTP_CMD die Verbindungsparameter für den FTP-Server.
4. Erzeugen Sie einen Global-DB und innerhalb dieses Global-DBs eine Variable vom Type FTP_CONNECT_IPV4, FTP_CONNECT_IPV6 oder FTP_CONNECT_NAME.
5. Verschalten Sie die Variable innerhalb des Datenbausteins mit der Anweisung FTP_CMD.
6. Für die Verbindung zum FTP-Server geben Sie im DB an:
   – den Benutzernamen, das Kennwort und die IP-Adresse für den FTP-Zugang im entsprechenden Datentyp (FTP_CONNECT_IPV4, FTP_CONNECT_IPV6 oder FTP_CONNECT_NAME)
7. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Anwendungsbeispiele

- Anwendungsbeispiel: FTP-Kommunikation mit S7-1500 und CP 1543-1
  Das Anwendungsbeispiel finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/103550797).
- Anwendungsbeispiel: FTP-Client Kommunikation mit S7-1200/1500
  Das Anwendungsbeispiel finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/81367009).

Weitere Informationen

In der Online-Hilfe STEP 7 finden Sie beschrieben:

- die Systemdatentypen
- die Anweisungen für offene Kommunikation
- die Verbindungsparameter

8.10 Auf- und Abbau von Kommunikationsbeziehungen

Auf- und Abbau von Kommunikationsbeziehungen

Die folgende Tabelle zeigt den Auf- und Abbau von Kommunikationsbeziehungen im Rahmen der offenen Kommunikation.

Tabelle 8-5 Auf- und Abbau von Kommunikationsbeziehungen

Einrichten der Verbindung	Aufbau der Kommunikationsbeziehung	Abbau der Kommunikationsbeziehung
Über Anwenderprogramm	Nach dem Laden des Anwenderprogramms in die CPUs: Der passive Kommunikationspartner richtet mit dem Aufruf von TSEND_C/TRCV_C bzw. TCON den lokalen Verbindungszugang ein. Der Aufruf von TSEND_C/TRCV_C bzw. TCON im aktiven Partner startet den Verbindungsaufbau. Konnte die Verbindung aufgebaut werden, erfolgt eine positive Rückmeldung an den Anweisungen im Anwenderprogramm. Nachdem Sie eine Verbindung mit der Anweisung T_RESET abgebaut haben, wird die Verbindung neu aufgebaut. Bei einem Verbindungsabbruch versucht der aktive Partner, die eingerichtete Verbindung wieder aufzubauen. Dies gilt nur, wenn zuvor der Verbindungsaufbau mit TCON erfolgreich war.	• Über die Anweisungen TSEND_C/TRCV_C, TDISCON und T_RESET • Wenn die CPU vom Betriebszustand RUN in STOP übergeht • Bei NETZ-AUS/NETZ-EIN Ein an einer CPU
Über Verbindungsprojektierung	Nach dem Laden der Verbindungsprojektierung und des Anwenderprogramms in die CPUs.	Durch Löschen der Verbindungsprojektierung in STEP 7 und Laden der geänderten Projektierung in die CPU.

S7-Kommunikation

9

Merkmale S7-Kommunikation

Die S7-Kommunikation als SIMATIC-homogene Kommunikation zeichnet sich aus durch herstellerspezifische Kommunikation zwischen SIMATIC-CPUs (kein offener Standard). Die S7-Kommunikation dient der Migration und Anbindung an bestehende Systeme (S7-300, S7-400).

Für die Datenübertragung zwischen zwei Automatisierungssystemen S7-1500 empfehlen wir Ihnen, die offene Kommunikation zu nutzen (siehe Kapitel Open User Communication (Seite 126)).

Eigenschaften der S7-Kommunikation

Über S7-Kommunikation tauscht die CPU Daten mit einer weiteren CPU aus. Sobald der Anwender die Daten auf der Empfängerseite empfangen hat, wird der Empfang der Daten automatisch an die Sende-CPU quittiert.

Der Datenaustausch erfolgt über projektierte S7-Verbindungen. S7-Verbindungen können einseitig oder zweiseitig projektiert werden.

S7-Kommunikation ist möglich über:

- Integrierte PROFINET- oder PROFIBUS DP-Schnittstelle einer CPU
- Schnittstelle eines CP/CM

Einseitig projektierte S7-Verbindungen

Bei einer einseitig projektierten S7-Verbindung erfolgt die Projektierung für diese Verbindung nur in einem Kommunikationspartner und wird auch nur in diesen geladen.

Eine einseitige S7-Verbindung kann zu einer CPU projektiert werden, die nur Server einer S7-Verbindung ist (z. B. CPU 315-2 DP). Die CPU ist projektiert und damit sind die Adressparameter und Schnittstellen bekannt.

Außerdem kann eine einseitige S7-Verbindung zu einem Partner projektiert werden, der nicht im Projekt vorhanden ist und dessen Adressparameter und Schnittstelle daher nicht bekannt sind. Die Adresse müssen Sie eingeben; sie wird von STEP 7 nicht überprüft. Der Partner ist initial unspezifiziert (beim Anlegen der S7-Verbindung ist noch keine Partneradresse eingetragen). Sobald Sie die Adresse eingeben, ist er "unbekannt" (das heißt: er ist spezifiziert, aber dem Projekt unbekannt).

Damit besteht die Möglichkeit, über Projektgrenzen hinweg S7-Verbindungen einzusetzen. Der Kommunikationspartner ist für das lokale Projekt unbekannt (unspezifiziert) und wird in einem anderen STEP 7- oder Fremd-Projekt projektiert.

Zweiseitig projektierte S7-Verbindungen

Bei einer zweiseitig projektierten S7-Verbindung erfolgt die Projektierung und das Laden der projektierten S7-Verbindungsparameter in beide Kommunikationspartner.

Anweisungen für S7-Kommunikation

Für S7-Kommunikation in S7-1500 sind folgende Anweisungen einsetzbar:

- PUT/GET
  Mit der Anweisung PUT schreiben Sie Daten in eine remote CPU. Mit der Anweisung GET lesen Sie Daten aus einer remoten CPU aus. Die Anweisungen PUT und GET sind einseitige Anweisungen, d. h. Sie benötigen nur Anweisung in einem Kommunikationspartner. Die Anweisungen PUT und GET können Sie bequem über die Verbindungsparametrierung einrichten.

  HINWEIS
  Datenbausteine für Anweisungen PUT/GET
  Bei Verwendung der Anweisungen PUT/GET dürfen Sie nur Datenbausteine mit absoluter Adressierung einsetzen. Symbolische Adressierung von Datenbausteinen ist nicht möglich.
  Desweitern müssen Sie diesen Service im Bereich "Schutz" in der CPU-Projektierung freischalten.

  Wie Sie für den Datenaustausch zwischen zwei S7-1500 CPUs eine S7-Verbindung und die Kommunikationsanweisungen PUT und GET projektieren und programmieren, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/82212115).
- BSEND/BRCV
  Die Anweisung BSEND sendet Daten an eine remote Partneranweisung vom Typ BRCV. Die Anweisung BRCV empfängt Daten von einer remoten Partneranweisung vom Typ BSEND. Die S7-Kommunikation über das Anweisungspaar BSEND/BRCV verwenden Sie für sicheres Übertragen von Daten.
- USEND/URCV
  Die Anweisung USEND sendet Daten an eine remote Partneranweisung vom Typ URCV. Die Anweisung URCV empfängt Daten von einer remoten Partneranweisung vom Typ USEND. Die S7-Kommunikation über das Anweisungspaar USEND/URCV verwenden Sie für schnelles, ungesichertes Übertragen von Daten unabhängig von der zeitlichen Bearbeitung des Kommunikationspartners; z. B. für Betriebs- und Wartungsmeldungen.

S7-Kommunikation über PROFIBUS DP-Schnittstelle im Slave-Betrieb

Sie finden in STEP 7, in den Eigenschaften der PROFIBUS DP-Schnittstelle von Kommunikationsmodulen (z. B. CM 1542-5) das Kontrollkästchen "Test, Inbetriebnahme und Routing". Über dieses Kontrollkästchen stellen Sie ein, ob die PROFIBUS DP-Schnittstelle des DP-Slaves aktiver oder passiver Teilnehmer am PROFIBUS ist.

- Kontrollkästchen aktiviert: DP-Slave ist aktiver Teilnehmer am PROFIBUS.
- Kontrollkästchen deaktiviert: DP-Slave ist passiver Teilnehmer am PROFIBUS. Zu diesem DP-Slave können Sie nur einseitig projektierte S7-Verbindungen einrichten.

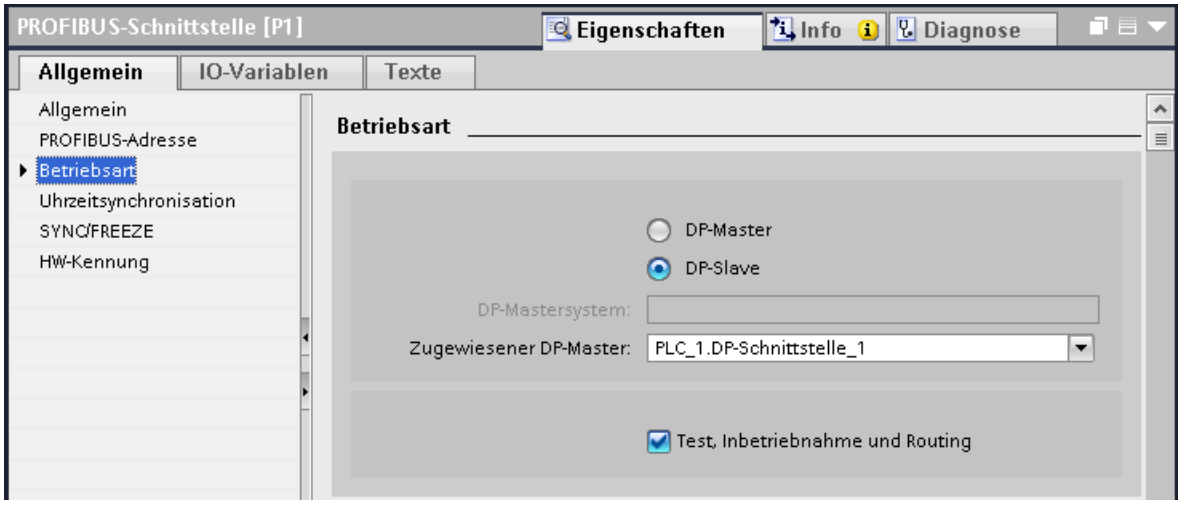


Bild 9-1 Kontrollkästchen "Test, Inbetriebnahme und Routing"

S7-Verbindungen für PUT/GET-Anweisungen parametrieren

In der Verbindungsparametrierung der PUT/GET-Anweisungen können Sie S7-Verbindungen anlegen und parametrieren. Geänderte Werte werden von der Verbindungsparametrierung sofort auf Eingabefehler geprüft.

Voraussetzung: Im Programmiereditor ist eine Anweisung PUT bzw. GET angelegt.

Um eine S7-Verbindung über PUT/GET-Anweisungen zu projektieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie im Programmiereditor den Aufruf der Anweisung PUT oder GET.
2. Öffnen Sie im Inspektorfenster das Register "Eigenschaften > Konfiguration".
3. Selektieren Sie die Gruppe "Verbindungsparameter". Solange Sie noch keinen Verbindungspartner selektiert haben, ist nur die leere Klappliste für den Partner-Endpunkt aktiv. Alle anderen Eingabemöglichkeiten sind deaktiviert.

   Es werden die bereits bekannten Verbindungsparameter angezeigt:

   – Name des lokalen Endpunkts
   – Schnittstelle des lokalen Endpunkts

– IPv4-Adresse des lokalen Endpunkts

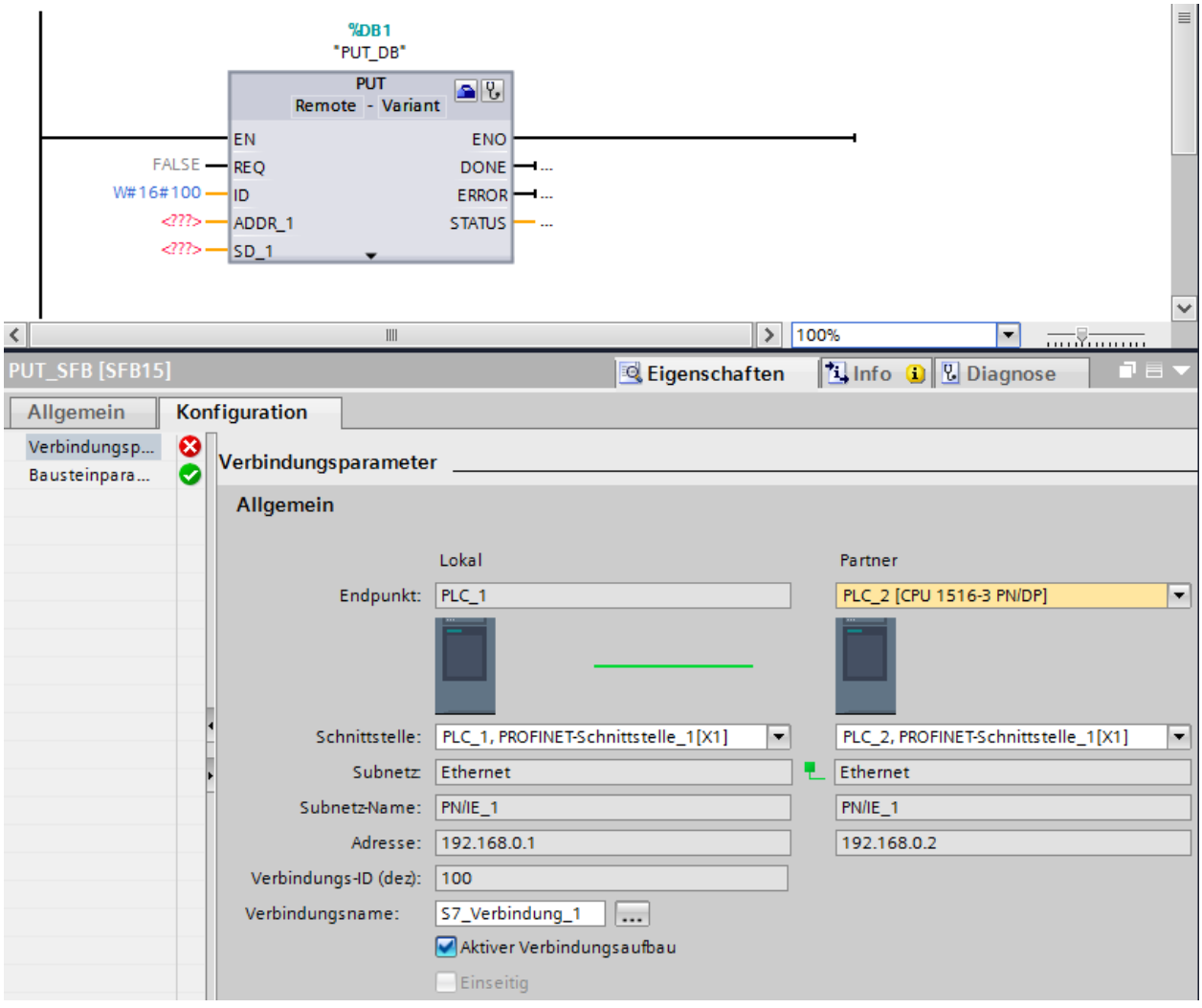


Bild 9-2 Verbindungsparametrierung für PUT-Anweisung

4. Wählen Sie in der Klappliste des Partner-Endpunkts einen Verbindungspartner. Als Kommunikationspartner kommt ein unspezifiziertes Gerät oder eine im Projekt vorhandene CPU in Frage.

   Die folgenden Parameter werden automatisch eingetragen, sobald Sie den Verbindungspartner gewählt haben:

   – Name des Partner-Endpunkts
   – Schnittstelle des Partner-Endpunkts. Wenn mehrere Schnittstellen zur Verfügung stehen, dann können Sie die Schnittstelle bei Bedarf ändern.
   – Schnittstellentyp des Partner-Endpunkts
   – Subnetz-Name beider Endpunkte
   – IPv4-Adresse des Partner-Endpunkts
   – Name der Verbindung, die für die Kommunikation genutzt wird.

5. Ändern Sie bei Bedarf den Verbindungsnamen im Eingabefeld "Verbindungsname" ab. Wenn Sie eine neue Verbindung erstellen, oder eine vorhandene Verbindung bearbeiten möchten, klicken Sie auf die Schaltfläche "Verbindung auswählen" rechts neben dem Eingabefeld für den Verbindungsnamen.

HINWEIS

Die Anweisungen PUT und GET zwischen zwei Kommunikationspartnern sind erst dann lauffähig, wenn sowohl die Hardware-Konfiguration wie auch der Programmteil für den Partner-Endpunkt in die Hardware geladen wurden. Achten Sie darauf, dass Sie für eine funktionierende Kommunikation nicht nur die Verbindungsbeschreibung der lokalen CPU in das Gerät laden, sondern auch die der Partner-CPU.

S7-Verbindungen für z. B. BSEND/BRCV projektieren

Wenn Sie z. B. die Anweisungen für BSEND/BRCV für S7-Kommunikation nutzen wollen, müssen Sie zunächst eine S7-Verbindung projektieren.

Um eine S7-Verbindung zu projektieren, gehen Sie folgendermaßen vor:

1. Konfigurieren Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 die Kommunikationspartner.
2. Wählen Sie die Schaltfläche "Verbindungen" und aus der Klappliste den Eintrag "S7-Verbindung".
3. Verbinden Sie per Drag & Drop die Kommunikationspartner miteinander (über Schnittstelle oder lokalen Endpunkt). Falls noch kein zugehöriges S7-Subnetz vorhanden ist, wird dieses automatisch angelegt.
   Alternativ können Sie auch eine Verbindung zu unspezifizierten Partnern einrichten.
4. Wählen Sie im Register "Verbindungen" die Zeile der S7-Verbindung.
5. Stellen Sie im Bereich "Allgemein", im Register "Eigenschaften" ggf. die Eigenschaften der S7-Verbindung ein, z. B. den Namen der Verbindung und die verwendeten Schnittstellen der Kommunikationspartner.
   Für S7-Verbindungen zu einem unspezifizierten Partner stellen Sie die Adresse des Partners ein.
   Im Bereich "Lokale ID" finden Sie die lokale ID (Referenz der S7-Verbindung im Anwenderprogramm).
6. Wählen Sie in der Projektnavigation für eine der beiden CPUs den Ordner "Programmbausteine" und öffnen Sie im Ordner den OB 1 durch Doppelklick. Der Programmeditor öffnet sich.
7. Rufen Sie im Programmeditor die entsprechenden Anweisungen für S7-Kommunikation im Anwenderprogramm des Kommunikationspartners (einseitig) bzw. in den Anwenderprogrammen der Kommunikationspartner (zweiseitig) auf. Wählen Sie z. B. aus der Task Card "Anweisungen", Bereich "Kommunikation" die Anweisungen BSEND und BRCV und ziehen Sie sie per Drag & Drop in ein Netzwerk des OB 1.
8. Vergeben Sie am Parameter ID der Anweisung die lokale ID der projektierten Verbindung, die für die Übertragung der Daten verwendet werden soll.
9. Parametrieren Sie die Anweisungen, welche Daten wohin geschrieben bzw. woher gelesen werden.
10. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU(s).

S7-Kommunikation über CP 1543-1

Wenn Sie die S7-Kommunikation über die Industrial Ethernet-Schnittstelle des CP 1543-1 einrichten, können Sie in den Eigenschaften der S7-Verbindung unter "Allgemein" das Transportprotokoll für die Datenübertragung wählen:

- Kontrollkästchen "TCP/IP" aktiviert (voreingestellt): ISO-on-TCP (RFC 1006): für die S7-Kommunikation zwischen CPUs S7-1500
- Kontrollkästchen "TCP/IP" deaktiviert: ISO-Protokoll (ISO/IEC 8073): Adressierung über MAC-Adressen

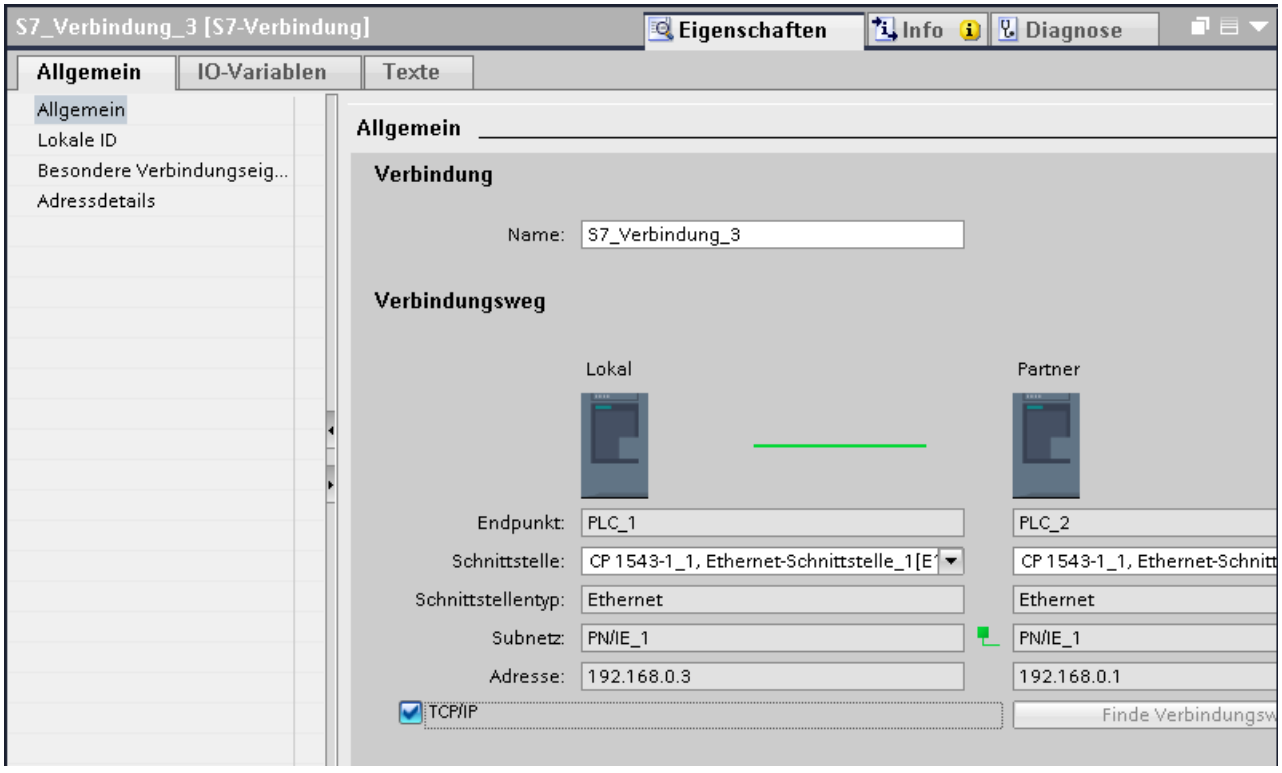


Bild 9-3 CP 1543-1 Transportprotokoll wählen

Vorgehen zum Einrichten einer S7-Verbindung über unterschiedliche S7-Subnetze

Sie haben die Möglichkeit eine S7-Verbindung über mehrere S7-Subnetze (PROFIBUS, PROFINET/Industrial Ethernet) hinweg zu nutzen (S7-Routing (Seite 380)).

1. Konfigurieren Sie in der Netzsicht des Hardware- und Netzwerkeditors von STEP 7 die Kommunikationspartner.
2. Wählen Sie die Schaltfläche "Vernetzen".
3. Verbinden Sie per Drag & Drop die entsprechenden Schnittstellen mit den jeweiligen S7-Subnetzen (PROFIBUS oder PROFINET / Industrial Ethernet).
4. Wählen Sie die Schaltfläche "Verbindungen" und aus der Klappliste den Eintrag "S7-Verbindung".

5. Verbinden Sie per Drag & Drop in unserem Beispiel die PLC_1 im linken S7-Subnetz (PROFIBUS) mit der PLC_3 rechten S7-Subnetz (PROFINET).
   Die S7-Verbindung von CPU 1 zu CPU 3 ist konfiguriert.

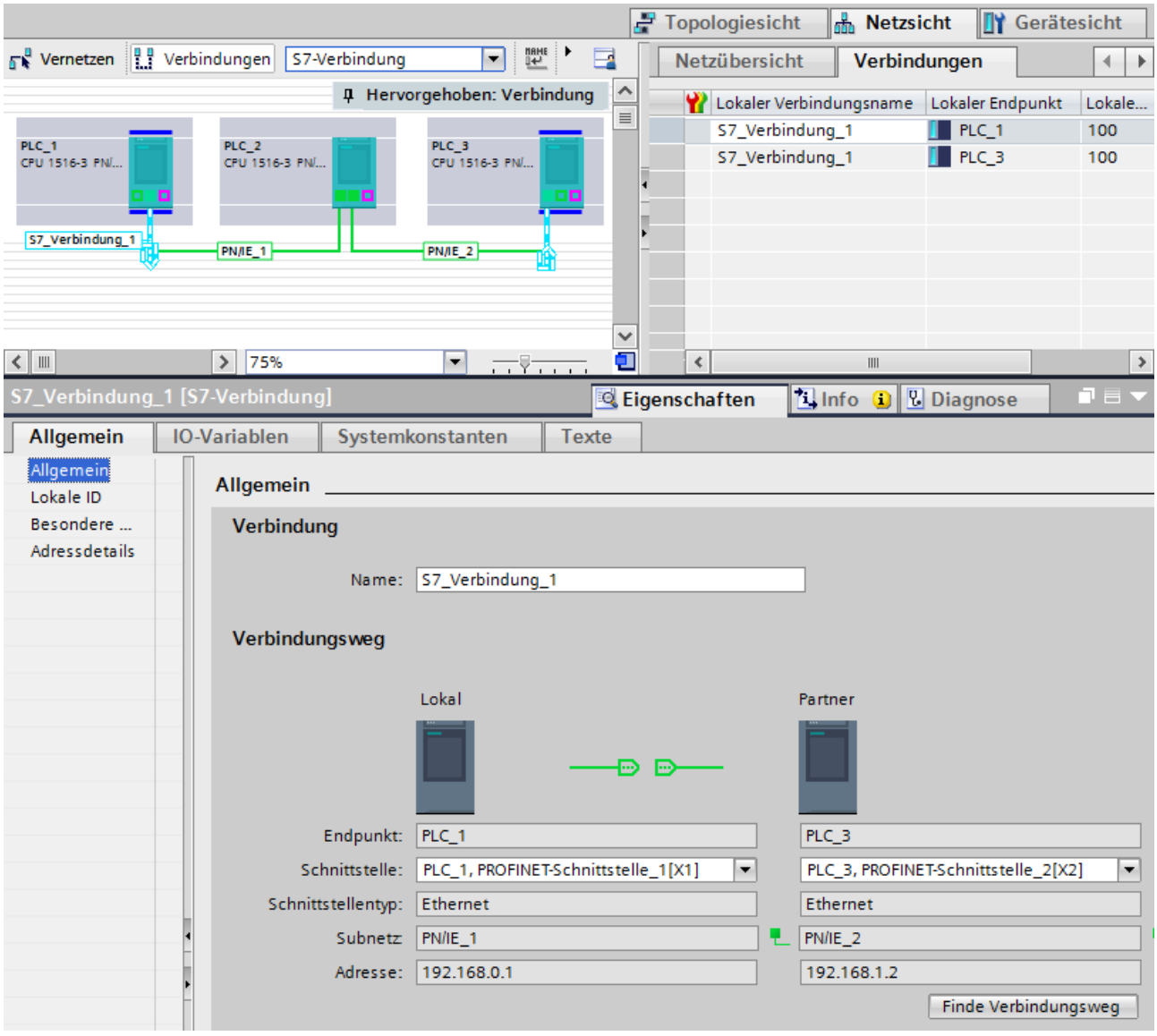


Bild 9-4 S7-Verbindungen über unterschiedliche Subnetze

ET 200SP Open Controller als Router für S7-Verbindungen

Wenn Sie die Schnittstelle "PROFINET onboard [X2]" der CPU 1515SP PC (F) der SIMATIC PC-Station zuweisen, kann die CPU 1515SP PC (F) als Router für S7-Verbindungen verwendet werden. Wenn Sie die CP-Schnittstelle für "Keine oder eine andere Windows-Einstellung" verwenden, dann können Sie den Open Controller nicht als Router für geroutete S7-Verbindungen verwenden.

Eine bestehende durch die CPU 1515SP PC (F) geroutete S7-Verbindung wird ungültig, wenn die Zuweisung der Schnittstelle der CPU 1515SP PC (F) von "SIMATIC PC-Station" zu "Keine oder eine andere Windows-Einstellung" geändert wird. Da die PLC nun keine Routingfunktion mehr für diese Verbindung übernimmt, wird beim Übersetzen der CPU 1515SP PC (F) kein Hinweis auf die ungültige Verbindung angezeigt. Die ungültige geroutete S7-Verbindung wird Ihnen erst beim Übersetzen der Endpunkte der Verbindung angezeigt.

Die für geroutete S7-Verbindungen benötigten Schnittstellen müssen in der CPU 1515SP PC (F) explizit zugewiesen bleiben. Sie können die Zuweisung der Schnittstelle der CPU 1515SP PC (F) in den Eigenschaften unter "PROFINET onboard [X2] > Schnittstellenzuweisung" bearbeiten.

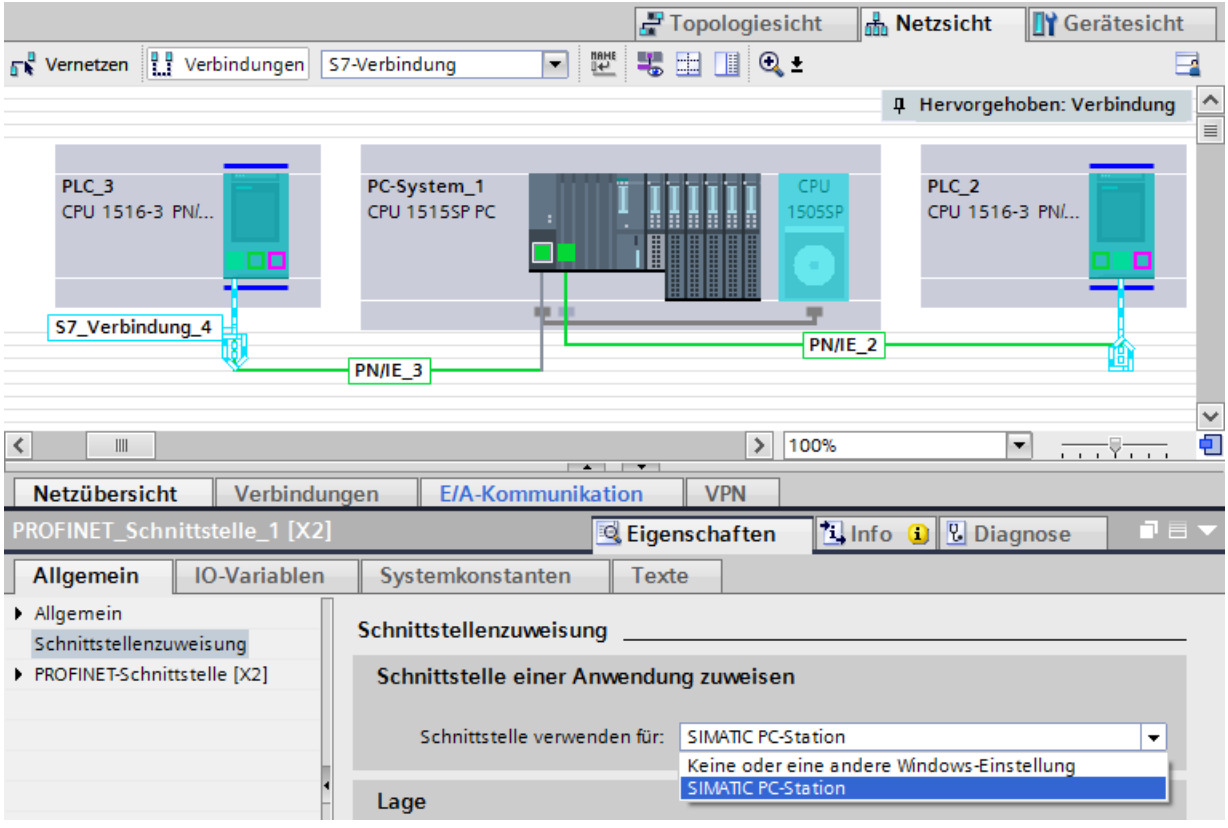


Bild 9-5 S7-Routing PC-Station

Weitere Informationen

Detaillierte Informationen zum Projektieren von S7-Verbindungen und wie Sie Anweisungen für die S7-Kommunikation im Anwenderprogramm nutzen, finden Sie in der Online-Hilfe STEP 7 beschrieben.

10

Punkt-zu-Punkt-Kopplung

Funktionalität

Die Kommunikation über Punkt-zu-Punkt-Kopplung bei S7-1500, ET 200MP und ET 200SP erfolgt über Kommunikationsmodule (CM) mit seriellen Schnittstellen (RS232, RS422 oder RS485):

- S7-1500/ET 200MP:
  - CM PtP RS232 BA
  - CM PtP RS422/485 BA
  - CM PtP RS232 HF
  - CM PtP RS422/485 HF
- ET 200SP:
  - CM PtP

Der bidirektionale Datenaustausch über Punkt-zu-Punkt-Kopplung funktioniert zwischen Kommunikationsmodulen oder kommunikationsfähigen Fremdsystemen oder -geräten. Zur Kommunikation sind mindestens 2 Kommunikationspartner notwendig ("Punkt zu Punkt"). Bei RS422 und RS485 sind mehr als zwei Kommunikationspartner möglich.

Protokolle für die Kommunikation über Punkt-zu-Punkt-Kopplung

- Freeport-Protokoll (auch ASCII-Protokoll genannt)
- Prozedur 3964(R)
- Modbus-Protokoll im RTU-Format (RTU: Remote Terminal Unit)
- USS-Protokoll (Universelles-serielles-Schnittstellen-Protokoll)

Die Protokolle nutzen unterschiedliche Schichten nach dem ISO/OSI-Referenzmodell:

- Freeport: nutzt Schicht 1 (Bitübertragungsschicht)
- 3964(R), USS und Modbus: nutzen Schicht 1 und 2 (Bitübertragungsschicht und Sicherungsschicht; damit höhere Übertragungssicherheit als bei Freeport). USS und Modbus nutzen zusätzlich Schicht 4.

Eigenschaften des Freeport-Protokolls

- Der Empfänger erkennt das Ende der Datenübertragung über ein parametrierbares Endekriterium (z. B. Ablauf Zeichenverzugszeit, Empfang Endezeichen, Empfang feste Anzahl Daten).
- Der Sender kann nicht erkennen, ob die gesendeten Daten beim Empfänger fehlerfrei angekommen sind.

Eigenschaften Prozedur 3964(R)

- Beim Senden werden den Daten Steuerzeichen hinzugefügt (Start-, Ende- und Blockprüfzeichen). Dabei müssen Sie beachten, dass diese Steuerzeichen nicht als Daten in dem Telegramm vorhanden sind.
- Der Verbindungsauf- und -abbau erfolgt über Steuerzeichen.
- Bei Übertragungsfehlern wird die Datenübertragung automatisch wiederholt.

Datenaustausch über Freeport- bzw. 3964(R)-Kommunikation

Die Sendedaten werden im Anwenderprogramm der zugehörigen CPU in Datenbausteinen (Sendepuffer) abgelegt. Für die Empfangsdaten steht im Kommunikationsmodul ein Empfangspuffer zur Verfügung. Kontrollieren Sie die Eigenschaften des Empfangspuffers und passen Sie diese gegebenenfalls an. In der CPU müssen Sie einen Datenbaustein für den Empfang anlegen.

Im Anwenderprogramm der CPU übernehmen die Anweisungen "Send_P2P" und "Receive_P2P" den Datentransfer zwischen CPU und CM.

Vorgehen zum Einrichten von Freeport- bzw. 3964(R)-Kommunikation

1. Konfigurieren Sie in der Gerätesicht des Hardware- und Netzwerkeditors von STEP 7 einen S7-1500-Aufbau mit CPU und CM.
2. Selektieren Sie die Schnittstelle des CM in der Gerätesicht von STEP 7.
3. Parametrieren Sie die Schnittstelle (z. B. Anschlusskommunikation, Konfiguration des Nachrichtensendens) im Inspektorfenster von STEP 7 unter "Eigenschaften" >"Allgemein".
4. Wählen Sie in der Task Card "Anweisungen" unter "Kommunikation" > "Kommunikationsprozessor" die Anweisungen "Send_P2P" bzw. "Receive_P2P" und ziehen Sie sie die Anweisung per Drag & Drop in das Anwenderprogramm (z. B. in einen FB).
5. Parametrieren Sie die Anweisungen entsprechend Ihren Vorgaben.
6. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Alternative: Dynamische Parametrierung des Kommunikationsmoduls

In bestimmten Anwendungsbereichen ist es vorteilhaft, die Kommunikation dynamisch, d. h. über eine spezifische Applikation programmgesteuert, einzurichten.

Typische Anwendungsfälle finden sich z. B. bei Herstellern von Serienmaschinen. Um ihren Kunden möglichst komfortable Bedienoberflächen anzubieten, passen diese Hersteller die Kommunikationsdienste an die jeweiligen Bedieneingaben an.

Anweisungen für Freeport-Kommunikation

Es stehen Ihnen 3 Anweisungen für die dynamische Projektierung im Anwenderprogramm für Freeport-Kommunikation zur Verfügung. Für alle 3 Anweisungen gilt: die bisher gültigen Konfigurationsdaten werden überschrieben, aber nicht dauerhaft im Zielsystem gespeichert.

- Die Anweisung "Port_Config" dient der programmgesteuerten Konfiguration des entsprechenden Ports des Kommunikationsmoduls.
- Die Anweisung "Send_Config" dient der dynamischen Projektierung von z. B. Zeitabständen und Pausen bei der Übertragung (serielle Übertragungsparameter) für den entsprechenden Port.
- Die Anweisung "Receive_Config" dient der dynamischen Projektierung von z. B. Bedingungen für Anfang und Ende einer zu übertragenden Nachricht (serielle Empfangsparameter) für den entsprechenden Port.

Anweisungen für 3964(R)-Kommunikation

Es stehen Ihnen 2 Anweisungen für die dynamische Projektierung im Anwenderprogramm für 3964(R)-Kommunikation zur Verfügung. Für die Anweisungen gilt: die bisher gültigen Konfigurationsdaten werden überschrieben, aber nicht dauerhaft im Zielsystem gespeichert.

- Die Anweisung "Port_Config" dient der programmgesteuerten Konfiguration des entsprechenden Ports des Kommunikationsmoduls.
- Die Anweisung "P3964_Config" dient der dynamischen Projektierung von Protokollparametern.

Eigenschaften USS-Protokoll

- Einfaches serielles Datenübertragungsprotokoll mit zyklischem Telegrammverkehr im Halbduplexbetrieb, das auf die Anforderungen in der Antriebstechnologie zugeschnitten ist.
- Die Datenübertragung funktioniert nach dem Master-Slave-Prinzip.
  - Der Master hat Zugriff auf die Funktionen des Antriebs und kann u. a. den Antrieb steuern, Statuswerte lesen sowie die Antriebsparameter lesen und schreiben.

Datenaustausch über USS-Kommunikation

Das Kommunikationsmodul ist der Master. Der Master sendet kontinuierlich Telegramme (Auftragstelegramme) an die bis zu 16 Antriebe und erwartet jeweils ein Antworttelegramm vom angesprochenen Antrieb.

Ein Antrieb sendet bei folgenden Bedingungen ein Antworttelegramm:

- Wenn ein Telegramm fehlerfrei empfangen wurde
- Wenn der Antrieb in diesem Telegramm adressiert wurde

Ein Antrieb darf nicht senden, wenn diese Bedingungen nicht erfüllt sind oder der Antrieb im Broadcast angesprochen wurde.

Für den Master besteht die Verbindung zu den betreffenden Antrieben dann, wenn er nach einer definierten Bearbeitungszeit (Antwortverzugszeit) vom Antrieb ein Antworttelegramm erhält.

Vorgehen zum Einrichten von USS-Kommunikation

1. Konfigurieren Sie in der Gerätesicht des Hardware- und Netzwerkeditors von STEP 7 einen S7-1500-Aufbau mit CPU und CM.
2. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie im Ordner den OB 1 durch Doppelklick. Der Programmeditor öffnet sich.
3. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation", Ordner "Kommunikationsprozessor" die Anweisungen für USS-Kommunikation entsprechend Ihrer Aufgabenstellung und ziehen Sie sie per Drag & Drop in ein Netzwerk des OB 1:
   - Die Anweisung "USS_Port_Scan" ermöglicht die Kommunikation über das USS-Netzwerk.
   - Die Anweisung "USS_Drive_Control" bereitet Sendedaten für den Antrieb vor und wertet die Antwortdaten des Antriebs aus.
   - Die Anweisung "USS_Read_Param" dient dem Auslesen von Parametern aus dem Antrieb.
   - Die Anweisung "USS_Write_Param" dient dem Ändern von Parametern im Antrieb.
4. Parametrieren Sie die Anweisungen entsprechend Ihren Vorgaben.
5. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Eigenschaften Modbus-Protokoll (RTU)

- Die Kommunikation läuft über serielle asynchrone Übertragungen mit einer Übertragungsgeschwindigkeit bis 115,2 kbit/s, halbduplex ab.
- Die Datenübertragung funktioniert nach dem Master-Slave-Prinzip.
- Der Modbus-Master kann Aufträge zum Lesen und Schreiben von Operanden an den Modbus-Slave senden:
  - Lesen von Eingängen, Zeiten, Zählern, Ausgängen, Merkern, Datenbausteinen
  - Schreiben von Ausgängen, Merkern, Datenbausteinen
- Broadcast an alle Slaves ist möglich.

Datenaustausch über Modbus-Kommunikation (RTU)

Das Kommunikationsmodul kann Modbus-Master oder Modbus-Slave sein. Ein Modbus-Master kann mit einem oder mehreren Modbus-Slaves kommunizieren (Anzahl hängt von der Schnittstellen-Physik ab). Nur der vom Modbus-Master explizit angesprochene Modbus-Slave darf Daten an den Modbus-Master zurücksenden. Der Slave erkennt das Ende der Datenübertragung und quittiert diese. Im Fehlerfall stellt er dem Master einen Fehlercode zur Verfügung.

Vorgehen zum Einrichten von Modbus-Kommunikation (RTU)

1. Konfigurieren Sie in der Gerätesicht des Hardware- und Netzwerkeditors von STEP 7 einen S7-1500-Aufbau mit CPU und CM.
2. Wählen Sie in der Projektnavigation für die CPU den Ordner "Programmbausteine" und öffnen Sie im Ordner den OB 1 durch Doppelklick. Der Programmeditor öffnet sich.
3. Wählen Sie aus der Task Card "Anweisungen", Bereich "Kommunikation", Ordner "Kommunikationsprozessor" die Anweisungen für Modbus-Kommunikation entsprechend Ihrer Aufgabenstellung und ziehen Sie sie per Drag & Drop in ein Netzwerk des OB 1:
   - Die Anweisung "Modbus_Comm_Load" konfiguriert den Port des CM für die Modbus-Kommunikation.
   - Die Anweisung "Modbus_Master" wird eingesetzt für die Modbus-Master-Funktionalität.
   - Die Anweisung "Modbus_Slave" wird eingesetzt für die Modbus-Slave-Funktionalität.
4. Parametrieren Sie die Anweisungen entsprechend Ihren Vorgaben.
5. Laden Sie die Hardware-Konfiguration und das Anwenderprogramm in die CPU.

Weitere Informationen

- Weitere Informationen zur Kommunikation über Punkt-zu-Punkt-Kopplung und Grundlagen der seriellen Datenübertragung finden Sie im Funktionshandbuch CM PtP - Konfigurationen für Punkt-zu-Punkt-Kopplungen (https://support.industry.siemens.com/cs/ww/de/view/59057093).
- Wie Sie die genannten Anweisungen für die Punkt-zu-Punkt-Kopplung im Anwenderprogramm nutzen, finden Sie in der Online-Hilfe STEP 7 beschrieben.
- Informationen zu den Kommunikationsmodulen mit serieller Schnittstelle finden Sie im jeweiligen Gerätehandbuch des Kommunikationsmoduls.

11

OPC UA-Kommunikation

11.1 Wissenswertes zu OPC UA

11.1.1 OPC UA und Industrie 4.0

Einheitlicher Standard für den Informations- und Datenaustausch

Industrie 4.0 steht für die intensive Nutzung, Auswertung und Analyse der zahlreichen Daten aus der Produktion in IT-Systemen der Unternehmensebene. Mit Industrie 4.0 nimmt der Datenaustausch zwischen Produktions- und Unternehmensebene sehr stark zu. Eine Voraussetzung für das Gelingen ist ein einheitlicher Standard zum Informations- und Datenaustausch.

Das klassische OPC läuft nur unter Windows-Betriebssystemen. Um diese Einschränkung zu umgehen, entwickelte die OPC Foundation den Standard OPC UA (OPC Unified Architecture).

Der Standard OPC UA ist wegen seiner Unabhängigkeit von bestimmten Betriebssystemen, seinem sicheren Übertragungsverfahren und der semantischen Beschreibung der Daten für den Ebenen übergreifenden Datenaustausch besonders geeignet. Auf diese Weise können auch die Maschinendaten (Regelgrößen, Messwerte oder Parameter) übertragen werden.

Wichtiger Bestandteil dieses Konzepts ist, dass für die zeitkritische maschinennahe Datenübertragung die OPC UA-Kommunikation parallel zu der Echtzeitkommunikation stattfinden kann.

OPC UA weist eine sehr hohe Skalierungsfähigkeit auf, so dass ein durchgängiger Informationsaustausch zwischen Sensoren, Steuerungen und MES- oder ERP-Systemen möglich ist.

OPC UA stellt nicht nur Daten bereit, sondern auch Informationen zu den Daten (z. B. Datentypen, Objekttypen), und damit wird ein maschineninterpretierbarer Zugriff auf die Daten möglich.

Themenseite OPC UA

Einen Überblick der wichtigsten Beiträge und Links zum Thema OPC UA finden Sie im Siemens Industry Online Support.
Themenseite OPC UA (https://support.industry.siemens.com/cs/ww/de/view/109770435)

11.1.2 Allgemeine Eigenschaften von OPC UA

OPC UA und PROFINET

OPC UA und PROFINET können gemeinsam genutzt werden. Beide Protokolle nutzen dieselbe Netzwerk-Infrastruktur.

Unabhängigkeit vom Betriebssystem

Der OPC UA Standard ist plattformunabhängig und verwendet ein optimiertes, TCP-basiertes Binärprotokoll für High-Performance Anwendungen.

OPC UA kann zum Beispiel unter Windows, Linux, Mac OS X, einem Echtzeitbetriebssystem oder einem mobilen Betriebssystem (Android oder iOS) eingesetzt werden.

Unabhängigkeit von einer bestimmten Transportschicht

OPC UA unterstützt aktuell die folgenden Transportmechanismen und Protokolle:

- Übertragen von Nachrichten als Binärstrom direkt über TCP/IP.
- Übertragen von Nachrichten mit XML über TCP/IP und HTTP. Dieser Transportmechanismus lässt nur eine langsame Übertragung zu und wird deshalb kaum verwendet. Die S7-1500 CPUs unterstützen diesen Transportmechanismus nicht.

Den binären Datenaustausch unterstützt jede OPC UA-Anwendung (durch die OPC UA-Spezifikation vorgeschrieben).

Einfaches Client-Server-Prinzip

Ein OPC UA-Server stellt innerhalb eines Netzwerks sehr viele Informationen bereit, z. B. zur CPU, zum OPC UA-Server selbst, zu den Daten und zu den Datentypen. Ein OPC UA-Client ruft diese Informationen ab.

Umsetzung in verschiedenen Programmiersprachen

Die OPC Foundation hat den Standard OPC UA in mehreren Programmiersprachen implementiert: Stacks für .NET, ANSI C, und Java sind verfügbar, wobei die Stacks für ANSI C und Java nicht mehr gewartet werden.

Die OPC Foundation bietet den .NET-Stack sowie Beispielprogramme als Open Source Software an. Siehe Github (https://github.com/opcfoundation).

Mehrere Unternehmen bieten Software Development Kits (SDK) an. Diese Entwicklungspakete enthalten die Stacks der OPC Foundation und weitere Funktionen, welche die Entwicklung von Lösungen erleichtern.

Vorteil der Nutzung von SDKs:

- Support durch den Zulieferer
- Getestete Software
- Ausführliche Dokumentation
- Klare Lizenzbedingungen (wichtig für Weiterverkauf von Lösungen)

Skalierbarkeit

OPC UA kann für Geräte verschiedener Leistungsklassen verwendet werden:

- Sensoren
- Eingebettete Systeme
- Steuerungen
- PC-Systeme
- Smartphones
- Server, auf denen MES- oder ERP-Anwendungen laufen.

Die Leistungsklasse der Geräte wird über Profile differenziert. Verschiedene OPC UA Profile bieten die Möglichkeit, OPC UA sowohl für sehr kleine und einfache Geräte als auch für sehr leistungsfähige Geräten zu skalieren.

Ein OPC UA Profil beschreibt Funktionen und Dienste, die vom Server und vom Client unterstützt werden müssen. Darüber hinaus können optional weitere Funktionen/Dienste erbracht werden, die nicht vom Profil gefordert werden.

OPC UA Profile unterscheiden sich von den PROFINET Profilen; letztere legen zusätzliche herstellerübergreifende Eigenschaften und Verhaltensweisen für Geräte und Systeme fest im Sinne einer herstellerneutralen Softwareschnittstelle.

Nano Embedded Device 2017 Server Profile

Für die kleinsten Geräte mit stark eingeschränkter Funktionalität gibt es das "Nano Embedded Device 2017 Server Profile" der OPC Foundation. Dieses Profil entspricht funktional der Core Server-Facette und definiert das binäre OPC UA-TCP-Protokoll als das erforderliche Transportprofil. Das Profil ermöglicht Verbindungen ohne UA-Security, keine Subscriptions und keine Methodenaufrufe. Die Unterstützung von Diagnoseobjekten und -variablen ist für dieses Profil optional.

Weitere Profile bauen auf dem "Nano Embedded Device 2017 Server Profile" auf, erfordern mehr Ressourcen und bieten mehr Funktionalität.

Micro Embedded Device 2017 Server Profile

Dieses Profil bietet eingeschränkte Funktionalität, es erfordert mindestens zwei parallele Verbindungen, zusätzlich Subscriptions/Datenmonitoring, aber keine UA-Security und keine Methodenaufrufe.

- Die Basic Controller S7-1200 unterstützen das "Micro Embedded Device 2017 Server Profile"; zusätzlich unterstützt die S7-1200 UA-Security.

Embedded 2017 UA Server Profile

Dieses Profil wurde für die Geräte mit mehr als 50 MB Arbeitsspeicher und einem leistungsfähigeren Prozessor entwickelt. Es baut auf dem Micro Embedded Device Server-Profil auf. Zusätzlich erfordert es UA-Security und Methodenaufrufe.

Zusätzlich müssen die Server ihr genutztes Typmodell (Datentypen, Referenztypen, Variablentypen, ...) verfügbar machen.

- Die Advanced Controller S7-1500 unterstützen das "Embedded 2017 UA Server Profile".

Standard- und Global-Discovery-Profile

Die "OPC UA Specification Part 7" definiert weitere Profile:

- Das "Standard 2017 UA Server Profile", das für die PC-basierte OPC UA-Server geeignet ist
- 2 globale Profile, "Global Discovery Server 2017 Profile" und "Global Discovery and Certificate Management 2017 Server Profile", die erforderliche Services- und Informationsmodelle eines Global Discovery Servers abdecken

Typ-Instanz-Konzept

OPC UA bietet ein voll vernetztes (full-meshed-network) objektorientiertes Informationsmodell für Namensräume, inklusive Metadaten zur Objektbeschreibung. Über die Referenzierung der Instanzen untereinander inklusive ihrer Typen sind beliebige Objektstrukturen erzeugbar. Da Server ihr Instanz- und Typsystem offenlegen, können Clients durch dieses Netz navigieren und sich alle erforderlichen Informationen beschaffen. Sowohl Instanzen als auch deren Typdefinitionen sind zur Laufzeit verfügbar.

Verfahren oder Konzepte, wie Referenzen auf Typen zu handhaben sind, werden im Laufe der Zeit optimiert. Diese Optimierungen führen zu neuen Versionen der OPC UA-Spezifikation (z. B. V1.03 => V1.04).

Abbildung der PLC-Variablen

Die Informationen des OPC UA-Servers (z. B. PLC-Variablen) sind als Knoten (Nodes) modelliert, die miteinander über Referenzen verbunden sind. Die Semantik wird vom Server im Adressraum angezeigt und kann von Clients (beim Navigieren) erfasst werden. Dadurch ist es möglich, mit einem OPC UA-Client von Knoten zu Knoten zu browsen und zu erfahren, welche Inhalte gelesen, beobachtet oder geschrieben werden können.

Integrierte Sicherheitsmechanismen

OPC UA verwendet Sicherheitsmechanismen auf unterschiedlichen Ebenen:

- Der Aufbau sicherer Verbindungen zwischen einem OPC UA-Server und einem OPC UA-Client ist nur möglich, wenn sich Client und Server mit Hilfe von X.509-v3 Zertifikaten anmelden können und gegenseitig die Zertifikate anerkennen (Sicherheit auf Anwendungsebene). Verschiedene Security Policys sind möglich, auch eine ungesicherte Verbindung zwischen Server und Client (Security Policy: "Keine Security").
- Ein Server kann grundsätzlich für den autorisierten Zugriff (Authentifizierung) folgende Informationen vom Anwender fordern:
  - ein Benutzerzertifikat (nicht in STEP 7 projektierbar)
  - Benutzernamen und Passwort
  - keine Legitimation des Anwenders

Die Sicherheitsmechanismen sind optional und konfigurierbar.

Weitere Informationen

Weitere Informationen finden Sie auf der Internetseite der OPC Foundation (https://opcfoundation.org).

11.1.3 OPC UA bei S7-1200/S7-1500 CPUs

Bei OPC UA arbeitet ein System als Server und stellt anderen Systemen (Clients) die vorhandenen Informationen zur Verfügung.

OPC UA-Clients greifen z. B. lesend und schreibend auf Daten eines OPC UA-Servers zu. OPC UA-Clients rufen Methoden im OPC UA-Server auf.

Sie können mit einem Client online auf diese Daten zugreifen, auch z. B. auf Informationen zur Leistungsfähigkeit und Diagnose. Im Sprachgebrauch von OPC UA heißt diese Funktion "Browsen". Die Funktion "Subscription" erspart das regelmäßige Lesen einer Variablen - ein Client wird dabei vom Server über Wertänderungen informiert.

Ein System kann gleichzeitig sowohl Client als auch Server sein.

OPC UA-Server der S7-1500 CPU

Ab Firmware V2.0 ist eine S7-1500 CPU mit einem OPC UA-Server ausgestattet.

Die folgenden Kapitel beschreiben, wie Sie den OPC UA-Server der S7-1500 CPU konfigurieren und damit Daten und Methoden für OPC UA-Clients zur Verfügung stellen, sodass Clients lesend oder schreibend auf PLC-Variablen der CPU zugreifen und Server-Methoden aufrufen können.

Außerdem zeigen die folgenden Kapitel, wie Sie Companion Spezifikationen in den Adressraum des OPC UA-Servers einbinden.

OPC UA-Server der S7-1200 CPU

Ab Firmware V4.4 ist eine S7-1200 CPU mit einem OPC UA-Server ausgestattet.

Der OPC UA-Server wird prinzipiell genauso konfiguriert wie bei einer S7-1500 CPU, Funktionsumfang und Mengengerüste sind gemäß dem unterstützten Profil "Micro Embedded Device 2017 Server Profile" eingeschränkt. Die Funktionen "Registered Read" und "Registered Write" stehen daher im Gegensatz zu einer S7-1500 CPU **nicht** zur Verfügung.

Ab Firmware V4.5 unterstützen S7-1200 CPUs sowohl Server-Methoden also auch strukturierte Datentypen (Strukturen und Arrays).

Weitere Informationen finden Sie in der Onlinehilfe von STEP 7.

OPC UA-Client der S7-1500 CPU

Ab Firmware V2.6 ist eine S7-1500 CPU zusätzlich auch mit einem OPC UA-Client ausgestattet.

Die folgenden Kapitel zeigen, wie Sie mit standardisierten Anweisungen (PLCopen-Funktionsbausteine) ein Anwenderprogramm erstellen, das als OPC UA-Client folgende Funktionen bereitstellt:

- Daten von einem OPC UA-Server lesen
- Daten zu einem OPC UA-Server schreiben
- Methoden eines OPC UA-Servers aufrufen

STEP 7 (TIA Portal) hilft Ihnen beim Erstellen von Anwenderprogrammen mit einem Editor für Client-Schnittstellen und einer Parametrierung für OPC UA-Verbindungen.

Die OPC UA-Anweisungen für eine S7-1500 CPU als Client sind ausführlich in der Hilfe zu den Anweisungen beschrieben (Anweisungen > Kommunikation > OPC UA).

OPC UA-Client für Testzwecke

Um den Umgang mit OPC UA-Clients zu verdeutlichen, verwendet die folgende Beschreibung unterschiedliche OPC UA-Clients:

- "UaExpert" von Unified Automation. Ein umfangreicher Client, der kostenlos verwendet werden kann:
  Link zum Download von UaExpert (https://www.unified-automation.com/downloads/opc-ua-clients.html)
- "UA Sample Client" der OPC Foundation. Dieser Client ist kostenlos verfügbar für Anwender, die bei der OPC Foundation registriert sind:
  Link zum Download des Beispiel-Clients der OPC Foundation (https://opcfoundation.org)

Anwendungsbeispiel im Industry Online Support

Der Siemens Industry Online Support stellt ein kostenloses Anwendungsbeispiel mit einer Client-API für verschiedene Anwendungsfälle zur Verfügung. Mit den Funktionen dieser Schnittstelle können Sie eigene OPC UA-Clients passend für Ihren Anwendungsfall erstellen. Um den Umgang mit der API zu vereinfachen, bieten wir Ihnen eine übergeordnete .NET-Helper-Klasse an.

Die Client-API setzt auf dem .NET OPC UA-Stack der OPC Foundation auf.

Das Anwendungsbeispiel zeigt z. B. den Aufbau von Verbindungen zwischen Server und Client. Das Beispiel zeigt ebenso das Lesen und Schreiben von PLC-Variablen.

Link zum Download: OPC UA .NET Client für den SIMATIC S7-1500 OPC UA Server (https://support.industry.siemens.com/cs/ww/de/view/109737901)

11.1.4 Zugang zu OPC UA-Applikationen

Im Folgenden sind die Zugriffsmöglichkeiten beschrieben, die eine S7-1500 CPU mit einer OPC UA-Applikation (Client oder Server) über einen CP in derselben Station hat. Außerdem wird gezeigt, wie diese Zugriffsmöglichkeiten mit der Funktion "IP-Forwarding" kombiniert werden können, um über eine S7-1500-Station auf Geräte eines anderen IP-Subnetzes zugreifen zu können.

Alle Einstellungen erreichen Sie über die Eigenschaften der CPU, Bereich "Erweiterte Konfiguration" im Inspektorfenster.

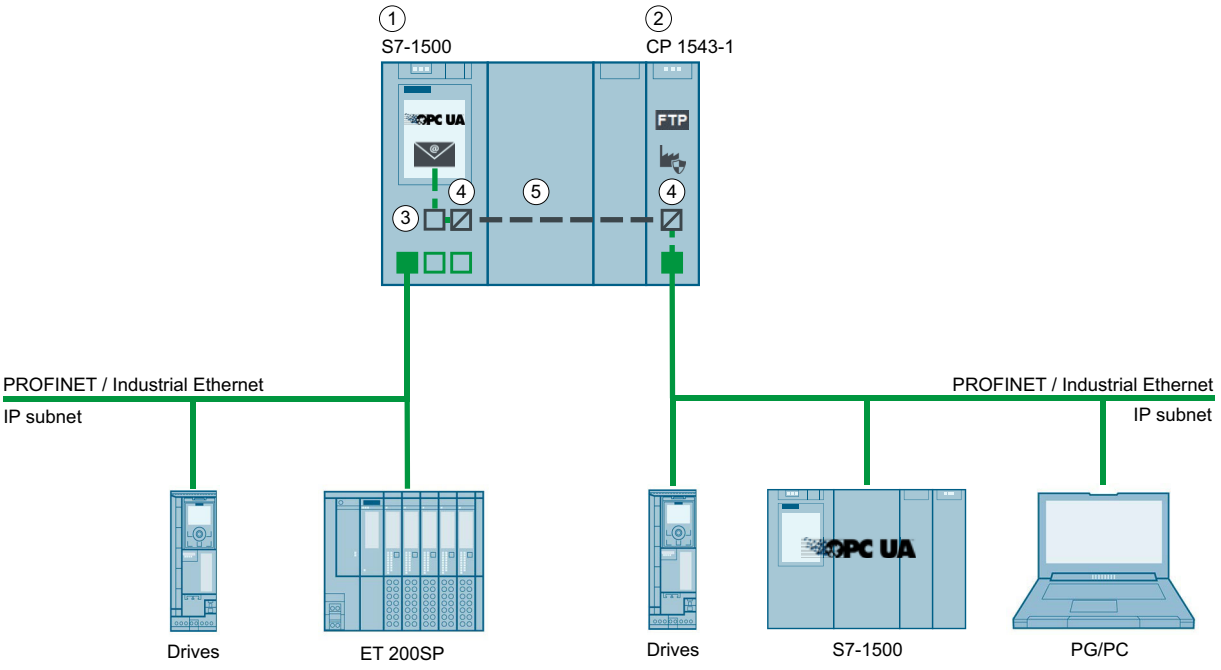
Die Option, über eine CP-Schnittstelle die OPC UA-Applikation in der CPU zu erreichen, ist an folgende Voraussetzungen gebunden:

- S7-1500 CPU (Advanced Controller) ab Firmware-Version V2.8, S7-1500R/H CPUs ab Firmware-Version V3.1
- CP 1543-1 ab Firmware-Version V2.2

Empfehlung: Verwenden Sie einen CP 1543-1 ab Firmware V3.0. Ab dieser Version sind die Security-Funktionen (Firewall) auch für die virtuelle Schnittstelle (W1) verfügbar und es muss keine zusätzliche Firewall zwischen der Station und einem unsicheren Netz installiert werden.

Prinzip: Schnittstelle für Zugriff über Kommunikationsmodul

Damit eine CPU-Applikation wie OPC UA über eine CP-Schnittstelle erreicht werden kann, müssen Sie eine virtuelle Schnittstelle (W1) projektieren. Über die IP-Adressparameter dieser virtuellen Schnittstelle können dann IP-basierende Applikationen erreicht werden.
Das Prinzip ist in folgendem Bild dargestellt.



① CPU S7-1500 ab FW V2.8 (z. B. CPU 1515-2 PN)

② CP 1543-1 (ab FW V2.2)

③ Virtuelle Schnittstelle (W1)

④ Protokollumsetzung PROFINET / Industrial Ethernet auf Rückwandbus bzw. Rückwandbus auf PROFINET / Industrial Ethernet

⑤ Rückwandbus

Bild 11-1 Prinzip: Schnittstelle für Zugriff über Kommunikationsmodul

Beispiel: Zugang von OPC UA Clients zum OPC UA-Server der CPU

Für den Zugang eines OPC UA-Clients auf den OPC UA-Server der CPU stehen Ihnen folgende Schnittstellen der S7-1500 Station zur Verfügung:

- Die lokalen PROFINET-Schnittstellen der S7-1500 CPU
- Die Ethernet-Schnittstelle eines CP 1543-1 (ab Firmware-Version V2.2)

Das folgende Bild zeigt beispielhaft eine mögliche Konfiguration. Die CPU könnte auch die Rolle OPC UA-Client haben und das Gerät am Subnetz des CP die Rolle OPC UA-Server.

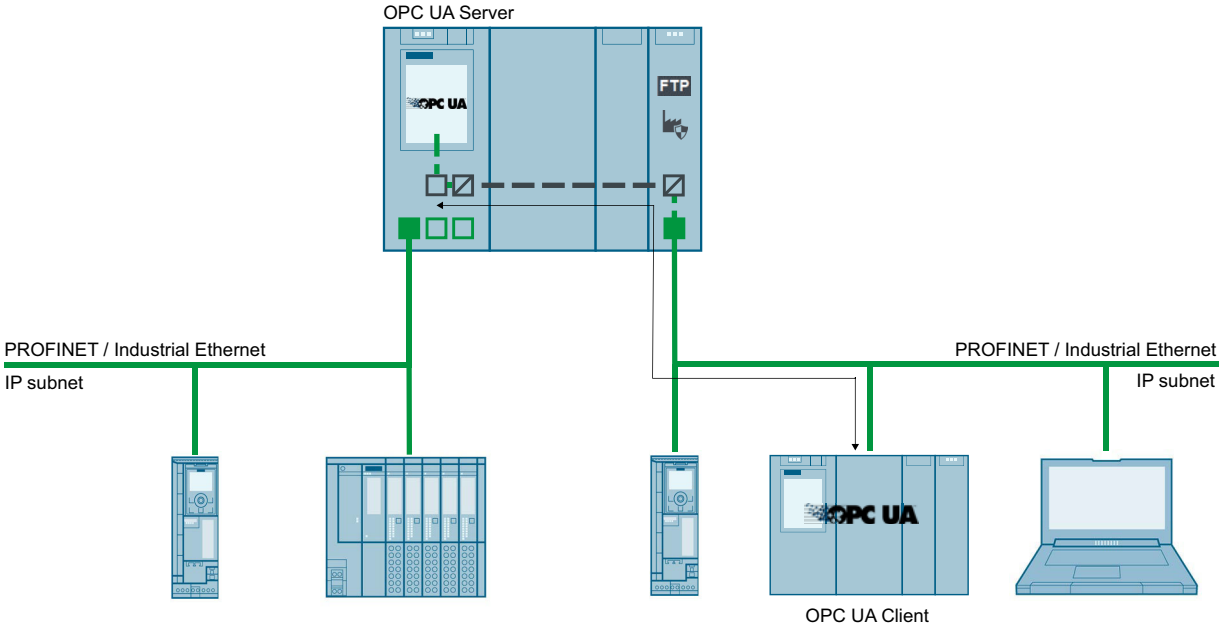


Bild 11-2 Beispiel: Zugang von OPC UA Clients zum OPC UA-Server der CPU

Beispiel: Zugang von OPC UA-Clients zu OPC UA-Servern über S7-1500 CPU mit aktiviertem IP-Forwarding

OPC UA-Client und OPC UA-Server können auch über eine S7-1500 CPU miteinander verbunden sein, wobei die S7-1500 CPU als IP-Forwarder arbeitet. Diese Konfigurationsmöglichkeit erlaubt eine flexible Erweiterung bestehender Anlagen.

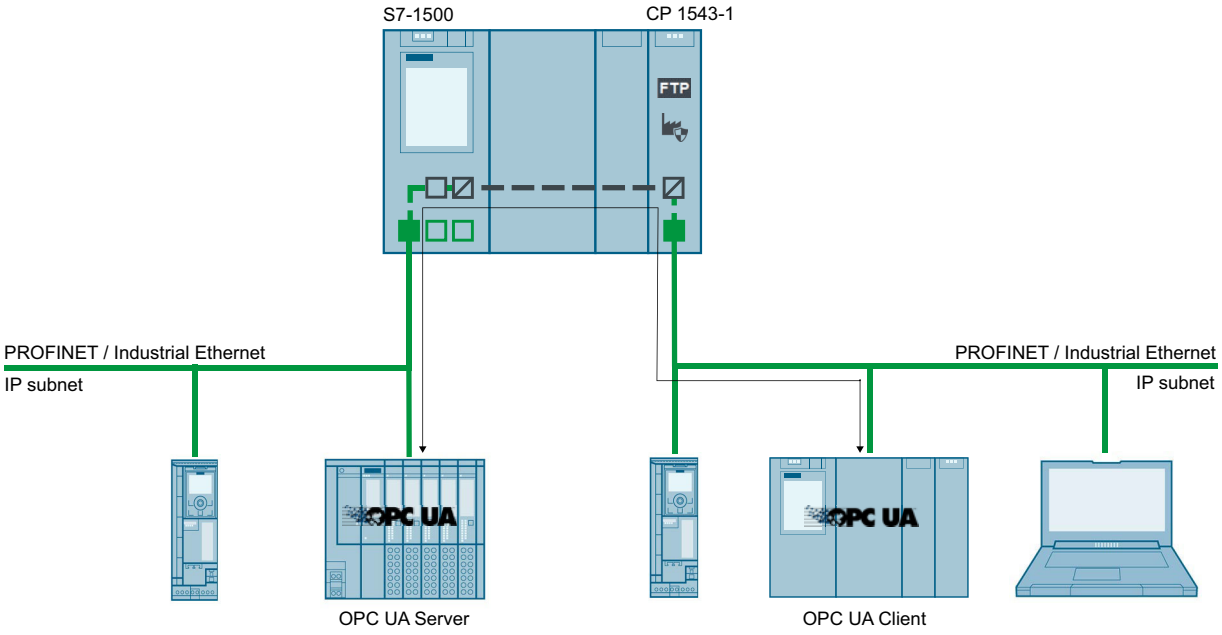


Bild 11-3 Beispiel: Zugang von OPC UA-Clients zu OPC UA-Servern über S7-1500 CPU mit aktiviertem IP-Forwarding

Weitere Informationen

Weitere Informationen zu den Zugriffsmöglichkeiten über die virtuelle Schnittstelle und über IP-Forwarding finden Sie in folgenden Abschnitten:

- IP-Forwarding (Seite 384)
- Virtuelle Schnittstelle für IP-basierte Anwendungen (Seite 393)

11.1.5 Adressierung von Knoten

Knoten (Nodes) sind die grundlegenden Elemente von OPC UA, sie sind vergleichbar mit Objekten aus der objektorientierten Programmierung. Knoten werden z. B. für Nutzdaten (Variablen) oder andere Metadaten verwendet. Mit Knoten wird ein OPC UA-Adressraum modelliert, der auch ein Typenmodell mit Typdefinitionen enthält.

Knoten-ID (NodeId)

Knoten im OPC UA-Adressraum werden durch eine NodeId (Node Identifier) eindeutig bestimmt.

Die NodeId besteht aus dem Identifier, Identifier Type und einem Namensraumindex. Namensräume werden verwendet, um Namenskonflikte zu vermeiden.

Die OPC Foundation hat eine Reihe von Knoten definiert, die Auskunft über den jeweiligen OPC UA-Server geben. Diese Knoten sind im Namensraum der OPC Foundation zu finden und besitzen den Index 0.

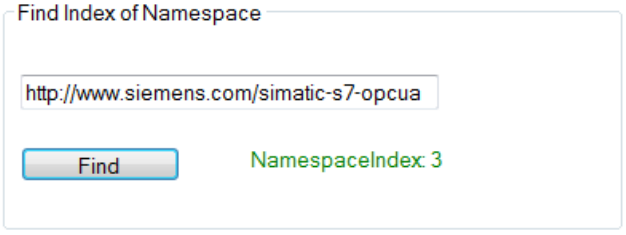
Weiterhin hat die OPC Foundation Daten- und Variablentypen definiert.

Namensraum (Namespace)

Neben dem oben beschriebenen Namensraum der OPC Foundation ist der Namensraum für den Zugriff auf CPU-Daten von Interesse: Alle Variablen bzw. Methoden eines S7-1500 OPC UA-Servers befinden sich im Namensraum (Namespace) der Standard-Server-Schnittstelle "http://www.siemens.com/simatic-s7-opcua".

Standardmäßig besitzt dieser Namensraum den Index 3. Wenn weitere Namensräume in den Server eingefügt oder vorhandene gelöscht werden, kann sich der Index ändern. Deshalb ist es erforderlich, dass ein OPC UA-Client den aktuellen Index des Namensraums (z. B. "http://www.siemens.com/simatic-s7-opcua") beim Server erfragt, bevor er dessen Werte liest oder schreibt.

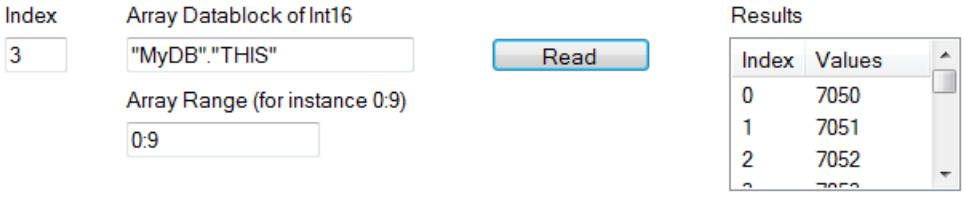
Das folgende Bild zeigt ein beispielhaftes Ergebnis einer solchen Anfrage.



Identifier

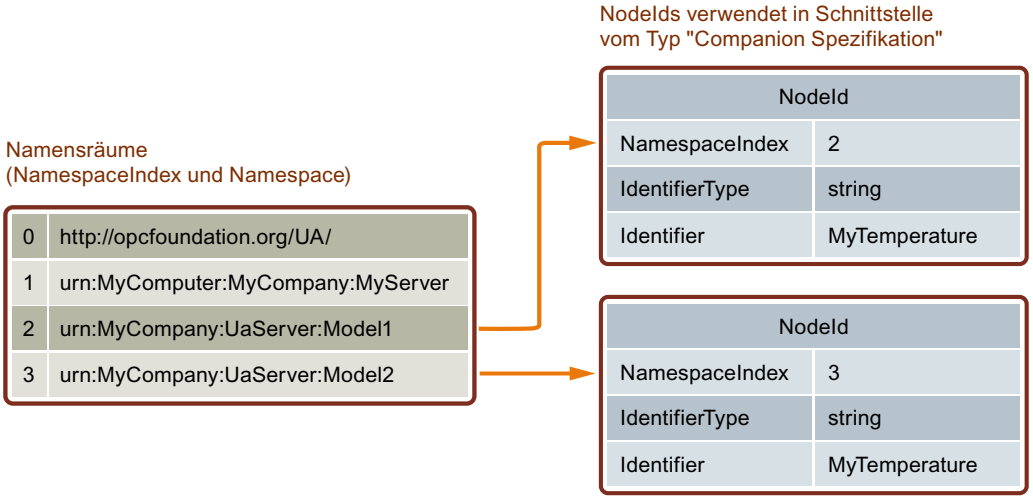
Der Identifier entspricht dem Namen der PLC-Variablen in Anführungszeichen. Das Anführungszeichen ist das einzige Zeichen, das in STEP 7 nicht als Namensbestandteil erlaubt ist. Durch die Anführungszeichen werden Namenskonflikte vermieden.
Das folgende Beispiel liest den Wert der Variablen "StartTimer":

Der Identifier kann aus mehreren Komponenten bestehen. Die einzelnen Komponenten sind dann durch einen Punkt getrennt.
Das folgende Beispiel liest den Array-Datenbaustein "MyDB" komplett ein. In diesem Datenbaustein befindet sich ein Array mit zehn Integer-Werten. Alle zehn Werte sollen auf einmal gelesen werden. Deshalb ist bei Array Range "0:9" eingetragen:



Beispiel für NodeIds, Identifier und Namensräume

Den Zusammenhang zwischen NodeIds, Identifier und Namensräumen verdeutlicht folgendes Bild: Es macht kein Problem, wenn zwei Knoten dieselben Identifier haben aber unterschiedlichen Namensräumen angehören.
STEP 7 (TIA Portal) erlaubt auf einfache Weise, Namensräume über eine Server-Schnittstelle zu importieren.



PLC-Variablen im Adressraum des OPC UA-Servers

Das folgende Bild zeigt, wo sich die PLC-Variablen des Beispiels im Adressraum des OPC UA-Servers befinden (Ausschnitt aus UA Client):
Der Datenbaustein "MyDB" ist ein globaler Datenbaustein. Deshalb befindet sich der Datenbaustein unterhalb des Knotens "DataBlocksGlobal". "StartTimer" ist eine Merker-Variable und wird deshalb unterhalb des Knotens "Memory" dargestellt.

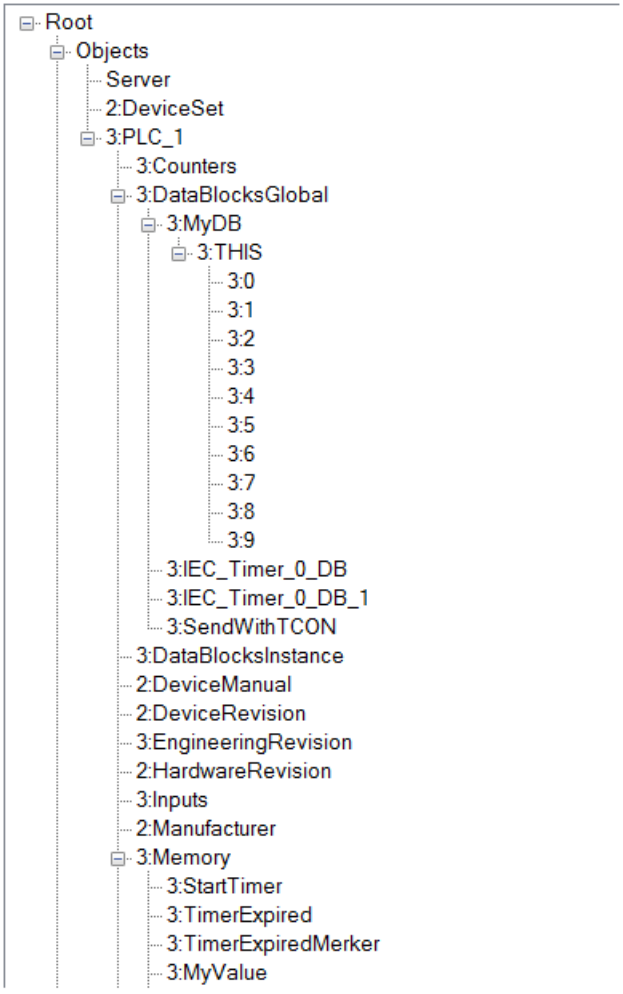


Bild 11-4 PLC-Variablen im Adressraum des OPC UA-Servers

Methoden im Adressraum des OPC UA-Servers

Wenn Sie über Ihr Anwenderprogramm eine Methode implementieren, dann sieht das im Adressraum des OPC UA-Servers folgendermaßen aus
(siehe Methoden auf dem OPC UA-Server bereitstellen (Seite 290)):

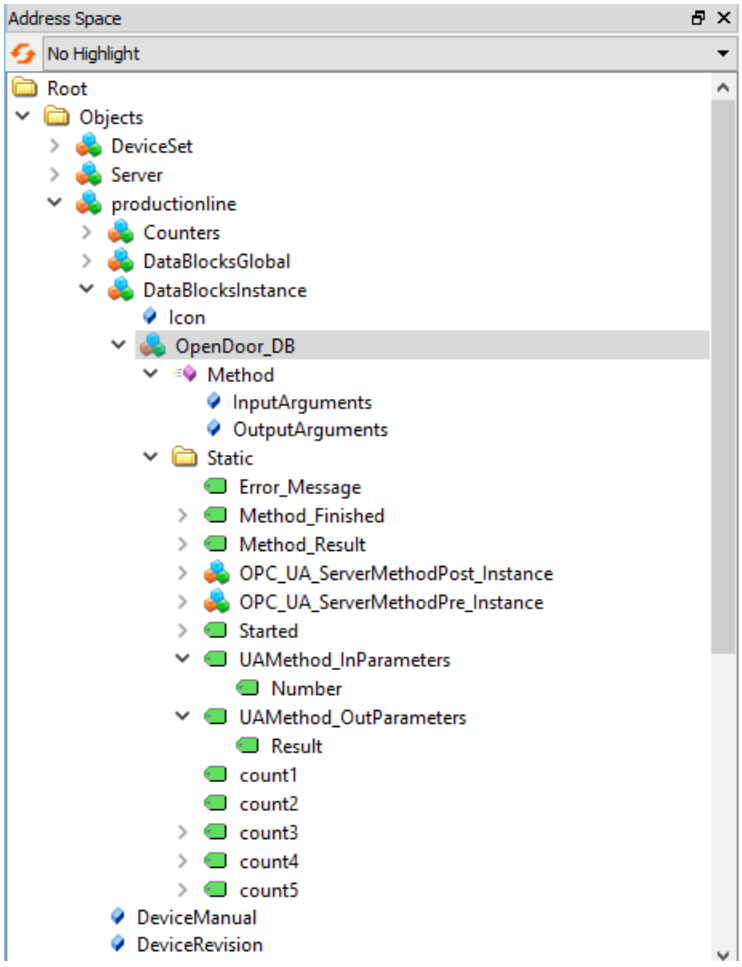


Bild 11-5 Methoden im Adressraum des OPC UA-Servers

11.1.6 Wissenswertes zu OPC UA-Clients

Grundlagen zu OPC UA-Clients

OPC UA-Clients sind Programme, die Folgendes leisten:

- Zugriff auf Informationen von einem OPC UA-Server (z. B. einer S7-1500 CPU): lesend/browsend, schreibend, Subscriptions
- Methoden durch den OPC UA-Server ausführen lassen

OPC UA-Clients können jedoch nur auf Daten zugreifen, die dafür freigegeben sind (siehe "Schreib- und Leserechte verwalten (Seite 222)").

Um eine Verbindung zu einem OPC UA-Server aufzubauen, benötigen Sie den Endpunkt des Servers (siehe "Endpunkte der OPC UA-Server (Seite 213)").

Informationen aus dem OPC UA-Server auslesen

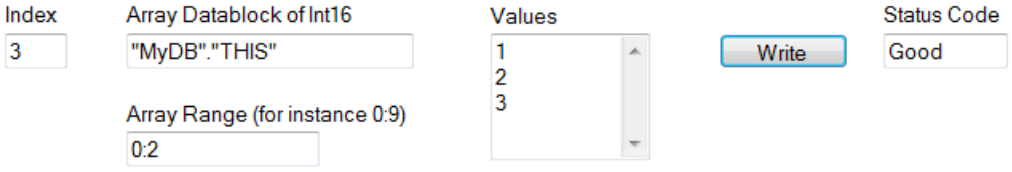
Wenn eine Verbindung zu einem Endpunkt des Servers besteht, dann können Sie die Navigationsfunktion des Clients nutzen: Sie navigieren von einem definierten Ausgangspunkt ausgehend (vom Wurzelknoten "Root" aus) durch den Adressraum des Servers.
Dadurch erhalten Sie unter anderem die folgenden Informationen:

- Freigegebene PLC-Variablen, Datenbausteine und Datenbaustein-Komponenten
- Namensraumindex und Identifier dieser PLC-Variablen, Datenbausteine und DB-Komponenten
- Datentypen der PLC-Variablen und DB-Komponenten
- Anzahl von Komponenten in Arrays (fürs Lesen und Schreiben von Arrays erforderlich)

Darüber hinaus erhalten Sie Information über den OPC UA-Server selbst, sowie Informationen über die S7-1500 basierend auf dem Standard "OPC UA for Devices" der OPC Foundation, zum Beispiel Seriennummer, Firmware-Version.

Daten vom Server lesen und zum Server schreiben

Sie kennen nun den Namensraumindex, Identifier und Datentyp von PLC-Variablen. Damit können Sie gezielt einzelne PLC-Variablen und DB-Komponenten wie auch ganze Arrays und Strukturen lesen.
Beispiele für das Lesen von booleschen Variablen und Array-Datenbausteinen finden Sie unter Knoten adressieren (Seite 172).
Regeln für den Zugriff auf Strukturen finden Sie hier (Seite 348).
Mit den Informationen, die Sie beim Navigieren durch den Adressraum des Servers erhalten (Index, Identifier und Datentyp), können Sie mit dem OPC UA-Client auch Werte in die S7-1500 übertragen. Das folgende Beispiel überschreibt im Array-Datenbaustein "MyDB" die ersten drei Werte.



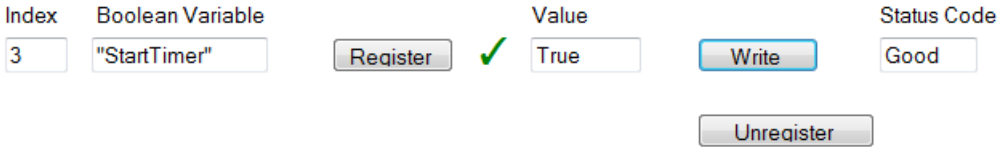
Bei "Array Range" geben Sie an, welche Komponenten des Arrays Sie überschreiben wollen. Am Status Code "Good" ist zu sehen, dass die Werte erfolgreich übertragen werden konnten. Sie können jedoch nur die Werte zur S7-1500 schreiben, nicht die Zeitstempel dieser Werte. Die Zeitstempel können nur gelesen werden.

Schnellerer Zugriff durch Registrierung

Registered Read/Write bietet sich an für den wiederholten, optimierten Zugriff auf Daten – bei höchster Performance. Beim Registrieren von Variablenknoten erstellt der OPC UA-Server einen numerischen Identifier (numerische NodeId), der direkt auf den registrierten Knoten verweist. Bei Lese- oder Schreibaufträgen des Clients auf diesen numerischen Identifier muss der Server keine Zeichenketten als Identifier auflösen und kann optimiert auf die geforderte Variable zugreifen.

Dieser Identifier gilt ausschließlich für die aktuelle Session und muss im Falle des Abbaus/Verlustes der Session neu erfragt werden.

Beim folgenden Beispiel wurde zunächst die Variable "StartTimer" beim Server registriert. Anschließend wird für das Setzen des Wertes die schnelle Funktion "RegisteredWrite" genutzt.



Nach dem gleichen Schema lässt sich auch die Funktion "RegisteredRead" nutzen, was insbesondere beim wiederkehrendem Auslesen von Daten sinnvoll ist. Beachten Sie jedoch, dass es je nach Anwendung sinnvoller sein kann, eine Subscription zu verwenden.

Empfehlung: Registrierungen platzieren Sie am besten im Hochlauf-Programm des OPC UA-Clients, da die Registrierung Zeit in Anspruch nimmt.

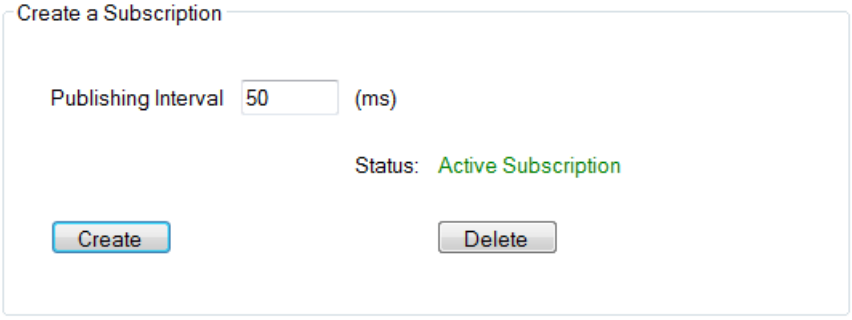
Beachten Sie, dass Sie in den Eigenschaften der S7-1500 CPU die maximale Anzahl registrierter Knoten einstellen können und die Clients diese Anzahl berücksichtigen müssen, siehe Allgemeine Einstellungen des OPC UA-Servers (Seite 237).

Subscription

Mit "Subscription" wird eine Funktion bezeichnet, bei der nur Variablen übertragen werden, für die sich ein OPC UA-Client beim OPC UA-Server angemeldet hat. Der OPC UA-Server sendet für diese angemeldeten Variablen (überwachte Elemente = monitored Items) nur dann eine Nachricht an den OPC UA-Client, wenn sich ein Wert geändert hat. Durch die Überwachung dieser Variablen entfällt ein ständiges Abfragen durch den OPC UA-Client (Polling); die Netzlast wird reduziert.

Um diese Funktion zu nutzen, müssen Sie eine Subscription anlegen. Dazu geben Sie beim UA-Client das Sendeintervall ("Publishing Interval") vor und klicken auf die Schaltfläche "Create". Das Sendeintervall ist das Zeitintervall, in dem der Server neue Werte in einer Benachrichtigung (data change notification) an den Client sendet.

Beim folgenden Beispiel wurde eine Subscription angelegt: Alle 50 Millisekunden erhält hier der Client eine Nachricht mit den neuen Werten (Sendeintervall 50 ms).



Server vor Überlast schützen

Den OPC UA-Server der S7-1500 CPU können Sie mithilfe des Parameters "Kleinstes Sendeintervall" so einstellen, dass er nicht extrem kurze, vom Client gewünschte Sendeintervalle bedient, siehe Einstellungen des Servers für Subscriptions (Seite 239).

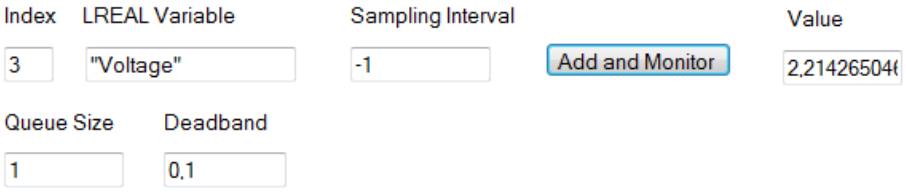
Beispiel: Ein Client möchte wie oben beschrieben in einem Sendeintervall von 50 ms bedient werden. Ein so kurzes Sendeintervall würde aber eine hohe Netzlast und eine hohe Belastung des Servers hervorrufen. Daher stellen Sie als "Kleinstes Sendeintervall" beim Server 1000 ms ein. Clients, die kürzere Sendeintervalle in ihrer Subscription fordern, werden so auf 1000 ms "heruntergebremst", der Server vor Überlastung geschützt.

Abtastung und Senden (Sampling & Publishing) im Rahmen einer Subscription sind Kommunikationsprozesse, die ebenso wie andere Kommunikationsprozesse (TCP/UDP/Webserver-Kommunikation ...) mit der Priorität 15 von der CPU abgearbeitet werden. OBs mit höherer Priorität unterbrechen die Kommunikation. Wenn Sie die Abtast- und Sendeintervalle zu kurz einstellen, verursacht diese Einstellung eine hohe Kommunikationslast. Wählen Sie daher möglichst große Intervalle, die für die Anwendung noch ausreichend sind.

Informationen zur Konsistenz von Variablen finden Sie in Konsistenz von CPU-Variablen (Seite 228).

Überwachung von PLC-Variablen

Wenn die Subscription angelegt ist, teilen Sie dem Server mit, welche Variablen er damit überwachen soll. Im folgenden Beispiel wurde die Variable "Voltage" der Subscription hinzugefügt.



Die Variable "Voltage" enthält z. B. den Wert einer Spannungsgröße, die von einer S7-1500 CPU erfasst wird.

Das Abtastintervall ("Sampling Interval") enthält einen negativen Wert (-1). Dadurch wird festgelegt, dass die Default-Einstellung des OPC UA-Servers für das Abtastintervall verwendet wird.

Die Default-Einstellung ist durch das Sendeintervall ("Publishing Interval") der Subscription festgelegt. Wenn Sie das kleinstmögliche Abtastintervall einstellen wollen, dann wählen Sie den Wert "0".
Die Länge der Warteschlange ist hier im Beispiel auf "1" festgelegt: Es wird immer nur ein Wert im Intervall von 50 Millisekunden aus der CPU gelesen und anschließend an den OPC UA-Client gesendet, wenn sich der Wert verändert hat.
Der Parameter "Deadband" ist im Beispiel "0,1": Wertänderungen müssen mindestens 0,1 Volt betragen, nur dann sendet der Server den neuen Wert an den Client. Kleinere Wertänderungen sendet der Server nicht. Mit diesem Parameter können Sie zum Beispiel Rauschen ausblenden: geringfügige Änderungen einer Prozessgröße, denen keine reale Bedeutung zukommt.

11.1.7 Mapping von Datentypen

SIMATIC- und OPC UA-Datentypen

SIMATIC-Datentypen stimmen nicht immer mit OPC UA-Datentypen überein.
S7-1500 CPUs stellen SIMATIC-Variablen (mit SIMATIC-Datentypen) dem eigenen OPC UA-Server als OPC UA-Datentypen bereit. OPC UA-Clients können dann über die Server-Schnittstelle auf diese Variablen mit OPC UA-Datentypen zugreifen.
Ein Client kann von einer solchen Variablen das Attribut "DataType" lesen und darüber den Original-Datentyp in SIMATIC rekonstruieren.

Beispiel

Eine Variable hat den SIMATIC-Datentyp "COUNTER". Sie lesen in der Tabelle COUNTER → UInt16. Sie wissen nun, dass Sie nicht umkodieren müssen, der COUNTER-Wert geht als UInt16 Datentyp über die Leitung.
Der Client erkennt am Attribut "DataType", dass die Variable eigentlich vom SIMATIC-Datentyp "COUNTER" ist. Mit diesem Wissen rekonstruiert der Client den Datentyp.

Tabelle 11-1 SIMATIC- und OPC UA-Datentypen

SIMATIC-Datentyp	OPC UA-Datentyp
BOOL	Boolean
BYTE	BYTE → Byte
WORD	WORD → UInt16
DWORD	DWORD → UInt32
LWORD	LWORD → UInt64
SINT	SByte
INT	Int16
DINT	Int32
LINT	Int64
USINT	Byte
UINT	UInt16
UDINT	UInt32

SIMATIC-Datentyp	OPC UA-Datentyp
ULINT	UInt64
REAL	Float
LREAL	Double
S5TIME	S5TIME → UInt16
TIME	TIME → Int32
LTIME	LTIME → Int64
DATE	DATE → UInt16
TIME_OF_DAY (TOD)	TOD → UInt32
LTIME_OF_DAY (LTOD)	LTOD → UInt64
DATE_AND_TIME (DT)	DT → Byte[8]
LDT	DateTime
DTL Besonderheit: Die Struktur können Sie mit einem OPC UA-Client nur komplett beschreiben. Lesend können Sie auf einzelne Elemente dieser Struktur zugreifen (z. B. auf "YEAR")	als Struktur gemappt
CHAR	CHAR → Byte
WCHAR	WCHAR → UInt16
STRING (Codepage 1252 bzw. Windows-1252)	STRING → String
WSTRING (UCS-2; Universal Coded Character Set)	String
TIMER	TIMER → UInt16
COUNTER	COUNTER → UInt16

Arrays

Ein Lese- bzw. Schreibauftrag bei OPC UA ist immer ein Array-Zugriff, d. h. grundsätzlich mit Index und Länge versehen. Eine Einzelvariable ist ein Sonderfall eines Arrays (Index 0 und Länge 1). Auf der Leitung wird der Datentyp einfach mehrfach hintereinander gesendet. Bei der Variablen zeigt das Attribut "DataType" auf den Basisdatentyp. Aus den Attributen "ValueRank" und "ArrayDimensions" ergibt sich, ob es sich um ein Array handelt und wie groß das Array ist.

Auf Arrays basierende Datentypen

Es gibt SIMATIC-Datentypen, bei denen ein OPC UA-Wert auf ein Array von Bytes abgebildet wird. Ein Array dieser Datentypen wird dann auf ein zweidimensionales Array abgebildet.

Beispiel: Der SIMATIC-Datentyp DATE_AND_TIME (DT) wird auf ein 8-Byte-Array (Byte[8]) gemappt, siehe Tabelle oben. Wenn Sie ein Array des SIMATIC-Datentyps DATE_AND_TIME (DT) definieren, dann zählt es als zweidimensionales Array.

Auswirkungen hat diese Tatsache z. B. auf die Verwendung von Systemdatentypen wie OPC_UA_NodeAdditionalInfo und OPC_UA_NodeAdditionalInfoExt:

Für die oben beschriebenen Datentypen müssen Sie den Systemdatentyp OPC_UA_NodeAdditionalInfoExt für mehrdimensionale Arrays verwenden statt OPC_UA_NodeAdditionalInfo.

Strukturen

Strukturen werden als ExtensionObject übertragen. Der Server der S7-1500 nutzt die binäre Darstellung für die Übertragung des ExtensionObjects über die Leitung, wobei die einzelnen Strukturelemente direkt hintereinanderliegen. Vorne befindet sich die NodeId des Datentyps, mit deren Hilfe ein Client den Aufbau der Struktur herausfindet.

Bei der OPC UA Specification <= V1.03 muss ein Client dazu das komplette DataTypeDictionary lesen, dekodieren und interpretieren (sofern er es nicht bereits vorher Offline durch einen XML-Import gelernt hat).

Ab OPC UA V1.04 gibt es dazu das Attribut DataTypeDefinition, das leichter und schneller zu lesen und zu interpretieren ist. Ein Client ermittelt den Aufbau der Struktur nur einmal, vor bzw. während des ersten Zugriffs, und nutzt diese Information dann für die Dauer der Session.

Spezielle Simatic-Datentypen

Simatic-Datentypen, die weder in der Tabelle oben vorkommen noch als Elemente einer Struktur bzw. eines PLC-Datentyps definiert werden können, werden vom OPC UA-Client nicht unterstützt.

Es handelt sich z. B. um Zeiger "ANY" oder "POINTER", Funktionsbaustein "Block_FB", Funktion "Block_FC" oder Hardwaredatentyp "REMOTE".

Die Auswahl eines nicht unterstützten Datentyps führt zu einer Fehlermeldung.

Weitere Information

Nähere Informationen zur Abbildung der Basisdatentypen, aber auch von Arrays und Strukturen, finden Sie in der OPC UA Spezifikation Part 6, "Mappings" (siehe dort "OPC UA BINARY").

Was müssen Sie bei Arrays und Datentypen DTL und LDT im OPC UA-Server einer SIMATIC S7-1500 beachten? FAQ (https://support.industry.siemens.com/cs/ww/de/view/109766726)

11.2 Security bei OPC UA

11.2.1 Security-Einstellungen

Gefahren begegnen

OPC UA erlaubt den Datenaustausch zwischen unterschiedlichen Systemen, sowohl innerhalb der Prozess- und Produktionsebene, als auch zu Systemen der Leit- und Unternehmensebene.

Diese Möglichkeit birgt auch Security-Risiken. Deshalb bietet OPC UA eine Reihe von Sicherheitsmechanismen:

- Prüfung der Identität von OPC UA-Server und -Clients.
- Prüfung der Identität der Anwender.
- Signierter/verschlüsselter Datenaustausch zwischen OPC UA-Server und -Clients.

Die Sicherheitseinstellungen sollten nur in begründeten Fällen umgangen werden:

- Während der Inbetriebnahme
- Bei Inselprojekten ohne Ethernet-Verbindung nach außen

Wenn Sie z. B. beim "UA Sample Client" der OPC Foundation den Endpunkt "None" auswählen, dann gibt das Programm eine deutliche Warnung aus:

Ebenso prüft STEP 7 beim Übersetzen Ihres Projekts, ob Sie die Einstellungsmöglichkeiten für den Schutz berücksichtigt haben und warnt bei möglichen Risiken. Dazu gehört auch eine OPC UA Security Policy mit der Einstellung "keine Security", was dem Endpunkt "None" entspricht.

HINWEIS

Nicht erwünschte Security Policys deaktivieren

Wenn Sie bei den Secure-Channel-Einstellungen des S7-1500 OPC UA-Servers alle Security Policys aktiviert haben - also auch den Endpunkt "None" (Keine Security) - dann ist der Datenverkehr zwischen Server und Client auch ungesichert möglich (weder signiert noch verschlüsselt). Der OPC UA-Server der S7-1500 CPU sendet auch bei "None" (Keine Security) sein öffentliches Zertifikat an den Client. Und manche Clients prüfen dieses Zertifikat. Doch der Client ist nicht gezwungen, ein Zertifikat an den Server zu senden. Die Identität des Clients bleibt möglicherweise unbekannt. Jeder OPC UA-Client kann sich dann mit dem Server verbinden, unabhängig von sämtlichen noch folgenden Security-Einstellungen.

Achten Sie bei der Projektierung des OPC UA-Servers darauf, dass nur Security Policys aktiviert sind, die mit dem Schutzkonzept für Ihre Maschine oder Anlage vereinbar sind. Alle anderen Security Policys sind zu deaktivieren.

Empfehlung: Verwenden Sie die Einstellung "Basic256Sha256 - Signieren & Verschlüsseln", bei der der Server nur Sha256-Zertifikate akzeptiert. Die Security Policys "Basic128Rsa15" und "Basic256" sind in der Voreinstellung deaktiviert und sollten nicht als Endpunkt verwendet werden. Wählen Sie Endpunkte mit einer höheren Security-Policy.

Weitere Security-Regeln

- Nutzen Sie nur im Ausnahmefall den Endpunkt "None".
- Verwenden Sie nur im Ausnahmefall die "Gast-Authentifizierung" des Benutzers.
- Erlauben Sie nur dann den Zugriff auf PLC-Variablen und DB-Komponenten über OPC UA, wenn es tatsächlich erforderlich ist.
- Nutzen Sie die Listen vertrauenswürdiger Clients in den Einstellungen des S7-1500 OPC UA-Clients, um nur bestimmten Clients Zugriff zu erlauben.

11.2.2 Zertifikate gemäß X.509 der ITU

Bei OPC UA sind Sicherheitsmechanismen in mehreren Schichten integriert. Dabei spielen digitale Zertifikate eine wichtige Rolle. Ein OPC UA-Client kann nur eine gesicherte Verbindung zu einem OPC UA-Server aufbauen, wenn der Server das digitale Zertifikat des Clients akzeptiert und als vertrauenswürdig einstuft.

Siehe Kapitel Handling der Client- und Server-Zertifikate (Seite 241).

Außerdem muss auch der Client das Zertifikat des Servers prüfen und ihm vertrauen. Server und Client müssen sich ausweisen und beweisen, dass sie tatsächlich der sind, der sie zu sein behaupten: Sie müssen ihre Identität nachweisen. Die gegenseitige Authentifizierung von Client und Server verhindert zum Beispiel Angriffe durch einen "Man in the Middle".

Angriffe durch "Man in the Middle"

Zwischen Server und Client könnte sich ein "Man in the Middle" befinden, ein Programm, das die Kommunikation zwischen Server und Client abfängt und behauptet, selbst Client oder Server zu sein, und so wichtige Informationen über das S7-Progamm erhält oder Werte in der CPU setzt und damit eine Maschine oder Anlage angreifen kann.

Bei OPC UA werden digitale Zertifikate verwendet, die dem Standard X.509 der International Telecommunication Union (ITU) entsprechen.

Damit lässt sich die Identität eines Programms, eines Rechners oder einer Organisation nachweisen (authentifizieren).

X.509-Zertifikate

Ein X.509-Zertifikat enthält unter anderem die folgenden Informationen:

- Versionsnummer des Zertifikats
- Seriennummer des Zertifikats
- Informationen über den Algorithmus, den die Zertifizierungsstelle zum Signieren des Zertifikats verwendete.
- Name der Zertifizierungsstelle
- Beginn und Ende der Gültigkeit des Zertifikats
- Name des Programms, der Person oder Organisation, für die das Zertifikat von der Zertifizierungsstelle signiert wurde.
- Der öffentliche Schlüssel des Programms, der Person oder Organisation.

Somit verknüpft ein X509-Zertifikat eine Identität (Name eines Programms, einer Person oder einer Organisation) mit dem öffentlichen Schlüssel des Programms, der Person oder Organisation.

Prüfung beim Verbindungsaufbau

Beim Verbindungsaufbau zwischen Client und Server prüfen die Teilnehmer alle Informationen aus dem Zertifikat, die zur Feststellung der Integrität notwendig sind, z. B. Signatur, Gültigkeitsdauer, Applikationsname (URN) und bei Firmware Version V2.5 (nur bei dieser Version) auch die IP-Adressen des Clients im Client-Zertifikat.

HINWEIS

Außerdem wird beim Verbindungsaufbau der Gültigkeitszeitraum geprüft, der im Zertifikat hinterlegt ist. Die CPU-Uhr muss daher gestellt sein und Datum/Uhrzeit müssen sich im Gültigkeitszeitraum befinden, sonst findet keine Kommunikation statt.

Signieren und Verschlüsseln

Damit überprüft werden kann, ob ein Zertifikat manipuliert wurde, werden Zertifikate signiert.

