Programmierleitfaden Safety für SIMATIC S7-1200/1500

SIMATIC Safety Integrated

https://support.industry.siemens.com/cs/ww/de/view/109750255

Siemens Industry Online Support

Rechtliche Hinweise

Nutzung der Anwendungsbeispiele

In den Anwendungsbeispielen wird die Lösung von Automatisierungsaufgaben im Zusammenspiel mehrerer Komponenten in Form von Text, Grafiken und/oder Software-Bausteinen beispielhaft dargestellt. Die Anwendungsbeispiele sind ein kostenloser Service der Siemens AG und/oder einer Tochtergesellschaft der Siemens AG („Siemens“). Sie sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit und Funktionsfähigkeit hinsichtlich Konfiguration und Ausstattung. Die Anwendungsbeispiele stellen keine kundenspezifischen Lösungen dar, sondern bieten lediglich Hilfestellung bei typischen Aufgabenstellungen. Sie sind selbst für den sachgemäßen und sicheren Betrieb der Produkte innerhalb der geltenden Vorschriften verantwortlich und müssen dazu die Funktion des jeweiligen Anwendungsbeispiels überprüfen und auf Ihre Anlage individuell anpassen.
Sie erhalten von Siemens das nicht ausschließliche, nicht unterlizenzierbare und nicht übertragbare Recht, die Anwendungsbeispiele durch fachlich geschultes Personal zu nutzen. Jede Änderung an den Anwendungsbeispielen erfolgt auf Ihre Verantwortung. Die Weitergabe an Dritte oder Vervielfältigung der Anwendungsbeispiele oder von Auszügen daraus ist nur in Kombination mit Ihren eigenen Produkten gestattet. Die Anwendungsbeispiele unterliegen nicht zwingend den üblichen Tests und Qualitätsprüfungen eines kostenpflichtigen Produkts, können Funktions- und Leistungsmängel enthalten und mit Fehlern behaftet sein. Sie sind verpflichtet, die Nutzung so zu gestalten, dass eventuelle Fehlfunktionen nicht zu Sachschäden oder der Verletzung von Personen führen.

Haftungsausschluss

Siemens schließt seine Haftung, gleich aus welchem Rechtsgrund, insbesondere für die Verwendbarkeit, Verfügbarkeit, Vollständigkeit und Mangelfreiheit der Anwendungsbeispiele, sowie dazugehöriger Hinweise, Projektierungs- und Leistungsdaten und dadurch verursachte Schäden aus. Dies gilt nicht, soweit Siemens zwingend haftet, z.B. nach dem Produkthaftungsgesetz, in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der schuldhaften Verletzung des Lebens, des Körpers oder der Gesundheit, bei Nichteinhaltung einer übernommenen Garantie, wegen des arglistigen Verschweigens eines Mangels oder wegen der schuldhaften Verletzung wesentlicher Vertragspflichten. Der Schadensersatzanspruch für die Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegen oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist mit den vorstehenden Regelungen nicht verbunden. Von in diesem Zusammenhang bestehenden oder entstehenden Ansprüchen Dritter stellen Sie Siemens frei, soweit Siemens nicht gesetzlich zwingend haftet.
Durch Nutzung der Anwendungsbeispiele erkennen Sie an, dass Siemens über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden kann.

Weitere Hinweise

Siemens behält sich das Recht vor, Änderungen an den Anwendungsbeispielen jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in den Anwendungsbeispielen und anderen Siemens Publikationen, wie z. B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.
Ergänzend gelten die Siemens Nutzungsbedingungen (https://support.industry.siemens.com).

Securityhinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-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 Security-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 Security finden Sie unter https://www.siemens.com/industrialsecurity.

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 Security RSS Feed unter https://www.siemens.com/cert.

Inhaltsverzeichnis

1 Einleitung

Die Steuerungsgeneration SIMATIC S7-1200 und S7-1500 weist eine zeitgemäße Systemarchitektur auf und bietet zusammen mit dem TIA Portal effiziente Möglichkeiten der Programmierung und Projektierung.

Durch diesen Programmierleitfaden ermöglichen wir ihnen:

- Die Reduzierung von CPU-Stopps
- Kurze Übersetzungszeiten
- Weniger und einfachere Abnahmen

Dieses Dokument gibt Ihnen viele Empfehlungen und Hinweise zur optimalen Projektierung und Programmierung von S7-1200/1500 Steuerungen. Dies hilft Ihnen, eine standardisierte und optimale Programmierung Ihrer Automatisierungslösungen zu erstellen.

Die beschriebenen Beispiele können universell auf den Steuerungen S7-1200 und S7-1500 eingesetzt werden.

Vorteile

Mit der Einhaltung der hier genannten Empfehlungen erzielen Sie viele Vorteile:

- Wiederverwendbarkeit von Programmteilen
- Einfachere Abnahme (Code-Review, Fehlererkennung und -korrektur)
- Höhere Flexibilität bei Programmänderungen
- Reduzierung von Programmierfehlern
- Erhöhte Anlagenverfügbarkeit durch Vermeidung von CPU-Stopps
- Leichtere Lesbarkeit für Dritte
- Verringerte Laufzeit des Sicherheitsprogramms

Hinweis

Nicht alle Empfehlungen dieses Dokuments können gleichzeitig angewandt werden. In diesen Fällen müssen Sie als Anwender entscheiden, welcher Empfehlung Sie eine höhere Priorität geben (z. B. Standardisierung oder Laufzeitoptimierung des Sicherheitsprogramms).

Programmierleitfaden und -styleguide

Bei der Programmierung von Sicherheitsprogrammen gelten grundsätzlich dieselben Empfehlungen wie aus dem Programmierleitfaden und dem Programmierstyleguide.

Programmierleitfaden für SIMATIC S7-1200/1500:

https://support.industry.siemens.com/cs/ww/de/view/90885040

Programmierstyleguide für SIMATIC S7-1200/1500:

https://support.industry.siemens.com/cs/ww/de/view/109478084

Leitfaden zur Bibliothekshandhabung im TIA Portal:

https://support.industry.siemens.com/cs/ww/de/view/109747503

Dieses Dokument dient als Ergänzung zu den genannten Dokumenten und behandelt die Besonderheiten bei der Programmierung von Sicherheitsprogrammen mit STEP 7.

Hinweis

Unabhängig von diesem Dokument sind die Aussagen im Handbuch "SIMATIC Safety - Projektieren und Programmieren" zu beachten - insbesondere darin aufgeführte Warnungen sind unbedingt einzuhalten, da eine Nichtbeachtung bedeutet, dass Tod oder schwere Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Warnhinweiskonzept

Dieses Dokument 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.

2 Fehlersichere Steuerungen projektieren

2.1 Die geeignete F-CPU auswählen

Die Auswahl der F-CPU ist von folgenden Faktoren abhängig:

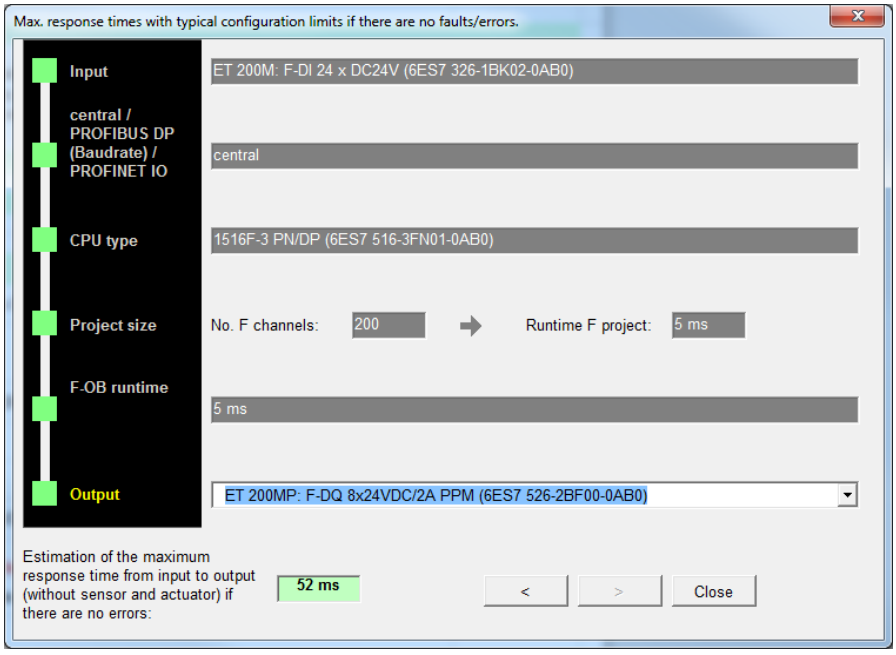
- Laufzeit des Sicherheitsprogramms
- PROFIsafe-Kommunikationszeit
- Reaktionszeit der Sicherheitsfunktion
- Anzahl der benötigten Ein- und Ausgänge
- Anzahl der angebundenen Peripherie
- Speicherbedarf des Programms

Abschätzung der Reaktionszeit

Wenn Sie bereits eine grobe Vorstellung haben, welches Automatisierungssystem Sie einsetzen möchten, können Sie die Reaktionszeit Ihres Sicherheitsprogramms mit der SIMATIC STEP 7 Reaktionszeittabelle abschätzen oder verschiedene Szenarien durchspielen und so die geeignete F-CPU auswählen:

https://support.industry.siemens.com/cs/ww/de/view/93839056

Abbildung 2-1: Reaktionszeit-Assistent der SIMATIC STEP 7 Reaktionszeittabelle



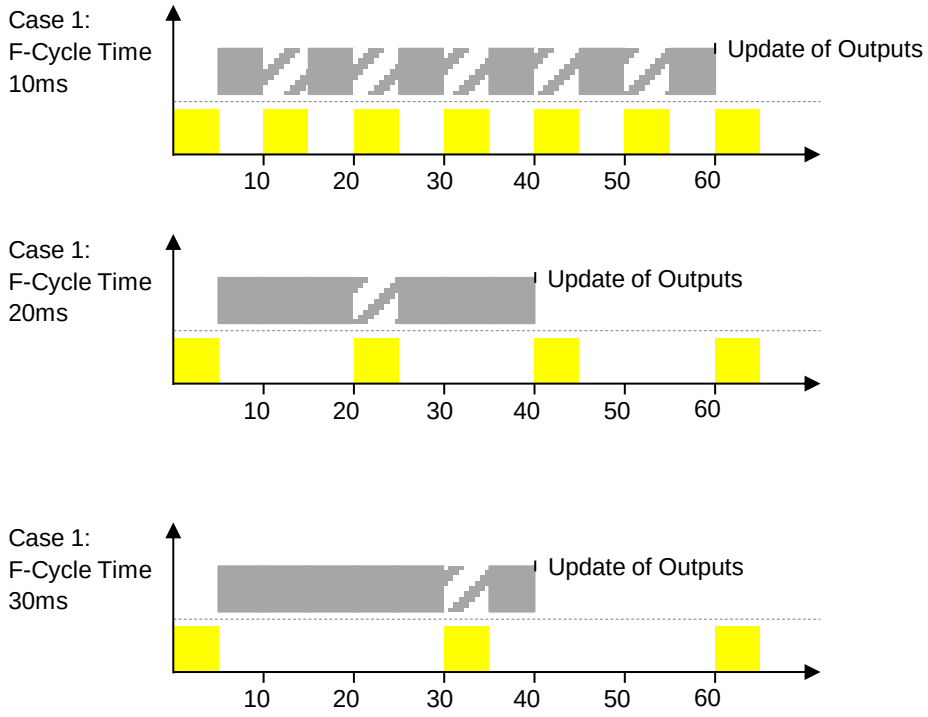
Einfluss der Zykluszeit des Sicherheitsprogramms auf das Standard-Anwenderprogramm

Eine hohe Zykluszeit des Sicherheitsprogramms verlangsamt die Reaktionszeit Ihrer Sicherheitsfunktionen, lässt dafür aber mehr Zeit für die Bearbeitung des Standard-Anwenderprogramms zu.

Eine kurze Zykluszeit des Sicherheitsprogramms verkürzt die Reaktionszeit Ihrer Sicherheitsfunktionen, lässt dafür aber weniger Zeit für die Bearbeitung des Standard-Anwenderprogramms zu.

Die folgende Abbildung zeigt den Einfluss der Zykluszeit des Sicherheitsprogramms der Ereignisklasse „Cyclic interrupt“ auf die Zeit, die für die Bearbeitung des Standard-Anwenderprogramms zur Verfügung steht.

Abbildung 2-2 Einfluss der Zykluszeit des Sicherheitsprogramms auf das Standard-Anwenderprogramm



Duration of standard user program: 30ms
Priority: ≤11

Duration of safety program: 5ms
Priority: 12

Hinweis

Beachten Sie, dass höherpriore Organisationsbausteine (z. B. Weckalarm-OBs oder Motion Control-OBs) auf dieselbe Weise, wie in Abbildung 2-2, auch das Sicherheitsprogramm unterbrechen können.

