HANDBUCH

AC500-S

Sicherheitshandbuch V1.3.0

Übersetzung der englischen Originalausgabe. Im Zweifelsfall gilt das englische Originaldokument.

Inhaltsverzeichnis

1 Einführung

1.1 Verwendungszweck

Dieses Sicherheitshandbuch beschreibt die AC500-S-Sicherheitssteuerung. Es bietet detaillierte Informationen zur korrekten Installation, Ausführung, Programmierung und Wartung des Systems in Anwendungen der funktionalen Sicherheit bis SIL 3 gemäß IEC 61508, max. SIL 3 gemäß IEC 62061 und Performance Level e (Kat. 4) gemäß ISO 13849-1.

Die AC500-Serie von ABB ist eine SPS-basierte modulare Automationslösung, die die Kombination von Sicherheits- und Standard-E/A-Modulen erleichtert, um so die Anforderungen des Marktes im Bereich Automation zu erfüllen.

1.2 Dokumentenhistorie

Rev.	Beschreibung der Version / Änderungen	Wer	Datum
1.3.0	Verschiedene Verbesserungen im Text. Der Unternehmensname wurde geändert. Die Programmierumgebung für Sicherheitsgeräte wurde in „AC500-S Programming Tool“ umbenannt. Größere Änderungen: ● Features des neuen PROFIsafe V2.6 Protokolls wurden hinzugefügt, z. B.: – FLOAT32, INT32, UINT32 werden unterstützt – Kapitel 4.3.5: PROFIsafe V2.6 F-Parameter wurden hinzugefügt – Kapitel 4.3.6.1: PROFIsafe V2.6 F-(Sub-)Module wurden hinzugefügt – Kapitel 4.6.3: Aktualisierung gemäß der neuen F-Host Bibliothek SafetyBase_PROFIsafe_LV210_AC500_V22.lib – Anhang B.2.1: PROFIsafe V2.6 F-Device Diagnosemeldungen wurden hinzugefügt ● Neues Kapitel 4.6.6.4: Spezifische Funktionen für benutzerdefinierte CRC (neue Funktionsbausteine in der Bibliothek SafetyExt2_LV110_AC500_V27.lib) ● Neuer Anhang D: Firmware- / Software-Versionshistorie	ABB	04.02.2022
1.2.1	Verschiedene Verbesserungen im Text. Größere Änderungen: ● Kapitel 3.4.7 und 3.5.7: Neues Beschaltungsbeispiel für DX581-S und AI581-S wurde hinzugefügt. ● Kapitel 4.1: Informationen zu neuem Sicherheits-Engineering wurden hinzugefügt. ● Kapitel 6.2: Neue Checklistenposition Nr. 23 zur Prüfung der Byte-Reihenfolge wurde hinzugefügt.	ABB	24.03.2021

Rev.	Beschreibung der Version / Änderungen	Wer	Datum
1.2.0	Diverse Tippfehler wurden korrigiert und verschiedene Verbesserungen in Bezug auf den Text und die Abbildungen wurden vorgenommen. Das Layout wurde gemäß der aktuellen ABB-Markenstrategie geändert. Größere Änderungen: ● Kapitel 4.3.7.1: Das neue Sicherheitsverifizierungs-Tool SVT (Safety Verification Tool) wurde hinzugefügt. ● Sicherheitsmodule werden von AC500-V3-Standard-CPUs unterstützt. Spezifische Informationen zur Verwendung von Sicherheitsmodulen mit Standard-CPUs wurden in Anhänge B + C verlagert. Anhang B enthält alle spezifischen Informationen zu Sicherheitsmodulen mit V2-Standard-CPUs PM5xx. Anhang C enthält alle spezifischen Informationen zu Sicherheitsmodulen mit V3-Standard-CPUs PM56xx. ● Kapitel 3.1.2.6: „Aktualisierung von Firmware, Bootcode und Bootprojekt“ wurde aktualisiert. ● Die Montageanleitung der Sicherheits-E/A-Module wurde aktualisiert.	ABB	19.06.2020
1.1.0	Diverse Tippfehler wurden korrigiert. Verschiedene Verbesserungen im Text. Größere Änderungen: ● Informationen zu Sicherheits-CPUs SM560-S-FD-1(-XC) und SM560-S-FD-4(-XC) wurden hinzugefügt. ● Kap. 4.6.7: Die neue PROFIsafe F-Device Library SafetyDeviceExt_LV100_PROFIsafe_AC500_V27.lib wurde hinzugefügt. ● Kap. 4.6.8: Die neue Safety Library SafetyExt2_LV100_AC500_V27.lib wurde hinzugefügt. ● Detaillierte Informationen über relevante Normen wurden hinzugefügt. ● Die Checklisten für die Inbetriebnahme der AC500-S in Kapitel 6 wurden aktualisiert.	ABB	16.03.2018
1.0.5	Diverse Tippfehler wurden korrigiert. Kleinere Verbesserungen im Text und Entfernung der Screenshots älterer Versionen des Automation Builder. Größere Änderungen: ● Die neue PROFIsafe F-Host Library SAFETY-BASE_PROFIsafe_LV200_AC500_V22.lib wird im Dokument verwendet. ● Die Liste der häufig gestellten Fragen wurde ergänzt. ● Kap. 2.4: Details zu den Sicherheitswerten für AC500-S-Sicherheitsmodule wurden bereitgestellt. ● Kap. 4.3.6: „GEFAHR!“ zur Erklärung der Verwendung des PROFIsafe-Bits Device_Fault hinzugefügt. ● Kap. 6.3: Punkt 9 der Checkliste wurde hinzugefügt.	ABB	23.10.2017

Rev.	Beschreibung der Version / Änderungen	Wer	Datum
1.0.4	Diverse Tippfehler wurden korrigiert. Kleine Verbesserungen im Text. Größere Änderungen: Lizenzinformation wurde aktualisiert: ● Kap. 4.1: Hinweisblock mit Verweis auf die Installation der Lizenz PS501-S wurde entfernt. ● Kap. 4.2: Abbildung 63 aktualisiert (Programmier-Workflow, Schritt 2), wurde erweitert für die Lizenzhandhabung des Automation Builder, Version V2.0.2 (oder höher). ● Kap. 4.3.2: Die „Lizenzfreischaltung“ wurde um eine zusätzliche Lizenzinformation zur Verwendung der Automation Builder Version V2.0.2 (oder höher) erweitert. Zusatzinformationen wurde in Übereinstimmung mit der neuen F-Host-Bibliothek „SAFETYBASE_PROFIsafe_AC500_V22_Ext.lib“ hinzugefügt: ● Kap. 4.6.1: Tabelle für Bibliothek „SAFETY-BASE_PROFIsafe_AC500_V22_Ext.lib“ wurde aktualisiert. ● Kap. 4.6.3: Das Kapitel wurde in Übereinstimmung mit dem neuen Bibliotheknamen „SAFETYBASE_PROFIsafe_AC500_V22_Ext.lib“ aktualisiert. ● Kap. 6.2: Punkt 20 der Checkliste wurde entsprechend dem neuen Bibliotheknamen „SAFETYBASE_PROFIsafe_AC500_V22_Ext.lib“ aktualisiert.	ABB	27.03.2017
1.0.3	Diverse Tippfehler wurden korrigiert. Der Abkürzungsliste wurden zusätzliche Abkürzungen hinzugefügt. Das gesamte Dokumentenlayout wurde überarbeitet: ● Im Zuge der Standardisierung von Dokumenten wurde der gelbe Hintergrund bei Anmerkungen und Empfehlungen durch einen hellgrauen Hintergrund ersetzt. ● Die Symbole „GEFAHR“ und „HINWEIS“ wurden durch Standardsymbole in Textfeldern aus der deutschen Norm DIN 4844-2 ersetzt. Folgender Text wurde im Dokument geändert: ● Im Dokument werden nun mehr genormte Ausdrücke verwendet. ● Die Wertebereiche für Lager- und Transporttemperaturen wurden erweitert. ● Für die Sicherheits-CPU SM560-S wurde eine senkrechte Montageoption (mit Leistungsreduzierung) hinzugefügt und für die Sicherheits-E/A-Module DI581-S und AI581-S wurde diese korrigiert. ● LREAL wird von den Sicherheits-CPUs SM560-S nicht unterstützt und wurde aus dem Dokument entfernt. ● Die Beschreibung von POE SF_MAX_POWER_DIP_GET wurde geändert. ● Das Textfeld „GEFAHR“ wurde zur POE SF_DPRAM_PM5XX_S_SEND hinzugefügt, um Grenzen der POE-Verwendung zu erläutern. ● Die Begriffsdefinitionen für F_WD_Time2 und Device_WD2 in Kapitel 5.3 wurden korrigiert. ● In Kapitel 5.3 wurde „F_Host_WD“ in der Box „ACHTUNG“ durch „den mit SF_WDOG_TIME_SET eingestellten Wert“ ersetzt.	ABB	28.05.2015
1.0.2	Dem Titel des Dokumentes wurde das Wort „Originalanweisungen“ hinzugefügt.	ABB	17.04.2015

Rev.	Beschreibung der Version / Änderungen	Wer	Datum
1.0.1	Einige kleinere Tippfehler wurden korrigiert. Die TÜV SÜD-Bescheinigung wurde hinzugefügt. Folgender Text wurde im Dokument geändert: ● Die Ein-/Ausgänge der Sicherheits-E/A-Module sind von den anderen Schaltkreisen des Moduls nicht galvanisch getrennt. ● Die Sicherheitswerte der Sicherheitsausgänge des Moduls DX581-S(-XC) gelten nur, wenn der Parameter „Überwachung“ „Ein“ ist. ● Der Diagnosedeckungsgrad für das Modul DX581-S(-XC) muss ≥ 94 % sein. ● Es wurde deutlich gemacht, dass das Bootprojekt nur auf der SM560-S aktualisiert werden kann, wenn kein Bootprojekt in SM560-S geladen ist. ● In 100 Stunden ist max. ein Kommunikationsfehler (Ausgangssignale von CE_CRC oder Host_CE_CRC werden TRUE) zulässig, der vom Bediener mit dem Eingangssignal OA_C quittiert wird, ohne dass das verantwortliche Sicherheitspersonal kontaktiert werden muss. ● Die Zykluszeit der SM560-S muss dreimal statt nur zweimal in der Berechnung der Antwortzeit der Sicherheitsfunktion berücksichtigt werden. ● Die Werte für die Genauigkeit der Eingangsverzögerung der Eingänge bei der Berechnung der Antwortzeit der Sicherheitsfunktion wurden aktualisiert. ● Aktualisierung von Anhang A mit Systemdaten für AC500-S-XC.	ABB	08.03.2013
1.0.0	Erstausgabe	ABB	19.12.2012

1.3 Gültigkeit

Die Daten und Bilder in diesem Dokument sind nicht bindend. ABB behält sich das Recht vor, seine Produkte im Rahmen seiner Strategie der kontinuierlichen Produktentwicklung zu verändern.

1.4 Wichtige Anwenderinformation

Diese Dokumentation richtet sich an Fachpersonal, das mit den Grundsätzen und Anforderungen der funktionalen Sicherheit vertraut ist. Die AC500-S-Sicherheitssteuerung darf nur in Betrieb genommen werden, wenn die Sicherheitskonzepte und Anforderungen in diesem Sicherheitshandbuch vorher gelesen und verstanden wurden.

Die folgenden besonderen Hinweise werden in der gesamten Dokumentation verwendet, um Sie vor potenziellen Gefahren zu warnen oder auf bestimmte Informationen hinzuweisen.

GEFAHR!

Hinweise für Ihre persönliche Sicherheit sind im Handbuch durch dieses Sicherheitswarnsymbol hervorgehoben. Es zeigt an, dass Tod oder schwere Körperverletzung folgen können, sofern keine ausreichenden Vorsichtsmaßnahmen getroffen werden.

HINWEIS!

Dieses wichtige Symbol weist auf Informationen hin, die für eine erfolgreiche Anwendung und das Verständnis des Produktes von Belang sind. Es gibt an, dass unerwünschte Ergebnisse auftreten können, wenn die entsprechende Information nicht berücksichtigt wird.

1.5 Definitionen, Begriffe, Abkürzungen

1oo2	1oo2-Sicherheitsarchitektur: Dies bedeutet, dass es zwei parallel geschaltete Kanäle gibt, sodass jeder Kanal die Sicherheitsfunktion ausführen kann.
AC500	Nicht sicherheitsgerichtete SPS von ABB
AC500-XC	Nicht sicherheitsgerichtete SPS von ABB, geeignet für extreme Umgebungsbedingungen
AC500-S	ABB Sicherheitssteuerung für Anwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1)
AC500-S-XC	ABB Sicherheitssteuerung für Anwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1), geeignet für extreme Umgebungsbedingungen
AC500-S Programming Tool	Editor IEC 61131-3, enthalten in der Engineering Suite Automation Builder
ADC	Analog- zu Digitalwandler
AOPD	Aktive optoelektronische Schutzeinrichtung
Automation Builder	Integrierte Engineering Suite für ABB SPS, einschließlich AC500-S Programming Tool
CCF	Fehler gemeinsamer Ursache
Control Builder Plus PS501	Integrierte Engineering Suite für ABB SPS, einschließlich AC500-S Programming Tool, Vorgänger von Automation Builder
CPU	Prozessor
CRC	Cyclic Redundancy Check (zyklische Blockprüfung). Eine Nummer, die aus einem Datenbaustein abgeleitet oder zusammen mit diesem übertragen wird, um korrupte Daten festzustellen.
DC	Diagnosedeckungsgrad
DPRAM	Dual-Port RAM
DUT	Datentypobjekt
IEC	Internationale Elektrotechnische Kommission
EDM	Externes Geräteüberwachungssignal, das den Zustandsübergang eines Aktors wiedergibt.
EMV	Elektromagnetische Verträglichkeit
EN	Europäische Norm (EN)
EPROM	Löschbarer programmierbarer Nur-Lese-Speicher
Fehlerschwere	Durch eine Zahl angegeben. Je niedriger die Zahl, desto kritischer der angezeigte Fehler. Beispiel: „1“ = Prozessor startet nicht, da der Fehler keinen Normalbetrieb gestattet, „11“ = andere Parametereinstellungen
ESD	Elektrostatische Entladung
BWS (Englisch: ESPE)	Berührungslos wirkende Schutzeinrichtung (z. B. Lichtvorhang)
F-Host	Datenverarbeitungseinheit, die ein spezielles Protokoll ausführen und den „Black Channel“ bedienen kann ↯ *[2]*
F-Device	Passiver Kommunikationsteilnehmer, der ein spezielles Protokoll ausführen kann, das im Normalfall vom F-Host zum Datenaustausch angestoßen wird ↯ *[2]*
F-Parameter	Failsafe-Parameter, wie unter ↯ *[2]* definiert

FAQ	Häufig gestellte Fragen
FB	Funktionsbaustein
FUP	Funktionsbausteinsprache (Programmiersprache laut IEC 61131)
Flash-Speicher	Nichtflüchtiger Computerspeicherchip, der elektrisch gelöscht und erneut programmiert werden kann
FSCP	Functional safety communication profile
FV	Failsafe-Wert
GSDML	Generic Station Description Markup Language
ID	Kennung
IO-Controller	Controller, der im Kontext von PROFINET die Automatisierungs-Task steuert
IO-Device	Feldgerät, das im PROFINET-Kontext von einem IO-Controller überwacht und gesteuert wird
iParameter	Individueller Sicherheits-Geräteparameter
KOP	Kontaktplan (Programmiersprache laut IEC 61131)
Loopback	Das programmierbare Routing-Feature eines Busgerätes leitet eine F-Host-Nachricht unbeabsichtigt zurück zum F-Host, der eine Nachricht derselben Länge erwartet (siehe *www.profisafe.net*).
LSB	Niederwertigstes Bit (least significant bit)
Max. SIL	Maximales Safety Integrity Level (IEC 62061)
MSB	Höchstwertigstes Bit (most significant bit)
MTBF	Mittlerer Ausfallabstand
MTTF	Mittlere Zeit bis zum ersten Ausfall
Muting	Muting ist die gewollte Unterdrückung der Sicherheitsfunktion. Dies ist z. B. erforderlich, wenn Material in den Gefahrenbereich transportiert wird.
NC	Öffner. Diese Kontakte unterbrechen den Stromkreis, wenn das Relais aktiviert wird. Der Stromkreis ist intakt, wenn das Relais nicht aktiv ist.
NO	Schließer. Diese Kontakte schließen den Stromkreis, wenn das Relais aktiviert wird. Der Stromkreis ist unterbrochen, wenn das Relais nicht aktiv ist.
OEM	Erstausrüster
OSSD	Ausgangssignal-Schaltelement (Output Signal Switching Device)
Passivierung	Die Passivierung ist der besondere Zustand der Sicherheits-E/A-Module, der zur Lieferung von sicheren Ersatzwerten, den 0-Werten bei AC500-S, an die Sicherheits-CPU führt.
PC	Personal Computer
PELV	Funktionskleinspannung mit sicherer Trennung
PES	Programmierbares elektronisches System (siehe IEC 61508)
PFD	Wahrscheinlichkeit eines Ausfalls bei Anforderung
PFH	Ausfallwahrscheinlichkeit pro Stunde
PL	Performance Level gemäß ISO 13849-1
SPS	Speicherprogrammierbare Steuerung
POE	Programmorganisationseinheit
Power Cycle	Ein Power Cycle umfasst das Ausschalten der Sicherheits-CPU, ein anschließendes Warten von mindestens 1,5 s und das erneute Einschalten der Sicherheits-CPU.

PROFIsafe	Sicherheitsbezogenes Busprofil von PROFIBUS DP/PA und PROFINET IO zur Kommunikation zwischen Sicherheitsprogramm und Sicherheits-E/A im Sicherheitssystem.
PROFINET	Technischer Industriestandard für Datenkommunikation über Industrial Ethernet
Prüfintervall	Das Prüfintervall bezieht sich auf eine regelmäßig durchgeführte Prüfung, um Fehlfunktionen im Sicherheitssystem zu finden, sodass gegebenenfalls der vorherige Neuzustand des Systems so gut wie möglich wiederhergestellt werden kann. Die Zeitspanne zwischen diesen Prüfungen ist das Prüfintervall.
PS	Programmiersystem
PTC	Positiver Temperaturkoeffizient
RAM	Random Access Memory
Reintegration	Das Schalten von Ersatzwerten „0“ zu Prozessdaten.
RIOforFA	Profil für dezentrale E/As für die Fabrikautomation. Um die Qualitätsinformationen eines Kanals parallel zum Diagnosesystem zu bekommen ↳ *[12]*.
Sicherheitsvariable	Eine Variable zur Ausführung einer Sicherheitsfunktion in einem Sicherheitssystem.
SCA	Safety Code Analysis (Sicherheitscodeanalyse) – ein Software-Tool von ABB, um automatisch CODESYS Safety-Programmierrichtlinien zu überprüfen.
SD-Karte	Sichere digitale Speicherkarte
SELV	Schutzkleinspannung
SFRT	Antwortzeit der Sicherheitsfunktion (= Safety Function Response Time)
SIL	Safety Integrity Level (IEC 61508)
ST	Strukturierter Text (Programmiersprache laut IEC 61131)
SVT	Safety Verification Tool – ein Software-Tool von ABB zur Verifizierung der AC500-S-Sicherheitskonfiguration im Automation Builder
TÜV	Technischer Überwachungsverein
TWCDT	Gesamt-Worst-Case-Verzögerungszeit (= Total Worst Case Delay Time)
ULP	ULP (Unit in the Last Place) ist der Abstand zwischen Gleitpunktzahlen, d. h. der Wert, den das niederwertigste Bit darstellt, wenn es 1 ist (siehe *http://en.wikipedia.org/wiki/Unit_in_the_last_place* für weitere Details).
WLAN	Wireless Local Area Network

1.6 Zertifizierung zur funktionalen Sicherheit

Die AC500-S-Sicherheitsmodule sind sicherheitstechnische Komponenten bis SIL 3 laut IEC 61508, max. SIL 3 laut IEC 62061 und PL e laut ISO 13849-1 gemäß Zertifizierung durch die TÜV SÜD Rail GmbH (Deutschland).

AC500-S ist eine Sicherheitssteuerung, deren Zuverlässigkeit im Betrieb durch die Verwendung von 1oo2-Redundanz in der Hardware und zusätzlichen Diagnosefunktionen für Hard- und Software im Vergleich zu einer nicht sicherheitsgerichteten SPS deutlich verbessert ist. Die integrierten Diagnosefunktionen für Sicherheitsintegrität basieren auf den zum Zertifizierungszeitpunkt gültigen Sicherheitsstandards ↳ *Zertifizierungsbericht von TÜV SÜD Rail für AC500-S [1]*. Diese Sicherheitsintegritätstests schließen Testroutinen, die während des gesamten Betriebs laufen, mit ein, sodass die AC500-S-Sicherheitssteuerung geeignet ist für Anwendungen der Maschinensicherheit sowie Prozessanwendungen bis zu SIL 3 laut IEC 61508, max. SIL 3 laut IEC 62061 und PL e laut ISO 13849-1.

HINWEIS!
Siehe Zertifizierungsbericht von TÜV SÜD Rail für AC500-S ↳ *[1]* für eine vollständige Liste der Normen und weitere Details wie Versionen der Normen etc.

Das Prüfintervall der AC500-S-Sicherheitssteuerung beträgt 20 Jahre.

Werte zu PFH, PFD, MTTFd, Kategorie und DC aus IEC 61508, IEC 62061 und ISO 13849-1 erfüllen für AC500-S-Sicherheitsmodule die Anforderungen für SIL 3, max. SIL 3 und PL e ↳ *Kapitel 2.4 „Sicherheitswerte“ auf Seite 20*.

1.7 Referenzen / zugehörige Dokumente

[1] - Zertifizierungsbericht von TÜV SÜD Rail für Sicherheitssteuerung AC500-S, Version ab 2018, verfügbar unter *www.abb.com/plc*

[2] - PROFIsafe – Profil für Sicherheitstechnologie auf PROFIBUS DP und PROFINET IO-Profil, in Bezug auf IEC 61784-3-3, ab Version 2.6MU1, 2018/08

[3] - AC500-Anwenderdokumentation für Automation Builder / Control Builder Plus, verfügbar unter *www.abb.com/plc*

[4] - IEC 61131, ab 2003, speicherprogrammierbare Steuerungen, Teil 3 – Programmiersprachen

[5] - Computer Science and Engineering at University of California, Riverside, Chapter 14, Ch14_Floating Point Calculations and its drawbacks.pdf

[6] - Anwenderbeispiele mit PLCopen Safety-Funktionen, ab Version 1.0.1, 2008

[7] - PROFIsafe-Systembeschreibung, Version – ab Nov. 2007

[8] - PLCopen Safety: Konzepte und Funktionsbausteine, ab Version 1.0, 2006

[9] - ISO 13849-1: Sicherheit von Maschinen – Sicherheitsbezogene Teile von Steuerungen – Teil 1: Allgemeine Gestaltungsleitsätze, ab 2015

[10] - PROFIBUS-Richtlinie: PROFIsafe – Umweltanforderungen, V2.5, ab März 2007

[11] - PROFIBUS-Richtlinie: Kommunikations-Funktionsbausteine bei PROFIBUS DP und PROFINET IO, V2.0, November 2005 Best.-Nr. 2.182 (oder neuer)

[12] - RIO-FA_3242_V110_Aug18.pdf, 2018/10/30, Version 1.1.0, Best.-Nr. 3.242, *https://de.profibus.com/downloads/remote-io-for-factory-automation-rio-for-fa*

1.8 Anwendbare Normen

Norm	Datum	Titel
IEC 61508	2010	Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme
IEC 62061	2021	Sicherheit von Maschinen – Funktionale Sicherheit sicherheitsbezogener elektrischer, elektronischer und programmierbarer elektronischer Systeme
ISO 13849-1	2015	Sicherheit von Maschinen – Sicherheitsbezogene Teile von Steuerungen – Teil 1: Allgemeine Gestaltungsleitsätze
IEC 60204-1	2016	Sicherheit von Maschinen – Elektrische Ausrüstung von Maschinen – Teil 1: Allgemeine Anforderungen
IEC 61496-1	2020	Sicherheit von Maschinen – Berührungslos wirkende Schutzeinrichtungen

Norm	Datum	Titel
IEC 61511-1 + AMD1	2016 2017	Funktionale Sicherheit – Sicherheitstechnische Systeme für die Prozessindustrie – Teil 1: Allgemeines, Begriffe, Anforderungen an Systeme, Hardware und Software
IEC 61326-3-1	2017	EMV für funktionale Sicherheit
IEC 61131-2	2017	Speicherprogrammierbare Steuerungen – Teil 2: Betriebsmittelanforderungen und Prüfungen
ISA-71.04-2013 Harsh Group A	2016	Environmental Conditions for Process Measurement and Control Systems – Airborne Contaminants
IEC 60721-3-3	2002	Klassifizierung von Umweltbedingungen – Teil 3-3: Klassen von Umwelteinflussgrößen und deren Grenzwerte – Ortsfester Einsatz, wettergeschützt
CISPR 16-1-2	2014	Anforderungen an Geräte und Einrichtungen sowie Festlegung der Verfahren zur Messung der hochfrequenten Störaussendung (Funkstörungen) und Störfestigkeit – Teil 1-2: Geräte und Einrichtungen zur Messung der hochfrequenten Störaussendung (Funkstörungen) und Störfestigkeit – Koppeleinrichtungen zur Messung der leitungsgeführten Störaussendung
CISPR 16-2-1	2017	Anforderungen an Geräte und Einrichtungen sowie Festlegung der Verfahren zur Messung der hochfrequenten Störaussendung (Funkstörungen) und Störfestigkeit – Teil 2-1: Verfahren zur Messung der hochfrequenten Störaussendung (Funkstörungen) und Störfestigkeit – Messung der leitungsgeführten Störaussendung
CISPR 16-2-3	2016	Anforderungen an Geräte und Einrichtungen sowie Festlegung der Verfahren zur Messung der hochfrequenten Störaussendung (Funkstörungen) und Störfestigkeit – Teil 2-3: Verfahren zur Messung der hochfrequenten Störaussendung (Funkstörungen) und Störfestigkeit – Messung der gestrahlten Störaussendung
IEC 61000-4-2	2008	Elektromagnetische Verträglichkeit (EMV) – Teil 4-2: Prüf- und Messverfahren – Prüfung der Störfestigkeit gegen die Entladung statischer Elektrizität
IEC 61000-4-3	2010	Elektromagnetische Verträglichkeit (EMV) – Teil 4-3: Prüf- und Messverfahren – Prüfung der Störfestigkeit gegen hochfrequente elektromagnetische Felder
IEC 61000-4-4	2012	Elektromagnetische Verträglichkeit (EMV) – Teil 4-4: Prüf- und Messverfahren – Prüfung der Störfestigkeit gegen schnelle transiente elektrische Störgrößen/Burst
IEC 61000-4-5	2017	Elektromagnetische Verträglichkeit (EMV) – Teil 4-5: Prüf- und Messverfahren – Prüfung der Störfestigkeit gegen Stoßspannungen
IEC 61000-4-6	2013	Elektromagnetische Verträglichkeit (EMV) – Teil 4-6: Prüf- und Messverfahren – Störfestigkeit gegen leitungsgeführte Störgrößen, induziert durch hochfrequente Felder
IEC 61000-4-8	2009	Elektromagnetische Verträglichkeit (EMV) – Teil 4-8: Prüf- und Messverfahren – Prüfung der Störfestigkeit gegen Magnetfelder mit energietechnischen Frequenzen
IEC 60715	2017	Abmessungen von Niederspannungsschaltgeräten – Genormte Tragschienen für die mechanische Befestigung von elektrischen Geräten in Schaltanlagen
IEC 60068-2-1	2009	Umgebungseinflüsse – Teil 2-1: Prüfverfahren – Prüfung A: Kälte
IEC 60068-2-6	2007	Umgebungseinflüsse – Teil 2-6: Prüfverfahren – Prüfung Fc: Schwingen (sinusförmig)

Norm	Datum	Titel
IEC 60068-2-27	2008	Umgebungseinflüsse – Teil 2-27: Prüfverfahren – Prüfung Ea und Leitfaden: Schocken
IEC 60068-2-30	2005	Umgebungseinflüsse – Teil 2-30: Prüfverfahren – Prüfung Db: Feuchte Wärme, zyklisch (12 + 12 Stunden)
IEC 60068-2-52	2017	Umgebungseinflüsse – Teil 2-52: Prüfverfahren – Prüfung Kb: Salznebel, zyklisch (Natriumchloridlösung)
IEC 60068-2-64	2008	Umgebungseinflüsse – Teil 2-64: Prüfverfahren – Prüfung Fh: Schwingen, Breitbandrauschen (digital geregelt) und Leitfaden
IEC 60068-2-78	2012	Umgebungseinflüsse – Teil 2-78: Prüfverfahren – Prüfung Cab: Feuchte Wärme, konstant

HINWEIS!
Wenden Sie sich an den technischen Support von ABB für weitere Details.

2 Übersicht AC500-S-Sicherheitssteuerung

2.1 Übersicht

Die AC500-S ist ein 1oo2-System (Sicherheits-CPU und Sicherheits-E/A-Module), das zur Ausführung von Sicherheitsfunktionen, die SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1) erfordern, in Systemen mit hoher Anforderungsrate in Anwendungen der Maschinensicherheit und in Systemen mit niedriger Anforderungsrate in sicherheitsgerichteten Prozessanwendungen eingesetzt wird. Ein 1oo2-System verfügt über zwei Mikroprozessoren. Jeder davon führt die Sicherheitslogik in seinem eigenen Speicherbereich aus und beide vergleichen die Ergebnisse der Ausführung. Sobald eine Diskrepanz bei der Ausführung oder ein Fehler festgestellt wird, wechselt das System in einen sicheren Zustand, der für jedes Sicherheitsmodul separat beschrieben wird.

2.1.1 System

Die AC500-S-Sicherheitssteuerung ist in die AC500-Plattform integriert, sodass der Look-&-Feel-Ansatz gleich bleibt. Aufgrund der Integration in die AC500-SPS-Plattform sind die allgemeinen AC500-Systemcharakteristika (Mechanik, Programmierung, Konfiguration etc.) auch für die AC500-S-Sicherheitsmodule gültig.

Alle AC500-Standardmodule gelten als rückwirkungsfreie Module für die AC500-S-Sicherheitssteuerung. Im Gegensatz zu den Sicherheitsmodulen werden rückwirkungsfreie Module nicht für Sicherheitsfunktionen verwendet. Ein Fehler in einem dieser Module beeinflusst die Ausführung der Sicherheitsfunktionen nicht negativ.

Der Begriff „Integrated Safety“, der für die AC500-S-Sicherheitssteuerung und die AC500-Plattform verwendet wird, bedeutet Folgendes:

- Ein PROFINET IO Feldbus wird für die sicherheitsgerichtete und nicht sicherheitsgerichtete Kommunikation verwendet.
- Dieselbe Entwicklungsumgebung wird für die Programmierung von Sicherheits- und Standardmodulen für gleiches „Look & Feel“ verwendet.
- Dieselbe Hardware und Verkabelung wird für Sicherheits- und Standardmodule für gleiches „Look & Feel“ verwendet.
- Dasselbe Diagnosekonzept wird für Sicherheits- und Standardmodule verwendet.

Abb. 1: Überblick der AC500-Familie von ABB mit Sicherheits- und Standardmodulen

1 **Standard-Kommunikationsmodul**
AC500 deckt alle üblichen Kommunikationsstandards wie Ethernet, EtherCAT, PROFINET IO, PROFIBUS DP, CANopen, DeviceNet, Modbus TCP, Modbus serial, Serial, ABB CS31 und PROFIsafe via PROFINET ab. Es kann mit anderen Systemen kombiniert werden, um so optimale Netzwerkknoten zu erzielen; dadurch ist AC500 von ABB sowohl für kleinere als auch große industrielle Systeme geeignet.

2 **Sicherheits-CPU**
Sicherheits-CPUs, zertifiziert bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1). Eine Reihe von Funktionen wie Systemdiagnose über LEDs und Onboard-Displays von Standard-CPUs bietet zusätzlich das Diagnosekonzept, das für Sicherheitsintegrität erforderlich ist.

3 **Standard-CPU**
Die komplette Bandbreite an AC500-Standard-CPUs von ABB kann mit Sicherheits-CPUs verwendet werden, um kundenspezifische Lösungen zu schaffen – sogar für die anspruchsvollsten Anforderungen. Die Programmierung von sicherheitsgerichteten und nicht sicherheitsgerichteten Anwendungen wird über die Schnittstelle einer nicht sicherheitsgerichteten SPS durchgeführt.

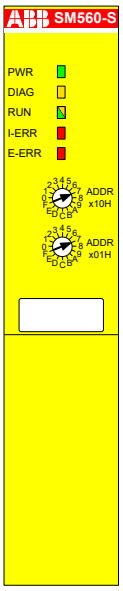
4 **Sicherheits-E/A-Modul**
Sicherheits-E/A-Module, zertifiziert bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1). Durch Produktmerkmale wie die Fehlerdiagnostik pro Kanal und die Flexibilität, zwischen der Abschaltung pro Kanal oder der Modulabschaltung bei Kanalfehlern zu wählen, wird sicheres Arbeiten so viel einfacher.

5 **Standard-E/A-Modul**
Mit den Standard-E/A-Modulen von ABB kann die gesamte Bandbreite an E/A-Modulen, d. h. S500 und S500-eCo, an eine nicht sicherheitsgerichtete SPS angeschlossen werden. Die große Zahl an Funktionen der konfigurierbaren E/A-Module AC500 ermöglicht kundenspezifische und kostengünstige Lösungen zur Optimierung industrieller Anwendungen.

2.1.2 Sicherheitskomponenten

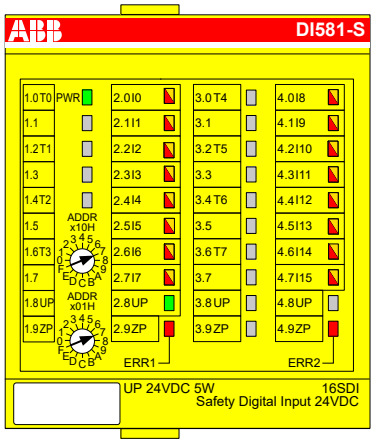
Die AC500-S-Sicherheitssteuerung besteht aus folgenden sicherheitsbezogenen Hardwarekomponenten:

SM560-S /
SM560-S-FD-1 /
SM560-S-FD-4



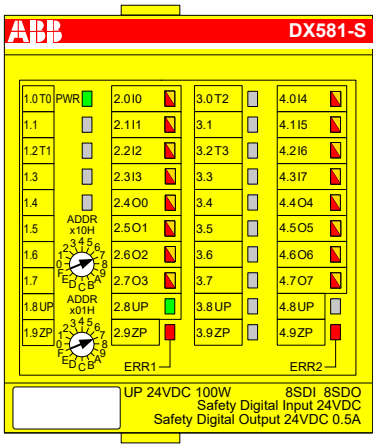
Sicherheits-CPU (Sicherheitsmodul) für Sicherheitsanwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1).

DI581-S



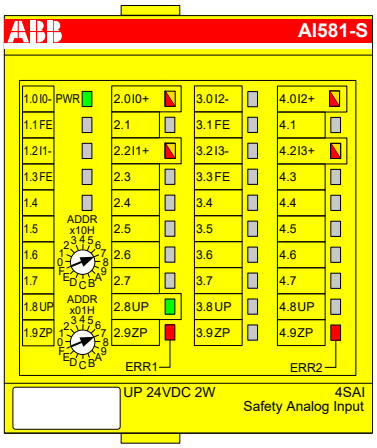
Binäres digitales Sicherheits-Eingabemodul DI581-S mit 16 sicherheitsgerichteten Eingangskanälen (bis SIL 2 oder PL d) oder 8 sicherheitsgerichteten Eingangskanälen (bis SIL 3 oder PL e) mit 8 Testimpuls-Ausgangskanälen.

DX581-S



Binäres Sicherheits-E/A-Modul DX581-S mit 8 sicherheitsgerichteten Ausgangskanälen (bis SIL 3 oder PL e) und 8 sicherheitsgerichteten Eingangskanälen (bis SIL 2 oder PL d) oder 4 sicherheitsgerichteten Eingangskanälen (bis SIL 3 oder PL e) mit 4 Testimpuls-Ausgangskanälen.

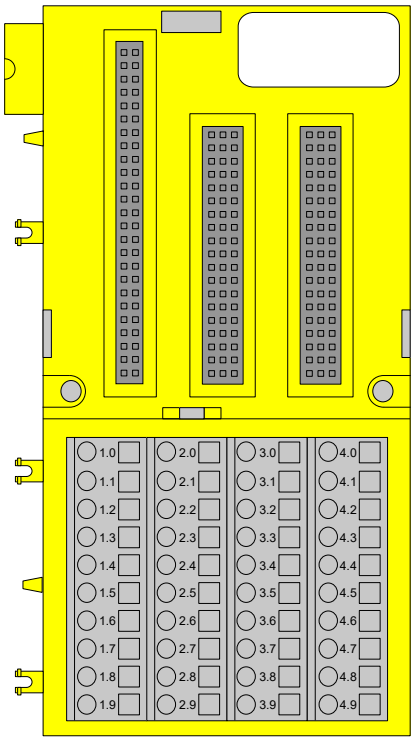
AI581-S



Analoges Sicherheits-Eingabemodul AI581-S mit 4 Sicherheits-Stromeingangskanälen 0 … 20 mA (bis SIL 2 oder PL d) oder 2 Sicherheits-Stromeingangskanälen (bis SIL 3 oder PL e).

Die folgende rückwirkungsfreie Komponente muss zum Einbau von Sicherheits-E/A-Modulen verwendet werden:

TU582-S



Klemmenblock mit Federzugklemmen TU582-S für Sicherheits-E/A-Module.

2.2 Bestimmungsgemäße Verwendung

AC500-S-Sicherheitskomponenten von ABB in Kundenapplikationen müssen von den zuständigen Zertifizierungsbehörden abgenommen und zugelassen werden. ABB übernimmt keine Haftung für eventuelle Konsequenzen, die aus einer unsachgemäßen Verwendung entstehen:

- Nichterfüllung von Normen und Richtlinien
- Unautorisierte Veränderungen der Geräte, Anschlüsse und Einstellungen
- Verwendung von nicht autorisierten oder ungeeigneten Geräten
- Nichtbeachtung der Sicherheitsanweisungen in diesem Handbuch

2.3 Sicherheitskreis

Der Sicherheitskreis, zu dem die AC500-S-Sicherheitssteuerung gehört, besteht aus Sensoren, Sicherheitssteuerung und Aktoren.

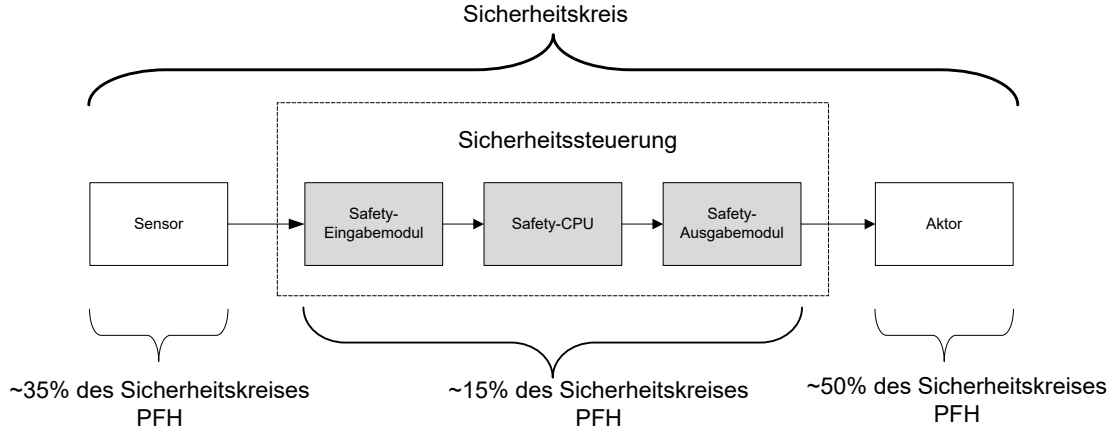


Abb. 2: Typischer Sicherheitskreis mit AC500-S-Sicherheitssteuerung

Zur Berechnung der Werte PFH/PFD eines Beispiel-Sicherheitssystems wird normalerweise ein Wert von 15 % für die Sicherheitssteuerung angenommen.

2.4 Sicherheitswerte

Tab. 1: Die folgenden Sicherheitswerte müssen für die AC500-S-Sicherheitsmodule verwendet werden:

Typ	**SIL(1)** **max. SIL(2)**	**PL(3)**	**DC(4)**	**MTTFd(5)**	**PFHd(6)**	**PFHd(7)**	**PFDg(8)**	**T1(9)**	**SFF(10)**	**β(11)**
SM560-S(-XC) / SM560-S-FD-1(-XC) / SM560-S-FD-4(-XC)	3	e	97	1280	1,90E-09	8,95E-11	7,90E-06	20	98	2
AI581-S(-XC)	3	e	97	920	2,95E-09	4,50E-10	3,80E-05	20	99	2
DI581-S(-XC)	3	e	95	2270	1,45E-09	4,40E-10	3,70E-05	20	98	2
Eingänge von DX581-S(-XC)	3	e	94	2250	1,45E-09	4,50E-10	3,80E-05	20	98	2

Typ	SIL[1] max. SIL[2]	PL[3]	DC[4]	MTTFd[5]	PFHd[6]	PFHd[7]	PFDg[8]	T1[9]	SFF[10]	β[11]
Ausgänge von DX581-S(-XC) mit Parameter Detection = „Ein“	3	e	94	1985	1,60E-09	4,50E-10	3,80E-05	20	99	2
Ausgänge von DX581-S(-XC) mit Parameter Detection = „Aus“	2	d	85	200	1,19E-08	1,08E-08	4,70E-04	20	auf Anfrage	2

(1) - SIL (Safety Integrity Level) gemäß IEC 61508

(2) - Max. SIL (maximales Safety Integrity Level) gemäß IEC 62061

(3) - PL (Performance Level) gemäß ISO 13849-1

(4) - Diagnosedeckungsgrad, % (siehe ISO 13849-1)

(5) - Mittlere Zeit bis zu einem gefährlichen Ausfall (Jahre) gemäß ISO 13849-1

(6) - Wahrscheinlichkeit eines gefahrbringenden Ausfalls pro Stunde gemäß IEC 62061

(7) - Wahrscheinlichkeit eines gefahrbringenden Ausfalls pro Stunde gemäß IEC 61508 (Betriebsart mit hoher Anforderungsrate)

(8) - Durchschnittliche Wahrscheinlichkeit eines Versagens der vorgesehenen Funktion bei Anforderung gemäß IEC 61508 (Betriebsart mit niedriger Anforderungsrate)

(9) - Prüfintervall – Einsatzzeitraum – Lebenszeit

(10) - SFF (Anteil sicherer Ausfälle), % gemäß IEC 61508

(11) - β (Betafaktor), % für Ausfälle infolge gemeinsamer Ursachen gemäß IEC 61508

GEFAHR!

Bei der Berechnung der Sicherheitswerte wird von der durchschnittlichen Temperatur ausgegangen. Die durchschnittliche Temperatur für den erweiterten Temperaturbereich (-40 °C … +70 °C) sowie den Normaltemperaturbereich (0 °C … +60 °C) ist auf +40 °C definiert.

Stellen Sie sicher, dass die durchschnittliche Betriebstemperatur für in Betrieb befindliche AC500-S- und AC500-S-XC-Module +40 °C nicht überschreitet.

2.5 Fachpersonal

AC500-S-Sicherheitssteuerungen dürfen nur in Verbindung mit dieser Dokumentation eingerichtet und verwendet werden.

Sicherheits-Anwendungstechniker für die AC500-S-Sicherheitssteuerung
Nur Fachpersonal, das für die Inbetriebnahme von sicherheitsgerichteten Geräten, Systemen und Schaltkreisen entsprechend anerkannter funktionaler Sicherheitspraktiken und Normen autorisiert ist, darf die AC500-S-Sicherheitssteuerung in Betrieb nehmen und betreiben.

Das folgende Grundwissen über AC500-Systeme ist erforderlich, um dieses Sicherheitshandbuch für AC500-S korrekt zu verstehen:

- AC500-Automatisierungssystem
- Programmierumgebung Automation Builder / Control Builder Plus (Systemkonfiguration und -programmierung in den Programmiersprachen ST, KOP und FUP).

2.6 Lebenszyklus

Alle AC500-S-Sicherheitsmodule haben eine maximale Lebensdauer von 20 Jahren. Das bedeutet, dass alle AC500-S-Sicherheitsmodule mindestens eine Woche vor Ablauf dieser 20 Jahre (ab dem Lieferdatum durch ABB) außer Betrieb genommen und durch neue AC500-S-Sicherheitsmodule ersetzt werden müssen.

2.7 Installation der Sicherheitsmodule

Folgende Regeln gelten für das Installieren von Sicherheitsmodulen:

- Die Installation hat mit den geeigneten Hilfsmitteln und Werkzeugen in Übereinstimmung mit der Dokumentation zu erfolgen.
- Die Installation der Geräte darf nur im spannungslosen Zustand und durch Fachpersonal erfolgen.
- Die allgemeinen und national gültigen Sicherheitsnormen müssen strikt beachtet werden.
- Die Elektroinstallation erfolgt unter Beachtung der relevanten Normen.
- Die erforderlichen Schutzmaßnahmen gegen elektrostatische Entladung sind zu treffen.

HINWEIS!

Schäden an der SPS durch unsachgerecht ausgewählte Schaltschränke

AC500-S-Sicherheitsmodule müssen in geschlossenen Schaltschränken verwendet werden, die für ein Modul mit Schutzart IP 20 geeignet sind. ↳ *Weitere Informationen siehe [3].*

2.8 Modulaustausch

Die Sicherheits-CPU SM560-S / SM560-S-FD-1 / SM560-S-FD-4 erkennt einen Austausch der Sicherheits-E/A-Module automatisch, wenn das System hochgefahren wird. Das Gesamtsystem (Sicherheits-CPU und PROFIsafe-Features mit eindeutigen Adressen für sicherheitsgerichtete Geräte ↳ *[2]*) verfügt über einen Mechanismus, der automatisch sicherstellt, dass ausgetauschte Sicherheitsmodule mit den korrekten Parametern betrieben und nicht kompatible Modulversionen zurückgewiesen werden. Ein unsicherer Zustand ist nicht möglich, wenn der falsche Sicherheits-E/A-Modultyp in den Klemmenblock TU582-S eingesteckt wird.

2.9 Neustartverhalten von AC500-S

Wenn eine Sicherheits-CPU SM560-S / SM560-S-FD-1 / SM560-S-FD-4 durch einen Power Cycle neu gestartet wird, geht die zuvor gespeicherte Fehlerinformation verloren. Zusätzliche Maßnahmen im Sicherheitsprogramm wie das Speichern von Fehlerinformationen und anderen Informationen im Flash-Speicher der Sicherheits-CPU müssen auf der Sicherheits-CPU programmiert sein, damit diese Informationen dort dauerhaft gespeichert sind. Die Sicherheits-E/A-Module erhalten ihre Parameter jedes Mal, wenn das System neu gestartet wird. Die Sicherheits-CPU kann Sicherheits-E/A-Module durch das PROFIsafe-Startverhalten wieder integrieren ↳ *[2]*. Wenn Ihr Prozess keinen automatischen Neustart nach Aus- und Einschalten zulässt, müssen Sie einen Neustartschutz in Ihrem Sicherheitsprogramm programmieren. Die sicherheitsgerichteten Prozessdatenausgänge müssen bis zur manuellen Quittierung blockiert werden. Diese Sicherheitsausgänge dürfen erst dann aktiviert werden, wenn dies sicher ist und Fehler korrigiert worden sind.

2.10 Austausch von Komponenten der AC500-S-Sicherheitssteuerung

Wenn Sie für die Software Ihrer Programmiereinheit oder Ihres PCs eine neuere Version installieren, beachten Sie Hinweise zur Auf- und Abwärtskompatibilität in der Dokumentation und die Readme-Dateien für diese Produkte.

Hardwarekomponenten für AC500-S (Sicherheits-CPU und Sicherheits-E/As) werden genau wie bei Standard-AC500-Automationssystemen ausgetauscht.

2.11 Umweltgerechte Entsorgung

Sämtliche Sicherheitskomponenten der Serie AC500-S von ABB wurden so entwickelt, dass sich die umweltschädliche Wirkung auf ein Minimum beschränkt. Für eine umweltgerechte Entsorgung können die AC500-S-Sicherheitskomponenten teilweise auseinandergenommen und getrennt entsorgt werden. Dafür ist die gültige nationale und internationale Gesetzgebung zu beachten.

2.12 Sichere Kommunikation

Sicherheitsdaten werden mithilfe des PROFIsafe-Profils von Sicherheits-CPU zu Sicherheits-E/As übertragen ↬ *[2]*. Die Sicherheits-CPU SM560-S / SM560-S-FD-1 / SM560-S-FD-4 benötigt eine Standard-CPU, um mit Sicherheits-E/A-Modulen zu kommunizieren. Die gesamte sicherheitsbezogene Kommunikation erfolgt über die Standard-CPU mithilfe des „Black-Channel“-Prinzips zur Datenübertragung ↬ *[2]*.

Die Kommunikation der Sicherheits-CPU mit Sicherheits-E/A-Modulen erfolgt über einen PROFINET IO Feldbus mit einem PROFIsafe-Profil zur sicheren Datenübertragung ↬ *[2]*. Sicherheits- und Standard-E/A-Module können auf einem lokalen I/O-Bus sowohl bei zentraler als auch dezentraler Konfiguration kombiniert werden. Das PROFINET IO-Controller-Kommunikationsmodul (CM579-PNIO) wird in Verbindung mit den Standard-CPUs als Teil des „Black Channel“ zur Übertragung von Sicherheitsdaten an die PROFINET IO-Geräte verwendet. Die PROFINET-Geräte CI501-PNIO, CI502-PNIO, CI504-PNIO und CI506-PNIO können zum Anschluss der Sicherheits-E/A-Module in dezentralen Konfigurationen verwendet werden.

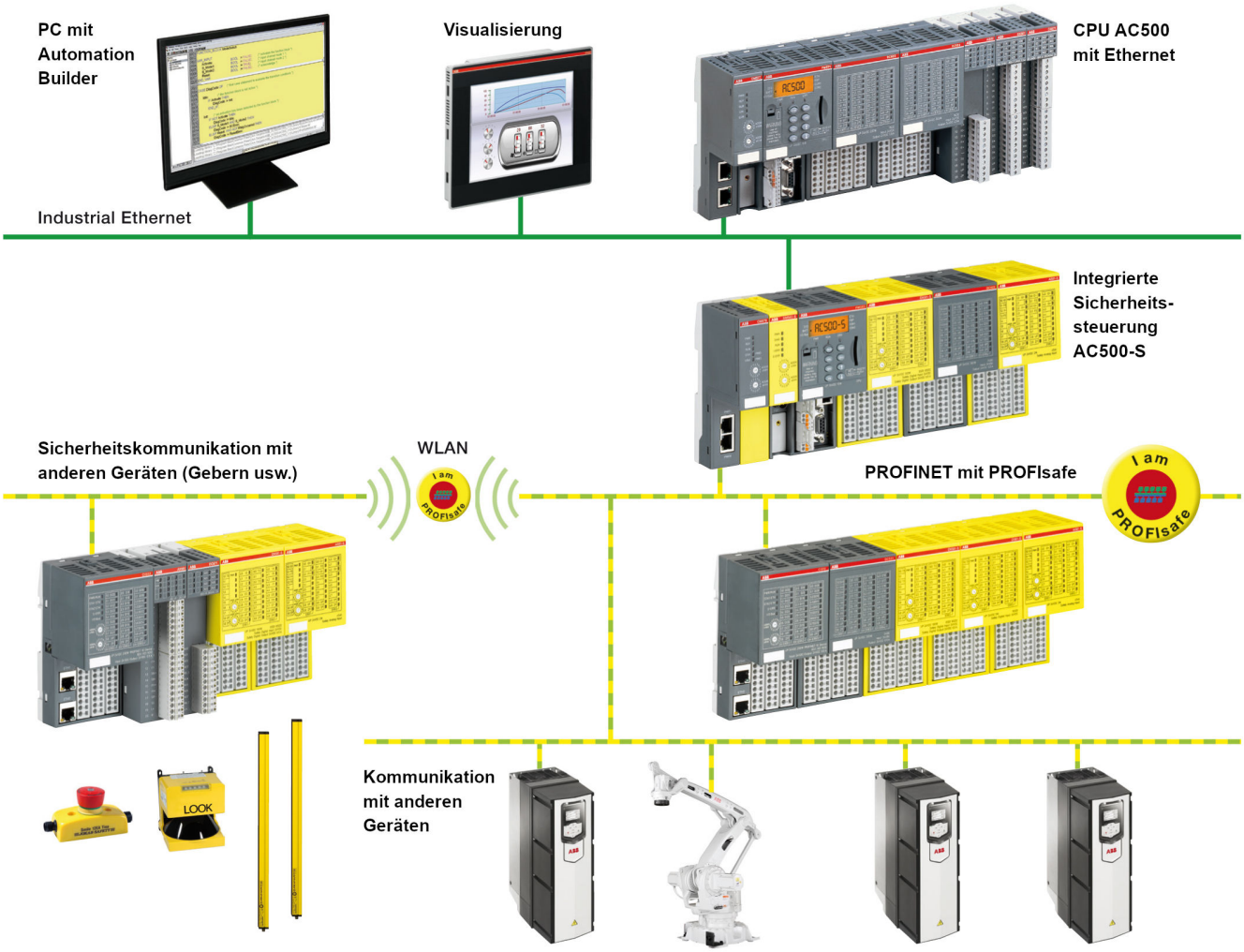


Abb. 3: AC500-S-Systemaufbau mit PROFINET/PROFIsafe für im Netzwerk verteilte Sicherheits-E/As, Sensoren und Aktoren

Die PROFINET/PROFIsafe-Kommunikation zwischen AC500-S-Sicherheits-CPUs wird bei Verwendung von PROFINET IO-Device-Kommunikationsmodulen CM589-PNIO und/oder CM589-PNIO-4 gemeinsam mit Sicherheits-CPUs SM560-S-FD-1 und/oder SM560-S-FD-4 mit F-Device-Funktionalität auf einer Seite und CM579-PNIO mit einer AC500-S Sicherheits-CPU mit F-Host-Funktionalität auf der anderen Seite unterstützt (Abb. 4, Seite 25). Die Sicherheits-CPUs SM560-S-FD-1 und SM560-S-FD-4 können durch Konfigurieren von bis zu 32 F-Submodulen eine große Menge Sicherheitsdaten mit F-Hosts austauschen (auch PROFIsafe F-Hosts von Drittanbietern werden unterstützt), wenn PROFINET / PROFIsafe verwendet wird.

Bei Verwendung von PROFIsafe F-Submodulen mit kurzen Telegrammlängen (unterstützt für PROFIsafe V2.4 and V2.6) können bis zu 384 Bytes ausgetauscht werden (max. 32 F-Device-Instanzen mit 12 Bytes Sicherheitsdaten für jede Ein-/Ausgangsrichtung).

Bei Verwendung von PROFIsafe F-Submodulen mit langen Telegrammlängen (nur unterstützt für PROFIsafe V2.6) können bis zu 1353 Bytes ausgetauscht werden (max. 11 F-Device-Instanzen mit 123 Bytes Sicherheitsdaten für jede Ein-/Ausgangsrichtung).

SM560-S-FD-1 mit F-Device(s) unterstützt die sichere Kommunikation mit maximal einem F-Host. SM560-S-FD-4 mit F-Device(s) unterstützt die sichere Kommunikation mit maximal vier F- Hosts. Abb. 4 zeigt, dass man mit den Sicherheits-CPUs SM560-S-FD-1 und SM560-S-FD-4 mit zusätzlicher F-Device-Funktionalität eine sichere Kommunikation von CPU zu CPU zwischen verschiedenen Steuerstationen an PROFINET/PROFIsafe einrichten kann. Die Sicherheits-CPUs SM560-S-FD-4 können gleichzeitig nicht nur mit 1 PROFINET IO-Controller / F-Host (Master), sondern mit bis zu 4 PROFINET IO-Controllern / F-Hosts (Mastern) kommunizieren. Zusätzlich zu den Sicherheits-CPUs SM560-S-FD-1 und SM560-S-FD-4 werden CM589-PNIO bzw. CM589-PNIO-4 PROFINET IO-Device-Kommunikationsmodule für die PROFINET-Konnektivität als „Black Channel“ zu 1 oder bis zu 4 PROFINET IO-Controllern benötigt.

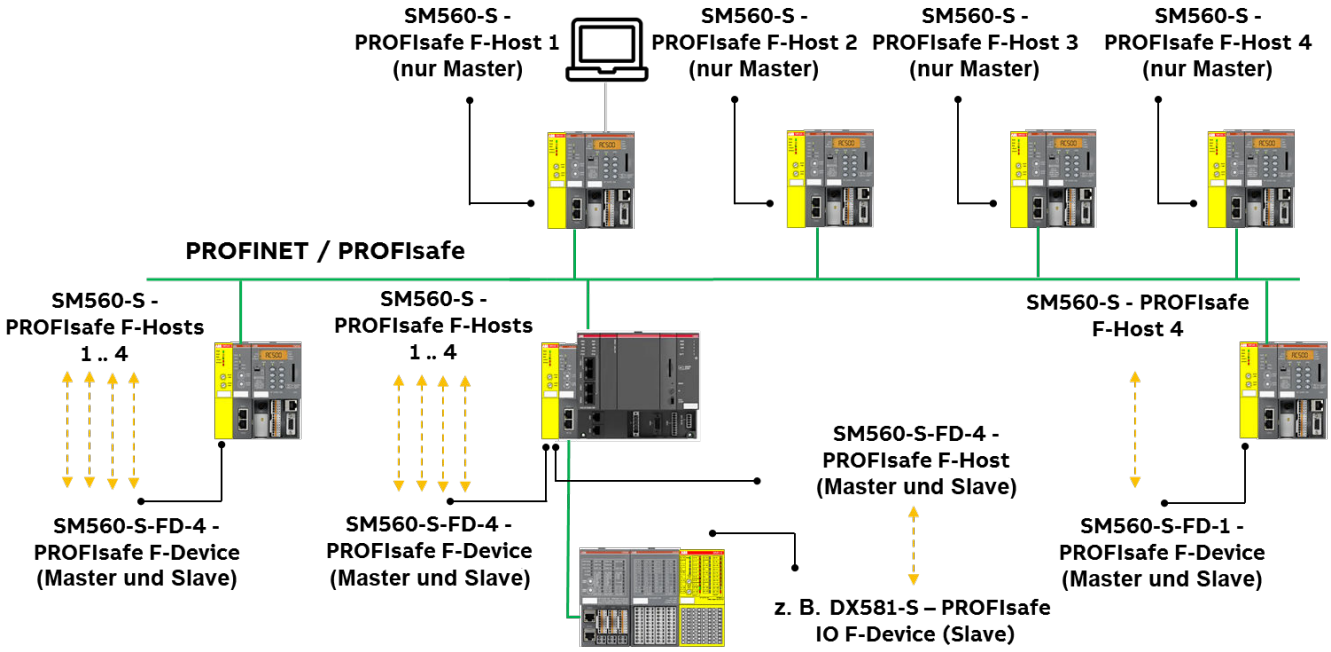


Abb. 4: Beispielhafter Aufbau für eine sichere Kommunikation von CPU zu CPU zwischen verschiedenen Sicherheits-CPUs (SM560-S / SM560-S- FD-1 / SM560-S-FD-4)

Die folgenden Kommunikationsanforderungen sollen für die Verwendung der-AC500-S-Sicherheitssteuerung erfüllt werden:

- Sicherheitsdaten dürfen nicht über öffentliche Netzwerke, z. B. das Internet, übertragen werden. Wenn Sicherheitsdaten über Firmen-/Fabriknetzwerke übertragen werden, stellen Sie sicher, dass es einen ausreichenden Schutz gegen Manipulation gibt (Firewall oder Router für Netzwerktrennung).
- Die an die Kommunikationsgeräte angeschlossenen Geräte müssen über eine sichere galvanische Trennung verfügen.

HINWEIS!

AC500-S-Sicherheits-E/A-Module und SM560-S-FD-1- / SM560-S-FD-4-Sicherheits-CPUs können mit F-Hosts von Drittanbietern im PROFINET verwendet werden. Gültige GSDML-Dateien von ABB können in der F-Host-Entwicklungsumgebung von Drittanbietern installiert werden und befinden sich zum Download auf www.abb.com/plc.

Danach können Sie AC500-S-Module mit einem F-Host von Drittanbietern konfigurieren und verwenden. Bitte wenden Sie sich an den technischen Support von ABB, um die F_iPar_CRC-Werte der AC500-S-Sicherheits-E/A-Module für F-Hosts von Drittanbietern zu erhalten.

Stellen Sie mithilfe von geeigneten Funktionsprüfungs- und Verifizierungsverfahren sicher, dass alle iParameter (Eingangsverzögerung, Kanalkonfiguration usw.) für alle AC500-S-Sicherheits-E/As und anderen F-Devices mit einem gegebenen F_iPar_CRC-Wert korrekt sind ↬ *Kapitel 6.5 „Verifizierung einer sicheren iParameter-Einstellung in den AC500-S-Sicherheits-E/As“ auf Seite 382*.

Die in GSDML-Dateien für AC500-S-Sicherheits-E/A-Module verwendeten F_iPar_CRC-Standardwerte stimmen nicht mit den iParameter-Standardkonfigurationen für AC500-S-Sicherheits-E/A-Module überein und müssen vor der Verwendung in den Konfigurations-Werkzeugen neu berechnet werden. Auf diese Weise wird die nicht bestimmungsgemäße Verwendung von AC500-S-Sicherheits-E/A-Modulen mit F-Hosts von Drittanbietern verhindert.

2.13 Sicherheitsfunktion und Reaktion auf Fehler

Die Hauptsicherheitsfunktion der AC500-S-Sicherheitssteuerung ist es, Digital- und Analogeingänge zu lesen, um die sicherheitsgerichteten Digitalausgänge gemäß dem vom Anwender laut IEC 61131 definierten Anwendungsprogramm und der Konfiguration durch das Sicherheitslogikmodul der Sicherheits-CPU zu steuern.

Die AC500-S-Sicherheitssteuerung kann nach dem Ruhestromprinzip („de-energize to trip“) verwendet werden. Der sichere Zustand der Ausgänge wird gemäß der folgenden Tabelle definiert:

Tab. 2: Verhalten des Ruhestrom-Sicherheitssystems

	Ruhestromprinzip
Modus gemäß IEC 61508	Hohe Anforderungsrate oder niedrige Anforderungsrate
Sicherheitsfunktion	Ausschalten der Ausgänge
Sicherer Zustand	Abgeschaltete Ausgänge

Der Zweck der AC500-S-Sicherheitsfunktion ist es, den sicheren Zustand der Maschine oder des Prozesses (als System) bei gegebenem SIL (IEC 61508 und IEC 61511), max. SIL (IEC 62061) und PL (ISO 13849-1) zu garantieren. Eine Beispiel-Sicherheitsfunktion auf Anwendungsebene, die von einer AC500-S in Maschinenanwendungen ausgeführt werden kann, ist der Not-Halt.

2.13.1 Sicherheits-CPU (SM560-S / SM560-S-FD-1 / SM560-S-FD-4)

Die Sicherheitsfunktion der Sicherheits-CPU ist es, Signalinformationen korrekt zu verarbeiten. Sie verarbeitet Signale der Sicherheitseingänge und die interne Datenspeicherung, um Signale für Sicherheits-Ausgabemodule zu generieren und ihrem internen Datenspeicher einen neuen Zustand zuzuweisen.

Wenn diese Funktion nicht korrekt ausgeführt werden kann, wechselt die Sicherheits-CPU in einen SAFE-STOP-Zustand, in dem keine gültigen Sicherheitstelegramme generiert werden und infolgedessen alle sicheren Ausgangskanäle abgeschaltet werden („0“-Zustand), nachdem die Watchdog-Zeit abgelaufen ist.

Fehler in der zyklischen Kommunikation zwischen Sicherheits-CPU und Sicherheits-E/A-Modulen oder anderen F-Devices, z. B. den Sicherheits-CPUs SM560-S-FD-1 oder SM560-S-FD-4, werden von der Sicherheits-CPU festgestellt. Daraufhin werden „0“-Werte an das Sicherheitsprogramm übergeben.

Der Entwickler des Anwendungsprogramms muss eine spezielle Reaktion im Fehlerfall einbauen, z. B. das Abschalten der sicheren Ausgangskanäle („0“-Zustand), sofern erforderlich.

2.13.2 Sicherheitsmodule mit sicheren Eingangskanälen (DI581-S, DX581-S und AI581-S)

Die Sicherheitsfunktion der Sicherheitsmodule (DI581-S, DX581-S und AI581-S) mit digitalen und analogen Eingangskanälen ist es, externe analoge und/oder digitale Signale korrekt zu lesen. Wenn diese Funktion nicht korrekt ausgeführt werden kann, wird das Sicherheitsmodul oder nur sein Eingangskanal (je nach Fehler) in den sicheren Zustand geschaltet. Bei einem Kanalfehler wird der sichere Wert (abgeschaltet = „0“) zusammen mit zusätzlichen Informationen über den entsprechenden Kanalfehler an das Sicherheitslogikmodul (z. B SM560-S) übertragen.

Bei einem Modulfehler werden keine gültigen Telegramme vom Sicherheitsmodul zum Sicherheitslogikmodul generiert. Die Werte dieser sicheren Eingangskanäle werden sicheren Werten (abgeschaltet = „0“) auf der Sicherheits-CPU zugeordnet.

Fehler in der zyklischen Kommunikation zwischen Sicherheits-CPU und Sicherheitsmodulen werden von den Sicherheitsmodulen mit Eingangskanälen festgestellt. Wenn ein Kommunikationsfehler auftritt, gehen alle Eingänge des entsprechenden Sicherheitsmoduls in den sogenannten Zustand der Passivierung, in dem „0“-Werte als Prozesswerte gesendet werden, sobald die Kommunikation mit der Sicherheits-CPU wieder hergestellt ist. Das Umschalten (Reintegration) von „0“-Sicherheitswerten zu Prozessdaten erfolgt erst nach Quittierung durch den Anwender.

2.13.3 Sicherheitsmodule mit sicheren Ausgangskanälen (DX581-S)

Die Sicherheitsfunktion der Sicherheitsmodule (DX581-S) mit sicheren Ausgangskanälen ist es, ihre Ausgangssignale korrekt zu schreiben. Wenn diese Funktion nicht korrekt ausgeführt werden kann, werden das Sicherheitsmodul oder seine Ausgangskanäle (je nach Fehler) in den sicheren Zustand geschaltet. Bei einem Kanalfehler wird der sichere Wert (abgeschaltet = „0“) für die entsprechenden sicheren Ausgangskanäle gesetzt. Bei einem Modulfehler werden keine gültigen Telegramme vom Sicherheits-Ausgabemodul zur Sicherheits-CPU generiert. Den Werten aller sicheren Ausgangskanäle werden sicheren Werten (abgeschaltet = „0“) zugeordnet.

Fehler in der zyklischen Kommunikation zwischen Sicherheits-CPU und sicheren Ausgabemodulen werden von dem sicheren Ausgabemodul DX581-S erkannt. Bei einem Kommunikationsfehler werden alle Ausgänge des entsprechenden sicheren Ausgabemoduls abgeschaltet („0“). Das Umschalten (Reintegration) von „0“-Sicherheitswerten zu Prozessdaten erfolgt erst nach Quittierung durch den Anwender, sobald die Kommunikation wiederhergestellt wurde.

2.14 Sicherheitsfunktionstest

Nach Erstellen des Sicherheitsprogramms und der Systemkonfiguration muss ein kompletter Funktionstest für die entsprechende Automations-Task durchgeführt werden. Bei Veränderungen eines Sicherheitsprogramms, für das bereits ein kompletter Funktionstest durchgeführt wurde, müssen nur die Veränderungen getestet werden, sofern zuvor eine korrekte Untersuchung der Auswirkungen durchgeführt wurde.

Das Sicherheitsprogramm, die Sicherheits-E/A-Konfiguration usw. müssen verifiziert, ausgedruckt, für den Projektdatenbericht gespeichert und archiviert werden. Auf den Sicherheitsfunktionstest folgt die Systemabnahme. Nach der Konfiguration der Hardware und Zuordnung der Parameter für Sicherheits-CPU und Sicherheits-E/A-Module können Sie einen Abnahmetest durchführen. Während des Abnahmetests des Systems müssen alle anwendungsspezifischen Normen befolgt werden.

2.15 Fehlerbehebung

Fehlermeldungen im Diagnosepuffer der Standard-CPU enthalten eine Beschreibung, mit der Sie mögliche Probleme der AC500-S-Konfiguration beheben können. Wenn manche der Probleme weiter bestehen und es keine Fehlermeldungen im Diagnosepuffer gibt, wenden Sie sich an den technischen Support von ABB für weitere Angaben.

HINWEIS!

Stellen Sie sicher, dass die Sicherheits-E/A-Module korrekt mit einem guten elektrischen Kontakt am Klemmenblock TU582-S angeschlossen sind, um unerwünschte Systemzustände mit möglicherweise fehlerhaften LED-Anzeigen zu vermeiden ↳ *Kapitel 3.3.3 „Montage, Abmessungen und elektrischer Anschluss“ auf Seite 76* ↳ *Kapitel 3.4.3 „Montage, Abmessungen und elektrischer Anschluss“ auf Seite 103* ↳ *Kapitel 3.5.3 „Montage, Abmessungen und elektrischer Anschluss“ auf Seite 124*.

Unten finden Sie eine Liste der bekannten Probleme und Lösungen im Zusammenhang mit den Komponenten der AC500-S-Sicherheitssteuerung:

ID	Verhalten	Mögliche Ursache	Abhilfe
1.	Die Sicherheits-CPU ist im Zustand RUN oder DEBUG RUN, aber alle Sicherheits-E/A-Module schalten plötzlich in den Zustand RUN (Modulpassivierung).	Ihr Programm enthält möglicherweise eine Endlosschleife, weshalb die Sicherheits-CPU keine gültigen Sicherheitstelegramme in einer angemessenen Zeit (innerhalb der konfigurierten Watchdog-Zeit) an die Sicherheits-E/A-Module senden kann.	Überprüfen (debuggen) Sie Ihr Sicherheitsprogramm und stellen Sie sicher, dass es keine Endlosschleife(n) gibt.
2.	Aus dem Sicherheitsprojekt ist kein Login bei der Sicherheits-CPU möglich.	Die Visualisierung wurde direkt an die Sicherheits-CPU angeschlossen, was die Verbindung zur Sicherheits-CPU blockiert. Es ist jeweils nur eine Verbindung zur Sicherheits-CPU zulässig.	Trennen Sie die Visualisierung von der Sicherheits-CPU.
3.	Während des Schließens oder Speicherns des Projekts, des Änderns des Sicherheitsprojekts usw. mit Automation Builder kommt eventuell keine Reaktion vom Automation Builder und/oder vom Sicherheitsprojekt. Es ist, als hätte sich die Anwendung aufgehängt.	Die Benutzerverwaltung des Automation Builder fordert Sie auf, Ihre Login-Informationen der Sicherheitskomponenten zu bestätigen; diese Meldung ist nicht im Vordergrund. Ihre vorherige Sitzung ist abgelaufen.	Suchen Sie die Login-Meldung im Hintergrund Ihres Windows-Desktops, loggen Sie sich ein und fahren Sie mit Ihren vorherigen Tätigkeiten fort. Stellen Sie eine längere Sitzungsdauer für Automation Builder ein, wenn sich dieses Ereignis wiederholt ↳ *[3]*.
4.	Ihr sicherheitsgerichteter Digitaleingangskanal wurde mit einer internen Fehlerdiagnosemeldung auf einer Standard-CPU zeitweise passiviert. Mit AC500 V2-Standard-CPU: Fehlerschwere: E3, Komponente: 14, Gerät: 1 … 10, Modul: 31, Kanal: 31, Fehler: 43 Mit AC500 V3-Standard-CPU: Fehlerschwere: 3, Fehlercode: 16171	Ein möglicher Grund ist, dass Ihre Eingangssignalfrequenz eine zulässige Eingangskanal-Signalfrequenz überschritten hat ↳ *erlaubte Frequenzbereiche: Kapitel 3.3.2 , Seite 71*.	Stellen Sie sicher, dass Ihr Eingangssignal nicht die zulässige Eingangssignalfrequenz überschreitet.
5.	Das Modul DX581-S ist eingeschaltet, aber an die UP-Klemme des Moduls DX581-S ist keine Spannungsversorgung angeschlossen.	Verdrahtungsfehler im Modul DX581-S, wenn +24 V DC an mindestens einer der sicherheitsgerichteten Digitalausgangsklemmen des DX581-S anliegt. Dies führt dazu, dass das DX581-S über sicherheitsgerichtete Digitalausgänge eingeschaltet wird.	Überprüfen Sie die Verdrahtung des DX581-S und trennen Sie +24 V DC von der/den sicherheitsgerichteten Digitalausgangsklemme(n).
6.	Einige Kanäle eines Sicherheits-E/A-Moduls oder ein komplettes Modul sind zwischenzeitlich ohne Grund (korrekte Verdrahtung usw.) passiviert worden.	Es gibt keinen zuverlässigen elektrischen Kontakt zwischen dem Sicherheits-E/A-Modul und dem Klemmenblock TU582-S.	Stellen Sie sicher, dass Sie das Sicherheits-E/A-Modul mit einer Kraft von mind. 100 N in den Klemmenblock TU582-S drücken, wie es in den Checklisten für AC500-S angegeben ist.
7.	Mit einer größeren Zahl Sicherheits-E/A-Module im System braucht die Sicherheits-CPU länger, um den Befehl *„Bootprojekt erzeugen“* auszuführen.	Die Sicherheits-CPU ist ein „Single-Threaded“-System. Je mehr Sicherheits-E/A-Module sich in einem System befinden, desto höher ist die interne Zykluszeit der Sicherheits-CPU, um die relevanten Daten der sicheren Ein- und Ausgänge zu verarbeiten.	Aktuell gibt es nur eine Möglichkeit, dieses Verhalten zu ändern, und zwar eine Aufteilung der sicheren Ein- und Ausgänge auf verschiedene Sicherheits-CPUs, damit jede einzelne weniger sichere Ein- und Ausgänge verarbeiten muss.

ID	Verhalten	Mögliche Ursache	Abhilfe
8.	Nach dem Login bei der Sicherheits-CPU unter Verwendung von AC500-S Programming Tool wird eine lange Liste mit internen Konstanten in grüner Schrift für PROFIsafe F-Host-Instanzen angezeigt.	Die Option *„Konstanten ersetzen“* wurde ausgewählt.	Gehen Sie im AC500-S Programming Tool-Menü zu *„Projekt ➔ Optionen ➔ Aufbau“*. Entfernen Sie die Markierung der Option *„Konstanten ersetzen“*.
9.	Es kann kein gültiges Sicherheitsprojekt generiert werden (PROFIsafe Callback-Funktionen fehlen und kein Sicherheits-E/A-Abbild wird erstellt).	Ein möglicher Grund ist, dass Sie unter *„Objekteigenschaften … ➔ Zugriffsrechte“* für eine der POEs im Verzeichnisbaum des Sicherheitsprojekts die folgende Option ausgewählt haben: *„Kein Zugriff“* oder *„Lesezugriff“* für alle *„Benutzergruppen“* mit der Auswahl *„Für alle übernehmen“*.	Starten Sie das Sicherheitsprojekt, loggen Sie sich ein und gehen Sie zu *„Objekteigenschaften … ➔ Zugriffsrechte“* für eine der POEs im Verzeichnisbaum des Sicherheitsprojekts, um *„Vollzugriff“* für eine der Benutzergruppen einzustellen, und wählen Sie dann *„Für alle übernehmen“*. Danach können Sie den Befehl *„Sicherheits-Konfigurationsdaten erzeugen“* für Ihr Sicherheitsprojekt im Automation Builder erfolgreich wiederholen.
10.	Bei der Anwahl des FB CurTimeEx aus der Bibliothek Safety_SysLibTime.lib werden immer „0“-Werte an den Ausgängen zurückgegeben.	Der FB CurTimeEx ist in der aktuellen Version der Sicherheits-CPU nicht implementiert und für zukünftige Verwendung reserviert.	Verwenden Sie den FB CurTimeEx nicht in Ihrem Sicherheitsprogramm.