Hier gibt es verschiedene Vorgehensweisen:

- Innerhalb des TIA Portals haben Sie die Möglichkeit, Zertifikate zu erzeugen und zu signieren. Wenn Sie Ihr Projekt geschützt haben und als Benutzer angemeldet sind mit dem Funktionsrecht, Security-Einstellung vornehmen zu dürfen, dann sind auch die globalen Security-Einstellungen nutzbar. Die globalen Security-Einstellungen erlauben Zugriff auf den Zertifikatsmanager und damit auch auf die Zertifizierungsstelle (CA) des TIA Portals.
- Ihnen stehen weitere Möglichkeiten für die Erzeugung und Signierung von Zertifikaten zur Verfügung. Im TIA Portal können Sie Zertifikate in den globalen Zertifikatsmanager importieren.
  - Sie wenden sich an eine Zertifizierungsstelle (CA) und lassen Ihr Zertifikat signieren.
    In diesem Fall überprüft die Zertifizierungsstelle Ihre Identität und signiert Ihr Zertifikat mit dem privaten Schlüssel der Zertifizierungsstelle. Senden Sie dazu einen CSR (Certificate Signing Request) an die Zertifizierungsstelle.
  - Sie erstellen selbst ein Zertifikat und signieren es.
    Dazu nutzen Sie zum Beispiel das Programm "Opc.Ua.CertificateGenerator" der OPC Foundation. Oder Sie verwenden OpenSSL.
    Weiterführende Informationen finden Sie unter PKI-Schlüsselpaare und Zertifikate selbst erzeugen (Seite 188).

Exkurs: Zertifikatstypen

- Selbst signiertes Zertifikat:
  Jeder Teilnehmer erzeugt sein eigenes Zertifikat und signiert es. Beispielanwendungen: Statische Konfiguration mit begrenzter Anzahl von Kommunikationsteilnehmern.
  Aus einem selbst signierten Zertifikat können keine neuen Zertifikate abgeleitet werden. Allerdings müssen Sie alle selbst signierten Zertifikate der Partnergeräte in die CPU laden (STOP erforderlich).
- CA-Zertifikat:
  Alle Zertifikate werden von einer Zertifizierungsstelle erstellt und signiert. Beispielanwendungen: Dynamisch wachsende Anlagen.
  Sie müssen nur das Zertifikat der Zertifizierungsstelle in die CPU laden. Die Zertifizierungsstelle kann neue Zertifikate erzeugen (Hinzufügen von Partnergeräten ohne STOP der CPU möglich).

Signieren

Durch die Signatur lässt sich die Integrität und Herkunft einer Nachricht nachweisen, wie im Folgenden beschrieben ist.
Beim Signieren bildet der Sender zunächst aus dem Klartext (Klarnachricht) einen Hashwert. Dann verschlüsselt der Sender den Hashwert mit seinem privaten Schlüssel und überträgt schließlich den Klartext zusammen mit dem verschlüsselten Hashwert zum Empfänger. Der Empfänger benötigt zur Überprüfung der Signatur den öffentlichen Schlüssel des Senders (ist im X509-Zertifikat des Senders enthalten). Mit dem öffentlichen Schlüssel des Senders entschlüsselt der Empfänger den erhaltenen Hashwert. Dann bildet der Empfänger selbst aus dem empfangenen Klartext den Hashwert (das Hashverfahren ist im Zertifikat des Senders enthalten). Anschließend vergleicht der Empfänger die beiden Hashwerte:

- Wenn die beiden Hashwerte gleich sind, dann ist die Klarnachricht unverändert beim Empfänger angekommen und wurde nicht manipuliert.
- Wenn die beiden Hashwerte nicht gleich sind, dann ist die Klarnachricht nicht identisch beim Empfänger angekommen. Die Klarnachricht wurde manipuliert oder bei der Übertragung verfälscht.

Verschlüsseln

Durch Verschlüsseln von Daten verhindern Sie, dass Unbefugte Kenntnis vom Inhalt erhalten. X509-Zertifikate werden nicht verschlüsselt; sie sind öffentlich und jedermann kann sie einsehen.
Beim Verschlüsseln verschlüsselt der Sender die Klarnachricht mit dem öffentlichen Schlüssel des Empfängers. Dazu benötigt der Sender das X509-Zertifikat des Empfängers, weil darin der öffentliche Schlüssel des Empfängers enthalten ist. Der Empfänger entschlüsselt die Nachricht mit seinem privaten Schlüssel. Nur der Empfänger kann die Nachricht entschlüsseln. Er allein besitzt den privaten Schlüssel. Deshalb darf der private Schlüssel nie weitergegeben werden.

Secure Channel

OPC UA verwendet die privaten und öffentlichen Schlüssel von Client und Server beim Aufbau einer gesicherten Verbindung, des Secure Channels. Wenn die gesicherte Verbindung aufgebaut ist, dann erzeugen Client und Server einen internen, nur ihnen bekannten Schlüssel, den sie zum Signieren und Verschlüsseln von Nachrichten verwenden. Dieses symmetrische Verfahren (ein gemeinsamer Schlüssel) ist sehr viel schneller als unsymmetrische Verfahren (private und öffentliche Schlüssel).

Weitere Informationen

Ein Anwendungsbeispiel für die Verwendung von Zertifikaten mit dem TIA Portal finden Sie hier: Verwendung von Zertifikaten mit TIA Portal (https://support.industry.siemens.com/cs/ww/de/view/109769068).

11.2.3 Zertifikate bei OPC UA

Verwendung von X.509-Zertifikaten bei OPC UA

OPC UA verwendet beim Aufbau einer Verbindung von Client zu Server verschiedene Arten von X.509-Zertifikaten:

- OPC UA Applikationszertifikate
  Solche X.509-Zertifikate identifizieren die Software-Instanz, die jeweilige Installation einer Client- oder Server-Software. Beim Attribut "Organisation Name" tragen Sie den Namen des Unternehmens ein, das die Software einsetzt.

  HINWEIS

  Der OPC UA-Server der S7-1500 verwendet Applikationszertifikate auch bei der Security-Einstellung "None" (Keine Security). Dadurch wird die Kompatibilität zu OPC UA V1.1 und früher gewahrt.

- OPC UA Software-Zertifikate
  Dieses X.509-Zertifikat identifiziert eine konkrete Version der Client- oder Server-Software. Solche Zertifikate enthalten Attribute, die beschreiben, welche Tests diese Version der Software bei der Zertifizierung durch die OPC Foundation (bzw. anerkannte Testlabors) bestanden hat. Beim Attribut "Organisation Name" tragen Sie den Namen des Unternehmens ein, das die Software entwickelt hat oder vertreibt.

  HINWEIS

  Bei STEP 7 werden keine Software-Zertifikate unterstützt.

- OPC UA Benutzerzertifikate
  Dieses X.509-Zertifikat identifiziert den konkreten Benutzer, der zum Beispiel von einem OPC UA-Server Prozessdaten abruft. Dieses Zertifikat ist nicht erforderlich, wenn der Benutzer seine Berechtigung mit seinem Passwort nachweisen kann oder wenn ein anonymer Zugang konfiguriert ist.

  HINWEIS

  Bei STEP 7 werden keine Benutzerzertifikate unterstützt.

Die beschriebenen Zertifikate sind End-Entity-Zertifikate: Sie identifizieren zum Beispiel eine Person, eine Organisation, ein Unternehmen, eine Instanz (Installation) einer Software.

11.2.4 Selbst-signierte Zertifikate erzeugen

Zertifikatsgenerator des Clients verwenden

Viele OPC UA-Client-Applikationen bzw. SDKs sind in eine Beispielanwendung eingebunden, die es Ihnen erlaubt, aus dieser Applikation heraus Zertifikate für den Client zu erzeugen.
Die Beschreibung zur Zertifikatserzeugung finden Sie im Allgemeinen im Kontext zur Beschreibung der OPC UA-Client-Applikation.

Beispiel-Client aus dem Online-Support
Der OPC UA .NET Client für den SIMATIC S7-1500 OPC UA-Server (https://support.industry.siemens.com/cs/ww/de/view/109737901) erstellt während des ersten Programmstarts ein selbst-signiertes Softwarezertifikat der Client-Applikation im Windows Certificate Store. Die Dokumentation zu diesem Beispiel beschreibt die Vorgehensweise zur Handhabung dieser Zertifikate.

Zertifikatsgenerator des TIA Portals verwenden

Wenn Sie einen OPC UA-Client verwenden, der kein Client-Zertifikat erzeugt, können Sie selbst-signierte Zertifikate mit STEP 7 erstellen.
Dazu gehen Sie folgendermaßen vor:

1. In den Eigenschaften der CPU, unter "Schutz & Security > Zertifikatsmanager > Gerätezertifikate" doppelklicken Sie auf "<Neu hinzufügen>"
2. Klicken Sie auf "Hinzufügen".
3. Im Dialog "Neues Zertifikat erzeugen" wählen Sie bei "Verwendungszweck" die Option "OPC UA-Client".
4. Klicken Sie auf "OK".

Im Feld "Alternativer Antragstellername" (Subject Alternative Name) trägt STEP 7 automatisch die URI für das erstellte Zertifikat ein. Bei der programmtechnischen Zertifikatserstellung über den .NET-Stack der OPC Foundation heißt das Feld z. B. "ApplicationUri", bei anderen Tools zur Zertifikatserstellung kann es anders heißen.

Weitere Informationen

Weitere Informationen zur Handhabung von Client-Zertifikaten finden Sie im Kapitel Handling der Client-Zertifikate der S7-1500 CPU (Seite 353).

11.2.5 PKI-Schlüsselpaare und Zertifikate selbst erzeugen

Dieses Kapitel ist für Sie nur dann relevant, wenn Sie einen OPC UA-Client verwenden wollen, der nicht selbst ein PKI-Schlüsselpaar und ein Client-Zertifikat erzeugen kann. Sie erzeugen in diesem Fall mit OpenSSL einen privaten und öffentlichen Schlüssel, generieren ein X.509-Zertifikat und signieren das Zertifikat selbst.

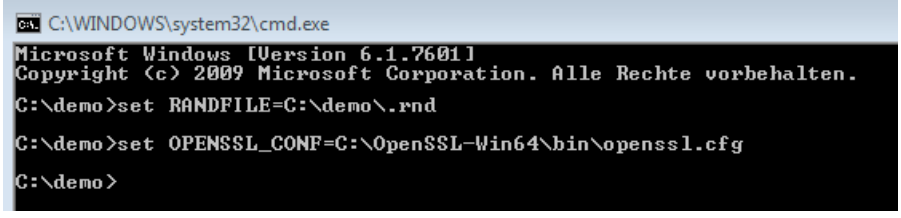
OpenSSL verwenden

OpenSSL ist ein Tool für Transport Layer Security, das Sie zum Erzeugen von Zertifikaten nutzen können. Sie können auch andere Tools verwenden, z. B. XCA, eine Schlüsselverwaltungssoftware mit grafischer Oberfläche für eine bessere Übersicht von ausgestellten Zertifikaten.

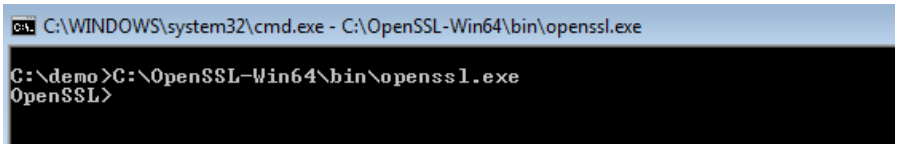
Um mit OpenSSL unter Windows zu arbeiten, gehen Sie folgendermaßen vor:

1. Installieren Sie OpenSSL unter Windows. Wenn Sie eine 64-Bit-Version des Betriebssystems verwenden, dann installieren Sie OpenSSL zum Beispiel im Verzeichnis "C:\OpenSSL-Win64". Sie erhalten OpenSSL-Win64 als Download bei verschiedenen Anbietern für Open Source Software.
2. Legen Sie ein Verzeichnis an, zum Beispiel "C:\demo".
3. Öffnen Sie die Kommandozeile (Eingabeaufforderung). Dazu klicken Sie auf "Start" und tragen im Suchfeld "cmd" oder "Eingabeaufforderung" ein. Klicken Sie in der Ergebnisliste mit der rechten Maustaste auf "cmd.exe" bzw. "Eingabeaufforderung" und führen Sie das Programm als Administrator aus. Windows öffnet die Eingabeaufforderung.
4. Wechseln Sie zum Verzeichnis "C:\demo". Dazu geben Sie den folgenden Befehl ein: "cd C:\demo".
5. Setzen Sie die folgenden Umgebungsvariablen:
   - set RANDFILE=c:\demo\.rnd
   - set OPENSSL_CONF=C:\OpenSSL-Win64\bin\openssl.cfg

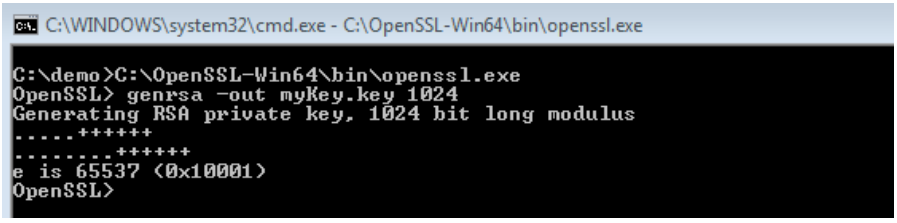
   Das folgende Bild zeigt die Kommandozeile mit den Befehlen:



6. Starten Sie nun OpenSSL. Wenn OpenSSL im Verzeichnis C:\OpenSSL-Win64 installiert wurde, dann geben Sie ein: C:\OpenSSL-Win64\bin\openssl.exe. Das folgende Bild zeigt die Kommandozeile mit dem Befehl:



7. Generieren Sie einen privaten Schlüssel. Speichern Sie den Schlüssel in die Datei "myKey.key". Der Schlüssel ist in diesem Beispiel 1024 Bit lang; für eine erhöhte Sicherheit von RSA verwenden Sie in der Praxis 2048 Bit! Geben Sie den folgenden Befehl ein: "genrsa -out myKey.key 2048" ("genrsa -out myKey.key 1024" im Beispiel). Das folgende Bild zeigt die Kommandozeile mit dem Befehl und der Ausgabe von OpenSSL:



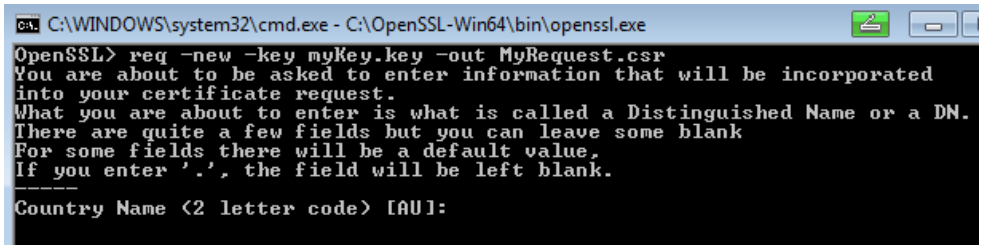
8. Erzeugen Sie einen CSR (Certificate Signing Request), eine Aufforderung, ein Zertifikat zu signieren. Dazu geben Sie den folgenden Befehl ein: "req -new -key myKey.key -out myRequest.csr". Während der Ausführung dieses Befehls fragt OpenSSL Sie nach Angaben zu Ihrem Zertifikat:
   - Country Name: z. B. "DE", für Deutschland, "FR" für Frankreich
   - State or Province Name: z. B. "Bayern".
   - Location Name: z. B. "Augsburg".
   - Organisation Name: Tragen Sie den Namen Ihres Unternehmens ein.
   - Organisational Unit Name: z. B. "IT"
   - Common Name: z. B. "OPC UA Client der Maschine A"
   - Email Address:

HINWEIS

Hinweis für S7-1500 CPU als Server mit Firmware-Version V2.5

Im Feld "Alternativer Antragstellername" (Subject Alternative Name) des erstellten Zertifikats muss bei S7-1500 CPUs Version V2.5 (nur bei dieser Version) die IP-Adresse des Client-Programms hinterlegt sein, sonst akzeptiert die CPU das Zertifikat nicht.

Ihre Angaben werden in das Zertifikat eingefügt. Das folgende Bild zeigt die Kommandozeile mit dem Befehl und der Ausgabe von OpenSSL:



Der Befehl legt im Verzeichnis C:\demo eine Datei an, die den Certificate Signing Request (CSR) enthält, im Beispiel "myRequest.csr".

Verwendung des CSR

Sie können einen CSR auf zwei Arten verwenden:

- Sie senden den CSR an eine Zertifizierungsstelle (CA): Beachten Sie dabei die Hinweise der jeweiligen Zertifizierungsstelle. Die Zertifizierungsstelle (CA) überprüft Ihre Angaben und Identität (Authentifizierung) und signiert das Zertifikat mit dem privaten Schlüssel der Zertifizierungsstelle. Sie erhalten das signierte X.509-Zertifikat und verwenden dieses Zertifikat zum Beispiel für OPC UA, HTTPS oder Secure OUC (secure open user communication). Ihre Kommunikationspartner überprüfen durch den öffentlichen Schlüssel der Zertifizierungsstelle, ob Ihr Zertifikat wirklich von dieser Stelle herausgegeben und signiert wurde (d. h. die Zertifizierungsstelle Ihre Angaben im Zertifikat bestätigt hat).
- Sie signieren den CSR selbst: Dazu verwenden Sie Ihren privaten Schlüssel. Diese Möglichkeit zeigt der nächste Schritt.

Zertifikat selber signieren

Damit Sie Ihr Zertifikat (Self-Signed Certificate) selber erzeugen und signieren können, geben Sie den folgenden Befehl ein: "x509 -req -days 365 -in myRequest.csr -signkey myKey.key -out myCertificate.crt".

Das folgende Bild zeigt die Kommandozeile mit dem Befehl und der Ausgabe von OpenSSL:

Der Befehl erzeugt ein X.509-Zertifikat mit den Attributen, die Sie mit dem CSR (im Beispiel "myRequest.csr") übergeben, z. B. mit einer Gültigkeit von einem Jahr (-days 365). Außerdem signiert der Befehl das Zertifikat mit Ihrem privaten Schlüssel (im Beispiel "myKey.key"). Ihre Kommunikationspartner können durch Ihren öffentlichen Schlüssel (in Ihrem Zertifikat enthalten) überprüfen, dass Sie im Besitz des zu diesem öffentlichen Schlüssel gehörenden privaten Schlüssels sind. Damit ist auch ausgeschlossen, dass Ihr öffentlicher Schlüssel von einem Angreifer missbräuchlich verwendet wurde.

Bei Self-Signed Zertifikaten bestätigen Sie selbst, dass Ihre Angaben in Ihrem Zertifikat richtig sind. Es gibt keine unabhängige Stelle, die Ihre Angaben überprüft.

Weitere Informationen

Informationen zur Handhabung von Client-Zertifikaten der S7-1500 CPU finden Sie im Kapitel Handling der Client-Zertifikate der S7-1500 CPU (Seite 353).

11.2.6 Nachrichten gesichert übertragen

Aufbau sicherer Verbindungen bei OPC UA

OPC UA verwendet sichere Verbindungen zwischen Client und Server. Dabei überprüft OPC UA die Identität der Kommunikationspartner. Für die Authentifizierung von Client und Server nutzt OPC UA Zertifikate gemäß X.509-V3 der ITU (International Telecommunication Union). Ausnahme: Bei der Security Policy "Keine Security" wird keine sichere Verbindung aufgebaut.

Message Security Modus

OPC UA verwendet die folgenden Security Policys zum Schutz von Nachrichten:

- Keine Security
  Alle Nachrichten sind ungesichert. Um diese Security Policy zu verwenden, bauen Sie eine Verbindung zu einem None-Endpunkt eines Servers auf.
- Signieren
  Alle Nachrichten werden signiert. Dadurch lässt sich die Integrität der empfangenen Nachrichten überprüfen. Manipulationen werden erkannt. Um diese Security Policy zu verwenden, bauen Sie eine Verbindung zu einem Sign-Endpunkt eines Servers auf.
- Signieren & Verschlüsseln
  Alle Nachrichten werden signiert und verschlüsselt. Dadurch lässt sich die Integrität der empfangenen Nachrichten überprüfen. Manipulationen werden erkannt. Außerdem kann kein Angreifer den Inhalt der Nachricht lesen (Schutz der Vertraulichkeit). Um diese Security Policy zu verwenden, bauen Sie eine Verbindung zu einem "Signieren & Verschlüsseln"-Endpunkt eines Servers auf.

Die Security Policys sind zusätzlich nach den verwendeten Algorithmen benannt. Beispiel: "Basic256Sha256 - Signieren & Verschlüsseln" bedeutet: Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.

Erforderliche Schichten

Das folgende Bild zeigt die drei Schichten Transport-Schicht, Secure Channel und Session, die für den Aufbau einer Verbindung stets erforderlich sind.

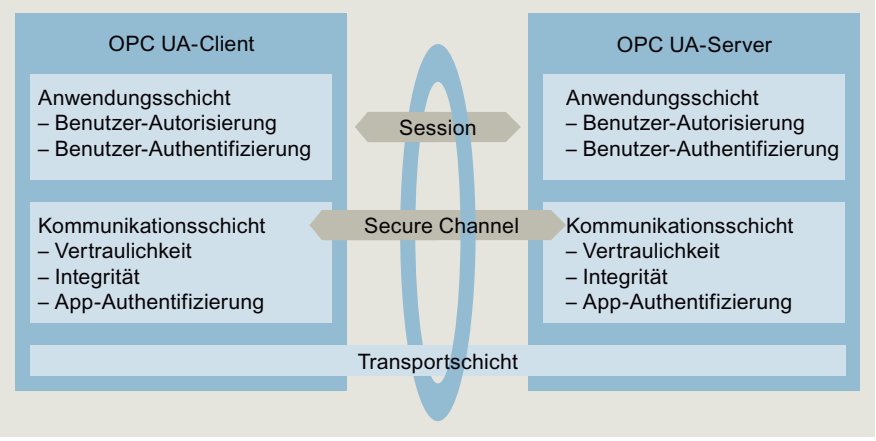


Bild 11-6 Erforderliche Schichten Transport-Schicht, Secure Channel und Session

- Transportschicht:
  Diese Schicht sendet und empfängt Nachrichten. Dazu verwendet OPC UA ein optimiertes TCP-basiertes Binärprotokoll. Die Transportschicht ist die Grundlage für den nachfolgenden Secure Channel.
- Secure Channel
  Der Secure Channel erhält von der Transportschicht die empfangenen Daten und leitet sie an die Session weiter. Daten von der Session, die gesendet werden sollen, leitet der Secure Channel an die Transportschicht weiter.
  Beim Security-Modus "Signieren" signiert der Secure Channel Daten (Nachrichten), die gesendet werden. Bei ankommenden Nachrichten überprüft der Secure Channel die Signatur, um Manipulationen erkennen zu können
  Bei einer "Signieren & Verschlüsseln" Security Policy signiert und verschlüsselt der Secure Channel die Sendedaten. Empfangene Daten entschlüsselt der Secure Channel. Anschließend überprüft der Secure Channel die Signatur.
  Bei der Security Policy "Keine Security" passieren die Nachrichtenpakete den Secure Channel unverändert (die Nachrichten werden im Klartext empfangen und gesendet).
- Session
  Die Session gibt die Nachrichten vom Secure Channel an die Anwendung weiter, bzw. erhält von der Anwendung Nachrichten, die gesendet werden sollen. Die Anwendung verwendet die Prozesswerte, bzw. stellt die Werte bereit.

Aufbau des Secure Channels

Der Aufbau des Secure Channels läuft folgendermaßen ab:

1. Der Server beginnt mit dem Aufbau des Secure Channels, wenn er eine Aufforderung dazu vom Client erhält. Dieser Request ist entweder signiert, signiert und verschlüsselt, oder die Nachricht wird im Klartext gesendet (Security-Modus des gewählten Server-Endpunkts). Bei "Signieren" und "Signieren & Verschlüsseln" sendet der Client ein "Geheimnis" (eine Zufallszahl) mit dem Request.
2. Der Server validiert das Client-Zertifikat (unverschlüsselt im Request enthalten) und überprüft die Identität des Clients. Vertraut der Server dem Client-Zertifikat, dann
   – entschlüsselt der Server die Nachricht und überprüft die Signatur ("Signieren & Verschlüsseln")
   – oder überprüft nur die Signatur ("Signieren")
   – oder lässt die Nachricht unverändert ("Keine Security").
3. Danach sendet der Server eine Antwort zum Client (gleichermaßen gesichert wie der Request). Im Response ist das Server-Geheimnis enthalten. Aus dem Client- und Server-Geheimnis errechnen Client und Server einen symmetrischen Schlüssel. Damit ist der Secure Channel aufgebaut.

Der symmetrische Schlüssel wird nun für das Signieren und Verschlüsseln von Nachrichten verwendet (anstelle der privaten und öffentlichen Schlüssel von Client und Server).

Aufbau der Session

Der Aufbau der Session läuft folgendermaßen ab:

1. Der Client startet den Session-Aufbau, indem er einen CreateSessionRequest an den Server schickt. Diese Nachricht enthält ein Nonce, eine nur einmal verwendete Zufallszahl. Der Server muss diese Zufallszahl (Nonce) signieren, um zu beweisen, dass er Inhaber des privaten Schlüssels ist. Der private Schlüssel gehört zu dem Zertifikat, den der Server beim Aufbau des Secure Channels verwendet. Diese Nachricht (und alle nachfolgenden) ist entsprechend den Sicherheitseinstellungen des gewählten Server-Endpunkts (ausgewählte Security Policys) gesichert.
2. Der Server antwortet mit der CreateSession Response. Diese Nachricht enthält den öffentlichen Schlüssel des Servers sowie das signierte Nonce. Der Client überprüft das signierte Nonce.
3. Wenn der Server den Test bestanden hat, dann sendet der Client einen SessionActivateRequest an den Server. Diese Nachricht enthält die Angaben, die für die Authentifizierung des Benutzers erforderlich sind:
   - entweder Benutzername und Passwort
   - oder das X.509-Zertifikat des Benutzers (in STEP 7 nicht unterstützt)
   - oder keine Daten (wenn ein anonymer Zugang konfiguriert ist).
4. Wenn der Benutzer über die notwendigen Rechte verfügt, sendet der Server eine Nachricht an den Client zurück (ActivateSessionResponse). Damit ist die Session aktiviert.

Die sichere Verbindung zwischen OPC UA-Client und -Server ist aufgebaut.

Aufbau einer Verbindung mit PLCopen Funktionsbausteinen

Die PLCopen Spezifikation hat eine Reihe von IEC 61131 Funktionsbausteinen für OPC UA-Clients definiert. Die Anweisung UA_Connect initiiert hierbei sowohl einen Secure Channel als auch eine Session nach oben beschriebenem Muster.

11.2.7 Zertifikatsmanagement über Global Discovery Server (GDS)

11.2.7.1 Automatisiertes Zertifikatsmanagement mit GDS

Ab TIA Portal V17 und S7-1500 CPU-Fimware-Version V2.9 können Sie Zertifikatsmanagement-Services des OPC UA-Servers für die Übertragung von OPC UA Server-Zertifikaten zur Laufzeit nutzen.
Über GDS Push-Managementfunktionen können OPC UA-Zertifikate, Vertrauenslisten und Zertifikatsperrlisten (CRLs) für den OPC UA-Server der S7-1500 CPU automatisiert aktualisiert werden. Die Automatisierung des Zertifikatsmanagements erspart den manuellen Aufwand für eine Neuprojektierung der CPU, z. B. nach Ablauf der Gültigkeitsdauer eines Zertifikats, und ein erneutes Laden der CPU. Außerdem können Sie über die GDS-Push-Managementfunktionen aktualisierte Zertifikate und Listen in den Betriebszuständen STOP und RUN der CPU übertragen.
Das Zertifikatsmanagement-Informationsmodell ist in der Spezifikation OPC UA Part 12 spezifiziert (OPC 10000-12: OPC Unified Architecture, Part 12: Discovery and Global Services).

Ab TIA Portal Version V18 und S7-1500 CPU-Firmware-Version V3.0 können Sie die GDS Push-Managementfunktionen auch für Webserver-Zertifikate nutzen. Der Ablauf von z. B. Zertifikats-Updates über GDS-Push-Managemementfunktionen ist hier prinzipiell identischzum Zertifikats-Update von OPC UA Server-Zertifikaten. Statt OPC UA-Server-Zertifikate übertragen Sie Webser-Zertifikate zur Laufzeit bzw. im laufenden Betrieb in die CPU. Unterschiede oder Einschränkungen werden an den entsprechenden Stellen in der folgenden Beschreibung erläutert.
Die folgenden Abschnitte geben einen Überblick über Global Discovery Services allgemein sowie die ab TIA Portal V17 / CPU Firmware-Version V2.9 unterstützte Funktion eines automatisierten Zertifikats-Updates.

Discovery Server

Ein OPC UA-Client benötigt, um sich mit einem OPC UA-Server zu verbinden, Informationen zu dessen Endpoint wie z. B. die Endpoint-URL und die Security-Policy. Bei einer großen Anzahl infrage kommender Server im Netz kann die Suche und Verwaltung dieser Server-Informationen ein Discovery-Server übernehmen.

- OPC UA-Server registrieren sich beim Discovery-Server.
- OPC UA-Clients fordern eine Liste erreichbarer Server vom Discovery-Server an und verbinden sich dann mit dem gewünschten OPC UA-Server.

Global Discovery Server (GDS)

Das OPC UA GDS-Konzept erlaubt die Konfiguration von Subnetz-übergreifenden Discovery-Diensten einerseits und andererseits stellt es Schnittstellen zur Verfügung, um zentrales Zertifikatsmanagement zu betreiben.
Ein Global Discovery Server (GDS) stellt Mechanismen für die zentrale Verwaltung folgender Komponenten zur Verfügung:

- CA-signierte Zertifikate und selbstsignierte Zertifikate
- Vertrauenslisten (Trusted Lists) und Zertifikatsperrlisten (Certificate Revocation Lists, CRL)

Ein GDS stellt damit einen Zugriffspunkt zum zentralen Zertifikatsmanagement zur Verfügung - er übernimmt damit die Aufgabe eines Security-Servers innerhalb eines OPC UA-Netzwerks.
Hauptanwendung von GDS ist das Management von CA-signierten Zertifikaten mit den entsprechenden CRLs, im Einzelnen geht es um folgende Aufgaben:

- Initiales Erstellen eines OPC UA-Applikationszertifikats
- Regelmäßiges Update der Vertrauensliste (Trust List) und der CRLs
- Erneuerung eines Applikationszertifikats

Zertifikatsmanagement

Zertifikatsmanagement hat die Aufgabe, die Verwaltung und Verteilung von Zertifikaten sowie von Vertrauenslisten für verschiedene Services bzw. UA-Applikationen automatisiert durchzuführen.

In diesem Kontext unterscheidet man folgende Rollen:

- Zertifikatsmanager - eine OPC UA-Applikation, die Zertifikatsmanagement-Funktionen bereitstellt
- Zertifikatsempfänger - eine OPC UA-Applikation, die Zertifikate, Vertrauenslisten und CRLs vom Zertifikatsmanager erhält.

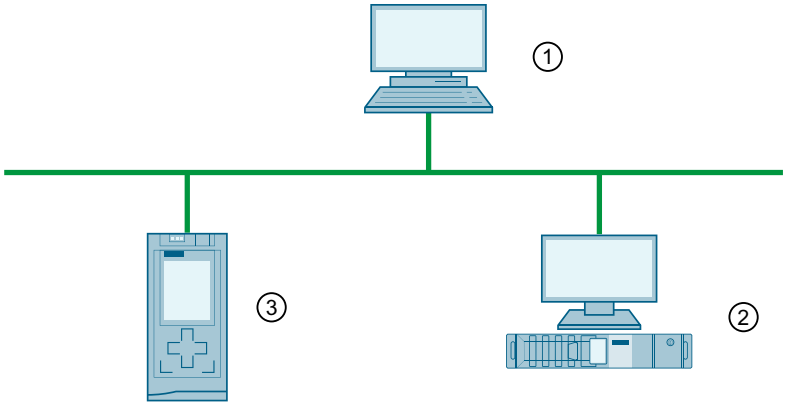
Es gibt zwei Modelle für Zertifikatsmanagement: Pull- und Push-Management.

- Beim Pull-Management agiert die OPC UA-Applikation als ein Client des GDS-Servers und nutzt Zertifikatsmanagement-Methoden, um Zertifikats-Updates und Vertrauenslisten-Updates anzufordern.
- Beim Push-Management agiert die OPC UA-Applikation als ein Server und stellt Methoden für einen OPC UA GDS als OPC UA-Client bereit. Der GDS in der Rolle Zertifikatsmanager nutzt diese Methoden, um Zertifikate und Vertrauenslisten-Updates zu übertragen (zu "pushen"), siehe Erläuterungen zum Konzept für automatisiertes Zertifikats-Update weiter unten.

Die S7-1500 CPU unterstützt ab Firmware-Version V2.9 zunächst nur das Push-Management für den OPC UA-Server der CPU.

Anlagenkonfiguration mit GDS

Im Bild unten ist beispielhaft dargestellt, welche Aufgaben die beteiligten Geräte im Zusammenspiel mit einem GDS, der Zertifikatsmanagement-Funktionen zur Verfügung stellt.



① Root CA - Gerät, das Zertifikate für die Anlage ausstellt (diese Zertifikate können auch auf anderem Weg übermittelt werden, z. B. per Mail)

② OPC UA GDS mit Zertifikatsmanager, erzeugt bzw. signiert Gerätezertifikate, verwaltet Vertrauenslisten und Zertifikatsperrlisten (Certificate Revocation List (CRL)), schreibt Zertifikate und Listen in die Geräte (Push-Funktion). Für die Push-Funktion benötigt dieses Gerät OPC UA Client-Funktionalität.

③ Gerät mit OPC UA-Applikation, empfängt "gepushte" Zertifikate und Listen

Konzept für automatisiertes Zertifikats-Update für STEP 7 ab Version V17

GDS und Zertifikatsmanager werden typischerweise in einer Applikation kombiniert - im Bild unten sind es allerdings zwei getrennte Komponenten.

Als Zertifikatsmanager eignen sich auch Geräte wie "normale" OPC UA-Clients, sie müssen allerdings den Datentyp Bytestring unterstützen, der für die Übertragung von Zertifikaten erfoderlich ist, wie z. B. eine S7-1500 CPU ab Firmware V2.9 als OPC UA-Client oder das Tool UA Expert (Unified Automation) mit GDS-Plugin.

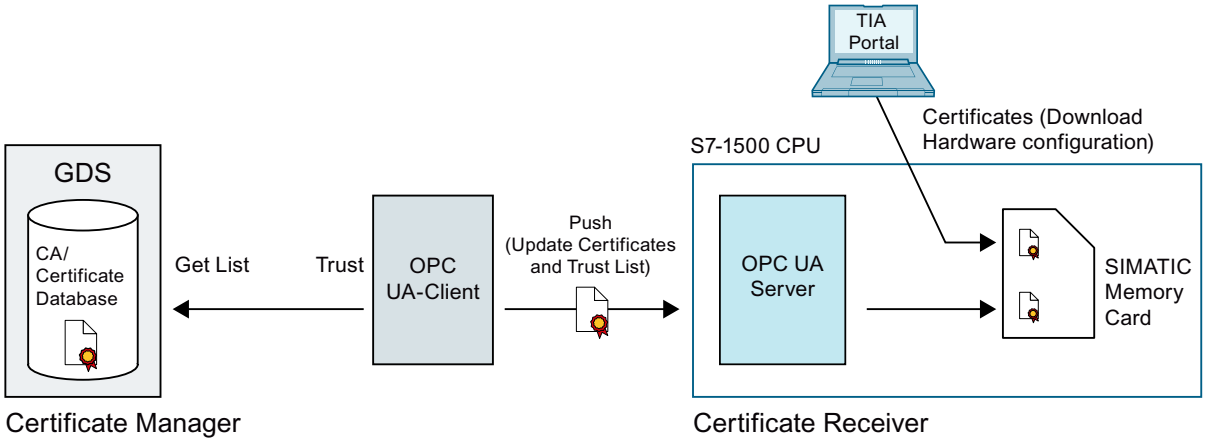
Der OPC UA-Server der S7-1500 CPU als Zertifikatsempfänger stellt die benötigten standardisierten Methoden und Attribute zur Verfügung, über die OPC UA Clients Zertifikate, Vertrauenslisten und CRLs schreiben und lesen können.

Der Schwerpunkt im Kontext OPC UA Server der S7-1500 CPU liegt hier auf der Beschreibung der Push-Funktion in Abgrenzung zur bekannten Art und Weise, Zertifikate in die CPU zu bringen: Durch Laden der Hardware-Konfiguration.

Im Bild unten ist gezeigt, über welche Wege Zertifikate und Listen für OPC UA in einer S7-1500 CPU ab FW V2.9 übertragen werden können:

- Entweder über Laden der Hardware-Konfiguration im STOP der CPU; die Zertifikate sind Teil der Hardware-Konfiguration.
- Oder über GDS Push-Methoden im RUN oder im STOP der CPU.

Beide Übertragungswege parallel zu nutzen ist nicht möglich. Wenn Sie sich für die Übertragung z. B. von OPC UA Server-Zertifikaten mit GDS-Push-Funktionen zur Laufzeit entschieden haben, müssen Sie auch alle anderen Zertifikatstypen über diesen Weg in die CPU übertragen.



Weitere Informationen

Weitere Informationen zu den Zertifikaten bei OPC UA finden Sie im Kapitel Zertifikate bei OPC UA (Seite 186).

11.2.7.2 Mengengerüst für Push-Funktion

Anzahl Zertifikate für Push-Funktion

Für die OPC UA Push-Funktion hat eine S7-1500 CPU unabhängig vom Typ mit der Firmware-Version ab V2.9 ein Mengengerüst von 62 Vertrauenslisten-Einträgen.

- Jeder aktivierte zertifikatsbasierte Service (CPU-Applikation) "verbraucht" jeweils einen Eintrag für ein Zertifikat und einen Eintrag für den privaten Schlüssel.
- Ein Sperrlisten-Eintrag (CRL) zählt dabei genauso wie ein Eintrag in die Liste der Zertifikate, denen vertraut wird.
- Ein Zertifikat, das von verschiedenen Services (CPU-Applikationen) genutzt wird, zählt als ein einziger Vertrauenslisten-Eintrag.

Größe von Elementen für Push-Funktion (z. B. Zertifikate)

Max. 4096 Bytes.

Beispiel

Sie wollen für max. 62 OPC UA-Clients Zugriff auf den OPC UA-Server gewähren und die Vertrauensliste entsprechend füllen.
Wenn Sie einen Sperrlisten-Eintrag in die Vertrauensliste einfügen, dann können Sie nur noch maximal 61 Client-Zertifikaten vertrauen.
Zusätzliche Zertifikate für OPC UA können **nicht** durch Laden der Hardware-Konfiguration in die CPU übertragen werden.

Tipp

Um die Anzahl notwendiger Zertifikate gering zu halten, empfehlen wir, die OPC UA-Client-Zertifikate möglichst von derselben CA signieren zu lassen.
In diesem Fall benötigt die CPU als OPC UA-Server nur das entsprechende CA-Zertifikat und CRLs. Mit diesen Elementen kann der OPC UA-Server dann alle CA-signierten Client-Zertifikate verifizieren. Die einzelnen Client-Zertifikate brauchen Sie dann nicht der Vertrauensliste hinzufügen.

11.2.7.3 GDS-Parameter einstellen und laden

Im Folgenden ist beschrieben, welche Einstellungen für den Zertifikate-Update notwendig sind.

Voraussetzung

- Je nach Applikationszertifikat ist die entsprechende STEP 7/TIA Portal-Version und S7-1500 CPU-Firmware-Version erforderlich.
  Siehe auch hier: Wissenswertes zum Zertifikatsmanagement (Seite 62)
  - Für OPC UA-Server-Zertifikate z. B. TIA Portal ab V17, CPU-Firmware-Version V2.9
  - Für Webserver-Zertifikate z. B. ab TIA Portal V18, CPU Firmware-Version V3.0
- Uhrzeit/Datum der CPU ist eingestellt (gilt generell für zertifikatsbasierte Kommunikation)
- Der OPC UA-Server ist aktiviert
- Der Service, der das GDS-Push-Management nutzt, muss aktiviert sein. Z. B. muss der Webserver für die Übertragung von Webserver-Zertifikaten aktiviert sein.
- Mindestens ein Endpunkt mit der Security Policy "Signieren & Verschlüsseln" muss projektiert sein, der Partner muss diesen Endpunkt nutzen
- Authentifizierter Benutzer mit ausreichenden Funktionsrechten ist parametriert
  Der Benutzer muss eine Rolle haben, die das Funktionsrecht "Zertifikate verwalten" hat.
  Dieses Funktionsrecht erfordert wiederum folgende Voraussetzungen:
  - In der Projektnavigation muss der **Projektschutz aktiviert** sein: Projektnavigation: "Security-Einstellungen > Einstellungen > Projektschutz".
  - In den CPU-Einstellungen muss wiederum im Bereich "OPC UA > Allgemein" folgende Allgemeine-Benutzerverwaltung-Einstellung aktiviert sein: "Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts aktivieren".

Wie die Funktionsrechte eingestellt werden, ist beschrieben im Abschnitt Benutzer und Rollen mit OPC UA-Funktionsrechten (Seite 251).

GDS aktivieren

Wenn die oben genannten Voraussetzungen erfüllt sind, muss GDS noch aktiviert werden:

1. Navigieren Sie im Inspektorfenster (CPU-Parameter) zum Bereich "OPC UA > Server > Allgemein".
2. Aktivieren Sie die Option "Global Discovery Services (Push) aktivieren"

Genutzten Zertifikatspeicher bestimmen

Zertifikate, die über GDS verwaltet werden, liegen in einem anderen Speicherbereich als die Zertifikate, die über das TIA Portal (STEP 7) heruntergeladen werden.

Wenn GDS-Push-Zertifikatsmanagement aktiviert ist, bedienen sich die Services (Applikationen) der CPU auch aus dem Zertifikatsspeicher, dessen Zertifikate zur Laufzeit verwaltet werden.

1. Navigieren Sie in den CPU-Einstellungen in den Bereich "Schutz & Security > Zertifikatsmanagement".
2. Wählen Sie die Option "Von Zertifikatsverwaltung zur Laufzeit bereitgestellte Zertifikate verwenden".

   Die andere Option (durch TIA Portal konfigurierte und heruntergeladene Zertifikate verwenden) verwendet die Zertifikate, die im CPU-STOP vom TIA Portal mit der Konfiguration in die CPU geladen werden. In diesem Zertifikatspeicher können keine Zertifikate oder Vertrauenslisten zur Laufzeit aktualisiert werden.

Diagnose für Ablauf von Zertifikaten aktivieren

Wenn Sie vorab über den Ablauf eines Zertifikats informiert werden möchten, wählen Sie die Option "Systemdiagnoseereignis für den Zertifikatsablauf aktivieren" im Bereich "Schutz & Security > Zertifikatsmanagement".

Tragen Sie im Eingabefeld "Ereignis anzeigen bei Restlaufzeit des Zertifikats von ..." einen Prozentwert ein.

Wirkung dieser Einstellungen:

- Zu dem Zeitpunkt, zu dem dieser Wert von einem Zertifikat erreicht wird, erscheint eine entsprechende Systemdiagnosemeldung bis das Zertifikat abläuft oder aktualisiert wurde.
- Wenn das Ende des Gültigkeitszeitraums eines Zertifikats erreicht ist, erzeugt die CPU eine entsprechende Systemdiagnosemeldung sowie einen Eintrag im Diagnosepuffer und die Maintenance-LED leuchtet.

Beispiel:
Das per GDS am 1.6.22 übermittelte Zertifikat hat eine Gültigkeit vom 1.6.22 bis 30.6.22 (30 Tage). Sie haben einen Prozentwert für das Diagnoseereignis von 10 eingegeben. Am 27.6.22, nach Ablauf von 90% der Gültigkeitsdauer, erscheint eine entsprechende Meldung, dass das übermittelte Zertifikat am 30.06.2022 ablaufen wird.
Unabhängig vom parametrierten Prozentwert erscheint beim Ablauf des Gültigkeitszeitraums eines Zertifikats auf jeden Fall eine entsprechende Meldung, ein Eintrag im Diagnosepuffer und die Maintenance-LED leuchtet.

CPU laden

Beim Laden der Konfiguration in die CPU haben Sie die Möglichkeit, die Zertifikate, die über GDS verwaltet werden, vor dem Laden zu löschen. Wenn Sie das Löschen bestätigen, folgt nach dem Laden eine Bereitstellungsphase (siehe Abschnitt zur Inbetriebnahme).

Wenn Sie die Memory Card außerhalb der CPU laden (Card Reader), dann wird dieser Zertifikatsspeicher immer gelöscht.

Wenn Global Discovery Services (Push) aktiviert ist und keine gepushten Zertifikate vorhanden sind, dann liegt für den OPC UA-Server weder ein eigenes Zertifikat noch eine Vertrauensliste und auch keine CRL vor.

11.2.7.4 GDS Inbetriebnahme

Part 12 der OPC UA-Spezifikation unterscheidet beim Zertifikatsmanagement eine Bereitstellungsphase (provisioning phase) und eine Laufzeitphase (run time phase).

In der Bereitstellungsphase liefert ein GDS bzw. ein OPC UA-Client dem OPC UA-Server der CPU initiale Vertrauenslisten und CRLs. In dieser Phase akzeptiert der OPC UA-Server der CPU alle Zertifikate und Listen, die ihm angeboten werden - vergleichbar mit der Einstellung "Vertrauenswürdige Clients" für den OPC UA-Server, dass alle Client-Zertifikate zur Laufzeit akzeptiert werden. Nur so ist der Verbindungsaufbau mit Clients möglich, die der Server nicht kennt, d. h. die er nicht durch vorhandene Zertifikate bzw. Vertrauenslisten authentifizieren kann bis er das entsprechende Client-Zertifikat bzw. die entsprechende Vertrauensliste erhalten hat.

Die Bereitstellungsphase zeichnet sich durch eine geringere Sicherheit aus (geringere Security); daher wird die Bereitstellungsphase durch Leuchten der Maintenance-LED und einen entsprechenden Diagnosepuffer-Eintrag angezeigt (Maintenance demanded).

In der Laufzeitphase werden z. B. die bestehenden CRLs aktualisiert sowie die Zertifikate und Vertrauenslisten erneuert. Die Kommunikation ist in dieser Phase sicher (secure).

Voraussetzung

Nur autorisierte Benutzer mit ausreichenden Funktionsrechten können eine Verbindung in der Bereitstellungsphase aufbauen. Die Benutzer müssen eine Rolle mit dem Funktionsrecht "Zertifikate verwalten" haben.

Siehe auch GDS-Parameter einstellen und laden (Seite 198).

Regeln für die Bereitstellungsphase

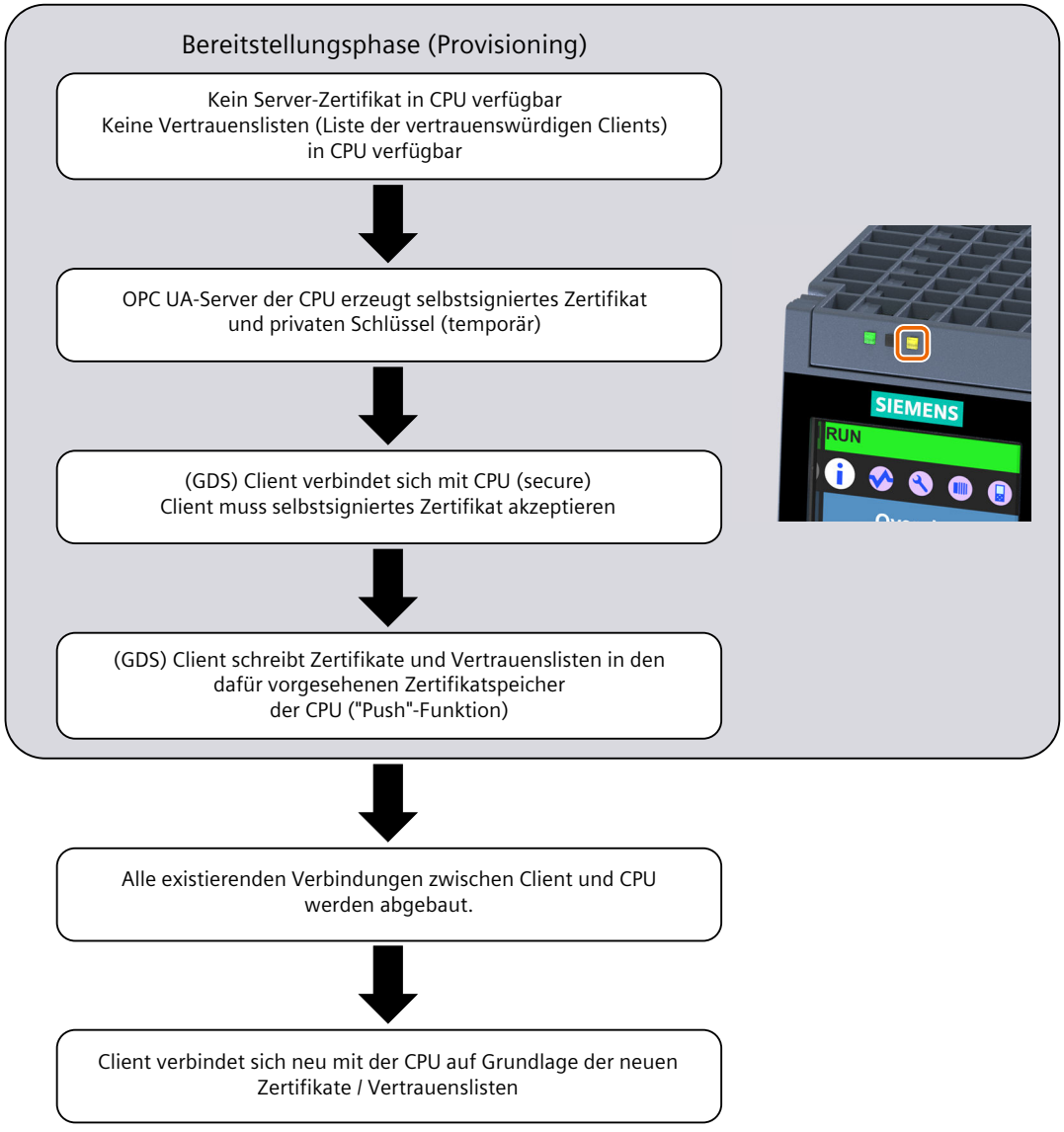
In der Bereitstellungsphase kann der OPC UA-Server der CPU die OPC UA-Clients, die einen Verbindungsaufbau initiieren, nicht authentifizieren. Daher sind folgende Regeln zu beachten:

- Sorgen Sie für eine sichere Umgebung, z. B. Zugriff auf die CPU nur für den Inbetriebnehmer ermöglichen. Überprüfen Sie, dass die richtigen Geräte miteinander kommunizieren.
- Begrenzen Sie die Zeit für diese Phase.

Dass sich die CPU in der Bereitstellungsphase befindet, signalisiert die CPU durch die leuchtende Maintenance-LED sowie durch einen entsprechenden Diagnosepuffer-Eintrag (Maintenance demanded).

Ablauf der Bereitstellungsphase

Im Folgenden ist der Ablauf der Bereitstellungsphase für OPC UA-Server-Zertifikate und Vertrauenslisten skizziert.
Der Ablauf der Bereitstellungsphase für Webserver-Zertifikate ist vergleichbar. Im Unterschied zu OPC UA pusht der GDS-Client nur Webserver-Zertifikate und keine Vertrauenslisten in den entsprechenden Zertifikatsspeicher.



Eintritt in die Bereitstellungsphase

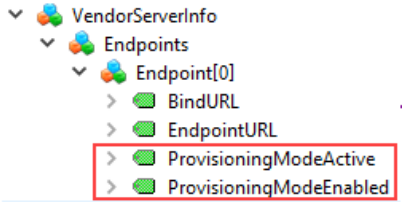
Die CPU ist nach einem Anlauf des OPC UA-Servers automatisch in der Bereitstellungsphase, wenn eine der folgenden Bedingungen erfüllt ist:

- Das OPC UA-Server-Zertifikat ist das initiale, von der CPU generierte selbstsignierte Zertifikat und wurde noch nicht durch ein gültiges Server-Zertifikat ersetzt.
- Die Vertrauensliste (Liste vertrauenswürdiger Clients) ist leer.

Das von der CPU generierte OPC UA-Server-Zertifikat enthält die wichtigsten Parameter des OPC UA-Servers und wird inklusive dem privaten Schlüssel nach jedem Hochlauf des OPC UA-Servers nach NETZ-EIN neu generiert - bis das gültige Server-Zertifikat vorliegt. Deswegen kann der Start des OPC UA Servers nach NETZ-EIN länger dauern.
Nach dem Laden der Hardware-Konfiguration wird der Zertifikatsspeicher für Zertifikate, die zur Laufzeit aktualisiert werden können, je nach Einstellung beim Laden gelöscht oder die Zertifikate bleiben erhalten. D. h. falls GDS aktiviert ist und der Zertifikatsspeicher gelöscht wurde, befindet sich die CPU nach einem Laden der Hardware-Konfiguration in der Bereitstellungsphase.

Diagnose der Bereitstellungsphase

Neben der leuchtenden Maintenance-LED verfügt das GDS-Adressmodell über zwei Knoten, die Auskunft darüber geben, ob der OPC UA-Server der CPU sich in der Bereitstellungsphase befindet:



Nur wenn die Voraussetzungen für GDS erfüllt sind (Endpunkt-Security Signiert & Verschlüsselt sowie Administrator-Funktionsrechte vorhanden), können Sie die beiden Knoten wie im Bild markiert zur Diagnose nutzen.
ProvisioningModeEnabled: Zeigt an, dass eine Bereitstellungsphase unterstützt wird
ProvisioningModeActive: Zeigt an, dass sich der OPC UA-Server der CPU in der Bereitstellungsphase befindet.

Ende der Bereitstellungsphase

Die CPU beendet die Bereitstellungsphase automatisch, wenn folgende Bedingungen erfüllt sind:

- Das von der CPU für die Bereitstellungsphase generierte und selbstsignierte Zertifikat wurde durch ein gültiges Server-Zertifikat überschrieben. Dieses gültige Server-Zertifikat kann ein selbstsigniertes Zertifikat oder ein CA-signiertes Zertifikat sein.
- Die Vertrauensliste in der CPU ist nicht leer, d. h. Client-Zertifikate der OPC UA-Clients, denen vertraut werden soll, bzw. CA-Zertifikate zum Prüfen der Client-Zertifikate sind vorhanden.

Wenn der OPC UA-Client ein CA-signiertes Zertifikat übermittelt und zusätzlich das CA-Zertifikat zur Vertrauensliste hinzufügt, dann kann der OPC UA-Server der CPU alle weiteren Zertifikate von OPC UA-Clients, die von derselben CA signiert wurden, automatisch akzeptieren.

Anforderung eines gültigen Server-Zertifikats

Ab TIA Portal Version V18 / S7-1500 CPU Version V3.0 können neben OPC UA Server-Zertifikaten auch Zertifikate für weiteres Services in die CPU übertragen werden, z. B. für den Webserver.

Ein gültiges Zertifikat erhält der entsprechende Service, z. B. der OPC UA-Server der CPU, in folgenden Schritten:

1. Ein GDS-Client (OPC UA-Client) ruft die Methode "CreateSigningRequest"auf, um ein Server-Zertifikat mit einem Certificate Signing Request (CSR) anzufordern.
2. Dieser CSR muss von einer Certificate Authority (CA) signiert werden.
3. Der signierte CSR muss dann als Zertifikat zurück zum OPC UA-Server der CPU transferiert werden.

Der OPC UA-Server der CPU stellt diese Methode zur Verfügung, vorausgesetzt, der Client hat das erforderliche Funktionsrecht "Zertifikate verwalten".

Die Methode "CreateSigningRequest" erlaubt folgende Varianten:

- Zertifikat-Update ohne Erzeugung eines neuen Schlüsselpaars (CPU-interne Schlüssel, die bereits vorhanden sind, werden verwendet)
- Zertifikat-Update mit Erzeugung eines neuen Schlüsselpaars (CPU intern)

Daneben gibt es noch die Möglichkeit, Zertifikate mit extern erzeugten Schlüsselpaaren zu erzeugen.

ACHTUNG
Empfohlene Vorgehensweise bei der Zertifikatsgenerierung Ein Transport von privaten Schlüsseln sollte nach Möglichkeit vermieden werden; ein privater Schlüssel sollte ein Gerät nicht verlassen. Aus diesem Grund empfehlen wir die Zertifikatsgenerierung ohne Erzeugung eines neuen Schlüsselpaars bzw. mit CPU-interner Erzeugung eines Schlüsselpaars.

Zertifikat erzeugen ohne Schlüsselpaar

- Die Methode "CreateSigningRequest" gibt einen Certificate Signing Request (CSR) zurück, also eine Datei (*.csr) mit spezifischen Informationen zum Server bzw. zum Service, z. B. Applikationsname und URL.
- Außerhalb der CPU muss dieser CSR von einer Certificate Authority (CA) validiert, signiert und dann das Zertifikat zurückgeschickt werden.
- Das Zertifikat muss anschließend mit der Methode "UpdateCertificate" in die CPU übertragen ("gepusht") werden.

Bei dieser Vorgehensweise verlässt der Schlüssel die CPU nicht.

Zertifikat erzeugen mit intern erzeugtem Schlüsselpaar

Die Vorgehensweise ähnelt der im vorhergehenden Abschnitt erläuterten Methode; nur dass neben dem CSR zusätzlich ein Schlüsselpaar erzeugt wird. Sie geben im Parameter der Methode "CreateSigningRequest" an, dass sie ein Schlüsselpaar erzeugen soll.

Auch bei dieser Vorgehensweise verlässt der private Schlüssel die CPU nicht.

Die Generierung eines neuen Schlüsselpaars erzeugt massive CPU-Last. Die CPU arbeitet diesen Auftrag über einen längeren Zeitabschnitt mit geringerer Priorität im reservierten Bereich für die Kommunikationslast ab. Die Dauer dieses Zeitabschnitts ist abhängig von der Leistungsfähigkeit der CPU.

Weil während der Schlüsselgenerierung über eine längere Zeit der Anteil der eingestellten Kommunikationslast voll ausgenutzt wird, stellen Sie den Anteil der "Zyklusbelastung durch Kommunikation" so ein, dass die maximale Zykluszeit nicht überschritten wird und ausreichend Reserven vorhanden sind. Nutzen Sie dazu z. B. die Webserver-Seite "Diagnose > Laufzeitinformationen" der CPU. Diese Seite zeigt Informationen zur aktuellen Programm-/Kommunikationslast und Zykluszeit Ihres Anwenderprogramms. Über einen Regler bekommen Sie Hilfestellung zu den Auswirkungen einer geänderten Kommunikationslast auf die Zykluszeit.

Zertifikat erzeugen mit extern erzeugtem Schlüsselpaar

Die Zertifikatserzeugung läuft z. B. mithilfe von Tools ab, die zusätzliche Schlüssel erzeugen können.

Zertifikat und Schlüssel werden über die Methode "UpdateCertificate" in die CPU übertragen.

Diese Vorgehensweise ist wegen geringerer Security nicht empfehlenswert.

ACHTUNG
Unterschiedliche Schlüssel für unterschiedliche Zielsysteme Nutzen Sie für ein Produktivsystem immer neu erzeugte Schlüssel. Wenn Sie Ihr Projekt z. B. mit PLCSIM Advanced auf Ihrem PC simulieren und testen, nutzen Sie auf keinen Fall die für die Simulation genutzen Schlüssel auch für ein Produktivsystem. Beschränken Sie den Zugriff auf PC-basierte Steuerungen durch die Einrichtung von entsprechenden Berechtigungen.

11.2.7.5 Adressmodell für das Push-Zertifikatsmanagement

Die OPC UA-Spezifikation Part 12 (OPC 10000-12: Discovery, Global Services) definiert Methoden und Attribute für z. B. OPC UA-Server, um GDS bzw. OPC UA-Clients zu ermöglichen, Zertifikate und Vertrauenslisten im Server zu aktualisieren ("Push-Zertifikatsmanagement"). Diese Methoden und Attribute finden sich auch im Adressmodell des OPC UA-Servers wieder.

Im Folgenden ist der relevante Abschnitt im Adressmodell des OPC UA-Servers der S7-1500 CPU erläutert.

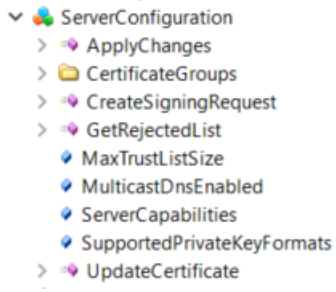
Voraussetzung

Damit die relevanten Methoden und Attribute für die GDS-Push-Funktionalität sichtbar werden, sind folgende Voraussetzungen erforderlich:

- GDS ist aktiviert.
- Eingestellte Security Policy unterstützt Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln (Sign & Encrypt).
- Zugriff mit Runtime-Funktionsrecht "Zertifikate verwalten".

Adressmodell für GDS-Push-Funktionalität

Das Adressmodell für die GDS-Push-Funktionalität entspricht dem "Information Model for Push Certificate Management" der OPC UA-Spezifikation OPC 10000-12: Discovery, Global Services.
Unter dem Knoten "ServerConfiguration" befindet sich folgende Struktur:



Methoden und Attribute für den Zugriff auf das Adressmodell

Im Folgenden sind die Methoden und Attribute in Kurzform erläutert mit Besonderheiten und Einschränkungen des konkreten Adressmodells der S7-1500 CPU. Die oben genannte OPC UA-Spezifikation enthält die allgemeine Beschreibung.
Im Anschluss an diese Übersichtstabelle finden Sie die detailliertere Beschreibung der einzelnen Methoden.

Methode / Attribut (Variable)	Beschreibung
CreateSigningRequest	Methode zum Erzeugen eines PKCS#10-kodierten Certificate Request, der mit dem privaten Schlüssel des Services (z. B. OPC UA-Server) signiert ist.
UpdateCertificate	Methode zum Aktualisieren des Server-Zertifikats z. B. für den OPC UA-Server.
ApplyChanges	Methode zum Anwenden einer security-relevanten Änderung, falls beim Ausführen einer vorher ausgeführten Methode das "ApplyChangesRequired"-Attribut gesetzt wurde. **Hinweis** Wenn als Folge von "ApplyChanges" ein Zertifikat geändert wird, unterbricht die CPU die Verbindungen/Sessions, die über dieses Zertifikat gesichert sind. Hintergrund: Die Basis für die gesicherten Verbindungen - das Zertifikat - ist nicht mehr gültig.
GetRejectedList	Methode, die eine Liste von Zertifikaten zurückliefert, die vom OPC UA-Server abgelehnt wurden. Abgelehnte Zertifikate werden aktuell nicht vom OPC UA-Server der S7-1500 CPUs gespeichert. Die Methode liefert ein leeres Array (RejectedList) zurück.
ServerCapabilities	Variable wird vom OPC UA-Server der S7-1500 CPU nicht unterstützt.
SupportedPrivateKeyFormats	Variable, die erlaubte Formate des privaten Schlüssels angibt. Für S7-1500 CPUs nur "PEM" (String Array)

Methode / Attribut (Variable)	Beschreibung
MaxTrustListSize	Variable, die die maximale Größe der Vertrausensliste angibt.
MulticastDnsEnabled	Variable, die angibt, ob Multicast DNS unterstützt wird. Für S7-1500 CPUs ist der Wert "False".
CertificateGroups	Objekt (Ordner), das alle vom OPC UA-Server unterstützten Zertifikatsgruppen organisiert. In den Zertifikatsgruppen befinden sich die Objekte, die dynamisch zur Laufzeit aktualisiert werden können: Z. B. jeweils eine Vertrauensliste und ein oder mehrere Zertifikate, die einem Service (z. B. OPC UA-Applikation) zugeordnet sind. Details zum Aufbau des CertificateGroups Objekt und welche Methoden und Attribute dort verfügbar sind ist im nächsten Kapitel beschrieben.

CreateSigningRequest

Die Methode hat folgende Parameter:

Parameter	Datentyp	Beschreibung
[in] certificateGroupId	NodeId	NodeId des CertificateGroup Objekts.
[in] certificateTypeId	NodeId	Angeforderter Zertifikatstyp. Liste der erlaubten Zertifikatstypen ist spezifiziert durch die Variable "CertificateTypes" der Zertifikatsgruppe. Für den OPC UA-Server z. B. Zertifikatetyp "RsaSha256ApplicationCertificateType", für den Webserver der Zertifikatetyp "HttpsCertificateType".
[in] subjectName	String	Subject Name, der im Certificate Request angefordert wird. Wenn nicht spezifiziert, wird der aktuelle Subject Name des Zertifikats verwendet.
[in] regeneratePrivateKey	Boolean	True: Server erzeugt einen neuen privaten Schlüssel. Dieser Schlüssel wird solange gespeichert bis die UpdateCertificate-Methode mit dem passenden signierten Zertifikat aufgerufen wird. False: Server verwendet vorliegenden privaten Schlüssel.
[in] nonce	ByteString	Zusätzliches Nonce für die Erzeugung des neuen privaten Schlüssels (siehe regeneratePrivateKey). Muss mind. 32 Bytes lang sein.
[out] certificateRequest	ByteString	PKCS #10 - DER kodierter Certificate Request.

Method Result Codes

Result Code	Beschreibung
Bad_InvalidArgument	certificateTypeId, certificateGroupId oder subjectName ist nicht gültig.
Bad_UserAccessDenied	Der aktuelle Benutzer hat nicht die erforderlichen Funktionsrechte.

UpdateCertificate

Anwendungsfälle:

- Zertifikatserzeugung mit CreateSigningRequest. Kein privater Schlüssel bereitgestellt.
- Neuer privater Schlüssel und neues Zertifikat wurden außerhalb des Servers erzeugt. Beides wird mit UpdateCertificate aktualisiert.
- Zertifikat erzeugt und signiert mit dem privaten Schlüssel des bestehenden Zertifikats. Kein privater Schlüssel bereitgestellt.

Parameter	Datentyp	Beschreibung
[in] certificateGroupId	NodeId	NodeId des CertificateGroup Objekts.
[in] certificateTypeId	NodeId	Angeforderter Zertifikatstyp. Liste der erlaubten Zertifikatstypen ist spezifiziert durch die Variable "CertificateTypes" der Zertifikatsgruppe.
[in] certificate	ByteString	DER-kodiertes Zertifikat, das das bestehende Zertifikat ersetzt.
[in] issuerCertificates	ByteString	Aussteller-Zertifikate
[in] privateKeyFormat	String	Format des privaten Schlüssels. Aktuell nur PEM unterstützt. Falls der privateKey nicht spezifiziert ist: null oder Leerstring.
[in] privateKey	ByteString	Privater Schlüssel kodiert wie in privateKeyFormat angegeben.
[out] applyChangesRequired	Boolean	Zeigt an, dass die Methode "ApplyChanges" vor Benutzung des neuen Zertifikats aufgerufen werden muss.

Method Result Codes

Result Code	Beschreibung
Bad_InvalidArgument	certificateTypeId oder certificateGroupId ist nicht gültig.
Bad_CertificateInvalid	Das Zertifikat ist ungültig oder das Format wird nicht unterstützt.
Bad_NotSupported	Der private Schlüssel ist ungültig oder das Format wird nicht unterstützt.
Bad_UserAccessDenied	Der aktuelle Benutzer hat nicht die erforderlichen Funktionsrechte.
Bad_SecurityChecksFailed	Fehler aufgetreten beim Verifizieren der Integrität des Zertifikats.

Apply Changes

Die Methode hat keine Parameter.

Method Result Codes

Result Code	Beschreibung
Bad_UserAccessDenied	Der aktuelle Benutzer hat nicht die erforderlichen Funktionsrechte.

GetRejectedList

Die Methode hat folgende Parameter:

Parameter	Datentyp	Beschreibung
[out] certificates	ByteStrings	DER-kodierte Liste abgelehnter Zertifikate. Aktuell liefert die Methode eine leere Liste (leeres Array) zurück, da abgelehnte Zertifikate nicht gespeichert werden.

Method Result Codes

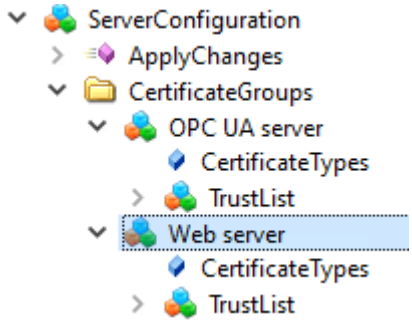
Result Code	Beschreibung
Bad_UserAccessDenied	Der aktuelle Benutzer hat nicht die erforderlichen Funktionsrechte.

11.2.7.6 CertificateGroups im Adressmodell

Zur Laufzeit aktualisierbare Zertifikate und Vertrauenslisten für Services bzw. Applikationen der CPU (z. B. OPC UA-Server) befinden sich im Adressmodell im Objekt "CertificateGroups" - für die verschiedenen Services der S7-1500 CPU gibt es jeweils eine Zertifikategruppe. Für das OPC UAServer-Zertifikat hat die Zertifikategruppe den Namen "OPC UA server".

CertificateGroup im Adressmodell

Das folgende Bild zeigt die Struktur des Objekts "CertificateGroups" unterhalb des Knotens "ServerConfiguration".



Den Display Name der CertificateGroups (z. B. "OPC UA server" können Sie in STEP 7 (TIA Portal) ändern:

1. Navigieren Sie im Inspektorfenster (CPU-Eigenschaften) zum Bereich "Schutz & Security > Zertifikatsmanagement".
2. Aktivieren Sie die Option "Von Zertifikatsverwaltung während der Laufzeit bereitgestellte Zertifikate verwenden".
3. Ändern Sie den Gruppennamen (DisplayName) der Zertifikatsgruppe in der Tabelle darunter. Zulässig sind 1-64 Zeichen im Format 7-Bit ASCII.
   In der ersten Spalte ist der aktivierte Service angegeben, für den Zertifikate zur Laufzeit übertragen werden können und in der Spalte "ID" ist ein festgelegter numerischer Identifier angegeben, der CPU-intern zur Referenzierung der Zertifikate genutzt wird.

Hier ein Beispiel für die Anzeige im Bereich "Zertifikatsmanagement":

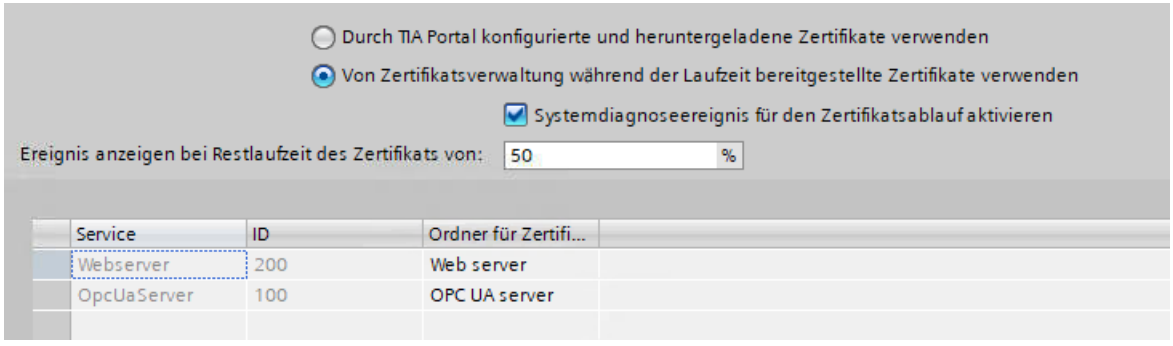


Bild 11-7 Einstellungen Zertifikatsmanagement

Knoten "CertificateTypes"

Die Variable "CertificateTypes" spezifiziert die NodeIds der Zertifikatstypen, die der Server-Applikation zugeordnet sind.

Für den Service OPC UA-Server z. B. wird der CertificateType "RsaSha256ApplicationCertificateType" unterstützt, für den Service Webserver ist es der CertificateType "HttpsCertificateType".

Knoten "TrustList"

Der Knoten für das Vertrauenslisten-Objekt (TrustList Datei) definiert einen OPC UA Dateityp (Binary encoded stream), der die Information enthält, welche Zertifikate und CRLs im Verzeichnis "pki store\trusted\issuer" der Memory Card gelesen und aktualisiert werden können. Dieser Knoten stellt Methoden und Attribute zur Verfügung, die ein Lesen und Aktualisieren möglich machen.

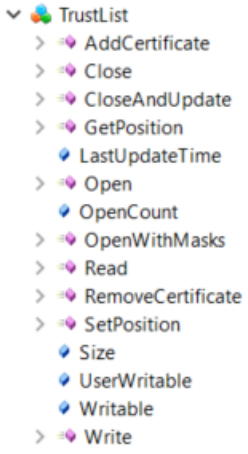
Der Knoten ist eine Instanz des OPC UA Datentyps "TrustListDataType" mit folgendem Aufbau:

Parameter	Datentyp	Beschreibung
specifiedLists	TrustListsMasks	Bitmaske, die anzeigt, welche Listen Informationen enthalten.
trustedCertificates	ByteStrings	Liste der Applikationszertifikate und CA-Zertifikate, welchen vertraut wird.

Parameter	Datentyp	Beschreibung
trustedCrls	ByteStrings	CRLs für die Zertifikate in der Liste "trustedCertificates".
issuerCertificates	ByteStrings	Liste der CA-Zertifikate, die notwendig sind zum Validieren der CA-signierten Zertifikate.
issuerCrls	ByteStrings	CRLs der CA-Zertifikate in der Liste "issuerCertificates".

Aufbau des Knotens "TrustList"

Der Knoten "TrustList" hat folgenden Aufbau:



Methoden und Attribute für den Knoten "TrustList"

Im Folgenden finden Sie eine Beschreibung der Knoten unter "TrustList", die zusätzlich zu den Methoden des Object Type "FileType" hinzukommen. Der TrustList Type ist von FileType abgeleitet (siehe OPC 10000-5: OPC Unified Architecture, Part 5: Information Model).

Methode / Attribut (Variable)	Beschreibung
LastUpdateTime	Variable, die den Zeitpunkt der letzten Aktualisierung anzeigt.
OpenWithMasks	Methode, die einem Client erlaubt, nur einen Teil der TrustList zu lesen.
CloseAndUpdate	Methode zum Schließen der TrustList-Datei und Anwenden der Änderungen.
AddCertificate	Methode zum Hinzufügen eines einzelnen Zertifikats zur TrustList.
RemoveCertificate	Methode zum Entfernen eines einzelnen Zertifikats aus der TrustList.

Beschreibung der Methoden

Die Beschreibung der Methoden mit ihren Result Codes, Attributen und Typen des TrustList-Objekts finden Sie in der OPC UA Spezifikation Part 12, Discovery and Global Services.

11.3 S7-1500 CPU als OPC UA-Server nutzen

11.3.1 Wissenswertes zum OPC UA-Server der S7-1500 CPUs

11.3.1.1 Der OPC UA-Server der S7-1500 CPUs

Die S7-1500 CPUs ab Firmware V2.0 sind mit einem OPC UA-Server ausgestattet. Dies betrifft neben den Standard-S7-1500 CPUs auch die Varianten S7-1500F, S7-1500T, S7-1500C, S7-1500pro CPUs, ET 200SP CPUs, SIMATIC S7-1500 SW Controller und PLCSIM Advanced.
Konvention: Mit "S7-1500 CPUs" sind auch die oben genannten CPU-Varianten gemeint.

Grundlagen zum OPC UA-Server der S7-1500 CPU

Der Zugriff auf den OPC UA-Server der CPU ist über alle integrierten Ethernet-Schnittstellen der S7-1500 CPU möglich.
Über CPs ist unter folgenden Bedingungen ein direkter Zugriff über den Rückwandbus des Automatisierungssystems auf den OPC UA-Server der CPU möglich:

- Projektierung mit TIA Portal Version ab V16
- S7-1500-CPU ab Firmware Version 2.8 sowie CP 1543-1 ab Firmware Version V2.2

Zur Projektierung siehe Zugang zu OPC UA-Applikationen (Seite 168).
Über CMs ist kein direkter Zugriff über den Rückwandbus des Automatisierungssystems auf den OPC UA-Server der CPU möglich.
Für den Zugriff durch Clients speichert der Server die freigegebenen PLC-Variablen und andere Informationen in Form von Knoten ab (siehe Zugriffsmöglichkeiten auf Daten des OPC UA-Servers (Seite 217)). Diese Knoten sind miteinander verbunden und bilden ein Netzwerk. OPC UA definiert Einstiegspunkte in dieses Netzwerk (Well-known Nodes), die das Navigieren zu unterlagerten Knoten ermöglichen.
Mit einem OPC UA-Client können Sie Werte von Variablen des SPS-Programms lesen, beobachten oder schreiben sowie Methoden aufrufen, die der Server zur Verfügung stellt. Ab Firmware Version 2.5 können Sie Methoden implementieren, siehe Wissenswertes zu Server-Methoden (Seite 290).

Knotenklassen

OPC UA-Server stellen Informationen in Form von Knoten (Nodes) zur Verfügung. Ein Knoten kann zum Beispiel ein Objekt, eine Variable, eine Methode oder eine Property sein.
Das folgende Beispiel zeigt den Adressraum des OPC UA-Servers einer S7-1500 CPU (Ausschnitt aus dem OPC UA-Client "UaExpert" von Unified Automation).

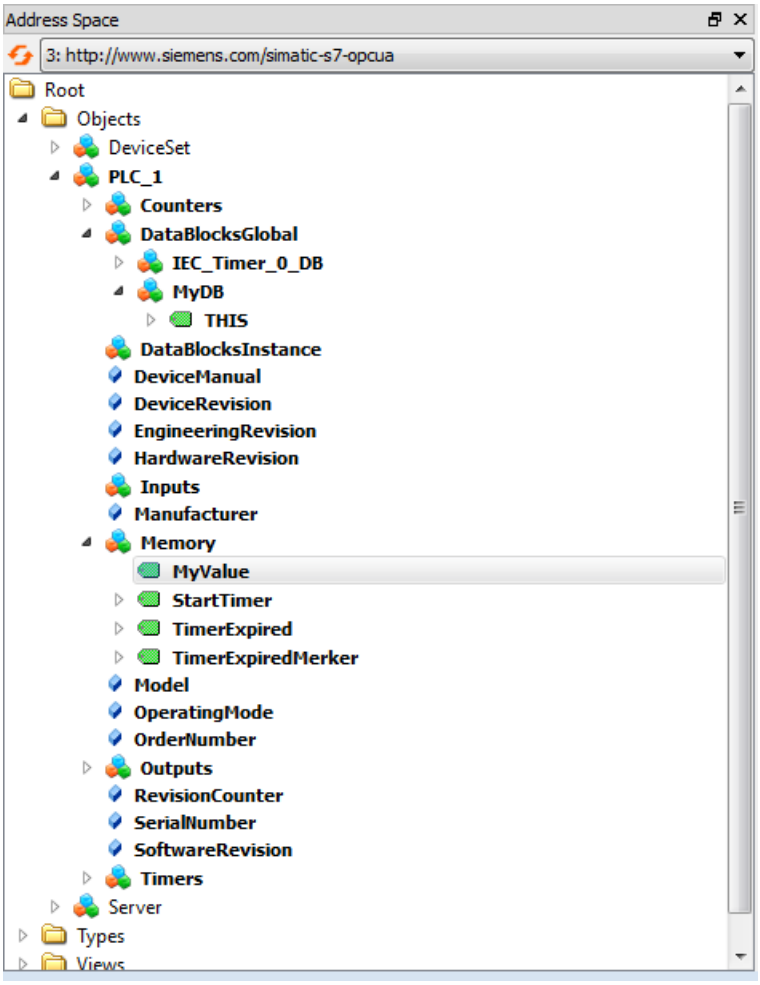


Bild 11-8 Beispiel für den Adressraum des OPC UA-Servers einer S7-1500 CPU

Im Bild oben ist die Variable "MyValue" markiert (grau unterlegt).
Diese Variable befindet sich unterhalb des Knotens "Memory", der die Knotenklasse "Object" besitzt.
"Memory" wiederum befindet sich unterhalb des Knotens "PLC_1" (ebenfalls ein Object).

Adressraum

Die Knoten sind untereinander über Referenzen verbunden, zum Beispiel über die Referenz "HasComponent", die eine hierarchische Beziehung zwischen einem Knoten und seinen unterlagerten Knoten wiedergibt. Über ihre Referenzen bilden die Knoten ein Netzwerk, das zum Beispiel die Form eines Baums besitzen kann.
Ein Netzwerk aus Knoten wird auch als Adressraum bezeichnet. Von der Wurzel ausgehend sind alle Knoten im Adressraum erreichbar.

11.3.1.2 Endpunkte der OPC UA-Server

Die Endpunkte der OPC UA-Server definieren die Sicherheitsstufe für eine Verbindung. Je nach Einsatzzweck oder gewünschter Sicherheitsstufe müssen Sie am Endpunkt die entsprechenden Einstellungen für die Verbindung vornehmen.

Verschiedene Security-Einstellungen

Vor dem Aufbau einer gesicherten Verbindung erfragen OPC UA-Clients beim Server, mit welchen Security-Einstellungen Verbindungen möglich sind. Der Server sendet eine Liste zurück mit allen Security-Einstellungen (Endpunkten), die der Server anbietet.

Aufbau von Endpunkten

Endpunkte bestehen aus den folgenden Komponenten:

- Kennung für OPC: "opc.tcp"
- IP-Adresse: 192.168.178.151 (in dem Beispiel)
- Port-Nummer für OPC UA: 4840 (Standard-Port)
  Die Port-Nummer ist konfigurierbar.
- Security-Einstellung für Nachrichten (Message Security-Modus): None, Sign, SignAndEncrypt.
- Verschlüsselungs- und Hash-Verfahren (Security Policy): None, Basic128Rsa15, Basic256, Basic256Sha256 (in dem Beispiel).

Das folgende Bild zeigt das Programm "UA Sample Client" der OPC Foundation.
Der Client hat eine gesicherte Verbindung zum OPC UA-Server einer S7-1500 CPU aufgebaut, zum Endpunkt "opc.tcp://192.168.178.151:4840 - [SignAndEncrypt: Basic256Sha256:Binary]". Die Security-Einstellungen "SignAndEncrypt:Basic256Sha256" sind im Endpunkt enthalten.

HINWEIS

Endpunkt mit möglichst hoher Security Policy auswählen

Wählen Sie für die Endpunkte eine für die Anwendung angemessene hohe Security Policy aus und deaktivieren Sie am OPC UA-Server die Security Policy mit geringerer Security Policy.

Für die sichersten Endpunkte (Basic256Sha256) des OPC UA-Servers der S7-1500 CPU ist ein Sha256-Zertifikat notwendig.

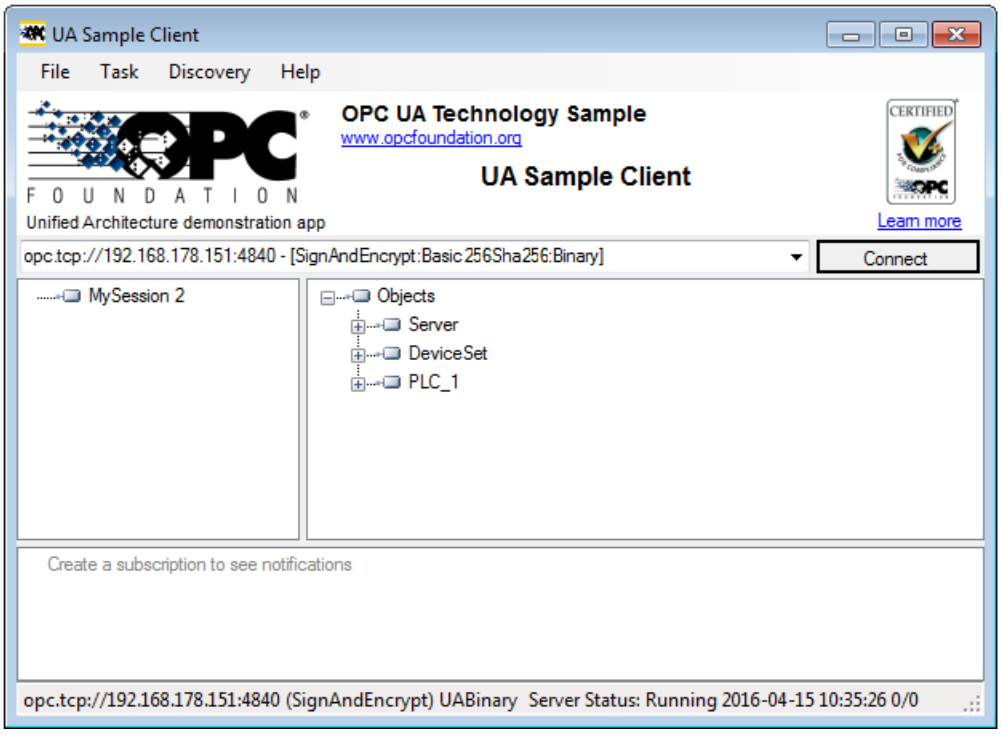


Bild 11-9 Programm "UA Sample Client" der OPC Foundation

Der Aufbau einer Verbindung zu einem Endpunkt des Servers kommt nur zu Stande, wenn der OPC UA-Client die geforderten Sicherheitseinstellungen dieses Endpunkts erfüllt.

Durch den OPC UA-Server bereitgestellte Informationen

OPC UA-Server stellen zahlreiche Informationen bereit:

- Die Werte von PLC-Variablen und DB-Komponenten, auf die Clients zugreifen dürfen.
- Die Datentypen dieser PLC-Variablen und DB-Komponenten.
- Angaben zum OPC UA-Server selbst und zur CPU.

Dadurch können sich Clients einen Überblick verschaffen und gezielt Informationen auslesen. Ein vorhergehendes Wissen über das SPS-Programm und den Datenhaushalt der CPU ist nicht erforderlich. Es ist nicht nötig, beim Entwickler des SPS-Programms nachzufragen, wenn PLC-Variablen gelesen werden sollen. Im Server selbst sind alle erforderlichen Angaben gespeichert (zum Beispiel die Datentypen der PLC-Variablen).

Anzeige der Informationen des OPC UA-Servers

Sie haben folgende Möglichkeiten:

- Online: Sie lassen sich zur Laufzeit des OPC UA-Servers alle verfügbaren Informationen anzeigen. Navigieren (Browsen) Sie dazu im Adressraum des Servers.
- Offline: Sie exportieren eine XML-Datei, die auf den XML-Schemata der OPC Foundation basiert.
  Selbst erstellte Server-Methoden (FB-Instanz, die von einem OPC UA-Client aufgerufen werden kann) werden ab STEP 7 V15.1 mit exportiert, siehe Methoden auf dem OPC UA-Server bereitstellen (Seite 290).
- Offline mit dem Openness-API: Sie verwenden in Ihrem Programm das API (Application Programming Interface) des TIA Portals, um die Funktion zum Export aller von OPC UA lesbaren PLC-Variablen aufzurufen. Dafür ist .NET Framework 4.0 erforderlich, siehe TIA Portal Openness, SIMATIC Projekte über Skripte automatisieren (https://support.industry.siemens.com/cs/ww/de/view/109477163).
- Wenn Sie bereits die Syntax und das SPS-Programm kennen, können Sie ohne vorhergehende Recherche auf den OPC UA-Server zugreifen.

11.3.1.3 Verhalten des OPC UA-Servers im Betrieb

Der OPC UA-Server im Betrieb

Der OPC UA-Server der S7-1500 CPU startet, wenn Sie den Server aktivieren und das Projekt in die CPU laden.
Wie Sie den OPC UA-Server aktivieren, ist hier beschrieben.

Verhalten beim Betriebszustand STOP der CPU

Ein aktivierter OPC UA-Server bleibt im Betrieb, auch wenn die CPU in den Betriebszustand "STOP" wechselt. Der OPC UA-Server antwortet dann nach wie vor auf Anfragen von OPC UA-Clients.
Das Verhalten des Servers im Einzelnen:

- Wenn Sie die Werte von PLC-Variablen abfragen, dann erhalten Sie die Werte, die aktuell waren, bevor die CPU in den Betriebszustand "STOP" wechselte oder gesetzt wurde.
- Wenn Sie Werte zum OPC UA-Server schreiben, dann übernimmt der OPC UA-Server diese Werte.
  Aber die CPU verarbeitet die Werte nicht, weil das Anwenderprogramm im Betriebszustand "STOP" nicht ausgeführt wird.
  Jedoch kann ein OPC UA-Client die bei STOP geschriebenen Werte aus dem OPC UA-Server der CPU lesen.
  Bei Wiederanlauf überschreibt die CPU die bei STOP geschriebenen Werte mit den Startwerten der PLC-Variablen.
- Wenn Sie eine Server-Methode aufrufen, dann erhalten Sie die Fehlermeldung 16#00AF_0000 (BadInvalidState), da die Server-Methode (Anwenderprogramm) nicht ausgeführt wird.
- Verbindungen zum OPC UA-Server bleiben bei einem Betriebszustandsübergang (STOP > RUN bzw. RUN > STOP) bestehen. Aunahme: Es werden OPC UA-relevante Daten geladen, siehen nächsten Abschnitt.

Laden der CPU kann OPC UA-Server beeinflussen

Wenn Sie eine CPU mit laufendem OPC UA-Server laden, dann kann es in Abhängigkeit von den geladenen Objekten notwendig sein, dass der Server stoppen und neu starten muss. In diesem Fall werden aktive Verbindungen unterbrochen und müssen nach dem Server-Neustart wieder aufgebaut werden.

Die Dauer des Neustarts ist vor allem von folgenden Parametern abhängig:

- Vom Umfang der Datenstruktur
- Von der Anzahl der Variablen, die im OPC UA Adressraum sichtbar sind
- Von der Einstellung zu abwärts kompatibler Datentyp-Definition nach der OPC UA-Spezifikation bis V1.03 (TypeDictionary aktiviert)
- Von den Einstellungen zur Kommunikationslast und Mindestzykluszeit, weitere Informationen finden Sie hier. (Seite 362)

Bei den FW-Versionen der CPU kleiner als V2.8 wurde der OPC UA-Server bei jedem Laden in die CPU gestoppt und anschließend neu gestartet.

Ab der FW-Version V2.8 wurde das Verhalten des OPC UA-Servers folgendermaßen optimiert:

- Beim Laden von Objekten im Betriebszustand STOP der CPU stoppt der OPC UA Server weiterhin grundsätzlich und startet anschließend wieder. STEP 7 zeigt in diesem Fall keine Warnung.
- Beim Laden von Objekten im Betriebszustand RUN der CPU stoppt der OPC UA-Server nur dann, wenn die geladenen Objekte OPC UA-relevant sind, bzw. OPC UA-relevant sein könnten. Nach einer Reinitialisierung wegen geänderter OPC UA-Daten startet der OPC UA-Server wieder.

  Bevor OPC UA-relevante Objekte in die CPU geladen werden und den OPC UA-Server stoppen, zeigt STEP 7 im Vorschau-Dialog zum Laden eine Warnung an. Sie können dann entscheiden, ob ein Server-Neustart verträglich ist für den laufenden Prozess oder ob Sie das Laden abbrechen. Diese Warnungen werden nur bei einem laufenden OPC UA-Server angezeigt. Wenn der OPC UA-Server nicht aktiviert ist, haben geänderte OPC UA-Daten keinen Einfluss auf den Ladevorgang.

Beispiele

- Sie wollen nur einen weiteren Codebaustein zum Programm hinzufügen.
  Weder Datenbausteine noch Eingänge, Ausgänge, Merker, Zeiten oder Zähler sind betroffen.
  Reaktion beim Laden: Ein laufender OPC UA-Server wird nicht unterbrochen.
- Sie wollen einen neuen Datenbaustein laden und Sie haben den Datenbaustein als nicht-OPC-UA-relevant gekennzeichnet.
  Reaktion beim Laden: Ein laufender OPC UA-Server wird nicht unterbrochen.
- Sie wollen einen Datenbaustein überschreiben.
  Reaktion beim Laden: Eine Warnung erscheint, dass der Server neu gestartet wird.
  Hintergrund: STEP 7 kann nicht ermitteln, ob sich die Änderungen auf OPC UA-relevante Daten beziehen oder nicht.

Betriebszustand der CPU über OPC UA-Server auslesen

Der OPC UA-Server erlaubt Ihnen, den Betriebszustand der CPU auszulesen, siehe folgendes Bild:

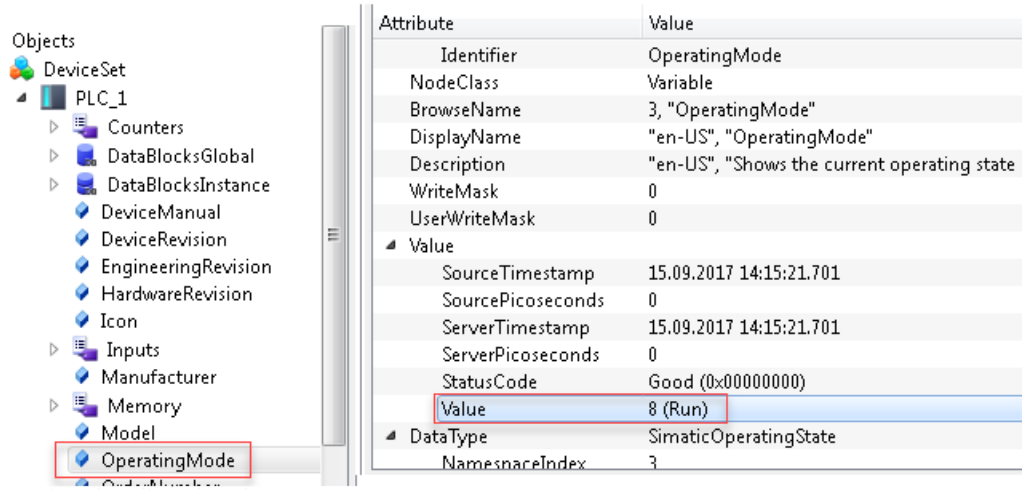


Bild 11-10 Betriebszustand der CPU über OPC UA-Server auslesen

Neben dem Betriebszustand der CPU können Sie z. B. auch Informationen über das Handbuch (DeviceManual) oder die Firmware Version (HardwareRevision) auslesen.

11.3.2 Zugriffsmöglichkeiten auf Daten des OPC UA-Servers

11.3.2.1 Client-Zugriffe und lokale Zugriffe auf den OPC UA-Server

Ein OPC UA-Server stellt innerhalb eines Netzwerks viele Informationen für OPC UA-Clients bereit. Im Folgenden geht es um die Möglichkeiten, wie CPU-Variablen (PLC-Variablen und DB-Elemente) im Adressraum des eigenen OPC UA-Servers zur Verfügung gestellt werden.

CPU-Variablen über Server-Schnittstellen im OPC UA-Adressraum bereitstellen

Der einfachste Weg, CPU-Variablen automatisch in den Adressraum des OPC UA-Servers zu übernehmen:

- Aktivieren Sie in den OPC UA-Eigenschaften der CPU die Standard-SIMATIC-Server-Schnittstelle.
  Dann sind alle für OPC UA freigegebenen CPU-Variablen automatisch auch im OPC UA-Adressraum unter dem Namen der CPU vorhanden.

Bild 11-11 Standard-SIMATIC-Server-Schnittstelle des OPC UA-Servers

Flexibler und übersichtlicher ist die Verwendung von OPC UA-Server-Schnittstellen; Sie projektieren die Server-Schnittstellen in der Projektnavigation (unterhalb der CPU, Ordner "OPC UA-Kommunikation"). Benutzerdefinierte OPC UA-Server-Schnittstellen ermöglichen Ihnen auf einfache Weise, OPC UA-Variablen und CPU-Variablen (Lokaldaten) zuzuordnen ("Mapping").

OPC UA-Server-Schnittstelle

Browse Name	Knotentyp	Zugriffsstufe	Lokaldaten
myServerInterface	Schnittstelle	---	
myOPC_UA_Variable	BOOL	RD	"myDataBlock"."myVarBool"
myOPC_UA_Variable2	Word	RD/WR	"myDataBlock"."myVarWord"

Bild 11-12 Benutzerdefinierte Server-Schnittstelle mit gemappten CPU-Variablen

Der Datenaustausch zwischen OPC UA-Client und OPC UA-Server ist in folgendem Beispiel zweier S7-1500 CPUs anschaulich dargestellt.
Eine S7-1500 CPU als Client schreibt hier Werte auf eine OPC UA-Variable des OPC UA-Servers. Durch das Mapping zwischen CPU-Variable und OPC UA-Variable wirkt es so, als würde der OPC UA-Client einen Wert direkt in die CPU-Variable schreiben. Für eine S7-1500-Client-CPU verwenden Sie die Anweisung "OPC_UA_WriteList" in Verbindung mit weiteren notwendigen Anweisungen für den Datenaustausch.

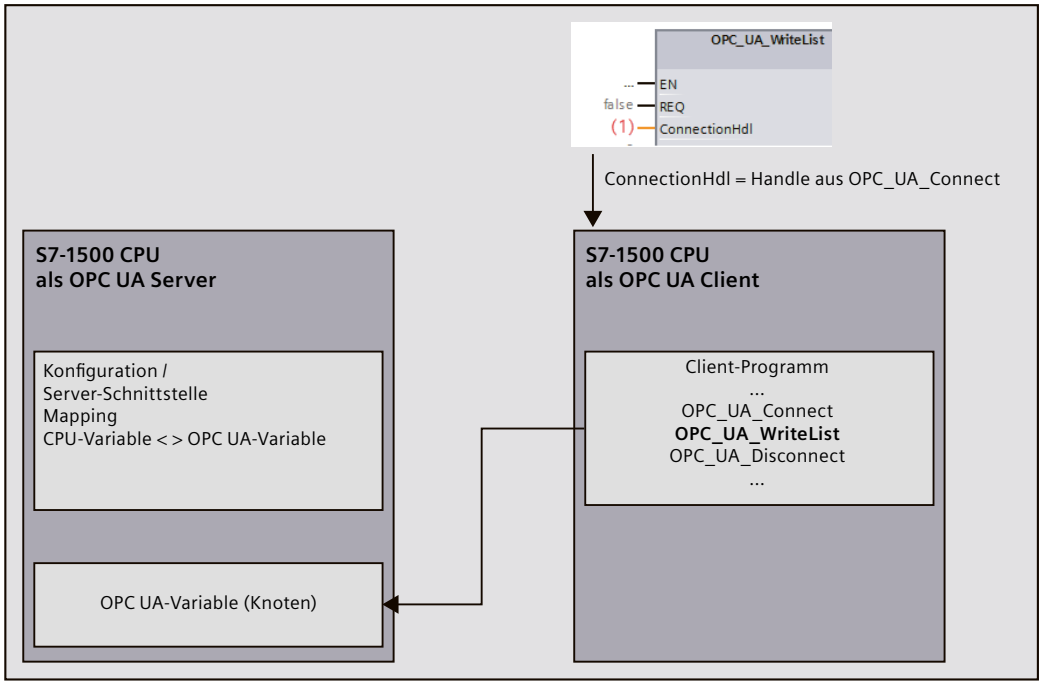


Bild 11-13 Client-Zugriff auf OPC UA-Variable des Servers

CPU-Variablenwerte direkt in OPC UA-Variable schreiben (OPC UA-DataValue setzen)

Ab Firmware-Version V3.0 bieten S7-1500 CPUs neben dem Mappen von Variablen zusätzlich die Möglichkeit, über die Anweisung "OPC_UA_WriteList" Werte direkt in lokale OPC UA-Variablen-Knoten des Servers zu schreiben. Normalerweise wird die Anweisung "OPC_UA_WriteList" im Client-Programm der CPU dazu benutzt, um Werte in OPC UA-Variablen eines entfernten OPC UA-Servers zu schreiben.

Vorteil der Verwendung von "OPC_UA_WriteList" im Server: Neben dem Wert (Value) können Sie dem OPC UA-Variablen-Knoten noch folgende Zusatzinformationen mitgeben:

- SourceTimestamp
- StatusCode

OPC UA stellt einen Built-In Datenyp "DataValue" zur Verfügung. DataValue ist eine Struktur, die sowohl den Wert (Value) als auch SourceTimestamp und StatusCode als Zusatzinformation zum Wert aufnimmt. Die DataValue-Struktur wird nur von OPC UA-Services genutzt und Sie können nicht direkt im Programm der CPU auf die Elemente dieser Struktur schreiben. Ein Schreibzugriff ist nur über eine besondere Verwendung der Anweisung "OPC_UA_WriteList" möglich.

Anwendungsmöglichkeiten

CPU-Variablen können keinen Zeitstempel aufnehmen, der angibt, wann ein Wert zuletzt auf die CPU-Variable geschrieben wurde. Wenn Sie also CPU-Variablen und OPC UA-Variablen über Server-Schnittstellen mappen, setzt der OPC UA-Server den SourceTimestamp nicht auf den Zeitpunkt, zu dem die CPU-Variable sich geändert hat, sondern auf den Zeitpunkt, zu dem der Wert im Server "aufgesammelt" wurde; z. B. durch eine Read-Dienst oder durch die Abtastung im Rahmen einer Subscription.

Wenn Sie DataValue direkt mit "OPC_UA_WriteList" in einen OPC UA-Variablen-Knoten schreiben, können Sie z. B. einen im Programm ermittelten Zeitstempel als SourceTimestamp für den Wert (Value) mitgeben.

Prinzipielle Funktionsweise der Anweisung "OPC_UA_WriteList" für das Setzen von DataValues

Die Struktur DataValue bilden Sie z. B. als UDT nach und übergeben der Anweisung "OPC_UA_WriteList" eine Variable dieses Datentyps. Die Anweisung transferiert dann die Elemente der Variable konsistent in den OPC UA-Variablen-Knoten.

Über den Wert des Anweisungsparameters "ConnectionHdl" wird die Funktionsweise der Anweisung "OPC_UA_WriteList" festgelegt: "Normale" Client-Anweisung oder Anweisung zum Schreiben auf lokale OPC UA-Variablen-Knoten. OPC UA-Clients können im letzteren Fall den Wert mit Zusatzinformationen lesen und entsprechend auswerten.

Das Prinzip finden Sie in den folgenden Bildern dargestellt, einmal mit einem beliebigen Client und einmal mit einer S7-1500 CPU als OPC UA-Client. Im Fall des S7-1500-CPU-Clients ist die Zuordnung der DataValue-Elemente zu den entsprechenden Anweisungsparametern der "OPC_UA_ReadList"-Anweisung dargestellt. Sie haben vollen Zugriff auf alle Elemente der DataValue-Struktur.

Der Wert von "Read" (-42) der Anweisung "OPC_UA_WriteList" bewirkt im Server, dass in lokale OPC UA-Variablen-Knoten geschrieben wird.

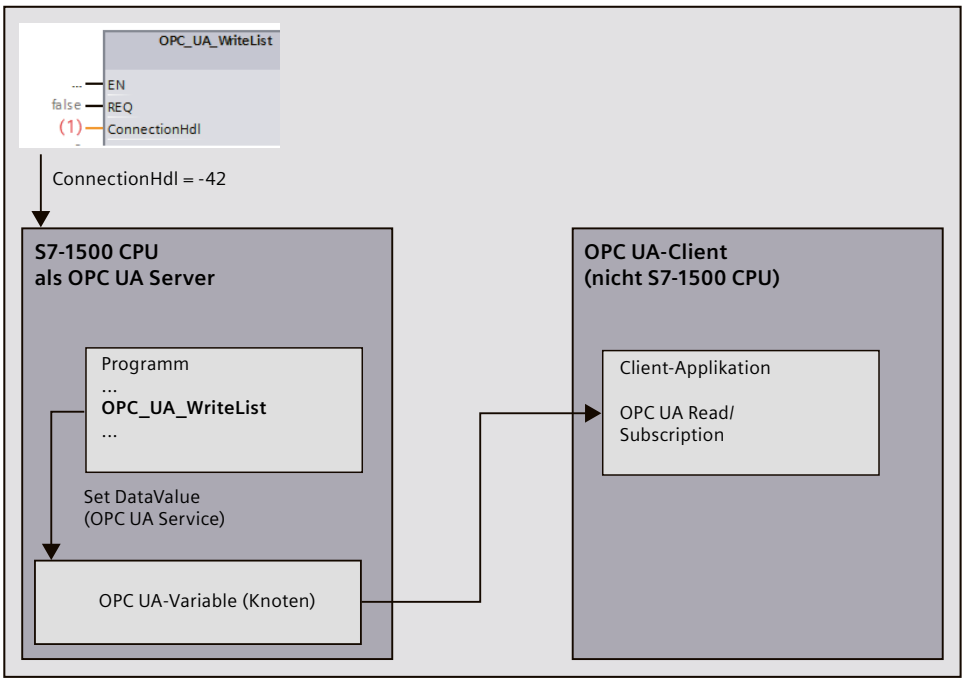


Bild 11-14 Set Data Value auf lokale OPC UA-Variable des Servers

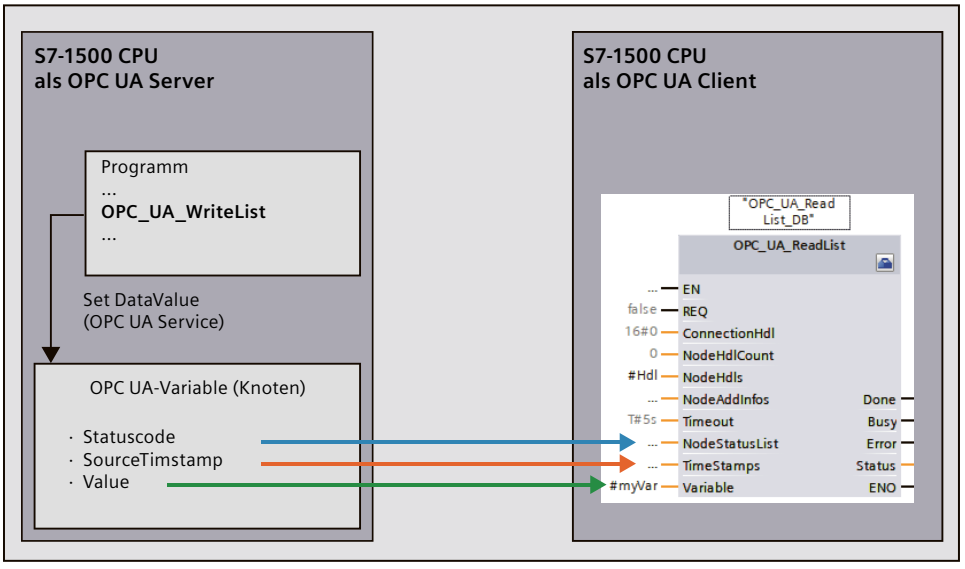


Bild 11-15 Client liest Data Value (OPC UA-Variable des Servers der S7-1500 CPU)

Weitere Anwendungsmöglichkeiten

Wenn sich OPC UA-Clients im Rahmen einer Subscription bei einer S7-1500 CPU für Wertänderungen anmelden (Monitored Items) und Sie den entsprechenden DataValue sowohl mit dem Wert (Value) als auch mit den oben genannten Zusatzinformationen versorgen, dann können auch Änderungen der Zusatzinformationen eine Benachrichtigung (Notification) auslösen.