Um sicherzustellen, dass das Sicherheitsprogramm nicht unterbrochen wird, können Sie die Prioritäten in den Eigenschaften der jeweiligen OBs anpassen.

Hinweis

Falls die Zykluszeit geringer ist als die Bearbeitungsdauer des Sicherheitsprogrammes, wechselt die CPU in den STOP-Zustand.

Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 5.2 – F-Ablaufgruppefestlegen.

2.2 F-Änderungshistorie

Die F-Änderungshistorie verhält sich wie die Änderungshistorie des Standard-Anwenderprogramms. In der Projektnavigation wird unter "Gemeinsame Daten > Protokolle" ("Common data > Logs") für jede F-CPU eine F-Änderungshistorie angelegt.

Empfehlung

Aktivieren Sie die Änderungshistorie zu Beginn der Projektierung oder spätestens nach endgültiger Festlegung des projektspezifischen CPU-Namens, da die Änderungshistorie an den CPU-Namen gekoppelt ist.

Vorteile

- Sicherstellen, dass die letzte Änderung geladen wurde durch Vergleich von Online- und Offline-Stand des CRC (Cyclic Redundancy Check).
- Nachverfolgung in Multiuser-Projekten, welcher Anwender das Sicherheitsprogramm geändert oder geladen hat.
- Abgleich von Online- und Offline-Stand ohne Online-Verbindung zwischen CPU und PG/PC.

ACHTUNG	Die F-Änderungshistorie dürfen Sie nicht für das Erkennen von Änderungen im Sicherheitsprogramm oder in der Projektierung der F-Peripherie bei der Abnahme von Änderungen verwenden.

Hinweis	Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 10.8 – F-Änderungshistorie

2.3 Konsistenter Upload von F-CPUs

Mit TIA Portal V14 SP1 und höher können Sie fehlersichere SIMATIC S7-1500 CPUs konsistent aus dem Automatisierungssystem ins TIA Portal hochladen.

Empfehlung

Ein Upload aus dem Automatisierungssystem ist nur möglich, wenn das Projekt dafür freigegeben ist.

Aktivieren Sie bei Beginn der Projektierung die Option "Konsistenter Upload" in der Safety Administration im TIA Portal.

Vorteile

Ein Programmierer auf der Anlage kann das jeweilige Programm auf sein PG laden und somit den Serviceaufwand reduzieren.

Hinweis	Das Aktivieren der Option zum konsistenten Upload von einer F-CPU verlängert das Laden der sicherheitsrelevanten Projektdaten. Außerdem wird auf der F-CPU mehr Ladespeicher benötigt.

2.4 Know-how-Schutz

Ab STEP 7 Safety V14 können Sie den Know-how-Schutz für fehlersichere Bausteine (FCs und FBs) aktivieren.

Der Know-how-Schutz schützt vor dem Zugriff durch unberechtigte Personen auf bestimmte Programmteile, unabhängig vom Zugriffschutz der F-CPU und des Sicherheitsprogramms. Der Inhalt eines FC oder FB kann ohne Passwort nicht eingesehen oder verändert werden.

Empfehlung

Prüfen Sie während der Projektphase, inwieweit es sinnvoll ist, Bausteine eines Sicherheitsprogramms vor dem Zugriff Dritter zu schützen.

Vorteile

- Schutz Ihres Know-hows über den Inhalt der Programmteile.
- Abgenommene Bausteine können nicht verändert werden.

Weitere Informationen

Die nachfolgende Dokumentation bietet eine Anleitung zum Umgang mit dem Know-how-Schutz für unterschiedliche Szenarien:

Knowhow-Schutz in fehlersicheren Programmen:
https://support.industry.siemens.com/cs/ww/de/view/109742314

3 Methoden für die Safety-Programmierung

3.1 Programmstrukturen

Empfehlung

Achten Sie bei der Programmerstellung darauf, dass ihr Programm wiederverwendbar gestaltet ist. Regeln und Empfehlungen zur Programmierung finden Sie im Dokument Programmierstyleguide für SIMATIC S7-1200 / S7-1500 https://support.industry.siemens.com/cs/ww/de/view/109478084 .

3.1.1 Programmstruktur definieren

Empfehlung

- Teilen Sie den Programmcode modular auf, z.B.
  - in Teilbereiche für Erfassen, Auswerten, Reagieren oder
  - nach Sicherheitsfunktionen oder
  - nach Anlagenteilen
- Erstellen Sie im Vorfeld eine Spezifikation für jedes Modul (basierend auf den Anforderungen der Risikobeurteilung).
- Vermeiden Sie komplexe Signalpfade.

Vorteile

- Komplexität wird minimiert.
- Programmierfehler werden reduziert.
- Erlaubt den Programmcode ohne Ausführung des Programms (z. B. Code-Review oder PLCSIM) zu analysieren/testen.
- Leichtere Erweiterbarkeit und Vereinfachung der erneuten Abnahme.
- Wiederverwendbarkeit von Programmteilen ohne erneute Abnahme.
- Fertige Programmteile können vorab getestet und abgenommen werden.

Beispiel

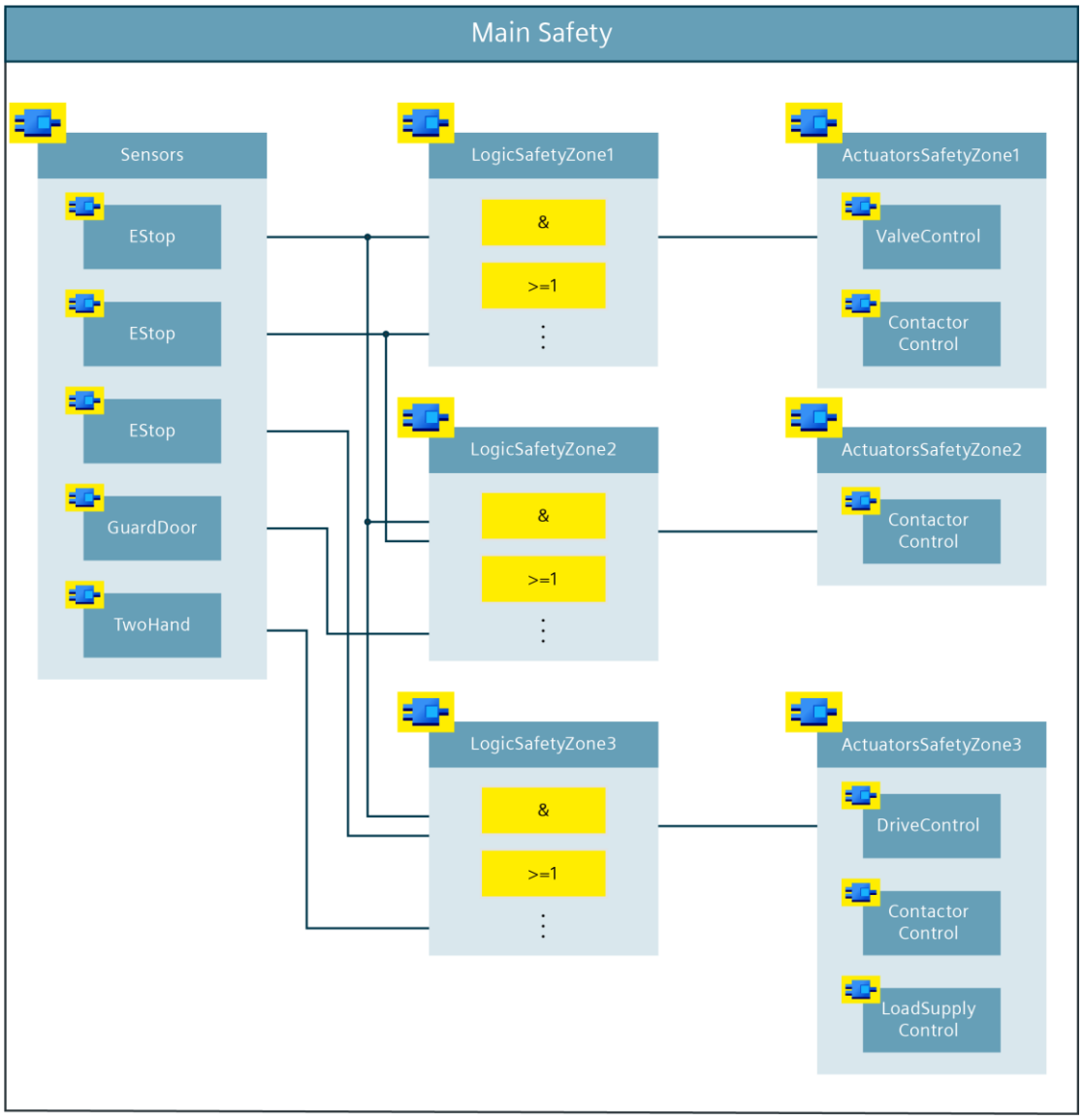
Die folgende Abbildung zeigt eine Sicherheitsapplikation, die in drei Maschinenbereiche (Safety Zones) aufgeteilt ist.

Da die Sensorsignale teilweise bereichsübergreifend verschaltet werden (z. B. global-wirkende Not-Halt-Funktionen), werden sie in einem FB "Sensors" gruppiert (eine Aufteilung in physikalische oder logische Bereiche wäre ebenso möglich). Die Auswertung der jeweiligen Sensoren erfolgt über standardisierte Funktionsbausteine (z. B. "GuardDoor").

Auch die Bausteine der Mobile Panel werden hier aufgerufen.

Für jeden Maschinenbereich werden eigene Logik- und Aktor-FBs erstellt. Die Ansteuerung der Aktoren erfolgt über standardisierte Funktionsbausteine (z. B. "ContactorControl").

Abbildung 3-1: Beispiel einer Programmstruktur



Hinweis

Die hier dargestellte Strukturierung ist beispielhaft. Je nach Größe und Komplexität des Sicherheitsprogramms kann auch eine andere Aufteilung gewählt werden. In kleineren Applikationen wäre es z.B. auch möglich, die Logik und Aktoransteuerung in einem gemeinsamen Funktionsbaustein zu realisieren.

3.1.2 Safety Unit

Software Units bilden eine komfortable Möglichkeit ihr TIA Portal Projekt zu strukturieren. Durch den Einsatz der Safety Unit kann auch das Sicherheitsprogramm Teil dieser Struktur sein.

Folgende Voraussetzungen müssen Sie erfüllen, um eine Safety Unit verwenden zu können:

- Ab TIA Portal V18/ STEP 7 V18
- F-CPU S7-1500 ab FW V2.6
- Unter "Extras/Einstellungen/STEP 7 Safety" ist das Optionskästchen "Verwaltet Sicherheitsprogramm in 'Safety Unit'-Umgebung" angewählt.
- Die F-CPU wird neu angelegt.

Informationen zum Datenaustausch zwischen Safety Unit und Standard Units erhalten sie im Kapitel 3.7.3.

Empfehlung

Wir empfehlen den Einsatz der Safety Unit.

Vorteil

- Größere Übersichtlichkeit
- Zeitersparnis durch Standardisierung
- Einfache Sicherstellung der Datenintegrität

3.1.3 Aufrufebenen von F-FBs/F-FCs

Bei Sicherheitsprogrammen können Sie maximal acht Aufrufebenen verwenden. Ab dieser Grenze erscheint eine Warnung und bei reinen FC- und Multiinstanzaufrufketten eine Fehlermeldung.

Hinweis Funktionen werden auf Systemseite im Absicherungsprogramm als FBs mit Multiinstanzaufruf abgebildet, weshalb auch für FC-Aufrufketten ab acht Aufrufebenen eine Fehlermeldung erscheint.

Die in Abbildung 3-1 dargestellte Programmstruktur zeigt eine Möglichkeit auf, wie die Aufrufebenen flach gehalten werden können, sodass das Sicherheitsprogramm innerhalb der hier spezifizierten Grenzen bleibt.

3.1.4 Aufrufreihenfolge der Bausteine im Main Safety

Empfehlung

Rufen Sie Bausteine innerhalb vom Main Safety in folgender Reihenfolge auf:

1. Empfangs-Bausteine von anderen CPUs (F-CPU-F-CPU-Kommunikation)
2. Fehlerquittierung/Wiedereingliederung von F-Modulen/-Kanälen
3. Auswertebaustein der Sensoren
4. Betriebsartenauswertung
5. Logische Verknüpfungen, Berechnungen, Auswertungen usw.
6. Ansteuerbausteine für sichere Aktoren
7. Sende-Bausteine zu anderen CPUs (F-CPU-F-CPU-Kommunikation)

Vorteile

- CPU arbeitet immer mit den aktuellen Werten
- Erleichtert die Orientierung im Main Safety

Hinweis

Zusätzlich haben Sie mit Vor-/Nachverarbeitung die Möglichkeit, Standardbausteine (FCs) unmittelbar vor bzw. nach einer F-Ablaufgruppe aufzurufen. z. B. für den Datentransfer bei der fehlersicheren Kommunikation über Flexible F-Link.