ID	Verhalten	Mögliche Ursache	Abhilfe
11.	● Setzen Sie den Parameter „Debug-Modus aktivieren“ auf der Sicherheits-CPU auf *„Aus“*. ● Erzeugen Sie Bootprojekte für die Sicherheits-CPU und die Standard-CPU. ● Führen Sie einen Power Cycle aus. ● Vergleichen Sie die CRCs der Bootprojekte auf Ihrem PC und der Sicherheits-CPU. Der Vergleich zeigt, dass sie identisch sind, was in Ordnung ist. ● Versuchen Sie, ein Bootprojekt für die Sicherheits-CPU zu erzeugen. Eine Fehlermeldung wird angezeigt, da „Debug-Modus aktivieren“ für die Sicherheits-CPU *„Aus“* ist; dies ist in Ordnung. ● Wiederholen Sie den Vergleich von Bootprojekt-CRCs auf Ihrem PC und der Sicherheits-CPU. Jetzt wird gemeldet, dass sie nicht gleich sind (Bootprojekt-CRC für die Sicherheits-CPU wird als CDCDCDCD angezeigt), was irreführend sein kann, da das Bootprojekt auf der Sicherheits-CPU nicht verändert wurde.	AC500-S Programming Tool unterstützt den beschriebenen Anwendungsfall nicht.	Nach einem Power Cycle der Sicherheits-CPU wird jetzt die korrekte Bootprojekt-CRC für die Sicherheits-CPU angezeigt.
12.	Der serielle Treiber wird verwendet, um den Anschluss zur Sicherheits-CPU herzustellen. In AC500-S Programming Tool werden kurz hintereinander der *„Login“*- und der *„Logout“*-Befehl ausgeführt, gefolgt von einer erneuten Ausführung des *„Login“*-Befehls. Nach dem zweiten Login-Versuch wird ein Kommunikationsfehler in AC500-S Programming Tool angezeigt.	Der serielle Treiber hat nicht genügend Zeit, um erneut initialisiert zu werden.	Warten Sie mindestens 20 Sekunden, bevor Sie einen erneuten *„Login“*-Befehl nach einem *„Logout“* eingeben.

ID	Verhalten	Mögliche Ursache	Abhilfe
13.	● Der Befehl *„Login“* wird zum Anmelden in AC500-S Programming Tool ausgeführt; danach wird der SPS-Browser-Befehl *„setpwd“* eingegeben, um ein neues Passwort zu definieren, z. B. „PWD1“ für die Sicherheits-CPU. ● Für die Sicherheits-CPU wird ein Power Cycle ausgeführt, aber das Fenster von AC500-S Programming Tool bleibt auf dem Endanwender-PC offen. ● Der *„Login“*-Befehl wird zum Anmelden ausgeführt und das in Schritt 1 neu eingerichtete Passwort „PWD1“ wird eingegeben. Der SPS-Browser-Befehl *„setpwd“* wird eingegeben, um ein neues Passwort für die Sicherheits-CPU zu definieren, z. B. „PWD2“. ● Für die Sicherheits-CPU wird ein Power Cycle ausgeführt, aber das Fenster von AC500-S Programming Tool bleibt auf dem Endanwender-PC offen. ● Der *„Login“*-Befehl wird zum Anmelden ausgeführt und die Fehlermeldung „Sie haben ein falsches Passwort für die SPS eingegeben“ wird angezeigt. Nach dem Klicken auf die Schaltfläche *„OK“* können Sie noch das neue Passwort „PWD2“ eingeben und sich erfolgreich an der Sicherheits-CPU anmelden.	AC500-S Programming Tool versucht, sich mit einem alten Passwort bei der Sicherheits-CPU anzumelden.	Nach dem Zurücksetzen des Passworts der Sicherheits-CPU schließen Sie AC500-S Programming Tool und öffnen Sie das Tool erneut. Die Fehlermeldung erscheint nicht mehr.
14.	Nach dem Einschalten geht das Sicherheits-E/A-Modul in den Zustand SAFE STOP und beide Fehler-LEDs leuchten.	Der im Automation Builder-Projekt konfigurierte F_Dest_Add-Wert stimmt nicht mit dem PROFIsafe-Adress-Schalter-Wert des Sicherheits-E/A-Moduls überein.	Stellen Sie sicher, dass der im Automation Builder-Projekt konfigurierte F_Dest_Add-Wert mit dem PROFIsafe-Adress-Schalter-Wert des Sicherheits-E/A-Moduls übereinstimmt.
15.	Es ist kein Login an der Sicherheits-CPU möglich.	Falsche *„Kommunikationsparameter“*-Einstellungen werden verwendet.	Stellen Sie sicher, dass in AC500-S Programming Tool die korrekten *„Kommunikationsparameter“*-Einstellungen für die Verbindung zur Sicherheits-CPU verwendet werden.
16.	Nachdem das Bootprojekt in die Sicherheits-CPU geladen wurde, scheint die V2-Standard-CPU manchmal 45 Sekunden lang nichts zu tun, bis ihre Fehler-LED zu leuchten beginnt.	Zeitüberschreitung bei der V2-Standard-CPU.	Diese Situation ist sehr selten. Aktuell gibt es für dieses Verhalten der V2-Standard-CPU keine Abhilfe.

ID	Verhalten	Mögliche Ursache	Abhilfe
17.	Nach dem Einschalten der Sicherheits-CPU kann es vorkommen, dass diese nicht in den Modus RUN schaltet. Die Diagnose-LED leuchtet und kein Bootprojekt wird in die Sicherheits-CPU geladen. Wenn Sie versuchen, sich in die Sicherheits-CPU einzuloggen, wird in AC500-S Programming Tool die Fehlermeldung „Kein Programm im Controller! Soll das neue Programm geladen werden?“ angezeigt.	Die Spannungseinbruch-Funktion der Sicherheits-CPU wird ausgelöst, wenn die Pause zwischen Aus- und Einschalten weniger als 1,5 s beträgt. Das Bootprojekt befindet sich noch immer in der Sicherheits-CPU, ist jedoch aufgrund der Erkennung eines Spannungseinbruchs nicht geladen. Daher ist es nicht notwendig, ein Bootprojekt neu in die Sicherheits-CPU zu laden.	Schalten Sie die Sicherheits-CPU mit einer Pause zwischen Aus- und Einschalten von mehr als ≥ 1,5 s aus und wieder ein.
18.	Wenn während des Debug-Prozesses der Breakpoint erreicht wird und Sie eine Variable zu forcieren versuchen, wird diese erst im nächsten Zyklus der Sicherheits-CPU mit dem forcierten Wert aktualisiert.	Die Sicherheits-CPU ist ein „Single-Threaded“-System.	Dieses Verhalten ist beabsichtigt.
19.	Wenn das Projekt in die Sicherheits-CPU geladen wird, bleibt der geladene Code bei 0 Byte im Download-Fenster stehen oder eine Fehlermeldung wird angezeigt.	Der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU ist *„AUS“* und diese Konfigurationsdaten wurden auf die Standard-CPU heruntergeladen.	Setzen Sie den Parameter „Debug-Modus aktivieren“ auf *„EIN“*, generieren Sie eine neue Konfiguration und laden Sie das Projekt in die Standard-CPU. Der neue Projektcode kann über AC500-S Programming Tool in die Sicherheits-CPU geladen werden.
20.	Nach dem Ausloggen ist das Einloggen in die Sicherheits-CPU nicht möglich.	Zu schnelles Einloggen in die Sicherheits-CPU nach dem Ausloggen.	Warten Sie nach dem Ausloggen von der Sicherheits-CPU einige Sekunden (~ 5–10 s), bevor Sie sich in die Sicherheits-CPU einloggen.
21.	Die Diagnosemeldung mit Fehlerschweregrad 3 und dem Fehlertext „Messwertunterschreitung am E/A-Modul“ erscheint im Diagnosesystem der Standard-CPU, obwohl es für den gegebenen Eingangskanal AI581-S einen Überstrom und keinen Unterstrom gab.	Der interne Erkennungsmechanismus kann nicht immer zwischen Über- und Unterstrom unterscheiden, weil in der Elektronik von AI581-S auf einen Überstrom oft ein Unterstrom folgt.	Bis jetzt gibt es für dieses Problem keine Abhilfe.
22.	Der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU ist *„EIN“* und korrekt in die Standard-CPU geladen worden. Ein Debugging der Sicherheits-CPU ist jedoch immer noch nicht möglich.	Die Sicherheitsprojekte auf Ihrem PC und in der Sicherheits-CPU sind nicht dieselben. Möglicherweise wird die folgende Meldung angezeigt: „Das Programm wurde geändert! Soll das neue Programm geladen werden?“.	Laden Sie Ihr Sicherheitsprojekt von Ihrem PC auf die Sicherheits-CPU, dann wird das Debugging möglich sein.

ID	Verhalten	Mögliche Ursache	Abhilfe
23.	Nach Verwendung des Menüpunkts *„Online ➔ Reset“* in AC500-S Programming Tool geht die Sicherheits-CPU in den Zustand DEBUG STOP (nicht sicher). Die Sicherheits-E/A-Module werden passiviert. Wenn Sie sich in der Sicherheits-CPU einloggen, können Sie OA_Req_S = TRUE Bits in PROFIsafe-Instanzen der F-Devices sehen. Die Sicherheitsanwendung wird von der Sicherheits-CPU nicht ausgeführt, aber Sie können weiterhin OA_C = TRUE für die F-Devices setzen, woraufhin diese in den Modus RUN schalten. Die Sicherheits-CPU verbleibt die ganze Zeit im Zustand DEBUG STOP (nicht sicher).	Der PROFIsafe F-Host läuft nach Verwendung des Menüpunkts *„Online ➔ Reset“* nicht im Failsafe-Modus.	Dieses Verhalten ist bei der Sicherheits-CPU beabsichtigt ↳ *Kapitel 3.1.5.1 „Beschreibung der Zustände der Sicherheits-CPU“ auf Seite 51*.
24.	Die Fehlermeldung „Fehler in den Konfigurationsdaten, die Sicherheitssteuerung kann die Konfigurationsdaten nicht lesen“ ist auf der Sicherheits-CPU verfügbar.	• Heruntergeladene Konfigurationen von Standard-CPU und Sicherheits-CPU passen nicht zusammen. • Kein Bootprojekt wird auf die Sicherheits-CPU geladen.	• Laden Sie gültige Konfigurationen als Teil von Bootprojekten auf die Standard-CPU bzw. Sicherheits-CPU und stellen Sie sicher, dass diese zusammenpassen. • Laden Sie ein gültiges Bootprojekt auf die Sicherheits-CPU.

2.16 FAQ – AC500-S-Sicherheitssteuerung

- **Verfügbarkeit eines Bootprojekts auf der Sicherheits-CPU nach Spannungsabfall oder unvollständigem „Power Cycle“**
  Bei Unter- oder Überspannung, die auch durch einen unvollständigen „Power Cycle“ (Abschalten und Wiedereinschalten in unter 1,5 s) bedingt sein kann, geht die Sicherheits-CPU in den Zustand SAFE STOP und die LED I-ERR leuchtet. Dennoch ist das Bootprojekt weiter intakt. Um die Sicherheits-CPU wieder in den Modus RUN zu bringen, müssen zwei aufeinanderfolgende Power Cycles durchlaufen werden. Nach dem ersten Power Cycle geht die Sicherheits-CPU in den Zustand DEBUG STOP (nicht sicher) und die LED DIAG leuchtet. Der zweite Power Cycle versetzt die Sicherheits-CPU wieder in den Modus RUN (sicher).
- **Es ist unmöglich, ein Bootprojekt für die Sicherheits-CPU zu erstellen**
  Prüfen Sie, ob der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU im Automation Builder-Projekt „EIN“ ist, das erzeugte Bootprojekt in die Standard-CPU geladen wurde und dies von einem Power Cycle gefolgt wurde.
- **Nach dem Power Cycle geht die Sicherheits-CPU in den Zustand SAFE STOP (I-ERR leuchtet)**
  Dies könnte bedingt sein durch ein korruptes Bootprojekt oder eine falsche Einstellung des Drehschalters der Sicherheits-CPU auf einen dieser Werte: 0xFE, 0xFD oder 0xFC. Eine weitere Möglichkeit ist, dass die Sicherheits-CPU zu kurz ausgeschaltet war. Um einen zuverlässigen Neustart sicherzustellen, muss die Abschaltzeit ≥ 1,5 s betragen.

- **Die Kanalreintegration des AI581-S-Sicherheitsmoduls ist nach Behebung des Fehlerzustands nicht möglich**

  Nur bei einer Kanalpassivierung durch Über- oder Unterstrom bleibt der analoge Sicherheitskanal 30 s lang passiviert, um die ursprünglichen Eigenschaften wiederherzustellen. Dann wird geprüft, ob der Fehlerzustand weiterhin vorliegt oder nicht. Liegt der Fehler nicht mehr vor, wird das Signal für die Reintegrationsanforderung für den entsprechenden Kanal auf TRUE gesetzt, sodass der Kanal wieder integriert werden kann. Während der zuvor genannten 30 s kann der analoge Sicherheitskanal nicht reintegriert werden.
- **Der Prozesswert bestimmter konfigurierter Eingänge ist immer FALSE (nur im 2-Kanal-Evaluierungsmodus)**

  Unsere Module sind so konstruiert, dass in einem 2-Kanal-Modus der niedrigere Kanal (z. B. Kanäle 0/4 ➔ Kanal 0, Kanäle 1/5 ➔ Kanal 1 etc. beim DX581-S-Modul) immer den gesammelten Prozesswert, das PROFIsafe-Diagnosebit, die Quittierungsanforderung und die Acknowledge-Reintegrationsinformation transportiert. Der höhere Kanal liefert immer den passivierten Wert „0“. Somit ist ein Namensabbild des höheren Kanals bei einem 2-Kanal-Evaluierungsmodus nicht erforderlich.
- **Der azyklische nicht sichere Datenaustausch dauert sehr lange.**

  Dieses Verhalten hängt von der Task-Konfigurationseinstellung der Standard-CPU ab. Passen Sie die Zykluszeit Ihrer Task an der Standard-CPU dort an (stellen Sie z. B. die Zykluszeit auf 1 ms), wo die Funktionsbausteine für den azyklischen nicht sicheren Datenaustausch programmiert werden, um die beste Leistung zu erzielen.
- **Wann sollte ich anstelle des azyklischen nicht sicheren Datenaustauschs den zyklischen nicht sicheren Datenaustausch verwenden?**

  Wenn 84 Bytes im azyklischen nicht sicheren Datenaustausch nicht ausreichen oder der Datenaustausch zu langsam ist, können Sie den zyklischen nicht sicheren Datenaustausch für den Austausch von Daten mit bis zu 2 kB mit minimalem Programmieraufwand verwenden.

  Für die meisten Sicherheitsanwendungen wird diese Funktion nicht benötigt und sollte daher auch nicht verwendet werden. Wenn Sie die Funktion dennoch benötigen, finden Sie weitere Informationen unter *Anhang B.5 „Datenaustausch zwischen Sicherheits-CPU und AC500 V2-Standard-CPU“ auf Seite 430* *Anhang C.5 „Datenaustausch zwischen Sicherheits-CPU und AC500 V3-Standard-CPU“ auf Seite 448*.
- **Ist die Datenkommunikation über den azyklischen bzw. zyklischen nicht sicheren Datenaustausch sicher?**

  Die Datenkommunikation über den azyklischen bzw. zyklischen nicht sicheren Datenaustausch ist nicht sicher, da sie nicht durch funktionale Sicherheitsmaßnahmen für die Datenkommunikation geschützt wird. Allerdings können Nutzer mit dem sogenannten „Black Channel“-Prinzip ihre eigenen Sicherheitsprofile auf diese nicht sichere Kommunikation implementieren. Für nähere Einzelheiten wenden Sie sich an den technischen Support von ABB.
- **Keine Erkennung von Übersprechen oder Kurzschlüsse auf 24 V DC für S-DOs des DX581-S. Warum und wie soll dieses Problem gelöst werden?**

  Die Ausgänge des DX581-S-Sicherheitsmoduls sind von der angeschlossenen Last entkoppelt. Dies ist erforderlich, um einen Einfluss der angeschlossenen Last auf den internen Prüfkreis zu vermeiden und somit eine hohe Robustheit zu garantieren (kein zufälliges Auslösen durch falsche Fehlererkennung aufgrund einer unerwarteten Änderung der elektrischen Eigenschaften der angeschlossenen Last). Daher können ein Übersprechen und ein Kurzschluss auf 24 V DC nur bis zur Ausgangsklemme des DX581-S-Sicherheitsausgangs erkannt werden, nicht aber an der zugehörigen Ausgangsleitung. Bei den meisten Kunden ist ein Fehlerausschluss durch Isolierung der Ausgangsleitung oder alternativ durch Neustart der Maschine (mit richtiger Startprüfmethode im Programm der Sicherheits-CPU für bestimmte S-DOs implementiert, um diese nacheinander zu aktivieren) mindestens einmal im Monat ausreichend. Der Anwender kann auch andere geeignete Maßnahmen treffen (z. B. durch Festlegung geeigneter Prüfzeiträume für die Sicherheitsfunktion oder durch Wiedergabe des Zustands der Ausgangsleitung mit einem sicherheitsgerichteten Digitaleingang), um die entsprechenden Anforderungen der IEC 62061 und ISO 13849-1 zu erfüllen, wenn ein Übersprechen oder ein Kurzschluss auf 24 V DC realisiert werden soll.

- **Ist mein Sicherheitsprogramm OK, wenn nicht alle von der AC500-S-Sicherheitscode-analyse (SCA) geprüften, sicherheitsgerichteten Programmierrichtlinien und -regeln erfüllt sind?**
  Das SCA-Werkzeug prüft nur, ob die statischen Sicherheitsprogrammierrichtlinien oder -regeln befolgt werden. Daher müssen vom SCA-Werkzeug identifizierte Fehler nicht unbedingt zu einer Maschinenstörung führen, sondern erfordern eine zusätzliche Begründung, warum diese Ausnahmen (nicht erfüllte Sicherheitsprogrammierrichtlinien oder -regeln) in der gegebenen Sicherheitsanwendung des Kunden erlaubt sind. Letzteres kann die Zertifizierung des Sicherheitsanwendungsprogramms des Kunden verzögern.
- **Was bedeutet die eingebaute Stromversorgung im Sicherheits-E/A-Modul?**
  Dies bedeutet, dass kein separates Stromversorgungsmodul für Sicherheits-E/As der AC500-S gekauft werden muss. 24 V DC können direkt über UP- und ZP-Anschlüsse am Klemmenblock angeschlossen werden.
- **Wie wirkt sich der Anschluss eines Testimpulses des gleichen Typs (z. B. T0, T1, T2, T3 etc.) von einem Modul zum sicherheitsgerichteten Digitaleingangskanal eines anderen Moduls aus? Sind die Testimpulse modulspezifisch?**
  Ja, die Testimpulse sind modulspezifisch. Da Testimpulse modulspezifisch sind, würde der Anschluss eines Testimpulses gleichen Typs von einem Modul und gleichem Kanal am anderen Modul eine Passivierung des Kanals bewirken. Diese Art des Anschlusses ist nicht erlaubt und wird nicht empfohlen.
- **Gibt es eine andere Verzögerung des Sicherheitstelegramms, wenn das Sicherheitsmodul in einem anderen physischen Steckplatz platziert wird (Slot für Kommunikationsmodul oder E/A-Modul)?**
  Die Telegrammverzögerung ist in solchen Fällen zu vernachlässigen, da der mögliche Unterschied weit unter 1 ms liegt.
- **Ist die interne 1oo2-Sicherheitsstruktur nur für Sicherheitseingänge anwendbar, wenn es sich um einen 2-Kanal-Eingang handelt?**
  Nein, das gesamte Hardwaresystem der AC500-S ist für die Verwendung der internen 1oo2-Sicherheitsstruktur konzipiert. Somit wird auch ein einzelner angeschlossener Eingang intern gesplittet und mit der 1oo2-Sicherheitsarchitektur verarbeitet.
- **Wie werden sichere Kontaktmatten/sichere Kontaktpuffer und sichere Kontaktleisten gekoppelt?**
  Die meisten auf dem Markt erhältlichen Sicherheitsmatten und Stoßfänger beinhalten die ASi-Safety-Option. Mithilfe von ASi-Safety zum PROFINET/PROFIsafe-Gateway können solche Signale an die AC500-S angeschlossen werden.
- **Können Zweileitergeber mit einem analogen Eingang verwendet werden?**
  Ja, das Analogmodul AI581-S kann Zweileitergeber verarbeiten.
- **Was bedeutet die „EIN“-Zeit des Testimpulses bei DI581-S/DX581-S-Modulen? Wie oft wird sie wiederholt?**
  Die Anschlussklemmen für Testimpulse liefern ein Signal mit 24 V DC zur Überwachung passiver Sensoren mit Testimpulsen. Dieses Testimpulssignal wird für eine festgelegte Zeit (1 ms) in den Zustand LOW abgeschaltet. Dies gilt sowohl für das DI581-S- als auch das DX581-S-Modul. Der Testimpuls wird auf jedem Testimpulskanal alle 58 ms beim DI581-S- und alle 27 ms beim DX581-S-Modul wiederholt.
- **Wie oft ist der Sicherheitsausgang AUS, wenn die Erkennungsfunktion beim DX581-S-Modul EIN ist?**
  Ist die Erkennung aktiviert, wird der Ausgang des Sicherheitsmoduls DX581-S alle 55 ms geprüft. Bitte beachten Sie, dass der Testimpuls des internen Hauptschalters ebenfalls an jedem Ausgang beobachtet werden kann. Der Testimpuls des Hauptschalters kann nicht deaktiviert werden und liegt immer an. Seine Dauer liegt im schlimmsten Fall knapp unter 1 ms (wenn der Ausgangsstrom 500 mA beträgt) und ist im besten Fall fast nicht sichtbar (wenn der Ausgangsstrom unter 50 mA liegt).
- **Können AC500-S-Sicherheitsmodule in Anwendungen mit niedriger Anforderungsrate eingesetzt werden?**
  Ja.

- **Wie wird die Adressschaltereinstellung der Sicherheits-CPU mit SIL 3 / PL e konform, wenn der Wert im Programm der Sicherheitsanwendung verwendet werden soll?**
  Man kann den Ausführungspfad des Sicherheitsprogramms der Sicherheits-CPU je nach Einstellung des Sicherheits-CPU-Konfigurationsschalters ändern, der mit dem Funktionsbaustein SF_SM5XX_OWN_ADR in das Sicherheitsprogramm eingelesen werden kann. Eine Änderung des Ausführungspfads des Sicherheitsprogramms der Sicherheits-CPU je nach Einstellung des Sicherheits-CPU-Adressschalters alleine ist nicht immer ausreichend, um SIL 3 / PL e zu erreichen. Man muss einen zusätzlichen Mechanismus implementieren, z. B. um einen zweiten Einstiegspunkt für die Programmkonfigurationseinstellung auf Anwendungsebene zu erzielen. Dies kann beispielsweise durch Einlesen vorkonfigurierter (vorab gespeicherter) Werte von einer SD-Karte auf die Standard-CPU geschehen. Dieser zusätzliche, vorkonfigurierte (vorab gespeicherte) Wert muss auf die Sicherheits-CPU übertragen und mit der Adressschalterstellung der Sicherheits-CPU verglichen werden, bevor die Adressschalterstellung der Sicherheits-CPU akzeptiert wird, um den Ausführungspfad des Sicherheitsprogramms der Sicherheits-CPU zu ändern. Dadurch kann ein höheres funktionales Sicherheitsniveau bis SIL 3 / PL e erzielt werden.
- **Bei welchen Anwendungsarten werden FBs wie SF_APPL_MEASURE_BEGIN und SF_APPL_MEASURE_END verwendet?**
  Diese FBs können für die Zeitprofilerstellung Ihres Sicherheitsanwendungsprogramms genutzt werden, die oft beim Debugging hilfreich ist, um Leistungsengpässe in Sicherheitsanwendungen zu finden. So kann beispielsweise die tatsächlich von der Sicherheits-CPU benötigte Zeit für die Ausführung eines bestimmten Teils der Sicherheitsprogrammlogik veranschlagt werden.
- **Wie können Nutzerdaten auf der Sicherheits-CPU persistent gemacht werden?**
  Nutzerdaten können im nichtflüchtigen Flash-Speicher der Sicherheits-CPU gespeichert und von dort mit speziellen FBs (SF_FLASH_WRITE, SF_FLASH_READ und SF_FLASH_DEL) gelesen oder gelöscht werden.
- **Können Fehler im Zusammenhang mit dezentralen PROFINET/PROFIsafe-Sicherheitsmodulen im Diagnosepuffer der Standard-CPU erfasst werden?**
  Mit AC500 V2-Standard-CPU:
  Ja, Sie können mit speziellen Diagnose-FBs Diagnosemeldungen von dezentralen Sicherheitsmodulen auf die V2-Standard-CPU lesen. Diese FBs finden Sie in der Bibliothek Profinet_AC500_V13.lib der V2-Standard-CPU.
  Mit AC500 V3-Standard-CPU:
  Die Fehler im Zusammenhang mit PROFINET/PROFIsafe können automatisch im Diagnosepuffer der V3-Standard-CPU erfasst werden.
- **Warum startet der Neustartbefehl der Standard-CPU dezentrale Sicherheits-E/A-Module nicht neu?**
  Dieses Verhalten ist beabsichtigt. Nach dem Neustartbefehl der Standard-CPU werden nur zentrale Sicherheits-E/A-Module neu initialisiert. Alle dezentralen Sicherheits-E/A-Module können nicht neu initialisiert werden und müssen vom Sicherheitsprogramm bestätigt werden, um sie wieder zu integrieren, wenn der Neustart der Standard-CPU und der Sicherheits-CPU abgeschlossen ist. Dieses Verhalten (Neuinitialisierung oder nicht) hängt von der PROFINET CI50x-PNIO-Einstellung ab und kann geändert werden.
- **Ist die Umwandlung von ST in LAD/FBD möglich?**
  Ja, bei einfachen Projekten mit grundlegendem Befehlssatz ist die Umwandlung möglich. Allerdings können nicht alle ST-Standardkonstrukte in LAD/FBD umgewandelt werden. Bitte beachten Sie, dass bei einer Umwandlung von ST in LAD/FBD der Sicherheitsprogrammcode nicht wieder auf ST zurückgesetzt werden kann.
- **Bei antivalenter Schaltung ist der NO-Kanal immer mit dem niedrigeren Kanal verbunden (dem Kanal, der einen gesammelten 2-Kanal-Sicherheitswert für die Sicherheits-CPU liefert). Gibt es dafür einen speziellen Grund?**
  Dieses Verhalten ist beabsichtigt, um Fehler bei der antivalenten Sensorverdrahtung und eine potenzielle Fehlinterpretation davon zu vermeiden, welcher Kanal einen gesammelten 2-Kanal-Sicherheitswert liefert.

- **Wenn unsere Sicherheits- und Standard-E/As mit Sicherheitssteuerungen von Drittanbietern verwendet werden, sind die Diagnosemeldungen der Sicherheits- und Standard-E/As dann im Diagnosepuffer dieser Sicherheitssteuerungen von Drittanbietern verfügbar?**

  Alle Diagnosemeldungen von Sicherheits- und Standard-E/As sind nicht sichere Daten, die von Standard-CPUs gesammelt werden (auch von Drittanbieter-CPUs). Alle Diagnosemeldungen von Sicherheits- und Standard-E/As sind derzeit im AC500-Diagnosemeldungsformat verfügbar und können in den Diagnosepuffer von Standard-CPUs von Drittanbietern gelesen und abgelegt werden, indem spezielle FBs aufgerufen werden oder die PROFINET-Standarddiagnose verwendet wird.
- **Wer könnte ein Sicherheitsprogramm zertifizieren?**

  Alle internationalen und nationalen zugelassenen Zertifizierungsstellen wie TÜV, EXIDA, UL etc. (wovon einige weltweit arbeiten) könnten ein Sicherheitsprogramm zertifizieren.
- **Welche Schritte sind bei der Entwicklung eines Sicherheitsprogramms richtig?**

  Bitte beachten Sie die Richtlinien ISO 13849-1 und IEC 62061 für die Entwicklung von Maschinensicherheitsanwendungen und IEC 61511 für die Entwicklung von Prozesssicherheitsanwendungen.
- **Dürfen FOR-Schleifen in ST-Programmen als Alternative zu IF- und CASE-Blöcken für Index-Grenzwertprüfungen bei Zugriff auf Felder verwendet werden?**

  Nein, diese dürfen nicht als Alternative verwendet werden.

  Wenn Arrays in FOR-Schleifen verwendet werden, muss der Programmierer weiterhin Index-Grenzwertprüfungen bei Zugriff auf Felder implementieren.

3 AC500-S-Sicherheitsmodule

3.1 Sicherheits-CPU — SM560-S / SM560-S-FD-1 / SM560-S-FD-4

Elemente des Moduls

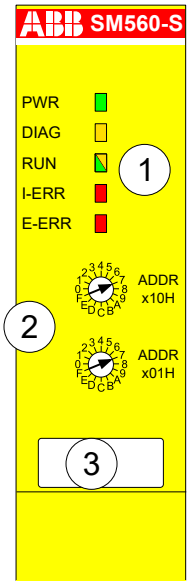


Abb. 5: SM560-S / SM560-S-FD-1 / SM560-S-FD-4

1 Fünf LEDs als Statusanzeigen
2 Drehschalter zur Adress-/Konfigurationseinstellung
3 Schild

3.1.1 Verwendungszweck

SM560-S / SM560-S-FD-1 / SM560-S-FD-4 sind Sicherheits-CPUs für Sicherheitsanwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1). Die Sicherheits-CPU wird auf der linken Seite der Standard-CPU auf demselben Modulträger montiert. Die Kommunikation zwischen Standard-CPU und Sicherheits-CPU erfolgt über den im Modulträger integrierten internen Kommunikationsbus.

Abhängig vom eingesetzten Modulträger und von der verwendeten Standard-CPU können mehrere Kommunikationsmodule gleichzeitig für eine Standard-CPU eingesetzt werden. An jeder Standard-CPU kann jedoch immer nur eine Sicherheits-CPU gleichzeitig betrieben werden.

Die Sicherheits-CPU wird über DPRAM mit Sicherheits-System-Konfigurator und AC500-S Programming Tool programmiert und konfiguriert. Beide sind Teil der Software Automation Builder.

Die Konfiguration der Sicherheits-CPU wird nichtflüchtig in ihren Flash-EPROMs gespeichert.

Informationen zur Kombination von Sicherheits-CPUs mit der zugehörigen nicht sicherheitsgerichteten Umgebung finden Sie in den Kompatibilitätsinformationen ↳ *Anhang B.1 „Kompatibilität mit AC500 V2-Standard-CPU“ auf Seite 415* ↳ *Anhang C.1 „Kompatibilität mit AC500 V3-Standard-CPU“ auf Seite 437*.

3.1.2 Funktionalität

3.1.2.1 Übersicht

AC500-Sicherheits-CPUs werden immer mit Standard-CPUs eingesetzt.

Die Sicherheits-CPU wird in AC500-S Programming Tool auf ähnliche Weise wie bei einer AC500-CPU programmiert, allerdings unter Beachtung der Richtlinien zur Sicherheitsprogrammierung. Die Programmierung erfolgt durch Routing über die AC500-CPU unter Verwendung der seriellen Schnittstelle oder Ethernet. Das Anwenderprogramm setzt sich zusammen aus:

- dem kompilierten Code aller im Programm aufgerufenen POEs
- dem Initialisierungscode für die Variablen.

SM560-S-FD-1 / SM560-S-FD-4 enthält alle Funktionen der Sicherheits-CPU SM560-S. Bei den Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 sind folgende zusätzliche Funktionen verfügbar:

- PROFIsafe F-Device-Funktionalität
  - SM560-S-FD-1 kann mit 1 PROFIsafe F-Host (Steuerung) kommunizieren
  - SM560-S-FD-4 kann mit bis zu 4 PROFIsafe F-Hosts (Steuerungen) kommunizieren
- Größeres Sicherheitsprogramm: 1,3 MB (die Sicherheits-CPU SM560-S hat 1,0 MB).

Jede Variante der Sicherheits-CPU hat ihre eigene Produktkennung in den Produktionsdaten. Somit wird der Download eines Bootprojektes auf eine falsche Produktvariante anhand der Firmware erkannt.

3.1.2.2 Fließkommaoperationen

Sicherheits-CPUs können Fließkommaoperationen durchführen.

GEFAHR!

Teilen durch Null ist nicht zulässig und sollte spätestens bei der formellen Sicherheits-CPU-Codeüberprüfung gemäß den Richtlinien zur Sicherheitsprogrammierung erkannt werden ↳ *Kapitel 4.4 „Sicherheitsprogrammierrichtlinien“ auf Seite 196*.

Bei Ausnahmen in den Fließkommaoperationen (z. B. durch die Verwendung ungültiger Argumente) wechselt die Sicherheits-CPU in den Zustand SAFE STOP oder gibt den Wert „unendlich“ zurück.

Der Bereich der gültigen Argumente für Fließkommaoperationen in der Sicherheits-CPU ist wie folgt:

- SIN und COS: $[-9 \times 10^{15} \ldots 9 \times 10^{15}]$
- TAN: $[-4{,}5 \times 10^{15} \ldots 4{,}5 \times 10^{15}]$
- ATAN: $[-3{,}402823 \times 10^{38} \ldots 3{,}402823 \times 10^{38}]$
- LOG, LN und SQRT: bis zu $3{,}402823 \times 10^{38}$

Die Argumente, die nicht im oben angegebenen Bereich liegen, führen zu einem SAFE STOP der Sicherheits-CPU.

GEFAHR!

Die Gültigkeit des Endergebnisses der Fließkommaoperation muss überprüft werden, bevor diese im Sicherheitsprogramm verwendet wird.

GEFAHR!

Es ist wichtig, Folgendes bei der Programmierung von Fließkomma-Rechnungen zu berücksichtigen *[5]*:

- Runden oder Abschneiden der Ergebnisse nach jeder Fließkommaoperation gemäß definierten ULPs (MOD, EXPT, EXP, ABS, TAN, ASIN, ACOS, ATAN, SIN, COS, LOG und LN werden mit einem max. erwarteten Fehler von 2 ULP ausgeführt; ADD, SUB, MUL, DIV und SQRT mit einem max. Fehler von 1 ULP in der Sicherheits-CPU). Weitere Details zu ULPs unter *http://en.wikipedia.org/wiki/Unit_in_the_last_place*.
- Wenn Sie einen Wert berechnen, der das Ergebnis einer Reihe von Fließkommaoperationen ist, kann der Fehler zunehmen und die eigentliche Berechnung stark beeinflussen.
- Wenn zwei Zahlen mit demselben Vorzeichen subtrahiert oder zwei mit verschiedenen Vorzeichen addiert werden, kann die Genauigkeit des Ergebnisses schlechter sein als die Genauigkeit, die im Fließkommaformat erreicht wird.
- Die Reihenfolge der Berechnung kann die Genauigkeit des Ergebnisses beeinflussen.
- Bei einer Reihe Berechnungen mit Addition, Subtraktion, Multiplikation und Division versuchen Sie, zuerst die Multiplikation und Division durchzuführen.
- Versuchen Sie beim Multiplizieren und Dividieren einer Reihe von Zahlen die Multiplikationen so aufzuteilen, dass große und kleine Zahlen zusammen multipliziert werden. Versuchen Sie ebenso, Zahlen mit derselben Größe zu dividieren.
- Beim Vergleichen von zwei Fließkommazahlen vergleichen Sie immer, ob ein Wert in dem Bereich liegt, der von einem zweiten Wert (plus oder minus einem kleinen Fehlerwert) vorgegeben wird.

3.1.2.3 Systemfunktionen

Die Sicherheits-CPU verfügt über keine Batterie. Deshalb werden alle Operanden initialisiert, sobald die Steuerspannung aktiviert wird. Der Datenaustausch zwischen Sicherheits- und Standard-CPUs ist möglich *Anhang B.5 „Datenaustausch zwischen Sicherheits-CPU und AC500 V2-Standard-CPU“ auf Seite 430* *Anhang C.5 „Datenaustausch zwischen Sicherheits-CPU und AC500 V3-Standard-CPU“ auf Seite 448*.

GEFAHR!

Es wird nicht empfohlen, Datenwerte von der Standard-CPU auf die Sicherheits-CPU zu übertragen. Hierbei müssen die Endanwender zusätzliche prozessspezifische Validierungsverfahren in ihrem Sicherheitsprogramm definieren, um die Korrektheit der übertragenen nicht sicheren Daten zu überprüfen, wenn sie diese nicht sicheren Werte für Sicherheitsfunktionen verwenden möchten.

Datenwerte von der Sicherheits-CPU auf die Standard-CPU zu übertragen, z. B. für Diagnose und spätere Darstellung auf Bedienpanels, ist kein Problem.

Selbsttests und Diagnosefunktionen (sowohl beim Starten als auch während des Betriebs), wie CPU- und RAM-Tests, Programmablauf-Überwachung usw., werden in die Sicherheits-CPU gemäß den Anforderungen von IEC 61508 implementiert.

Ausgewählte Daten können „failsafe“ und dauerhaft im Flash-Speicher der Sicherheits-CPU mit den speziellen Bibliotheken-POEs SF_FLASH_READ, SF_FLASH_WRITE und SF_FLASH_DEL gespeichert werden *Kapitel 4.6.7.10 „SF_FLASH_READ“ auf Seite 351* *Kapitel 4.6.7.11 „SF_FLASH_WRITE“ auf Seite 354* *Kapitel 4.6.7.12 „SF_FLASH_DEL“ auf Seite 357*.

Die Sicherheits-CPU ist eine „Single-Threaded“ und „Single-Task“-CPU. Nur eine freilaufende Task ist für die Ausführung des Sicherheitsprogramms verfügbar. Die freilaufende Task wird verarbeitet, sobald das Sicherheitsprogramm gestartet wird, und nach Abschluss eines Laufs in einer Endlosschleife automatisch neu gestartet. Für diese Task ist keine Zykluszeit einstellbar, die Anwender können die Zykluszeit der Sicherheits-CPU jedoch mithilfe der Bibliotheks-POE SF_WDOG_TIME_SET überwachen ↳ *Kapitel 4.6.7.3 „SF_WDOG_TIME_SET“ auf Seite 346*.

Die Watchdog-Zeit der Sicherheits-CPU wird mit SF_WDOG_TIME_SET gesetzt; dies ist die maximal zulässige Zeit für ihren Zyklus. Wenn die mit SF_WDOG_TIME_SET gesetzte Zeit während der Programmausführung auf der Sicherheits-CPU überschritten wird, schaltet sie in den Zustand SAFE STOP (keine gültigen Telegramme werden vom Gerät generiert) und die LED I-ERR leuchtet.

HINWEIS!

Im Anwenderprogramm muss die POE SF_WDOG_TIME_SET nur einmal aufgerufen werden, um einen Watchdog-Wert über 0 einzustellen. Wenn SF_WDOG_TIME_SET nicht im Anwenderprogramm aufgerufen wird, wird die Standard-Watchdog-Zeit = 0 verwendet, wodurch es in der Sicherheits-CPU unmittelbar zu einem SAFE STOP-Zustand mit I-ERR LED = EIN kommt.

Um gelegentliche Stopps der Sicherheits-CPU aufgrund eines Überschreitens der Zyklusdauer zu vermeiden, was von der Zyklusdauer-Überwachung festgestellt wurde, sollte die CPU-Last beim Testlauf des Anwenderprogramms überwacht werden, um sicherzustellen, dass der gewählte Watchdog-Wert korrekt gesetzt wurde.

HINWEIS!

Der Watchdog-Wert, der in der POE SF_WDOG_TIME_SET gesetzt wird, wird zur Überwachung der Zyklusdauer der Sicherheits-CPU ausschließlich im Modus RUN (sicher) verwendet. In den Modi DEBUG RUN (nicht sicher) und DEBUG STOP (nicht sicher) der Sicherheits-CPU wird der Watchdog-Wert ignoriert.

Mit dem speziellen SPS-Browser-Befehl „setpwd“ kann ein Passwort für die Sicherheits-CPU definiert werden, um nicht autorisierten Zugang zu den Daten (Anwendungsprojekt usw.) zu verhindern. Ohne Passwort kann keine Verbindung zur Sicherheitssteuerung hergestellt werden.

3.1.2.4 Überwachung der Spannungsversorgung

Die interne Spannungsversorgung (+3,3 V) der Sicherheits-CPU wird auf Unter- und Überspannung überwacht. Wenn eine Unter- oder Überspannung festgestellt wird, schaltet die Sicherheits-CPU in den Zustand SAFE STOP (keine gültigen Telegramme werden vom Gerät generiert) und die LED I-ERR leuchtet. Um das Neustarten der Sicherheits-CPU zu regeln, nachdem die Spannungsversorgung wieder im zulässigen Spannungsbereich ist, kann man eine max. erlaubte Anzahl von Neustarts der Sicherheits-CPU mit der POE SF_MAX_POWER_DIP_SET definieren ↳ *Kapitel 4.6.7.2 „SF_MAX_POWER_DIP_SET“ auf Seite 344*.

3.1.2.5 Adress-/Konfigurationsschalter-/F_Dest_Add-Einstellungen

Die Einstellung der zwei Drehschalter für die Adresse und/oder Systemkonfiguration von PROFIsafe (diese Schalter können z. B. für die Programmablauf-Überwachung des Sicherheitsprogramms verwendet werden) kann im Sicherheitsprogramm mit der POE SF_SM5XX_OWN_ADR ausgelesen werden ↳ *Kapitel 4.6.7.8 „SF_SM5XX_OWN_ADR“ auf Seite 350*. Die Adresswerte der Schalter 0xFF, 0xFE, 0xFD und 0xFC werden für interne Systemfunktionen der Sicherheits-CPU verwendet (siehe unten):

- Der Adresswert 0xFF während des Startens der Sicherheits-CPU verhindert das Laden des Bootprojekts in die Sicherheits-CPU beim Starten (das Bootprojekt bleibt weiterhin im Flash-Speicher der Sicherheits-CPU). Infolgedessen kann der Anwender sich an der Sicherheits-CPU anmelden und ein neues korrektes Bootprojekt laden. Das kann erforderlich sein, wenn das Bootprojekt korrupt ist, was zu einem SAFE STOP der Sicherheits-CPU führen könnte. Die Sicherheits-CPU geht nach dem Start und der erfolgreichen Ausführung des 0xFF-Befehls in den Zustand DEBUG STOP (nicht sicher).
- Der Adresswert 0xFE während des Startens der Sicherheits-CPU erlaubt das Löschen des Bootprojekts aus ihrem Flash-Speicher. Das Bootprojekt wird nach einem Power Cycle der Sicherheits-CPU endgültig gelöscht. Das kann erforderlich sein, wenn das Bootprojekt korrupt ist, was zu einem SAFE STOP der Sicherheits-CPU führen könnte. Die Sicherheits-CPU geht nach dem Start und der Ausführung des 0xFE-Befehls in den Zustand SAFE STOP.
- Der Adresswert 0xFD während des Startens der Sicherheits-CPU erlaubt das Löschen der Nutzerdaten aus ihrem Flash-Speicher. Die Nutzerdaten sind nach einem Power Cycle der Sicherheits-CPU endgültig gelöscht. Das kann erforderlich sein, wenn die Nutzerdaten korrupt sind, was zu einem SAFE STOP der Sicherheits-CPU führen könnte. Die Sicherheits-CPU geht nach dem Start und der Ausführung des 0xFD-Befehls in den Zustand SAFE STOP.
- Der Adresswert 0xFC während des Startens der Sicherheits-CPU erlaubt das Löschen sämtlicher Daten der Sicherheits-CPU, d. h. zusätzlich zu Bootprojekt und Nutzerdaten auch das Passwort der Sicherheits-CPU und den definierten Spannungseinbruchwert aus dem Flash-Speicher. Das bedeutet ein Rücksetzen der Sicherheits-CPU in ihren Originalzustand. Die Daten sind nach einem Power Cycle der Sicherheits-CPU endgültig gelöscht. Die Sicherheits-CPU geht nach dem Start und der Ausführung des 0xFC-Befehls in den Zustand SAFE STOP.

Der Adressbereich der Schalter von 0xF0 … 0xFB ist für zukünftige interne Systemfunktionen reserviert.

HINWEIS!

Die Verwendung der Adresswerte aus dem Systembereich 0xF0 … 0xFF kann zum Verlust wichtiger Anwenderinformationen im Flash-Speicher der Sicherheits-CPU, z. B. Bootprojekt, Nutzerdaten, Passwort oder Spannungseinbruchwert, führen. Es ist deshalb wichtig, dass die Anwender während der Veränderung der Schalteradressen-Position auf der Sicherheits-CPU besondere Vorsicht walten lassen.

GEFAHR!

Ungeachtet der Tatsache, dass die SF_SM5XX_OWN_ADR-Funktion eine Sicherheits-POE ist, ist der Adresswert des Hardwareschalters ein nicht sicherer Wert und es sind zusätzliche Maßnahmen nötig, um die funktionalen sicherheitsbezogenen Anforderungen zu erfüllen.

PROFIsafe F_Dest_Add-Adressen für F-Devices in Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 werden über den Adress-Drehschalter festgelegt. Dies bedeutet, dass hinter dem Adress-Drehschalter an Sicherheits-CPUs mehr als eine Funktion liegen kann. Dies muss bei der Konzeption von Sicherheitsanwendungen sorgfältig bedacht werden, beispielsweise wenn Systemfunktionen (Werte 0xFF, 0xFE, 0xFD und 0xFC am Adress-Drehschalter) bei den Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 genutzt werden müssen. Im letzteren Fall muss der zuvor festgelegte Wert des Adress-Drehschalters für F_Dest_Add-Adressen entsprechend dokumentiert und auf den ursprünglich dokumentierten Wert zurückgesetzt werden, wenn die Systemfunktionen bei den Sicherheits-CPUs erfolgreich durchgeführt wurden.

Die Nutzung des Adress-Drehschalters für die F_Dest_Add-Einstellung ermöglicht die Verwendung des gleichen Bootprojekts der Sicherheits-CPU für verschiedene Maschinen, sofern jede Maschine eine eindeutige, mit dem Adress-Drehschalter voreingestellte F_Dest_Add-Adresse hat und im Automation Builder-Projekt richtig konfiguriert wird.

Der zulässige Wert des Adress-Drehschalters für die F_Dest_Add-Einstellung ist 1 bis 239 (0 würde keine Nutzung von F-Devices bei SM560-S-FD-1 / SM560-S-FD-4 bedeuten). Ein Adress-Drehschalter repräsentiert F_Dest_Add für alle möglichen F-Device-Instanzen (maximal 32 F-Device-Instanzen mit je 12 Bytes Sicherheitsdaten) bei den Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4.

Folgende Regel gilt für die Zuweisung von F_Dest_Add zu F-Devices:

– F_Dest_Add für F-Device = Adress-Drehschalterwert × 100 + F-Device-Instanznummer (0..31, eine „consecutive number“, da die F-Device-Instanzen im Modul-/Gerätebau des Automation Builder erstellt werden).
– Für die richtige Konfiguration vom F-Device in den Sicherheits-CPUs SM560-S-FD-1 und SM560-S-FD-4 müssen die richtige Konfiguration der F_Dest_Add mit dem Adress-Drehschalterwert und die Konfiguration des F-Parameters vom F-Host und von dessen Controller zur Verfügung gestellt werden.

Ein komplexes System mit mehreren AC500-S-Untersystemen, die untereinander über PROFIsafe verbunden sind, erfordert eine zusätzliche Betrachtung der Zuweisung von F_Dest_Add- und F_Source_Add-Adressen, da sich Meldungen von verschiedenen F-Hosts im „Black Channel“ überschneiden können, z. B. in der Standard-CPU. Die mögliche Überschneidung kann die Wahrscheinlichkeit eines gefährlichen Fehlers in der Sicherheitskonfiguration und -kommunikation erhöhen. Der typische PFH-Wert für die PROFIsafe-Kommunikation ist 3,0E-10.

GEFAHR!

Bei jedem AC500-S-Untersystem, bei dem sich die sichere PROFIsafe-Kommunikation im „Black Channel“ mit der PROFIsafe-Kommunikation von einem anderen F-Host überschneiden kann, muss ein Paar aus F_Dest_Add und F_Source_Add (in der PROFIsafe-Terminologie der sogenannte Codename ↳ *[2]*) eindeutig sein. Wenn nur F_Dest_Add vom F-Device geprüft wird (wenn beispielsweise Hardware-Adresseinstellungen an ihm verwendet werden), dann müssen nicht nur die Codenamen, sondern auch F_Dest_Add eindeutig sein. Bei SM560-S-FD-1 und SM560-S-FD-4 müssen aufgrund der Tatsache, dass sich die PROFIsafe-Kommunikation von verschiedenen F-Hosts (PROFIsafe-Telegramme vom eigenen F-Host an SM560-S-FD-1 oder SM560-S-FD-4 und PROFIsafe-Telegramme von externen F-Hosts) an der Sicherheits-CPU überschneidet, zusätzliche Maßnahmen für eindeutige Codenamen angewandt werden:

– Eindeutige F_Dest_Add für alle F-Devices, die zu(m) externen F-Host(s) und zum eigenen F-Host bei den Sicherheits-CPUs SM560-S-FD-1 oder SM560-S-FD-4 gehören.

HINWEIS!

FSCP 3/1 Adresstyp 1 wird in SM560-S-FD-1 und SM560-S-FD-4 verwendet:

Nur F_Dest_Add wird für die Identifizierung des PROFIsafe F-Device in SM560-S-FD-1 und SM560-S-FD-4 verwendet.

Der zulässige Bereich für F_Dest_Add-Adressen ist beschrieben in ↳ *Kapitel 4.3.5 „Instanziierung und Konfiguration von Sicherheitsmodulen / Definition von Variablennamen“ auf Seite 150*.

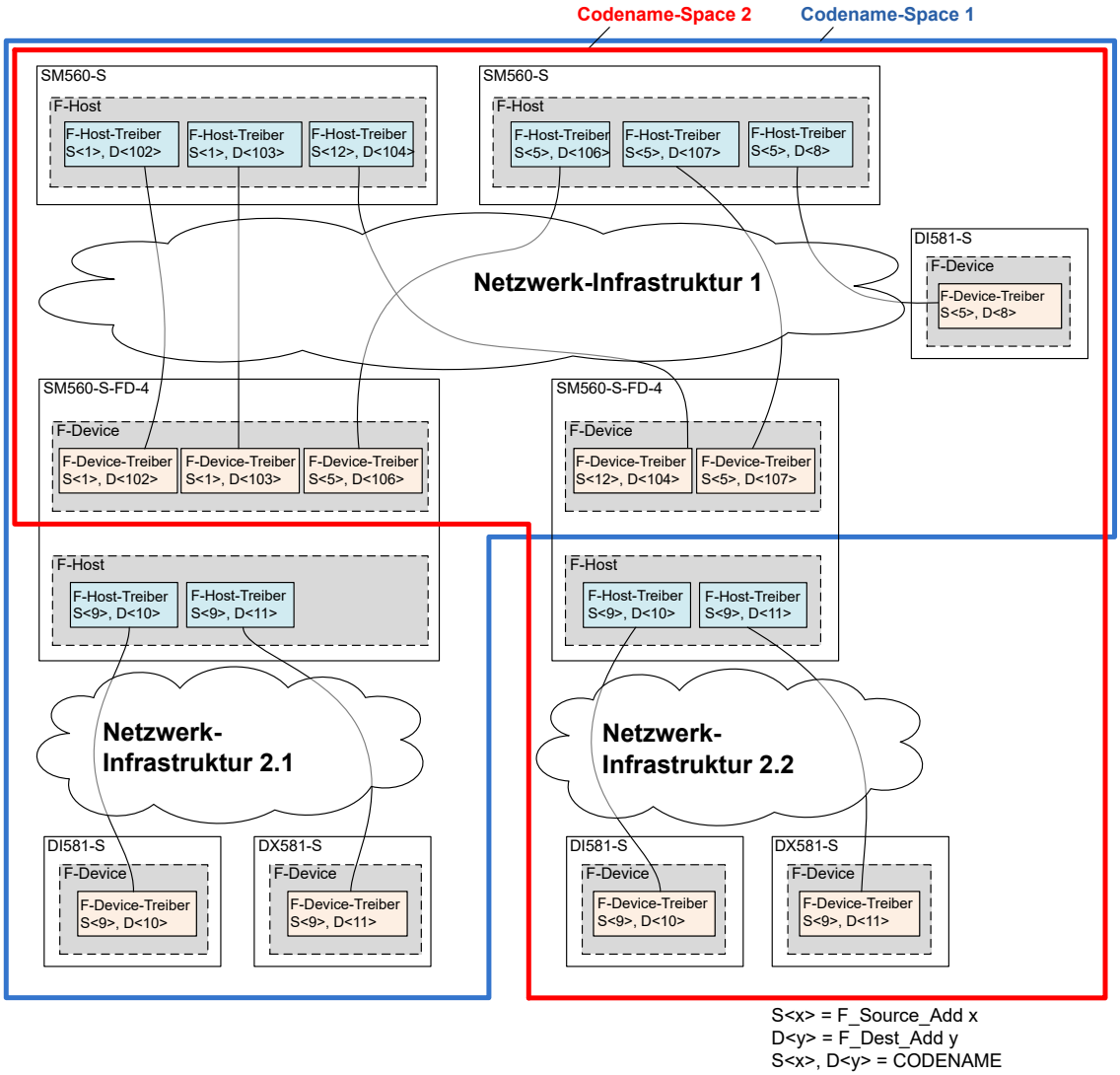


Abb. 6: Beispielsystem mit sich überschneidenden PROFIsafe-Netzwerken und PROFIsafe-Adresszuweisung und einer allgemeinen generischen Netzwerkinfrastruktur, die WLAN, Telekommunikationsnetz, Direktanschluss etc. umfassen kann.

GEFAHR!

Zusammenfassend müssen folgende Regeln mit organisatorischen Abläufen für eine sichere Kommunikation von CPU zu CPU mit den CPUs SM560-S-FD-1 und SM560-S-FD-4 angewandt werden. (Dies muss manuell geprüft werden und ist Teil von ↳ *Kapitel 6.3 „Checkliste für Konfiguration und Verkabelung“ auf Seite 378*.):

- Im gleichen Codename-Space muss F_Dest_Add eindeutig sein (Abb. 6, Seite 45).
- Im gleichen Codename-Space darf F_Source_Add nicht in anderen F-Hosts wiederverwendet werden. Im gleichen F-Host ist eine Wiederverwendung für mehrere F-Host-Treiber erlaubt.
- Im gleichen Codename-Space darf F_Dest_Add nicht als F_Source_Add verwendet werden und umgekehrt.

Um sicherzustellen, dass die richtige Sicherheitskonfiguration und Sicherheitsanwendung in das richtige System geladen wird, können Kunden den Adressschalter der SM560-S-FD-1 / SM560-S-FD-4 verwenden, um zu prüfen, ob die Konfiguration zum ausgewählten System passt. Der Adressschalter bei SM560-S-FD-1 / SM560-S-FD-4 schützt die gegebene Sicherheits-CPU bedingungslos, da er für die Festlegung der F_Dest_Add für PROFIsafe F-Device-Instanzen verwendet wird. Wird ein falsches Bootprojekt auf die gegebene SM560-S-FD-1 / SM560-S-FD-4 geladen, dann passt dieses nicht zu den vom F-Host übertragenen F-Parametern und führt zum Konfigurationsfehler der entsprechenden PROFIsafe-Instanz.

3.1.2.6 Aktualisierung von Firmware, Bootcode und Bootprojekt

Die Aktualisierungen der Sicherheits-CPU für Bootprojekt, Firmware und Bootcode erfolgen über eine Standard-CPU, entweder über Automation Builder oder SD-Karte.

GEFAHR!

Jeder Aktualisierung von Firmware und Bootcode muss eine komplette Validierung der funktionalen Sicherheit für die gegebene Sicherheitssteuerungsanwendung folgen.

3.1.2.6.1 Aktualisierung über Automation Builder

Es wird empfohlen, Firmware, Bootcode und Bootprojekt über den Automation Builder zu aktualisieren. Diese Funktion ist beschrieben in *[3]*.

3.1.2.6.2 Aktualisierung über SD-Karte

GEFAHR!

Wenn Sie eine SD-Karte für die Aktualisierung von Firmware, Bootcode und Bootprojekt über eine Standard-CPU verwenden, ist es wichtig, dass die Kunden zur Vermeidung ungewollter Software-Aktualisierungen auf der Sicherheits-CPU spezielle organisatorische Abläufe festlegen (z. B. beschränkten Zugriff auf den Schaltschrank, in dem sich die Sicherheits-CPU befindet).

Die aktuelle Firmware-Version kann mit der POE SF_RTS_INFO auf der Sicherheits-CPU gelesen werden (*Kapitel 4.6.7.9 „SF_RTS_INFO“ auf Seite 350*). Auf diese Weise können Sie die Ausführung des Sicherheitsprogramms auf bestimmte vordefinierte Firmware-Versionen beschränken.

HINWEIS!

Die gleichzeitige Aktualisierung von Bootprojekt und Firmware/Bootcode für die Sicherheits-CPU ist nicht möglich. Führen Sie diese Aktualisierungen in zwei Schritten durch. Das bedeutet, dass Sie unter Umständen zwei SD-Karten benötigen. Eine SD-Karte mit dem Firmware-/Bootcode-Update und eine weitere mit dem Bootprojekt-Update.

Firmware- und Bootcode-Aktualisierung

Die Vorgehensweise zum Erstellen einer SD-Karte mit Firmware/Bootcode für eine Sicherheits-CPU entspricht der Vorgehensweise für Kommunikationsmodule *[3]*.

Bootprojekt-Aktualisierung

Das Bootprojekt der Sicherheits-CPU kann nur aktualisiert werden, wenn kein Bootprojekt in die Sicherheits-CPU geladen ist. Auf diese Weise wird eine unbeabsichtigte Aktualisierung des Bootprojekts auf der Sicherheits-CPU vermieden. Bevor Sie ein neues Bootprojekt aktualisieren, löschen Sie das vorhandene Bootprojekt auf der Sicherheits-CPU, z. B. indem Sie den Adressschalter auf den Wert 0xFE/0xFC (*Kapitel 3.1.2.5 „Adress-/Konfigurationsschalter-/F_Dest_Add-Einstellungen “ auf Seite 42*) einstellen, über den SPS-Browser-Befehl `delappl` in AC500-S Programming Tool oder über *„Online ➔ Reset Origin“* im Automation Builder.

3.1.3 Montage, Abmessungen und elektrischer Anschluss

Die Sicherheits-CPU wird auf der linken Seite der Standard-CPU auf demselben Modulträger montiert. Der elektrische Anschluss wird beim Einbau der Sicherheits-CPU automatisch hergestellt. Hier werden grundlegende Informationen zur Montage des Systems angezeigt. Ausführliche Informationen finden Sie unter *[3]*.

Installation und Wartung dürfen nur von Elektro-Fachkräften nach den technischen Regeln, Richtlinien und einschlägigen Normen, z. B. EN 60204 Teil 1, vorgenommen werden.

Montage der Sicherheits-CPU

GEFAHR!
Einbau und Austausch im laufenden Betrieb sind bei Modulen unter Spannung nicht zulässig. Für jegliche Arbeiten an Sicherheitsmodulen müssen immer alle Spannungsquellen (Versorgungs- und Prozessspannungen) ausgeschaltet sein.

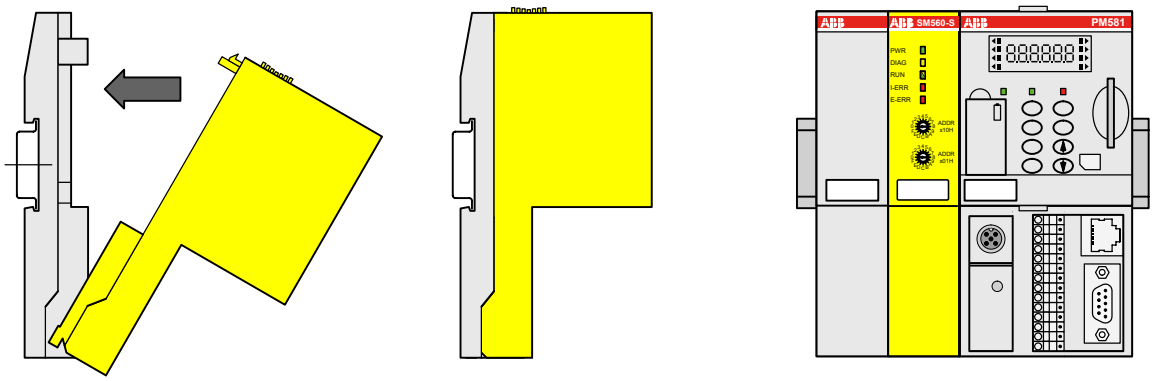


Abb. 7: Montageanleitung

▷ Setzen Sie das Modul unten ein und lassen Sie es oben einrasten.

Demontage der Sicherheits-CPU

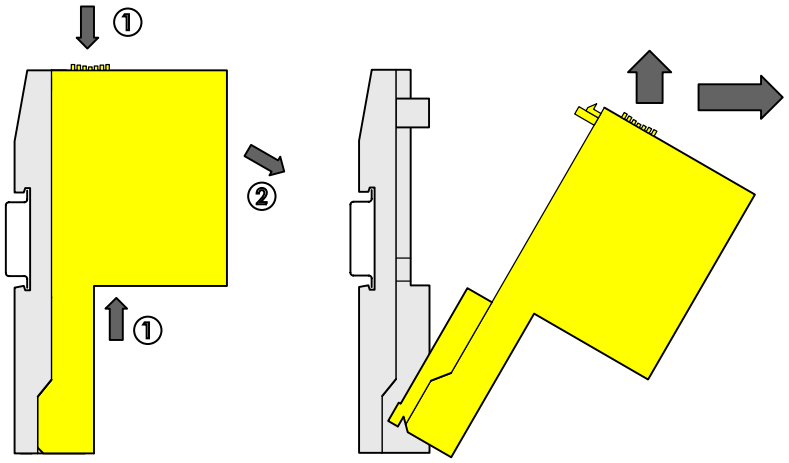


Abb. 8: Demontageanleitung

▷ Drücken Sie oben und unten, dann klappen Sie das Modul nach außen und entfernen es.

Abmessungen der Sicherheits-CPU

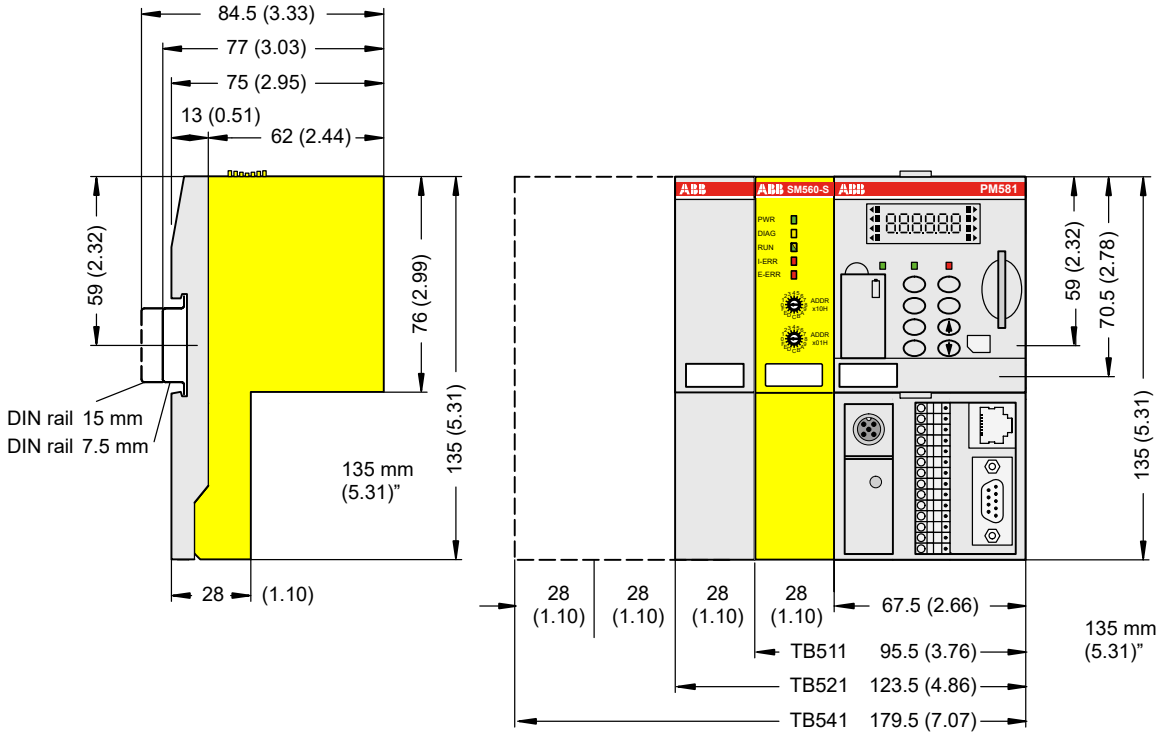


Abb. 9: Abmessungen der Sicherheits-CPU

3.1.4 Diagnose und LED-Statusanzeige

Der Zustand der Sicherheits-CPU wird durch LEDs angezeigt. Die LED RUN ist zweifarbig. Die folgende Abbildung und Tabelle zeigen die Positionen und Funktionen der 5 LEDs:

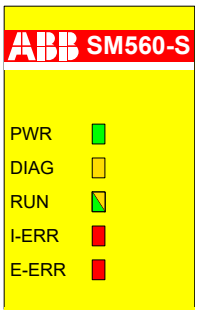


Abb. 10: LEDs als Statusanzeigen

Tab. 3: Statusanzeige und deren Bedeutung

LED	Beschreibung	Farbe	Status	Bedeutung
PWR	Modul-Spannungs-versorgung	Grün	EIN	Interne Spannungsversorgung +3,3 V ist verfügbar
			BLINKT	Nicht zutreffend
			AUS	Interne Spannungsversorgung +3,3 V ist nicht verfügbar
DIAG	Diagnose	Gelb	EIN	Konfigurationsfehler
			BLINKT	Nicht zutreffend
			AUS	Kein Konfigurationsfehler
RUN	RUN-Modus-Anzeige	Grün	EIN	Die Sicherheits-CPU ist im Modus RUN (sicher). Das Anwendungsprogramm wird ausgeführt.
			BLINKT	Nicht zutreffend
			AUS	Die Sicherheits-CPU ist im Modus DEBUG STOP (nicht sicher). Das Anwendungsprogramm wird nicht ausgeführt.
		Gelb	EIN	Die Sicherheits-CPU ist im Modus DEBUG RUN (nicht sicher). Das Anwendungsprogramm wird ausgeführt.
			BLINKT	Anzeige für Firmware, Bootprojekt oder Bootcode-Aktualisierung
			AUS	Die Sicherheits-CPU ist im Modus DEBUG STOP (nicht sicher). Das Anwendungsprogramm wird nicht ausgeführt.
I-ERR	Anzeige für internen Fehler im Gerät	Rot	EIN	Interner Fehler im Gerät, der zu einem SAFE STOP führt (vom Gerät werden keine gültigen PROFIsafe-Telegramme generiert)
			BLINKT	Firmware- oder Bootcode-Aktualisierung
			AUS	Kein interner Fehler im Gerät, der zu einem sicheren Zustand führt
E-ERR	Anzeige für einen externen Fehler	Rot	EIN	Diese LED kann nur aus dem Anwenderprogramm mit der speziellen Bibliotheken-POE SF_E_ERR_LED_Set eingeschaltet werden ↬ *Kapitel 4.6.7.1 „SF_E_ERR_LED_SET“ auf Seite 344*. Ein möglicher Fall wäre die Anzeige von wichtigen Fehlern der externen Geräte.

LED	Beschreibung	Farbe	Status	Bedeutung
			BLINKT	Diese LED kann nur aus dem Anwenderprogramm mit der speziellen Bibliotheken-POE SF_E_ERR_LED_Set eingeschaltet werden ↳ *Kapitel 4.6.7.1 „SF_E_ERR_LED_SET“ auf Seite 344*. Ein möglicher Fall wäre die Anzeige von geringeren Fehlern der externen Geräte.
			AUS	Keine externen Fehler wurden erkannt.

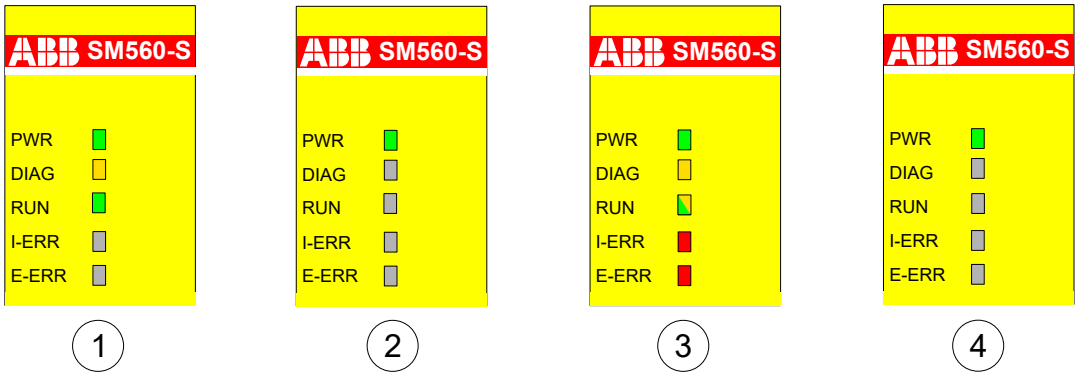


Abb. 11: LED-Anzeige der Sicherheits-CPU während des Starts

1 Zustand 1 – Hardware-Reset
2 Zustand 2 – Initialisierung
3 Zustand 3 – LED-Test
4 Zustand 4 – Ende der Startphase

Fehlermeldungen

Die Fehlermeldungen der Sicherheits-CPU werden zusammen mit den Fehlermeldungen anderer Kommunikationsmodule in Standard-CPUs gespeichert. Alle Fehlermeldungen werden an der Standard-CPU angezeigt. Darüber hinaus können Fehlermeldungen der Sicherheits-CPU an der Sicherheits-CPU selbst abgelesen werden.

Mit AC500 V2-Standard-CPU: ↳ *Anhang B.2.1 „Fehlermeldungen für Sicherheits-CPUs“ auf Seite 417*

Mit AC500 V3-Standard-CPU: ↳ *Anhang C.2.1 „Fehlermeldungen für Sicherheits-CPUs“ auf Seite 438*

Die komplette Liste der AC500-Fehlermeldungen finden Sie in ↳ *[3]*.

HINWEIS!

Die Fehlermeldungen der Sicherheits-CPU und der Sicherheits-E/A-Module werden auf der Anzeige der Standard-CPU angezeigt.

Ein Überlauf der Fehlermeldungen auf der Sicherheits-CPU ist nicht möglich. Die maximale Zahl der Einträge im Diagnosesystem der Sicherheits-CPU ist 100. Wenn alle 100 Einträge des Diagnosesystems belegt sind, werden die ältesten Einträge durch die neuesten überschrieben.

Nach einem Power Cycle der Sicherheits-CPU sind Fehlermeldungen vom Diagnosesystem der Sicherheits-CPU gelöscht.

3.1.5 Zustände der Sicherheits-CPU

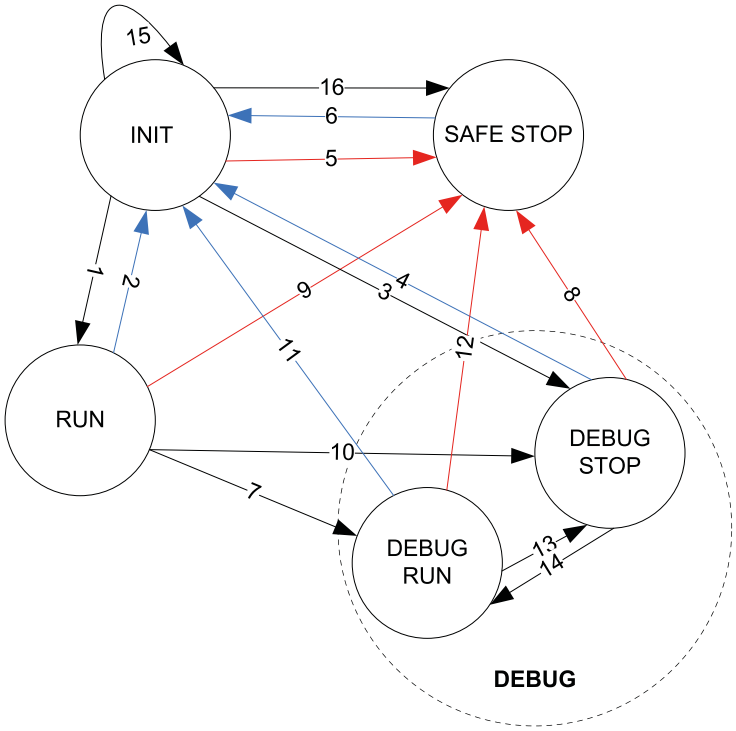


Abb. 12: Zustände ↳ Kapitel 3.1.5.1 „Beschreibung der Zustände der Sicherheits-CPU“ auf Seite 51 und Übergänge ↳ Kapitel 3.1.5.2 „Übergänge zwischen Zuständen der Sicherheits-CPU “ auf Seite 53 der Sicherheits-CPU

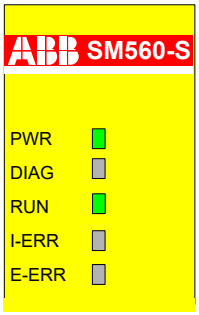
Power Cycle oder SPS-Browser-/SPS-Shell-Befehl *„Reboot“* von der Standard-CPU
Fehler mit Schweregrad 1 oder 2
Weitere Übergänge

3.1.5.1 Beschreibung der Zustände der Sicherheits-CPU

INIT

Dies ist ein zeitlich beschränkter Systemstatus während des internen Sicherheitsdiagnosetests und Startvorgangs. Informationen zu den LED-Zuständen finden Sie unter Abb. 11, Seite 50.

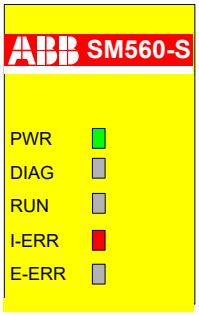
RUN



In diesem Zustand wird die Sicherheitsanwendung regulär ausgeführt, sofern das Bootprojekt geladen ist. Kein Fehler mit Schweregrad 1 oder 2 vorhanden.