Beispiel: Ein binärer Wert ändert sich so schnell, dass er im Abtastintervall wieder auf seinen ursprünglichen Wert zurückfällt (schneller Wechsel TRUE > FALSE > TRUE). Eine Änderung des Values wird nicht erkannt. Wohl aber die Änderung des Zeitstempels. Ähnlich kann eine Benachrichtigung bei einer Änderung des StatusCodes ausgelöst werden - auch ohne dass sich der Value ändert.

Randbedingungen

- OPC UA-Clients dürfen die OPC UA-Variable nur lesen; das Attribut "AccessLevel" für Schreib-/Leserechte ist entsprechend für die OPC UA-Variable zu setzen.
- Nur OPC UA-Variablen der benutzerdefinierten Server-Schnittstellen lassen sich lokal setzen.
- In der benutzerdefinierten Server-Schnittstelle darf es für die direkt beschriebenen OPC UA-Variablen kein Mapping auf CPU-Variablen geben.

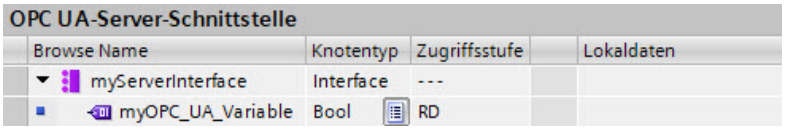


Bild 11-16 Benutzerdefinierte Server-Schnittstelle

Details zur Verwendung der Anweisung "OPC_UA_WriteList" im Kontext "OPC UA-DataValue setzen" finden Sie im entsprechenden Abschnitt der Hilfe zu Kommunikationsanweisungen.

OPC UA-DataValue-Attribute setzen bei Arrays und Strukturen

Beim Setzen von OPC UA-Variablen vom Typ Struktur oder Array über "OPC_UA_WriteList" für das Setzen von OPC UA-DataValue-Attributen werden alle Elemente dieses Arrays bzw. dieser Struktur befüllt.

Einzelne Elemente von Strukturen oder Arrays sollten Sie nicht zusätzlich als unterlagerte OPC UA-Variablen modellieren.

Grund: Wenn Sie einzelne Elemente von Arrays bzw. Strukturen zusätzlich als separaten Knoten im Adressraum des Servers als unterlagerten Knoten modellieren, dann werden diese Knoten **nicht** automatisch mitbefüllt. Diese separaten Knoten sind für den OPC UA-Server unabhängig von der übergeordneten OPC UA-Variable vom Typ Struktur bzw. Array, da sie nicht auf CPU-Variablen gemappt sind.

Um diese separat modellierten Knoten zu befüllen, müssen Sie die einzelnen Elemente als jeweils eigene DataValue-Strukturen im Programm anlegen.

Tipp: Damit OPC UA-Clients in diesem Fall die Möglichkeit haben, zeitgleich von Änderungen an den betroffenen Knoten zu erfahren, setzen Sie die Werte aller betroffenen OPC UA-Variablen im selben "OPC_UA_WriteList"-Aufruf.

Weitere Informationen

Zum Thema "OPC UA-DataValue-Attribute setzen" unterstützt Sie ein Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/109820694).

Informationen, wie Sie Schreib- und Leserechte für CPU-Variablen koordinieren, finden Sie im Kapitel Schreib- und Leserechte für CPU-Variablen koordinieren (Seite 225).

Informationen, wie Sie Benutzerdefinierte Server-Schnittstelle anlegen, finden Sie im Kapitel Benutzerdefinierte Server-Schnittstelle anlegen (Seite 269).

11.3.2.2 Schreib- und Leserechte verwalten

PLC-Variablen und DB-Variablen für OPC UA frei geben

OPC UA-Clients können auf PLC-Variablen und DB-Variablen lesend und schreibend zugreifen, wenn die Variablen für OPC UA freigegeben sind (Voreinstellung). Bei freigegebenen Variablen ist das Optionskästchen bei "Erreichbar aus HMI/OPC UA" aktiviert.

Die Voreinstellung können Sie in den Einstellungen des TIA Portals ändern: Befehl "Einstellungen > PLC-Programmierung > Allgemein" im Menü "Extras". Im Bereich "Bausteinschnittstelle/Datenbausteinelemente" finden Sie die entsprechenden Optionen.

Das folgende Beispiel zeigt einen Array-Datenbaustein:

MyDB

Name	Datentyp	Erreichbar aus HMI/OPC UA	Schreibbar aus HMI/OPC UA	Sichtbar in HMI Engineering
MyDB	Arr...	☑	☑	☑
MyDB[0]	Int	☑	☑	☑
MyDB[1]	Int	☑	☑	☑
MyDB[2]	Int	☑	☑	☑
MyDB[3]	Int	☑	☑	☑
MyDB[4]	Int	☑	☑	☑
MyDB[5]	Int	☑	☑	☑
MyDB[6]	Int	☑	☑	☑
MyDB[7]	Int	☑	☑	☑
MyDB[8]	Int	☑	☑	☑
MyDB[9]	Int	☑	☑	☑

Bild 11-17 PLC-Variablen und DB-Variablen für OPC UA-Variablen frei geben

Das Array kann von OPC UA-Clients komplett in einem Zug gelesen werden (siehe Adressierung von Knoten (Seite 172)). Bei allen Komponenten des Arrays sind die Optionskästchen "Erreichbar aus HMI/OPC UA" und "Schreibbar aus HMI/OPC UA" aktiviert.

Folge: OPC UA-Clients dürfen diese Komponenten sowohl lesen als auch schreiben.

Schreibrechte entziehen

Wenn Sie eine Variable gegen Schreibzugriffe schützen wollen, dann deaktivieren Sie bei dieser Variablen die Option "Schreibbar aus HMI/OPC UA". Dadurch entziehen Sie OPC UA-Clients und HMI-Geräten das Schreibrecht.

Folge: Ausschließlich lesende Zugriffe durch OPC UA-Clients und durch HMI-Geräte sind möglich. OPC UA-Clients können dieser Variable keine Werte zuweisen und dadurch keinen Einfluss auf den Ablauf des S7-Programms nehmen.

Schreib- und Leserechte entziehen

Um eine Variable gegen Schreib- und Lesezugriffe zu schützen, deaktivieren Sie bei dieser Variablen die Option "Erreichbar aus HMI/OPC UA" (Häkchen nicht gesetzt). Dadurch entfernt der OPC UA-Server diese Variable aus seinem Adressraum. OPC UA-Clients sehen diese CPU-Variable nicht mehr.

Folge: OPC UA-Clients und HMI-Geräte können diese Variable weder lesen noch schreiben.

Schreib- und Leserechte von Strukturen

Wenn Sie für eine Komponente einer Struktur das Schreib- oder Leserecht entziehen, dann kann die Struktur oder der Datenbaustein nicht mehr als Ganzes beschrieben oder gelesen werden.
Wenn Sie Schreib- und Leserechte einzelnen Komponenten eines PLC-Datentyps (UDT) entziehen, dann sind die Rechte auch in einem auf dem UDT basierenden Datenbaustein entzogen!

Sichtbar in HMI Engineering

Die Option "Sichtbar in HMI Engineering" bezieht sich auf Engineering-Tools von Siemens. Wenn Sie die Option "Sichtbar in HMI Engineering" deaktivieren (Häkchen nicht gesetzt), dann können Sie die Variable nicht mehr in WinCC (TIA Portal) projektieren.
Die Option hat keine Auswirkungen auf OPC UA.

Regeln

- Erlauben Sie in STEP 7 nur dann lesende Zugriffe auf PLC-Variablen und Variablen von Datenbausteinen, wenn es für die Kommunikation zu anderen Systemen (Steuerungen, eingebetteten Systemen, MES) erforderlich ist.
  Andere PLC-Variablen sollten Sie nicht frei geben.
- Gewähren Sie nur dann schreibende Zugriffe über OPC UA, wenn Schreibrechte tatsächlich bei bestimmten PLC-Variablen und Variablen von Datenbausteinen erforderlich sind.
- Wenn Sie für alle Elemente eines Datenbausteins die Option "Erreichbar aus HMI/OPC UA" zurückgesetzt haben, dann ist der Datenbaustein für einen OPC UA-Client im Adressraum des OPC UA-Servers der S7-1500 CPU nicht mehr sichtbar.
- Sie können auch zentral den Zugriff auf einen gesamten Datenbaustein verhindern (siehe Schreib- und Leserechte für kompletten DB verwalten (Seite 224)). Diese Einstellung "überstimmt" die Einstellungen an den Komponenten im DB-Editor.

Weitere Informationen

Informationen, wie Sie Schreib- und Leserechte für CPU-Variablen koordinieren, finden Sie im Kapitel Schreib- und Leserechte für CPU-Variablen koordinieren (Seite 225).

11.3.2.3 Schreib- und Leserechte für kompletten DB verwalten

DBs oder DB-Inhalte für OPC UA-Clients verbergen

Sie haben die Möglichkeit, den Zugriff auf einen kompletten Datenbaustein durch einen OPC UA-Client auf einfache Weise zu verhindern.

Auf diese Weise bleiben die Daten des entsprechenden DBs, auch Instanz-DBs von Funktionsbausteinen, für OPC UA-Clients verborgen.

Voreingestellt ist, dass Datenbausteine von OPC UA-Clients lesbar und schreibbar sind. Diese Voreinstellung können Sie in den Einstellungen des TIA Portals ändern: Befehl "Einstellungen > PLC-Programmierung > Allgemein" im Menü "Extras". Im Bereich "Voreinstellung für neue Bausteine" finden Sie die entsprechende Option.

Vorgehen

Um einen Datenbaustein für OPC UA-Clients komplett zu verbergen bzw. um einen Datenbaustein vor Schreibzugriffen von OPC UA-Clients zu schützen, gehen Sie folgendermaßen vor:

1. Markieren Sie in der Projektnavigation den zu schützenden Datenbaustein.
2. Wählen Sie das Kontextmenü "Eigenschaften".
3. Wählen Sie den Bereich "Attribute".
4. Aktivieren/Deaktivieren Sie die Optionskästchen "DB erreichbar aus OPC UA" nach Ihren Erfordernissen.

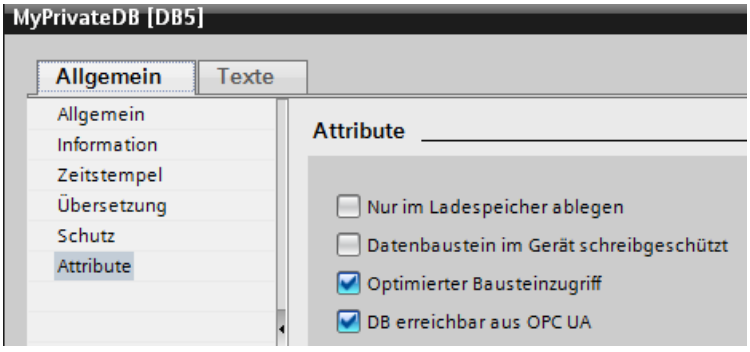


Bild 11-18 DBs oder DB-Inhalte für OPC UA-Clients verbergen

HINWEIS

Beeinflussung der Einstellungen im DB-Editor

Wenn Sie über das hier beschriebene DB-Attribut einen DB verbergen, dann sind die Einstellungen an den Komponenten im DB-Editor nicht mehr relevant; einzelne Komponenten können nicht mehr erreicht oder beschrieben werden.

Tipp: Übersicht aller Programmbausteine nutzen

Falls Sie mehrere Datenbausteine nutzen, bietet es sich an, die Detailübersicht des Ordners "Programmbausteine" für ein gezieltes Ein- oder Ausschalten der OPC UA-Erreichbarkeit zu verwenden.

Gehen Sie folgendermaßen vor:

1. Markieren Sie in der Projektnavigation den Ordner "Programmbausteine".
2. Aktivieren Sie den Befehl "Übersicht" im Menü "Ansicht".
3. Wählen Sie das Register "Details".
   Eine Übersicht der Bausteine mit ihren Attributen wird angezeigt.
4. Vergewissern Sie sich, das die Spalte "Datenbaustein erreichbar über OPC UA" aktiviert ist.
5. Aktivieren Sie nur die Datenbausteine, die über OPC UA erreichbar sein sollen.

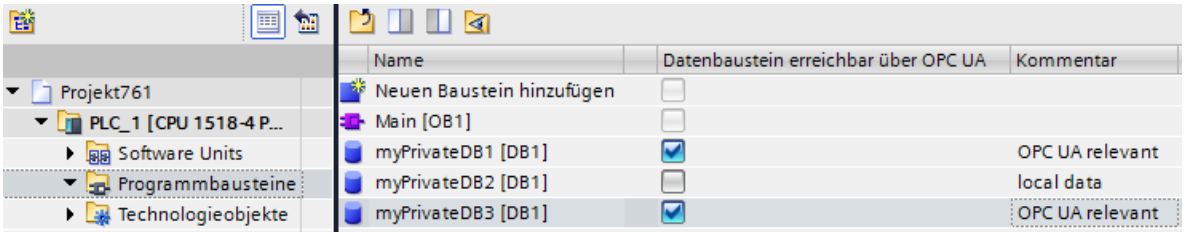


Bild 11-19 Übersicht über die Programmbausteine

11.3.2.4 Schreib- und Leserechte für CPU-Variablen koordinieren

Definition von Schreib- und Leserechten im Informationsmodell (OPC UA-XML)

Im OPC UA-Informationsmodell regelt das Attribut "AccessLevel" den Zugriff auf Variablen.

AccessLevel ist bitweise definiert:

Bit 0 = CurrentRead und Bit 1 = CurrentWrite. Die Bedeutung der Bitkombinationen ergibt sich wie folgt:

- AccessLevel = 0: kein Zugriff
- AccessLevel = 1: read only
- AccessLevel = 2: write only
- AccessLevel = 3: read+write

Beispiel für die Vergabe von Schreib- und Leserechten (read+write)

```
<UAVariable NodeId="ns=3;s=&quot;Data_block_2&quot;.&quot;Static_1&quot;"
BrowseName="3:Static_1"
ParentNodeId="ns=3;s=&quot;Data_block_2&quot;"
DataType="INT"
AccessLevel="3">
    <DisplayName>Static_1</DisplayName>
```

Definition von Schreib- und Leserechten in STEP 7

Beim Definieren von Variablen legen Sie die Zugriffsrechte fest mit den Eigenschaften "Erreichbar aus HMI/OPC UA" und "Schreibbar aus HMI/OPC UA".

Beispiel für die Vergabe von Schreib- und Leserechten

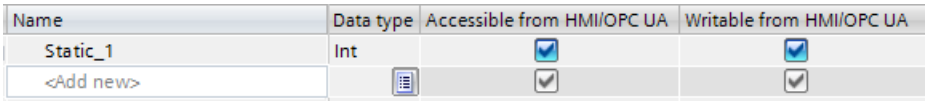


Bild 11-20 Beispiel für die Vergabe von Schreib- und Leserechten

Zusammenspiel der Schreib- und Leserechte

Wenn Sie eine OPC UA-Server-Schnittstelle importiert haben und in dieser OPC UA-XML-Datei sind AccessLevel-Attribute gesetzt, dann resultieren die Schreib- und Leserechte aus der Regel: Wirksam sind die geringsten Zugriffsrechte aus beiden Einstellungen.

Beispiel

- AccessLevel = 1 (read only) in der OPC UA-Server-Schnittstelle
- Sowohl "Erreichbar aus HMI/OPC UA" als auch "Schreibbar aus HMI/OPC UA" ist in der PLC-Variablentabelle aktiviert

Ergebnis: Die Variable kann nur gelesen werden.

Regeln

Wenn Schreibrechte erforderlich sind:

- AccessLevel = 2 oder 3
- "Schreibbar aus HMI/OPC UA" aktiviert

Wenn Leserechte erforderlich sind:

- AccessLevel = 1 (AccessLevel 3 ist auch möglich, aber irreführend. Die Einstellung suggeriert, ein OPC UA-Client hat Schreib- und Leserechte)
- "Erreichbar aus HMI/OPC UA" aktiviert, "Schreibbar aus HMI/OPC UA" deaktiviert

Wenn weder Lese- noch Schreibrechte eingeräumt werden sollen (kein Zugriff):

- AccessLevel = 0
- "Erreichbar aus HMI/OPC UA" deaktiviert
  Um jeglichen Zugriff zu sperren, muss nur eine der beiden Bedingungen erfüllt sein. Überdenken Sie in diesem Fall, ob die Variable in der OPC UA-Server-Schnittstelle überhaupt benötigt wird.

Zugriffstabelle

"Erreichbar aus HMI/OPC UA" muss gesetzt sein, damit überhaupt ein Zugriff per OPC UA möglich ist. "Schreibbar aus HMI/OPC UA" muss gesetzt sein, damit ein OPC UA-Client auf eine Variable / ein DB-Element schreiben kann.
Der Tabelle entnehmen Sie das resultierende Zugriffsrecht.

Tabelle 11-2 Zugriffstabelle

OPC UA-XML	**STEP 7 (TIA Portal) z. B. Variablentabelle**		
AccessLevel	**Erreichbar aus HMI/OPC UA**	**Schreibbar aus HMI/OPC UA**	**Resultierendes Zugriffs-recht**
0	x	x	Kein Zugriff
x	0	x	Kein Zugriff
1	aktiviert	x	Read only
2	aktiviert	deaktiviert	Kein Zugriff
3	aktiviert	deaktiviert	Read only
2	aktiviert	aktiviert	Write only
3	aktiviert	aktiviert	Read+write

(x = don't care)

11.3.2.5 Konsistenz von CPU-Variablen

Attribut "AccessLevelEx" erweitert Zugriffseigenschaften

Der OPC UA-Server der S7-1500 CPU unterstützt ab Firmware Version V2.6 neben dem Attribut "AccessLevel" (siehe Schreib- und Leserechte für CPU-Variablen koordinieren (Seite 225)) auch das Attribut "AccessLevelEx", das neben den bereits erläuterten Bits für den Lese- und Schreibzugriff zusätzlich Auskunft über die Konsistenz einer OPC UA-Variable gibt. Das neue Attribut wurde mit Version V1.04 der OPC UA-Spezifikation eingeführt (Part 3, Address Space Model).

Konsistenz-Eigenschaften auslesen

Im OPC UA-Informationsmodell des OPC UA-Servers definiert das Attribut "AccessLevelEx" den Zugriff auf Variablen.

AccessLevelEx ist bitweise definiert, die hier relevanten Bits sind folgende:

- Bit 0 = CurrentRead
- Bit 1 = CurrentWrite
- Bit 2 bis 7 sind nicht relevant für den OPC UA-Server einer S7-1500 CPU

Die Bedeutung der Bitkombinationen ist im Abschnitt zu den Lese- und Schreibrechten erläutert.

Zusätzlich kommen folgende Bits für die Konsistenzeigenschaften hinzu:

- Bit 8 = NonatomicRead; das Bit ist gesetzt, wenn die Variable nicht konsistent gelesen werden kann. Bei Lese-Konsistenz der Variablen ist Bit 8=0.
- Bit 9 = NonatomicWrite; ist gesetzt, wenn die Variable nicht konsistent geschrieben werden kann. Bei Schreib-Konsistenz der Variablen, bzw. wenn kein schreibender Zugriff gewährt ist, dann ist Bit 9=0.

Beispiele

Eine OPC UA-Variable (Struktur) ist lesbar und schreibbar; aber inkonsistent für lesenden und für schreibenden Zugriff.

Daraus folgt: Die Bits 0, 1, 8 und 9 sind gesetzt: AccessLevelEx = "771" (1+2+256+512).

Eine andere Struktur ist nur lesbar.

Daraus folgt: Die Bits 0 und 8 sind gesetzt, Bit 1 und Bit 9 sind nicht gesetzt: AccessLevelEx = "257" (1+0+256+0).

Behandlung des Attributs im Server

Das Attribut "AccessLevelEx" steht nur im OPC UA-Server zur Verfügung. Das Attribut ist nicht in einer Nodeset-Datei (XML-Export-Datei) vorhanden.
Allerdings nimmt das Attribut "AccessLevel", das exportiert wird, die Information von "AccessLevelEx" mit auf, siehe nächsten Abschnitt.

Export
Beim XML-Export der Standard-SIMATIC-Server-Schnittstelle setzt der Server das Attribut "AccessLevel", das im Unterschied zur V1.03-Spezifikation in V1.04 auf 32 Bit erweitert wurde, auf den Wert von Attribut "AccessLevelEx".

Import
Beim Import einer Nodeset-Datei (z. B. aus einem Export einer Server-Schnittstelle) setzt die S7-1500 CPU das Attribut "AccessLevelEx" nach ihrem eigenen Wissen um die Konsistenz des importierten Datentyps, siehe nächsten Abschnitt. Der importierte Wert wird ignoriert.

Konsistenz von Datentypen an der Server-Schnittstelle

Die Konsistenz von Variablen (im Sprachgebrauch von OPC UA: "atomicity") innerhalb eines Programmzyklus einer S7-1500 CPU ist an den Knoten der Server-Schnittstelle sichergestellt für folgende Datentypen:

- BOOL, BYTE, WORD, DWORD, LWORD
- SINT, INT, LINT, DINT, USINT, UINT, ULINT, UDINT
- REAL, LREAL
- DATE, LDT, TIME, LTIME, TIME_OF_DAY, LTIME_OF_DAY, S5TIME
- CHAR, WCHAR
- Ebenfalls konsistent sind Systemdatentypen bzw. Hardware-Datentypen, die auf den oben genannten Datentypen basieren.
  Beispiel: HW_ANY, von UINT (UInt16) abgeleitet.

Tipp: Wenn Sie im Adressraum der S7-1500 CPU browsen (z. B. mit dem OPC UA Client UaExpert), dann finden Sie die konsistenten Datentypen unter Types > BaseDataType > Enumeration/Number/String.

Variablen von folgenden Datentypen sind **nicht** konsistent (im Sprachgebrauch von OPC UA: "nonatomic"):

- SIMATIC-Strukturen sind generell nicht konsistent. D. h. alle Variablen, die z. B. unbenannte Strukturen sind oder einen UDT-Datentyp haben, sind nicht konsistent.
- Systemdatentypen wie z. B. DTL, IEC_Counter, IEC_TIMER etc. – das sind Datentypen, die von Strukturen abgeleitet sind.
- Strings (Array of Char) sind nicht konsistent.

Tipp: Wenn Sie im Adressraum der S7-1500 CPU browsen (z. B. mit dem OPC UA Client UaExpert), dann finden Sie Datentypen, die auf Strukturen basieren, unter Types > BaseDataType > Structure.

11.3.2.6 Schreibzugriffe auf OPC UA-Variablen von S7-1500 Motion Control

Zusätzlich zur Konsistenz der Datentypen überprüft die CPU die Variablen der Technologieobjekte auf Plausibilität und Gültigkeit.

Wenn ein OPC UA-Client einen ungültigen oder nicht plausiblen Wert in eine Variable schreibt, dann bleibt der ursprüngliche Wert in der Varible des Technologieobjekts erhalten.

Trotz erfolglosem Schreibzugriff wird der Status "Good" ausgegeben.

Beispiel 1

Interpolationsart der Kurvenscheibe

Die Variable "Cam_1".InterpolationSettings.InterpolationMode ist vom Typ INT, darf aber nur die Werte 1..2 annehmen.

Wenn Sie die Variable über OPC UA auf einen ungültigen Wert ändern wollen, z. B. 3, dann wird der Status Code "Good" ausgegeben, die Variable aber nicht geändert.

Beispiel 2

Positionen der SW-Endschalter bei einer Positionierachse

Die Position des positiven HW-Endschalters muss positiver sein als die Position des negativen SW-Endschalters.

"PosAxis_1".PositionLimits_SW.MaxPosition > "PosAxis_1".PositionLimits_SW.MinPosition

Wenn Sie eine Variable über OPC UA auf einen Wert ändern wollen, der diese Bedingung nicht erfüllt, dann wird der Status Code "Good" ausgegeben, die Variable aber nicht geändert.

Welche Werte für die Variablen der Technologieobjekte gültig sind, entnehmen Sie der Dokumentation zu den Technologieobjekten (https://support.industry.siemens.com/cs/ww/de/view/109751049).

11.3.2.7 Zugriffsmöglichkeiten auf Daten des OPC UA-Servers

Hohe Performance abhängig vom Anwendungsfall

OPC UA ist für die Übertragung vieler Daten in kurzer Zeit ausgelegt. Sie können die Leistung deutlich steigern, wenn Sie nicht auf einzelne PLC-Variablen zugreifen, sondern Arrays und Strukturen als Ganzes lesen und schreiben.

Am schnellsten ist der Zugriff auf Arrays. Deshalb sollten Sie die Daten für OPC UA-Clients in Arrays zusammenfassen.

Empfehlungen für den Zugriff auf den OPC UA-Server durch den OPC UA-Client

- Nutzen Sie für den einmaligen oder seltenen Datenzugriff den normalen Read/Write-Zugriff.
- Nutzen Sie für den zyklischen Zugriff auf wenige Daten (bis ca. alle 5 Sekunden) Subscriptions.

  Optimieren Sie am OPC UA-Server die Einstellungen für das kleinste Sendeintervall und das kleinste Abtastintervall.
- Nutzen Sie für einen regelmäßigen (wiederkehrenden) Zugriff auf bestimmte Variablen die Funktionen "RegisteredRead" und "RegisteredWrite".

Gewähren Sie der CPU mehr Kommunikationslast, indem Sie den Wert für "Zyklusbelastung durch Kommunikation" erhöhen. Vergewissern Sie sich, dass Ihre Anwendung mit den geänderten Einstellungen noch ordnungsgemäß funktioniert.

Vorgehensweise zum Anlegen eines Arrays-DBs

Arrays können sie z. B. in globalen Datenbausteinen, im Instanzdatenbaustein eines Funktionsbausteins oder als Array-DB anlegen. Im Folgenden ist beschrieben, wie Sie einen Array-DB anlegen.

Um einen Datenbaustein mit einem Array (Array-Datenbaustein) anzulegen, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation die CPU mit dem OPC UA-Server.
2. Doppelklicken Sie auf "Programmbausteine".
3. Doppelklicken Sie auf "Neuen Baustein hinzufügen".
4. Klicken Sie auf die Schaltfläche "Datenbaustein".
5. Wählen Sie einen eindeutigen Namen für den Datenbaustein oder übernehmen Sie den bereits eingetragenen Namen.
6. Wählen Sie in der Klappliste bei "Typ" den Eintrag "Array-DB".
7. Wählen Sie in der Klappliste bei "Array-Datentyp" den Datentyp für die einzelnen Komponenten des Arrays.
8. Tragen Sie die obere Grenze des Arrays bei "Array-Grenze" ein.
9. Klicken Sie auf die Schaltfläche "OK".

11.3.2.8 Attribut MinimumSamplingInterval

MinimumSamplingInterval-Attribut von Variablen

Neben "Value", "DataType" und "AccessLevel" können Sie in der XML-Datei, die den Server-Adressraum repräsentiert, auch das Attribut "MinimumSamplingInterval" für eine Variable setzen.

Das Attribut gibt an, wie schnell der Server den Variablenwert abtasten kann.

Der OPC UA-Server der S7-1500 CPU geht folgendermaßen mit den Werten für MinimumSamplingInterval um:

- Negative Werte und Werte größer als 4294967 werden auf -1 gesetzt; das bedeutet: Die minimale Abtastrate ist unbestimmt; der Server gibt nicht an, wie schnell der Variablenwert abgetastet werden kann.
- Kommazahlen werden auf drei Nachkommastellen gerundet.

11.3.2.9 OPC UA-XML-Datei exportieren

OPC UA-Exportdatei erzeugen

Die OPC Foundation hat ein Standard-Format, basierend auf XML, zum Beschreiben von Informationsmodellen spezifiziert. Damit kann einem Client vorab das Informationsmodell eines OPC UA-Servers zur Verfügung gestellt werden oder es können Informationsmodelle auf einen OPC UA-Server geladen werden. Eine Datei in diesem Format wird Nodeset-Datei genannt, weil es ein Informationsmodell als Menge (Set) von Knoten (Nodes) beschreibt.

Mit STEP 7 (TIA Portal) können Sie auf einfache Weise das Standard-SIMATIC-Informationsmodell der S7-1500 CPU als Server in eine OPC UA-XML-Datei (Nodeset-Datei) exportieren; inklusive folgender Elemente, die Sie für OPC UA freigegeben haben:

- CPU-Variablen (PLC-Variablen und DB-Elemente)
- Funktionsbausteine mit ihren Eingängen/Ausgängen

Elemente, die in der CPU existieren, aber nicht im Programm verwendet werden, tauchen nach dem Export auch nicht in der OPC UA-XML-Datei auf. Beispiele für solche nicht verwendeten Elemente sind:

- UDTs, die keinem Datenbaustein zugeordnet sind
- Funktionsbausteine mit Ein-/Ausgängen ohne Zuordnung der Ein-/Ausgänge zu CPU-Variablen

Die OPC UA-XML-Datei nutzen Sie für die Offline-Projektierung eines OPC UA-Clients; sie ist gemäß der OPC UA-Spezifikation aufgebaut und fungiert als Standard-SIMATIC-Server-Schnittstelle.

Um die OPC UA-XML-Datei zu erzeugen und zu exportieren, gehen Sie folgendermaßen vor:

1. Wählen Sie die CPU aus. Klicken Sie dazu auf das Symbol der CPU (z. B. in der Netzsicht).
2. Klicken Sie in den Eigenschaften der CPU auf "Allgemein > OPC UA > Server > Exportieren".
3. Klicken Sie auf die Schaltfläche "OPC UA XML-Datei exportieren":
4. Wählen Sie das Verzeichnis, in dem Sie die Exportdatei speichern wollen.
5. Wählen Sie einen neuen Namen für die Datei. Oder behalten Sie den Namen bei, der bereits eingetragen ist.
6. Klicken Sie auf "Speichern".

HINWEIS

Server-Methoden sind ab STEP 7 (TIA Portal) V15.1 in der OPC UA-Exportdatei (Nodeset) mit ihren Ein- und Ausgangsparametern enthalten.

Alle Arrayelemente separat exportieren

Wenn in den CPU-Eigenschaften "OPC UA > Server > Exportieren" die Option "Alle Arrayelemente als separate Knoten exportieren" aktiviert ist, dann enthält die OPC UA XML-Datei alle Elemente von Arrays jeweils als einzelne XML-Elemente. In der XML-Datei sind außerdem die Arrays selbst jeweils in einem XML-Element beschrieben.

Wenn viele Array-Elemente in einem Array enthalten sind, kann die XML-Datei sehr umfangreich werden.

Tipp

Im folgenden FAQ finden Sie einen Konverter, mit dem Sie die Exportdatei in das CSV-Format wandeln können. Sie erhalten damit eine Liste der für OPC UA erreichbaren Variablen der CPU.
Den FAQ finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/view/109742903).

11.3.3 OPC UA-Server konfigurieren

11.3.3.1 OPC UA-Server aktivieren

Voraussetzung

- Wenn Sie **Zertifikate** zur gesicherten Kommunikation nutzen z. B. HTTPS, Secure OUC, OPC UA, dann achten Sie darauf, dass die betroffenen Baugruppen über die **aktuelle Uhrzeit und das aktuelle Datum** verfügen. Die Baugruppen werten die verwendeten Zertifikate sonst als ungültig und die gesicherte Kommunikation funktioniert nicht.
- Sie haben eine Runtime-Lizenz für den Betrieb der OPC UA-Funktionen erworben, siehe Lizenzen für OPC UA (Seite 255).

OPC UA-Server in Betrieb nehmen

In der Grundeinstellung ist der OPC UA-Server der CPU aus Sicherheitsgründen nicht freigegeben: OPC UA-Clients können weder schreibend noch lesend auf die S7-1500 CPU zugreifen.
Um den OPC UA-Server der CPU zu aktivieren, gehen Sie folgendermaßen vor.

1. Wählen Sie die CPU aus. Klicken Sie dazu auf das Symbol der CPU (z. B. in der Netzsicht).
2. Klicken Sie in den Eigenschaften der CPU auf "OPC UA > Server".
3. Aktivieren Sie den OPC UA-Server der CPU.
4. Bestätigen Sie die Sicherheitshinweise.
5. Wählen Sie bei den CPU-Eigenschaften den Bereich "Runtime-Lizenzen" und stellen die erworbene Runtime-Lizenz für den OPC UA-Server ein.
6. Kompilieren Sie das Projekt.
7. Laden Sie das Projekt in die CPU.
   Der OPC UA-Server der CPU startet nun.

Einstellungen bleiben gespeichert

Falls Sie den Server bereits aktiviert und Einstellungen vorgenommen hatten, dann gehen diese Einstellungen nicht verloren, wenn Sie den Server deaktivieren. Die Einstellungen sind nach wie vor gespeichert und stehen wieder zur Verfügung, wenn Sie den Server wieder aktivieren.

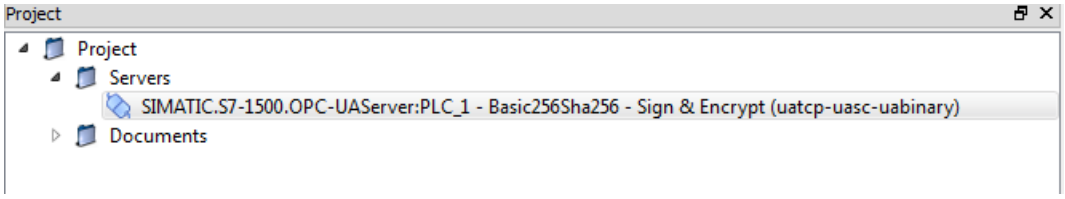
Applikationsname

Der Applikationsname ist der Name der OPC UA-Applikation und gilt für den Server und den Client. Der Name wird angezeigt unter "OPC UA > Allgemein":

- Die Voreinstellung für den Applikationsnamen lautet: "SIMATIC.S7-1500.OPC-UA.Application:PLC_1".
- Die Voreinstellung setzt sich aus "SIMATIC.S7-1500.OPC-UA.Application:" und dem Namen der CPU zusammen, wie er unter "Allgemein > Produktinformation > Name" gewählt wurde, hier "PLC_1".
- Über diesen Applikationsnamen identifiziert sich der OPC UA-Server gegenüber einem Kommunikationspartner (OPC UA Client), z. B. wenn ein OPC UA-Client den Discovery-Service nutzt, um erreichbare Server zu ermitteln.
- Den angezeigten Applikationsnamen verwendet der OPC UA-Client der CPU beim Verbindungsaufbau zu einem OPC UA-Server. D. h. die CPU trägt diesen Applikationsnamen automatisch als "ApplicationName" für die Anweisung "OPC_UA_Connect" ein (Variable vom Typ "OPC_UA_SessionConnectInfo" am Parameter "SessionConnectInfo" der Anweisung "OPC_UA_Connect").

  Daher müssen Sie beim Programmieren der Anweisung "OPC_UA_Connect" den "ApplicationName" mit einem Leerstring belegen. Mit dem Applikationsnamen können Sie z. B. den Client mit seinen Sessions (SessionNames) zu Diagnosezwecken identifizieren.

Wenn Sie den Server aktiviert haben, können Sie auch einen anderen Namen verwenden, der in Ihrem Projekt aussagekräftig ist und der die in ihrem Projekt geltenden Anforderungen z. B. nach weltweiter Eindeutigkeit erfüllt.
Das folgende Beispiel stammt von UaExpert:



Applikationsname ändern

Um den Applikationsnamen zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie die CPU aus. Klicken Sie dazu auf das Symbol der CPU (z. B. in der Netzsicht).
2. Klicken Sie in den Eigenschaften der CPU auf "OPC UA > Allgemein"
3. Tragen Sie einen aussagekräftigen Namen ein.

Beachten Sie, dass der Applikationsname auch im Zertifikat eingetragen ist (Subject Alternative Name) und Sie nach Änderung des Applikationsnamens gegebenenfalls ein zuvor erstelltes Zertifikat nochmals erstellen müssen.

11.3.3.2 Zugang zum OPC UA-Server

Server-Adressen

Der OPC UA-Server der S7-1500 CPU ist über alle integrierten PROFINET-Schnittstellen der CPU (ab Firmware V2.0) erreichbar.

Über CPs ist unter folgenden Bedingungen ein direkter Zugriff über den Rückwandbus des Automatisierungssystems auf den OPC UA-Server der CPU möglich:

- Projektierung mit TIA Portal Version ab V16 S7-1500-CPU ab Firmware Version 2.8 sowie CP 1543-1 ab Firmware Version V2.2.

  Zur Projektierung siehe Zugang zu OPC UA-Applikationen (Seite 168).

Über CMs ist kein direkter Zugriff über den Rückwandbus des Automatisierungssystems auf den OPC UA-Server der CPU möglich.

Bei SIMATIC S7 1500 SW Controllern ist der OPC UA-Server über die PROFINET-Schnittstellen erreichbar, die der Software-PLC zugewiesen sind.

Weitere Zugangsmöglichkeiten von SW Controllern sind in folgendem Anwendungsbeispiel beschrieben: Interne und externe OPC-UA Anbindung über die virtuelle Ethernet-Schnittstelle des Software Controllers ab V2.5
(https://support.industry.siemens.com/cs/ww/de/view/109760541).

Beispiel für URLs (Uniform Resource Locator), über die Verbindungen zum OPC UA-Server der CPU aufgebaut werden können:

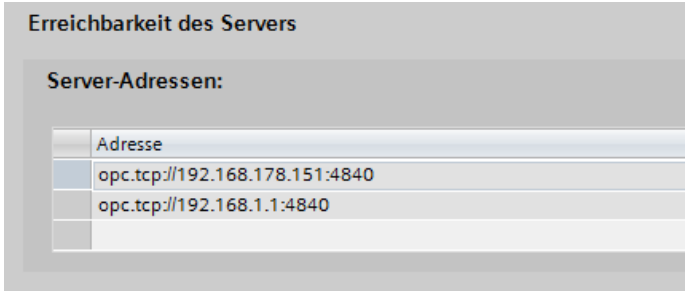


Bild 11-21 Anzeige der Server-Adressen

Die URLs gliedern sich folgendermaßen:

- Protokollkennung "opc.tcp://"
- IP-Adresse
  - 192.168.178.151

    Die IP-Adresse, über die OPC UA-Server aus dem Ethernet-Subnetz 192.168.178 erreichbar ist.
  - 192.168.1.1

    Die IP-Adresse, über die OPC UA-Server aus dem Ethernet-Subnetz 192.168.1 erreichbar ist.
- TCP-Portnummer
  - Voreinstellung: 4840 (Standardport)

    Die Portnummer kann geändert werden, unter "OPC UA > Server > Einstellungen > Port".

Dynamische IP-Adressen

Im folgenden Beispiel ist die IP-Adresse der PROFINET-Schnittstelle [X2] noch nicht festgelegt.

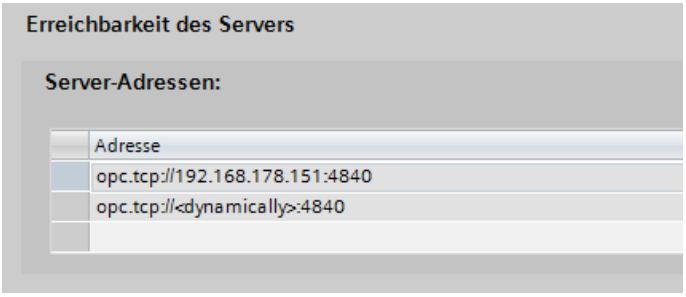


Bild 11-22 Anzeige der Server-Adressen mit dynamischer IP-Adresse

In der Tabelle erscheint der Platzhalter "<dynamically>".
Die IP-Adresse dieser PROFINET-Schnittstelle wird später am Gerät gesetzt, z. B. über das Display der CPU.

Standard-SIMATIC-Server-Schnittstelle aktivieren

Wenn die Option "Standard-SIMATIC-Server-Schnittstelle aktivieren" aktiviert ist, dann stellt der OPC UA-Server der CPU die freigegebenen PLC-Variablen und Server-Methoden den Clients zur Verfügung, wie es von SIEMENS im selbst definierten Namespace festgelegt wurde.
In der Voreinstellung ist diese Option aktiviert.
Lassen Sie die Option aktiviert, damit OPC UA-Clients die Möglichkeit haben, sich automatisch mit dem OPC UA-Server der CPU zu verbinden und Daten auszutauschen.
Wenn Sie diese Option nicht aktivieren, müssen Sie die Server-Schnittstelle über den Eintrag "OPC UA-Kommunikation" in der Projektnavigation hinzufügen. Diese Schnittstelle wird dann als OPC UA-Server-Schnittstelle verwendet, siehe OPC UA-Server-Schnittstelle projektieren (Seite 255).

HINWEIS

Allgemeine Geräteinformationen auch bei deaktivierter Standard-SIMATIC-Server-Schnittstelle lesbar

Auch wenn Sie die Standard-SIMATIC-Server-Schnittstelle deaktivieren, sind für OPC UA-Clients allgemeine Geräteinformationen über den OPC UA-Server der CPU lesbar.

Beispiele für solche Geräteinformationen: DeviceManual, DeviceRevision, OrderNumber. Sämtliche Objekte des Anwenderprogramms bleiben aber in diesem Fall für Clients unsichtbar.

Wenn Sie verhindern wollen, dass auch diese Geräteinformationen nicht sichtbar sind, dann müssen Sie den OPC UA-Server der CPU deaktivieren.

11.3.3.3 Allgemeine Einstellungen des OPC UA-Servers

TCP-Port für OPC UA

Standardmäßig verwendet OPC UA den TCP-Port 4840. Sie können jedoch auch einen anderen Port wählen. Eingaben von 1024 bis 49151 sind möglich. Sie müssen jedoch darauf achten, dass keine Konflikte mit anderen Anwendungen entstehen. Den gewählten Port müssen OPC UA-Clients beim Verbindungsaufbau verwenden.

Im folgenden Beispiel wurde der Port 48400 gewählt:

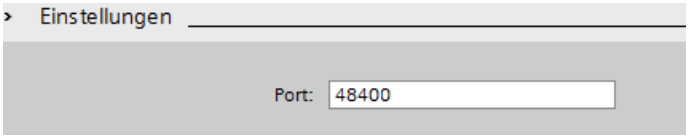


Bild 11-23 TCP-Port für OPC UA

Einen Überblick über die unterstützten Protokolle und die verwendeten Portnummern der S7-1500 CPUs finden Sie im Kapitel Kommunikationsprotokolle und verwendete Portnummern bei Ethernet-Kommunikation (Seite 36).

Einstellungen für Sessions

- Maximales Timeout für Sessions
  In diesem Feld legen Sie fest, wie lange die Zeitspanne höchstens sein darf, bis der OPC UA-Server eine Session ohne Datenaustausch abbaut.
  Mögliche Werte zwischen 1 und 600000 Sekunden.
- Maximale Anzahl OPC UA-Sessions
  In diesem Feld legen Sie fest, wie viele Sessions der OPC UA-Server der CPU höchstens aufbaut und gleichzeitig betreibt.
  Die maximale Anzahl der Sessions ist abhängig von der Leistungsfähigkeit der CPU. Jede Session bindet Ressourcen.

Maximale Anzahl registrierter Knoten

In diesem Feld legen Sie fest, wie viele Knoten (Nodes) der OPC UA-Server höchstens registriert.

Die maximale Anzahl der registrierten Knoten ist abhängig von der Leistungsfähigkeit der CPU und wird beim Projektieren des Feldinhalts angezeigt (Mauszeiger in das Feld setzen). Jede Registrierung bindet Ressourcen.

HINWEIS

Keine Fehlermeldung beim Versuch, mehr Knoten zu registrieren als die projektierte maximale Anzahl registrierbarer Knoten

Wenn ein Client zur Laufzeit mehr Knoten registrieren will als die projektierte maximale Anzahl registrierbarer Knoten, dann registriert der Server der S7-1500 CPU nur die projektierte maximale Anzahl. Der Server liefert dem Client ab der projektierten maximalen Anzahl registrierbarer Knoten die regulären String-Node-Ids unverändert zurück, sodass der Geschwindigkeitsvorteil durch Registrierung für diese Knoten entfällt. Der Client erhält keine Fehlermeldung.

Achten Sie beim Projektieren auf eine ausreichende Reserve; berücksichtigen Sie dazu die maximale Anzahl registrierbarer Knoten (z. B. über die technischen Daten der CPU).

Weitere Informationen

Welche Ports die verschiedenen Dienste für die Datenübertragung über TCP und UDP verwenden und was bei der Verwendung von Routern und Firewalls zu beachten ist, finden Sie im FAQ (https://support.industry.siemens.com/cs/ww/de/view/8970169).

Abwärtskompatible Datentyp-Definitionen nach OPC UA-Spezifikation ≤ V1.03

Die OPC UA-Spezifikation (<= V1.03) definiert Mechanismen, um mit Hilfe von TypeDictionaries Datentyp-Definitionen, z. B. für benutzerdefinierte Strukturen (UDTs) von einem Server auslesen zu können.

Sie können in den OPC UA-Server-Eigenschaften der CPU einstellen, ob die CPU diese abwärtskompatiblen Datentyp-Definitionen nach OPC UA-Spezifikation ≤ V1.03 für die Standard-SIMATIC-Server-Schnittstelle erzeugt oder nicht.

Da TypeDictionaries komplex sind und zu großen OPC UA-XML-Dateien (Server-Schnittstellen) führen, die der Client interpretieren muss, wurde mit OPC UA Spezifikation V1.04 eine einfachere Lösung eingeführt (Attribut "DataTypeDefinition" am DataType-Knoten). Wenn Ihr Client die OPC UA-Spezifikation ab V1.04 unterstützt, dann deaktivieren Sie die Option.

Vorteile der Datentyp-Definitionen nach OPC UA-Spezifikation ab V1.04:

- Der Server startet schneller
- Der Speicher wird effizienter genutzt
- Die Funktion "Browsen" reagiert schneller

11.3.3.4 Einstellungen des Servers für Subscriptions

Subscription statt zyklisches Abfragen

Eine Alternative zum zyklischen Abfragen einer PLC-Variablen (Polling) ist das Beobachten dieses Werts. Nutzen Sie dazu eine Subscription (Abonnement): Wenn sich der Wert von PLC-Variablen geändert hat, informiert der Server den Client. Siehe "Der OPC UA-Client".

Ein Server überwacht meist sehr viele PLC-Werte. Deshalb sendet der Server in regelmäßigen Abständen Nachrichten (Notifikations) an den Client, in denen die neuen Werte der PLC-Variablen enthalten sind.

Vorteile von Subscriptions:

- Der Server startet schneller
- Der Speicher wird effizient genutzt

Wie oft sendet der Server Nachrichten?

Beim Anlegen einer Subscription gibt der OPC UA-Client seinen Wunsch an, in welchen Abständen der OPC UA-Client bei Wertänderung die neuen Werte erhalten möchte. Um die Kommunikationslast durch OPC UA zu begrenzen, legen Sie einen zeitlichen Mindestabstand für die Nachrichten fest. Dazu verwenden Sie die Parameter für das kleinste Sendeintervall und das kleinste Abtastintervall.

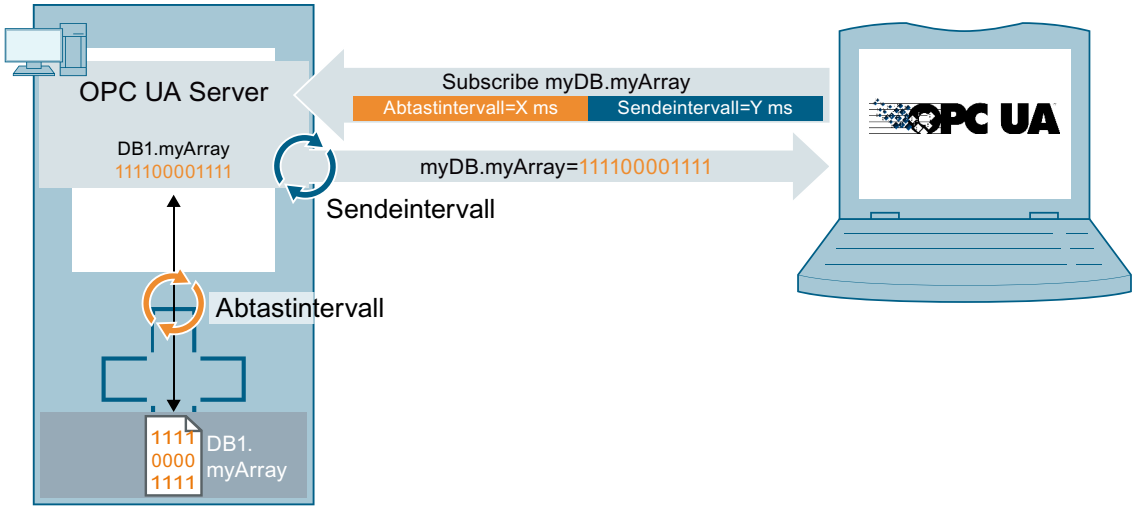


Bild 11-24 Prinzip einer Subscription

Kleinstes Sendeintervall

Bei "Kleinstes Sendeintervall" stellen Sie die zeitlichen Abstände (Intervalle) ein, in denen der Server bei Wertänderung eine Nachricht mit den neuen Werten an den Client schickt.
Im folgenden Bild wird als "Kleinstes Abtastintervall" der Wert 250 ms verwendet. Als "Kleinstes Sendeintervall" ist der Wert von 200 ms eingetragen.

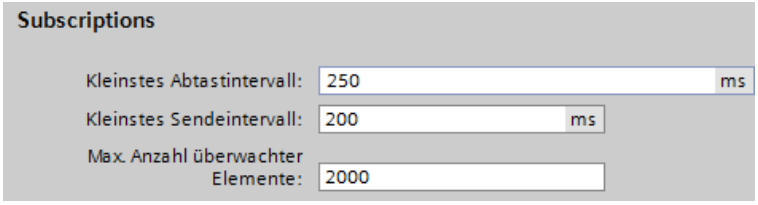


Bild 11-25 Einstellungen Subscriptions

Im Beispiel sendet der OPC UA-Server bei Wertänderung alle 200 ms eine neue Nachricht, falls der OPC UA-Client eine Aktualisierung fordert.
Wenn der OPC UA-Client eine Aktualisierung alle 1000 ms fordert, dann sendet der OPC UA-Server auch nur einmal in 1000 ms (eine Sekunde) eine Nachricht mit den neuen Werten.
Wenn der OPC UA-Client eine Aktualisierung alle 100 ms fordert, dann sendet der Server trotzdem nur alle 200 ms (kleinstes Sendeintervall).

Kleinstes Abtastintervall

Bei "Kleinstes Abtastintervall" stellen Sie die zeitlichen Abstände (Intervalle) ein, in denen der OPC UA-Server den Wert einer CPU-Variablen erfasst und mit dem bisherigen Wert vergleicht, um eine Wertänderung festzustellen.
Wenn Sie das Abtastintervall kleiner gewählt haben als das Sendeintervall und ein OPC UA-Client für bestimmte PLC-Variablen eine hohe Abtastrate fordert, dann können pro Sendeintervall zwei oder mehr Werte anfallen.
In diesem Fall schreibt der OPC UA-Server die Wertänderungen in die Warteschlange und sendet nach Ablauf des Sendeintervalls alle Wertänderungen an den Client. Wenn mehr Wertänderungen im Sendeintervall anfallen als in die Warteschlange passen, dann überschreibt der OPC UA-Server die ältesten Werte (abhängig von der eingestellten "Discard Policy" des Daten-abonnierenden Clients, die Option "Discard Oldest" muss in diesem Fall aktiviert sein). Die aktuellsten Werte werden an den Client gesendet.

Maximale Anzahl überwachter Elemente (Monitored Items)

In diesem Feld legen Sie die maximale Anzahl an Elemente fest, die der OPC UA-Server der CPU gleichzeitig auf Wertänderung überwacht.
Die Überwachung bindet Ressourcen. Die maximale Anzahl überwachter Elemente ist abhängig von der verwendeten CPU.

Weitere Informationen

Informationen zu den Systemgrenzen des OPC UA-Servers der S7-1500 CPUs (Firmware V2.0 und V2.1) hinsichtlich Subscriptions, Abtastintervalle und Sendeintervalle entnehmen Sie folgendem FAQ (https://support.industry.siemens.com/cs/ww/de/view/109755846).
Bei der Verwendung von Subscriptions geben im Fehlerfall bestimmte Statuscodes Auskunft über den aufgetretenen Fehler. Informationen zu Ursachen und Abhilfen bei auftretenden Statuscodes im OPC UA Client finden Sie in der Liste der Fehlercodes in der Online-Hilfe zu STEP 7 (TIA Portal) oder im folgenden FAQ (https://support.industry.siemens.com/cs/ww/de/view/109755860).
Die Regeln für Subscriptions finden Sie im Kapitel Regeln für Subscriptions (Seite 362).
Informationen zur Diagnose von Subscriptions finden Sie im Kapitel Subscriptions diagnostizieren (Seite 324).

11.3.3.5 Handling der Client- und Server-Zertifikate

Eine gesicherte Verbindung zwischen OPC UA-Server und einem OPC UA-Client kommt nur dann zu Stande, wenn sich der Server gegenüber dem Client ausweisen kann. Dazu dient das Zertifikat des Servers.

Zertifikat des OPC UA-Servers

Wenn Sie den OPC UA-Server aktiviert und die Sicherheitshinweise bestätigt haben, erzeugt STEP 7 automatisch das Zertifikat für den Server und speichert es im lokalen Zertifikate-Verzeichnis der CPU. Dieses Verzeichnis können Sie mit dem lokalen Zertifikatsmanager der CPU einsehen und verwalten (Zertifikate exportieren oder löschen).
Das folgende Bild zeigt den lokalen Zertifikatsmanager der CPU mit dem automatisch erzeugten Zertifikat für den OPC UA-Server:

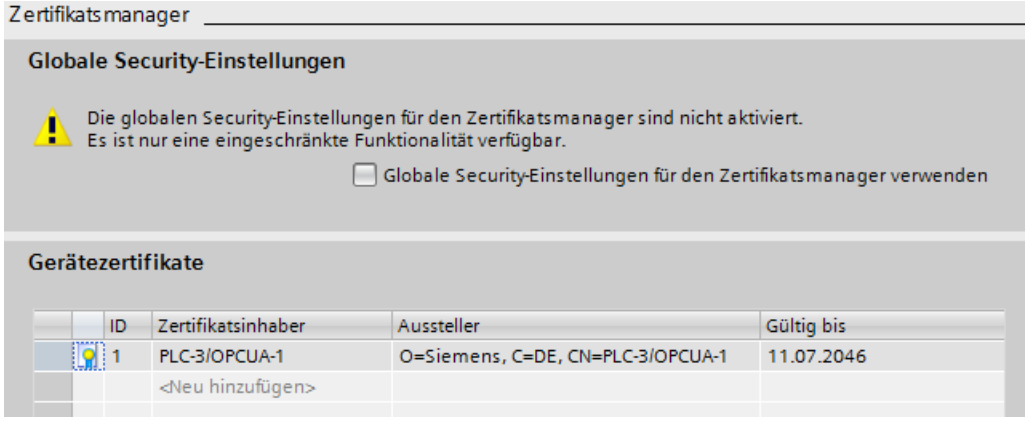


Bild 11-26 Lokaler Zertifikatsmanager der CPU

Alternativ können Sie auch selbst ein Server-Zertifikat erzeugen.
Das Zertifikat des Servers wird während des Aufbaus einer Verbindung vom Server zum Client übertragen, der Client überprüft das Zertifikat.

Der Client-Anwender entscheidet, ob er dem Zertifikat des Servers vertraut

Auf der Client-Seite muss nun der Anwender entscheiden, ob er dem Server-Zertifikat vertraut. Wenn der Anwender dem Server-Zertifikat vertraut, dann speichert der Client das Server-Zertifikat in seinem Verzeichnis, das die vertrauenswürdigen Server-Zertifikate enthält.
Das folgende Beispiel zeigt einen Dialog des Clients "UA Sample Client". Wenn der Anwender auf die Schaltfläche "Ja" klickt, vertraut der Client dem Server-Zertifikat:

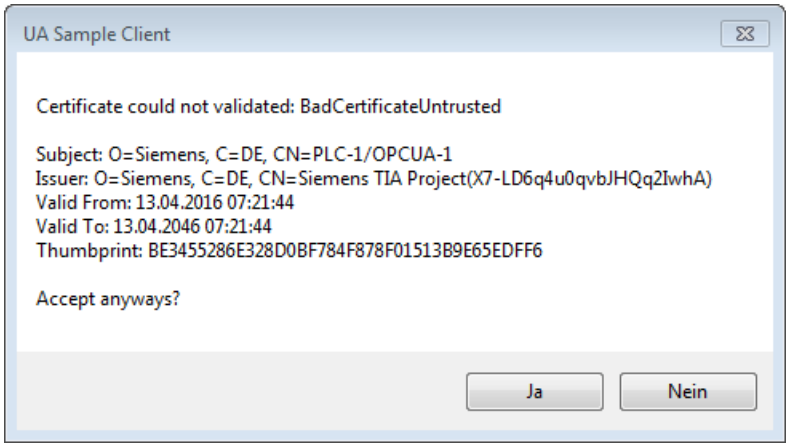


Bild 11-27 Dialog des Clients "UA Sample Client"

Woher kommt das Zertifikat eines Clients?

Client der S7-1500

Wenn Sie den OPC UA-Client einer S7-1500 CPU nutzen (OPC UA-Client aktiviert), dann können Sie mit STEP 7 ab V15 für diese Clients Zertifikate erzeugen:

1. In der "Projektnavigation" wählen Sie die CPU aus, die als Client fungiert.
2. Doppelklicken Sie auf "Gerätekonfiguration"
3. In den Eigenschaften der CPU klicken Sie auf "Schutz & Security > Zertifikatsmanager".
4. In der Tabelle "Gerätezertifikate" doppelklicken Sie auf "<neu hinzufügen>".
   STEP 7 öffnet einen Dialog.
5. Klicken Sie auf die Schaltfläche "Hinzufügen".
6. Bei "Verwendungszweck" wählen Sie aus der Liste den Eintrag "OPC UA-Client".
   Achtung:
   Unter "Alternativer Name des Zertifikatsinhabers (SAN)" müssen die IP-Adressen eingetragen sein, unter denen die CPU in Ihrer Anlage erreichbar ist.
   Sie müssen also die IP-Schnittstellen der CPU konfigurieren, bevor Sie ein Client-Zertifikat erzeugen.
7. Klicken Sie auf "OK".
   STEP 7 zeigt nun das Client-Zertifikat in der Tabelle "Gerätezertifikate" an.
8. Klicken Sie mit der rechten Maustaste auf diese Zeile und wählen Sie aus dem Kontextmenü den Eintrag "Zertifikat exportieren".
9. Wählen Sie ein Verzeichnis aus, in dem Sie das Client-Zertifikat speichern.

Clients anderer Hersteller
Wenn Sie UA-Clients von Herstellern oder der OPC Foundation verwenden, wird bei der Installation oder beim ersten Programmaufruf automatisch ein Client-Zertifikat erzeugt.
DieseZertifikate müssen Sie über den globalen Zertifikatsmanager in STEP 7 importieren und für die jeweilige CPU verwenden (wie oben gezeigt).
Wenn Sie selbst einen OPC UA-Client programmieren, dann können Sie Zertifikate programmtechnisch erstellen, siehe "Instanz-Zertifikat für den Client". Oder Sie erzeugen Zertifikate mit Tools, zum Beispiel mit OpenSSL oder dem Zertifikate-Generator der OPC Foundation:

- Wie Sie bei OpenSSL vorgehen, lesen Sie hier: "PKI-Schlüsselpaare und Zertifikate selbst erzeugen".
- Wie Sie mit dem Zertifikate-Generator der OPC Foundation arbeiten, lesen Sie hier: "Selbstsignierte Zertifikate erzeugen".

Client-Zertifikate dem Server bekanntmachen

Client-Zertifikate müssen Sie dem Server zur Verfügung stellen, damit eine gesicherte Verbindung aufgebaut werden kann.
Dazu gehen Sie folgendermaßen vor:

1. Aktivieren Sie im lokalen Zertifikatsmanager des Servers die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden". Dadurch ist der globale Zertifikatsmanager verfügbar.
   Sie finden diese Option in den Eigenschaften der CPU, die als Server dient, unter "Schutz & Security > Zertifikatsmanager".
   Falls dieses Projekt noch nicht geschützt ist, dann klicken Sie in der "Projektnavigation" von STEP 7 unter "Security-Einstellungen > Einstellungen" auf die Schaltfläche "Dieses Projekt schützen" und melden Sie sich an.
   STEP 7 zeigt nun in der "Projektnavigation" unter "Security-Einstellungen" der Eintrag "Globale Security-Einstellungen" an.
2. Doppelklicken Sie auf "Globale Security-Einstellungen".
3. Doppelklicken Sie auf "Zertifikatsmanager".
   STEP 7 öffnet den globalen Zertifikatsmanager.
4. Klicken Sie auf das Register "Vertrauenswürdige Zertifikate".
5. Klicken Sie mit der rechten Maustaste im Register auf eine freie Fläche (nicht auf ein Zertifikat).
6. Wählen Sie aus dem Kontextmenu den Eintrag "Importieren".
   Der Dialog zum Importieren von Zertifikaten wird angezeigt.
7. Wählen Sie das Client-Zertifikat aus, dem der Server vertrauen soll.
8. Klicken Sie auf die Schaltfläche "Öffnen", um das Zertifikat zu importieren.
   Das Zertifikat des Clients ist nun im globalen Zertifikatsmanager enthalten.
   Merken Sie sich die ID des gerade importierten Client-Zertifikats.
9. Klicken Sie nun in den Eigenschaften der CPU, die als Server dient, auf das Register "Allgemein".
10. Klicken Sie auf den Bereich "OPC UA > Server > Security > Secure Channel".
11. Scrollen Sie im Dialog "Secure Channel" nach unten zum Abschnitt "Vertrauenswürdige Clients".
12. Doppelklicken Sie in der Tabelle auf die leere Zeile mit "<neu hinzufügen>". In der Zeile wird eine Schaltfläche mit drei Punkten angezeigt.
13. Klicken Sie auf diese Schaltfläche.

14. Wählen Sie das Client-Zertifikat aus, das Sie importiert haben.
15. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
16. Übersetzen Sie das Projekt.
17. Laden Sie die Konfiguration in die S7-1500 CPU.

Ergebnis:
Der Server vertraut nun dem Client. Wenn außerdem das Server-Zertifikat als vertrauenswürdig gilt, dann können Server und Client eine gesicherte Verbindung aufbauen.

Client-Zertifikate automatisch akzeptieren

Wenn Sie die Option "Client-Zertifikate zur Laufzeit automatisch akzeptieren" aktivieren (unterhalb der Liste "Vertrauenswürdige Clients"), dann akzeptiert der Server alle Client-Zertifikate.

ACHTUNG
Einstellung nach der Inbetriebnahme Um Sicherheitsrisiken zu vermeiden, deaktivieren Sie die Option "Client-Zertifikate zur Laufzeit automatisch akzeptieren" wieder nach der Inbetriebnahme.

Security-Einstellungen des Servers konfigurieren

Das folgende Bild zeigt die verfügbaren Security-Einstellungen des Servers zum Signieren und Verschlüsseln von Nachrichten.

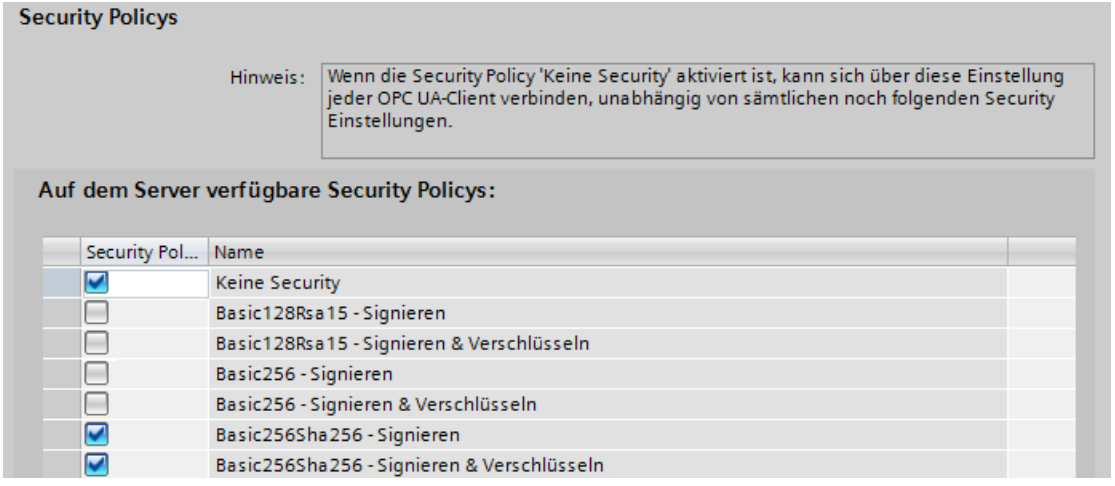


Bild 11-28 Security-Einstellungen des Servers konfigurieren

Per Voreinstellung wird ein Server-Zertifikat erstellt, welches SHA256-Signierung nutzt. Die folgenden Security Policys sind freigegeben:

- Keine
  Ungesicherter Endpoint

  HINWEIS

  Nicht erwünschte Security Policys deaktivieren

  Wenn Sie bei den Secure-Channel-Einstellungen des S7-1500 OPC UA-Servers alle Security Policys aktiviert haben (Voreinstellung) - also auch den Endpunkt "Keine Security" - dann ist der Datenverkehr zwischen Server und Client auch ungesichert möglich (weder signiert noch verschlüsselt). Die Identität des Clients bleibt bei "Keine Security" unbekannt. Jeder OPC UA-Client kann sich dann mit dem Server verbinden, unabhängig von sämtlichen noch folgenden Security-Einstellungen.

  Achten Sie bei der Projektierung des OPC UA-Servers darauf, dass nur Security Policys aktiviert sind, die mit dem Schutzkonzept für Ihre Maschine oder Anlage vereinbar sind. Alle anderen Security Policys sind zu deaktivieren.

  Empfehlung: Verwenden Sie, wenn möglich, die Einstellung "Basic256Sha256".

- Basic128Rsa15 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 128-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic128Rsa15 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 128-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
- Basic256Rsa15 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 256-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic256Rsa15 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen, die den Hash-Algorithmus RSA15 und 256-Bit-Verschlüsselung verwenden.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
- Basic256Sha256 - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
- Basic256Sha256 - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Hashing und 256-Bit-Verschlüsselung.
  Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
- Aes256_Sha256_RsaPss - Signieren
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Verschlüsselung und 256-Bit-Hashing. Alle Zertifikate müssen wenigstens Sha256 Signaturen nutzen.
  Dieser Endpoint sichert die Integrität der Daten durch Signieren.
  Für hohe Security-Anforderungen. PKI-Infrastruktur erforderlich.

- Aes256_Sha256_RsaPss - Signieren & Verschlüsseln
  Gesicherter Endpoint, unterstützt eine Reihe von Algorithmen für 256-Bit-Verschlüsselung und 256-Bit-Hashing. Alle Zertifikate müssen wenigstens Sha256 Signaturen nutzen. Dieser Endpoint sichert die Integrität und Vertraulichkeit der Daten durch Signieren und Verschlüsseln.
  Für hohe Security-Anforderungen. PKI-Infrastruktur erforderlich.

Um eine Security-Einstellung frei zu geben, klicken Sie auf das Kästchen in der jeweiligen Zeile.

HINWEIS

Wenn Sie die Einstellungen "Basic256Sha256 -Signieren" und "Basic256Sha256 -Signieren & Verschlüsseln" nutzen, dann müssen OPC UA-Server und OPC UA-Clients "SHA256"-signierte Zertifikate verwenden.

Bei den Einstellungen "Basic256Sha256 -Signieren" und "Basic256Sha256 -Signieren & Verschlüsseln" signiert die Zertifizierungsstelle von STEP 7 die Zertifikate automatisch mit "SHA256".

Security Policy "Keine Security" und Authentifizierung über Benutzername und Passwort

Folgende Kombination können Sie einstellen:

Security Policy = "Keine Security" und Authentifizierung über Benutzername und Passwort.

- Der OPC UA-Server der S7-1500 unterstützt diese Kombination. OPC UA-Clients können sich verbinden und die Authentifizierungsdaten verschlüsseln oder auch nicht.
- Der OPC UA-Client der S7-1500 CPU unterstützt ebenfalls diese Kombination: Zur Laufzeit verbindet er sich aber nur dann, wenn er die Authentifizierungsdaten verschlüsselt über die Leitung schicken kann!

11.3.3.6 Server-Zertifikate mit STEP 7 erzeugen

Die folgende Beschreibung zeigt die Vorgehensweise zur Erzeugung neuer Zertifikate mit STEP 7 und gilt prinzipiell für verschiedene Verwendungen der Zertifikate. Je nachdem, über welchen Bereich der CPU-Eigenschaften Sie den folgenden Dialog starten, stellt STEP 7 den passenden Verwendungszweck bereits ein - hier "OPC UA-Client & -Server".

Empfehlung: Um die volle Funktionalität für die Security des OPC UA-Servers zu nutzen, verwenden Sie die globalen Security-Einstellungen.

Die globalen Security-Einstellungen aktivieren Sie in den CPU-Eigenschaften, Bereich "Schutz & Security > Zertifikatsmanager".

Server-Zertifikate individuell anpassen

STEP 7 erzeugt automatisch ein Zertifikat für den OPC UA-Server der S7-1500, wenn Sie den Server aktivieren (siehe "OPC UA-Server aktivieren (Seite 233)"). Dabei verwendet STEP 7 für die Parameter des Zertifikats voreingestellte Werte. Wenn Sie die Parameter ändern wollen, dann gehen Sie folgendermaßen vor:

1. Klicken Sie in den Eigenschaften der CPU unter "Allgemein > OPC UA > Server > Security > Secure Channel > Server-Zertifikat" auf die Drei-Punkte-Schaltfläche. Ein Dialog wird eingeblendet, der lokal vorhandene Zertifikate anzeigt.
2. Klicken Sie auf die Schaltfläche "Hinzufügen".
3. Der Dialog zum Erzeugen neuer Zertifikate wird angezeigt (folgendes Bild). Die Werte für ein Beispiel sind bereits eingetragen:

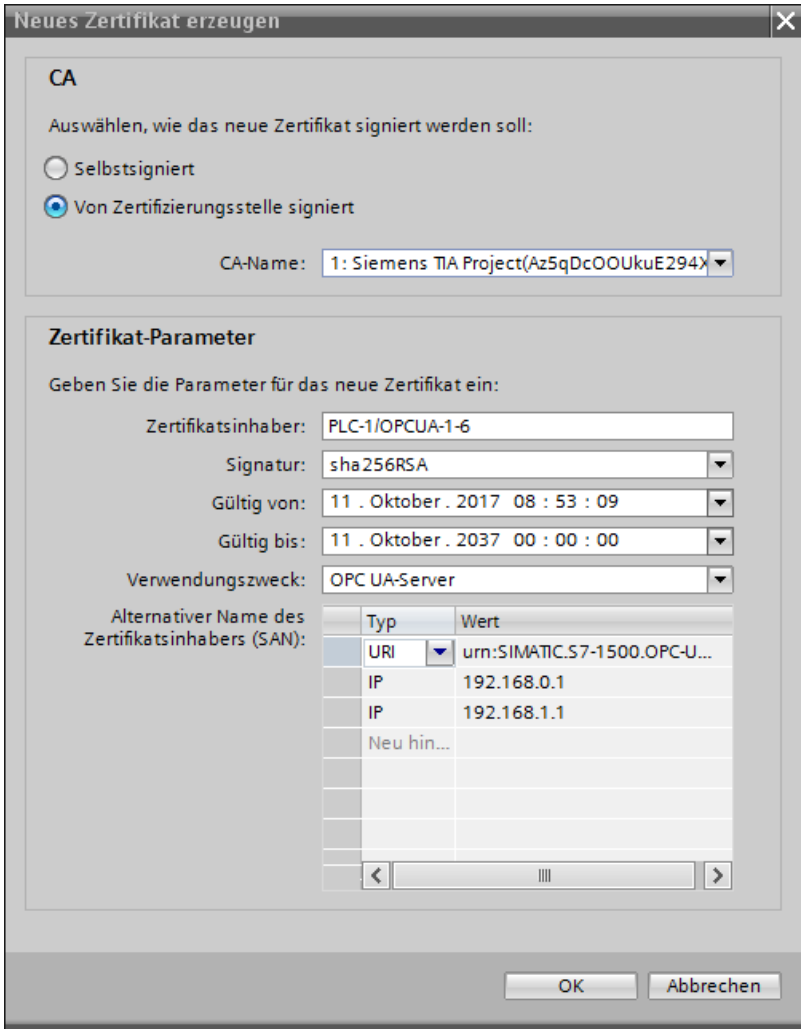


Bild 11-29 Server-Zertifikate individuell anpassen

4. Verwenden Sie andere Parameter, falls dies nach den Sicherheitsvorgaben in Ihrem Unternehmen oder des Auftraggebers erforderlich ist.

Erläuterung der Felder für die Zertifikatserzeugung

- CA
  Wählen Sie aus, ob das Zertifikat selbst-signiert sein soll oder von einem der CA-Zertifikate des TIA Portals. Die Zertifikate sind unter "Zertifikate bei OPC UA" beschrieben. Wenn Sie ein Zertifikat erzeugen wollen, das von einem der CA-Zertifikate des TIA-Portals signiert sein soll, dann muss das Projekt geschützt sein und Sie müssen als Benutzer mit den erforderlichen Funktionsrechten angemeldet sein. Informationen hierzu erhalten Sie unter "Grundlagen der Benutzerverwaltung im TIA Portal".
- Zertifikatsinhaber
  Die Voreinstellung setzt sich zusammen aus dem Namen des Projekts und "\OPCUA-1". Im Beispiel lautet der Projekt-Name "PLC1". In den Eigenschaften der CPU stellen Sie den Projektnamen ein unter "Allgemein > Projektinformation > Name". Behalten Sie die Voreinstellung bei oder tragen Sie bei "Zertifikatsinhaber" einen anderen Namen für den OPC UA-Server ein, der in Ihrem Projekt aussagekräftiger ist.
- Signatur
  Wählen Sie hier das Hash- und Verschlüsselungsverfahren aus, das beim Signieren des Server-Zertifikats verwendet werden soll. Die folgenden Einträge sind verfügbar:
  - "sha1RSA",
  - "sha256RSA".
- Gültig von
  Tragen Sie hier das Datum und die Uhrzeit ein für den Beginn der Gültigkeit des Server-Zertifikats.
- Gültig bis
  Tragen Sie hier das Datum und die Uhrzeit ein für das Ende der Gültigkeit des Server-Zertifikats. Achten Sie darauf, dass das Zertifikat nicht nur ein Jahr oder wenige Jahre gültig ist. Im Beispiel ist das Zertifikat 30 Jahre gültig. Aus Sicherheitsgründen sollten Sie allerdings das Zertifikat in viel kürzeren Abständen erneuern. Die lange Gültigkeit überlässt aber Ihnen die Entscheidung, wann dazu der passende Zeitpunkt ist, etwa wenn eine Anlage gewartet werden muss.
- Verwendungszweck
  Voreingestellt ist "OPC UA-Client & -Server". Behalten Sie diese Voreinstellung für den OPC UA-Server bei. Der Dialog "Neues Zertifikat erzeugen" kann von mehreren Stellen aus in STEP 7 aufgerufen werden. Wenn Sie zum Beispiel diesen Dialog für den Webserver der CPU aufrufen, dann wird unter "Verwendungszweck" "Webserver" eingetragen. In der Klappliste zum Verwendungszweck sind die folgenden Einträge verfügbar:
  - "OPC UA-Client"
  - "OPC UA-Client & -Server"
  - "OPC UA-Server"
  - "TLS"
  - "Webserver"
- Alternativer Name des Zertifikatsinhabers (SAN)
  Im Beispiel oben ist eingetragen:
  "URI:urn:SIMATIC.S7-1500.OPC-UAServer:PLC1,IP:192.168.178.151,IP:192.168.1.1". Diese URI muss korrekt eingetragen sein, da sie gegen die übermittelte Applikationsbeschreibung geprüft wird.
  Gültig wäre auch der folgende Eintrag: "IP: 192.168.178.151, IP: 192.168.1.1". Wichtig ist, dass hier die IP-Adressen eingetragen sind, über die der OPC UA-Server der CPU erreichbar ist.
  Siehe "Zugang zum OPC UA-Server (Seite 235)".

Dadurch können OPC UA-Clients überprüfen, ob tatsächlich eine Verbindung zum OPC UA-Server der S7-1500 aufgebaut werden soll, oder möglicherweise ein Angreifer versucht, dem OPC UA-Client von einem anderen PC aus manipulierte Werte zuzusenden.

11.3.3.7 Authentifizierung des Benutzers

Arten der Benutzer-Authentifizierung

Sie können beim OPC UA-Server der S7-1500 einstellen, wie sich ein Benutzer des OPC UA-Clients legitimieren muss, wenn er auf den Server zugreifen will.
Dazu gibt es die folgenden Möglichkeiten:

- **Gast-Authentifizierung**
  Der Anwender muss seine Berechtigung nicht nachweisen (anonymer Zugang). Der OPC UA-Server überprüft nicht die Berechtigung des Client-Anwenders.
  Für CPUs bis Firmware-Version V3.0: Wenn Sie diese Art der Benutzer-Authentifizierung nutzen wollen, dann wählen Sie unter "OPC UA > Server > Security > Benutzer-Authentifizierung" die Option "Gast-Authentifizierung aktivieren".
  Für CPUs ab Firmware-Version ab V3.1: Nutzen Sie die lokale Benutzerverwaltung, um die Gast-Authentifizierung über den Benutzer "Anonymous" zu realisieren.

  HINWEIS

  Um die Security zu erhöhen, sollten Sie den Zugriff auf den OPC UA-Server nur mit Benutzer-Authentifizierung erlauben!

- **Authentifizierung über Benutzername und Passwort**
  Der Anwender muss seine Berechtigung nachweisen (kein anonymer Zugang). Der OPC UA-Server überprüft, ob der Client-Anwender berechtigt ist, auf den Server zuzugreifen. Als Nachweis gilt der Benutzername mit dem richtigen Passwort.
  Für CPUs bis bis Firmware-Version V3.0: Wenn Sie diese Art der Benutzer-Authentifizierung nutzen wollen, dann gehen Sie folgendermaßen vor:
  - Wählen Sie unter "OPC UA > Server > Security > Benutzer-Authentifizierung" die Option "Authentifizierung über Benutzername und Passwort aktivieren".
  - Deaktivieren Sie die Gast-Authentifizierung.
  - Tragen Sie in der Tabelle "Benutzerverwaltung" die Anwender ein.
    Klicken Sie dazu jeweils auf den Eintrag "<Neuen Benutzer hinzufügen>". Ein neuer Benutzer mit einem automatisch vergebenen Namen wird angelegt. Sie können den Benutzernamen editieren und das für den Benutzernamen zugehörige Passwort eingeben. Sie können maximal 21 Benutzer hinzufügen.
- **Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts**
  Für CPUs bis Firmware-Version V3.0 gibt es die Option "Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts aktivieren". Diese Einstellung befindet sich unter den allgemeinen OPC UA-Einstellungen (CPU-Eigenschaften: OPC UA > Allgemein). Wenn Sie diese Option aktivieren, dann wird die Benutzerverwaltung des geöffneten Projekts auch für die Benutzer-Authentifizierung des OPC UA-Servers verwendet: Bei OPC UA sind dann dieselben Benutzernamen und Passwörter gültig wie im aktuellen Projekt.

Um die Benutzerverwaltung des Projekts zu aktivieren, gehen Sie folgendermaßen vor:

- Klicken Sie in der "Projektnavigation" auf "Security-Einstellungen > Einstellungen".
- Klicken Sie auf Schaltfläche "Dieses Projekt schützen".
- Tragen Sie Ihren Benutzernamen und Ihr Passwort ein.
- Unter "Security-Einstellungen > Benutzer und Rollen" tragen Sie weitere Benutzer ein.

Wenn Sie in Ihrem Projekt einen weiteren OPC UA-Server projektieren, dann aktivieren Sie in dem Projekt ebenfalls die Option "Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts aktivieren". Dadurch ist eine erneute Eingabe von Benutzernamen und Passwörtern unnötig.

Übertragung der Benutzer-Authentifizierung über den Secure Channel

Identifikationsdaten zur Benutzer-Authentifizierung wie z. B. Benutzername und Passwort werden bei OPC UA mit einer gesonderten Security Policy übertragen. Diese Security Policy wird als "UserTokenPolicy" bezeichnet.

Unabhängig von den eingestellten Security Policys für den Secure Channel wählt ein OPC UA-Client bei geforderter Benutzer-Authentifizierung beim Verbindungsaufbau eine angemessene "UserTokenPolicy" aus. Diese UserTokenPolicy sorgt dafür, dass ein UserIdentityToken (z. B. Benutzername und Passwort) immer mit angemessenen Security-Einstellungen übertragen wird.

Der OPC UA-Client hat damit die Möglichkeit, Benutzername und Passwort auch dann verschlüsselt zu übertragen, wenn aufgrund der eingestellten Security Policy "Keine Security" für den Secure Channel gilt. Zum Ablauf eines gesicherten Verbindungsaufbaus bei OPC UA siehe Nachrichten gesichert übertragen (Seite 191).

11.3.3.8 Benutzer und Rollen mit OPC UA-Funktionsrechten

Folgende Optionen zur Benutzer-Authentifizierung greifen auf zentrale Projekteinstellungen für Projektbenutzer zurück:

- Beim Server:
  Beim Parametrieren der CPU-Eigenschaften (OPC UA > Server > Security > Benutzer-Authentifizierung). Dort ist es die Option "Zusätzliche Benutzerverwaltung über die Security-Einstellungen des Projekts aktivieren"
- Beim Client:
  Beim Konfigurieren der Client-Schnittstelle (Register "Konfiguration", Bereich "Security"). Dort ist es die Option "Benutzer (TIA Portal - Security-Einstellungen)"

Voraussetzung

Um die Security-Einstellungen bearbeiten zu können, muss das Projekt geschützt sein und Sie sind mit ausreichenden Rechten, z. B. als Administrator, angemeldet.

Einstellungen in der Projektnavigation > "Security-Einstellungen"

Die zentralen Benutzereinstellungen und Rollen erreichen Sie im geschützten Projekt in der Projektnavigation, Bereich "Security-Einstellungen". Hier definieren Sie zentral Benutzer mit ihrem Benutzernamen, Passwort sowie Funktionsrechten. Diese Einstellungen können Sie an anderer Stelle einfach wiederverwenden.

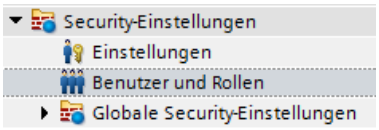


Bild 11-30 Einstellung Benutzen und Rollen

Wiederverwenden zentraler Security-Einstellungen

Beispiele für andere Stellen der Wiederverwendung sind:

- Benutzer-Auswahl für die Benutzer-Authentifizierung beim OPC UA-Server
  Bei dieser Einstellung teilen Sie dem Server mit, welcher Client (Benutzer) mit welchem Benutzernamen und welchem Passwort überhaupt auf den Server zugreifen darf.
- Benutzer-Auswahl für die OPC UA-Client-Authentifizierung
  Bei dieser Einstellung teilen Sie dem Client mit, mit welchem Benutzernamen und welchem Passwort er sich als Client beim Server authentifiziert.

Die Einstellungen für den Client und den Server müssen zusammenpassen: Der Benutzername und das Passwort, mit dem der Client sich anmeldet, muss beim Server eingerichtet und mit den entsprechenden Berechtigungen versehen sein.

Funktionsrechte für Server und Client

Für die Benutzer der Client-Funktionalität und für die Benutzer der Server-Funktionalität bei einer S7-1500 CPU müssen auch die entsprechenden Funktionsrechte für den Client bzw. für den Server aktiviert sein. Es reicht nicht aus, nur Benutzernamen und Passwort zentral zu hinterlegen.

Ein Beispiel erläutert diese Art der Rechteverwertung.

1. Sie definieren im Bereich "Security-Einstellungen > Benutzer und Rollen" eine neue Rolle im Register "Rollen" mit dem Namen z. B. "PLC-opcua-role-all-inclusive".
   Tipp: Möglicherweise wird das Register durch ein Informationsfenster verdeckt ("Der aktuelle Status wurde noch nicht überprüft..."). Schließen Sie in dem Fall zunächst das Informationsfenster.
2. Im Bereich "Kategorien von Funktionsrechten" navigieren Sie zu den Runtime-Rechten, dann zu den CPU-Funktionsrechten und markieren darunter die CPU, deren Funktionsrechte Sie einstellen wollen.

3. Im Bereich "Funktionsrechte" finden Sie folgende Funktionsrechte:
   - **OPC UA-Server-Zugriff**
     Dieses Funktionsrecht wirkt am OPC UA-Server der S7-1500 CPU. Nur wenn diese Option markiert ist, kann der Benutzer, der die Rolle "PLC-opcua-role-all-inclusive" hat, Zertifikate, CRLs oder Vertrauenslisten zur Laufzeit in die CPU übertragen (Push-Funktion). Dieses Funktionsrecht wird für ein automatisiertes Zertifikatshandling benötigt z. B. im Rahmen von GDS (Global Discovery Service).
   - **Zertifikate verwalten**
     Dieses Funktionsrecht wirkt am OPC UA-Server der S7-1500 CPU. Nur wenn diese Option aktiviert ist, kann der Benutzer, der die Rolle "PLC-opcua-role-all-inclusive" hat, Zertifikate, CRLs oder Vertrauenslisten zur Laufzeit in die CPU übertragen (Push-Funktion). Dieses Funktionsrecht wird für ein automatisiertes Zertifikatshandling benötigt z. B. im Rahmen von GDS (Global Discovery Service).
   - **Benutzer-Authentifizierung des OPC UA-Clients**
     Dieses Funktionsrecht wirkt am OPC UA-Client der S7-1500 CPU (an den Client-Anweisungen). Nur wenn diese Option markiert ist, kann der Benutzer, dem die Rolle "PLC-opcua-role-all-inclusive" zugewiesen ist, den entsprechenden Benutzernamen und das Passwort nutzen, um sich für den Aufbau einer Session mit einem Server zu authentifizieren.

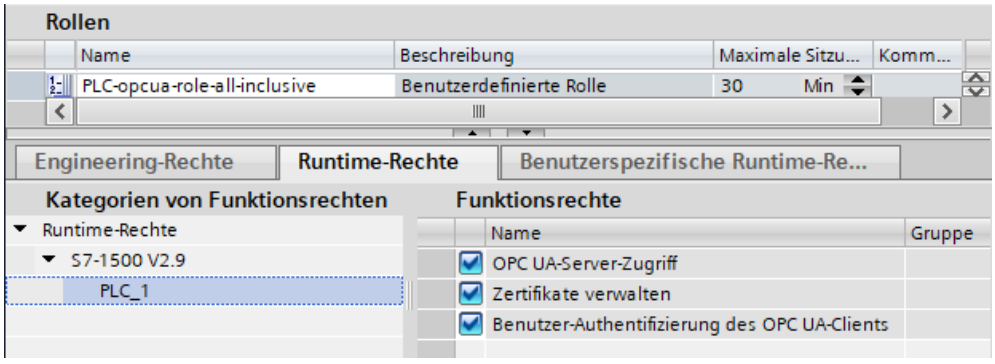


Bild 11-31 Funktionsrechte einstellen

4. Die Rolle "PLC-opcua-role-all-inclusive" müssen Sie noch den entsprechenden Benutzern zuweisen (Register "Benutzer" im Bereich "Security-Einstellungen" in der Projektnavigation).

HINWEIS

"Runtime-Timeout" für Benutzer mit OPC UA-Funktionsrechten

Die Werte in der Spalte "Runtime-Timeout" (Max. Sitzungsdauer) in der Tabelle zur Konfiguration von Benutzern wertet die CPU nicht aus für OPC UA-Runtime-Rechte.

Daher wird ein Benutzer nicht automatisch nach einer bestimmten Zeitspanne abgemeldet. Nutzen Sie für diesen Zweck OPC UA spezifische Mechanismen wie z. B. den Parameter "Max. Session-Timeout" (Bereich OPC UA > Server > Einstellungen).

11.3.3.9 Diagnoseeinstellungen des Servers

Diagnose

Den Umfang der Diagnosen des OPC UA-Servers können Sie in den CPU-Einstellungen festlegen.
Um den Diagnoseumfang zu ändern, navigieren Sie zum Bereich "OPC UA > Server > Diagnose".

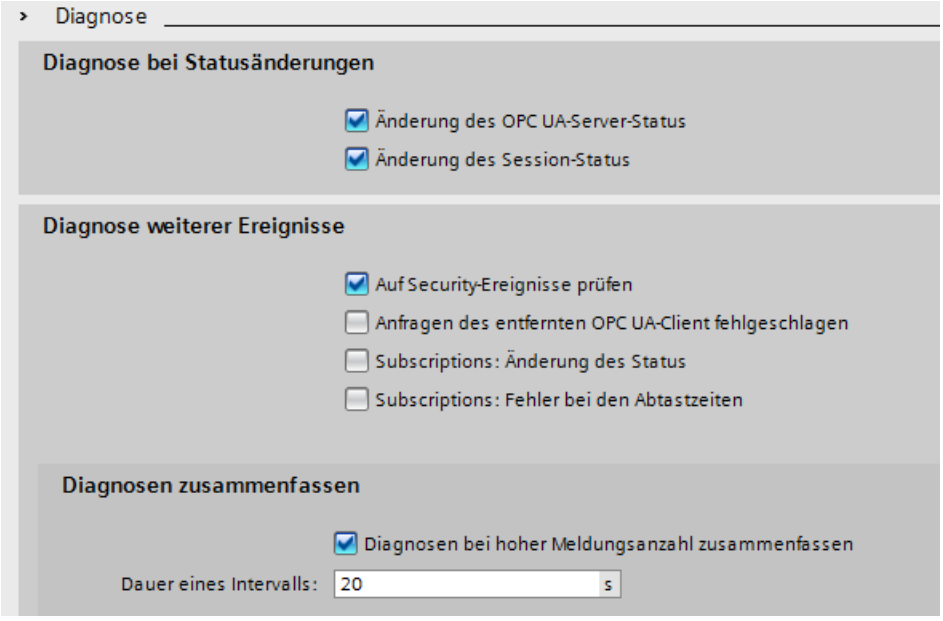


Bild 11-32 Diagnoseeinstellungen OPC UA-Server

Voreinstellung

Voreingestellt ist ein Diagnoseverhalten, das die wichtigsten Diagnosen ermöglicht, ohne die Kommunikationsbelastung nennenswert zu erhöhen.
Diagnosen für Subscriptions aktivieren Sie dann, wenn der OPC UA-Server auch Subscriptions bedient, ggf. nur in der Inbetriebnahmephase.
Grund: Viele Diagnosen erzeugen eine hohe Kommunikationslast in der CPU und verdrängen unter Umständen andere wichtige Meldungen. Oder wichtige Meldungen gehen bei hohem Diagnoseaufkommen in der Masse der Meldungen unter und werden nicht beachtet.

Weitere Informationen

Weitere Informationen zu Bedeutung und Wirkung der oben gezeigten Einstellungen finden Sie hier (Seite 317).

11.3.3.10 Lizenzen für OPC UA

Runtime-Lizenzen

Für den Betrieb des OPC UA-Servers der S7-1500 CPU ist eine Lizenz erforderlich. Der Typ der erforderlichen Lizenz ist abhängig von der Leistung der jeweiligen CPU. Die folgenden Lizenz-Typen werden unterschieden:

- SIMATIC OPC UA S7-1500 small (erforderlich für CPU 1511, CPU 1512, CPU 1513, ET 200SP CPUs, CPU 1515SP PC)
- SIMATIC OPC UA S7-1500 medium (erforderlich für CPU 1515, CPU 1516, Softwarecontroller CPU 1507, CPU 1516pro-2PN)
- SIMATIC OPC UA S7-1500 large (erforderlich für CPU 1517, CPU 1518)

Der erforderliche Lizenz-Typ wird angezeigt unter "Eigenschaften > Allgemein > Runtime-Lizenzen > OPC-UA > Typ der benötigten Lizenz":

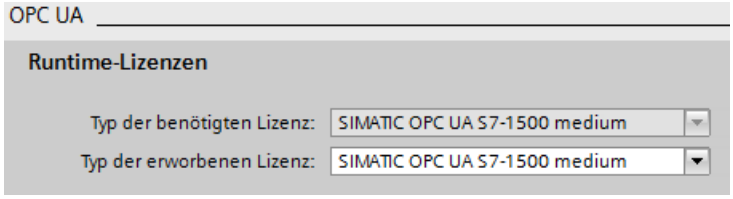


Bild 11-33 Runtime-Linzenzen OPC UA-Server

Um den Erwerb der erforderlichen Lizenz zu bestätigen, gehen Sie folgendermaßen vor:

1. Klicken Sie in den Eigenschaften der CPU auf "Runtime-Lizenzen > OPC UA".
2. Wählen Sie in der Klappliste bei "Typ der erworbenen Lizenz" die notwendige Lizenz aus.

11.3.4 OPC UA-Server-Schnittstelle projektieren

11.3.4.1 Was ist eine Server-Schnittstelle?

Definition

Eine Server-Schnittstelle fasst Knoten eines OPC UA-Adressraums einer CPU zu einer Einheit zusammen, sodass eine bestimmte Sicht auf diese CPU für OPC UA-Clients bereitgestellt wird.
Jede Server-Schnittstelle definiert einen oder mehrere Namensräume im OPC UA-Server der CPU.
STEP 7 (TIA Portal) unterscheidet die folgenden Typen von Server-Schnittstellen:

- Companion-Spezifikation
  Für diesen Typ einer Server-Schnittstelle verwenden Sie z. B. eine Companion Spezifikation, die eine Arbeitsgruppe erstellt hat.
  Die Arbeitsgruppe setzt sich typischerweise zusammen aus Mitgliedern der OPC Foundation und einer anderen Industrieorganisation, die gemeinsam ein OPC UA-Informationsmodell für einen bestimmten Zweck spezifiziert haben (zum Beispiel für den Datenaustausch mit RFID-Geräten oder mit Spritzgießmaschinen).
  Dieses Informationsmodell wird in Form von OPC UA-Knoten im Adressraum eines OPC UA-Servers realisiert. Auf diese OPC UA-Knoten können OPC UA-Clients zugreifen.

Den Server-Schnittstellentyp "Companion-Spezifikation" können Sie auch verwenden, um z. B. firmenintern erstellte Informationsmodelle zu laden, die z. B. in SiOME erstellt wurden.

Falls Sie in Ihrem Projekt eine bestimmte Companion-Spezifikation umsetzen, dann übernehmen Sie die Vorgaben dieser Companion-Spezifikation in Ihr Projekt als Server-Schnittstelle.

Für Server-Schnittstellen vom Typ Companion-Spezifikationen können Sie mehrere Namensräume importieren, die die Companion-Spezifikation verwendet.

Weitere Informationen zu Companion Spezifikationen finden Sie hier (Seite 257).

Weitere Informationen zu SiOME finden Sie hier (https://support.industry.siemens.com/cs/ww/de/view/109755133).

- Wenn Companion Spezifikationen auf Typdefinitionen in abhängigen Spezifikationen zurückgreifen, dann nutzen Sie hierfür Referenz-Namensräume.
  Referenz-Namensräume importieren Sie genau so wie die Companion Spezifikationen selber.
  Siehe Server-Schnittstelle für Companion Spezifikation anlegen (Seite 264).
- Wenn Sie Instanzdaten von FBs oder UDTs der CPU für OPC UA Clients zugänglich machen wollen, haben Sie die Möglichkeit, ab TIA Portal Version V17 diese Instanzdaten-Zuweisungen automatisch vornehmen zu lassen. Sie müssen lediglich die FB-Typen bzw. die UDTs auf geeignete OPC UA-Datentypen von einem importierten Referenz-Namensräumen mappen. Damit dieses Mapping möglich ist, aktivieren Sie die Option "Erzeuge OPC UA-Knoten basierend auf dem Lokaldaten-Mapping" im Dialog zum Anlegen einer OPC UA-Server-Schnittstelle vom Typ Companion-Spezifikation/Referenz-Namensraum.
  Siehe OPC UA-Knoten erzeugen basierend auf Lokaldaten-Mappings von FB-Typen und UDTs (Seite 285)

- Benutzerdefinierte Server-Schnittstelle:
  Für diesen Typ einer Server-Schnittstelle fassen Sie OPC UA-Knoten eines OPC UA-Servers zu einer Einheit zusammen.
  Sie orientieren sich dabei an den Vorgaben für Ihr Projekt oder an den Erfordernissen für Ihre Maschine oder Ihre Anlage.
  Weitere Informationen zur benutzerdefinierten Server-Schnittstelle finden Sie hier (Seite 269).

Spritzgießmaschine als Beispiel für Companion Spezifikation

In diesem Beispiel enthält eine Server-Schnittstelle folgende Elemente:

- OPC UA-Knoten, die Sie mit einem OPC UA-Client lesen können, um Informationen über diese Spritzgießmaschine zu erhalten (aus lesbaren PLC-Variablen),
- OPC UA-Knoten, die Sie mit einem OPC UA-Client schreiben können, um Werte in die Spritzgießmaschine zu übertragen (in schreibbare PLC-Variablen),
- OPC UA-Knoten, die Sie mit einem OPC UA-Client aufrufen können, um Funktionen der Spritzgießmaschine zu starten (über Server-Methoden).

Diese Server-Schnittstelle erlaubt eine standardisierte Sicht auf eine CPU, die als Steuerung einer Spritzgießmaschine dient.

Für Spritzgießmaschinen definieren die Companion Spezifikationen "OPC UA specifications for plastics and rubber machines" (früher "Euromap") eine ganze Reihe von OPC UA-Knoten, die Sie als Server-Schnittstelle übernehmen können.

Andere OPC UA-Knoten der CPU sind nicht in dieser Server-Schnittstelle enthalten. Das erhöht den Überblick.

Beispiel für benutzerdefinierte Server-Schnittstelle

Eine CPU soll die Herstellung von Werkstücken steuern. Wenn vom übergeordneten Leitsystem ein Produktionsauftrag eintrifft, beginnt die Produktion.
Die Produktionsaufträge werden über eine Server-Methode übergeben: Ein Leitsystem überträgt Informationen zu einem Werkstück, indem es die Server-Methode in der CPU aufruft. Zudem startet diese Server-Methode auch die Produktion.
Das Leitsystem, d. h. der angeschlossene OPC UA-Client, soll lediglich diese eine Server-Methode sehen. Deshalb legen Sie eine benutzerdefinierte Server-Schnittstelle in der CPU an und ordnen dieser Server-Schnittstelle die Server-Methode zu. Sie geben nur diese Server-Schnittstelle für OPC UA-Clients frei und begrenzen dadurch die Sicht auf die CPU auf diese eine Funktion.

11.3.4.2 OPC UA Companion Spezifikationen verwenden

Einleitung

OPC UA ist universell einsetzbar: Der Standard selbst macht z. B. keine Aussagen darüber, wie PLC-Variablen zu benennen sind. Auch steht es im Ermessen des einzelnen Nutzers (Anwendungsentwicklers), Server-Methoden zu programmieren und zu benennen, die über OPC UA aufrufbar sind.

Informationsmodellierung und Standardisierung für Geräte und Branchen

Für gleichartige Anwendungen bietet es sich an, mit dem "OPC UA-Baukasten" sein Geräteinterface oder Maschineninterface zu standardisieren.
Viele Gremien und Arbeitskreise haben die Standardisierung vorangetrieben und unterschiedliche Companion Spezifikationen erarbeitet.
In diesen Spezifikationen ist festgelegt:

- Mit welchen Objekten, Methoden und Variablen ist ein typisches Gerät oder eine typische Maschine zu beschreiben.
- Welcher Namensraum ist für die genannten Objekte vorgesehen.

Maschinen werden dabei typischerweise in funktionale bzw. technologische Einheiten strukturiert und diese Einheiten standardisiert.
Den Betreibern von Maschinen und Anlagen bieten Companion Spezifikationen den Vorteil einer einheitlichen Schnittstelle. So sind zum Beispiel alle RFID-Reader, welche die Spezifikation AutoID einhalten, auf die gleiche Weise integrierbar. Das heißt, alle RFID-Reader, die konform der Spezifikation AutoID sind, lassen sich auf die gleiche Weise von OPC UA-Clients ansprechen, unabhängig vom Hersteller der RFID-Reader.
Ein anderes Beispiel für eine Companion Spezifikation aus dem Bereich Spritzgießmaschinen ist die Euromap 77 Companion Spezifikation.

Das folgende Kapitel beschreibt am Beispiel von Euromap 77, wie Sie eine Companion Spezifikation in STEP 7 (TIA Portal) übernehmen und dafür PLC-Variablen anlegen.

HINWEIS

EUROMAP und die OPC Foundation haben die Joint Working Group "OPC UA Plastics and Rubber Machinery" etabliert

Die bestehenden EUROMAP Empfehlungen EUROMAP 77 (data exchange between injection moulding machines and MES), 82.1 (temperature control devices) und 83 (general definitions) sind veröffentlicht worden unter dem neutralen Schirm der OPC Foundation als OPC 40077, 40082-1 and 40083.

Wesentliche Änderung ist die Änderung des Namensraums, z. B. für EUROMAP 77: Aktuell "http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/".

Die im Folgenden aufgeführten Beispiele verwenden die bisher gültigen Bezeichnungen und Verweise.

Beispiel Euromap 77 (aktuell: OPC 40077)

Euromap 77 bzw. der Nachfolge-Standard OPC 40077 standardisiert den Datenaustausch zwischen Spritzgießmaschinen und dem übergelagerten MES (Manufacturing Execution System). Dadurch kann das MES alle unterlagerten Spritzgießmaschinen gleich anbinden.

Die einheitliche Datenschnittstelle erleichtert die Integration von Spritzgießmaschinen in eine Anlage.

Companion Spezifikation verwenden: Übersicht

Die Euromap 77 ist in der OPC UA-XML-Datei "Opc_Ua.EUROMAP77.NodeSet2.xml" beschrieben.

HINWEIS

Euromap 77, Euromap 83 und OPC UA for Devices (DI)

Mit dem Release Candidate 2 wurde ein Teil der Definitionen von Euromap 77 nach Euromap 83 (aktuell OPC 40083) übertragen. Deshalb müssen Sie auch die OPC UA-Server-Schnittstelle von Euromap 83 importieren.

"OPC UA for Devices" ist ein allgemein gültiges Informationsmodell für die Konfiguration von Hardware- und Softwarekomponenten. Das Informationsmodell dient auch als Basis für weitere Companion Standards und wird daher ebenfalls importiert.

Sie erhalten die OPC UA-XML-Dateien hier:
Euromap77 (https://www.euromap.org/euromap77)
Euromap83 (https://www.euromap.org/euromap83)
OPC UA for Devices (https://opcfoundation.org/UA/schemas/DI/)

Diese XML-Dateien definieren eine OPC UA-Schnittstelle einer Spritzgießmaschine, die konform der Euromap 77 ist.