3.1.5 F-konforme PLC-Datentypen

Auch bei Sicherheitsprogrammen ist es möglich, Daten optimal mit PLC-Datentypen zu strukturieren.

Empfehlung

- Legen Sie F-konforme PLC-Datentypen (F-UDTs) an, um auch im Sicherheitsprogramm Daten zu strukturieren.
- Verwenden Sie F-konforme PLC-Datentypen, um große Anzahlen an Variablen an Bausteine zu übergeben.
- Nutzen Sie die Möglichkeit, F-konforme PLC-Datentypen zu schachteln.

Vorteile

- Eine Änderung in einem PLC-Datentyp wird an allen Verwendungsstellen im Anwenderprogramm automatisch aktualisiert.
- Größere Übersichtlichkeit durch Strukturierung der Daten.

Hinweis

Versuchen Sie die F-konformen PLC-Datentypen möglichst modular aufzubauen, um eine Wiederverwendbarkeit der Datentypen sowie der Bausteine zu erreichen.

Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 5.1.5 – F-konforme PLC-Datentypen

Beispiel

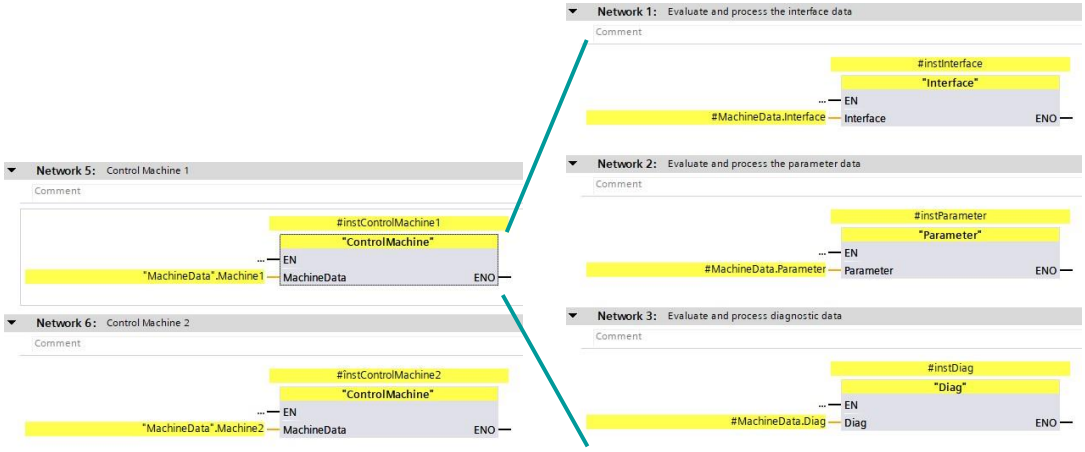
Das im Folgenden dargestellte Beispiel zeigt die Verwendung von F-konformen PLC-Datentypen. Der F-UDT „typeMachine“ (Abbildung 3-2) enthält maschinenbezogene Daten. Durch die Verwendung der weiteren F-UDTs „typeInterface“, „typeParameter“ sowie „typeDiag“ und die Schachtelung werden die Daten strukturiert. Abbildung 3-3 zeigt, wie auf die entsprechenden Daten zugegriffen werden kann.

Abbildung 3-2 Geschachtelter F-konformer PLC-Datentyp

typeMachine

	Name	Data type
1	Interface	"typeInterface"
2	Parameter	"typeParameter"
3	Diag	"typeDiag"
4	<Add new>	

Abbildung 3-3 Verwendung von geschachtelten F-UDTs



3.1.6 Bausteininformationen und Kommentare

Allgemein

In SIMATIC Safety stehen Ihnen die Programmiersprachen Funktionsplan (FUP) und Kontaktplan (KOP) zur Verfügung. Beide Sprachen bieten die Möglichkeit, Block- und Netzwerkkommentare zu hinterlegen.

Kommentare haben keinen Einfluss auf die Signatur von F-FBs/F-FCs und können daher auch nach der Abnahme noch bearbeitet werden.

Empfehlung

Tragen Sie formale Informationen zum Baustein mithilfe der folgenden Schablone in den Blockkommentar Ihres Bausteins ein.

```
  (*Add your company*) / (c)Copyright 2022
-------------------------------------------------------------------------------
Title:
Comment/Function:
Library/Family:
Author:
Tested with:
Engineering:
Restrictions:
Requirements:
DC: |
Category:
-------------------------------------------------------------------------------
Change log table:
Version | Date | Expert in charge | Changes applied
---------|------------|-----------------------|-----------------------------
01.00.00 | 10.10.2022 | *add your Signature here* | *add the experts name*
|First released
```

Wenn Sie in einem F-FB Diagnosefunktionen umsetzen, die für den PL bzw. SILCL eines anderen Teilsystems (Erfassen oder Auswerten) relevant sind, tragen Sie die normativen

Parameter wie PL bzw. SILCL und Kategorie (nach ISO 13849-1), DC-Maßnahmen, CCF-Maßnahmen usw. ebenfalls in den Blockkommentar ein.

Tragen Sie nach erfolgreicher Abnahme des Bausteins die Signatur ebenfalls in den Blockkommentar ein. Dies erleichtert die Nachverfolgbarkeit bei funktionalen Änderungen des Bausteins.

3.2 Funktionale Bezeichner von Variablen

Bei Safety wird sprachlich oftmals von Abschaltungen oder Abschaltsignalen gesprochen. In der Praxis wird auch die Beschreibung einer Sicherheitsfunktion in diesem Wortlaut ausgeführt:

"Wird eine Schutztür geöffnet, muss der Antrieb XY sicher abgeschaltet werden."

Bei der technischen Realisierung als Sicherheitsprogramm werden in der Regel jedoch Freigabesignale programmiert. Dies liegt daran, dass Sicherheitsverschaltungen nach dem Ruhestrom-Prinzip ausgelegt werden.

Wenn zum Beispiel eine Schutztür geschlossen ist, gibt sie die Freigabe, einen sicheren Aktor einzuschalten.

Empfehlung

Legen Sie vor Projektbeginn eine einheitliche Bezeichnung der Variablen mit entsprechenden Suffixen fest. Der Bezeichner gibt den Sinn und Zweck der Variablen im Kontext des Quellcodes wieder.

Wählen Sie den Bezeichner der Variablen so, dass er den logischen "1"-Zustand ("true") widerspiegelt.

Zum Beispiel "mainDoorEnable" oder "conveyorSafetyRelease".

Hinweis Beachten Sie, dass die standardisierten Bezeichnungen der Antriebsfunktionen (z. B. STO und SLS) nach IEC61800-5-2 nicht der obigen Empfehlung entsprechen.

3.3 Bausteine standardisieren

Neben der eigentlichen Auswertung eines Sensors bzw. der Ansteuerung eines Aktors sind häufig dieselben Aufbereitungen von Ein- und Ausgangsparametern notwendig (z. B. Flankenauswertung, Zeitfunktionen, Quittierung usw.).

Hierfür eignet es sich, modulare Bausteine zu erstellen und wiederzuverwenden.

Bausteinbibliotheken

Siemens bietet dafür im Industry Online Support Bausteinbibliotheken an, welche Sie in Ihrem Projekt einsetzen können

- LSafe, TÜV geprüfte Bibliothek für grundlegende Sicherheitsfunktionen.
  https://support.industry.siemens.com/cs/ww/de/view/109793462
- LDrvSafe, bietet fehlersichere Bausteine im Zusammenspiel zwischen CPU, SINUMERIK ONE; SINAMICS via PROFIsafe und SIMATIC Micro-Drive.
  https://support.industry.siemens.com/cs/ww/de/view/109485794

Empfehlung

Erstellen Sie modulare Bausteine, die Sie wiederverwenden können:

- Bausteine für typische fehlersichere Sensoren
- Bausteine für typische fehlersichere Aktoren
- Bausteine für oft genutzte Funktionen (z. B. Reintegration, Betriebsart)

Vorteile

- Wiederverwendete Bausteine müssen nur einmal abgenommen werden
- Schnellere Programmierung weiterer Funktionen und Projekte
- Versionierung mit dem TIA Portal-Bibliothekskonzept möglich
- Standardisierung von Formalparametern über Projekte und Programmierer hinweg und dadurch leichte Lesbarkeit und Prüfbarkeit

Hinweis Nachfolgende Programmierung der Bausteine sind Beispiele. Die tatsächliche Funktion ist abhängig von der Risikobeurteilung der Applikation bzw. den Anforderungen des Projekts.

3.4 Logische Verknüpfungen programmieren

Aufgabe der Bausteine

- Erzeugung von Freigabesignalen zur Ansteuerung der sicherheitsgerichteten Aktoren basierend auf den relevanten Sicherheitsfunktionen
- Verknüpfung der Sensorfreigaben, Betriebsartenfreigaben usw. mit den Ansteuersignalen der Aktoren

Empfehlung

- Verwenden Sie vorrangig UND- und ODER-Logikelemente
- Vermeiden Sie Sprünge in binärer Logik

3.5 Betriebsartenabhängige Sicherheitsfunktionen programmieren

Empfehlung

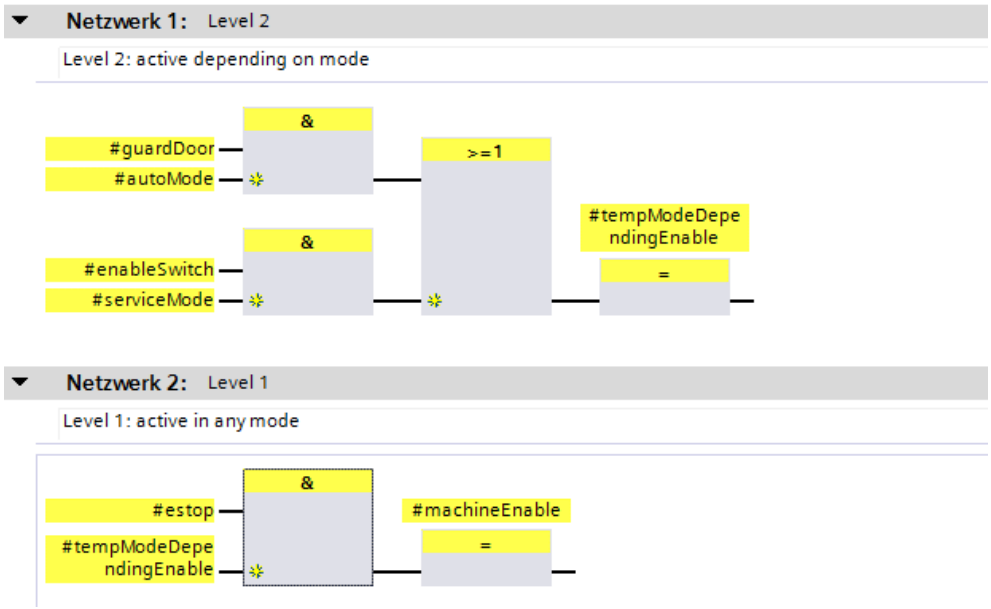
Teilen Sie die Logik in unterschiedliche Level auf:

- Level 1: alle Sicherheitsfunktionen, die von Betriebsarten bzw. Anlagenzuständen unabhängig sind.
  - Logische UND-Verknüpfungen aller Sicherheitsfunktionen, die immer aktiv sind.
  - Typischerweise Not-Halt-Einrichtungen.
- Level 2: alle Sicherheitsfunktionen, die betriebsartenabhängig sind.
  - Logische ODER-Verknüpfung der Sicherheitsfunktionen, die nur in bestimmten Betriebsarten wirken.
  - z.B. Schutztüren im Automatikbetrieb, oder Zustimmtaster im Servicebetrieb.

Beispiel

An einer Maschine sind drei Sicherheitsfunktionen realisiert, wobei die Not-Halt-Funktion "estop" in jeder Betriebsart wirkt und die Schutztürüberwachung "guardDoor" und die Zustimmfunktion "enablingSwitch" jeweils nur in einer Betriebsart wirken.