In AC500-S Programming Tool stehen den Anwendern alle Online-Menüpunkte aus dem *„Online“*-Menü zur Verfügung, aber nur drei davon können ohne Verlassen des Zustands RUNs ausgeführt werden: *„Einloggen“*, *„Ausloggen“* und *„Prüfe Bootprojekt der Steuerung“*. Durch sämtliche anderen Optionen (z. B. Setzen eines Breakpoints) wird die Sicherheits-CPU in einen nicht sicheren DEBUG-Zustand versetzt (DEBUG RUN oder DEBUG STOP).

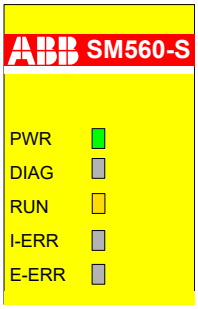
SAFE STOP



Nachdem ein Fehler mit Schweregrad 1 oder 2 erkannt wurde, geht die Sicherheits-CPU in den Zustand SAFE STOP. Sämtliche PROFIsafe-Ausgangstelegramme werden auf Null gesetzt (in diesem Zustand werden keine gültigen PROFIsafe-Telegramme generiert). Den Anwendern stehen in AC500-S Programming Tool keine Online-Menüpunkte aus dem *„Online“*-Menü zur Verfügung.

Dieser Zustand kann nur durch einen Power Cycle oder durch Eingabe des SPS-Browser-/SPS-Shell-Befehls *„Reboot“* in der Standard-CPU verlassen werden.

DEBUG RUN



Der Zustand DEBUG RUN (nicht sicher) wird erreicht, wenn die Online-Menüpunkte aus dem *„Online“*-Menü aus dem sicheren Modus RUN verwendet werden (außer *„Einloggen“*, *„Ausloggen“* und *„Prüfe Bootprojekt der Steuerung“*). Der Anwender kann einen Breakpoint im Sicherheitsprogramm definieren, einen *„Einzelschritt“* ausführen, Variablenwerte erzwingen und schreiben und andere in AC500-S Programming Tool verfügbare Debugging-Funktionen ausführen.

Wenn der Online-Menüpunkt *„Stop“* aufgerufen wird oder ein Breakpoint im Sicherheitsprogramm erreicht wird, schaltet die Sicherheits-CPU in den Zustand DEBUG STOP (nicht sicher).

Im Zustand DEBUG RUN werden gültige PROFIsafe-Telegramme generiert. Der Zustand DEBUG RUN ist nicht sicher; deshalb liegt die Verantwortung für einen sicheren Prozessablauf ausschließlich bei der Person oder Organisation, die den Modus DEBUG RUN (nicht sicher) aktiviert hat.

Zum sicheren Modus RUN kann man erst nach einem Power Cycle oder durch Eingabe des SPS-Browser-/SPS-Shell-Befehls *„Reboot“* an der Standard-CPU zurückkehren.

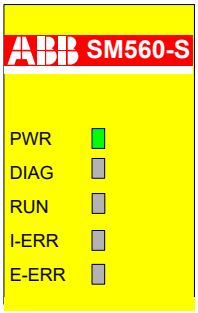
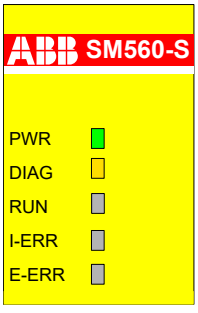
GEFAHR!

Die Sicherheitsfunktionalität und infolgedessen der sichere Prozessablauf werden von der Sicherheits-CPU im Modus DEBUG RUN (nicht sicher) oder DEBUG STOP (nicht sicher) nicht länger garantiert.

Wenn der Modus DEBUG RUN (nicht sicher) oder DEBUG STOP (nicht sicher) auf der Sicherheits-CPU aktiviert wird, liegt die **Verantwortung für einen sicheren Prozessablauf ausschließlich bei der Person oder Organisation**, die den Modus DEBUG RUN (nicht sicher) oder DEBUG STOP (nicht sicher) aktiviert hat.

Mit der POE SF_SAFETY_MODE kann die Information abgefragt werden, ob die Sicherheits-CPU im Modus SAFETY oder DEBUG (nicht sicher) läuft, sodass ggf. die Ausführung des Anwenderprogramms beschränkt oder gestoppt werden kann *Kapitel 4.6.7.7 „SF_SAFETY_MODE“ auf Seite 349*.

DEBUG STOP

Ohne Fehler mit Schweregrad 3 oder 4	Mit Fehler mit Schweregrad 3 oder 4
	

In diesem nicht sicheren Zustand kann ein Anwender, ähnlich wie in DEBUG RUN, durch Setzen von Breakpoints etc. in die Ausführung des Sicherheitsprogramms eingreifen. Das Sicherheitsprogramm wird im Zustand DEBUG STOP (nicht sicher) nicht ausgeführt. Der PROFIsafe F-Host und die F-Devices (SM560-S-FD-1 und SM560-S-FD-4) der Sicherheits-CPU senden PROFIsafe-Telegramme mit Failsafe-„0“-Werten und setzen FV_activated für alle Sicherheits-E/A-Module und F-Devices.

GEFAHR!

Da der PROFIsafe F-Host weiter im Zustand DEBUG STOP (nicht sicher) läuft, ist es möglich, passivierte Sicherheits-E/A-Module zu reintegrieren und in den sicherheitsgerichteten Zustand RUN zu bringen. Man kann Variablen für Sicherheits-E/A-Module forcieren, beispielsweise um Sicherheitsausgänge zu aktivieren.

Wenn der Modus DEBUG RUN (nicht sicher) oder DEBUG STOP (nicht sicher) auf der Sicherheits-CPU aktiviert wird, **liegt die Verantwortung für einen sicheren Prozessablauf ausschließlich bei der Person oder Organisation,** die den Modus DEBUG RUN (nicht sicher) oder DEBUG STOP (nicht sicher) aktiviert hat.

Wenn der Online-Menüpunkt *„RUN“* im Sicherheitsprogramm aufgerufen wird, schaltet die Sicherheits-CPU in den Zustand DEBUG RUN.

Alle Online-Menüpunkte stehen den Anwendern in diesem Zustand zur Verfügung.

Bei den Online-Befehlen *„Einzelschritt in“*, *„Einzelschritt über“* und *„Einzelschritt“* oder wenn ein Breakpoint erreicht wird, wird zwischen DEBUG RUN und DEBUG STOP umgeschaltet (Übergänge 13 und 14 in Abb. 12, Seite 51).

Zum sicheren Modus RUN kann man erst nach einem Power Cycle oder durch Eingabe des SPS-Browser-/SPS-Shell-Befehls *„Reboot“* an der Standard-CPU zurückkehren.

3.1.5.2 Übergänge zwischen Zuständen der Sicherheits-CPU

Übergang (Abb. 12, Seite 51)	Von	Zu	Beschreibung
(1)	INIT	RUN	● Erfolgreiche Initialisierung ● Bootprojekt ist verfügbar; es gibt keinen Konfigurations- oder anderen Fehler mit Schweregrad 1 oder 2.
(2)	RUN	INIT	Power Cycle oder SPS-Browser-/SPS-Shell-Befehl *„Reboot“* von der Standard-CPU
(3)	INIT	DEBUG STOP	● Erfolgreiche Initialisierung ● Kein Bootprojekt vorhanden oder Fehler mit Schweregrad 3 ● Schalteradresse 0xFF wurde in der Sicherheits-CPU gesetzt
(4)	DEBUG STOP	INIT	Power Cycle oder SPS-Browser-/SPS-Shell-Befehl *„Reboot“* von der Standard-CPU
(5)	INIT	SAFE STOP	● Ein Fehler mit Schweregrad 1 oder 2 wurde während der Initialisierung erkannt ● Aktualisierung von Firmware oder Bootcode war nicht erfolgreich
(6)	SAFE STOP	INIT	Power Cycle oder SPS-Browser-/SPS-Shell-Befehl *„Reboot“* von der Standard-CPU

Übergang (Abb. 12, Seite 51)	Von	Zu	Beschreibung
(7)	RUN	DEBUG RUN	Die Online-Menüpunkte *„Breakpoint ein/aus“*, *„Werte schreiben“*, *„Werte forcen“* oder *„Einzelschritt“* von AC500-S Programming Tool wurden verwendet.
(8)	DEBUG STOP	SAFE STOP	Ein Fehler mit Schweregrad 1 oder 2 wurde erkannt
(9)	RUN	SAFE STOP	Ein Fehler mit Schweregrad 1 oder 2 wurde erkannt
(10)	RUN	DEBUG STOP	● Online-Menüpunkte *„Stop“*, *„Quellcode laden“* oder *„Reset“* (verschiedene) von AC500-S Programming Tool ● *[RUN]*-Knopf der Standard-CPU (Standard-CPU war im Modus RUN) wurde gedrückt ● Online-Menüpunkte *„Stop“* oder *„Reset“* (verschiedene) von der Standard-CPU ● Neues sicherheitsgerichtetes Bootprojekt wird geladen
(11)	DEBUG RUN	INIT	Power Cycle oder SPS-Browser-/SPS-Shell-Befehl *„Reboot“* von der Standard-CPU
(12)	DEBUG RUN	SAFE STOP	Ein Fehler mit Schweregrad 1 oder 2 wurde erkannt
(13)	DEBUG RUN	DEBUG STOP	● Online-Menüpunkte *„Stop“* oder *„Reset“* (verschiedene) von AC500-S Programming Tool ● *[RUN]*-Knopf der Standard-CPU (Standard-CPU war im Modus RUN) wurde gedrückt ● Online-Menüpunkte *„Stop“* oder *„Reset“* (verschiedene) von der Standard-CPU ● Breakpoint wurde beim Debugging erreicht ● Am Ende des Sicherheits-CPU-Zyklus im „Einzelschritt“-Debugging-Modus ● Neues sicherheitsgerichtetes Bootprojekt wird geladen
(14)	DEBUG STOP	DEBUG RUN	● Online-Menüpunkte *„Einzelschritt über“*, *„Einzelschritt in“* und *„Run“* von AC500-S Programming Tool ● Online-Menüpunkt *„Run“* von der Standard-CPU ● *[RUN]*-Knopf der Standard-CPU (Standard-CPU war im STOP-Modus) wurde gedrückt
(15)	INIT	INIT	Power Cycle oder SPS-Browser-/SPS-Shell-Befehl *„Reboot“* von der Standard-CPU
(16)	INIT	SAFE STOP	Schalteradresse 0xFE, 0xFD oder 0xFC wurde in der Sicherheits-CPU gesetzt

3.1.6 Interaktion zwischen Sicherheits- und Standard-CPU

Die Sicherheits-CPU und die Standard-CPU haben jeweils ihre eigene Firmware, ein eigenes Bootprojekt und Anwendungsprogramm, die getrennt voneinander ausgeführt werden. Das einzige Steuerelement der Standard-CPU-Hardware, das den Status von Standard-CPU und Sicherheits-CPU ändern kann, ist der *[RUN]*-Knopf der Standard-CPU. Der *[RUN]*-Knopf der Standard-CPU kann sowohl Sicherheits-CPU als auch Standard-CPU gleichzeitig stoppen und starten. Dieses Verhalten des *[Run]*-Knopfs hängt von den Einstellungen der Standard-CPU ab ↳ *[3]*. Eine gestoppte Sicherheits-CPU bedeutet, dass nur die Ausführung des Anwen-

dungsprogramms gestoppt wurde. PROFIsafe F-Host und F-Device-Stacks ↳ *[2]* laufen im Failsafe-Modus weiter. Alle Sicherheits-E/A-Module werden passiviert und „0“-Ersatzwerte werden für Sicherheits-E/As und F-Devices verwendet. Die Ausführung von PROFIsafe F-Host und F-Device-Stack kann nur durch den Wechsel in den Zustand SAFE STOP gestoppt werden. In diesem Fall werden keine PROFIsafe-Telegramme generiert und die LED I-ERR leuchtet.

GEFAHR!

Es ist nicht möglich, die Sicherheits-CPU mit der Taste *[RUN]* der Standard-CPU zu starten. Die Sicherheits-CPU wechselt immer in den nicht sicheren DEBUG-Modus (DEBUG RUN oder DEBUG STOP), sobald der *[RUN]*-Knopf an der Standard-CPU gedrückt wird ↳ *Kapitel 3.1.5.1 „Beschreibung der Zustände der Sicherheits-CPU“ auf Seite 51*. Durch einen Power Cycle der Sicherheits-CPU oder den SPS-Browser-/SPS-Shell-Befehl *„Reboot“* der Standard-CPU wird die Sicherheits-CPU wieder in den sicheren Modus RUN geschaltet.

Die Befehle *„Run“* und *„Stop“* in der Engineering Suite haben dieselbe Auswirkung auf die Sicherheits-CPU und die Standard-CPU wie die Taste *[RUN]* an der Standard-CPU.

Einige Parameter der Standard-CPU-Konfiguration beeinflussen das Gesamtverhalten der Sicherheits- und der Standard-CPU ↳ *Anhang B.3 „Konfiguration der AC500 V2-Standard-CPU-Parameter“ auf Seite 427* ↳ *Anhang C.3 „Konfiguration der AC500 V3-Standard-CPU-Parameter“ auf Seite 445*.

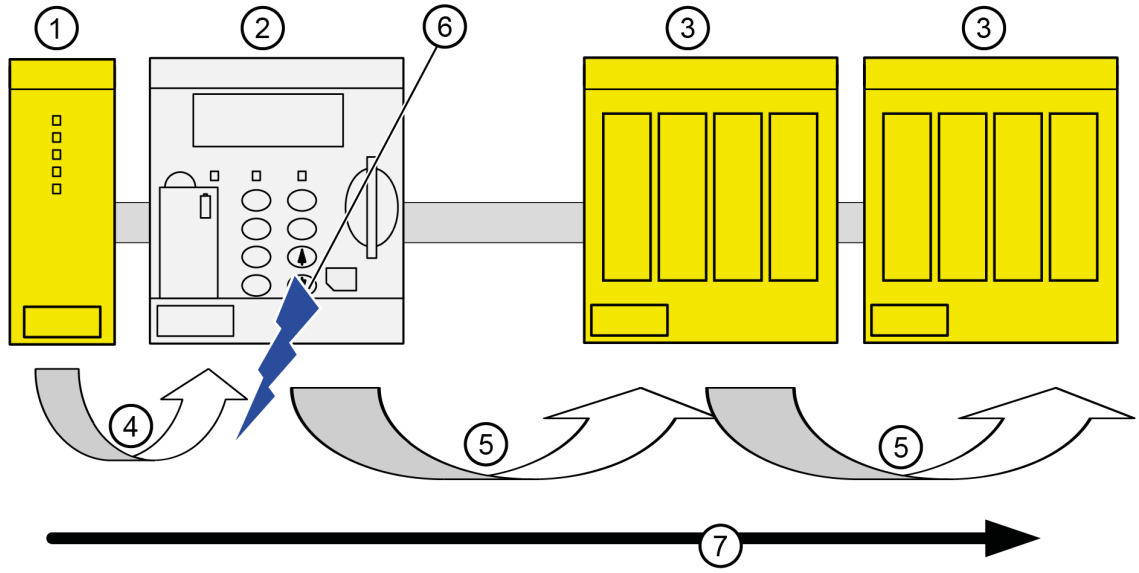


Abb. 13: Einfluss von Parametereinstellungen der Standard-CPU auf den Sicherheitstelegrammfluss

1 Sicherheits-CPU
2 Standard-CPU
3 Sicherheits-E/A-Modul
4 Gültiges Sicherheitstelegramm
5 Telegramm mit „0“-Werten oder gültiges Sicherheitstelegramm
6 Einstellungen der Standard-CPU
7 Sicherheitstelegramme der Sicherheits-CPU mit Ausgangswerten

3.1.7 Parametrierung

Die Einrichtung der Parameterdaten wird mit der System-Konfigurationssoftware Automation Builder durchgeführt.

Nr.	Name	Werte	Standard
1	Minimale Aktualisierungs-Zeit (mit AC500 V2-Standard-CPU) Aktualisierungszyklus-Zeit (mit AC500 V3-Standard-CPU)	1-20000 ms	„10 ms“
2	Debug-Modus aktivieren	„Ein“, „Aus“	„Aus“
3	Zeitüberschreitung PROFIsafe Start	0-65535 ms	„0 ms“

„Minimale Aktualisierungs-Zeit“ / „Aktualisierungszyklus-Zeit“

Abhängig von der eingesetzten AC500-CPU wird der Parameter für die Zykluszeit unterschiedlich bezeichnet. Die Bedeutung beider Parameter ist identisch.

Beachten Sie, dass dieser Parameter die Antwortzeit der Sicherheitsfunktion beeinflusst. Je kleiner der Wert ist, desto kürzer ist die Antwortzeit der Sicherheitsfunktion ↬ *Kapitel 5.1 „Übersicht“ auf Seite 363*. Gleichzeitig steigt jedoch die Standard-CPU-Last, je kleiner der Wert von *„Minimale Aktualisierungs-Zeit“* bzw. *„Aktualisierungszyklus-Zeit“* ist.

GEFAHR!

Hohe Werte (z. B. > 10 ms) für den Parameter *„Minimale Aktualisierungs-Zeit“* bzw. *„Aktualisierungszyklus-Zeit“* erhöhen die Wahrscheinlichkeit, dass keine Eingangs-Impulssignale mit einer Länge von < *„Minimale Aktualisierungs-Zeit“* bzw. *„Aktualisierungszyklus-Zeit“* an die Sicherheits-CPU gesendet werden.

„Debug-Modus aktivieren“

Steht dieser Parameter auf *„Aus“*, kann kein neues Bootprojekt in die Sicherheits-CPU geladen werden, und ein Debuggen ist nicht möglich.

Wenn ein neues Bootprojekt in die Sicherheits-CPU geladen werden muss, muss zuvor ein neues Bootprojekt in die Standard-CPU geladen werden. Dabei muss der Parameter *„Debug-Modus aktivieren“* für die Sicherheits-CPU *„EIN“* sein. Nach einem Neustart der Standard-CPU kann ein neues Bootprojekt in die Sicherheits-CPU geladen werden.

Beachten Sie, dass die folgenden SPS-Browser-Befehle in der Sicherheits-CPU nur unterstützt werden, wenn der Parameter *„Debug-Modus aktivieren“* auf *„EIN“* gesetzt ist ↬ *Liste aller SPS-Browser-Befehle*:

- resetprg – Reset des Programms der Sicherheits-CPU
- resetprgorg – Originalzustand des Programms der Sicherheits-CPU wiederherstellen
- setpwd – Login-Passwort der Sicherheits-CPU setzen
- delpwd – Login-Passwort der Sicherheits-CPU löschen
- delappl – Anwenderprogramm löschen
- deluserdat – Nutzerdatensegmente löschen

„Zeitüberschreitung PROFIsafe Start“

Dieser Sicherheits-CPU-Parameter definiert, wie lange die Sicherheits-CPU während des Startens auf die F-Device-Kommunikation warten soll. Bei einer abgelaufenen Zeitüberschreitung passiviert die Sicherheits-CPU das F-Device, wodurch eine Reintegration durch den Anwender erzwungen wird.

Wert = 0 deaktiviert jegliche Überwachung der Zeitüberschreitung für alle F-Devices (keine Überwachung der Zeitüberschreitung).

Der Wert dieses Parameters ist für alle F-Devices gültig.

3.1.8 Technische Daten

Weitere technische Daten stehen im SPS-Katalog von ABB zur Verfügung: www.abb.com/plc.

HINWEIS!
Die Version -XC der Sicherheits-CPU ist für eine Verwendung unter extremen Umgebungsbedingungen erhältlich ↬ *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Speicher

Angabe	Wert	Einheit
Anwender-Programmspeicher von SM560-S	1	MB
Anwender-Programmspeicher von SM560-S-FD-1 und SM560-S-FD-4	1,3	MB
Speicher für Nutzerdaten (davon 120 kB gespeichert)	1	MB

Leistung

Angabe	Wert	Einheit
Zyklusdauer – binär	0,05	µs/Befehl
Zyklusdauer – Wort	0,06	µs/Befehl
Zyklusdauer – Fließkomma	0,50	µs/Befehl

Spannungen laut EN 61131-2

Angabe	Wert	Einheit
Prozess- und Versorgungsspannung (ohne Restwelligkeit)	24 (-15 %, +20 %)	V DC
Absolute Grenzwerte (inklusive Restwelligkeit)	19,2 … 30	V DC
Restwelligkeit	< 5	%
Verpolschutz	10	s

GEFAHR!
Das Überschreiten der zulässigen Prozess- oder Versorgungsspannung (< -35 V DC bzw. > +35 V DC) kann zu irreparablen Schäden am System führen.

Erlaubte Unterbrechungen der Spannungsversorgung laut EN 61131-2

Angabe	Wert	Einheit
Unterbrechungen der Gleichstromversorgung	< 10	ms
Zeit zwischen 2 Unterbrechungen der Gleichstromversorgung, PS2	> 1	s

Umgebungsbedingungen

Angabe	Wert	Einheit
Betriebstemperatur*	0 … +60	°C
Lagerungstemperatur	-40 … +85	°C
Transporttemperatur	-40 … +85	°C
Luftfeuchtigkeit ohne Kondensation	max. 95	%
Betriebsluftdruck	> 800	hPa

Angabe	Wert	Einheit
Lagerluftdruck	> 660	hPa
Betriebshöhe	< 2000	m über NN
Lagerhöhe	< 3500	m über NN

* Erweiterte Temperaturbereiche (unter 0 °C und über +60 °C) werden von Sonderversionen der Sicherheits-CPU unterstützt ↳ *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Kriech- und Luftstrecken

Die Kriech- und Luftstrecken entsprechen der Überspannungskategorie II, Verschmutzungsgrad 2.

Netzteile

Zur Versorgung der Module müssen Netzteile gemäß PELV-/SELV-Spezifikationen verwendet werden.

Elektromagnetische Verträglichkeit

Informationen zur elektromagnetischen Verträglichkeit finden Sie im neuesten TÜV SÜD Report ↳ *[1]*.

Mechanische Eigenschaften

Angabe	Wert	Einheit
Montage	Horizontal (oder vertikal mit Leistungsreduzierung (maximale Betriebstemperatur auf +40 °C reduziert))	
Schutzart	IP 20	
Gehäuse	gemäß UL94	
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 3,5 mm	2 … 15	Hz
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 1 g *	15 … 150	Hz
Stoßprüfung (alle drei Achsen), 11 ms Halbsinus	15	g
MTBF	168	Jahre

* Höhere Werte auf Anfrage

Selbsttest und Diagnosefunktionen

Tests während Start und Betrieb: Programmablauf-Überwachung, RAM, CPU usw.

Abmessungen, Gewicht

Angabe	Wert	Einheit
B × H × T	28 × 135 × 75	mm
Gewicht	~ 100	g

Zertifizierungen

CE, cUL (weitere Zertifizierungen unter *www.abb.com/plc*)

3.1.9 Bestelldaten

Typ	Beschreibung	Bestellnummer
SM560-S	Sicherheitsmodul – CPU, sicherheitsgerichtetes Modul bis SIL 3	1SAP 280 000 R0001
SM560-S-XC	Sicherheitsmodul – CPU, sicherheitsgerichtetes Modul bis SIL 3, extreme Umgebungsbedingungen	1SAP 380 000 R0001
SM560-S-FD-1	Sicherheitsmodul – CPU, sicherheitsgerichtetes Modul bis SIL 3 mit F-Device-Funktionalität für 1 PROFIsafe-Netzwerk	1SAP 286 000 R0001
SM560-S-FD-1-XC	Sicherheitsmodul – CPU, sicherheitsgerichtetes Modul bis SIL 3 mit F-Device-Funktionalität für 1 PROFIsafe-Netzwerk, extreme Umgebungsbedingungen	1SAP 386 000 R0001
SM560-S-FD-4	Sicherheitsmodul – CPU, sicherheitsgerichtetes Modul bis SIL 3 mit F-Device-Funktionalität für bis zu 4 PROFIsafe-Netzwerke	1SAP 286 100 R0001
SM560-S-FD-4-XC	Sicherheitsmodul – CPU, sicherheitsgerichtetes Modul bis SIL 3 mit F-Device-Funktionalität für bis zu 4 PROFIsafe-Netzwerke, extreme Umgebungsbedingungen	1SAP 386 100 R0001

3.2 Allgemeines Verhalten des Sicherheits-E/A-Moduls

3.2.1 Übersicht

Alle Sicherheits-E/A-Module (AI581-S, DI581-S und DX581-S) können in einer zentralen oder dezentralen Konfiguration mit PROFINET/PROFIsafe (Abb. 3, Seite 24) verwendet werden. PROFINET-Geräte CI501-PNIO, CI502-PNIO, CI504-PNIO und CI506-PNIO können zum Anschluss der Sicherheits-E/A-Module in dezentralen Konfigurationen verwendet werden. Sicherheits-E/A-Module können frei mit Standard-E/As aus den Produktfamilien AC500 und AC500-eCo kombiniert werden.

HINWEIS!
Die Firmware-Aktualisierung für Sicherheits-E/A-Module kann aktuell nur durch Fachpersonal im Werk von ABB vorgenommen werden.

3.2.2 Zustände des Sicherheits-E/A-Moduls

Die Zustände der Sicherheits-E/A-Module können mit den folgenden zwei Zustandsdiagrammen beschrieben werden.

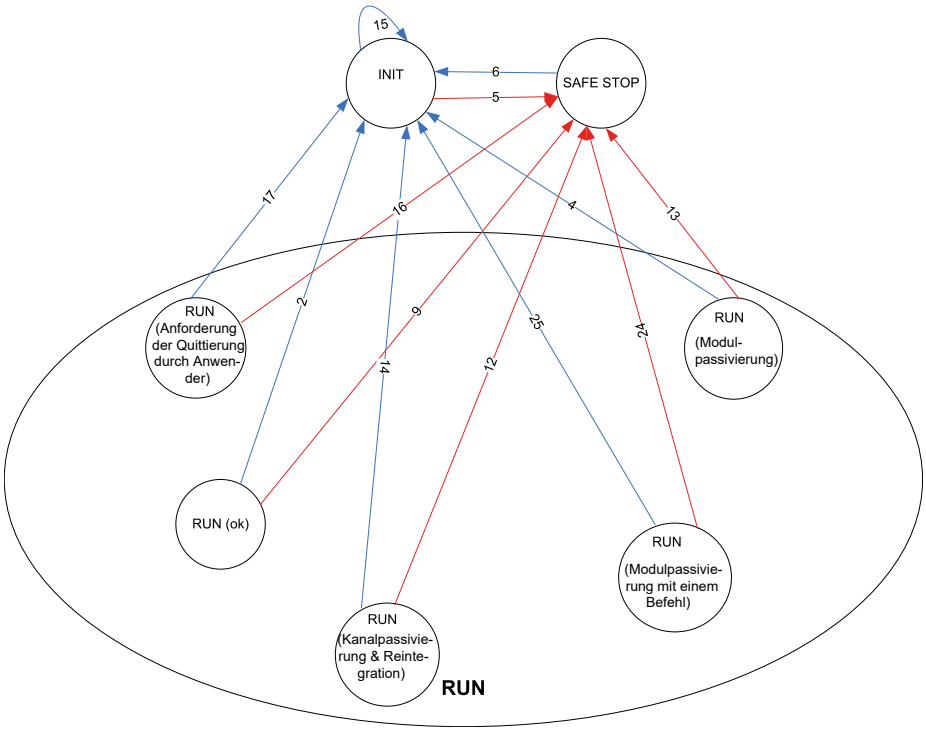


Abb. 14: Überblick der Übergänge im Zusammenhang mit Power Cycles und Fehlern mit Schweregrad 1 bei Sicherheits-E/A-Modulen

Power Cycle
Fehler mit Schweregrad 1

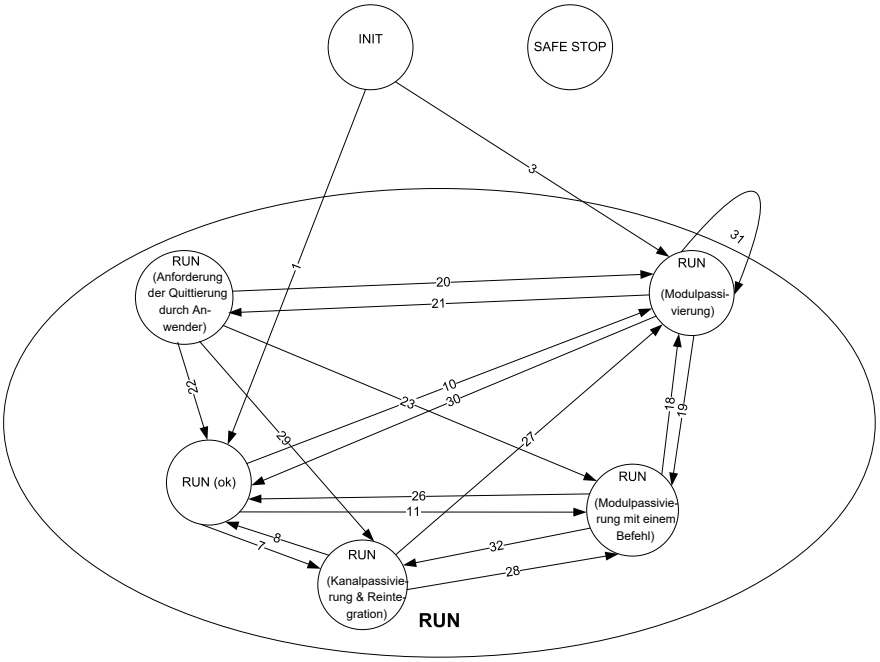


Abb. 15: Überblick der Übergänge von Sicherheits-E/A-Modulen (mit Ausnahme von Power Cycles und Fehlern mit Schweregrad 1)

Übergänge

3.2.2.1 Beschreibung der Zustände des Sicherheits-E/A-Moduls

INIT

Die Hardware wird initialisiert und interne Starttests der Sicherheits-E/A-Module werden durchgeführt. Informationen zu den LED-Zuständen finden Sie unter Abb. 16, Seite 69. Nach einer erfolgreichen Parametrierung wird der Start der PROFIsafe-Kommunikation durch den PROFIsafe F-Host erwartet.

Das Sicherheits-E/A-Modul bleibt in diesem Zustand:

- solange eine Unterspannung erkannt wird
- wenn die Parametrierung fehlgeschlagen ist oder noch aussteht
- wenn die PROFIsafe-Kommunikation noch aussteht

Die Anwender müssen sicherstellen, dass ein dediziertes Qualifier-Ausgangsbit (PROFIsafe-Diagnose) für mindestens einen Kanal des entsprechenden Sicherheits-E/A-Moduls auf „1“ gesetzt ist, um zu prüfen, ob die PROFIsafe F-Devices initialisiert wurden.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul:

OA_Req_S = 0

FV_activated_S = 1

Device_Fault = 0

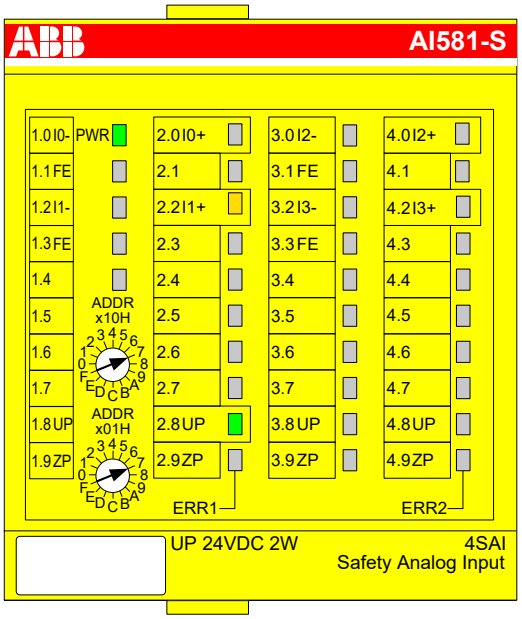
Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 0

Kanal-Prozesswert = 0

Bit für Reintegrationsanforderung = 0

RUN (OK)



Die PROFIsafe-Kommunikation läuft. Die Sicherheitsanwendung läuft ohne Fehler.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul:

OA_Req_S = 0

FV_activated_S = 0

Device_Fault = 0

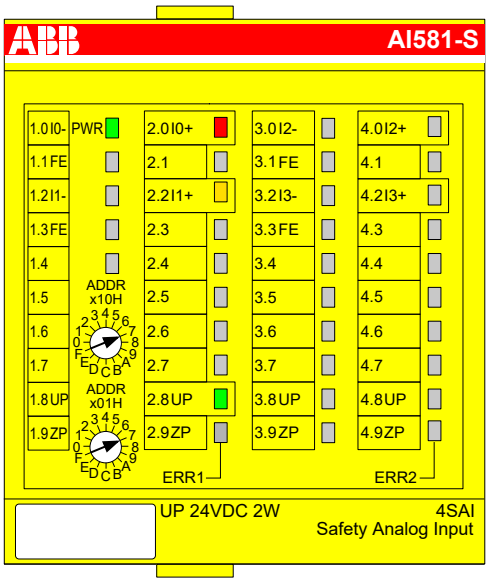
Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 1

Kanal-Prozesswert = Prozesswert

Bit für Reintegrationsanforderung = 0

RUN (Kanalpassivierung und -reintegration)



Die PROFIsafe-Kommunikation läuft. Die Sicherheitsanwendung läuft mit erkannten Kanalfehlern.

Ein Kanalfehler (z. B. kein erwarteter Testimpuls, Diskrepanzzeit usw.) wurde an mindestens einem Kanal festgestellt. Der Failsafe-Wert („0“) wird für den/die passivierten Eingangskanal/-kanäle zum PROFIsafe F-Host übertragen. Die entsprechenden PROFIsafe-Diagnosebits werden auch auf „0“ gesetzt, um die Verwendung von Failsafe-Werten anzuzeigen.

Ein passivierter Ausgangskanal hat den Status „0“. Die entsprechenden PROFIsafe-Diagnosebits werden auch auf „0“ gesetzt, um die Verwendung von Failsafe-Werten anzuzeigen.

Sobald der Kanalfehler behoben wurde (z. B. falsche Verdrahtung wurde korrigiert; dies gilt nur für Fehler, die quittierbar sind), schaltet das Bit zur Reintegrationsanforderung des entsprechenden Kanals auf „1“, was der Sicherheitsanwendung, die auf der Sicherheits-CPU läuft, anzeigt, dass der Kanal wieder integriert werden kann. Durch Setzen des Reintegrationsquittierungs-Bits („Acknowledge Reintegration“) von „0“ auf „1“ wird die Reintegration des angegebenen Kanals veranlasst. Eine positive Flanke von „0“ auf „1“ ist erforderlich, um die Reintegration des Kanals zu quittieren.

Sobald alle Kanalfehler behoben und quittiert wurden, ist der Zustand RUN (OK) wieder erreicht.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul:

OA_Req_S = 0

FV_activated_S = 0

Device_Fault = 0

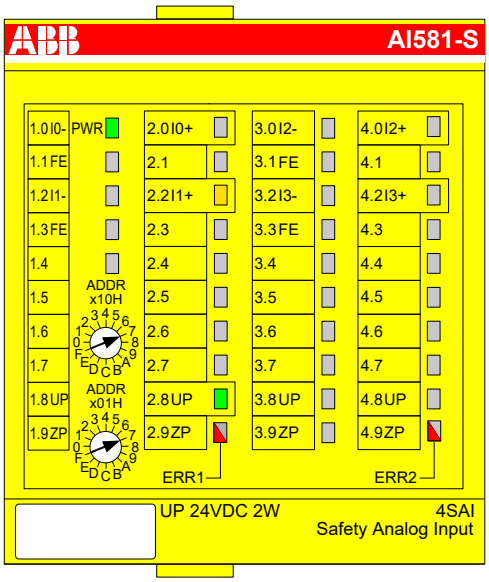
Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 0

Kanal-Prozesswert = 0

Bit für Reintegrationsanforderung = 0, wenn ein Fehler vorliegt; „1“, wenn der Kanal wieder integriert werden kann.

RUN (Modulpassivierung): Abwechselndes Blinken der LEDs ERR1 und ERR2



Die PROFIsafe-Kommunikation läuft. Die Sicherheitsanwendung läuft mit einem Modulfehler.

Das Modul und seine Kanäle werden passiviert. Mögliche Gründe für die Modulpassivierung sind folgende:

- Fehler der PROFIsafe-Kommunikation (CRC-Fehler)
- Zeitüberschreitung beim PROFIsafe-Watchdog
- Über- oder Unterspannung erkannt (Device_Fault-Statusbit = 1)

Der Failsafe-Wert „0“ wird für alle passivierten Eingangskanäle an die Sicherheitssteuerung übertragen, wenn die Verbindung zum PROFIsafe F-Host möglich ist. Die Sicherheitsanwendung versucht ständig, eine Kommunikation zur Sicherheits-CPU aufzubauen, wenn diese abgebrochen ist. Alle passivierten Ausgangskanäle haben den Status „0“.

Ein Übergang in einen anderen Modus RUN ist nur möglich, wenn der festgestellte Fehler behoben wurde.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul (wenn die Kommunikation möglich ist):

OA_Req_S = 0

FV_activated_S = 1

Device_Fault = 1 (bei erkannter Unter- oder Überspannung) und/oder CE_CRC = 1 (bei Kommunikationsfehler) und/oder WD_timeout = 1 (bei Watchdog-Zeitüberschreitung)

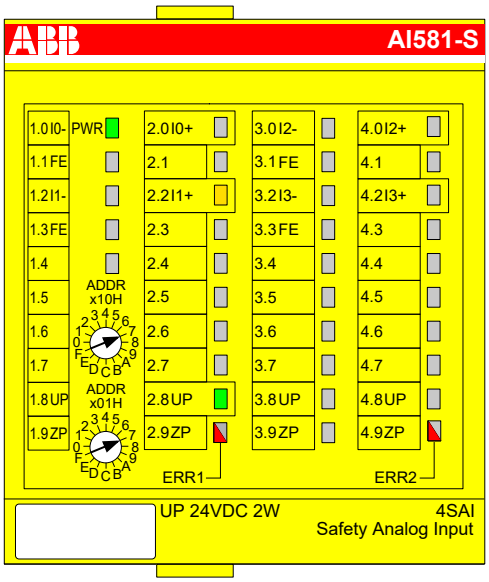
Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 0

Kanal-Prozesswert = 0

Bit für Reintegrationsanforderung = 0

RUN (Modulpassivierung mit einem Befehl): Abwechselndes Blinken der LEDs ERR1 und ERR2



Die PROFIsafe-Kommunikation läuft. Die Sicherheitsanwendung läuft ohne Fehler.

Das Modul und alle seine Kanäle werden passiviert, weil die Sicherheitsanwendung auf der Sicherheits-CPU eine Modulpassivierung angefordert hat (activate_FV_C = 1 wurde gesetzt).

Der Failsafe-Wert („0“) wird für alle passivierten Eingangskanäle an die Sicherheits-CPU übertragen. Alle passivierten Ausgangskanäle haben den Status „0“. Das/die PROFIsafe-Diagnosebit(s) für alle Kanäle haben den Status „0“, um anzuzeigen, dass Failsafe-Werte übertragen werden.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul:

FV_activated_S = 1

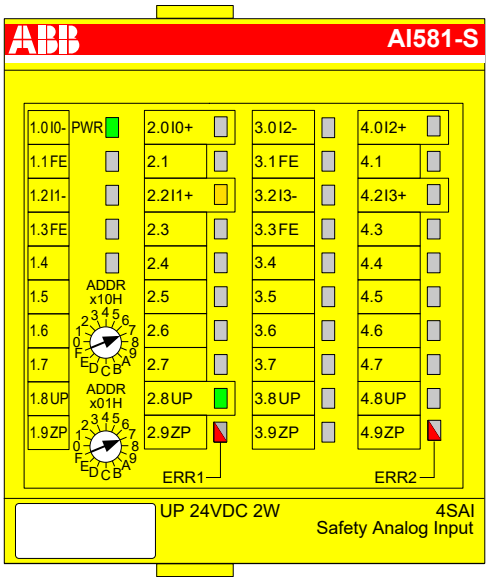
Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 0

Kanal-Prozesswert = 0

Bit für Reintegrationsanforderung = 0

RUN (Anforderung der Quittierung durch Anwender): Abwechselndes Blinken der LEDs ERR1 und ERR2



Die PROFIsafe-Kommunikation läuft. Die Sicherheitsanwendung läuft ohne Fehler, wartet aber auf die Quittierung einer Modul-Reintegration (Modulfehler wurde behoben).

Der Failsafe-Wert „0“ wird weiterhin für alle passivierten Eingangskanäle an die Sicherheits-CPU übertragen. Alle passivierten Ausgangskanäle haben den Status „0“. Die PROFIsafe-Diagnosebits für alle Kanäle haben den Status „0“, um anzuzeigen, dass Failsafe-Werte übertragen werden.

Das Bit OA_Req_S ist „1“.

Sobald die Sicherheitsanwendung der Sicherheits-CPU OA_C (positive Flanke) setzt, geht das Sicherheits-E/A-Modul in den Zustand RUN (OK), wenn keine weiteren Fehler erkannt werden. Die positive Flanke muss an das Sicherheits-E/A-Modul gesendet werden, bis OA_Req_S „0“ zurückgibt.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul:

OA_Req_S = 1

FV_activated_S = 1

Device_Fault = 0

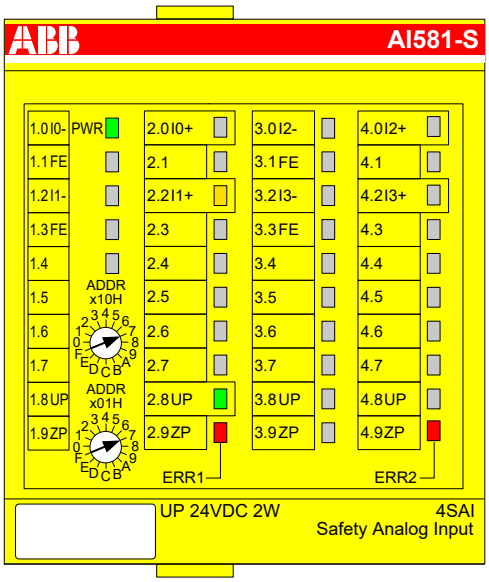
Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 0

Kanal-Prozesswert = 0

Bit für Reintegrationsanforderung = 0

SAFE STOP



Die Ausführung der Sicherheitsanwendung wurde gestoppt. Die PROFIsafe-Kommunikation ist nicht möglich.

Dieser Zustand wird bei einem Fehler mit Schweregrad 1 erreicht (z. B. CPU-Test oder RAM-Test usw. fehlgeschlagen).

Dieser Zustand kann nur durch einen Power Cycle oder den Befehl *„reboot“* von der Standard-CPU oder dem Kommunikationsschnittstellen-Modul verlassen werden.

PROFIsafe-Statusbits im F-Host für Sicherheits-E/A-Modul:

OA_Req_S = 0

FV_activated_S = 1

Device_Fault = 0

Prozessdatenbits im Prozessabbild des Sicherheits-E/A-Moduls:

PROFIsafe-Diagnosebit = 0

Kanal-Prozesswert = 0

Bit für Reintegrationsanforderung = 0

3.2.2.2 Übergänge zwischen Zuständen des Sicherheits-E/A-Moduls

Übergang (Abb. 14, Seite 60, Abb. 15, Seite 60)	Von	Zu	Beschreibung
(1)	INIT	RUN (OK)	Das Sicherheits-E/A-Modul geht direkt nach INIT während eines normalen Starts in diesen Zustand über
(2)	RUN (OK)	INIT	Power Cycle
(3)	INIT	RUN (Modulpassivierung)	PROFIsafe-Watchdog, PROFIsafe-Kommunikationsfehler oder Unter-/Überspannung direkt nach INIT. Das Sicherheits-E/A-Modul geht auch nach einem Power Cycle des Sicherheits-E/A-Moduls in diesen Zustand über, wenn die Sicherheits-CPU mit PROFIsafe F-Host weiterläuft und das Sicherheits-E/A-Modul nach dem Aus-/Einschalten in einen Failsafe-Zustand RUN (Modulpassivierung) versetzt.
(4)	RUN (Modulpassivierung)	INIT	Power Cycle
(5)	INIT	SAFE STOP	Fehler mit Schweregrad 1 (CPU-Test, RAM-Test usw. fehlgeschlagen)
(6)	SAFE STOP	INIT	Power Cycle
(7)	RUN (OK)	RUN (Kanalpassivierung und -reintegration)	Ein Kanalfehler wurde vom Sicherheits-E/A-Modul erkannt. Die Tests werden, wenn möglich, für den entsprechenden Kanal fortgesetzt, um zu prüfen, ob der Kanalfehler behoben wurde (z. B. Verdrahtung wurde korrigiert). Sobald der Fehler behoben wurde, setzt das Modul das Bit für Reintegrationsanforderung für den entsprechenden Kanal auf „1“.
(8)	RUN (Kanalpassivierung und -reintegration)	RUN (OK)	● Der Kanalfehler liegt nicht mehr vor. ● Das Bit „Reintegrationsanforderung“ wird vom Sicherheits-E/A-Modul für den entsprechenden Kanal auf „1“ gesetzt. ● Das Bit „Reintegrationsquittierung“ (positive Flanke) wird vom PROFIsafe F-Host für den entsprechenden Kanal gesetzt.
(9)	RUN (OK)	SAFE STOP	Fehler mit Schweregrad 1 (CPU-Test, RAM-Test usw. fehlgeschlagen)
(10)	RUN (OK)	RUN (Modulpassivierung)	PROFIsafe-Watchdog, PROFIsafe-Kommunikationsfehler oder Unter-/Überspannung wurde erkannt.
(11)	RUN (OK)	RUN (Modulpassivierung mit einem Befehl)	Der Befehl „activate_FV_C = 1“ wurde von der Sicherheits-CPU gesendet
(12)	RUN (Kanalpassivierung und -reintegration)	SAFE STOP	Fehler mit Schweregrad 1 (CPU-Test, RAM-Test usw. fehlgeschlagen)
(13)	RUN (Modulpassivierung)	SAFE STOP	Fehler mit Schweregrad 1 (CPU-Test, RAM-Test usw. fehlgeschlagen)
(14)	RUN (Kanalpassivierung und -reintegration)	INIT	Power Cycle
(15)	INIT	INIT	Power Cycle

Übergang (Abb. 14, Seite 60, Abb. 15, Seite 60)	Von	Zu	Beschreibung
(16)	RUN (Anforderung der Quittierung durch Anwender)	SAFE STOP	Fehler mit Schweregrad 1 (CPU-Test, RAM-Test usw. fehlgeschlagen)
(17)	RUN (Anforderung der Quittierung durch Anwender)	INIT	Power Cycle
(18)	RUN (Modulpassivierung mit einem Befehl)	RUN (Modulpassivierung)	PROFIsafe-Watchdog, PROFIsafe-Kommunikationsfehler oder Unter-/Überspannung wurde erkannt. **Hinweis:** Bei diesem Übergang ist es möglich, dass das Bit WD_timeout der PROFIsafe F-Host-Instanz hin- und herschaltet, wenn eine Zeitüberschreitung des Watchdog vom Sicherheits-E/A-Modul periodisch erkannt wird.
(19)	RUN (Modulpassivierung)	RUN (Modulpassivierung mit einem Befehl)	Wenn der Grenzabschaltwert während der Prozess-Unter- oder Überspannung nicht erreicht wird und die Prozessspannung wieder im normalen Bereich liegt, wird das Sicherheits-E/A-Modul wieder integriert und würde automatisch in den Zustand RUN (OK) gehen; aber kurze Zeit zuvor wurde der Befehl „activate_FV_C = 1“ vom PROFIsafe F-Host-Stack gesendet, wodurch das Sicherheits-E/A-Modul in den Zustand RUN (Modulpassivierung durch Befehl) wechselt.
(20)	RUN (Anforderung der Quittierung durch Anwender)	RUN (Modulpassivierung)	Prozess-Unterspannung/-Überspannung wurde identifiziert.
(21)	RUN (Modulpassivierung)	RUN (Anforderung der Quittierung durch Anwender)	● Modulfehler (Watchdog oder Kommunikationsfehler (CRC)) wurde behoben. **und** ● Befehl activate_FV_C = 0 **dann** ● Sicherheits-E/A-Modul setzt OA_Req_S = 1
(22)	RUN (Anforderung der Quittierung durch Anwender)	RUN (OK)	● OA_Req_S = 1 wurde vom Sicherheits-E/A-Modul gesetzt, nachdem der Modulfehler behoben wurde. ● OA_C (positive Flanke) wurde vom PROFIsafe F-Host für das entsprechende Sicherheits-E/A-Modul gesetzt.
(23)	RUN (Anforderung der Quittierung durch Anwender)	RUN (Modulpassivierung mit einem Befehl)	Der Befehl „activate_FV_C = 1“ wurde vom PROFIsafe F-Host gesendet
(24)	RUN (Modulpassivierung mit einem Befehl)	SAFE STOP	Fehler mit Schweregrad 1 (CPU-Test, RAM-Test usw. fehlgeschlagen)
(25)	RUN (Modulpassivierung mit einem Befehl)	INIT	Power Cycle
(26)	RUN (Modulpassivierung mit einem Befehl)	RUN (OK)	● Kein Modulfehler ● Befehl activate_FV_C = 0

Übergang (Abb. 14, Seite 60, Abb. 15, Seite 60)	Von	Zu	Beschreibung
(27)	RUN (Kanalpassivierung und -reintegration)	RUN (Modulpassivierung)	PROFIsafe-Watchdog, PROFIsafe-Kommunikationsfehler oder Unter-/Überspannung wurde erkannt. **Hinweis:** Bei diesem Übergang ist es möglich, dass das Bit WD_timeout der PROFIsafe F-Host-Instanz hin- und herschaltet, wenn eine Zeitüberschreitung des Watchdog vom Sicherheits-E/A-Modul periodisch erkannt wird.
(28)	RUN (Kanalpassivierung und -reintegration)	RUN (Modulpassivierung mit einem Befehl)	Der Befehl „activate_FV_C = 1“ wurde vom PROFIsafe F-Host-Stack gesendet.
(29)	RUN (Anforderung der Quittierung durch Anwender)	RUN (Kanalpassivierung und -reintegration)	Dieser Übergang ist nur möglich, wenn der Kanalfehler vor oder während der Modulpassivierung erkannt wurde. Als Folge davon versetzt einer der Kanaltests das Sicherheits-E/A-Modul nach der Modul-Reintegration direkt in den Zustand RUN (Kanal-Passivierung und Reintegration).
(30)	RUN (Modulpassivierung)	RUN (OK)	Wenn der Grenzabschaltwert während der Prozess-Unterspannung nicht erreicht wird und die Prozessspannung wieder im normalen Bereich liegt, wird das Sicherheits-E/A-Modul wieder integriert und geht automatisch in den Zustand RUN (OK) über. Wenn der Grenzwert für die Gerätesicherung während der Prozess-Überspannung nicht erreicht wird und die Prozessspannung wieder im normalen Bereich liegt, wird das Sicherheits-E/A-Modul wieder integriert und geht automatisch in den Zustand RUN (OK) über.
(31)	RUN (Modulpassivierung)	RUN (Modulpassivierung)	Wenn zweimal innerhalb 1 s eine Prozess-Unterspannung vorliegt, bleibt das Sicherheits-E/A-Modul im Zustand RUN (Modulpassivierung).
(32)	RUN (Modulpassivierung mit einem Befehl)	RUN (Kanalpassivierung und -reintegration)	Dieser Übergang ist nur möglich, wenn der Kanalfehler während RUN (Modulpassivierung mit einem Befehl) erkannt wurde. Als Folge davon geht das Sicherheits-E/A-Modul nach dem Befehl activate_FV_C = 0 in den Zustand RUN (Kanal-Passivierung und Reintegration) über.

3.2.3 Unterspannung / Überspannung

Wenn beim E/A-Modul eine Unterspannung (< 18 V) erkannt wird, wechselt das Modul in den Zustand RUN (Modulpassivierung), solange die Prozessspannung den Abschaltwert (16 V) nicht unterschreitet, bei dem keine Kommunikation zum PROFIsafe F-Host mehr möglich ist. Wenn der Grenzabschaltwert (16 V) während der Prozess-Unterspannung nicht erreicht wurde und die Prozessspannung wieder im normalen Bereich (≥ ~ 18 V) liegt, wird das Sicherheits-E/A-Modul wieder integriert und geht automatisch in den Zustand RUN (OK) über.

Um eine andauernde ungewollte Modulpassivierung und Reintegration zu verhindern, ist das folgende Feature für Unterspannung verfügbar:

- Anwender müssen permanent das Bit Device_Fault des Sicherheits-E/A-Moduls überwachen; wenn es „1“ ist, muss das Modul mit activate_FV_C = 1 passiviert werden.

Wenn beim E/A-Modul eine Überspannung (> 31,2 V) erkannt wird, wechselt das Modul in den Zustand RUN (Modulpassivierung), bis die Prozessspannung nicht den Grenzwert für die Gerätesicherung (> 35 V) übersteigt, bei der das E/A-Modul beschädigt wird und ausgetauscht werden muss. Wenn der Grenzwert für die Gerätesicherung während der Prozess-Überspannung nicht erreicht wird und die Prozessspannung wieder im normalen Bereich liegt, wird das Sicherheits-E/A-Modul wieder integriert und geht automatisch in den Zustand RUN (OK) über. Um eine andauernde ungewollte Modulpassivierung und Reintegration zu verhindern, ist dasselbe Feature (Überwachung des Bits Device_Fault) wie für Unterspannung verfügbar.

3.2.4 Diagnose

GEFAHR!

Die Diagnosedaten sind nicht sicherheitsrelevant und sollten deshalb in der Sicherheitsanwendung nicht zur Ausführung von Sicherheitsfunktionen verwendet werden.

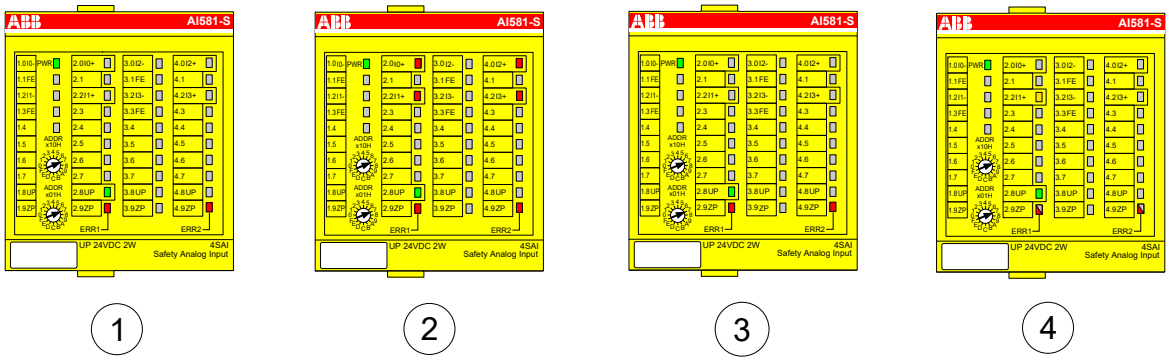


Abb. 16: LED-Anzeige der Sicherheits-E/A-Module während des Starts (Beispiel mit AI581-S)

1 Zustand 1 – Hardware-Reset und Initialisierung
2 Zustand 2 – LED-Test
3 Zustand 3 – Ende der Initialisierung
4 Zustand 4 – Parametrierung beendet, aber noch keine PROFIsafe-Kommunikation

Fehlermeldungen

HINWEIS!

Externe Fehler (an Verkabelung oder Sensoren) an den Sicherheits-E/A-Modulen führen zur Kanalpassivierung („0“-Werte werden zurückgegeben). Sobald ein externer Fehler behoben wird und dies in einem internen Sicherheits-E/A-Modul-Test erkannt wird, verlangen die Kanäle des Sicherheits-E/A-Moduls eine Quittierung zur Reintegration in den normalen Sicherheitsprozessmodus. Anwender können solche Kanäle mit dedizierten Kanalbits quittieren (siehe Abb. 73 auf Seite 159).

Die Fehlermeldungen der Sicherheits-E/A-Module werden zusammen mit den anderen Fehlermeldungen des Moduls in der Standard-CPU gespeichert.

Mit AC500 V2-Standard-CPU: ↬ *Anhang B.2.2 „Fehlermeldungen für Sicherheits-E/A-Module“ auf Seite 425*

Mit AC500 V3-Standard-CPU: ↬ *Anhang C.2.2 „Fehlermeldungen für Sicherheits-E/A-Module“ auf Seite 443*

Die komplette Liste der AC500-Fehlermeldungen finden Sie in ↬ *[3]*.

3.3 Digitales Sicherheits-Eingabemodul DI581-S

Elemente des Moduls

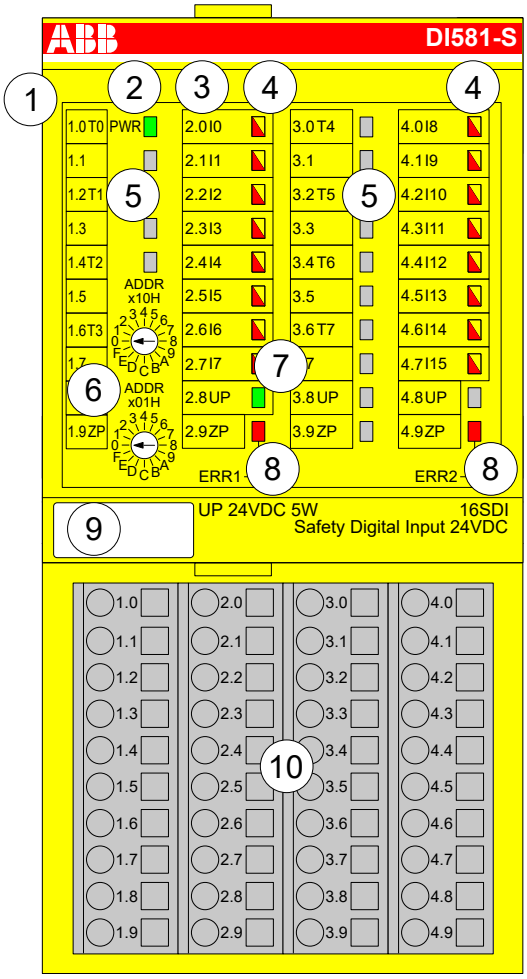


Abb. 17: Digitales Sicherheits-Eingabemodul DI581-S, eingesteckt in Klemmenblock TU582-S

1 I/O-Bus
2 System-LED
3 Zuordnung Klemmennummer – Signalname
4 16 gelb/rote LEDs Signalstatus I0 … I7/I8 … I15
5 8 eindeutige phasenverschobene Testimpuls-Ausgänge T0 … T3/T4 … T7
6 2 Drehschalter für PROFIsafe-Adresse
7 Grüne LED für Prozessspannung UP
8 Rote LEDs zur Anzeige von Modulfehlern
9 Beschriftungsschild (TA525)
10 E/A-Klemmenblock (TU582-S)

3.3.1 Verwendungszweck

Das digitale Sicherheits-Eingabemodul DI581-S kann als dezentrales Erweiterungsmodul für die PROFINET-Module CI501-PNIO, CI502-PNIO, CI504-PNIO und CI506-PNIO oder lokal an CPUs der AC500-Serie für Sicherheitsanwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1) verwendet werden.

HINWEIS!
Die Werte, die mit Ihrer Sicherheitsanwendung für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreicht werden können, hängen von der Verdrahtung der Sensoren im DI581-S-Modul ab ↬ *Kapitel 3.3.7 „Anschluss-beispiele DI581-S“ auf Seite 79*.

DI581-S enthält 16 sicherheitsgerichtete Digitaleingänge 24 V DC aufgeteilt in zwei Gruppen (2.0 … 2.7 und 4.0 … 4.7) ohne Potentialtrennung zwischen den Kanälen.

Die Eingänge sind gegenüber den anderen Schaltkreisen des Moduls nicht galvanisch getrennt.

3.3.2 Funktionalität

Digitaleingänge	16 (24 V DC)
LED-Anzeigen	Für Signalzustand, Modulfehler, Kanalfehler und Versorgungsspannung
Interne Spannungsversorgung	über I/O-Bus-Schnittstelle
Externe Spannungsversorgung	Über Klemmen ZP und UP (Prozessspannung 24 V DC)

Selbsttests und Diagnosefunktionen (sowohl beim Starten als auch während des Betriebs) wie CPU- und RAM-Tests, Programmablauf-Überwachung, Kanalübersprechen und dauerhaftes 1-Signal usw. werden im DI581-S gemäß den Anforderungen von IEC 61508 SIL 3 implementiert.

HINWEIS!
Nur F_Dest_Add wird für die PROFIsafe F-Device-Identifizierung im DI581-S verwendet.

Das DI581-S verfügt über 16 sicherheitsgerichtete Digitaleingangskanäle mit den folgenden Funktionen:

- Phasenverschobene (eindeutige) Testimpulse T0 … T7 können für den Anschluss mechanischer Sensoren verwendet werden. Die Testimpuls-Ausgänge T0 … T7 liefern ein 24-V-Signal mit einem kurzen phasenverschobenen eindeutigen Impuls (0 V) von 1 ms. Da die Testimpulse der Testimpuls-Ausgangskanäle eindeutig sind (aufgrund der Phasenverschiebung), können sie verwendet werden zur Überwachung des Kanalübersprechens zwischen einem gegebenen Eingangskanal mit verbundenem Testimpuls-Ausgang und einer anderen Leitung, z. B. 24 V DC, eines anderen Testimpuls-Ausgangs usw. Testimpuls-Ausgänge sind dediziert:
  - T0 kann nur mit Eingangskanälen I0 und I1 verwendet werden
  - T1 kann nur mit Eingangskanälen I2 und I3 verwendet werden
  - T2 kann nur mit Eingangskanälen I4 und I5 verwendet werden
  - T3 kann nur mit Eingangskanälen I6 und I7 verwendet werden
  - T4 kann nur mit Eingangskanälen I8 und I9 verwendet werden
  - T5 kann nur mit Eingangskanälen I10 und I11 verwendet werden
  - T6 kann nur mit Eingangskanälen I12 und I13 verwendet werden
  - T7 kann nur mit Eingangskanälen I14 und I15 verwendet werden
- Eingangsverzögerung mit den folgenden Werten: 1 ms, 2 ms, 5 ms, 10 ms, 15 ms, 30 ms, 50 ms, 100 ms, 200 ms, 500 ms. Eine Eingangsverzögerung von 1 ms ist der Mindestwert.

HINWEIS!

Die zulässige Signalfrequenz bei sicherheitsgerichteten Digitaleingängen hängt vom Wert der Eingangsverzögerung für einen Kanal ab:

- Bei einer Kanal-Eingangsverzögerung von 1 … 10 ms muss die Impulslänge des Eingangssignals ≥ 15 ms (~ 65 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 15 ms muss die Impulslänge des Eingangssignals ≥ 20 ms (~ 50 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 30 ms muss die Impulslänge des Eingangssignals ≥ 40 ms (~ 25 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 50 ms muss die Impulslänge des Eingangssignals ≥ 60 ms (~ 15 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 100 ms muss die Impulslänge des Eingangssignals ≥ 120 ms (~ 8 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 200 ms muss die Impulslänge des Eingangssignals ≥ 250 ms (~ 4 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 500 ms muss die Impulslänge des Eingangssignals ≥ 600 ms (~ 1,5 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.

GEFAHR!

Der Parameter Eingangsverzögerung besagt, dass Signale mit einer kürzeren Dauer als die Eingangsverzögerung vom Sicherheitsmodul nicht erkannt werden.

Die Signale mit einer Dauer von mehr als „Eingangsverzögerung“ + „Genauigkeit der Eingangsverzögerung“ werden immer vom Sicherheitsmodul erkannt, vorausgesetzt, dass die zulässige Frequenz (siehe vorangehender Hinweis) des Sicherheitseingangssignals nicht überschritten wird.

Die „Genauigkeit der Eingangsverzögerung“ kann mit den folgenden Annahmen geschätzt werden:

- Wenn für den entsprechenden sicherheitsgerichteten Digitaleingang keine Testimpulse konfiguriert wurden, kann die Genauigkeit der Eingangsverzögerung berechnet werden als 1 % der eingestellten Eingangsverzögerung (die Genauigkeit der Eingangsverzögerung muss jedoch mindestens 0,5 ms sein!).
- Wenn für den entsprechenden sicherheitsgerichteten Digitaleingang des DI581-S-Moduls Testimpulse konfiguriert wurden, können die Werte für die Genauigkeit der Eingangsverzögerung mithilfe des Parameterwertes für die Eingangsverzögerung abgeschätzt werden ↬ *Tab. 4 „Genauigkeit der Eingangsverzögerung für DI581-S“ auf Seite 72*.

Tab. 4: Genauigkeit der Eingangsverzögerung für DI581-S

Eingangsverzögerung (ms)	Genauigkeit der Eingangsverzögerung (ms)
1	2
2	2
5	3

Eingangsverzögerung (ms)	Genauigkeit der Eingangsverzögerung (ms)
10	4
15	5
30	6
50	7
100	10
200	15
500	25

- Überprüfung der Prozess-Spannungsversorgung (eine Diagnosemeldung, die über die fehlende Prozess-Spannungsversorgung für ein entsprechendes Sicherheits-E/A-Modul informiert, wird vom Sicherheits-E/A-Modul an die CPU gesendet). Diese Funktion ist nicht sicherheitsbezogen und steht nicht im Zusammenhang mit der internen sicherheitsrelevanten Über- oder Unterspannungserkennung.
- 2-Kanal äquivalent oder 2-Kanal antivalent mit Diskrepanzzeit-Überwachung (konfigurierbar von 10 ms … 30 s).

HINWEIS!

In einem 2-Kanal-Modus transportiert der niedrigere Kanal (Kanäle 0/8 ➔ Kanal 0, Kanäle 1/9 ➔ Kanal 1 usw.) gesammelt den Prozesswert, das PROFIsafe-Diagnosebit, die Quittierungsanforderung und die Acknowledge-Reintegrationsinformation. Der höhere Kanal liefert immer den passivierten Wert „0“.

GEFAHR!

Nach einem Diskrepanzzeit-Fehler werden die relevanten Kanäle passiviert. Sobald ein gültiger Sensorzustand erkannt wird (äquivalent oder antivalent, je nach ausgewähltem Modus), wird das Statusbit für die Reintegrationsanforderung eines Kanals TRUE. Ein Fehler kann mit der Reintegrationsquittierung des entsprechenden Kanals quittiert werden. Dies kann direkt zu einem Maschinenstart führen, da sowohl TRUE – TRUE als auch FALSE – FALSE gültige Zustände für Äquivalenz und TRUE – FALSE und FALSE – TRUE gültige Zustände für Antivalenz sind.

Stellen Sie sicher, dass dies in Ihrer Sicherheitsanwendung in Ordnung ist. Trifft dies nicht zu, können Sie entweder die mitgelieferten PLCopen Safety-POEs für 2-Kanal-Evaluierung in Ihrem Sicherheitsprogramm verwenden oder Ihre eigenen POEs für 2-Kanal-Evaluierung in der Sicherheits-CPU schreiben.

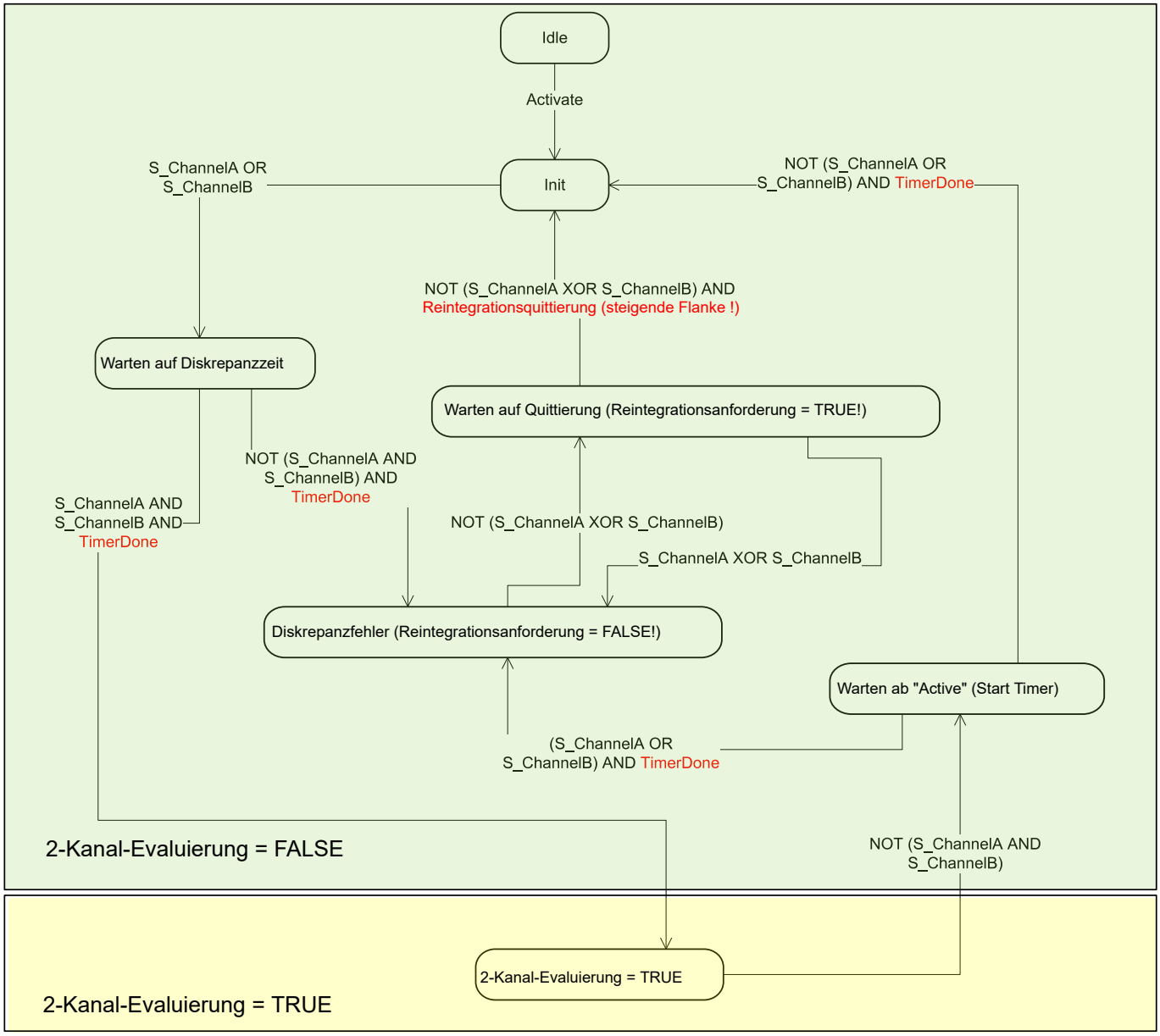


Abb. 18: Modus 2-Kanal äquivalent in DI581-S implementiert

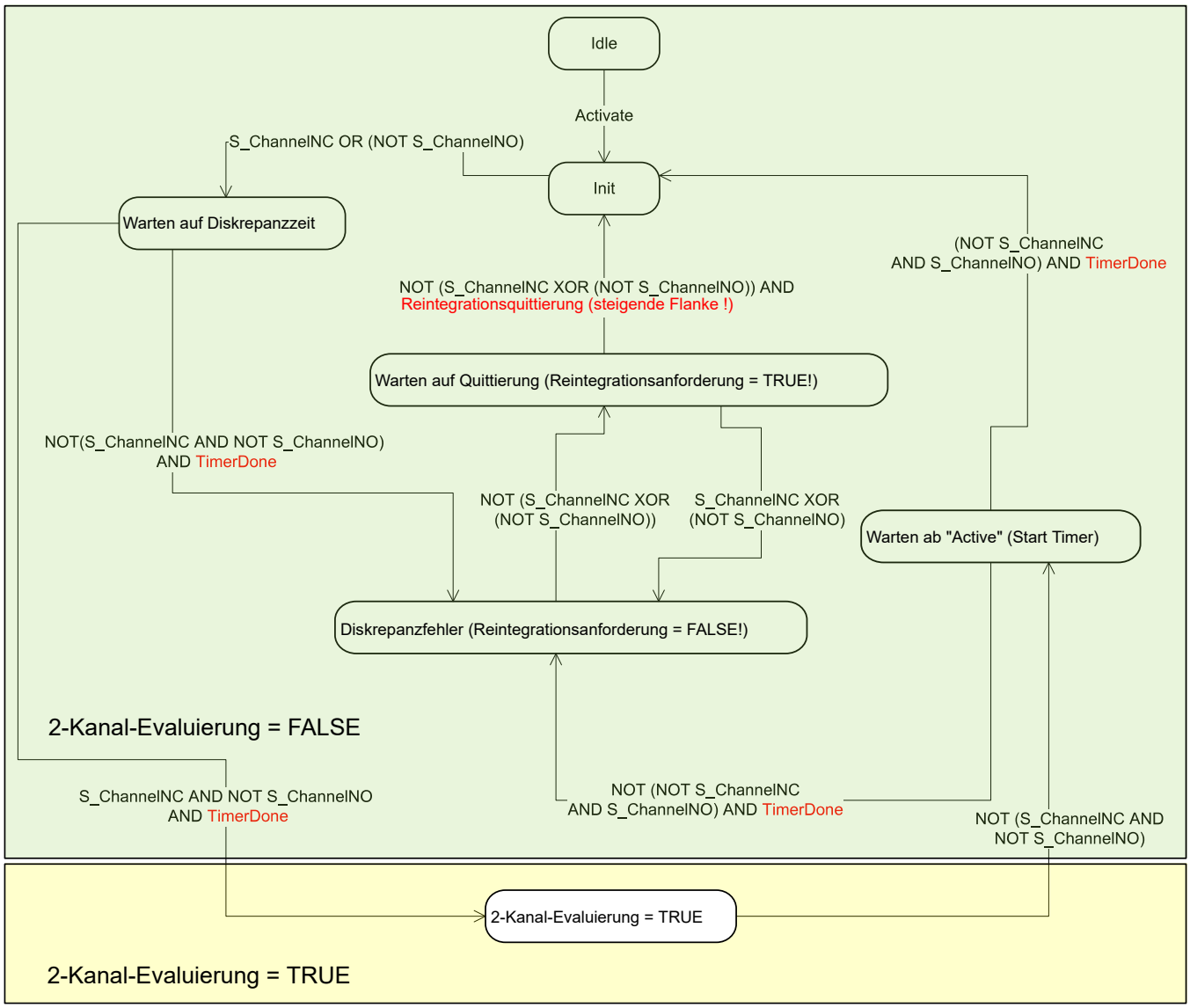


Abb. 19: Modus 2-Kanal antivalent in DI581-S implementiert

HINWEIS!

Die Modi „2-Kanal äquivalent“ und „2-Kanal antivalent“ werden in DI581-S und DX581-S implementiert, um relativ statische Sicherheitssignale, z. B. für Not-Halt, zu verarbeiten.

Wenn sich häufig ändernde Signale, z. B. für Lichtvorhänge, Laserscanner, Türschalter usw., von DI581-S und DX581-S verarbeitet werden müssen, wird dringend empfohlen, eine Eingangsverzögerung von 1 ms für diese Kanäle zu verwenden oder die entsprechenden Kanäle im 1-Kanal-Modus zu konfigurieren und die Evaluierung für 2-Kanal äquivalent und 2-Kanal antivalent in der Sicherheits-CPU mit den PLCopen Safety-Funktionsbausteinen SF_Equivalent ↳ *Kapitel 4.6.4.2 „SF_Equivalent“ auf Seite 222* und SF_Antivalent ↳ *Kapitel 4.6.4.3 „SF_Antivalent“ auf Seite 227* vorzunehmen.