Euromap 77 verwenden: Übersicht

Um die Euromap 77 zu verwenden, gehen Sie folgendermaßen vor:

1. Erstellen Sie mit dem Programm SiOME eine XML-Datei, in der Sie eine Instanz des Typs "IMM_MES_InterfaceType" anlegen.
   Wie Sie dabei vorgehen ist unten im "Schritt 1: In SiOME Instanzen erstellen" beschrieben.
2. Legen Sie in STEP 7 (TIA Portal) PLC-Variablen und Server-Methoden an, die der Instanz des Typs "IMM_MES_InterfaceType" entsprechen (erstellt im Schritt 1).
   Wie Sie dabei vorgehen, das ist unten, im "Schritt 2: In STEP 7 PLC-Variablen anlegen" beschrieben.
   Ein Beispiel für OPC UA-Knoten und den entsprechenden PLC-Variablen ist im Kapitel "Server-Schnittstelle für Companion Spezifikation anlegen (Seite 264)" zu finden.
3. Fügen Sie in STEP 7 (TIA Portal) eine neue Server-Schnittstelle vom Typ Companion Spezifikation hinzu und importieren Sie die XML-Datei, die Sie im Schritt 1 erstellt haben.
   Wie Sie dabei vorgehen, das ist im Kapitel "Server-Schnittstelle für Companion Spezifikation anlegen (Seite 264)" beschrieben.
4. Ordnen Sie den OPC UA-Knoten der neuen Server-Schnittstelle die entsprechenden PLC-Variablen zu, die Sie im Schritt 2 angelegt haben.
   Wie Sie dabei vorgehen, das ist ebenfalls im Kapitel "Server-Schnittstelle für Companion Spezifikation anlegen (Seite 264)" beschrieben.

Schritt 1: In SiOME Instanzen erstellen

Im folgenden ist beschrieben, wie Sie das kostenlose Programm "SiOME" nutzen, den "Siemens OPC UA Modeling Editor".

Sie können mit SiOME eine OPC UA-XML-Datei erstellen, die eine Server-Schnittstelle (ein Informationsmodell) beschreibt.

Download-Link und Erläuterungen zu SiOME finden Sie hier (https://support.industry.siemens.com/cs/ww/de/view/109755133).

Vorgehensweise in STEP 7

Um die neue Server-Schnittstelle zu nutzen, importieren Sie die Server-Schnittstelle in das STEP 7-Projekt, siehe Kapitel "Server-Schnittstelle für Companion Spezifikation anlegen (Seite 264)".

Wenn das Projekt in die CPU geladen ist, steht die neue Server-Schnittstelle für OPC UA-Clients zur Verfügung.

Vorgehensweise in SiOME 1.7.3

HINWEIS

Die folgende Beschreibung zeigt die Arbeitsschritte in SiOME 1.7.3.

Folgeversionen von SiOME erleichtern Ihnen z. B. das Anlegen von entsprechenden DBs, Strukturen, Variablen oder Methoden im Anwenderprogramm. Über Drag & Drop können Sie z. B. Daten von SiOME in das TIA Portal (Anwenderprogramm) übertragen. Die Variablen etc. sind dann auch schon korrekt gemappt bzw. bei Methoden werden auch die entsprechenden FB-Elemente im Anwenderprogramm korrekt generiert.

Laden Sie sich die aktuelle Version von SiOME unter dem oben angegebenen Download-Link herunter und folgen Sie den Anweisungen in der dort bereitgestellten Dokumentation.

Die folgende Beschreibung zeigt die Arbeitsschritte in SiOME 1.7.3.

Um die Euromap 77 zu verwenden, erzeugen Sie eine XML-Datei mit einer Instanz von "IMM_MES_InterfaceType".

Erst durch die Instanziierung des Objekttyps entsteht die Ausprägung (das Informationsmodell) der konkreten Maschine im Adressraum des OPC UA-Servers.

Der Objekttyp "IMM_MES_InterfaceType" ist der Root-Objekttyp der Euromap 77. "IMM" steht für "Injection Moulding Machine".

Gehen Sie folgendermaßen vor:

1. Laden Sie die Dateien "Opc_Ua.EUROMAP77.NodeSet2.xml" und "Opc_Ua_EUROMAP83_NodeSet2.xml" von der Website der Euromap (siehe oben).
2. Laden Sie die Datei "Opc.Ua.Di.NodeSet2.xml" von der Website der OPC Foundation.
   In der Datei "Opc.Ua.Di.NodeSet2.xml" sind Typ-Definitionen enthalten, die die Euromap 77 verwendet.
3. Starten Sie SiOME.
4. Importieren Sie zunächst den Namensraum "http://opcfoundation.org/UA/DI/"
   Dazu klicken Sie im Bereich "Information model" auf die Schaltfläche "Import XML".

   Bild 11-34 Schaltfläche "Import XML" in SiOME

   SiOME zeigt den Dialog zum Öffnen von Dateien an.
5. Wählen Sie die Datei "Opc.Ua.Di.NodeSet2.xml" aus und klicken Sie auf die Schaltfläche "Öffnen", um die Datei zu importieren.
   Ergebnis: SiOME importiert die XML-Datei und zeigt im Bereich "Namespaces" den Namensraum "http://opcfoundation.org/UA/DI/" an.
   Der Standard-Namensraum "http://opfoundation.org/UA/" ist in SiOME immer vorhanden und muss nicht importiert werden.
6. Importieren Sie nun den Namensraum "http://www.euromap.org/euromap83/"
   Dazu klicken Sie wieder im Bereich "Information model" auf die Schaltfläche "Import XML".
   Wählen Sie die Datei "Opc_Ua.EUROMAP83.NodeSet2.xml" aus.
   Ergebnis: SiOME importiert die XML-Datei und zeigt im Bereich "Namespaces" nun auch den Namensraum "http://www.euromap.org/euromap83/" an.
7. Importieren Sie nun den Namensraum "http://www.euromap.org/euromap77/"
   Dazu klicken Sie wieder im Bereich "Information model" auf die Schaltfläche "Import XML".
   Wählen Sie die Datei "Opc_Ua.EUROMAP77.NodeSet2.xml" aus.
8. Legen Sie für Ihr Projekt einen eigenen Namensraum an.
   Dazu klicken Sie im Bereich "Namespaces" mit der rechten Maustaste auf "OPC UA Modelling Editor Project" oder auf "Namespaces" und wählen "Add Namespace".
   SiOME öffnet den Dialog "Add Namespace".

9. Tragen Sie den Namen eines neuen Namensraums ein.
   Im Beispiel ist der Namensraum "YourCompany.org" verwendet.
   SiOME zeigt nun auch den neuen Namensraum an:

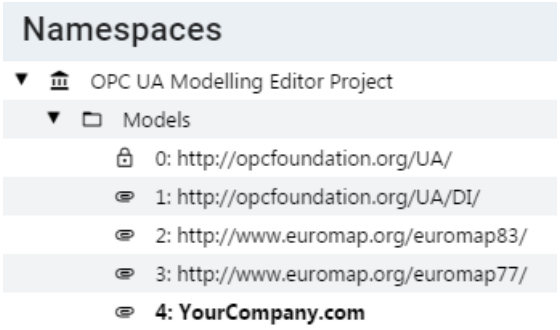


Bild 11-35 Anzeige des Namensraums in SiOME

10. Erzeugen Sie eine Instanz aus dem Root-Objekttyp IMM_MES_InterfaceType der Companion Spezifikation Euromap 77.
    Klicken Sie dazu im Bereich "Information model" mit der rechten Maustaste auf das Verzeichnis "DeviceSet" und wählen Sie "Add Instance".
    SiOME zeigt den Dialog "Add Instance" an.
11. Fügen Sie bei "Name" einen aussagekräftigen Namen für die neue Instanz ein.
    Im Beispiel tragen Sie "IMM_Manufacturer_01234" ein.
    Bei "TypeDefinition" wählen Sie "IMM_MES_InterfaceType".
    Dieser Objekttyp ist der Root-Objekttyp der Euromap 77: Wenn Sie eine Instanz dieses Objekttyps bilden, dann verwenden Sie die Euromap 77 einmal im Adressraum Ihres OPC UA-Servers.

12. Klicken Sie auf "OK".
    SiOME zeigt die neue Instanz "IMM_Manufacturer_01234" im Bereich "Information model" unter "DeviceSet" an:

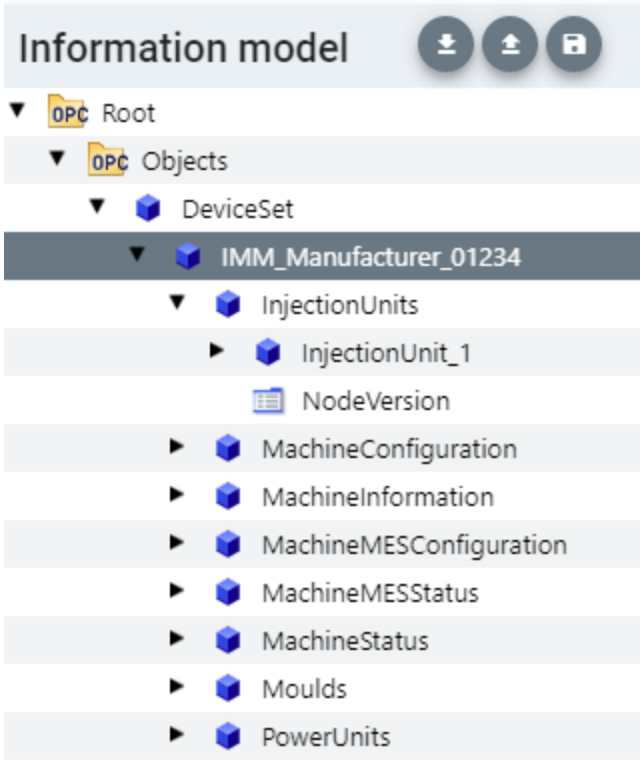


Bild 11-36 Anzeige information modek

13. Legen Sie eine Instanz des Objekttyps "InjectionUnitType" an.
    Dazu klicken Sie im Bereich "Information model" mit der rechten Maustaste auf das Verzeichnis "InjectionUnits" und wählen "Add Instance".
    SiOME zeigt den Dialog "Add Instance" an.
    Fügen Sie bei "Name" einen aussagekräftigen Namen für die Instanz ein.
    Im Beispiel tragen Sie "InjectionUnit_1" ein.
    Bei "TypeDefinition" wählen Sie "InjectionUnitType".
    Klicken Sie auf "OK".
14. Legen Sie im Verzeichnis "Moulds" eine neue Instanz "Mould_1" des Objekttyps "MouldType" an.
15. Legen Sie im Verzeichnis "PowerUnits" eine neue Instanz "PowerUnit_1" des Objekttyps "PowerUnitType" an.
16. Speichern Sie die XML-Datei.
    Dazu klicken Sie im Bereich "Information model" auf die Schaltfläche "Quick save":

Bild 11-37 Schaltfläche "Quick save" in SiOME

17. Exportieren Sie die XML-Datei.
    Dazu klicken Sie im Bereich "Information model" auf die Schaltfläche "Export XML".

Bild 11-38 Schaltfläche "Export XML" in SiOME

    SiOME blendet den Dialog "Export XML" ein.
18. Lassen Sie alle Namensräume aktiviert und klicken Sie auf "OK".
    SiOME zeigt den Dialog "Speichern unter" an.
19. Wählen Sie einen aussagekräftigen Namen und speichern Sie die exportierte Datei.
    Im Beispiel nennen Sie die XML-Datei "IMM_Manufacturer_01234".

Ergebnis:
Sie haben eine XML-Datei erstellt, die die Companion Spezifikation Euromap 77 einmal verwendet (mit einer Instanz).

Schritt 2: In STEP 7 PLC-Variablen für die Euromap-77-Instanz anlegen

Für die Euromap 77 müssen Sie in Ihrem Anwenderprogramm PLC-Variablen und Server-Methoden bereitstellen und der Instanz des Typs "IMM_MES_InterfaceType" zuordnen.
Gehen Sie folgendermaßen vor, um PLC-Variablen für die Instanz des Typs "IMM_MES_InterfaceType" anzulegen.

1. Legen Sie einen Anwender-definierten Datentyp (UDT) an.
   Das folgende Bild zeigt als Beispiel den Anfang des Anwender-definierten Datentyps "InjectionUnit".
   Dieser Datentyp besitzt die selbe Struktur wie "InjectionUnit" im Typ "IMM_MES_InterfaceType".
   Achten Sie darauf, dass Sie SIMATIC-Datentypen verwenden, die zu den OPC UA-Datentypen kompatibel sind (siehe unten "Mapping der Datentypen").

InjectionUnit

	Name	Datentyp	Defaultwert	Erreichbar aus HMI/OPC UA	Schreibbar aus HMI/OPC UA
	BarrelId	String	''	☑	☑
	Index	UDInt	0	☑	☑
	InProduction	Bool	false	☑	☑
	IsPresent	Bool	false	☑	☑

Bild 11-39 UDT in STEP 7 anlegen

2. Fügen Sie Ihrem STEP 7-Projekt einen neuen globalen Datenbaustein hinzu.
   Im Beispiel nennen Sie den Datenbausein "IMM_Manufacturer_01234", sodass ein Bezug zur Spritzgießmaschine des jeweiligen Herstellers und der Seriennummer besteht.
3. Legen Sie in diesem Datenbaustein ein neues Element an.
   Im Beispiel nennen Sie dieses Element "InjectionUnit_1"
4. Weisen Sie diesem Element den neuen Anwender-definierten Datentyp "InjectionUnit" zu.

Ergebnis

In Ihrem STEP 7-Projekt haben Sie im Datenbaustein "IMM_Manufacturer_01234" eine Variable für die Euromap 77 angelegt.

11.3.4.3 Server-Schnittstelle für Companion Spezifikation anlegen

Grundlegende Informationen zu Companion Spezifikationen finden Sie im Kapitel "OPC UA Companion Spezifikationen verwenden (Seite 257)". Dort wird auch ausführlich auf den Nutzen der Companion Spezifikation Euromap 77 eingegangen, die ein Modell für Spritzgießmaschinen zur Verfügung stellt.

Die S7-1500-CPU kann mithilfe dieses Companion Standards z. B. eine Spritzgießmaschine steuern und einem OPC UA-Client, z. B. einem übergeordneten MES-System, eine Schnittstelle für den Zugriff auf Funktionen und Variablen der Spritzgießmaschine ermöglichen.

Eine OPC UA-Server-Schnittstelle vom Typ "Companion Standard" begrenzt den Zugriff von Clients auf genau die Funktionen und Variablen, die z. B. für übergeordnete Systeme (MES-Systeme) erforderlich sind.

Die folgende Beschreibung zeigt, wie Sie in STEP 7 (TIA Portal) eine Server-Schnittstelle anlegen, die nur die Companion Spezifikation Euromap 77 enthält.

Wenn Sie OPC UA-Clients darüber hinaus weitere Variablen oder Methoden zugänglich machen wollen als die, die für das Management einer Spritzgießmaschine notwendig sind, dann legen Sie einfach eine weitere OPC UA-Server-Schnittstelle an. Auf diese Weise können Sie die Funktionalität der CPU als OPC UA-Server übersichtlich ordnen.

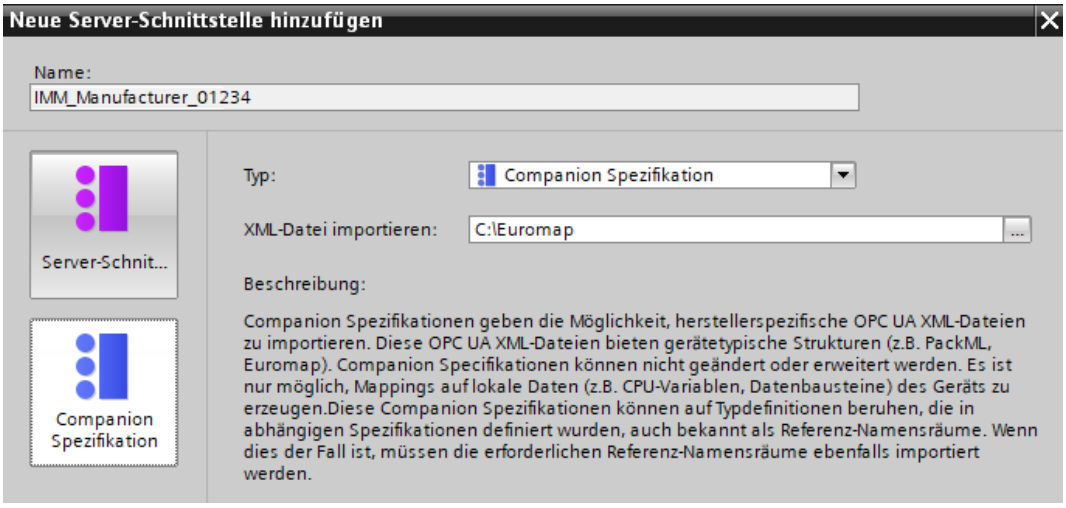
Server-Schnittstelle für eine Companion Spezifikation anlegen

Um mit STEP 7 (TIA Portal) eine Server-Schnittstelle für eine Companion Spezifikation zu erstellen, gehen Sie folgendermaßen vor:

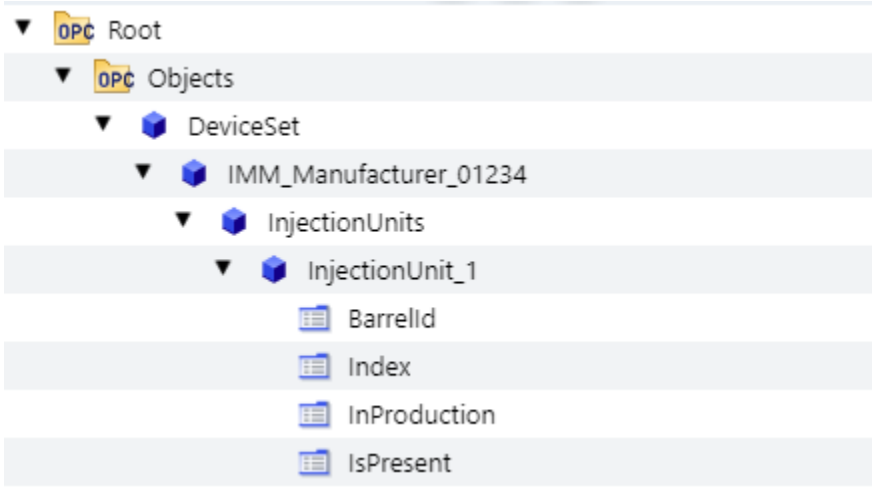
1. Wählen Sie die CPU aus, die Sie als OPC UA-Server nutzen.
2. Klicken Sie in der Projektnavigation auf "OPC UA-Kommunikation > Server-Schnittstellen".
3. Doppelklicken Sie auf "Neue Server-Schnittstelle hinzufügen".
4. Klicken Sie auf die Schaltfläche "Companion Spezifikation", um diesen Typ einer Server-Schnittstelle auszuwählen.

   Im Dialog ist ein allgemeiner Name für die neue Server-Schnittstelle eingetragen, zum Beispiel "Server-Schnittstelle_1".

5. Ändern Sie den Namen der neuen Server-Schnittstelle, sodass er in Ihrem Projekt aussagekräftig ist.
   Der Name sollte gemäß der Euromap 77 die folgende Struktur besitzen: "IMM_<Hersteller>_<Seriennummer>".
   Das Beispiel verwendet den Namen "IMM_Manufacturer_01234".



6. Im Feld "Import XML-Datei" wählen Sie eine XML-Datei aus, in der ein Informationsmodell beschrieben ist.
   Wie Sie eine solche XML-Datei mit dem Tool SiOME erstellen, beschreibt das Kapitel "OPC UA Companion Spezifikationen verwenden (Seite 257)".
   Das folgende Bild zeigt einen Ausschnitt aus dem Informationsmodell: "IMM_MANUFACTURER_0123456" eine Instanz (Verwendung) des Typs "IMM_MES_InterfaceType", der von Euromap 77 festgelegt wurde. "InjectionUnit_1" ist eine Instanz des Typs "InjectionUnitType" der Euromap 77.

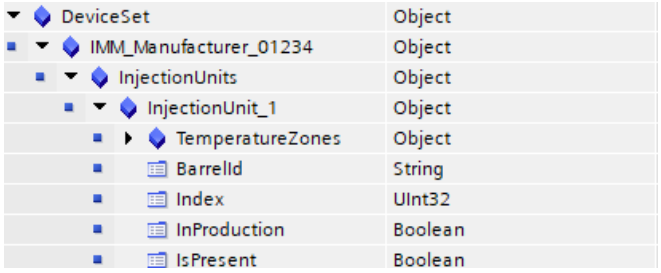


7. Klicken Sie auf die Schaltfläche "OK".
   STEP 7 (TIA Portal) importiert das Informationsmodell, das in der gewählten XML-Datei beschrieben ist.
   Ein Fehler tritt auf, wenn in der importierten XML-Datei Typ-Definitionen verwendet werden, die noch nicht in STEP 7 (TIA Portal) vorliegen und die auch nicht in der importierten XML-Datei enthalten sind.

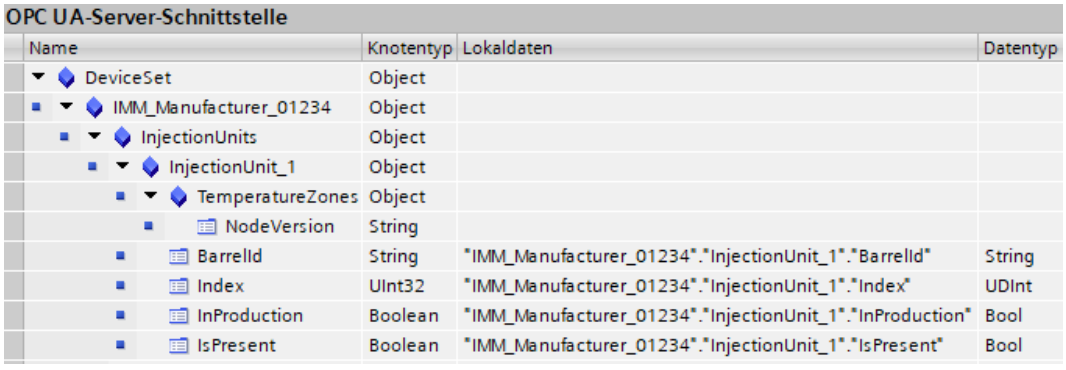
Im Beispiel wird eine XML-Datei importiert, die Typ-Definitionen verwendet, die in den folgenden Namensräume (Namespaces) definiert sind:

- http://opcfoundation.org/UA/DI/
- http://www.euromap.org/euromap83/
- http://www.euromap.org/euromap77/

Tipp: Fehlende Namensräume zeigt STEP 7 im unteren Bereich des OPC UA-Schnittstellen-Editors an (Register "Eigenschaften").
Dazu markieren Sie in der Projektnavigation die Server-Schnittstelle (hier: IMM_Manufacturer_01234) und wählen im Inspektorfenster den Bereich "Namensräume". Fehlende Namensräume sind markiert.
Falls in Ihrem STEP 7-Projekt ein oder mehrere Namensräume fehlen, dann legen Sie für jeden Namensraum eine neue Server-Schnittstelle des Typs "Referenz-Namensraum" an. Wie Sie dabei vorgehen, das zeigt das Kapitel "Server-Schnittstelle für Referenz-Namensraum anlegen (Seite 282)".
Liegen alle Referenz-Namensräume vor, dann zeigt STEP 7 die Tabelle ohne Fehler an:



8. Ziehen Sie per Drag&Drop vom rechten Bereich der Tabelle (OPC UA-Elemente) zum linken Teil der Tabelle (OPC UA-Server-Schnittstelle), sodass die jeweiligen OPC UA-Elemente (die lokalen PLC-Variablen) den jeweiligen OPC UA-Knoten der Euromap 77 zugeordnet sind.
   Das folgende Bild zeigt einen Ausschnitt aus der Zuordnung der lokalen Daten (PLC-Variablen) zu den OPC UA-Knoten der Euromap 77:



ACHTUNG
Mapping von Lokaldaten der CPU auf Knoten der OPC UA-Server-Schnittstelle prüfen Wenn ungütige Zuweisungen (Mappings) in der Server-Schnittstelle existieren, dann können falsche Lese- und Schreiboperationen die Folge sein. Prüfen Sie die Zuweisungen und führen Sie einen Konsistenzcheck durch.

Informationen zur Server-Schnittstelle

Der Editor zur Projektierung der OPC UA-Server-Schnittstelle ist als Tabelle aufgebaut und stellt die folgenden Informationen bereit:

- **Name**
  Der oberste Knoten (Root-Knoten) trägt im Beispiel den Namen "IMM_Manufacturer_01234". Wenn ein Client im Adressraum des Servers browst, dann ist dieser Knoten der Container für alle unterlagerten Knoten. BrowseName und der DisplayName dieses Knotens sind abhängig vom Namen, den Sie für die Server-Schnittstelle vergeben haben.
  Dieser Name steht hier z. B. für die Spritzgießmaschine als Ganzes. Er ist der Name der hier verwendeten Instanz der Companion Spezifikation Euromap 77. Gemäß der Companion Spezifikation soll der Instanzname mit "IMM" beginnen; dann folgt der Name des Herstellers der Spritzgießmaschine; am Ende wird die Seriennummer der Maschine angefügt. Dadurch lässt sich eine Maschine eindeutig identifizieren.
  Die Namen aller anderen (unterlagerten) Knoten sind durch die Spezifikation festgelegt (im Beispiel oben durch die Euromap 77). Diese Knotennamen dürfen nicht geändert werden. Das sichert eine einheitliche Sicht auf alle Spritzgießmaschinen, die sich an die Spezifikation halten.
- **Knotentyp**
  Typ des OPC UA-Knotens. Der Typ ist durch die verwendete Companion Spezifikation festgelegt.
  STEP 7 markiert in folgenden Fällen einen Knotentyp in der Tabelle farbig:
  – wenn dafür in der importierten XML-Datei keine Definition enthalten ist oder
  – wenn der Namensraum, in dem die Typ-Definition erfolgte, nicht in STEP 7 vorhanden ist.
  In diesem Fall legen Sie für den oder die fehlenden Namensräume jeweils eine Server-Schnittstelle des Typs "Referenz-Namensraum" an.
  Die fehlenden Namensräume finden Sie in den Eigenschaften der Server-Schnittstelle, unter "Namensräume".
- **Lokaldaten**
  STEP 7 zeigt den Datenbaustein an, der den OPC UA-Knoten zugeordnet ist: Von diesem Datenbaustein liest die CPU die Werte für die OPC UA-Knotens.
  Wenn ein Datenbaustein farbig hinterlegt ist (z. B. nach Konsistenzcheck), dann ist der angegebene Datenbaustein in der CPU nicht vorhanden.
  In diesem Fall müssen Sie den fehlenden Datenbaustein in der CPU (im Anwenderprogramm) anlegen und mit Werten versorgen.
- **Datentyp**
  Der SIMATIC-Datentyp der PLC-Variable (z. B. Element eines Datenbausteins) in der CPU, von dem der Wert eines OPC UA-Knotens (des Typs UAVariable) gelesen oder dem ein Wert zugewiesen wird.

Schnittstelle aktualisieren

Sie haben die Möglichkeit, die Server-Schnittstelle für Companion Speifikationen zu aktualisieren.
Beispiel: Nach dem Import von Companion Spezifikationen und anschließendem Import von Referenz-Namensräumen, von denen die Companion Spezifikationen abhängig sind, sind die Typdefinitionen der Referenz-Namensräume noch nicht wirksam.
Wenn Sie auf die Schaltfläche "Schnittstelle aktualisieren" klicken, werden die fehlenden Typdefinitionen der Companion Spezifikationen aufgelöst.
Schaltfläche "Schnittstelle aktualisieren":

Lokaldaten erzeugen

Sie haben die Möglichkeit, für alle oder für ausgewählte Knoten der Server-Schnittstelle, die nicht bereits Lokaldaten der CPU zugeordnet ("gemappt") sind, Lokaldaten zu erzeugen. Die neu erzeugten Lokaldaten werden automatisch gemappt.
Sie lösen das automatische Erzeugen von Lokaldaten aus, indem Sie auf die Schaltfläche "Lokaldaten erzeugen" klicken (alle nicht bereits gemappten Knoten) oder durch Auswahl einzelner Knoten und anschließendem Kontextmenü "Lokaldaten erzeugen".
Schaltfläche "Lokaldaten erzeugen":

Bild 11-40 Schaltfläche "Lokaldaten erzeugen"

Sie können nur Knoten erzeugen lassen, die auf Lokaldaten abbildbar sind, also keine Objekte, keine Ordner, keine Methoden oder Input-/Output-Argumente von Methoden.
Nachdem Sie auf die Schaltfläche geklickt oder das Kontextmenü gewählt haben, müssen Sie im Folgedialog wählen, ob die Lokaldaten in einem neuen DB angelegt werden sollen oder in einem bereits bestehenden DB.

Konsistenzcheck

Sie haben die Möglichkeit, die Konsistenz der Server-Schnittstelle zu überprüfen.
Dabei prüft STEP 7 (TIA Portal), ob den OPC UA-Knoten der Server-Schnittstelle PLC-Variablen (Datenbausteine) mit kompatiblen SIMATIC-Datentypen zugeordnet sind.
Um die Konsistenz der Server-Schnittstelle zu überprüfen, klicken Sie auf das folgende Symbol in der Funktionsleiste des OPC UA-Server Schnittstellen-Editors:

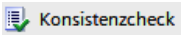


Bild 11-41 Schaltfläche "Konsistenzcheck"

Schnittstelle exportieren

Sie haben die Möglichkeit, die OPC UA-Server-Schnittstelle als XML-Datei zu exportieren. Diese XML-Datei enthält alle Datentyp-Definitionen, die die Server-Schnittstelle referenziert.
Um die OPC UA-Server-Schnittstelle zu exportieren, klicken Sie auf das folgende Symbol in der Funktionsleiste des OPC UA-Server Schnittstellen-Editors:

Bild 11-42 Schaltfläche "Schnittstelle exportieren"

11.3.4.4 Benutzerdefinierte Server-Schnittstelle anlegen

Einleitung

Diese Beschreibung geht von dem folgenden Beispiel aus:
Ein Schutzzaun umgibt die Fertigungszelle "Cell_1". Der Zaun besitzt das Tor "Gate_1".
Eine CPU S7-1500 steuert die gesamte Fertigungszelle und kontrolliert auch den Zugang durch Gate_1.
Ein Roboter verpackt in der Fertigungszelle Medikament in Kartons und stapelt die Kartons anschließend auf Paletten.
Selbstfahrende Flurförderzeuge transportieren die Paletten in das Zentrallager, dabei passieren sie Gate_1.
Die CPU veröffentlicht eine Server-Schnittstelle, über die fahrerlose Transportsysteme das Öffnen von Gate_1 veranlassen.
Die Server-Schnittstelle enthält die Server-Methode "smOpenGate" zum Öffnen des Tors sowie die Variable "Gate_1_State", die den Status des Tors anzeigt (geöffnet oder geschlossen).

Benutzerdefinierte Server-Schnittstelle anlegen

Um eine Server-Schnittstelle zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie die CPU aus, die Sie als OPC UA-Server nutzen und konfiguriert haben.
2. Klicken Sie auf "OPC UA-Kommunikation > Server-Schnittstellen".
3. Doppelklicken Sie auf "Neue Server-Schnittstelle hinzufügen".
   STEP 7 zeigt den folgenden Dialog an:

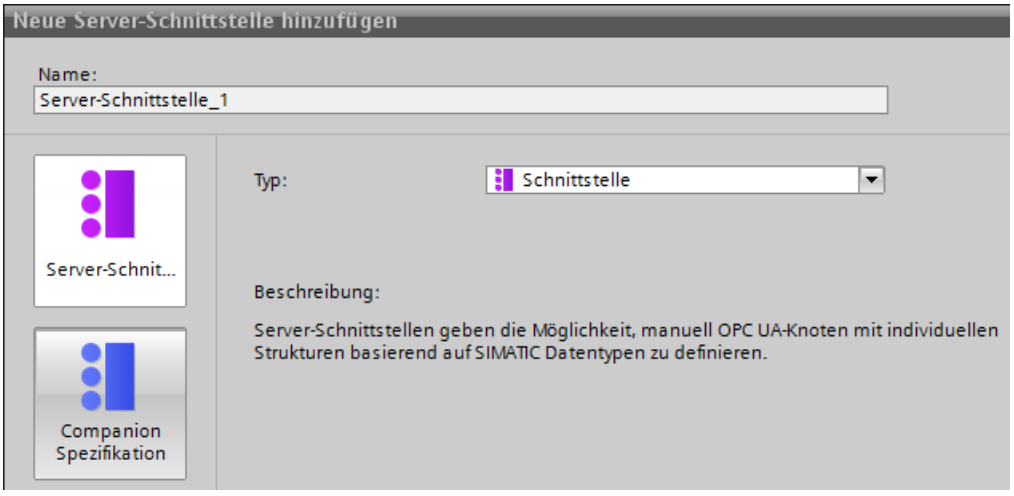


Bild 11-43 Server-Schnittstelle hinzufügen

4. Ändern Sie den Namen der neuen Server-Schnittstelle, sodass er in Ihrem Projekt aussagekräftig ist.
   Im Beispiel ändern Sie den Namen "Server-Schnittstelle_1", der von STEP 7 vorgeschlagen wird, in "Cell_1".
5. Klicken Sie auf die Schaltfläche "Server-Schnittstelle" und dann auf "OK".
6. Klicken Sie dazu auf das Dreieck vor "Programmbausteine" im Bereich "OPC UA-Elemente", um den Ordner "Programmbausteine" zu öffnen.
   STEP 7 zeigt die folgende Tabelle zum Editieren an:

Bild 11-44 Server-Schnittstelle editieren

Der Editor ist in zwei Bereiche geteilt:

- **OPC UA-Server-Schnittstelle**
  Auf der linken Seite befindet sich der Wurzelknoten der Server-Schnittstelle "Cell_1".
  Diese Schnittstelle ist aktuell noch leer: Der Server-Schnittstelle wurden noch keine OPC UA-Elemente hinzugefügt.
- **OPC UA-Elemente**
  Auf der rechten Seite befinden sich die OPC UA-Elemente.
  OPC UA-Elemente sind Objekte, die bislang im STEP 7-Projekt angelegt und die Eigenschaft "Erreichbar aus HMI/OPC UA" haben.
  Sie können die OPC UA-Elemente der neuen Server-Schnittstelle "Cell_1" hinzufügen.

7. Ziehen Sie per Drag&Drop OPC UA-Elemente in die Zeile "<Neu hinzufügen>" der neuen Server-Schnittstelle.

HINWEIS

Generell gilt: Wenn Sie Datenbausteine oder auch Technologieobjekte im linken Bereich der Tabelle ablegen, erzeugt STEP 7 (TIA Portal) ein Objekt in der Server-Schnittstelle. Darunter sind die Elemente der Datenbausteine als eigene Knoten angeordnet.

Wenn Sie Strukturen im linken Bereich der Tabelle ablegen, erzeugt STEP 7 einen Knoten für die Struktur als Ganzes und Knoten für jedes Element der Struktur.

Ebenso verhält es sich mit Arrays: Auch in diesem Fall erzeugt STEP 7 einen Knoten für das Array als Ganzes und Knoten für jedes Element des Arrays.

Wenn Sie eine Methode im linken Bereich der Tabelle ablegen, erzeugt STEP 7 einen einzigen Knoten; die Argumente der eingefügten Methode werden zur Information angezeigt.

Im Beispiel ziehen Sie die Variable "Gate_1_State" aus dem rechten Bereich in den linken Bereich zu "<Neu hinzufügen>".
Dann ziehen Sie die Server-Methode in den linken Bereich.
Diese Server-Methode befindet sich innerhalb des Datenbausteins "smOpenGate_DB [DB3]" im rechten Bereich.
STEP 7 (TIA Portal) zeigt den Dialog folgendermaßen an:

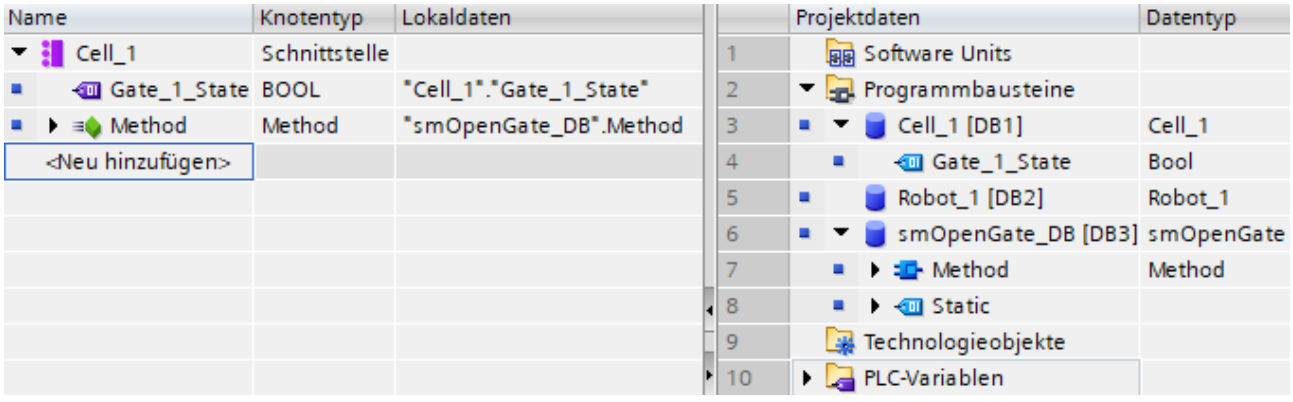


Bild 11-45 OPC UA-Elemente der Server-Schnittstelle hinzufügen

ACHTUNG
Mapping von Lokaldaten der CPU auf Knoten der OPC UA-Server-Schnittstelle prüfen Wenn ungütige Zuweisungen (Mappings) in der Server-Schnittstelle existieren, dann können falsche Lese- und Schreiboperationen die Folge sein. Prüfen Sie die Zuweisungen und führen Sie einen Konsistenzcheck durch. Dass das TIA Portal im Fall ungültiger Zuweisungen nur Warnungen und keine Fehler erzeugt, ermöglicht Ihnen, schrittweise vorzugehen: Sie können z. B. im ersten Schritt das Programm/die Lokaldaten ändern, sodass das Programm fehlerfrei läuft. Im nächsten Schritt ziehen Sie die OPC UA-Server-Schnittstelle nach und beseitigen die Inkonsistenzen. An den Stellen, wo das TIA Portal Warnungen erzeugt, funktioniert die OPC UA-Server-Schnittstelle zur Laufzeit nicht; der OPC UA-Server erzeugt Laufzeitfehler.

Den Blick auf OPC UA-Server begrenzen

Durch die Auswahl der OPC UA-Elemente begrenzen Sie den Blick auf den OPC UA-Server und die Möglichkeiten der OPC UA-Clients.

In der Server-Schnittstelle des Beispiels fehlt der Datenbaustein "Robot_1", weil Flurförderfahrzeuge keinen Zugriff auf die Server-Methoden und Variablen des Roboters benötigen.

Sinnvollerweise deaktivieren Sie in diesem Fall die Standard-Server-Schnittstelle (SIMATIC-Namensraum) in den OPC UA-Eigenschaften der S7-1500 CPU, damit die gefilterten Knoten nicht auf anderem Weg erreichbar sind.

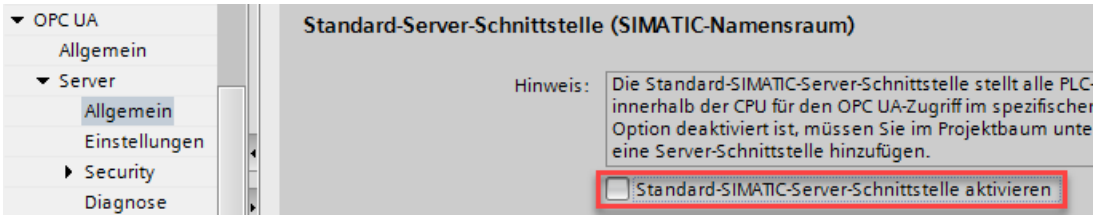


Bild 11-46 Standard-Server-Schnittstelle deaktivieren

Sie haben auch die Möglichkeit, die Sichtbarkeit jeder parametrierten OPC UA-Server-Schnittstelle in den Eigenschaften der Server-Schnittstelle zu deaktivieren und damit zu verhindern, dass diese Server-Schnittstelle im Betrieb von Clients genutzt werden kann.

- Markieren Sie dazu die Server-Schnittstelle und wählen mit der rechten Maustaste den Befehl "Eigenschaften".

Diese Option gibt Ihnen die Möglichkeit, z. B. zentral mehrere Server-Schnittstellen zu definieren und jeweils nur die benötigte Server-Schnittstelle zu aktivieren und zu laden.

Eine einmal definierte Server-Schnittstelle können Sie per Drag&Drop in der Projektnavigation auf eine andere CPU kopieren.

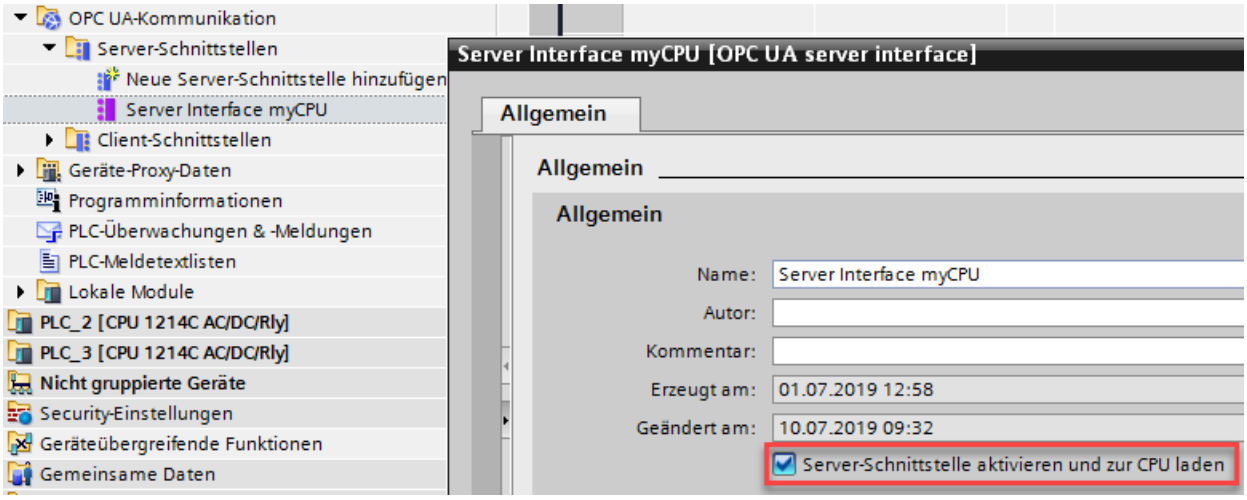


Bild 11-47 Sichtbarkeit der Server-Schnittstelle deaktivieren

Informationen zur Server-Schnittstelle

Der Editor für die Bearbeitung der OPC UA-Server-Schnittstelle ist als Tabelle aufgebaut und stellt die im Folgenden beschriebenen Informationen bereit.

Beachten Sie, dass zunächst nicht alle Spalten angezeigt werden. Was angezeigt wird, bestimmen Sie über Rechtsklick mit der Maustaste auf die Überschriften-Zeile der Tabelle.

Wenn eine Zeile markiert ist, können Sie die OPC UA-Attribute des Knotens im Inspektorfenster (Bereich "OPC UA-Attribute") einblenden, z. B. Knoten-ID, Knotenklasse, Knotentyp und die Beschreibung.

- **BrowseName**
  An oberster Stelle steht der sprachneutrale Name der benutzerdefinierten Server-Schnittstelle (BrowseName). Dieser Name ist frei wählbar.
  Unter dem Namen der Schnittstelle stehen die Namen (BrowseNames) der einzelnen OPC UA-Knoten, die der Server-Schnittstelle hinzugefügt wurden.
  Sie können den Namen eines OPC UA-Knotens in diesem Dialog nicht ändern. Die Namen stammen aus dem STEP 7-Projekt.
  Sie können einen OPC UA-Knoten aus der Tabelle löschen. Dadurch gehört er nicht mehr zur Server-Schnittstelle und ist für OPC UA-Clients nicht mehr sichtbar.
- **DisplayName**
  Wie BrowseName; allerdings ist der Name übersetzbar und wird, wenn vorhanden, in der entsprechenden Sprache angezeigt.
- **Knoten-ID**
  NodeId des OPC UA-Knotens, z. B. http://Server-Node_1; i=1
- **Knotentyp**
  Typ des OPC UA-Knotens, zum Beispiel BOOL, BYTE, INT, usf.
  Diese Knotentypen wurden von Siemens definiert, nicht von der OPC Foundation. Die OPC Foundation verwendet zum Beispiel für BOOL den Knotentyp Boolean. BOOL ist direkt von Boolean abgeleitet.

Der angegebene Knotentyp ist in diesem Dialog nicht änderbar: Wenn Sie einen anderen Knotentyp verwenden wollen, dann müssen Sie den Typ der jeweiligen PLC-Variablen im STEP 7-Projekt ändern.

- **Datentyp**
  Angegeben ist der SIMATIC-Datentyp, der im STEP 7-Projekt verwendet wird, zum Beispiel Bool, Byte, Int, usw.
- **Zugriffsstufe**
  - Falls ein OPC UA-Knoten eine Variable ist (Typ UAVariable), dann kann der Knoten nur lesbar sein (RD) oder les- und schreibbar (RD/WR).
  - Falls ein OPC UA-Knoten eine Methode ist (Typ UAMethod), dann ist dieser Knoten stets aufrufbar.
- **Lokaldaten**
  Der SIMATIC-Datentyp des Datenbausteins in der CPU, von dem der Wert eines OPC UA-Knotens (des Typs UAVariable) gelesen oder dem ein Wert zugeschrieben wird.

Lokaldaten erzeugen

Sie haben die Möglichkeit, für alle oder für ausgewählte Knoten der Server-Schnittstelle, die nicht bereits Lokaldaten der CPU zugeordnet ("gemappt") sind, Lokaldaten zu erzeugen. Die neu erzeugten Lokaldaten werden automatisch gemappt.

Sie lösen das automatische Erzeugen von Lokaldaten aus, indem Sie auf die Schaltfläche "Lokaldaten erzeugen" klicken (alle nicht bereits gemappten Knoten) oder durch Auswahl einzelner Knoten und anschließendem Kontextmenü "Lokaldaten erzeugen".

Schaltfläche "Lokaldaten erzeugen":

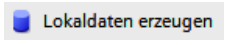


Bild 11-48 Schaltfläche "Lokaldaten erzeugen"

Sie können nur Knoten erzeugen lassen, die auf Lokaldaten abbildbar sind, also keine Objekte, keine Ordner, keine Methoden oder Input-/Output-Argumente von Methoden.

Nachdem Sie auf die Schaltfläche geklickt oder das Kontextmenü gewählt haben, müssen Sie im Folgedialog wählen, ob die Lokaldaten in einem neuen DB angelegt werden sollen oder in einem bereits bestehenden DB.

Konsistenzcheck

Sie haben die Möglichkeit, die Konsistenz der Server-Schnittstelle zu kontrollieren.

Beim Konsistenzcheck prüft STEP 7, ob die OPC UA-Knoten der Server-Schnittstelle jeweils einem passenden OPC UA-Element zugeordnet sind (gleicher Datentyp) oder ob das verwendete Element überhaupt noch existiert in der CPU.

Um die Konsistenz der Server-Schnittstelle zu überprüfen, klicken Sie auf das folgende Symbol in der Funktionsleiste des OPC UA-Server Schnittstellen-Editors:

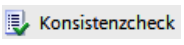


Bild 11-49 Schaltfläche "Konsistenzcheck"

Schnittstelle exportieren

Sie haben die Möglichkeit, die OPC UA-Server-Schnittstelle als XML-Datei zu exportieren. Diese XML-Datei enthält alle Datentyp-Definitionen, die die Server-Schnittstelle referenziert.
Um die OPC UA-Server-Schnittstelle zu exportieren, klicken Sie auf das folgende Symbol in der Funktionsleiste des OPC UA-Server Schnittstellen-Editors:

Bild 11-50 Schaltfläche "Schnittstelle exportieren"

Weitere Informationen

Informationen zu Kopiervorlagen für die OPC UA-Kommunikation finden Sie im Kapitel Kopiervorlagen für OPC UA-Kommunikation (Seite 364).

Siehe auch

Client-Zugriffe und lokale Zugriffe auf den OPC UA-Server (Seite 217)

11.3.4.5 Datentypen für Companion Spezifikationen

Mapping der Datentypen

Die folgende Tabelle zeigt den kompatiblen SIMATIC-Datentyp zum jeweiligen OPC UA-Datentyp.
Ordnen Sie die Datentypen zu wie unten gezeigt (SIMATIC-Datentyp - OPC UA-Datentyp). Andere Zuordnungen sind nicht zugelassen. STEP 7 prüft nicht die Einhaltung dieser Regel und verhindert nicht eine falsche Zuordnung. Sie sind für die regelkonforme Auswahl und Zuordnung der Datentypen verantwortlich.
Die aufgelisteten Datentypen können Sie auch z. B. als Elemente von Strukturen/UDTs für Eingangs- und Ausgangsparameter von selbst erstellten Server-Methoden (UAMethod_InParameters und UAMethod_OutParameters) verwenden.

Tabelle 11-3 Mapping der Datentypen

SIMATIC-Datentyp	OPC UA-Datentyp
BOOL	Boolean
SINT	SByte
INT	Int16
DINT	Int32
LINT	Int64
USINT	Byte
UINT	UInt16
UDINT	UInt32
ULINT	UInt64
REAL	Float
LREAL	Double

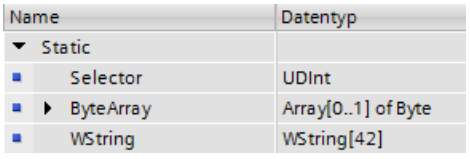
SIMATIC-Datentyp	OPC UA-Datentyp
LDT	DateTime
WSTRING	String
DINT	Enumeration (Encoding Int32) und alle davon abgeleiteten Datentypen
Anwenderdefinierter Datentyp erforderlich (UDT, user-defined data type) Der anwenderdefinierte Datentyp muss mit dem Präfix "Union_" angelegt werden, z. B. "Union_MyDatatype", siehe Beispiel unter der Tabelle. Das erste Element (Selector) in diesem UDT muss den Datentyp "UDINT" besitzen.	UNION und alle davon abgeleiteten Datentypen
siehe Datentypen LocalizedText und ByteString (Seite 276)	LocalizedText ByteString

Anwenderdefinierter Datentyp für UNION erforderlich

Das folgende Bild zeigt die Variable "MyVariable", die den Datentyp "Union_MyDatatype" besitzt.

Dieser SIMATIC-Datentyp entspricht einer OPC UA-Variaben mit dem Datentyp UNION.

Das Bild zeigt ein Beispiel für die Deklaration: Bei Selector = 1, nimmt die Union einen ByteArray auf, bei Selector = 2 einen WString.



11.3.4.6 Datentypen LocalizedText und ByteString

Ab TIA Portal Version V17 und S7-1500 CPU Firmwareversion V2.9 stehen Ihnen die beiden OPC UA Built-in Datentypen "LocalizedText" und "ByteString" für ein Mapping auf entsprechende SIMATIC-Datenstrukturen zur Verfügung. Zur Definition dieser OPC UA Datentypen siehe auch OPC 10000-3 DataType Definitions.

Diese Datentypen werden z. B. in Companion Spezifikationen verwendet und sind mit dem OPC UA Schnittstellen-Editor leicht für das Anwenderprogramm zu handhaben.

LocalizedText

Eine Struktur, die eine Zeichenfolge (String) mit Gebietsschemabezeichner (Locale Identifier, z. B. 'en-US') enthält.

Die Struktur hat 3 Elemente mit festgelegter Reihenfolge und ist bei SIMATIC folgendermaßen aufgebaut:

- **Encoding** (Datentyp OPC_UA_LocalizedTextEncodingMask): Zeigt im Bit 0 an, ob das Feld "Locale" einen Inhalt hat und im Bit 1 an, ob das Feld "Text" einen Inhalt hat. Beide Felder sollten einen Inhalt haben, daher empfehlen wir den Wert von "Encoding" für SIMATIC auf 2#00000011 zu setzen.
- **Locale** (Datentyp WString): Gebietsschema, z. B. 'en-US'.
- **Text** (Datentyp WString): Textfeld, z. B. 'Text'.

ByteString
Eine Folge von Oktetts.
Die Struktur ist folgendermaßen aufgebaut:

- **ActualLength** (Datentyp "OPC_UA_ByteStringActualLength"): Länge des ByteString Arrays, das befüllt ist
- **ByteString** (Datentyp "Array of Byte"): Byte-Array

Voraussetzung

Eine OPC UA-Server-Schnittstelle ist erstellt.

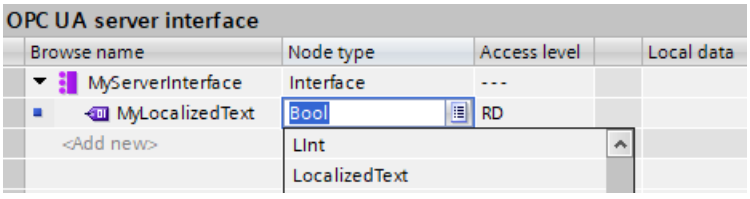
Anwendung

Sie können eine Companion-Spezifikation oder einen Referenz-Namensraum importieren, die Definitionen vom Typ "LocalizedText" bzw. "ByteString"enthalten.
Ebenso können Sie eine Server-Schnittstelle erstellen und selbst ein Adressmodell mit den Datentypen "LocalizedText" bzw. "ByteString" definieren. Die Vorgehensweise ist im nächsten Abschnitt beschrieben.

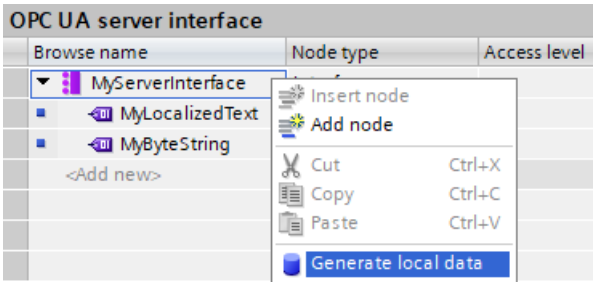
Vorgehensweise

Im Folgenden ist beschrieben, wie Sie mit dem Schnittstellen-Editor einen Knoten vom Typ "LocalizedText" bzw. "ByteString" erzeugen und anschließend automatisch eine SIMATIC-Datenstruktur für diesen Knoten erstellen lassen.
Um OPC UA-Knoten vom Typ "Localized Text"/"ByteString" in einer Server-Schnittstelle zu definieren, gehen Sie folgendermaßen vor:

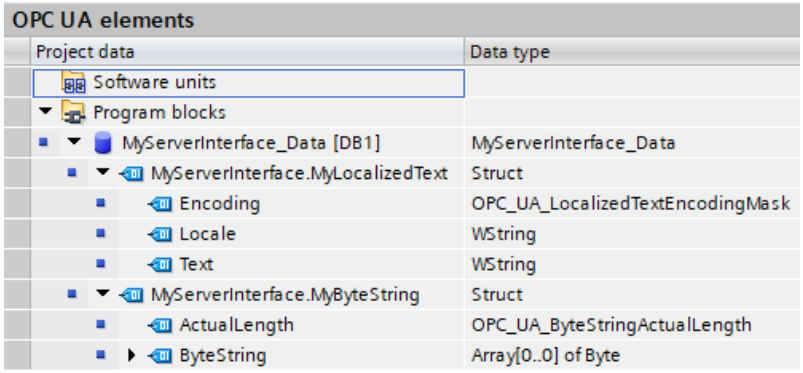
1. Erzeugen Sie im Bereich "OPC UA-Server-Schnittstelle" Knoten vom Typ "LocalizedText" bzw. "ByteString". Diese Knotentypen sind in der Liste der wählbaren Knotentypen enthalten.



2. Wählen Sie im Kontextmenü den Befehl "Lokaldaten erzeugen". Wählen Sie für die Erzeugung der Lokaldaten einen Datenbaustein, z. B. einen neuen DB mit Namen "MyServerInterface_Data".



Ergebnis: STEP 7 erzeugt die passende Struktur für das Mapping, in der Sie noch für "LocalizedText" die benötigte Textlänge (Text) und das erforderliche Gebietsschema (Locale) anpassen müssen.
Für "ByteString" gilt entsprechendes; hier müssen Sie die Länge und das Array anpassen.
Der Konsistenzcheck erzeugt eine Warnung, um auf die erforderlichen Anpassungen hinzuweisen.



Regeln

- Sie können für die Knotentypen "LocalizedText" bzw. "ByteString" auch UDTs mit entsprechender Struktur wie oben gezeigt anlegen und für DB-Elemente verwenden.
- Sie können die Knotentypen "LocalizedText" bzw. "ByteString" in anderen Strukturen verwenden (Schachteln).
- Die SIMATIC-Strukturen für "LocalizedText" bzw. "ByteString" dürfen nur komplett verwendet werden; ein isolierter Datentyp z. B. "OPC_UA_LocalizedTextEncodingMask" für andere Zwecke ist nicht vorgesehen.
- Ein- und Ausgangsparameter von Methoden können auch vom Datentyp/Knotentyp "LocalizedText" bzw. "ByteString" sein.

11.3.4.7 Weitere OPC UA Datentypen für Companion Spezifikationen nutzen

Neben den im Abschnitt "Mapping von Datentypen" aufgeführten OPC UA-Datentypen und deren Entsprechungen auf SIMATIC-Seite gibt es noch folgende OPC UA-Basisdatentypen, die Sie ebenfalls nutzen können:

- OpcUa_NodeId
- OpcUa_QualifiedName
- OpcUa_Guid
- OpcUa_XmlElement
- OpcUa_ByteString (Seite 276)
- OpcUa_LocalizedText (Seite 276)

Voraussetzung für die Nutzung der oben genannten Basisdatentypen als Variablen im Anwenderprogramm: Die Basisdatentypen müssen als zusammengesetzte Datentypen vorliegen, die genauso strukturiert sind wie die entsprechenden OPC UA-Basisdatentypen.

- OpcUa_NodeId und OpcUa_QualifiedName liegen als Systemdatentyp vor; deshalb können Sie diese Datentypen für einzelne Variablen aber auch als Elemente einer Struktur nutzen.
- Für die Basisdatentypen bzw. Built-in Datentypen GUID und XmlElement müssen Sie einen PLC-Datentyp entsprechend der OPC UA-Spezifikation anlegen und anschließend als Element in einer Struktur verwenden, damit die Datentypen der Elemente aufgelöst werden können. Wie der PLC-Datentyp jeweils aussehen muss, ist im Folgenden für jeden einzelnen Basisdatentyp beschrieben.
- Für OpcUa_ByteString und OpcUa_LocalizedText wurden im TIA Portal V17 die Voraussetzungen geschaffen, diese Datentypen einfach in der Server-Schnittstelle vom Typ "Companion Spezifikation" zu nutzen:
  - Sie erzeugen in der Server-Schnittstelle den entsprechenden Knotentyp (z. B. OpcUa_LocalizedText)
  - Sie klicken auf "Lokaldaten erzeugen"

  STEP 7 erzeugt dann automatisch die passenden Datenstrukturen in einem DB.
- Für OpcUa_Guid wurden diese Voraussetzungen im TIA Portal V19 geschaffen. Gehen Sie vor wie im vorhergehenden Abschnitt beschrieben.

Systemdatentyp "OPC_UA_NodeId"

Für den OPC UA Basisdatentyp "OpcUa_NodeId" entnehmen Sie der folgenden Tabelle die Bedeutung der Parameter. OPC_UA_NodeId nutzen Sie zur Identifizierung eines Knotens im OPC UA-Server.

Parameter	S7-Datentyp	Bedeutung
NamespaceIndex	UINT	Namensraumindex des Knotens im OPC UA-Server. Ein Knoten kann zum Beispiel eine Variable sein.
Identifier	WSTRING[254]	Die Bezeichnung für den Knoten (Objekt oder Variable) ist abhängig vom Identifier-Typ: • Numerischer Identifier: Der Knoten wird mit einer Zahl bezeichnet, zum Beispiel "12345678". • String-Identifier: Der Knoten wird mit einem Namen bezeichnet, zum Beispiel "MeineVariable". Groß- und Kleinschreibung wird unterschieden.

Parameter	S7-Datentyp	Bedeutung
IdentifierType	UDINT	Typ des Identifiers • 0: Numerischer Identifier • 1: String-Identifier • 2: GUID • 3: Opaque

Systemdatentyp "OPC_UA_QualifiedName"

Entnehmen Sie der folgenden Tabelle den Aufbau des Systemdatentyps "OPC_UA_QualifiedName":

Name	S7-Datentyp	Bedeutung
NamespaceIndex	UINT	Der Namespaceindex des Namens.
Name	WSTRING[64]	Name des Knotens oder der Variablen.

Systemdatentyp "GUID"

Der Datentyp "Guid" steht ab TIA Portal V19 als Knotentyp zur Verfügung. Zur Definition dieses OPC UA Datentyps siehe auch OPC 10000-6 Mappings.

Das Bild zeigt die Zuweisung des Datentyps "Guid" für einen Variablen-Knoten in einer Server-Schnittstelle.

Das nächste Bild zeigt den automatisch "Lokaldaten erzeugen" angelegten Datenbaustein mit den GUID-Elementen.

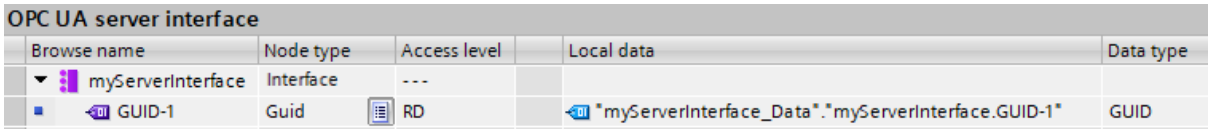


Bild 11-51 GUID-Knoten in der Server-Schnittstelle

Das nächste Bild zeigt den automatisch "Lokaldaten erzeugen" angelegten Datenbaustein mit den GUID-Elementen.

Name	Data type	Start value
Static		
myServerInterface.GUID-1	GUID	
Data1	UDInt	16#C4965784
Data2	UInt	16#0DFE
Data3	UInt	16#4B8F
Data4	Array[0..7] of Byte	
Data4[0]	Byte	16#87
Data4[1]	Byte	16#0A
Data4[2]	Byte	16#74
Data4[3]	Byte	16#52
Data4[4]	Byte	16#38
Data4[5]	Byte	16#C6
Data4[6]	Byte	16#AE
Data4[7]	Byte	16#AE

Bild 11-52 Datenbaustein mit SDT GUID

UDT für Basisdatentyp XmlElement

Ein XmlElement ist ein serialisiertes XML-Fragment (UTF-8-String).
Legen Sie für den Basisdatentyp "XmlElement" folgenden PLC-Datentyp an:

XmlElement

	Name	Data type	Default value
1	XmlElement	WString	WSTRING#''

11.3.4.8 Regeln für OPC UA-XML-Dateien

Import von exportieren OPC UA-XML-Dateien in eine S7-1500 CPU

Beachten Sie folgenden Hinweis, wenn Sie Server-Schnittstellen importieren, die aus dem OPC UA-XML-Export einer S7-1500 stammen.

HINWEIS

Gesperrter Import für Namensraum "http://www.siemens.com/simatic-s7-opcua"

Sie können keine Server-Schnittstelle mit dem Namensraum "http://www.siemens.com/simatic-s7-opcua" in eine S7-1500 CPU importieren, da dieser Namensraum für S7-1500 CPUs reserviert (Standard-SIMATIC Server-Schnittstelle) und für den Import gesperrt ist.

Wenn Sie eine Server-Schnittstelle mit dem Namensraum "http://www.siemens.com/simatic-s7-opcua" importieren wollen, dann öffnen Sie die zu importierende Server-Schnittstelle (OPC UA-XML-Datei) und ändern den Namensraum an den entsprechenden Stellen. Die so geänderte Datei können Sie dann importieren.

Integrität der OPC UA-XML-Dateien

OPC UA-XML-Dateien repräsentieren den Server-Adressraum. Diese Dateien werden z. B. im Kontext von OPC UA Companion Spezifikationen von Ihnen nach Anpassung an die Applikation als Server-Schnittstelle importiert, in die S7-1500 CPU geladen und getestet.

⚠ WARNUNG

Keine Prüfung von importierten OPC UA-XML-Dateien

Schützen Sie diese OPC UA-XML-Dateien vor nicht autorisierten Manipulationen, da STEP 7 die Integrität dieser Dateien nicht prüft.

Empfehlung

Um bei einer Erweiterung bzw. Anpassung des Server-Adressraums Risiken zu minimieren, gehen Sie folgendermaßen vor:

1. Schützen Sie das Projekt (Projektnavigation: Security-Einstellungen > Einstellungen).
2. Exportieren Sie vor der Erweiterung oder Anpassung die entsprechende Server-Schnittstelle.
3. Überarbeiten Sie diese OPC UA-XML-Datei.
4. Importieren Sie die Datei erneut als Server-Schnittstelle.

11.3.4.9 Server-Schnittstelle für Referenz-Namensraum anlegen

Companion Spezifikationen und referenzierte Namensräume

In einer Companion Spezifikation sind eine Reihe von OPC UA-Objekttypen (sowie weitere Definitionen) festgelegt. Diese Objekttypen sind jeweils in Namensräumen definiert, damit die Namen der Objekttypen (Typ-Definitionen) eindeutig sind.

Um eine Companion Spezifikation in Ihrem Projekt zu verwenden, erzeugen Sie Instanzen von Objekttypen dieser Companion Spezifikation.

Dafür müssen die Objekt-Definitionen in Ihrem STEP 7-Projekt vorliegen. Falls das nicht der Fall ist, müssen Sie die Objekt-Definitionen importieren. Um alle Definitionen eines Namenraums zu importieren, legen Sie in STEP 7 für jeden Namensraum eine Server-Schnittstelle des Typs "Referenz-Namensraum" an.

HINWEIS

EUROMAP und die OPC Foundation haben die Joint Working Group "OPC UA Plastics and Rubber Machinery" etabliert

Die bestehenden EUROMAP Empfehlungen EUROMAP 77 (data exchange between injection moulding machines and MES), 82.1 (temperature control devices) und 83 (general definitions) sind veröffentlicht worden unter dem neutralen Schirm der OPC Foundation als OPC 40077, 40082-1 and 40083. Die im Folgenden aufgeführten Beispiele verwenden jedoch die bisher gültigen Bezeichnungen und Verweise.

Beispiel Euromap 77 (aktuell OPC 40077)

Sie haben für die Companion Spezifikation Euromap 77 (aktuell OPC 40077) eine Server-Schnittstelle hinzugefügt.
Die Server-Schnittstelle verwendet Objekttypen, die in OPC UA DI sowie in Euromap 83 und Euromap 77 in ihren entsprechenden Namensräumen definiert sind.
Deshalb legen Sie in STEP 7 neben der Server-Schnittstelle Euromap 77 vom Typ "Companion Spezifikation" weitere Server-Schnittstellen vom Typ "Referenz-Namensraum" an, jeweils für folgende Namensräume:

- http://opcfoundation.org/UA/DI/
- http://www.euromap.org/euromap83/
- http://www.euromap.org/euromap77/

Die folgende Beschreibung zeigt, wie Sie dazu vorgehen.

Server-Schnittstelle für einen Referenz-Namensraum anlegen

Um eine Server-Schnittstelle für einen Namensraum zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie die CPU aus, die Sie als OPC UA-Server nutzen.
2. Klicken Sie auf "OPC UA-Kommunikation > Server-Schnittstellen".
3. Doppelklicken Sie auf "Neue Server-Schnittstelle hinzufügen".
   STEP 7 (TIA Portal) zeigt nun den Dialog "Neue Server-Schnittstelle hinzufügen" an.
   Im Dialog ist ein allgemeiner Name für die neue Server-Schnittstelle eingetragen, zum Beispiel "Server-Schnittstelle_1".
4. Vergeben Sie einen aussagekräftigen Namen für die neue Server-Schnittstelle.
   Im Beispiel wählen Sie den Namen "Opc.Ua.Di" oder einen ähnlichen Namen, der eindeutig auf den Namensraum "http://opcfoundation.org/UA/DI/" hinweist.
   Dieser Namensraum muss als Erster importiert werden. Er enthält grundlegende Definitionen (zum Beispiel den UAObjectType "DeviceType").
5. Klicken Sie auf die Schaltfläche "Companion Spezifikation" und wählen Sie den Typ "Referenz-Namensraum".
6. Klicken Sie auf die 3 Punkte neben dem Feld "XML-Datei importieren".
7. Wählen Sie eine XML-Datei aus, die die Definitionen des Namensraums "http://opcfoundation.org/UA/DI/" enthält.
   Im Beispiel wählen Sie die Datei "Opc.Ua.Di.NodeSet2.xml" aus. Diese Datei können Sie hier laden:
   Opc.Ua.Di.NodeSet2.xml (https://opcfoundation.org/UA/schemas/DI/)
   Hinweis: Wenn die Datei nicht importiert werden kann, liegt das möglicherweise daran, dass die im TIA Portal vorhandene OPC UA ("CORE") Model Version nicht zur OPC UA for Devices ("DI") Model Version passt. Wählen Sie in diesem Fall eine andere Version vom DI Model (z. B. eine Vorgänger-Version).

Das folgende Bild zeigt den Dialog mit den Einträgen:

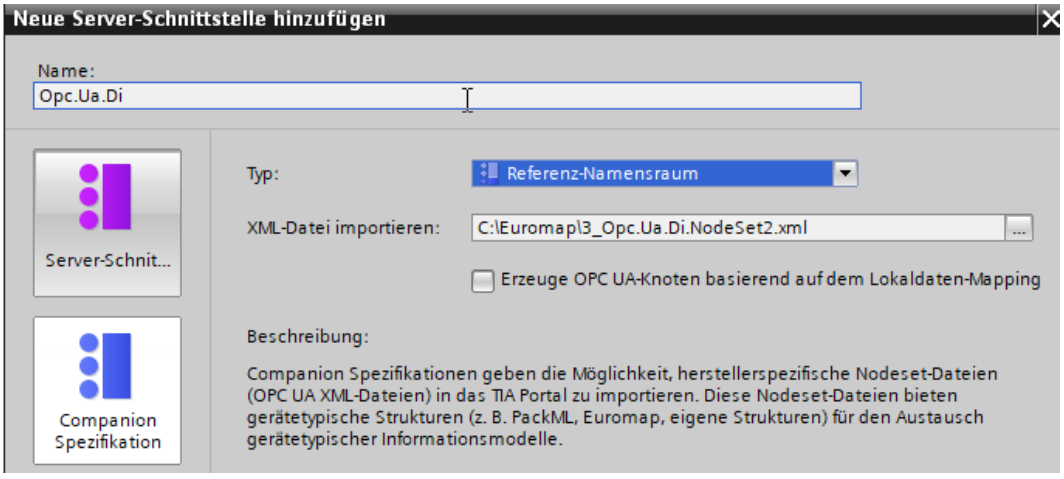


Bild 11-53 Referenz-Namensraum hinzufügen

8. Klicken Sie auf die Schaltfläche "OK".
   STEP 7 (TIA) erzeugt nun die neue Server-Schnittstelle.

Sie finden die Server-Schnittstelle in der Projektnavigation von STEP 7 (TIA Portal), unter "OPC UA-Kommunikation > Server-Schnittstellen > Referenz-Namensraum".
Falls eine Companion Spezifikation weitere Namensräume verwendet, dann fügen Sie jeweils eine neue Server-Schnittstelle für jeden Namensraum hinzu.

Weitere Server-Schnittstellen für die Euromap77 hinzufügen

Für die Euromap 77 benötigen Sie noch die folgenden Namensräume:

- http://www.euromap.org/euromap83/
- http://www.euromap.org/euromap77/

Fügen Sie zunächst eine Server-Schnittstelle für den Namensraum "http://www.euromap.org/euromap83/" hinzu.
Dieser Namenraum enthält grundlegende Definitionen für die Euromap 77, deshalb wird er hier zunächst benötigt. Alle Definitionen dieses Namensraums sind in der XML-Datei "Opc_Ua.EUROMAP83.NodeSet2.xml" enthalten, die Sie von der Website der Euromap (https://www.euromap.org/en/euromap83) laden können.
Fügen Sie anschließend eine Server-Schnittstelle für den Namensraum "http://www.euromap.org/euromap77" hinzu. Alle Definitionen dieses Namensraums sind in der XML-Datei "Opc_Ua.EUROMAP77.NodeSet2.xml" enthalten, die Sie ebenfalls von der Website der Euromap (https://www.euromap.org/en/euromap77) laden können.

11.3.4.10 OPC UA-Knoten erzeugen basierend auf Lokaldaten-Mappings von FB-Typen und UDTs

Wenn Sie Instanzdaten von FBs oder UDTs der CPU für OPC UA Clients zugänglich machen wollen, haben Sie die Möglichkeit, ab TIA Portal Version V17 diese Instanzdaten-Zuweisungen automatisch vornehmen zu lassen.

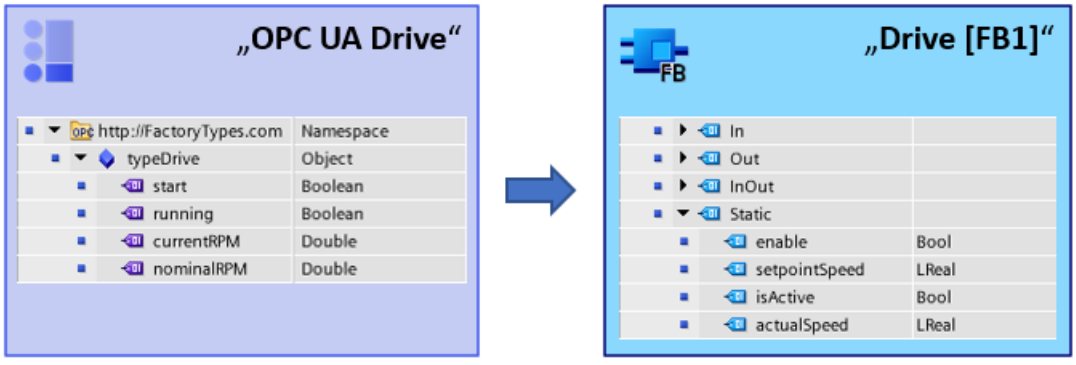
Sie müssen lediglich die FB-Typen bzw. die UDTs auf geeignete OPC UA-Datentypen von importierten Referenz-Namensräumen mappen. Basierend auf diesen Mappings erzeugt STEP 7 (TIA Portal) dann beim Übersetzen für jede FB-Instanz bzw. für jede UDT-Verwendung die erforderlichen Knoten in der Server-Schnittstelle.

Wenn Sie Ihr Programm erweitern und weitere FB-Instanzen bzw. UDT-Verwendungen ergänzen oder bestehende Instanzen löschen, brauchen Sie sich nicht um die Anpassung der Server-Schnittstelle zu kümmern: STEP 7 passt die Server-Schnittstelle automatisch beim Übersetzen des Programms an.

Beispiel

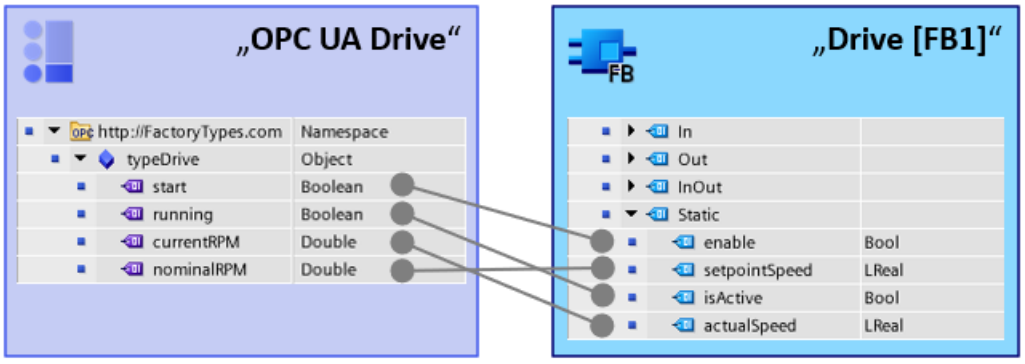
- Sie erstellen einen Funktionsbaustein (FB) im Anwenderprogramm der CPU und legen im "Static"-Bereich der Schnittstelle des FBs die Parameter fest, die das "Gedächtnis" des FBs bilden. Die Instanzen (Werte) dieser Parameter sollen für OPC UA-Clients zugreifbar sein.
- Sie erstellen einen OPC UA-Datentyp (z. B. mit SiOME) mit Elementen, die vom Datentyp den Parametern im Static-Bereich der Schnittstelle des FBs entsprechen. Die Reihenfolge der Elemente spielt keine Rolle. Anschließend importieren Sie die Referenz-Nodeset-Datei als Referenz-Namensraum.

Das folgende Bild zeigt die Zuordnung der Elemente als Gegenüberstellung der Referenz-Namensraum-Sicht (Server-Schnittstelle) und der OPC UA-Elemente-Sicht (Programm).



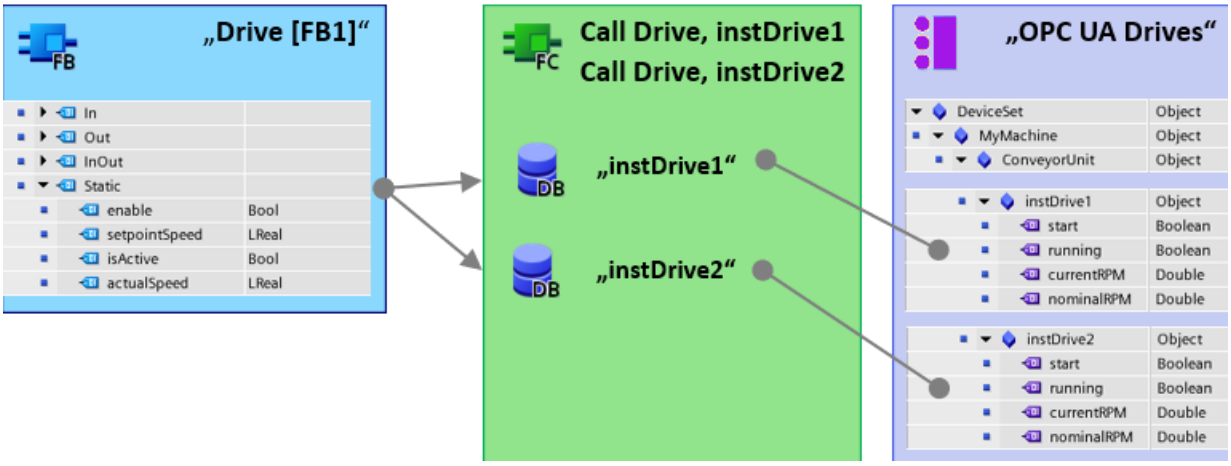
Mapping der Datentypen (FB-Schnittstelle - OPC UA-Schnittstelle): Prinzip

Das folgende Bild zeigt die Zuordnung der Elemente aus dem Anwenderprogramm der CPU zu den Elementen der OPC UA-Server-Schnittstelle. Die Reihenfolge der Elemente muss nicht übereinstimmen.

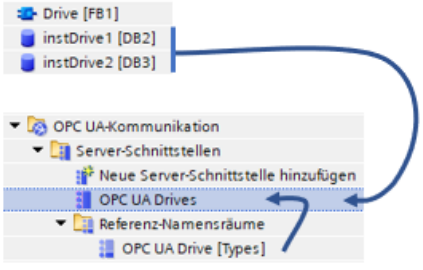


Automatische Erzeugung der OPC UA-Server-Instanzen in der Server-Schnittstelle: Prinzip

Das folgende Bild zeigt das Ergebnis der Übersetzung des Projekts: Die Instanzen des Anwenderprogramms werden auch in der Server-Schnittstelle erzeugt.



Durch das Mapping zwischen FB-Typinformation / UDT-Typinformation und OPC UA-Typinformationen ist STEP 7 in der Lage, alle im Programm vorhandenen Instanzen als Knoten in der Server-Schnittstelle zu erzeugen.



Regeln

- Nur die FB-Elemente im "Static"-Bereich einer FB-Schnittstellen lassen sich auf OPC UA-Typbeschreibungen mappen.
- Beim Mappen der Datentypen müssen immer OPC UA-Elemente aus derselben FB-Schnittstelle bzw. aus demselben UDT für ein Objekt gewählt werden. Mappen aus unterschiedlichen FBs oder UDTs auf ein Objekt ist nicht erlaubt.

Voraussetzung

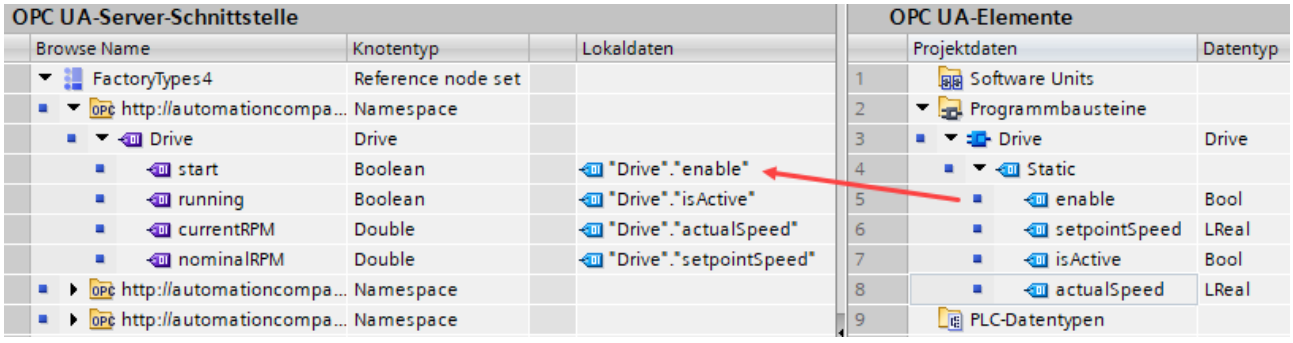
- Die verwendeten FB-Typen, definiert im "Static"-Bereich eines FBs, müssen als "Erreichbar für OPC UA" projektiert sein
- Die verwendeten UDTs müssen als "Erreichbar für OPC UA" projektiert sein
- Eine Nodeset-Datei (XML-Datei) liegt vor mit OPC UA Datentyp-Definitionen, die zu den im Anwenderprogramm definierten FB-Typen bzw. UDTs passen (sich mappen lassen). Nutzen Sie das Tool "SiOME" zur Erzeugung Ihrer Nodeset-Datei (Siemens Industry Online Support).
- Das Anwenderprogramm mit den FB-Instanzen und UDT-Verwendungen liegt vor.

Vorgehensweise

Um einen Datentyp aus einem Referenz-Namensraum auf einen FB-Typ oder UDT-Datentyp zu mappen, gehen Sie folgendermaßen vor:

1. Markieren Sie die CPU, die Sie als OPC UA-Server nutzen.
2. Importieren Sie die vorbereitete Nodeset-Datei (XML-Datei) mit den Typdefinitionen als Referenz-Namensraum.
   – Aktivieren Sie im Dialog "Neue Server-Schnittstelle hinzufügen" die Option "Erzeuge OPC UA-Knoten basierend auf dem Lokaldaten-Mapping".
   Nur wenn diese Option aktiviert ist, lassen sich FB-Typen bzw. UDTs per Drag & Drop auf OPC UA-Typbeschreibungen mappen.
3. Doppelklicken Sie auf das Symbol für die soeben erzeugte Server-Schnittstelle vom Typ "Referenz-Namensraum".
   Der Editor für das Mapping zwischen OPC UA-Server-Schnittstelle und OPC UA-Elementen wird geöffnet. Im Eigenschaften-Bereich des Editors, Bereich "Mapping der lokalen Daten", ist die Option "Erzeuge OPC UA-Knoten basierend auf dem Lokaldaten-Mapping" aktiviert. Wenn nicht, dann aktivieren Sie die Option.
   Passen Sie im Feld "Schnittstellenname" den Namen der zu erzeugenden Server-Schnittstelle an. Beim Übersetzen wird eine neue Server-Schnittstelle vom Typ "Companion Spezifikation" mit diesem Namen erzeugt.

4. Weisen Sie den Knoten der Server-Schnittstelle (Referenz-Namensraum) die bestehenden FB-Typen bzw. UDTs zu, indem Sie das OPC UA-Element (rechte Seite des Editors) per Drag & Drop auf den entsprechenden Knoten der Server-Schnittstelle (Referenz-Namensraum, Spalte "Lokaldaten") ziehen.



5. Übersetzen Sie das Projekt.
   Nach dem Übersetzen befinden sich die neu erzeugten Knoten der Instanzen in der erzeugten Server-Schnittstelle vom Typ "Companion Spezifikation". Für jeden Instanz-DB erzeugt STEP 7 ein Objekt. Die generierten Elemente befinden sich jeweils unter diesen Objekten.
   Ähnlich dazu erzeugt STEP 7 für jeden Global-DB, der beim Instanziieren eines UDTs erzeugt wird, ebenfalls ein Objekt.
   Diese erzeugte Companion-Spezifikation-Schnittstelle kann nicht geändert werden (nicht mehr möglich: Lokaldaten erzeugen, Companion Spezifikation importieren). Diese Maßnahme schützt vor Datenverlust für den Fall, dass die Server-Schnittstelle manuell erweitert und folglich bei einem erneuten Übersetzen überschrieben wurde. Lediglich der Name der erzeugten Companion-Spezifikation-Schnittstelle kann angepasst werden.
   Außerdem können Gruppenordner, die Sie zur Strukturierung Ihrer Daten in der Projektnavigation Ihres TIA Projekts erstellt haben, in der erzeugten Companion-Spezifikation-Schnittstelle als "Folder" (Knotentyp) übernommen werden.

Anwenderprogramm mit FB-Typen bzw. UDT erstellen

Wie Sie FBs und UDTs erstellen, ist hier nicht im Detail erklärt; dazu nutzen Sie die Beschreibung zur Erstellung eines Anwenderprogramms, z. B. Bausteinschnittstelle deklarieren und PLC-Datentypen (UDT) deklarieren.

Konsistenzcheck

Der Konsistenzcheck (Schaltfläche "Konsistenzcheck" des Editors) prüft auch das Mapping der Datentypen und aktualisiert die Anzeige der Datentypen in der entsprechenden Spalte des Editors.

11.3.4.11 Hinweise zu Mengengerüsten bei Nutzung von Server-Schnittstellen

Wenn Sie OPC UA-Server-Schnittstellen verwenden, dann müssen Sie abhängig von Leistungsklasse der S7-1500 CPU Obergrenzen für folgende Objekte berücksichtigen:

- Anzahl der Server-Schnittstellen
- Anzahl der OPC UA-Knoten (Nodes)
- Datenmenge der Ladeobjekte
- Falls Sie Methoden implementiert haben: Anzahl Server-Methoden bzw. Server-Methoden-Instanzen

Mengengerüste für OPC UA-Server-Schnittstellen und Methoden

In folgender Tabelle sind die Mengengerüste der S7-1500 CPUs dokumentiert, die auch beim Übersetzen und Laden einer Konfiguration berücksichtigt werden (tagesaktuelle Technische Daten der CPUs finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/ps/td)).
Eine Verletzung der Mengengerüste wird mit einer Fehlermeldung quittiert.

Tabelle 11-4 Mengengerüste für OPC UA-Server-Schnittstellen

Technisches Datum	**CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)**	**CPU 1505 (S/SP/SP F/SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)**	**CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)**
Nutzung von importierten Companion Spezifikationen (Informationsmodelle)			
Maximale Anzahl OPC UA-Server-Schnittstellen: • Typ "Companion-Spezifikation" • Typ "Referenz-Namensraum" • Typ "Server-Schnittstelle"	 10 20 10	 10 20 10	 10 20 10
• Maximale Anzahl von OPC UA Knoten (Nodes) in benutzerdefinierten Server-Schnittstellen	1000	5000	30000
• Maximale Größe ladbarer OPC UA-Server-Schnittstellen	1024 KB	5120 KB	8192 KB
Bereitstellung von Methoden			
Maximale Anzahl nutzbarer Server-Methoden bzw. max. Anzahl Server-Methoden-Instanzen (Anweisungen OPC_UA_ServerMethodPre, OPC_UA_ServerMethodPost)	20	50	100

11.3.5 Methoden auf dem OPC UA-Server bereitstellen

11.3.5.1 Wissenswertes zu Server-Methoden

Anwenderprogramm für Server-Methoden bereitstellen

Auf dem OPC UA-Server einer S7-1500 CPU (ab Firmware V2.5) haben Sie die Möglichkeit, Methoden über Ihr Anwenderprogramm bereitzustellen. Diese Methoden können von OPC UA-Clients genutzt werden, um z. B. einen Fertigungsauftrag über den Methodenaufruf von der S7-1500 CPU zu starten.

OPC UA-Methoden, eine Realisierung von "Remote Procedure Calls", bieten einen effizienten Mechanismus für die Interaktionen zwischen verschiedenen Kommunikationsteilnehmern. Der Mechanismus liefert sowohl eine Auftragsbestätigung als auch Rückgabewerte, sodass Sie auf die Ausprogrammierung von Handshaking-Mechanismen verzichten können.

Mit OPC UA-Methoden können Sie z. B. Daten konsistent ohne Triggerbits/Handshaking übertragen oder bestimmte Aktionen auf der Steuerung auslösen.

Wie funktioniert eine OPC UA-Methode?

Eine OPC UA-Methode funktioniert im Prinzip wie ein Knowhow geschützter Funktionsbaustein, der von einem externen OPC UA-Client zur Laufzeit aufgerufen wird.

Der OPC UA-Client "sieht" lediglich die definierten Ein- und Ausgänge. Das Innere des Funktionsbausteins, die Methode oder der Algorithmus, bleibt dem externen OPC UA-Client verborgen. Der OPC UA-Client bekommt eine Rückmeldung über die erfolgreiche Ausführung und Rückgabewerte, die der Funktionsbaustein (Methode) liefert. Oder eine Fehlermeldung bei nicht erfolgreicher Ausführung.

Sie haben als Programmierer die vollständige Kontrolle und Verantwortung darüber, in welchem Programmkontext die OPC UA-Methode abläuft.

Regeln für die Programmierung einer Methode und Laufzeitverhalten

- Sorgen Sie dafür, dass die über die OPC UA-Methode gelieferten Rückgabewerte konsistent zu den vom OPC UA-Client gelieferten Eingabewerten sind.
- Berücksichtigen Sie die Regeln zur Namensvergabe und Aufbau von Parametern sowie die verwendbaren Datentypen (siehe Beschreibung der OPC UA-Server-Anweisungen).
- Verhalten zur Laufzeit: Der OPC UA-Server nimmt **einen** Aufruf pro Instanz an. Erst wenn dieser Aufruf vom Anwenderprogramm abgearbeitet wurde oder Timeout hatte, ist diese Methoden-Instanz wieder für andere OPC UA-Clients aufrufbar.

Im Folgenden wird die prinzipielle Vorgehensweise gezeigt, mit der Sie ein Anwenderprogramm als Server-Methode implementieren.

Implementierung einer Server-Methode

Ein Programm (Funktionsbaustein) zur Implementierung einer Server-Methode hat folgenden Aufbau:

1. **Aufruf der Server-Methode abfragen mit OPC_UA_ServerMethodPre**
   In Ihrem Anwenderprogramm (d. h. in Ihrer Server-Methode) rufen Sie zunächst die Anweisung "OPC_UA_ServerMethodPre" auf.
   Diese Anweisung hat folgende Aufgaben:
   - Mit dieser Anweisung fragen Sie beim OPC UA-Server der CPU nach, ob Ihre Server-Methode von einem OPC UA-Client aufgerufen wurde.
   - Wenn die Methode aufgerufen wurde und die Server-Methode über Eingangsparameter verfügt, dann erhält Ihre Server-Methode nun die Eingangsparameter.
     Die Eingangsparameter der Server-Methode stammen vom aufrufenden OPC UA-Client.
2. **Server-Methode bearbeiten**
   In diesem Abschnitt der Server-Methode stellen Sie das eigentliche Anwenderprogramm zur Verfügung.
   Sie haben die gleichen Möglichkeiten wie in anderen Anwenderprogrammen (zum Beispiel Zugriff auf andere Funktionsbausteine oder auf globale Datenbausteine).
   Wenn die Server-Methode Eingangsparameter verwendet, stehen Ihnen die Eingangsparameter der Server-Methode zur Verfügung.
   Dieser Abschnitt der Server-Methode sollte nur dann ausgeführt werden, wenn ein OPC UA-Client die Server-Methode aufgerufen hat.
   Nach der erfolgreichen Ausführung der Methode setzen Sie die Ausgangsparameter der Server-Methode, falls die Server-Methode Ausgangsparameter besitzt.
3. **Server-Methode beantworten mit OPC_UA_ServerMethodPost**
   Um die Server-Methode abzuschließen, rufen Sie die Anweisung "OPC_UA_ServerMethodPost" auf.
   Geben Sie der Anweisung "OPC_UA_ServerMethodPost" über die Parameter die Information mit, ob das Anwenderprogramm abgearbeitet ist oder nicht.
   Wenn das Anwenderprogramm erfolgreich ausgeführt wurde, wird der OPC UA-Server über die einsprechenden Parameter informiert. Der OPC UA-Server sendet nun die Ausgangsparameter der Server-Methode an den OPC UA-Client.

Rufen Sie die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" immer pärchenweise auf, unabhängig davon, ob das Anwenderprogramm zwischen den beiden Anweisungen abgearbeitet oder im nächsten Zyklus fortgesetzt wird.

Ein Beispiel für die Implementierung einer Server-Methode finden Sie in der Onlinehilfe von STEP 7.

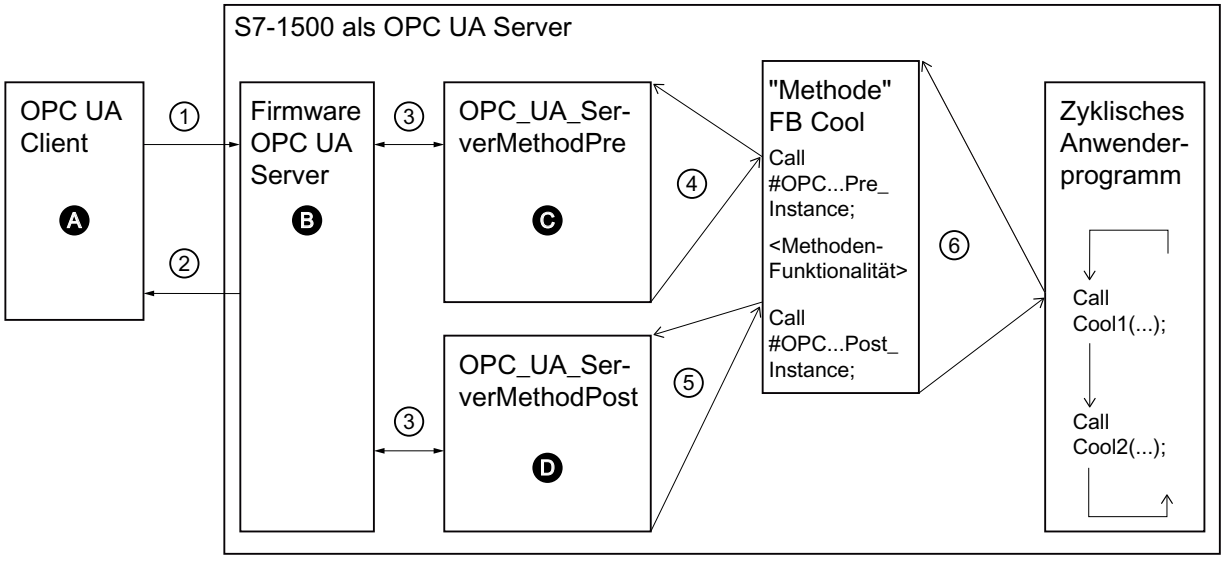
Einbindung der Server-Methode

Die folgende Grafik zeigt, wie ein OPC UA-Client (A) die Server-Methode "Cool" aufruft:
Die CPU führt im zyklischen Anwenderprogramm die Instanz "Cool1" der Server-Methode "Cool" aus ⑥.

Die CPU fragt zunächst mit der Anweisung "OPC_UA_ServerMethodPre" nach ④, ob ein OPC UA-Client die Server-Methode "Cool" aufgerufen hat ①.

- Wenn die Server-Methode nicht aufgerufen wurde, kehrt die Programmausführung über ④ und ⑥ direkt zum zyklischen Anwenderprogramm zurück. Die CPU setzt das zyklische Anwenderprogramm nach "Cool1" fort.
- Wenn die Server-Methode aufgerufen wurde, gelangt diese Information über ④ zurück zur Server-Methode Cool. In der Server-Methode Cool wird nun die eigentliche Funktionalität ausgeführt, siehe "<Methoden-Funktionalität>" in der Grafik.
  Anschließend teilt die Server-Methode über die Anweisung "OPC_UA_ServerMethodPost" ⑤ der Firmware (B) mit, dass die Anweisung ausgeführt wurde ③.

  Die Firmware sendet diese Information über ② an den aufrufenden OPC UA-Client (A) zurück.
  Die CPU setzt das zyklische Anwenderprogramm nach "Cool1" fort.



A Aufruf der Server-Methode und Management der "Done"-Information (Methode beendet)

① Asynchroner Aufruf der Server-Methode

② Asynchrone "Done"-Information der aufgerufenen Methode (Methode beendet)

B Warten auf OPC UA-Client-Aufrufe, Management von Aufrufen in der Warteschlange, "Done"-Information aus dem zyklischen Anwenderprogramm an den OPC UA-Client weiterleiten

③ Datentransfer vom OPC UA-Server zur Methoden-Instanz des Anwenderprogramms und umgekehrt

C Prüfen, ob Methode aufgerufen wurde.
Wenn ja, dann Eingangsdaten vom OPC UA-Server zur Methoden-Instanz des Anwenderprogramms weiterreichen und der Methoden-Instanz zurückmelden, dass die Methode aufgerufen wurde ("called")

④ Synchroner Aufruf der Anweisung OPC_UA_ServerMethodPre als Multiinstanz mit Angabe des Speicherbereichs für die Eingangsdaten vom OPC UA-Server.
Der Return-Value informiert, ob die Methode vom OPC UA-Client aufgerufen worden ist.

⑤ Prüfen, ob die Methode beendet wurde oder noch aktiv ist ("busy").

D Prüfen, ob die Methode beendet wurde.
Wenn ja, werden die Ausgangsdaten der Methoden-Instanz zum OPC UA Server weitergeleitet und der Methoden-Instanz zurückgemeldet, dass die Methode beendet ist. Der OPC UA-Server wird darüber informiert.

⑥ Aufruf des Methoden-FB (hier: FB Cool) mit der gewünschten Instanz und den Prozessparametern

Bild 11-54 Beispiel: Aufruf der Server-Methode "Cool"

Informationen zu den Server-Anweisungen

Die Anweisungen "OPC_UA_ServerMethodPre" und "OPC_UA_ServerMethodPost" sind detailliert beschrieben in der Hilfe zu den Anweisungen > Kommunikation > OPC UA > OPC UA-Server.

11.3.5.2 Randbedingungen zum Einsatz von Server-Methoden

Zulässige Datentypen

Wenn Sie Server-Methoden bereitstellen, dann achten Sie auf folgende Regel:

- Ordnen Sie die Datentypen zu wie unten gezeigt (SIMATIC-Datentyp - OPC UA-Datentyp). Andere Zuordnungen sind nicht zugelassen.

STEP 7 prüft nicht die Einhaltung dieser Regel und verhindert nicht eine falsche Zuordnung. Sie sind für die regelkonforme Auswahl und Zuordnung der Datentypen verantwortlich.
Die aufgelisteten Datentypen können Sie auch z. B. als Elemente von Strukturen/Arrays/UDTs für Eingangs- und Ausgangsparameter von selbst erstellten Server-Methoden (UAMethod_InParameters und UAMethod_OutParameters) verwenden.

SIMATIC-Datentyp	OPC UA-Datentyp
BOOL	Boolean
SINT	SByte
INT	Int16
DINT	Int32
LINT	Int64
USINT	Byte
UINT	UInt16
UDINT	UInt32
ULINT	UInt64
REAL	Float
LREAL	Double
LDT	DateTime

SIMATIC-Datentyp	OPC UA-Datentyp
WSTRING	String
DINT	Enumeration (Encoding Int32) und alle davon abgeleiteten Datentypen
Anwenderdefinierter Datentyp erforderlich (UDT, user-defined data type) Der Anwenderdefinierte Datentyp muss mit dem Prefix "Union_" angelegt werden, z. B. "Union_MyDatatype". Das erste Element (Selector) in diesem UDT muss den Datentyp "UDINT" besitzen.	UNION und alle davon abgeleiteten Datentypen

Anzahl implementierbarer Server-Methoden und Anzahl der Argumente

Wenn Sie Server-Methoden über Ihr Anwenderprogramm implementieren, dann ist die Anzahl nutzbarer Methoden je nach CPU-Typ begrenzt, siehe folgende Tabelle (tagesaktuelle Technischen Daten der CPUs finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/ps/td)).

Technisches Datum	CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)	CPU 1505 (S/SP/SP F/SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)	CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)
Maximale Anzahl nutzbarer Server-Methoden bzw. max. Anzahl Server-Methoden-Instanzen (Anweisungen OPC_UA_ServerMethodPre, OPC_UA_ServerMethodPost)	20	50	100
Maximale Anzahl Argumente pro Methode (Mehr als die angegebene Anzahl Argumente ist projektierbar und ladbar in die CPU, allerdings kann ein OPC UA-Client die Methode dann nicht aufrufen.)	20	20	20

Fehlermeldung bei Überschreitung

Wenn die maximale Anzahl von Server-Methoden überschritten ist, melden die Anweisungen OPC_UA_ServerMethodPre bzw. OPC_UA_ServerMethodPost den Fehlercode 0xB080_B000 (TooManyMethods).

Versorgung von strukturierten Datentypen mit geschachtelten Arrays

Wenn ein strukturierter Datentyp (Struct/UDT) ein Array enthält, stellt der OPC UA-Server keine Information über die Länge dieses Arrays bereit.

Falls Sie eine solche Struktur z. B. als Eingangs- oder Ausgangsparameter einer Server-Methode verwenden, dann müssen Sie sicherstellen, dass das geschachtelte Array beim Aufruf der Methode mit der richtigen Länge versorgt wird.

Wenn Sie diese Regel nicht berücksichtigen, schlägt die Methode fehl mit dem Fehlercode "BadInvalidArgument".

11.3.6 Meldungen auf dem OPC UA-Server bereitstellen

11.3.6.1 Wissenswertes zu Meldungen

Meldungen ermöglichen Ihnen, Fehler bei der Prozessbearbeitung in den Automatisierungssystemen schnell zu erkennen, genau zu lokalisieren und zu beheben. Stillstandszeiten einer Anlage können so wesentlich verkürzt werden. Eine standardisierte und plattformunabhängige Möglichkeit der Meldungsverarbeitung ist mit dem OPC UA-Informationsmodell "Alarms & Conditions" gegeben.

Ab Firmware-Version V2.9 unterstützt der OPC UA-Server einer S7-1500 CPU das OPC UA-Informationsmodell "Alarms and Conditions". Damit bietet der OPC UA-Server Zugriff auf Steuerungsmeldungen.

Die folgenden Abschnitte beschreiben, welche in der SIMATIC verfügbaren Meldungstypen an der OPC UA-Schnittstelle des OPC UA-Servers unterstützt werden.

In den darauf folgenden Kapiteln ist weiterhin beschrieben, wie Sie den OPC UA-Server der S7-1500 CPU für Alarms and Conditions konfigurieren, wie das Alarms-and-Conditions-Modell bei OPC UA in Grundzügen strukturiert ist und welche Besonderheiten bei Verwendung von Meldungen aus dem Adressraum des OPC UA-Servers im Unterschied zu den SIMATIC-Steuerungsmeldungen des CPU-Meldesystems zu berücksichtigen sind.

Basis für die Umsetzung von Meldungen nach OPC UA Alarms & Conditions

Spezifiziert ist das Informationsmodell Alarms and Conditions in der Spezifikation "OPC 10000-9: OPC Unified Architecture Part 9: Alarms & Conditions".

Steuerungsmeldungen in der SIMATIC

Der OPC UA-Server der S7-1500 CPU unterstützt die unten aufgeführten Steuerungsmeldungen, die S7-1500 CPUs zur Verfügung stehen. Sie projektieren bzw. programmieren diese Meldungen wie gewohnt, ohne zusätzliche Regeln für die Nutzung dieser Meldungen durch OPC UA-Clients berücksichtigen zu müssen.

Der zusätzliche Nutzen von OPC UA Alarms and Conditions besteht darin, dass diese Meldungsarten nicht nur von HMI-Geräten, von Webbrowsern, vom CPU-Display oder vom TIA Portal angezeigt werden können, sondern auch von allen OPC UA-Clients, die OPC UA Alarms and Conditions unterstützen.