Abbildung 3-4: Betriebsarten programmieren



3.6 Anbindung von Global-Daten

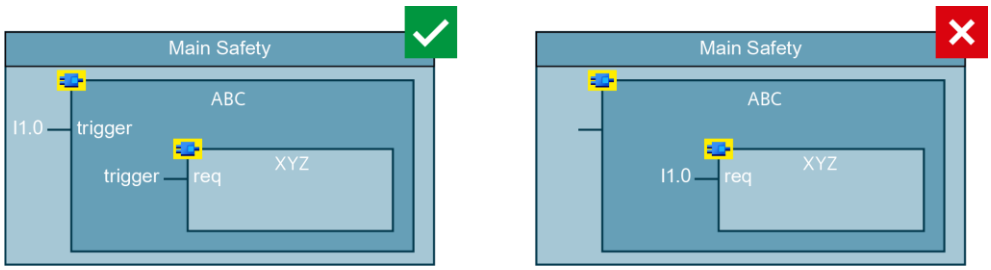
Empfehlung

- Verbinden Sie Global-Daten (Eingänge, Ausgänge, Datenbausteine) in der höchsten Ebene der Baustein-Hierarchie (Main Safety).
- Verwenden Sie die Bausteinschnittstellen, um Signale an unterlagerte Ebenen weiterzugeben.

Vorteile

- Modulares Bausteinkonzept
- Programmteile können ohne Anpassungen in anderen Projekten wiederverwendet werden
- Programmierfehler werden reduziert
- Das Gesamtprogramm wird leichter lesbar, da die generelle Funktion eines Bausteins bereits anhand der Schnittstellen abgeschätzt werden kann.

Abbildung 3-5: Anbindung von Global-Daten



3.7 Datenaustausch zwischen Standard- und Sicherheitsprogramm

Prinzipiell hat das Sicherheitsprogramm die Aufgabe alle Funktionen auszuführen, die eine risikomindernde Maßnahme darstellen. Alle anderen betrieblichen Funktionen, wie auch Funktionen zur Bedienung und Wartung, gehören in das Standard-Anwenderprogramm.

Da in der Praxis auch im Sicherheitsprogramm Informationen für das Diagnose- und Meldekonzept anfallen und auch betriebliche Informationen für das Sicherheitsprogramm relevant sind, können beide Programmteile nicht komplett getrennt werden.

Um nicht-sicherheitsrelevante Funktionen ins Standard-Anwenderprogramm auszulagern, wird eine klar definierte Schnittstelle empfohlen.

Hinweis Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 8.1 – Datentransfer vom Sicherheits- zum Standard-Anwenderprogramm

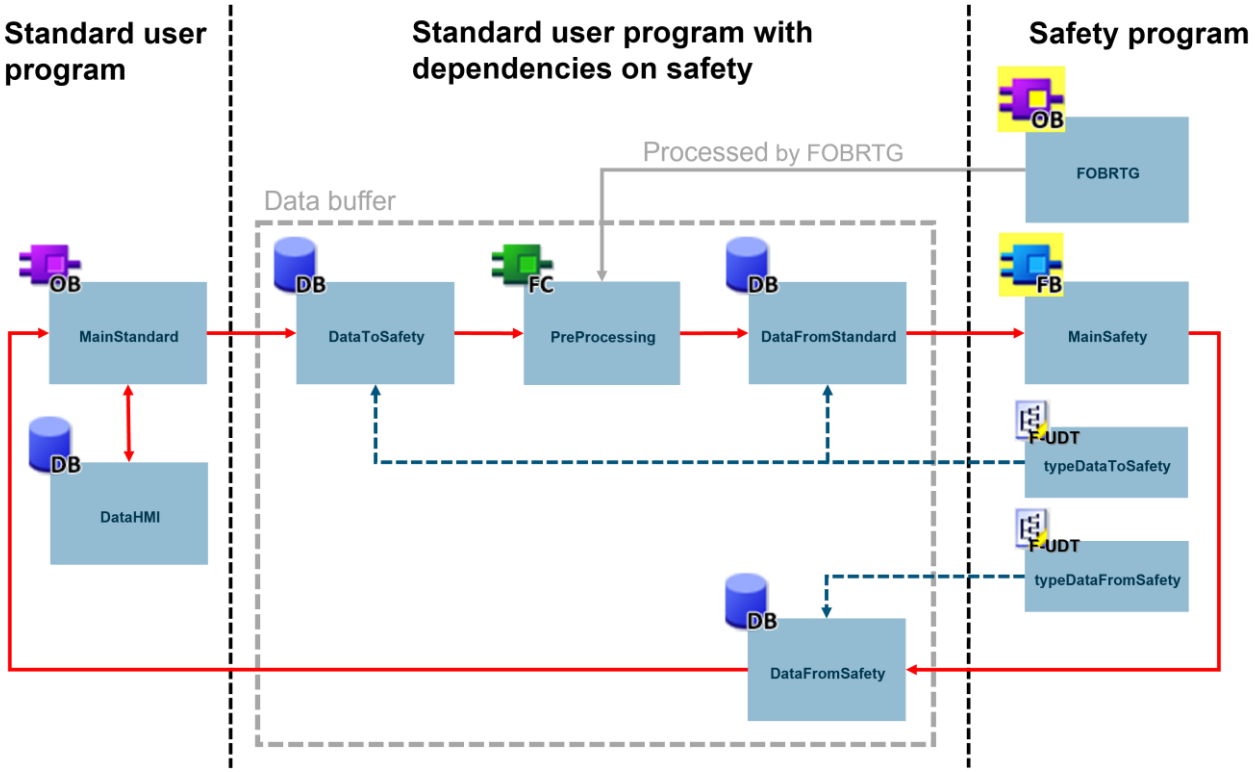
Hinweis Bei jedem Zugriff vom Sicherheitsprogramm auf Standard-Variablen erfolgt vom Compiler jeweils eine separate Codierung. Aus Performance-Gründen kann es daher sinnvoll sein, einem leicht veränderten Verfahren zu folgen.

Mehr Informationen finden Sie im Kapitel 4.1.4.

3.7.1 Grundlegende Übertragung

Nutzen Sie zur Übertragung drei Standard-Datenbausteine (auch Koppel-DBs), um Daten zwischen Anwenderprogram und Sicherheitsprogramm auszutauschen. Trennen Sie die Informationsübertragung in einen ersten Baustein, welcher Informationen aus dem Standardprogramm in das Sicherheitsprogramm überträgt, einen zweiten, in den die Daten hineinkopiert werden können und einen dritten, welcher Daten in die entgegengesetzte Richtung transferiert. Eine Übersicht erhalten sie in Abbildung 3-6.

Abbildung 3-6 Prozessübersicht Datenaustausch zwischen Standard und Safety



F-UDT

Verwenden Sie Datenbausteine vom Typ eines F-UDT. Der Datenbaustein enthält genau die im F-UDT definierten Variablen. Somit wird gewährleistet, dass Änderungen an den Schnittstellen nur durch Benutzer mit Safety-Passwort durchgeführt werden können.

Hinweis

Durch Änderungen an Standardbausteinen, auf die vom Sicherheitsprogramm lesend oder schreibend zugegriffen wird verliert das F-Programm seine Konsistenz. Ein erneutes Übersetzen des Sicherheitsprogramms ist notwendig, ein Download auf die CPU ist nur über den Systemzustand STOP möglich.

Bei Änderungen solcher Standardbausteine werden Sie seit TIA Portal V16 zur Eingabe des F-Passworts aufgefordert. Mit Hilfe von F-konformen PLC-Datentypen (F-UDTs) können Sie die Schnittstelle vor Änderungen ohne Safety Passwort schützen.

Vorverarbeitung

Kopieren Sie die Daten aus dem Standardprogramm in der Vorverarbeitung der F-Ablaufgruppe in einen weiteren Datenbaustein. Dadurch kann sichergestellt werden, dass während der Verarbeitung des Sicherheitsprogrammes keine Daten geändert werden.

Hinweis

Änderungen sicherheitsgerichteter Daten während der Ausführung der Ablaufgruppe führen zum CPU-Stopp.

Datenverarbeitung

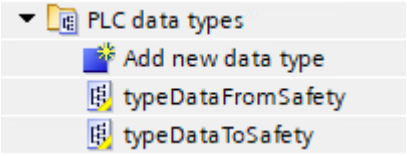
Achten Sie darauf, jeweilige Weiterverarbeitung der Daten, beispielsweise für ein Diagnose- und Meldekonzept, innerhalb des Standardprogrammes durchzuführen, um das Sicherheitsprogramm kompakt zu halten.

Ablaufgruppen

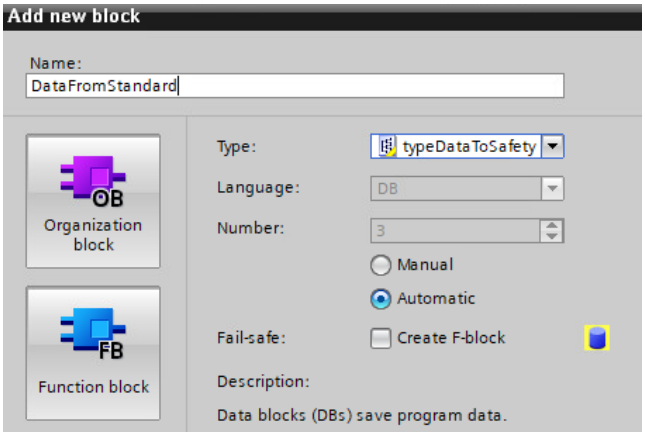
Verwenden Sie für jede Ablaufgruppe eigene Koppel-DBs.

3.7.2 Schrittanleitung

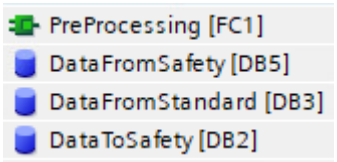
1. Legen Sie jeweils einen F-UDT für die Übertragung in die jeweilige Richtung an und füllen Sie ihn mit den jeweils zu übertragenden Daten.



2. Erstellen sie drei Datenbausteine, zwei vom Typ „typeDataToSafety“ und einen vom Typ „typeDataFromSafety“.

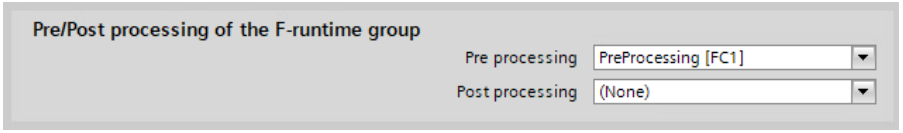


3. Erstellen Sie zusätzlich einen FC „PreProcessing“. Nutzen Sie diesen, um die Daten vom „DataToSafety“ auf den Datenbaustein „DataFromStandard“ zu kopieren.




```
1  "DataFromStandard" := "DataToSafety";
```

4. Öffnen sie die Safety Administration, wählen sie die korrekte Ablaufgruppe aus und rufen Sie den FB „PreProcessing“ in der Vorverarbeitung der jeweiligen Ablaufgruppe auf:

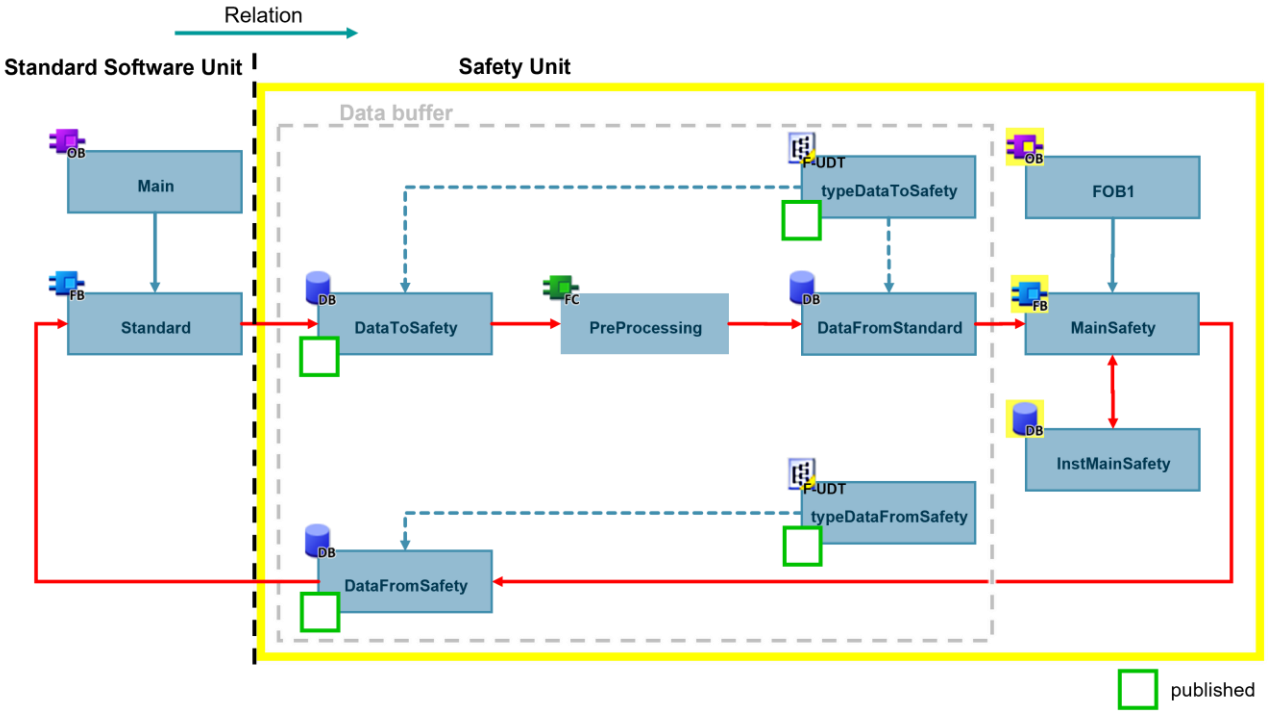


5. Nun können Sie Daten, welche Sie im Standardprogramm benötigen auf den „DataFromStandard“ schreiben und im Sicherheitsprogramm von dem „DataToSafety“ lesen. Ebenso können Sie Daten aus dem Sicherheitsprogramm vom „DataFromSafety“ lesen.