3.3.3 Montage, Abmessungen und elektrischer Anschluss

Die Eingabemodule können nur in den E/A-Klemmenblock mit Federzugklemmen TU582-S eingesteckt werden. Die eindeutige mechanische Codierung auf den E/A-Klemmenblöcken verhindert eventuelle Fehler, sodass keine Standard-E/A-Module in den Sicherheits-E/A-Klemmenblock eingesteckt werden können und umgekehrt. Hier werden grundlegende Informationen zur Montage des Systems angezeigt. Ausführliche Informationen finden Sie unter ↬ *[3]*.

Installation und Wartung dürfen nur von Elektro-Fachkräften nach den technischen Regeln, Richtlinien und einschlägigen Normen, z. B. EN 60204 Teil 1, vorgenommen werden.

Montage von DI581-S

GEFAHR!

Einbau und Austausch im laufenden Betrieb sind bei Modulen unter Spannung nicht zulässig. Für jegliche Arbeiten an Sicherheitsmodulen müssen immer alle Spannungsquellen (Versorgungs- und Prozessspannungen) ausgeschaltet sein.

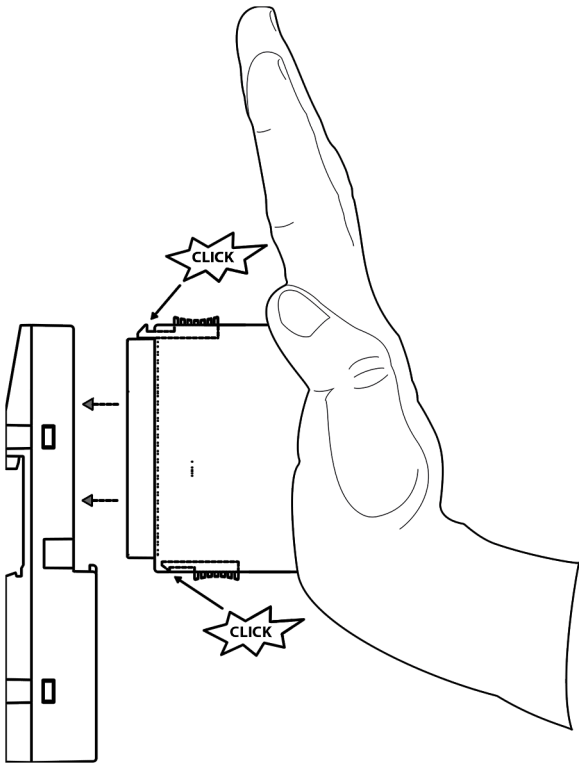


Abb. 20: Montageanleitung

1. Positionieren Sie das Modul auf dem Klemmenblock.

   ⇨ Das Modul rastet ein.

2. Drücken Sie das Modul dann mit einer Kraft von mindestens 100 N in den Klemmenblock, um einen zuverlässigen elektrischen Kontakt herzustellen.

Demontage von DI581-S

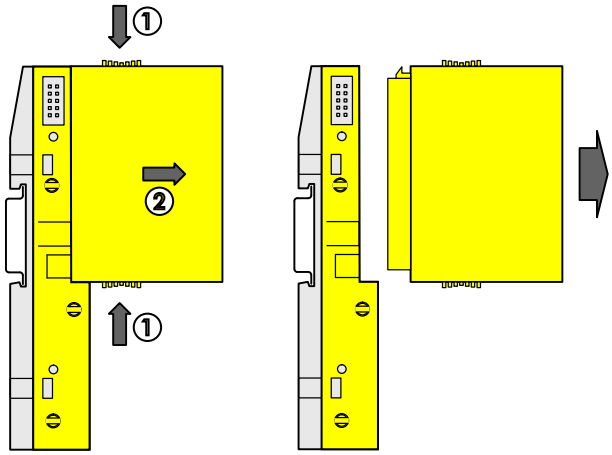


Abb. 21: Demontageanleitung

▷ Drücken Sie oben und unten, dann entfernen Sie das Modul.

Abmessungen

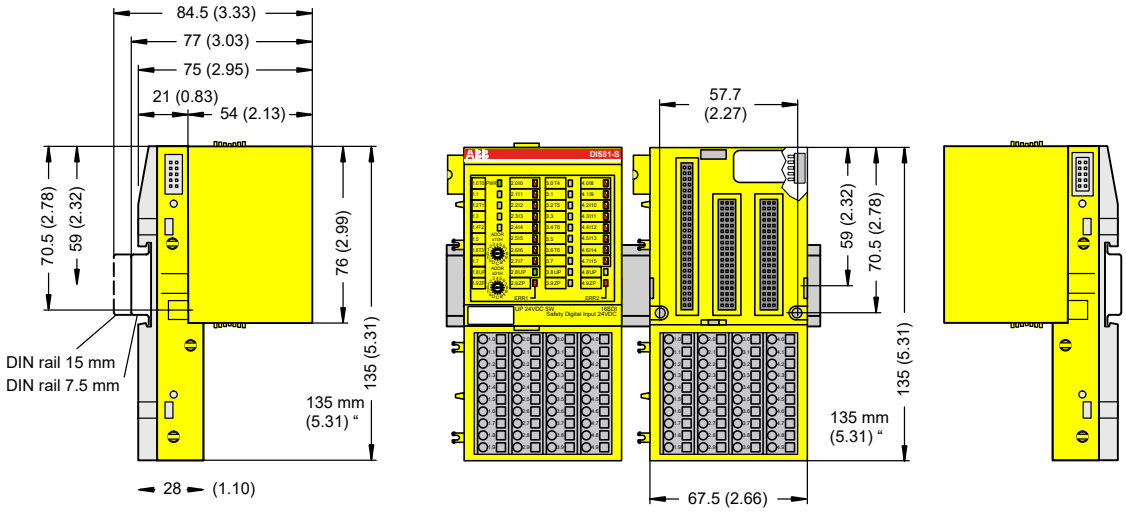


Abb. 22: Abmessungen des Sicherheits-E/A-Moduls DI581-S

Elektrischer Anschluss

HINWEIS!

Derselbe TU582-S wird für alle Sicherheits-E/A-Module der Serie AC500-S verwendet. Wenn der TU582-S für ein DX581-S mit sicherheitsgerichteten Digitalausgängen verdrahtet wird und ein DI581-S oder AI581-S versehentlich in diesen Klemmenblock gesteckt wird, ist es nicht möglich, dass die sicherheitsgerichteten Digitalausgangsklemmen am TU582-S durch falsch eingesteckte Sicherheits-E/A-Module DI581-S oder AI581-S unter Spannung gesetzt werden.

Der elektrische Anschluss der Ein- und Ausgangskanäle erfolgt an den 40 Klemmen des E/A-Klemmenblocks. Auf diese Weise können die Module ausgetauscht werden, ohne dass die Verkabelung an den Klemmenblöcken gelöst werden muss.

Die Klemmen 1.8, 2.8, 3.8 und 4.8 bzw. 1.9, 2.9, 3.9 und 4.9 sind im Inneren des E/A-Klemmenblocks jeweils elektrisch miteinander verbunden und haben unabhängig vom eingesetzten Modul immer dieselbe Belegung:

- Klemmen 1.8, 2.8, 3.8 und 4.8: Prozessspannung UP = +24 V DC
- Klemmen 1.9, 2.9, 3.9 und 4.9: Prozessspannung ZP = 0 V

Belegung der weiteren Klemmen:

Klemmen	Signal	Bedeutung
1.0, 1.2, 1.4, 1.6, 3.0, 3.2, 3.4, 3.6	T0, T1, T2, T3, T4, T5, T6, T7	Anschlüsse der 8 Testimpuls-Ausgänge T0, T1, T2, T3, T4, T5, T6, T7
2.0 … 2.7, 4.0 … 4.7	I0, I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13, I14, I15	16 sicherheitsgerichtete Digitaleingänge
1.8, 2.8, 3.8, 4.8	UP	Prozessversorgung +24 V DC
1.9, 2.9, 3.9, 4.9	ZP	Zentraler Erdanschluss der Prozessversorgungsspannung
1.1, 1.3, 1.5, 1.7, 3.1, 3.3, 3.5, 3.7	Frei	Nicht belegt

HINWEIS!
Die Prozessspannung muss in das Erdungskonzept des Steuerungssystems einbezogen werden (z. B. Erdung des Minuspols).

Anschlussbeispiele

Beispiele der elektrischen Anschlüsse des DI581-S und des Einzelkanals Ix.

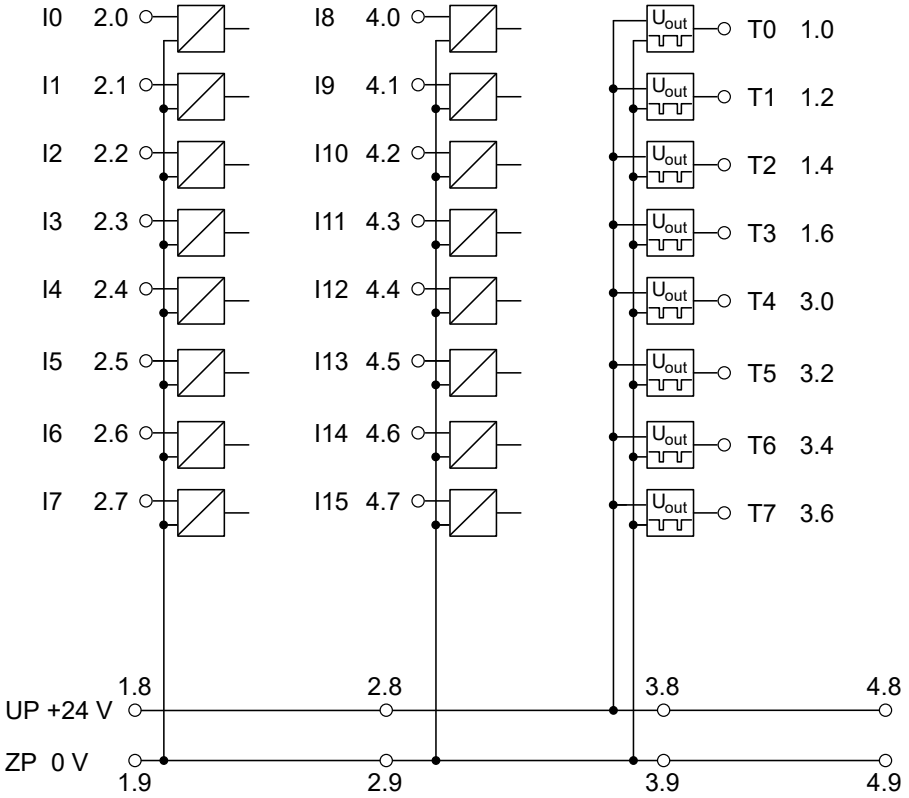


Abb. 23: Beispiel für elektrische Anschlüsse des DI581-S

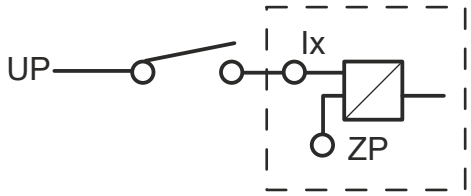


Abb. 24: Beispiel für Einzelkanal des DI581-S

3.3.4 Interner Datenaustausch

Eingänge (Byte)	6
Ausgänge (Byte)	2

3.3.5 Konfiguration der Ein- und Ausgänge

Im digitalen Sicherheits-Eingabemodul DI581-S selbst werden keine Konfigurationsdaten gespeichert. Die Konfigurationsdaten werden in den Sicherheits- und Standard-CPUs gespeichert.

3.3.6 Parametrierung

Die Einrichtung der Parameterdaten wird mit der System-Konfigurationssoftware Automation Builder durchgeführt. Die GSDML-Datei von ABB für PROFINET-Geräte kann zum Konfigurieren der Parameter für DI581-S mit PROFINET F-Hosts von Drittanbietern verwendet werden.

Die Parametereinstellung hat unmittelbaren Einfluss auf die Funktionalität der Module und die für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreichbaren Werte.

Nr.	Name	Werte	Standard
1	Überwachung Spannung	„Ein“, „Aus“	„Ein“
2	Konfiguration	„Nicht belegt“, „1 Kanal“, „2-Kanal äquivalent“, „2-Kanal antivalent“	„Nicht belegt“
3	Testimpuls	„Nicht verfügbar“, „Verfügbar“	„Nicht verfügbar“
4	Eingangsverzögerung	„1 ms“, „2 ms“, „5 ms“, „10 ms“, „15 ms“, „30 ms“, „50 ms“, „100 ms“, „200 ms“, „500 ms“	„5 ms“
5	Diskrepanzzeit*	„10 ms“, „20 ms“, „30 ms“, „40 ms“, „50 ms“, „60 ms“, „70 ms“, „80 ms“, „90 ms“, „100 ms“, „150 ms“, „200 ms“, „250 ms“, „300 ms“, „400 ms“, „500 ms“, „750 ms“, „1 s“, „2 s“, „3 s“, „4 s“, „5 s“, „10 s“, „20 s“, „30 s“	„50 ms“

* Nur für Konfigurationen „2-Kanal äquivalent“ und „2-Kanal antivalent“ verfügbar

3.3.7 Anschlussbeispiele DI581-S

Beispiele elektrischer Anschlüsse und der Werte, die für das Modul DI581-S für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreichbar sind, finden Sie weiter unten.

HINWEIS!

Wenn DC = hoch in den Beschaltungsbeispielen mit sicherheitsgerichteten Digitaleingängen verwendet wird, wird die folgende Maßnahme aus ISO 13849-1 ↳ *[9]* für das DI581-S-Modul verwendet: Querschlussüberwachung von Eingangssignalen und Zwischenergebnissen innerhalb der Logik (L) sowie temporale und logische Software-Überwachung des Programmflusses und Erkennung von statischen Fehlern und Kurzschlüssen (bei mehreren E/As).

Wenn DC = mittel in den Beschaltungsbeispielen mit sicherheitsgerichteten Digitaleingängen verwendet wird, kann eine der Maßnahmen für die Eingabegeräte aus ISO 13849-1 ↳ *[9]* mit DC ≥ 90 % verwendet werden.

1-Kanal-Sensor

Max. SIL / PL [1), 2)]	Max. SIL 1/PL c
SIL[3)]	SIL 2

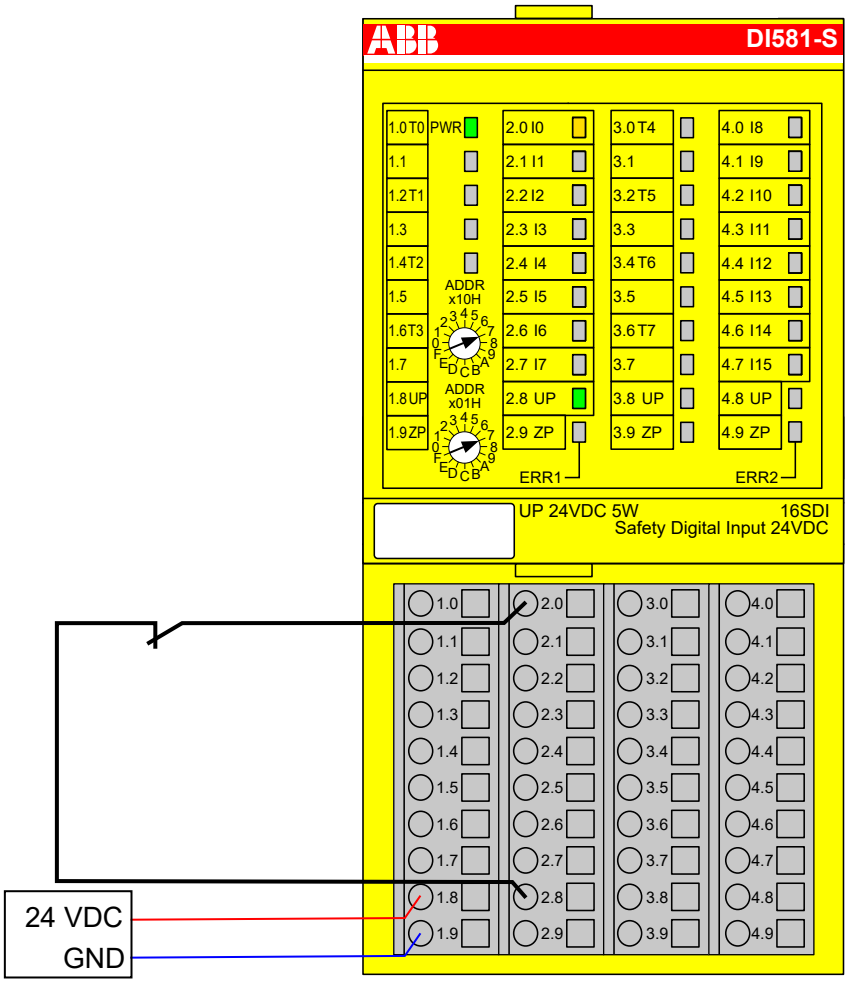


Abb. 25: Beschaltungsbeispiel DI581-S, 1-Kanal-Sensor

1) - MTTFd = hoch, DC = 0

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu SIL 3 erreicht werden)

1-Kanal OSSD-Ausgang (mit internen Tests)

Max. SIL / PL [1), 2)]	Max. SIL 1/PL c
SIL[3)]	SIL 2

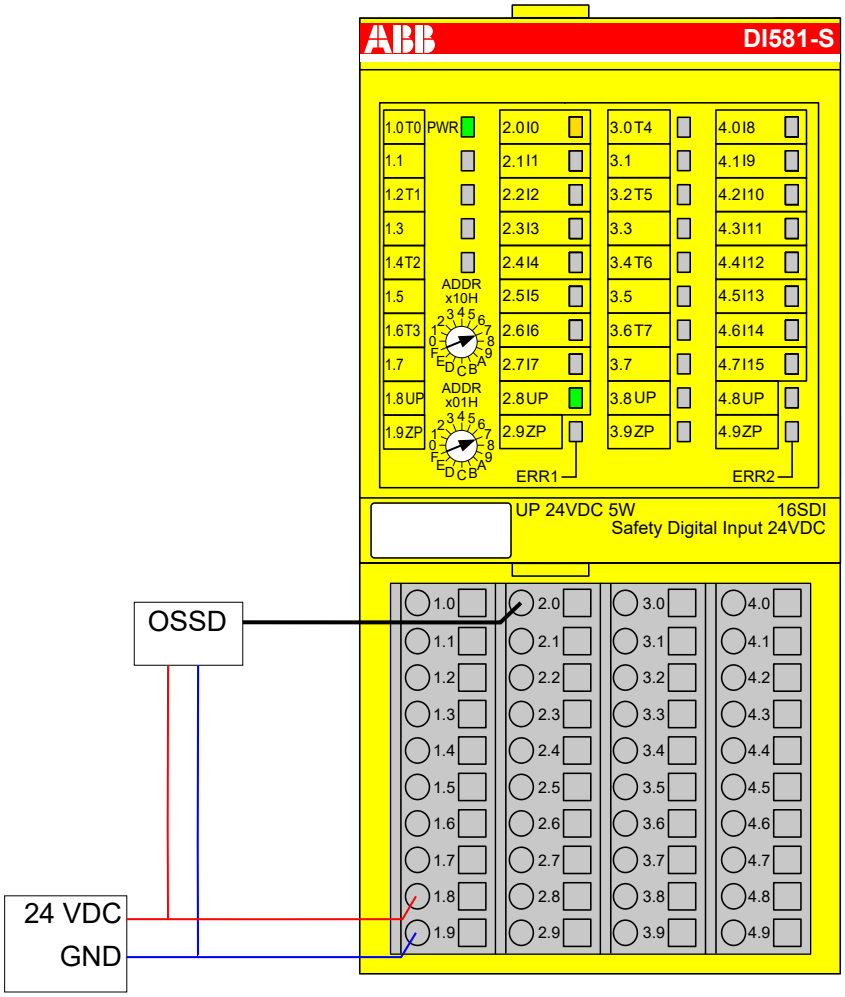


Abb. 26: Beschaltungsbeispiel DI581-S, 1-Kanal OSSD-Ausgang (mit internen Tests)

1) - MTTFd = hoch, DC = 0

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu SIL 3 erreicht werden)

2-Kanal-Sensor (äquivalent)

2-Kanal-Auswertung	Im DI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 3

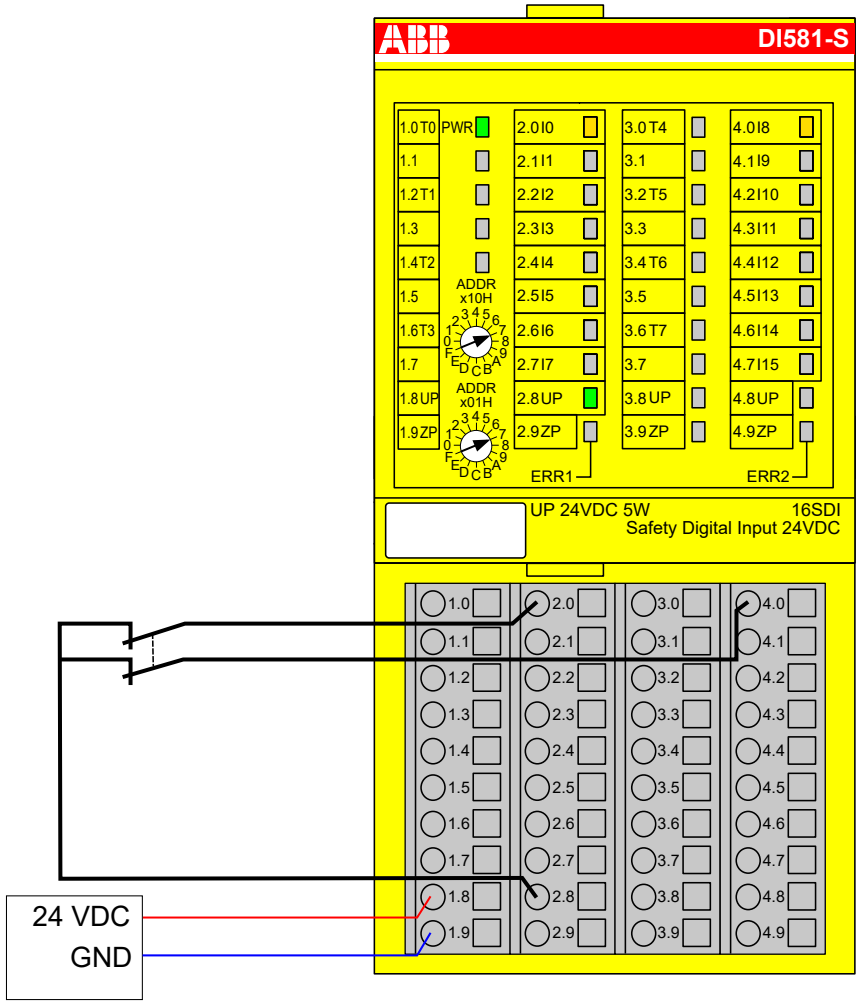


Abb. 27: Beschaltungsbeispiel DI581-S, 2-Kanal-Sensor (äquivalent)

1) - MTTFd = hoch, DC = mittel

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2-Kanal-Sensor (antivalent)

2-Kanal-Auswertung	Im DI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 3

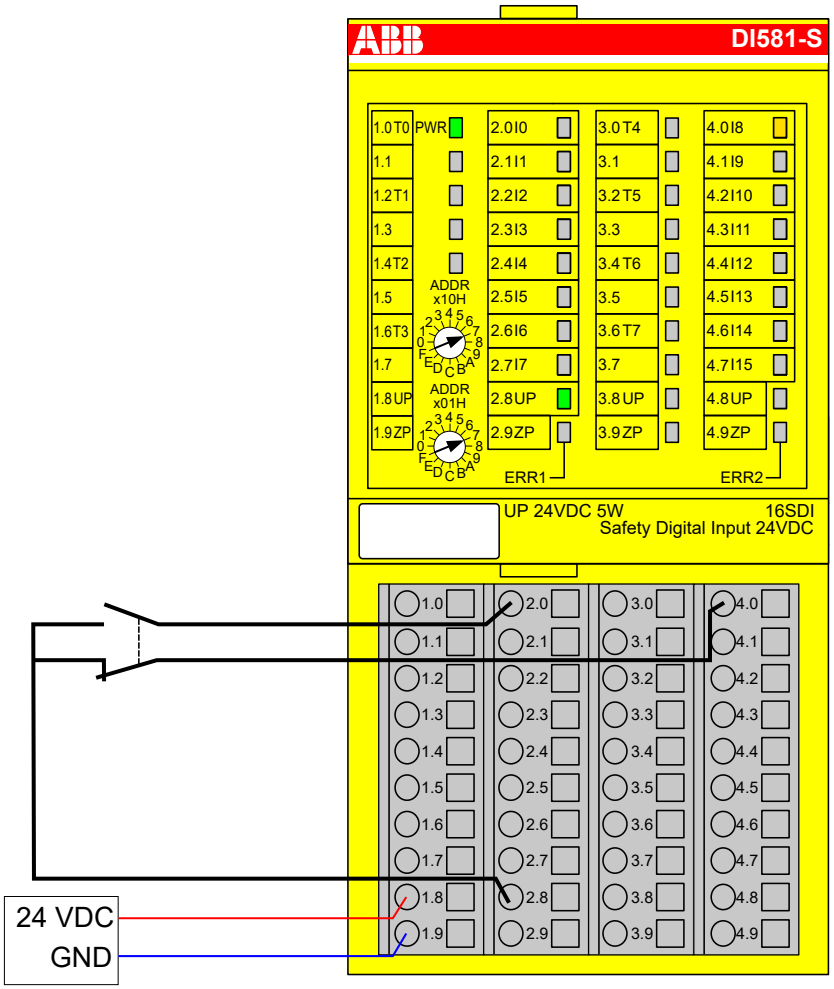


Abb. 28: Beschaltungsbeispiel DI581-S, 2-Kanal-Sensor (antivalent)

1) - MTTFd = hoch, DC = mittel

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2-Kanal OSSD-Ausgang (mit internen Tests)

2-Kanal-Auswertung	Im DI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 3 / PL e
SIL[3)]	SIL 3

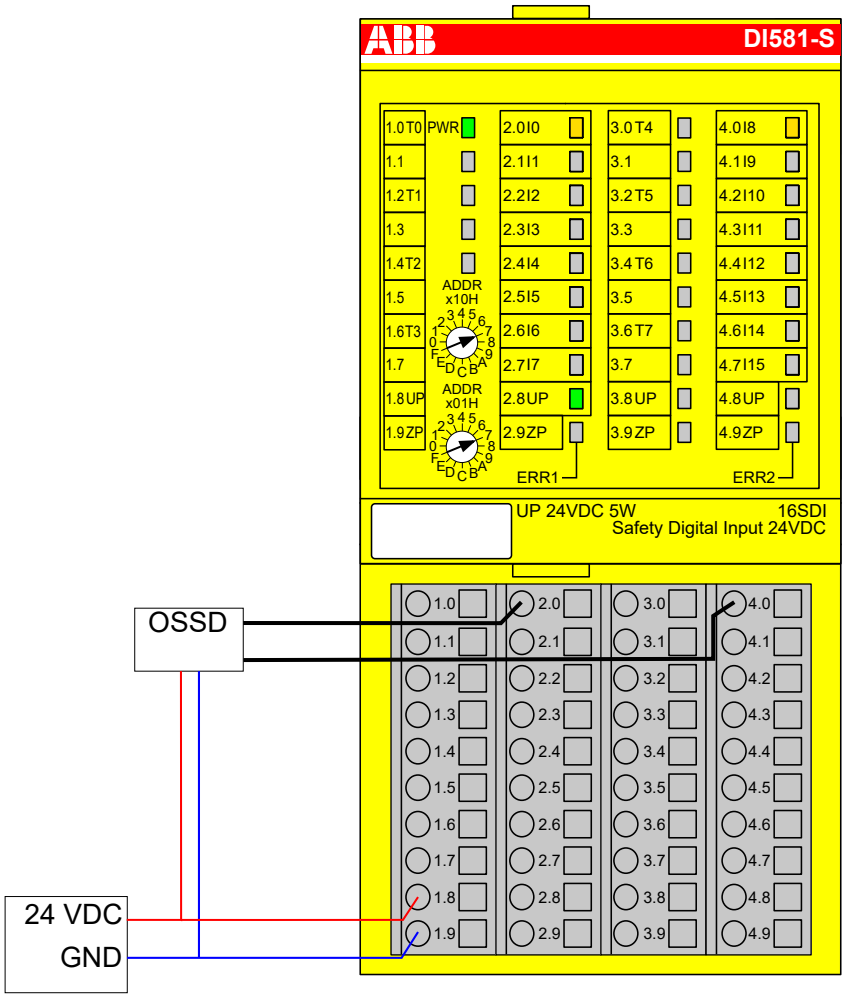


Abb. 29: Beschaltungsbeispiel DI581-S, 2-Kanal OSSD-Ausgang (mit internen Tests)

1) - MTTFd = hoch, DC = hoch

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

1-Kanal-Sensor mit Testim-pulsen

Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 3

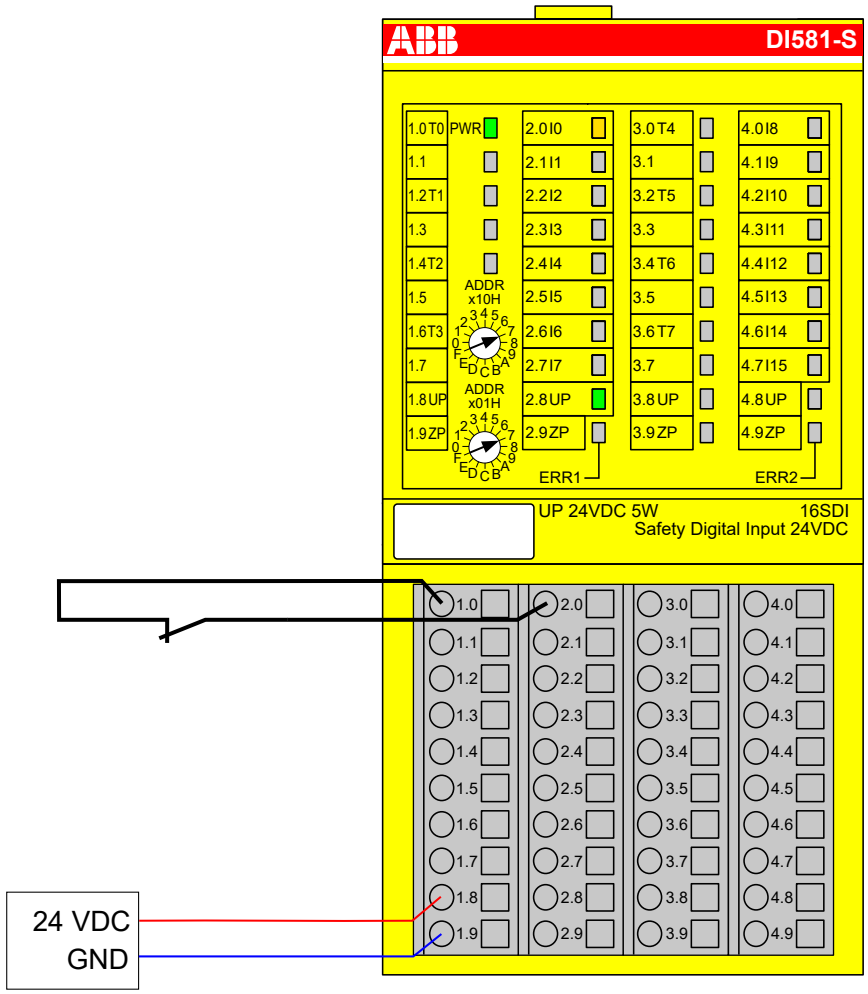


Abb. 30: Beschaltungsbeispiel DI581-S, 1-Kanal-Sensor mit Testimpulsen

[1)] - MTTFd = hoch, DC = mittel

[2)] - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

[3)] - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2-Kanal-Sensor (äquivalent) mit Testimpulsen

2-Kanal-Auswertung	In der Sicherheits-CPU
Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 3

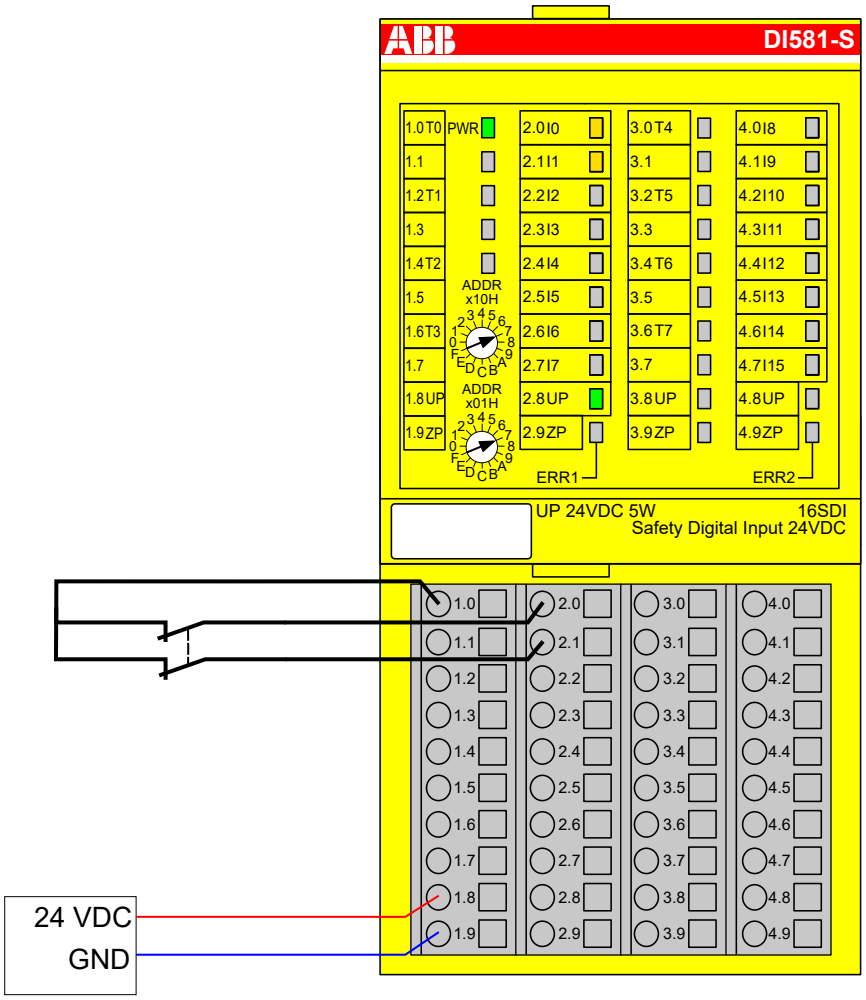


Abb. 31: Beschaltungsbeispiel DI581-S, 2-Kanal-Sensor (äquivalent) mit Testimpulsen

1) - MTTFd = hoch, DC = mittel

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2-Kanal-Sensor (äquivalent) mit Testimpulsen

2-Kanal-Auswertung	Im DI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 3 / PL e
SIL[3)]	SIL 3

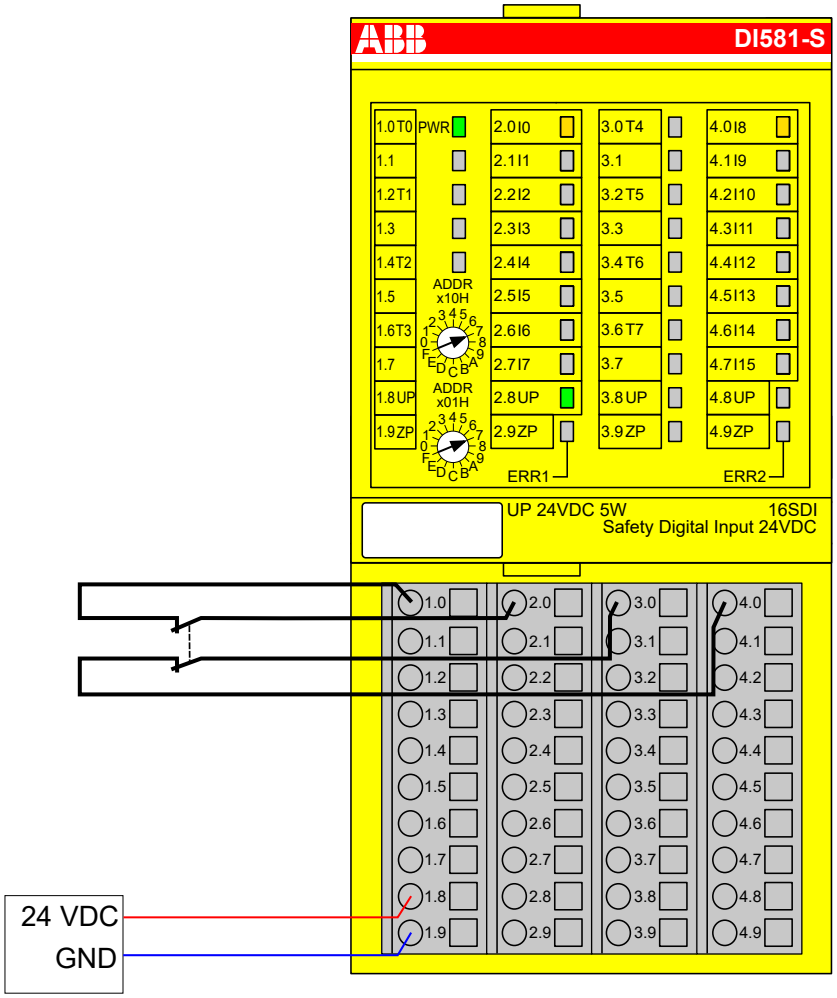


Abb. 32: Beschaltungsbeispiel DI581-S, 2-Kanal-Sensor (äquivalent) mit Testimpulsen

1) - MTTFd = hoch, DC = hoch

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2 × OSSD-Ausgang (mit internen Tests)

2-Kanal-Auswertung	Im DI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 3 / PL e
SIL[3)]	SIL 3

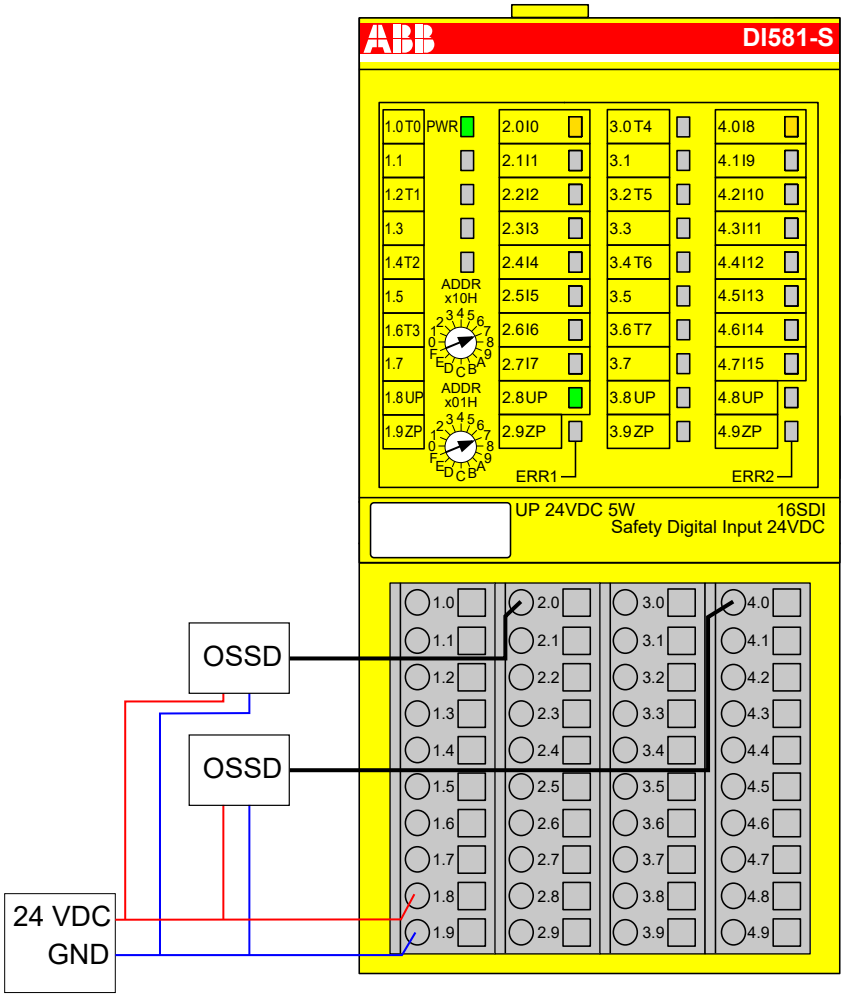


Abb. 33: Beschaltungsbeispiel DI581-S, 2 x OSSD-Ausgang (mit internen Tests)

1) - MTTFd = hoch, DC = hoch

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2 separate Sensoren mit Testimpulsen

2-Kanal-Auswertung	In der Sicherheits-CPU
Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 3

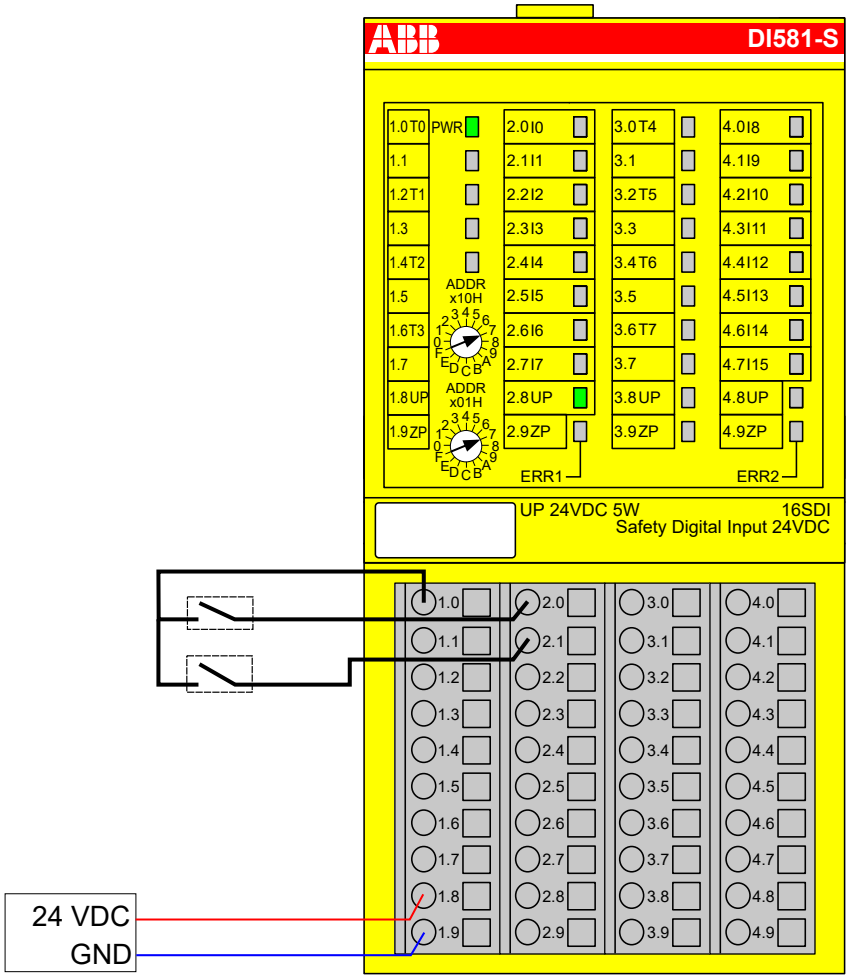


Abb. 34: Beschaltungsbeispiel DI581-S, 2 separate Sensoren mit Testimpulsen

1) - MTTFd = hoch, DC = mittel

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2 × 2-Kanal-Sensor (antivalent) mit Testimpulsen

2-Kanal-Auswertung	Zuerst im Modul DI581-S und dann in der Sicherheits-CPU
Max. SIL / PL [1), 2)]	Max. SIL 3 / PL e
SIL[3)]	SIL 3

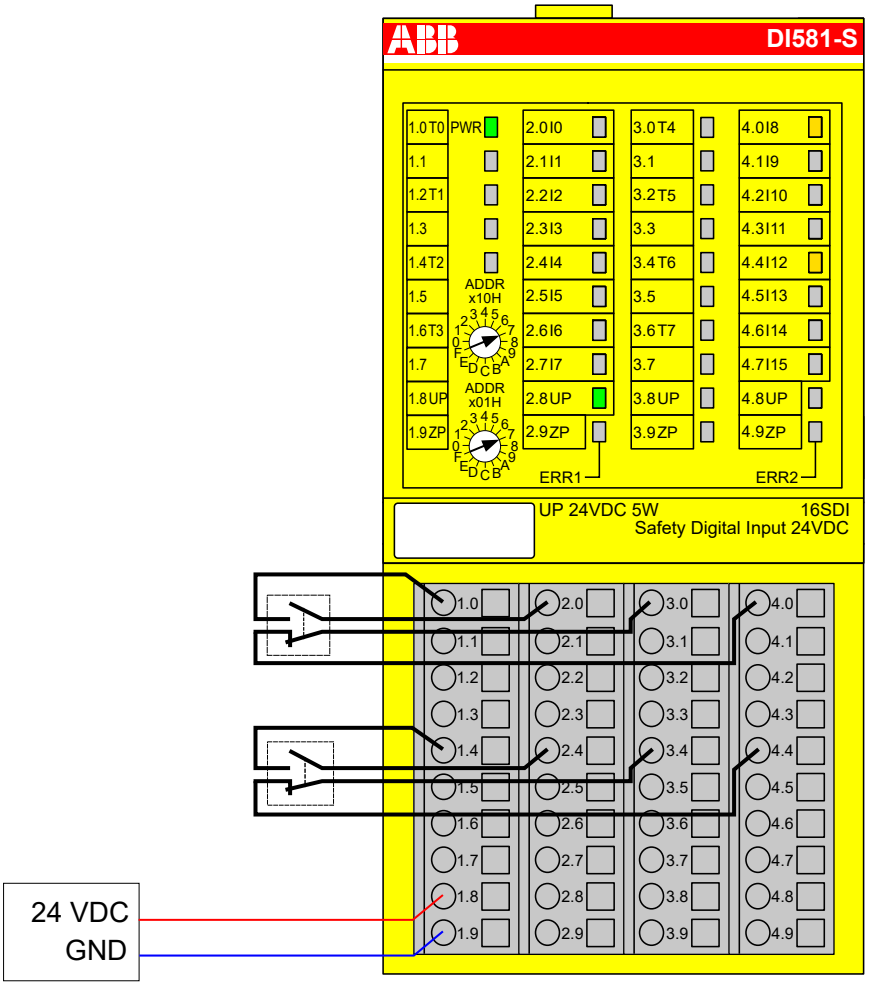


Abb. 35: Beschaltungsbeispiel DI581-S, 2 x 2-Kanal-Sensor (antivalent) mit Testimpulsen

1) - MTTFd = hoch, DC = hoch

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Modusschalter 1 von 4

Modusschalter-Auswertung	In der Sicherheits-CPU
Max. SIL / PL [1), 2)]	Max. SIL 1/PL c
SIL[3)]	SIL 2

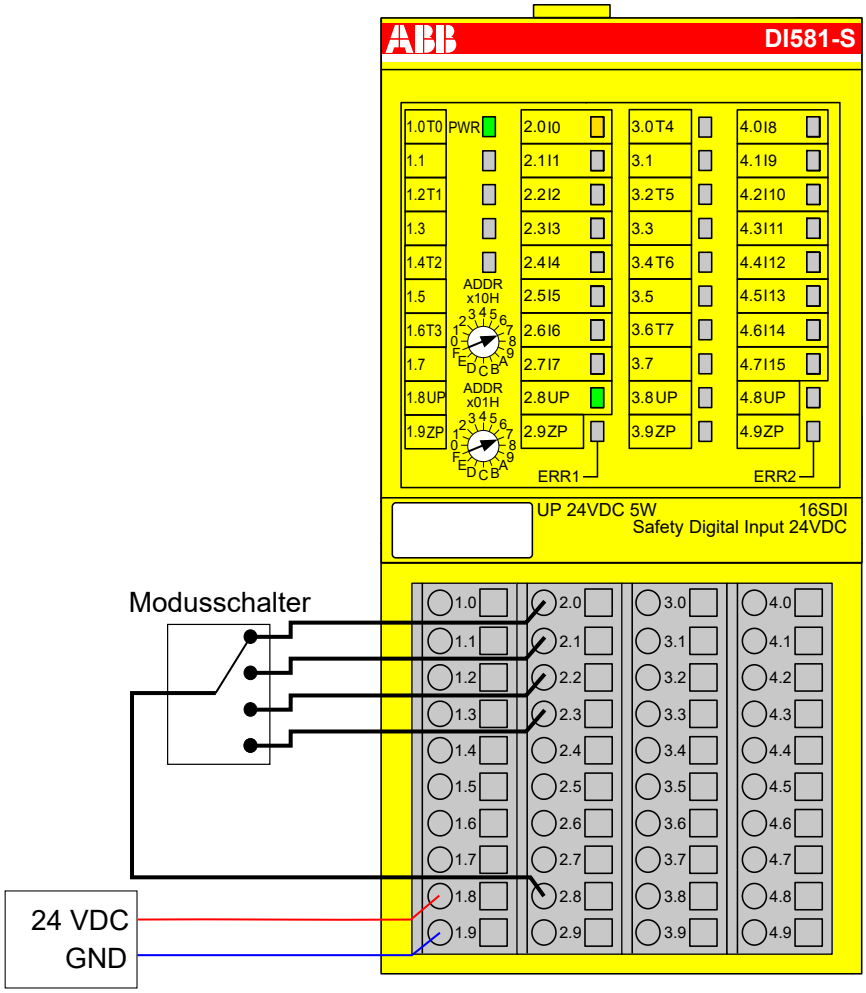


Abb. 36: Beschaltungsbeispiel DI581-S, Modusschalter 1 von 4

[1)] - MTTFd = hoch, DC = gering

[2)] - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

[3)] - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu SIL 3 erreicht werden)

3.3.8 LED-Statusanzeige

Tab. 5: Statusanzeige und deren Bedeutung

LED	Beschreibung	Farbe	LED = AUS	LED = EIN	LED blinkt
Eingänge 0 … 15	Digitaleingang	Gelb	Eingang = AUS	Eingang = EIN (die Eingangsspannung wird auch angezeigt, wenn die Versorgungsspannung AUS ist)	–
	Kanalfehler	Rot	Kein Kanalfehler	Kanalfehler	–

LED	Beschreibung	Farbe	LED = AUS	LED = EIN	LED blinkt
UP	Prozessspannung +24 V DC an Klemme	Grün	Prozess-Versorgungs-spannung fehlt	Prozess-Versorgungs-spannung ist OK	–
PWR	+3,3 V Spannung von I/O-Bus	Grün	+3,3 V Spannung von I/O-Bus ist nicht verfügbar	+3,3 V Spannung von I/O-Bus verfügbar	–
ERR1	Modulfehler-Anzeige 1	Rot	Kein Modulfehler	Modulfehler, der zu einem SAFE STOP führt	Modulpassivierung und/oder Quittieranforderung (abwechselndes Blinken)
ERR2	Modulfehler-Anzeige 2	Rot			

3.3.9 Technische Daten

HINWEIS!
Die Version DI581-S-XC ist für eine Verwendung unter extremen Umgebungsbedingungen erhältlich (*Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*).

Weitere technische Daten stehen im SPS-Katalog von ABB zur Verfügung: *www.abb.com/plc*.

Prozess-Versorgungsspannung UP

Angabe	Wert	Einheit
Anschlussklemmen 1.8 … 4.8 (UP)	+24	V
Anschlussklemmen 1.9 … 4.9 (ZP)	0	V
Nennwert (-15 %, +20 %, ohne Restwelligkeit)	24	V DC
Max. Restwelligkeit	5	%
Verpolschutz	Ja	
Nennwert für Absicherung für UP (schnell)	10	A
Galvanische Trennung	pro Modul	
Verarbeitungsmechanismen von Ein-/Ausgängen	Regelmäßige Aktualisierung	
Stromaufnahme über UP im Normalbetrieb mit + 24 V DC (für Modulelektronik)	0,18	A
Einschaltstrom von UP bei 30 V (beim Einschalten)	0,1	A^2s
Einschaltstrom von UP bei 24 V (beim Einschalten)	0,06	A^2s

HINWEIS!
Alle Kanäle des DI581-S (einschließlich Testimpuls-Ausgänge) sind gegen Verpolung, Rückspeisung, Kurzschluss und andauernde Überspannung bis 30 V DC geschützt.

Einbaulage Horizontal oder vertikal mit Leistungsreduzierung (maximale Betriebstemperatur auf +40 °C reduziert)

Kühlung

Die natürliche Konvektionskühlung darf nicht durch Kabelkanäle oder andere Einbauten im Schaltschrank behindert werden.

Erlaubte Unterbrechungen der Spannungsversorgung laut EN 61131-2

Angabe	Wert	Einheit
Unterbrechungen der Gleichstromversorgung	< 10	ms
Zeit zwischen 2 Unterbrechungen der Gleichstromversorgung, PS2	> 1	s

Umgebungsbedingungen

Angabe	Wert	Einheit
Betriebstemperatur*	0 … +60	°C
Lagerungstemperatur	-40 … +85	°C
Transporttemperatur	-40 … +85	°C
Luftfeuchtigkeit ohne Kondensation	max. 95	%
Betriebsluftdruck	> 800	hPa
Lagerluftdruck	> 660	hPa
Betriebshöhe	< 2000	m über NN
Lagerhöhe	< 3500	m über NN

* Erweiterte Temperaturbereiche (unter 0 °C und über +60 °C) werden von Sonderversionen von DI581-S unterstützt ↳ *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Kriech- und Luftstrecken

Die Kriech- und Luftstrecken entsprechen der Überspannungskategorie II, Verschmutzungsgrad 2.

Netzteile

Zur Versorgung der Module müssen Netzteile gemäß PELV-/SELV-Spezifikationen verwendet werden.

Elektromagnetische Verträglichkeit

Informationen zur elektromagnetischen Verträglichkeit finden Sie im neuesten TÜV SÜD Report ↳ *[1]*.

Mechanische Eigenschaften

Angabe	Wert	Einheit
Schutzart	IP 20	
Gehäuse	gemäß UL94	
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 3,5 mm	2 … 15	Hz
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 1 g *	15 … 150	Hz
Stoßprüfung (alle drei Achsen), 11 ms Halbsinus	15	g
MTBF	102	Jahre

* Höhere Werte auf Anfrage

Selbsttest und Diagnosefunktionen

Tests während Start und Betrieb: Programmablauf-Überwachung, RAM, CPU, Kanalübersprechen, dauerhaftes 1-Signal usw.

Abmessungen, Gewicht

Angabe	Wert	Einheit
B × H × T	67,5 × 76 × 62	mm
Gewicht	~ 130	g

Zertifizierungen CE, cUL (weitere Zertifizierungen unter *www.abb.com/plc*)

3.3.9.1 Technische Daten der sicherheitsgerichteten Digitaleingänge

Angabe	Wert	Einheit
Anzahl Eingangskanäle je Modul	16	
Klemmen für Kanäle I0 bis I7	2.0 … 2.7	
Klemmen für Kanäle I8 bis I15	4.0 … 4.7	
Anschlüsse mit Bezugspotential für alle Eingänge (Minuspol der Prozess-Versorgungsspannung, Signalname ZP)	1.9 … 4.9	
Galvanische Trennung von den restlichen Teilen des Moduls (I/O-Bus)	Ja	
Eingangstyp gemäß EN 61131-2	Typ 1	
Eingangsverzögerung (0 ➔ 1 oder 1 ➔ 0), konfigurierbar	1 … 500	ms

Anzeige Eingangssignal Eine gelbe LED pro Kanal. Die LED ist EIN bei Eingangssignal „High“ (Signal 1).

Signalspannung

Angabe	Wert	Einheit
Eingangssignalspannung	24	V DC
Signal 0	-3 … +5	V
Undefiniertes Signal	> +5 … < +15	V
Signal 1	+15 … +30	V

Eingangsstrom je Kanal

Angabe	Wert	Einheit
Eingangsspannung +24 V, typisch	7	mA
Eingangsspannung +5 V	> 1	mA
Eingangsspannung +15 V	> 4	mA
Eingangsspannung +30 V	< 8	mA

Kabellänge

Angabe	Wert	Einheit
Max. Kabellänge, geschirmt	1000	m
Max. Kabellänge, ungeschirmt	600	m

3.3.9.2 Technische Daten der nicht sicheren Testimpuls-Ausgänge

GEFAHR!
Das Überschreiten der zulässigen Prozess- oder Versorgungsspannung (< -35 V DC bzw. > +35 V DC) kann zu irreparablen Schäden am System führen.

Angabe	Wert	Einheit
Anzahl Testimpuls-Kanäle pro Modul (Ausgänge für Transistor-Testimpuls)	8	
Klemmen für Kanäle T0 bis T3	1.0, 1.2, 1.4, 1.6	
Klemmen für Kanäle T4 bis T7	3.0, 3.2, 3.4, 3.6	
Anschlüsse mit Bezugspotential für alle Testimpuls-Ausgänge (Minuspol der Prozess-Versorgungsspannung, Signalname ZP)	1.9 … 4.9	
Anschlüsse der gemeinsamen Versorgungsspannung für alle Ausgänge (Pluspol der Prozess-Versorgungsspannung, Signalname UP)	1.8 … 4.8	
Ausgangsspannung für 1-Signal	UP – 0,8	V
Länge der Testimpuls-0-Phase	1	ms

Ausgangsstrom

Angabe	Wert	Einheit
Nennwert, je Kanal	10	mA
Maximalwert (alle Kanäle zusammen)	80	mA
Kurzschluss-/Überlastfestigkeit	Ja	
Ausgangsstrombegrenzung	65	mA
Rückspannungsfestigkeit gegen 24-V-Signalanschluss	Ja	

Kabellänge

Angabe	Wert	Einheit
Max. Kabellänge, geschirmt	1000	m
Max. Kabellänge, ungeschirmt	600	m

3.3.10 Bestelldaten

Typ	Beschreibung	Bestellnummer
DI581-S	Digitales Sicherheits-Eingabemodul 16SDI	1SAP 284 000 R0001
DI581-S-XC	Digitales Sicherheits-Eingabemodul 16SDI, extreme Umgebungsbedingungen	1SAP 484 000 R0001

3.4 Digitales Sicherheits-E/A-Modul DX581-S

Elemente des Moduls

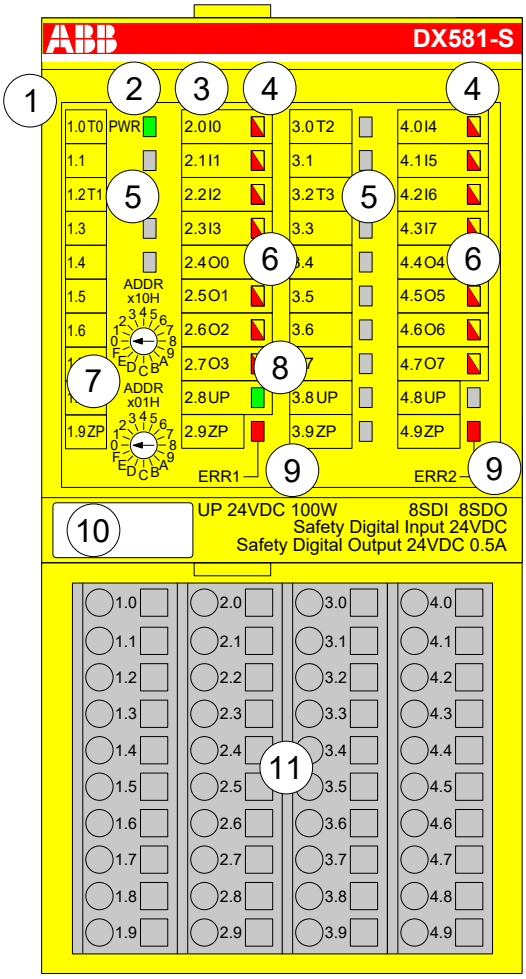


Abb. 37: Digitales Sicherheits-E/A-Modul DX581-S, eingesteckt in Klemmenblock TU582-S

1 I/O-Bus
2 System-LED
3 Zuordnung Klemmennummer – Signalname
4 8 gelb/rote LEDs Signalstatus I0 … I3/I4 … I7
5 4 Testimpuls-Ausgänge T0 … T1/T2 … T3
6 8 gelb/rote LEDs Signalstatus O0 … O3 / O4 … O7
7 2 Drehschalter für PROFIsafe-Adresse
8 Grüne LED für Prozessspannung UP
9 Rote LEDs zur Anzeige von Modulfehlern
10 Beschriftungsschild (TA525)
11 E/A-Klemmenblock (TU582-S)

3.4.1 Verwendungszweck

Das digitale Sicherheits-E/A-Modul DX581-S kann als dezentrales Erweiterungsmodul für die PROFINET-Module CI501-PNIO, CI502-PNIO, CI504-PNIO und CI506-PNIO oder lokal an CPUs der AC500-Serie für Sicherheitsanwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1) verwendet werden.

HINWEIS!
Die Werte, die mit Ihrer Sicherheitsanwendung für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreicht werden können, hängen von der Verdrahtung der Sensoren und Aktoren mit dem DX581-S-Modul ab *Kapitel 3.4.7 „Anschlussbeispiele DX581-S“ auf Seite 108*.

DX581-S enthält 8 sicherheitsgerichtete Digitaleingänge 24 V DC aufgeteilt in zwei Gruppen (2.0 … 2.3 und 4.0 … 4.3) und 8 digitale Sicherheits-Transistorausgänge ohne Potentialtrennung zwischen den Kanälen.

Die Ein-/Ausgänge sind von den anderen Schaltkreisen des Moduls nicht galvanisch getrennt.

3.4.2 Funktionalität

Digitaleingänge	8 (24 V DC)
Digitalausgänge	8 (24 V DC)
LED-Anzeigen	Für Signalzustand, Modulfehler, Kanalfehler und Versorgungsspannung
Interne Spannungsversorgung	über I/O-Bus-Schnittstelle
Externe Spannungsversorgung	Über Klemmen ZP und UP (Prozessspannung 24 V DC)

Selbsttests und Diagnosefunktionen (sowohl beim Starten als auch während des Betriebs), wie CPU- und RAM-Tests, Programmablauf-Überwachung, Kanalübersprechen und dauerhaftes 1-Signal usw., werden in DX581-S gemäß den Anforderungen von IEC 61508 SIL 3 implementiert.

HINWEIS!
Nur F_Dest_Add wird für die PROFIsafe F-Device-Identifizierung im DX581-S verwendet.

Das DX581-S verfügt über 8 sicherheitsgerichtete Digitaleingangskanäle mit den folgenden Funktionen:

- Phasenverschobene (eindeutige) Testimpulse T0 … T3 können für den Anschluss mechanischer Sensoren verwendet werden. Die Testimpuls-Ausgänge T0 … T3 liefern ein 24-V-Signal mit einem kurzen phasenverschobenen eindeutigen Impuls (0 V) von 1 ms. Da die Testimpulse der Testimpuls-Ausgangskanäle eindeutig sind (aufgrund der Phasenverschiebung), können sie verwendet werden zur Überwachung des Kanalübersprechens zwischen einem gegebenen Eingangskanal mit verbundenem Testimpuls-Ausgang und einer anderen Leitung, z. B. 24 V DC, eines anderen Testimpuls-Ausgangs usw. Testimpuls-Ausgänge sind dediziert:
  - T0 kann nur mit Eingangskanälen I0 und I1 verwendet werden
  - T1 kann nur mit Eingangskanälen I2 und I3 verwendet werden
  - T2 kann nur mit Eingangskanälen I4 und I5 verwendet werden
  - T3 kann nur mit Eingangskanälen I6 und I7 verwendet werden
- Eingangsverzögerung mit den folgenden Werten: 1 ms, 2 ms, 5 ms, 10 ms, 15 ms, 30 ms, 50 ms, 100 ms, 200 ms, 500 ms. Eine Eingangsverzögerung von 1 ms ist der Mindestwert.

HINWEIS!

Die zulässige Signalfrequenz bei sicherheitsgerichteten Digitaleingängen hängt vom Wert der Eingangsverzögerung für einen Kanal ab:

- Bei einer Kanal-Eingangsverzögerung von 1 … 10 ms muss die Impulslänge des Eingangssignals ≥ 15 ms (~ 65 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 15 ms muss die Impulslänge des Eingangssignals ≥ 20 ms (~ 50 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 30 ms muss die Impulslänge des Eingangssignals ≥ 40 ms (~ 25 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 50 ms muss die Impulslänge des Eingangssignals ≥ 60 ms (~ 15 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 100 ms muss die Impulslänge des Eingangssignals ≥ 120 ms (~ 8 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 200 ms muss die Impulslänge des Eingangssignals ≥ 250 ms (~ 4 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.
- Bei einer Kanal-Eingangsverzögerung von 500 ms muss die Impulslänge des Eingangssignals ≥ 600 ms (~ 1,5 Hz) sein, um eine gelegentliche Passivierung des Eingangskanals zu vermeiden.

GEFAHR!

Der Parameter Eingangsverzögerung besagt, dass Signale mit einer kürzeren Dauer als die Eingangsverzögerung vom Sicherheitsmodul nicht erkannt werden.

Die Signale mit einer Dauer von mehr als „Eingangsverzögerung“ + „Genauigkeit der Eingangsverzögerung“ werden immer vom Sicherheitsmodul erkannt, vorausgesetzt, dass die zulässige Frequenz (siehe vorangehender Hinweis) des Sicherheitseingangssignals nicht überschritten wird.

Die „Genauigkeit der Eingangsverzögerung“ kann mit den folgenden Annahmen geschätzt werden:

- Wenn für den entsprechenden sicherheitsgerichteten Digitaleingang keine Testimpulse konfiguriert wurden, kann die Genauigkeit der Eingangsverzögerung berechnet werden als 1 % der eingestellten Eingangsverzögerung (die Genauigkeit der Eingangsverzögerung muss jedoch mindestens 0,5 ms sein!).
- Wenn für den entsprechenden sicherheitsgerichteten Digitaleingang des DX581-S-Moduls Testimpulse konfiguriert wurden, können die Werte für die Genauigkeit der Eingangsverzögerung mithilfe des Parameterwertes für die Eingangsverzögerung abgeschätzt werden ↬ *Tab. 6 „Genauigkeit der Eingangsverzögerung für DX581-S“ auf Seite 99*.

Tab. 6: Genauigkeit der Eingangsverzögerung für DX581-S

Eingangsverzögerung (ms)	Genauigkeit der Eingangsverzögerung (ms)
1	2
2	2
5	3

Eingangsverzögerung (ms)	Genauigkeit der Eingangsverzögerung (ms)
10	4
15	5
30	6
50	10
100	15
200	25
500	50

- Überprüfung der Prozess-Spannungsversorgung (eine Diagnosemeldung, die über die fehlende Prozess-Spannungsversorgung für ein entsprechendes Sicherheits-E/A-Modul informiert, wird vom Sicherheits-E/A-Modul an die CPU gesendet). Diese Funktion ist nicht sicherheitsbezogen und steht nicht im Zusammenhang mit der internen sicherheitsrelevanten Über- oder Unterspannungserkennung.
- 2-Kanal äquivalent oder 2-Kanal antivalent mit Diskrepanzzeit-Überwachung (konfigurierbar von 10 ms … 30 s).

HINWEIS!

In einem 2-Kanal-Modus transportiert der niedrigere Kanal (Kanäle 0/4 ➔ Kanal 0, Kanäle 1/5 ➔ Kanal 1 usw.) gesammelt den Prozesswert, das PROFIsafe-Diagnosebit, die Quittierungsanforderung und die Acknowledge-Reintegrationsinformation. Der höhere Kanal liefert immer den passivierten Wert „0“.

GEFAHR!

Nach einem Diskrepanzzeit-Fehler werden die relevanten Kanäle passiviert. Sobald ein gültiger Sensorzustand erkannt wird (äquivalent oder antivalent, je nach ausgewähltem Modus), wird das Statusbit für die Reintegrationsanforderung eines Kanals TRUE. Ein Fehler kann mit der Reintegrationsquittierung des entsprechenden Kanals quittiert werden. Dies kann direkt zu einem Maschinenstart führen, da sowohl TRUE – TRUE als auch FALSE – FALSE gültige Zustände für Äquivalenz und TRUE – FALSE und FALSE – TRUE gültige Zustände für Antivalenz sind.

Stellen Sie sicher, dass dies in Ihrer Sicherheitsanwendung in Ordnung ist. Trifft dies nicht zu, können Sie entweder die mitgelieferten PLCopen Safety-POEs für 2-Kanal-Evaluierung in Ihrem Sicherheitsprogramm verwenden oder Ihre eigenen POEs für 2-Kanal-Evaluierung in der Sicherheits-CPU schreiben.

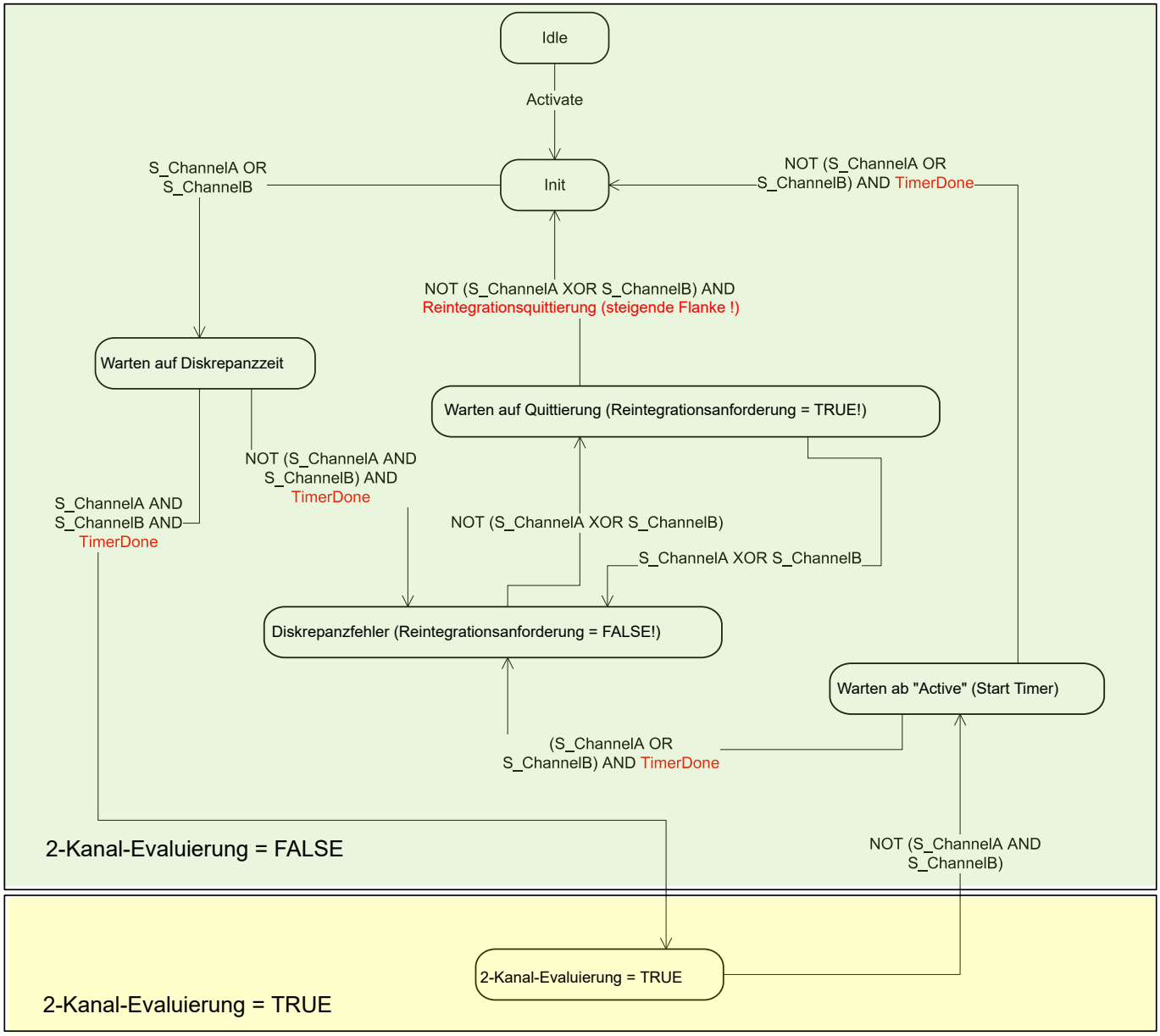


Abb. 38: Modus 2-Kanal äquivalent in DX581-S implementiert

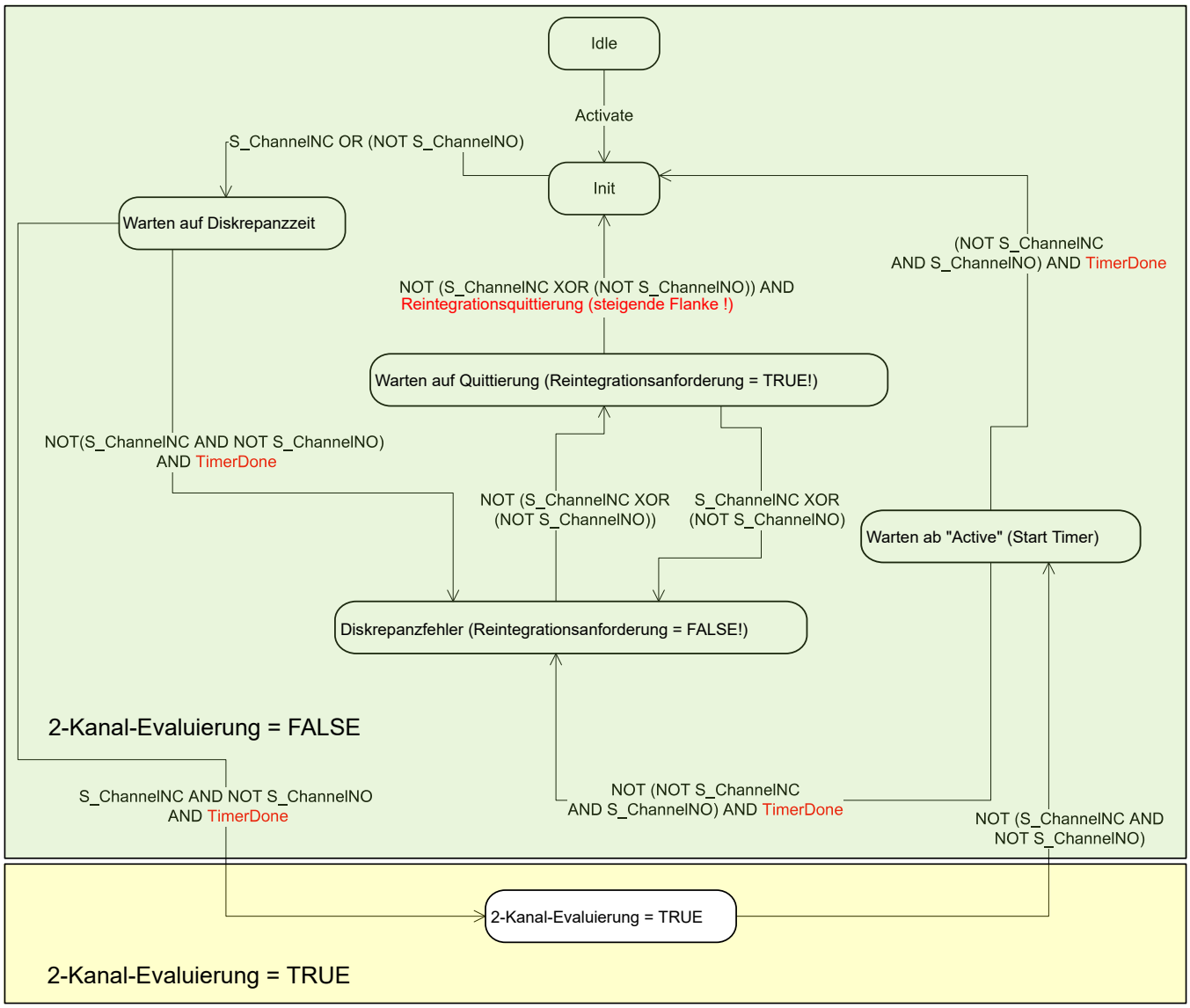


Abb. 39: Modus 2-Kanal antivalent in DX581-S implementiert

HINWEIS!