- PLC-Überwachungsmeldungen mit ProDiag
  Mit wenigen Projektierungsschritten integrieren Sie einfache Überwachungen in Ihr Programm, ohne dabei den Programmcode zu verändern. Die Projektierung der Überwachungen ist unabhängig von den Programmiersprachen des TIA Portals, da nur einzelne Operanden überwacht werden und Sie keine weiteren Programmierabschnitte benötigen.
- Systemdiagnosemeldungen
  Konfigurationsabhängige Baugruppenereignisse sind durch die Hardware-Konfiguration der CPU bekannt und sind über angeschlossene Anzeigegeräte auswertbar. Sie können im Meldungseditor nur angesehen, aber nicht bearbeitet werden.
- Programmmeldungen (Anweisung Program_Alarm)
  Zum Melden von programmsynchronen Ereignissen sind Programmmeldungen jeweils einem Baustein zugeordnet. Sie werden im Programmiereditor erstellt und im Meldungseditor (TIA Portal) bearbeitet.
- GRAPH-Meldungen
  Für GRAPH-Funktionsbausteine können Sie ebenfalls Meldungen aktivieren; z. B. für Interlocks, Supervisionen und GRAPH-Warnungen (Schrittzeit-Überwachungen).

Wesentliche Informationen zu den Meldungsarten

Wesentlich für die Unterschiede im Verhalten von Meldungen sind folgende Merkmale:

- Haben Meldungen einen Zustand (z. B. sind sie gekommen, gegangen - mit den entsprechenden Zeitstempeln)?
- Sind Meldungen quittierpflichtig?

Wenn keines dieser Merkmale zutrifft, d. h. die Meldungen sind zustandslos und nicht quittierpflichtig, dann informieren Meldungen einfach ein aufgetretenes Ereignis ("Fire and Forget"). Es bleibt dem Meldungs-empfangenden Gerät überlassen, die Meldung für eine weitere Verwendung oder auch nur zur Anzeige zwischenzuspeichern.

Meldeklasse bestimmt Quittierverhalten

In diesem Abschnitt geht es um die Einstellungsmöglichkeiten bei Programmmeldungen. Für Systemdiagnosemeldungen und PLC-Überwachungsmeldungen können Sie ebenfalls das Verhalten der Meldungen einstellen (z. B. ProDiag-Überwachungseinstellungen) - Details lesen Sie in den verlinkten weiterführenden Informationen.

Die Einstellungsmöglichkeiten für Programmmeldungen finden Sie im Meldungseditor (Doppelklick auf "PLC-Überwachungen und Meldungen" in der Projektnavigation, dann Register "Meldungen" wählen).

Für die S7-1500 CPU stellen Sie hier über die Meldungsklasse ein, ob eine Meldung quittiert werden muss oder nicht. Zusätzlich zum Quittierverhalten legen Sie beim Anlegen einer neuen Meldungsklasse die Default-Priorität der Meldungen dieser Meldeklasse fest.

Ob eine Meldung zustandsbehaftet ist oder nicht, stellen Sie über die Option "nur Information" z. B. am Meldungstyp ein; damit erreichen Sie ein "Fire-and-Forget-Verhalten" der Meldung.

Hier ein Beispiel mit Einstellungen im Meldungseditor mit unterschiedlichen Meldeklassen ("PLC-Überwachungen und Meldungen" in der Projektnavigation):

- Erste Zeile "Program_Alarm": nicht quittierpflichtig, nur Information ("Fire and Forget").
- Zweite Zeile "Program_Alarm_1": quittierpflichtig und zustandsbehaftet, d. h. mit der Meldung wird angezeigt, ob sie gekommen oder gegangen ist.
- Dritte Zeile "Program_Alarm_2": nicht quittierpflichtig und zustandsbehaftet, d. h. mit der Meldung wird angezeigt, ob sie gekommen oder gegangen ist.

Meldungstypen

Name	Typ	ID	Ort	Meldetext	Infotext	Meldeklasse	Quittierung	Nur Information
Program_Alarm	PLC-Meldung		AlarmType	myText	myINFO	No Acknowledgement	☐	☑
Program_Alarm_1	PLC-Meldung		AlarmType			Acknowledgement	☑	☐
Program_Alarm_2	PLC-Meldung		AlarmType			No Acknowledgement	☐	☐

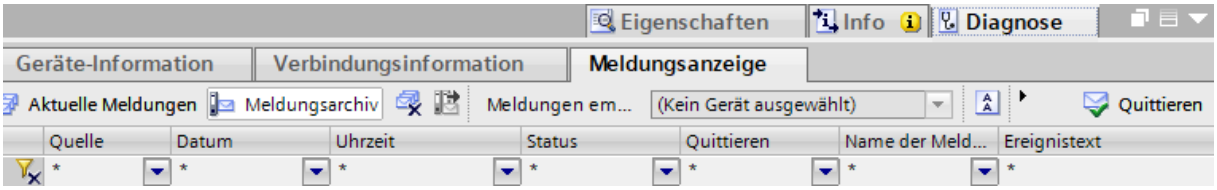
Anzeige von Meldungen im TIA Portal

Zur Laufzeit haben Sie die Möglichkeit, sich die Meldungen im TIA Portal anzeigen zu lassen: Die Meldungsanzeige befindet sich direkt unter dem Meldungseditor (Register "Diagnose" > Register "Meldungsanzeige").

Bezüglich des Zustands- und Quittierverhaltens gilt Folgendes:

- Wenn Sie auf die Schaltfläche "Aktuelle Meldungen" klicken, werden Meldungen angezeigt, die aktuell gekommen, gegangen oder quittiert wurden. Nur zustandsbehaftete Meldungen und quittierpflichtige Meldungen werden hier angezeigt. Quittieren können Sie eine quittierpflichtige Meldung (blaue Schrift) ebenfalls in dieser Ansicht über Kontextmenü oder mithilfe der Schaltfläche "Quittieren".
- Wenn Sie den zeitlichen Verlauf beobachten wollen (z. B. Meldung kam, ist quittiert worden und ist dann gegangen), müssen Sie auf die Schaltfläche "Meldungsarchiv" klicken. Nur in dieser Ansicht sind die drei Ereignisse, die zu dieser Meldung gehören, nacheinander aufgeführt. In der Ansicht "Aktuelle Meldungen" finden Sie nur den jeweils aktuellen Zustand.
- Info-Reports (Meldungen mit der Eigenschaft "Nur Information") werden nur in der Ansicht "Meldungsarchiv" angezeigt. Da diese Meldungen nur einmalig getriggert und nicht zwischengespeichert werden, tauchen sie nicht in der Ansicht "Aktuelle Meldungen" auf.
- PLC-Überwachungen werden ebenfalls in der Meldungsanzeige angezeigt.

- Systemmeldungen gehören i. d. R. zur Meldungsklasse "No Acknowledgement" mit der Option "Nur Information". Diese Meldungen werden in der CPU im Diagnosepuffer geloggt und erlauben damit eine Analyse der Abfolge von Systemmeldungen über einen begrenzten Zeitraum. Betriebszustandsänderungen, die zwar auch im Diagnosepuffer geloggt werden, sind dagegen zustandsbehaftet, d. h. es wird z. B. angezeigt, dass bzw. wann eine CPU in den Betriebszustand STOP gegangen ist und wenn bzw. wann sie diesen Zustand wieder verlassen hat, d. h. in den Betriebszustand RUN gegangen ist. Diese Informationen werden mit den Zuständen "gekommen/gegangen" angezeigt.



Bereitstellung von Steuerungsmeldungen durch den OPC UA-Server

Wenn ein OPC UA-Client Meldungen der S7-1500 CPU empfangen will, dann muss er sich auf OPC UA Events subscriben (MonitoredEventItems).

Dazu befinden sich im Adressraum des OPC UA-Servers der S7-1500 CPU entsprechende Knoten, die "Event-Notifier" sind und für die OPC UA-Clients eine Subscription anlegen, um die Meldungen empfangen zu können.

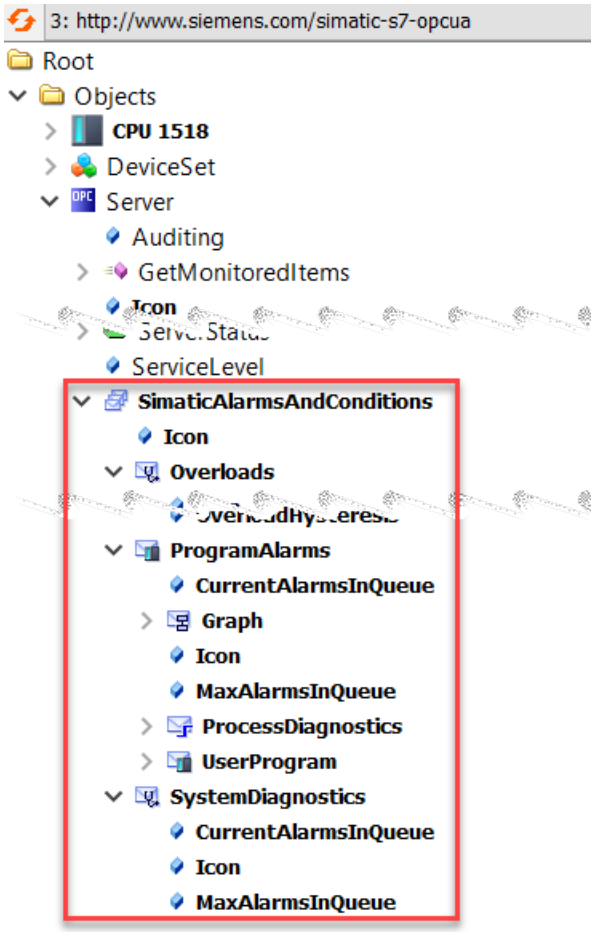
Der Vollständigkeit halber sei noch erwähnt, dass dazu im Server-Adressraum noch weitere Typ-Definitionen für diesen Zweck unter "Types" als Knoten eingetragen sind.
Typ-Definitionen unter "BaseEventType" und "ConditionType" stellen sicher, dass die von SIMATIC-Meldungen verwendeten Felder auch im OPC UA-Server bereitgestellt werden.

Der OPC UA-Adressraum der S7-1500 CPU spiegelt damit nach Aktivierung von OPC UA Alarms and Conditions (CPU-Eigenschaften in der Hardware-Konfiguration) die verschiedenen Meldungsarten (Steuerungsmeldungen) wie oben beschrieben:

- ProcessDiagnostics
  Entspricht den PLC-Überwachungsmeldungen mit ProDiag
- SystemDiagnostics
  Entspricht den Systemdiagnosemeldungen
- UserProgram
  Entspricht den Programmmeldungen
- Graph
  Entspricht den GRAPH-Meldungen

Über die Auswahl des Knotens für eine Subscription bestimmen Sie, welche Meldungsarten vom OPC UA-Client empfangen werden. Der Knoten "Server" z. B. ermöglicht den Empfang aller Meldungen, der Knoten "UserProgram" nur den Empfang von Programmmeldungen.

Einzelheiten zum OPC UA-Modell für Alarms and Conditions finden Sie im nächsten Kapitel, speziell zum Knoten "Overloads" finden Sie weitere Informationen hier: Speichergrenzen für OPC UA Alarms and Conditions hantieren (Seite 314).



Weitere Informationen zu den Meldungsarten

Die Konzepte und Projektierungsmöglichkeiten für Steuerungsmeldungen werden hier nicht weiter erläutert. In der Onlinehilfe zu STEP 7 finden Sie Informationen zu Meldungsprojektierung, Meldungsanzeige und den dazugehörigen Anweisungen, z. B. "Progam_Alarm".

11.3.6.2 OPC UA Events

Im Folgenden werden die Grundkonzepte für die Meldungsverarbeitung bei OPC UA vertieft - hier geht es um das Grundkonzept der "Events". Dabei sind an dieser Stelle die englischen Benennungen, wie sie in den verschiedenen Teilen der OPC UA-Spezifikation verwendet werden, beibehalten worden.

Eigenschaften von Events

Im Adressmodell des OPC UA-Servers haben Sie ab CPU-Firmware-Version V2.9 nicht nur die Möglichkeit, über Knoten auf PLC-Variablen zuzugreifen (lesen, schreiben) und Methoden zu nutzen - Sie können auch über Knoten Ereignisse bzw. Meldungen erhalten. Im Sprachgebrauch von OPC UA sind das Events.

Ein Event enthält u. A. einen Ereignistext (Message), Zeitstempel (Time) und Event-Quelle (SourceNode).

Die Informationen, die mit einem Event vom Server geliefert werden, hängen ab vom Eventtyp. OPC UA definiert im Part 5 der Spezifikation (Information Model) einen BaseEventType.

Weitere Eventtypen, die unterschiedliches Meldeverhalten bereitstellen, sind vom BaseEventType abgeleitet. Diese Typinformationen von den verschiedenen Eventtypen sind im Adressraum eines OPC UA-Servers sichtbar (Ordner "Types"). Das gilt z. B. auch für die Eventtypen von "Conditions" und "Alarms", die im nächsten Kapitel behandelt werden.

Welche Eigenschaften (Felder) eines Events obligatorisch (mandatory) sind und welche Ereignisse optional sind, legt die OPC UA-Spezifikation für den BaseEventType und für abgeleitete EventTypes fest.

Das folgende Bild zeigt die hierarchische Struktur von BaseEventType.

Die folgenden Abschnitte zeigen, wie von der Wurzel der Ableitungshierarchie, dem BaseEventType, spezialisierte EventTypes abgeleitet sind. Die SIMATIC-spezifischen Ableitungen sorgen dafür, dass die Informationen, die in der SIMATIC mit einer Meldung mitgeliefert und z. B. auf einem HMI-Gerät angezeigt werden, auch im Adressraum des OPC UA-Servers von einem OPC UA-Client abonniert werden können.

Ein Event selbst ist nicht als Knoten im Adressraum verfügbar. Events werden nur von Knoten bzw. Objekten ausgelöst, die die Eigenschaft "Event-Notifier" haben. Diese Knoten werden oft auch Eventmeldeobjekte genannt. Nur Knoten mit dieser Eigenschaft können in einer Subscription als EventMonitoredItem angegeben werden, um im Client zugehörige Events zu empfangen.

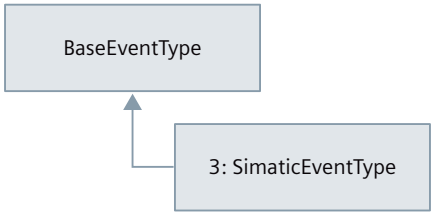
Knoten, die bei einer S7-1500 CPU Events auslösen können, sind Objekte wie "Server", darunter das Objekt "SimaticAlarmsAndConditions" und die drei darunter liegenden Objekte ProcessDiagnostics, SystemDiagnostics und UserProgram. Bei diesen Objekten im Adressraum des OPC UA-Servers der CPU ist das Attribut "EventNotifier" gesetzt.

Definition von SimaticEventType

Das folgende Bild zeigt, dass der Typ "SimaticEventType" direkt von BaseEventType abgeleitet ist.

BaseEventType ist die grundlegende Typdefinition für Events bei OPC UA.

Von BaseEventType ausgehend werden - direkt oder indirekt - alle Event-Typen für OPC UA definiert.



Der Typ "SimaticEventType" ist im SIMATIC-Namensraum definiert (http://www.siemens.com/simatic-s7-opcua).

SimaticEventType besitzt alle Eigenschaften von BaseEventType sowie die speziellen Eigenschaften, die ein Abbild der Feldstruktur von SIMATIC-Meldungen sind.

Beschreibung der Eventfelder für SimaticEventType

Die folgende Tabelle enthält Informationen zu den Feldern von SimaticEventType für Meldungen vom Typ "Nur Information". Felder, die laut OPC UA optional sind und nicht vom OPC UA-Server der CPU genutzt werden, wurden weggelassen. Eine allgemeine Beschreibung der Felder finden Sie außerdem in der Spezifikation OPC 10000-5: OPC Unified Architecture, Part 5: Information Model, Release 1.04.

BrowsePath	DataType	Erläuterungen
EventId	ByteString	Eindeutige Event-ID des Events
EventType	NodeId	Knoten-ID des Eventtyps
Time	UtcTime	Zeitstempel des Ereignisses (Event occurrence)
ReceiveTime	UtcTime	Aktueller Zeitstempel, zu dem das OPC UA Event erzeugt wurde.
Message	LocalizedText	Ereignistext
Severity	UInt16	Priorität der Meldung aus der SIMATIC (0..16) gespreizt in einen Bereich 1..1000 bei OPC UA, siehe nachfolgende Tabelle. Die Priorität kennzeichnet die Dringlichkeit, mit der auf das Ereignis zu reagieren ist.
3:AdditionalText_01	LocalizedText	Optionaler Zusatztext 1
...	...	...
3:AdditionalText_09	LocalizedText	Optionaler Zusatztext 9
3:AssociatedValue_01	3:SimaticAssociated-AlarmValue	Optionaler Begleitwert 1 (nicht für Systemdiagnose)
...	...	...
3:AssociatedValue_10	3:SimaticAssociated-AlarmValue	Optionaler Begleitwert 10 (nicht für Systemdiagnose)
3:InfoText	LocalizedText	Infotext

BrowsePath	DataType	Erläuterungen
3:ID	UInt16	Meldungsnummer - CPU-weit eindeutige Nummer (ID), die vom System vergeben wird und eine Meldung identifiziert.
3:DisplayClass	UInt16	Anzeigeklasse (genutzt von HMI-Geräten. Bestimmt, welche Ereignisse auf welchen HMI-Geräten angezeigt werden.
3:GroupID	UInt8	Quittiergruppe für Meldungen, die gemeinsam quittiert werden.

Zuordnung Priorität (SIMATIC) - Severity (OPC UA)

Die folgende Tabelle zeigt, wie die 17 Prioritäten, die Sie Meldungen im SIMATIC-Umfeld zuweisen können, auf die 1000-stufige Severity beim OPC UA-Server der S7-1500 CPU gemappt werden.
Diese Zuordnung ist herstellerspezifisch. Für andere Geräte können andere Zuordnungen gelten.

OPC Range	Priorität 0..16 (SIMATIC)	Severity 1..1000 (OPC UA)
HIGH (667 – 1 000)	16	1000
	15	930
	14	860
	13	790
	12	720
MEDIUM (334 – 666)	11	650
	10	600
	9	550
	8	500
	7	450
	6	400
	5	350
LOW (1 – 333)	4	300
	3	225
	2	150
	1	75
	0	1

11.3.6.3 OPC UA Conditions und OPC UA Alarms

Im Folgenden werden die Grundkonzepte für OPC UA Conditions und für OPC UA Alarms vertieft, basierend auf den Erläuterungen zu Events in den vorhergehenden Abschnitten. Dabei sind an dieser Stelle wieder die englischen Benennungen, wie sie in den verschiedenen Teilen der OPC UA-Spezifikation verwendet werden, beibehalten worden.

Eigenschaften von Conditions

Voraussetzung für das Verständnis ist das Konzept der "Events" bei OPC UA.

Wenn ein Eventmeldeobjekt zusätzlich zu seiner Eigenschaft, Events feuern zu können, auch noch Zustandsinformationen bietet, spricht man bei OPC UA von Conditions. Conditions repräsentieren einen Zustand eines Systems oder einer seiner Komponenten. Basiszustände sind "enabled" und "disabled", andere Zustandsdefinitionen sind auch möglich.

Zustandsänderungen werden interessierten OPC UA-Clients wiederum über Events mitgeteilt (Condition Events).

Ein Beispiel für eine Condition ist z. B. die Zustandsinformation, dass ein Gerät eine Wartung benötigt.

Eigenschaften von Alarms

Die Eigenschaften von ConditionType reichen allerdings nicht aus, um die Merkmale von SIMATIC-Meldungen vollständig im OPC UA-Server abzubilden.

Von ConditionType, der von BaseEventType abgeleitet ist, definiert OPC UA weitere abgeleitete Eventtypen wie AcknowledgeableConditionType und AlarmConditionType.

AcknowledgeableConditionType ergänzt die Eigenschaften von ConditionType um das Merkmal "Quittierbar" (AckedState).

AlarmConditionType ergänzt also die Eigenschaften von ConditionType und AcknowledgeableConditionType um das Merkmal, einen "ActiveState" einnehmen zu können. Im SIMATIC-Sprachgebrauch ist das eine gekommene Meldung. Der ActiveState signalisiert, dass die Situation, welche die Condition repräsentiert, aktuell existiert bzw. eingetreten ist.

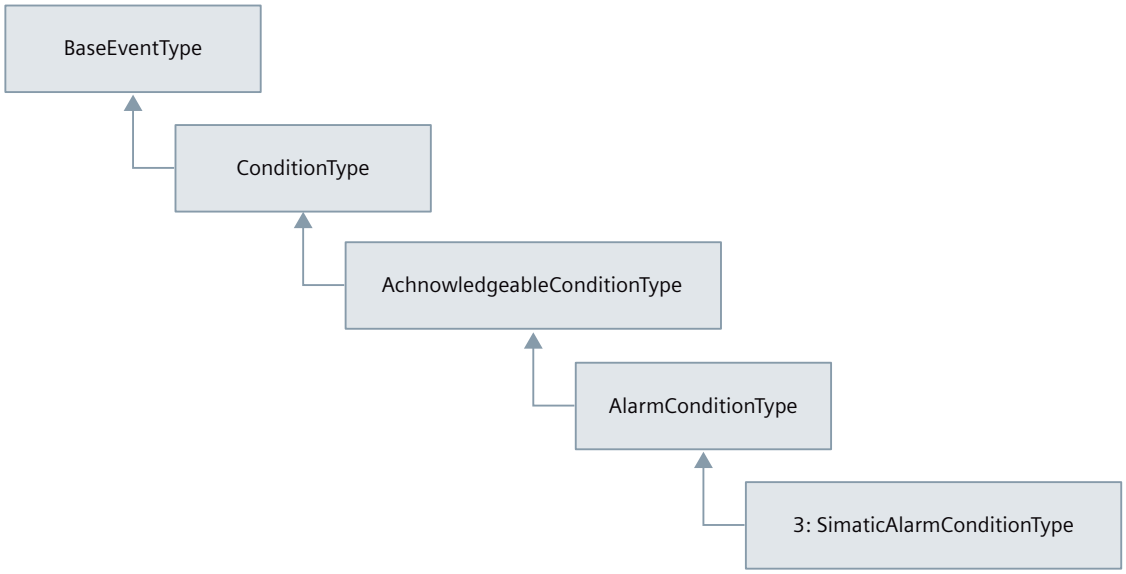
Beispiel: Eine Temperatur hat einen Grenzwert überschritten. Wenn "ActiveState" nicht gesetzt ist, dann existiert die Situation, welche die Condition repräsentiert, gerade nicht mehr - i. d. R. spricht man hier von einem "Normalzustand". Im Sprachgebrauch der SIMATIC entspricht das einer gegangenen Meldung.

Bei OPC UA sind noch weitere Status definiert wie z. B. SilenceState oder ShelvingState, die aber für das Mapping zum SIMATIC-Meldesystem keine Rolle spielen und daher hier nicht näher erläutert werden.

Vom AlarmConditionType ist der SimaticAlarmConditionType abgeleitet, der alle Eventfelder enthält, um die Zustands- und Quittierinformationen von SIMATIC-Meldungen abbilden zu können.

Definition von SimaticAlarmConditionType

Das folgende Bild zeigt, wie Events des Typs "SimaticAlarmConditionType" durch eine Reihe von Erweiterungen zum OPC UA "BaseEventType" definiert sind.



Beschreibung der Eventfelder für SimaticAlarmConditionType

Die folgende Tabelle enthält Informationen zu den Feldern von SimaticAlarmConditionType für zustandsbehaftete und quittierbare Meldungen, die zusätzlich zu den Eventfeldern wie bei SimaticEventType hinzukommen. Felder, die laut OPC UA optional sind und nicht vom OPC UA-Server der CPU genutzt werden, wurden weggelassen. Eine Beschreibung der Felder finden Sie außerdem in der Spezifikation OPC 10000-9: OPC Unified Architecture, Part 9: Alarms & Conditions, Release 1.04.

BrowsePath	DataType	Erläuterungen
ConditionClassId	NodeId	Möglich sind Knoten-IDs für SystemConditionClassType, ProcessConditionClassType bzw. BaseConditionClassType
ConditionClassName	LocalizedText	DisplayName der ConditionClassId
Retain	Boolean	Zeigt an, dass die Meldung noch relevant ist für OPC UA Clients (Gesetzt, falls Meldung noch ansteht und nicht quittiert ist).
Comment	LocalizedText	• Jüngster Kommentar, der über die Methode "AddComment" oder "Acknowledge" eingetragen wurde. • NULL nach einem Server Neustart und wenn kein Kommentar eingetragen wurde.
Comment.SourceTimestamp	UtcTime	Zeitstempel für die letzte Änderung des Kommentarfelds
AckedState	LocalizedText	"Acknowledged" or "Unacknowledged"

BrowsePath	DataType	Erläuterungen
AckedState.Id	Boolean	Gesetzt, wenn quittiert wurde
AckedState.TransitionTime	UtcTime	Zeitpunkt, an dem die Meldung quittiert wurde. NULL wenn nicht quittiert oder nicht quittierbar.
ActiveState	LocalizedText	"Active" oder "Inactive"
ActiveState.Id	Boolean	Gesetzt wenn "Aktive"

11.3.6.4 Alarms and Conditions aktivieren

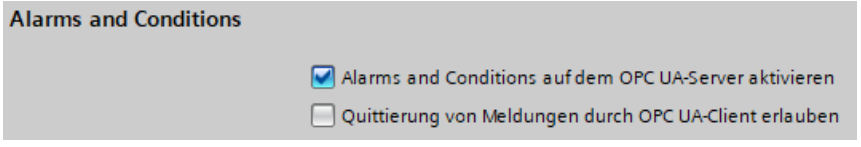
Voraussetzungen

- S7-1500 CPU ab Firmware Version V2.9
- Runtime-Lizenz für OPC UA entsprechend den Lizenzvorgaben erworben und in den CPU-Eigenschaften eingestellt
- Parameter "Zentrale Meldungsverwaltung in der PLC" ist aktiviert (Bereich "PLC-Meldungen" in den CPU-Eigenschaften).

Vorgehen

Um Meldungen über OPC UA Alarms and Conditions zu aktivieren, gehen Sie folgendermaßen vor:

1. Navigieren Sie in den CPU-Eigenschaften zum Bereich "OPC UA > Server > Allgemein".
2. Aktivieren Sie die Option "Alarms and Conditions auf dem OPC UA-Server aktivieren".
   Erst wenn die Option aktiviert ist, werden die entsprechenden Typen und Objekte, die Events auslösen können, sichtbar im Adressraum.
3. Falls erforderlich, aktivieren Sie zusätzlich die Option "Quittierung von Meldungen durch OPC UA-Client erlauben".
   In diesem Fall kann jeder verbundene OPC UA-Client eine quittierpflichtige Meldung mit der Methode "Acknowledge" quittieren.



Empfehlung: Diagnose "Requests eines entfernten OPC UA-Clients fehlgeschlagen" aktivieren

Wenn der OPC UA-Server nicht ausreichend Speicher allokieren kann, dann ist in diesem Zustand kein Erzeugen von OPC UA-Alarmen möglich; ein Meldungsverlust für OPC UA-Clients ist möglich.

Sie sollten daher die Diagnose "Requests eines entfernten OPC UA-Cients fehlgeschlagen" aktivieren, um diesen Zustand diagnostizieren zu können (Eigenschaften der CPU > OPC UA > Server > Diagnose).

Außerdem sollten Sie zusätzlich die Option "Diagnosen bei hoher Meldungszahl zusammenfassen" ebenfalls aktivieren.

Sobald wieder ausreichend Speicher zur Verfügung steht, sollten OPC UA-Clients die ConditionRefresh-Methode aufrufen, um den aktuellen Zustand des Alarmsystems zu erhalten.

Weitere Informationen

Informationen zu Methoden für OPC UA Alarms and Conditions finden Sie im Kapitel Methoden für OPC UA Alarms and Conditions (Seite 310).

Informationen zu fehlgeschlagenen Anfragen eines entfernten Clients finden Sie im Kapitel Anfrage eines entfernten Clients fehlgeschlagen (Seite 322).

11.3.6.5 Events eines OPC UA-Servers abonnieren

Alle Events über Knoten "Server" abonnieren

OPC UA-Server stellen Events über den Knoten "Server" und über unterlagerte Knoten bereit. Wenn OPC UA-Clients den Knoten "Server" abonnieren, dann erhalten sie alle Ereignisse und Meldungen des OPC UA-Servers.

Der Knoten "Server" befindet sich im Ordner "Objects" unterhalb von "Root".

OPC UA-Server informieren OPC UA-Clients darüber, welche Ereignistypen sie verwenden (unter "Root > Types > EventTypes" im Adressraum).

Filtermöglichkeiten auf Events

OPC UA-Clients können eine Auswahl treffen und nur bestimmte Knoten unterhalb des Knotens "Server" und damit bestimmte Eventtypen abonnieren, z. B. nur den Knoten "UserProgram". Dadurch wird die Anzahl der Ereignisse vom OPC UA-Server reduziert auf Programmmeldungen.

Eine weitere Möglichkeit zu filtern ergibt sich durch die Auswahl der Eventfelder, im OPC UA-Sprachgebrauch "Select Clause".

D. h. in der Subscription wird im OPC UA-Client neben dem Eventmeldeobjekt (z. B. der Knoten "UserProgram") auch eine Selektion der Eventfelder vorgenommen. Sie wählen die Eventfelder über den Browse Name des entsprechenden Felds aus.

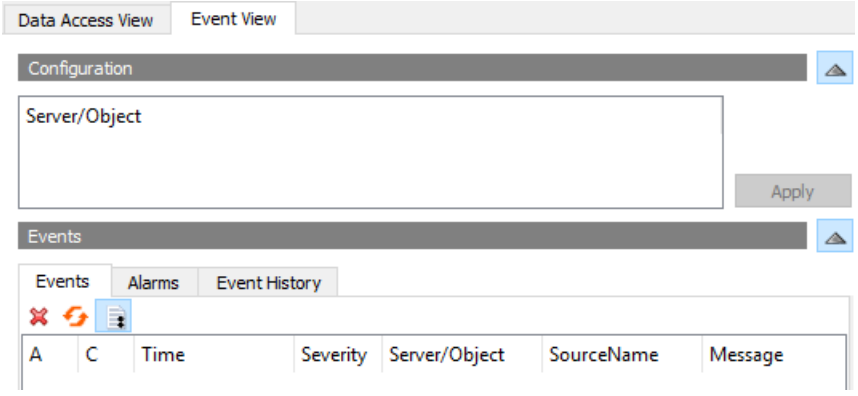
OPC UA definiert daneben noch sogenannte "Where Clauses". Eine Where Clause im Eventfilter wird verwendet, um die Anzahl der Events, die vom OPC UA-Server für das gewählte Objekt geliefert werden, weiter zu beschränken, z. B. durch einen Filter auf einen Severity-Bereich.

Beispiel-Client UaExpert

Am Beispiel des OPC UA-Clients UaExpert lässt sich zeigen, wie Events eines OPC UA Servers über eine Subscription empfangen werden können. Hier die wichtigsten Informationen zu den angezeigten Events/Meldungen:

- Event View ist eine eigene Sicht auf Events neben Data Access View.
- Im Bereich "Configuration" befindet sich das gewählte Eventmeldeobjekt mit den Feldern für die Select Clause. Eine Konfiguration von Where-Clauses ist im UaExpert aktuell nicht möglich.
- Im Bereich "Events", Register "Events": Entspricht der Sicht in der TIA Meldungsanzeige mit aktivierter Schaltfläche "Meldungsarchiv"; dort sind auch Events der Kategorie "Nur Information" und gegangene Meldungen sichtbar, da UaExpert sie im Hintergrund für die Anzeige zwischenspeichert. Im Register "Alarms" sind diese Events nicht sichtbar.
- Im Feld "Events", Register "Alarms": Entspricht der Sicht in der TIA Meldungsanzeige mit aktivierter Schaltfläche "Aktuelle Meldungen"; dort sind Meldungen mit ihrem Status z. B. "active" (entspricht "gekommen") sichtbar und über das Kontextmenü sind diese Meldungen auch quittierbar. Gegangene Meldungen sind in dieser Sicht nicht mehr sichtbar.

In den Spalten des Events-Bereichs wird eine Auswahl der Eventfelder angeboten, z. B. der Ereignistext (Message) und ob die Meldung quittiert wurde (A=Acknowledged).



Besonderheiten der Anzeige von Meldungen über den OPC UA-Server der CPU

Im Folgenden sind die Besonderheiten der Meldungsanzeige über OPC UA Alarms and Conditions für den aktuellen Stand noch einmal zusammengefasst.

Thema	Erläuterung
Kommentar (Comment)	Über OPC UA können Sie mit der Methode "AddComment" oder mit der Methode "Acknowledge"einen Kommentar zu einer Meldung hinzufügen. Dieser steht nach einem Server-Neustart nicht mehr zur Verfügung.
Anstehende Meldungen gehen nach Server-Neustart nicht verloren	Der OPC UA-Server unterstützt die Methode "ConditionRefresh", mit der er z. B. nach dem Laden eines neuen Datenbausteins (erfordert Server-Neustart und erneuten Verbindungsaufbau) den aktuellen Zustand des Systems dem OPC UA-Client zur Verfügung stellt.

11.3.6.6 Begleitwerte von Meldungen verarbeiten

Sie können für SIMATIC-Meldungen Platzhalter spezifizieren. MIt Platzhaltern integrieren Sie bis zu 10 Begleitwerte (SD_1 bis SD_10) in den Meldungstext. Platzhalter können auch spezifische Textlisten-Einträge sein.

Wenn Sie Meldungen mit Platzhaltern nutzen, sind folgende Regeln zu beachten:

- Platzhalter, die Werte in der Meldung repräsentieren, werden nur für Systemdiagnosemeldungen oder Security-Ereignismeldungen automatisch eingefügt. Für andere Kategorien von Meldungen (z. B. Programmmeldungen) werden die Platzhalter für Werte nicht aufgelöst. Diese Auflösungen müssen die OPC UA-Clients leisten.
- Platzhalter, die Textlisten referenzieren, werden durch die CPU aufgelöst (Format z. B: %t#<Name der Textliste>).

Beispiel wie Werte und Platzhalter durch UaExpert zugeordnet werden

1. Stellen Sie sicher, dass alle Felder, die Sie benötigen, in der UaExpert Konfiguration aktiviert sind.
   Beachten Sie, dass jedes Feld, das nicht benötigt wird, Kommunikationslast erzeugt. Vermeiden Sie daher eine Komplett-Selektion wie aus Gründen der Einfachheit im unteren Beispiel gezeigt ist.

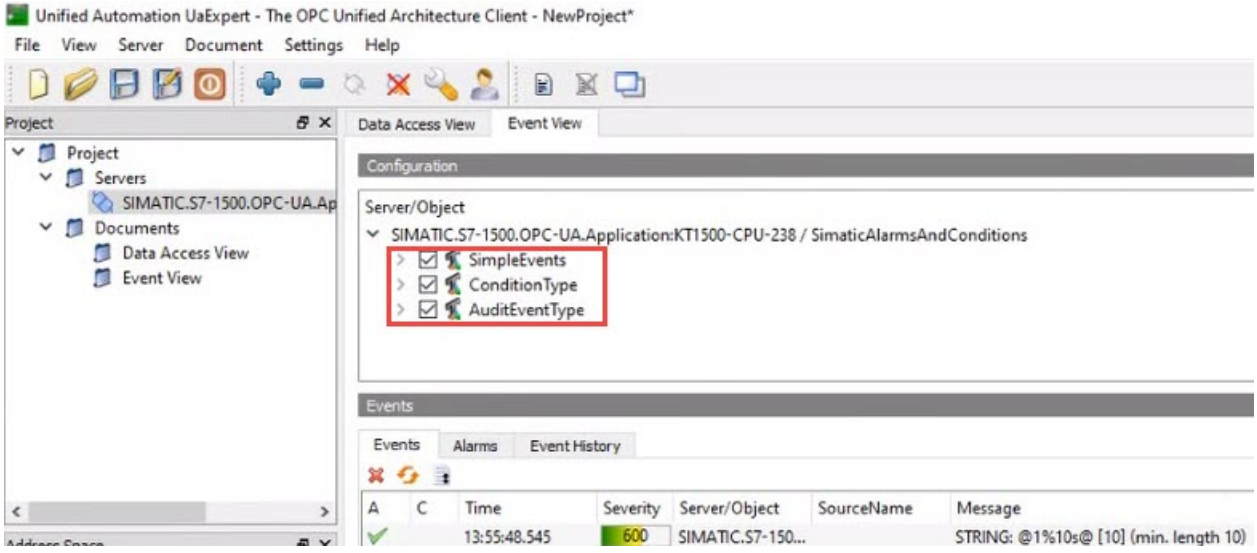


Bild 11-55 ServerObjekct-Configuration_neutral120

2. Im Register "Events" von UaExpert wählen Sie die Meldung mit den integrierten Begleitwerten.
   Im Bereich "Details" der Meldung finden Sie den Wert, der in die Meldung zu integrieren ist.
   Beispiel: "AssociatedValue_01" wird SD_1 zugeordnet (Format @1% ...@).
   Sie finden Erläuterungen zu den Formaten für Begleitwerte im TIA Portal Informationssystem (z. B. durch Suche nach "Beispiel für Begleitwerte").

Unterstützung von einfachen Datentypen als Begleitwerte

Der Feldtyp von "AssociatedValue_01 bis ..._10 ist vom Typ Union und ist beschränkt auf einige einfache Typen. Der OPC UA Datentyp ist SimaticAssociatedAlarmValue. Folgende einfache Datentypen werden unterstützt:

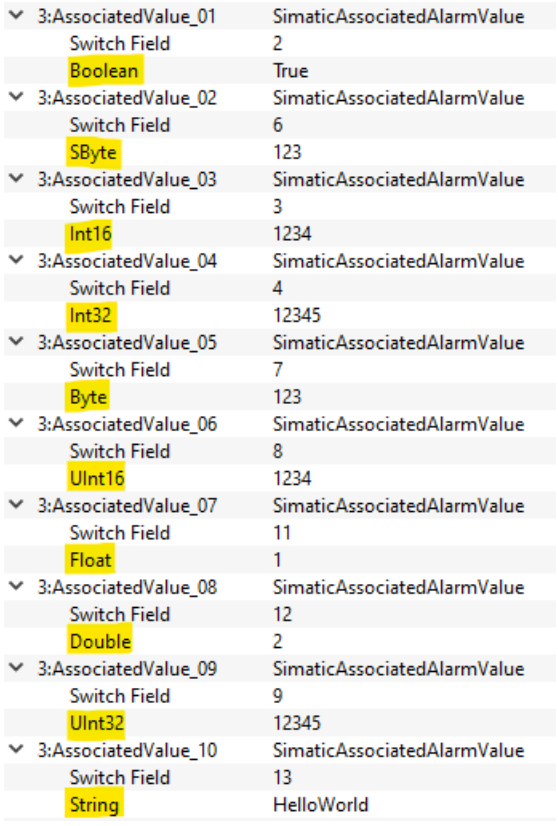


Bild 11-56 UnionDataTypes_neutral120

Mapping auf SIMATIC Datentypen

Folgende Zuordung SIMATIC-Datentyp => OPC UA Datentyp gilt:

Unterstützte Datentypen für SD_1 bis SD_10	Mapping auf OPC UA
BOOL	Boolean
BBOOL	Boolean
BYTE	Byte
CHAR	Byte
SINT	SByte
USINT	Byte
WORD	UInt16
WChar	UInt16
INT	Int16
UINT	UInt16

Unterstützte Datentypen für SD_1 bis SD_10	Mapping auf OPC UA
DWORD	UInt32
DINT	Int32
UDINT	UInt32
REAL	Float
LREAL	Double
String	String
WString	String

11.3.6.7 Methoden für OPC UA Alarms and Conditions

Die OPC UA-Spezifikation Part 9 (OPC 10000-9: Alarms & Conditions) definiert Methoden für OPC UA-Server, um OPC UA-Clients zu ermöglichen, z. B. auf Zustandsänderungen zu reagieren.

Im Folgenden sind die Methoden erläutert, die der OPC UA-Servers der S7-1500 CPU unterstützt mit ihren Besonderheiten.

Voraussetzung

Damit Sie die relevanten Methoden für die Alarms-and-Conditions-Funktionalität nutzen können, sind folgende Voraussetzungen erforderlich:

- Alarms and Conditions ist aktiviert
- Für die Methode "Acknowledge" muss Server-seitig die Quittierung von Meldungen durch OPC UA-Clients erlaubt sein

Methoden für OPC UA Alarms and Conditions

Im Folgenden sind die Methoden in Kurzform erläutert mit den Besonderheiten und Einschränkungen, die durch die Umsetzung für den OPC UA-Server der S7-1500 CPU bedingt sind.

Die Methoden sind im Typraum sichtbar.

Die oben genannte OPC UA-Spezifikation enthält die allgemeine Beschreibung.

Im Anschluss an diese Übersichtstabelle finden Sie die detailliertere Beschreibung der einzelnen Methoden.

Methode	Beschreibung
Acknowledge	Methode zum Quittieren eines Alarm-Objekts, das durch eine EventId eindeutig gekennzeichnet ist.
ConditionRefresh	Methode zum Anfordern einer Aktualisierung aller Alarm-Objekte (im SIMATIC-Sprachgebrauch: Aktualisieren aller anstehenden Meldungen). Alle überwachten Elemente (Monitored Items) der Subscription werden aktualisiert.

Methode	Beschreibung
	Synchronisieren anstehender Alarms-Objekte vom OPC UA-Server der CPU ist z. B. angezeigt bei folgenden Situationen: • Erstmaliges Verbinden oder Wiederaufnahme der Verbindung (nach Unterbrechung der Kommunikation) • Bildwechsel auf einem Bedienbildschirm eines HMI-Geräts
AddComment	Methode zum Hinzufügen von Kommentaren zu Alarm-Objekten.

Aufruf der Methoden "Acknowledge" und "AddComment"

Methodenaufrufe bei OPC UA nutzen MethodId und ObjectId.
Die ObjectId im Fall eines Alarm-Objekts ist die Knoten-Id für die Instanz des Alarm-Objekts.
Da im Adressmodell von Simatic Alarms and Conditions keine Instanzen für Alarm-Objekte vorgesehen sind, sieht die OPC UA Spezifikation in diesem Fall vor, dass der OPC UA-Client die ConditionId als ObjectId verwendet.
Informationen zur Ermittlung der ConditionId mittels eines SimpleAttributeOperands in der SelectClause des Event-Filters finden Sie ebenfalls in der OPC UA-Spezifikation Part 9 (OPC 10000-9: Alarms & Conditions):

Name	Type	Description
SimpleAttributeOperand		
typeId	NodeId	*NodeId* of the *ConditionType Node*
browsePath[]	QualifiedName	empty
attributeId	IntegerId	Id of the *NodeId Attribute*

Acknowledge

Die Acknowledge-Methode (MethodId: i=9111) hat folgende Parameter:

Parameter	Datentyp	Beschreibung
[in] EventId	ByteString	EventId identifiziert eine bestimmte Event Notification. Nur Events, deren AckedState.Id-Feld den Wert "False" hat, können mit der Methode "Acknowledge" quittiert werden.
[in] comment	LocalizedText	Text zur Kommentierung der Quittierung, z. B. durch einen Bediener. Siehe auch ergänzende Beschreibung zur Methode "AddComment".

Method Result Codes

Result Code	Beschreibung
Good	Methode wurde erfolgreich ausgeführt.
BadNotSupported	Methode kann nicht aufgerufen werden, da in den CPU-Eigenschaften für OPC UA die Option zum Quittieren für Alarms and Conditions durch OPC UA-Clients deaktiviert ist.

Result Code	Beschreibung
BadConditionBranchAlreadyAcked	Quittierung wurde bereits durchgeführt.
BadNodeIdUnknown	Methode wurde mit der falschen ConditionId aufgerufen (siehe Anmerkungen zur ObjekctId).
BadEventIdUnknown	Methode wurde mit der falschen EventId aufgerufen.

ConditionRefresh

Die ConditionRefresh-Methode (MethodId: i=3875) hat folgende Parameter:

Parameter	Datentyp	Beschreibung
[in] SubscriptionId	Uint32	SubscriptionId der Subscription, die aktualisiert werden soll.

Method Result Codes

Result Code	Beschreibung
Bad_SubscriptionIdInvalid	Die SubscriptinId ist nicht gültig.
Bad_RefreshInProgress	Die Methode "ConditionRefresh" läuft gerade.
Bad_UserAccessDenied	Die Methode "ConditionRefresh" läuft im Kontext einer falschen Session. D. h. die Subscription gehört zu einer anderen Session.

HINWEIS

Methode ConditionRefresh2

Der OPC UA-Server der S7-1500 CPU unterstützt nicht die Methode ConditionRefresh2, die gezielt ein überwachtes Element (MonitoredItem) in einer Subscription synchronisieren kann. In diesem Fall gibt der OPC UA-Server den Result Code "Bad_MethodInvalid" zurück. Verwenden Sie stattdessen die Methode "ConditionRefresh".

AddComment

Sie haben die Möglichkeit, Kommentare zu Alarms-Objekten vom Typ SimaticAlarmConditionType hinzuzufügen, da die Unterstützung von Kommenaren obligatorisch für OPC UA Alarms and Conditions ist.

Ein Kommentar wird im Eventfeld "Comment" gespeichert.

Folgende Zeitstempel-Eventfelder gehören zum Comment:

- "Comment.SourceTimestamp" für den Zeitpunkt der Übernahme des Kommentars in die CPU
- "Time" für den Änderungszeitpunkt am Alarms-Objekt

Wenn die Methode "AddComment" aufgerufen wird, sind "Time" und "Comment.SourceTimestamp" identisch.

Besonderheiten bei Alarms-and-Conditions-Kommentaren für den OPC UA-Server der CPU

Die Add-Comment-Methode (MethodId: i=9029) hat folgende Parameter:

Parameter	Datentyp	Beschreibung
[in] EventId	ByteString	EventId identifiziert die Event Notification, zu der ein Zustand gemeldet wurde.
[in] comment	LocalizedText	Text zur Kommentierung des spezifizierten Alarms-Objekts.

Method Result Codes

Result Code	Beschreibung
Good	Methode wurde erfolgreich ausgeführt.
BadNodeIdUnknown	Methode wurde mit der falschen ConditionId aufgerufen (siehe Anmerkungen zum Aufruf der Methoden "Acknowlege" und "AddComment").
BadEventIdUnknown	Methode wurde mit der falschen EventId aufgerufen.

Besonderheiten bei Alarms-and-Conditions-Kommentaren für den OPC UA-Server der CPU

Sie haben die Möglichkeit, mit der AddComment-Methode Kommentare zu den Alarm-Objekten vom Typ "SimaticAlarmConditionType" hinzuzufügen. Auch beim Aufruf der Acknowledge-Methode wird ein Kommentar gesetzt. Die Methode "AddComment" kann mehrfach aufgerufen werden.

- Ein Kommentar wird im Eventfeld "Comment" gespeichert. Der "Comment.SourceTimestamp" kennzeichnet den letzten Zeitpunkt, an dem ein **Kommentar** gesetzt wurde.
- Der Zeitstempel "Time" kennzeichnet den **letzten Änderungszeitpunkt am Alarm-Objekt**.

Wenn die Methode "AddComment" aufgerufen wird, sind "Time" und "Comment.SourceTimestamp" identisch.

Wenn die Methode "Acknowledge" aufgerufen wird, können die beiden Zeitstempel voneinander abweichen, da das Quittieren asynchron erfolgt.

Die Unterstützung von Kommentaren für OPC UA Alarms and Conditions ist obligatorisch. Das SIMATIC-Meldungssystem kennt keine entsprechenden Kommentare für Meldungen. Daher sind einige Besonderheiten zu beachten:

- Es gibt nur einen Kommentar:
  Für ein Alarm-Objekt gibt es nur einen Kommentar, so dass bei mehreren Methoden-Aufrufen hintereinander ein bestehender Kommentar immer wieder überschrieben wird.
- Lebensdauer und Zeitstempel:
  Kommentare werden nur am aktuellen Alarm-Objekt gespeichert. Wenn das Alarm-Objekt nicht mehr existiert wie z. B. nach einem Server-Neustart, existiert auch der Kommentar nicht mehr. Die zugehörigen Eventfelder '"Comment" und "Comment.SourceTimestamp" sind dann zurückgesetzt (null).

Das Eventfeld "Time"ist dann so gesetzt, als habe es den Methodenaufruf "AddComment"nicht gegeben. Beispiel: Wenn Sie ein nicht quittiertes Alarms-Objekt kommentieren, erhält das Eventfeld "Time"den Zeitpunkt dieser Kommentaränderung. Nach einem Server-Neustart zeigt das Eventfeld "Time" nicht den Zeitpunkt an, zu dem der Kommentar gesetzt wurde, sondern den Zeitpunkt, zu dem das entsprechende Event gekommen ist.

11.3.6.8 Speichergrenzen für OPC UA Alarms and Conditions hantieren

Der OPC UA-Server der S7-1500 CPU hat für die Funktion "Alarms and Conditions" produktspezifisch begrenzte Speicherkapazität (siehe Technische Daten der CPUs).
Zwei Speicherpools für verschiedene Kategorien von Meldungen stehen zur Verfügung:

- Pool nur für ProgramAlarms (enspricht programmbezogenen Meldungs-Verursachern (Producer) wie z. B. Programmmeldungen über Program_Alarm, ProDiag, Graph)
- Pool nur für SystemDiagnostics (entspricht den Systemdiagnosemeldungen)

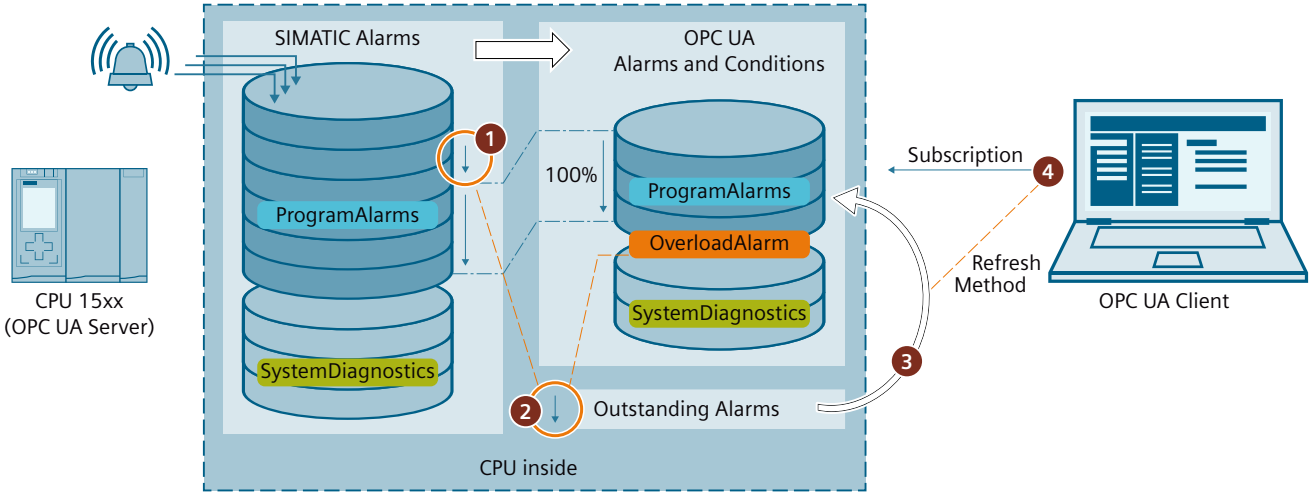
Unter ungünstigen Bedingungen (z. B. Meldeschwall) kann die CPU nicht alle anstehenden Meldungen (ProgramAlarms oder SystemDiagnostics) aus dem SIMATIC-Alarms-Bereich für das OPC UA Alarms-and-Conditions-System verfügbar machen. Meldungen gehen aber trotzdem in diesem Fall nicht verloren.
Sie haben die Möglichkeit, auf dieses Überlastereignis im Anwenderprogramm zu reagieren. Ihrer Anwendung entsprechend können Sie Meldungen, die es sozusagen "nicht ins OPC UA Alarms-and-Conditions-System geschafft haben", über die Methode "ConditionRefresh" wieder dem OPC UA Alarms-and-Conditons-System zur Verfügung stellen.

Voraussetzung

- Alarms and Conditions ist aktiviert
- Event-Subscriptions sind im OPC UA-Client eingerichtet

Prinzip

Das folgende Bild zeigt vereinfacht den Prozess der Zwischenspeicherung von ProgramAlarms-Meldungen, um sie zu einem anderen Zeitpunkt für das OPC UA Alarms and Condition System wieder verfügbar zu machen. Die in der Bildlegende genannten Knoten sind im darauffolgenden Bild des Adressmodells sichtbar.



① Anzahl aktiver Meldungen ist zu hoch, um alle Meldungen über OPC UA Alarms and Conditions zugänglich zu machen.

② Overloads Alarm (Überlastungsmeldung) wird ausgelöst. Diese Überlastungsmeldung ist solange aktiv, bis folgende Situation eingetreten ist:

- Es stehen keine Alarme mehr für das OPC UA Alarms-and-Conditions-System aus (OutstandingProgramAlarms = 0) und
- Anzahl Alarme im OPC UA Alarms-and-Conditions-System < Hysterese-bereinigter Maximalwert für OPC UA Alarme (= MaxAlarmsInQueue - OverloadHysteresis)

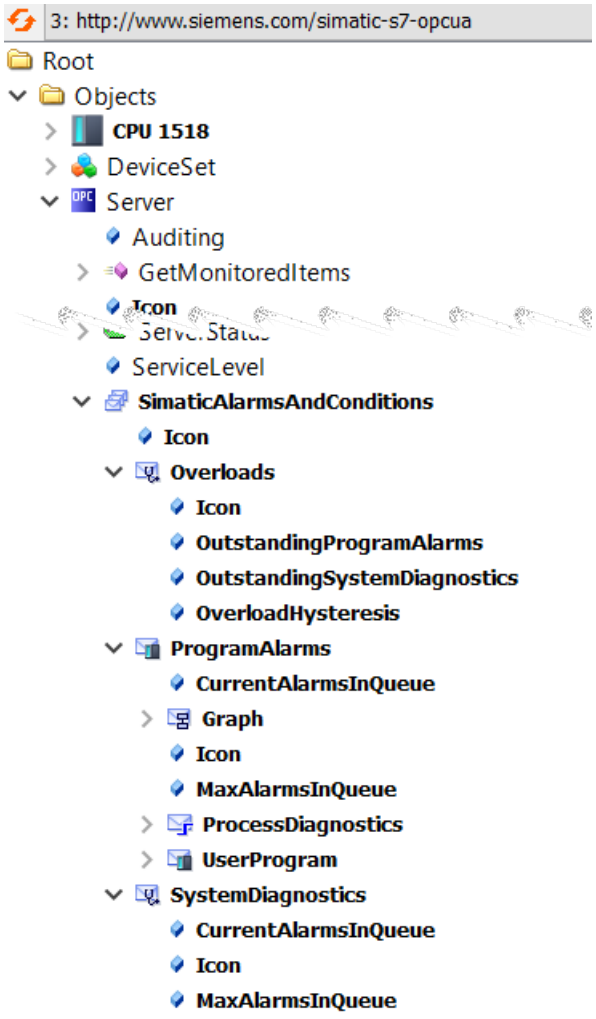
Meldungen, die aufgrund der Überlastungsituation nicht im OPC UA Alarms-and-Conditions-System vorhanden sind, werden von der CPU als "OutstandingAlarms" zwischengespeichert.

③ Wenn ein OPC UA Client die ConditionRefresh-Methode ausführt, werden nicht nur alle Alarm Objekte der betreffenden Subscription synchronisiert, sondern auch die für OPC UA Alarms and Conditions ausstehenden Meldungen (OutstandingAlarms) in den Alarms and Conditions Speicherbereich überführt - soviel, bis die maximale Anzahl Alarme erreicht ist. "Älteste" Alarme werden zuerst überführt. Danach erhält jede Subscription auf diese Alarme die überführten Alarme - nicht nur der OPC UA-Client, der die ConditionRefresh-Methode aufgerufen hat.

④ OPC UA-Client steuert über die Informationen der Overloads-Knoten den Umgang mit ausstehenden Meldungen.

Adressmodell für Alarms and Conditions

Das folgende Bild zeigt die Knoten des OPC UA Alarms-and-Conditions-Adressmodells.



Besonderheiten

- Wenn ausstehende Alarme gehen oder quittiert werden, gelangen sie nicht mehr über die ConditionRefresh-Methode in den OCP UA Alarms-and-Condition-Systembereich. Sie sind dann für OPC UA Alarms and Conditions und damit für die verbundenen OPC UA-Clients "unsichtbar". Dieser Sachverhalt beeinflusst z. B. statistische Auswertungen von Meldungsverläufen.
- Um eine hohe Alarmfrequenz für den Overloads Alarm zu vermeiden, falls das Meldungsaufkommen um den Maximalwert pendelt, ist die Grenze für ein Auslösen des Alarms größer als die Grenze für die Rücknahme dieses Alarms: Der Wert für diese Differenz wird im Knoten "OverloadHysteresis" angezeigt.
  Beispiel: Maximale Anzahl Meldungen: 200, OverloadHysteresis: 3.
  Overloads Alarm wird ausgelöst ab 200 Meldungen und wird erst bei weniger als 197 Meldungen zurückgenommen. Steigt das Meldungsaufkommen wieder, wird ab 200 Meldungen erneut ausgelöst.

11.3.7 Diagnosemöglichkeiten nutzen

11.3.7.1 Diagnose des OPC UA-Servers

Online-Diagnose des OPC UA-Servers

Den OPC UA-Server der S7-1500 CPU können Sie mit gängigen OPC UA-Clients, z. B. UaExpert, online diagnostizieren.
Die Diagnoseinformationen sind in folgende Bereiche aufgeteilt:

- Server Diagnostics
- Sessions Diagnostics
- Subscriptions Diagnostics

Im Adressraum des Servers sind z. B. folgende Knoten mit Diagnoseinformationen vorhanden:

- **ServerDiagnosticsSummary**: Server Diagnose-Zusammenfassung
  - CurrentSessionCount: Anzahl der aktiven Sessions
  - SecurityRejectedSessionCount: Anzahl der Sessions, die wegen nicht zusammenpassender Endpunkt-Security-Einstellungen zwischen Client und Server zurückgewiesen wurden
- **SessionsDiagnosticsSummary**: Session Diagnose-Zusammenfassung
  - ActualSessionTimeout: Eingestellte Zeit, die eine Session z. B. bei Verbindungsabbruch überdauert
- **SubscriptionsDiagnosticsArray**: Array mit je einem Element pro Subscription für die jeweilige Session

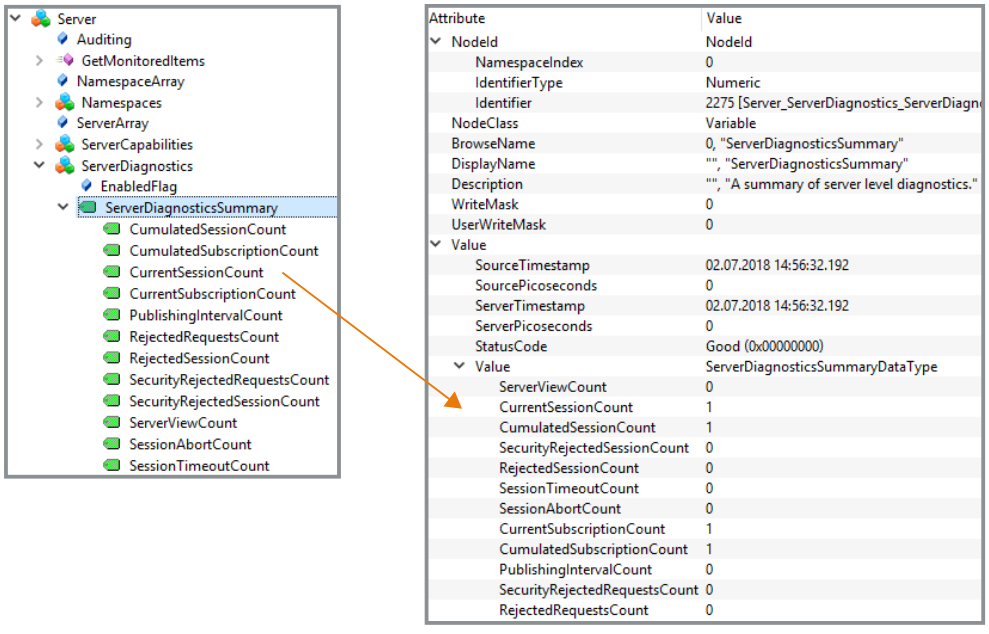


Bild 11-57 Server Diagnostics

Der Knoten SessionsDiagnosticsSummary zeigt auch die Eigenschaften der Client-Applikation, die innerhalb der Session auf den Server zugreift.

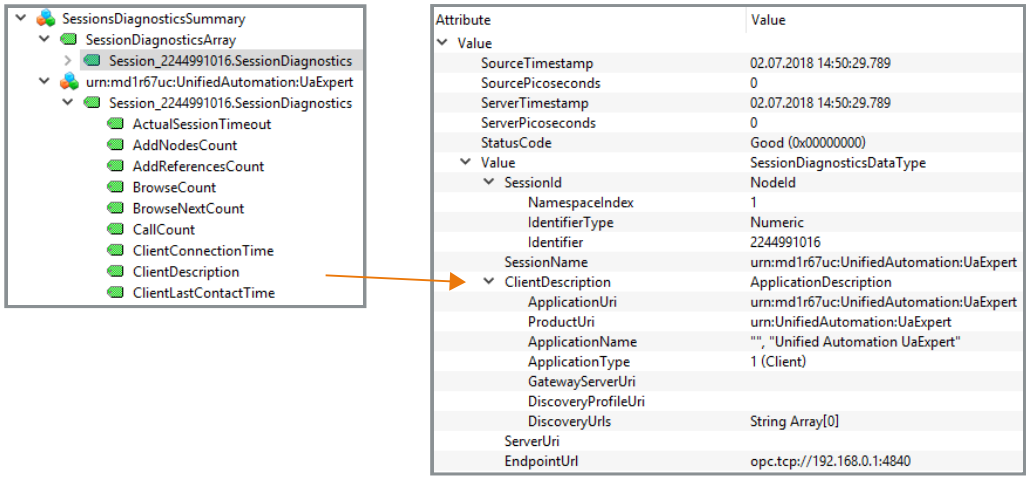


Bild 11-58 Sessions Diagnostics mit den Eigenschaften der Client-Applikation

Diagnose der Verbindung zwischen Client und Server

Um den Status der Verbindung während der Laufzeit des Programms im Client zu diagnostizieren, verwenden Sie die folgende Anweisung:
OPC_UA_ConnectionGetStatus: Verbindungsstatus lesen.

11.3.7.2 OPC UA-Server im Programm diagnostizieren

Ab STEP 7 (TIA Portal) V18 können Sie bei einer S7-1500 CPU (ab Firmware-Version V3.0) auf Knoten im OPC UA-Adressraum zugreifen, um den Inhalt zu Diagnosezwecken im Programm auszuwerten.

Funktionsweise

Im lokalen Adressraum der CPU gibt es zahlreiche Knoten, in die der OPC UA-Server der CPU Daten und Zustände hinterlegt. Mithilfe der Anweisung "OPC_UA_ReadList" können Sie auf diese Informationen zugreifen und sie im Anwenderprogramm auswerten.
Beispiel: "ServerState" ist ein Knoten im Adressraum der CPU, der Werte für den Server-Zustand bzw. für Zustandsübergänge enthält (Running, Shutdown, Failed, ...).
Die Anweisung verwenden Sie dabei nicht als Client-Anweisung, sondern als Anweisung zum Lesen von Knoten des eigenen lokalen OPC UA-Adressraums. Insofern gelten für diesen Anwendungsfall besondere Regeln und Voraussetzungen.

Weitere Informationen

Weitere Informationen zum Aufruf der Anweisung "OPC_UA_ReadList" zu Diagnosezwecken finden Sie in der Hilfe zum TIA Portal, Thema "OPC UA-Server diagnostizieren mit OPC_UA_ReadList".

11.3.7.3 Server-Zustandsübergänge diagnostizieren

Informationen zum Server-Zustand

S7-1500 CPUs sind ab Firmware-Version V2.8 in der Lage, bei Zustandsänderungen des OPC UA-Servers einen Eintrag im Diagnosepuffer zu erzeugen.

Der Diagnosepuffer zeigt den neuen Zustand an.

Ebenso wird die Ursache der Zustandsänderung angezeigt, z. B. Laden in die CPU, NETZ-AUS - NETZ-EIN-Übergang, Anweisung des Anwenderprogramms oder Serviceanfrage von einem Partner (Client).

Voraussetzung

In den OPC UA-Eigenschaften der CPU ist die Option "Änderung des OPC UA-Server-Status" aktiviert (OPC UA > Server > Diagnose).

HINWEIS

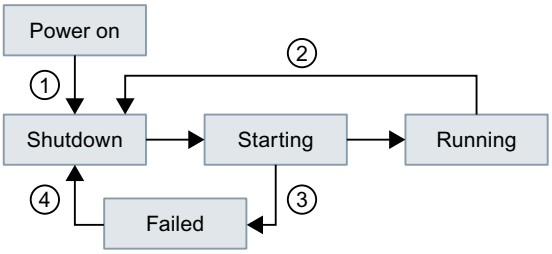
Wenn diese Option aktiviert ist, trägt die CPU auch automatisch nach dem Anlauf die niedrigste eingestellte Security Policy in den Diagnosepuffer ein.

Beispiele

Wenn durch einen Ladevorgang der OPC UA-Server der CPU herunterfährt und anschließend mit einer gültigen neuen Konfiguration startet, dann zeigt der Diagnosepuffer jeweils neuen Server-Zustand an, z. B. Shutdown => Starting => Running.

Wenn durch einen Ladevorgang der OPC UA-Server herunterfährt und wegen eines zu umfangreichen Typedictionarys der Server nicht starten kann, dann zeigt der Diagnosepuffer letztlich den Zustand "Failed" an (Shutdown => Starting => Failed).

Server-Zustände und Zustandsübergänge



①, ④ NETZ-EIN oder Laden im RUN, wenn OPC UA-relevante Daten betroffen sein können.

② Laden der Hardware-Konfiguration mit deaktiviertem OPC UA-Server. Der Server bleibt im Shutdown.
Laden der Hardware-Konfiguration mit aktiviertem OPC UA-Server und fehlerhaften OPC UA-Daten (z. B. zu viele Strukturen mit der Folge, dass das Typedictionary zu groß wird). Der Server kann in diesem Fall nicht starten (siehe ③).

③ OPC UA-Server kann aufgrund z. B. fehlerhafter Daten nicht starten.

Bild 11-59 Server-Zustände und Zustandsübergänge

Beschreibung der Server-Zustände

Im Folgenden sind die einzelnen Zustände erläutert, die der OPC UA-Server einnehmen kann.

Server-Zustände	Erläuterung
Shutdown	Initialer Status • nach NETZ-EIN • nach Laden der Hardware-Konfiguration mit aktiviertem oder deaktiviertem OPC UA-Server • nach dem Laden OPC UA-relevanter Daten
Starting	OPC UA-Adressraum im Server wird initialisiert.
Running	OPC UA-Server läuft (normaler produktiver Zustand für OPC UA-Server).
Failed	Fehlerzustand. OPC UA-Server kann aufgrund z. B. fehlerhafter Daten nicht starten.

11.3.7.4 Session-Zustandsübergänge diagnostizieren

Informationen zum Session-Zustand

S7-1500 CPUs sind ab Firmware-Version V2.8 in der Lage, bei Zustandsänderungen einer OPC UA-Session einen Eintrag im Diagnosepuffer zu erzeugen.
Der Diagnosepuffer zeigt den neuen Zustand an. Ebenso wird die entsprechende Session-ID angezeigt.

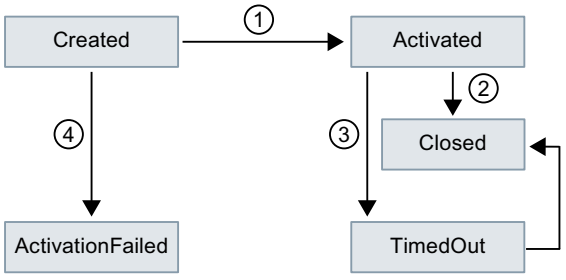
Voraussetzung

In den OPC UA-Eigenschaften der CPU ist die Option "Änderung des Session-Status" aktiviert (OPC UA > Server > Diagnose).

Beispiel

Ein Client überträgt beim Verbindungsaufbau falsche Authentifizierungsdaten (z. B. falsches Passwort). Im Diagnosepuffer wird der neue Zustand der Session "ActivationFailed" eingetragen mit der entsprechenden Session-ID.

Session-Zustände und Zustandsübergänge



① Client verbindet sich mit Server, Login mit korrekten Authentifizierungsdaten (korrekten Credentials).

② Client schließt Verbindung korrekt.

③ Client sendet keine Nachrichten mehr; Session wird mit Timeout beendet.

④ Client verbindet sich mit Server, Login mit falschen Authentifizierungsdaten.

Bild 11-60 Session-Zustände und Zustandsübergänge

11.3.7.5 Auf Security-Ereignisse prüfen

Wenn die CPU im Rahmen der OPC UA-Kommunikation ein Security-Ereignis diagnostiziert, kann sie es in den Diagnosepuffer eintragen.

Voraussetzungen

- S7-1500 CPUs ab FW-Version 2.8
- Die Option "Auf Security-Ereignisse prüfen" ist aktiviert (Eigenschaften der CPU > OPC UA > Server > Diagnose)

Diagnostizierte Security-Ereignisse

S7-1500 CPUs diagnostizieren die folgenden OPC UA-relevanten Security-Ereignisse:

- Client-Zertifikat ist ungültig (z. B. syntaktisch oder semantisch inkorrekt, Signatur inkorrekt, aktuelles Datum liegt nicht im Gültigkeitszeitraum)
- Benutzername/Passwort-Anmeldung ist fehlgeschlagen (deaktiviert oder falsche Daten)
- Client möchte eine bestimmte Security Policy oder einen bestimmten Message Security Mode nutzen; der Server unterstützt die Security Policy bzw. den geforderten Security Mode nicht
- Client baut Verbindung nicht spezifikationskonform (OPC UA Spec) auf (z. B. unerwartete SecureChannelID/SessionID/Client Nonce)

Beispiel

Wenn versucht wird, die Kommunikation zu kompromittieren (z. B. durch Session-Hijacking, Man-in-the-Middle-Angriffe etc.) erkennt der Server dies anhand einer Überprüfung.

11.3.7.6 Anfrage eines entfernten Clients fehlgeschlagen

S7-1500 CPUs sind ab Firmware-Version V2.8 in der Lage, einen Eintrag im Diagnosepuffer zu erzeugen bei folgenden Ereignissen:

- Fehlerhafte Client-Anfragen (Falsche Verwendung)
- Service-Fehler aufgetreten
- CPU-spezifische Grenzen des OPC UA-Servers wurden überschritten

Beispiel für fehlerhafte Client-Anfrage

Eine fehlerhafte Anfrage liegt z. B. vor, wenn ein Client einen Knoten (Variable) adressiert, die nicht vorhanden ist bzw. wenn eine Ressource angefordert wird, die nicht vorhanden ist.
In diesem Fall wird der entsprechende Service, der den Fehler verursacht hat, in den Diagnosepuffer eingetragen und zusätzlich die entsprechende Session-ID.

Service-Fehler

Wenn ein Service selbst fehlschlägt, dann gibt der Server ein Service-Fehler zurück (ServiceFault). In diesem Fall wird der Statuscode (Bad...) sowie die entsprechende Session-ID in den Diagnosepuffer eingetragen.

Beispiel für Grenzen überschritten

Falls eine Service-Anforderung eine CPU-spezifische Grenze überschreitet, z. B. Anzahl Sessions, Anzahl überwachter Elemente (Monitored Items), Anzahl Subscriptions etc., dann wird diese Diagnose in den Diagnosepuffer eingetragen. Zusammen mit der Meldung wird angegeben, welche Grenze überschritten wurde.

Ausnahme: Falls Sie Diagnosen zusammenfassen und die Meldung gehäuft auftritt, wird die fehlerverursachende Grenze nicht eingetragen. Sie erhalten einen allgemeinen Hinweis, dass das unterstützte Mengengerüst überschritten wurde.

Mögliche Einträge für den fehlerverursachenden Service

Je nach verwendetem Client-Anwendung können die Anfragen an den Server aus Anwendersicht unterschiedlich ausgelöst werden, z. B. durch ein Online-Tool mit grafischer Benutzeroberfläche oder durch Anweisungen im Programm eines Clients.

OPC UA mit seiner service-orientierten Architektur folgt einem Request-Response-Paradigma, daher setzt die jeweilige Client-Anwendung die Anfragen in die bei OPC UA definierten Service-Requests um.

Die Namen dieser Services sind festgelegt und entsprechend ihrer Verwendung gruppiert, siehe auch opcfoundation.org.

Genau diese Namen der Services finden Sie zusammen mit der entsprechenden Session-ID als fehlerverursachenden Service bei einer falschen Verwendung im Diagnosepuffer wieder.

Im Folgenden sind die bei OPC UA möglichen Services aufgelistet.

Discovery Service Set
FindServers
GetEndpoints

Session Service Set
CreateSession
ActivateSession
CloseSession
Cancel

View Service Set
Browse
BrowseNext
TranslateBrowsePathsToNodeIds
RegisterNodes
UnregisterNodes

Attribute Service Set
Write
Read

Method Service Set
Call

Monitored Item Service Set
CreateMonitoredItems
ModifyMonitoredItems
DeleteMonitoredItems
SetMonitoringMode
SetTriggering

Subscription Service Set
CreateSubscription
ModifySubscription
DeleteSubscriptions
Publish
Republish
SetPublishingMode

11.3.7.7 Subscriptions diagnostizieren

Informationen zu einer Subscription

S7-1500 CPUs sind ab Firmware-Version V2.8 in der Lage, bei Zustandsänderungen einer Subscription einen Eintrag im Diagnosepuffer zu erzeugen.
Der Diagnosepuffer zeigt den neuen Zustand an; Ausnahme: "KeepAlive".

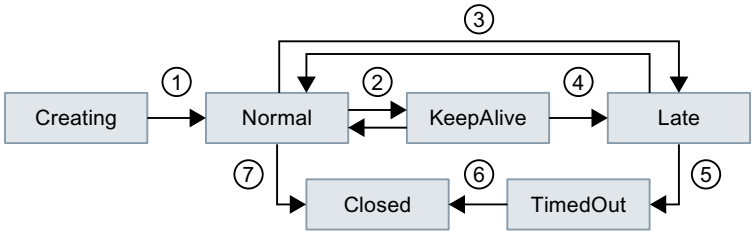
Voraussetzung

In den OPC UA-Eigenschaften der CPU ist die Option "Subscriptions: Änderung des Status" aktiviert (OPC UA > Server > Diagnose).

Beispiel

Ein OPC UA-Client ist mit einer S7-1500 CPU als OPC UA-Server verbunden und erzeugt eine Subscription im Server.
Die Diagnoseoptionen für Subscriptions sind in den OPC UA-Eigenschaften der CPU aktiviert.
Im Diagnosepuffer werden nacheinander die Zustände "Creating" und "Normal" mit der entsprechenden Subscription-ID eingetragen.

Subscription-Zustände und Zustandsübergänge



① Subscription wird erzeugt und ist danach aktiv.

② Zustandswechsel wird nicht im Diagnosepuffer eingetragen, da je nach Datenaufkommen möglicherweise zu viele Einträge im Diagnosepuffer die Folge sind.

③ Siehe Erläuterung in Tabelle zu "Late"; z. B. keine Sendeanforderungen vom Client.

④ Maximaler KeepAlive-Wert erreicht.

⑤ Siehe Erläuterung in Tabelle zu "TimedOut".

⑥ Maximale Lifetime der Subscription erreicht.

⑦ Client hat Subscription gelöscht.

Bild 11-61 Subscription-Zustände und Zustandsübergänge

Beschreibung der Subscription-Zustände

Folgende Zustände kann eine Subscription im OPC UA-Server einnehmen:

Zustand	Bedeutung
Creating	Client hat eine Subscription im Server angefordert; der Server erzeugt die Subscription.
Normal	Subscription ist im Server angelegt und aktiv.
Closed	Client hat die Subscription gelöscht.
KeepAlive	Zustand, wenn sich über längere Zeit die überwachten Elemente (monitored items) nicht ändern. Diese Zustandsübergänge werden nicht in den Diagnosepuffer eingetragen.
Late	Client hat eine Subscription mit minimalen Abtast- und Sendeintervallen erzeugt. Die Menge überwachter Elemente wird in dieser Zeit nicht an den Client übermittelt. Client übermittelt keine Sendeanforderungen mehr (z. B. wegen Ausfall).
TimedOut	Der Client hat eine Subscription angefordert. Der Server kann nur dann die Subscription bedienen (Publish Response senden), wenn ausreichend Sendeanforderungen (Publish Requests) vom Client vorhanden sind. Wenn der Client keine Sendeanforderungen mehr sendet, geht die Subscription nach einer gewissen Zeit in den Zustand "TimedOut".

Subscription: Fehler bei den Abtastzeiten

Ab Firmware V2.5 der SIMATIC S7-1500 CPU kann der OPC UA-Server bei der Verwendung von Subscriptions den Statuscode "GoodOverload" übermitteln, wenn beim Sampling der Items eine Überlast der CPU entsteht.
Ab Firmware V2.8 der SIMATIC S7-1500 CPU kann der OPC UA-Server dieses Ereignis auch in den Diagnosepuffer eintragen.

Voraussetzung

In den OPC UA-Eigenschaften der CPU ist die Option "Subscriptions: Fehler bei den Abtastzeiten" aktiviert (OPC UA > Server > Diagnose).

Fehlerfreie Subscription

Bei einer OPC UA-Subscription auf verschiedene Elemente (Items, z. B. Variablen) muss der OPC UA-Server der SIMATIC S7-1500 in vorgegebenen Intervallen (Abtastintervall) die Elemente auf Wertänderung überprüfen. Diese Überprüfung, das sogenannte "Sampling", benötigt eine gewisse Zeit, die abhängig von der Anzahl und dem Datentyp der Items ist. Nachdem das Sampling abgeschlossen ist und ein Sendeauftrag (Publishing-Request) vorliegt, sendet der Server die Elemente an den Client.

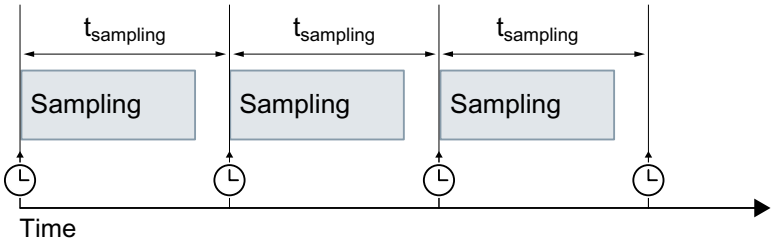
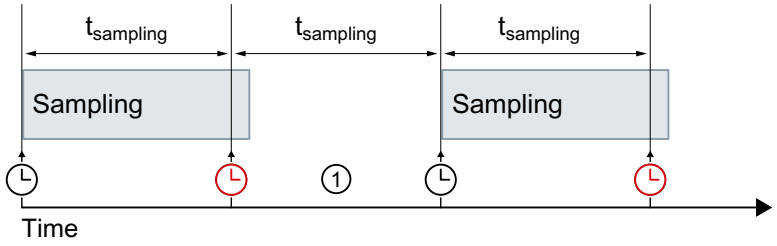


Bild 11-62 Fehlerfreie Subscription

Subscription mit Fehler

Wenn sich zu viele Elemente in der Warteschlange (Queue) befinden, dann kann es zu einer Überlast des Kommunikations-Stacks kommen, einem sogenannten "Overload". Hierbei kann die CPU nicht alle Elemente im vorgegebenen Sampling-Intervall überprüfen und muss deswegen den nächsten Sampling-Auftrag überspringen.
In diesem Fall sendet die CPU den Statuscode "GoodOverload" (0x002F0000) pro Element, obwohl die Elemente nicht überprüft wurden. Die Bedeutung des Statuscodes gemäß IEC 61131-3 lautet: "Sampling has slowed down due to resource limitations".



① Sampling-Auftrag wird übersprungen

Bild 11-63 Subscription mit Fehler

Siehe auch FAQ 109763090 (https://support.industry.siemens.com/cs/ww/de/view/109763090).

Weitere Informationen

Informationen zu den Servereinstellungen für Subscriptions finden Sie im Kapitel Einstellungen des Servers für Subscriptions (Seite 239).

11.3.7.8 Diagnosen zusammenfassen

Um zu verhindern, dass der Diagnosepuffer von sehr vielen identischen OPC UA-Diagnosen "überschwemmt" wird, können Sie ab STEP 7 V16 parametrieren, dass diese Diagnosen als Sammelmeldung in den Diagnosepuffer eingetragen werden. Pro Interval (Überwachungszeitraum) erzeugt die CPU dann nur noch eine Sammelmeldung pro OPC UA-Diagnose.
Welche Diagnosen die CPU zusammenfasst und wie der Prozess bei hoher Meldungsanzahl abläuft, erfahren Sie in den nächsten Abschnitten.

Voraussetzung

In den OPC UA-Eigenschaften der CPU ist die Option "Diagnosen bei hoher Meldungsanzahl zusammenfassen" aktiviert (OPC UA > Server > Diagnose, Bereich "Diagnosen zusammenfassen").

Beispiel

Ein OPC UA-Client "überfordert" eine S7-1500 CPU als OPC UA-Server wiederholt mit einer Abtastrate, die der Server nicht bedienen kann (Überlast).
Die Einstellung "Diagnosen bei hoher Meldungsanzahl zusammenfassen" ist aktiviert.
Im Diagnosepuffer erscheint für diese Diagnose die Meldung, dass die Abtastrate nicht erreicht werden konnte; gefolgt von der Anzahl dieser Ereignisse im projektierten Intervall.

OPC UA-Diagnosen, die zusammengefasst werden können

Die im Folgenden aufgezählten Diagnosen bilden jeweils eigene Gruppen (Typ). Mit der Einstellung "Diagnosen bei hoher Meldungsanzahl zusammenfassen" werden Diagnosen aus derselben Gruppe verdichtet:

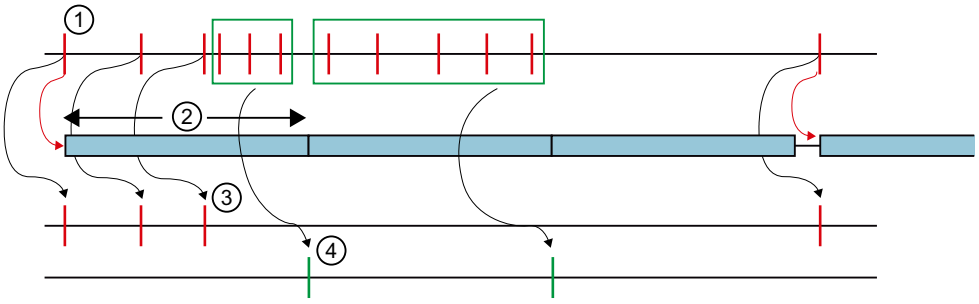
- Falsche Verwendung eines OPC UA-Services
- OPC UA-Service-Fehler
- Subscription-Status hat sich geändert
- Abtastrate konnte nicht erreicht werden (Subscriptions, Überlast)
- OPC UA-Security-Überprüfung fehlgeschlagen
- Mengengerüst des OPC UA-Servers überschritten

Funktionsweise

Die CPU trägt die ersten drei Diagnosen einer Gruppe sofort in den Diagnosepuffer ein. Alle danach folgenden Diagnosen dieser Gruppe ignoriert sie.

Am Ende des Überwachungszeitraums (Intervall) erzeugt die CPU eine Sammelmeldung, in der sie die Diagnose und die Häufigkeit dieser Diagnose während des abgelaufenen Intervalls einträgt. Treten diese Diagnosen auch in den folgenden Intervallen auf, so generiert die CPU nur noch jeweils eine Sammelmeldung pro Folgeintervall.

Im Diagnosepuffer hinterlässt ein Diagnoseschwall also folgendes Muster: Drei Einzelmeldungen gefolgt von einer Reihe von Sammelmeldungen. Diese Reihe kann aus zwei, drei oder mehr Sammelmeldungen bestehen; das ist abhängig von der gewählten Überwachungszeit und der Dauer des Diagnoseschwalls.



① Diagnosen einer Gruppe (eines Typs), zum Beispiel "Abtastrate konnte nicht erreicht werden"

② Intervall (Überwachungszeitraum): Wenn eine Diagnose zum ersten Mal auftritt (oder erneut auftritt), wird die Überwachungszeit gestartet (oder erneut gestartet).

③ Einzelmeldungen: Die drei ersten Diagnosen aus derselben Gruppe werden sofort in den Diagnosepuffer eingetragen. Ab der vierten Diagnose erzeugt die CPU nur noch Sammelmeldungen. Tritt nach einer Pause von mindestens einem Intervall wieder eine Diagnose dieser Gruppe auf, trägt die CPU eine Einzelmeldung in den Diagnosepuffer ein und startet den Überwachungszeitraum erneut.

④ Sammelmeldungen: Die CPU erzeugt nach drei Diagnosen nur noch eine Sammelmeldung als Zusammenfassung aller weiteren Diagnosen in diesem Intervall. Treten diese Diagnosen auch in den folgenden Intervallen auf, so generiert die CPU jeweils nur eine Sammelmeldung pro Folgeintervall.

Bild 11-64 Zusammenfassung von Diagnosen

11.4 S7-1500 CPU als OPC UA-Client nutzen

11.4.1 Übersicht und Voraussetzungen

Mit STEP 7 (TIA Portal) können Sie ab der Version V15.1 einen OPC UA-Client parametrieren und programmieren, der PLC-Variablen in einem OPC UA-Server lesen kann. Ferner ist es möglich, zu einem OPC UA-Server neue Werte für PLC-Variablen zu übertragen. Außerdem können Sie in Ihrem Anwenderprogramm Methoden aufrufen, die ein OPC UA-Server bereitstellt. Sie verwenden dazu in Ihrem Anwenderprogramm die Anweisungen für OPC UA-Clients.

Die Anweisungen des OPC UA-Clients sind angelehnt an den Standard "PLCopen OPC-UA Client for IEC61131-3".

Spezifikation PLCopen

Mit diesen standardisierten Anweisungen können Sie in Ihrem Anwenderprogramm OPC UA-Client-Funktionen nutzen, die in einer S7-1500 CPU ausgeführt werden.

Außerdem ist es mit geringen Anpassungen möglich, dieses Anwenderprogramm auch in Steuerungen anderer Hersteller auszuführen, sofern diese Hersteller ebenfalls die OPC UA Spezifikation "PLCopen OPC-UA Client for IEC61131-3" umgesetzt haben.

Komfortable Editoren in STEP 7

Für die Parametrierung der Anweisungen für OPC UA-Clients steht Ihnen im TIA Portal ein komfortabler Editor zur Verfügung - die Verbindungsparametrierung (Seite 233).

Außerdem verfügt STEP 7 ab Version 15.1 auch über einen Editor für Client-Schnittstellen (Seite 336).

Dieses Kapitel beschreibt, wie Sie mit diesen Editoren arbeiten.

Zunächst soll gezeigt werden, wie Sie mit dem Schnittstellen-Editor eine neue Schnittstelle anlegen und konfigurieren, da Sie eine solche Schnittstelle für die spätere Verbindungsparametrierung benötigen.

Die Beschreibung verwendet zur besseren Verständlichkeit ein Beispiel, siehe Beschreibung des Beispiels (Seite 335).

Voraussetzungen

- Sie verfügen über die erforderliche Runtime-Lizenz für OPC UA und haben die Lizenz in STEP 7 projektiert (Eigenschaften der CPU > Runtime-Lizenzen).
- Der Client der S7-1500 CPU ist aktiviert.

Um den Client der S7-1500 CPU zu nutzen, müssen Sie den Client aktivieren:

1. Wählen Sie in den Eigenschaften der CPU den Bereich "OPC UA > Client".
2. Aktivieren Sie die Option "OPC UA-Client aktivieren".

Wenn Sie den Client nicht aktivieren, kommt kein Verbindungsaufbau zustande. Sie erhalten eine entsprechende Fehlermeldung an den Anweisungen, z.B. "OPC_UA_Connect".

Informationen zum Applikationsnamen, der auch für den Server und den Client gilt, finden Sie hier (Seite 233).

Übersicht

Um den Editor und die Verbindungsparametrierung zu verwenden, gehen Sie folgendermaßen vor:

1. Legen Sie zunächst eine Client-Schnittstelle an. Fügen Sie der Schnittstelle PLC-Variablen und -Methoden hinzu, auf die Sie zugreifen möchten ("Erster Schritt (Seite 336)").
2. Parametrieren Sie anschließend die Verbindung zum OPC UA-Server (Zweiter Schritt (Seite 350)).
3. Schließlich verwenden Sie die parametrierte Verbindung bei den OPC UA-Client-Anweisungen (Dritter Schritt (Seite 357)).

11.4.2 Wissenswertes zu den Client-Anweisungen

Die S7-1500 CPU als OPC UA-Client ermöglicht Ihnen, mit standardisierten OPC UA Client-Anweisungen die Kommunikation für folgende Aufgaben zu steuern:

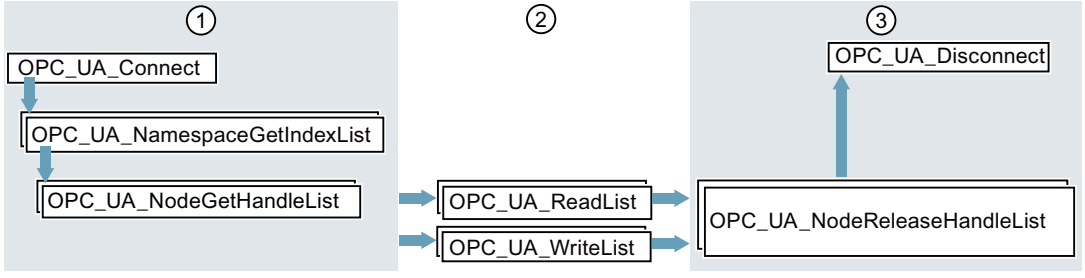
- Lesen/Schreiben von Variablen des OPC UA-Servers
- Methoden im OPC UA-Server aufrufen

Optional verwendbare Anweisungen erlauben Ihnen folgende Informationen zu ermitteln:

- Den Status der Verbindung zwischen OPC UA-Client und OPC UA-Server
- NodeIds von Knoten bei bekannter Hierarchie des Adressraums

Standardisierter Ablauf der OPC UA Kommunikation

Der Ablauf der Kommunikation und damit die Reihenfolge der Anweisungen folgt einem Muster, das im Folgenden veranschaulicht ist.

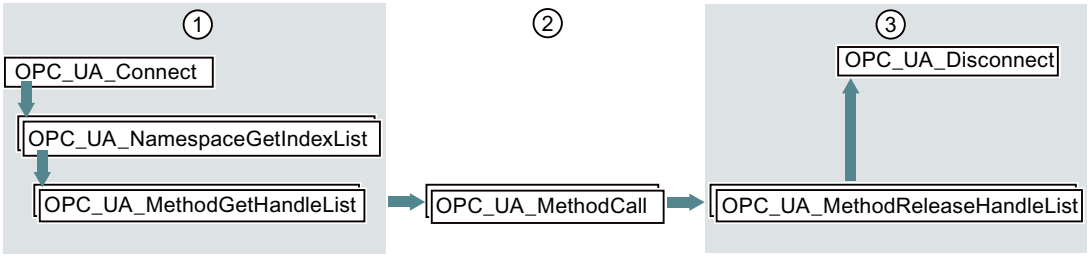


① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum "Clean-up" nach durchgeführten Lese- oder Schreibvorgängen

Bild 11-65 Ablaufreihenfolge für Lese- bzw. Schreibvorgang



① Anweisungen zur Vorbereitung der Methodenaufrufe

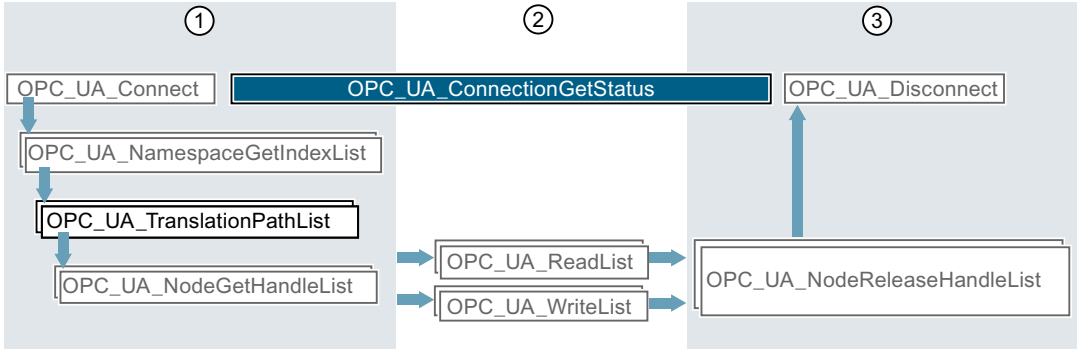
② Methodenaufrufe

③ Anweisungen zum "Clean-up" nach durchgeführten Methodenaufrufen

Bild 11-66 Ablaufreihenfolge für Methodenaufruf im OPC UA-Server

Optionale Anweisungen (Auslesen des Status einer Verbindung / Auslesen von NodeIds von Knoten bei bekannter Hierarchie des Adressraums)

- OPC_UA_ConnectionGetStatus
- OPC_UA_TranslatePathList



① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge mit eingeschobener Anweisung zum Erfragen z. B. der NodeIds von Knoten des OPC UA-Servers.

② Parallel zu anderen Anweisungen können Sie zwischen Verbindungsaufbau und Verbindungsabbau den Verbindungsstatus ermitteln

③ Anweisungen zum "Clean-up"

Bild 11-67 Ablaufreihenfolge optionale Anweisungen

Komfortable Editoren in STEP 7

Die OPC UA-Client Anweisungen sind detailliert im Referenzteil (STEP 7 Informationssystem) beschrieben. Für die Parametrierung der Anweisungen steht Ihnen im TIA Portal ein komfortabler Editor zur Verfügung - die Verbindungsparametrierung (Seite 350).

Wir empfehlen für den ersten Programmentwurf, mit der Verbindungsparametrierung zu beginnen und bei Bedarf weitere Anweisungen zu nutzen und das Programm manuell zu optimieren.

Informationen zu den Client-Anweisungen

Die Client-Anweisungen sind detailliert beschrieben in der Hilfe zu den Anweisungen > Kommunikation > OPC UA > OPC UA-Client.

Anwendungsbeispiel im Online-Support

In diesem Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/109762770) steht Ihnen ein S7-Anwenderbaustein "OpcUaClient" zur Verfügung, der die wichtigsten Funktionen der OPC UA-Anweisungen zusammenfasst, die Implementierung für Sie beschleunigt und die Programmierung vereinfacht. Als OPC UA-Server im Beispiel dient eine S7-1500-Steuerung mit einem einfachen Simulationsprogramm für Prozesswerte.

Der S7-Anwenderbaustein leistet Folgendes:

- Verbindungsaufbau und Verbindungsabbau zum Server
- Diagnose der Verbindung und automatisches Neu-Verbinden bei Verbindungsabbrüchen
- Registered Read
- Registered Write
- Registered Method Call

11.4.3 Anzahl gleichzeitig nutzbarer Client-Anweisungen

Gleichzeitig nutzbare OPC UA-Client-Anweisungen

Folgende Grenzen gelten für die gleichzeitige Nutzung von OPC UA-Client-Anweisungen (tagesaktuelle Technische Daten der CPUs finden Sie im Internet (https://support.industry.siemens.com/cs/ww/de/ps/td)):

Tabelle 11-5 Mengengerüste für OPC UA-Client-Anweisungen

OPC UA-Anweisung	**Max. Anzahl für CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)**	**Max. Anzahl für CPU 1505 (S/SP/SP F/SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)**	**Max. Anzahl für CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)**
OPC_UA_Connect	4	10	40
OPC_UA_NamespaceGetIndex-List	4*	10*	40*
OPC_UA_NodeGetHandleList	4*	10*	40*
OPC_UA_MethodGetHandleList	4*	10*	40*
OPC_UA_TranslatePathList	4*	10*	40*
OPC_UA_ReadList	20 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	50 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	200 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)
OPC_UA_WriteList	20 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	50 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	200 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)

* max. 1 pro Verbindung

OPC UA-Anweisung	**Max. Anzahl für CPU 1510SP (F) CPU 1511 (C/F/T/TF) CPU 1512C CPU 1512SP (F) CPU 1513 (F)**	**Max. Anzahl für CPU 1505 (S/SP/SP F/SP T/SP TF) CPU 1515 (F/T/TF) CPU 1515 SP PC (F/T/TF) CPU 1516 (F/T/TF)**	**Max. Anzahl für CPU 1507S (F) CPU 1517 (F/T/TF) CPU 1518 (F)**
OPC_UA_MethodCall	20 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	50 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)	200 insgesamt (max. 5 pro Verbindung, siehe OPC_UA_Connect)
OPC_UA_NodeReleaseHandleList	4*	10*	40*
OPC_UA_MethodReleaseHandle-List	4*	10*	40*
OPC_UA_Disconnect	4*	10*	40*
OPC_UA_ConnectionGetStatus	4*	10*	40*

* max. 1 pro Verbindung

Maximale Anzahl nutzbarer OPC UA-Client-Schnittstellen

Wenn Sie mit Hilfe der Verbindungsparametrierung OPC UA-Client Schnittstellen anlegen, dann ist die Anzahl von Client-Schnittstellen auf max. 40 begrenzt.

Sie legen die OPC UA-Client-Schnittstellen an, indem Sie im Bereich "OPC UA-Kommunikation" in der Projektnavigation auf das Symbol "Neue Client-Schnittstelle hinzufügen" doppelklicken.

Die maximale Anzahl von OPC UA-Client-Schnittstellen ist unabhängig davon, ob Sie die CPU zusätzlich als OPC UA-Server nutzen.

11.4.4 Beispiel-Konfiguration für OPC UA

In den folgenden Kapiteln ist beschrieben, wie Sie den Editor für Client-Schnittstellen sowie die Verbindungsparametrierung nutzen.

Die Beschreibung geht von einem konkreten Beispiel aus: In der Anlage arbeiten zwei S7-1500 CPUs: Eine CPU dient als OPC UA-Client, die andere als OPC UA-Server.

Selbstverständlich können Sie auch Steuerungen, Sensoren oder IT-Systeme anderer Hersteller als OPC UA-Clients oder -Server verwenden. Gerade der Datenaustausch zwischen unterschiedlichen Systemen (Interoperabilität) ist ein großer Vorteil von OPC UA.

Verbindungsparametrierung anhand eines Beispiels:

Die Anlage produziert in einer Fertigungslinie Rohlinge.

Die folgenden Steuerungen werden eingesetzt:

1. Als Controller der Fertigungslinie dient eine S7-1511 CPU.
   Der Controller wird im Beispiel "**Productionline**" genannt.
   Der OPC UA-Server der Steuerung ist aktiviert.
   Die CPU besitzt im Beispiel die IP-Adresse 192.168.1.1.
   Diese CPU veröffentlicht über den OPC UA-Server die Werte der folgenden Variablen:
   - **NewProduct**
     Die Variable besitzt den Datentyp "Bool".
     Wenn die Variable den Wert TRUE besitzt, dann hat die Fertigungslinie einen Rohling bearbeitet.
     Der Rohling steht für die Abholung bereit.
   - **ProductNumber**
     Diese Variable enthält die Identnummer des Rohlings.
     Die Variable besitzt den Datentyp "Int".
   - **Temperature**
     Diese Variable enthält Temperaturwerte, die während der Fertigung des Rohlings erfasst wurden.
     Die Variable ist ein Array mit Elementen des Datentyps "Real".

   Außerdem stellt diese CPU die folgende beschreibbare Variable bereit:
   - **ProductionEnabled**
     Die Variable wird vom OPC UA-Client gesetzt.
     Die Variable besitzt den Datentyp "Bool".
     Wird der Wert auf TRUE gesetzt, dann ist die Fertigungslinie freigegeben und darf Rohlinge herstellen.

   Außerdem stellt diese CPU über den OPC UA-Server die folgende Methode bereit:
   - **OpenDoor**
     Damit können OPC UA-Clients veranlassen, dass eine Zugangstür zur Fertigungslinie geöffnet wird.
2. Eine S7-1516 CPU kontrolliert die Zusammenarbeit mit anderen Fertigungslinien.
   Diese CPU wird im Beispiel "**Supervisor**" genannt.
   Der OPC UA-Client dieser CPU ist aktiviert.
   Über OPC UA kann diese CPU die Variablen NewProduct und ProductNumber lesen, die Variable ProductionEnabled setzen sowie die Methode OpenDoor aufrufen.
   Diese CPU besitzt im Beispiel die IP-Adresse 192.168.1.2.

Das folgende Bild zeigt das Beispiel in der Netzsicht des TIA Portals:

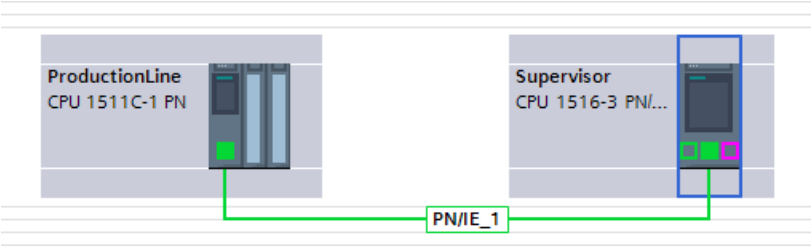


Bild 11-68 Beispiel für Verbindungsparametrierung in der Netzsicht

11.4.5 Client-Schnittstellen anlegen

Das TIA Portal verfügt ab der Version 15.1 über einen Editor für Client-Schnittstellen.
In einer Client-Schnittstelle fassen Sie alle PLC-Variablen zusammen, die Sie von einem OPC UA-Server lesen oder schreiben wollen.
Außerdem enthält die Client-Schnittstelle alle Methoden, die der OPC UA-Server bereitstellt und die Sie mit Ihrem Anwenderprogramm, das als OPC UA-Client fungiert, aufrufen wollen.
Wenn Sie eine Client-Schnittstelle anlegen, dann erstellt STEP 7 auch Datenbausteine für die Parametrierung der Verbindung zu dem OPC UA-Server, von dem Sie Daten lesen oder zu dem Sie Daten schreiben wollen.

Maximale Anzahl von Client-Schnittstellen

Sie können maximal 40 Client-Schnittstellen anlegen.

Editor für Client-Schnittstellen

Um eine Client-Schnittstelle zu erstellen, gehen Sie folgendermaßen vor:

1. Wählen Sie die Projektansicht im TIA Portal
2. Wählen Sie im Bereich "Geräte" die CPU aus, die Sie als OPC UA-Client nutzen wollen.
3. Klicken Sie auf "OPC UA-Kommunikation > Client-Schnittstellen".
4. Doppelklicken Sie auf "Neue Client-Schnittstelle hinzufügen".
   STEP 7 erzeugt eine neue Client-Schnittstelle und zeigt diese im Editor an:

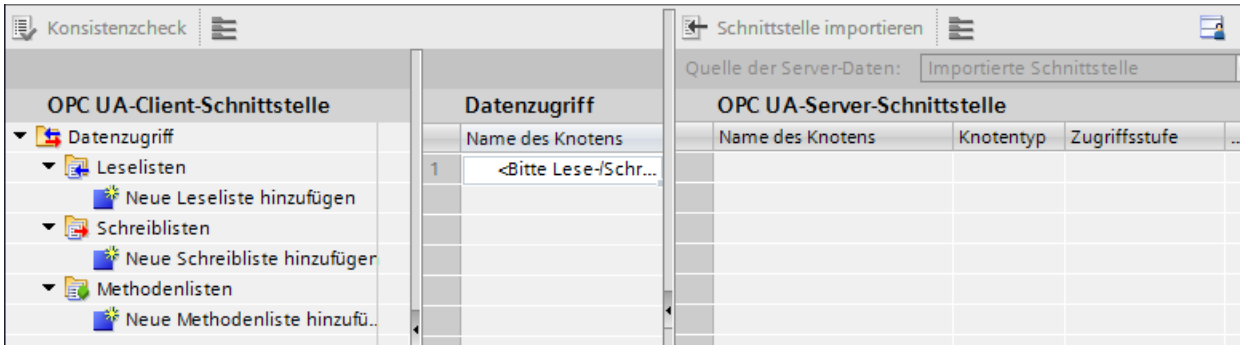


Bild 11-69 OPC UA-Client-Schnittstelle hinzufügen

STEP 7 nennt die neue Schnittstelle "Client-Schnittstelle_1". Wenn es schon eine "Client-Schnittstelle_1" gibt, dann erhält die neue Schnittstelle die Bezeichnung "Client-Schnittstelle_2" usw.