3.7.3 Datenaustausch mit Units

Der Datenaustausch mit Units funktioniert ähnlich dem Austausch zwischen Standard und Safety ohne Units.

Abbildung 3-7 Prozessübersicht Datenaustausch zwischen Standard und Safety mit Units



Vorgehen

1. Legen Sie zwei F-UDTs („typeDataToSafety“ und „typeDataFromSafety“) an. Legen Sie anschließend drei Standard-Datenbausteine, vom Typ der entsprechenden F-UDTs, in der Safety Unit an („DataToSafety“, „DataFromStandard“ und „DataFromSafety“).
2. Veröffentlichen Sie die F-UDTs und die Datenbausteine in der Safety Unit.
3. Erstellen Sie eine Relation der Standard Software Units auf die Safety Unit. Veröffentlichen Sie die notwendigen Datenbausteine und F-UDTs, diese sind in Abbildung 3-7, wie auch im TIA-Portal mit grünem Kästchen gekennzeichnet.

3.7.4 HMI-Signale ans Sicherheitsprogramm übergeben

Die Datenübertragung vom HMI zum Sicherheitsprogramm ist über das Standardprogramm zu realisieren. Kopieren sie die Daten aus dem HMI im Standardprogramm in den Koppel-DB (Abbildung 3-6).

Signale sicher übertragen

Die Kommunikation zwischen HMI und CPU ist nicht sicher. Um sicherheitsgerichtete Daten zu übertragen, sind Maßnahmen notwendig, die die sichere Übertragung gewährleisten. Dieses Anwendungsbeispiel zeigt Ihnen ein geeignetes Sicherheitskonzept.

Fehlersicheres Übertragen sicherheitsgerichteter Kennwerte über Webserver/HMI:

https://support.industry.siemens.com/cs/ww/de/view/109780314

Sicherheitsfunktionen rückstellen

Für das Rückstellen von Sicherheitsfunktionen oder das Quittieren von Fehlern über ein HMI stellt TIA Portal den Systembaustein "ACK_OP" bereit.

Eine Quittierung besteht aus zwei Schritten:

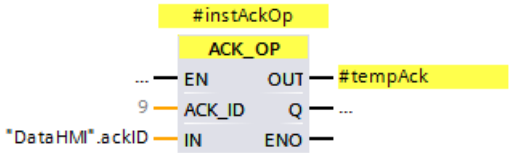
1. Wechsel des Durchgangs IN für genau einen Zyklus auf den Wert "6".
2. Wechsel des Durchgangs IN für genau einen Zyklus auf den Wert am Eingang "ACK_ID" innerhalb einer Minute.

Dieser Systembaustein stellt eine Ausnahme zum empfohlenen Datenaustausch dar.

Der Systembaustein setzt in jedem Zyklus den InOut-Parameter "IN" auf "0" zurück. Werden die Daten vom HMI im Standard-Anwenderprogramm umkopiert, wird die "0" in jedem Zyklus mit dem Wert aus dem HMI überschrieben und die Bedingung, dass die Werte für genau einen Zyklus anstehen, ist nicht erfüllt.

Beschreiben Sie die Variable am Eingang "IN" daher direkt vom HMI aus und setzen Sie die Priorität des Sicherheitsprogramms höher als die der Kommunikation, um eine mögliche Datenverfälschung zu vermeiden.

Abbildung 3-8: Systembaustein "ACK_OP"



3.7.5 Nicht-sicheren Eingänge im Sicherheitsprogramm verwenden

Empfehlung

Das Einlesen von Standard-Eingängen, die direkt im Sicherheitsprogramm benötigt werden, sollen direkt im Sicherheitsprogramm gelesen werden. Ein "Umweg" über das Standard-Anwenderprogramm ist zu vermeiden.

Hintergrund dafür ist, dass auch nicht-sicherheitsgerichtete Signale in die systematische Integrität der Applikation eingehen. Typische Beispiel dafür sind Quittier-/Rückstelltaster oder Betriebsartenwahlschalter. Welcher Taster welche Sicherheitsfunktion rückstellen darf, ist ein direktes Ergebnis der Risikobewertung. Daher muss eine Änderung der Befehlsgeräte einen Einfluss auf die Signatur haben und darf nur in Begleitung einer Neubeurteilung und Änderungsabnahme einhergehen. Außerdem wird nur so eine mögliche Datenverfälschung im Standardsignal aufgedeckt.

ACHTUNG	Die Bewertung, welche Signale Einfluss auf die systematische Integrität einer Applikation haben und abhängig davon in Standard-Anwenderprogramm oder Sicherheitsprogramm ausgewertet werden, ist abhängig von der Risikobeurteilung einer Applikation.

Empfehlung

Unter bestimmten Umständen kann es abweichend zu vorheriger Empfehlung sinnvoll sein, das Einlesen von Standard-Eingängen, die im Sicherheitsprogramm benötigt werden, im Standard-Programm einzulesen und über einen Standard-Datenbaustein (wie in Kap. 3.7.1 beschrieben) an das Sicherheitsprogramm weiterzugeben. Hierdurch wird eine höhere Unabhängigkeit zwischen Hard- und Software erzielt. Dies ist insbesondere im Umfeld von Serienmaschinen und modularen Maschinenkonzepten gefordert.

Vorteil

- Bessere Modularisierung und Wiederverwendbarkeit
- Entkopplung von Hard- und Software

 **WARNUNG**	**Im Sicherheitsprogramm dürfen grundsätzlich nur fehlersichere Daten oder fehlersichere Signale von F-Peripherie und anderen Sicherheits-programmen (in anderen F-CPUs) verarbeitet werden, da alle Variablen aus dem Standard nicht abgesichert sind.** **Durch die Entkopplung von Hard- und Software können Fehler in der Verschaltung nicht durch Änderungen an der Signatur aufgedeckt werden.** **Im Weiteren gelten die Angaben in den entsprechenden Handbüchern.**

Hinweis	Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 8.2 – Datentransfer vom Standard-Anwenderprogramm zum Sicherheitsprogramm

3.8 F-Signaturen

Die F-Signaturen dienen zur eindeutigen Identifikation der sicherheitsgerichteten Programminformationen. Sie werden im Safety-Administration-Editor (SAE) angezeigt und sind Bestandteil des Sicherheitsausdrucks. Bei Änderung der Signatur muss das Sicherheitsprogramm erneut validiert und abgenommen werden.

Folgende Signaturen werden im Sicherheitsteil bereitgestellt:

Tabelle 3-1: Übersicht der F-Signaturen

Bezeichnung	Bedeutung
F-Gesamtsignatur	Kennzeichnet einen eindeutigen Stand der fehlersicheren Projektdaten
F-HW-Gesamtsignatur	Bei Änderungen an fehlersicherer HW-Konfiguration
F-SW-Gesamtsignatur	Bei Änderungen am Sicherheitsprogramm
F-Kommunikations-Adress-Signatur	Bei Änderungen am Namen oder der F-Kommunikations-UUID von Kommunikationsverbindungen mit Flexible F-Link

Empfehlung

Im Gegensatz zur F-Änderungshistorie können sie durch die F-Signaturen Änderungen im Sicherheitsprogramm sicher erkennen. Nutzen Sie die Signaturen, um Bausteinstände sicher zu dokumentieren.

Vorteil

- Eindeutige Identifikation von fehlersicherer Hardware, Software und Kommunikation
- Sichere Nachverfolgbarkeit möglicher Änderungen und eindeutige Dokumentation
- Nach einem Hardwaretausch kann über die F-SW-Gesamtsignatur einfach nachgewiesen werden, dass die Software unverändert ist

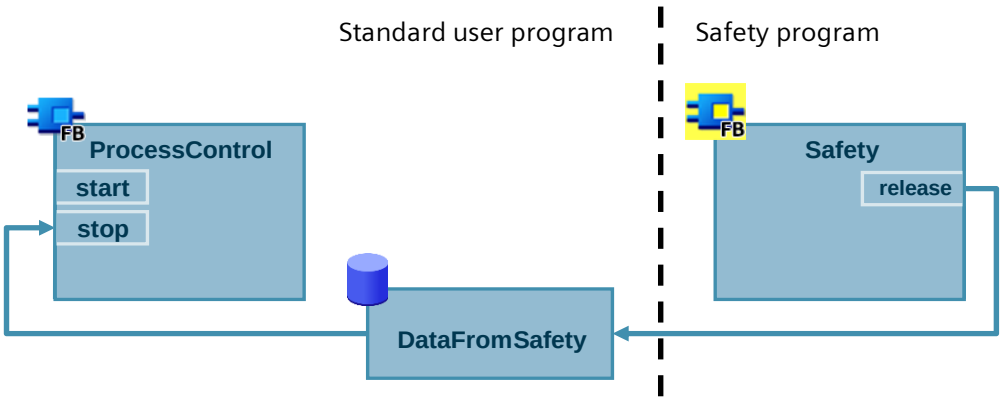
3.9 Betriebsmäßiges Schalten zurücksetzen

Sichere Aktoren werden oft auch für betriebsmäßiges Schalten verwendet. Die einschlägigen Sicherheitsnormen fordern, dass ein Rückstellen der Sicherheitsfunktion keinen Wiederanlauf der Maschine auslöst. Beim Auslösen der Sicherheitsfunktion muss daher das betriebsmäßige Schalten zurückgesetzt und ein erneutes Einschaltsignal erforderlich werden.

Empfehlung

- Verriegeln Sie die Prozesssteuerung im Standard-Anwenderprogramm mit dem Freigabesignal aus dem Sicherheitsprogramm. Eine sichere Abschaltung setzt dadurch auch die Prozesssteuerung zurück.
- Übergeben Sie das Freigabesignal aus dem Sicherheitsprogramm über einen globalen Datenbaustein (siehe auch Kapitel 3.7).

Abbildung 3-9: Prozesssteuerung mit dem Freigabesignal verriegeln



3.10 Wiedereingliederung von fehlersicheren Peripheriemodulen/-kanälen

Erkennt die F-CPU einen sicherheitsrelevanten Fehler, passiviert sie den betroffenen fehlersicheren Kanal bzw. das gesamte Modul. Nachdem der Fehler behoben wurde, muss der passivierte Kanal wiedereingegliedert (depassiviert) werden.

Solange ein Kanal passiviert ist, arbeitet er mit Ersatzwerten. Ein Eingang liefert den Ersatzwert "0" an das Prozessabbild. Ein Ausgang wird mit dem Ersatzwert "0" beschaltet, unabhängig davon, ob das Programm den Ausgang ansteuert oder nicht.

Hinweis

Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 6.5 – Passivierung und Wiedereingliederung der F-Peripherie

3.10.1 Passivierte Module/Kanäle auswerten

Allgemein

Ob ein Kanal passiviert ist, können Sie folgendermaßen auswerten:

- Wertstatus des Kanals ist "false"
- Variable "QBAD" des F-Peripherie-Datenbausteins des Moduls ist "true"
- LEDs des Kanals und des Moduls leuchten rot
- Eintrag im Diagnosepuffer

Die Wiedereingliederung kann entweder manuell oder automatisch erfolgen. Abhängig von der Risikobeurteilung legen Sie das Quittierverhalten fest.

Nachdem ein Fehler behoben wurde, wird Ihnen die Quittierbereitschaft folgendermaßen angezeigt:

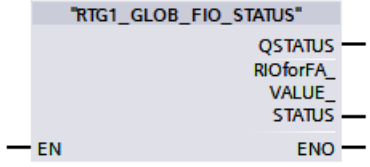
- Variable "ACK_REQ" des F-Peripherie-Datenbausteins des Moduls ist "true"
- LEDs des Kanals und des Moduls blinken abwechselnd rot und grün

Status der F-Peripherien/-Kanäle global auswerten

Ab STEP 7 V14 SP1 können Sie einen Baustein vom System generieren lassen, um den Status aller F-Peripherien/-Kanäle einer F-Ablaufgruppe global auszuwerten.