Die Modi „2-Kanal äquivalent“ und „2-Kanal antivalent“ werden in DI581-S und DX581-S implementiert, um relativ statische Sicherheitssignale, z. B. für Not-Halt, zu verarbeiten.

Wenn sich häufig ändernde Signale, z. B. für Lichtvorhänge, Laserscanner, Türschalter usw., von DI581-S und DX581-S verarbeitet werden müssen, wird dringend empfohlen, eine Eingangsverzögerung von 1 ms für diese Kanäle zu verwenden oder die entsprechenden Kanäle im 1-Kanal-Modus zu konfigurieren und die Evaluierung für 2-Kanal äquivalent und 2-Kanal antivalent in der Sicherheits-CPU mit den PLCopen Safety-Funktionsbausteinen SF_Equivalent ↳ *Kapitel 4.6.4.2 „SF_Equivalent“ auf Seite 222* und SF_Antivalent ↳ *Kapitel 4.6.4.3 „SF_Antivalent“ auf Seite 227* vorzunehmen.

DX581-S verfügt über 8 sicherheitsgerichtete Digitalausgangskanäle mit den folgenden Funktionen:

- Die internen Ausgangskanal-Tests können deaktiviert werden.

GEFAHR!

Parameter *„Erkennung“* der Ausgangskanäle

Wenn für einen der Ausgangskanäle der Parameter *„Erkennung“* = AUS gesetzt wird, erscheint eine Warnung, dass der Ausgangskanal in diesem Fall nicht den Anforderungen gemäß max. SIL 3 (IEC 62061) und PL e (ISO 13849-1) entspricht. Zwei Sicherheits-Ausgangskanäle müssen verwendet werden, um die entsprechenden max. SIL- und PL-Werte zu erreichen.

Der Parameter *„Erkennung“* wurde für Anwender entwickelt, die Sicherheitsausgänge des DX581-S für Sicherheitsfunktionen gemäß max. SIL 1 (oder max. SIL 2 unter speziellen Bedingungen) oder PL c (oder max. PL d unter speziellen Bedingungen) nutzen möchten und weniger interne Impulse des DX581-S auf der Sicherheits-Ausgangsleitung sichtbar haben möchten. Solche internen Impulse könnten als LOW-Signal z. B. von Antriebseingängen erkannt werden, was zu einem ungewollten Maschinenstopp führen würde.

GEFAHR!

Verhalten unabhängig von der Einstellung des Parameters *„Erkennung“*

Kurzschluss gegenüber Erde wird für Ausgangskanäle des Moduls DX581-S überwacht. Kurzschlüsse gegenüber 24 V DC an der Ausgangsleitung werden jedoch nicht überwacht. Die Endanwender müssen entsprechende Maßnahmen durchführen (z. B. auf Anwendungsseite durch die Definition geeigneter Testintervalle für die Sicherheitsfunktion oder durch das Zurücklesen des Status der Ausgangsleitung unter Verwendung eines der verfügbaren sicherheitsgerichteten Digitaleingänge), um die entsprechenden Anforderungen nach IEC 62061 und ISO 13849-1 zu erfüllen, wenn ein Kurzschluss gegenüber 24 V DC nicht ausgeschlossen werden kann.

GEFAHR!

Wenn für den entsprechenden Sicherheits-Ausgangskanal ein Fehler erkannt wird, wird er direkt durch das Modul DX581-S passiviert.

Beachten Sie bitte, dass das Bit zur Reintegrationsanforderung für passivierte Ausgangskanäle automatisch auf HIGH gesetzt wird, sobald der Kanal passiviert ist und der erwartete LOW-Zustand („0“) vom Ausgangskanal erreicht wurde. Solch ein Verhalten tritt bei bestimmten Fehlern auf, weil das Modul DX581-S im LOW-Zustand des Ausgangskanals („0“) nicht überprüfen kann, ob zuvor erkannte Fehler, die zur Passivierung des Kanals führten, noch bestehen oder nicht.

Wenn die Anwender solche Ausgangskanäle mithilfe der relevanten Reintegrationsquittierungs-Bits wieder integrieren möchten, wird dies erfolgreich sein. Sollte der Fehler allerdings noch fortbestehen, werden die entsprechenden Kanäle im nächsten Fehlererkennungszyklus des DX581-S passiviert werden.

Bei internen Fehlern des Ausgabemoduls wird das gesamte Modul passiviert.

3.4.3 Montage, Abmessungen und elektrischer Anschluss

Die Eingabe-/Ausgabemodule können nur in den E/A-Klemmenblock mit Federzugklemmen TU582-S eingesteckt werden. Die eindeutige mechanische Codierung auf den E/A-Klemmenblöcken verhindert eventuelle Fehler, sodass keine Standard-E/A-Module in den Sicherheits-E/A-Klemmenblock eingesteckt werden können und umgekehrt. Hier werden grundlegende Informationen zur Montage des Systems angezeigt. Ausführliche Informationen finden Sie unter *[3]*.

Installation und Wartung dürfen nur von Elektro-Fachkräften nach den technischen Regeln, Richtlinien und einschlägigen Normen, z. B. EN 60204 Teil 1, vorgenommen werden.

Montage von DX581-S

GEFAHR!
Einbau und Austausch im laufenden Betrieb sind bei Modulen unter Spannung nicht zulässig. Für jegliche Arbeiten an Sicherheitsmodulen müssen immer alle Spannungsquellen (Versorgungs- und Prozessspannungen) ausgeschaltet sein.

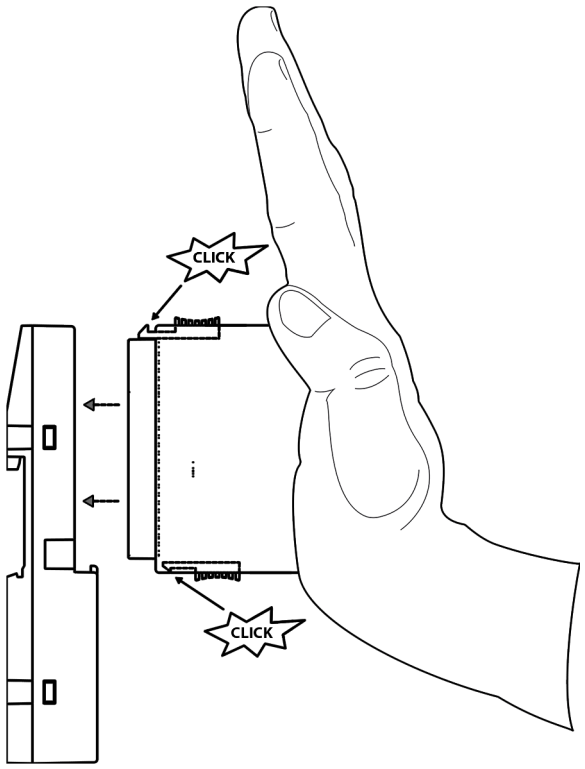


Abb. 40: Montageanleitung

1. Positionieren Sie das Modul auf dem Klemmenblock.
   ⇨ Das Modul rastet ein.
2. Drücken Sie das Modul dann mit einer Kraft von mindestens 100 N in den Klemmenblock, um einen zuverlässigen elektrischen Kontakt herzustellen.

Demontage von DX581-S

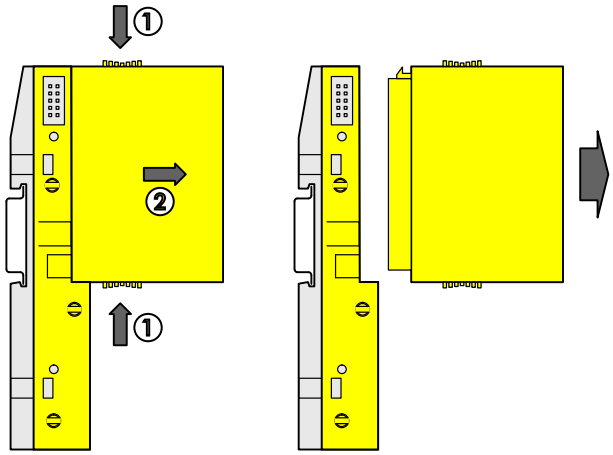


Abb. 41: Demontageanleitung

▷ Drücken Sie oben und unten, dann entfernen Sie das Modul.

Abmessungen

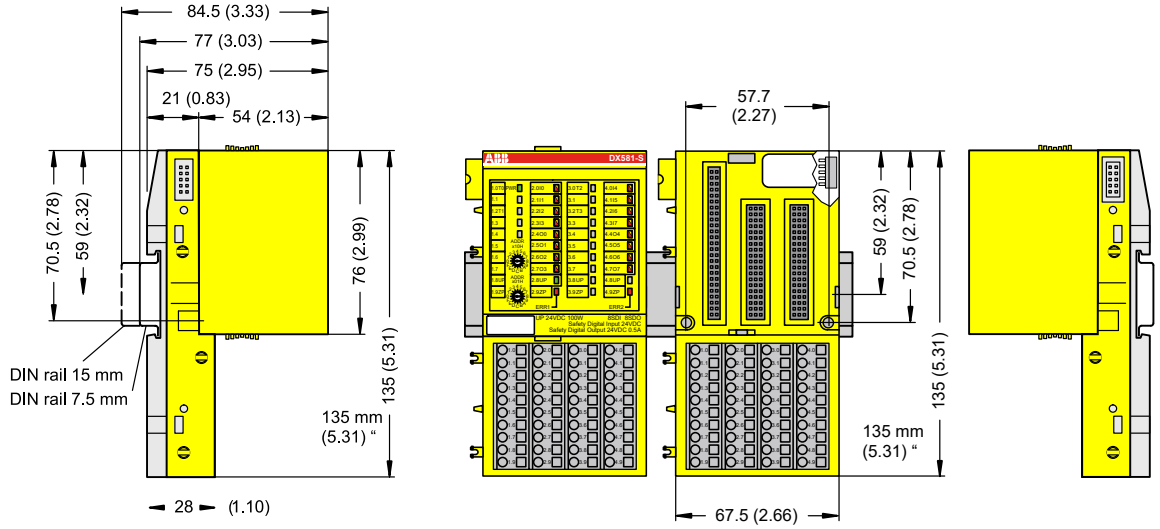


Abb. 42: Abmessungen des Sicherheits-E/A-Moduls DX581-S

Elektrischer Anschluss

HINWEIS!

Derselbe TU582-S wird für alle Sicherheits-E/A-Module der Serie AC500-S verwendet. Wenn der TU582-S für ein DX581-S mit sicherheitsgerichteten Digitalausgängen verdrahtet wird und ein DI581-S oder AI581-S versehentlich in diesen Klemmenblock gesteckt wird, ist es nicht möglich, dass die sicherheitsgerichteten Digitalausgangsklemmen am TU582-S durch falsch eingesteckte Sicherheits-E/A-Module DI581-S und AI581-S unter Spannung gesetzt werden.

Der elektrische Anschluss der Ein- und Ausgangskanäle erfolgt an den 40 Klemmen des E/A-Klemmenblocks. Auf diese Weise können die Module ausgetauscht werden, ohne dass die Verkabelung an den Klemmenblöcken gelöst werden muss.

Die Klemmen 1.8, 2.8, 3.8 und 4.8 bzw. 1.9, 2.9, 3.9 und 4.9 sind im Inneren des E/A-Klemmenblocks jeweils elektrisch miteinander verbunden und haben unabhängig vom eingesetzten Modul immer dieselbe Belegung:

- Klemmen 1.8, 2.8, 3.8 und 4.8: Prozessspannung UP = +24 V DC
- Klemmen 1.9, 2.9, 3.9 und 4.9: Prozessspannung ZP = 0 V

Belegung der weiteren Klemmen:

Klemmen	Signal	Bedeutung
1.0, 1.2, 3.0, 3.2	T0, T1, T2, T3	Anschlüsse der 4 Testimpuls-Ausgänge T0, T1, T2, T3
2.0 … 2.3, 4.0 … 4.3	I0, I1, I2, I3, I4, I5, I6, I7	8 sicherheitsgerichtete Digitaleingänge
2.4… 2.7, 4.4 … 4.7	O0, O1, O2, O3, O4, O5, O6, O7	8 sicherheitsgerichtete Digitalausgänge
1.8, 2.8, 3.8, 4.8	UP	Prozessversorgung +24 V DC
1.9, 2.9, 3.9, 4.9	ZP	Zentraler Erdanschluss der Prozessversorgungsspannung
1.1, 1.3, 1.4, 1.5, 1.6, 1.7, 3.1, 3.3, 3.4, 3.5, 3.6, 3.7	Frei	Nicht belegt

HINWEIS!
Die Prozessspannung muss in das Erdungskonzept des Steuerungssystems einbezogen werden (z. B. Erdung des Minuspols).

Anschlussbeispiele

Beispiele für elektrische Anschlüsse des DX581-S-Moduls und der Einzelkanäle Ix und Ox.

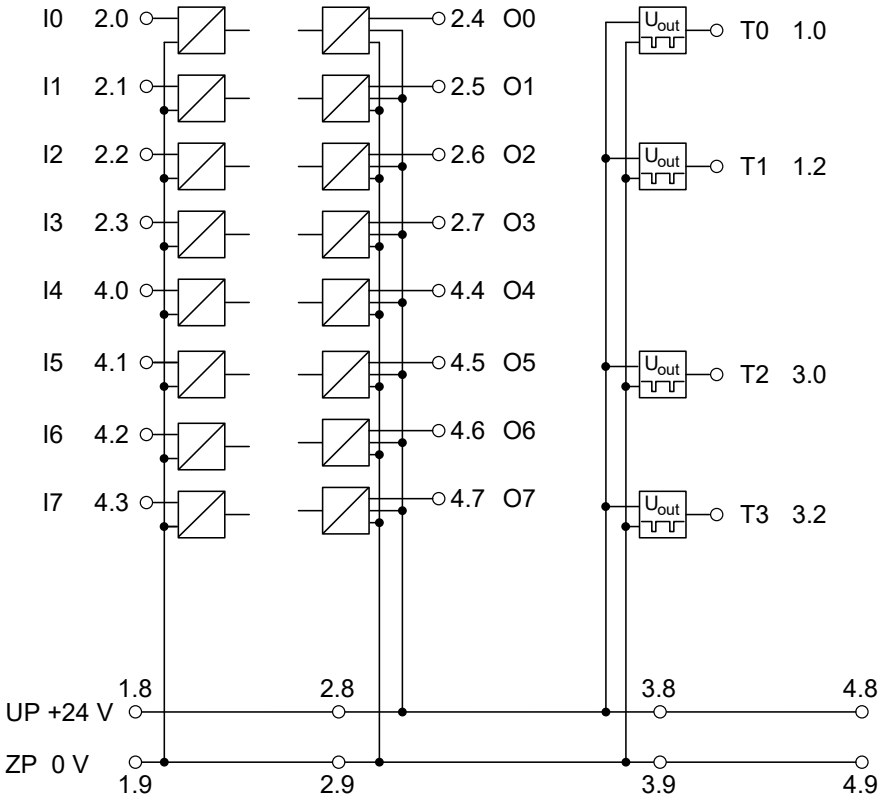


Abb. 43: Beispiel für elektrische Anschlüsse des DX581-S

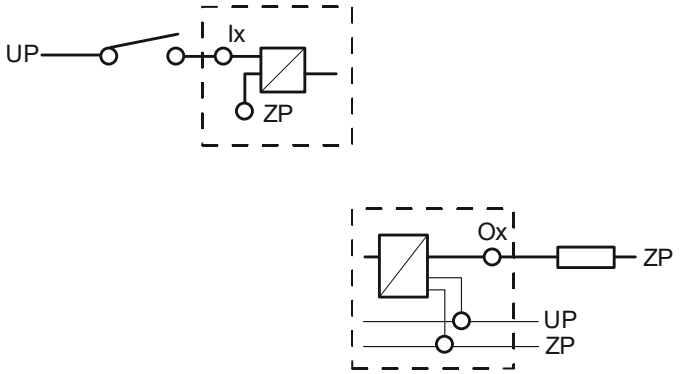


Abb. 44: Beispiel für Einzelkanäle des DX581-S

3.4.4 Interner Datenaustausch

Eingänge (Byte)	5
Ausgänge (Byte)	3

3.4.5 Konfiguration der Ein- und Ausgänge

Im digitalen Sicherheits-E/A-Modul DX581-S selbst werden keine Konfigurationsdaten gespeichert. Die Konfigurationsdaten werden in den Sicherheits- und Standard-CPUs gespeichert.

3.4.6 Parametrierung

Die Einrichtung der Parameterdaten wird mit der System-Konfigurationssoftware Automation Builder durchgeführt. Die GSDML-Datei von ABB für PROFINET-Geräte kann zum Konfigurieren der Parameter für DX581-S mit PROFINET F-Hosts von Drittanbietern verwendet werden.

Die Parametereinstellung hat unmittelbaren Einfluss auf die Funktionalität der Module und die für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreichbaren Werte.

Nr.	Name	Werte	Standard
1	Überwachung Spannung	„Ein“, „Aus“	„Ein“
Eingänge			
2	Eingangskanal-Konfiguration	„Nicht belegt“, „1 Kanal“, „2-Kanal äquivalent“, „2-Kanal antivalent“	„Nicht belegt“
3	Testimpuls	„Nicht verfügbar“, „Verfügbar“	„Nicht verfügbar“
4	Eingangsverzögerung	„1 ms“, „2 ms“, „5 ms“, „10 ms“, „15 ms“, „30 ms“, „50 ms“, „100 ms“, „200 ms“, „500 ms“	„5 ms“
5	Diskrepanzzeit*	„10 ms“, „20 ms“, „30 ms“, „40 ms“, „50 ms“, „60 ms“, „70 ms“, „80 ms“, „90 ms“, „100 ms“, „150 ms“, „200 ms“, „250 ms“, „300 ms“, „400 ms“, „500 ms“, „750 ms“, „1 s“, „2 s“, „3 s“, „4 s“, „5 s“, „10 s“, „20 s“, „30 s“	„50 ms“
Ausgänge			
6	Ausgangskanal-Konfiguration	„Nicht belegt“, „Belegt“	„Nicht belegt“
7	„Erkennung“ (interner Ausgangskanal-Test) ↳ *„Parameter „Erkennung“ der Ausgangskanäle“ auf Seite 103*	„Aus“, „Ein“	„Ein“

* Nur für Konfigurationen „2-Kanal äquivalent“ und „2-Kanal antivalent“ verfügbar

3.4.7 Anschlussbeispiele DX581-S

Beispiele der elektrischen Anschlüsse und der Werte, die für das Modul DX581-S für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreichbar sind, finden Sie weiter unten. Beachten Sie, dass die elektrischen Anschlüsse, die für sicherheitsgerichtete Eingangskanäle des DI581-S aufgeführt sind, auch für das DX581-S gelten.

HINWEIS!

Wenn DC = hoch in den Beschaltungsbeispielen mit sicherheitsgerichteten Digitaleingängen verwendet wird, wird die folgende Maßnahme aus ISO 13849-1 ↳ *[9]* für Modul DX581-S verwendet: Querschlussüberwachung von Eingangssignalen und Zwischenergebnissen innerhalb der Logik (L) sowie temporale und logische Software-Überwachung des Programmflusses und Erkennung von statischen Fehlern und Kurzschlüssen (bei mehreren E/As).

Wenn DC = mittel in den Beschaltungsbeispielen mit sicherheitsgerichteten Digitaleingängen verwendet wird, kann eine der Maßnahmen für die Eingabegeräte aus ISO 13849-1 ↳ *[9]* mit DC ≥ 90 % verwendet werden.

HINWEIS!

Wenn DC = hoch in den Beschaltungsbeispielen mit sicherheitsgerichteten Digitalausgängen verwendet wird, wird die folgende Maßnahme aus ISO 13849-1 ↳ *[9]* für Modul DX581-S verwendet: Querschlussüberwachung von Ausgangssignalen und Zwischenergebnissen innerhalb der Logik (L) sowie temporale und logische Softwareüberwachung des Programmflusses und Erkennung von statischen Fehlern und Kurzschlüssen (bei mehreren E/As).

Wenn in den Beschaltungsbeispielen DC = mittel mit sicherheitsgerichteten Digitalausgängen verwendet wird, kann eine beliebige Maßnahme für die Ausgabegeräte aus ISO 13849-1 ↳ *[9]* mit DC ≥ 90 % verwendet werden.

GEFAHR!

Die erreichbaren Werte für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) bei den Sicherheitsausgängen des DX581-S-Moduls sind nur gültig, wenn der Parameter „Erkennung“ = „Ein“ ist. Wenn der Parameter „Erkennung“ „Aus“ ist, wenden Sie sich an den technischen Support von ABB, um die für SIL, max. SIL und PL erreichbaren Werte zu erhalten.

Relais

Interner Ausgangskanal-Test	Ja
Max. SIL / PL [1)]	Max. SIL 1/PL c
SIL[2)]	SIL 2
Max. SIL / PL [3)]	Max. SIL 2 / PL d
SIL[4)]	SIL 3

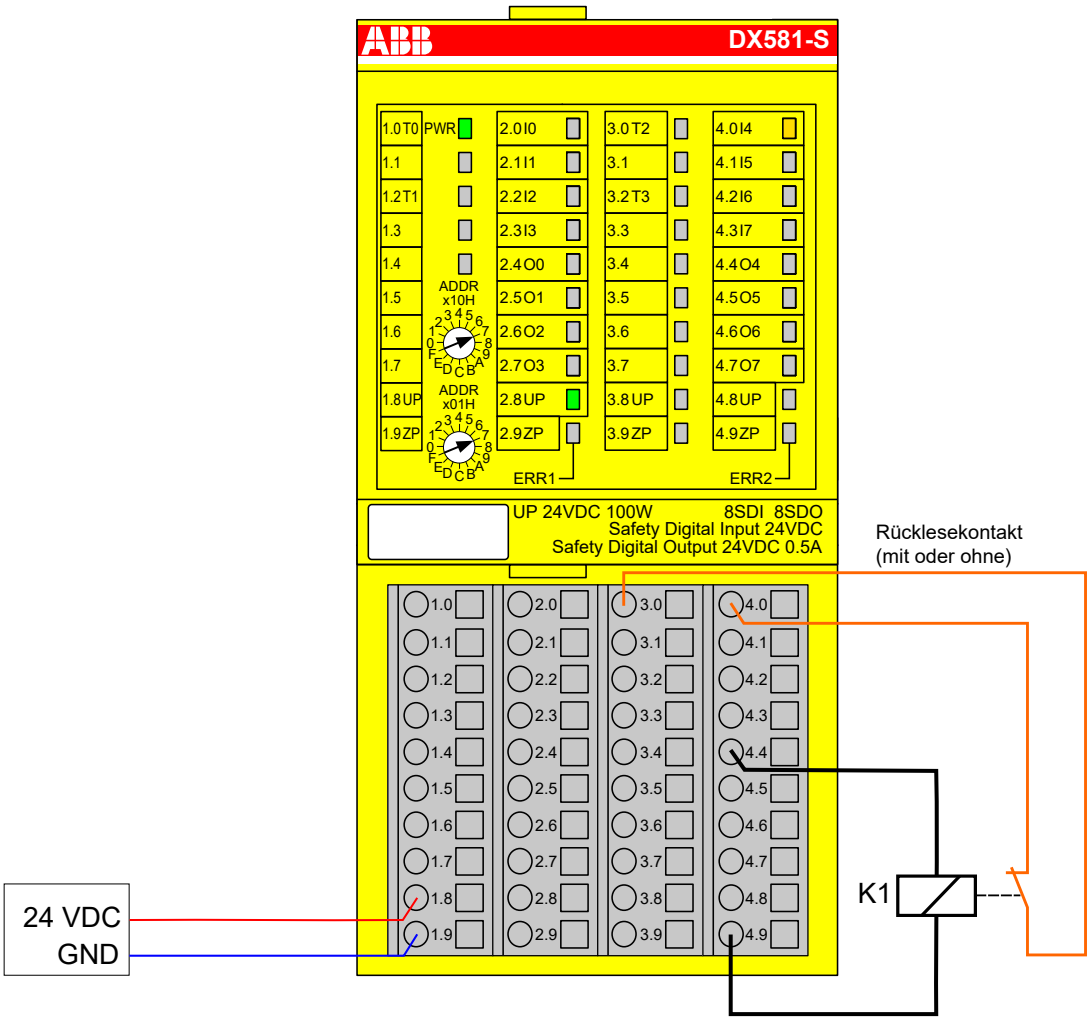


Abb. 45: Beschaltungsbeispiel DX581-S, Relais

[1)] - Ohne Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = 0

[2)] - Ohne Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss kann eine höhere Ebene bis zu SIL 3 erreicht werden)

[3)] - Mit Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = mittel

[4)] - Mit Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

2 Relais

Interner Ausgangskanal-Test	Ja
Max. SIL / PL [1)]	Max. SIL 1/PL c
SIL[2)]	SIL 3
Max. SIL / PL [3)]	Max. SIL 3 / PL e
SIL[4)]	SIL 3

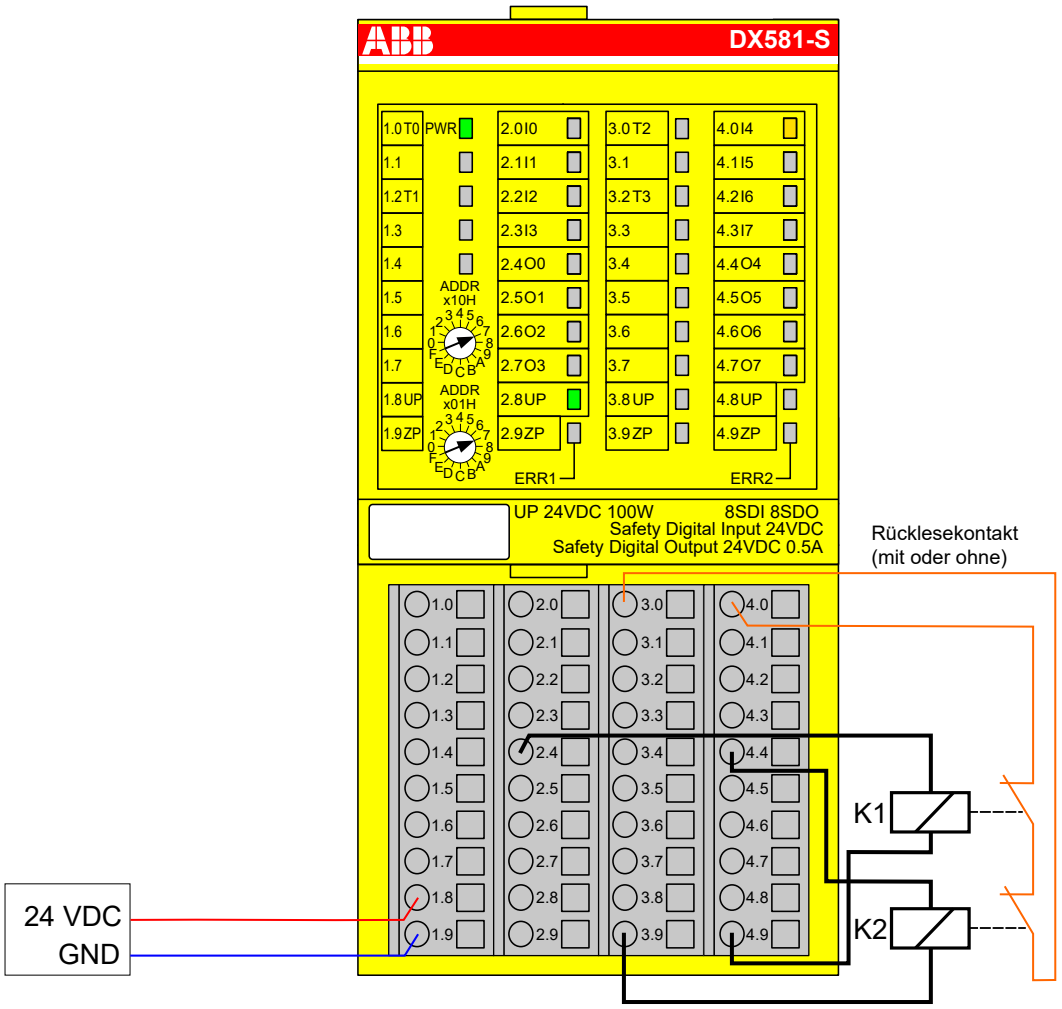


Abb. 46: Beschaltungsbeispiel DX581-S, 2 Relais

[1)] - Ohne Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = 0

[2)] - Ohne Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

[3)] - Mit Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) MTTFd = hoch; DC = hoch

[4)] - Mit Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Gerät mit Transistoreingang (1-Kanal)

Interner Ausgangskanal-Test	Ja
Max. SIL / PL [1)]	Max. SIL 1/PL c
SIL[2)]	SIL 2
Max. SIL / PL [3)]	Max. SIL 2 / PL d
SIL[4)]	SIL 3

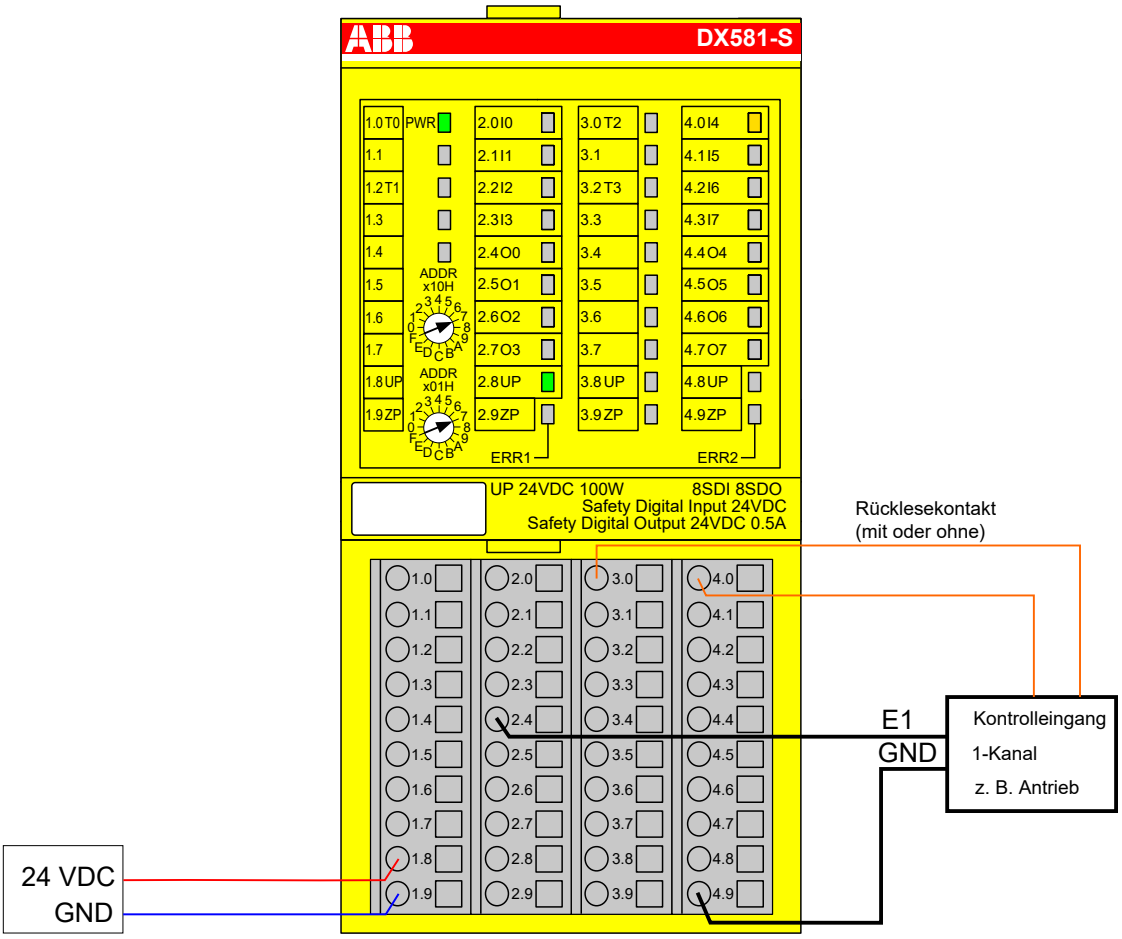


Abb. 47: Beschaltungsbeispiel DX581-S, Gerät mit Transistoreingang (1-Kanal)

[1)] - Ohne Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = 0

[2)] - Ohne Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss kann eine höhere Ebene bis zu SIL 3 erreicht werden)

[3)] - Mit Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = mittel

[4)] - Mit Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Gerät mit Transistoreingang (2-Kanal)

Interner Ausgangskanal-Test	Ja
Max. SIL / PL [1]	Max. SIL 1/PL c
SIL[2]	SIL 3
Max. SIL / PL [3]	Max. SIL 3 / PL e
SIL[4]	SIL 3

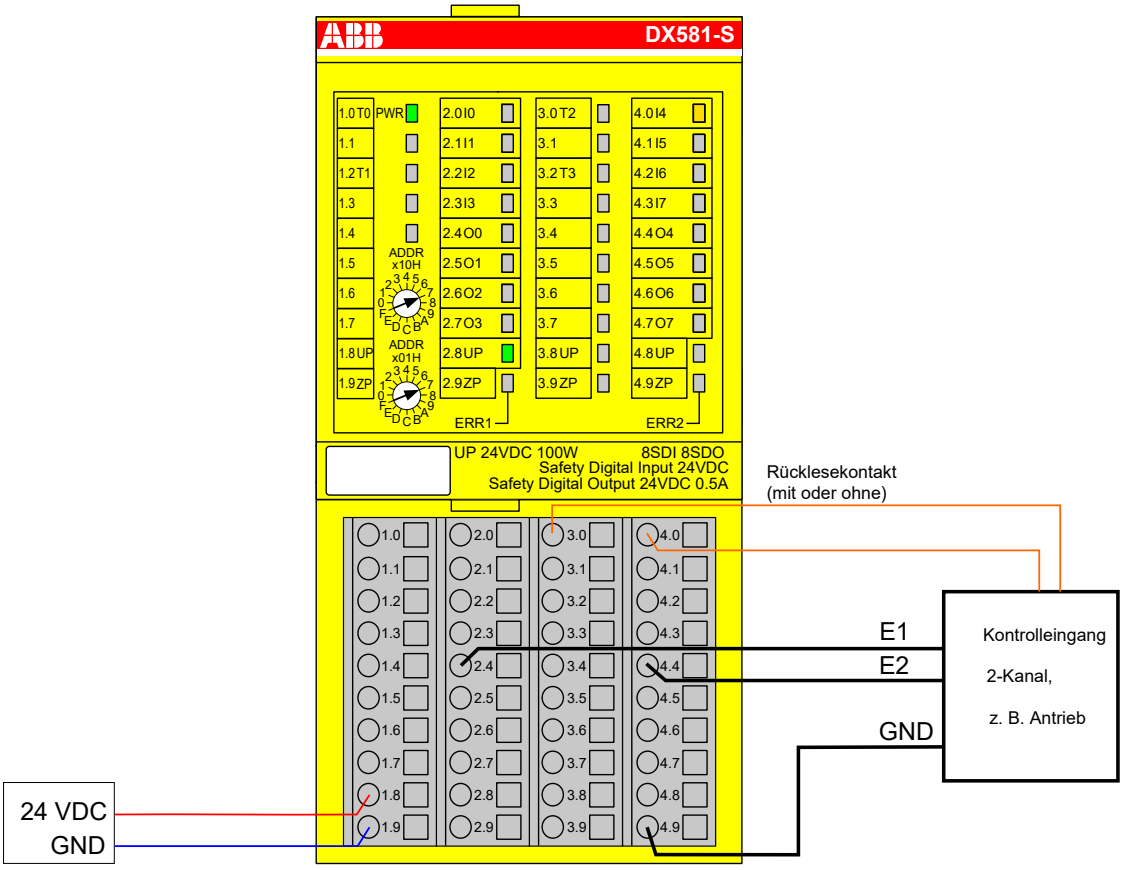


Abb. 48: Beschaltungsbeispiel DX581-S, Gerät mit Transistoreingang (2-Kanal)

[1] - Ohne Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = 0

[2] - Ohne Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

[3] - Mit Rücklesekontakt: Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) MTTFd = hoch; DC = mittel

[4] - Mit Rücklesekontakt: Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Fehlererkennung an der Ausgangsleitung der Lampe, des Ventils etc.

Interner Ausgangskanal-Test	Ja
Max. SIL / PL [1]	Max. SIL 2 / PL d Zusätzliche dynamische, anwendungsspezifische Tests zur Verdrahtung sind in Abhängigkeit von der Anwendung und der erforderlichen Verdrahtungsfehlererkennung (Kurzschluss zu 24 V DC, Fehler Kanalübersprechen an sicherheitsgerichteten Digitalausgängen etc.) erforderlich
SIL[2]	SIL 3

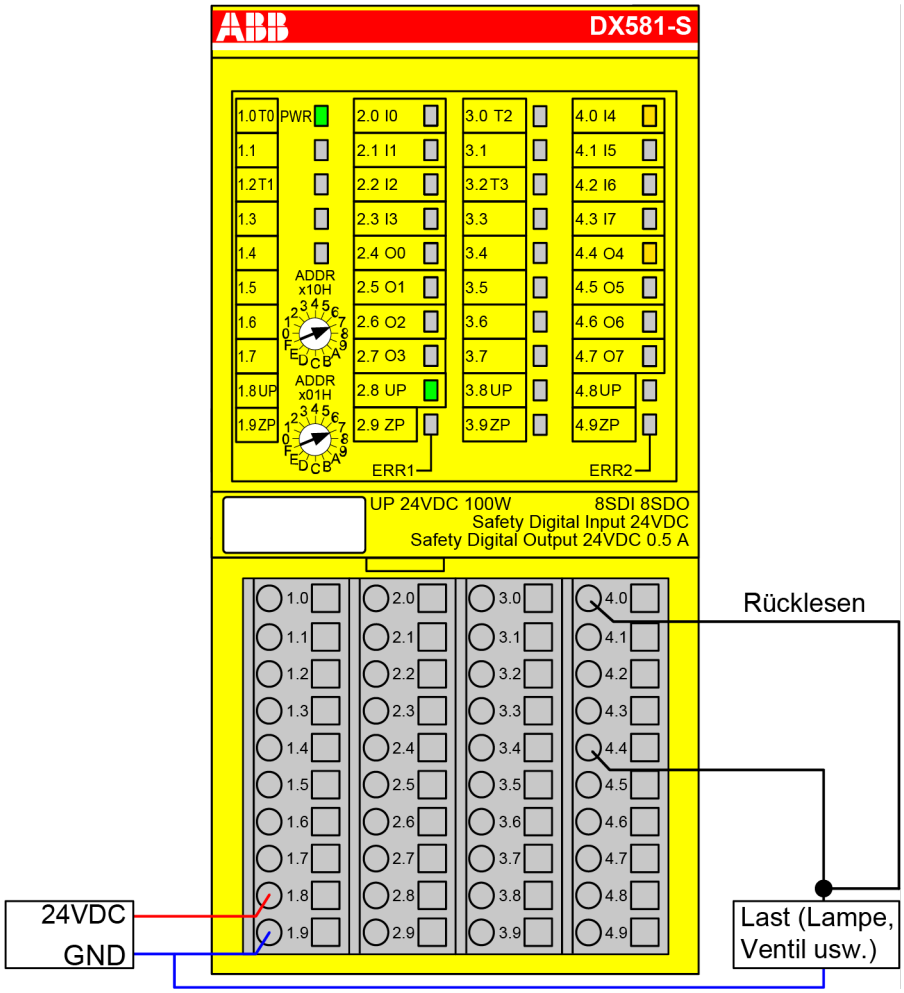


Abb. 49: Beschaltungsbeispiel DX581-S, Fehlererkennung an der Ausgangsleitung der Lampe, des Ventils etc.

[1] - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden) MTTFd = hoch; DC = mittel

[2] - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Anwendungs-beispiel

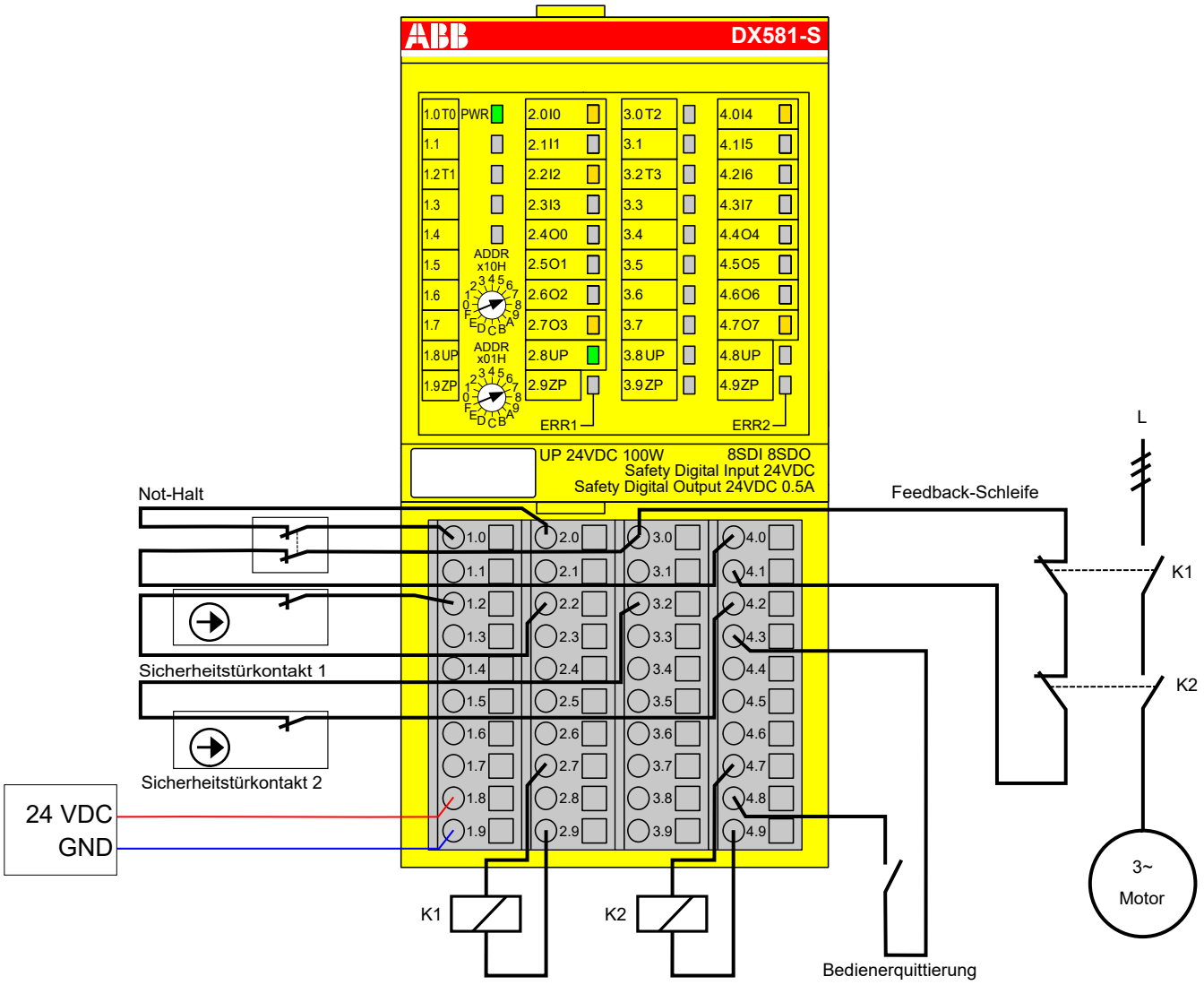


Abb. 50: Anwendungsbeispiel mit DX581-S

3.4.8 LED-Statusanzeige

Tab. 7: Statusanzeige und deren Bedeutung

LED	Beschreibung	Farbe	LED = AUS	LED = EIN	LED blinkt
Eingänge 0 … 7	Digitaleingang	Gelb	Eingang = AUS	Eingang = EIN (die Eingangsspannung wird auch angezeigt, wenn die Versorgungsspannung AUS ist)	–
	Kanalfehler	Rot	Kein Kanalfehler	Kanalfehler	–
Ausgänge 0 … 7	Digitalausgang	Gelb	Ausgang = AUS	Ausgang = EIN	–
	Kanalfehler	Rot	Kein Kanalfehler	Kanalfehler	–
UP	Prozessspannung +24 V DC an Klemme	Grün	Prozess-Versorgungsspannung fehlt	Prozess-Versorgungsspannung ist OK	–

LED	Beschreibung	Farbe	LED = AUS	LED = EIN	LED blinkt
PWR	+3,3 V Spannung von I/O-Bus	Grün	+3,3 V Spannung von I/O-Bus ist nicht verfügbar	+3,3 V Spannung von I/O-Bus verfügbar	–
ERR1	Modulfehler-Anzeige 1	Rot	Kein Modulfehler	Modulfehler, der zu einem SAFE STOP führt	Modulpassivierung und/oder Quittieranforderung (abwechselndes Blinken)
ERR2	Modulfehler-Anzeige 2	Rot			

3.4.9 Technische Daten

HINWEIS!
Die Version DX581-S-XC ist für eine Verwendung unter extremen Umgebungsbedingungen erhältlich (*Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*).

Weitere technische Daten stehen im SPS-Katalog von ABB zur Verfügung: *www.abb.com/plc*.

Prozess-Versorgungsspannung UP

Angabe	Wert	Einheit
Anschlussklemmen 1.8 … 4.8 (UP)	+24	V
Anschlussklemmen 1.9 … 4.9 (ZP)	0	V
Nennwert (-15 %, +20 %, ohne Restwelligkeit)	24	V DC
Max. Restwelligkeit	5	%
Verpolschutz	Ja	
Nennwert für Absicherung für UP (schnell)	10	A
Galvanische Trennung	pro Modul	
Verarbeitungsmechanismen von Ein-/Ausgängen	Regelmäßige Aktualisierung	
Stromaufnahme über UP im Normalbetrieb mit + 24 V DC (für Modulelektronik)	0,18	A
Einschaltstrom von UP bei 30 V (beim Einschalten)	0,1	A^2s
Einschaltstrom von UP bei 24 V (beim Einschalten)	0,06	A^2s

HINWEIS!
Alle Kanäle des DX581-S (einschließlich Testimpuls-Ausgänge) sind gegen Verpolung, Rückspeisung, Kurzschluss und andauernde Überspannung bis 30 V DC geschützt.

Einbaulage
Horizontal oder vertikal mit Leistungsreduzierung (Ausgangslast um 50 % bei +40 °C pro Gruppe reduziert und maximale Betriebstemperatur auf +40 °C reduziert)

Kühlung
Die natürliche Konvektionskühlung darf nicht durch Kabelkanäle oder andere Einbauten im Schaltschrank behindert werden.

Erlaubte Unterbrechungen der Spannungsversorgung laut EN 61131-2

Angabe	Wert	Einheit
Unterbrechungen der Gleichstromversorgung	< 10	ms
Zeit zwischen 2 Unterbrechungen der Gleichstromversorgung, PS2	> 1	s

Umgebungsbedingungen

Angabe	Wert	Einheit
Betriebstemperatur*	0 … +60	°C
Lagerungstemperatur	-40 … +85	°C
Transporttemperatur	-40 … +85	°C
Luftfeuchtigkeit ohne Kondensation	max. 95	%
Betriebsluftdruck	> 800	hPa
Lagerluftdruck	> 660	hPa
Betriebshöhe	< 2000	m über NN
Lagerhöhe	< 3500	m über NN

* Erweiterte Temperaturbereiche (unter 0 °C und über +60 °C) werden von Sonderversionen des DX581-S unterstützt *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Kriech- und Luftstrecken

Die Kriech- und Luftstrecken entsprechen der Überspannungskategorie II, Verschmutzungsgrad 2.

Netzteile

Zur Versorgung der Module müssen Netzteile gemäß PELV-/SELV-Spezifikationen verwendet werden.

Elektromagnetische Verträglichkeit

Informationen zur elektromagnetischen Verträglichkeit finden Sie im neuesten TÜV SÜD Report *[1]*.

Mechanische Eigenschaften

Angabe	Wert	Einheit
Schutzart	IP 20	
Gehäuse	gemäß UL94	
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 3,5 mm	2 … 15	Hz
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 1 g *	15 … 150	Hz
Stoßprüfung (alle drei Achsen), 11 ms Halbsinus	15	g
MTBF	73	Jahre

* Höhere Werte auf Anfrage

Selbsttest und Diagnosefunktionen

Tests während Start und Betrieb: Programmablauf-Überwachung, RAM, CPU, Kanalübersprechen, dauerhaftes 1-Signal usw.

Abmessungen, Gewicht

Angabe	Wert	Einheit
B × H × T	67,5 × 76 × 62	mm
Gewicht	~ 130	g

Zertifizierungen CE, cUL (weitere Zertifizierungen unter *www.abb.com/plc*)

3.4.9.1 Technische Daten der sicherheitsgerichteten Digitaleingänge

Angabe	Wert	Einheit
Anzahl Eingangskanäle je Modul	8	
Klemmen für Kanäle I0 bis I3	2.0 … 2.3	
Klemmen für Kanäle I4 bis I7	4.0 … 4.3	
Anschlüsse mit Bezugspotential für alle Eingänge (Minuspol der Prozess-Versorgungsspannung, Signalname ZP)	1.9 … 4.9	
Galvanische Trennung von den restlichen Teilen des Moduls (I/O-Bus)	Ja	
Eingangstyp gemäß EN 61131-2	Typ 1	
Eingangsverzögerung (0 ➔ 1 oder 1 ➔ 0), konfigurierbar	1 … 500	ms

Anzeige Eingangssignal Eine gelbe LED pro Kanal. Die LED ist EIN bei Eingangssignal „High“ (Signal 1).

Signalspannung

Angabe	Wert	Einheit
Eingangssignalspannung	24	V DC
Signal 0	-3 … +5	V
Undefiniertes Signal	> +5 … < +15	V
Signal 1	+15 … +30	V

Eingangsstrom je Kanal

Angabe	Wert	Einheit
Eingangsspannung +24 V, typisch	7	mA
Eingangsspannung +5 V	> 1	mA
Eingangsspannung +15 V	> 4	mA
Eingangsspannung +30 V	< 8	mA

Kabellänge

Angabe	Wert	Einheit
Max. Kabellänge, geschirmt	1000	m
Max. Kabellänge, ungeschirmt	600	m

3.4.9.2 Technische Daten der sicherheitsgerichteten Digitalausgänge

GEFAHR!
Das Überschreiten der zulässigen Prozess- oder Versorgungsspannung (< -35 V DC bzw. > +35 V DC) kann zu irreparablen Schäden am System führen.

Angabe	Wert	Einheit
Anzahl Kanäle pro Modul (Transistorausgänge)	8	
Anschlüsse mit Bezugspotential für alle Ausgänge (Minuspol der Prozess-Versorgungsspannung, Signalname ZP)	1.9 … 4.9	
Anschlüsse der gemeinsamen Versorgungsspannung für alle Ausgänge (Pluspol der Prozess-Versorgungsspannung, Signalname UP)	1.8 … 4.8	
Ausgangsspannung für 1-Signal	UP – 3	V
Ausgangsverzögerung (0 ➔ 1 oder 1 ➔ 0): Ausgangsstrom 5 mA	1	ms
Ausgangsverzögerung (0 ➔ 1 oder 1 ➔ 0): Ausgangsstrom 500 mA	4	ms
Möglichkeit des Schaltens einer kapazitiven Last von mindestens	300	µF
Möglichkeit des Schaltens einer induktiven Last von mindestens	1	H

Ausgangsstrom

Angabe	Wert	Einheit
Nennwert, je Kanal bei UP = 24 V	500	mA
Maximalwert (alle Kanäle zusammen)	4	A
Leckstrom bei 0-Signal	< 0,5	mA
Kurzschluss-/Überlastfestigkeit	Ja	
Überlastmeldung (Kanalpassivierung), I > 0,7 A	Ja	
Ausgangsstrombegrenzung (automatische Reaktivierung nach Kurzschluss/Überlast)	Ja	
Rückspannungsfestigkeit gegen 24-V-Signalanschluss	Ja	
Demagnetisierung durch interne Löschdioden beim Ausschalten induktiver Lasten	Ja	
Nennwert für Absicherung an UP	4,5	A

Kabellänge

Angabe	Wert	Einheit
Max. Kabellänge, geschirmt	1000	m
Max. Kabellänge, ungeschirmt	600	m

3.4.9.3 Technische Daten der nicht sicheren Testimpuls-Ausgänge

Angabe	Wert	Einheit
Anzahl Testimpuls-Kanäle pro Modul (Ausgänge für Transistor-Testimpuls)	4	
Klemmen für Kanäle T0, T1	1.0, 1.2	
Klemmen für Kanäle T2, T3	3.0, 3.2	
Anschlüsse mit Bezugspotential für alle Testimpuls-Ausgänge (Minuspol der Prozess-Versorgungsspannung, Signalname ZP)	1.9 … 4.9	

Angabe	Wert	Einheit
Anschlüsse der gemeinsamen Versorgungsspannung für alle Ausgänge (Pluspol der Prozess-Versorgungs-spannung, Signalname UP)	1.8 … 4.8	
Ausgangsspannung für 1-Signal	UP – 0,8	V
Länge der Testimpuls-0-Phase	1	ms

Ausgangsstrom

Angabe	Wert	Einheit
Nennwert, je Kanal	10	mA
Maximalwert (alle Kanäle zusammen)	40	mA
Kurzschluss-/Überlastfestigkeit	Ja	
Ausgangsstrombegrenzung	65	mA
Rückspannungsfestigkeit gegen 24-V-Signalanschluss	Ja	

Kabellänge

Angabe	Wert	Einheit
Max. Kabellänge, geschirmt	1000	m
Max. Kabellänge, ungeschirmt	600	m

3.4.10 Bestelldaten

Typ	Beschreibung	Bestellnummer
DX581-S	Digitales Sicherheits-E/A-Modul 8SDI/SDO	1SAP 284 100 R0001
DX581-S-XC	Digitales Sicherheits-E/A-Modul 8SDI/SDO, extreme Umgebungs-bedingungen	1SAP 484 100 R0001

3.5 Analoges Sicherheits-Eingabemodul AI581-S

Elemente des Moduls

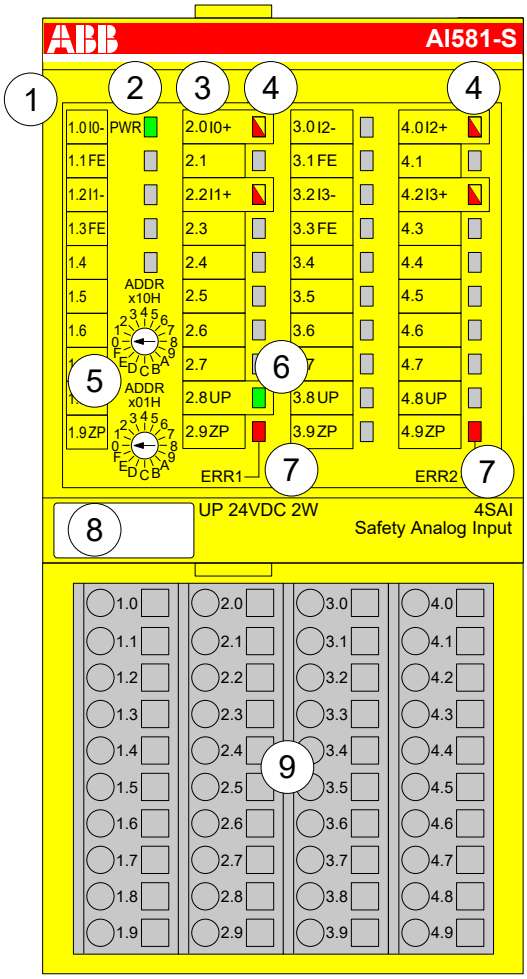


Abb. 51: Analoges Sicherheits-Eingabemodul AI581-S, eingesteckt in Klemmenblock TU582-S

1 I/O-Bus
2 System-LED
3 Zuordnung Klemmennummer – Signalname
4 4 gelb/rote LEDs Signalstatus I0 … I1/I2 … I3
5 2 Drehschalter für PROFIsafe-Adresse
6 Grüne LED für Prozessspannung UP
7 Rote LEDs zur Anzeige von Modulfehlern
8 Beschriftungsschild (TA525)
9 E/A-Klemmenblock (TU582-S)

3.5.1 Verwendungszweck

Das analoge Sicherheits-Eingabemodul AI581-S kann als dezentrales Erweiterungsmodul für die PROFINET-Module CI501-PNIO, CI502-PNIO, CI504-PNIO und CI506-PNIO oder lokal an CPUs der AC500-Serie für Sicherheitsanwendungen bis SIL 3 (IEC 61508), max. SIL 3 (IEC 62061) und PL e (ISO 13849-1) verwendet werden.

HINWEIS!

Die Werte, die mit Ihrer Sicherheitsanwendung für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreicht werden können, hängen von der Verdrahtung der Sensoren im AI581-S-Modul ab ↯ *Kapitel 3.5.7 „Anschluss-beispiele AI581-S“ auf Seite 128*.

AI581-S enthält 4 Sicherheits-Strom-Analogeingänge in zwei Gruppen (2.0 ... 2.2 und 4.0 … 4.2) ohne Potentialtrennung zwischen den Kanälen.

Die Eingänge sind gegenüber den anderen Schaltkreisen des Moduls nicht galvanisch getrennt.

3.5.2 Funktionalität

Analogeingänge	4 (0 … 20 mA oder 4 … 20 mA)
LED-Anzeigen	Für Signalzustand, Modulfehler, Kanalfehler und Versorgungsspannung
Interne Spannungsversorgung	über I/O-Bus-Schnittstelle
Externe Spannungsversorgung	Über Klemmen ZP und UP (Prozessspannung 24 V DC)

Selbsttests und Diagnosefunktionen (sowohl beim Starten als auch während des Betriebs), wie CPU- und RAM-Tests, Programmablauf-Überwachung und Kanalübersprechen usw., werden in AI581-S gemäß den Anforderungen von IEC 61508 SIL 3 implementiert.

HINWEIS!
Nur F_Dest_Add wird für die PROFIsafe F-Device-Identifizierung im AI581-S verwendet.

AI581-S verfügt über 4 Sicherheits-Analogeingangskanäle mit den folgenden Funktionen:

- 14-Bit-Auflösung
- Überprüfung der Prozess-Spannungsversorgung (eine Diagnosemeldung, die über die fehlende Prozess-Spannungsversorgung für ein entsprechendes Sicherheits-E/A-Modul informiert, wird vom Sicherheits-E/A-Modul an die CPU gesendet). Diese Funktion ist nicht sicherheitsbezogen und steht nicht im Zusammenhang mit der internen sicherheitsrelevanten Über- oder Unterspannungserkennung.
- Störfrequenzunterdrückung 50 Hz oder 60 Hz
- Modi 1-Kanal (0 … 20 mA), 1-Kanal (4 … 20 mA) oder 2-Kanal (4 … 20 mA) (Minimal- oder Maximalwert kann für die Übertragung zur Sicherheits-CPU im 2-Kanal-Modus (4 … 20 mA) gewählt werden; Toleranzbereich 4 … 12 % kann für 2-Kanal-Modus gesetzt werden)

HINWEIS!
In einem 2-Kanal-Modus transportiert der niedrigere Kanal (Kanäle 0/2 ➔ Kanal 0, Kanäle 1/3 ➔ Kanal 1 usw.) gesammelt den Prozesswert, das PROFIsafe-Diagnosebit, die Quittierungsanforderung und die Acknowledge-Reintegrationsinformation. Der höhere Kanal liefert immer den passivierten Wert „0“.

HINWEIS!
Die maximale interne Diskrepanzzeit zwischen zwei internen Kanalwerten (1-Kanal- oder 2-Kanal-Modus) ist beim AI581-S-Modul 67,5 ms; dies ist auch der interne Wert für die Worst-Case-Eingangsverzögerung.

Die Diskrepanzzeit zwischen zwei Kanalwerten (2-Kanal-Modus) mit dem gewählten überwachten Toleranzbereich (4 … 12 %) beträgt ebenfalls 67,5 ms.

HINWEIS!
Die Analogeingangskanäle haben integrierte Hardware-Tiefpassfilter von 100 Hz.

HINWEIS!
Wenn am Sicherheitskanal des Analogeingangs ein Überstrom/Unterstrom erkannt wird, wird die Kanalpassivierung nach spätestens 200 ms ausgelöst. Nachdem der Kanal für 30 s passiviert war, wird erneut auf anliegenden Überstrom/Unterstrom geprüft. Ist kein Überstrom/Unterstrom vorhanden, wird das Signal für die Reintegrationsanforderung für den entsprechenden Kanal auf TRUE gesetzt, sodass der Kanal wieder integriert werden kann.

Die folgende Tabelle zeigt das Abbild der Prozesswerte der Sicherheits-CPU auf die entsprechenden mA-Werte des AI581-S-Moduls. Für einen Analogeingang sind zwei Modi definiert, 0 … 20 mA und 4 … 20 mA.

HINWEIS!
Sowohl Überlauf als auch Überschreitung stellen einen Überstrom dar. Sowohl Unterlauf als auch Unterschreitung stellen einen Unterstrom dar.

Nur im Falle von Über- und Unterlauf werden die Analogkanäle passiviert und „0“-Prozesswerte an die Sicherheits-CPU übertragen.

Bereich	**0 … 20 mA**	**4 … 20 mA**	**Digitalwert (dez)**		**Digitalwert (hex)**	
Überlauf*	: > 23,519	: > 22,81	32767* : 32512*		7FFF* : 7F00*	
Überschreitung	23,519 : 20,000723	22,81 : 20,000578	32511 : 27649		7EFF : 6C01	
Nennbereich	20 : 0	20 : 16 : 4	27648 : 20736 : 0		6C00 : 5100 : 0000	
Unterschreitung	-0,000723 : -1,481	3,999421 : 1,185	**0 … 20 mA** -1 : -2048	**4 … 20 mA** -1 : -4864	**0 … 20 mA** FFFF : F800	**4 … 20 mA** FFFF : ED00
Unterlauf*	< -1,481	< 1,185	**0 … 20 mA** -2049* : -32768*	**4 … 20 mA** -4865* : -32678*	**0 … 20 mA** F7FF* : 8000*	**4 … 20 mA** ECFF* : 8000*

* In diesen Fällen werden die Analogkanäle passiviert und „0“-Prozesswerte an die Sicherheits-CPU übertragen.

3.5.3 Montage, Abmessungen und elektrischer Anschluss

Die Eingabemodule können nur in den E/A-Klemmenblock mit Federzugklemmen TU582-S eingesteckt werden. Die eindeutige mechanische Codierung auf den E/A-Klemmenblöcken verhindert eventuelle Fehler, sodass keine Standard-E/A-Module in den Sicherheits-E/A-Klemmenblock eingesteckt werden können und umgekehrt. Hier werden grundlegende Informationen zur Montage des Systems angezeigt. Ausführliche Informationen finden Sie unter ↬ *[3]*.

Installation und Wartung dürfen nur von Elektro-Fachkräften nach den technischen Regeln, Richtlinien und einschlägigen Normen, z. B. EN 60204 Teil 1, vorgenommen werden.

Montage von AI581-S

GEFAHR!
Einbau und Austausch im laufenden Betrieb sind bei Modulen unter Spannung nicht zulässig. Für jegliche Arbeiten an Sicherheitsmodulen müssen immer alle Spannungsquellen (Versorgungs- und Prozessspannungen) ausgeschaltet sein.

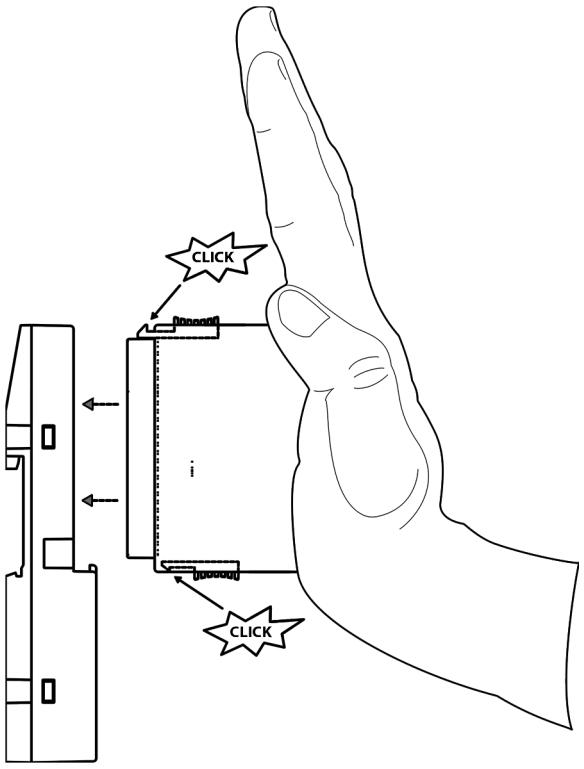


Abb. 52: Montageanleitung

1. Positionieren Sie das Modul auf dem Klemmenblock.
   ⇨ Das Modul rastet ein.
2. Drücken Sie das Modul dann mit einer Kraft von mindestens 100 N in den Klemmenblock, um einen zuverlässigen elektrischen Kontakt herzustellen.

Demontage von AI581-S

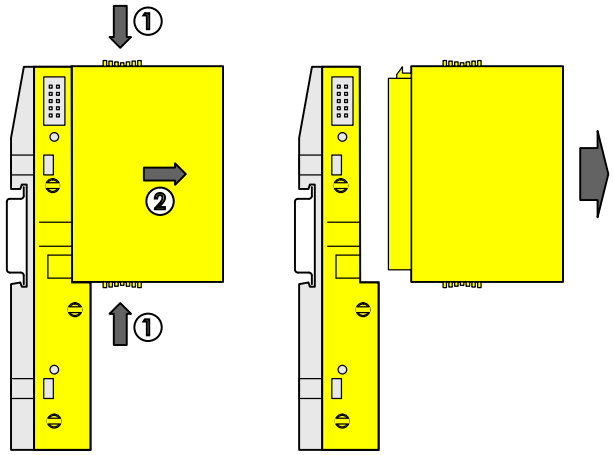


Abb. 53: Demontageanleitung

▷ Drücken Sie oben und unten, dann entfernen Sie das Modul.

Abmessungen

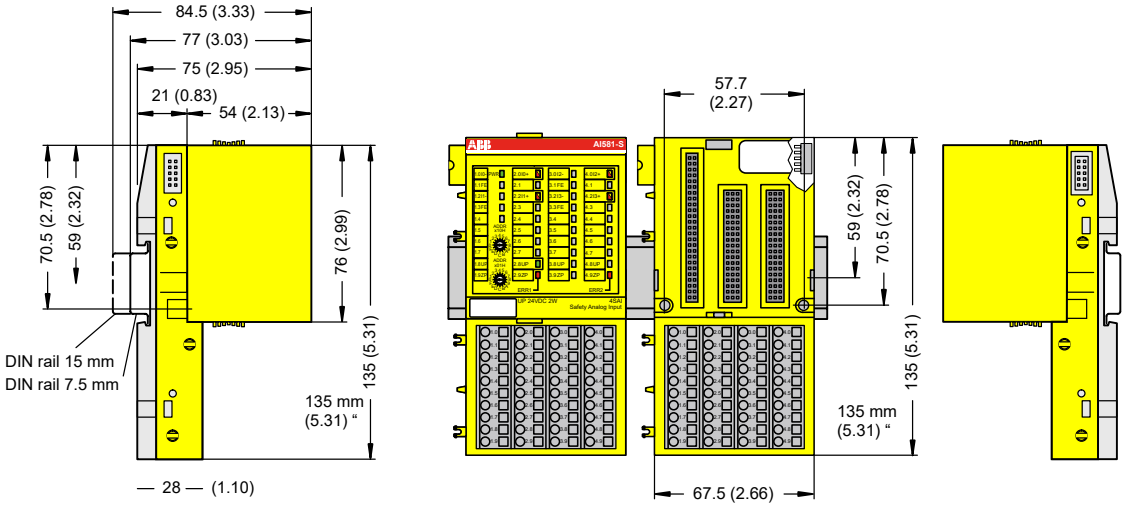


Abb. 54: Abmessungen des Sicherheits-E/A-Moduls AI581-S

Elektrischer Anschluss

HINWEIS!

Derselbe TU582-S wird für alle Sicherheits-E/A-Module der Serie AC500-S verwendet. Wenn der TU582-S für ein DX581-S mit sicherheitsgerichteten Digitalausgängen verdrahtet wird und ein DI581-S oder AI581-S versehentlich in diesen Klemmenblock gesteckt wird, ist es nicht möglich, dass die sicherheitsgerichteten Digitalausgangsklemmen am TU582-S durch falsch eingesteckte Sicherheits-E/A-Module DI581-S und AI581-S unter Spannung gesetzt werden.

Der elektrische Anschluss der Ein- und Ausgangskanäle erfolgt an den 40 Klemmen des E/A-Klemmenblocks. Auf diese Weise können die Module ausgetauscht werden, ohne dass die Verkabelung an den Klemmenblöcken gelöst werden muss.

Die Klemmen 1.8, 2.8, 3.8 und 4.8 bzw. 1.9, 2.9, 3.9 und 4.9 sind im Inneren des E/A-Klemmenblocks jeweils elektrisch miteinander verbunden und haben unabhängig vom eingesetzten Modul immer dieselbe Belegung:

- Klemmen 1.8, 2.8, 3.8 und 4.8: Prozessspannung UP = +24 V DC
- Klemmen 1.9, 2.9, 3.9 und 4.9: Prozessspannung ZP = 0 V

Belegung der weiteren Klemmen:

Klemmen	Signal	Bedeutung
1.0, 1.2, 3.0, 3.2	I0-, I1-, I2-, I3-	Negative Anschlüsse der 4 Analogeingänge
2.0, 2.2, 4.0, 4.2	I0+, I1+, I2+, I3+	Positive Anschlüsse der 4 Analogeingänge
1.1, 1.3, 3.1, 3.3	FE	Funktionserde
1.8, 2.8, 3.8, 4.8	UP	Prozessversorgung +24 V DC
1.9, 2.9, 3.9, 4.9	ZP	Zentraler Erdanschluss der Prozessversorgungsspannung
1.4 … 1.7, 2.1, 2.3 … 2.7, 3.4 … 3.7, 4.1, 4.3 … 4.7	Frei	Nicht belegt

HINWEIS!
Die Prozessspannung muss in das Erdungskonzept des Steuerungssystems einbezogen werden (z. B. Erdung des Minuspols).

HINWEIS!
Die Minuspole der Analogeingänge sind untereinander elektrisch verbunden. Sie bilden ein „Analogmasse“-Signal für das Modul.

Aufgrund des gemeinsamen Bezugspotenzials können analoge Stromeingänge nicht in Reihe hintereinander geschaltet werden, weder innerhalb des Moduls selbst noch mit Kanälen anderer Module.

HINWEIS!
Es besteht keine galvanische Trennung zwischen den analogen Kreisen und ZP/UP. Daher müssen die analogen Sensoren galvanisch getrennt sein, um Schleifen über das Erdpotenzial oder die Versorgungsspannung zu verhindern.

HINWEIS!
Analoge Signalleitungen werden grundsätzlich in geschirmten Kabeln geführt. Die Abschirmung wird an beiden Kabelenden geerdet. Um unzulässige Potenzialdifferenzen zwischen verschiedenen Anlagenteilen zu vermeiden, müssen niederohmige Potenzialausgleichsleitungen verlegt werden.

Bei einfachen Anwendungen (wenig Störungen, geringe Anforderungen an die Präzision) kann auch auf die Schirmung verzichtet werden.

Anschlussbeispiele

Beispiele für elektrische Anschlüsse des AI581-S-Moduls und der Einzelkanäle Ix.

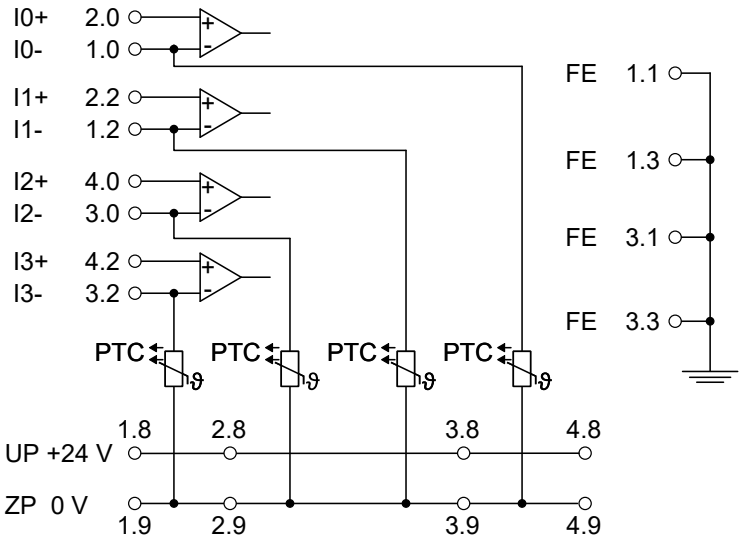


Abb. 55: Beispiel für elektrische Anschlüsse des AI581-S

HINWEIS!
Der PTC, der im Anschlussschema angegeben ist, ist in das Modul AI581-S eingebaut.

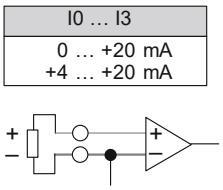


Abb. 56: Beispiel für Einzelkanäle des AI581-S

3.5.4 Interner Datenaustausch

Eingänge (Byte)	9
Ausgänge (Byte)	1

3.5.5 Konfiguration der Ein- und Ausgänge

Im analogen Sicherheits-Eingabemodul AI581-S selbst werden keine Konfigurationsdaten gespeichert. Die Konfigurationsdaten werden in den Sicherheits- und Standard-CPUs gespeichert.

3.5.6 Parametrierung

Die Einrichtung der Parameterdaten wird mit der System-Konfigurationssoftware Automation Builder durchgeführt. Die GSDML-Datei von ABB für PROFINET-Geräte kann zum Konfigurieren der Parameter für AI581-S mit PROFINET F-Hosts von Drittanbietern verwendet werden.

Die Parametereinstellung hat unmittelbaren Einfluss auf die Funktionalität der Module und die für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreichbaren Werte.

Nr.	Name	Werte	Standard
1	Überwachung Spannung	„Ein“, „Aus“	„Ein“
2	Konfiguration	„Nicht belegt“, „1-Kanal (0 … 20 mA)“, „1-Kanal (4 … 20 mA)“, „2-Kanal (4 … 20 mA)“	„Nicht belegt“
3	Störfrequenzunterdrückung	„50 Hz“, „60 Hz“, „keine“	„50 Hz“
4	Toleranzbereich (nur für „2-Kanal (4 … 20 mA)“-Modus)	„4 %“, „5 %“, „6 %“, „7 %“, „8 %“, „9 %“, „10 %“, „11 %“, „12 %“	„4 %“
5	Verwendeter Wert (min./max.) (nur verwendet für „2-Kanal (4 … 20 mA)“-Modus)	„Minimum“, „Maximum“	„Minimum“

3.5.7 Anschlussbeispiele AI581-S

Beispiele der elektrischen Anschlüsse und der Werte, die für das Modul AI581-S für SIL (IEC 61508), max. SIL (IEC 62061) und PL (ISO 13849-1) erreichbar sind, finden Sie weiter unten.