Außerdem erzeugt STEP 7 die folgenden Datenbausteine:

- Client-Schnittstelle_1_Configuration
  Der Datenbaustein enthält bereits alle Systemdatentypen, die für die Anweisungen des OPC UA-Clients benötigt werden.
  Wenn Sie die Verbindung zum OPC UA-Server parametrieren, wird dieser Datenbaustein gefüllt.
  Sie parametrieren eine Verbindung in den Eigenschaften der Client-Schnittstelle, siehe: Beispiel-Konfiguration für OPC UA (Seite 335).
- Client-Schnittstelle_1_Data
  Ein Datenbaustein für die PLC-Variablen, die Sie von einem OPC UA-Server lesen oder schreiben wollen sowie für Methoden, die Sie im OPC UA-Server aufrufen wollen.
  Diesen Datenbaustein verwenden Sie in Ihrem Anwenderprogramm.
  Aktuell ist dieser Datenbaustein noch leer.

5. Verwenden Sie einen aussagekräftigen Namen für die neue Client-Schnittstelle.
   Im Beispiel wählen Sie "Productionline".
   Dadurch ändern sich auch die Namen der zugehörigen Datenbausteine in:
   - Productionline_Data
   - Productionline_Configuration
6. Um eine OPC UA-Server-Schnittstelle zu importieren, klicken Sie rechts oben im Editor auf die Schaltfläche "Schnittstelle importieren".
   Dadurch können Sie eine XML-Datei importieren, welche die Server-Schnittstelle eines OPC UA-Servers beschreibt.
   Alternative: Sie ermitteln online die Server-Schnittstelle eines verbundenen OPC UA-Servers, siehe: Server-Schnittstelle online ermitteln (Seite 343).
7. STEP 7 zeigt einen Dialog an, mit dem Sie eine XML-Datei auswählen.
   Diese XML-Datei beschreibt den Adressraum eines OPC UA-Servers.
   Der Adressraum eines OPC UA-Servers enthält alle PLC-Variablen und Server-Methoden, die ein OPC UA-Server veröffentlicht.
   Auf diesen Adressraum können OPC UA-Clients zugreifen:
   - PLC-Variablen lesen
   - PLC-Variablen schreiben
   - Server-Methoden aufrufen

   Der Adressraum eines OPC UA-Servers kann in eine oder mehrere Server-Schnittstellen unterteilt sein.
   Für das Erstellen von Server-Schnittstellen, siehe: Server-Schnittstelle für Companion Spezifikation anlegen (Seite 264).
8. Erzeugen Sie eine **Leseliste** in dieser Client-Schnittstelle.
   Dazu gehen Sie folgendermaßen vor:
   - Klicken Sie im linken Abschnitt des Editors auf "Neue Leseliste hinzufügen".
     STEP 7 fügt eine neue Liste hinzu mit dem Namen "Leseliste_1".
     Für das Beispiel ändern Sie den Namen in "ReadListProduct"
   - Fügen Sie nun der neuen Leseliste die PLC-Variablen hinzu, die Sie vom OPC UA-Server lesen wollen.
     Im Beispiel werden in der Leseliste "ReadListProduct" die Variablen "NewProduct" und "ProductNumber" hinzugefügt.

Wählen Sie die Variable "NewProduct" im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") aus. Ziehen Sie die Variable "NewProduct" per Drag & Drop zur Leseliste "ReadProduct" im mittleren Feld des Editors. Verfahren Sie genau so mit der Variable "ProductNumber".

Das folgende Bild zeigt das rechte Feld des Editors:

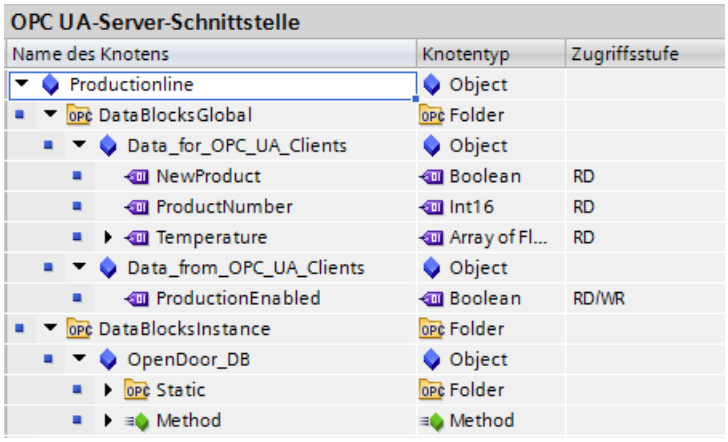


Bild 11-70 Leseliste in der OPC UA-Server-Schnittstelle

Alternative:

Sie können eine neue Leseliste auch erzeugen, indem Sie im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") einen Knoten des Typs Object oder Folder auswählen und dann per Drag & Drop zu "Neue Leseliste hinzufügen" im linken Feld des Editors ziehen. Die neue Leseliste enthält dann alle PLC-Variablen des gezogenen Knotens.

Im Beispiel wählen Sie das Objekt "Data_for_OPC_UA_Clients" aus, das die Variablen "NewProduct" und "ProductNumber" enthält. STEP 7 erzeugt die neue Leseliste "Data_for_OPC_UA_Clients". Außerdem enthält das Objekt auch die Variable "Temperature". Löschen Sie die Variable "Temperature" aus der Leseliste. Da sie im Beispiel nicht gelesen werden soll.

Ändern Sie den Namen der Leseliste in "ReadListProduct".

Das folgende Bild zeigt den Inhalt der Leseliste:

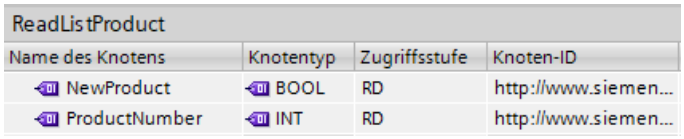


Bild 11-71 Leseliste

HINWEIS

Lese- und Schreiblisten unterstützen nicht alle Knotentypen

Der OPC UA-Client der S7-1500 CPU unterstützt nicht sämtliche OPC UA-Datentypen (Knotentypen), die über eine OPC UA-Server-Schnittstelle bereitgestellt werden können. Wenn Sie einen nicht-unterstützten Knotentyp z. B. in einer Leseliste oder Schreibliste platzieren, erscheint eine entsprechende Fehlermeldung. In diesem Fall können Sie den entsprechenden Knoten nicht in die Lese- oder Schreibliste aufnehmen.

Welche Typen unterstützt werden, ist hier beschrieben: Mapping von Datentypen (Seite 179)

9. Falls Sie PLC-Variablen neue Werte zuweisen wollen, erzeugen Sie eine **Schreibliste** in dieser Client-Schnittstelle.
   Dazu gehen Sie folgendermaßen vor:
   – Klicken Sie im linken Abschnitt des Editors auf "Neue Schreibliste hinzufügen".
     STEP 7 fügt eine neue Liste hinzu mit dem Namen "Schreibliste_1".
     Für das Beispiel ändern Sie den Namen in "WriteListStatus".
   – Fügen Sie nun der neuen Schreibliste alle Variablen des OPC UA-Servers hinzu, denen Sie neue Werte zuweisen wollen.
     Im Beispiel fügen Sie der Schreibliste "WriteListStatus" die Variable "ProductionEnabled" hinzu.
     Wählen Sie die Variable im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") aus. Ziehen Sie die Variable per Drag & Drop zur Schreibliste im mittleren Feld des Editors.

   Alternative:
   Sie können eine neue Schreibliste auch erzeugen, indem Sie im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") einen Knoten des Typs Object oder Folder auswählen und dann per Drag & Drop zu "Neue Schreibliste hinzufügen" im linken Feld des Editors ziehen.
   Die neue Schreibliste enthält dann alle Variablen des betreffenden Knotens.
   Im Beispiel wählen Sie das Objekt "Data_from_OPC_UA_Clients" aus, das die Variable "ProductionEnabled" enthält. STEP 7 erzeugt die neue Schreibliste "Data_from_OPC_UA_Clients". Ändern Sie den Namen in "WriteListStatus".
   Das folgende Bild zeigt den Inhalt der Schreibliste:

WriteListStatus

Name des Knotens	Knotentyp	Zugriffsstufe	Knoten-ID
ProductionEnabled	BOOL	RD/WR	http://www.siemen...

Bild 11-72 Schreibliste

10. Falls Sie eine Methode dieses OPC UA-Servers aufrufen wollen, erzeugen Sie eine neue Methodenliste.
    Dazu gehen Sie folgendermaßen vor:
    – Klicken Sie im linken Abschnitt des Editors auf "Neue Methodenliste hinzufügen".
      STEP 7 fügt eine neue Liste hinzu mit dem Namen "Methodenliste_1".
      Für das Beispiel ändern Sie den Namen in "MethodListOpenDoor".
    – Fügen Sie nun der neuen Methodenliste eine Methode des OPC UA-Servers hinzu.
      In diesem Beispiel fügen Sie der Methodenliste "MethodListOpenDoor" die Methode "OpenDoor" hinzu.
      Wählen Sie die Methode im rechten Feld des Editors ("OPC UA-Server-Schnittstelle") aus. Ziehen Sie die Methode per Drag & Drop zur Methodenliste im mittleren Feld des Editors.

    Alternative:
    Sie können eine neue Methodenliste auch erzeugen, indem Sie im rechten Feld des Editors (OPC UA-Server-Schnittstelle) eine Methode (Knoten des Typs Object) auswählen und dann per Drag & Drop zu "Neue Methodenliste hinzufügen" im linken Feld des Editors ziehen. Die neue Methodenliste enthält dann die Methode des betreffenden Knotens.
    Das folgende Bild zeigt den Inhalt der Methodenliste:

MethodListOpenDoor

Name des Knotens	Knotentyp	Zugriffsstufe	Knoten-ID
Method			http://www.siemen...

Bild 11-73 Methodenliste

Falls Sie eine weitere Methode des OPC UA-Servers aufrufen wollen, dann müssen Sie eine neue Methodenliste anlegen. Jede Methodenliste enthält nur eine Methode.
Siehe auch Wissenswertes zu Server-Methoden (Seite 290).

11. Übersetzen Sie das Projekt.
Dazu wählen Sie das Projekt aus und klicken dann in der Symbolleiste auf das folgende Symbol:

STEP 7 übersetzt das Projekt und aktualisiert die Datenbausteine, die zu der Client-Schnittstelle "Productionline" gehören.

HINWEIS

STEP 7 überschreibt beim Übersetzen sämtliche Daten in den Datenbausteinen, die zur Client-Schnittstelle gehören. Aus diesem Grund sollten Sie diese Datenbausteine weder manuell ergänzen noch korrigieren.

HINWEIS

Umbenennen von Knotennamen (DisplayNames)

Sie können in Leselisten, Schreiblisten und Methodenlisten den Namen eines Knotens über das Kontextmenü umbenennen. Im OPC UA Sprachgebrauch ist das der "DisplayName".

Wenn Sie den Namen des Knotens einer Methodenliste umbenennen und der Knoten bereits in einem programmierten Baustein für den Methodenaufruf "OPC_UA_MethodCall" verwendet wird, führt die Übersetzung des Projekts zu Konsistenzfehlern: Die UDTs der Methode werden bei der Übersetzung mit dem geänderten Namen neu erzeugt. Die im Programm verwendeten Referenzen auf die Methode stimmen dann nicht mehr.

Um die Konsistenzfehler zu beheben, können Sie entweder die Änderung des Namens der Methode in der Client-Schnittstelle rückgängig machen oder Sie navigieren zu dem Methodenaufruf und weisen dort unter "Eigenschaften > Bausteinparameter" (Register "Konfiguration") die betreffenden Parameter erneut zu.

Datenbausteine der Client-Schnittstelle

Folgende Datenbausteine gehören zur Client-Schnittstelle "Productionline":

- **Productionline_Configuration**
Ein Datenbaustein für die Konfiguration.
In dem Beispiel heißt dieser Datenbaustein "Productionline_Configuration".
Der Datenbaustein enthält bereits alle Systemdatentypen, die für die Anweisungen des OPC UA-Clients benötigt werden.
Außerdem enthält der Datenbaustein allgemeine Vorgabenwerte für die Parametrierung der Verbindung zu einem OPC UA-Server.
Wenn Sie mit der Verbindungsparametrierung arbeiten, wird dieser Datenbaustein gefüllt.

- **Produktionline_Data**
  Ein Datenbaustein für die PLC-Variablen, die Sie in dem Client-Schnittstellen-Editor eingegeben haben.
  In dem Beispiel heißt dieser Datenbaustein "Productionline_Data".
  Das folgende Bild zeigt den Datenbaustein.

Static	
ReadListProduct	Struct
Variable	"Productionline.ReadListProduct"
NewProduct	Bool
ProductNumber	Int
NodeStatusList	Array[0..1] of DWord
NodeStatusList[0]	DWord
NodeStatusList[1]	DWord
TimeStamps	Array[0..1] of LDT
TimeStamps[0]	LDT
TimeStamps[1]	LDT
WriteListStatus	Struct
Variable	"Productionline.WriteListStatus"
ProductionEnabled	Bool
NodeStatusList	Array[0..0] of DWord
NodeStatusList[0]	DWord
MethodListOpenDoor	Struct
MethodStatusList	Array[0..0] of DWord
MethodStatusList[0]	DWord
MethodResultList	Array[0..0] of DWord
MethodResultList[0]	DWord
Method	Struct
Inputs	"Productionline.MethodListOpendoor.Method.Inputs"
Number	Int
Outputs	"Productionline.MethodListOpendoor.Method.Outputs"
Result	Int

Bild 11-74 Datenbaustein "Productionline_Data"

Den Datenbaustein "Productionline_Data" verwenden Sie in Ihrem Anwenderprogramm und greifen auf die gelesenen Werte der PLC-Variablen "NewProduct" und "ProductNumber" zu. Im folgenden Abschnitt ist dieser Sachverhalt beispielhaft erläutert.

PLC-Variablen der Client-Schnittstelle lesen und schreiben

Beispiel: Wert "ProductNumber" lesen

In einem SCL-Programm schreiben Sie zum Beispiel:

```
#MyLocalVariable :=
"Productionline_Data".ReadListProduct.Variable.ProductNumber;
```

Damit weisen Sie z. B. der lokalen Variablen "#MyLocalVariable" die Nummer des Rohlings zu, der gerade in der Fertigungslinie hergestellt wurde.

Voraussetzungen:

- Eine Verbindung besteht zum OPC UA-Server der CPU, welche die Fertigungslinie steuert
- Der OPC UA-Client hat die aktuellen Werte gelesen

Deshalb überprüfen Sie, ob ein gelesener Wert gültig (valide) ist:

- Prüfen Sie, ob der Wert in "Productionline_Data".ReadListProduct.NodeStatusList[1] gleich 0 ist
- Optional: Prüfen Sie, wann dieser Wert vom OPC UA-Server gelesen wurde. Dieser Wert steht in "Productionline_Data".ReadProduct.TimeStamps[1]. Wenn keine Zeitstempel angefordert werden, verringert sich die Kommunikationslast.

Beispiel: Wert "ProductEnabled" schreiben

Mit dem Datenbaustein übertragen Sie zum OPC UA-Server neue Werte für PLC-Variablen, in dem Beispiel für die Variable "ProductionEnabled".

Mit der folgenden Zuweisung geben Sie in der Beispielanlage die Fertigungslinie frei:

```
"Productionline_Data".WriteListStatus.Variable.ProductionEnabled := TRUE;
```

Die Freigabe ist aber nur dann erfolgreich, wenn folgende Voraussetzungen erfüllt sind:

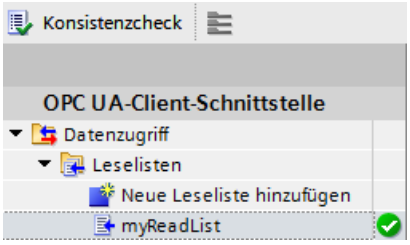
- Eine Verbindung besteht zum OPC UA-Server der CPU, welche die Fertigungslinie steuert
- Aktuelle Werte werden über den OPC UA-Client geschrieben

Konsistenzcheck

Zu guter Letzt prüfen Sie die Konsistenz der Lese-/Schreib- bzw. Methodenliste.

1. Markieren Sie die Liste, die Sie prüfen wollen
2. Klicken Sie auf die Schaltfläche "Konsistenzcheck" über dem Bereich "OPC UA-Client-Schnittstelle"

Eine fehlerfreie Zuordnung der Variablen bzw. Methoden zu den korrespondierenden Elementen der Server-Schnittstelle wird durch ein grünes Häkchen angezeigt.



Sie können davon ausgehen, dass der Datenaustausch zwischen Client und Server sowie Methodenaufrufe zur Laufzeit ohne Probleme funktionieren.

Im Fehlerfall wird eine Liste im Inspektorfenster angezeigt. Aus dieser Liste heraus können Sie zum jeweiligen Fehler springen.

STEP 7 prüft beim Konsistenzcheck:

- Sind alle Elemente, die Sie in der jeweiligen Liste verwenden, auch im Server vorhanden?
- Stimmen die verwendeten Datentypen überein?
- Bei Methoden: Stimmen Anzahl, Namen, Reihenfolge und Datentypen von Methoden-Argumenten überein?

11.4.6 Server-Schnittstelle online ermitteln

Mit STEP 7 (TIA Portal) können Sie die Schnittstelle eines OPC UA-Servers online ermitteln. Damit bringen Sie in Erfahrung, welche Variablen eines verbundenen OPC UA-Server Sie mit OPC UA-Clients lesen oder setzen (schreiben) können. Außerdem erfahren Sie, welche Server-Methoden der OPC UA-Server für OPC UA-Clients bereitstellt.

Wenn Sie offline arbeiten, können Sie die Schnittstelle des OPC UA-Servers über eine OPC UA XML-Datei erstellen. In der OPC UA XML-Datei ist der Adressraum des Servers beschrieben, siehe: OPC UA-XML-Datei exportieren (Seite 232).

Online Server-Schnittstellen ermitteln

Um eine Server-Schnittstelle online zu ermitteln, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Projektnavigation von STEP 7 die CPU aus, die als OPC UA-Client projektiert ist (im Beispiel Supervisor).
2. Wählen Sie die Client-Schnittstelle aus (im Beispiel OPC UA-Kommunikation > Client-Schnittstellen > Productionline).
   Falls noch keine Client-Schnittstelle angelegt wurde, dann doppelklicken Sie auf "Neue Client-Schnittstelle hinzufügen".
3. Doppelklicken Sie auf die ausgewählte Client-Schnittstelle.
   Der Editor für Client-Schnittstellen wird angezeigt:

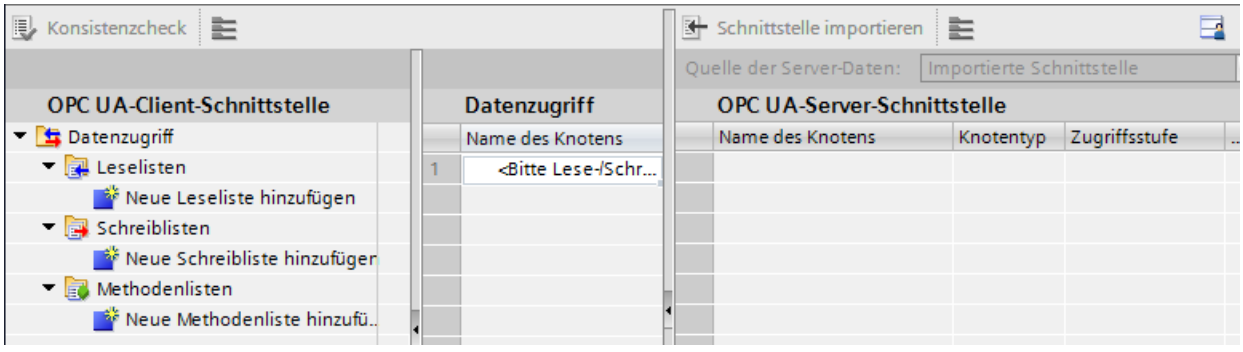


Bild 11-75 Editor für Client-Schnittstelle

4. Klicken Sie im linken Feld des Editors auf "Neue Leseliste hinzufügen", "Neue Schreibliste hinzufügen" oder auf "Neue Methodenliste hinzufügen".
5. Wählen Sie im rechten Feld des Editors bei "Quelle der Server-Daten" als Datenquelle "Online[]":

6. Klicken Sie auf die Schaltfläche "Online-Zugänge".
   STEP 7 zeigt den Dialog "Mit OPC UA-Server verbinden":

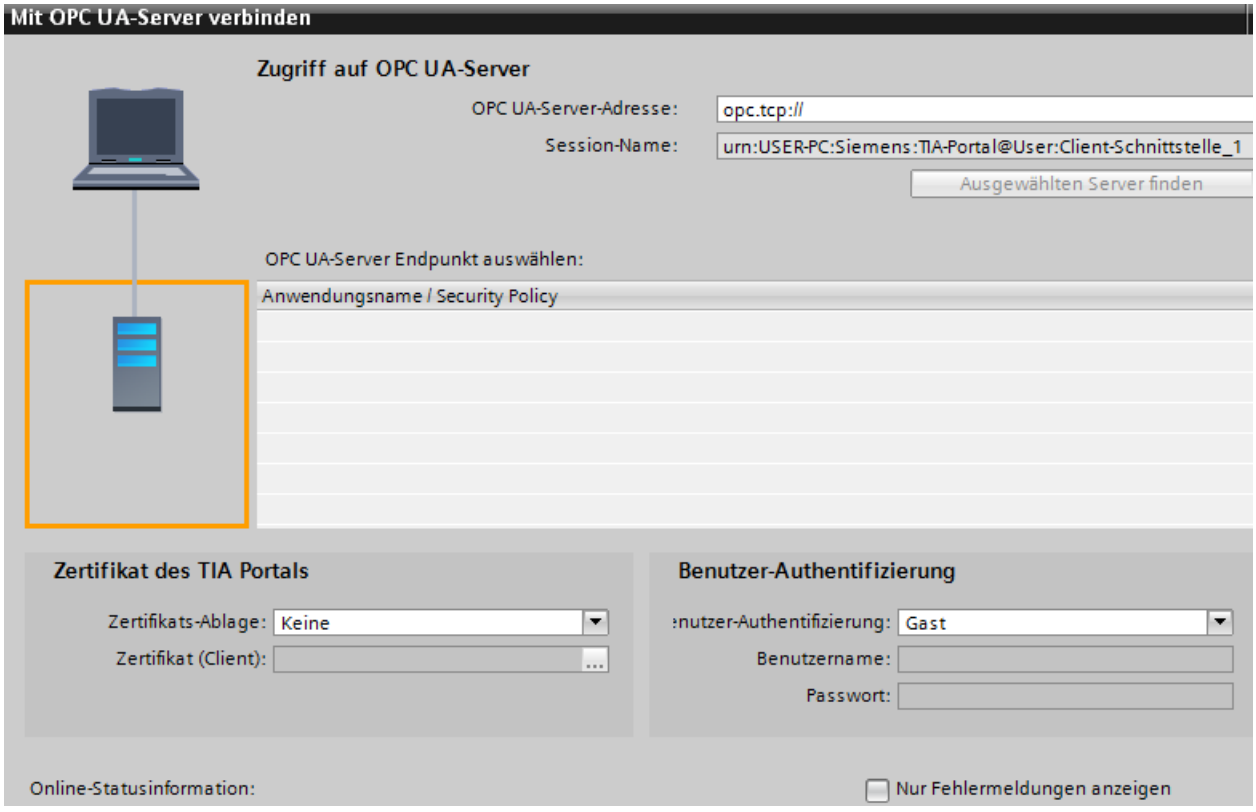


Bild 11-76 Dialog "Mit OPC UA-Server verbinden"

Tipp: Wenn Sie erstmals eine Online-Verbindung zu einem OPC UA-Server aufbauen, nutzen Sie die Schaltfläche "Online-Zugänge". Beim erneuten Verbinden nach einem Verbindungsabbruch wählen Sie die Schaltfläche "Mit Online-Server verbinden" neben dem Auswahlfeld "Online".

Tragen Sie oben rechts die IP-Adresse des OPC UA-Servers ein, dessen Server-Schnittstelle Sie online ermitteln wollen.

7. Klicken Sie auf die Schaltfläche "Ausgewählten Server finden".
   STEP 7 baut eine Verbindung zum OPC UA-Server auf und ermittelt alle Sicherheitseinstellungen (Server-Endpoints), die der Server bereithält.
   STEP 7 zeigt die Endpunkte als Liste an:

Bild 11-77 Gefundener OPC UA-Server mit allen Server-Endpoints

8. Klicken Sie auf den Endpunkt, den Sie für eine Verbindung von STEP 7 zum OPC UA-Server nutzen wollen.

9. Wollen Sie eine gesicherte Verbindung verwenden?
   – Falls Sie einen sicheren Endpunkt ausgewählt haben, dann wählen Sie jetzt bei "Zertifikats-Ablage" den Eintrag "TIA Portal" aus.
     Und bei "Zertifikat (Client)" wählen Sie ein Client-Zertifikat für Ihren PC aus, auf dem STEP 7 (TIA Portal) gerade ausgeführt wird.
     Wenn noch kein Client-Zertifikat für Ihren PC vorliegt, dann können Sie hier im TIA Portal ein Client-Zertifikat erzeugen.
     Um ein Zertifikat für Ihren PC zu erzeugen, gehen Sie folgendermaßen vor:
     - Klicken Sie auf die Schaltfläche im Eingabefeld "Zertifikat (Client)".
     - Klicken Sie auf die Schaltfläche "Hinzufügen".
     - Bei "Zertifikatsinhaber" tragen Sie "STEP 7 (TIA Portal)" ein.
     - Bei "Verwendungszweck" wählen Sie den Eintrag "OPC UA-Client".
     - Bei "Alternativer Name des Zertifikatsinhabers (SAN)" tragen Sie unter "Wert" die IP-Adresse Ihres PCs ein, auf dem Sie gerade STEP 7 (TIA Portal) ausführen. Überschreiben Sie die bereits eingetragene IP-Adresse.
     - Falls Ihr PC eine zweite IP-Adresse verwendet, so tragen Sie diese ebenfalls ein. Wenn Ihr PC keine zweite IP-Adresse verwendet, dann löschen Sie die zweite bereits eingetragene IP-Adresse.
     - Klicken Sie auf die Schaltfläche "OK".
   – Falls Sie keinen sicheren Endpunkt ausgewählt haben, dann behalten Sie die Voreinstellung ("Keine") bei.
10. Wie wollen Sie sich anmelden?
   – Falls Sie sich als Gast beim OPC UA-Server anmelden möchten, dann übernehmen Sie die Voreinstellung bei "Benutzer-Authentifizierung".
   – Falls Sie sich mit Benutzernamen und Passwort anmelden möchten, dann wählen Sie "Benutzername und Passwort".
     Verwenden Sie den Benutzernamen und das Passwort, das bei der Konfiguration des OPC UA-Servers hinterlegt wurde, in den Eigenschaften der CPU, unter "Allgemein > OPC UA > Server > Security > Benutzer-Authentifizierung > Benutzerverwaltung".
11. Klicken Sie auf die Schaltfläche "Verbinden".
    Bei einer gesicherten Verbindung erscheint noch eine Meldung, dass Sie das Server-Zertifikat akzeptieren müssen, damit die sichere Verbindung zustande kommt. Im Meldungsfenster können Sie sich die weiteren Details zum Server-Zertifikat über einen Link anzeigen lassen.
    Dieses Standard-Windows-Fenster liefert nur Informationen zum Server-Zertifikat. Wenn Sie auf über die Schaltfläche zum Installieren des Server-Zertifikats klicken, wird das Server-Zertifikat aber nicht in den Zertifikatsspeicher des TIA Portals aufgenommen, d. h. beim nächsten Verbindungsversuch werden Sie erneut aufgefordert, das Server-Zertifikat zu akzeptieren.
    STEP 7 baut anschließend eine Verbindung zum OPC UA-Server auf und zeigt wieder den Editor für Client-Schnittstellen an.
    Im rechten Feld des Editors stellt STEP 7 die oberste Ebene des Adressraums des OPC UA-Servers dar:

12. Klicken Sie auf das kleine schwarze Dreieck neben "Objects".
    STEP 7 zeigt nun auch die Ebene unterhalb von Objects an.
13. Klicken Sie auf das kleine schwarze Dreieck neben "Productionline".
    STEP 7 zeigt nun auch die Ebene unterhalb von Productionline an.

14. Öffnen Sie nun weitere unterlagerte Ordner:

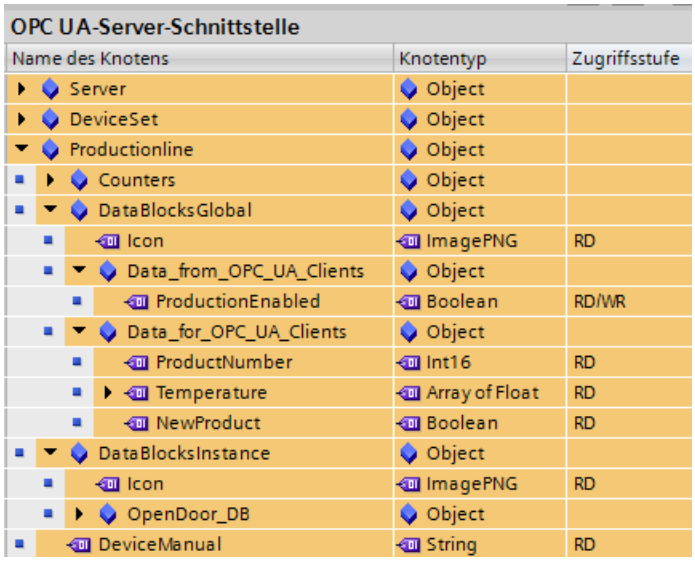


Bild 11-78 Onlineansicht OPC UA-Server-Schnittstelle

Weitere Informationen

Informationen zum Mappen von Datentypen finden Sie im Kapitel Mapping von Datentypen (Seite 179).
Informationen, wie Sie eine Client-Schnittstelle anlegen, finden Sie im Kapitel Client-Schnittstellen anlegen (Seite 336).

11.4.7 Mehrsprachige Texte nutzen

Im Editor für Client-Schnittstellen importieren Sie mit den OPC UA-XML-Dateien (Informationsmodelle) auch Texte, die in verschieden Sprachen angezeigt werden können. Die Mehrsprachigkeit ist optional, jeder Knoten (Node) kann hinsichtlich der angebotenen Sprachen unterschiedlich definiert sein.
In der XML-Datei sind das folgende Felder, die für unterschiedlichen Sprachen vorbereitet sein können:

- DisplayName
- Description

Beispiel für mehrsprachig definierte Texte in einer OPC UA-XML-Datei

In der dargestellten XML-Datei sind z. B. der Displayname und die Description sowohl mit einem "Default"-Text und mehreren lokalisierten Texten eingetragen.

- Default-Text ist jeweils der erste Eintrag ohne Lokalisierungsinformation
- Lokalisierter Text ist jeweils der Text hinter "Locale=" gefolgt von einem Sprachkürzel, z. B. "it-IT" für italienisch

```
<UAVariable NodeId="ns=3;i=6070" BrowseName="3:EngineeringRevision" ParentNodeId="ns=3;i=1002"
DataType="String">
    <DisplayName>EngineeringRevision</DisplayName>
    <DisplayName Locale="en-US">EngineeringRevision</DisplayName>
    <DisplayName Locale="de-DE">Revisionsstand</DisplayName>
    <Description>Revision Level of the engineering environment.</Description>
    <Description Locale="en-US">Revision Level of the engineering environment.</Description>
    <Description Locale="de-DE">Revisionsstand der Engineeringumgebung.</Description>
    <Description Locale="fr-FR">Niveau de révision de l'environnement d'ingénierie.</Description>
    <Description Locale="it-IT">Livello di revisione dell'ambiente di ingegneria.</Description>
<References>
    <Reference ReferenceType="HasTypeDefinition">i=68</Reference>
    <Reference ReferenceType="HasModellingRule">i=78</Reference>
</References>
</UAVariable>
```

Bild 11-79 Beispiel für mehrsprachig definierten Text in einer OPC UA-XML-Datei

Anzeige mehrsprachiger Texte

Beim Importieren einer Server-Schnittstelle werden die verfügbaren mehrsprachigen Texte intern gespeichert und auch mit dem Projekt in eine CPU geladen.

Den Text aus der OPC UA-XML-Datei zeigt der Client-Editor in den Spalten "Name des Knotens" (entspricht "DisplayName") und "Beschreibung" (entspricht "Description") an.

Welche Sprache für einen Knoten angezeigt wird, hängt von folgenden kaskadierenden Regeln ab:

- Wenn der Knoten Text in der aktuell verwendeten Editiersprache enthält, wird der Text auch in der Editiersprache angezeigt.
  (Einstellung der Editiersprache: In der Projektnavigation wählen Sie den Bereich "Sprachen & Ressourcen > Projektsprachen")
- Wenn der Knoten keinen Text in der Editiersprache enthält, aber dort ein Default-Text definiert ist (ohne Sprachkürzel), dann wird der Default-Text angezeigt.
- Spalte "Name des Knotens": Wenn auch kein Default-Text definiert ist, aber ein Text in irgendeiner anderen Sprache, dann wird der DisplayName-Text in der ersten vorliegenden Sprache angezeigt. Für Beschreibungstexte gilt diese Regel nicht.
- Wenn keine der oben genannten Bedingungen erfüllt ist, wird kein Text angezeigt.

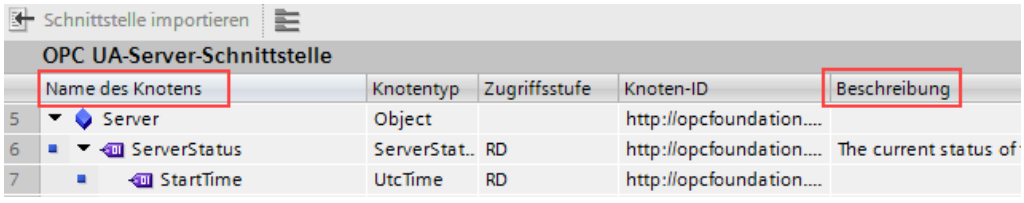


Bild 11-80 Anzeige für mehrsprachige Texte

Wenn Sie die Editiersprache wechseln, wechselt auch der mehrsprachige Text nach den oben erläuterten Regeln in der importierten Schnittstelle.

Die Knoten können Sie dann in die entsprechenden Listen (Leseliste, Schreibliste, Methodenliste) per Drag&Drop übernehmen.

In den Listen (Leseliste, Schreibliste, Methodenliste) ist kein Sprachwechsel möglich.

Übernahme der angezeigten Beschreibungstexte als Kommentar in PLC-Datentypen

Wenn Sie das Programm übersetzen, erzeugt STEP 7 automatisch PLC-Datentypen (UDTs) für jede Leseliste, für jede Schreibliste und für Eingänge bzw. Ausgänge jeder Methode. Diese UDTs haben jeweils ein Element für jeden Knoten.

Die UDTs übernehmen den Beschreibungstext als Kommentar nach den oben erläuterten Regeln. STEP 7 erzeugt den Kommentar nur in einer Sprache, wie auch die Texte in der OPC UA-Server-Schnittstelle nur in einer Sprache angezeigt werden können.

11.4.8 Regeln für den Zugriff auf Strukturen

Im Folgenden sind die Regeln für den Zugriff auf Strukturen erläutert. Beachten Sie diese Regeln beim Lesen und Schreiben von Werten kompletter Strukturen, die ein OPC UA-Server zur Verfügung stellt.

Wie der Client der S7-1500 CPU auf Strukturen zugreift

Der OPC UA-Client der S7-1500 CPU nutzt für den performanten Zugriff auf Strukturen zur Laufzeit weder TypeDictionaries noch DataTypeDefinition-Attribute, die ein Server zur Auflösung dieser Strukturen anbietet.

Daher sind die Möglichkeiten des OPC UA-Clients zur Überprüfung von Strukturelementen zur Laufzeit des Anwenderprogramms im Client begrenzt.

Regeln für den Zugriff auf Strukturen

Wenn Sie zur Projektierung der Lese- und Schreiblisten die Client-Schnittstellen verwenden (Verbindungsparametrierung) und die PLC-Datentypen passend zum importierten bzw. online ermittelten Adressmodell des Servers zuweisen, dann funktionieren Lese- und Schreibzugriffe auf Strukturen zur Laufzeit problemlos.

Denn die Projektierung mithilfe der Client-Schnittstelle sorgt automatisch dafür, dass die Reihenfolge und die Datentypen der Strukturelemente client- und server-seitig aufeinander abgestimmt sind.

Empfehlung: Aktualisieren Sie eine S7-1500 CPU (als Server) auf die aktuellen Firmware-Version (z. B. V2.0 > V2.5.2 oder höher).

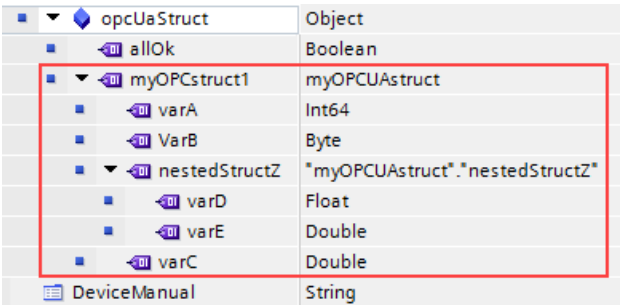
Zur Laufzeit überprüft der OPC UA-Client nur die Gesamtlänge des übermittelten Werts; detailliertere Prüfungen sind nicht möglich.

In Strukturen sind auch Strings (WSTRING, STRING und OPC UA ByteString) erlaubt. Strings haben zwar eine variable Länge, aber OPC UA beherrscht diese Variabilität: Bei der Übertragung wird jedem String ein Längenfeld vorangestellt, in dem die Länge des Strings kodiert ist. Eine S7-1500 CPU als OPC UA Client kann daher die Länge eines Strings prüfen und ermitteln, ob der String in die zugeordnete CPU-Variable "hineinpasst". Damit kann die CPU auch die Gesamtlänge der Struktur prüfen.

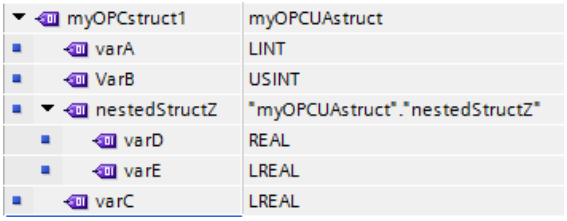
Für die Zuweisung von OPC UA-Strukturen zu PLC-Variablen bzw. DB-Variablen gelten die Mapping-Regeln (siehe Mapping von Datentypen (Seite 179)).

Beispiel für eine fehlerfreie Zuweisung von Strukturelementen

In der importierten Nodeset-Datei (XML-Export) ist die Struktur folgendermaßen definiert:



Die in der Leseliste abgebildete Struktur stimmt sowohl in der Reihenfolge als auch von den zugewiesenen Datentypen mit dem entsprechenden Knoten der Nodeset-Datei überein.



Wenn die Struktur sich jetzt auf dem Server ändert, z. B. varA und varB werden vertauscht, und die Leseliste im Client bleibt gleich, dann stimmt die Zuweisung nicht mehr:

- Die Gesamtlänge der Daten bleibt gleich (es hat sich nur die Reihenfolge geändert)
- Der Aufbau der Struktur ist bei Client und Server unterschiedlich!

⚠ WARNUNG

Keine Fehlermeldung bei unterschiedlichem Strukturaufbau zwischen Client und Server

Wenn die Strukturen bei Client und Server nicht übereinstimmen, dann erzeugt diese Regelverletzung möglicherweise keinen Fehler beim Übersetzen und auch nicht zur Laufzeit.

Achten Sie darauf, dass sich die projektierten Zuweisungen bei Strukturen zur Laufzeit nicht ändern. Projektieren Sie bei Bedarf die Zuweisung in den Lese- und Schreiblisten neu!

11.4.9 Verbindungsparametrierung nutzen

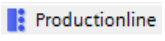
11.4.9.1 Verbindungen anlegen und parametrieren

Mit den Anweisungen für OPC UA-Clients erstellen Sie ein Anwenderprogramm, das Daten mit einem OPC UA-Server austauscht. Dafür sind eine Reihe von Systemdatentypen erforderlich.
Um die Arbeit mit diesen Systemdatentypen zu vereinfachen, verfügt STEP 7 (TIA Portal) ab der Version 15.1 über eine Verbindungsparametrierung für OPC UA-Clients.
Die Verbindungsparametrierung ist eine Option, die Sie nutzen können, aber nicht müssen. Sie können die erforderlichen Systemdatentypen auch manuell anlegen.
Damit die Beschreibung nachvollziehbar wird, verwenden wir ein Beispiel, siehe Beschreibung des Beispiels (Seite 335).

Verbindungsparametrierung öffnen

Um die Verbindung zu einem OPC UA-Server zu parametrieren, gehen Sie folgendermaßen vor:

1. Doppelklicken Sie im Bereich "OPC UA-Kommunikation" in der Projektnavigation auf die Client-Schnittstelle, die Sie parametrieren wollen.
   Für die Beispiel-Konfiguration: Doppelklicken Sie auf die Client-Schnittstelle "Productionline":

   Wie Sie eine Client-Schnittstelle anlegen, ist im Kapitel "Client-Schnittstelle anlegen (Seite 336)" beschrieben.
2. Klicken Sie auf das Register "Eigenschaften" (Inspektorfenster), falls das Register nicht bereits angezeigt wird.
   STEP 7 zeigt nun die Verbindungsparametrierung für die Anweisungen des OPC UA-Clients an.
   Das Register "Allgemein" ist geöffnet.
3. Klicken Sie auf das Register "Konfiguration" und stellen Sie die Verbindung zum OPC UA-Server ein.

Verbindungsparameter einstellen

1. Wählen Sie einen aussagekräftigen Namen für die Session. Für das Beispiel wählen Sie den Namen "OPC UA Connection to Productionline".
2. Tragen Sie im Feld "Adresse" die IP-Adresse des OPC UA-Servers ein, zu dem Ihr Anwenderprogramm, das als OPC UA-Client arbeitet, eine Verbindung aufbauen soll. In der Beispielkonfiguration besitzt die CPU, die die Fertigungslinie steuert, die IP-Adresse "192.168.1.1". Es soll eine Verbindung zu dem OPC UA-Server dieser CPU aufgebaut werden, deshalb tragen Sie diese IP-Adresse im Feld "Adresse" ein. Der OPC UA-Server verwendet in diesem Fall den Standard-Port 4840.
   Alternativ können Sie in Feld "Adresse" einen gültigen DNS-Namen eingeben. Die Länge des DNS-Namens ist auf 242 Zeichnen beschränkt.
   Wenn die Adresse nicht gültig ist, wird die Fehlermeldung angezeigt: "Geben Sie eine gültige Adresse ein".
   Wenn die Zeichenfolge der Felder "Adresse", "Port" und "Pfad" mehr als 254 Zeichen umfasst, erscheint ebenfalls eine Fehlermeldung.

3. Tragen Sie einen Pfad innerhalb des OPC UA-Servers ein, um den Zugriff auf diesen Pfad zu beschränken. Die Angabe ist optional. Einige Server bauen jedoch nur dann eine Verbindung auf, wenn ein Server-Pfad angegeben ist.
   Wenn Sie einen Pfad angeben, wird dieser automatisch an den Eintrag "ServerEndpointUrl" im Konfigurations-DB für die Client-Schnittstelle eingetragen. Der Eintrag besteht dann aus den Komponenten "OPC-Schema Präfix", "IP-Adresse", "Port-Nummer" und "Server-Pfad", z. B.: "opc.tcp://192.168.0.10:4840/Beispiel/Pfad".
   Das folgende Bild zeigt die Eingabe der IP-Adresse des OPC UA-Servers:

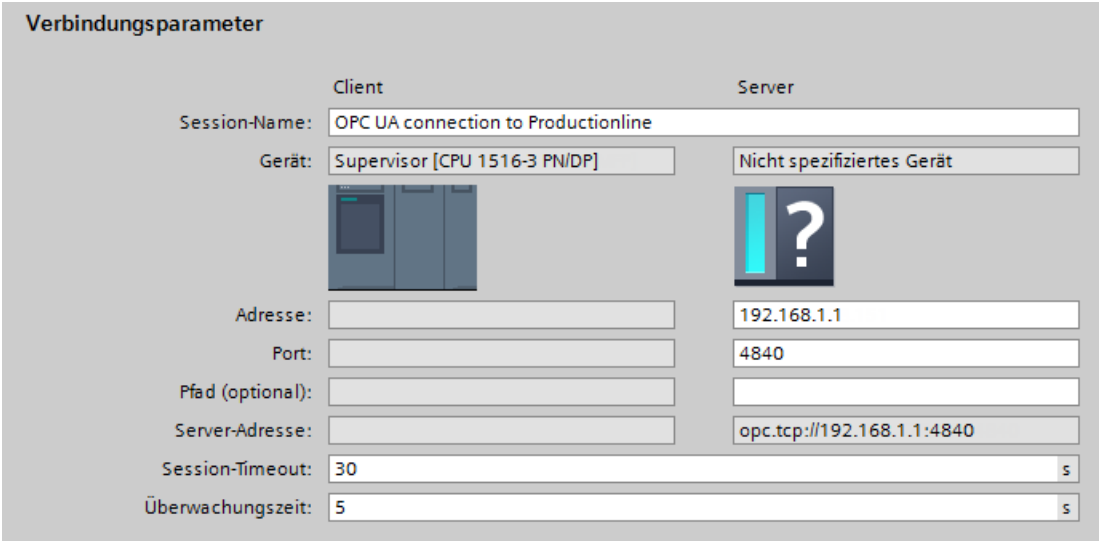


Bild 11-81 Verbindungsparameter

4. Falls der OPC UA-Server nicht den Standard-Port 4840 verwendet, müssen Sie hier die Portnummer einfügen.
   Tragen Sie in das Feld zum Beispiel die Nummer 65535 ein, falls der OPC UA-Server, zu dem Sie eine Verbindung aufbauen wollen, diese Portnummer verwendet.
5. Außerdem übernehmen Sie die Voreinstellungen für "Session-Timeout" (30 Sekunden) und "Überwachungszeit" (5 Sekunden).

Security Parameter einstellen

1. Klicken Sie auf den Bereich "Security" im Register "Konfiguration".
   In diesem Bereich befinden sich alle Sicherheitseinstellungen für die Verbindung zum OPC UA-Server.

Folgende Einstellungen sind z. B. möglich:

Bereich "Allgemein"

Security-Modus:

Wählen Sie aus der Klappliste den Sicherheitsmodus, den die Verbindung zum OPC UA-Server erfüllen muss.

Wenn der Server den gewählten Modus nicht erfüllt, dann wird keine Session aufgebaut.

Die folgenden Einstellungen stehen zur Wahl:

- Keine Security: Keine gesicherte Verbindung!
- Signieren: OPC UA-Server und -Client signieren die Datenübertragung (alle Nachrichten): Manipulationen sind so erkennbar.
- Signieren & Verschlüsseln: OPC UA-Server und -Client signieren und verschlüsseln die Datenübertragung (alle Nachrichten):

Security Policy:

Stellen Sie die Verschlüsselungstechniken für das Signieren und Verschlüsseln von Nachrichten ein.

Die folgenden Einstellungen sind möglich:

- Keine Security
- Basic128Rsa15
- Basic256
- Basic256Sha256

Wenn Sie eine gesicherte Verbindung konfigurieren, dann müssen Sie die folgenden Punkte beachten:

- Für eine gesicherte Verbindung ist ein Zertifikat für den Client erforderlich.
- Dieses Client-Zertifikat müssen Sie dem Server bekanntmachen.

Wie Sie dazu vorgehen, ist im Kapitel "Handling der Client- und Server-Zertifikate (Seite 241)" unter "Zertifikat des OPC UA-Clients" beschrieben.

Bereich "Zertifikate"

Client-Zertifikat:

Das Zertifikat bestätigt die Authentizität des OPC UA-Clients.

Um ein Zertifikat auszuwählen, klicken Sie dazu auf das folgende Symbol:

STEP 7 zeigt eine Liste mit Zertifikaten an.

Wählen Sie davon das Zertifikat aus, das Sie dem Server bekanntgemacht haben.

Klicken Sie auf das Symbol mit dem grünen Häkchen:

Oder erzeugen Sie ein neues Zertifikat. Dazu klicken Sie auf das Symbol "Hinzufügen".

Wenn Sie ein neues Zertifikat erzeugen, dann müssen Sie dieses Zertifikat dem Server bekanntmachen.

Bereich "Benutzer-Authentifizierung"

Bei Benutzer-Authentifizierung sind die folgenden Einstellungen möglich:

- Gast
- Benutzername und Passwort
- Benutzer (TIA Portal - Security-Einstellungen)

Weitere Informationen siehe Benutzer und Rollen mit OPC UA-Funktionsrechten (Seite 251).

Sprachen einstellen

UA-Variablen vom Typ String können bei OPC UA lokalisiert werden, d. h. Texte (Werte für die UA-Variable) können beim Server in verschiedenen Landessprachen vorliegen. Zum Beispiel können für DisplayName (Name des Knotens) und Description (Beschreibung) lokalisierte Texte vorliegen.

Im Bereich "Sprachen" des Registers "Konfiguration" haben Sie die Möglichkeit, die Sprache der vom Server zurückgelieferten Texte in folgender Weise zu beeinflussen:

Geben Sie im Bereich "Sprachen" eine Reihenfolge von Sprachen ein, die der Server beim Verbindungsaufbau dem Client übermittelt.

Die Sprache bzw. die damit verbundene Lokale ID ("Sprachkürzel"), die Sie in der ersten Zeile eingeben, ist die vom Client bevorzugte Sprache.

- Wenn der Server die UA-Variable in der gewünschten Sprache liefern kann, dann wird sie dem Client übermittelt.
- Wenn der Server die UA-Variable nicht in der gewünschten Sprache liefern kann, dann prüft er, ob er die UA-Variable in der Sprache liefern kann, die Sie in der zweiten Zeile eingegeben haben (1. Ersatzsprache).
- So arbeitet der Server die Liste ab und wenn er weder die gewünschte Sprache noch eine Ersatzsprache liefern kann, dann liefert er die Default-Sprache.

Weitere Informationen

Welche Ursachen gibt es, wenn die Verbindung zu einem OPC UA-Server fehlschlägt? FAQ. (https://support.industry.siemens.com/cs/ww/de/view/109766709)

11.4.9.2 Handling der Client-Zertifikate der S7-1500 CPU

Woher kommt das Zertifikat des Clients?

Wenn Sie den OPC UA-Client einer S7-1500 CPU nutzen (OPC UA-Client aktiviert), dann können Sie mit STEP 7 ab V15.1 für diese Clients Zertifikate erzeugen wie es in den folgenden Abschnitten beschrieben ist.

Wenn Sie UA-Clients von Herstellern oder der OPC Foundation verwenden, wird bei der Installation oder beim ersten Programmaufruf automatisch ein Client-Zertifikat erzeugt. Diese Zertifikate müssen Sie über den globalen Zertifikatsmanager in STEP 7 importieren und für die jeweilige CPU verwenden.

Wenn Sie selbst einen OPC UA-Client programmieren, dann können Sie Zertifikate programmtechnisch erstellen. Oder Sie erzeugen Zertifikate mit Tools, zum Beispiel mit OpenSSL oder dem Zertifikate-Generator der OPC Foundation:

- Wie Sie bei OpenSSL vorgehen, lesen Sie hier: "PKI-Schlüsselpaare und Zertifikate selbst erzeugen (Seite 188)".
- Wie Sie mit dem Zertifikate-Generator der OPC Foundation arbeiten, lesen Sie hier: "Selbstsignierte Zertifikate erzeugen (Seite 187)".

Zertifikat des OPC UA-Clients der S7-1500 CPU

Eine gesicherte Verbindung zwischen OPC UA-Server und einem OPC UA-Client kommt nur dann zu Stande, wenn der Server das Zertifikat des Clients als vertrauenswürdig einstuft.

Dazu müssen Sie dem Server das Client-Zertifikat bekanntmachen.

Die folgenden Abschnitte beschreiben, wie Sie zunächst für den OPC UA-Client der S7-1500 CPU ein Zertifikat erzeugen und es dann dem Server zur Verfügung stellen.

1. Zertifikat für den Client erzeugen und exportieren

Für eine gesicherte Verbindung müssen Sie ein Client-Zertifikat erzeugen und - falls Server und Client sich in unterschiedlichen Projekten befinden - das Zertifikat exportieren.

Wenn Client und Server sich im selben Projekt befinden, entfällt das Exportieren des Client-Zertifikats und das anschließende Importieren.

Voraussetzungen

Die IP-Schnittstelle der CPU ist konfiguriert, eine IP-Adresse vorhanden.

Hintergrund: Unter "Alternativer Name des Zertifikatsinhabers (SAN)" wird die IP-Adressen eingetragen, unter der die CPU in Ihrer Anlage erreichbar ist.

OPC UA-Client-Zertifikat erzeugen

Die einfachste Möglichkeit, für eine S7-1500 CPU ein Client-Zertifikat zu erzeugen, ist über die Projektierung einer Client-Schnittstelle.

Die Projektierung der Client-Schnittstelle sieht die Auswahl oder Erzeugung eines Client-Zertifikats vor, siehe Verbindungen anlegen und parametrieren (Seite 350).

Alternativ können Sie das Client-Zertifikat auch folgendermaßen erzeugen:

1. In der "Projektnavigation" wählen Sie die CPU aus, die als Client fungiert.
2. Doppelklicken Sie auf "Gerätekonfiguration"
3. In den Eigenschaften der CPU klicken Sie auf "Schutz & Security > Zertifikatsmanager".
4. In der Tabelle "Gerätezertifikate" doppelklicken Sie auf "<neu hinzufügen>".
   STEP 7 öffnet einen Dialog.
5. Klicken Sie auf die Schaltfläche "Hinzufügen".
6. Bei "Verwendungszweck" wählen Sie aus der Liste den Eintrag "OPC UA-Client".
7. Klicken Sie auf "OK".
   STEP 7 zeigt nun das Client-Zertifikat in der Tabelle "Gerätezertifikate" an.
8. Falls der Server in einem anderen Projekt liegt: Klicken Sie mit der rechten Maustaste auf diese Zeile und wählen Sie das Kontextmenü "Zertifikat exportieren".
9. Wählen Sie ein Verzeichnis aus, in dem Sie das Client-Zertifikat speichern.

2. Das Client-Zertifikat dem Server bekanntmachen

Das Client-Zertifikat müssen Sie dem Server zur Verfügung stellen, damit eine gesicherte Verbindung aufgebaut werden kann.

Dazu gehen Sie folgendermaßen vor:

1. Falls der Client in einem anderen Projekt konfiguriert wurde und Sie das Client-Zertifikat dort erstellt und exportiert haben:
   - Aktivieren Sie im lokalen Zertifikatsmanager des Servers die Option "Globale Security-Einstellungen für den Zertifikatsmanager verwenden". Dadurch ist der globale Zertifikatsmanager verfügbar.
     Sie finden diese Option in den Eigenschaften der CPU, die als Server dient, unter "Schutz & Security > Zertifikatsmanager".
   - Falls dieses Projekt noch nicht geschützt ist, dann klicken Sie in der Projektnavigation von STEP 7 unter "Security-Einstellungen > Einstellungen" auf die Schaltfläche "Dieses Projekt schützen" und melden Sie sich an.
     STEP 7 zeigt nun in der Projektnavigation unter "Security-Einstellungen" der Eintrag "Globale Security-Einstellungen" an.
   - Doppelklicken Sie auf "Globale Security-Einstellungen".
   - Doppelklicken Sie auf "Zertifikatsmanager".
     STEP 7 öffnet den globalen Zertifikatsmanager.
   - Klicken Sie auf das Register "Gerätezertifikate".
   - Klicken Sie mit der rechten Maustaste im Register auf eine freie Fläche (nicht auf ein Zertifikat).
   - Wählen Sie das Kontextmenü "Importieren".
     Der Dialog zum Importieren von Zertifikaten wird angezeigt.
   - Wählen Sie das Client-Zertifikat aus, dem der Server vertrauen soll.
   - Klicken Sie auf die Schaltfläche "Öffnen", um das Zertifikat zu importieren.
     Das Zertifikat des Clients ist nun im globalen Zertifikatsmanager enthalten. Merken Sie sich die ID des gerade importierten Client-Zertifikats.
2. Klicken Sie nun in den Eigenschaften der CPU, die als Server dient, auf das Register "Allgemein".
3. Klicken Sie auf den Bereich "OPC UA > Server > Security > Secure Channel".
4. Scrollen Sie im Dialog "Secure Channel" nach unten zum Abschnitt "Vertrauenswürdige Clients".
5. Doppelklicken Sie in der Tabelle auf die leere Zeile mit "<neu hinzufügen>". In der Zeile wird eine Schaltfläche mit drei Punkten angezeigt.
6. Klicken Sie auf diese Schaltfläche.
7. Wählen Sie das vorbereitete Client-Zertifikat aus.
8. Klicken Sie auf die Schaltfläche mit dem grünen Häkchen.
9. Übersetzen Sie das Projekt.
10. Laden Sie die Konfiguration in die S7-1500 CPU (Server).

Ergebnis

Der Server vertraut nun dem Client. Wenn außerdem das Server-Zertifikat als vertrauenswürdig gilt, dann können Server und Client eine gesicherte Verbindung aufbauen.

11.4.9.3 Authentifizierung des Benutzers

Sie können in der OPC UA-Client-Schnittstelle der S7-1500 einstellen, wie sich ein Benutzer des OPC UA-Clients legitimieren muss, wenn er auf den Server zugreifen will. Dazu müssen Sie in der Projektnavigation der gewünschten S7-1500 CPU unter "OPC UA-Kommunikation > Client-Schnittstellen" die entsprechende Client-Schnittstelle auswählen und im Inspektorfenster unter "Eigenschaften > Konfiguration > Security" die Art der Benutzer-Authentifizierung auswählen.

Arten der Benutzer-Authentifizierung

Für die Benutzer-Authentifizierung stehen Ihnen folgenden Möglichkeiten zur Verfügung:

- **Gast**
  Der Benutzer muss seine Berechtigung nicht nachweisen (anonymer Zugang). Die CPU erzeugt zu diesem Zweck für den Anwender eine anonyme Session und der OPC UA-Server überprüft nicht die Berechtigung des Client-Anwenders.
- **Benutzername und Passwort**
  Der Benutzer muss seine Berechtigung nachweisen (kein anonymer Zugang). Der OPC UA-Server überprüft, ob der Client-Anwender berechtigt ist, auf den Server zuzugreifen. Als Nachweis gilt der Benutzername mit dem richtigen Passwort. Diese Eingaben können nicht von der Client-Schnittstelle überprüft werden, daher werden hier alle Werte als gültig akzeptiert.

  HINWEIS

  Benutzername und Passwort speichert STEP 7 unverschlüsselt im Datenbaustein/Instanzdatenbaustein. Empfehlung: Verwenden Sie die Benutzer-Authentifizierung für das TIA-Projekt "Benutzer (TIA Portal - Security-Einstellungen)".

- **Benutzer (TIA Portal - Security-Einstellungen)**
  Sie können einen Benutzernamen aus der Liste der im Projekt eingetragenen Benutzer für die Authentifizierung eintragen. Die Namen der eingetragenen Benutzer des aktuellen Projekts finden Sie in der Benutzerverwaltung in der Projektnavigation unter "Security-Einstellungen > Benutzer und Rollen". Dort können Sie auch weitere Benutzer eintragen.
  Sie können auch einen Namen eintragen, der nicht in der Benutzerverwaltung des Projekts steht oder auch das Feld leer lassen. Dies ist dann notwendig, wenn der entsprechende Benutzername erst zur Laufzeit aus einer anderen Quelle kommt, beispielsweise über HMI oder von einem anderen OPC UA-Client.

Security Policy "Keine Security" und Authentifizierung über Benutzername und Passwort

Folgende Kombination können Sie einstellen:
Security Policy = "Keine Security" und Authentifizierung über Benutzername und Passwort.

- Der OPC UA-Server der S7-1500 unterstützt diese Kombination. OPC UA-Clients können sich verbinden und die Authentifizierungsdaten verschlüsseln oder auch nicht.
- Der OPC UA-Client der S7-1500 CPU unterstützt ebenfalls diese Kombination: Zur Laufzeit verbindet er sich aber nur dann, wenn er die Authentifizierungsdaten verschlüsselt über die Leitung schicken kann!

Folge: Bei folgender Konfiguration kann keine Verbindung zur Laufzeit hergestellt werden:

- S7-1500 als OPC UA-Client
- OPC UA-Server, der bei eingestellter Security Policy "Keine Security" (= "none") keine Verschlüsselung der Authentifizierungsdaten unterstützt

Weitere Informationen

Informationen zu den Benutzern und Rollen mit OPC UA-Funktionsrechten finden Sie im Kapitel Benutzer und Rollen mit OPC UA-Funktionsrechten (Seite 251).

11.4.9.4 Parametrierte Verbindung nutzen

Einleitung

Dieses Kapitel zeigt, wie Sie eine parametrierte Verbindung bei OPC UA-Anweisungen verwenden (Dritter Schritt).

Voraussetzungen

- Sie haben eine Client-Schnittstelle angelegt und dieser Schnittstelle PLC-Variablen und -Methoden hinzugefügt, siehe (Erster Schritt (Seite 336)).
- Sie haben eine Verbindung zu einem OPC UA-Server parametriert (Zweiter Schritt (Seite 350)).

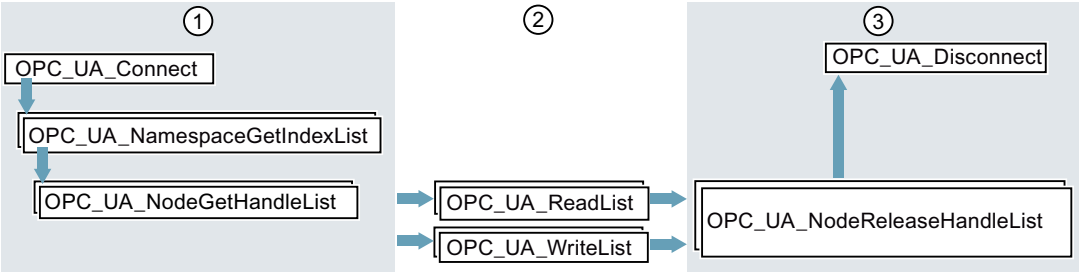
Übersicht

Um Daten von einem OPC UA-Server zu lesen oder Daten zu einem OPC UA-Server zu schreiben, verwenden Sie die folgenden Anweisungen:

- OPC_UA_Connect
- OPC_UA_NamespaceGetIndexList
- OPC_UA_NodeGetHandleList
- OPC_UA_ReadList oder OPC_UA_WriteList
- OPC_UA_NodeReleaseHandleList
- OPC_UA_Disconnect

Reihenfolge der OPC UA-Anweisungen

Das folgende Bild zeigt die Reihenfolge, in der die OPC UA-Anweisungen in einem Anwenderprogramm aufgerufen werden, um damit PLC-Variablen zu lesen oder zu schreiben:



① Anweisungen zur Vorbereitung der Lese- und Schreibvorgänge

② Lese- und Schreibanweisungen

③ Anweisungen zum "Clean-up" nach durchgeführten Lese- oder Schreibvorgängen
Die Anweisung "OPC_UA_NodeReleaseHandleList" kann entfallen, wenn anschließend gleich "OPC_UA_Disconnect" aufgerufen wird.

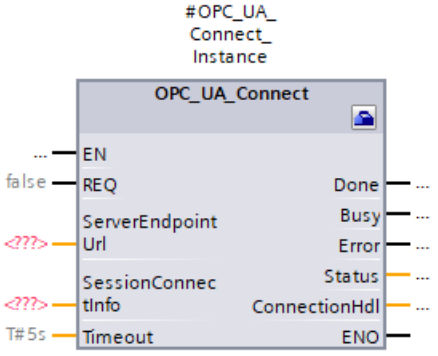
Bild 11-82 Aufrufreihenfolge für Schreib- und Lesevorgänge

STEP 7 (TIA Portal) versorgt die Parameter dieser Anweisungen automatisch, wenn Sie eine Client-Schnittstelle und eine parametrierte Verbindung zu einem OPC UA-Server verwenden. Wie Sie dabei vorgehen, zeigt das folgende Kapitel.

Client-Schnittstelle und parametrierte Verbindung verwenden

Um eine parametrierte OPC UA-Verbindung zu verwenden, gehen Sie folgendermaßen vor:

1. Öffnen Sie Ihr Anwenderprogramm im TIA Portal.
2. Ziehen Sie per Drag & Drop die Anweisung "**OPC_UA_Connect**" in den Programmeditor.
   Sie finden die Anweisung unter "Anweisungen > Kommunikation > OPC UA" im TIA Portal.
3. Wählen Sie eine Aufrufoption für die Anweisung
   Das Beispiel verwendet eine Multi-Instanz.
   STEP 7 stellt die Anweisung im Programmeditor dar.
   Der Editor für die Programmiersprache Funktionsplan (FUP) verwendet die folgende Darstellung:

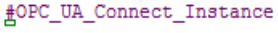


Der Editor für die Programmiersprache Kontaktplan (KOP) zeigt die Anweisung ähnlich an.

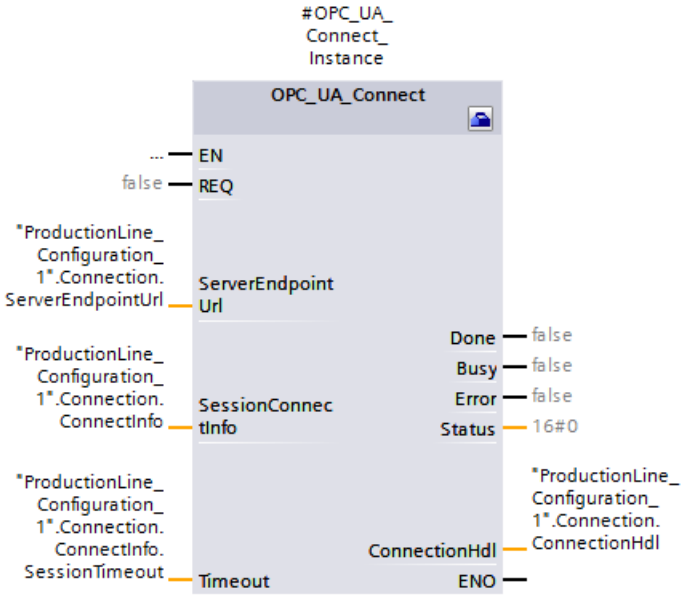
4. Klicken Sie im Editor für FUP oder KOP auf das Symbol für einen Werkzeugkasten.
   Das Symbol befindet sich im Kopf der Anweisung:

   Falls Sie den Editor für AWL oder SCL verwenden: Klicken Sie auf das kleine grüne Rechteck unter dem ersten Zeichen des Instanznamens:

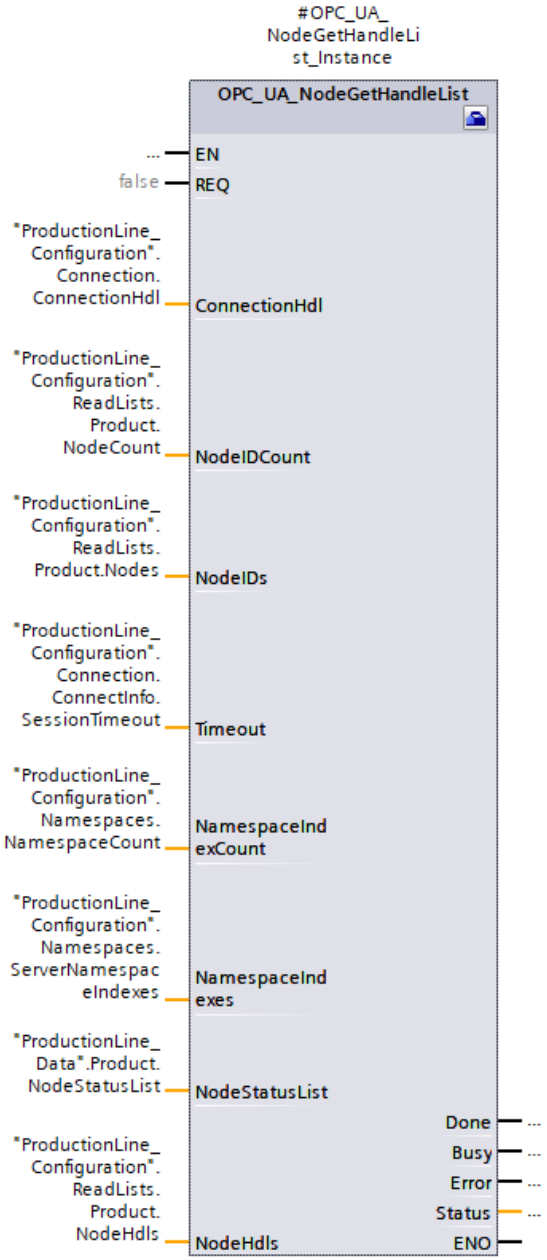
   

   Das Beispiel (Seite 335) verwendet "#OPC_UA_Connect_Instance" als Instanzname.
   STEP 7 stellt nun die Eigenschaften der Anweisung in einem eigenen Dialog dar.
5. Wählen Sie bei "Client-Schnittstelle" die Client--Schnittstelle aus, das Sie für die Anweisung verwenden wollen.
   Im Beispiel wählen wir die Client--Schnittstelle "ProductionLine".
   STEP 7 verschaltet nun die Client--Schnittstelle "ProductionLine" mit den Parametern der Anweisung OPC_UA_Connect:

   "ProductionLine" ist die Schnittstelle, die der OPC UA-Client des Beispiels (Seite 335) für den Datenaustausch mit dem OPC UA-Server "ProductionLine" verwendet.
6. Ziehen Sie per Drag & Drop die Anweisung "**OPC_UA_NamespaceGetIndexList**" in den Programmeditor.
   Sie finden die Anweisung unter "Anweisungen > Kommunikation > OPC UA" im TIA Portal.
   Wählen Sie die Aufrufoption "Multi-Instanz"
   Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
   Wählen Sie die Client-Schnittstelle aus, das Sie verwenden wollen (Im Beispiel "ProductionLine").
   STEP 7 verschaltet nun automatisch alle Parameter der Anweisung "OPC_UA_NamespaceGetIndexList":

7. Ziehen Sie per Drag & Drop die Anweisung "**OPC_UA_NodeGetHandleList**" in den Programmeditor.
   Wählen Sie die Aufrufoption "Multi-Instanz".
   Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
   Wählen Sie die Client-Schnittstelle aus, das Sie verwenden wollen. Das Beispiel nutzt die Client-Schnittstelle "ProductionLine".
   Wählen Sie unter "Datenzugriff > Lese-/Schreibliste" die Lese- oder Schreibliste aus, die Sie verwenden wollen (im Beispiel die Leseliste "Product").
   STEP 7 verschaltet nun automatisch alle Parameter der Anweisung "OPC_UA_NodeGetHandleList":



Falls Sie Daten zu einem OPC UA-Server schreiben wollen, dann wählen Sie unter "Datenzugriff > Schreibliste" die Schreibliste aus, die Sie verwenden wollen (im Beispiel wäre das die Schreibliste "ProductionStatus").

8. Ziehen Sie per Drag & Drop die Anweisung "**OPC_UA_ReadList**" in den Programmeditor.
   Wählen Sie die Aufrufoption "Multi-Instanz"
   Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
   Wählen Sie die Client-Schnittstelle aus, das Sie verwenden wollen. Das Beispiel nutzt das Client-Schnittstelle "ProductionLine".
   Wählen Sie unter "Datenzugriff > Leseliste" die Leseliste aus, die Sie verwenden wollen (im Beispiel die Leseliste "Product").
   STEP 7 verschaltet nun automatisch alle Parameter der Anweisung OPC_UA_ReadList.
   Falls Sie Daten zu einem OPC UA-Server schreiben wollen, dann verwenden Sie die Anweisung "**OPC_UA_WriteList**" und wählen Sie unter "Datenzugriff > Schreibliste" die Liste der Variablen, die Sie zum Server senden wollen (im Beispiel die Schreibliste "ProductionStatus").
9. Falls Sie in Ihrem Anwenderprogramm mit unterschiedlichen Leselisten oder Schreiblisten arbeiten sollten, die Sie programmgesteuert verwenden, ziehen Sie per Drag&Drop die Anweisung "**OPC_UA_NodeReleaseHandleList**" in den Programmeditor.
   Wählen Sie die Client-Schnittstelle aus, das Sie verwenden wollen.
   Wählen Sie nun eine Lese- oder Schreibliste aus, die Sie frei geben wollen: Geben Sie nur Lese-/ oder Schreiblisten frei, die Sie selten nutzen, da das erneute Registrieren zeitintensiv ist.
   Wiederholen Sie dann die Schritte ab Schritt 7 mit der Anweisung "OPC_UA_NodeGetHandleList".
10. Ziehen Sie per Drag & Drop die Anweisung "**OPC_UA_Disconnect**" in den Programmeditor.
    Wählen Sie die Aufrufoption "Multi-Instanz".
    Klicken Sie auf das Symbol für einen Werkzeugkasten (KOP und FUP) oder auf das grüne Kästchen unter dem Instanznamen (AWL und SCL), falls der Editor nicht bereits geöffnet ist.
    Wählen Sie die Client-Schnittstelle aus, das Sie verwenden wollen. Das Beispiel nutzt die Client-Schnittstelle "ProductionLine".
    STEP 7 verschaltet nun automatisch alle Parameter der Anweisung OPC_UA_Disconnect.

Unterstützte Anweisungen

Bei den folgenden Anweisungen versorgt STEP 7 automatisch die Parameter, wenn Sie eine Client-Schnittstelle und eine parametrierte Verbindung zu einem OPC UA-Server nutzen:

- OPC_UA_Connect
- OPC_UA_NamespaceGetIndexList
- OPC_UA_NodeGetHandleList
- OPC_UA_MethodGetHandleList
- OPC_UA_MethodReleaseHandleList
- OPC_UA_ReadList
- OPC_UA_WriteList
- OPC_UA_MethodCall

- OPC_UA_NodeReleaseHandleList
- OPC_UA_Disconnect

Kompaktanweisungen

Ab TIA Portal V17 stehen Ihnen Kompaktanweisungen für OPC UA zur Verfügung, die Verbindungsaufbau und Schreibauftrag//Leseauftrag/Methodenauruf zusammenfassen:

- OPC_UA_ReadList_C für die Erzeugung einer Verbindung und Lesen von Variablen
- OPC_UA_WriteList_C für die Erzeugung einer Verbindung und Schreiben von Variablen
- OPC_UA_MethodCall_C für die Erzeugung einer Verbindung und Aufruf von Methoden

Informationen zu den Kompaktanweisungen finden Sie in der TIA Portal-Hilfe.

11.5 Tipps und Empfehlungen

11.5.1 Regeln für Subscriptions

Folgende Regeln gelten für Subscriptions:

- Gruppieren Sie Subscriptions im Client nach unterschiedlichen Abtast- und Sendeintervallen und verteilen Sie die überwachten Elemente (Variablen) auf diese Gruppen.
  Beispiel: Bilden Sie eine Subscription für größere Sendeintervalle (z. B. 5 Sekunden) und eine Subscription für kleinere Sendeintervalle (z. B. 0,1 Sekunden).
- Deaktivieren Sie nicht benötigte Subscriptions.
  Grund: Der Subscription-Modus "Deaktiviert" reduziert den Ressourcenverbrauch.
- Um den Ressourcenverbrauch weiter zu opitimieren, reduzieren Sie beim Client den Subscription-Timeout. Der Subscription-Timeout kann nicht direkt vorgegeben werden; diese Zeit ergibt sich aus den vom Server bestätigten den Subscription-Einstellungen "PublishingInterval" und "LifetimeCount".
  Hintergrund: Wenn ein Client Subscriptions angelegt hat und die Session beendet wird, dann bleiben die Subscriptions zunächst im Server bestehen und belegen Speicherressourcen. Erst wenn der Subscriptions-Timeout die Lebenszeit der Subscriptons beendet, gibt der OPC UA-Server die hierfür benötigten Ressourcen wieder frei.
- Berücksichtigen Sie die maximale Anzahl überwachter Elemente (Monitored Items) von Subscriptions für die entsprechende S7-1500-CPU.
  Die Angaben finden Sie in den Technischen Daten der jeweiligen CPU, sie sind bezogen auf ein Abtast-/Sendeintervall von 1 Sekunde.
  Weitere Informationen finden Sie im FAQ 109755846 (https://support.industry.siemens.com/cs/de/de/view/109755846).
- Wählen Sie für den OPC UA-Client und für den OPC UA-Server gleiche Abtast- und Sendeintervalle.

- Vermeiden Sie Arrays und Strukturen als Elemente von Subscriptions – falls der Prozess das erlaubt.
  Grund: Wenn sich auch nur ein Wert eines Arrays / einer Struktur ändert, dann wird die gesamte Struktur übertragen, was unnötige Kommunikationslast erzeugt.
- Gelegentliche Nicht-Einhaltung der geforderten Abtastrate quittiert der OPC UA-Server der S7-1500 CPU entsprechend OPC UA Spezifikation mit einem "GoodOverload" Fehlercode, siehe auch TIA Portal-Hilfe. Unterschiedliche OPC UA-Clients gehen unterschiedlich mit "Good"-Fehlercodes ungleich "0" um. Berücksichtigen Sie dieses Verhalten und mindern, falls erforderlich, die Kommunikationslast entsprechend den oben vorgestellten Maßnahmen.

Weitere Informationen

Informationen, wie Sie den Server für Subscriptions einstellen, finden Sie im Kapitel Einstellungen des Servers für Subscriptions (Seite 239).

11.5.2 Regeln für das Anwenderprogramm

Anwenderprogramme für OPC UA

Folgende Regeln gelten für Anwenderprogramme:

- Wenn es Ihre Anwendung erlaubt und die Kommunikationsbelastung groß ist, sollten Sie eine Mindestzeit für Zyklus-OBs einstellen.
  Vorteile:
  – Die Zykluszeit bleibt weitgehend konstant
  – Der CPU bleibt durchgängig mehr Zeit für Kommunikationsaufgaben

  Tipp: Verwenden Sie zur Analyse der CPU-Auslastung (z. B. Kommunikation) die Anweisung "Runtime_Info"; Modus 21 bzw. Modus 25 (siehe TIA Portal-Hilfe).
- Reduzieren Sie die Anzahl der Variablen bzw. Datenbausteine, die aus OPC UA/HMI erreichbar sind. Standardmäßig sind beim Anlegen von Variablen/DBs/IDBs alle Variablen aus OPC UA/HMI erreichbar. Diese Maßnahme führt zu einer verbesserten Performance beim Laden im RUN.
  Tipp: Mit Hilfe der detaillierten Objektanzeige im TIA Portal können Sie einfach die nicht-OPC UA-relevanten Datenbausteine kennzeichnen als "nicht erreichbar aus OPC UA".
- Eine konsistente Übertragung von Daten über die Grenze von einfachen Datentypen hinaus ist nur bei OPC UA-Methoden möglich. Wenn Sie andere OPC UA-Funktionen nutzen (Subscriptions, Read/Write), müssen Sie die Datenkonsistenz applikativ sicherstellen.
- Für sich wiederholende Lese- bzw. Schreibzugriffe auf dieselben Variablen bietet OPC UA den Service "RegisterNodes". Mit Hilfe dieses Services können Server den optimierten Zugriff auf Variablen vorbereiten. Die Anweisung "OPC_UA_NodeGetHandleList" der S7-1500 als OPC UA-Client ruft implizit diesen Service auf, um den Server auf optimierte Zugriffe (im Sprachgebrauch von OPC UA "Registered Read/Write") vorzubereiten.

Detaillierte Objektanzeige im TIA Portal aufrufen

Um die detaillierte Objektanzeige aufzurufen, gehen Sie folgendermaßen vor:

1. Wechseln Sie in der Portalansicht in das Portal "PLC-Programmierung".
2. Wählen Sie "Alle Objekte anzeigen".
3. Wechseln Sie im Auswahlfenster in das Register "Details".
4. Deaktivieren Sie in der Spalte "DB erreichbar aus OPC UA" die Erreichbarkeit aus OPC UA für einzelne Objekte.

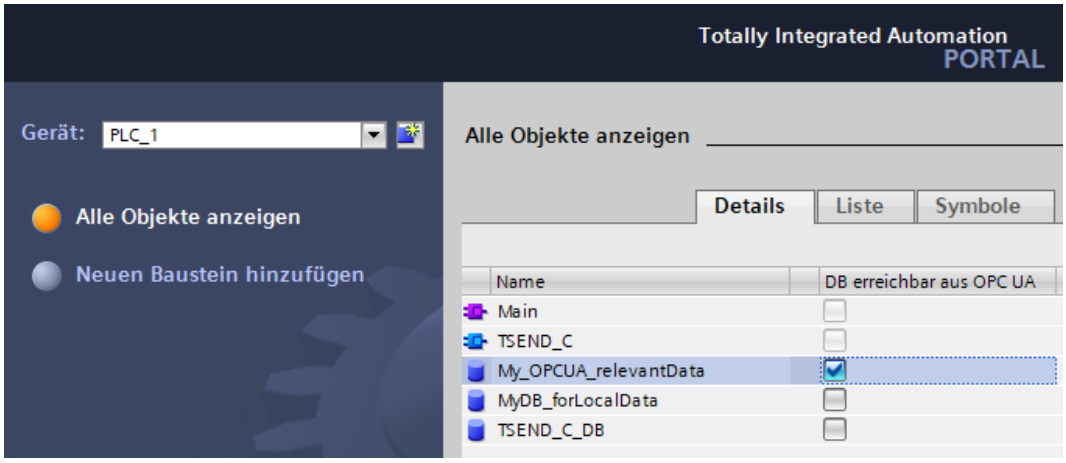


Bild 11-83 Detaillierte Objektanzeige im TIA Portal aufrufen

11.5.3 Kopiervorlagen für OPC UA-Kommunikation

Kopiervorlagen für die OPC UA-Schnittstellen

Schnittstellen von OPC UA-Servern und OPC UA-Clients, die Sie mehrfach verwenden möchten, können Sie entweder in der Projektbibliothek oder in einer globalen Bibliothek ablegen. Kopiervorlagen in der Projektbibliothek können Sie nur innerhalb des Projekts verwenden. Wenn Sie die Kopiervorlage in einer globalen Bibliothek erstellen, lässt sie sich in unterschiedlichen Projekten verwenden.

Die OPC UA-fähige CPUs unterscheiden 3 Schnittstellentypen des OPC UA-Servers:

- Übliche Server-Schnittstelle
- Companion-Spezifikation Schnittstelle
- Namensraumreferenz

Beim Hinzufügen der OPC-UA Schnittstelle in der Projektnavigation unter "OPC UA-Kommunikation" erhält jeder Schnittstellentyp ein eigenes Symbol. Das gleiche Symbol übernimmt auch die Kopiervorlage.

Erstellen Sie entweder einzelne Kopiervorlagen oder eine Kopiervorlage mit mehreren Schnittstellen.

Mehrere Kopiervorlagen aus Auswahl erzeugen

Sie selektieren ein oder mehrere Elemente und erzeugen daraus einzelne Kopiervorlagen.

1. Öffnen Sie die Bibliothek in der Task Card "Bibliotheken"
2. Selektieren Sie die gewünschten Elemente
3. Ziehen Sie die Elemente per Drag&Drop in den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner von "Kopiervorlagen"

Eine Kopiervorlage aus Auswahl erzeugen

Sie selektieren mehrere Elemente und erzeugen daraus eine einzelne Kopiervorlage, die alle selektierten Elemente enthält.

1. Kopieren Sie die Elemente, die Sie als Kopiervorlagen anlegen möchten, in die Zwischenablage
2. Klicken Sie in der Bibliothek mit der rechten Maustaste auf den Ordner "Kopiervorlagen" oder einen beliebigen Unterordner
3. Wählen Sie im Kontextmenü den Befehl "Als einzelne Kopiervorlage einfügen"

Wenn mehrere Schnittstellen vom OPC UA-Server oder OPC UA-Client zu einer Kopiervorlage hinzugefügt werden, werden die Beschriftung und das Symbol in der Bibliothek entsprechend geändert.
Anstelle des einfachen Symbols wird ein Symbol mit "+" angezeigt.

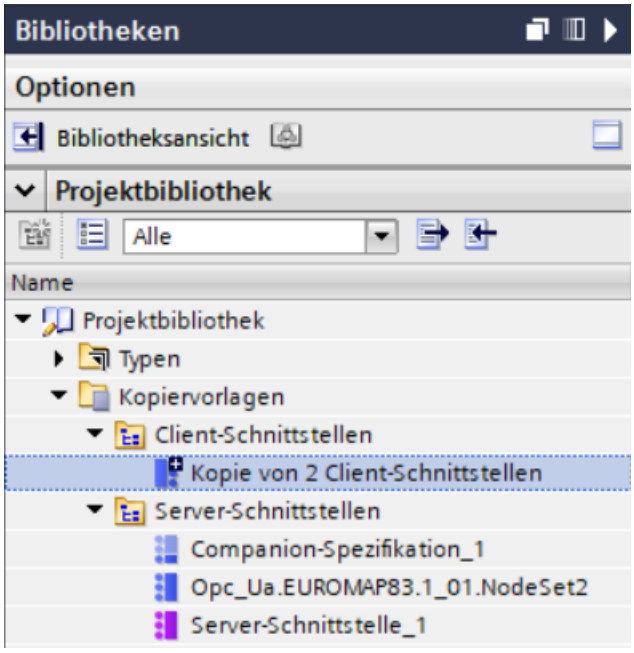


Bild 11-84 Kopiervorlage in STEP 7 erzeugen

Weitere Informationen

Informationen, wie Sie eine benutzerdefinierte Server-Schnittstelle anlegen, finden Sie im Kapitel Benutzerdefinierte Server-Schnittstelle anlegen (Seite 269).

12

Adressvergabe über DHCP

Um zukunftssicher, leistungsfähig und flexibel zu automatisieren, unterstützen immer mehr Komponenten aus dem Fertigungsbereich IT-Standards. Weltweite Ethernet-Standards, durchgängige Kommunikation sowie Vielseitigkeit machen IT-gestützte Automatisierung zu einer wirtschaftlichen Lösung für Ihre Anforderungen. Funktionale Erweiterungen der Kommunikationsmöglichkeien der S7-1500 CPUs in dieser Richtung geben Ihnen mehr Freiräume für die Einsatzmöglichkeiten Ihrer Anlage oder Maschine. Sie nutzen IT-Technologie, um effizient zu automatisieren. Mit der Einführung von DHCP und der Erweiterung von DNS für S7-1500-CPU ab Firmware-Version V2.9 gewinnen Sie mehr Flexibilität für die Konzeption Ihrer Automatisierungslösung.

Für die Schnittstellen einer S7-1500-CPU können Sie einstellen, dass Adressparameter, wie z. B. die IP-Adresse mit Subnetzmaske, von einem DHCPv4-Server (im Folgenden DHCP-Server) bezogen werden.

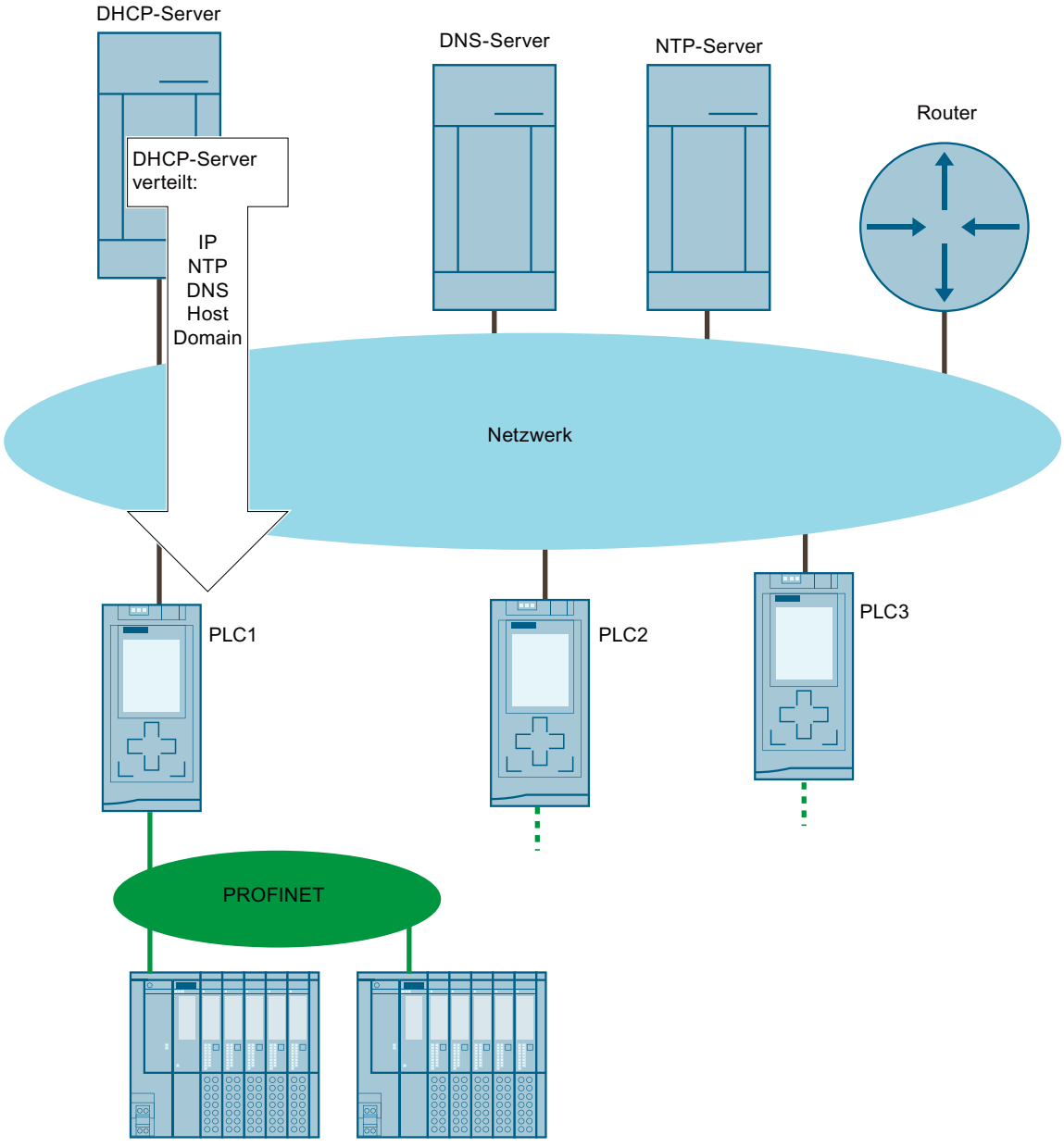


Bild 12-1 Überblick DHCP

Anwendungbereiche

- Einsatz der S7-1500 CPU in einer gemanagten IT-Umgebung
- Hinfügen von neuen Geräten in einer modularen Fertigungsstruktur

12.1 Prinzip der Adressvergabe über DHCP

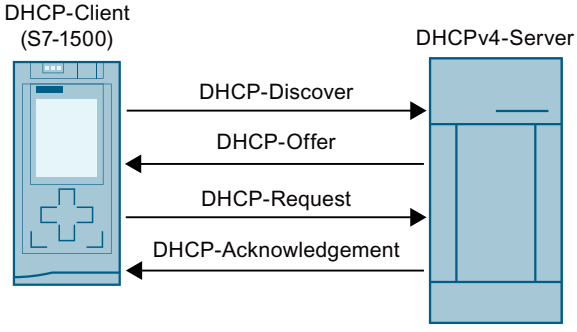
Voraussetzungen Projektierung

Damit eine PROFINET-Schnittstelle der S7-1500 CPU IP-Adressparameter über einen DHCP-Server beziehen kann, müssen folgende Voraussetzungen erfüllt sein:

- Die Adressvergabe über einen DHCP-Server ist konfiguriert.
  DHCP aktivieren (Seite 374)
- Für die Schnittstelle darf keine PROFINET IO-Kommunikation projektiert sein.

Prinzip der DHCP-Adressvergabe

Sobald das Projekt in die CPU-geladen oder die CPU mit projektierter DHCP-Adressvergabe eingeschaltet und hochgefahren ist, beginnt der Prozess der DHCP-Adressvergabe:



DHCP-Discover	Der DHCP-Client sucht über Broadcast nach einem geeigneten DHCP-Server. Der DHCP-Client identifiziert sich gegenüber dem DHCP-Server mit der konfigurierten Client-ID bzw. mit seiner MAC-Adresse.
DHCP-Offer	Der DHCP-Server bietet dem DHCP-Client IP-Adressparameter (IPv4-Adresse, Subnetzmaske, optional Default-Router) und gegebenenfalls weitere Daten (Optionen) an.
DHCP-Request	Der DHCP-Client fordert die im DHCP-Offer angebotenen IP-Adressparameter und Optionen. Der DHCP-Client der S7-1500 CPU akzeptiert immer die erste DHCP-Offer eines DHCP-Servers, die den Erfordernissen genügt (IP-Adresse mit Subnetz-Maske).
DHCP-Acknowledgement	Der DHCP-Server bestätigt und übermittelt die der im DHCP-Offer angebotenen IP-Adressparameter und Optionen. Der DHCP-Server teilt dem DHCP-Client auch mit, wie lange der DHCP-Client die Adressparameter verwenden kann (Lease-Time).

Bild 12-2 Prinzip der Adressvergabe bei DHCP

Die IP-Adressparameter und Optionen werden nicht im Ladespeicher der CPU abgelegt. Nach einem Urlöschen oder Neustart der CPU werden die IP-Adressparameter und Optionen erneut über DHCP bezogen.

Optionen bei der DHCP-Adressvergabe

Für die S7-1500 CPU können Sie konfigurieren, dass die folgenden Optionen über einen DHCP-Server bezogen werden:

- Adressen von bis zu vier DNS-Servern
  Adressen der DNS-Server über DHCP beziehen (Seite 376)
- Adresse von bis zu 4 NTP-Servern
  Adressen der NTP-Server über DHCP beziehen (Seite 377)
- Hostname und Domain
  Hostname und Domain über DHCP beziehen (Seite 377)

Falls erforderlich, liefert der DHCP-Server auch die Adresse eines Routers (Standard-Gateway) als Option mit.

Wie lange kann die S7-1500 CPU die DHCP-Adressparameter verwenden?

Zusätzlich zu den Adressparametern teilt der DHCP-Server der S7-1500 CPU (DHCP-Client) auch die Lease-Time mit. Die Lease-Time legt fest, wie lange die CPU die Adressparameter verwenden kann.

Wenn die Lease-Time vollständig abgelaufen ist, gibt die CPU die zugewiesenen Adressparameter wieder zurück. Die CPU verfügt über eine interne Zeitüberwachung für die Lease-Time.

Für die CPU gibt es zu bestimmten Zeitpunkten beim Ablauf der Lease-Time die Möglichkeit, die Lease-Time zu verlängern:

- Renewal: Die Hälfte der Lease-Time ist abgelaufen: Die CPU kontaktiert den ursprünglichen DHCP-Server und fragt nach einer Verlängerung der Lease-Time. Der ursprüngliche DHCP-Server kann die bestehende Lease-Time entweder bestätigen oder eine neue Lease-Time vergeben. Bei einer neuen Lease-Time wird die Zeitüberwachung in der CPU zurückgesetzt.
- Rebinding: 7/8 der Lease-Time sind abgelaufen: Die CPU kontaktiert alle verfügbaren DHCP-Server über Broadcast und fragt nach einer Verlängerung der Lease-Time. Ein DHCP-Server kann die bestehende Lease-Time entweder bestätigen oder eine neue Lease-Time vergeben. Bei einer neuen Lease-Time wird die Zeitüberwachung in der CPU zurückgesetzt.
  Bei einer negativen Antwort des DHCP-Server beim Rebinding oder wenn kein DHCP-Server antwortet, dann gibt die CPU die Adressparameter nach 8/8 der Lease-Time wieder zurück.

Wenn die CPU die Adressparameter nach Ablauf der Lease-Time zurückgegeben hat, dann startet die CPU mit einem neuen DHCP-Discover einen neuen Zyklus zur DHCP-Adressvergabe.

Weitere Informationen

Informationen, wie Sie die Client-ID konfigurieren, finden Sie im Kapitel Client-ID konfigurieren (Seite 375).

12.2 DHCP mit DNS

Ab STEP 7 V17 unterstützt die S7-1500 CPU die in der namensbasierten Kommunikation (DNS) verwendeten Adressparameter Hostname und Domain.

Für bestimmte Kommunikationsdienste ist die namensbasierte Addressierung über den vollständigen Namen bestehend aus Hostname und Domain sinnvoll:

- Die CPU ist unter dem vollständigen Namen addressierbar, z. B. bei OPC UA, Open User Communication. Im Falle einer dynamischen IP-Adressvergabe durch den DHCP-Server ist durch den DNS-Namen immer eine eindeutige Adressierung möglich.
- In Zertifikaten der S7-1500 CPU kann der vollständige Name enthalten sein, z.B. für OPC UA-Kommunikation, Webserver, secure Communication.
  - Nur wenn Sie Hostname und Domain für die S7-1500 CPU in STEP 7 konfigurieren, dann wird der vollständige Name in den Gerätezertifikaten im Projekt als subject alternate name (SAN) eingetragen.
  - Sobald Sie Hostname und/oder Domain über DHCP beziehen bzw. über das Anwenderprogamm vergeben, wird der vollständige Name nicht in den Gerätezertifikaten im Projekt hinterlegt.

Wie Sie die DNS-Konfiguation für Ihre CPU einstellen, hängt davon ab, wie Sie Hostname und Domain in Ihrem Netzwerk vergeben.

- Zentrale Vergabe von Hostname und Domain
  Sie vergeben Hostname und Domain in Ihrem Netz zentral, z. B. über einem konfigurierten DNS-Server. Sie konfigurieren in STEP 7, dass die CPU Hostname und Domain über DHCP bezieht.
  In der folgenden Konfiguration ist in der S7-1500 CPU nur die Client-ID konfiguriert. Bei der DHCP-Adressvergabe liefert der DHCP-Server die Optionen Hostname und Domain an die CPU.

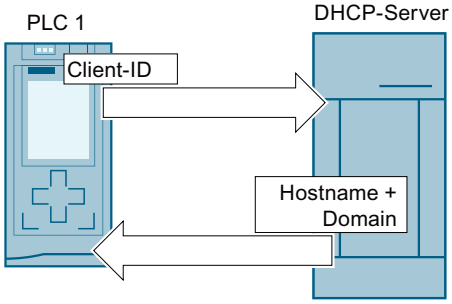


Bild 12-3 Hostname und Domain über DHCP beziehen

Für diese Konfiguration müssen Sie in STEP 7 zuerst die Hostname- und Domain-Konfiguration aktivieren. Anschließend konfigurieren Sie, dass Hostname und Domain über DHCP bezogen werden.

Hostname und Domain über DHCP beziehen (Seite 377)

- Lokale Vergabe von Hostname und Domain
  Sie können Hostname und Domain in STEP 7 projektieren oder im Anwenderprogramm vergeben.

HINWEIS

Gültigkeit der von DHCP bezogenen Daten

Wenn Sie Hostname und/oder Domain im Anwenderprogramm ändern, dann werden alle über DHCP bezogenen Daten (IP-Suite, Hostname, Domain, NTP-Server, DNS-Server) ungültig und werden von DHCP-Server neu bezogen. Deshalb solten Sie Hostname und/oder Domain nur in dringenden Fällen und nicht im laufenden Betrieb ändern.

Alle Verbinungen können abgeworfen werden, wenn sich die IP-Adresse der Schnittstelle ändert.

In der folgenden Konfiguration ist in der S7-1500-CPU neben ihrer Client-ID auch ihr Hostname und ihre Domain konfiguriert. Bei der DHCP-Adressvergabe liefert die CPU neben der Client-ID den Hostname und die Domain an den DHCP-Server. Der DHCP-Server erhält die Informationen, um z. B. einen DNS-Server mit den Adressdaten der CPU zu updaten.

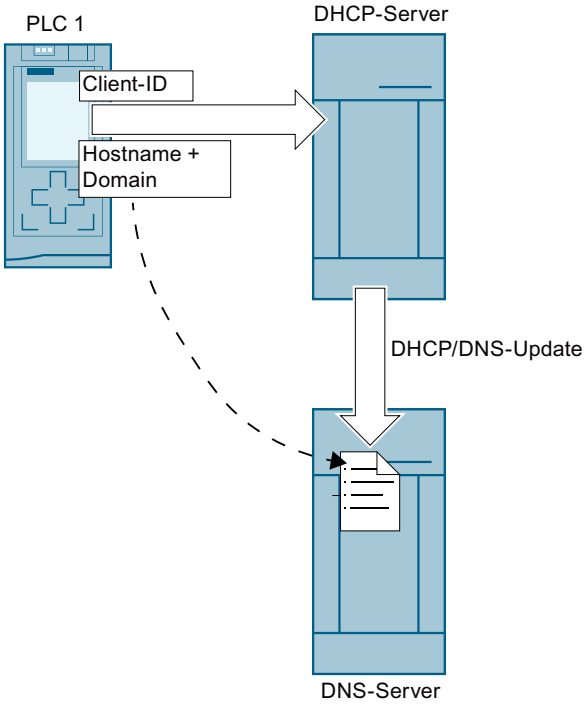


Bild 12-4 Hostname und Domain konfigurieren

Für diese Konfiguration müssen Sie in STEP 7 zuerst die Hostname- und Domain-Konfiguration aktivieren. Anschließend konfigurieren Sie Hostname und Domain in STEP 7.

- Zentrale Vergabe der Domain und lokale Vergabe von Hostname.
  - Sie konfigurieren in STEP 7,dass die CPU die Domain über DHCP bezieht.
  - Sie können den Hostname in STEP 7 projektieren oder über das Anwenderprogramm vergeben.

  In der folgenden Konfiguration ist in der S7-1500 CPU neben ihrer Client-ID auch der Hostname konfiguriert. Bei der DHCP-Adressvergabe liefert die CPU neben der Client-ID den Hostname an den DHCPv4-Server. Der DHCP-Server liefert die Option Domain an die CPU.

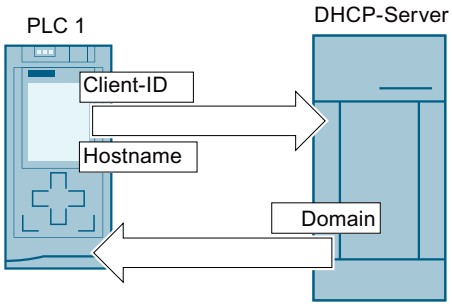


Bild 12-5 Hostname konfigurieren, Domain über DHCP beziehen

Für diese Konfiguration müssen Sie in STEP 7 zuerst die Hostname- und Domain-Konfiguration aktivieren. Anschließend konfigurieren Sie den Hostname in STEP 7 und konfigurieren, dass die Domain über DHCP bezogen wird.

Voraussetzungen

- Sie haben für mindestens eine Schnittstelle der S7-1500 CPU die Adressvergabe über DHCP aktiviert.

Konfiguration von Hostname- und Domain aktivieren

Um die Hostname- und Domain-Konfiguration in STEP 7 zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in der Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration".
3. Aktivieren Sie das Optionskästchen "Hostname und Domain aktivieren".

Hostname in STEP 7 konfigurieren

Um den Hostname in STEP 7 zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration" > "Hostname".
3. Wählen Sie bei "Hostname-Konfiguration:" in der Klappliste "Hostname im Projekt einstellen" aus.
4. Geben Sie bei "Hostname:" den Hostname ein.
   - Geben Sie Ihren gewünschten Hostname ein.
   - Wenn Sie das Optionskästchen "Hostname identisch mit Gerätename" aktivieren, dann vergibt STEP 7 automatisch den Gerätenamen als Hostname.

Nur wenn Sie in STEP 7 Hostname und Domain konfiguriert haben, dann wird bei "Vollständiger Name:" der vollständige Name angezeigt.

Hostname im Anwenderprogramm vergeben

Um den Hostname im Anwenderprogramm zu vergeben, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration" > "Hostname".
3. Wählen Sie bei "Hostname-Konfiguration:" in der Klappliste "Hostname direkt am Gerät einstellen (z. B. PLC-Programm, Dispay)" aus.
4. Rufen Sie im Anwenderprogramm die Anweisung "CommConfig" auf. Der Parameter DATA muss auf einen UDT "Conf_Hostname" zeigen, in dem der Hostname festgelegt ist.

Weitere Informationen zur Anweisung "CommConfig" und zum UDT "Conf-Hostname" finden Sie in der Online-Hilfe von STEP 7.