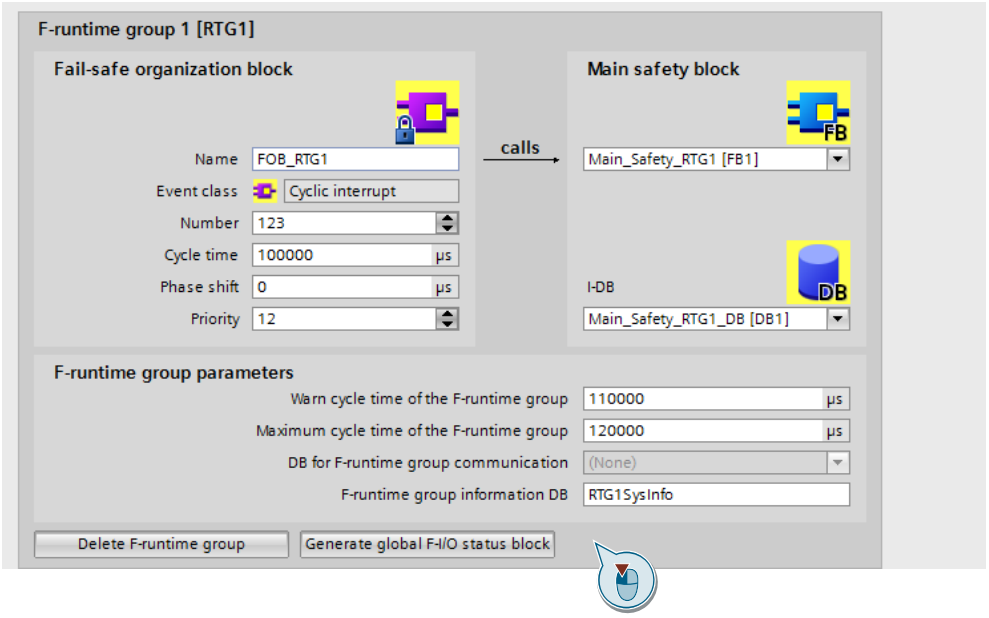
Dieser Baustein wertet aus, ob für mindestens eine F-Peripherie oder mindestens einen Kanal einer F-Peripherie einer F-Ablaufgruppe statt der Prozesswerte Ersatzwerte ausgegeben werden. Das Ergebnis der Auswertung steht am Ausgang "QSTATUS" an. Dabei bleiben F-Peripherien, die Sie mit der Variable DISABLE im F-Peripherie-DB deaktiviert haben, unberücksichtigt.

Abbildung 3-10: Systemgenerierter Baustein zur globalen Auswertung der F-Peripherien



Den Baustein generieren Sie in der Safety Administration in den Einstellungen der jeweiligen F-Ablaufgruppe.

Abbildung 3-11: Baustein zur globalen Auswertung der F-Peripherien generieren



Hinweis

Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 3.3.1 – Bereich "F-Ablaufgruppe"

3.10.2 Automatische Wiedereingliederung

Abhängig davon, ob das jeweilige Modul den Standard "RIOforFA" (siehe Kapitel 5) unterstützt, können Sie die automatische Wiedereingliederung auf unterschiedliche Weise realisieren.

WARNUNG

Automatische Wiedereingliederung kann eine gefährliche Situation einleiten

Ob eine automatische Wiedereingliederung für betreffenden Prozess sicherheitstechnisch zulässig ist, ist abhängig von der Risikobeurteilung.

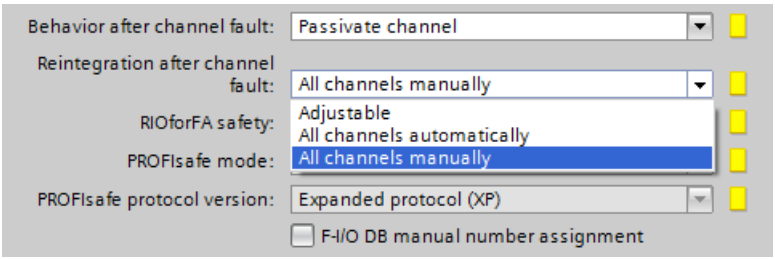
Hinweis

Die automatische Wiedereingliederung bezieht sich auf F-Peripherie-/Kanalfehler (z. B. Diskrepanzfehler, Kurzschluss). Kommunikationsfehler müssen weiterhin manuell quittiert werden (siehe Kapitel 3.10.3).

Module, die "RIOforFA" unterstützen

Bei Modulen, die "RIOforFA" unterstützen, können Sie eine automatische Wiedereingliederung entweder für das gesamte Modul oder auch nur für einzelne Kanäle parametrieren.

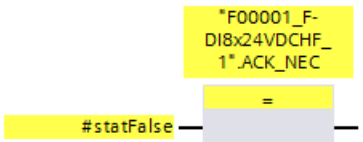
Abbildung 3-12: Automatische Wiedereingliederung parametrieren



Module, die "RIOforFA" nicht unterstützen

Bei Modulen, die "RIOforFA" nicht unterstützen, programmieren Sie die automatische Wiedereingliederung im Sicherheitsprogramm. Setzen Sie dazu die Variable "ACK_NEC" des jeweiligen F-Peripherie-Datenbausteins auf "false":

Abbildung 3-13: Automatische Wiedereingliederung programmieren

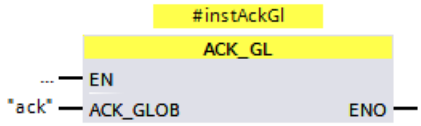


3.10.3 Manuelle Wiedereingliederung

Globale Reintegration aller passivierten F-Module

Um alle passivierten F-Module bzw. -Kanäle einer F-Ablaufgruppe wiedereinzugliedern, verwenden Sie die Anweisung "ACK_GL":

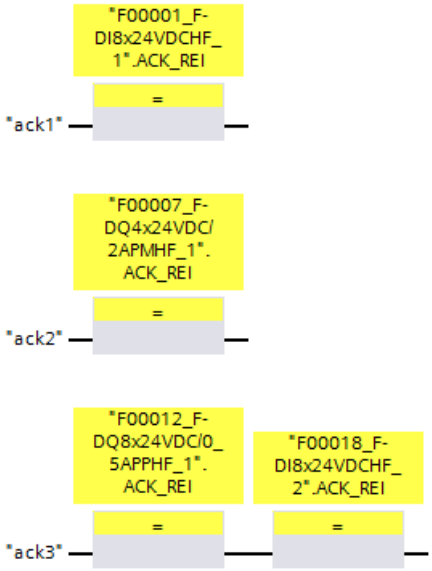
Abbildung 3-14: Anweisung "ACK_GL"



Separate Wiedereingliederung von Modulen (oder einer Gruppe von Modulen)

Bei verteilten Anlagen kann es erforderlich sein, dass nur lokal wiedereingegliedert werden darf (z. B. separate Befehlsgeräte am Schaltschrank). Verschalten Sie dazu die Variablen "ACK_REI" der jeweiligen F-Peripherie-Datenbausteine:

Abbildung 3-15: Separate Wiedereingliederung von Modulen



4 Sicherheitsprogramme optimieren

4.1 Übersetzungsdauer und Laufzeit optimieren

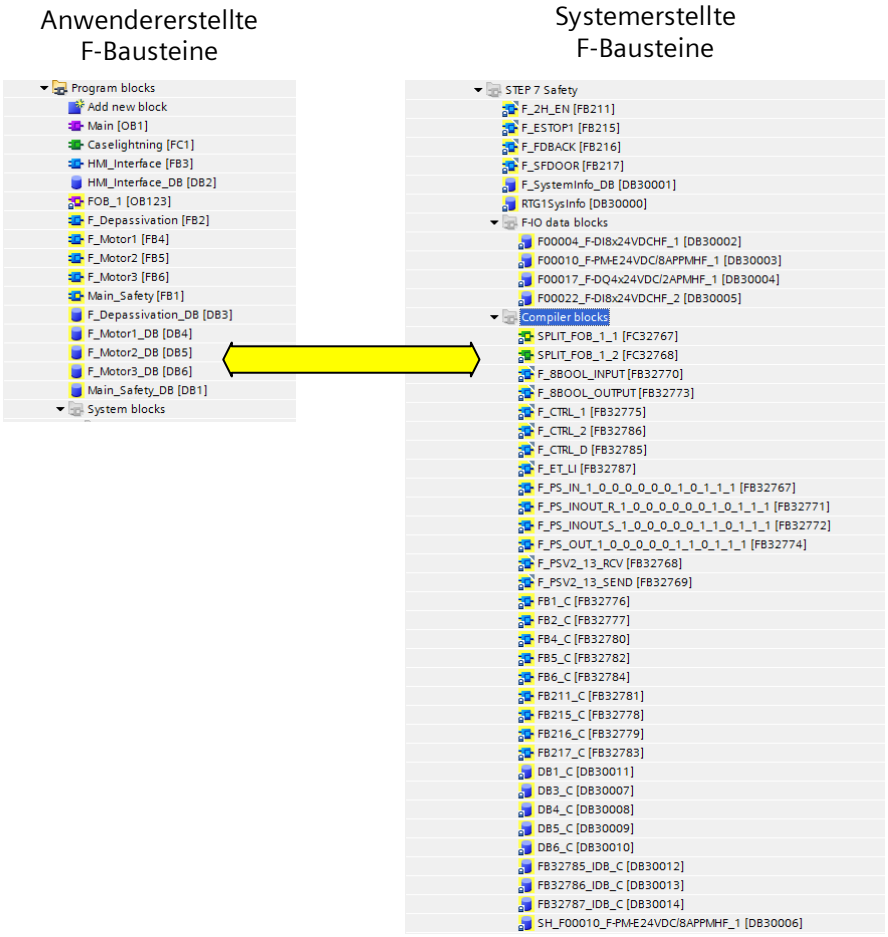
Einleitung

Ein wichtiger Bestandteil eines Sicherheitsprogramms ist die Absicherung der Anwenderprogrammierung durch das Coded Processing (siehe Kapitel 5). Ziel ist es, jegliche Verfälschung im Sicherheitsprogramm aufzudecken und damit unsichere Zustände zu verhindern.

Dieses Absicherungsprogramm wird während der Übersetzung erzeugt und verlängert so die Übersetzungsdauer. Auch die Laufzeit der F-CPU wird durch das Absicherungsprogramm verlängert, da die F-CPU dieses zusätzlich bearbeitet und die Ergebnisse mit dem Anwenderprogramm vergleicht.

Das Absicherungsprogramm, das automatisch vom System generiert wird, finden Sie im Systembausteinordner Ihrer F-CPU.

Abbildung 4-1: Absicherungsprogramm



Dabei haben manche Anweisungen, die im Sicherheitsprogramm verwendet werden können, stärkeren Einfluss auf die Performance einer fehlersicheren Steuerung als andere.

In diesem Kapitel werden Ihnen verschiedene Möglichkeiten zur Verkürzung der Übersetzungs- und Programmlaufzeit aufgezeigt.

Hinweis

Es ist je nach Anwendung nicht immer möglich, alle Vorschläge zu nutzen. Sie geben aber Aufschluss, warum bestimme Programmiermethoden kürzere Übersetzungs- und Programmlaufzeiten als ein nicht-optimiertes Programm verursachen.

Laufzeit ermitteln

TIA Portal erstellt für jede F-Ablaufgruppe automatisch einen Datenbaustein "RTGxSysInfo", der unter anderem die aktuelle sowie die längste Laufzeit dieser F-Ablaufgruppe enthält.

Diesen systemgenerierten Baustein finden Sie in der Projektnavigation unter ("Program blocks > System blocks > STEP 7 Safety").