HINWEIS!

Wenn DC = hoch in den Beschaltungsbeispielen mit sicheren Analogeingängen verwendet wird, wird die folgende Maßnahme aus ISO 13849-1 ↬ *[9]* für Modul AI581-S verwendet: Querschlussüberwachung von Eingangssignalen und Zwischenergebnissen innerhalb der Logik (L) sowie temporale und logische Software-Überwachung des Programmflusses und Erkennung von statischen Fehlern und Kurzschlüssen (bei mehreren E/As).

Wenn DC = mittel in den Beschaltungsbeispielen mit sicheren Analogeingängen verwendet wird, kann eine der Maßnahmen für die Eingabegeräte aus ISO 13849-1 ↬ *[9]* mit DC ≥ 90 % verwendet werden.

Analogsensor (0 … 20 mA)

Max. SIL / PL [1), 2)]	Max. SIL 1/PL c
SIL[3)]	SIL 1

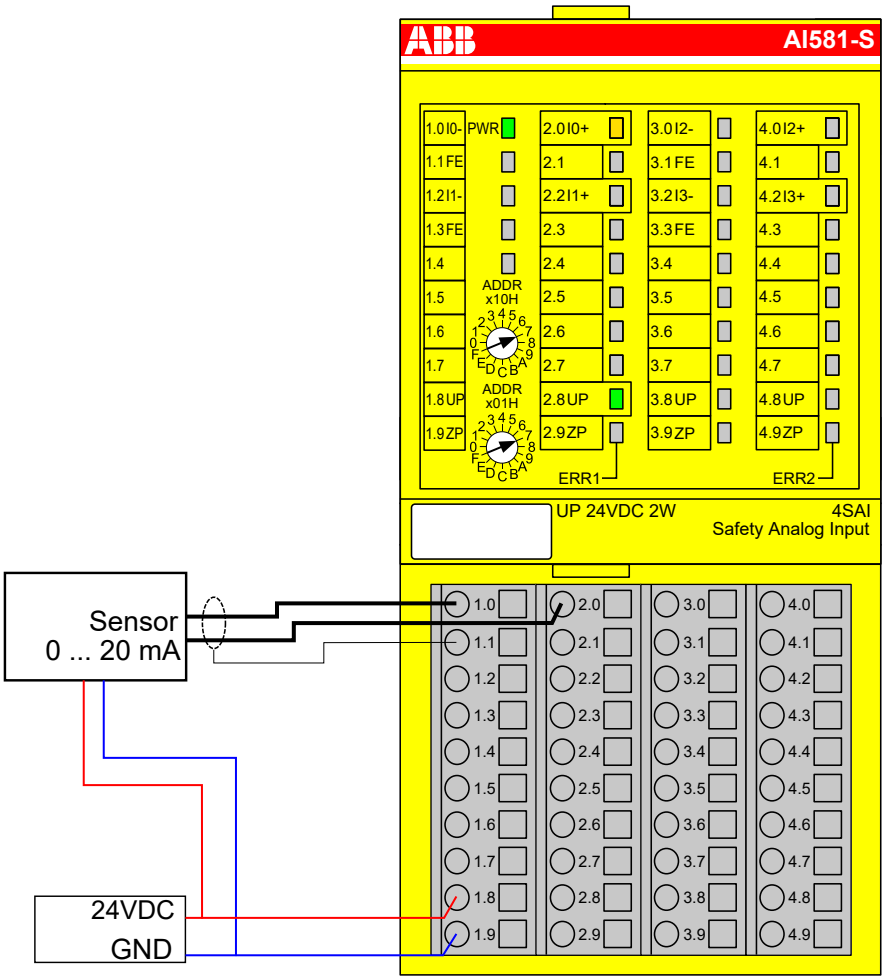


Abb. 57: Beschaltungsbeispiel AI581-S, Analogsensor (0 … 20 mA)

[1)] - MTTFd = hoch, DC = gering

[2)] - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

[3)] - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu SIL 3 erreicht werden)

2 Analogsensoren (0 … 20 mA)

2-Kanal-Auswertung	Im AI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 3

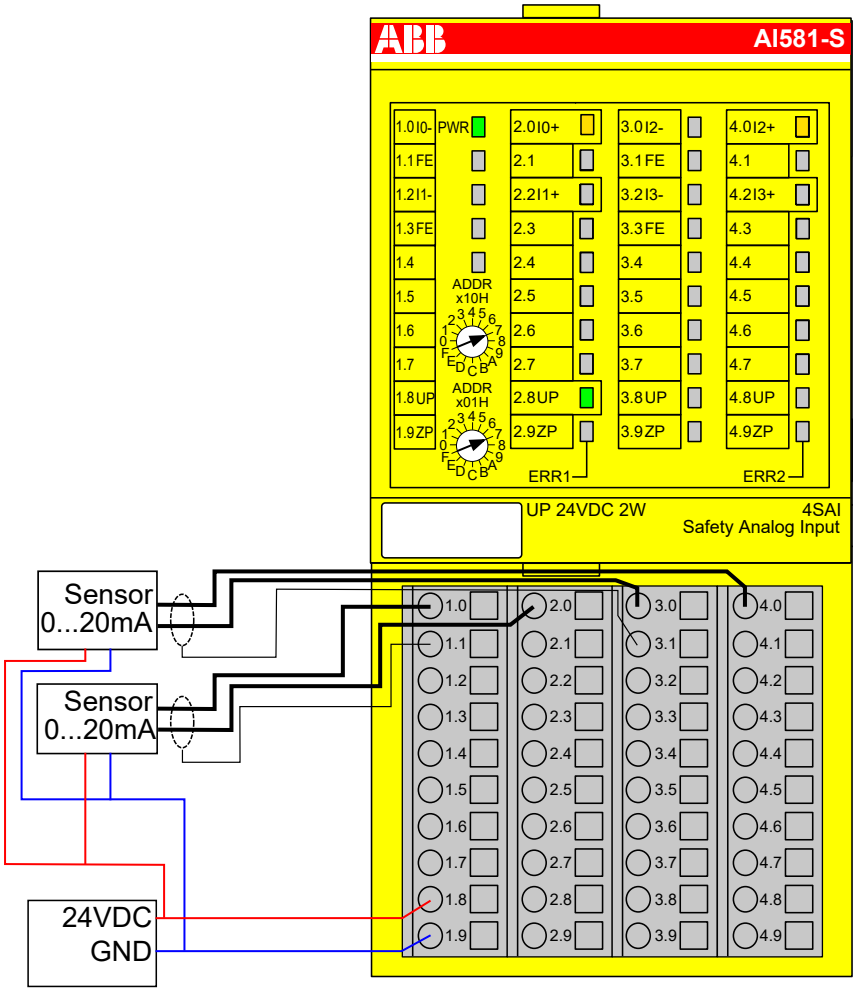


Abb. 58: Beschaltungsbeispiel AI581-S, 2 Analogsensoren (0 … 20 mA)

1) - MTTFd = hoch, DC = mittel

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Analogsensor (4 … 20 mA)

Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 2

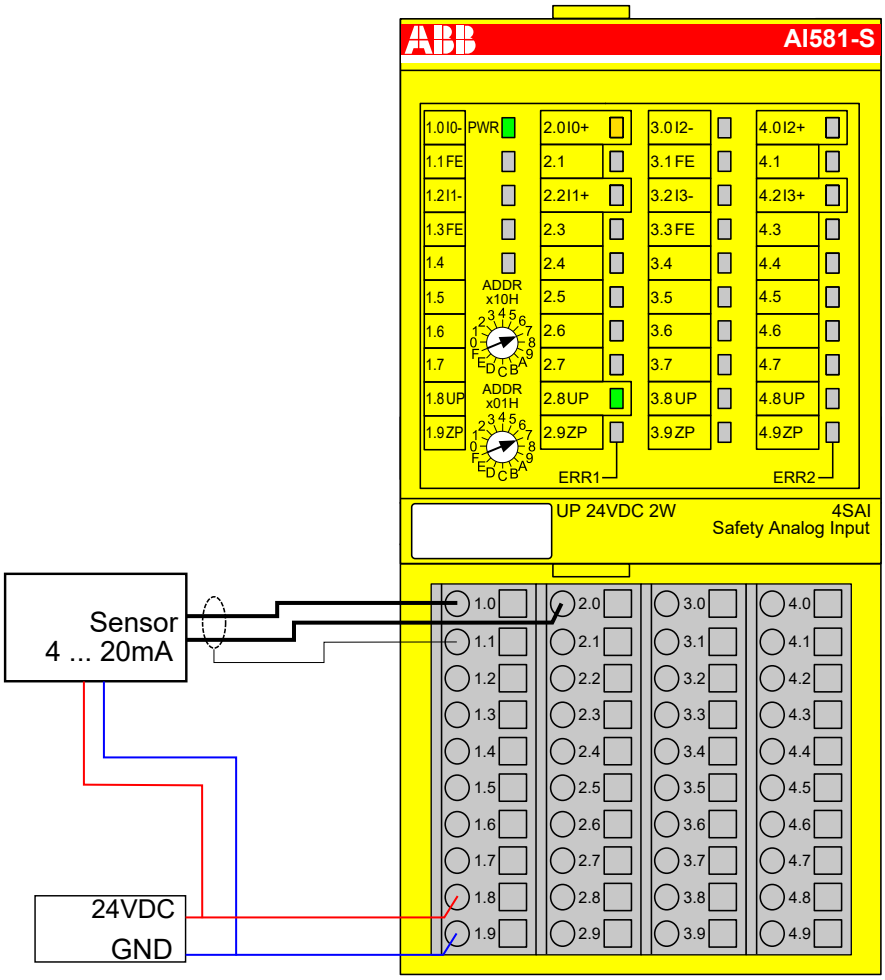


Abb. 59: Beschaltungsbeispiel AI581-S, Analogsensor (4 … 20 mA)

1) - MTTFd = hoch, DC = mittel

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu SIL 3 erreicht werden)

2 Analogsensoren (4 … 20 mA)

2-Kanal-Auswertung	Im AI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 3 / PL e
SIL[3)]	SIL 3

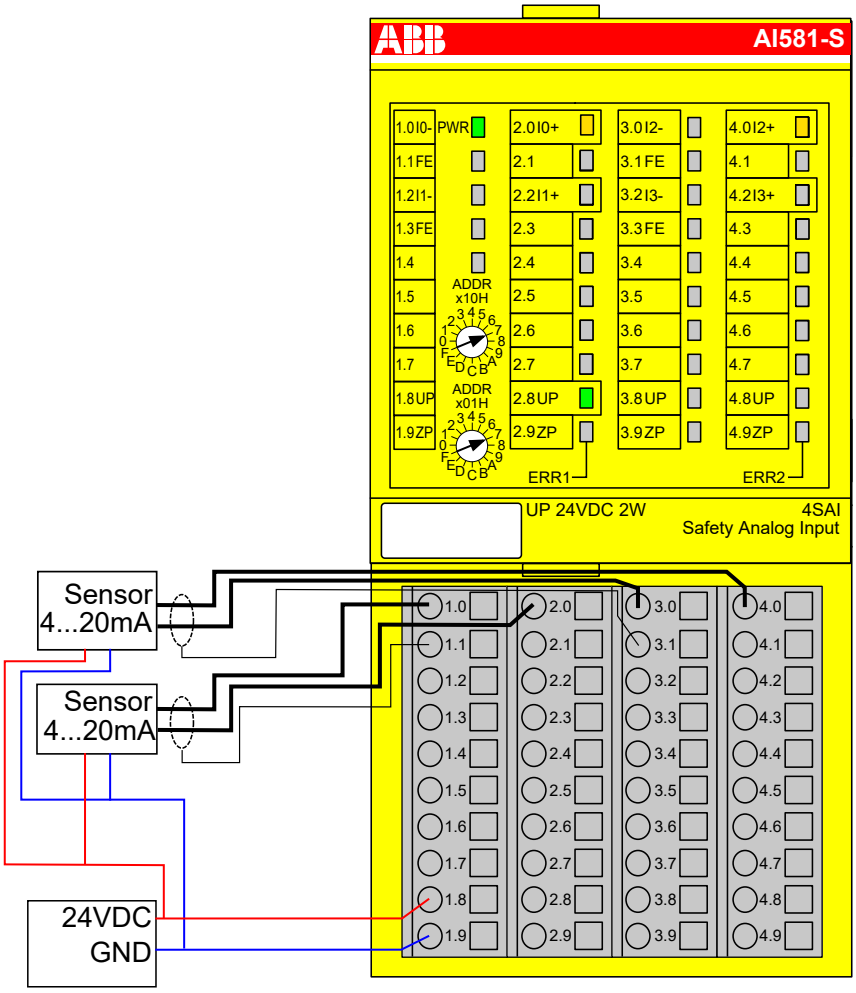


Abb. 60: Beschaltungsbeispiel AI581-S, 2 Analogsensoren (4 … 20 mA)

1) - MTTFd = hoch, DC = hoch

2) - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1)

3) - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

Transmitter (4 ... 20 mA)

Max. SIL / PL [1), 2)]	Max. SIL 2 / PL d
SIL[3)]	SIL 2

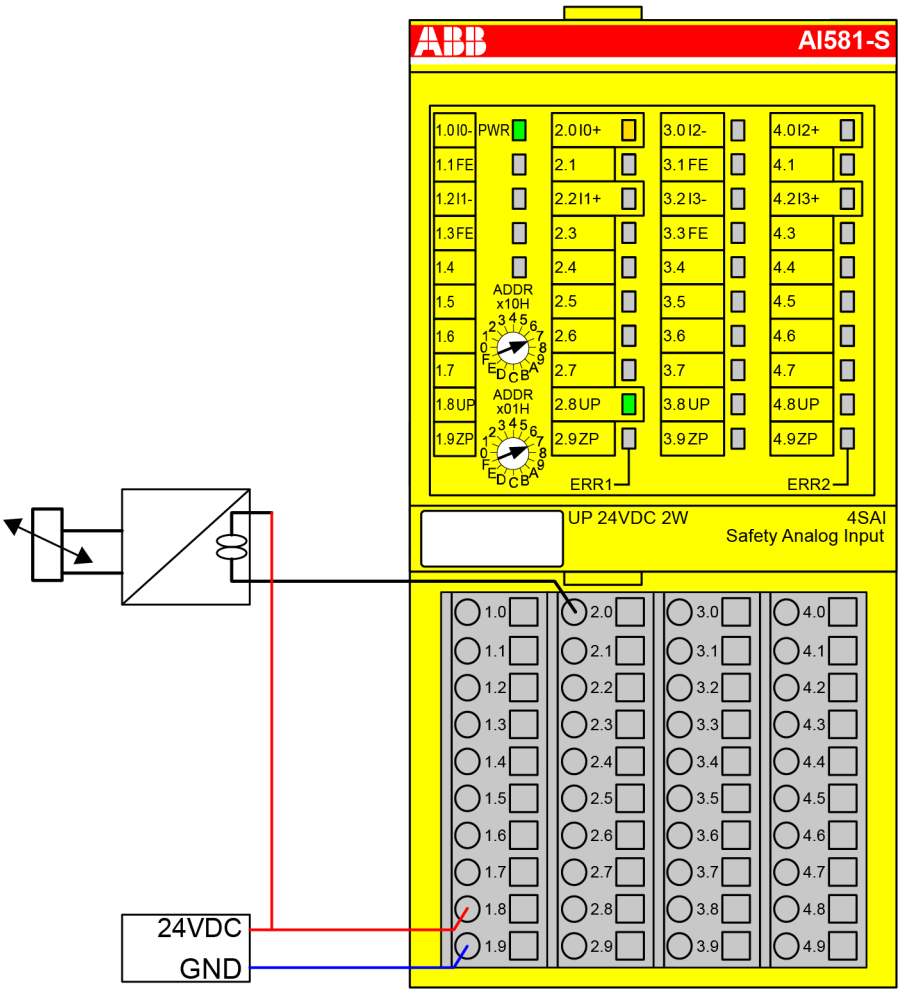


Abb. 61: Beschaltungsbeispiel AI581-S, Transmitter (4 ... 20 mA)

[1)] - MTTFd = hoch, DC = mittel

[2)] - Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu PL e, max. SIL 3 erreicht werden)

[3)] - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich) ➔ ohne Fehlerausschluss (mit Fehlerausschluss können höhere Ebenen bis zu SIL 3 erreicht werden)

2 Transmitter (4 … 20 mA)

2-Kanal-Evaluierung	Im AI581-S-Modul
Max. SIL / PL [1), 2)]	Max. SIL 3 / PL e
SIL[3)]	SIL 3

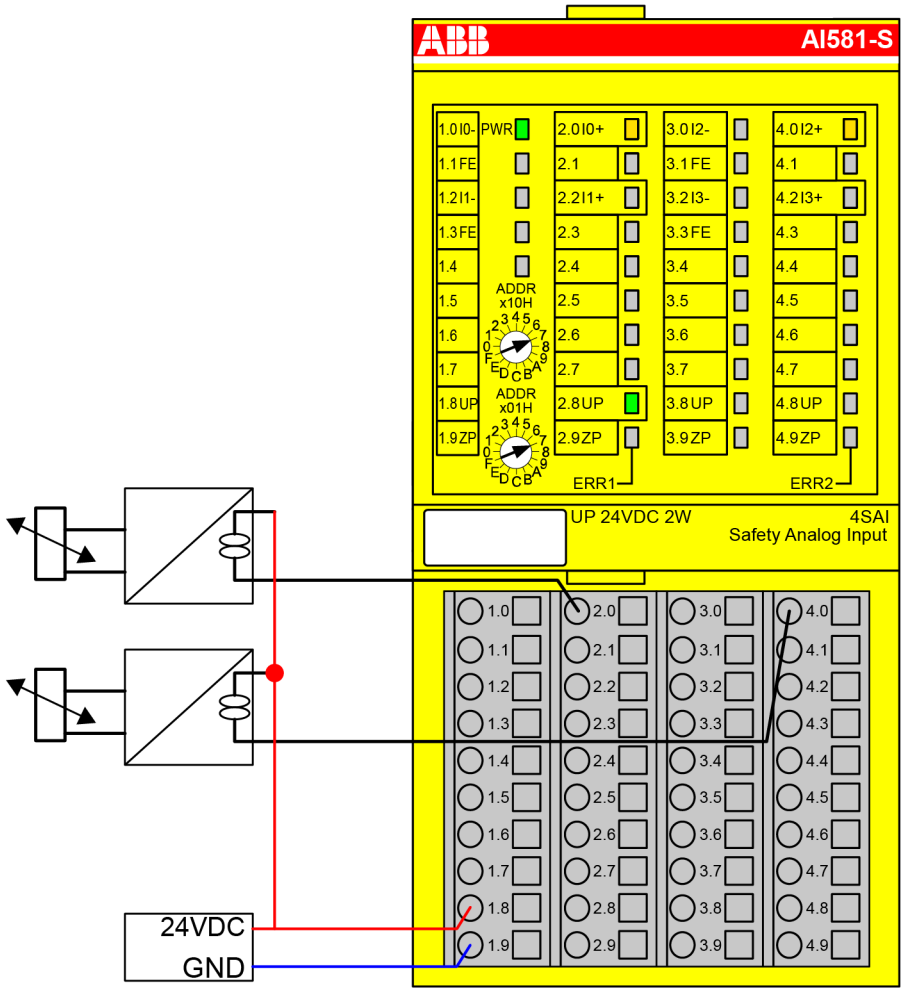


Abb. 62: Beschaltungsbeispiel AI581-S, 2 Transmitter (4 ... 20 mA)

[1)] - MTTFd = hoch, DC = hoch

[2)] - Max. Max. SIL (IEC 62061) / max. erreichbares PL (ISO 13849-1)

[3)] - Max. erreichbares SIL nach IEC 61508 (Komponenten des Typs A sind erforderlich)

3.5.8 LED-Statusanzeige

Tab. 8: Statusanzeige und deren Bedeutung

LED	Beschreibung	Farbe	LED = AUS	LED = EIN	LED blinkt
Eingänge 0 … 3	Analogeingang	Gelb	Analogeingang = ca. 0 mA	Eingang = EIN (Lichtintensität der LED hängt vom Eingabewert ab)	--
	Kanalfehler	Rot	Kein Kanalfehler	Kanalfehler	--
UP	Prozessspannung +24 V DC an Klemme	Grün	Prozess-Versorgungsspannung fehlt	Prozess-Versorgungsspannung ist OK	--

LED	Beschreibung	Farbe	LED = AUS	LED = EIN	LED blinkt
PWR	+3,3 V DC Spannung von I/O-Bus	Grün	+3,3 V DC I/O-Busspannung ist nicht vorhanden	+3,3 V DC I/O-Busspannung ist vorhanden	--
ERR1	Modulfehler-Anzeige 1	Rot	Kein Modulfehler	Modulfehler, der zu einem SAFE STOP führt	Modulpassivierung und/oder Quittieranforderung (abwechselndes Blinken)
ERR2	Modulfehler-Anzeige 2	Rot			

3.5.9 Technische Daten

HINWEIS!
Die Version AI581-S-XC ist für eine Verwendung unter extremen Umgebungsbedingungen erhältlich ↳ *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Weitere technische Daten stehen im SPS-Katalog von ABB zur Verfügung: www.abb.com/plc.

Prozess-Versorgungsspannung UP

Angabe	Wert	Einheit
Anschlussklemmen 1.8 … 4.8 (UP)	+24	V
Anschlussklemmen 1.9 … 4.9 (ZP)	0	V
Nennwert (-15 %, +20 %, ohne Restwelligkeit)	24	V DC
Max. Restwelligkeit	5	%
Verpolschutz	Ja	
Nennwert für Absicherung für UP (schnell)	10	A
Galvanische Trennung	pro Modul	
Verarbeitungsmechanismen von Ein-/Ausgängen	Regelmäßige Aktualisierung	
Wandlungsfehler der Analogwerte durch Nichtlinearitäten, Abgleichfehler im Werk und Auflösung im Nennbereich, üblicherweise	±1	%
Wandlungsfehler der Analogwerte durch Nichtlinearitäten, Abgleichfehler im Werk und Auflösung im Nennbereich, maximal	±1,5	%
Maximale Signalfrequenz	70	Hz
Stromaufnahme über UP im Normalbetrieb mit + 24 V DC (für Modulelektronik)	0,18	A
Einschaltstrom von UP bei 30 V (beim Einschalten)	0,1	A²s
Einschaltstrom von UP bei 24 V (beim Einschalten)	0,06	A²s

Einbaulage

Horizontal oder vertikal mit Leistungsreduzierung (maximale Betriebstemperatur auf +40 °C reduziert)

Kabellänge

Angabe	Wert	Einheit
Leiterquerschnitt von Analogkabeln	> 0,14	mm²

Angabe	Wert	Einheit
Max. Analogkabellänge, geschirmt	100	m

Kühlung

Die natürliche Konvektionskühlung darf nicht durch Kabelkanäle oder andere Einbauten im Schaltschrank behindert werden.

Erlaubte Unterbrechungen der Spannungsversorgung laut EN 61131-2

Angabe	Wert	Einheit
Unterbrechungen der Gleichstromversorgung	< 10	ms
Zeit zwischen 2 Unterbrechungen der Gleichstromversorgung, PS2	> 1	s

Umgebungsbedingungen

Angabe	Wert	Einheit
Betriebstemperatur*	0 … +60	°C
Lagerungstemperatur	-40 … +85	°C
Transporttemperatur	-40 … +85	°C
Luftfeuchtigkeit ohne Kondensation	max. 95	%
Betriebsluftdruck	> 800	hPa
Lagerluftdruck	> 660	hPa
Betriebshöhe	< 2000	m über NN
Lagerhöhe	< 3500	m über NN

* Erweiterte Temperaturbereiche (unter 0 °C und über +60 °C) werden von Sonderversionen von AI581-S unterstützt ↳ *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Kriech- und Luftstrecken

Die Kriech- und Luftstrecken entsprechen der Überspannungskategorie II, Verschmutzungsgrad 2.

Netzteile

Zur Versorgung der Module müssen Netzteile gemäß PELV-/SELV-Spezifikationen verwendet werden.

Elektromagnetische Verträglichkeit

Informationen zur elektromagnetischen Verträglichkeit finden Sie im neuesten TÜV SÜD Report ↳ *[1]*.

Mechanische Eigenschaften

Angabe	Wert	Einheit
Schutzart	IP 20	
Gehäuse	gemäß UL94	
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 3,5 mm	2 … 15	Hz
Vibrationsfestigkeit gemäß EN 61131-2 (alle drei Achsen), kontinuierlich 1 g *	15 … 150	Hz
Stoßprüfung (alle drei Achsen), 11 ms Halbsinus	15	g
MTBF	102	Jahre

* Höhere Werte auf Anfrage

Selbsttest und Diagnosefunktionen Tests während Start und Betrieb: Programmablauf-Überwachung, RAM, CPU, ADC usw.

Abmessungen, Gewicht

Angabe	Wert	Einheit
B × H × T	67,5 × 76 × 62	mm
Gewicht (ohne Klemmenblock)	~ 130	g

Zertifizierungen CE, cUL (weitere Zertifizierungen unter *www.abb.com/plc*)

3.5.9.1 Technische Daten der sicheren Analogeingänge

GEFAHR!
Das Überschreiten der zulässigen Prozess- oder Versorgungsspannung (< -35 V DC bzw. > +35 V DC) kann zu irreparablen Schäden am System führen.

Angabe	Wert	Einheit
Anzahl Kanäle pro Modul	4	
Konfigurierbarkeit, 1-kanaliger Modus	0 … 20	mA
Konfigurierbarkeit, 1-kanaliger Modus	4 … 20	mA
Konfigurierbarkeit, 2-kanaliger Modus	4 … 20	mA
Eingangswiderstand der Kanäle im aktiven Modus	~ 125	Ω
Eingangswiderstand der Kanäle im inaktiven Modus	~ 15	kΩ

Kanalaufteilung in Gruppen 2 Gruppen zu je 2 Kanälen.

Angabe	Wert	Einheit
Zeitkonstante des Eingangsfilters	1	ms
Zyklusdauer für Wandlung	0,33	ms
Auflösung	14	Bits
Temperaturkoeffizient ± % des Skalenendwertes (0 … 20 mA)	± 0,005	%/K
Max. Fehler bei +25 °C ± % des Skalenendwertes (0 … 20 mA)	± 0,25	%
Max. Fehler über den gesamten Temperaturbereich ± % des Skalenendwertes (0 … 20 mA)	± 0,25	%
Wert des niederwertigsten Bits (LSB = least significant bit)	2,03	µA
Maximal zulässige permanente Überlast (keine Schäden) (Selbstschutz), Spannung	32	V DC
Maximal zulässige permanente Überlast (keine Schäden) (Selbstschutz), Strom	24	mA
Nichtlinearität (des Skalenendwertes)	± 0,05	%
Abtastwiederholungszeit	3,3	ms

Angabe	Wert	Einheit
Eingangsfiltereigenschaften – erste Ordnung, Filterzeitkonstante	1	ms
Übergangsfrequenz	160	Hz
Überspannungsschutz	Ja	

Galvanische Trennung

Gegen interne Versorgung und andere Module.

Anzeige Eingangssignal

Eine LED pro Kanal.

Max. temporäre Abweichung während elektrischer Interferenztests in ± % des Skalenendwertes

Angabe	Wert	Einheit
Abweichung während Störstrahlung und leitungsgeführter Störung	< 0,1	%
Abweichung während Burst-Test	max. 0,33	%
Abweichung während Surge-Test	bis zu 50	%
Abweichung während elektrostatischen Entladungen	keine Abweichung	

Analogeingangsschutz

Angabe	Wert
Schutzart des Analogeingangs	Suppressordiode

Kabellänge

Angabe	Wert	Einheit
Max. Kabellänge, geschirmt	100	m

3.5.10 Bestelldaten

Typ	Beschreibung	Bestellnummer
AI581-S	Analoges Sicherheits-Eingabemodul 4SAI	1SAP 282 000 R0001
AI581-S-XC	Analoges Sicherheits-Eingabemodul 4SAI, extreme Umgebungsbedingungen	1SAP 482 000 R0001

3.6 Sicherheits-E/A-Klemmenblock TU582-S

Elemente des Moduls

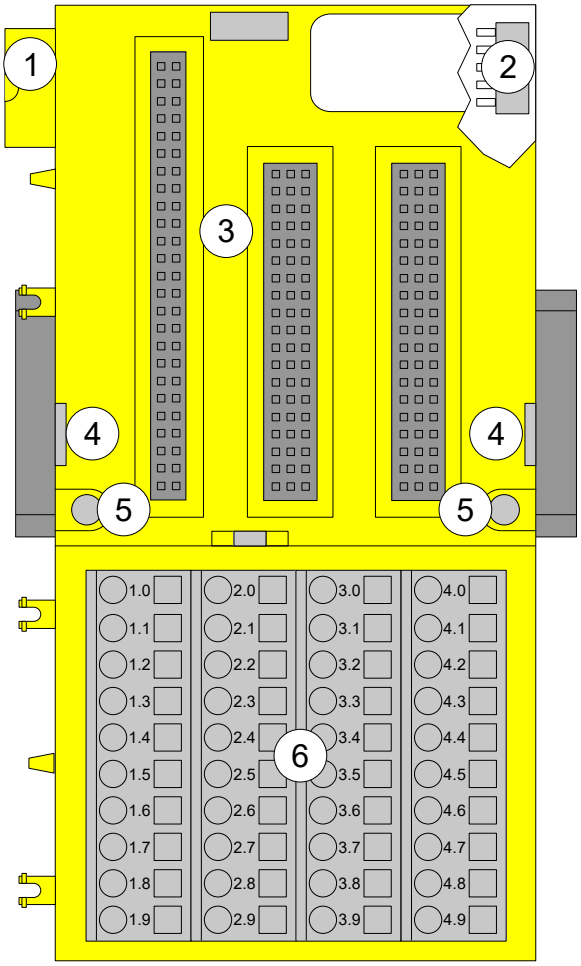


Abb. 63: Sicherheits-E/A-Klemmenblock mit Federzugklemmen TU582-S für Sicherheits-E/A-Erweiterungsmodule

1 I/O-Bus (10-polig, Stecker)
2 I/O-Bus (10-polig, Buchse)
3 Steckplatz für E/A-Modul
4 Durch Einführen eines Schraubendrehers an dieser Stelle lassen sich nebeneinander montierte Klemmenblöcke auseinanderschieben.
5 Bohrungen für Wandmontage
6 40 Anschlussklemmen in Federzugtechnik (Signale und Prozessspannung)

3.6.1 Funktionalität

Die E/A-Klemmenblöcke TU582-S (mit Federzugklemmen) sind speziell für die Verwendung mit den Sicherheits-E/A-Modulen AI581-S, DI581-S und DX581-S der Serie AC500-S geeignet.

Die Sicherheits-E/A-Module werden am E/A-Klemmenblock eingesteckt. Bei korrektem Sitz der Module werden diese durch zwei mechanische Verriegelungen gesichert. Alle elektrischen Anschlüsse werden am Klemmenblock vorgenommen, was einen einfachen Ausbau bzw. Austausch der E/A-Module ermöglicht, ohne dass die Verdrahtung am Klemmenblock gelöst und neu angeschlossen werden muss.

Die Klemmen 1.8 bis 4.8 und 1.9 bis 4.9 sind im Inneren des E/A-Klemmenblocks jeweils elektrisch miteinander verbunden und haben unabhängig vom eingesetzten Modul immer dieselbe Belegung:

- Klemmen 1.8 bis 4.8: Prozessspannung UP = +24 V DC
- Klemmen 1.9 bis 4.9: Prozessspannung ZP = 0 V

Die Belegung der anderen Klemmen ist abhängig vom eingesetzten Sicherheits-E/A-Modul ↳ *DI581-S* ↳ *DX581-S* ↳ *AI581-S*.

3.6.2 Montage, Abmessungen und elektrischer Anschluss

Die Sicherheits-E/A-Module können nur in den E/A-Klemmenblock mit Federzugklemmen TU582-S eingesteckt werden. Die eindeutige mechanische Codierung auf den E/A-Klemmenblöcken verhindert eventuelle Fehler, sodass keine Standard-E/A-Module in den Sicherheits-E/A-Klemmenblock eingesteckt werden können und umgekehrt. Hier werden grundlegende Informationen zur Montage des Systems angezeigt. Ausführliche Informationen finden Sie unter ↳ *[3]*.

Installation und Wartung dürfen nur von Elektro-Fachkräften nach den technischen Regeln, Richtlinien und einschlägigen Normen, z. B. EN 60204 Teil 1, vorgenommen werden.

Montage von TU582-S auf Hutschiene

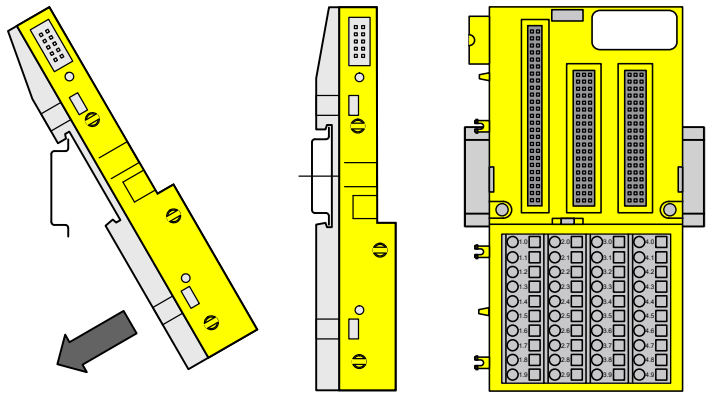
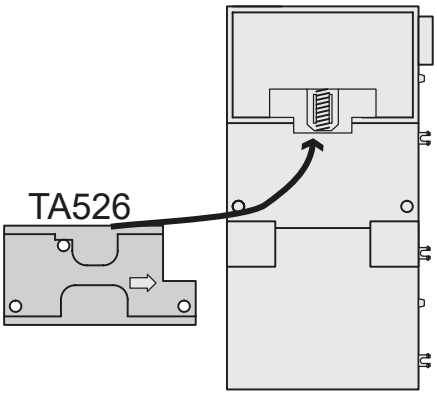


Abb. 64: Montageanweisung zur Montage auf einer Hutschiene

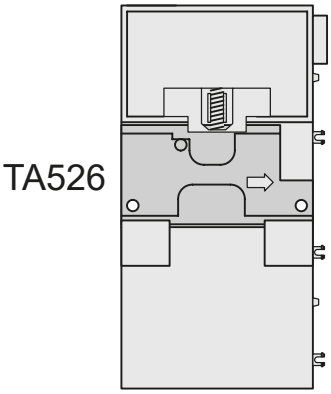
▷ Stecken Sie den Klemmenblock von oben in die Hutschiene und lassen Sie ihn unten einrasten.

Montage von TU582-S mit Schrauben

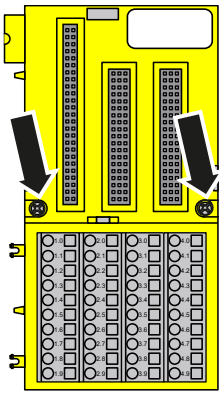


Das Anbringen der Zubehörteile TA526 für Wandmontage ist erforderlich.

1. Lassen Sie die TA526 auf der Rückseite des Klemmenblocks wie bei Hutschienen einrasten.

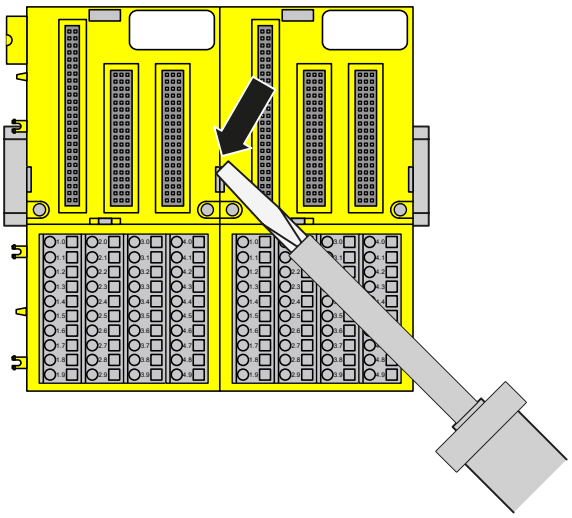


2. Befestigen Sie den Klemmenblock mit 2 M4-Schrauben (max. 1,2 Nm).

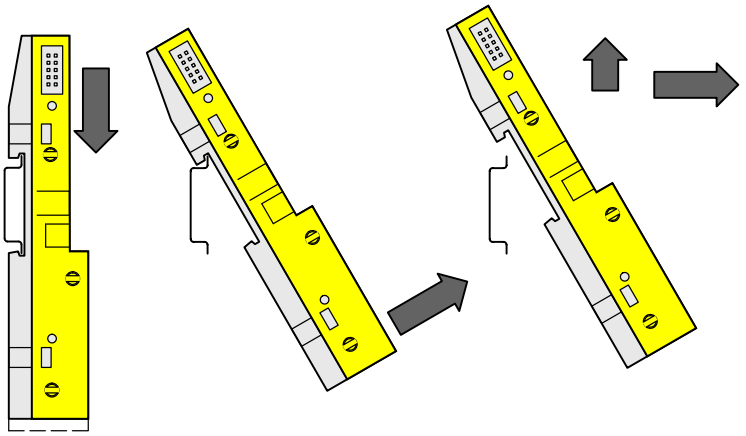


Demontage von TU582-S

1. Schieben Sie die Klemmenblöcke auseinander.



2. Ziehen Sie den Klemmenblock herunter und entnehmen Sie ihn.



Abmessungen

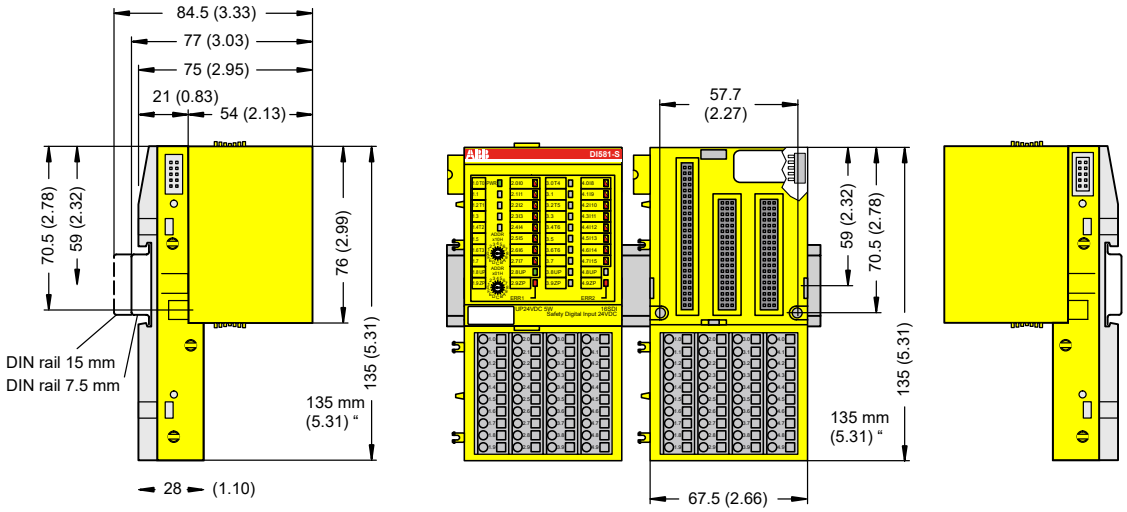


Abb. 65: Abmessungen des Sicherheits-E/A-Klemmenblocks TU582-S

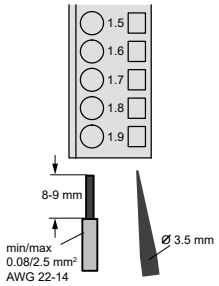


Abb. 66: Anschlussklemmen in Federzugtechnik (Öffnung mit Schraubendreher)

3.6.3 Technische Daten

HINWEIS!
Die Version TU582-S-XC ist für eine Verwendung unter extremen Umgebungsbedingungen erhältlich *Anhang A „Systemdaten für AC500-S-XC“ auf Seite 409*.

Weitere technische Daten stehen im SPS-Katalog von ABB zur Verfügung: *www.abb.com/plc*.

Typ

Anschluss von vorne, Leiteranschluss erfolgt senkrecht zur Platine.

Angabe	Wert	Einheit
Anzahl Kanäle pro Modul	32	
Nennspannung	24	V DC
Max. zulässiger Gesamtstrom (zwischen den Klemmen 1.8 … 4.8 und 1.9 … 4.9)	10	A

Kanalaufteilung in Gruppen

4 Gruppen zu je 8 Kanälen (1.0 … 1.7, 2.0 … 2.7, 3.0 … 3.7, 4.0 … 4.7). Die Zuweisung der Kanäle wird durch das eingesetzte E/A-Erweiterungsmodul vorgegeben.

Einbaulage

Horizontal oder vertikal

Erdung

Durch direkte Verbindung mit der geerdeten Hutschiene oder bei Wandmontage über die Schrauben.

Leiter

Angabe	Wert	Einheit
Leiterquerschnitt, starr	0,08 … 2,5	mm²
Leiterquerschnitt, flexibel	0,08 … 2,5	mm²
Leiterquerschnitt, mit Aderendhülse	0,25 … 1,5	mm²
Abisolierte Länge, Minimum	5	mm
Abisolierte Länge	7	mm

Mechanische Eigenschaften

Angabe	Wert	Einheit
Schutzart	IP 20	
MTBF	2757	Jahre
Gewicht	~ 200	g

3.6.4 Bestelldaten

Typ	Beschreibung	Bestellnummer
TU582-S	Sicherheits-E/A-Klemmenblock, 24 V DC	1SAP 281 200 R0001
TU582-S-XC	Sicherheits-E/A-Klemmenblock, 24 V DC, extreme Umgebungsbedingungen	1SAP 481 200 R0001

4 Konfiguration und Programmierung

4.1 Übersicht

4.1.1 Automation Builder

Die Engineering Suite Automation Builder ist eine Plattform für Konfiguration und Programmierung von IEC-61131-bezogenen Anwendungen.

Für das Konfigurieren und Programmieren von Sicherheitsanwendungen müssen Sie Automation Builder mit installiertem und lizenziertem Safety Engineering mit seinen Sicherheitskomponenten (AC500-S Programming Tool und Safety Configurator) nutzen.

Das Sicherheitskonzept für Sicherheitskomponenten in der Automation Builder-Software stellt sicher, dass das Programmiersystem für die Implementierung der Sicherheitsfunktionen in AC500-S korrekt funktioniert, d. h. dass Fehler des Programmiersystems erkannt werden können. Die Kommunikation zwischen AC500-S Programming Tool und der Sicherheits-CPU ist nicht Teil des Sicherheitskreises. Überprüfungen werden jedoch durchgeführt, z. B. wird eine CRC-Prüfung während des Downloads eines Projekts durchgeführt, um zu überprüfen, ob die übertragenen Daten korrekt sind und kein Kommunikationsfehler vorliegt. Anwender müssen zudem die Version und Funktionalität ihrer Projekte und die richtige Konfiguration der Sicherheits- und Standardmodule überprüfen.

Die Automation Builder Sicherheitskomponenten ermöglichen das Erstellen von Sicherheitsanwendungen bis zum Sicherheitsintegritätslevel SIL 3 (IEC 61508, IEC 62061 und IEC 61511) / PL e (ISO 13849).

Die Kompatibilität der Automation Builder Version ist von den verwendeten Sicherheits- und Standard-CPUs abhängig ↳ *Anhang B.1 „Kompatibilität mit AC500 V2-Standard-CPU“ auf Seite 415* ↳ *Anhang C.1 „Kompatibilität mit AC500 V3-Standard-CPU“ auf Seite 437*.

4.1.2 Safety Engineering

Sie können leicht prüfen, welche Safety Engineering Version und welche ihrer Sicherheitskomponenten bei Ihnen installiert und lizenziert sind. Diese Funktion ist ab Automation Builder 2.3.0 verfügbar.

☑ Automation Builder ist offen.

▷ Wechseln Sie in das Menü *„„Hilfe ➔ Informationen ➔ Safety Version Information“*.

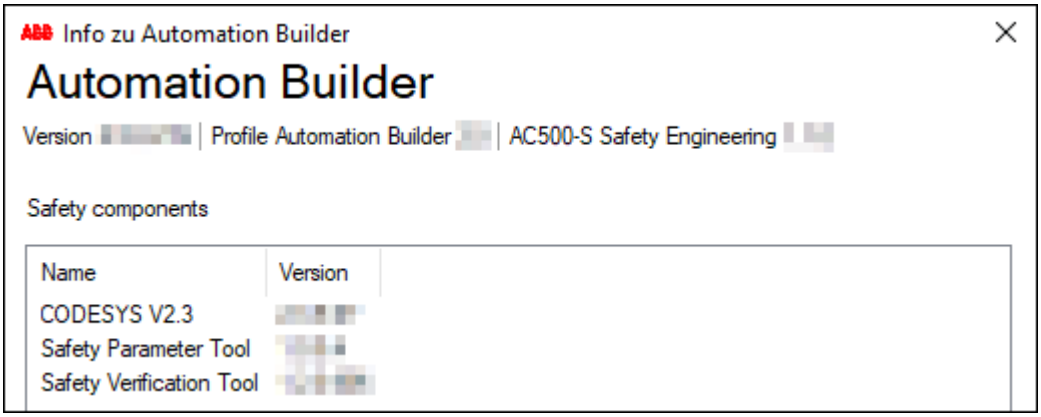


Abb. 67: Informationen zu Safety Engineering und Sicherheitskomponenten

Wenn eine Safety Engineering Version und die Versionen von Sicherheitskomponenten angezeigt werden, ist gewährleistet, dass Sie freigegebene und begutachtete Sicherheitskomponenten nutzen.

Die Sicherheitskomponenten werden unabhängig von Automation Builder Versionen freigegeben. Nach der Installation von Automation Builder 2.3.0 oder höher muss der Anwender die Safety Engineering Version prüfen ↳ *Kapitel 4.2 „Ablauf“ auf Seite 145*.

HINWEIS!
Werden kein Safety Engineering und keine Sicherheitskomponenten angezeigt, wiederholen Sie die Installation von Automation Builder und stellen Sie sicher, dass Sie die entsprechende Lizenz aktiviert haben. Lässt sich der Fehler nicht beseitigen, wenden Sie sich an den technischen Support von ABB.

4.1.3 Sicherheitsmaßnahmen

Eine vollständige Überprüfung von Programmlogik und Konfiguration ist erforderlich, um zu prüfen, ob die Logik die funktionellen und sicherheitsbezogenen Anforderungen Ihrer Sicherheitsanwendung korrekt und vollständig erfüllt. Die Projektdaten müssen nach jeder Änderung erneut überprüft werden.

GEFAHR!
Beim ersten Start der Sicherheits-CPU und nach einer Änderung des Anwendungsprogramms oder der Konfiguration muss die Sicherheit des gesamten Systems mit einem kompletten Funktionstest überprüft werden. Dieser umfasst auch eine Überprüfung der korrekten Kodierung der Sicherheitsanwendung basierend auf den Funktionsspezifikationen.

4.1.4 Schutz vor ungewollten Änderungen

In die Sicherheits-CPU und im Automation Builder sind Sicherheitsfeatures als Schutzmechanismen eingebaut, die die ungewollte oder unautorisierte Änderung des Sicherheitssystems verhindern:

- Eine Änderung des Sicherheitsprogramms generiert eine neue Bootprojekt-CRC-Versionsnummer.
- Anwender müssen sich in der Sicherheits-CPU einloggen, um Zugriff auf die Bedienfunktionen zu haben.
- Sicherheitsanforderungen und andere relevante Anwendungsnormen zum Schutz vor Manipulationen müssen beachtet werden. Die Autorisierung der Mitarbeiter und die erforderlichen Schutzmaßnahmen liegen in der Verantwortung des Betreibers.

Ein unautorisierter Zugriff auf die Sicherheits-CPU und das Sicherheitsprogramm kann durch mehrere Passwörter verhindert werden ↬ *Kapitel 4.3.3 „Anlegen eines neuen Projekts und Benutzerverwaltung“ auf Seite 146*.

4.2 Ablauf

Der in diesem Kapitel vorgestellte Entwicklungsablauf beschreibt nur die Schritte, die zum Instanziieren, Konfigurieren und Programmieren der Sicherheitsmodule und der Standardmodule, die Teil des „Black Channel“ ↬ *[2]* für sichere Kommunikation sind, erforderlich sind. Alle anderen Standardmodule werden separat behandelt in ↬ *[3]*. Nähere Informationen zu diesen Schritten finden Sie unter ↬ *Kapitel 4.3 „Konfiguration und Programmierung des Systems“ auf Seite 146*.

Ablauf für Konfiguration und Programmierung des AC500-S-Systems

1. Automation Builder installieren (siehe Installationsanweisungen).
2. Lizenz aktivieren.
3. Ab Automation Builder 2.3.0: Prüfen, dass Safety Engineering und die Sicherheitskomponenten verfügbar sind ↬ *Kapitel 4.1.2 „Safety Engineering“ auf Seite 144*.

4. Neues Projekt anlegen und Benutzerverwaltung konfigurieren, um Zugriff auf Sicherheitsmodule und ihre Konfiguration auf Sicherheitspersonal zu beschränken.
5. GSDML-Dateien für die Konfiguration von PROFIsafe F-Devices von Drittanbietern installieren (optionaler Schritt).
6. Sicherheitsmodule und Standardmodule instanziieren und konfigurieren. Variablennamen entsprechend den Richtlinien zur Sicherheitsprogrammierung definieren *Kapitel 4.3.5 „Instanziierung und Konfiguration von Sicherheitsmodulen / Definition von Variablennamen“ auf Seite 150*.
7. Programm für die Sicherheitsanwendung schreiben und Startvorgang des Systems beachten.
8. Programm und Systemkonfiguration überprüfen. Das SCA-Tool für die statische Codeanalyse Ihres Programms nutzen *Kapitel 4.5 „Sicherheitscodeanalyse-Tool“ auf Seite 206*. Entsprechend den Anweisungen zur Überprüfung Ihrer Konfiguration vorgehen *Kapitel 4.3.7 „Überprüfen von Programm- und Systemkonfiguration“ auf Seite 180*.

4.3 Konfiguration und Programmierung des Systems

In diesem Kapitel wird Schritt für Schritt erklärt, wie die AC500-S-Sicherheitssteuerung konfiguriert und programmiert wird.

4.3.1 Installation

▷ Automation Builder installieren (siehe zugehörige Installationsanweisungen).

4.3.2 Lizenzaktivierung

Automation Builder 2.0.2 (oder höher)

1. Bestellen Sie das Add-On DM220-FSE oder DM221-FSE-NW über die Bestellnummern 1SAS010020R0102 und 1SAS010021R0102.
2. Aktivieren Sie die Lizenz auf Ihrem PC entsprechend der Anweisungen zur Lizenzaktivierung.

Automation Builder bis 1.2.4

1. Bestellen Sie eine Lizenz PS501-S über die Bestellnummer 1SAP198000R0001.
2. Aktivieren Sie die Lizenz auf Ihrem PC entsprechend der Anweisungen zur Lizenzaktivierung.

4.3.3 Anlegen eines neuen Projekts und Benutzerverwaltung

Neues Projekt anlegen und Benutzerverwaltung konfigurieren, um den Zugriff auf Sicherheitsmodule und ihre Konfiguration ausschließlich für Sicherheitspersonal zuzulassen.

1. Verwenden Sie den Menüpunkt *„Neues Projekt …“* im Automation Builder, um ein neues Projekt anzulegen.
2. Wählen Sie im Menü eine AC500-Standard-CPU aus. Stellen Sie sicher, dass Sie die richtige Standard-CPU auswählen, die Sicherheits-CPUs unterstützt ↳ *Anhang B.1 „Kompatibilität mit AC500 V2-Standard-CPU“ auf Seite 415* ↳ *Anhang C.1 „Kompatibilität mit AC500 V3-Standard-CPU“ auf Seite 437*.

> **HINWEIS!**
> Beachten Sie die Einstellungen der Standard-CPU ↳ *Anhang B.3 „Konfiguration der AC500 V2-Standard-CPU-Parameter“ auf Seite 427* ↳ *Anhang C.3 „Konfiguration der AC500 V3-Standard-CPU-Parameter“ auf Seite 445*.

3. Um neue Benutzer anzulegen und vorhandene Benutzer zu verwalten, wählen Sie *„Projekt ➔ Projekteinstellungen …“*.

> **HINWEIS!**
> In allen neuen Projekten im Automation Builder gibt es den Standardbenutzer „Owner“ mit einem leeren Passwort. Dies ist der Projektadministrator. Der Projektadministrator ist dafür verantwortlich, ein neues Passwort für „Owner“ festzulegen und zusätzlich je nach Projektorganisation und Bedarf dedizierte Sicherheits- und Standardbenutzer anzulegen.

Nur Mitglieder der Sicherheitsgruppe dürfen Sicherheitsmodule bearbeiten, ihre Konfiguration ändern usw. Benutzer ohne die entsprechenden Anmeldedaten und Zugriffsrechte können standardmäßig nicht auf Sicherheitsmodule zugreifen.

Der Zugriff auf die Sicherheits-CPU und das Sicherheitsprogramm kann durch drei Passwörter geschützt werden.

- Passwort für die Sicherheits-CPU
- Passwort für das Sicherheitsprogramm in AC500-S Programming Tool
  - max. 200 Zeichen
  - zulässige Zeichen: (A-Z) (a-z) (0-9) Ä Ö Ü ä ö ü ß # § % ° ^ + - & _ ! @ ´ ~ * | ( ) { } [ ] , ; . : <> = / ' ?
- Passwort für Sicherheitsmodule und ihre Konfigurationsdaten im Automation Builder mit Sicherheitsfeatures

Der Projektadministrator darf alle verfügbaren Optionen für die Benutzerverwaltung nutzen, um die bestmöglichen Benutzereinstellungen mit den entsprechenden Rechten zu finden ↳ *[3]*.

> **GEFAHR!**
> Der Projektadministrator ist dafür verantwortlich, für ein entsprechendes Sicherheitsprogramm eine geeignete Benutzerverwaltung aufzustellen, um unautorisierten Zugriff auf Sicherheitsmodule zu verhindern.
>
> Passwörter für Benutzer der Sicherheitsgruppe sollten selbst ausgewählt werden (mindestens 8 Zeichen und eine Kombination aus Zahlen und Buchstaben werden empfohlen). Der Zugang zu Passwörtern muss streng kontrolliert werden.
>
> Stellen Sie sicher, dass Sie die *„Einschränkung“* von Rechten für Benutzer und Gruppen (z. B alle) korrekt im Menü *„Projekt ➔ Benutzerverwaltung ➔ Zugriffsrechte ….“* eingestellt haben, um ein nicht autorisiertes Erstellen neuer Benutzer in der Sicherheitsgruppe zu vermeiden.

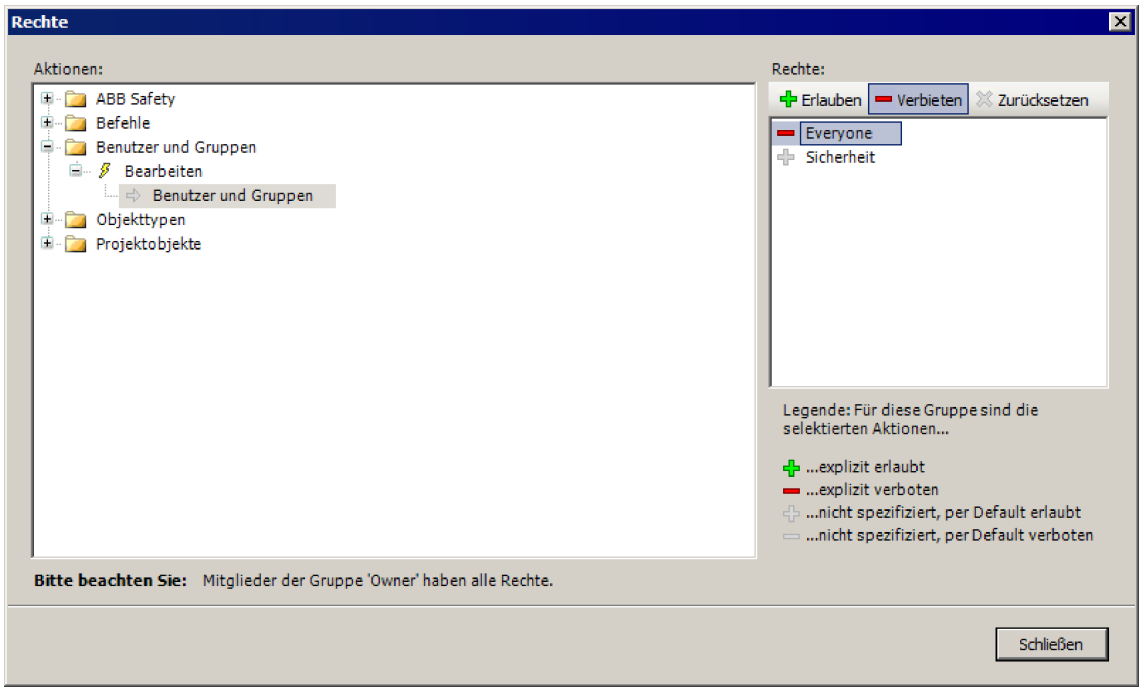


Abb. 68: Zugriffsrechte für Benutzer und Benutzergruppen

4.3.4 Arbeit mit PROFINET/PROFIsafe F-Devices

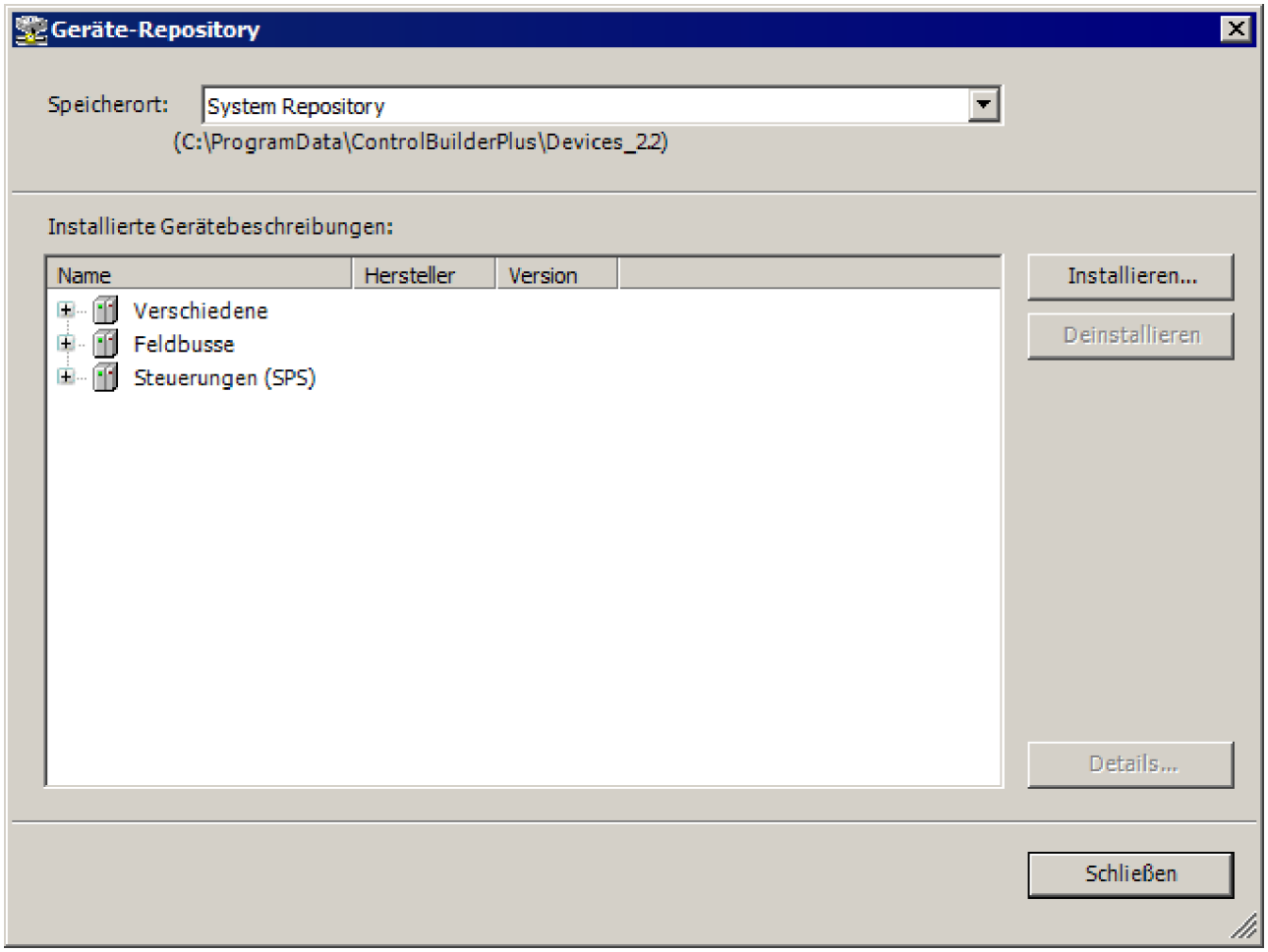
Die Installation von GSDML-Dateien ist erforderlich, um PROFIsafe F-Devices von Drittanbietern konfigurieren zu können.

Um F-Devices von Drittanbietern mit der AC500-S-Sicherheitssteuerung zu nutzen, müssen die sicherheitsgerichteten Geräte auf dem PROFINET-I/O liegen und das PROFIsafe-Busprofil im V2-Modus unterstützen ↳ *[2]*. Die Basis für die Konfiguration von allen PROFINET-Geräten (Sicherheit und Standard) ist die Spezifikation des Gerätes in der GSDML-Datei (Generic Station Description Markup Language).

Die Eigenschaften der E/A-Geräte werden in der GSDML-Datei gespeichert. Für PROFINET/PROFIsafe-Geräte sind Teile der GSDML-Dateidaten durch CRC geschützt ↳ *[2]*. GSDML-Dateien werden von den Geräteherstellern geliefert.

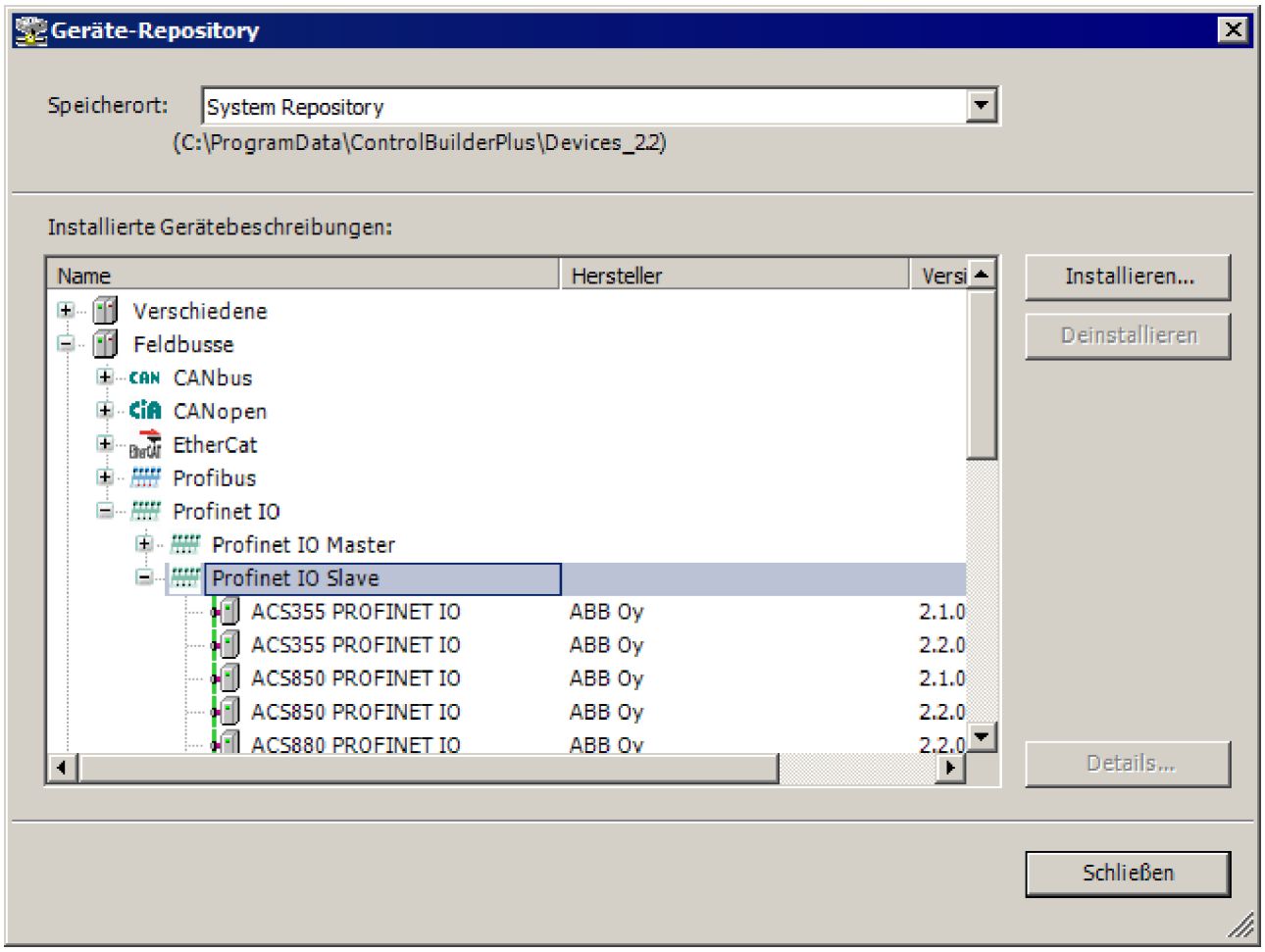
HINWEIS!
Bitte kontaktieren Sie den technischen Support von ABB zu unterstützten GSDML-Dateiversionen. Sie hängt von der installierten Version des Automation Builder ab.

1. Zum Installieren der GSDML-Datei öffnen Sie das Menü *„Werkzeuge ➔ Gerätepool…“*.



2. Klicken Sie auf die Schaltfläche *[Installieren…]*, um eine GSDML-Datei für die Installation auszuwählen.

   ⇨ Nach einer erfolgreichen Installation werden neue Geräte im *„Gerätepool“* unter dem Objekt *„Profinet IO“* angezeigt.



4.3.5 Instanziierung und Konfiguration von Sicherheitsmodulen / Definition von Variablennamen

Sicherheits- und Standardmodule, die Teil des „Black Channel“ für sichere Kommunikation sind, instanziieren und konfigurieren. Variablennamen für Eingangs-, Ausgangs- und PROFIsafe-Signale entsprechend den Richtlinien zur Sicherheitsprogrammierung definieren.

1. Wählen Sie einen der vier für Kommunikationsmodule und Sicherheits-CPU verfügbaren Steckplätze und instanziieren Sie die Sicherheits-CPU. Beachten Sie, dass die Steckplatznummer mit dem Steckplatz, in den die Sicherheits-CPU eingesteckt ist, übereinstimmen muss.
2. Doppelklicken Sie auf die Sicherheits-CPU und setzen Sie die zugehörigen Parameter nach Bedarf ↳ *Kapitel 3.1.7 „Parametrierung“ auf Seite 55*.

> **HINWEIS!**
> Beachten Sie den Parameter *„Debug-Modus aktivieren“*. Steht dieser Parameter auf *„Aus“*, kann kein neues Bootprojekt in die Sicherheits-CPU geladen werden.

3. Um dezentrale Einheiten im System zu integrieren, kann das Kommunikationsmodul CM579-PNIO des PROFINET IO-Controllers beispielsweise in Slot 2 instanziiert werden. Bitte beachten Sie, dass PROFINET das einzige von der PROFIsafe-Kommunikation der Sicherheitssteuerung AC500-S unterstützte Bussystem ist.
4. Wählen Sie nun das neu erstellte Modul CM579-PNIO aus und instanziieren Sie die erforderliche Anzahl von PROFINET-Modulen, z. B. CI501-PNIO, CI502-PNIO, usw. oder von anderen PROFINET-Modulen von Drittanbietern, die zuvor mithilfe von GSDML-Dateien in den *„Gerätepool“* importiert wurden.

   Informationen zur korrekten Einstellung von PROFINET-Gerätenamen und IP-Adressen finden Sie unter ↳ *[3]*.
5. Am Objekt *„IO_Bus“* können bis zu 10 E/A-Module (Sicherheits- oder Standardmodule) instanziiert werden, die sich zentral auf der Standard-CPU befinden.
6. Gleichermaßen können bis zu 10 E/A-Module (Sicherheits- und Standardmodule) auf jedem PROFINET IO-Device von ABB instanziiert werden.

In der GSDML-Datei wird die maximale Anzahl unterstützter Module auf PROFINET IO-Devices von Drittanbietern definiert.

Die Parameter der Sicherheits-E/A-Module können nach Doppelklick auf das jeweilige Modul eingestellt werden. Für jedes Modul gibt es zwei verschiedene Parameter: F-Parameter und iParameter.

F-Parameter sind Parameter, die speziell von der PROFIsafe-Gruppe ↳ *[2]* definiert wurden, um eine sichere Geräte-Kommunikation und Parametrierung zu garantieren. Die Namen der F-Parameter sind für alle F-Devices (ABB und Drittanbieter) dieselben. Die wichtigsten F-Parameter für Endanwender werden hier erläutert.

F_SIL - definiert das höchste nutzbare Sicherheits-Integritätslevel für das betreffende F-Device. Es darf über dem in der GSDLM-Datei des F-Device definierten Wert liegen.

F_Dest_Add - definiert die F-Device-Adresse; diese muss dieselbe Adresse wie die am physischen Sicherheits-E/A-Modul eingestellte sein.

HINWEIS!

Stellen Sie sicher, dass F_Dest_Add für alle F-Devices eindeutig ist; anderenfalls kann keine Sicherheitskonfiguration generiert werden.

Zur Einstellung von F_Dest_Add im Automation Builder können Dezimal- oder Hexadezimal-Zahlen mit dem Präfix 16# oder 0x verwendet werden.

F_Source_Add - definiert die F-Host-Adresse, die für das betreffende F-Device gültig ist.

F_WD_Time - definiert die Watchdog-Zeitüberschreitung an der F-Device-Verbindung. Der Parameter wird sowohl am F-Host als auch am F-Device überwacht. Wenn der F-Host eine Zeitüberschreitung erkennt, wird das F-Device passiviert, und es werden Failsafe-Wert gesendet. Wenn das F-Device eine Zeitüberschreitung erkennt, signalisiert es dies über PROFIsafe Status-Byte und sendet Failsafe-Werte. F_WD_Time wird außerdem in Berechnungen der Antwortzeit der Sicherheitsfunktion verwendet *Kapitel 5.3 „Antwortzeit der Sicherheitsfunktion (= Safety Function Response Time)“ auf Seite 363*.

F_CRC_Seed - definiert die unterstützte PROFIsafe Protokollversion. Wenn der Parameter F_CRC_Seed nicht existiert oder in der GSDML F_CRC_Seed = 0 ist (standardmäßiger, symbolischer Wert „CRC_Seed16“), wird die PROFIsafe Protokollversion V2.4 vom F-Device unterstützt und die mit PROFIsafe Protokollversion V2.6 eingeführten Verbesserungen (z. B. Nutzung langer Frames) werden nicht unterstützt. Dadurch ist sichergestellt, dass alle existierenden F-Devices (vor der Freigabe der PROFIsafe Protokollversion V2.6) weiter entsprechend PROFIsafe Protokollversion V2.4 identifiziert werden. F_CRC_Seed = 1 (symbolischer Wert „CRC_Seed24/32“) gibt an, dass PROFIsafe Protokollversion V2.6 unterstützt wird. Der Parameter ist nicht änderbar.

F_Passivation - existiert nur, wenn PROFIsafe Protokollversion V2.6 unterstützt wird (F_CRC_Seed = 1). Wenn F_Passivation = 1 (symbolischer Wert „Channel“), wird Unterstützung des RIOforFA Profils für das betreffende F-Device angefordert, wie in *[12]* spezifiziert. Wenn F-Passivation = 0 (symbolischer Wert „Device/Modul“) oder in der GSDML nicht existiert, wird dieses Profil nicht unterstützt. Der Parameter ist nicht änderbar.

F_WD_Time_2 - ist ein optionaler zweiter Parameter für eine Watchdog-Zeitüberschreitung, sofern vom F-Device unterstützt. Auf die AC500-S Sicherheits-CPU hat der betreffende Wert (sofern vorhanden) keine Auswirkungen, da die Sicherheits-CPU nicht über Verfahren verfügt, die diese zweite Watchdog-Zeit erfordern. Halten Sie diesen Parameter, sofern er im F-Parameter-Konfigurator vorhanden ist, auf seinem Standardwert.

HINWEIS!

Die Sicherheits-E/A-Module (AI581-S, DI581-S und DX581-S) und die F-Submodule „12 Byte In/Out (PROFIsafe V2.4)“ und „8 Byte and 2 Int In/Out (PROFIsafe V2.4)“ im SM560-S-FD-1 / SM560-S-FD-4 unterstützen nur die PROFIsafe Protokollversion V2.4. Die F-Parameter F_CRC_Seed und F_Passivation existieren in der F-Parameter-Konfiguration nicht.

Die F-Submodule „12 Byte In/Out (PROFIsafe V2.6)“ und „123 Byte In/Out (PROFIsafe V2.6)“ in SM560-S-FD-1 / SM560-S-FD-4 sind kompatibel mit PROFIsafe Protokollversion V2.6. F_CRC_Seed („CRC_Seed24/32“) ist in der F-Parameter-Konfiguration angegeben. Die F-Parameter F_Passivation und F_WD_Time_2 sind darauf nicht anwendbar und folglich auch nicht konfigurierbar (F_Passivation = 0 und nicht änderbar, F_WD_Time_2 existiert nicht).

F_iPar_CRC - ist ein spezieller F-Parameter, der für die sichere Übertragung von iParametern zu F-Devices verwendet wird. F_iPar_CRC wird außerhalb des F-Parameter-Editors berechnet und muss deshalb manuell vom Feld *„Prüfsumme iParameter“* in das Feld F_iPar_CRC der Registerkarte „F-Parameter“ kopiert werden, nachdem die Schaltfläche *[Berechnen]* für das entsprechende F-Device angeklickt wurde.

Beachten Sie, dass F_iPar_CRC für Sicherheits-E/A-Module AC500-S auch neu berechnet werden muss, wenn F_Dest_Add verändert wird, weil F_Dest_Add auch unsichtbar als iParameter zu den Sicherheits-E/A-Modulen AC500-S übertragen wird. Der Parameter wird in der AC500-S-Sicherheitssteuerung benötigt, um den physischen PROFIsafe-Adresswert des Sicherheits-E/A-Moduls mit dem in der Entwicklungsumgebung konfigurierten zu vergleichen.