Domain in STEP 7 konfigurieren

Um die Domain in STEP 7 zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration" > "Domain".
3. Wählen Sie bei "Domain-Konfiguration:" in der Klappliste "Domain im Projekt einstellen" aus.
4. Geben Sie bei "Domain:" ihre gewünschte Domain ein.

Nur Wenn Sie in STEP 7 Hostname und Domain konfiguriert haben, dann wird bei "Vollständiger Name:" der vollständige Name angezeigt.

Domain im Anwenderprogramm vergeben

Um die Domain im Anwenderprogramm zu vergeben, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration" > "Domain".
3. Wählen Sie bei "Hostname-Konfiguration:" in der Klappliste "Domain direkt am Gerät einstellen (z. B. PLC-Programm, Dispay)" aus.
4. Rufen Sie im Anwenderprogramm die Anweisung "CommConfig" auf. Der Parameter DATA muss auf ein UDT "Conf_Domainname" zeigen, in dem der Domainname festgelegt ist.

Weitere Informationen zur Anweisung "CommConfig" und zum UDT "Conf-Domainname" finden Sie in der Online-Hilfe von STEP 7.

Regeln für maximale Längen von Hostname, Domain und Client-ID

Beachten Sie die folgenden maximalen Längen in Byte. Dabei entspricht ein Byte einem Zeichen:

- Hostname: maximal 63 Byte
- Domain: maximal 252 Byte
- Client-ID: maximal 254 Byte
- Hostname + Domain: maximal 254 Byte
- Hostname + Domain + Client ID: maximal 260 Byte
  Gilt nur, wenn Hostname und Domain zum DHCP-Server gesendet werden müssen.

12.3 DHCP aktivieren

Voraussetzungen

- S7-1500-CPU ab Firmwarestand V2.9

Vorgehen

Um DHCP für die PROFINET-Schnittstelle einer S7-1500-CPU zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP7 die PROFINET-Schnittstelle der S7-1500-CPU.
2. Navigieren Sie in den Eigenschaften der Schnittstelle zu "Ethernet-Adressen" > "Internet Protocol Version 4 (IPv4)".
3. Wählen Sie die Option "IP-Adresse von DHCP-Server" aus.

Ergebnis

Sie haben für die Schnittstelle eingestellt, dass diese Ihre IP-Adresse über einen DHCP-Server bezieht.
Als Betriebsart für DHCP ist bei der S7-1500-CPU "MAC-Adresse als Client-ID verwenden" eingestellt. Wie Sie die Client-ID anpassen, finden Sie beschrieben bei Client-ID konfigurieren (Seite 375).

12.4 Client-ID konfigurieren

Die Client-ID

Die S7-1500 CPU identifiziert sich gegenüber einem DHCP-Server immer mit der Client-ID (DHCP-Option 61). Die Client-ID ist schnittstellenspezifisch.
Die S7-1500-CPU unterstützt hinsichtlich der Client-ID die beiden folgenden Betriebsarten:

- MAC-Adresse als Client-ID verwenden: Die MAC-Adresse der CPU wird als Client-ID für den DHCP-Client verwendet. Beachten Sie, wenn Sie bei dieser Betriebsart einen Gerätetausch der CPU durchführen, dann ändert sich die MAC-Adresse und damit auch die Client-ID.
- Anwenderdefinierte Client-ID: Bei dieser Option legen Sie die Client-ID in der Projektierung in STEP7 fest. Zusätzlich haben Sie die Möglichkeit, dass Sie die Client-ID zur Laufzeit anpassen, z. B. im Anwenderprogramm über die Anweisung "CommConfig".
  Wenn Sie bei dieser Betriebsart einen Gerätetausch der CPU durchführen, dann bekommt die neue CPU die projektierte Client-ID zugewiesen.

Voraussetzung

- Sie haben die Adressvergabe über DHCP für die Schnittstelle aktiviert.

Client-ID konfigurieren

Um die Client-ID in STEP 7 zu konfigurieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP7 die PROFINET-Schnittstelle der S7-1500-CPU.
2. Navigieren Sie in den Eigenschaften der Schnittstelle zu
   "Ethernet-Adressen" > "Internet Protocol Version 4 (IPv4)" > "IP-Adresse von DHCP-Server".
3. Wählen Sie bei "Betriebsart:" in der Klappliste die gewünschte Betriebsart aus:
   – MAC-Adresse als Client-ID verwenden (Voreinstellung)
   – Anwenderdefinierte Client-ID

   Wenn Sie die Option "MAC-Adresse als Client-ID verwenden" ausgewählt haben, dann sind Sie hier bereits fertig. Bei "Anwenderdefinierte Client-ID" fahren Sie mit Schritt 4. fort.
4. Geben Sie bei "Client ID" eine gültige Client-ID ein.
   – Eine Zeichenfolge mit 7-Bit-ASCII-Zeichen im Bereich von 0x21 bis 0x7e ist erlaubt.
   – Einige DHCP-Server erwarten eine vorangestellte "0" (z. B. einige SCALANCE-Geräte). Geben Sie in diesem Fall "\0" ein, gefolgt von Ihrer Client-ID.
   – Sie können das Feld auch frei lassen. In diesem Fall müssen Sie das Optionskästchen "Client-ID kann zur Laufzeit geändert werden" aktivieren.

5. Um die anwenderdefinierte Client-ID zur Laufzeit anpassbar zu machen, aktivieren Sie Optionskästchen "Client-ID kann zur Laufzeit geändert werden".

Client-ID zur Laufzeit anpassen

Mit der Anweisung "CommConfig" können Sie die Client-ID über das Anwenderprogramm anpassen. Rufen Sie die Anweisung auf. Der Parameter DATA muss auf einen UDT "Conf_ClientId" oder einen UDT "Conf_ClientId_Opaque" zeigen. Im UDT muss die Client-ID festgelegt ist.

Wenn Sie die anwenderdefinierte Client-ID in der Projektierung in STEP 7 frei gelassen haben, dann verwendet die CPU bis zur ersten Anpassung der Client-ID die MAC-Adresse als Client-ID.

HINWEIS

Gültigkeit der über DHCP bezogenen Daten

Wenn Sie ClientId mit "CommConfig" ändern, dann werden alle über DHCP bezogenen Daten ungültig: IP Suite, Domainname, NTP-Server, DNS-Server. Deshalb sollten Sie ClientId nur in dringenden Fällen und nicht im laufenden Betrieb ändern.

Weitere Informationen zur Anweisung "CommConfig" und zu den UDTs "Conf_ClientId" und "Conf_ClientId_Opaque" finden Sie in der Onlinehilfe von STEP 7.

12.5 Adressen der DNS-Server über DHCP beziehen

Voraussetzungen

- Sie haben für mindestens eine Schnittstelle der S7-1500 CPU die Adressvergabe über DHCP aktiviert.

Adressen von DNS-Servern über DHCP beziehen

Um die Adressen von bis zu 4 DNS-Servern über DHCP zu beziehen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "DNS-Konfiguration" > "Server-Liste".
3. Wählen Sie bei "Namensauflösung über DNS:" in der Klappliste "DNS-Server remote einstellen (z. B. DHCP)" aus.

Ergebnis: Wenn der DHCP-Server als Option Adressen von DNS-Servern liefert, dann verwendet die CPU bis zu 4 Adressen.

12.6 Adressen der NTP-Server über DHCP beziehen

Voraussetzungen

- Sie haben für mindestens eine Schnittstelle der S7-1500 CPU die Adressvergabe über DHCP aktiviert.

Adressen von NTP-Servern über DHCP beziehen

Um die Adressen von bis zu vier NTP-Servern über DHCP zu beziehen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Uhrzeit" > "Uhrzeitsynchronisation" > "NTP-Verfahren".
3. Wählen Sie bei "Uhrzeitsynchronisation:" in der Klappliste "NTP-Server remote einstellen (z. B. DHCP)" aus.

Ergebnis: Wenn der DHCP-Server als Option Adressen von NTP-Servern liefert, dann verwendet die CPU bis zu 4 Adressen.

12.7 Hostname und Domain über DHCP beziehen

Voraussetzung

- Sie haben für mindestens eine Schnittstelle der S7-1500 CPU die Adressvergabe über DHCP aktiviert.
- Sie haben die Hostname- und Domain-Konfiguration in STEP 7 aktiviert.

Hostname über DHCP beziehen

Um den Hostname über DHCP zu beziehen, gehen Sie folgendermaßen vor:

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration" > "Hostname".
3. Wählen Sie bei "Hostname-Konfiguration:" in der Klappliste "Hostname remote einstellen (z. B. DHCP)" aus.

Ergebnis: Wenn der DHCP-Server als Option einen Hostnamen liefert, dann verwendet die CPU diesen Hostnamen.

Domain über DHCP beziehen

Um die Domain über DHCP zu beziehen, gehen Sie folgendermaßen vor

1. Selektieren Sie in STEP 7 die S7-1500 CPU.
2. Navigieren Sie in den Eigenschaften der CPU zu "Erweiterte Konfiguration" > "Hostname und Domain" > "Hostname- und Domain-Konfiguration" > "Domain".
3. Wählen Sie bei "Domain-Konfiguration:" in der Klappliste "Domain remote einstellen (z. B. DHCP)" aus.

Ergebnis: Wenn der DHCP-Server als Option eine Domain liefert, dann verwendet die CPU diese Domain.

13

Routing

13.1 Überblick über die Routing-Mechanismen der S7-1500 CPUs

Die folgende Tabelle gibt einen Überblick über die Routing-Mechanismen der S7-1500-CPU.

Routing-Mechanismus	Beschreibung	Anwendungen	Kapitel
S7-Routing	S7-Routing ist die Übertragung von Daten über S7-Subnetzgrenzen hinweg. Hierbei können Sie Informationen von einem Sender über verschiedene S7-Subnetze hinweg zu einem Empfänger verschicken.	Anwenderprogramme laden Hardware-Konfiguration laden Test- und Diagnosefunktionen ausführen	S7-Routing (Seite 380)
IP-Forwarding	IP-Forwarding ist eine Funktion von Geräten, IP-Pakete zwischen 2 angeschlossenen IP-Subnetzen weiterzuleiten.	Einfacher Zugriff von der Leitebene auf die Feldebene für Konfiguration und Parametrierung von Devices, z. B. per PDM oder Webbrowser. Vereinfachte Einbindung von Devices für Remote-Zugriffe, z. B. für Diagnose bei Fernwartung oder Firmware-Update.	IP-Forwarding (Seite 384)
Datensatz-Routing	Daten können von einer Engineering Station von PROFINET aus über verschiedene Netzwerke hinweg an Feldgeräte gesendet werden. Da die Engineering Station die Feldgeräte über genormte Datensätze anspricht und diese Datensätze über S7-Geräte geroutet werden, hat sich für diese Art des Routings der Begriff "Datensatz-Routing" etabliert.	Datensatz-Routing wird z. B. eingesetzt, wenn Feldgeräte verschiedener Hersteller zum Einsatz kommen. Die Feldgeräte werden zur Parametrierung und Diagnose über genormte Datensätze (PROFINET) angesprochen.	Datensatz-Routing (Seite 391)

13.2 S7-Routing

Definition S7-Routing

S7-Routing ist die Übertragung von Daten über S7-Subnetzgrenzen hinweg. Hierbei können Sie Informationen von einem Sender über verschiedene S7-Subnetze hinweg zu einem Empfänger verschicken. Der Übergang von einem S7-Subnetz zu einem oder mehreren anderen Subnetzen erfolgt im S7-Router. Der S7-Router ist ein Gerät, welches über die Schnittstellen zu den betreffenden S7-Subnetzen verfügt. S7-Routing ist über verschiedene S7-Subnetze (PROFINET/Industrial Ethernet und/oder PROFIBUS) möglich.

Voraussetzungen für S7-Routing

- Alle erreichbaren Geräte in einem Netz sind innerhalb eines Projekts in STEP 7 konfiguriert und geladen worden.
- Alle am S7-Routing beteiligten Geräte müssen Informationen darüber erhalten, welche S7-Subnetze über welche S7-Router erreicht werden können (= Routing-Information). Die Routing-Information erhalten die Geräte durch das Laden der Hardwarekonfiguration in die CPUs, da die CPUs die Rolle eines S7-Routers spielt.
  Bei einer Topologie mit mehreren hintereinanderliegenden S7-Subnetzen müssen Sie folgende Reihenfolge beim Laden einhalten: zunächst laden Sie die Hardwarekonfiguration in die CPU(s), die direkt mit dem selben S7-Subnetz wie das PG/PC verbunden sind, dann laden Sie nacheinander die CPUs der dahinterliegenden S7-Subnetzen, vom nächsten S7-Subnetz bis zum am weitesten entfernten S7-Subnetz.
- Das PG/PC, mit dem Sie eine Verbindung über einen S7-Router herstellen wollen, muss dem S7-Subnetz zugeordnet sein, an dem es auch tatsächlich physikalisch angeschlossen ist. Das PG/PC können Sie in STEP 7 unter Online & Diagnose > Online-Zugänge > Verbindung mit Schnittstelle/Subnetz einen PG/PC zuordnen.
- Für S7-Subnetze vom Typ PROFIBUS: Die CPU muss entweder als DP-Master konfiguriert sein oder wenn sie als DP-Slave konfiguriert ist, muss in den Eigenschaften der DP-Schnittstelle des DP-Slaves das Kontrollkästchen "Test, Inbetriebnahme und Routing" aktiviert sein.
- S7-Routing für HMI-Verbindungen ist ab STEP 7 V13 SP1 möglich.

HINWEIS

Firewall und S7-Routing

Eine Firewall erkennt die IP-Adresse des Senders beim S7-Routing nicht, wenn der Sender sich außerhalb des an die Firewall angrenzenden S7-Subnetzes befindet.

Einen Überblick, welche Geräte die Funktion "S7-Routing" unterstützen, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/584459).

S7-Routing für Online-Verbindungen

Sie können mit dem PG/PC Geräte über S7-Subnetze hinweg erreichen, um beispielsweise:

- Anwenderprogramme zu laden
- eine Hardware-Konfiguration zu laden
- um Test- und Diagnosefunktionen ausführen zu können

Im folgenden Bild ist die CPU 1 S7-Router zwischen S7-Subnetz 1 und S7-Subnetz 2.

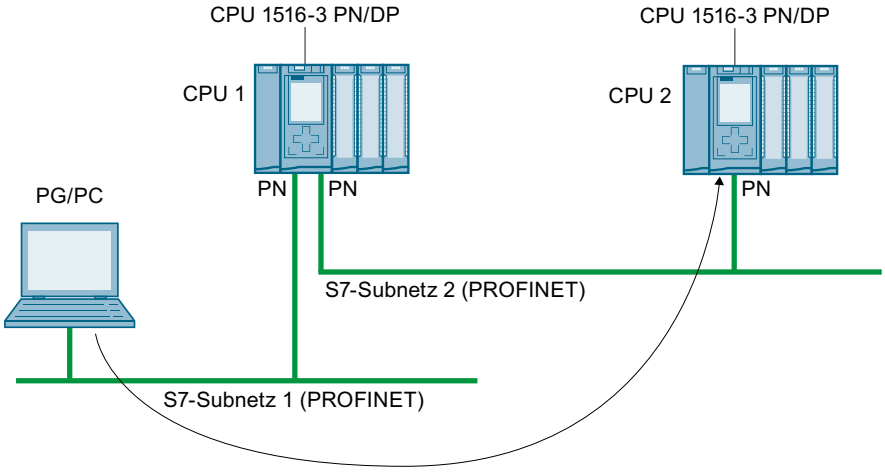


Bild 13-1 S7-Routing: PROFINET - PROFINET

Im folgenden Bild ist der Zugriff von einem PG über PROFINET nach PROFIBUS dargestellt. Die CPU 1 ist S7-Router zwischen S7-Subnetz 1 und S7-Subnetz 2; die CPU 2 ist S7-Router zwischen S7-Subnetz 2 und S7-Subnetz 3.

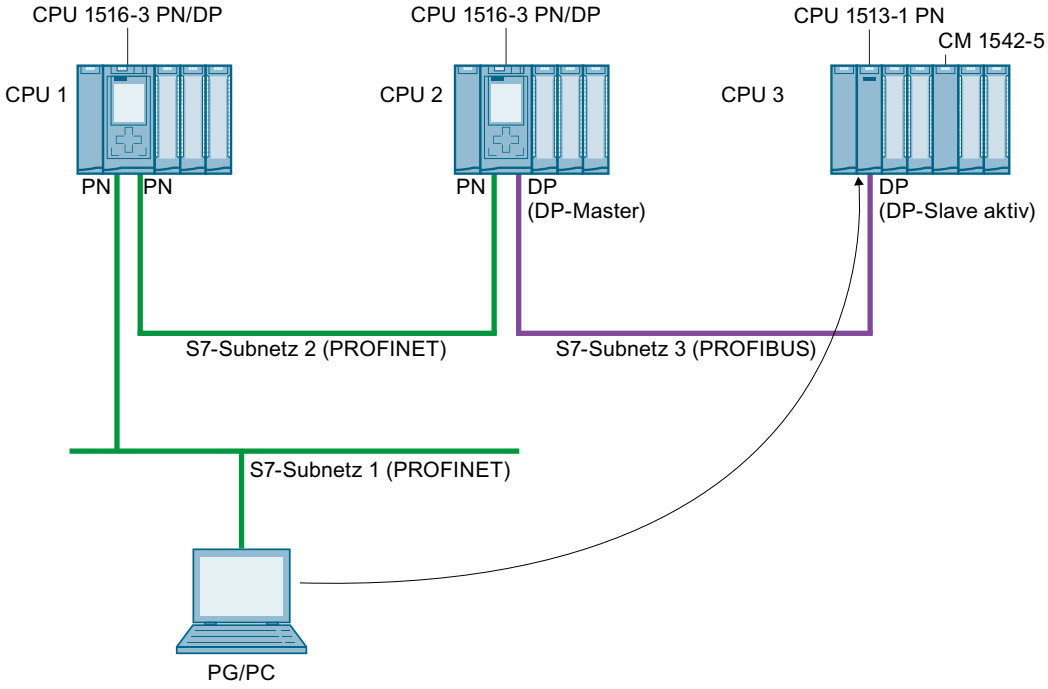


Bild 13-2 S7-Routing: PROFINET - PROFIBUS

S7-Routing für HMI-Verbindungen

Sie haben die Möglichkeit, eine S7-Verbindung von einem HMI zu einer CPU über unterschiedliche Subnetze (PROFIBUS und PROFINET bzw. Industrial Ethernet) einzurichten. Im folgenden Bild ist die CPU 1 S7-Router zwischen S7-Subnetz 1 und S7-Subnetz 2.

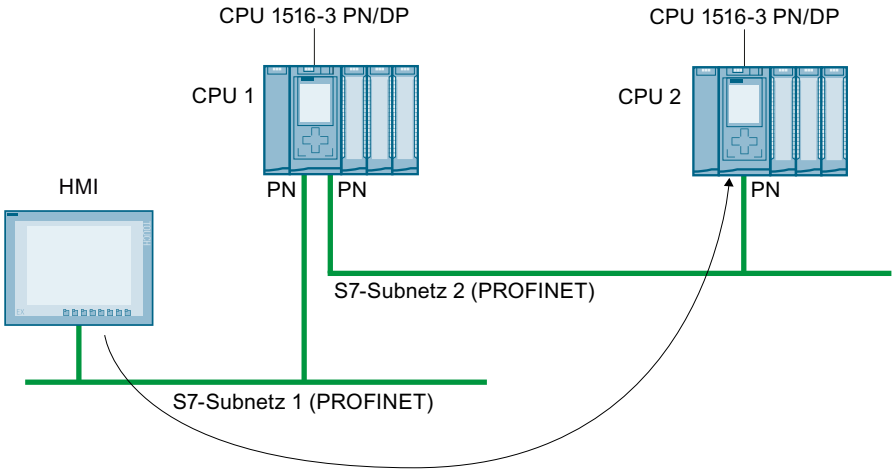


Bild 13-3 S7-Routing über HMI-Verbindung

S7-Routing für CPU-CPU-Kommunikation

Sie haben die Möglichkeit, eine S7-Verbindung von einer CPU zu einer anderen über unterschiedliche Subnetze (PROFIBUS und PROFINET bzw. Industrial Ethernet) einzurichten. Das Vorgehen ist an Beispielen im Kapitel S7-Kommunikation (Seite 150) beschrieben.

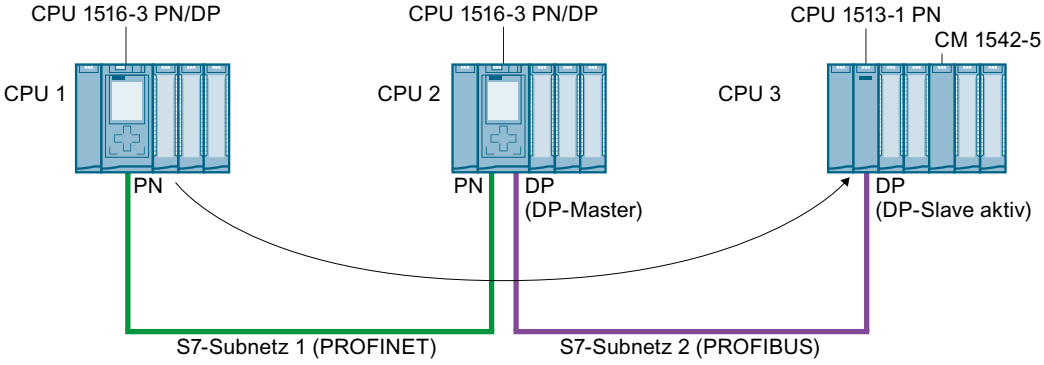


Bild 13-4 S7-Routing über CPU-CPU-Kommunikation

S7-Routing nutzen

Für die CPU wählen Sie im Dialog "Online verbinden" von STEP 7 die PG/PC-Schnittstelle und das S7-Subnetz aus. Das S7-Routing wird automatisch durchgeführt.

Anzahl der Verbindungen für S7-Routing

Die Anzahl der Verbindungen, die für S7-Routing in den S7-Routern (CPUs, CMs bzw. CPs) zur Verfügung stehen, finden Sie in den Technischen Daten, in den Gerätehandbüchern der jeweiligen CPU/CM/CP.

S7-Routing: Applikationsbeispiel

Das folgende Bild zeigt Ihnen als Applikationsbeispiel die Fernwartung einer Anlage durch ein PG. Die Verbindung kommt hierbei über zwei S7-Subnetz hinweg über eine Modemverbindung zu Stande.

Eine Fernverbindung über TeleService projektieren Sie in STEP 7 über "Online-Zugänge" bzw. "Online verbinden".

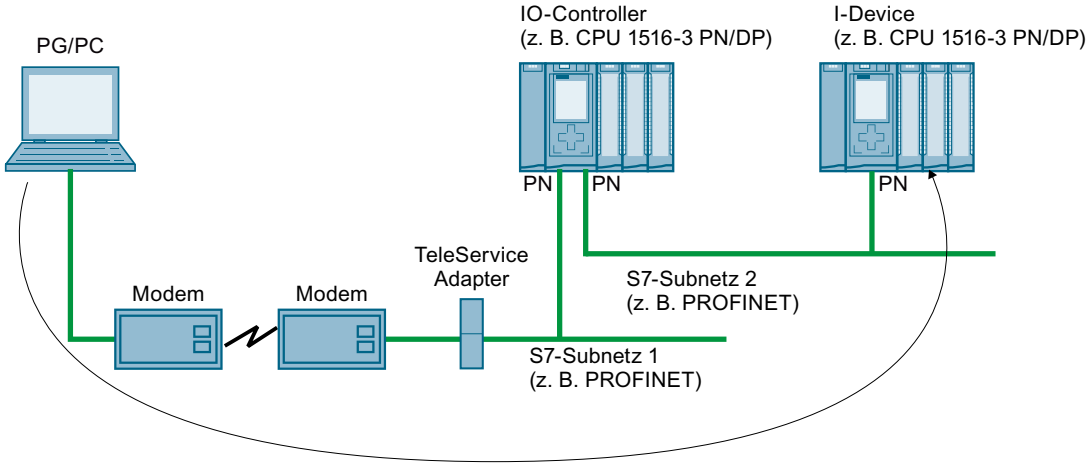


Bild 13-5 Fernwartung einer Anlage über TeleService

Weitere Informationen

- Die Belegung von Verbindungsressourcen beim S7-Routing ist beschrieben im Kapitel Belegung von Verbindungsressourcen (Seite 400).
- Ausführliche Informationen zum Einrichten von TeleService finden Sie in der Online-Hilfe STEP 7.
- Informationen zur HMI-Kommunikation finden Sie im Kapitel HMI-Kommunikation (Seite 124).
- Weitere Informationen zu S7-Routing und TeleService Adaptern finden Sie über die Suche im Internet unter folgenden Links:
  - Gerätehandbuch Industrie Software Engineering Tools TS Adapter IE Basic (https://support.industry.siemens.com/cs/ww/de/view/51311100)
  - Downloads zum TS Adapter (https://support.industry.siemens.com/cs/ww/de/ps/16006/dl)

13.3 IP-Forwarding

Weiterleitung von IP-Paketen mit IP-Forwarding

IP-Forwarding ist eine Funktion von Geräten, IP-Pakete zwischen 2 angeschlossenen IP-Subnetzen weiterzuleiten.

Die Funktion IP-Forwarding aktivieren/deaktivieren Sie in STEP 7. Wenn IP-Forwarding aktiviert ist, dann leitet die S7-1500 CPU empfangene, nicht an die CPU adressierte IP-Pakete an lokal angeschlossene IP-Subnetze weiter, bzw. an einen konfigurierten Router.

Das folgende Bild zeigt, wie ein PG auf Daten eines HMI-Geräts zugreift. PG und HMI-Gerät befinden sich in unterschiedlichen IP-Subnetzen. Die IP-Subnetze sind an die beiden Schnittstellen X1 und X2 der CPU angeschlossen.

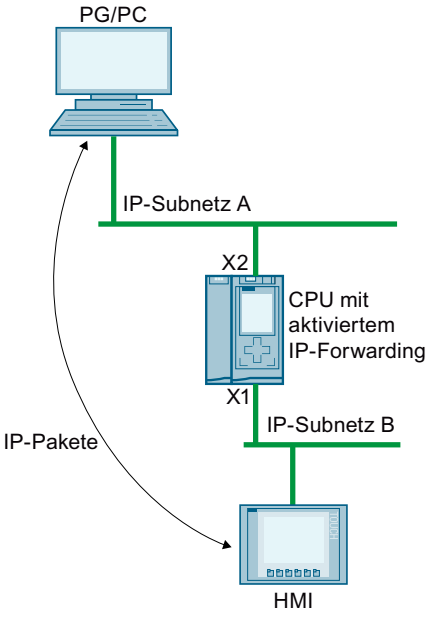


Bild 13-6 Zugriff eines PG auf ein HMI per IP-Forwarding

Einsatzgebiete

- Einfacher Zugriff von der Leitebene auf die Feldebene für Konfiguration und Parametrierung von Feldgeräten, z. B. per PDM oder Webbrowser
- Vereinfachte Einbindung von Devices für Remote-Zugriffe, z. B. für Diagnose bei Fernwartung oder Firmware-Update

Voraussetzungen für die Nutzung von IP-Forwarding

- S7-1500 CPU ab Firmware-Version V2.8
- Anzahl Ethernet-Schnittstellen:
  - Die CPU besitzt mindestens 2 Ethernet-Schnittstellen.
  - Oder die CPU besitzt eine Ethernet-Schnittstelle und ein CP 1543-1 ab Firmware-Version V2.2 stellt die andere Ethernet-Schnittstelle zur Verfügung. In diesem Fall muss die Funktion "Zugriff auf PLC über Kommunikationsmodul" in der CPU für den CP aktiviert sein.
- IP-Forwarding ist aktiviert.
- In jedem beteiligten Gerät entlang des Hin- und Rückwegs der IP-Pakete sind passende Default-Gateways/Routen parametriert.

IP-Routentabelle

Wenn IP-Forwarding aktiviert ist, dann leitet die CPU empfangene IP-Pakete weiter, die nicht an sie selbst adressiert sind. Wie die CPU die IP-Pakete weiterleitet, ist in ihrer internen IP-Routentabelle festgelegt.
Die CPU erstellt automatisch die IP-Routentabelle aus den folgenden Informationen der geladenen Hardware-Konfiguration:

- IP-Konfiguration der Ethernet-Schnittstellen
- Konfigurierter Router

Beispiel für eine Konfiguration mit IP-Forwarding

Das folgende Bild zeigt eine Beispielkonfiguration zusammen mit den erforderlichen IP-Adresseinstellungen und Router-Einstellungen.

- Ein PC am IP-Subnetz 192.168.4.0 kommuniziert mit einem HMI-Gerät am IP-Subnetz 192.168.2.0.
- An der CPU, Ethernet-Schnittstelle X3, ist die IP-Adresse eines Routers ("Standard-Gateway") konfiguriert; es handelt sich im Bild unten um das Gerät, das mit "IP-Router" bezeichnet ist.
  In STEP 7 konfigurieren Sie einen Router in den Eigenschaften der Schnittstelle unter "Ethernet-Adressen" > "IP-Protokoll".

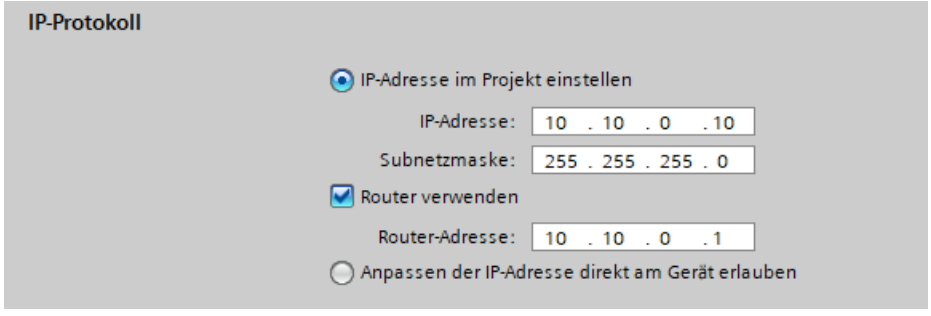


Bild 13-7 Router konfigurieren

- Für den PC, den IP-Router, das IO-Device und das HMI-Gerät sind ebenso die IP-Adressen eines Standard-Gateways bzw. die entsprechenden Routen eingetragen.

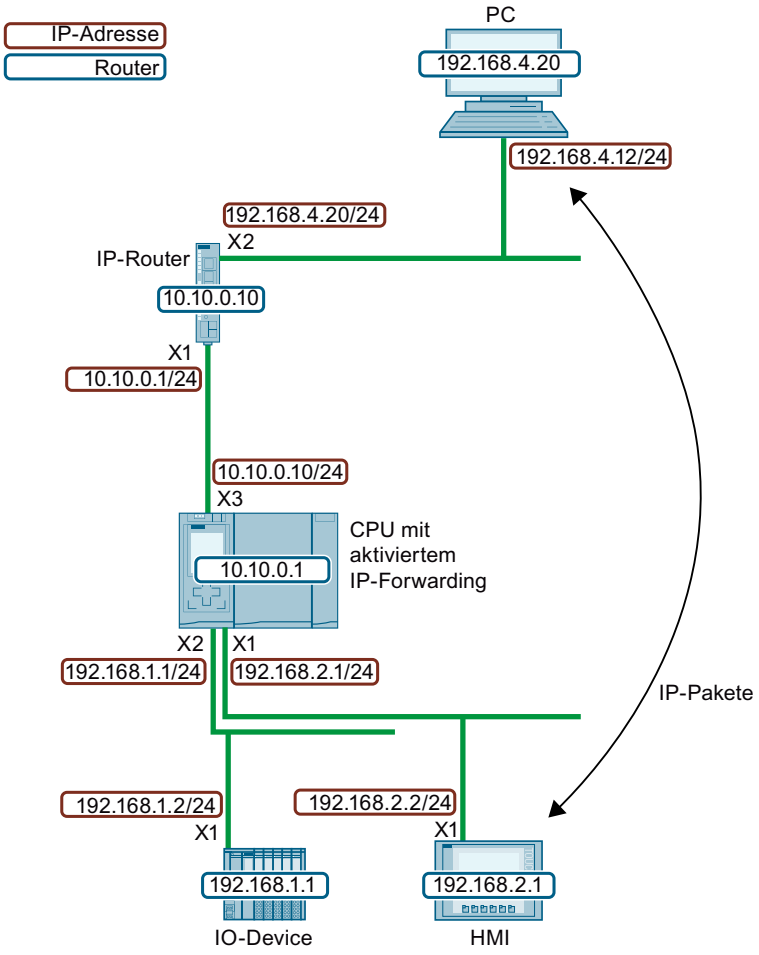


Bild 13-8 Beispielkonfiguration

Aus dieser Beispielkonfiguration ergibt sich für die CPU die folgende IP-Routentabelle.

Tabelle 13-1 IP-Routentabelle der CPU

Netzwerkziel	Schnittstelle	Gateway
0.0.0.0/0	10.10.0.10	10.10.0.1
192.168.1.0/24	192.168.1.1	-
192.168.2.0/24	192.168.2.1	-
10.10.0.0/24	10.10.0.10	-

Damit eine IP-Kommunikation zwischen dem PG/PC und dem HMI-Gerät möglich ist, müssen Sie sowohl im PC als auch im IP-Router zusätzliche IP-Routen zum IP-Subnetz des HMI-Geräts einrichten. Im HMI-Gerät projektieren Sie die IP-Adresse der CPU-Schnittstelle X1 als Standard-Gateway.

In einem Windows-Rechner richten Sie z. B. eine zusätzliche IP-Route in der Eingabeaufforderung ein über den Befehl "route add <Ziel-IP-Subnetz> mask <Subnetzmaske> <Gateway>". Dazu benötigen Sie allerdings gewisse Zugriffsrechte. Für dieses Beispiel geben Sie folgende Eingabeaufforderung ein:

- "route add 192.168.2.0 mask 255.255.255.0 192.168.4.20"

In einem IP-Router richten Sie zusätzliche Routen z. B. über ein Web-Interface ein. Für dieses Beispiel richten Sie folgende Route ein:

- Ziel-IP-Subnetz: 192.168.2.0
- Subnetzmaske: 255.255.255.0
- Gateway: 10.10.0.10

Einschränkungen

Für eine S7-1500 CPU können Sie neben dem Router ("Standard-Gateway") keine zusätzlichen IP-Routen konfigurieren. Das Netzwerkziel ist entweder ein angeschlossenes IP-Subnetz oder das Netzwerkziel ist über genau einen konfigurierbaren Router zu erreichen. Da die S7-1500 CPU keine zusätzlichen IP-Routen unterstützt, können Sie keine bidirektionalen IP-Router-Kaskaden aufbauen.

In der folgenden Konfiguration können Sie in der CPU entweder "Router 1" oder "Router 2" projektieren. Als Beispiel ist "Router 1" projektiert. In dem Fall können Sie "Router 2" nicht projektieren. Eine IP-Kommunikation zwischen PC und dem HMI-Gerät ist nicht möglich, weil die Route nicht in beide Richtungen durchgängig ist.

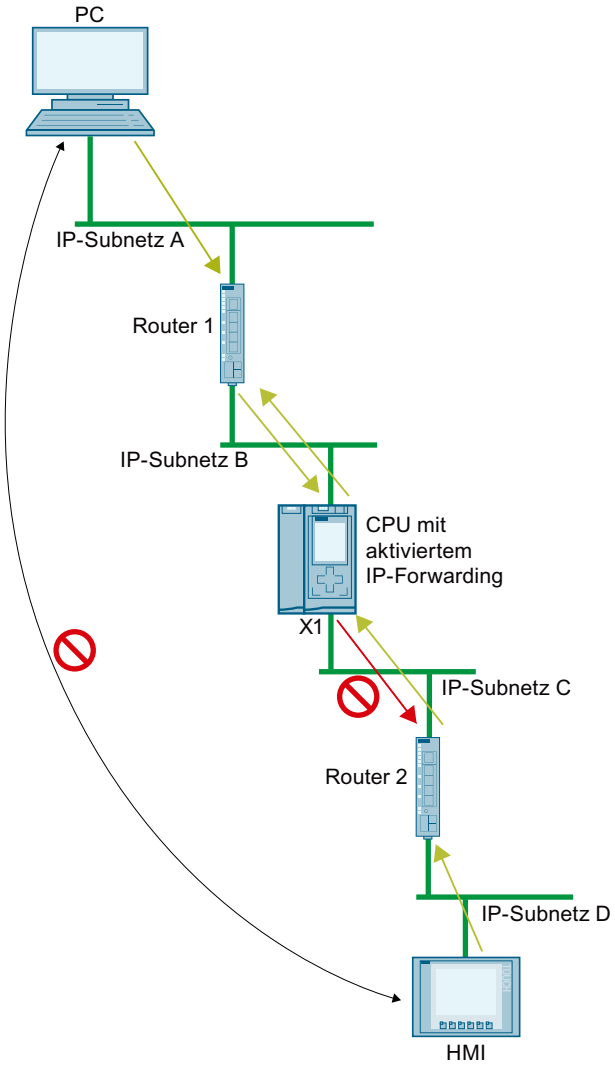


Bild 13-9 Nicht unterstützte IP-Routerkaskade

IP-Forwarding über die Schnittstelle eines CP

IP-Forwarding funktioniert auch über die Schnittstelle eines CP. Dafür müssen Sie für diesen CP die Funktion "Zugriff auf PLC über Kommunikationsmodul" in der CPU aktivieren.
Wie Sie die Funktion "Zugriff auf PLC über Kommunikationsmodul" aktivieren, finden Sie in der Onlinehilfe von STEP 7 beschrieben.

C/C++ Runtime der CPU-1518-4 PN/DP MFP über Schnittstellen X1 oder X2 erreichen

Wenn Sie für die CPU 1518-4 PN/DP MFP IP-Forwarding aktivieren, dann erreichen Sie über die Schnittstellen X1 und X2 nicht nur Geräte im IP-Subnetz von Schnittstelle X3, sondern auch die C/C++ Runtime. Von der C/C++ Runtime der CPU 1518-4 PN/DP MFP erreichen Sie alle Geräte in den IP-Subnetzen der Schnittstellen X1, X2 und X3.
Randbedingungen:

- IP-Forwarding ist für die CPU 1518-4 PN/DP MFP aktiviert.
- Die IP-Adresse der C/C++ Runtime und die IP-Adresse der Schnittstelle X3 liegen im selben IP-Subnetz.
- In der C/C++ Runtime sind die Routen zu den IP-Subnetzen an X1 und X2 eingetragen.
  In der C/C++ Runtime tragen Sie eine Route mit dem folgenden Befehl ein: "Route add-net <Ziel-IP-Subnetz> mask <Subnetzmaske> gw <Gateway>

Das folgende Bild zeigt eine Konfiguration, in der ein PC über die Schnittstelle X2 auf die C/C++ Runtime der CPU 1518-4 PN/DP MFP zugreift.

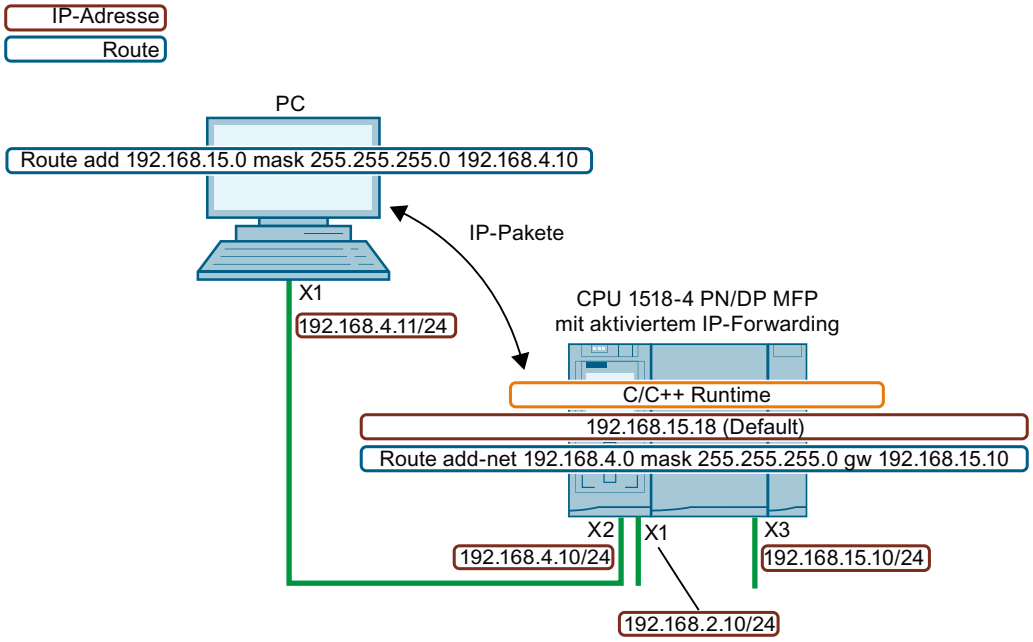


Bild 13-10 Zugriff auf C/C++ Runtime über die Schnittstelle X2

Netzwerksicherheit berücksichtigen beim IP-Forwarding

Wenn Sie IP-Forwardung für eine CPU aktivieren, dann ermöglichen Sie einen Zugriff "von außen" auf Geräte, die eigentlich nur von der CPU erreichbar sind und gesteuert werden. Diese Geräte sind deshalb auch in der Regel ungeschützt gegen Angriffe.
Das folgende Bild zeigt, wie Sie ihr Automatisierungssystem gegen unbefugten Zugriff schützen.

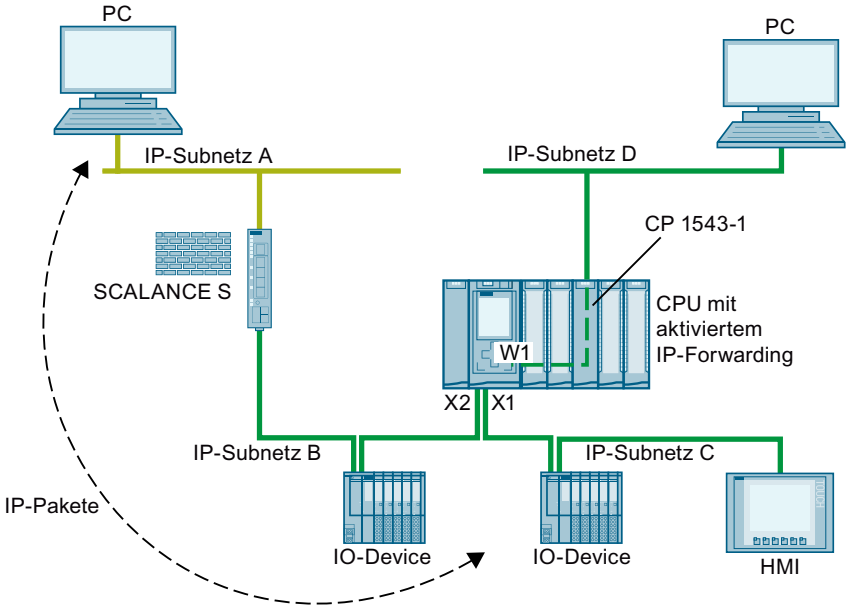


Bild 13-11 Netzwerksicherheit beim IP-Forwarding

- Über die Schnittstellen X1 und X2 erreicht die CPU alle Geräte innerhalb der CPU-nahen dunkelgrünen IP-Subnetze B und C.
- In der CPU ist ein Router SCALANCE S konfiguriert. Über den Router erreicht die CPU die Geräte im entfernten hellgrünen IP-Subnetz A.
- In der CPU ist für den CP 1543 die Funktion "Zugriff auf PLC über Kommunikationsmodul" aktiviert. Über die Schnittstelle W1 erreicht die CPU alle Geräte innnerhalb des IP-Subnetzes D.

Wenn IP-Forwarding in der CPU aktiviert ist, dann kann ein Gerät aus dem IP-Subnetz A auf jedes Gerät innerhalb der CPU-nahen IP-Subnetze B, C und D zugreifen.
Schützen Sie ihr Automatisierungssystem und angeschlossene Geräte gegen unbefugten Zugriff von außen.
Trennen Sie die CPU-nahen IP-Subnetze mit einer Firewall von den entfernten IP-Subnetzen. Verwenden Sie dafür zum Beispiel die Security Module SCALANCE S mit integrierter Firewall. Wie Sie eine Automatisierungszelle durch die Security Module SCALANCE S602 V3 und SCALANCE S623 mit einer Firewall schützen, finden Sie beschrieben in diesem Anwendungsbeispiel (https://support.industry.siemens.com/cs/ww/de/view/22376747).

IP-Forwarding aktivieren/deaktivieren

Um IP-Forwarding zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie die CPU in der Netzsicht von STEP 7 (TIA Portal).
2. Navigieren Sie im Inspektorfenster in den Eigenschaften der CPU zu "Allgemein" > "Erweiterte Konfiguration" > "IP-Forwarding".
3. Aktivieren Sie im Bereich "Konfiguration IPv4-Forwarding" das Optionskästchen "IPv4 für Schnittstellen dieser PLC aktivieren".

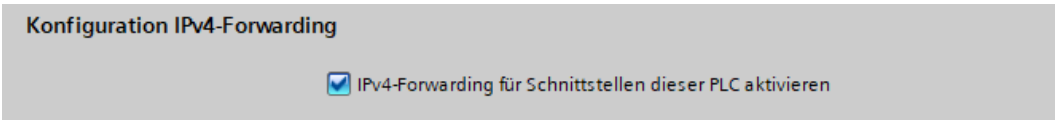


Bild 13-12 IP-Forwarding aktivieren

Ergebnis: IP-Forwarding ist für alle Schnittstellen der S7-1500 CPU aktiviert.
Sie deaktivieren IP-Forwarding, indem Sie das Optionskästchen "IPv4-Forwarding für Schnittstellen dieser PLC aktivieren" deaktivieren.

13.4 Datensatz-Routing

Definition Datensatz-Routing

Daten können von einer Engineering Station von PROFINET aus über verschiedene Netzwerke hinweg an Feldgeräte gesendet werden. Da die Engineering Station die Feldgeräte über genormte Datensätze anspricht und diese Datensätze über S7-Geräte geroutet werden, hat sich für diese Art des Routings der Begriff "Datensatz-Routing" etabliert.
Die Daten, die beim Datensatz-Routing versendet werden, beinhalten außer der Parametrierung für die beteiligten Feldgeräte (DP-Slaves) auch gerätespezifische Informationen, z. B. Sollwerte, Grenzwerte.
Datensatz-Routing wird z. B. eingesetzt, wenn Feldgeräte verschiedener Hersteller zum Einsatz kommen. Die Feldgeräte werden zur Parametrierung und Diagnose über genormte Datensätze (PROFINET) angesprochen.

Datensatz-Routing mit STEP 7

Sie können mit STEP 7 Datensatz-Routing durchführen, indem Sie über die TCI-Schnittstelle (Tool Calling Interface) ein Gerätetool (z. B. PCT) aufrufen und Aufrufparameter übergeben. Das Gerätetool nutzt für die Kommunikation mit dem Feldgerät die Kommunikationswege, die auch STEP 7 nutzt.
Für diese Art des Routings ist außer der Installation des TCI-Tools auf dem STEP 7-Rechner keine Projektierung erforderlich.

Beispiel: Datensatz-Routing mit dem Port Configuration Tool (PCT)

Mit dem Port Configuration Tool (PCT) ist es möglich, die IO-Link Master der ET200 zu konfigurieren und daran angeschlossene IO-Link Devices zu parametrieren. Die Subnetze sind über Datensatz-Router verbunden. Datensatz-Router sind z. B. CPUs, CPs, IMs, IO-Link Master.
Welche Konstellationen von Datensatz-Routern das PCT unterstützt, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/87611392).
Das folgende Bild zeigt eine Beispielkonfiguration für Datensatz-Routing mit PCT.

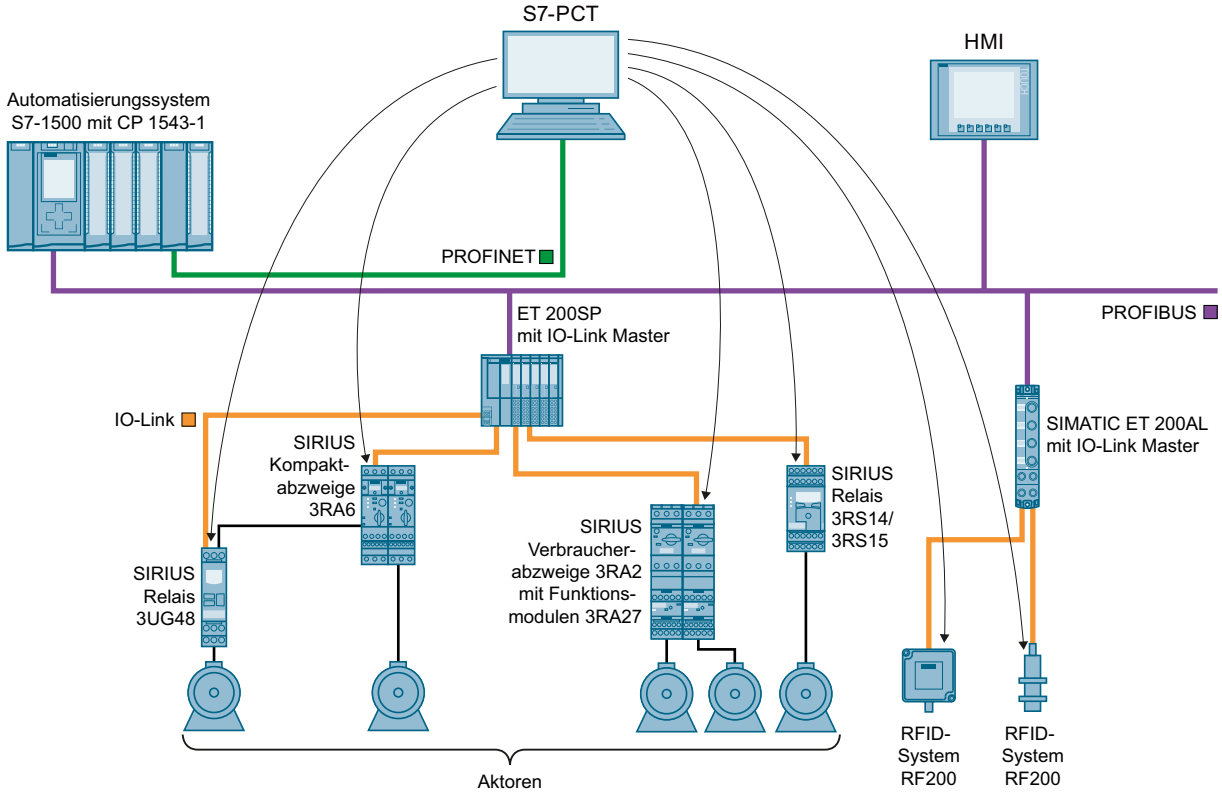


Bild 13-13 Beispielkonfiguration für Datensatz-Routing mit PCT

Weitere Informationen

- Welcher Unterschied zwischen "normalen Routing und Datensatzrouting besteht, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/7000978).
- Ob die eingesetzte CPU, der CP oder das CM Datensatz-Routing unterstützt, finden Sie in den entsprechenden Gerätehandbüchern beschrieben.
- Die Belegung von Verbindungsressourcen beim Datensatz-Routing ist beschrieben im Kapitel Belegung von Verbindungsressourcen (Seite 400).
- Weitere Informationen zur Konfiguration mit STEP 7 finden Sie in der Onlinehilfe STEP 7.

13.5 Virtuelle Schnittstelle für IP-basierte Anwendungen

Die S7-1500 CPU bietet ab Firmware-Version 2.8 die Möglichkeit, ihre IP-basierten Applikationen wie z. B. OPC UA nicht nur über ihre lokalen (PN-) Schnittstellen zu erreichen, sondern auch über die Schnittstellen von Kommunikationsprozessoren in derselben Station. Die unterstützten Kommunikationsprozessoren finden Sie in den Voraussetzungen. Ein Kommunikationspartner erreicht diese IP-basierten Anwendungen über eine virtuelle Schnittstelle, die im TIA Portal ab Version V16 konfigurierbar ist. Die virtuelle Schnittstelle hat die Bezeichnung W1 (gemäß IEC 81346-2).

Merkmale der virtuellen Schnittstelle

Die virtuelle Schnittstelle ist keine voll diagnosefähige Schnittstelle mit den bekannten Eigenschaften herkömmlicher Schnittstellen. In den grafischen Sichten wird die virtuelle Schnittstelle nicht angezeigt, da die interne Verbindung über den Rückwandbus kein S7-Subnetz darstellt und keine Ports hat. Eine physikalische Verbindung durch ein Netzwerkkabel kann daher nicht hergestellt werden.

Die IP-Adresse der virtuellen Schnittstelle wird angezeigt (im TIA Portal, im Display der CPU) und kann konfiguriert werden.

Die folgenden Kommunikationsmöglichkeiten können über die virtuelle Schnittstelle W1 genutzt werden:

- OPC UA (Client und Server)
- Programmierte OUC-Verbindungen
- PG/HMI-Kommunikation
- Partner-Zugriffe von S7-CPUs über PUT/GET-Anweisungen

Die aktivierte Schnittstelle kann in Dialogen verwendet werden, in denen IP-basierte Verbindungen konfiguriert werden.

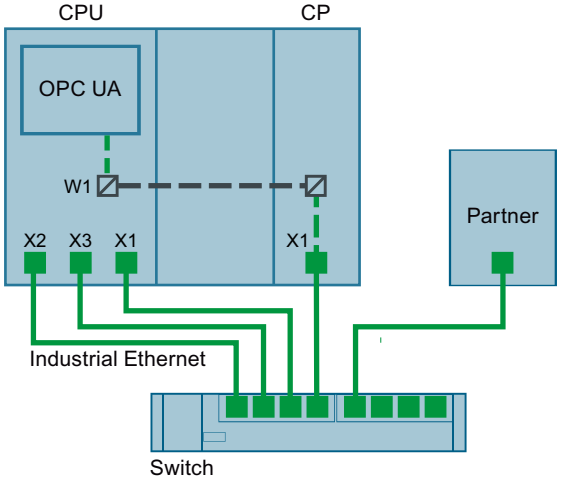


Bild 13-14 Prinzip der virtuellen Schnittstelle

Die virtuelle Schnittstelle hat gegenüber herkömmlichen Schnittstellen folgende Einschränkungen:

- Über die virtuelle Schnittstelle kann nicht auf den Webserver zugegriffen werden.
- Die Online-Sicherung ist über ein angeschlossenes Programmiergerät mit dem TIA Portal nicht möglich.
- Wenn CPU und Kommunikationspartner über die virtuelle Schnittstelle miteinander verbunden sind, können sie keine Daten über LLDP (Link Layer Discovery Protocol) austauschen.
- Der S7-Routing-Dienst verwendet die virtuelle Schnittstelle W1 nicht.

Voraussetzung

Für die Erreichbarkeit eines CPU-Dienstes über die Ethernet-Schnittstelle eines CPs müssen die folgenden Voraussetzungen erfüllt sein:

- S7-1500 CPU ab Firmware-Version V2.8
- CP 1543-1 ab Firmware-Version V2.2

Empfehlung: Verwenden Sie einen CP 1543-1 ab Firmware-Version V3.0. Ab dieser Version sind die Security-Funktionen (Firewall) auch für die virtuelle Schnittstelle verfügbar und es muss keine zusätzliche Firewall zwischen der Station und einem unsicheren Netz installiert werden.

Konfiguration der virtuellen Schnittstelle W1

In den Eigenschaften einer S7-1500 CPU ab Firmware-Version V2.8 können Sie unter "Erweiterte Konfiguration > Zugriff auf PLC über Kommunikationsmodul" der virtuellen Schnittstelle W1 ein gestecktes Kommunikationsmodul zuweisen. Über dieses kann dann von außen auf die CPU zugegriffen werden. Wenn keine CPs gesteckt sind oder die gesteckten CPs den Zugriff auf die CPU nicht unterstützen, ist die Auswahl leer.

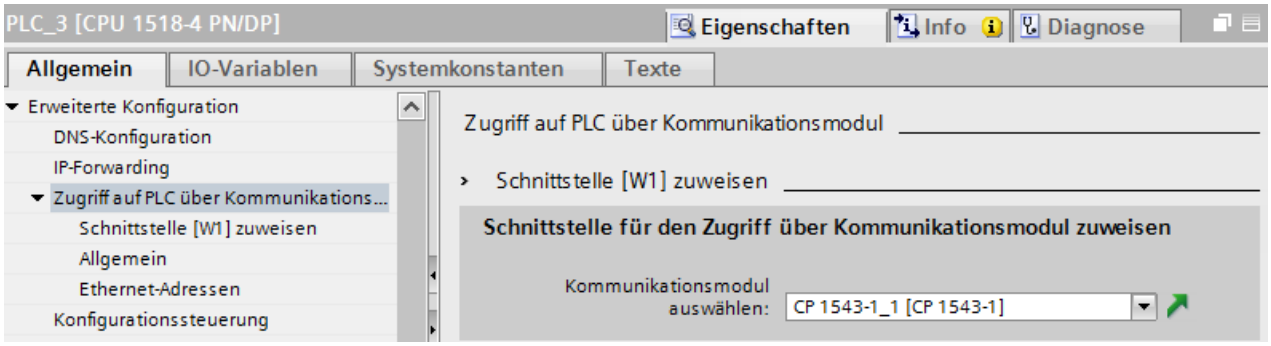


Bild 13-15 Auswahl des CP in den Eigenschaften der CPU

Nach der Auswahl des CP werden die Angaben und Parameter für die virtuelle Schnittstelle angezeigt. Hier können Sie die Einstellungen für das IP-Protokoll und die PROFINET-Parameter bearbeiten:

- Das IP-Subnetz ist frei wählbar, ebenso wie beim CP. Das IP-Subnetz geben Sie über die Subnetzmaske und IP-Adresse der virtuellen Schnittstelle ein.
- Beachten Sie bei der Eingabe des IP-Subnetzes für die virtuelle Schnittstelle, dass Sie nicht dasselbe IP-Subnetz verwenden wie für die lokalen Schnittstellen der CPU.

Nach Eingabe der IP-Adresse ist diese im Eigenschaftsdialog des OPC UA-Servers in der Liste der Server-Adressen aufgeführt. Durch diese Einstellungen erhält die CPU die neue virtuelle Schnittstelle W1, über die oben beschriebene CPU-Dienste wie der OPC UA-Server über ein Kommunikationsmodul erreichbar sind. Die entsprechend angelegten Verbindungen und S7-Kommunikation (z. B. HMI und BSEND, BRCV) gehen über diese Schnittstelle. Der OPC UA-Server erlaubt keine Auswahl einer bestimmten Schnittstelle (Auswahl über eine IP-Adresse), es sind entweder alle oder keine möglich.

HINWEIS

Die IP-Adresse der virtuellen Schnittstelle steht nicht als W1 im Geräte-Display unter den aktuell angezeigten lokalen Schnittstellen (Xn), sondern unter den "Adressen" im Abschnitt "Einstellungen". Die virtuelle Schnittstelle ist auch dann sichtbar, wenn kein CP gesteckt oder die virtuelle Schnittstelle nicht aktiviert ist. Wenn keine IP-Suite verfügbar ist, sind die IP-Adresse und die Subnetzmaske 0.0.0.0.

Wenn Sie die konfigurierten und geladenen IP-Adressparameter der virtuellen Schnittstelle über Display, T_CONFIG-Anweisung oder online ändern, dann wird nach einem Neustart der CPU wieder die geladene Konfiguration aktiv.

Konfigurationsänderungen am CP

Eine Änderung des zugewiesenen Kommunikationsmoduls kann sich auf die Konfiguration der virtuellen Schnittstelle auswirken:

- In den Eigenschaften der CPU:
  - Zuweisung eines anderen CP: Die Konfiguration wird für den neuen CP übernommen.
  - Abwahl des zugewiesenen CP: Die virtuelle Schnittstelle W1 wird deaktiviert und die Konfiguration geht verloren.
    Bei erneuter Zuweisung eines CP muss die Konfiguration erneut vorgenommen werden.
- Am Gerät:
  - Verschieben des CP: Wenn der CP nur auf einen anderen Steckplatz des Geräts verschoben wird, bleibt die Konfiguration weiterhin gültig.
  - Entfernen des CP: Wenn der CP gelöscht oder in ein anderes Gerät verschoben wird, bleibt die Konfiguration erhalten. In der Klappliste der CPU wird der CP als fehlend angezeigt und die Übersetzung der Konfiguration zeigt einen Fehler an. Zur Behebung kann der fehlende CP abgewählt oder ein anderer CP zugewiesen werden.

Anzeige in der Diagnose und den Systemkonstanten

Die virtuelle Schnittstelle W1 wird in der Diagnosesicht unter "Online & Diagnose" angezeigt. In den CPU-Eigenschaften wird die Hardware-Kennung der virtuellen Schnittstelle in den Systemkonstanten angezeigt.

Einstellungen im Kommunikationsmodul (CP 1543-1 ab FW-Version V3.0)

Ab Firmware-Version V3.0 können Sie die CP-interne Firewall nutzen, um den Datenverkehr über die virtuelle Schnittstelle abzusichern. Um die Firewall im Kommunikationsmodul zu aktivieren gehen Sie im geschützten Projekt folgendermaßen vor:

1. Selektieren Sie im Arbeitsbereich von STEP 7 das Kommunikationsmodul.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften > Security".
3. Aktivieren Sie die Option "Aktiviere Security-Funktionen".
   Im Inspektorfenster erscheinen nun die konfigurierbaren Security-Funktionen.
4. Aktivieren Sie die Option "Firewall aktivieren".
   Im Inspektorfenster können Sie nun die Verwendung der virtuellen Schnittstelle W1 zulassen.

Den Datenverkehr über die virtuelle Schnittstelle W1 der CPU können Sie nur für den Dienst OPC UA absichern.

HINWEIS

Prüfung der manuellen Konfiguration

Wenn die Firewall aktiviert ist, müssen Sie manuell prüfen, ob die gewünschten Dienste von der Firewall zugelassen werden. Aktivieren Sie für die IP- und MAC-Filter nur die Dienste, die Sie auch über die CP-Schnittstelle erreichen möchten. Beachten Sie im Infosystem des TIA Portals die Hinweise für die Security-Einstellungen und Firewall-Regeln der S7-1500-CPs.

Einstellungen im Kommunikationsmodul (CP 1543-1, FW-Version V2.2, < V3.0)

Die Security-Funktionen des CP 1543-1 mit einer Firmware-Version kleiner V3.0 können den Datenverkehr nicht über die virtuelle Schnittstelle absichern. Im TIA Portal können Sie zwar die Security-Funktionen aktivieren, eine solche Konfiguration ist aber nicht übersetzbar.

ACHTUNG

Verbindung mit unsicheren Netzen

Wenn Sie den CP mit einem unsicheren Netz verbinden, müssen Sie unbedingt eine zusätzliche Firewall zwischen dem CP und dem unsicheren Netz schalten. Verwenden Sie zum Beispiel die Security Module SCALANCE S602 V3 und SCALANCE S623 mit integrierter Firewall.

14

Verbindungsressourcen

14.1 Verbindungsressourcen einer Station

Einleitung

Einige Kommunikationsdienste benötigen Verbindungen. Verbindungen belegen im Automatisierungssystem (Station) Ressourcen. Die Verbindungsressourcen bekommt die Station von den CPUs, Kommunikationsprozessoren (CP) und Kommunikationsmodulen (CM) zur Verfügung gestellt.

Verbindungsressourcen einer Station

Die zur Verfügung stehenden Verbindungsressourcen sind abhängig von den eingesetzten CPUs, CPs und CMs und dürfen eine maximale Anzahl pro Station nicht überschreiten.
Die maximale Anzahl der Ressourcen einer Station wird durch die CPU bestimmt.

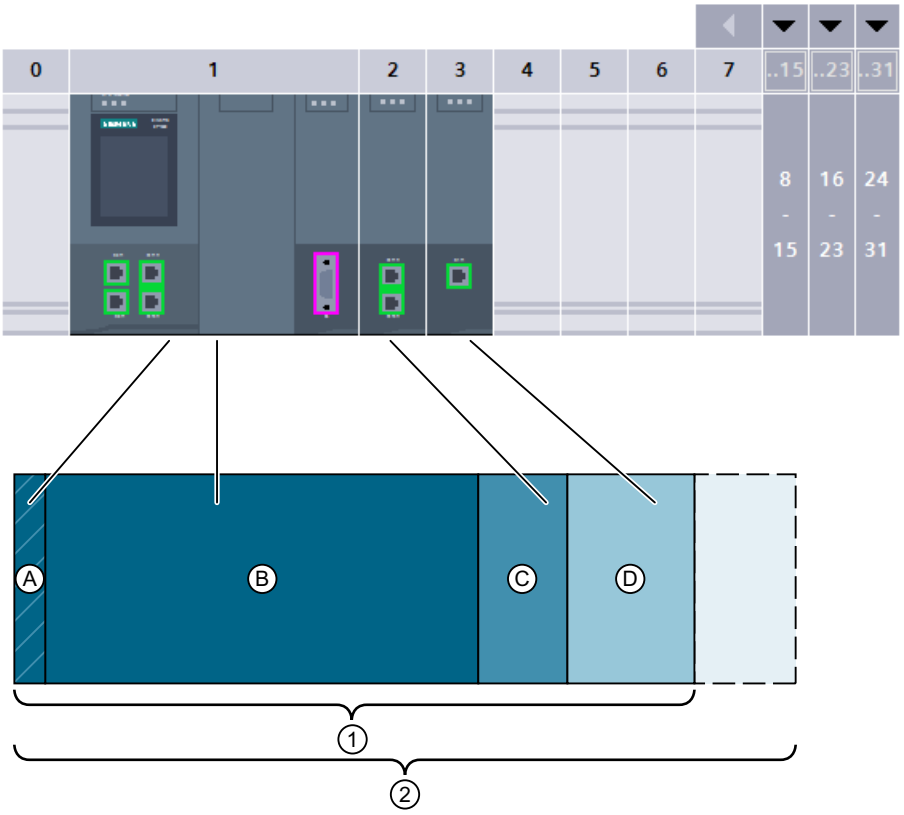
Reservierte Verbindungsressourcen

Jede CPU bringt reservierte Verbindungsressourcen für PG-, HMI- und Webserver-Kommunikation mit. Damit ist z. B. sichergestellt, dass ein PG immer mindestens eine Online-Verbindung mit der CPU aufbauen kann, unabhängig davon, wie viele andere Kommunikationsdienste bereits Verbindungsressourcen belegen.

Dynamische Verbindungsressourcen

Daneben gibt es dynamische Ressourcen. Die Differenz zwischen der maximalen Anzahl der Verbindungsressourcen und der Anzahl reservierter Verbindungsressourcen ist die maximale Anzahl dynamischer Verbindungsressourcen.
Aus dem Pool der dynamischen Verbindungsressourcen bedienen sich die Kommunikationsdienste PG-Kommunikation, HMI-Kommunikation, S7-Kommunikation, Open User Communication, Web-Kommunikation, OPC UA Client/Server-Kommunikation und sonstige Kommunikation.

Das folgende Bild zeigt beispielhaft, wie einzelne Komponenten Verbindungsressourcen einer S7-1500-Station zur Verfügung stellen.



① Verfügbare Verbindungsressourcen der Station, davon

- A: Reservierte Verbindungsressourcen der Station
- A + B: Verbindungsressourcen der CPU 1518
- C: Verbindungsressourcen des Kommunikationsmoduls CM 1542-1
- D: Verbindungsressourcen des Kommunikationsprozessors CP 1543-1

② Maximale Verbindungsressourcen der Station am Beispiel einer Konfiguration aus CPU 1518, CM 1542-1 und CP 1543-1

Bild 14-1 Verbindungsressourcen einer Station

Anzahl Verbindungsressourcen einer Station

Tabelle 14-1 Maximal unterstützte Verbindungsressourcen für einige CPU-Typen

Verbindungsressourcen einer Station	**1511C**	**1511 1512C 1513**	**1515 1516**	**1517**	**1518**
Maximale Verbindungsressourcen der Station	96	128	256	320	384
davon reserviert	10				
davon dynamisch	86	118	246	310	374
Verbindungsressourcen der CPU	64	88	128	288	320

Verbindungsressourcen einer Station	1511C	1511 1512C 1513	1515 1516	1517	1518
Max. zusätzlich nutzbare Verbindungsressourcen durch Stecken von CMs/CPs	32	40	128	32	64
Zusätzliche Verbindungsressourcen CM 1542-1	64				
Zusätzliche Verbindungsressourcen CP 1543-1	118				
Zusätzliche Verbindungsressourcen CM 1542-5	48				
Zusätzliche Verbindungsressourcen CP 1542-5	16				

Wie viele Verbindungsressourcen eine CPU bzw. ein Kommunikationsmodul unterstützt, finden Sie in den Gerätehandbüchern in den Technischen Daten beschrieben.

Beispiel

Sie haben eine CPU 1516-3 PN/DP mit einem Kommunikationsmodul CM 1542-1 und einem Kommunikationsprozessor CP 1542-5 konfiguriert.

- Maximale Verbindungsressourcen der Station: **256**
- Verfügbare Verbindungsressourcen:
  - CPU 1516-3 PN/DP: 128
  - CM 1542-1: 64
  - CP 1542-5: 16
  - Gesamt: **208**

Der Aufbau stellt 208 Verbindungsressourcen zur Verfügung. Durch Hinzufügen weiterer Kommunikationsmodule kann die Station maximal 48 weitere Verbindungsressourcen unterstützen.

Reservierte Verbindungsressourcen

Für Stationen mit S7-1500 CPU, ET 200SP CPU und ET 200pro CPU auf Basis S7-1500 sind 10 Verbindungsressourcen reserviert:

- 4 für PG-Kommunikation, die von STEP 7 benötigt werden für z. B. Test- und Diagnosefunktionen oder das Laden in die CPU
- 4 für HMI-Kommunikation, die durch die ersten, in STEP 7 projektierten HMI-Verbindungen belegt werden
- 2 für Kommunikation zum Webserver

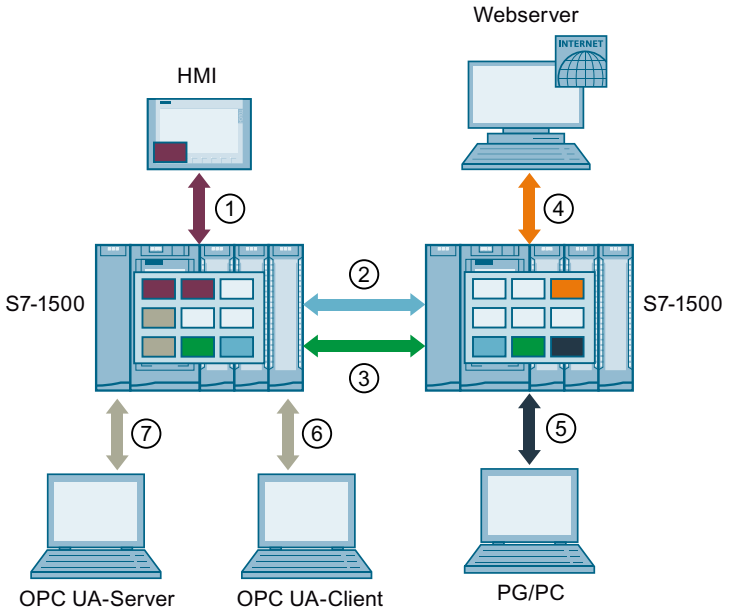
Weitere Informationen

Informationen zu den Verbindungsressourcen des redundanten Systems S7-1500R/H finden Sie im Kapitel Verbindungsressourcen des redundanten Systems S7-1500R/H (Seite 424).

14.2 Belegung von Verbindungsressourcen

Überblick - Belegung von Verbindungsressourcen

Das folgende Bild zeigt, wie verschiedene Verbindungen die Ressourcen der S7-1500 belegen.



① HMI-Kommunikation: siehe unten

② Open User Communication: Verbindungen der Open User Communication belegen in jedem Endpunkt eine Verbindungsressource.

③ S7-Kommunikation: Verbindungen der S7-Kommunikation belegen in jedem Endpunkt eine Verbindungsressource.

④ Web-Kommunikation: Die Webserver-Verbindung belegt in der Station mindestens eine Verbindungsressource. Die Anzahl der belegten Verbindungen hängt vom Browser ab.

⑤ PG-Kommunikation: Die PG-Verbindung belegt in der Station eine Verbindungsressource.

⑥ OPC UA-Client/Server-Kommunikation: Verbindungsressourcenbelegung beim Server siehe unten

⑦ OPC UA-Client/Server-Kommunikation: Verbindungsressourcenbelegung beim Client siehe unten

Verbindungsressource für HMI-Kommunikation

Verbindungsressource für OpenUser-Kommunikation

Verbindungsressource für S7-Kommunikation

Verbindungsressource für Web-Kommunikation

Verbindungsressource für PG-Kommunikation

Verbindungsressource für OPC UA Client/Server-Kommunikation

Bild 14-2 Belegung von Verbindungsressourcen

Verbindungsressourcen für HMI-Kommunikation

Bei HMI-Kommunikation ist die Belegung von Verbindungsressourcen in der Station abhängig vom eingesetzten HMI-Gerät.

Tabelle 14-2 Maximal belegte Verbindungsressourcen für verschiedene HMI-Geräte

HMI-Gerät	max. belegte Verbindungsressourcen der Station pro HMI-Verbindung
Basic Panel	1
Unified Basic Panel	3
Comfort Panel	2[1]
Unified Comfort Panel	3
Mobile Panel	2[1]
RT Advanced	2[1]
RT Professional	3
Unified PC	3

[1] Wenn Sie keine Systemdiagnose und keine Meldungsprojektierung nutzen, dann belegt die Station pro HMI-Verbindung nur eine Verbindungsressource.

Beispiel: Sie haben für eine CPU 1516-3 PN/DP folgende HMI-Verbindungen konfiguriert:

- Zwei HMI-Verbindungen zu einem HMI TP700 Comfort. (jeweils 2 Verbindungsressourcen)
- Eine HMI-Verbindung zu einem HMI KTP1000 Basic. (1 Verbindungsressource)

Insgesamt werden in der CPU 5 Verbindungsressourcen für HMI-Kommunikation belegt.

Verbindungsressourcen für OPC UA-Client-Kommunikation

Jede Verbindung, die der OPC UA-Client der CPU zu einem OPC UA-Server aufgebaut hat, belegt eine Verbindungsressource in der Station.
Beim Auf- und Abbau von einer OPC UA-Verbindung belegt der OPC UA-Client jeweils temporär eine weitere Verbindungsressource. Diese Verbindungsressource wird gemäß RFC 793 nach einer Wartezeit von ca. 60 s wieder freigegeben.

HINWEIS

Ressourcenmangel durch temporäre Verbindungsressourcen

Im folgenden Fall tritt ein Mangel der Verbindungsressourcen auf:

- Der OPC UA-Client der CPU baut gleichzeitig mehrere Verbindungen auf- oder ab.
- Die Anzahl der verfügbaren Verbindungsressourcen der Station reicht nicht für die permanenten und temporären Verbindungsressourcen der OPC UA Client-Kommunikation aus.

Achten Sie darauf, daß immer genügend verfügbare Verbindungsressourcen für den Auf- und Abbau von OPC UA-Verbindungen in der Station vorhanden sind.

Maßnahmen:

- Planen Sie genug Reserve für die OPC UA-Client-Verbindungen ein.
- Bauen Sie die OPC UA-Verbindungen gegebenenfalls nacheinander auf bzw. ab.

Verbindungsressourcen für Routing

Für die Übertragung von Daten über S7-Subnetze hinweg ("S7-Routing") wird eine S7-Verbindung zwischen zwei CPUs aufgebaut. Die S7-Subnetze sind über Netzübergänge, sogenannte S7-Router miteinander verbunden. CPUs, CMs und CPs in S7-1500 sind S7-Router.

Für eine geroutete S7-Verbindung gilt Folgendes:

- Eine geroutete Verbindung belegt in beiden Endpunkten jeweils eine Verbindungsressource, STEP 7 zeigt diese Verbindungsressourcen in der Tabelle "Verbindungsressourcen" an.
- Im S7-Router werden zwei spezielle Verbindungsressourcen für S7-Routing belegt. STEP 7 zeigt die speziellen Verbindungsressourcen für S7-Routing nicht in der Tabelle "Verbindungsressourcen" an. Die Anzahl der Ressourcen für S7-Routing ist CPU abhängig. Sie finden der Ressourcen für S7-Routing in den Technischen Daten der CPU unter "Anzahl S7-Routing Verbindungen".

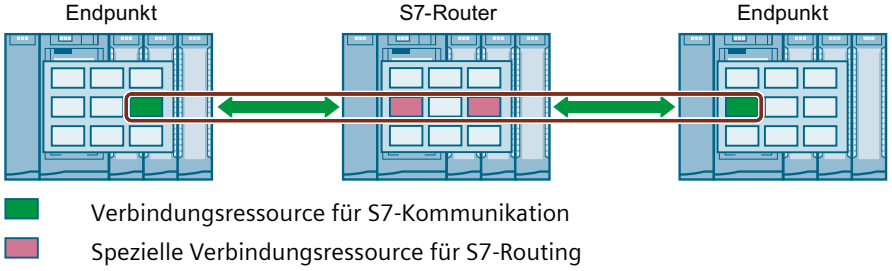


Bild 14-3 Verbindungsressourcen beim S7-Routing

Datensatz-Routing ermöglicht ebenfalls die Übertragung von Daten über S7-Subnetze hinweg, von einer Engineering Station, die am PROFINET angeschlossen ist, über PROFIBUS an diverse Feldgeräte.

Wie beim S7-Routing werden auch beim Datensatz-Routing in jedem Datensatz-Router zwei der speziellen Verbindungsressourcen für S7-Routing belegt.

HINWEIS

Verbindungsressourcen beim Datensatz-Routing

Im Datensatz-Router werden beim Datensatz-Routing zwei spezielle Verbindungsressourcen für S7-Routing belegt. Weder die Datensatzverbindung, noch die belegten Verbindungsressourcen werden in der Tabelle Verbindungsressourcen angezeigt.

Wann werden Verbindungsressourcen belegt?

Der Zeitpunkt für die Belegung der Verbindungsressourcen hängt davon ab, wie die Verbindung eingerichtet wird (siehe Kapitel Einrichten einer Verbindung (Seite 43)).

- **Programmiertes Einrichten einer Verbindung**:
  Sobald im Anwenderprogramm der CPU eine Anweisung zum Aufbau einer Verbindung (TSEND_C/TRCV_C oder TCON) aufgerufen wird, wird eine Verbindungsressource belegt.
  Durch entsprechende Parametrierung des CONT Parameters der Anweisungen TSEND_C/TRCV_C bzw. den Aufruf der Anweisung TDISCON kann die Verbindung nach der Datenübertragung abgebaut und die Verbindungsressource freigegeben werden. Wenn die Verbindung abgebaut ist, stehen die Verbindungsressourcen in der CPU/CP/CM wieder zur Verfügung.

- **Konfigurierte Verbindungen** (z. B. S7-Verbindung):
  Wenn Sie eine Verbindung in STEP 7 konfiguriert haben, dann wird die Verbindungsressource belegt, sobald die Hardware-Konfiguration in die CPU geladen wurde.
  Nach der Nutzung einer projektierten Verbindung zur Datenübertragung wird die Verbindung nicht abgebaut. Die Verbindungsressource bleibt dauerhaft belegt. Um die Verbindungsressource wieder frei zu geben, müssen Sie die projektierte Verbindung in STEP 7 löschen und die geänderte Projektierung in die CPU laden.
- **PG-Verbindung**:
  Sobald Sie das PG mit einer CPU online in STEP 7 verbunden haben, werden Verbindungsressourcen belegt.
- **Webserver**:
  Solange Sie den Webserver der CPU in einem Browser geöffnet haben, werden Verbindungsressourcen in der CPU belegt.
- **OPC UA-Server**
  Jede Verbindung zum OPC UA-Server der CPU belegt eine Verbindungsressource in der Station. Diese Verbindungsressource wird beim Verbindungsabbau sofort freigegeben.
- **OPC UA-Client**
  Jede Verbindung, die der OPC UA-Client der CPU zu einem OPC UA-Server aufgebaut hat, belegt eine Verbindungsressource in der Station. Beim Aufbau einer OPC UA-Verbindung belegt der OPC UA-Client temporär eine weitere Verbindungsressource. Beim Abbau einer OPC UA-Verbindung wird die Verbindungsressource gemäß RFC 793 erst nach einer Wartezeit von ca. 60 s wieder freigegeben.

Überwachung der maximal möglichen Anzahl Verbindungsressourcen

Offline

STEP 7 überwacht beim Konfigurieren von Verbindungen die Belegung von Verbindungsressourcen. Ein Überschreiten der maximal möglichen Anzahl von Verbindungsressourcen meldet STEP 7 mit einer entsprechenden Warnung.

Online

Die CPU überwacht den Verbrauch von Verbindungsressourcen im Automatisierungssystem. Wenn Sie im Anwenderprogramm mehr Verbindungen aufbauen, als das Automatisierungssystem Verbindungsressourcen bereitstellt, dann quittiert die CPU die Anweisung zum Aufbau der Verbindung mit einem Fehler.

Vergleich S7-1500 und S7-300

Einen Vergleich, wie die Kommunikationsressourcen der S7-1500 und S7-300 verwaltet werden, finden Sie in diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/109747092).

14.3 Anzeige der Verbindungsressourcen

Anzeige der Verbindungsressourcen in STEP 7 (Offline-Sicht)

Sie können sich die Verbindungsressourcen eines Automatisierungssystems in der Hardware-Konfiguration anzeigen lassen. Sie finden die Verbindungsressourcen im Inspektorfenster in den Eigenschaften der CPU.

Verbindungsressourcen

① ②

	Ressourcen der Station			Ressourcen des..	Ressourcen des..	Ressourcen des..
	Reserviert		Dynamisch ⚠	PLC_1 [CPU 15...	CM 1542-1_1 [..	CP 1543-1_1 [...
Maximale Anzahl der Ressourcen:		10	246	128	64	118
	Maximum	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte	Konfigurierte
PG-Kommunikation:	4	-	-	-	-	-
HMI-Kommunikation:	4	4	6	6	0	4
S7-Kommunikation:	0	-	7	4	3	0
Open User Communication:	0	-	13	8	5	0
Web-Kommunikation:	2	-	-	-	-	-
OPC UA-Client/Server-Kommunikati...	0	-	-	-	-	-
Sonstige Kommunikation:	-	-	0	0	0	0
Insgesamt verwendete Ressourcen:		4	26	18	8	4
Verfügbare Ressourcen:		6	220	110	56	114

Bild 14-4 Beispiel: Reservierte und verfügbare Verbindungsressourcen (Offline-Sicht)

① Stationsspezifische Verbindungsressourcen

Die Spalten der stationsspezifischen Verbindungsressourcen geben Informationen über die verwendeten und verfügbaren Verbindungsressourcen der Station.

Im Beispiel stehen für das Automatisierungssystem maximal 256 stationsspezifische Verbindungsressourcen zur Verfügung:

- 10 reservierte Verbindungsressourcen, davon sind 4 bereits verwendet und 6 noch verfügbar.
  Die verwendeten Ressourcen teilen sich wie folgt auf:
  – 4 Ressourcen für HMI-Kommunikation
- 246 dynamische Verbindungsressourcen, davon sind 26 bereits verwendet und 220 noch verfügbar.
  Die verwendeten Ressourcen teilen sich wie folgt auf:
  – 6 Ressourcen für HMI-Kommunikation
  – 7 Ressourcen für S7-Kommunikation
  – 13 Ressourcen für Open User Communication

Das Warndreieck in der Spalte der dynamischen Stationsressourcen wird deshalb angezeigt, weil die Summe der max. verfügbaren Verbindungsressourcen von CPU, CP und CM (= 310 Verbindungsressourcen) die Stationsgrenze von 256 überschreitet.

HINWEIS

Überschreiten der verfügbaren Verbindungsressourcen

Ein Überschreiten der stationsspezifischen Verbindungsressourcen meldet STEP 7 mit einer Warnung. Wenn Sie die verfügbaren Verbindungsressourcen aus CPU, CP und CM voll ausnutzen wollen, dann müssen Sie entweder eine CPU mit einer größeren max. Anzahl verfügbarer stationsspezifischer Verbindungsressourcen einsetzen oder die Anzahl der Kommunikationsverbindungen reduzieren.

② Modulspezifische Verbindungsressourcen

Die Spalten der modulspezifischen Verbindungsressourcen geben Informationen über die Ressourcenverwendung in den CPUs, CPs und CMs eines Automatisierungssystems:

Die Anzeige ist modul- und nicht schnittstellengranular.

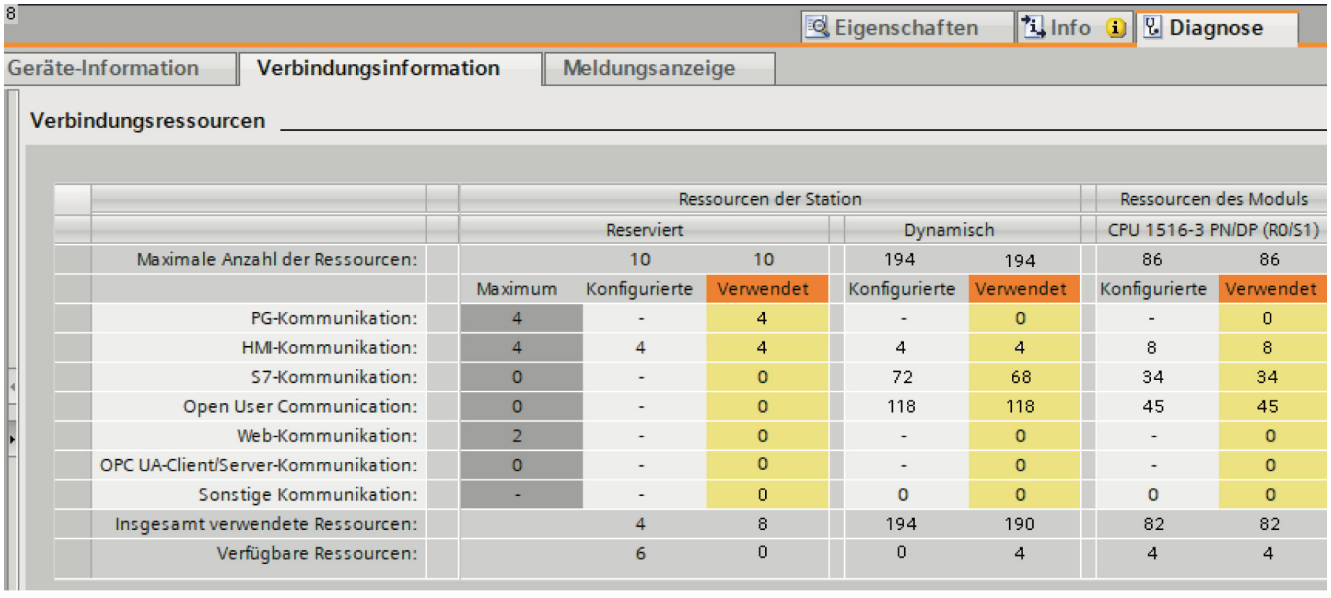
Im Beispiel stellt die CPU maximal 128 Verbindungsressourcen zur Verfügung, davon sind 18 bereits verwendet und 110 noch verfügbar:

Die verwendeten Ressourcen teilen sich wie folgt auf:

- 6 Ressourcen für HMI-Kommunikation
- 4 Ressourcen für S7-Kommunikation
- 8 Ressourcen für Open User Communication

Anzeige der Verbindungsressourcen in STEP 7 (Online-Sicht)

Wenn Sie online mit der CPU verbunden sind, dann können Sie sich unter "Verbindungsinformation" zusätzlich anzeigen lassen, wie viele Ressourcen jeweils aktuell verwendet werden.



	Ressourcen der Station					Ressourcen des Moduls	
	Reserviert			Dynamisch		CPU 1516-3 PN/DP (R0/S1)	
Maximale Anzahl der Ressourcen:		10	10	194	194	86	86
	Maximum	Konfigurierte	Verwendet	Konfigurierte	Verwendet	Konfigurierte	Verwendet
PG-Kommunikation:	4	-	4	-	0	-	0
HMI-Kommunikation:	4	4	4	4	4	8	8
S7-Kommunikation:	0	-	0	72	68	34	34
Open User Communication:	0	-	0	118	118	45	45
Web-Kommunikation:	2	-	0	-	0	-	0
OPC UA-Client/Server-Kommunikation:	0	-	0	-	0	-	0
Sonstige Kommunikation:	-	-	0	0	0	0	0
Insgesamt verwendete Ressourcen:		4	8	194	190	82	82
Verfügbare Ressourcen:		6	0	0	4	4	4

Bild 14-5 Verbindungsressourcen - online

Die Online-Sicht der Tabelle "Verbindungsressourcen" enthält zusätzlich zur Offline-Sicht Spalten mit den aktuell verwendeten Verbindungsressourcen. In der Online-Sicht werden **alle** verwendeten Verbindungsressourcen im Automatisierungssystem angezeigt, unabhängig davon, auf welche Art die Verbindung eingerichtet wurde.
In der Zeile "Sonstige Kommunikation" werden belegte Verbindungsressourcen für Kommunikation zu Fremdgeräten angezeigt. Die Tabelle wird automatisch aktualisiert.

HINWEIS

Wenn eine geroutete S7-Verbindung über eine CPU geht, dann werden die dafür benötigten Verbindungsressourcen der CPU nicht in der Tabelle der Verbindungsressourcen angezeigt.

Anzeige der Verbindungsressourcen für HMI

Informationen über die Verfügbarkeit und Belegung von Verbindungsressourcen für HMI-Verbindungen finden Sie in der Offline-Sicht im Kontext des HMI-Geräts (im Inspektorfenster, in den Eigenschaften im Bereich "Verbindungsressourcen").

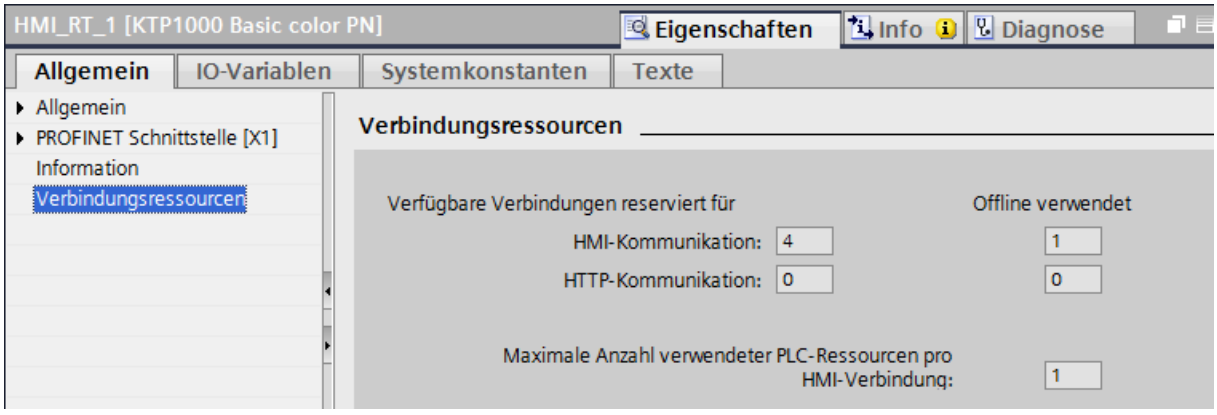


Bild 14-6 Verbindungsressourcen - HMI-Kommunikation

Im Bereich Verbindungsressourcen werden angezeigt:

- Anzahl der im HMI verfügbaren Verbindungen reserviert für HMI-Kommunikation und HTTP-Kommunikation
- Anzahl der offline im HMI verwendeten Verbindungsressourcen für HMI-Kommunikation und HTTP-Kommunikation
  Falls die Anzahl der maximal verfügbaren Verbindungsressourcen für ein HMI-Gerät überschritten ist, wird eine entsprechende Meldung von STEP 7 ausgegeben.
- "Maximale Anzahl verwendeter PLC-Ressourcen pro HMI-Verbindung": Dieser Parameter ist ein Faktor, der mit der Anzahl der offline verwendeten HMI-Verbindungen zu multiplizieren ist. Das Produkt ergibt die Anzahl der in der CPU belegten HMI-Ressourcen.

Anzeige der Verbindungsressourcen im Webserver

Sie können sich die Verbindungsressourcen nicht nur in STEP 7 anzeigen lassen, sondern auch mit einem Browser, der die entsprechende Seite des Webservers anzeigt.
Informationen zur Anzeige der Verbindungsressourcen im Webserver finden Sie im Funktionshandbuch Webserver (https://support.industry.siemens.com/cs/ww/de/view/59193560).

15

Diagnose und Störungsbeseitigung

15.1 Diagnose von Verbindungen

Verbindungstabelle in der Online-Sicht

Für eine im Hardware- und Netzwerkeditor von STEP 7 angewählte CPU erhalten Sie in der Online-Sicht der Verbindungstabelle den Status ihrer Verbindungen angezeigt.

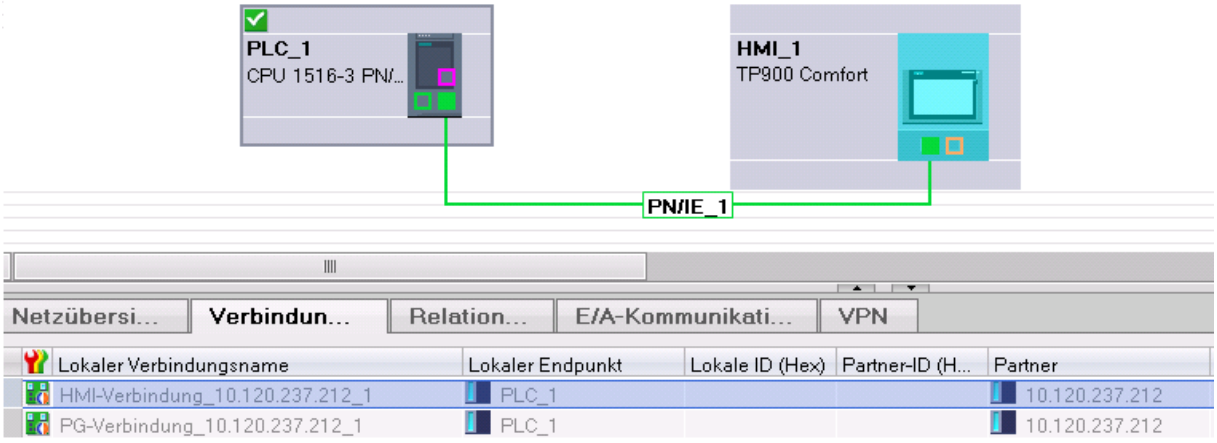


Bild 15-1 Online-Sicht der Verbindungstabelle

Für die angewählte Verbindung in der Verbindungstabelle erhalten Sie detaillierte Diagnoseinformationen im Register "Verbindungsinformationen".

Register "Verbindungsinformationen": Verbindungsdetails

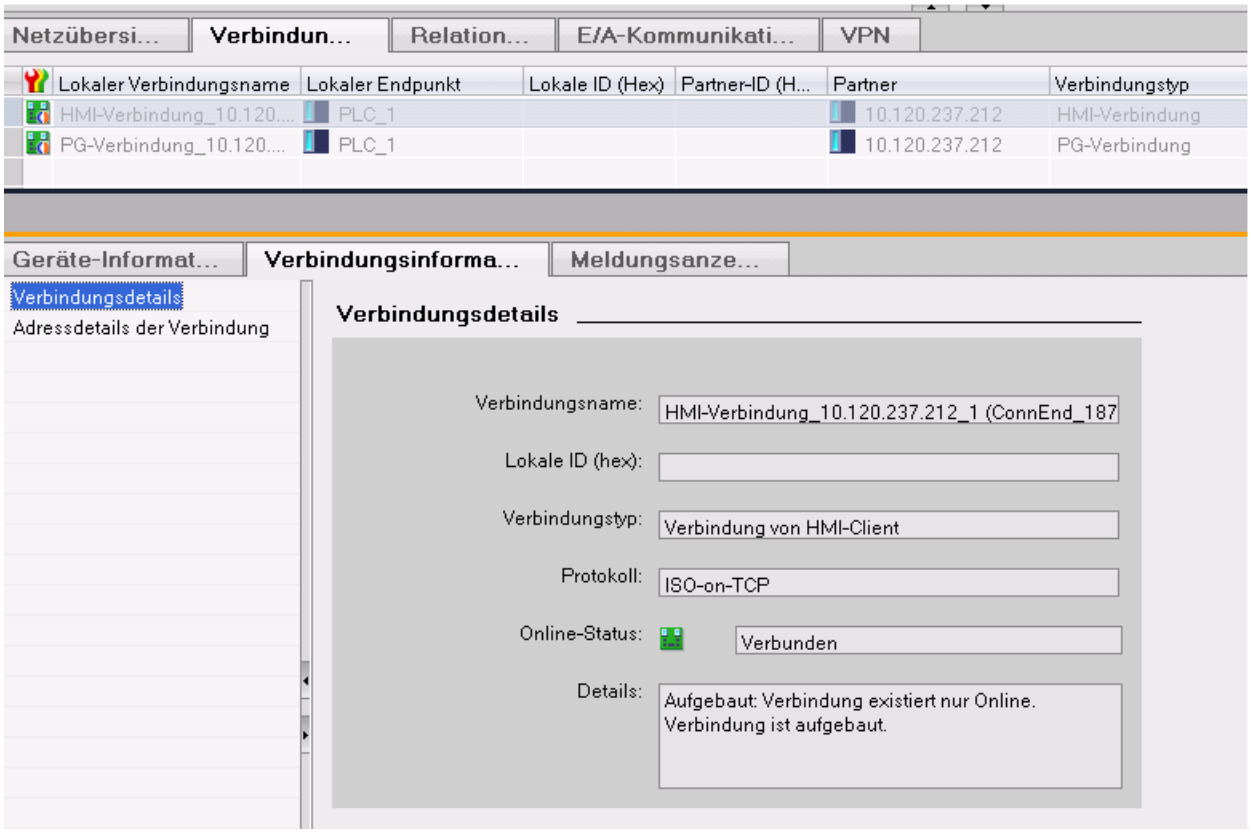


Bild 15-2 Diagnose von Verbindungen - Verbindungsdetails

Register "Verbindungsinformationen": Adressdetails

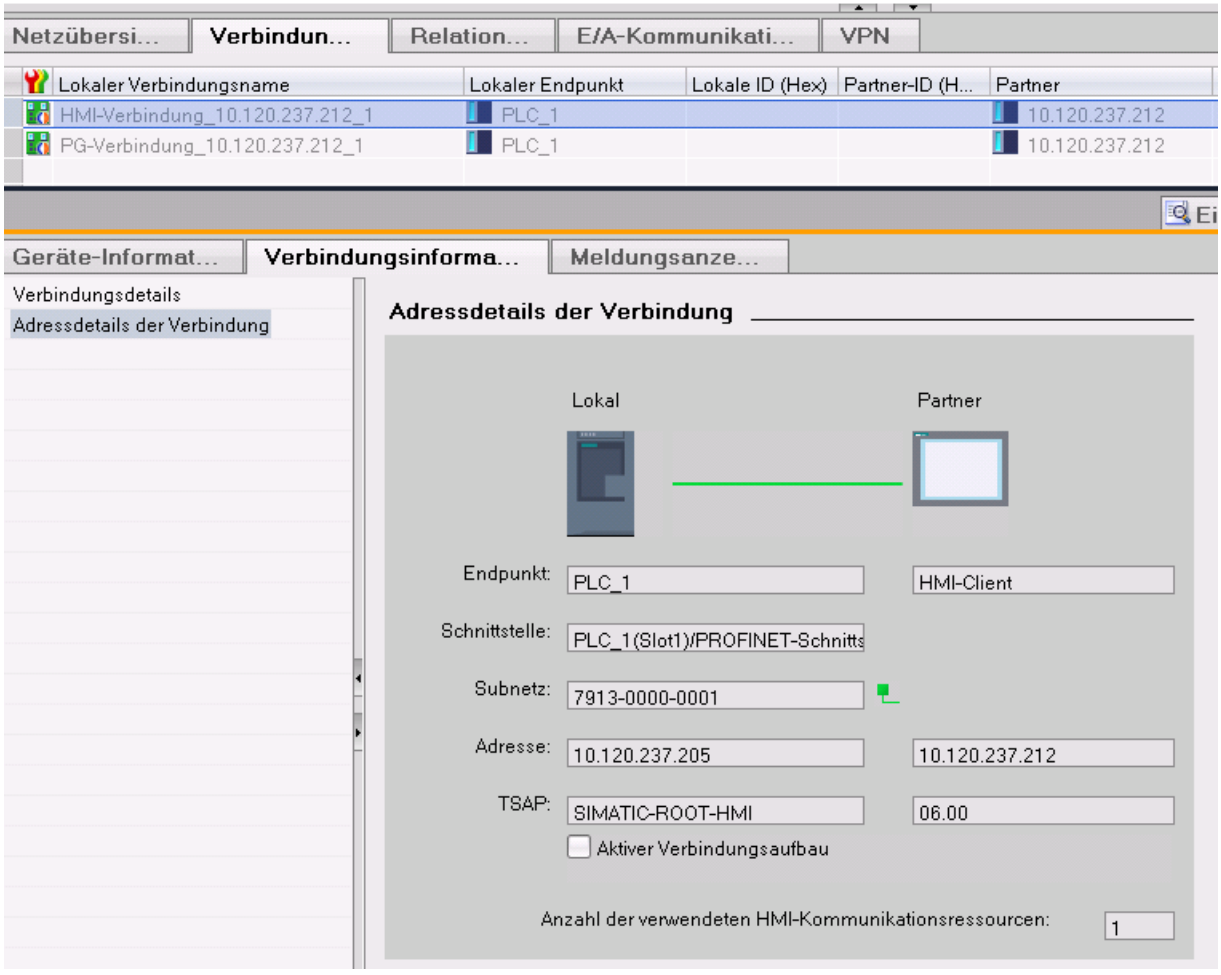


Bild 15-3 Diagnose von Verbindungen - Adressdetails

Diagnose über Webserver

Über den integrierten Webserver einer CPU haben Sie die Möglichkeit, Diagnoseinformationen von der CPU über einen Webbrowser auszuwerten.

Auf der Webseite "Kommunikation" finden Sie in verschiedenen Registern folgende Informationen zur Kommunikation über PROFINET:

- Informationen zu den PROFINET-Schnittstellen der CPU (z. B. Adressen, Subnetze, physikalische Eigenschaften)
- Informationen zur Qualität der Datenübertragung (z. B. Anzahl fehlerfrei gesendeter/empfangener Datenpakete)
- Informationen zur Belegung/Verfügbarkeit von Verbindungsressourcen
- Die Seite "Verbindungsstatus" ist ähnlich der Online-Sicht in STEP 7 und gibt auch eine Übersicht aller Verbindungen mit Detailansicht

Diagnose über Anwenderprogramm

Wenn Sie die Anweisung T_DIAG programmieren, können Sie über das Anwenderprogramm Diagnoseinformationen über die projektierten und programmierten Verbindungen von der CPU auswerten.

Weitere Informationen

Die Beschreibung der Webserverfunktionalität finden Sie im Funktionshandbuch Webserver (https://support.industry.siemens.com/cs/ww/de/view/59193560).

15.2 Notfalladresse

Wenn Sie die CPU nicht über die IP-Adresse erreichen, können Sie eine temporäre Notfalladresse (Emergency IP) für die CPU einstellen. Über die Notfalladresse können Sie die Verbindung mit der CPU wiederherstellen, um eine Gerätekonfiguration mit einer gültigen IP-Adresse zu laden.

Eine Notfalladresse können Sie unabhängig von der Schutzstufe der CPU einstellen

Wann benötigen Sie eine Notfalladresse?

In den folgenden Fällen ist Ihre CPU nicht erreichbar:

- Die IP-Adresse der PROFINET-Schnittstelle ist doppelt vorhanden.
- Die Subnetzmaske ist falsch eingestellt.

Voraussetzungen

- Sie haben in der Gerätekonfiguration in STEP 7 "IP-Adresse im Projekt einstellen" für das IP-Protokoll ausgewählt.
- Keine Projektierung mit aktiviertem DCP-Schreibschutz geladen.
- Die CPU befindet sich im Betriebszustand STOP.

Gültige Gerätekonfiguration mit einer Notfalladresse wiederherstellen

1. Stellen Sie die Notfalladresse für die Schnittstelle der CPU mit einem DCP-Tool ein. Zum Beispiel verfügt das SIMATIC Automation Tool über einen DCP-Befehl "IP-Adresse festlegen".
   Die Wartungs-LED der CPU leuchtet. Auch der Diagnosepuffer zeigt an, dass eine Notfalladresse für eine Ethernet-Schnittstelle aktiviert wurde.
2. Laden Sie ein STEP 7-Projekt mit einer gültigen IP-Adresse in die CPU.
3. Schalten Sie die CPU aus und wieder ein.
   Die Notfalladresse ist zurückgesetzt.