Abbildung 4-2: Systemgenerierter DB "RTGxSysInfo"

RTG1SysInfo

	Name	Data type	Start value	Monitor value
1	Input			
2	Output			
3	MODE	Bool	false	FALSE
4	F_SYSINFO	F_SYSINFO		
5	MODE	Bool	false	FALSE
6	TCYC_CURR	DInt	0	100
7	TCYC_LONG	DInt	0	101
8	TRTG_CURR	DInt	0	0
9	TRTG_LONG	DInt	0	2
10	T1RTG_CURR	DInt	0	0
11	T1RTG_LONG	DInt	0	0
12	F_PROG_SIG	DWord	DW#16#103E2...	16#103E_261A
13	F_PROG_DAT	DTL	DTL#2017-9-1...	DTL#2017-09-19-1...
14	F_RTG_SIG	DWord	DW#16#8A587...	16#8A58_7EBD
15	F_RTG_DAT	DTL	DTL#2017-9-1...	DTL#2017-09-19-1...
16	VERS_S7SAF	DWord	DW#16#14000...	16#1400_0100
17	InOut			
18	Static			

4.1.1 Sprünge im Sicherheitsprogramm

In einem Standard-Anwenderprogramm ist ein Sprung von einem Netzwerk in ein anderes (Jump auf Label) oder aus dem Baustein heraus (Return) eine einfache Programmverzweigung, die für jeden Zyklus neu berechnet, aber nicht extra abgesichert wird. Es wird also nicht geprüft, ob z. B. durch einen durch EMV erzeugten Speicherfehler ein Sprung trotz Bedingung "false" springt oder nicht.

In einem fehlersicheren Programm ist das nicht zulässig, da zu jeder Zeit garantiert werden muss, dass sich das Programm im korrekten Ablaufzweig befindet.

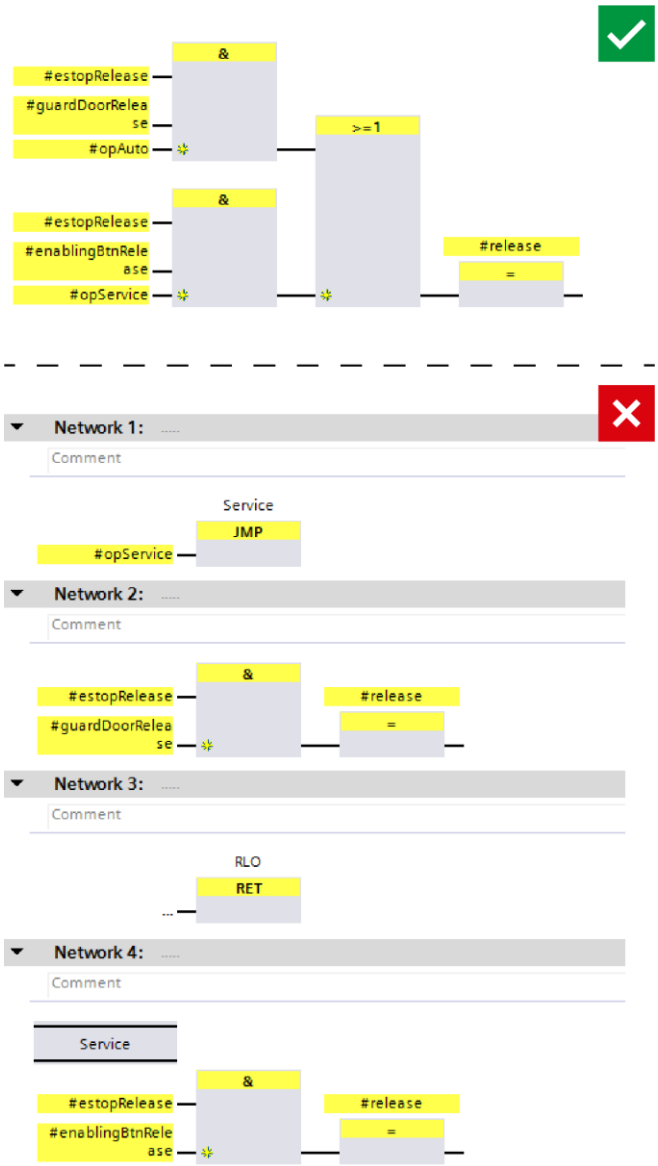
Dafür müssen im Absicherungsprogramm beide Alternativen (Jump auf Label ist "true" oder "false") vollständig berechnet werden.

Je mehr Sprünge Sie in einem Sicherheitsprogramm verwenden, desto stärker wird der Einfluss auf die Leistung der Steuerung.

Empfehlung

- Vermeiden Sie Sprünge im Sicherheitsprogramm.
- Verwenden Sie Zustandsautomaten statt Sprüngen in FBs mit binärer Logik.

Abbildung 4-3: Sprünge vermeiden



4.1.2 Timer-Bausteine

Timer sind für ein Sicherheitsprogramm ein zentraler Bestandteil, da auch viele der Systemfunktionen wie "ESTOP1" intern diese Timer verwenden. Trotzdem ist der Aufwand zur Erzeugung eines fehlersicheren Zeitwerts äußerst umfangreich und muss für jeden einzelnen Timer-Baustein erneut generiert werden.

Hinweis Beachten Sie dazu auch die Angaben im Handbuch SIMATIC Safety - Projektieren und Programmieren – Kapitel 5.2 – F-Ablaufgruppen festlegen

Empfehlung

Reduzieren Sie die Anzahl der Timer-Bausteine auf ein Minimum.

Folgende Bausteine greifen auf einen Timer zu:

- EV1oo2DI
- TWO_H_EN
- ACK_OP
- ESTOP1
- FDBACK
- MUT_P
- TOF
- TON
- TP

4.1.3 Multiinstanzen

Empfehlung

Verwenden Sie Multiinstanzen für fehlersichere Funktionsbausteine. Das bedeutet, dass die Baustein-internen Variablen in die Bausteinschnittstelle des aufrufenden Bausteins integriert werden.

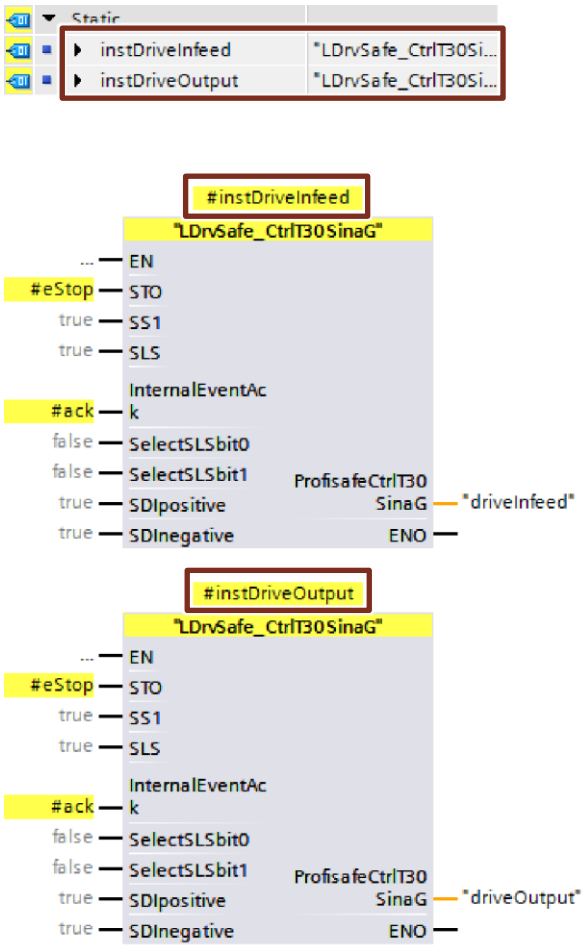
Vorteile

- Standarisierung von Sicherheitsprogrammen:
  Es werden keine Globaldaten für die Bausteinvariablen verwendet. Somit kann der aufrufende Baustein (inklusive der integrierten Bausteine) wiederverwendet werden.

Beispiel

Zwei Antriebe werden mit demselben Funktionsbaustein "LDrvSafe_CtrlT30SinaS" sicher angesteuert. Die Datenablage erfolgt in Multiinstanzen mit eindeutigen Namen.

Abbildung 4-4: Multiinstanzen



Die Bibliothek "LDrvSafe" zur Ansteuerung der Sicherheitsfunktionen von SINAMICS Antrieben finden Sie im Industry Online Support:

SIMATIC - Fehlersichere Bibliothek LDrvSafe zum Ansteuern von Safety Integrated Functions der Antriebsfamilie SINAMICS:

https://support.industry.siemens.com/cs/ww/de/view/109485794

4.1.4 Datenzugriff für Standardvariablen im Sicherheitsprogramm

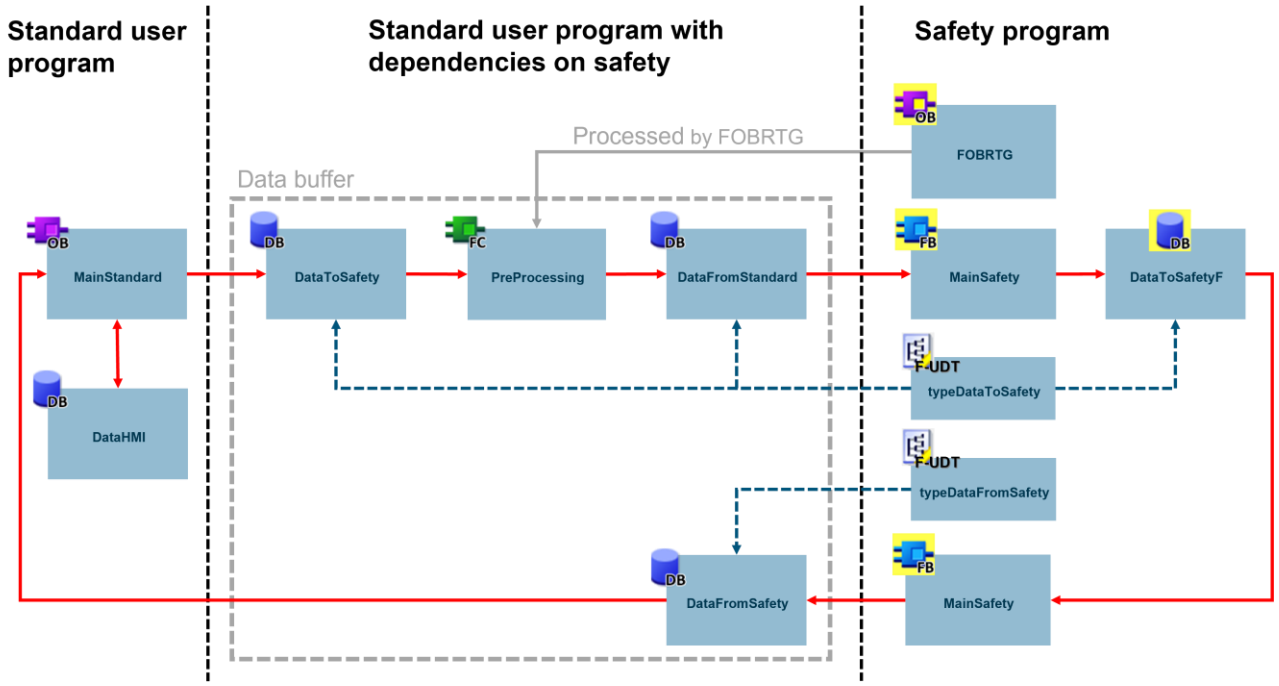
Bei jedem Zugriff vom Sicherheitsprogramm auf Standard-Variablen erfolgt vom Compiler jeweils eine separate Codierung. Dies gilt auch, wenn auf die gleiche Standard-Variable mehrfach zugegriffen wird. Aus Performance-Gründen kann es daher sinnvoll sein, die Standarddaten zu Beginn des Sicherheits-Anwenderprogramms einmalig in einen fehlersicheren Datenbereich zu kopieren und im weiteren Verlauf dann auf diese kopierten Variablen zuzugreifen.

Hinweis Informationen zum vereinfachten Verfahren entnehmen sie Kapitel 3.7.

Um die gewünschte Optimierung zu erreichen, nutzen Sie die folgenden Ressourcen. In Abbildung 4-5 ist eine Übersicht des Prozesses dargestellt. In der anschließenden Änderungsanleitung sind die Abweichungen zum vereinfachten Verfahren erklärt.

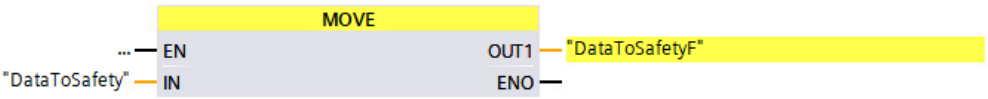
Die Notwendigkeit der Nutzung der Vorverarbeitung „PreProcessing“ bleibt hiervon unberührt.

Abbildung 4-5 Prozessübersicht der Laufzeitoptimierung des Datenaustausches



Änderungsanleitung

1. Fügen Sie einen zusätzlichen, fehlersicheren Datenbaustein „DataToSafety**F**“ zwischen „DataToSafety“ und der Verarbeitung im Sicherheitsprogramm ein.
2. Kopieren Sie zu Beginn des Sicherheitsprogrammes die Daten aus „DataToSafety“ in „DataToSafetyF“, nutzten Sie hierzu eine MOVE-Anweisung.