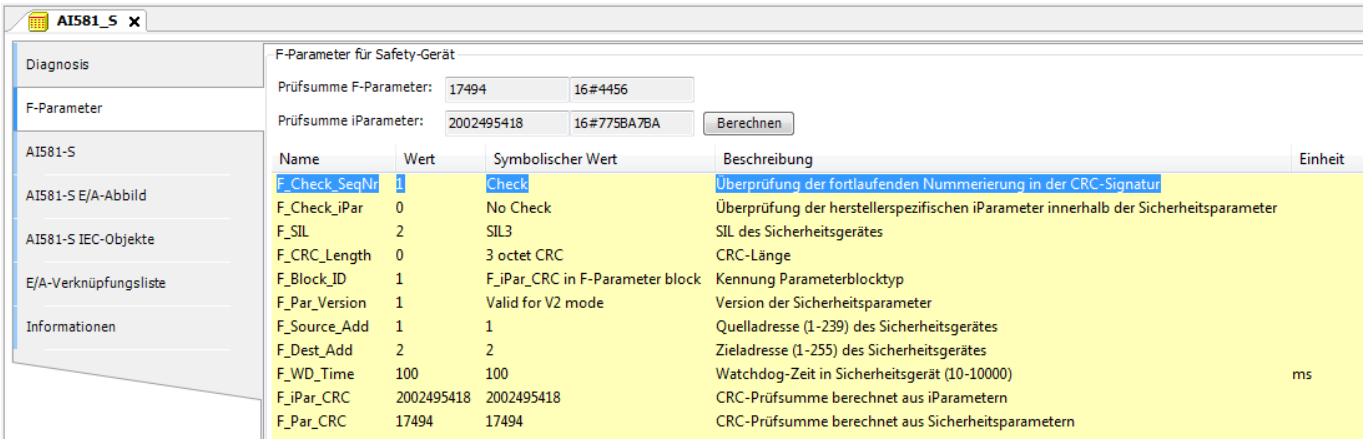


Abb. 69: Beispiel für F-Parameter-Einstellungen

Tab. 9: F-Parameter von AC500-S-Sicherheitsmodulen

F_Parameter	Definition	Zulässige Werte	Standardwert
F_Check_SeqNr	Dieser Parameter definiert, ob die „consecutive number“ in den CRC2 einbezogen werden soll. PROFIsafe V2-Modus ↳ *[2]*: Die „consecutive number“ wird immer in den CRC2 einbezogen. Hinweis: F_Check_SeqNr wird in der F-Parameter-Konfiguration für die SM560-S-FD-1 und SM560-S-FD-4 nicht angezeigt.	„No Check“ = 0 „Check“ = 1	„Check“ = 1
F_Check_iPar	Herstellerspezifische Verwendung mit homogenen Systemen	„No Check“ = 0 „Check“ = 1	„No Check“ = 0
F_SIL	Die verschiedenen Sicherheitsfunktionen, die sicherheitsrelevante Kommunikation verwenden, erfordern verschiedene Sicherheitsintegritätslevel. Die F-Devices können ihren eigenen SIL-Wert mit dem konfigurierten SIL-Wert (F_SIL) vergleichen. Wenn er höher als der SIL-Wert des angeschlossenen F-Device ist, wird das Statusbit „Gerätefehler“ gesetzt und eine Reaktion für den sicheren Zustand wird ausgelöst ↳ *[2]*.	„SIL1“ = 0 „SIL2“ = 1 „SIL3“ = 2 „NoSIL“ = 3	„SIL3“ = 2

F_Parameter	Definition	Zulässige Werte	Standardwert
F_CRC_Length	Abhängig von der Länge der F-E/A-Daten (12 oder 123 Oktette) und des SIL-Levels ist ein CRC von 2, 3 oder 4 Oktetten erforderlich.	„3 octet CRC“ = 0 „4 octet CRC“ = 2 Nicht unterstützt von SM560-S: „2 octet CRC“ = 1	„3 octet CRC“ = 0 für die AC500-S Sicherheits-E/A-Module und die F-Submodule „12 Byte In/Out (PROFIsafe V2.4)“ und „8 Byte and 2 Int In/Out (PROFIsafe V2.4)“ für SM560-S-FD-1 und SM560-S-FD-4. „4 octet CRC“ = 2 für die F-Submodule „12 Byte In/Out (PROFIsafe V2.6)“ und „123 Byte In/Out (PROFIsafe V2.6)“ für SM560-S-FD-1 und SM560-S-FD-4.
F_CRC_Seed	Dieser Parameter wird nur für PROFIsafe Protokollversion V2.6 unterstützt. Wenn F_CRC_Seed = 1, unterstützt das F-Device die PROFIsafe Protokollversion V2.6. Nur die F-Submodule „12 Byte In/Out (PROFIsafe V2.6)“ und „123 Byte In/Out (PROFIsafe V2.6)“ für SM560-S-FD-1 und SM560-S-FD-4 unterstützen die PROFIsafe Protokollversion V2.6.	"CRC_Seed16" = 0 "CRC_Seed24/32" = 1	Nicht sichtbar für die Sicherheits-E/A-Module und die F-Submodule „12 Byte In/Out (PROFIsafe V2.4)“ und „8 Byte and 2 Int In/Out (PROFIsafe V2.4)“ für SM560-S-FD-1 und SM560-S-FD-4. „CRC_Seed24/32“ = 1 für die F-Submodule „12 Byte In/Out (PROFIsafe V2.6)“ und „123 Byte In/Out (PROFIsafe V2.6)“ für SM560-S-FD-1 und SM560-S-FD-4.
F_Passivation	Dieser Parameter wird nur für PROFIsafe Protokollversion V2.6 unterstützt. Er definiert, ob kanalgranulare Passivierung gemäß RIOforFA unterstützt wird oder nicht. Von den Sicherheits-E/A-Modulen und den F-Submodulen für SM560-S-FD-1/SM560-S-FD-4 wird kanalgranulare Passivierung gemäß RIOforFA nicht unterstützt. Alle Sicherheits-E/A-Module unterstützen eigene kanalgranulare Passivierung. F-Submodule für SM560-S-FD-1/SM560-S-FD-4 erfordern keine kanalgranulare Passivierung gemäß RIOforFA.	„Device/Module“ = 0 „Channel“ = 1	Nicht sichtbar für die Sicherheits-E/A-Module und die F-Submodule „12 Byte In/Out (PROFIsafe V2.4)“ und „8 Byte and 2 Int In/Out (PROFIsafe V2.4)“ für SM560-S-FD-1 und SM560-S-FD-4. „Device/Module“ = 0 für die F-Submodule „12 Byte In/Out (PROFIsafe V2.6)“ und „123 Byte In/Out (PROFIsafe V2.6)“ für SM560-S-FD-1 und SM560-S-FD-4.

F_Parameter	Definition	Zulässige Werte	Standardwert
F_Block_ID	Identifizierung der Parameterart	„No F_iPar_CRC within F-Parameter block“ = 0 „F_iPar_CRC within F-Parameter block“ = 1	„F_iPar_CRC within F-Parameter block“ = 1 für Sicherheits-E/As (Sicherheits-E/A-Module der AC500-S können nur mit diesem Standardwert arbeiten) „F_iPar_CRC within F- Parameter block“ = 0 für SM560-S-FD-1 und SM560-S-FD-4
F_Par_Version	Versionsnummer des F-Parameter-Satzes	„Valid for V1-mode“ = 0 „Valid for V2-mode“ = 1	„Valid for V2-mode“ = 1 (Sicherheits-E/A-Module der AC500-S können nur mit diesem Standardwert arbeiten)
F_Source_Add	Ursprungsadresse des F-Host. Der Parameter F_Source_Add ist eine logische Adressbezeichnung, die frei zugewiesen werden kann, aber eindeutig sein muss. **F_Source_Add darf für ein F-Device nicht gleich F_Dest_Add sein.**	[1 – 511] für SM560-S-FD-1 und SM560-S-FD-4 [1 – 239] für Sicherheits-E/A-Module AC500-S. [1 – 65534] für PROFIsafe F-Devices von Drittanbietern (wenn vom Hersteller keine Begrenzungen von F_Source_Add festgelegt wurden) 0 und 65535 sind nicht zulässig.	1
F_Dest_Add	Eindeutige F-Device-Adresse, die mit der eingestellten Drehschalter-Adresse im F-Device verglichen wird. Der Parameter F_Dest_Add ist eine logische Adressbezeichnung, die frei zugewiesen werden kann, aber eindeutig sein muss.	[1 – 255] für Sicherheits-E/A-Module AC500-S. Für SM560-S-FD-1 und SM560-S-FD-4: ● F_Dest_Add = Wert des Adressschalters (1 – 239) * 100 + F-Device-Instanznr. (0..31) ● Die Werte des Adressschalters [240 – 255] sind für Systemfunktionen reserviert.	2 für Sicherheits-E/A-Module 100 für SM560-S-FD-1 oder SM560-S-FD-4
F_WD_Time	Watchdog-Zeit in ms für den Empfang von einem neuen gültigen Telegramm	[10 – 10000]	F-Devices von ABB: 100 F-Devices von Drittanbietern: gemäß GSDML-Datei

F_Parameter	Definition	Zulässige Werte	Standardwert
F_iPar_CRC	CRC für iParameter (herstellerspezifisch) von F-Devices (Sicherheits-E/As).	[0 – 4294967295] Hex [0 – FFFFFFFF]	Für Sicherheits-E/A-Module: abhängig von der iParameter-Standardkonfiguration für Sicherheits-E/A-Module. Nicht anwendbar auf SM560-S-FD-1 und SM560-S-FD-4.
F_Par_CRC	CRC1-Signaturberechnung für alle F-Parameter	[0 – 65535] Hex [0 – FFFF]	Abhängig vom Modultyp

iParameter sind individuelle F-Device-Parameter, die mit einem geeigneten F_iPar_CRC-Parameter an die F-Devices übertragen werden.

HINWEIS!

Die Implementierung des AC500-S PROFIsafe F-Host unterstützt nicht oder nur teilweise folgende Funktionen der PROFIsafe-Konformitätsklasse ↯ *[2]*:

- Kommunikations-Funktionsbausteinsatz RDREC, WRREC, RDIAG und RALRM, wie in ↯ *[11]* definiert
- iPar-Serverleistungen
- Schnittstelle zum Aufruf von Software-Tools, wie in ↯ *[2]* definiert.

HINWEIS!

Nach dem Verändern der iParameter öffnen Sie die Registerkarte *„F-Parameter“*, berechnen die Prüfsumme für iParameter neu und kopieren diese in die Zeile des F-Parameters F_iPar_CRC. Anderenfalls wird der neue Parametersatz vom F-Device nicht akzeptiert, weil F_iPar_CRC für einen bestimmten iParameter-Satz nicht gültig ist.

Für F-Devices von Drittanbietern, die mit einer GSDML-Datei importiert wurden, **gibt es die Funktion *„Prüfsumme iParameter“* nicht,** weil der Automation Builder den speziellen Algorithmus, der für die Berechnung von F_iPar_CRC bei Fremdgeräten verwendet wird, nicht kennt. F_iPar_CRC muss mit einem speziellen Software-Tool berechnet werden, das vom Hersteller des F-Device für die Entwicklung seiner F-Devices geliefert wird.

Eine weitere Möglichkeit ist, den Verkäufer des F-Device zu kontaktieren und den F_iPar_CRC für den entsprechenden F-Device iParameter anzufordern. Sobald F_iPar_CRC für ein F-Device von Drittanbietern verfügbar ist, können Sie ihn in die Zeile F_iPar_CRC im F-Parameter-Editor kopieren.

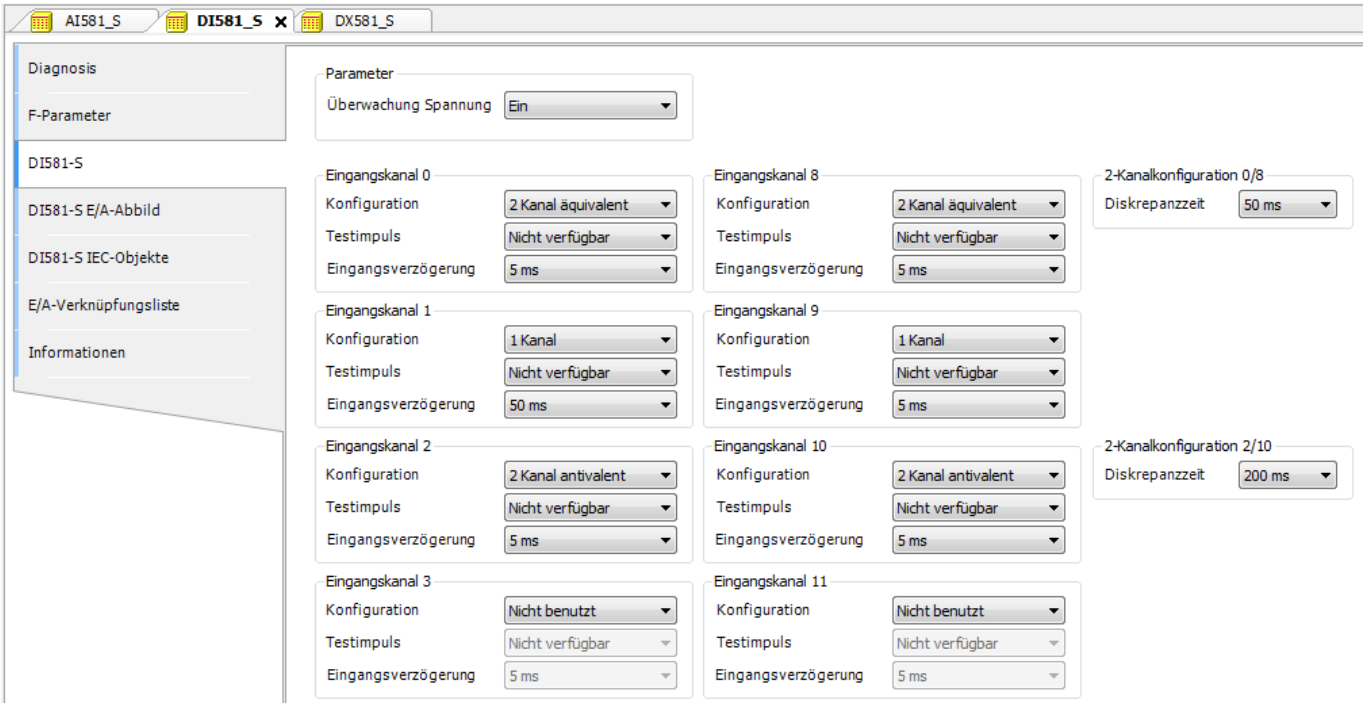


Abb. 70: Beispiele für iParameter-Einstellungen für das Sicherheitsmodul DI581-S; alle Eingangskanäle sind als „Kanal X mit Kanal X + 8“ gepaart

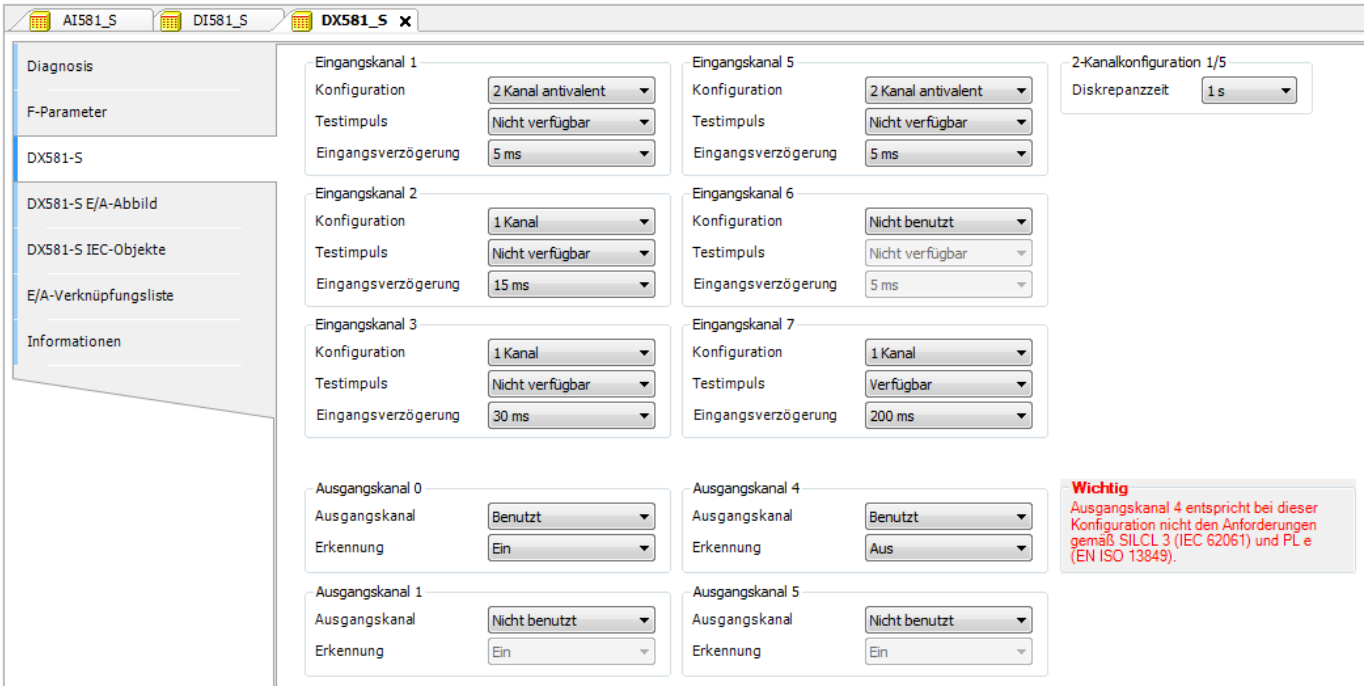


Abb. 71: Beispiele für iParameter-Einstellungen für das Sicherheitsmodul DX581-S; Eingangskanäle sind als „Kanal X mit Kanal X + 4“ gepaart

GEFAHR!

Wenn für einen der Ausgangskanäle „Erkennung = AUS“ gesetzt wird, erscheint eine Warnung, dass der Ausgangskanal in diesem Fall nicht den Anforderungen gemäß max. SIL 3 (IEC 62061) und PL e (ISO 13849-1) entspricht. Zwei Sicherheits-Ausgangskanäle müssen verwendet werden, um die entsprechenden SIL- und PL-Werte zu erreichen.

Der Parameter „Erkennung“ wurde für Anwender entwickelt, die Sicherheitsausgänge des DX581-S für Sicherheitsfunktionen gemäß max. SIL 1 (oder max. SIL 2 unter speziellen Bedingungen) oder PL c (oder max. PL d unter speziellen Bedingungen) nutzen möchten und weniger interne Impulse des DX581-S auf der Sicherheits-Ausgangsleitung sichtbar haben möchten. Solche internen Impulse könnten als LOW-Signal z. B. von Antriebseingängen erkannt werden, was zu einem ungewollten Maschinenstopp führen würde.

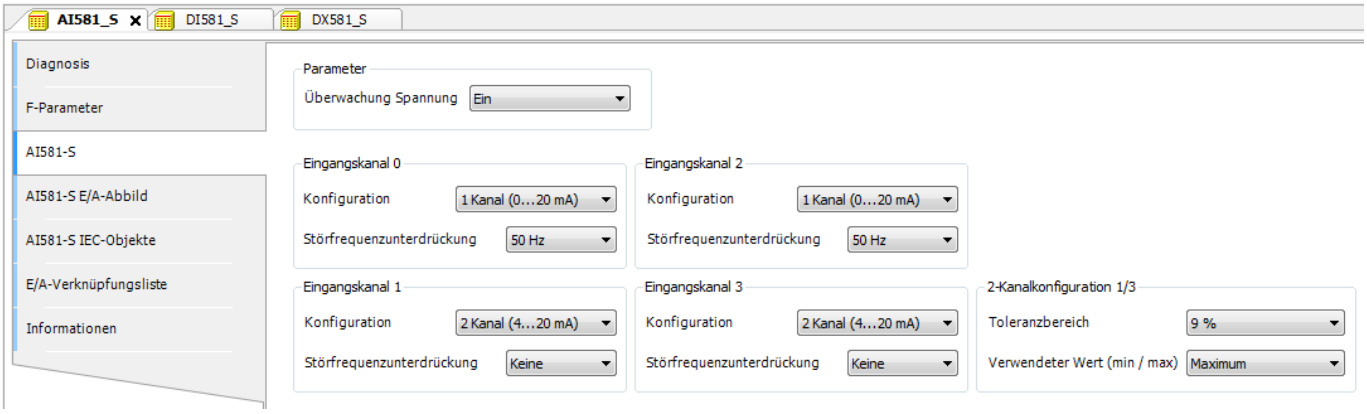


Abb. 72: Beispiele für iParameter-Einstellungen für das Sicherheitsmodul AI581-S; Eingangskanäle sind als „Kanal X mit Kanal X + 2“ gepaart

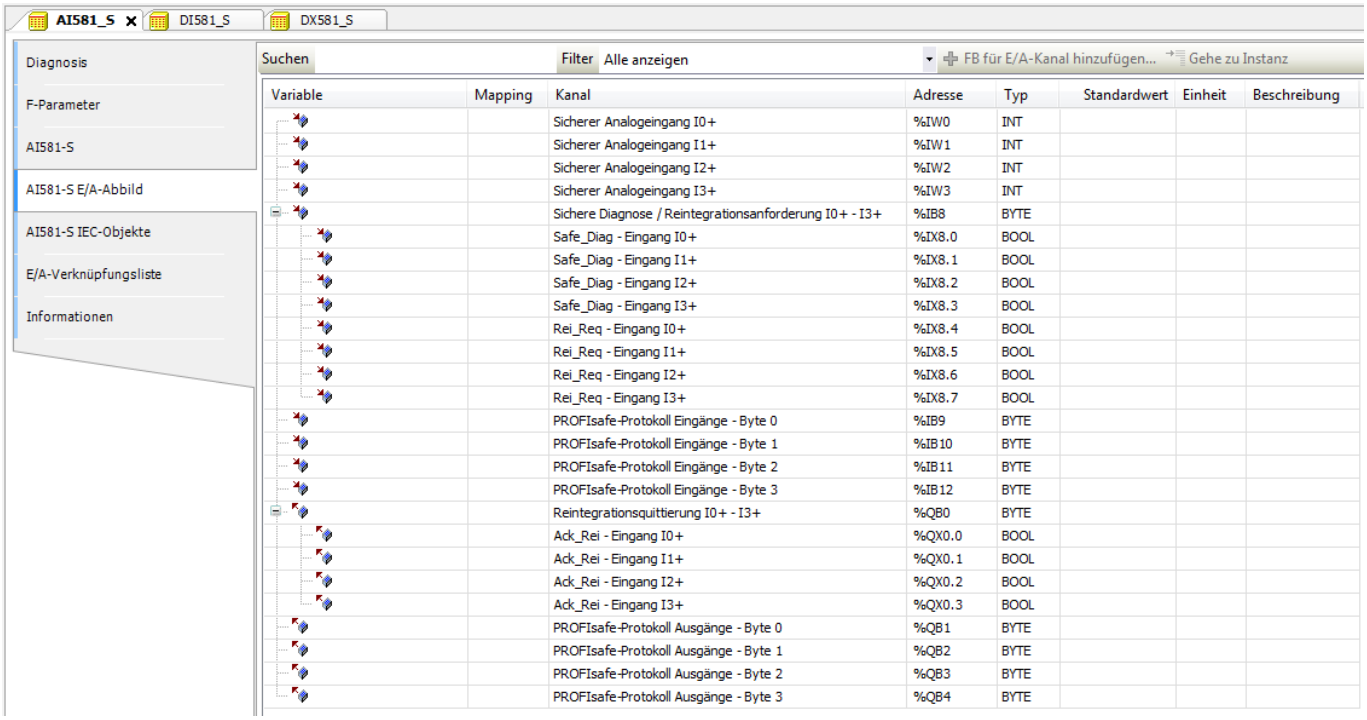
GEFAHR!

Zur Bearbeitung von Modul- und Kanalparametern kann auch die generische Gerätekonfiguration in den Registerkarten *„DI581-S-Parameter“*, *„DX581-S-Parameter“* oder *„AI581-S-Parameter“* verwendet werden. **Eine Veränderung der Sicherheits-E/A-Parameter über die generische Gerätekonfiguration wird jedoch nicht empfohlen**, da bei der Parametereinstellung eventuelle Fehler bei der Eingabe von Integerzahlen entstehen können.

Außerdem verfügt jedes F-Device über eine spezielle Registerkarte *„E/A-Abbild“*, auf der Variablennamen für Ein- und Ausgangssignale, PROFIsafe-Diagnosebits usw. definiert werden können.

GEFAHR!

Wenn Datentypen wie Unsigned16, Unsigned32, Integer16, Integer32 oder Float32, die mehr als ein Byte erfordern, in PROFIsafe-Daten verwendet werden, ist Folgendes zu beachten. Die Bytefolge bei diesen Datentypen hängt von der verwendeten Byte-Reihenfolge (Endianwert) des PROFIsafe-Gerätes und vom ausgewählten Typ der AC500-Standard-CPU ab. AC500 V2-Standard-CPU unterstützt Big Endian und AC500 V3-Standard-CPU unterstützt Little Endian. Stellen Sie sicher, dass die symbolischen Variablen ordnungsgemäß zugeordnet und die gelieferten Sicherheitsdaten in Ihrer Sicherheitsanwendung korrekt dargestellt sind.



Variable	Mapping	Kanal	Adresse	Typ	Standardwert	Einheit	Beschreibung
		Sicherer Analogeingang I0+	%IW0	INT			
		Sicherer Analogeingang I1+	%IW1	INT			
		Sicherer Analogeingang I2+	%IW2	INT			
		Sicherer Analogeingang I3+	%IW3	INT			
		Sichere Diagnose / Reintegrationsanforderung I0+ - I3+	%IB8	BYTE			
		Safe_Diag - Eingang I0+	%IX8.0	BOOL			
		Safe_Diag - Eingang I1+	%IX8.1	BOOL			
		Safe_Diag - Eingang I2+	%IX8.2	BOOL			
		Safe_Diag - Eingang I3+	%IX8.3	BOOL			
		Rei_Req - Eingang I0+	%IX8.4	BOOL			
		Rei_Req - Eingang I1+	%IX8.5	BOOL			
		Rei_Req - Eingang I2+	%IX8.6	BOOL			
		Rei_Req - Eingang I3+	%IX8.7	BOOL			
		PROFIsafe-Protokoll Eingänge - Byte 0	%IB9	BYTE			
		PROFIsafe-Protokoll Eingänge - Byte 1	%IB10	BYTE			
		PROFIsafe-Protokoll Eingänge - Byte 2	%IB11	BYTE			
		PROFIsafe-Protokoll Eingänge - Byte 3	%IB12	BYTE			
		Reintegrationsquittierung I0+ - I3+	%QB0	BYTE			
		Ack_Rei - Eingang I0+	%QX0.0	BOOL			
		Ack_Rei - Eingang I1+	%QX0.1	BOOL			
		Ack_Rei - Eingang I2+	%QX0.2	BOOL			
		Ack_Rei - Eingang I3+	%QX0.3	BOOL			
		PROFIsafe-Protokoll Ausgänge - Byte 0	%QB1	BYTE			
		PROFIsafe-Protokoll Ausgänge - Byte 1	%QB2	BYTE			
		PROFIsafe-Protokoll Ausgänge - Byte 2	%QB3	BYTE			
		PROFIsafe-Protokoll Ausgänge - Byte 3	%QB4	BYTE			

Abb. 73: Beispiel für Variablenabbild am Modul AI581-S

Dies gilt ebenso für die Sicherheitsmodule DX581-S und DI581-S; der einzige Unterschied liegt in der Anzahl der Ein- und Ausgangskanäle. Jeder Prozesskanal (Eingänge 0 bis 3 für AI581-S) verfügt zusätzlich über die folgenden Bits:

- Ein Bit für Sicherheitsdiagnose (Safe_Diag), um zu unterscheiden, ob der Prozesswert einen echten Prozesszustand widerspiegelt oder aufgrund von Kanal- oder Modulpassivierung ein „0“-Wert ist.
- Ein Bit für die Reintegrationsanforderung des Kanals (Rei_Req), das im Sicherheitsprogramm als Signal verwendet werden kann, dass ein externer Fehler (z. B. falsche Verdrahtung) behoben wurde und der Kanal wieder in die Sicherheitssteuerung integriert werden kann. Endanwender können so eine höhere Verfügbarkeit des Systems erwarten, weil sie von Fall zu Fall entscheiden können, welche Kanäle quittiert werden und welche nicht.
- Ein Bit für die Kanalreintegration (Ack_Rei), sobald der Fehler behoben wurde (weil z. B. die Verdrahtung des externen Sensors korrigiert wurde). Man kann auch eine Variable als BYTE für alle Ack_Rei-Bits definieren und 0xFF-Werte zur Quittierung sämtlicher Fehler auf einmal verwenden.

HINWEIS!

Wenn Sie Variablennamen für Eingangs-, Ausgangs- oder andere Sicherheitssignale definieren, beachten Sie die Sicherheitsprogrammierrichtlinien von
↳ *Kapitel 4.4 „Sicherheitsprogrammierrichtlinien“ auf Seite 196*.

HINWEIS!

Anstelle von WORD wird nur der Datentyp BYTE für Sicherheitsdaten des Moduls DI581-S unterstützt, wenn AC500 V3-Standard-CPU verwendet wird. Dies ist erforderlich, um die Byte-Reihenfolge einzuhalten, die bei AC500 V2-Standard-CPU (Big Endian) und AC500 V3-Standard-CPU (Little Endian) unterschiedlich ist. Dies ist bei der Migration des Sicherheitsprojekts von einer AC500 V2 auf eine V3-Standard-CPU zu berücksichtigen.

4.3.6 Programmierung der AC500-S-Sicherheits-CPU

Programm für die Sicherheitsanwendung schreiben und Startvorgang des Systems beachten.

HINWEIS!

Wie ein gültiges Bootprojekt für Standard-CPUs angelegt, konfiguriert und heruntergeladen wird, ist beschrieben in ↳ *[3]*.

Um unerwartete Konfigurationsfehler zu vermeiden, laden Sie zuerst ein gültiges Projekt auf die Standard-CPU. Laden Sie im nächsten Schritt ein Sicherheitsprojekt auf die Sicherheits-CPU.

1. Programmieren Sie ein gültiges Projekt und laden Sie dieses auf die Standard-CPU.
2. Starten Sie AC500-S Programming Tool durch einen Doppelklick auf den Knoten der Sicherheitsanwendung, z. B. *„AC500_S“*.

   ⇨ Bevor AC500-S Programming Tool gestartet wird, müssen Sie eventuell Ihre Konfiguration aktualisieren. Das ist erforderlich, um die aktualisierten Konfigurationsdaten (z. B. Variablennamen usw.) an AC500-S Programming Tool zu übertragen.

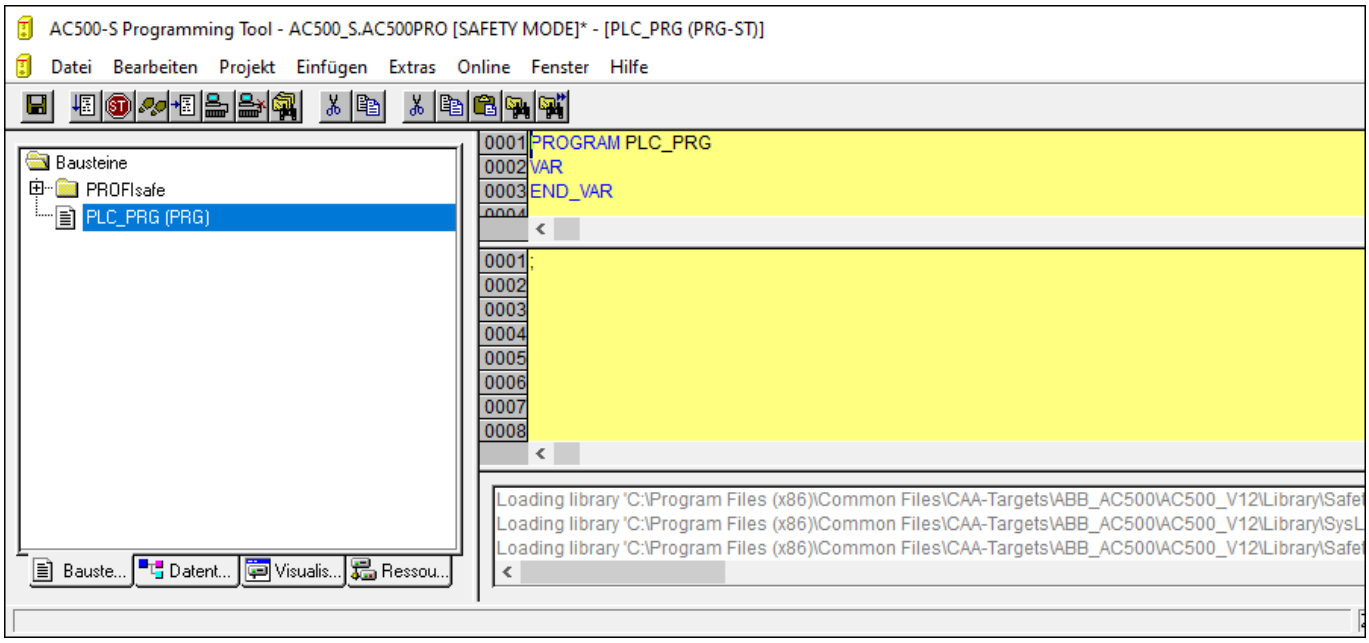


Abb. 74: AC500-S Programming Tool

GEFAHR!

Stellen Sie sicher, dass beim Start von AC500-S Programming Tool Folgendes angezeigt wird:

- Gelber Hintergrund
- SAFETY MODE wird auf der Titelleiste angezeigt.

HINWEIS!

Wenn AC500-S Programming Tool zum ersten Mal im Automation Builder-Projekt gestartet wird, werden Sie aufgefordert, die Identifikationsdaten der Sicherheitsbibliothek (Versionsnummer und CRC) manuell zu bestätigen. Danach werden diese Daten im Projekt gespeichert.

Wenn Sie den Inhalt der Sicherheitsbibliothek ändern und auf Ihrer Festplatte überschreiben, werden Sie beim nächsten Start von AC500-S Programming Tool darüber informiert, dass eine der Sicherheitsbibliotheken geändert wurde. **Im Eigenschaften-Fenster für Sicherheitsbibliotheken befindet sich immer noch der ursprünglich gespeicherte CRC-Wert.** Wenn Sie das Projekt kompilieren, wird jedoch eine CRC-Fehlermeldung angezeigt. Da die Bibliothek geändert wurde, wird das Projekt von AC500-S Programming Tool nicht kompiliert.

Um ein Projekt erfolgreich zu kompilieren, löschen Sie die gewählte Sicherheitsbibliothek manuell und fügen eine neue Sicherheitsbibliothek mit einer neuen CRC hinzu. Die neue Sicherheitsbibliothek wird mit neuer CRC akzeptiert und kein Kompilierfehler wird angezeigt.

3. Definieren Sie die Benutzerverwaltung für AC500-S Programming Tool.

   Alle Optionen der Benutzerverwaltung von AC500-S Programming Tool sind für den Projektadministrator verfügbar ↳ *[3]*.

   Der Projektadministrator muss ein Benutzerpasswort für ein neu angelegtes Sicherheitsprojekt festlegen. Unter *„Projekt ➔ Passwörter für Arbeitsgruppe …“* können Sie ein Passwort für die Benutzergruppe der Stufe 0 festlegen. Dies sind Benutzer der Sicherheitsgruppe im Automation Builder.

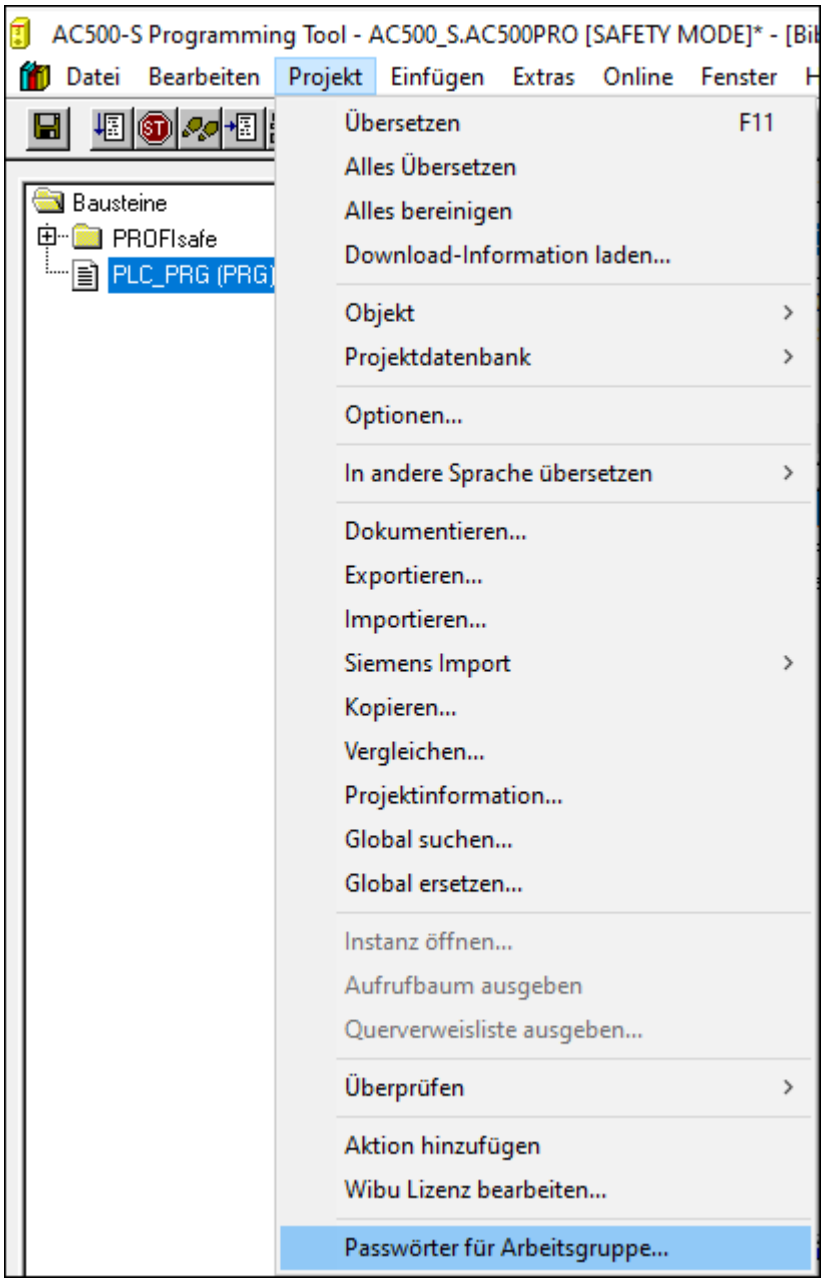


Abb. 75: Passwörter festlegen

4. Überprüfen Sie die F-Device-Konfiguration in AC500-S Programming Tool.

   Wenn Ihre Konfiguration der F-Devices endgültig ist, müssen Sie prüfen, ob die F-Parameter in der Registerkarte „F-Parameter“ dieselben sind wie jene, die in AC500-S Programming Tool importiert wurden: Öffnen Sie die Registerkarte *„Ressourcen“* im Sicherheitsprojekt. Navigieren Sie zu *„Globale Variablen ➔ PROFIsafe“* und wählen Sie die F-Device-Instanz aus, die Sie überprüfen möchten.

> **GEFAHR!**
> Sie müssen formell bestätigen, dass die F-Parameter-Werte der Registerkarte „F-Parameter“ dieselben sind wie jene, die in AC500-S Programming Tool importiert wurden (Punkt 3 in *Kapitel 6.2 „Checkliste für die Erstellung von Sicherheitsprogrammen“ auf Seite 374*).

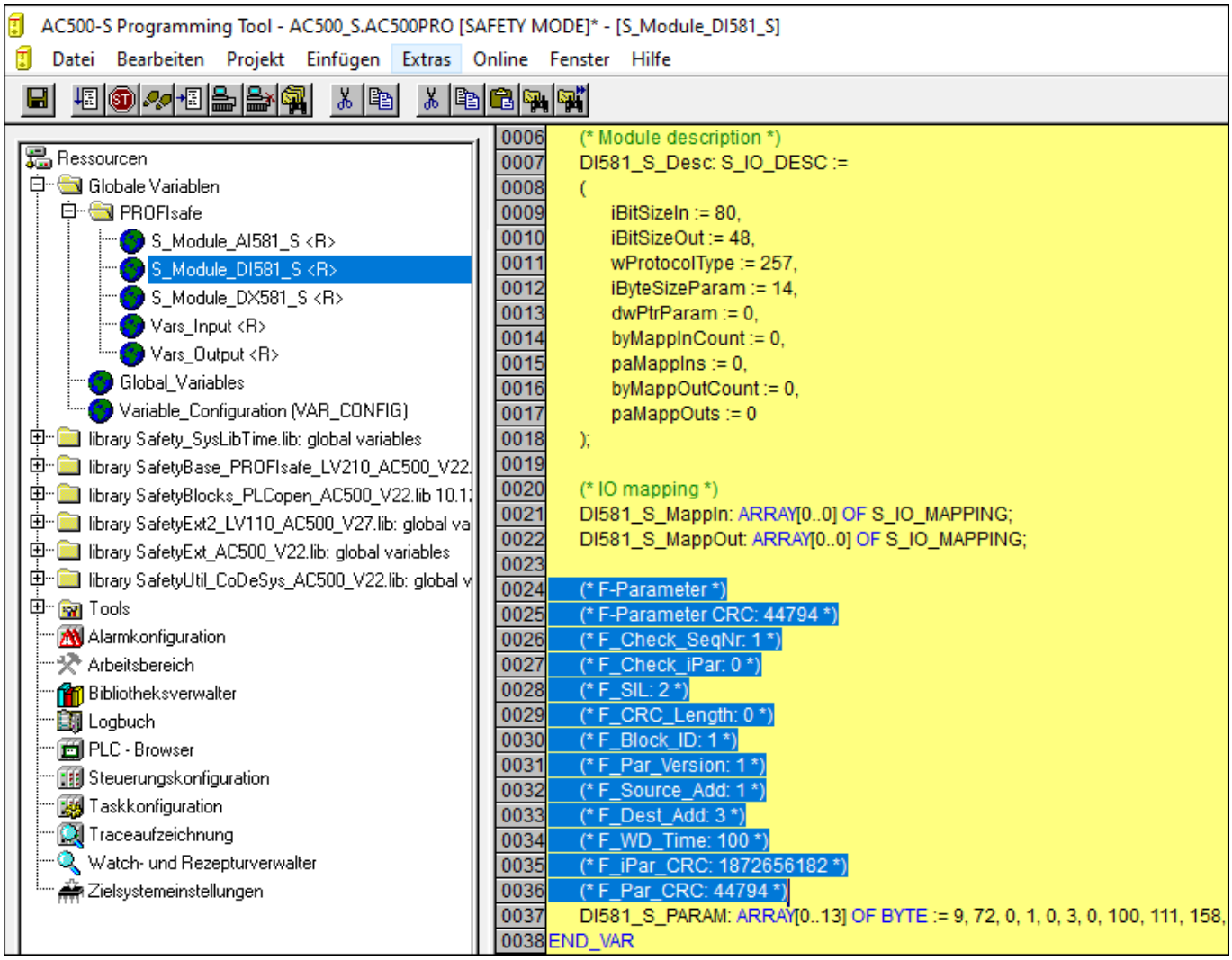


Abb. 76: F-Parameter-Werte in AC500-S Programming Tool

5. Alle konfigurierten Ein- und Ausgangsvariablen sind in der separaten Liste „Globale Variablen“ enthalten.

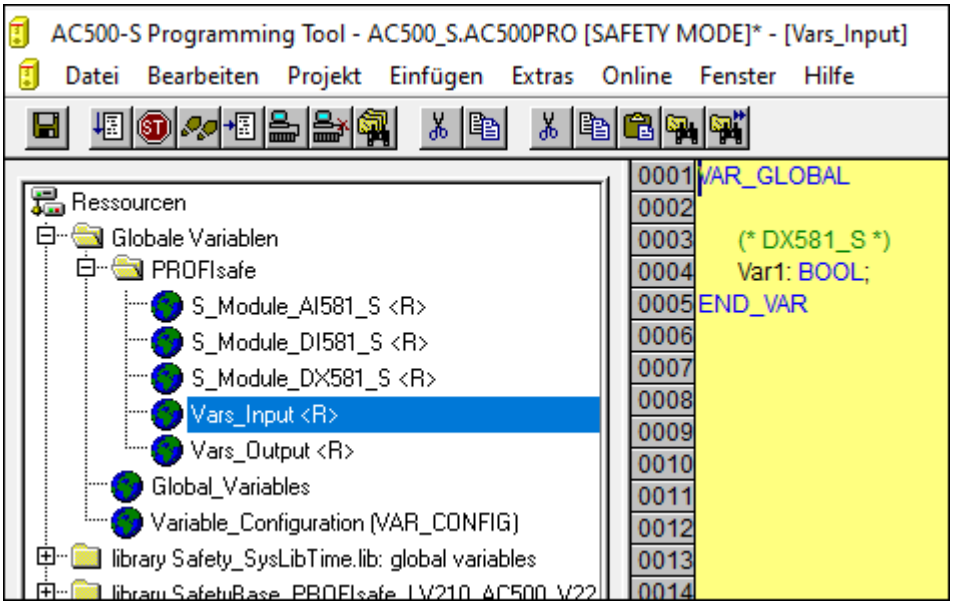


Abb. 77: Liste der globalen Variablen in AC500-S Programming Tool

⇨

GEFAHR!

Schreibgeschützte Ressourcen (<R>), die Task-Konfiguration und vorzertifizierte POEs (CallbackInit, CallbackReadInputs, CallbackWriteOutputs, InitPROFIsafe, ReadPROFIsafeInputs, WritePROFIsafeOutputs) im PROFIsafe-Ordner in AC500-S Programming Tool dürfen nicht geändert werden. Eine Veränderung von <R>-Ressourcen könnte zu Inkonsistenzen zwischen Automation Builder und dem Sicherheitsprojekt führen.

HINWEIS!

Alle konfigurierten sicherheitsgerichteten Ein-/Ausgangsvariablen werden auch im nicht sicherheitsgerichteten Projekt angezeigt (z. B. zur Anzeige auf Bedienpanels, Datenerfassung usw.).

Der Unterschied zum Sicherheitsprojekt ist, dass Endanwender die Werte dieser Sicherheitsvariablen aus dem nicht sicherheitsgerichteten Standardprojekt nicht ändern können. Dies ist per Design ausgeschlossen.

6. Überprüfen Sie die Gültigkeit der Sicherheitsbibliotheken.

   Überprüfen Sie in der Bibliotheksverwaltung, ob die CRCs der verwendeten Sicherheitsbibliotheken jenen entsprechen, die aufgelistet sind unter ↳ *Tab. 14 „Sicherheitsbibliotheken“ auf Seite 207*.

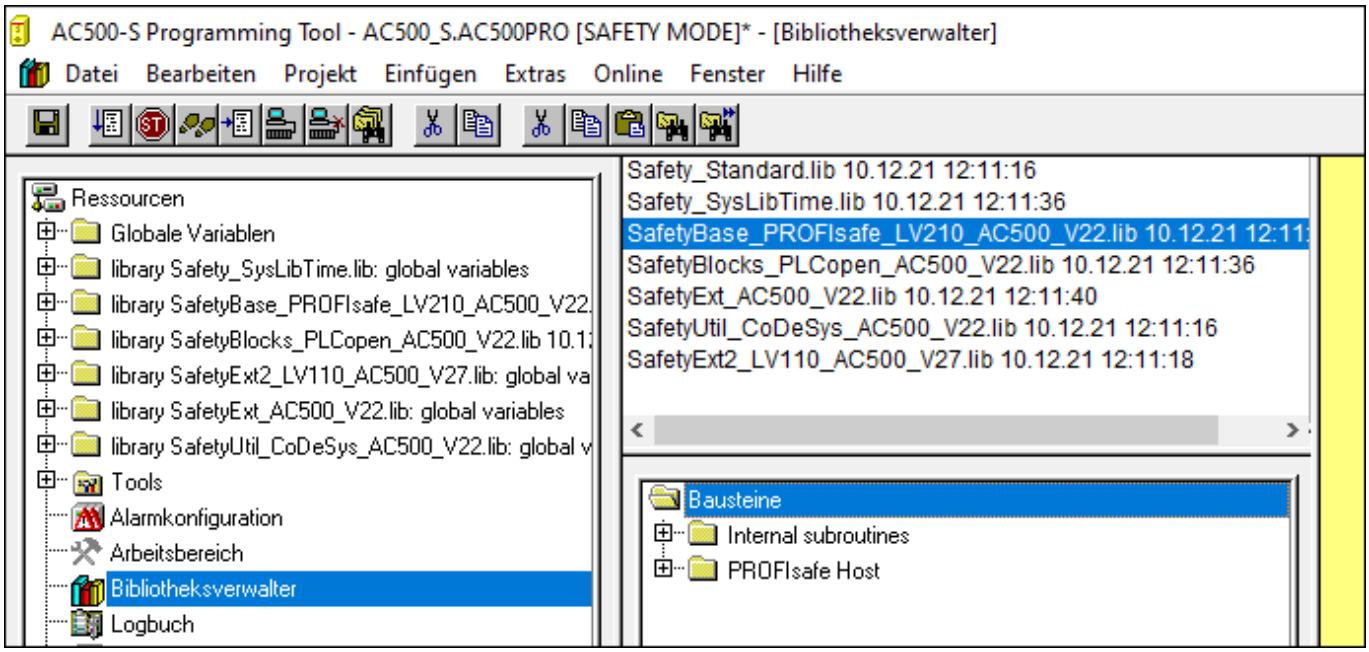


Abb. 78: Alle verfügbaren Sicherheitsbibliotheken sind in der Bibliotheksverwaltung aufgeführt.

⇨

GEFAHR!

Der Anwender ist dafür verantwortlich, sicherzustellen, dass nur zertifizierte Sicherheitsbibliotheken im Projekt verwendet werden. Weitere Informationen sind in der Übersicht der zertifizierten Sicherheitsbibliotheken und CRCs enthalten ↳ *Kapitel 4.6.1 „Übersicht“ auf Seite 207*.

Der Anwender ist allein verantwortlich für alle von ihm angelegten Bibliotheken, auf die er im Projekt zur Verwendung in Sicherheitsanwendungen verweist.

Eine formelle Bestätigung, dass in Ihrer Sicherheitsanwendung keine nicht sicherheitsgerichteten Bibliotheken verwendet werden, ist erforderlich (Punkt 19 in ↳ *Kapitel 6.2 „Checkliste für die Erstellung von Sicherheitsprogrammen“ auf Seite 374*).

HINWEIS!

Die Sicherheits-CPU AC500-S ist ein Single-Task-Gerät; deshalb ist eine Task-Konfiguration nicht erforderlich.

7. Starten Sie die Programmierung Ihrer Sicherheitsanwendung.

   Das Sicherheitsprogramm muss mit den folgenden Angaben gekennzeichnet werden: Projektname, Dateiname, Änderungsdatum, Titel, Autor, Version, Beschreibung und CRC. Mithilfe des Menüpunktes *„Online ➔ Prüfe Bootprojekt der Steuerung“* kann überprüft werden, ob das Offline-Projekt und das Bootprojekt in der Sicherheits-CPU identisch sind.

   Das Forcen der Variablen wird von der Sicherheits-CPU unterstützt, aber nur im DEBUG-Modus (nicht sicher); d. h. der Anwender trägt die volle Verantwortung für eventuelle Schäden aufgrund eines falschen Systemverhaltens im DEBUG-Modus (nicht sicher).

GEFAHR!

Das Forcen der Variablen in der Sicherheits-CPU ist nur gestattet, nachdem die in den operativen Kundenanwendungen schaltungsberechtigte Person für die Anlage konsultiert wurde. Beim Forcen muss der entsprechende Anwender sicherstellen, dass andere technische, organisatorische und strukturelle Maßnahmen ergriffen werden, um eine ausreichende sicherheitstechnische Überwachung des Prozesses sicherzustellen.

Für Sicherheitsanwendungen, die mit AC500-S entwickelt wurden, sind Visualisierungen mit AC500-S Programming Tool nur für Debugging- und Wartungszwecke gestattet.

GEFAHR!

Das Verändern von Werten mit Menübefehlen (z. B. „Werte schreiben“) versetzt die Sicherheits-CPU in den Modus DEBUG RUN; dieser ist nicht sicher.

Wenn der Modus DEBUG RUN (nicht sicher) auf der Sicherheits-CPU aktiviert wird, liegt die Verantwortung für einen sicheren Prozessablauf ausschließlich bei der Person oder Organisation, die den Modus DEBUG RUN (nicht sicher) aktiviert hat.

HINWEIS!

ST, FUP und KOP sind die einzigen Sprachen laut IEC 61131, die von der Sicherheits-CPU für Sicherheitsprogrammierung unterstützt werden. Beachten Sie die Richtlinien für die Sicherheitsprogrammierung von ↳ *Kapitel 4.4 „Sicherheitsprogrammierrichtlinien“ auf Seite 196*. ST mit einer Teilmenge gemäß ↳ *Kapitel 4.4* entspricht der Limited Variability Language laut IEC 61508.

HINWEIS!

Legen Sie keine Listen mit globalen Variablen an, deren Namen mit dem Präfix „S_Module_“ beginnen. Globale Variablen, die mit „S_Module_“ beginnen, werden automatisch von AC500-S Programming Tool aktualisiert und können zu einem Informationsverlust führen.

Beim Betrieb einer Sicherheitssteuerung ist es wichtig, dass alle F-Devices erfolgreich initialisiert wurden, bevor die Ausführung der Programmlogik gestartet wird. F-Devices starten im Modus FV_activated ↳ *weitere Details zum PROFIsafe F-Host-Stack siehe: Kapitel 4.6.3 SafetyBase_PROFIsafe_LV210_AC500_V22.lib, Seite 212*. Für einen simultanen Start empfehlen wir eine eigene spezielle POE, ähnlich wie SF_Startup (siehe unten), die verschiedene mögliche Startszenarien in der PROFIsafe-Spezifikation ↳ *[2]* verarbeitet und dann das Signal „Ready“ als Startsignal für die weitere normale Ausführung der Sicherheitsprogrammlogik gibt. Wie man in der Implementierung unten sieht, reicht es, wenn für mindestens einen der Kanäle des DI581-S das PROFIsafe-Diagnosebit „1“ ist, d. h. dass normale Prozesswerte geliefert werden können.

Deklarationsteil

```
FUNCTION_BLOCK SF_Startup

VAR_OUTPUT
   Ready: BOOL; (* TRUE setzen, sobald alle Sicherheitsmodule
initialisiert sind *)
END_VAR

VAR
    bTempReady: BOOL; (* Auf TRUE setzen, sobald Sicherheitsmodul
DI581-S bereit ist *)
END_VAR

VAR CONSTANT
    _TRUE: BOOL := TRUE; (* Konstant, da TRUE Buchstaben sind *)
    _FALSE: BOOL := FALSE; (* Konstant, da FALSE Buchstaben sind
*)
    wdNull: WORD := 16#0000; (* Konstant für Sicherheits-E/A-
Initialisierung *)
END_VAR

VAR_EXTERNAL
    DI581_S: PROFIsafeStack; (* Externe Deklaration *)
END_VAR
```

Implementierungsteil

```
(* Prüfen, ob Quittierung durch Bediener für F-Device verlangt
wird *)
IF DI581_S.OA_Req_S THEN (* Verlangt das Modul eine Quittierung?
*)
   DI581_S.OA_C := DI581_S.OA_Req_S; (* Ggf. Quittierung *)

(* IS_DI581_Started ist die Eingangsvariable für alle PROFIsafe-
Diagnosebits des Kanals, die in Control Builder Plus / Automation
Builder Plus für DI581-S gesetzt werden *)
ELSIF IS_DI581_Started > wdNull THEN (* Wurde dieses Modul
initialisiert? *)
   bTempReady := _TRUE; (* Ja, Modul wurde initialisiert *)
ELSE
   bTempReady := _FALSE; (* Nein, Modul wurde noch nicht
initialisiert *)
END_IF;

IF bTempReady THEN (* POE-Ausgangssignal setzen *)
   Ready := _TRUE;
ELSE
    Ready := _FALSE;
END_IF;
```

HINWEIS!

Zur Quittierung des F-Device nach Modulpassivierung muss das Befehlsbit OA_C von „0“ auf „1“ gesetzt werden, bis das Statusbit OA_Req_S „0“ wird.

8. Richten Sie die korrekten Kommunikationsparameter ein.

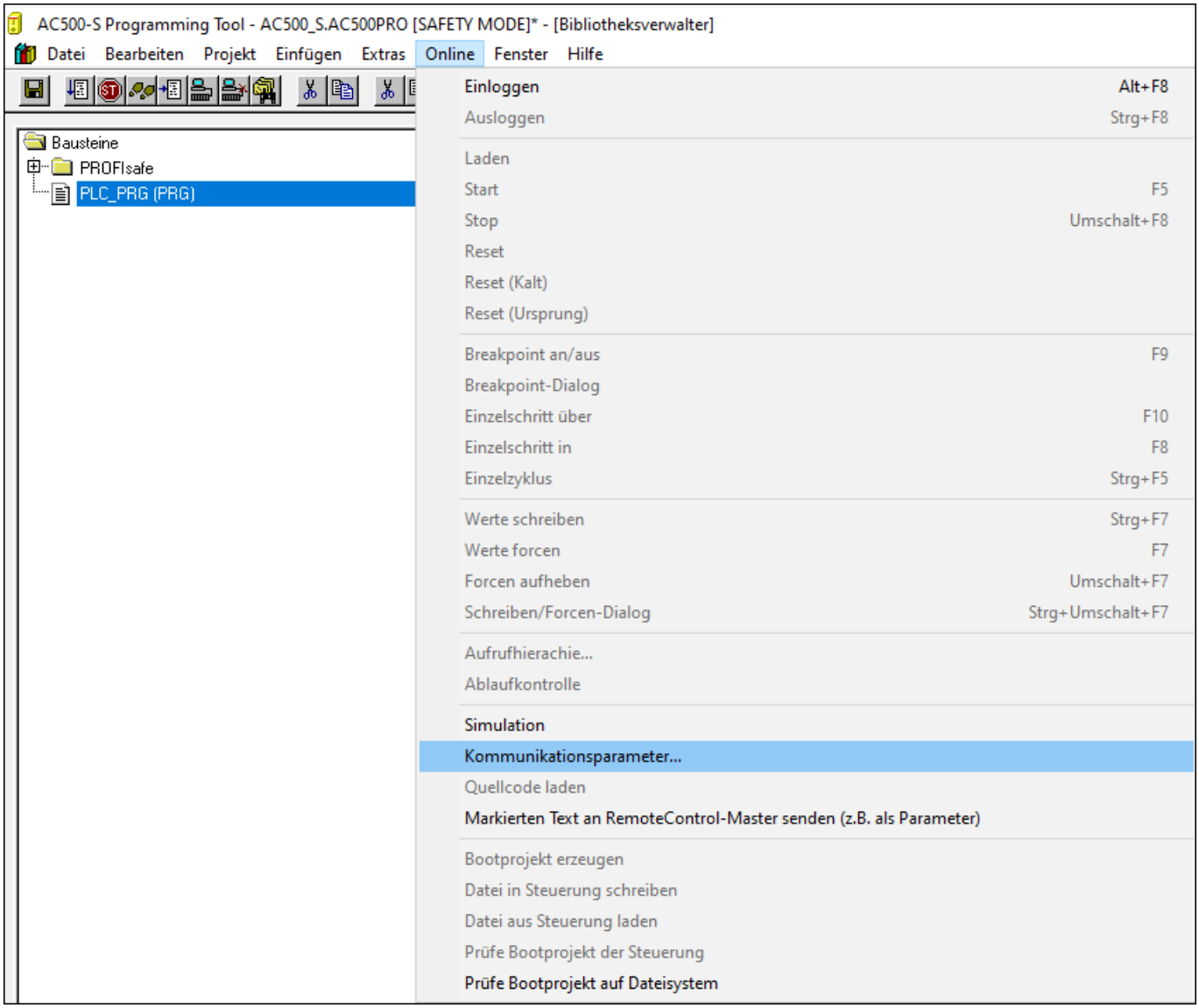


Abb. 79: Kommunikationsparameter festlegen

⇨

HINWEIS!

Stellen Sie sicher, dass zum Laden des Sicherheitsprojekts entweder der Kommunikationskanal *„ABB Tcp/Ip Level 2 AC“* oder der Kommunikationskanal *„ABB RS232 AC“* ausgewählt wurde.

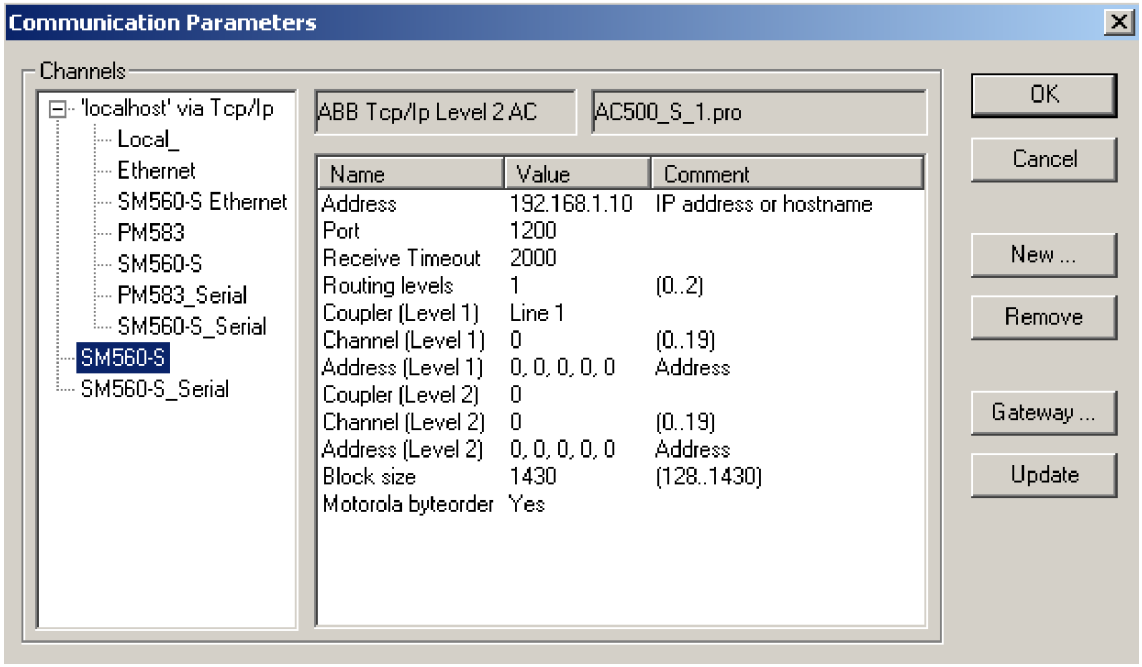


Abb. 80: Beispiel mit Ethernet-Anschluss

Beachten Sie, dass „Address“ die IP-Adresse der Standard-CPU ist, sofern diese auf der Standard-CPU unterstützt wird (man kann auch den COM-Port zum Programm-Download mit serieller Verbindung verwenden). Der Koppler (Level 1) definiert die Position der Sicherheits-CPU (Zeile 1 – Position 1, Zeile 2 – Position 2 usw.).

Weitere Informationen zu „Kommunikationsparametern“ finden Sie in *[3]*.

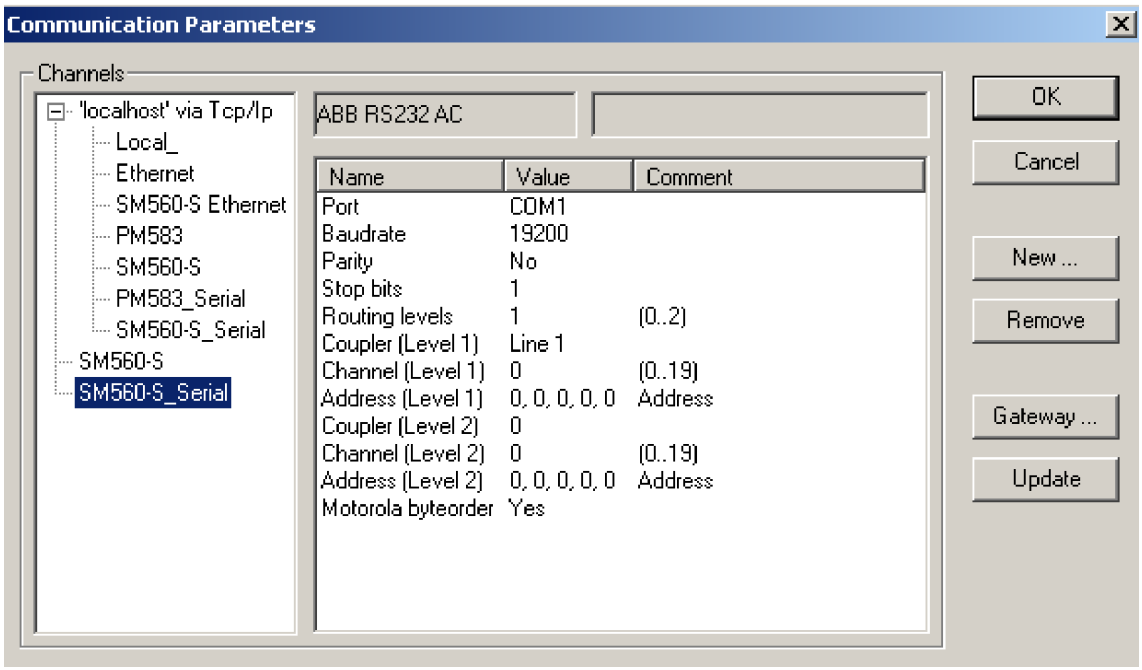


Abb. 81: Beispiel mit einer seriellen Verbindung

9. Laden Sie die Sicherheitsanwendung auf die Sicherheits-CPU.

   Das Sicherheitsprogramm kann von einem PC oder mithilfe einer SD-Karte auf die Sicherheits-CPU übertragen werden.

   ↳ *„Laden Sie Ihr Sicherheitsprogramm von einem PC auf die Sicherheits-CPU.“ auf Seite 173*

   ↳ *„Laden Sie Ihr Sicherheitsprogramm von einer SD-Karte auf die Sicherheits-CPU.“ auf Seite 175*

Laden Sie Ihr Sicherheitsprogramm von einem PC auf die Sicherheits-CPU.

10. Laden Sie Ihre Sicherheitsanwendung herunter und erzeugen Sie ein Bootprojekt, sodass Ihre Sicherheits-CPU die Ausführung des Sicherheitsprogramms nach dem Power Cycle beginnen kann.

> **HINWEIS!**
> Aus Sicherheitsgründen wird der Dienst *„Online-Change“* von der Sicherheits-CPU nicht unterstützt. Dies bedeutet, dass jede Änderung am Sicherheitsprojekt den Stopp der Sicherheits-CPU, das Laden eines neuen Bootprojekts und das Ausführen eines Power Cycle oder den Neustart durch die Standard-CPU erfordert, damit die Änderungen am Sicherheitsprogramm aktiviert werden.

> **HINWEIS!**
> Es kann jeweils nur ein Anwender an der Sicherheits-CPU angemeldet sein. Dies ist erforderlich, um verschiedene Änderungen von gleichzeitig arbeitenden Anwendern an der Sicherheits-CPU zu vermeiden.
>
> Die Beschränkung der Zahl offener Verbindungen gilt nur für die Sicherheits-CPU. Dies bedeutet, dass es weiterhin möglich ist, sich z. B. über das Internet und die OPC-Server-Funktionalität mit der Standard-CPU zu verbinden.

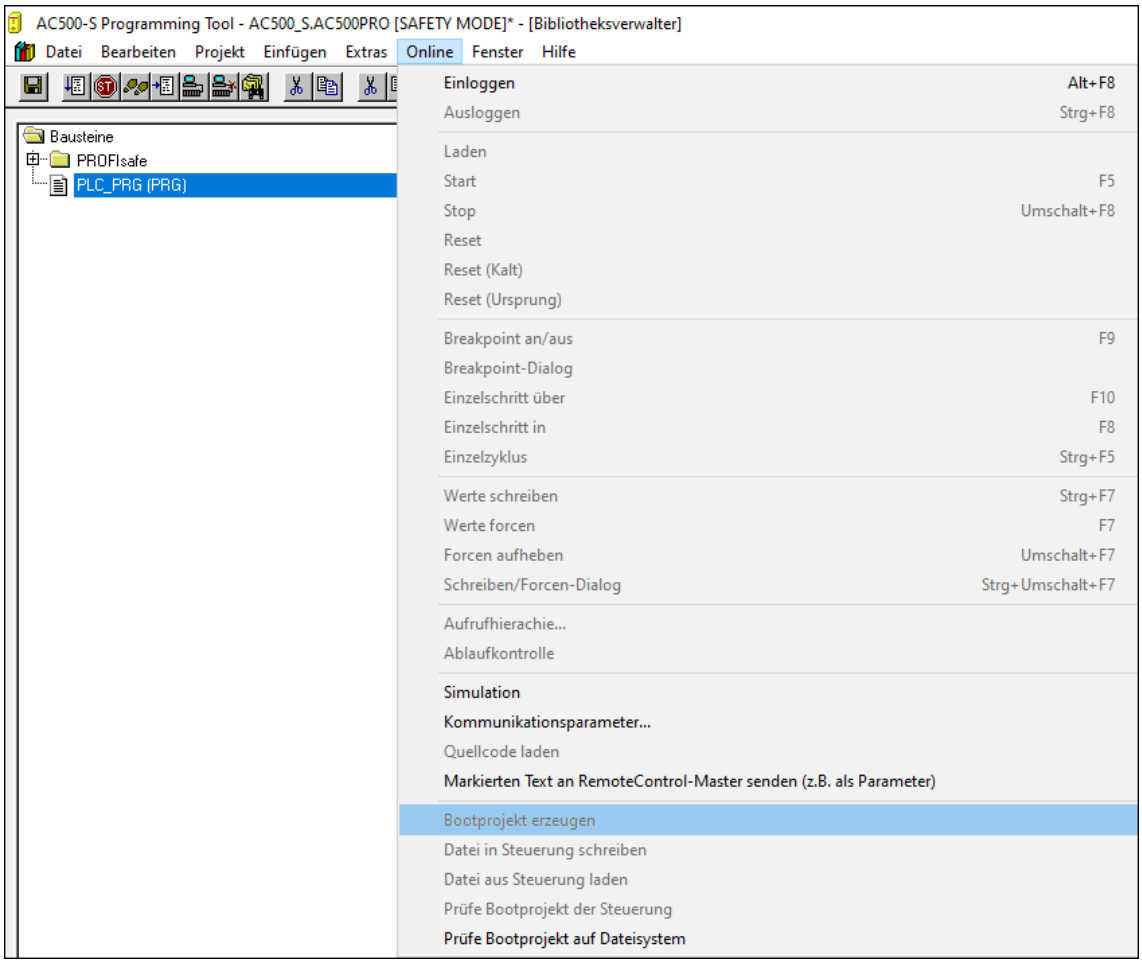


Abb. 82: Bootprojekt für die Sicherheits-CPU erzeugen

⇨

GEFAHR!

Wenn die Funktion *„Gerät aktualisieren …“* auf den Sicherheitsmodulen verwendet wurde, muss ein kompletter Funktionstest sämtlicher Teile der Sicherheitsanwendung durchgeführt werden. Für diesen Test muss die Maschine in ihrem endgültigen Zustand sein, d. h. einschließlich der mechanischen, elektrischen und elektronischen Komponenten, Sensoren, Aktoren und der Software.

HINWEIS!

Prüfen Sie über den Menüpunkt *„Online ➔ Prüfe Bootprojekt der Steuerung“*, ob das Offline-Projekt und das Bootprojekt in der Sicherheits-CPU identisch sind (Dateiname, Änderungsdatum, Titel, Autor, Version, Beschreibung und CRC).

Der gleiche Vergleich kann mit einem anderen Bootprojekt auf dem PC oder der SD-Karte mit dem Menüpunkt *„Online ➔ Prüfe Bootprojekt im Dateisystem“* durchgeführt werden.

Beachten Sie, dass das Bootprojekt mindestens einmal in die Sicherheits-CPU geladen werden muss, bevor es offline auf dem PC für Backup und spätere Verwendung gespeichert wird.

Es wird dringend empfohlen, die Befehle *„Alles bereinigen“* und *„Alles übersetzen“* aus dem *„Projekt“*-Menü vor dem Laden des Sicherheitsprogramms in die Sicherheits-CPU auszuführen.

HINWEIS!

Die Bootprojekt-CRC identifiziert eindeutig das Bootprojekt für die Sicherheits-CPU. Beachten Sie, dass nicht nur Code-Änderungen, sondern auch unterschiedliche Aktionen in der Programmierumgebung zu einer neuen Bootprojekt-CRC führen können.

Anwenderaktionen, die die CRC des sicherheitsgerichteten Bootprojekts ändern:

- In AC500-S Programming Tool:
  - Wählen Sie die Registerkarte *„Ressourcen“*, öffnen Sie *„Zielsystemeinstellungen“* und klicken Sie auf *[OK]*, ohne Änderungen im Dialog vorzunehmen.
  - Wählen Sie die Registerkarte *„Projekt ➔ Optionen“* und klicken Sie auf *[OK]*, ohne Änderungen im Dialog vorzunehmen.
  - Wählen Sie die Registerkarte *„Ressourcen“*, öffnen Sie *„Arbeitsbereich“* und klicken Sie auf *[OK]*, ohne Änderungen im Dialog vorzunehmen.
- Im Automation Builder:
  - Doppelklicken Sie auf die Sicherheits-CPU, öffnen Sie die Registerkarte *„CPU-Parameter“* und ändern Sie die entsprechenden Parameter, z. B. *„Debug-Modus aktivieren“*. Öffnen Sie anschließend AC500-S Programming Tool (Doppelklick auf Knoten der Sicherheitsanwendung).
  - Mit AC500 V2-Standard-CPU: Doppelklicken Sie auf die Sicherheits-CPU, nehmen Sie Änderungen über die Registerkarte *„Konfiguration Datenaustausch“* vor und öffnen Sie AC500-S Programming Tool (Doppelklick auf Knoten der Sicherheitsanwendung).

HINWEIS!

Bitte beachten Sie, dass die Standard-CPU an der iParameter-Übertragung zu F-Devices beteiligt ist und Sie daher nicht nur Ihr Sicherheits-Anwendungsprogramm in die Sicherheits-CPU laden müssen, sondern auf ähnliche Weise ↳ *[3]* auch das nicht sicherheitsgerichtete Programm in die Standard-CPU laden und ein Bootprojekt für die Standard-CPU erstellen müssen.

Eine Missachtung dieser Empfehlungen kann zu Konfigurationsfehlern oder Passivierung einiger F-Devices führen.

GEFAHR!

Verwenden Sie den Befehl *„Datei in Steuerung schreiben“* nicht für die Sicherheits-CPU, da es zu einem Verlust wichtiger Anwenderinformationen oder Laden von korrupten Daten in die Sicherheits-CPU kommen kann.

Überspringen Sie den nächsten Schritt und fahren Sie mit dem übernächsten fort.

Laden Sie Ihr Sicherheitsprogramm von einer SD-Karte auf die Sicherheits-CPU.

11.

GEFAHR!

Wenn Sie Ihr Sicherheitsprogramm mit einer SD-Karte auf die Sicherheits-CPU übertragen, vergewissern Sie sich, dass die eingesteckte SD-Karte das korrekte Sicherheitsprogramm enthält. Dies kann durch Programmidentifizierung (z. B. Bootprojekt-CRC) oder andere Maßnahmen wie eine eindeutige Kennung auf der SD-Karte geprüft werden.

HINWEIS!

Das Bootprojekt der Sicherheits-CPU kann nur über eine Speicherkarte aktualisiert werden, wenn kein Bootprojekt in die Sicherheits-CPU geladen ist ↳ *„Bootprojekt-Aktualisierung“ auf Seite 47*.