Ergebnis

Die CPU läuft mit der gültigen IP-Adresse an.

16

Kommunikation mit dem redundanten System S7-1500R/H

Einführung

Die Kommunikation mit dem redundanten System S7-1500R/H funktioniert grundsätzlich wie beim Standardsystem S7-1500.
Dieses Kapitel beschreibt die Besonderheiten und Einschränkungen für die Kommunikation mit dem redundanten System S7-1500R/H.

Kommunikationsmöglichkeiten für das redundante System S7-1500R/H

- Open User Communication über TCP/IP, UDP, ISO-on-TCP und Modbus/TCP
- Secure Open User Communication mit Ausnahme der Funktion "Secure OUC über E-Mail" (siehe auch Secure Open User Communication (Seite 86))
- S7-Kommunikation als Server
- HMI-Kommunikation, PG-Kommunikation
- Datenaustausch über OPC UA als Server
- Secure PG/HMI-Kommunikation (siehe auch Secure PG/HMI-Kommunikation (Seite 103))
- SNMP
- Uhrzeitsynchronisation über NTP
- Webserver (nur über Web API)
- Unterstützung des Kommunikationsprozessors CP 1543-1 als zentral gestecktes Modul (siehe auch Systemhandbuch S7-1500R/H (https://support.industry.siemens.com/cs/de/de/view/109754833))

Einschränkungen für die Kommunikation mit dem redundanten System S7-1500R/H

- Open User Communication:
  - keine projektierten Verbindungen
  - E-Mail: Die S7-1500R/H CPUs unterstützen nur die Versionen < V5.0 der Anweisung "TMAIL_C". Die Versionen ab V5.0 werden nicht unterstützt.
  - keine Unterstützung von Verbindungsbeschreibungen nach "TCON_Param"
- keine S7-Kommunikation als Client
- PG-Kommunikation: Es ist nicht möglich, online auf beide CPUs gleichzeitig zuzugreifen. Sie können entweder auf die Primary-CPU oder auf die Backup-CPU zugreifen.

16.1 System IP-Adressen bei R/H-CPUs

Einleitung

Zusätzlich zu den Geräte IP-Adressen der CPUs unterstützt das redundante System S7-1500R/H System IP-Adressen:

- System IP-Adresse für die PROFINET-Schnittstellen X1 der beiden CPUs (System IP-Adresse X1)
- System IP-Adresse für die PROFINET-Schnittstellen X2 der beiden CPUs (System IP-Adresse X2)
- System IP-Adresse für die PROFINET-Schnittstellen X3 der beiden CPUs (System IP-Adresse X3)

Die System IP-Adressen verwenden Sie für die Kommunikation mit anderen Geräten (z. B. HMI-Geräte, CPUs, PCs). Die Geräte kommunizieren über die System IP-Adresse immer mit der Primary-CPU des redundanten Systems. Dadurch wird z. B. sichergestellt, dass der Kommunikationspartner nach einem Ausfall der Primary-CPU im redundanten Betrieb mit der neuen Primary-CPU (vorher Backup-CPU) im Systemzustand RUN-Solo kommunizieren kann.

Zu jeder System IP-Adresse gehört eine virtuelle MAC-Adresse.

Die System IP-Adressen aktivieren Sie in STEP 7.

Vorteile der System IP-Adressen gegenüber den Geräte IP-Adressen

- Der Kommunikationspartner kommuniziert gezielt mit der Primary-CPU.
- Die Kommunikation des redundanten Systems S7-1500R/H über eine System IP-Adresse funktioniert auch bei Ausfall der Primary-CPU weiterhin.

Anwendungsfälle

Die System IP-Adressen nutzen Sie für folgende Anwendungen:

- HMI-Kommunikation mit dem redundanten System S7-1500R/H: Mit einem HMI-Gerät steuern bzw. beobachten Sie den Prozess auf dem redundanten System S7-1500R/H.
- Open User Communication mit dem redundanten System S7-1500R/H:
  - Eine andere CPU oder eine Applikation auf einem PC greift auf Daten des redundanten Systems S7-1500R/H zu.
  - Das redundante System S7-1500R/H greift auf ein anderes Gerät zu.

  Möglich sind TCP, UDP und ISO-on-TCP-Verbindungen.
- IP-Forwarding: Wenn Sie für IP-Routen durch das redundante System S7-1500R/H die System IP-Adressen als Gateway/Standard-Route verwenden, dann werden IP-Pakete auch bei Ausfall einer CPU weitergeleitet.

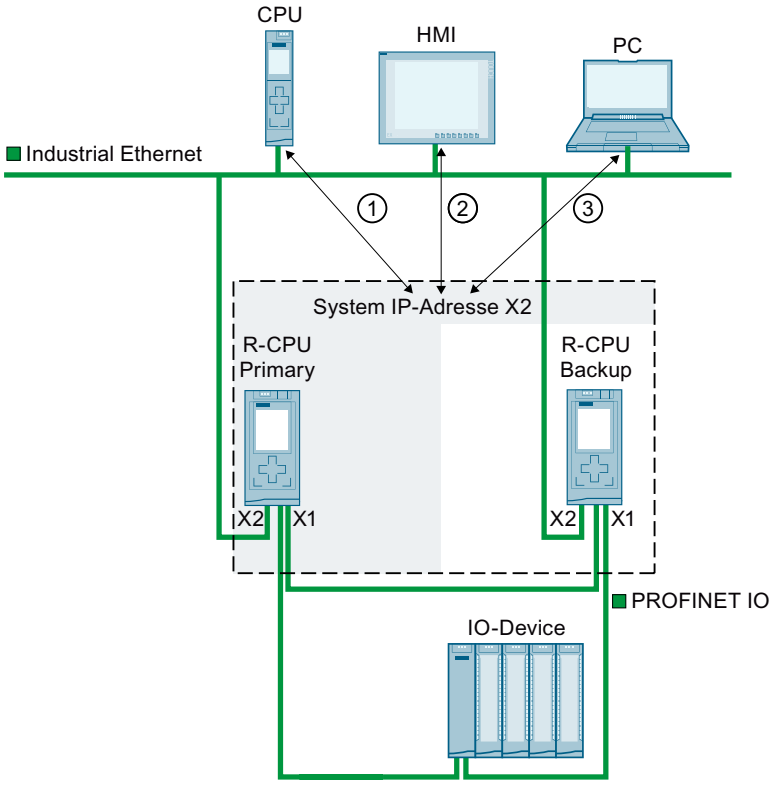
Voraussetzungen

- Die Schnittstelle des Kommunikationspartners ist mit beiden CPUs verbunden, jeweils über die gleiche Schnittstelle (z. B. X2).
- Die System IP-Adresse für die Schnittstellen des S7-1500R/H-Systems ist aktiviert.

Kommunikation über die System IP-Adressen X2 und X3

Wenn die CPUs des redundanten Systems S7-1500R/H zwei oder drei PROFINET-Schnittstellen haben, verwenden Sie für Kommunikation mit anderen Geräten bevorzugt die PROFINET-Schnittstelle X2 bzw. X3.

Das folgende Bild zeigt eine Konfiguration, bei der die Kommunikationspartner über die jeweiligen PROFINET-Schnittstellen X2 mit den CPUs des redundanten Systems S7-1500R/H verbunden sind.



① Open User Communication zwischen einer anderen CPU und dem redundanten System S7-1500R

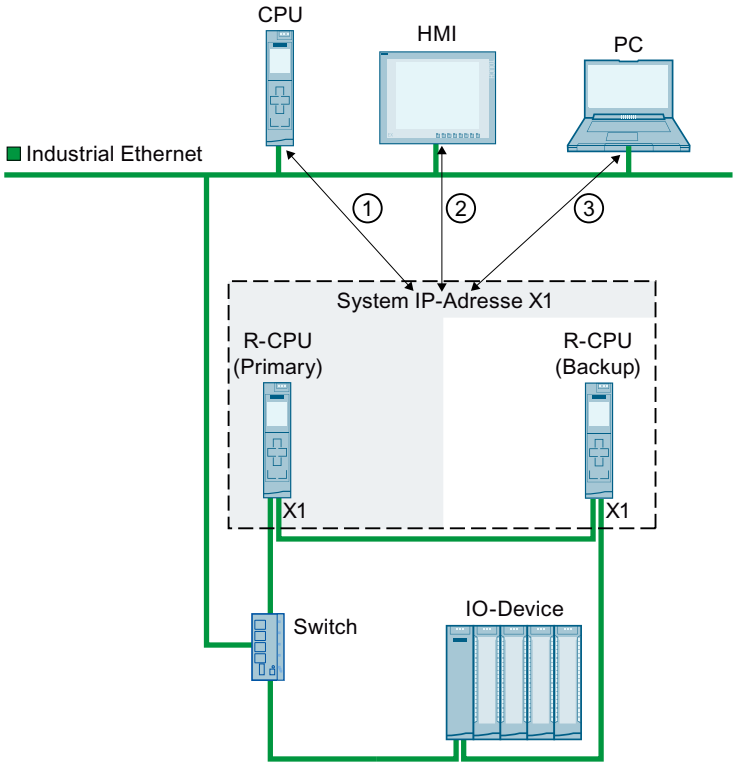
② HMI-Kommunikation mit dem redundanten System S7-1500R

③ Open User Communication zwischen dem redundanten System S7-1500R und einem PC

Bild 16-1 Beispiel: Kommunikation des redundanten Systems S7-1515R über die System IP-Adresse X2

Kommunikation über die System IP-Adresse X1

Das folgende Bild zeigt eine Konfiguration, bei der die Kommunikationspartner mit einem Switch an den PROFINET-Ring des redundanten Systems S7-1500R/H angeschlossen sind. Der PROFINET-Ring verbindet die Kommunikationspartner mit den jeweiligen PROFINET-Schnittstellen X1 der beiden CPUs.
Da die CPU 1513R-1 PN nur eine PROFINET-Schnittstelle hat, ist der Anschluss über den PROFINET-Ring die einzige Möglichkeit, um über die System IP-Adresse X1 zu kommunizieren.



① Open User Communication zwischen dem redundanten System S7-1500R und einer anderen CPU

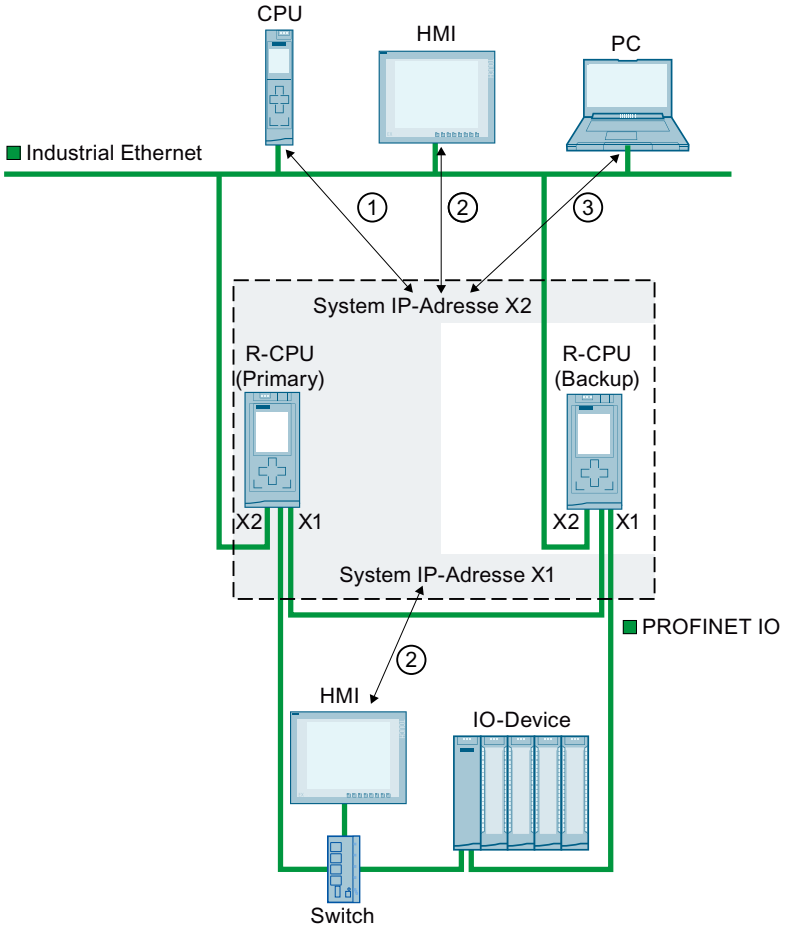
② HMI-Kommunikation mit dem redundanten System S7-1500R

③ Open User Communication zwischen dem redundanten System S7-1500R und einem PC

Bild 16-2 Beispiel: Kommunikation des redundanten Systems S7-1513R über die System IP-Adresse X1

Kommunikation über die System IP-Adressen X1, X2 und X3

Sie können für jede PROFINET-Schnittstelle des redundanten Systems S7-1500R/H eine System IP-Adresse verwenden. PROFINET-Geräte, die mit den Schnittstellen X1 der CPUs verbunden sind, kommunizieren über die System IP-Adresse X1. PROFINET-Geräte, die mit den Schnittstellen X2 der CPUs verbunden sind, kommunizieren über die System IP-Adresse X2. PROFINET-Geräte, die mit den Schnittstellen X3 der CPUs verbunden sind, kommunizieren über die System IP-Adresse X3.



① Open User Communication zwischen dem redundanten System S7-1500R und einer anderen CPU.

② HMI-Kommunikation mit dem redundanten System S7-1500R

③ Open User Communication zwischen dem redundanten System S7-1500R und einem PC

Bild 16-3 Beispiel: Kommunikation des redundanten Systems S7-1515R über die System IP-Adressen X1 und X2

Bei einem S7-1500H(F)-System haben Sie zusätzlich die Möglichkeit, Ihre Anlage in mehrere PROFINET-Ringe zu unterteilen.

Benötigte S1-/S2-Devices müssen Sie in diesem Fall in einem eigenen PROFINET-Ring hinter einem Y-Switch anschließen.

Empfehlung: Für eine erhöhte Verfügbarkeit der S1-/S2-Devices benötigen Sie 2 Y-Switche mit DNA-Redundanz (SCALANCE XF204-2BA DNA). Ein Y-Switch übernimmt die Rollen MRP-Manager und DNA-Manager. Ein weiterer Y-Switch die Rollen MRP-Client und DNA-Client. DNA-Redundanz ist nur mit einem angebundenen PROFINET-Ring möglich.

Weitere Informationen zu den Aufbauszenarien mit Y-Switch finden Sie im Systemhandbuch Redundantes System S7-1500R/H
(https://support.industry.siemens.com/cs/ww/de/view/109754833).

IP-Forwarding über die System IP-Adressen

Wenn Sie für IP-Routen durch das redundante System S7-1500R/H die System IP-Adressen als Gateway/Standard-Route verwenden, dann werden IP-Pakete auch bei Ausfall einer CPU weitergeleitet.

Im folgenden Bild ist der PC mit den beiden Schnittstellen X2 der S7-1500R CPUs verbunden. Für die Route zum HMI-Gerät ist im PC als Gateway die System IP-Adresse X2 eingetragen. Das HMI-Gerät ist über einen Switch an den PROFINET-Ring des redundanten Systems S7-1500 angeschlossen. Im HMI-Gerät ist als Router die System IP-Adresse X1 projektiert.

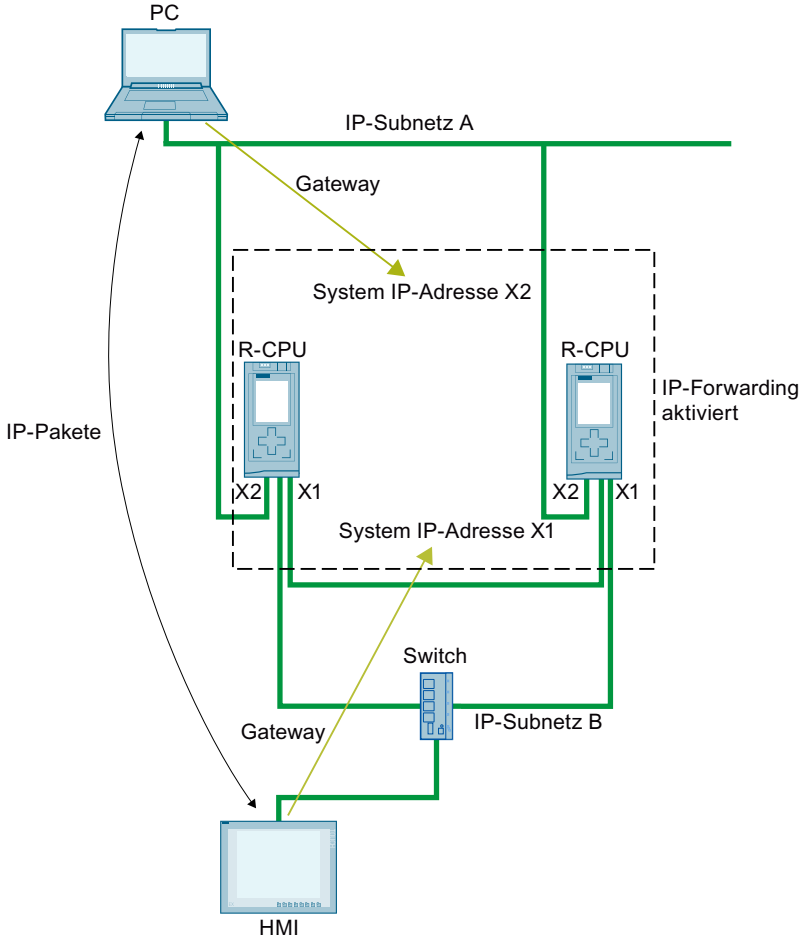


Bild 16-4 Beispiel: IP-Forwarding über die System IP-Adressen

System IP-Adressen aktivieren

Voraussetzungen:

- STEP 7 ab V15.1
- redundantes System S7-1500R/H mit zwei CPUs, z. B. zwei CPUs 1513R-1 PN

Wenn die CPUs des redundanten Systems S7-1500R/H zwei PROFINET-Schnittstellen (X1 und X2) besitzen, dann können Sie für beide PROFINET-Schnittstellen eine System IP-Adresse aktivieren. Im Folgenden ist beschrieben, wie Sie die System IP-Adresse für die Schnittstelle X1 aktivieren.

Um die System IP-Adresse für ihr redundantes System S7-1500R/H zu aktivieren, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 die Schnittstelle X1 einer der beiden CPUs.
2. Navigieren Sie im Inspektorfenster zu "Eigenschaften > Allgemein > Ethernet-Adressen" in den Bereich "System IP-Adresse für geschaltete Kommunikation".
3. Aktivieren Sie das Optionskästchen "Aktivieren Sie die System IP-Adresse für geschaltete Kommunikation".
   STEP 7 legt automatisch eine System IP-Adresse an.

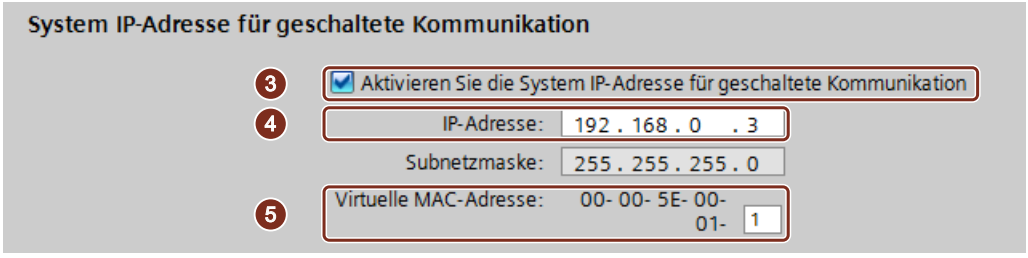


Bild 16-5 System IP-Adresse projektieren

4. Passen Sie die System IP-Adresse bei Bedarf an.
5. Passen Sie bei Bedarf die virtuelle MAC-Adresse an. Vergeben Sie dazu bei "Virtuelle MAC-Adresse" für das letzte Byte einen projektweit eindeutigen Wert (Wertebereich 01_H bis FF_H).

HINWEIS

Eindeutigkeit der virtuellen MAC-Adresse

Das redundante System S7-1500R/H verwendet für jede System IP-Adresse eine MAC-Adresse aus dem Adressbereich 00-00-5E-00-01-00 bis 00-00-5E-00-01-00. Dieser Adressbereich wird auch für VRRP (Virtual Redundancy Protocol) verwendet.

Wenn Sie Geräte mit VRRP einsetzen, z. B. Switche, achten Sie auf die Eindeutigkeit der MAC-Adressen innerhalb einer Ethernet-Broadcast-Domain.

Ergebnis: Die System IP-Adresse X1 für die PROFINET-Schnittstellen X1 der beiden CPUs ist aktiviert.

16.2 System IP-Adressen bei Kommunikationsprozessoren

Einleitung

Ab STEP 7 V19 haben Sie die Möglichkeit ein redundantes System S7-1500R/H ab FW-Version V3.1 mit den Kommunikationsprozessoren CP 1543-1 ab FW-Version V3.0 zu erweitern. R/H-CPUs unterstützen bei der Erweiterung durch die Kommunikationsprozessoren CP 1543-1 die Projektierung einer virtuellen Schnittstelle W1 mit Geräte- und System IP-Adresse. Die an den CPs angeschlossenen Kommunikationspartner kommunizieren über diese IP-Adressen mit den R/H-CPUs.

Die System IP-Adresse von W1 aktivieren Sie in STEP 7.

Zu jeder System IP-Adresse gehört eine virtuelle MAC-Adresse.

Ausfall eines Kommunikationsprozessors

Beim Ausfall eines Kommunikationsprozessors CPs 1543-1 verhält sich das redundante System S7-1500R/H wie folgt:

- S7-1500R:
  Beim Ausfall eines CPs wechselt die dazugehörige CPU in den Betriebszustand STOP. Das S7-1500R-System wechselt in den Systemzustand RUN-Solo.
  Wenn der CP an der Primary-CPU ausfällt, führt das redundante System S7-1500R eine Primary-Backup-Umschaltung durch. Die zugewiesene System IP-Adresse wechselt auf die neue Primary-CPU. Der Aufbau der neuen Kommunikationsverbindung findet dann über den redundanten CP statt.
  Wenn der CP an der Backup-CPU ausfällt, hat dieser Ausfall keine Auswirkung auf die bestehende Kommunikation über die System IP-Adresse.
- S7-1500H mit Aktivem Rückwandbus:
  Beim Ausfall eines CPs bleibt das S7-1500H-System im Systemzustand RUN-Redundant.
  Wenn der CP an der Primary-CPU ausfällt, führt das redundante System S7-1500H keine Primary-Backup-Umschaltung durch. In diesem Fall ist der Aufbau einer neuen Kommunikationsverbindung über den redundanten CP nur über die zugewiesene Geräte IP-Adresse möglich.
  Alternativ können Sie eine Primary-Backup-Umschaltung im Anwenderprogramm einrichten. Rufen Sie dazu die Anweisung "RH_CTRL" mit Mode=8 im Anwenderprogramm auf. Dadurch wird der Betriebszustand STOP der Primary-CPU angefordert. Die Backup-CPU wird zur Primary-CPU und wechselt in den Betriebszustand RUN. Auch die System IP-Adresse W1 wechselt auf die neue Primary-CPU. Der Aufbau der neuen Kommunikationsverbindung findet dann automatisch über den redundanten CP statt.

Weitere Informationen zur Anweisung "RH_CTRL" finden Sie in der Onlinehilfe zu STEP 7.

Informationen zur Projektierung der System IP-Adresse für die virtuelle Schnittstelle W1 und der virtuellen MAC-Adresse finden Sie im Systemhandbuch Redundantes System S7-1500R/H (https://support.industry.siemens.com/cs/de/de/view/109754833).

Vorteile der Erweiterung mit Kommunikationsprozessoren CP 1543-1

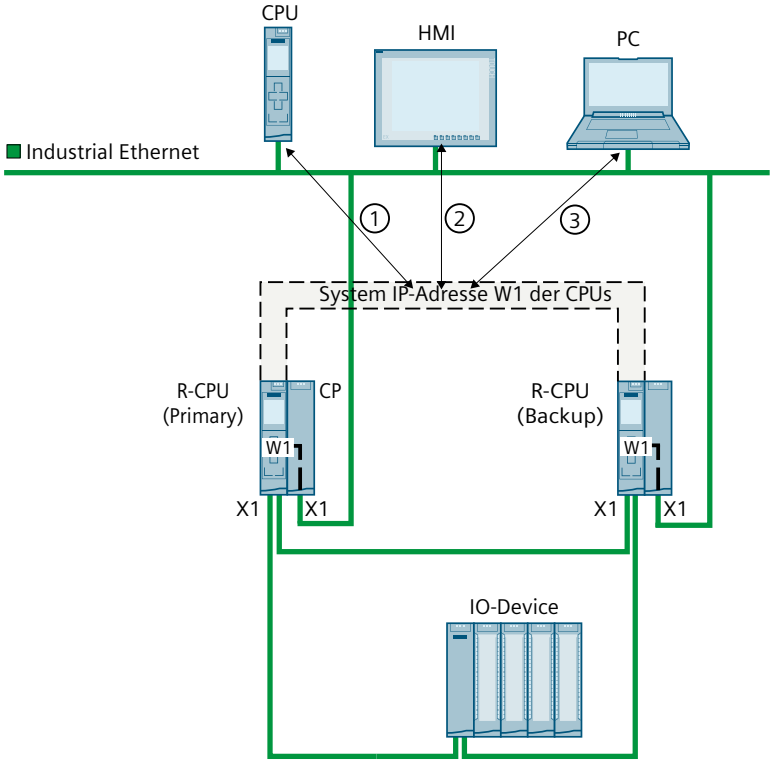
Wenn Sie Ihr redundantes System mit den Kommunikationsprozessoren CP 1543-1 erweitern, stehen Ihnen zusätzliche Kommunikationsschnittstellen zur Verfügung. Nutzen Sie diesen Vorteil bei einer CPU 1513R-1 PN mit nur einer PROFINET-Schnittstelle, z. B. für eine Netztrennung.

Voraussetzungen zum Verwenden der System IP-Adresse von W1

- Der virtuellen Schnittstelle W1 ist ein CP 1543-1 zugewiesen.
- Die System IP-Adresse der virtuellen Schnittstelle W1 ist aktiviert.
- Der virtuellen Schnittstelle W1 ist eine System IP-Adresse zugewiesen.
- Ein Kommunikationspartner erreicht beide CPs über die zugewiesene System IP-Adresse.

Kommunikation über die System IP-Adresse von W1

Das folgende Bild zeigt die Konfiguration eines redundanten Systems S7-1513R mit der Erweiterung durch CPs. Die Kommunikationspartner kommunizieren über die zugewiesene System IP-Adresse von W1 mit dem redundanten System.



① Kommunikation zwischen dem redundanten System S7-1500R und einer anderen CPU

② HMI-Kommunikation mit dem redundanten System S7-1500R

③ Kommunikation zwischen dem redundanten System S7-1500R und einem PC

Bild 16-6 Beispiel: Kommunikation des redundanten Systems S7-1513R über die zugewiesene System IP-Adresse von W1

IP-Forwarding über die System IP-Adressen

Wenn Sie für IP-Routen durch das redundante System S7-1500R die System IP-Adressen als Gateway/Standard-Route verwenden, dann werden IP-Pakete auch bei Ausfall eines CPs weitergeleitet.

HINWEIS

Primary-Backup-Umschaltung bei S7-1500H-Systemen mit Aktivem Rückwandbus

Beim Ausfall eines CPs in einem S7-1500H-System mit Aktivem Rückwandbus findet keine Primary-Backup-Umschaltung statt. Richten Sie in diesem Fall bei Bedarf die Primary-Backup-Umschaltung in Ihrem Anwenderprogramm ein.

Im folgenden Bild ist der PC mit den beiden Schnittstellen X1 der CPs 1543-1 verbunden. Für die Route zum HMI-Gerät ist im PC als Router die zugewiesene System IP-Adresse von W1 eingetragen. Das HMI-Gerät ist über einen Switch an den PROFINET-Ring des redundanten Systems S7-1500R angeschlossen. Im HMI-Gerät ist als Router die System IP-Adresse X1 der CPUs eingetragen.

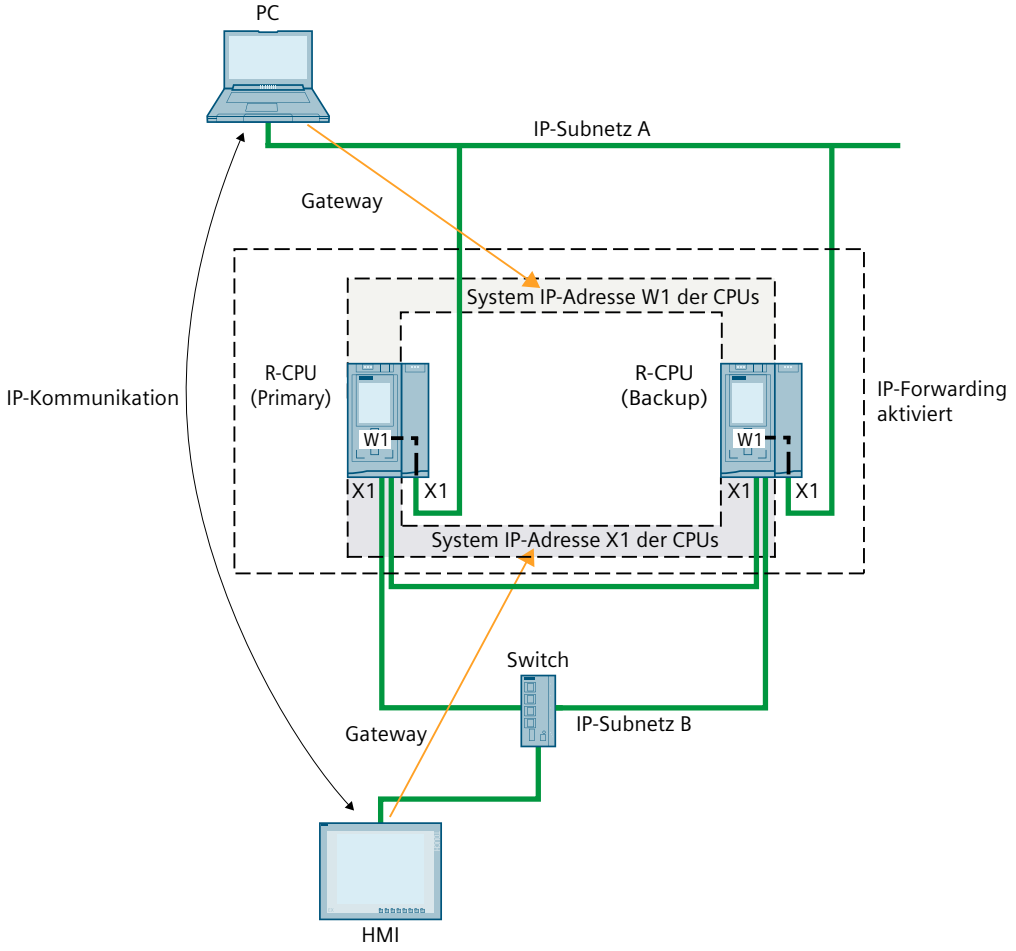


Bild 16-7 Beispiel: IP-Forwarding über die System IP-Adressen von CPUs und CPs

System IP-Adresse der virtuellen Schnittstelle W1 von R/H-CPUs zuweisen

Ab STEP 7 V19 können Sie der virtuellen Schnittstelle W1 von R/H-CPUs eine System IP-Adresse zuweisen. Die Kommunikationspartner, die Sie an den CPs angeschlossen haben, kommunizieren mit dem R/H-System über die zugewiesene System IP-Adresse.

Um die System IP-Adresse der virtuellen Schnittstelle W1 der R/H-CPUs zu ändern, gehen Sie folgendermaßen vor:

1. Wählen Sie in der Gerätesicht die R/H-CPU auf der Profilschiene_0 aus.
2. Navigieren Sie im Inspektorfenster zu "Erweiterte Konfiguration > Zugriff auf PLC über Kommunikationsmodul".
3. Wählen Sie im Abschnitt "Schnittstelle [W1] zuweisen" in der Auswahlliste "Kommunikationsmodul auswählen" den erforderlichen CP.
   Nach der Auswahl erscheint eine Warnmeldung, die Sie auf mögliche Risiken dieser Einstellung hinweist. Wenn Sie die Warnmeldung bestätigen, werden die Konfigurationsmöglichkeiten für die virtuelle Schnittstelle W1 angezeigt.
4. Navigieren Sie zum Bereich "Internet Protocol Version 4 (IPv4)".
5. Übernehmen Sie den Vorschlag oder vergeben Sie eine neue IP-Adresse.
6. Navigieren Sie zum Bereich "System IP-Adresse für geschaltete Kommunikation".
7. Aktivieren Sie die Option "Aktivieren Sie die System IP-Adresse für geschaltete Kommunikation".
8. Übernehmen Sie den Vorschlag oder vergeben Sie eine neue System IP-Adresse (und virtuelle MAC-Adresse).

9. Die R/H-CPU auf der Profilschiene_1 übernimmt die Einstellungen automatisch.

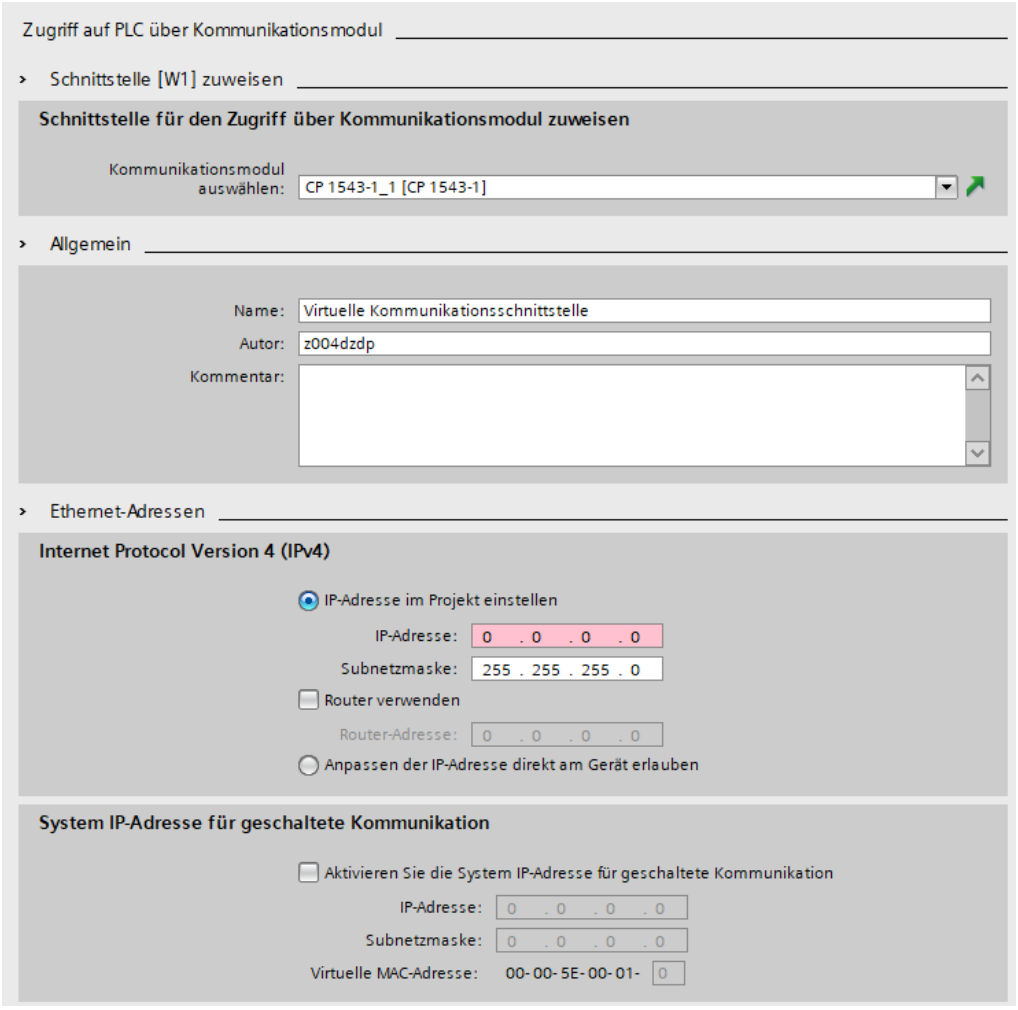


Bild 16-8 Konfigurieren der System IP-Adresse von W1

Ergebnis: Sie haben der virtuellen Schnittstelle W1 einen CP 1543-1 zugewiesen und die Geräte-/System IP-Adresse konfiguriert.

16.3 Verhalten beim Syncup

Verhalten von Kommunikationsverbindungen über die System IP-Adresse im Systemzustand SYNCUP

- HMI-, PG-, und S7-Verbindungen werden vorübergehend geschlossen. Für eine kurze Zeitdauer während des SYNCUPs ist es nicht möglich, Verbindungen zum redundanten System S7-1500R/H aufzubauen.
- Alle vorhandenen Verbindungen der Open User Communication werden unterbrochen:
  - Verbindungen, die die CPUs des redundanten Systems als aktiver Verbindungspartner eingerichtet haben, werden nach dem SYNCUP neu eingerichtet.
  - Das redundante System S7-1500R/H richtet nach dem SYNCUP Verbindungsendpunkte für den passiven Verbindungsaufbau neu ein.
- Die Bearbeitung von laufenden Instanzen der Anweisungen TSEND und TRCV wird gestoppt. Der Bausteinparameter STATUS liefert $80C4_H$ (Temporärer Kommunikationsfehler).

16.4 Verhalten bei Primary-Backup-Umschaltung

Verhalten von Kommunikationsverbindungen über die System IP-Adresse während einer Primary-Backup-Umschaltung

- Laufende Instanzen der Anweisungen TSEND und TRCV werden gestoppt und liefern den Status $80C4_H$ (Temporärer Kommunikationsfehler).
- Verbindungen, die das redundante System S7-1500R/H erfolgreich aktiv aufgebaut hatte, baut die neue Primary-CPU erneut auf.
- Die neue Primary-CPU richtet Verbindungsendpunkte für den passiven Verbindungsaufbau neu ein.

HINWEIS

Erhöhte Dauer der Verbindungsunterbrechung

Wenn das remote System nach der Primary-Backup-Umschaltung nicht aktiv sendet, kann es dazu kommen, dass die Verbindungsüberwachung (z. B. TCP-Keep-Alive oder Applikation) durch das remote System ausgeführt werden muss, bis die Verbindung wieder etabliert werden kann.

16.5 Verbindungsressourcen des redundanten Systems S7-1500R/H

Maximale Anzahl Verbindungsressourcen des redundanten Systems S7-1500R/H

Das redundante System S7-1500R/H unterstützt eine maximale Anzahl an Verbindungsressourcen.

Die eingesetzte CPU bestimmt die maximale Anzahl der Ressourcen für das redundante System:

- CPU 1513R: max. 88 Verbindungsressourcen
- CPU 1515R: max. 128 Verbindungsressourcen (ab V3.0), max. 108 Verbindungsressourcen (bis V2.9.x)
- CPU 1517H: max. 288 Verbindungsressourcen
- CPU 1518HF: max. 320 Verbindungsressourcen

Belegung der Verbindungsressourcen

Kommunikationsverbindungen belegen Verbindungsressourcen im redundanten System S7-1500R/H.

Jede Kommunikationsverbindung zum redundanten System S7-1500R/H belegt Verbindungsressourcen in der S7-1500R/H-Station. Die S7-1500R/H-Station umfasst den Hardwareaufbau von beiden CPUs des redundanten Systems S7-1500R/H.

Abhängig von der verwendeten IP-Adresse belegt eine Kommunikationsverbindung zusätzlich auch Verbindungsressourcen in einer oder beiden CPUs des redundanten Systems S7-1500R/H.

Die folgende Tabelle zeigt, in welcher CPU eine Kommunikationsverbindung in Abhängigkeit von der verwendeten IP-Adresse Verbindungsressourcen belegt.

Verbindung über...	Verbindungsressourcen der Station	Verbindungsressourcen CPU mit Redundanz-ID 1	Verbindungsressourcen CPU mit Redundanz-ID 2
eine System IP-Adresse	X	X	X
eine Geräte IP-Adresse der CPU mit Redundanz-ID 1	X	X	-
eine Geräte IP-Adresse der CPU mit Redundanz-ID 2	X	-	X

Anzeige der belegten Verbindungsressourcen in STEP 7

Voraussetzung: Online-Verbindung zum redundanten System S7-1500R/H
Sie finden die Online-Anzeige der Verbindungsressourcen im Inspektorfenster unter "Diagnose > Verbindungsinformation". STEP 7 zeigt immer die Verbindungsressourcen der selektierten CPU und der S7-1500R/H-Station an.

Verbindungsressourcen

	Ressourcen der Station					Ressourcen des Moduls	
	Reserviert			Dynamisch		CPU 1517H-3 PN (R0/S1)	
Maximale Anzahl der Ressourcen:		10	10	150	150	160	160
	Maximum	Konfigurierte	Verwendet	Konfigurierte	Verwendet	Konfigurier...	Verwendet
PG-Kommunikation:	4	-	2	-	0	-	2
HMI-Kommunikation:	4	0	0	0	0	0	0
S7-Kommunikation:	0	-	0	0	0	0	0
Open User Communication:	0	-	0	0	0	0	0
Web-Kommunikation:	2	-	0	-	0	-	0
Sonstige Kommunikation:	-	-	0	0	0	0	0
Insgesamt verwendete Ressourcen:		0	2	0	0	0	2
Verfügbare Ressourcen:		10	8	150	150	160	158

Bild 16-9 Anzeige der Verbindungsressourcen des redundanten Systems S7-1500R/H in STEP 7

16.6 HMI-Kommunikation mit dem redundanten System S7-1500R/H

16.6.1 HMI-Verbindung über die System IP-Adresse einrichten

Voraussetzungen

- Ein redundantes System S7-1500R/H, z. B. CPU 1513R-1PN
- System IP-Adresse ist aktiviert
- HMI-Gerät mit PROFINET-Schnittstelle

Vorgehen

Um eine HMI-Verbindung zu einem redundanten System S7-1500R/H einzurichten, gehen Sie folgendermaßen vor:

1. Selektieren Sie in der Netzsicht von STEP 7 eine PROFINET-Schnittstelle des HMI-Geräts.
2. Ziehen Sie per Drag&Drop eine Linie zwischen der PROFINET-Schnittstelle des HMI-Geräts und einer PROFINET-Schnittstelle des redundanten Systems S7-1500R/H.
   Das HMI-Gerät und das redundante System S7-1500R/H sind miteinander vernetzt.

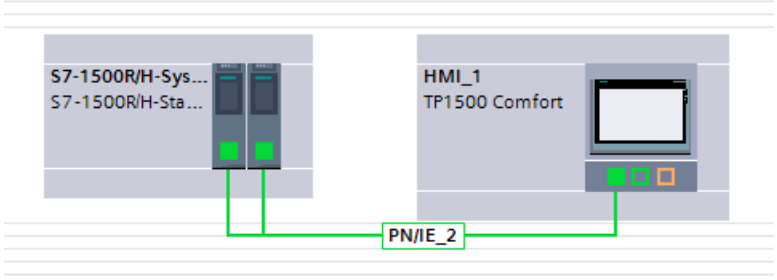


Bild 16-10 HMI-Gerät mit dem redundanten System S7-1500R/H vernetzen

3. Klicken Sie in der Funktionsliste auf das Symbol "Verbindungen". Sie aktivieren damit den Verbindungsmodus.
4. Ziehen Sie per Drag&Drop eine Linie zwischen HMI-Gerät und einer CPU des redundanten Systems S7-1500R/H.
   Die Liste "Verbindungspartner" öffnet sich.

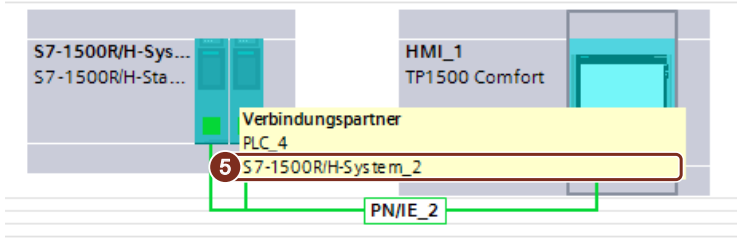


Bild 16-11 HMI-Verbindung zum redundanten System S7-1500R/H einrichten

5. Wählen Sie in der Liste "Verbindungspartner" das redundante System S7-1500R/H aus.

Ergebnis: Sie haben eine HMI-Verbindung zwischen dem HMI-Gerät und dem redundanten System S7-1500R/H eingerichtet. Die HMI-Verbindung nutzt die System-IP-Adresse. Das HMI-Gerät verbindet sich immer mit der Primary-CPU.

HMI-Verbindung auf die Geräte IP-Adresse umstellen

Um die HMI-Verbindung fest auf die ausgewählte CPU umzustellen, deaktivieren Sie das Kontrollkästchen "Verwenden Sie die System-IP-Adresse für geschaltete Kommunikation" in den Eigenschaften der HMI-Verbindung. Die HMI-Verbindung nutzt dann die Geräte IP-Adresse der PROFINET-Schnittstelle. Wenn diese CPU ausfällt, dann fällt die HMI-Verbindung zu dieser CPU dauerhaft aus.

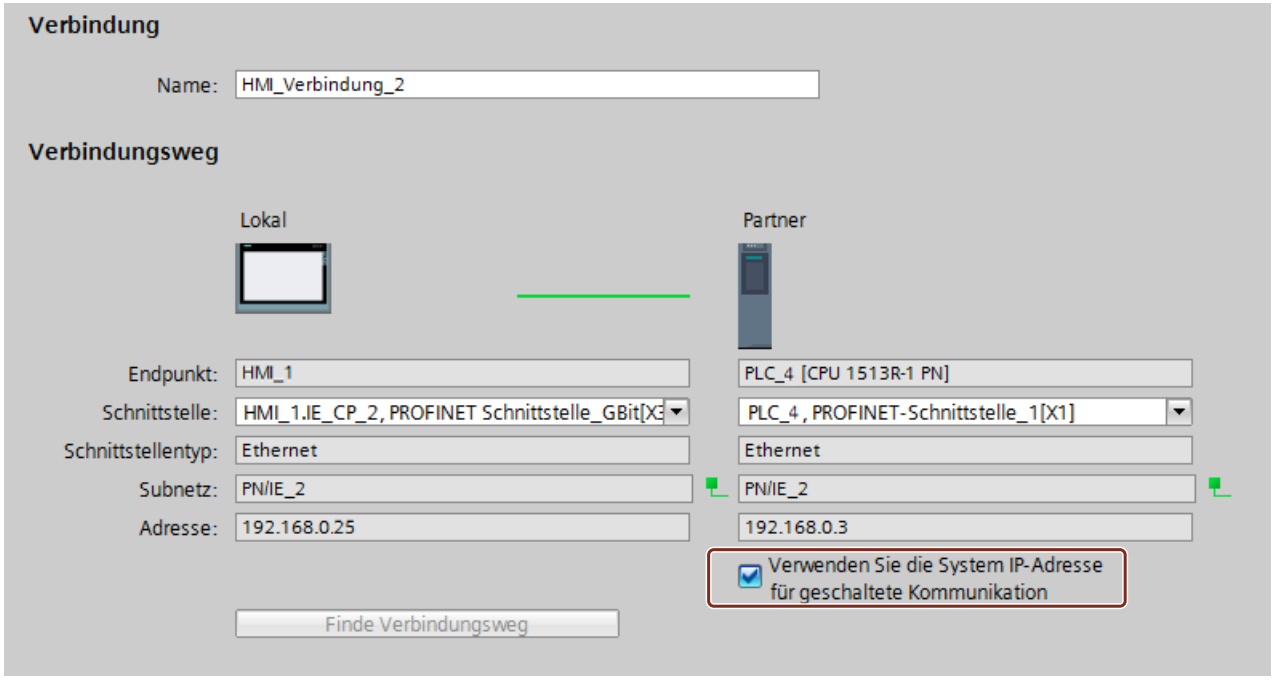


Bild 16-12 Eigenschaften der HMI-Verbindung

HINWEIS

Automatische Einrichtung HMI-Verbindung

Wenn Sie vom redundanten System S7-1500R/H eine Variable per Drag & Drop in ein HMI-Bild oder in die HMI-Variablentabelle hineinziehen, richtet STEP 7 automatisch eine HMI-Verbindung ein. Diese HMI-Verbindung besteht standardmäßig zwischen der PROFINET-Schnittstelle des HMI-Geräts und der PROFINET-Schnittstelle X1 der CPU mit Redundanz-ID 1. Die Verbindung verwendet die Geräte IP-Adresse der PROFINET-Schnittstelle X1.

In den Eigenschaften der HMI-Verbindung können Sie die HMI-Verbindung auf eine System IP-Adresse umstellen.

Weitere Informationen

Sie können eine HMI-Verbindung zum redundanten System S7-1500R/H auch über die Geräte IP-Adresse einrichten. Mit Hilfe von Skripten in der HMI-Projektierung wird die Verbindung von der ausgefallenen CPU auf die noch laufende CPU automatisch umgeschaltet. Die Beschreibung zu dieser Vorgehensweise finden Sie im folgenden FAQ (https://support.industry.siemens.com/cs/ww/de/view/109781687).

16.7 Open User Communication mit dem redundanten System S7-1500R/H

Einleitung

S7-1500R/H-Systeme ab FW-Version V3.1 unterstützen auch Secure Open User Communication (Secure OUC).
Wenn Sie ein S7-1500R/H-System ab FW-Version V3.1 mit den Kommunikationsprozessoren CP 1543-1 erweitern, können Sie Secure OUC auch über diese angeschlossenen CPs nutzen.
Voraussetzungen:

- STEP 7 ab V19
- CP 1543-1 ab FW-Version V3.0

Protokolle der Secure Open User Communication für das redundanten System S7-1500R/H

Die folgende Tabelle zeigt Ihnen, welche Protokolle der Open User Communication Sie für das redundante System S7-1500R/H einsetzen können und die dazu passenden Systemdatentypen und Anweisungen.

Tabelle 16-1 Protokolle, Systemdatentypen und einsetzbare Anweisungen für Open User Communication mit dem redundanten System S7-1500R/H

Protokoll	Systemdatentyp	Anweisungen
TCP	• TCON_QDN • TCON_QDN_SEC • TCON_IP_v4 • TCON_IP_V4_SEC	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C oder • TCON, TSEND/TRCV oder • TCON, TUSEND/TURCV (Abbau der Verbindung über TDISCON möglich)
TLS over TCP		
ISO-on-TCP	• TCON_IP_RFC	
UDP	• TCON_IP_v4 • TADDR_Param • TADDR_SEND_QDN • TADDR_RCV_IP	Verbindung herstellen und Daten senden/empfangen über: • TSEND_C/TRCV_C • TUSEND/TURCV/TRCV (Abbau der Verbindung über TDISCON möglich)
Modbus TCP	• TCON_IP_v4 • TCON_IP_V4_SEC • TCON_QDN • TCON_QDN_SEC	• MB_CLIENT • MB_RED_CLIENT • MB_SERVER • MB_RED_SERVER

16.7.1 Verbindung der Open User Communication mit dem redundanten System S7-1500R/H einrichten

Open User Communication über integrierte PROFINET-Schnittstellen der CPU

Das redundante System S7-1500R/H kann über Open User Communication mit anderen Geräten kommunizieren.

Die Verbindungen richten Sie im Anwenderprogramm ein, z. B. über die Anweisung "TSEND_C". Projektierte Verbindungen unterstützt das redundante System S7-1500R/H nicht.

Sie können die Verbindungen über die folgenden IP-Adressen der integrierten PROFINET-Schnittstellen einrichten:

- Geräte IP-Adressen:
  Im redundanten Betrieb kann das redundante System über alle Geräte IP-Adressen des redundanten Systems Verbindungen auf- bzw. abbauen und Daten senden bzw. empfangen.
  Wenn Sie die Verbindung über eine Geräte IP-Adresse einrichten, dann läuft die Kommunikation über die zugehörige CPU. Wenn die CPU ausfällt, dann fällt die gesamte Kommunikation über die Geräte IP-Adressen dieser CPU aus.
- System IP-Adressen:
  Wenn Sie die Verbindung über eine System IP-Adresse einrichten, dann läuft die Kommunikation immer über die Primary-CPU.

Entscheiden Sie anhand Ihres Anwendungsfalls, wie Sie die Verbindungen einrichten möchten.

Open User Communication über Kommunikationsprozessoren CP 1543-1

Wenn Sie ein redundantes System S7-1500R/H mit den Kommunikationsprozessoren CP 1543-1 erweitern, können Sie zusätzlich die folgenden Kommunikationsmöglichkeiten nutzen:

- Lokale Schnittstelle X1 des CP 1543-1
- Eigene Geräte IP-Adresse für die virtuelle Schnittstelle W1 jeder R/H-CPU
- Gemeinsame System IP-Adresse für die virtuelle Schnittstelle W1 des R/H-Systems

Informationen wie Sie einen CP 1543-1 für ein S7-1500R/H-System verwenden, finden Sie im Kapitel System IP-Adressen bei Kommunikationsprozessoren (Seite 418).

Verbindung über eine System IP-Adresse einrichten

Im Folgenden ist beschrieben, wie Sie eine Verbindung zu einer Partner-CPU über eine System IP-Adresse einer integrierten PROFINET-Schnittstelle des redundanten Systems S7-1500R/H einrichten.

Die Verbindung richten Sie im Anwenderprogramm des redundanten Systems S7-1500R/H z. B. mit einer Anweisung TSEND_C ein. Im Anwenderprogramm der Partner-CPU legen Sie eine entsprechende Anweisung TRCV_C an.

Die Vorgehensweise ist am Beispiel einer TCP-Verbindung zwischen dem redundanten System S7-1500R/H und einer CPU 1516-3PN/DP beschrieben.

Voraussetzungen

- Ein redundantes System S7-1500R/H als TCP-Client, z. B. 2 CPUs 1513-1PN
- System IP-Adresse der PROFINET-Schnittstelle X1 ist aktiviert.
- Verbindungspartner als TCP-Server, z. B. CPU 1516-3 PN/DP
- Die PROFINET-Schnittstellen X1 der redundanten CPUs 1513R und die PROFINET-Schnittstelle X2 der CPU 1516-3PN/DP befinden sich im selben Subnetz.

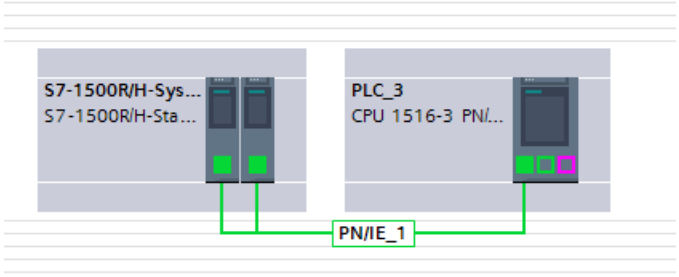


Bild 16-13 Beispiel Konfiguration für TCP-Verbindung

Anweisung TSEND_C im Anwenderprogramm des redundanten Systems S7-1500R/H

Um eine TCP-Verbindung zu einer anderen CPU einzurichten, gehen Sie folgendermaßen vor:

1. Legen Sie im Anwenderprogramm eine Anweisung "TSEND_C" an.

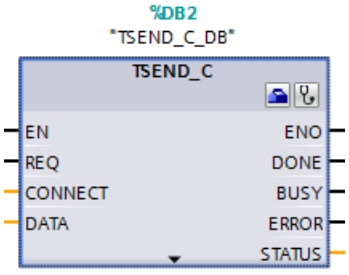


Bild 16-14 S7-1500R/H: Anweisung "TSEND_C"

2. Selektieren Sie die Anweisung "TSEND_C".
3. Navigieren Sie ins Inspektorfenster zu "Eigenschaften > Konfiguration > Verbindungsparameter".
   Auf der linken Seite sehen Sie das redundante System S7-1500R/H als lokalen Endpunkt der Verbindung:
   - "Schnittstelle:": Die Schnittstelle X1 ist voreingestellt.
   - "Subnetz:": Wenn die Schnittstelle X1 einem S7-Subnetz zugeordnet ist, dann zeigt STEP 7 den Namen des S7-Subnetzes an.
   - Das Optionskästchen "System IP-Adresse verwenden" ist aktiviert. Bei "Adresse:" steht die System IP-Adresse des redundanten Systems S7-1500R/H.

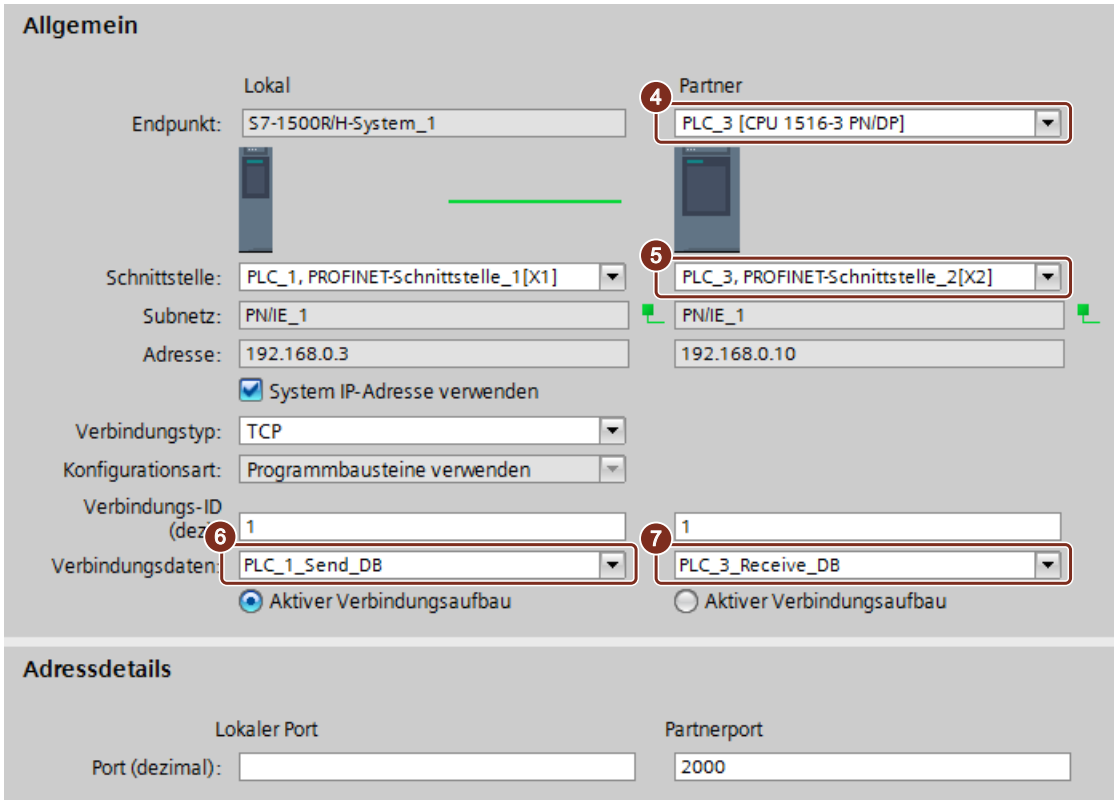


Bild 16-15 S7-1500R/H: Parametrierung der Anweisung TSEND_C in STEP 7

4. Wählen Sie bei "Partner" unter "Endpunkt:" die CPU 1516-3PN/DP als Kommunikationspartner aus.
5. Wählen Sie bei "Partner" unter "Schnittstelle:" die PROFINET-Schnittstelle X2 der CPU 1516-3PN/DP aus.
6. Wählen Sie bei "Lokal" unter "Verbindungsdaten" die Einstellung "<neu>" aus.
   STEP 7 legt im Anwenderprogramm des redundanten Systems S7-1500R/H einen Datenbaustein für die Verbindungsdaten an, z. B. "PLC_1_Send_DB".
   Bei "Verbindungstyp" ist "TCP" voreingestellt.
7. Wählen Sie bei "Partner" unter "Verbindungstyp:" die Einstellung "neu" aus.
   STEP 7 legt im Anwenderprogramm der anderen CPU einen Datenbaustein für die Verbindungsdaten an, z. B "PLC_3_Receive_DB".

Anweisung TRCV_C im Anwenderprogramm der CPU 1516-3PN/DP

Legen Sie im Anwenderprogramm der CPU 1516-3PN/DP eine Anweisung TRCV_C an und parametrieren Sie diese wie folgt:

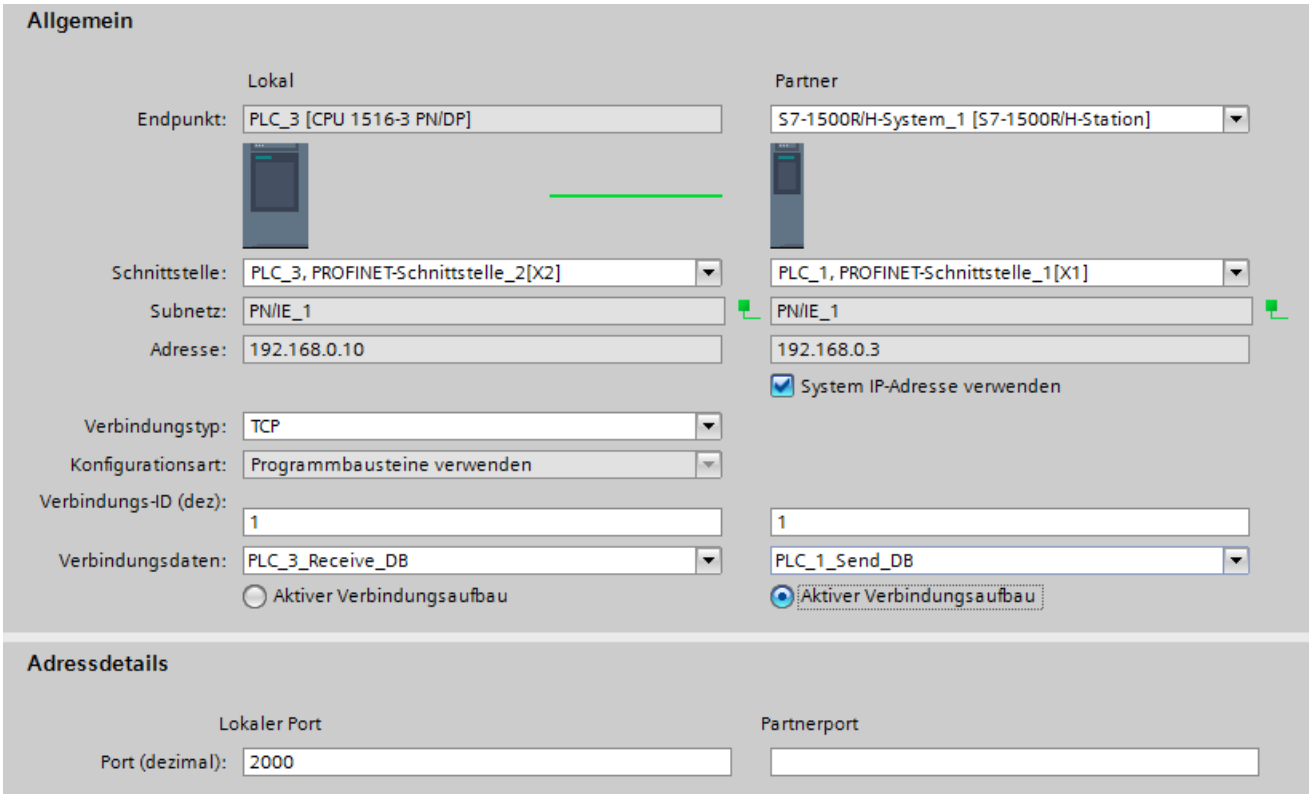


Bild 16-16 S7-1500-3PN/DP: Parametrierung der Anweisung TRCV_C in STEP 7

Verbindung über Geräte IP-Adresse einrichten

Um eine OUC-Verbindung über eine Geräte IP-Adresse einer der beiden CPUs einzurichten:

- Wählen Sie eine passende PROFINET-Schnittstelle des redundanten Systems S7-1500R/H aus.
- Deaktivieren Sie das Kontrollkästchen "System IP-Adresse verwenden".

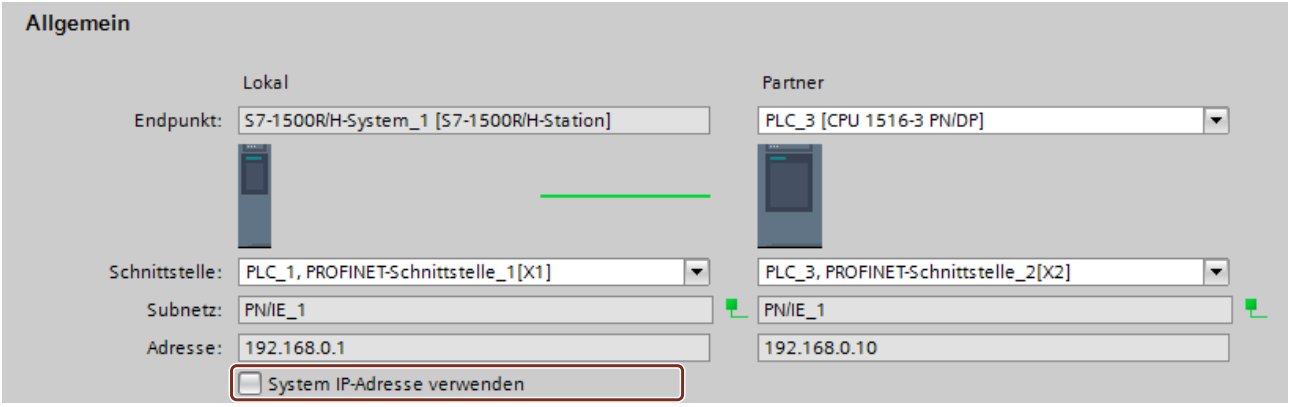


Bild 16-17 OUC-Verbindung über eine Geräte IP-Adresse

Weitere Informationen

Weitere Informationen zu den Systemzuständen finden Sie im Systemhandbuch S7-1500R/H (https://support.industry.siemens.com/cs/ww/de/view/109754833).
Weitere Informationen zur Konfiguration und Parametrierung Ihres PROFINET IO-Systems finden Sie im Funktionshandbuch PROFINET (https://support.industry.siemens.com/cs/ww/de/view/49948856).

16.7.2 Open User Communication mit Kommunikationsprozessoren CP 1543-1

Einleitung

Sie können über die Kommunikationsprozessoren CP 1543-1 sowohl Open User Communication (OUC) als auch Secure OUC anwenden.
Wenn Sie Secure OUC anwenden möchten, müssen zusätzlich die folgenden Voraussetzungen erfüllt sein, damit Sie die Geräte- und CA-Zertifikate hantieren können:

- Projektschutz ist aktiviert
- Security-Funktionen sind aktiviert in jedem CP 1543-1, den Sie für Secure OUC verwenden möchten
- Gleiche Security-Einstellungen in jedem CP an Primary- und Backup-CPU am selben Steckplatz
- Für beide CPs sind dieselben CA-Zertifikate konfiguriert
- Die Gerätezertifikate der CPs verweisen jeweils auf beide CPs, z. B. Konfigurieren eines alternativen Namen des Zertifikatsinhabers (SAN) in beiden Gerätezertifikaten

HINWEIS

Kein automatisches Synchronisieren der Sicherheitseinstellungen

In STEP 7 werden die Sicherheitseinstellungen zwischen den CPs nicht automatisch synchronisiert. Konfigurieren Sie deshalb die CPs am selben Steckplatz von Primary- und Backup-CPU identisch.

Verbindungsmöglichkeiten für Open User Communication über CPs

Für die folgenden Konfigurationen geht die OUC-Verbindung zu einem Partner immer über die lokale CP-Schnittstelle, unabhängig davon ob OUC oder Secure OUC verwendet wird. Je nachdem welche Konfiguration Sie verwenden, müssen Sie beim Einrichten der Kommunikationsverbindung die Verbindungsparameter anpassen. Wählen Sie bei den Verbindungsparametern die entsprechende HW-Kennung der Schnittstelle nach einer der folgenden Möglichkeiten:

- Verwenden der CP-Schnittstelle mit lokaler IP-Adresse:
  Geben Sie beim Verbindungsparameter "InterfaceId" die HW-Kennung der lokalen Ethernet-Schnittstelle eines CPs an, z. B. "Local1~CP_1543-1_1~Ethernet_interface_1". Im Systemzustand RUN-Redundant werden die CP-Schnittstellen sowohl der Primary- als auch der Backup-CPU genutzt.
- Verwenden der virtuellen Schnittstelle W1 der CPU mit konfigurierter Geräte IP-Adresse:
  Geben Sie beim Verbindungsparameter "InterfaceId" die HW-Kennung der virtuellen Schnittstelle einer CPU an, z. B. "Local1~Virtual_communication_interface". Im Systemzustand RUN-Redundant werden die virtuellen Schnittstellen W1 sowohl der Primary- als auch der Backup-CPU genutzt.
- Verwenden der virtuellen Schnittstelle W1 der CPU mit konfigurierter System IP-Adresse:
  Geben Sie beim Verbindungsparameter "InterfaceId" die HW-Kennung des Objekts mit der Zeichenfolge "...HSystemIPRef..." und der Kennzeichnung "Virtual" im Default-Namen an, z. B. "Local1~Virtual_communication_interface~HsystemIPRef_1". Dieses Objekt referenziert die System IP-Adresse der virtuellen Schnittstelle W1 der CPU.

Verbindung über CPs erstellen

Sie erstellen die Verbindung wie bei einer integrierten PROFINET-Schnittstelle des redundanten Systems S7-1500R/H. Jedoch müssen Sie die Verbindungsparameter entsprechend anpassen. Wählen Sie in der Auswahlliste "Schnittstelle" den zu verwendenden Kommunikationsprozessor CP 1543-1.
Wenn Sie die Option "System IP-Adresse verwenden" verwenden möchten, muss für die virtuelle Schnittstelle W1 eine System IP-Adresse konfiguriert sein.

Auswahl der Zertifikate für Secure Open User Communication über CPs

Abhängig davon, wie Sie die S7-1500R/H-Station mit einem CP 1543-1 konfigurieren, sind die folgenden Zertifikate-Regeln für die Kommunikation mit Verbindungspartnern gültig:

- Kommunikation über die CP-Schnittstelle mit lokaler IP-Adresse:
  Verwenden Sie das Gerätezertifikat des CPs zur Authentifizierung. Damit der Verbindungspartner das Gerätezertifikat validieren kann, muss dort das CA-Zertifikat des CPs unter "Zertifikate der Partner-Geräte" vorhanden sein.
- Kommunikation über die virtuellen Schnittstellen W1 der CPUs mit konfigurierter System IP-Adresse:
  Verwenden Sie ein Gerätezertifikat der CPU zur Authentifizierung. Damit der Verbindungspartner das Gerätezertifikat validieren kann, muss dort die Zertifizierungsstelle der CPU unter "Zertifikate der Partner-Geräte" vorhanden sein.

Weitere Informationen

Informationen, wie Sie Zertifikate erstellen oder zuweisen, finden Sie im Kapitel Verwalten von Zertifikaten (Seite 62).
Informationen zur Secure Open User Communication über die Schnittstelle eines CPs finden Sie im Kapitel Secure OUC über CP-Schnittstelle (Seite 93).

16.8 OPC UA-Server in einem S7-1500R/H-System nutzen

16.8.1 OPC UA-Server-Unterstützung für S7-1500R/H-Systeme

S7-1500R/H mit OPC UA-Server

Ab Firmware-Version V3.1 haben S7-1500R/H-CPUs einen OPC UA-Server. Alle weiteren Informationen dazu finden Sie nach Lieferfreigabe dieser Version hier: Produktinformation zur Dokumentation S7-1500/ET 200MP, S7-1500R/H (https://support.industry.siemens.com/cs/de/de/view/68052815).

17

Industrial Ethernet Security mit CP 1543-1

Umfassender Schutz - Aufgabe von Industrial Ethernet Security

Mit Industrial Ethernet Security können einzelne Geräte, Automatisierungszellen oder Netzsegmente eines Ethernet-Netzwerks abgesichert werden. Zusätzlich kann die Datenübertragung durch die Kombination unterschiedlicher Sicherheitsmaßnahmen geschützt werden vor:

- Datenspionage
- Datenmanipulation
- unberechtigten Zugriffen

Sicherheitsmaßnahmen

- Firewall
  - IP-Firewall mit Stateful Packet Inspection (Layer 3 und 4)
  - Firewall auch für Ethernet-"Non-IP"-Telegramme gemäß IEEE 802.3 (Layer2)
  - Bandbreitenbegrenzung
  - Globale Firewall-Regeln

  Alle Netzknoten, die sich im internen Netzsegment eines CP 1543-1 befinden, werden durch dessen Firewall geschützt. Ausnahme: Wenn Sie über die Schnittstelle des CP mit der Funktion ""Zugriff auf PLC über Kommunikationsmodul" auf die CPU zugreifen, dann schützt die Firewall diese Verbindung nicht.
- Logging
  Zur Überwachung lassen sich Ereignisse in Log-Dateien speichern, die mit Hilfe des Projektierwerkzeugs ausgelesen werden oder automatisch an einen Syslog-Server gesendet werden können.
- HTTPS
  Zur verschlüsselten Übertragung von Webseiten, z. B. bei der Prozesskontrolle.
- FTPS (expliziter Modus)
  Zur verschlüsselten Übertragung von Dateien.
- Gesichertes NTP
  Zur sicheren Uhrzeitsynchronisierung und -übertragung.
- SNMPv3
  Zur abhörsicheren Übertragung von Netzwerkanalyseinformationen.

- VPN-Gruppen
  Den CP 1543-1 können Sie mit anderen Security-Baugruppen per Projektierung zu VPN-Gruppen zusammenfassen. Zwischen allen Security-Baugruppen einer VPN-Gruppe werden IPsec-Tunnel aufgebaut (VPN). Alle internen Knoten dieser Security-Baugruppen können mittels dieser Tunnel gesichert miteinander kommunizieren.
- Schutz für Geräte und Netzsegmente
  Die Schutzfunktionen Firewall und VPN-Gruppen kann sich über den Betrieb einzelner Geräte, mehrerer Geräte wie auch ganzer Netzsegmente erstrecken.

Weitere Informationen

Eine Übersicht mit Links zu den wichtigsten Beiträgen zu Industrial Security finden Sie im diesem FAQ (https://support.industry.siemens.com/cs/ww/de/view/92651441).

17.1 Firewall

Aufgaben der Firewall

Die Firewall-Funktionalität hat die Aufgabe, Netze und Stationen vor Fremdbeeinflussungen und Störungen zu schützen. Das bedeutet, dass nur bestimmte, vorher festgelegte Kommunikationsbeziehungen erlaubt werden.

Zur Filterung des Datenverkehrs können u. a. IPv4-Adressen, IPv4-Subnetze, Portnummern oder MAC-Adressen verwendet werden.

Die Firewall-Funktionalität kann für folgende Protokollebenen konfiguriert werden:

- IP-Firewall mit Stateful Packet Inspection (Layer 3 und 4)
- Firewall auch für Ethernet-"Non-IP"-Telegramme gemäß IEEE 802.3 (Layer 2)

Firewall-Regeln

Firewall-Regeln beschreiben, welche Pakete in welche Richtung erlaubt bzw. verboten werden.

17.2 Logging

Funktionalität

Zu Test- und Überwachungszwecken verfügt das Security-Modul über Diagnose- und Logging-Funktionen.

- Diagnosefunktionen
  Hierunter sind verschiedene System- und Statusfunktionen zu verstehen, die Sie im Online-Modus anwenden können.
- Logging-Funktionen
  Hierbei geht es um die Aufzeichnung von System- und Sicherheitsereignissen. Die Aufzeichnung erfolgt je nach Ereignistyp in flüchtige oder dauerhafte lokale Pufferbereiche des CP 1543-1. Alternativ kann auch eine Aufzeichnung in einem Netzwerk-Server erfolgen.
  Die Parametrierung und Auswertung dieser Funktionen setzt eine Netzwerkverbindung voraus.

Ereignisse mit Logging-Funktionen aufzeichnen

Welche Ereignisse aufgezeichnet werden sollen, legen Sie mit den Log-Einstellungen fest. Dabei können Sie für die Aufzeichnung folgende Varianten konfigurieren:

- Lokales Logging
  Bei dieser Variante zeichnen Sie die Ereignisse in lokalen Puffern des CP 1543-1 auf. Im Online-Dialog des Security Configuration Tool können Sie dann auf diese Aufzeichnungen zugreifen, diese sichtbar machen und in der Service-Station archivieren.
- Netzwerk Syslog
  Beim Netzwerk Syslog nutzen Sie einen im Netz vorhandenen Syslog-Server. Dieser zeichnet die Ereignisse entsprechend der Konfiguration in den Log-Einstellungen auf.

17.3 NTP-Client

Funktionalität

Zur Überprüfung der zeitlichen Gültigkeit eines Zertifikats und für die Zeitstempel von Log-Einträgen werden auf dem CP 1543-1, genauso wie auf der CPU, Datum und Uhrzeit geführt. Diese Uhrzeit ist per NTP synchronisierbar. Der CP 1543-1 leitet die synchronisierte Uhrzeit über den Rückwandbus des Automatisierungssystems an die CPU weiter. So erhält auch die CPU eine synchronisierte Uhrzeit für die Zeitereignisse in der Anwenderprogrammbearbeitung.

Das automatische Stellen und der periodische Abgleich der Uhrzeit wird über einen gesicherten oder ungesicherten NTP-Server realisiert. Sie können dem CP 1543-1 max. 4 NTP-Server zuweisen. Eine gemischte Konfiguration von ungesicherten und gesicherten NTP-Servern ist nicht möglich.

17.4 SNMP

Funktionalität

Der CP 1543-1 unterstützt, genauso wie die CPU, die Übertragung von Managementinformationen über das Simple Network Management Protocol (SNMP). Dafür ist auf dem CP/der CPU ein "SNMP-Agent" installiert, der die SNMP-Anfragen entgegennimmt und beantwortet. Informationen über die Eigenschaften von SNMP-fähigen Geräten sind in sogenannten MIB-Dateien (Management Information Base) hinterlegt, für die der Benutzer die notwendigen Rechte haben muss.

Beim SNMPv1 wird der "Community String" mitgesendet. Der "Community String" ist wie ein Passwort, das zusammen mit der SNMP-Anfrage verschickt wird. Wenn der Community String korrekt ist, wird die angeforderte Information gesendet. Wenn der String falsch ist, wird die Anfrage verworfen.

Bei SNMPv3 können die Daten verschlüsselt übertragen werden. Dazu wählen Sie entweder ein Authentifizierungsverfahren (z. B. SHA) oder ein Authentifizierungs- und Verschlüsselungsverfahren (z. B. AES).

Sie können die Verwendung von SNMP für den CP/die CPU aktivieren und deaktivieren. Deaktivieren Sie SNMP, wenn die Sicherheitsrichtlinien in ihrem Netzwerk kein SNMP zulassen oder Sie eine eigene SNMP-Lösung verwenden.

Wie Sie SNMP für die CPU aktivieren und deaktivieren, finden Sie im Kapitel SNMP (Seite 114).

17.5 VPN

Funktionalität

Für Security-Baugruppen, die das interne Netz schützen, stellen VPN-Tunnel (Virtual Private Network) eine gesicherte Datenverbindung durch das unsichere externe Netz zur Verfügung.

Die Baugruppe verwendet für die Tunnelung das IPsec-Protokoll (Tunnelmodus von IPsec).

In STEP 7 können Sie Security-Baugruppen VPN-Gruppen zuordnen. Zwischen allen Baugruppen einer VPN-Gruppe werden automatisch VPN-Tunnel aufgebaut. Dabei kann eine Baugruppe in einem Projekt parallel mehreren verschiedenen VPN-Gruppen angehören.

Glossar

Anweisung

Die kleinste selbstständige Einheit eines Anwenderprogramms, durch Struktur, Funktion oder Verwendungszweck als abgegrenzter Teil des Anwenderprogramms charakterisiert. Die Anweisung stellt eine Arbeitsvorschrift für den Prozessor dar.

Anwenderprogramm

Bei SIMATIC wird zwischen dem Betriebssystem der CPU und Anwenderprogrammen unterschieden. Das Anwenderprogramm enthält alle Anweisungen, Deklarationen und Daten, durch die eine Anlage oder ein Prozess gesteuert werden können. Das Anwenderprogramm ist einem programmierbaren Modul (z. B. CPU, FM) zugeordnet und ist in kleinere Einheiten strukturierbar.

Automatisierungssystem

Speicherprogrammierbare Steuerung für die Regelung und Steuerung von Prozessketten der verfahrenstechnischen Industrie und der Fertigungstechnik. Je nach Automatisierungsaufgabe setzt sich das Automatisierungssystem aus unterschiedlichen Komponenten und integrierten Systemfunktionen zusammen.

Backup-CPU

Wenn sich das R-/H-System im Systemzustand RUN-Redundant befindet, dann führt die Primary-CPU den Prozess. Die Backup-CPU bearbeitet das Anwenderprogramm synchron und kann bei einem Ausfall der Primary-CPU die Prozessführung übernehmen.

Baumtopologie

Netzwerktopologie, gekennzeichnet von einer verzweigten Struktur: An jeden Busteilnehmer werden zwei oder weitere Busteilnehmer angeschlossen.

Betriebssystem

Software, die die Verwendung und den Betrieb eines Computers ermöglicht. Das Betriebssystem verwaltet Betriebsmittel wie Speicher, Ein- und Ausgabegeräte und steuert die Ausführung von Programmen.

Betriebszustände

Betriebszustände beschreiben das Verhalten einer einzelnen CPU zu jedem beliebigen Zeitpunkt.

Die CPUs von SIMATIC-Standardsystemen verfügen über die Betriebszustände STOP, ANLAUF und RUN.

Die Primary-CPU des redundanten Systems S7-1500R/H verfügt über die Betriebszustände STOP, ANLAUF, RUN, RUN-Syncup und RUN-Redundant. Die Backup-CPU verfügt über die Betriebszustände STOP, SYNCUP und RUN-Redundant.

Bus

Übertragungsmedium, das mehrere Teilnehmer miteinander verbindet. Die Datenübertragung kann elektrisch oder über Lichtwellenleiter sowohl seriell als auch parallel erfolgen.

Client

Teilnehmer in einem Netz, der von einem anderen Teilnehmer am Netz (Server) einen Dienst anfordert.

CM

→ Kommunikationsmodul

CP

→ Kommunikationsprozessor

CPU

Central **P**rocessing **U**nit - Zentralbaugruppe des S7-Automatisierungssystems mit Steuer- und Rechenwerk, Speicher, Betriebssystem und Schnittstelle für Programmiergerät.

DP-Master

Innerhalb von PROFIBUS DP ein Master in der Dezentralen Peripherie, der sich nach der Norm EN 50170, Teil 3, verhält.
→ *Siehe auch DP-Slave*

DP-Slave

Slave in der Dezentralen Peripherie, der am PROFIBUS mit dem Protokoll PROFIBUS DP betrieben wird und sich nach der Norm EN 50170, Teil 3, verhält.
→ *Siehe auch DP-Master*

Duplex

Datenübertragungsverfahren; es wird zwischen Voll- und Halbduplexverfahren unterschieden.
Halbduplex: Ein Kanal zum abwechselnden Datenaustausch steht zur Verfügung (abwechselnd Senden und Empfangen in jeweils eine Richtung).
Vollduplex: Zwei Kanäle zum gleichzeitigen Datenaustausch in beide Richtungen stehen zur Verfügung (gleichzeitiges Senden und Empfangen in beide Richtungen).

End-Entity-Zertifikat

→ *Siehe auch Gerätezertifikat*

Ethernet

Internationale Standardtechnologie für lokale Netzwerke (LAN), basierend auf Frames. Sie definiert Kabeltypen und Signalisierung für die Bitübertragungsschicht sowie Paketformate und Protokolle für die Medienzugriffskontrolle.

Ethernet-Netzwerkkarte

Elektronische Schaltung zur Verbindung eines Computers mit einem Ethernet-Netzwerk. Sie ermöglicht den Austausch von Daten/die Kommunikation innerhalb des Netzes.

Feldgerät

→ Gerät

FETCH/WRITE

Server-Dienste über TCP/IP, ISO-on-TCP und ISO für den Zugriff auf Systemspeicherbereiche von S7-CPUs. Der Zugriff (Client-Funktion) ist von einer SIMATIC S5 oder einem Fremdgerät/PC aus möglich. FETCH: Daten direkt lesen; WRITE: Daten direkt schreiben.

Freeport

Frei programmierbares ASCII-Protokoll; hier zur Datenübertragung über Punkt-zu-Punkt-Kopplung.

FTP

File **T**ransfer **P**rotocol; ein Netzwerkprotokoll zur Übertragung von Dateien über IP-Netzwerke. FTP wird benutzt, um Dateien vom Server zum Client herunter zu laden oder vom Client zum Server hochzuladen. Außerdem können über FTP Verzeichnisse angelegt und ausgelesen, sowie Verzeichnisse und Dateien umbenannt oder gelöscht werden.

Gerät

Oberbegriff für:

- Automatisierungssysteme (z. B. SPS, PC)
- Dezentrale Peripheriesysteme
- Feldgeräte (z. B. SPS, PC, Hydraulikgeräte, Pneumatikgeräte) und
- Aktive Netzkomponenten (z. B. Switches, Router)
- Netzübergänge zu PROFIBUS, AS-Interface oder anderen Feldbussystemen

Gerätezertifikate

Solche Zertifikate werden von einer Zertifizierungsstelle (CA) signiert.
Die Signatur eines End-Entity-Zertifikats wird mit dem öffentlichen Schlüssel des Zertifikats der Zertifizierungsstelle überprüft.
Die Attribute "Antragsteller" und "Aussteller" dürfen nicht identisch sein.
Bei "Antragsteller" steht zum Beispiel der Name eines Programms, wie beim OPC UA Applikations-Zertifikat.
Bei "Aussteller" steht die Zertifizierungsstelle, die dieses Zertifikat signiert hat.
Das Feld "CA" muss auf "False" stehen.

Geschaltete Kommunikation

Zusätzlich zu den Geräte IP-Adressen der CPUs unterstützt das redundante System S7-1500R/H System IP-Adressen:

- System IP-Adresse für die PROFINET-Schnittstellen X1 der beiden CPUs (System IP-Adresse X1)
- System IP-Adresse für die PROFINET-Schnittstellen X2 der beiden CPUs (System IP-Adresse X2)

Die System IP-Adressen verwenden Sie für die Kommunikation mit anderen Geräten (z. B. HMI-Geräte, CPUs, PG/PC). Die Geräte kommunizieren über die System IP-Adresse immer mit der Primary-CPU des redundanten Systems. Dadurch wird sichergestellt, dass der Kommunikationspartner nach einem Ausfall der Primary-CPU im redundanten Betrieb mit der neuen Primary-CPU (vorher Backup-CPU) im Systemzustand RUN-Solo kommunizieren kann.

HMI

Human **M**aschine **I**nterface, Gerät zur Visualisierung und Steuerung von Automatisierungsprozessen.

IE

→ Industrial Ethernet

IM

→ Interfacemodul

Industrial Ethernet

Richtlinie zum Aufbau eines Ethernets in industrieller Umgebung. Der wesentliche Unterschied zum Standard-Ethernet liegt in der mechanischen Belastbarkeit und Störunempfindlichkeit der einzelnen Komponenten.

Interfacemodul

Modul im Dezentralen Peripheriesystem. Das Interfacemodul verbindet das Dezentrale Peripheriesystem über einen Feldbus mit der CPU (IO-Controller/DP-Master) und bereitet die Daten für die Peripheriemodule auf.

Intermediate CA-Zertifikat

Das ist das Zertifikat einer Zertifizierungsstelle, das mit dem privaten Schlüssel einer Root-Zertifizierungsstelle signiert ist.
Eine solche zwischengeschaltete Zertifizierungsstelle signiert mit ihrem privaten Schlüssel End-Entity-Zertifikate.
Die Signatur dieser End-Entity-Zertifikate wird mit dem öffentlichen Schlüssel der Intermediate-Zertifizierungsstelle überprüft.
Die Attribute "Antragsteller" und "Aussteller" des Intermediate CA-Zertifikats dürfen nicht identisch sein: Diese Zertifizierungsstelle hat ihr Zertifikat ja nicht selbst signiert.
Das Feld "CA" muss auf "True" stehen.

IO-Controller, PROFINET IO-Controller

Zentrales Gerät in einem PROFINET-System, meist eine klassische speicherprogrammierbare Steuerung oder ein PC. Der IO-Controller richtet Verbindungen zu den IO-Devices ein, tauscht Daten mit ihnen aus, steuert und kontrolliert also das System.

IO-Device, PROFINET IO-Device

Gerät der Dezentralen Peripherie eines PROFINET-Systems, das von einem IO-Controller kontrolliert und gesteuert wird (z. B. dezentrale Ein-/Ausgänge, Ventilinseln, Frequenzumrichter, Switches).

IP-Adresse

Binäre Zahl, die im Zusammenhang mit dem Internetprotokoll (IP) als eindeutige Adresse in Computernetzwerken verwendet wird. Dadurch werden diese Geräte eineindeutig adressierbar und individuell erreichbar. Eine IPv4-Adresse kann mit Hilfe einer binären Subnetz-Maske so bewertet werden, dass sich ein Netzanteil bzw. ein Hostanteil als Struktur ergibt. Die textuelle Darstellung einer IPv4-Adresse besteht beispielsweise aus 4 Dezimalzahlen mit dem Wertebereich 0 bis 255. Die Dezimalzahlen sind durch einen Punkt voneinander getrennt.

IPv4-Subnetzmaske

Binäre Maske, mit der eine IPv4-Adresse (als binäre Zahl) in einen "Netzanteil" und einen "Hostanteil" aufteilt.

ISO-on-TCP-Protokoll

S7-Routing-fähiges Kommunikationsprotokoll für paketorientierte Übertragung von Daten im Ethernet; bietet eine Netzwerkadressierung. ISO-on-TCP-Protokoll ist für mittlere und große Datenmengen geeignet und erlaubt dynamische Datenlängen.

ISO-Protokoll

Kommunikationsprotokoll für nachrichten- bzw. paketorientierte Übertragung von Daten im Ethernet. Dieses Protokoll ist hardwarenahe, sehr schnell und lässt dynamische Datenlängen zu. Das ISO-Protokoll eignet sich für mittlere und große Datenmengen.

Kommunikationsmodul

Baugruppe für Kommunikationsaufgaben, die in einem Automatisierungssystem als Schnittstellenerweiterung der CPU (z. B. PROFIBUS) verwendet wird bzw. zusätzliche Kommunikationsmöglichkeiten (PtP) bietet.

Kommunikationsprozessor

Baugruppe für erweiterte Kommunikationsaufgaben, die spezielle Anwendungsfälle, z. B. im Bereich Security, abdeckt.

Konsistente Daten

Daten, die inhaltlich zusammengehören und beim Übertragen nicht getrennt werden dürfen.

Linientopologie

Netzwerktopologie, gekennzeichnet durch die Anordnung der Busteilnehmer in einer Reihe.

MAC-Adresse

Weltweit eindeutige Geräte-Identifikation für alle Ethernet-Geräte. Die MAC-Adresse wird bereits vom Hersteller vergeben und hat 3 byte Herstellerkennung und 3 byte Gerätekennung als laufende Nummer.

Master

Übergeordneter, aktiver Teilnehmer an der Kommunikation/am PROFIBUS-Subnetz. Der Master besitzt Buszugriffsrechte (Token), kann Daten anfordern und verschicken.
→ *Siehe auch DP-Master*

Modbus RTU

Remote **T**erminal **U**nit; Offenes Kommunikationsprotokoll für serielle Schnittstellen, welches auf einer Master-/Slave-Architektur basiert.

Modbus TCP

Transmission **C**ontrol **P**rotokoll; Offenes Kommunikationsprotokoll für Ethernet, welches auf einer Master/Slave-Architektur basiert. Die Daten werden als TCP/IP-Pakete übertragen.

Netz

Ein Netz besteht aus einem oder mehreren verknüpften Subnetzen mit einer beliebigen Zahl von Teilnehmern. Mehrere Netze können nebeneinander bestehen.

NTP

Das **N**etwork **T**ime **P**rotocol (NTP) ist ein Standard zur Synchronisierung von Uhren in Automatisierungssystemen über Industrial Ethernet. NTP verwendet das verbindungslose Transportprotokoll UDP für das Internet.

OPC UA

OPC Unified **A**utomation ist ein Protokoll für die Kommunikation zwischen Maschinen, entwickelt von der OPC Foundation.

PG

→ Programmiergerät

PNO

→ PROFIBUS-Nutzerorganisation

Port

Physikalische Anschlussmöglichkeit für Geräte, die PROFINET-Teilnehmer sind. PROFINET-Schnittstellen verfügen über einen oder mehrere Ports.

Primary-CPU

Wenn sich das R-/H-System im Systemzustand RUN-Redundant befindet, dann führt die Primary-CPU den Prozess. Die Backup-CPU bearbeitet das Anwenderprogramm synchron und kann bei einem Ausfall der Primary-CPU die Prozessführung übernehmen.

PROFIBUS

Process **Fi**eld **Bus** - europäische Feldbusnorm.

PROFIBUS DP

Ein PROFIBUS mit dem Protokoll DP, der sich konform zur EN 50170 verhält. DP steht für Dezentrale Peripherie (schnell, echtzeitfähig, zyklischer Datenaustausch). Aus Sicht des Anwenderprogramms wird die dezentrale Peripherie genauso angesprochen wie die zentrale Peripherie.

PROFIBUS-Adresse

Eindeutige Kennung eines am PROFIBUS angeschlossenen Teilnehmers. Zur Adressierung eines Teilnehmers wird die PROFIBUS-Adresse im Telegramm übertragen.

PROFIBUS-Gerät

Gerät mit mindestens einer PROFIBUS-Schnittstelle, entweder elektrisch (z. B. RS485) oder optisch (z. B. Polymer Optical Fiber).

PROFIBUS-Nutzerorganisation

Technisches Komitee, das den PROFIBUS- und PROFINET-Standard definiert und weiterentwickelt.

PROFINET

Offenes komponentenbasiertes industrielles Kommunikationssystem auf Ethernet-Basis für verteilte Automatisierungssysteme. Von der PROFIBUS-Nutzerorganisation geförderte Kommunikationstechnologie.

PROFINET IO

IO steht für Input/Output; Dezentrale Peripherie (schnell, echtzeitfähig, zyklischer Datenaustausch). Aus Sicht des Anwenderprogramms wird die dezentrale Peripherie genauso angesprochen wie die zentrale Peripherie.

PROFINET IO als Ethernet-basierter Automatisierungsstandard von PROFIBUS & PROFINET International definiert damit ein herstellerübergreifendes Kommunikations-, Automatisierungs- und Engineering-Modell.

Bei PROFINET IO wird eine Switching-Technologie eingesetzt, die es jedem Teilnehmer ermöglicht, zu jedem Zeitpunkt auf das Netz zuzugreifen. Damit kann das Netz durch gleichzeitige Datenübertragung mehrerer Teilnehmer wesentlich effektiver genutzt werden. Gleichzeitiges Senden und Empfangen wird durch den Vollduplex-Betrieb von Switched-Ethernet ermöglicht.

PROFINET IO basiert auf Switched-Ethernet mit Vollduplex-Betrieb und einer Übertragungsbandbreite von 100 Mbit/s.

PROFINET-Gerät

Gerät, das immer über eine PROFINET-Schnittstelle (elektrisch, optisch, drahtlos) verfügt.

PROFINET-Schnittstelle

Schnittstelle eines kommunikationsfähigen Moduls (z. B. CPU, CP) mit einem oder mehreren Ports. Bereits ab Werk ist der Schnittstelle eine MAC-Adresse zugewiesen. Zusammen mit der IP-Adresse und dem Gerätenamen (aus der individuellen Konfiguration) gewährleistet diese Adresse der Schnittstelle eine eindeutige Identifizierung des PROFINET-Geräts im Netz. Die Schnittstelle kann elektrisch, optisch oder drahtlos sein.

Programmiergerät

Programmiergeräte sind im Kern Personal Computer, die industrietauglich, kompakt und transportabel sind. Sie sind gekennzeichnet durch eine spezielle Hardware- und Software-Ausstattung für speicherprogrammierbare Steuerungen.

Protokoll

Vereinbarung, nach welchen Regeln die Kommunikation zwischen zwei oder mehreren Kommunikationspartnern abläuft.

Prozessabbild (E/A)

In diesen Speicherbereich überträgt die CPU die Werte aus den Ein- und Ausgabemodulen. Am Anfang des zyklischen Programms überträgt die CPU das Prozessabbild der Ausgänge als Signalzustand zu den Ausgabemodulen. Danach liest die CPU die Signalzustände der Eingabemodule in das Prozessabbild der Eingänge ein. Anschließend bearbeitet die CPU das Anwenderprogramm.

PtP

Point-**t**o-**P**oint, Schnittstelle und/oder Übertragungsprotokoll für bidirektionalen Datenaustausch zwischen genau zwei Kommunikationspartnern.

Punkt-zu-Punkt-Kopplung

Bidirektionaler Datenaustausch über Kommunikationsmodule mit serieller Schnittstelle zwischen genau zwei Kommunikationspartnern.

Redundante Systeme

Redundante Systeme sind dadurch gekennzeichnet, dass wichtige Automatisierungskomponenten mehrfach (redundant) vorhanden sind. Bei Ausfall einer redundanten Komponente wird die Kontrolle des Prozesses aufrechterhalten.

Ringtopologie

Alle Teilnehmer eines Netzes sind in einem Ring zusammengeschlossen.

Root-CA-Zertifikate

→ *Siehe auch Stammzertifikat*

Router

Netzwerkknoten mit eindeutiger Kennzeichnung (Namen und Adresse), der Subnetze miteinander verbindet und den Datentransport zu eindeutig gekennzeichneten Kommunikationsteilnehmern im Netz leistet.

RS232, RS422 und RS485

Standard für serielle Schnittstellen.

RTU

Modbus RTU (RTU: **R**emote **T**erminal **U**nit, entfernte Terminaleinheit) überträgt die Daten in binärer Form; ermöglicht einen guten Datendurchsatz. Die Daten müssen in ein lesbares Format umgesetzt werden, bevor sie ausgewertet werden können.

S7-Routing

Kommunikation zwischen S7-Automatisierungssystemen, S7-Anwendungen oder PC-Stationen in verschiedenen S7-Subnetzen über einen oder mehrere Netzwerkknoten, die als S7-Router fungieren.

SDA-Dienst

Send Data with Acknowledge. SDA ist ein elementarer Dienst, mit dem ein Initiator (z. B. DP-Master) eine Nachricht an einen anderen Teilnehmer abschicken kann und dafür unmittelbar eine Empfangsbestätigung erhält.

SDN-Dienst

Send Data with No Acknowledge. Dieser Dienst wird vorwiegend dafür eingesetzt, an mehrere Stationen Daten zu verschicken, der Dienst bleibt deswegen unquittiert. Geeignet für Synchronisationsaufgaben und Zustandsmeldungen.

Security

Oberbegriff für alle Maßnahmen zum Schutz vor

- Verlust der Vertraulichkeit durch unberechtigten Zugriff auf Daten
- Verlust der Integrität durch Manipulation von Daten
- Verlust der Verfügbarkeit durch Zerstörung von Daten

Selbst-signierte Zertifikate

Das sind Zertifikate, die Sie mit Ihrem privaten Schlüssel signieren und als End-Entiy-Zertifikate verwenden.
Die Signatur dieser Zertifikate wird mit Ihrem öffentlichen Schlüssel überprüft.
Die Attribute "Antragsteller" und "Aussteller" von selbst-signierten Zertifikaten müssen identisch sein: Sie haben Ihr Zertifikat ja selbst signiert.
Das Feld "CA" muss auf "False" stehen.
Sie können selbst-signierte Zertifikate zum Beispiel als Applikations-Zertifikat für einen OPC UA-Client verwenden.
Wie Sie ein selbst-signiertes Zertifikat mit dem Zertifikate-Generator der OPC Foundation erzeugen, ist hier (Seite %getreference) beschrieben.

Server

Gerät oder allgemein ein Objekt, das bestimmte Dienste erbringen kann; aufgrund der Anforderung durch einen Client wird der Dienst erbracht.

Slave

Dezentrales Gerät in einem Feldbussystem, das nur nach Aufforderung durch einen Master Daten mit diesem austauschen darf.
→ *Siehe auch DP-Slave*

SNMP

Simple **N**etwork **M**anagement **P**rotocol nutzt das verbindungslose Transportprotokoll UDP. SNMP funktioniert ähnlich dem Client/Server-Modell. Der SNMP Manager überwacht die Netzwerkknoten. Die SNMP Agenten sammeln in den einzelnen Netzwerkknoten verschiedene netzwerkspezifische Informationen und machen diese Informationen in strukturierter Form in der MIB (**M**anagement **I**nformation **B**ase) abrufbar und steuerbar. Mit Hilfe dieser Informationen kann ein Netzwerkmanagementsystem eine ausführliche Netzwerkdiagnose durchführen.

Stammzertifikat

Das ist das Zertifikat einer Zertifizierungsstelle: Sie signiert mit ihrem privaten Schlüssel End-Entity-Zertifikate und Intermediate CA-Zertifikate.
Die Attribute "Antragsteller" und "Aussteller" dieses Zertifikats müssen identisch sein: Diese Zertifizierungsstelle hat ihr Zertifikat selbst signiert.
Das Feld "CA" muss auf "True" stehen.
Das TIA Portal V14 besitzt ein solches Root-CA-Zertifikat:
Wenn Sie im TIA Portal den OPC UA Server einer S7-1500 konfigurieren, dann erzeugt das TIA Portal für den OPC UA-Server ein End-Entity-Zertifikat und signiert dieses Zertifikat mit dem eigenen privaten Schlüssel.
Die Signatur dieses End-Entity-Zertifikats ist überprüfbar mit dem öffentlichen Schlüssel des TIA Portals. Dieser Schlüssel ist im Root-CA-Zertifikat des TIA Portals enthalten.

Subnetz

Teil eines Netzes, dessen Parameter bei den Teilnehmern (z. B. bei PROFINET) abgeglichen sein müssen. Ein Subnetz umfasst die Buskomponenten und alle angeschlossenen Stationen. Subnetze können beispielsweise mittels Gateways oder Routern zu einem Netz gekoppelt werden.

Switch

Netzwerk-Komponente zur Verbindung mehrerer Endgeräte bzw. Netz-Segmente in einem lokalen Netz (LAN).

Systemzustände

Die Systemzustände des redundanten Systems S7-1500R/H resultieren aus den Betriebszuständen der Primary- und Backup-CPU. Der Begriff des Systemzustands wird benutzt, um einen vereinfachten Ausdruck zu erhalten, der die zeitgleich auftretenden Betriebszustände der beiden CPUs kennzeichnet. Beim redundanten System S7-1500R/H gibt es die Systemzustände STOP, ANLAUF, RUN-Solo, SYNCUP und RUN-Redundant.

TCP/IP

Transmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol, verbindungsorientiertes Netzwerk Protokoll, allgemein anerkannter Standard für den Datenaustausch in heterogenen Netzen.

Twisted Pair

Fast Ethernet über Twisted Pair-Leitungen basiert auf dem Standard IEEE 802.3u (100 Base-TX). Übertragungsmedium ist eine 2×2-adrige, verdrillte und geschirmte Leitung mit einem Wellenwiderstand von 100 Ohm (AWG 22). Die Übertragungseigenschaften dieser Leitung müssen die Anforderungen der Kategorie 5 erfüllen.
Die Maximallänge der Verbindung zwischen Endgerät und Netzkomponente darf 100 m nicht überschreiten. Die Anschlüsse erfolgen nach 100 Base-TX-Standard mit dem RJ45-Steckverbindungssystem.

UDP

User-**D**atagram-**P**rotokoll; Kommunikationsprotokoll zur schnellen und unkomplizierten Datenübertragung, ohne Quittierung. Auf Sicherungsmechanismen, wie sie bei TCP/IP vorhanden sind, wird verzichtet.

Uhrzeitsynchronisation

Fähigkeit zur Übertragung einer Standardsystemzeit von einer einzelnen Quelle an alle Geräte im System, so dass deren Uhren entsprechend der Standardzeit eingestellt werden können.

USS

Universelles **s**erielles **S**chnittstellen-Protokoll; definiert ein Zugriffsverfahren nach dem Master-Slave-Prinzip für die Kommunikation über einen seriellen Bus.

Webserver

Software/Kommunikationsdienst zum Datenaustausch über das Internet. Der Webserver überträgt die Dokumente über standardisierte Übertragungsprotokolle (HTTP, HTTPS) an einen Webbrowser. Dokumente können statisch sein oder dynamisch bei Anforderung durch den Webbrowser aus unterschiedlichen Quellen durch den Webserver zusammengesetzt werden.

Index

L

M

N

O

P

R

S

T

U

V

W

X

Z