3. Arbeiten Sie anschließend im Sicherheitsprogramm mit den fehlersicheren Variablen aus dem Datenbaustein: „DataToSafetyF“.

4.2 Datenverfälschung vermeiden

Die Absicherungsmechanismen im Rahmen des Coded Processing (siehe Kapitel 5) analysieren den Programmablauf zyklisch auf Datenverfälschungen. Im Falle einer solchen Verfälschung löst ein spezieller Systemfunktionsbaustein einen F-STOPP der CPU aus.

Dieser Mechanismus bezweckt, Einflüsse wie EMV, defekte Bauteile und ähnliches aufzudecken und das System in einen sicheren Zustand zu bringen, bevor die Maschine zu einer Gefahr für Mensch und Umgebung wird.

Neben äußeren Einflüssen kann auch eine falsche Programmierung Datenverfälschung verursachen. Die häufigste Ursache für Datenverfälschung ist, dass das Standard-Anwenderprogramm oder ein externes Gerät (z. B. HMI) Daten beschreibt, während das Sicherheitsprogramm diese liest.

Dies kann in folgenden Situationen auftreten:

- Schreibender Zugriff durch höherpriore Alarme
- Schreibender Zugriff durch HMI/Kommunikation
- Verwendung von Taktmerkern

Aktualisierung eines Teil-PAE (Prozessabbild der Eingänge) durch höherpriore Alarme

Wie Sie Zugriffe vom Standard-Anwenderprogramm auf das Sicherheitsprogramm korrekt programmieren, finden Sie im Kapitel 3.7.

Bei Arithmetischen Funktionen kann es zu einem Überlauf oder Unterlauf des verwendeten Datentyps kommen. Sie müssen dann mit einem geeigneten Ersatzwert ihre Berechnung abschließen. Die Fehlerfreie Berechnung wird bei den folgenden Funktionen am Ausgang ENO angezeigt:

- ADD
- SUB
- MUL
- DIV
- NEG
- ABS
- DWORD_TO_WORD

OPC UA

Deaktivieren Sie die Option „Schreibbar aus HMI/ OPC UA“ für alle fehlersicheren Variablen in jeglichen Organisations-, Funktions-, Datenbausteinen und Funktionen, um Datenverfälschung zu unterbinden.

Checkliste

Mit der folgenden Checkliste können Sie anwendererzeugte STOP-Ursachen identifizieren und beheben.

Tabelle 4-1: Checkliste

Mögliche Ursachen	Checked
Überlauf Mathematische Funktionen können Unter- bzw. Überlaufen, was der Anwender im Programm abfangen muss. Verschalten Sie daher den ENO Ausgang der Mathematischen Funktionen	
Division durch 0 Kommt es im Sicherheitsprogramm zu einer Division durch 0 geht die F-CPU in STOP. Verschalten Sie daher den ENO Ausgang der Mathematischen Funktionen	
Zugriff über HMI Über ein HMI werden schreibend Daten (Merker, DBs) verändert, die im Sicherheitsprogramm lesend verwendet werden. Da die Kommunikation voreingestellt eine höhere Priorität als Safety hat, kann dadurch eine Datenverfälschung entstehen. Mögliche Lösungen finden Sie im Kapitel 3.7.	
Standardzugriff auf F-Daten Das Standard-Anwenderprogramm ändert Daten von fehlersicheren Variablen oder Teile von deren Absicherungen. Der schreibende Zugriff auf F-Daten ist ausschließlich im Sicherheitsprogramm zulässig.	
Pointerzugriff auf F-Daten Identisch zum Standardzugriff, kann der Zugriff zur Laufzeit bei ungünstigen Standardwerten zur Bildung eines Pointers auf F-Bereiche (Eingänge, Ausgänge, Datenbausteine usw.) auftreten.	

Weitere Informationen

Weitere Informationen und Ursachen für Datenverfälschung finden Sie im Siemens Industry Online Support:

Wie gehen Sie vor, wenn die F-CPU in STOP geht und im Diagnosepuffer die Meldung "Datenverfälschung im Sicherheitsprogramm ..." angezeigt wird?

https://support.industry.siemens.com/cs/ww/de/view/19183712

5 Glossar

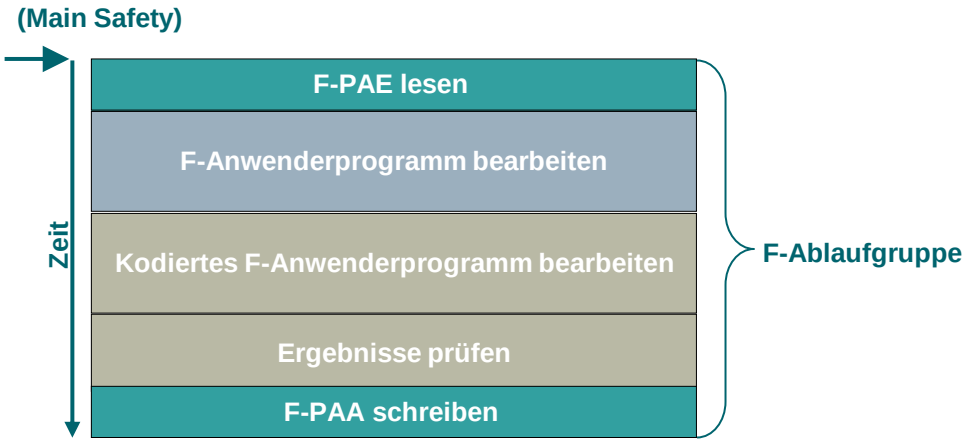
Coded Processing

Zur Erfüllung der normativen Anforderungen bezüglich Redundanz und Diversität nutzen alle SIMATIC F-CPUs das Prinzip des "Coded Processing". Bei diesem Prinzip wird das Sicherheitsprogramm von einem einzelnen Prozessor zweimal bearbeitet.

Dafür erzeugt der Compiler beim Übersetzen ein diversitäres (kodiertes) Sicherheitsprogramm, das als Absicherungsprogramm bezeichnet wird.

Im ersten Programmlauf wird das unveränderte Sicherheitsprogramm des Anwenders bearbeitet. Danach erfolgt die Bearbeitung des Absicherungsprogramms. Anschließend prüft die F-CPU die Ergebnisse. Bei korrekter Abarbeitung werden die sicheren Ausgänge geschrieben. Sollte die Prüfung versagen (z. B. aufgrund von Datenverfälschung), geht die F-CPU in den Stopp- Zustand und erstellt einen Eintrag im Diagnosepuffer.

Abbildung 5-1 Ablauf der Bearbeitung des Sicherheitsprogramms



Datenverfälschung

Datenverfälschung bedeutet, dass Daten des Sicherheitsprogramms durch äußere Einflüsse (z. B. EMV-Einflüsse) oder unzulässige, schreibende Zugriffe verfälscht werden.

F-CPU

Eine F-CPU ist eine Steuerung, die für sicherheitsgerichtete Aufgaben geeignet ist.

PROFIsafe

PROFIsafe ist ein Protokoll für die fehlersichere Kommunikation über PROFINET oder PROFIBUS.

Querschluss

Die Querschlusserkennung ist eine Diagnosefunktion eines Auswertegerätes, wodurch Kurz- bzw. Querschlüsse zwischen zwei Eingangskanälen (Sensorkreisen) erkannt werden.

Ein Querschluss kann beispielsweise durch das Quetschen einer Mantelleitung entstehen. Ohne Querschlusserkennung würde dies zur Folge haben, dass z. B. eine zweikanalige Not-Halt-Schaltung auch bei nur einem fehlerhaften Öffnerkontakt (Zweitfehler) keine Abschaltung auslöst.

RIOforFA

RIOforFA (Remote IO for Factory Automation) ist ein Standard der PROFIBUS & PROFINET International-Organisation und beschreibt unter anderem folgende Funktionen:

- Synchrone Bereitstellung kanalgranularer Diagnose vom Remote IOs für eine hohe Performance
- Kanalgranulare Passivierung und Wiedereingliederung von PROFIsafe Remote IOs

Rückführkreis

Ein Rückführkreis dient der Überwachung angesteuerter Aktoren (z. B. Relais oder Lastschütze) mit zwangsgeführten Kontakten bzw. Spiegelkontakten. Die Ausgänge können nur bei geschlossenem Rückführkreis aktiviert werden. Bei Verwendung eines redundanten Abschaltpfades muss der Rückführkreis beider Aktoren ausgewertet werden. Diese dürfen dafür auch in Reihe geschaltet werden.

Rückstellfunktion/rückstellen

Nach dem Auslösen einer Sicherheitsfunktion muss der Stoppzustand aufrechterhalten bleiben, bis ein sicherer Zustand für den Wiederanlauf gegeben ist.

Als Rückstellfunktion bzw. Rückstellen wird das Wiederherstellen der Sicherheitsfunktion und Aufheben des Stoppbefehls bezeichnet.

Oft wird hier auch vom "Quittieren der Sicherheitsfunktion" gesprochen.

Sicherheitsprogramm

Der Teil des Anwenderprogramms, in dem sicherheitsgerichtete Aufgaben bearbeitet werden.

STEP 7 Safety Basic/Advanced

STEP 7 Safety Basic und Advanced sind Optionspakete für STEP 7, mit denen F-CPUs projektiert und Sicherheitsprogramm erstellt werden können.

- Mit STEP 7 Safety Basic können Sie die fehlersicheren Steuerungen SIMATIC S7-1200 projektieren.
- Mit STEP 7 Safety Advanced können Sie alle fehlersicheren SIMATIC-Steuerungen projektieren.

6 Anhang

6.1 Service und Support

SiePortal

Die integrierte Plattform für Produktauswahl, Einkauf und Support - und Verbindung von Industry Mall und Online Support. Die neue Startseite, ersetzt die bisherigen Startseiten der Industry Mall sowie des Online Support Portals (SIOS) und fasst diese zusammen.

- Produkte & Services
  Unter Produkte & Services finden Sie alle unsere Angebote, die bisher im Mall Katalog verfügbar waren.
- Support
  Im Bereich Support finden Sie alle Informationen, die für die Lösung technischer Probleme mit unseren Produkten hilfreich sind.
- mySieportal
  mySiePortal ist Ihr persönlicher Bereich, der Funktionen, wie z.B. die Warenkorbverwaltung oder die Bestellübersicht anzeigt. Den vollen Funktionsumfang sehen Sie hier erst nach erfolgtem Login.

Das SiePortal rufen Sie über diese Adresse auf:
sieportal.siemens.com

Technical Support

Der Technical Support von Siemens Industry unterstützt Sie schnell und kompetent bei allen technischen Anfragen mit einer Vielzahl maßgeschneiderter Angebote - von der Basisunterstützung bis hin zu individuellen Supportverträgen.

Anfragen an den Technical Support stellen Sie per Web-Formular:
support.industry.siemens.com/cs/my/src

SITRAIN - Digital Industry Academy

Mit unseren weltweit verfügbaren Trainings für unsere Produkte und Lösungen unterstützen wir Sie praxisnah, mit innovativen Lernmethoden und mit einem kundenspezifisch abgestimmten Konzept.

Mehr zu den angebotenen Trainings und Kursen sowie deren Standorte und Termine erfahren Sie unter:
siemens.de/sitrain

Industry Online Support App

Mit der App "Industry Online Support" erhalten Sie auch unterwegs die optimale Unterstützung.

Die App ist für iOS und Android verfügbar:

6.2 Links und Literatur

Tabelle 6-1: Links und Literatur

Nr.	Thema
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link auf die Beitragsseite des Anwendungsbeispiels https://support.industry.siemens.com/cs/ww/de/view/109750255
\3\	Programmierleitfaden für SIMATIC S7-1200/1500 https://support.industry.siemens.com/cs/ww/de/view/90885040
\4\	Programmierstyleguide für SIMATIC S7-1200/1500 https://support.industry.siemens.com/cs/ww/de/view/109478084
\5\	SIMATIC Industrie Software SIMATIC Safety - Projektieren und Programmieren https://support.industry.siemens.com/cs/ww/de/view/54110126
\6\	Themenseite "Safety Integrated – Sicherheitstechnik in der Fertigungsautomatisierung" https://support.industry.siemens.com/cs/ww/de/view/109747812

6.3 Änderungsdokumentation

Tabelle 6-2: Änderungsdokumentation

Version	Datum	Änderung
V1.0.1	10/2017	Erste Ausgabe
V1.1.0	09/2020	Anpassungen und Korrekturen
V1.2.0	09/2021	Anpassungen und Korrekturen
V1.3.0	03/2023	Erweiterung Safety Unit und Anpassungen
V1.4.0	06/2023	Überarbeitung Datenaustausch Standard – Safety
V1.5.0	07/2023	Vereinfachung Datenaustausch
V1.6.0	07/2024	Erweiterung Datenaustausch zwischen Units