- Sicherheitsprogramm auf die Speicherkarte übertragen ↳ *„Bootprojekt-Aktualisierung“ auf Seite 47*.
- Programmidentifizierung durchführen – über *„Online ➔ Prüfe Bootprojekt im Dateisystem“* überprüfen, ob die CRCs der Sicherheitsprogramme auf der Speicherkarte und Offline (z. B. auf dem PC) übereinstimmen.
- Ein entsprechendes Beschriftungsschild auf der Speicherkarte anbringen.

Dieser Vorgang muss durch organisatorische Maßnahmen sichergestellt werden.

12. Nach dem Einloggen in die Sicherheits-CPU können Sie SPS-Browser-Befehle verwenden.

    Die folgenden SPS-Browser-Befehle, die über AC500-S Programming Tool aufgerufen werden können, werden von der Sicherheits-CPU unterstützt:

?	-	Liste der verfügbaren Browser-Befehle
reflect	-	Ausgabe von Browser-Befehlen (zu Testzwecken)
pid	-	Zeigt die Projekt-ID
pinf	-	Zeigt die Projektinformationen im AC500-Format an
getprgprop	-	Zeigt die Programmeigenschaften im AC500-Format an
getprgstat	-	Zeigt den Programmstatus im AC500-Format an
setpwd	-	Setzt ein neues Passwort für die Sicherheits-CPU (zum Einloggen erforderlich). Dieser Befehl ist nur aktiv, wenn der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU auf „EIN“ gesetzt und ein geeignetes Bootprojekt in die Standard-CPU geladen wurde.
delpwd	-	Löscht das Passwort für die Sicherheits-CPU. Dieser Befehl ist nur aktiv, wenn der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU auf „EIN“ gesetzt und ein geeignetes Bootprojekt in die Standard-CPU geladen wurde.
rtsinfo	-	Zeigt Firmware- und Bootprojekt-Informationen im AC500-Format an
proddata	-	Zeigt die Produktionsdaten der Sicherheits-CPU im AC500-Format an
diagreset	-	Setzt das Diagnosesystem der Sicherheits-CPU zurück
diagack all	-	Quittiert alle Fehler
diagack x	-	Quittiert alle Fehler der Klasse x (x= 1 .. 4)
diagshow all	-	Zeigt alle Fehler im AC500-Format an
diagshow X	-	Zeigt alle Fehler der Klasse x an
delappl	-	Löscht das Bootprojekt aus dem Flash-Speicher. Dieser Befehl wird nur ausgeführt, wenn die Sicherheits-CPU im Zustand DEBUG STOP ist. Nach dem Neustart der Sicherheits-CPU muss sichergestellt werden, dass kein Bootprojekt in der Sicherheits-CPU verfügbar ist. Dieser Befehl ist nur aktiv, wenn der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU auf „EIN“ gesetzt und ein geeignetes Bootprojekt in die Standard-CPU geladen wurde.
deluserdat:	-	Löscht die Nutzerdaten aus dem Flash-Speicher. Dieser Befehl wird nur ausgeführt, wenn die Sicherheits-CPU im Zustand DEBUG STOP ist. Er wird sofort ausgeführt und ist nur aktiv, wenn der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU auf „EIN“ gesetzt und ein geeignetes Bootprojekt in die Standard-CPU geladen wurde.
applinfo	-	Zeigt die Anwendungsinformationen, z. B. die Ergebnisse der Zeitprofilerstellung mit den Funktionen SF_APPL_MEASURE_BEGIN und SF_APPL_MEASURE_END.
applinfo reset	-	Setzt alle Anwendungsinformationen, z. B. Zeit-Messwerte, zurück.
flashstatus	-	Zeigt den Status der Flash-Programmierung in der Sicherheits-CPU in % an, wenn Bootcode, Firmware oder ein Bootprojekt heruntergeladen werden.

    Keiner der oben erwähnten SPS-Browser-Befehle ändert den Zustand der Sicherheits-CPU (z. B. von RUN in DEBUG RUN oder DEBUG STOP usw.).

HINWEIS!

Die folgenden SPS-Browser-Befehle von der Sicherheits-CPU ändern ihren Zustand:

resetprg:

Bereitet den Neustart der Sicherheits-CPU mit initialen Variablenwerten vor. Die Sicherheits-CPU ändert ihren Zustand, z. B. von RUN in DEBUG STOP. *Dieser Befehl wird nur akzeptiert, wenn der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU auf „EIN“ gesetzt und ein geeignetes Bootprojekt in die Standard-CPU geladen wurde.*

resetprgorg:

Versetzt die Sicherheits-CPU in ihren Originalzustand (sämtliche Variablen, Flash-Speicher usw. erhalten wieder die Initialwerte). Die Sicherheits-CPU ändert ihren Zustand, z. B. von RUN in DEBUG STOP. *Dieser Befehl wird nur akzeptiert, wenn der Parameter „Debug-Modus aktivieren“ für die Sicherheits-CPU auf „EIN“ gesetzt und ein geeignetes Bootprojekt in die Standard-CPU geladen wurde.*

GEFAHR!

Die Ergebnisse der Befehle *„delappl“*, *„resetprgorg“*, *„setpwd“* und *„delpwd“* müssen von den Endanwendern nach einem Power Cycle der Sicherheits-CPU durch Einloggen überprüft werden.

4.3.6.1 Sichere Kommunikation von CPU zu CPU mit SM560-S-FD-1 und SM560-S-FD-4

Die Sicherheits-CPUs SM560-S-FD-1 und SM560-S-FD-4 bieten bis zu 32 F-Device-Instanzen für die sichere Kommunikation von CPU zu CPU. Die Sicherheitsdaten jeder F-Device-Instanz werden auf den PROFINET IO-Device-Kommunikationsmodulen CM589-PNIO oder CM589-PNIO-4 abgebildet. Die Kommunikationsmodule CM589-PNIO und CM589-PNIO-4 ermöglichen eine physikalische Trennung ihres PROFINET-Netzwerks von dem des PROFINET IO Controller-Kommunikationsmoduls CM579-PNIO auf derselben Standard-CPU.

Die GSDML-Dateien von ABB für die PROFINET-Geräte CM589-PNIO/CM589-PNIO-4 können zum Konfigurieren der Prozess- und Sicherheitsdatenparameter für PROFINET/PROFIsafe F-Host-Systeme von Drittanbietern verwendet werden.

Um alle Arten von PROFIsafe F-Hosts von Drittanbietern zu unterstützen, darunter auch solche, die die Verwendung des PROFINET UseAsBits-Attributs in einem PROFIsafe-Modul auf 64 Bits begrenzen, z. B. die Siemens-CPUs S7 3xx-F, wurden verschiedene Arten von Sicherheitsdatenbeschreibungen festgelegt.

Sicherheitsdatenbeschreibungen, die mit der PROFIsafe Protokollversion V2.4 kompatibel sind:

- Typ 1: 12 Bytes, die als UseAsBits definiert sind.
- Typ 2 (für F-Hosts, die 12 Bytes, die als UseAsBits definiert sind, nicht unterstützen): 8 Bytes, die als UseAsBits definiert sind, und zwei Integer16-Werte.

Sicherheitsdatenbeschreibungen, die mit der PROFIsafe Protokollversion V2.6 kompatibel sind:

- Typ 3: 12 Bytes, die als UseAsBits definiert sind.
- Typ 4: 123 Bytes, die als UseAsBits definiert sind.

Sichere Kommunikation von CPU zu CPU mit PROFINET/ PROFIsafe herstellen

1. Definieren Sie Master- und Slave-Steuerungen in einer Steuerungssystemkonfiguration. Beachten Sie, dass das gleiche System auch gleichzeitig Master und Slave sein könnte.

   - Alle Steuerungen, die nur Master sind, müssen mindestens über eine Standard-CPU, einen IO-Controller CM579-PNIO und eine Sicherheits-CPU SM560-S verfügen.
   - Alle Steuerungen, die nur Slaves sind, müssen mindestens über eine Standard-CPU, ein E/A-Gerät CM589-PNIO (oder CM589-PNIO-4, falls die Kommunikation mit mehr als 1 PROFINET IO-Controller erforderlich ist; es wird auch die Verwendung von mehr als 1 Kommunikationsmodul CM589-PNIO unterstützt) und eine Sicherheits-CPU SM560-S-FD-1 (oder SM560-S-FD-4, falls die Kommunikation mit mehr als 1 PROFINET IO-Controller erforderlich ist) verfügen.
   - Alle Steuerungen, die gleichzeitig Master und Slave sind, müssen mindestens über eine Standard-CPU, einen IO-Controller CM579-PNIO, ein E/A-Gerät CM589-PNIO (oder CM589-PNIO-4, falls die Kommunikation mit mehr als 1 PROFINET IO-Controller erforderlich ist; es wird auch die Verwendung von mehr als 1 Kommunikationsmodul CM589-PNIO unterstützt) und eine Sicherheits-CPU SM560-S-FD-1 (oder SM560-S-FD-4, falls die Kommunikation mit mehr als 1 PROFINET IO-Controller erforderlich ist) verfügen.

   HINWEIS!
   Nur eine Sicherheits-CPU kann an eine Standard-CPU angebracht werden. Die Anzahl der PROFINET-Kommunikationsmodule für die vorhandene Standard-CPU wird nur von der Anzahl der daran verfügbaren Steckplätze begrenzt.

   HINWEIS!
   PROFINET IO-Controller von Drittanbietern mit F-Hosts können ebenfalls in der Konfiguration verwendet werden. Verwenden Sie die GSDML-Dateien für CM589-PNIO / CM589-PNIO-4 von *www.abb.com/plc* für den Anschluss einer SPS AC500-S als Slave an ein Mastersystem von Drittanbietern.

2. Nach der Auswahl der PROFINET-Kommunikationsmodule und Sicherheits-CPUs im Master- und Slavesystem muss die Anzahl der Sicherheitsbytes festgelegt werden, die zwischen Slave- und Mastersystemen ausgetauscht werden müssen.

   Maximal nahezu 2000 Sicherheitsbytes (einschließlich PROFIsafe-Status-/Kontrollbytes und CRC) können ausgetauscht werden (für jede Eingangs- und Ausgangsrichtung), abhängig von den verwendeten Typen der Sicherheitsdatenbeschreibung.

   - Beispielsweise können bei Verwendung des Sicherheitsdatentyps 1..3 (durch Konfigurieren von 32 F-Device-Objekten, was der maximalen konfigurierbaren Anzahl von F-Device-Objekten entspricht) maximal 384 Bytes (ohne PROFIsafe-Status-/Kontrollbytes und CRC) ausgetauscht werden.
   - Beispielsweise können bei Verwendung des Sicherheitsdatentyps 4 (durch Konfigurieren von 15 F-Device-Objekten) 1845 Bytes Sicherheitsdaten (ohne PROFIsafe-Status-/Kontrollbytes und CRC) ausgetauscht werden. Zu beachten ist, dass unter Verwendung der Kommunikationsmodule CM589-PNIO oder CM589-PNIO-4 maximal 1440 Bytes ausgetauscht werden können. Um diese Einschränkung zu überwinden, müssen Sie mehr als ein Kommunikationsmodul des Typs CM589-PNIO oder CM589-PNIO-4 ergänzen. Für Unterstützung wenden Sie sich an den technischen Support von ABB.

3. Die Sicherheitsbytes können an den Slavesystemen durch Auswahl der Module CM589-PNIO bzw. CM589-PNIO-4 und die Instanziierung der F-Device-Objekte an ihnen instanziiert werden. Die Konfiguration der Module CM589-PNIO oder CM589-PNIO-4 und die Instanziierung der nicht sicherheitsgerichteten Prozessdaten werden separat erläutert in *[3]*. SM560-S-FD-1 und SM560-S-FD-4 können insgesamt bis zu 32 F-Device-Objekte mit einer maximalen Datengröße von 1400 Bytes (für jede Kommunikationsrichtung) verarbeiten.

> **HINWEIS!**
> Die PROFIsafe F_Dest_Add-Werte werden entsprechend der Reihenfolge den Instanzen im Automation Builder-Projekt zugewiesen (ein Vermischen ist nicht möglich). Die erwartete Basisadresse für diese Gruppe wird mit dem Adress-Drehschalter der Sicherheits-CPU bzw. dem im Projekt des Mastersystems konfigurierten F-Parameterwert festgelegt *Kapitel 3.1.2.5 „Adress-/Konfigurationsschalter-/F_Dest_Add-Einstellungen “ auf Seite 42*.

Nach der Instanziierung der Objekte „12 Byte In/Out (Safety)“ / „8 Byte and 2 Int In/Out (Safety)“ können Variablennamen für die instanziierten IN- und OUT-Sicherheitsdaten vergeben werden. Die Variablen können später im Anwendungsprogramm der Sicherheits-CPU verwendet werden, wenn die AC500-S Programming Tool-Sicherheitsinstanz geöffnet wurde. Um auf die Sicherheitsdaten im Programm der Sicherheits-CPU zugreifen zu können, müssen unbedingt symbolische Namen für die erforderlichen Sicherheitsdaten vergeben werden.

4. In jeder Konfiguration des Mastersystems müssen CM589-PNIO bzw. CM589-PNIO-4 unter CM579-PNIO instanziiert werden, um die PROFINET-Verbindung mit Slavesystemen herzustellen *[3]*. Ebenso muss die von CM589-PNIO-4 unterstützte PROFINET-Gerätefunktionalität „Shared Device“ berücksichtigt werden, wenn die Daten des Slavesystems mit mehr als einem (bis zu 4) anderen Steuerungssystem ausgetauscht werden sollen.

5. Ähnlich wie bei der Konfiguration des Slavesystems müssen an jedem Mastersystem die entsprechenden F-Device-Objekte instanziiert werden. Beachten Sie, dass die Reihenfolge der Objekte und ihr Typ in der Masterkonfiguration die gleiche wie in der Slavekonfiguration sein muss, anderenfalls ist ein Konfigurationsfehler im Modus RUN zu erwarten. Die Namen der instanziierten F-Device-Objekte können frei gewählt werden.

6. Durch Doppelklicken auf jedes instanziierte F-Device-Objekt müssen entsprechende F-Parameterwerte zugewiesen werden. F_Dest_Add muss für jedes instanziierte Objekt korrekt eingestellt sein.

> **HINWEIS!**
> Weitere Informationen finden Sie in den Regeln der Adresseinstellung von F_Dest_Add. Beachten Sie, dass nur ein Aufwärtszählen gemäß der Reihenfolge der Module im Objektbaum des Automation Builder erlaubt ist (das oberste Objekt hat den niedrigsten F_Dest_Add-Wert) *Kapitel 3.1.2.5 „Adress-/Konfigurationsschalter-/F_Dest_Add-Einstellungen “ auf Seite 42*.

Wir haben beispielsweise den Adress-Drehschalter an der Sicherheits-CPU des Slavesystems (SM560-S-FD-1 oder SM560-S-FD-4) auf den Wert 0x01 eingestellt. Somit liegt unser verfügbarer Bereich für F_Dest_Add bei 100 … 131 *Kapitel 3.1.2.5 „Adress-/Konfigurationsschalter-/F_Dest_Add-Einstellungen “ auf Seite 42*. Das erste Sicherheitsobjekt (F-Device-Objekt) muss die niedrigste Zahl 100 verwenden. Das zweite muss 101 verwenden und so weiter.

7. Für F_Source_Add können alle Werte des erlaubten Bereichs (1 – 511) verwendet werden. Allerdings ist zu beachten, ob das Slavesystem auch eine Masterfunktionalität aufweist, z. B. für Sicherheits-E/A-Module. Ist dies der Fall, ist es nicht erlaubt, die gleiche F_Source_Add für F-Device-Objekte wie F_Source_Add im Slavesystem für eigene F-Devices, z. B. Sicherheits-E/A-Module zu verwenden (mehr Details zu den Regeln für die Zuweisung von F_Source_Add und F_Dest_Add: *Kapitel 3.1.2.5 „Adress-/Konfigurationsschalter-/F_Dest_Add-Einstellungen “ auf Seite 42*).
8. Nach der Instanziierung der F-Device-Objekte in der Konfiguration des Mastersystems können Variablennamen für die instanziierten IN- und OUT-Sicherheitsdaten vergeben werden. Die Variablen können später im Anwendungsprogramm der Sicherheits-CPU verwendet werden. Um auf die Sicherheitsdaten im Programm der Sicherheits-CPU zugreifen zu können, müssen unbedingt symbolische Namen für die erforderlichen Sicherheitsdaten vergeben werden. Die symbolischen Variablennamen können frei gewählt werden, müssen jedoch eindeutig sein.
9. Wenn SM560-S-FD-4 als Teil der gemeinsamen PROFINET-Gerätekommunikation verwendet wird (siehe Dokumentation für CM589-PNIO-4 in *[3]*), um Sicherheitsdaten mit bis zu 4 Mastersystemen auszutauschen, so müssen an jedem Mastersystem die nicht verwendeten Sicherheitskommunikationsmodule getrennt werden. Dies ermöglicht die Auswahl, welches der konfigurierten F-Submodule („12 Byte In/Out (PROFIsafe V2.4)“ / „8 Byte and 2 Int In/Out (PROFIsafe V2.4)“ / „12 Byte In/Out (PROFIsafe V2.6)“ / „123 Byte In/Out (PROFIsafe V2.6)“) im Slavesystem mit welchem Mastersystem kommuniziert. Jedes instanziierte Sicherheitskommunikationsmodul kann nur eine Verbindung zu einem der Mastersysteme haben. Daher müssen alle Sicherheitskommunikationsmodule, die mit anderen Mastersystemen verbunden sind, mit dem Befehl „Modul trennen“ im Menü des Mastersystemprojekts auf „Getrennt“ gesetzt werden. Die getrennten Module erhalten einen grauen Hintergrund. Mit dem Befehl „Modul verbinden“ im Menü des gegebenen Kommunikationsmoduls kann man dieses wieder mit dem gegebenen Mastersystem verbinden.

HINWEIS!
Wenn das gleiche Sicherheitskommunikationsmodul mit mehr als einem Mastersystem verbunden ist, wird die Verbindung während der Start- und Parametrierphase nur mit dem schnellsten Mastersystem hergestellt. In diesem Fall erhalten andere Mastersysteme keine Daten. Stellen Sie sicher, dass alle konfigurierten Sicherheitskommunikationsmodule (F-Device-Objekte) richtig mit den Mastersystemen verbunden sind. Eine falsche Konfiguration kann zu Fehlermeldungen führen *Anhang B.2 „Fehlermeldungen mit AC500 Standard-CPU V2“ auf Seite 416* *Anhang C.2 „Fehlermeldungen mit AC500 V3-Standard-CPUs“ auf Seite 438*.

4.3.7 Überprüfen von Programm- und Systemkonfiguration

Programm und Systemkonfiguration überprüfen. *Kapitel 6.2 „Checkliste für die Erstellung von Sicherheitsprogrammen“ auf Seite 374* verwenden.

Es ist wichtig, dass Sie die Checkliste erfolgreich ausfüllen können und unterzeichnen. Ein Sicherheitsprogramm kann nur abgenommen werden, wenn die Checkliste ausgefüllt ist. Wenn einige Punkte der Checkliste nicht ausgefüllt werden können, muss eine angemessene Begründung im Abschnitt für Kommentare angegeben werden.

4.3.7.1 Überprüfen von Programm- und Systemkonfiguration mit Safety Verification Tool (SVT)

Im Automation Builder 2.3.x (und höher) ist ein Safety Verification Tool (SVT) integriert, das als Teil der Installation des Automation Builder mit dem AC500-S-Softwarepaket installiert wird.

SVT prüft die AC500-S-Sicherheitskonfiguration im Automation Builder und erzeugt eine SVT-Checkliste, die AC500-S-Anwender für die manuelle Fertigstellung der funktionalen Sicherheitsprüfung des Automation Builder-Projekts verwenden sollen.

GEFAHR!
SVT ist für die Verwendung mit Automation Builder 2.3.x (und höher) zwingend erforderlich.

In Automation Builder 2.2.x und früheren Versionen war die Verwendung von SVT aufgrund anderer Verfahren zur Prüfung der funktionalen Sicherheitsintegrität des Automation Builder-Projekts nicht erforderlich.

Verwenden Sie SVT, um zu prüfen, ob das AC500-S Programming Tool-Projekt mit Ihrem Sicherheitsprojekt im Automation Builder übereinstimmt.

4.3.7.1.1 Funktionalität

SVT liest die IEC 61131-Programmobjekte aus dem mit AC500-S Programming Tool erstellten Sicherheitsprojekt und die Beschreibungsdateien für Sicherheitsgeräte im Automation Builder, überprüft die Daten aus beiden Quellen und erstellt die SVT-Checkliste. Die SVT-Checkliste ist eine Textdatei, die mit jedem beliebigen Texteditor geöffnet und bei Bedarf ausgedruckt werden kann. In den folgenden Abbildungen finden Sie Beispiele für SVT-Checklisten.

Die SVT-Checkliste umfasst mehrere Abschnitte:

- Einen Abschnitt mit Projektinformationen, der allgemeine Informationen zum Sicherheitsprojekt enthält *„Abschnitt mit Projektinformationen“ auf Seite 182*.
- Abschnitte für jedes Sicherheitsgerät im Sicherheitsprojekt *„Abschnitte zu Sicherheitsgeräten“ auf Seite 182*.
- Einen Abschnitt für die Sicherheits-CPU im Sicherheitsprojekt *„Abschnitt zur Sicherheits-CPU“ auf Seite 186*.
- Einen Abschnitt für die verwendeten Bibliotheken *„Bibliotheksabschnitt“ auf Seite 186*.

SVT prüft beispielsweise Folgendes:

- Die Integrität der globalen Variablen für das E/A-Abbild für jedes Sicherheitsgerät im Sicherheitsprojekt.
- Die Integrität der zugeordneten E/A-Variablen mit einer Beschreibung der E/A-Struktur.
- Die Prüfsumme der F-Parameter für jedes Sicherheitsgerät.
- Die Integrität der F-Parameter mit F-Parameter-Beschreibung.

GEFAHR!
Zusätzlich zu den erfolgreich bestandenen automatischen Prüfungen müssen Sie alle manuellen Prüfungen der SVT-Checkliste erfolgreich abschließen.

HINWEIS!
Verwenden Sie SVT im finalen Automation Builder-Projekt. Anschließend werden keine weiteren Änderungen im auf die funktionale Sicherheit bezogenen Projektteil erwartet, die zu einer neuen Bootprojekt-CRC führen würden.

Abschnitt mit Projektinformationen

Die SVT-Checkliste beginnt mit einem Abschnitt für die manuelle Prüfung von Informationen in Bezug auf das gesamte Sicherheitsprojekt.

```
################################################################################################################################
#                                                                                                                              #
# SVT (Safety Verification Tool) Checkliste                                                                                    #
#                                                                                                                              #
################################################################################################################################

GEFAHR! Sie müssen in funktionaler Sicherheit geschult sein, um an funktionalen Safety-Geräten zu arbeiten. Lesen und
verstehen Sie das AC500-S Sicherheitshandbuch und andere relevante Dokumente bevor Sie SVT benutzen. Weitere
Informationen finden Sie unter www.abb.com/plc.
Diese SVT-Checkliste wurde durch das Safety Verification Tool generiert. Benutzen Sie es um die Integrität Ihres
Safety-Projektes zu überprüfen. Es enthält die Ergebnisse der automatischen Überprüfungen durch das SVT und führt die
verwendeten Safety-Geräte des Projektes zur manuellen Überprüfung auf. Stellen Sie sicher, dass alle verwendeten
Safety-Geräte enthalten und deren Daten korrekt sind. Archivieren Sie die SVT-Checkliste für die weitere Nutzung, wenn
alle Überprüfungen erfolgreich waren.

(1)
Generiert am:     23.03.2022 14:29:11
SVT Version:      1.2.0.606

(2)
Die automatischen Überprüfungen sind erfolgreich.

(3)
Öffnen Sie in Ihrem Safety-Projekt die "Projektinformation..." im Menü "Projekt" des AC500-S Programming Tools.
[  ] Das "Verzeichnis" und der "Dateiname" sind identisch zum SVT-Checklisten Projektverzeichnis und Dateiname:
     C:\Users\Test\AppData\Local\Temp\CoDeSys\E2021EC2519AF8CB71B0445C370793FC__ecac4eb4-f666-4b1a-aa67-5c94b4199781\
     AC500_S.AC500PRO
[  ] Das Änderungsdatum ("Geändert am") ist identisch zum SVT-Checklisten Projektdatum: 03.02.2022 09:26:20
[  ] Im AC500-S Programming Tool, überprüfen Sie anhand des Menüs "Online" / "Prüfe Bootprojekt der Steuerung", dass
     das Safety Projekt und das Boot-Projekt auf der Safety-CPU identisch sind und geben Sie die BOOT-Projekt-CRC hier
     ein:
     _____

Die im Safety-Projekt gespeicherte "Projektinformation ..." (nur zur Information)
Titel:
Autor:
Version:
Beschreibung:

(4)
SVT Datenchecksumme:

36ba 1080 1f7f 7186 00c7 eecc 6e58 1079

Ist die SVT-Datenchecksumme identisch zu der SVT-Datenchecksumme von der vorherigen freigegebenen und gültigen
SVT-Checkliste?
[  ] Falls ja, ist die Automation Builder Projektkonfiguration identisch. Sie können die manuelle Überprüfung
     überspringen und die vorherige freigegebene und gültige SVT-Checkliste mit der identischen SVT-Datenchecksumme
     weiterhin nutzen.
[  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen durch.

(5)
Überprüfen Sie, dass alle konfigurierten Safety-Geräte des Automation Builder Projektes unten aufgeführt sind und dass
jedes von ihnen einen Abschnitt mit dem gleichen Namen in der SVT-Checkliste hat:
[  ] 1. AI581_S
[  ] 2. DI581_S
[  ] 3. DX581_S
[  ] Alle Safety-Geräte aus dem Automation Builder Projekt sind in der SVT-Checkliste enthalten.
[  ] Die SVT-Checkliste hat 'Ende der SVT-Checkliste' als letzte Textzeile.
```

Abb. 83: Beispiel für einen Abschnitt mit Projektinformationen in einer SVT-Checkliste

1 Zeitstempel und Versionsinformationen
2 Ergebnis der von SVT durchgeführten automatischen Konsistenzprüfungen
3 Bezug zum Sicherheitsprojekt
4 Datenprüfsumme für die gesamte SVT-Checkliste
5 Liste der Sicherheitsgeräte im Sicherheitsprojekt

Abschnitte zu Sicherheitsgeräten

Auf den Abschnitt mit Projektinformationen folgen in der SVT-Checkliste einzelne Abschnitte für jedes Sicherheitsgerät im Sicherheitsprojekt. Der Inhalt der einzelnen Abschnitte zu den Sicherheitsgeräten hängt vom Typ des jeweiligen Sicherheitsgerätes ab.

ABB-Sicherheitsgeräte

```
##############################################################################################################################
#                                                                                                                            #
# 1. DX581_S                                                                                                                 #
#                                                                                                                            #
##############################################################################################################################
```

(1)
```
Die automatischen Überprüfungen für dieses Safety-Gerät sind erfolgreich.
```

(2)
```
Safety-Gerät Datenchecksumme:

af76 7c66 3dfd 27be 33ca 1db6 f652 5a24

Ist die Safety-Gerätedatenchecksumme identisch zu der Safety-Gerätedatenchecksumme von der vorherigen freigegebenen und
gültigen SVT-Checkliste?
[  ] Falls ja, ist die Safety-Gerätekonfiguration identisch. Sie können die manuelle Überprüfung dieses Safety-Gerätes
     überspringen.
[  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen für das Safety-Gerät durch.
```

(3)
```
[  ] Öffnen Sie im Automation Builder Projekt das "Information" Tab von dem Safety-Gerät und überprüfen Sie in der
     oberen, linken Ecke, dass die Gerätetypbeschreibung identisch zu der SVT Gerätetypbeschreibung 'DX581-S' ist.
```

(4)
```
Öffnen Sie im Automation Builder Projekt das E/A Abbild von dem Safety Gerät und überprüfen Sie, dass alle Kanäle und
Variablen aus der nachfolgenden Tabelle identisch sind.
Die Spalten Datentyp und E/A (Eingang oder Ausgang) dienen nur der Information.
=============================================================================
|    Variable          | Kanal                              | Datentyp | E/A     |
=============================================================================
[  ]                   | Sichere Digitaleingänge I0 - I7    |          | Eingang |
[  ] IS_Estop1         |   Sicherer Digitaleingang I0       | BOOL     | Eingang |
[  ]                   |   Sicherer Digitaleingang I1       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I2       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I3       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I4       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I5       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I6       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I7       |          | Eingang |
-----------------------------------------------------------------------------
-
-
-
-----------------------------------------------------------------------------
```

(5)
```
Öffnen Sie im Automation Builder Projekt das "F-Parameter" Tab von dem Safety-Gerät und überprüfen Sie, dass die
F-Parameter Werte aus der nachfolgenden Tabelle identisch sind.
================================
|    F-Parameter   | Wert      |
================================
[  ] F_Check_SeqNr | 1         |
[  ] F_Check_iPar  | 0         |
[  ] F_SIL         | 2         |
[  ] F_CRC_Length  | 0         |
[  ] F_Block_ID    | 1         |
[  ] F_Par_Version | 1         |
[  ] F_Source_Add  | 1         |
[  ] F_Dest_Add    | 2         |
[  ] F_WD_Time     | 100       |
[  ] F_iPar_CRC    | 1455424635 |
[  ] F_Par_CRC     | 44422     |
--------------------------------
```

Abb. 84: Beispiel eines Sicherheitsgeräte-Abschnitts für Sicherheits-E/A-Modul DX581-S

1 Ergebnis der von SVT durchgeführten automatischen Konsistenzprüfungen
2 Datenprüfsumme für den Sicherheitsgeräte-Abschnitt
3 Beschreibung des Sicherheitsgerätetyps
4 Eingangs- und Ausgangs-Verknüpfungsliste für das Sicherheitsgerät
5 Liste der F-Parameter für das Sicherheitsgerät

F-Devices in AC500-S-Sicherheits-CPUs

F-Devices in den AC500-S-Sicherheits-CPUs SM560-S-FD-1 und SM560-S-FD-4 enthalten auch einen Abschnitt mit Informationen zur Position des Sicherheitsgerätes im Sicherheitsprojekt im Automation Builder.

```
################################################################################################################################
#                                                                                                                              #
# 2. _12_Byte_In_Out_Safety                                                                                                    #
#                                                                                                                              #
################################################################################################################################

Die automatischen Überprüfungen für dieses Safety-Gerät sind erfolgreich.

Safety-Gerät Datenchecksumme:

dc46 81e0 957f 6a3e 6628 c770 cae5 3adf

Ist die Safety-Gerätedatenchecksumme identisch zu der Safety-Gerätedatenchecksumme von der vorherigen freigegebenen und
gültigen SVT-Checkliste?
[  ] Falls ja, ist die Safety-Gerätekonfiguration identisch. Sie können die manuelle Überprüfung dieses Safety-Gerätes
     überspringen.
[  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen für das Safety-Gerät durch.

[  ] Öffnen Sie im Automation Builder Projekt das "Information" Tab von dem Safety-Gerät und überprüfen Sie in der
     oberen, linken Ecke, dass die Gerätetypbeschreibung identisch zu der SVT Gerätetypbeschreibung '12 Byte Ein-/
     Ausgänge (Safety)' ist.

(1) [  ] Überprüfen Sie, dass die Position des Safety-Gerätes (Standard E/A-Module und getrennte Safety-Geräte werden
         ignoriert) unter allen CM589-PNIO(-4) Knoten 1 ist.

Öffnen Sie im Automation Builder Projekt das E/A Abbild von dem Safety Gerät und überprüfen Sie, dass alle Kanäle und
Variablen aus der nachfolgenden Tabelle identisch sind.
Die Spalten Datentyp und E/A (Eingang oder Ausgang) dienen nur der Information.
=================================================
|    Variable    | Kanal           | Datentyp | E/A     |
=================================================
[  ] OS_CommOut1 | Ausgang Byte 0  | BYTE     | Ausgang |
[  ]             |   Bit 0         |          | Ausgang |
[  ]             |   Bit 1         |          | Ausgang |
[  ]             |   Bit 2         |          | Ausgang |
[  ]             |   Bit 3         |          | Ausgang |
[  ]             |   Bit 4         |          | Ausgang |
[  ]             |   Bit 5         |          | Ausgang |
[  ]             |   Bit 6         |          | Ausgang |
[  ]             |   Bit 7         |          | Ausgang |
-------------------------------------------------------
-
-
-
-------------------------------------------------------
```

Abb. 85: Beispiel eines Sicherheitsgeräte-Abschnitts für ein F-Device in AC500-S-Sicherheits-CPUs

1 Position des Sicherheitsgerätes im Sicherheitsprojekt im Automation Builder unter allen Knoten von CM589-PNIO(-4)

Sicherheitsgeräte von Drittanbietern

Die Abschnitte zu Sicherheitsgeräten von Drittanbietern enthalten auch die Modul-ID sowie Informationen zur GSDML-Datei in der SVT-Checkliste.

```
##################################################################################################################################
#                                                                                                                                #
# 3. SIO_02_02                                                                                                                   #
#                                                                                                                                #
##################################################################################################################################

Die automatischen Überprüfungen für dieses Safety-Gerät sind erfolgreich.

Safety-Gerät Datenchecksumme:

77b8 c99c 48b1 76e9 b7e2 7caa 00d0 4fb7

Ist die Safety-Gerätedatenchecksumme identisch zu der Safety-Gerätedatenchecksumme von der vorherigen freigegebenen und
gültigen SVT-Checkliste?
[  ] Falls ja, ist die Safety-Gerätekonfiguration identisch. Sie können die manuelle Überprüfung dieses Safety-Gerätes
     überspringen.
[  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen für das Safety-Gerät durch.

(1) Öffnen Sie im Automation Builder Projekt das "Information" Tab von dem Safety-Gerät und überprüfen Sie folgende Angaben:
[  ] Die Gerätetypbeschreibung in der oberen, linken Ecke ist identisch zu der SVT Gerätetypbeschreibung: SIO 02/02
[  ] Die Modul-ID ist identisch zu der SVT Modul-ID: sdio3-2_x1x2
(2) Überprüfen Sie, dass die in der SVT-Checkliste aufgeführte GSDML-Datei identisch ist zu der erwarteten Version des
Safety-Geräteherstellers.
[  ] C:\ProgramData\AutomationBuilder\AB_Devices_2.3\81\0x00B0_0x0100_DIM 1\SW%3D%2C HW%3D\
     GSDML-V2.3-Phoenix Contact-FL PN PN SDIO 2TX 2TX X1-X2-V1.1-20130408.xml

Öffnen Sie im Automation Builder Projekt das E/A Abbild von dem Safety Gerät und überprüfen Sie, dass alle Kanäle und
Variablen aus der nachfolgenden Tabelle identisch sind.
Die Spalten Datentyp und E/A (Eingang oder Ausgang) dienen nur der Information.
=================================================
|    Variable | Kanal           | Datentyp | E/A     |
=================================================
[  ] IS_Byte1 | SI_0            | USINT    | Eingang |
[  ]          |   Bit0          |          | Eingang |
[  ]          |   Bit1          |          | Eingang |
[  ]          |   Bit2          |          | Eingang |
[  ]          |   Bit3          |          | Eingang |
[  ]          |   Bit4          |          | Eingang |
[  ]          |   Bit5          |          | Eingang |
[  ]          |   Bit6          |          | Eingang |
[  ]          |   Bit7          |          | Eingang |
-------------------------------------------------
-
-
-
-------------------------------------------------

Öffnen Sie im Automation Builder Projekt das "F-Parameter" Tab von dem Safety-Gerät und überprüfen Sie, dass die
F-Parameter Werte aus der nachfolgenden Tabelle identisch sind.
==========================
|    F-Parameter   | Wert  |
==========================
[  ] F_SIL         | 2     |
[  ] F_Block_ID    | 0     |
[  ] F_Par_Version | 1     |
[  ] F_Source_Add  | 1     |
[  ] F_Dest_Add    | 11    |
[  ] F_WD_Time     | 150   |
[  ] F_Par_CRC     | 37844 |
--------------------------
```

Abb. 86: Beispiel eines Sicherheitsgeräte-Abschnitts für ein Sicherheitsgerät von einem Drittanbieter

1 Modul-ID
2 Informationen zur GSDML-Datei

Abschnitt zur Sicherheits-CPU

Wie der Sicherheitsgeräte-Abschnitt umfasst auch der Sicherheits-CPU-Abschnitt Informationen zu den automatischen Prüfungen, der Datenprüfsumme und den manuellen Prüfungen für die Sicherheits-CPU.

```
##############################################################################################################################
#                                                                                                                            #
#  Safety-CPU                                                                                                                #
#                                                                                                                            #
##############################################################################################################################

(1) Die automatischen Überprüfungen für dieses Safety-Gerät sind erfolgreich.

    Safety-Gerät Datenchecksumme:

    c987 217b 78dd 4405 6a9d a58c f06b 8c7a

(2) Ist die Safety-Gerätedatenchecksumme identisch zu der Safety-Gerätedatenchecksumme von der vorherigen freigegebenen und
    gültigen SVT-Checkliste?
    [  ] Falls ja, ist die Safety-Gerätekonfiguration identisch. Sie können die manuelle Überprüfung dieses Safety-Gerätes
         überspringen.
    [  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen für das Safety-Gerät durch.

(3) [  ] Öffnen Sie im Automation Builder Projekt das "CPU Parameter" Tab der Safety-CPU und überprüfen Sie, dass der
         konfigurierte Parameter "PROFIsafe startup timeout" identisch zu 0 ms ist, welcher aktuell im Safety-Projekt
         eingestellt ist.
```

Abb. 87: Beispiel eines Sicherheitsgeräte-Abschnitts für die Sicherheits-CPU

1 Ergebnis der von SVT durchgeführten automatischen Konsistenzprüfungen
2 Datenprüfsumme für den Sicherheits-CPU-Abschnitt
3 Parameter „PROFIsafe startup timeout“ der Sicherheits-CPU

Bibliotheksabschnitt

Der Bibliotheksabschnitt umfasst eine Datenprüfsumme zur Angabe von Veränderungen für die verwendeten Sicherheitsbibliotheken und die CRCs der verwendeten Sicherheitsbibliotheken.

```
##############################################################################################################################
#                                                                                                                            #
#  Bibliotheken                                                                                                              #
#                                                                                                                            #
##############################################################################################################################

(1) Bibliotheken Datenchecksumme:

    f7ba ca5d 042c aaa1 82a6 fe3c 10ea 0371

    Ist die Bibliotheken Datenchecksumme identisch zu der Bibliotheken Datenchecksumme von der vorherigen freigegebenen und
    gültigen SVT-Checkliste?
    [  ] Falls ja, sind die Bibliotheken identisch. Sie können die manuelle Überprüfung der Bibliotheken überspringen.
    [  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen für die Bibliotheken durch.

    Überprüfen Sie, dass die in diesem Abschnitt aufgeführten Bibliotheks-CRCs identisch sind zu denen im Abschnitt
    "AC500-S-Bibliotheken" des AC500-S Sicherheitshandbuchs. Alle weiteren Benutzer-spezifischen Bibliotheken und deren
    CRCs, welche nicht im AC500-S Sicherheitshandbuch aufgeführt sind, müssen separat überprüft und durch den Endkunden
    bestätigt werden um sie für Safety-spezifische Applikationen zu qualifizieren.
(2) ============================================================
    |    Bibliothek                                 | CRC Wert |
    ============================================================
    [  ] Safety_Standard.lib                        | fd5d3581 |
    [  ] Safety_SysLibTime.lib                      | 672b8325 |
    [  ] SafetyBase_PROFIsafe_LV210_AC500_V22.lib   | 8069df7b |
    [  ] SafetyBlocks_PLCopen_AC500_V22.lib         | b6e0bc60 |
    [  ] SafetyExt2_LV110_AC500_V27.lib             | aa3be9be |
    [  ] SafetyExt_AC500_V22.lib                    | 72a88162 |
    [  ] SafetyUtil_CoDeSys_AC500_V22.lib           | 6b29c54  |
    [  ] SYSLIBCALLBACK.LIB                         | 62ad210d |
    [  ] Target_AC500_V22.lib                       | 8daa436  |
    ------------------------------------------------------------
```

Abb. 88: Beispiel für den Bibliotheksabschnitt

1 Datenprüfsumme für die Sicherheitsbibliotheken
2 Bibliotheks-CRCs

Ende der SVT-Checkliste

Nach dem Bibliotheksabschnitt endet die SVT-Checkliste mit der Zeile `Ende der SVT-Checkliste`. Darauf folgen optionale Felder wie Datum, Signatur usw.

```
----------------------------------------
Ende der SVT-Checkliste
----------------------------------------

Optionale Felder:

  Reviewer:

  Maschine/Applikation <ID>:

  Datum:

  Unterschrift:
```

Abb. 89: Ende der SVT-Checkliste mit optionalen Feldern

4.3.7.1.2 SVT ausführen

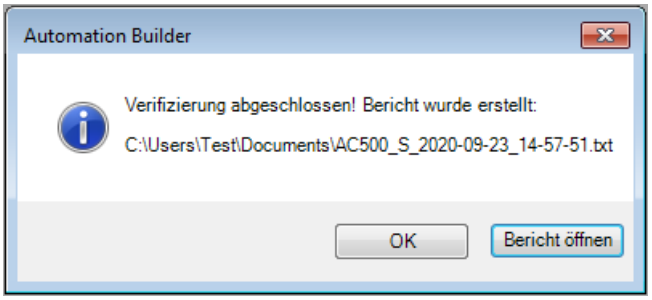
1. Navigieren Sie im Automation Builder zum Knoten der Sicherheits-CPU-Anwendung, z. B. *„AC500_S“*.
2. Führen Sie einen Rechtsklick auf den Knoten aus, um das Kontextmenü zu öffnen.
3. Wählen Sie *„Sicherheits-Konfigurationsdaten erzeugen“*.
4. Wählen Sie *„Sicherheitsprojektintegrität verifizieren“*.

> **HINWEIS!**
> Der Befehl *„Sicherheitsprojektintegrität verifizieren“* ist ggf. nicht aktiv, z. B. wenn das Sicherheitsprojekt geöffnet ist oder wenn Sie den Befehl *„Sicherheits-Konfigurationsdaten erzeugen“* nicht zuvor ausgeführt haben.
>
> Speichern und schließen Sie das Sicherheitsprojekt, bevor Sie SVT verwenden.

5. Wenn Sie mit passwortgeschützten Sicherheitsprojekten arbeiten, fordert Automation Builder das Passwort an ↬ *„Der Zugriff auf die Sicherheits-CPU und das Sicherheitsprogramm kann durch drei Passwörter geschützt werden.“ auf Seite 147*.

   ⇨ Während der Ausführung von SVT ist die Benutzeroberfläche des Automation Builder deaktiviert. Bei großen Sicherheitsprojekten kann dieser Vorgang mehrere Minuten dauern.

   Nach Abschluss der SVT-Ausführung zeigt Automation Builder eine Meldung darüber an, dass die SVT-Prüfung abgeschlossen wurde.



   Die Meldung zeigt den Pfad sowie den Namen der SVT-Checkliste. Der Dateiname enthält den Namen des Anwendungsknotens für die AC500-S-Sicherheits-CPU sowie das Datum und die Uhrzeit der SVT-Ausführung. Das Datum wird im ISO-Format (JJJJ-MM-TT) und die Uhrzeit im Stunden-Minuten-Sekunden-Format (hh-mm-ss) angegeben.

Die SVT-Checkliste enthält eine Datenprüfsumme für die gesamte Dokumentdatei sowie eine Datenprüfsumme für jedes Sicherheitsgerät im Sicherheitsprojekt.

Sie können die Datenprüfsummen verwenden, um zu prüfen, ob sich das Sicherheitsprojekt geändert hat. Wenn alle Datenprüfsummen in der SVT-Checkliste identisch sind, haben keine Änderungen in Bezug auf das Sicherheitsprojekt stattgefunden und Sie müssen die manuellen Prüfungen nicht wiederholen.

Das Upgraden eines Projekts auf eine neuere Version von Automation Builder kann zu einer veränderten SVT Datenprüfsumme *Handlungsschritt 3 auf Seite 189* führen.

HINWEIS!
Sie können SVT beliebig oft ausführen, um Ihr Sicherheitsprojekt zu prüfen. Es wird empfohlen, die für die abschließende Projektrevision verwendeten SVT-Checklisten zu archivieren. Sie können die archivierten SVT-Checklisten bei der Prüfung der Änderungen an Ihren Sicherheitsprojekten als Referenz verwenden. Mithilfe einer archivierten und verifizierten SVT-Checkliste können Sie Abschnitte und Sicherheitsgeräte, die Sie bereits geprüft haben, überspringen, wenn sich die Datenprüfsummen nicht geändert haben.

Sie können auch die manuellen Prüfungen für Abschnitte überspringen, die im Vergleich zur vorherigen validierten Version der SVT-Checkliste identische Datenprüfsummen aufweisen. Die manuellen Prüfungen sind nur für Abschnitte der SVT-Checkliste erforderlich, die eine abweichende Datenprüfsumme aufweisen.

Bei großen Sicherheitsprojekten können Sie ein entsprechendes Software-Tool verwenden, um zwei Textversionen der SVT-Checkliste zu vergleichen, um jegliche Unterschiede zu ermitteln.

4.3.7.1.3 SVT-Checkliste verifizieren

HINWEIS!
Die Auszüge einer SVT-Checkliste in diesem Abschnitt dienen nur als Beispiel und wurden für den entsprechenden Zweck bearbeitet. Ihre SVT-Checkliste kann abhängig von den Versionen von Automation Builder und SVT hiervon abweichen.

Lesen Sie die SVT-Checkliste sorgfältig durch und markieren Sie das entsprechende Ankreuzfeld für jeden Abschnitt und jede Frage in der SVT-Checkliste, sofern das Ergebnis Ihrer Prüfung positiv ist.

1. Verifizieren Sie den Abschnitt mit Projektinformationen *„Abschnitt mit Projektinformationen verifizieren“ auf Seite 189*.
2. Verifizieren Sie jeden Abschnitt zu Sicherheitsgeräten *„Abschnitte zu Sicherheitsgeräten verifizieren“ auf Seite 190*.
3. Verifizieren Sie den Abschnitt zur Sicherheits-CPU *„Sicherheits-CPU-Abschnitt verifizieren“ auf Seite 192*.
4. Verifizieren Sie den Bibliotheksabschnitt *„Bibliotheksabschnitt verifizieren“ auf Seite 192*.
5. Verifizieren Sie das Ende der SVT-Checkliste *„Ende der SVT-Checkliste verifizieren“ auf Seite 193*.

Wenn das Ergebnis Ihrer Prüfung für mindestens eine der manuellen Prüfungen in der SVT-Checkliste negativ oder nicht akzeptabel ist, stellen Sie sicher, dass die Sicherheits-Konfigurationsdaten auf dem neuesten Stand sind. Wenn die Probleme weiter bestehen, wenden Sie sich an den technischen Support von ABB, um Unterstützung zu erhalten.

Jeder Abschnitt der SVT-Checkliste beginnt mit einer Kopfzeile. Das Ende der SVT-Checkliste wird durch folgende Zeichenfolge angezeigt:

```
----------------------------------------
Ende der SVT-Checkliste
----------------------------------------
```

Abschnitt mit Projektinformationen verifizieren

Dieser Abschnitt enthält allgemeine Informationen zur SVT-Checkliste. Er beginnt mit dem Zeitstempel und der Version von SVT. Beispiel für einen Abschnitt mit Projektinformationen: Abb. 83, Seite 182.

1. Prüfen Sie, ob die von SVT durchgeführten automatischen Prüfungen bestanden wurden:

```
Die automatischen Überprüfungen sind erfolgreich.
```

   ⇨ Wenn die automatischen Prüfungen Fehler erzeugen, wird eine Fehlermeldung angezeigt *„Fehler bei den automatischen Prüfungen“ auf Seite 193*:

```
Die automatischen Überprüfungen sind fehlgeschlagen.
```

2. Prüfen Sie in AC500-S Programming Tool, ob die Projektinformationen korrekt sind. Kennzeichnen Sie die positive Verifizierung eines Punktes in der SVT-Checkliste mit einem „X“:

```
Öffnen Sie in Ihrem Safety-Projekt die "Projektinformation..." im Menü "Projekt" des AC500-S Programming Tools.
[ X ] Das "Verzeichnis" und der "Dateiname" sind identisch zum SVT-Checklisten Projektverzeichnis und Dateiname:
      C:\Users\Test\AppData\Local\Temp\CoDeSys\E2021EC2519AF8CB71B0445C370793FC__ecac4eb4-f666-4b1a-aa67-5c94b4199781\
      AC500_S.AC500PRO
[   ] Das Änderungsdatum ("Geändert am") ist identisch zum SVT-Checklisten Projektdatum: 03.02.2022 09:26:20
[   ] Im AC500-S Programming Tool, überprüfen Sie anhand des Menüs "Online" / "Prüfe Bootprojekt der Steuerung", dass
      das Safety Projekt und das Boot-Projekt auf der Safety-CPU identisch sind und geben Sie die BOOT-Projekt-CRC hier
      ein:
      _____________________________________________
```

   ⇨

> **HINWEIS!**
>
> Markieren Sie das entsprechende Ankreuzfeld für jede Frage in der SVT-Checkliste (siehe Beispiel oben). Sie können die bestätigten Punkte in einem Ausdruck oder in der Textdatei kennzeichnen.

3. Lesen Sie die Datenprüfsumme für die gesamte SVT-Checkliste.

   Verwenden Sie diese Datenprüfsumme zur Verifizierung von Änderungen in der gesamten SVT-Checkliste.

   Überprüfen Sie, ob die Datenprüfsumme identisch mit der Datenprüfsumme aus der vorherigen validierten SVT-Checkliste ist. Wenn diese SVT-Datenprüfsummen identisch sind, müssen Sie keine manuellen Prüfungen durchführen.

   Wenn die Datenprüfsummen nicht identisch sind oder Sie SVT erstmalig ausführen, fahren Sie mit den manuellen Prüfungen in der SVT-Checkliste fort.

```
SVT Datenchecksumme:

5f79 612c cb36 4ee7 4cb5 5202 4b6c 47bd

Ist die SVT-Datenchecksumme identisch zu der SVT-Datenchecksumme von der vorherigen freigegebenen und gültigen
SVT-Checkliste?
[   ] Falls ja, ist die Automation Builder Projektkonfiguration identisch. Sie können die manuelle Überprüfung
      überspringen und die vorherige freigegebene und gültige SVT-Checkliste mit der identischen SVT-Datenchecksumme
      weiterhin nutzen.
[   ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen durch.
```

4. Prüfen Sie, ob alle Sicherheitsgeräte im Automation Builder-Projekt in der SVT-Checkliste aufgeführt sind.

   Wenn ein Sicherheitsgerät nicht in der Liste aufgeführt ist, wählen Sie im Automation Builder *„Sicherheits-Konfigurationsdaten erzeugen“* und führen Sie SVT erneut aus. Nur konfigurierte und angeschlossene Sicherheitsgeräte werden im SVT aufgeführt, da alle getrennten Geräte außerhalb des betreffenden Projekts gehandhabt werden.

   ```
   Überprüfen Sie, dass alle konfigurierten Safety-Geräte des Automation Builder Projektes unten aufgeführt sind und dass
   jedes von ihnen einen Abschnitt mit dem gleichen Namen in der SVT-Checkliste hat:
   [  ] 1. DX581_S
   [  ] 2. _12_Byte_In_Out_Safety
   [  ] 3. SIO_02_02
   [  ] Alle Safety-Geräte aus dem Automation Builder Projekt sind in der SVT-Checkliste enthalten.
   [  ] Die SVT-Checkliste hat 'Ende der SVT-Checkliste' als letzte Textzeile.
   ```

   ⇨ Auf die Zeile `Ende der SVT-Checkliste` folgen optionale Felder, z. B. Datum, Signatur usw.

5. Setzen Sie die Verifizierung des Inhalts jedes Sicherheitsgeräte-Abschnitts fort.

Abschnitte zu Sicherheitsgeräten verifizieren

Jedes Sicherheitsgerät weist einen separaten Abschnitt in der SVT-Checkliste auf, der mit einer Kopfzeile mit dem Namen des Sicherheitsgerätes beginnt. Die Informationen in jedem Sicherheitsgeräte-Abschnitt hängen vom Typ des betreffenden Sicherheitsgerätes ab *„Abschnitte zu Sicherheitsgeräten“ auf Seite 182*.

1. Prüfen Sie, ob die für das betreffende Sicherheitsgerät von SVT durchgeführten automatischen Prüfungen bestanden wurden:

   ```
   Die automatischen Überprüfungen für dieses Safety-Gerät sind erfolgreich.
   ```

   ⇨ Wenn die automatischen Prüfungen Fehler erzeugen, wird eine Fehlermeldung angezeigt *„Fehler bei den automatischen Prüfungen“ auf Seite 193*:

   ```
   Die automatischen Überprüfungen für dieses Safety-Gerät sind fehlgeschlagen.
   ```

2. Lesen Sie die Datenprüfsumme für das Sicherheitsgerät.

   Verwenden Sie diese Datenprüfsumme zur Verifizierung von Änderungen der Daten für dieses Sicherheitsgerät. Wenn die Datenprüfsumme identisch mit der Datenprüfsumme aus einer vorherigen validierten SVT-Checkliste ist, sind die Daten für dieses Sicherheitsgerät identisch, sodass Sie die manuellen Prüfungen dafür überspringen können. Sind die Datenprüfsummen nicht identisch, wiederholen Sie alle manuellen Prüfungen für das Sicherheitsgerät.

   ```
   SVT Datenchecksumme:

   5f79 612c cb36 4ee7 4cb5 5202 4b6c 47bd

   Ist die SVT-Datenchecksumme identisch zu der SVT-Datenchecksumme von der vorherigen freigegebenen und gültigen
   SVT-Checkliste?
   [  ] Falls ja, ist die Automation Builder Projektkonfiguration identisch. Sie können die manuelle Überprüfung
        überspringen und die vorherige freigegebene und gültige SVT-Checkliste mit der identischen SVT-Datenchecksumme
        weiterhin nutzen.
   [  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen durch.
   ```

3. Verifizieren Sie die Beschreibung des Gerätetyps. Nur für Drittanbietergeräte muss auch die Modul-ID überprüft werden.

   ```
   Öffnen Sie im Automation Builder Projekt das "Information" Tab von dem Safety-Gerät und überprüfen Sie folgende Angaben:
   [  ] Die Gerätetypbeschreibung in der oberen, linken Ecke ist identisch zu der SVT Gerätetypbeschreibung: SIO 02/02
   [  ] Die Modul-ID ist identisch zu der SVT Modul-ID: sdio3-2_x1x2
   ```

4. Verifizieren Sie für Drittanbietergeräte, dass die in der SVT-Checkliste angezeigte GSDML-Datei identisch mit der erwarteten Version vom Hersteller des Sicherheitsgerätes ist.

   ```
   Überprüfen Sie, dass die in der SVT-Checkliste aufgeführte GSDML-Datei identisch ist zu der erwarteten Version des
   Safety-Geräteherstellers.
   [  ] C:\ProgramData\AutomationBuilder\AB_Devices_2.3\81\0x00B0_0x0100_DIM 1\SW%3D%2C HW%3D\
        GSDML-V2.3-Phoenix Contact-FL PN PN SDIO 2TX 2TX X1-X2-V1.1-20130408.xml
   ```

5. Prüfen Sie ggf., ob die Positionsnummer des Sicherheitsgerätes in der SVT-Checkliste seiner Position im Sicherheitsprojekt im Automation Builder entspricht. Die Positionsnummer für das betreffende Sicherheitsgerät kann sich ändern, wenn die CM589-PNIO(-4)-Knoten im Projekt verschoben werden.

```
[  ] Überprüfen Sie, dass die Position des Safety-Gerätes (Standard E/A-Module und getrennte Safety-Geräte werden
     ignoriert) unter allen CM589-PNIO(-4) Knoten 1 ist.
```

6. Überprüfen Sie die E/A-Abbild-Informationen für das Sicherheitsgerät.

   Beachten Sie, dass *„Datentyp“* und *„E/A“* ausschließlich zu Informationszwecken aufgeführt sind.

```
Öffnen Sie im Automation Builder Projekt das E/A Abbild von dem Safety Gerät und überprüfen Sie, dass alle Kanäle und
Variablen aus der nachfolgenden Tabelle identisch sind.
Die Spalten Datentyp und E/A (Eingang oder Ausgang) dienen nur der Information.
================================================================================
|    Variable          | Kanal                              | Datentyp | E/A     |
================================================================================
[  ]                   | Sichere Digitaleingänge I0 - I7    |          | Eingang |
[  ] IS_Estop1         |   Sicherer Digitaleingang I0       | BOOL     | Eingang |
[  ]                   |   Sicherer Digitaleingang I1       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I2       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I3       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I4       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I5       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I6       |          | Eingang |
[  ]                   |   Sicherer Digitaleingang I7       |          | Eingang |
--------------------------------------------------------------------------------
-
-
-
--------------------------------------------------------------------------
```

7. Überprüfen Sie die F-Parameter-Werte für das Sicherheitsgerät.

```
Öffnen Sie im Automation Builder Projekt das "F-Parameter" Tab von dem Safety-Gerät und überprüfen Sie, dass die
F-Parameter Werte aus der nachfolgenden Tabelle identisch sind.
================================
|    F-Parameter   | Wert      |
================================
[  ] F_Check_SeqNr | 1         |
[  ] F_Check_iPar  | 0         |
[  ] F_SIL         | 2         |
[  ] F_CRC_Length  | 0         |
[  ] F_Block_ID    | 1         |
[  ] F_Par_Version | 1         |
[  ] F_Source_Add  | 1         |
[  ] F_Dest_Add    | 2         |
[  ] F_WD_Time     | 100       |
[  ] F_iPar_CRC    | 1455424635 |
[  ] F_Par_CRC     | 44422     |
--------------------------------
```

⇨

> **HINWEIS!**
>
> Gemäß dem PROFIsafe V2.6 Protokoll ist der Wert „0“ (Null) für den F-Parameter F_Par_CRC unzulässig und wird automatisch in „1“ geändert. Für diesen Sonderfall wird in der SVT Checkliste ein entsprechender Hinweis angezeigt. Für weitere Informationen wenden Sie sich an den technischen Support von ABB.

Führen Sie diese manuellen Prüfungen für jedes Sicherheitsgerät in der SVT-Checkliste durch. Sie können die Abschnitte für die Sicherheitsgeräte nur überspringen, wenn die Datenprüfsumme für das Sicherheitsgerät identisch mit der Datenprüfsumme aus der vorherigen validierten und bestätigten SVT-Checkliste ist.

Sicherheits-CPU-Abschnitt verifizieren

1. Prüfen Sie, ob die von SVT durchgeführten automatischen Prüfungen für die Sicherheits-CPU bestanden wurden.

```
Die automatischen Überprüfungen für dieses Safety-Gerät sind erfolgreich.
```

   ⇨ Wenn die automatischen Prüfungen Fehler erzeugen, wird eine Fehlermeldung angezeigt *„Fehler bei den automatischen Prüfungen“ auf Seite 193*:

```
Die automatischen Überprüfungen für dieses Safety-Gerät sind fehlgeschlagen.
```

2. Lesen Sie die Datenprüfsumme für die Sicherheits-CPU.

   Verwenden Sie diese Datenprüfsumme zur Verifizierung von Änderungen der Daten für die Sicherheits-CPU. Wenn die Datenprüfsumme identisch mit der Datenprüfsumme aus einer vorherigen validierten SVT-Checkliste ist, sind die Daten für die Sicherheits-CPU identisch, sodass Sie die manuellen Prüfungen dafür überspringen können. Sind die Datenprüfsummen nicht identisch, wiederholen Sie alle manuellen Prüfungen für die Sicherheits-CPU.

```
SVT Datenchecksumme:

5f79 612c cb36 4ee7 4cb5 5202 4b6c 47bd

Ist die SVT-Datenchecksumme identisch zu der SVT-Datenchecksumme von der vorherigen freigegebenen und gültigen
SVT-Checkliste?
[  ] Falls ja, ist die Automation Builder Projektkonfiguration identisch. Sie können die manuelle Überprüfung
     überspringen und die vorherige freigegebene und gültige SVT-Checkliste mit der identischen SVT-Datenchecksumme
     weiterhin nutzen.
[  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen durch.
```

3. Überprüfen Sie den Wert des Parameters „PROFIsafe startup timeout“.

```
[  ] In the Automation Builder project, select "CPU Parameters" tab on the safety CPU and verify that the configured
     parameter "PROFIsafe startup timeout" is identical to 0 ms, which is currently configured in the safety project.
```

Bibliotheksabschnitt verifizieren

Dieser Abschnitt umfasst die Bibliotheks-CRCs der verwendeten Sicherheitsbibliotheken (Abb. 88, Seite 186).

1. Lesen Sie die Datenprüfsumme für die Bibliotheken.

   Verwenden Sie diese Datenprüfsumme zur Verifizierung von Änderungen der Bibliotheken. Wenn die Datenprüfsumme identisch mit der Datenprüfsumme aus einer vorherigen validierten SVT-Checkliste ist, sind die Bibliotheken identisch, sodass Sie die manuellen Prüfungen dafür überspringen können. Sind die Datenprüfsummen nicht identisch, wiederholen Sie alle manuellen Prüfungen für die Bibliotheken.

```
Bibliotheken Datenchecksumme:

f7ba ca5d 042c aaa1 82a6 fe3c 10ea 0371

Ist die Bibliotheken Datenchecksumme identisch zu der Bibliotheken Datenchecksumme von der vorherigen freigegebenen und
gültigen SVT-Checkliste?
[  ] Falls ja, sind die Bibliotheken identisch. Sie können die manuelle Überprüfung der Bibliotheken überspringen.
[  ] Falls nein, führen Sie die unten genannten manuellen Überprüfungen für die Bibliotheken durch.
```

2. Überprüfen Sie, ob die Bibliotheks-CRCs mit den AC500-S-Bibliotheken übereinstimmen
↳ *Kapitel 4.6 „AC500-S-Bibliotheken“ auf Seite 207*.

```
Überprüfen Sie, dass die in diesem Abschnitt aufgeführten Bibliotheks-CRCs identisch sind zu denen im Abschnitt
"AC500-S-Bibliotheken" des AC500-S Sicherheitshandbuchs. Alle weiteren Benutzer-spezifischen Bibliotheken und deren
CRCs, welche nicht im AC500-S Sicherheitshandbuch aufgeführt sind, müssen separat überprüft und durch den Endkunden
bestätigt werden um sie für Safety-spezifische Applikationen zu qualifizieren.
============================================================
|    Bibliothek                                | CRC Wert |
============================================================
[  ] Safety_Standard.lib                       | fd5d3581 |
[  ] Safety_SysLibTime.lib                     | 672b8325 |
[  ] SafetyBase_PROFIsafe_LV210_AC500_V22.lib | 8069df7b |
[  ] SafetyBlocks_PLCopen_AC500_V22.lib        | b6e0bc60 |
[  ] SafetyExt2_LV110_AC500_V27.lib            | aa3be9be |
[  ] SafetyExt_AC500_V22.lib                   | 72a88162 |
[  ] SafetyUtil_CoDeSys_AC500_V22.lib          | 6b29c54  |
[  ] SYSLIBCALLBACK.LIB                        | 62ad210d |
[  ] Target_AC500_V22.lib                      | 8daa436  |
------------------------------------------------------------
```

Ende der SVT-Checkliste verifizieren

Prüfen Sie, ob die SVT-Checkliste mit der Zeile *„Ende der SVT-Checkliste“* endet. Trifft dies zu, markieren Sie das entsprechende Ankreuzfeld im Abschnitt mit Projektinformationen (Abb. 89, Seite 187).

```
[  ] Die SVT-Checkliste hat 'Ende der SVT-Checkliste' als letzte Textzeile.
```

Fehler bei den automatischen Prüfungen

Wenn Fehler bei den automatischen Konsistenzprüfungen auftreten, zeigt SVT dies mittels einer Fehlermeldung im Abschnitt mit Projektinformationen in der SVT-Checkliste an.

```
################################################################################################################################
#                                                                                                                              #
# SVT (Safety Verification Tool) Checkliste                                                                                    #
#                                                                                                                              #
################################################################################################################################

GEFAHR! Sie müssen in funktionaler Sicherheit geschult sein, um an funktionalen Safety-Geräten zu arbeiten. Lesen und
verstehen Sie das AC500-S Sicherheitshandbuch und andere relevante Dokumente bevor Sie SVT benutzen. Weitere
Informationen finden Sie unter www.abb.com/plc.
Diese SVT-Checkliste wurde durch das Safety Verification Tool generiert. Benutzen Sie es um die Integrität Ihres
Safety-Projektes zu überprüfen. Es enthält die Ergebnisse der automatischen Überprüfungen durch das SVT und führt die
verwendeten Safety-Geräte des Projektes zur manuellen Überprüfung auf. Stellen Sie sicher, dass alle verwendeten
Safety-Geräte enthalten und deren Daten korrekt sind. Archivieren Sie die SVT-Checkliste für die weitere Nutzung, wenn
alle Überprüfungen erfolgreich waren.

Generiert am:     24.09.2020 16:40:50
SVT Version:      1.1.0.592

(1) Die automatischen Überprüfungen sind fehlgeschlagen.
      - Interner Fehler im Safety-Gerät 3. 'SIO_02_02'. Die Fehlernummer finden Sie im Abschnitt des Safety-Gerätes.

(2) Abhilfe:
      - Installieren Sie die GSDML-Datei zu dem Safety-Gerät 3. 'SIO_02_02' neu und aktualisieren Sie dieses im Automation
    Builder.
      - Führen Sie "Safety-Konfigurationsdaten erzeugen" für Ihr Safety-Projekt im Automation Builder erneut aus.

    Kontaktieren Sie den technischen ABB-Support falls der Fehler weiterhin besteht.

Öffnen Sie die "Projektinformation..." im Menü "Projekt" des Safety-Projektes.
[  ] Das "Verzeichnis" und der "Dateiname" sind identisch zum SVT-Checklisten Projektverzeichnis und Dateiname:
     C:\Users\Test\AppData\Local\Temp\CoDeSys\FCC992CECAD1D24ED4CAC646328F9E53__be8fc52d-fac0-49c3-8d1d-6b6e3a490d87\
     AC500_S.AC500PRO
[  ] Das Änderungsdatum ("Geändert am") ist identisch zum SVT-Checklisten Projektdatum: 24.09.2020 16:30:19
[  ] Überprüfen Sie im AC500-S Programming Tool anhand des Menüs "Online" / "Prüfe Bootprojekt der Steuerung", dass das
     Projekt im AC500-S Programming Tool und das Boot-Projekt auf der Safety-CPU identisch sind und geben Sie die
     BOOT-Projekt-CRC hier ein:
     _____

Die im Safety-Projekt gespeicherte "Projektinformation ..." (nur zur Information)
Titel:              SVT Projekt Titel
Autor:              SVT Projekt Autor
Version:            SVT Projekt Version
Beschreibung:       SVT Projekt Beschreibung

(3) SVT Datenchecksumme:

    Nicht verfügbar

(4) Überprüfen Sie, dass alle konfigurierten Safety-Geräte des Automation Builder Projektes unten aufgeführt sind und dass
    jedes von ihnen einen Abschnitt mit dem gleichen Namen in der SVT-Checkliste hat:
[  ] 1. DX581_S
[  ] 2. _12_Byte_In_Out_Safety
[  ] 3. SIO_02_02                                                                                          FEHLER
[  ] Alle Safety-Geräte aus dem Automation Builder Projekt sind in der SVT-Checkliste enthalten.
[  ] Die SVT-Checkliste hat 'Ende der SVT-Checkliste' als letzte Textzeile.
```

Abb. 90: Beispiel einer SVT-Checkliste mit Fehlern. Wenn Fehler bei den automatischen Konsistenzprüfungen auftreten, weicht der Inhalt des Abschnitts mit Projektinformationen in der SVT-Checkliste geringfügig ab.

1 Liste der bei automatischen Konsistenzprüfungen von SVT auftretenden Fehler
2 Liste der von SVT vorgeschlagenen Abhilfemaßnahmen, um die Ursachen von Fehlern zu beheben
3 Für die SVT-Checkliste wird keine Datenprüfsumme angegeben, wenn Fehler auftreten
4 Die Liste der Sicherheitsgeräte zeigt an, welche Sicherheitsgeräte Fehler erzeugt haben

HINWEIS!

Wenn Sie nicht alle Fehler mithilfe der vorgeschlagenen Abhilfemaßnahmen oder auf sonstige Weise beheben können, wenden Sie sich an den technischen Support von ABB, um Unterstützung zu erhalten.

Zusätzlich zum Abschnitt mit Projektinformationen liegt für jeden Sicherheitsgeräte-Abschnitt mit Fehlern eine entsprechende Meldung vor.

```
##########################################################################################################################
#                                                                                                                        #
# 3. SIO_02_02                                                                                                           #
#                                                                                                                        #
##########################################################################################################################

(1) Die automatischen Überprüfungen für dieses Safety-Gerät sind fehlgeschlagen.
      - Fehler in der GSDML-Datei. Fehlernummer 6602.

(2) Safety-Gerät Datenchecksumme:

    Nicht verfügbar

(3) Geräteidentifikation:
    Name                    SIO_02_02
    Gerätetypbeschreibung   SIO 02/02
    GSDML                   C:\ProgramData\AutomationBuilder\AB_Devices_2.3\81\0x00B0_0x0100_DIM 1\Sw%3D%2C Hw%3D\
                            GSDML-V2.3-Phoenix Contact-FL PN PN SDIO 2TX 2TX X1-X2-V1.1-20130408.xml
    Gerätetyp               81
    Geräte-ID               0x00B0_0x0100_DIM 1
    Geräteversion           Sw=, Hw=
    Modul-ID                sdio3-2_x1x2
```

Abb. 91: Beispiel eines Sicherheitsgeräte-Abschnitts mit Fehlern. Wenn Fehler bei den automatischen Prüfungen für ein Sicherheitsgerät auftreten, weicht der Inhalt des Sicherheitsgeräte-Abschnitts in der SVT-Checkliste geringfügig ab.

1 Liste der Fehler für dieses Sicherheitsgerät mit beispielhaften Fehlercodes
2 Wenn Fehler vorliegen, wird für das Sicherheitsgerät keine Datenprüfsumme angegeben
3 Informationen zur Geräteidentifikation als Hilfestellung bei der Fehlerbehebung

Zusammenfassung von Fehlermeldungen

Mögliche von SVT generierte Fehler:

- Allgemeine Fehler:
  - Interner Fehler in Gerät N. „XYZ“. Informationen zu Fehlercodes sind dem Abschnitt zu diesem Sicherheitsgerät zu entnehmen.
  - Interner Fehler im Sicherheitsprojekt. Fehlercode x.
  - Maximale Anzahl von 32 angeschlossenen F-Devices wurde überschritten.
- Fehler im Zusammenhang mit Sicherheitsgeräten:
  - Interner Fehler im Sicherheitsgerät. Informationen zu Fehlercodes sind dem Abschnitt zu diesem Sicherheitsgerät zu entnehmen.
  - Interner Fehler im Sicherheitsgerät. Fehlercode x.
  - Interner Fehler im Sicherheitsgerät oder in der GSDML-Datei. Informationen zu Fehlercodes sind dem Abschnitt zu diesem Sicherheitsgerät zu entnehmen.
  - Interner Fehler im Sicherheitsgerät oder in der GSDML-Datei. Fehlercode x.
  - Interner Fehler in F-Parametern. Informationen zu Fehlercodes sind dem Abschnitt zu diesem Sicherheitsgerät zu entnehmen.
  - Interner Fehler in F-Parametern. Fehlercode x.
  - Fehler in der GSDML-Datei. Fehlercode x.
  - Fehlende GSDML-Datei.
- Fehler im Zusammenhang mit F-Parametern oder der Kanalzuordnung:
  - Interner Fehler. Fehlercode x.
  - Mehrere Zuordnungen zu einem Ausgang sind nicht zulässig. Verwenden Sie entweder das übergeordnete Element oder Unterelemente.

Vom SVT vorgeschlagene mögliche Abhilfemaßnahmen zur Fehlerbehebung:

- Installieren Sie die GSDML-Datei von Sicherheitsgerät N. „XYZ“ neu und aktualisieren Sie dieses Gerät im Automation Builder.
- Verwenden Sie entweder das übergeordnete Element oder Unterelemente im Sicherheitsgerät N. „XYZ“
- Wiederholen Sie den Befehl *„Sicherheits-Konfigurationsdaten erzeugen“* für Ihr Sicherheitsprojekt im Automation Builder.
- Löschen oder trennen Sie F-Devices, um die maximal zulässige Anzahl von 32 nicht zu überschreiten.

4.4 Sicherheitsprogrammierrichtlinien

4.4.1 Übersicht

AC500-S Programming Tool ist zum Anlegen von Sicherheitsanwendungen bestimmter Klassen geeignet, wenn es in einer geeigneten Umgebung zusammen mit Steuerungen wie AC500-S, die speziell für diesen Zweck vorgesehen sind, eingesetzt wird. Dafür müssen jedoch bestimmte Richtlinien beachtet werden, die in diesem Dokument beschrieben werden.

4.4.1.1 Zielgruppe

Dieses Dokument richtet sich an Anwender, die Sicherheitsanwendungen mit AC500-S Programming Tool erstellen möchten.

Es dient auch als Grundlage für Prüfer, die Sicherheitsanwendungen abnehmen.

4.4.1.2 Anforderungen

Zum Verständnis dieses Dokuments ist die Kenntnis der IEC 61131-3 ↳ *[4]* erforderlich.

Erfahrung mit der Erstellung von Sicherheitsanwendungen ist hilfreich.

4.4.1.3 Begriffe

Ausgang	-	Variable, die auf eine IEC-Ausgangsadresse (%Q) abgebildet wird
Ausgangsparameter	-	VAR_OUTPUT eines Programms oder Funktionsbausteins
Eingänge	-	Variable, die auf eine IEC-Eingangsadresse (%I) abgebildet wird
Eingangsparameter	-	VAR_INPUT eines Programms, einer Funktion oder eines Funktionsbausteins

4.4.2 Framework

4.4.2.1 Safety Integrity Level (SIL)

AC500-S Programming Tool ist für die Erstellung von Anwendungen bis SIL 3 geeignet. Für höhere Stufen ist die Verwendung von AC500-S Programming Tool nicht gestattet.

4.4.2.2 Freigegebene AC500-S Programming Tool-Version

Die folgenden Produktversionen sind für Sicherheitsanwendungen zugelassen:

Produktkomponente	Name der Produktkomponente	Version (Datum)
Programmiersystem	AC500-S Programming Tool	ab 2.3.9.9

Unter *„Hilfe ➔ Info“* wird die AC500-S Programming Tool-Version angezeigt. Die korrekte Version des Laufzeitsystems wird über die SIL 3-Zulassung des Steuerungssystems durch den TÜV SÜD bestimmt.

4.4.2.3 Steuerungsspezifische Anwendungshinweise

Zum Laden der Sicherheitsanwendung muss bei Sicherheitssteuerungen ein spezielles Verfahren beachtet werden. In AC500-S Programming Tool wird das Laden des Bootprojekts als sicher eingestuft, da es durch geeignete Mechanismen abgesichert ist.

Vorgehensweise in AC500-S Programming Tool zum Laden von Sicherheitsanwendungen

1. Anwenderprogramm kompilieren.
2. Mit der Steuerung verbinden. Dies ist passwortgeschützt. Ggf. wird eine automatische Kompilierung des Anwenderprogramms angestoßen.
3. Menübefehl *„Online ➔ Bootprojekt erzeugen“* ausführen.
4. Steuerung neu starten.

   ⇨ Hierdurch wird die Anwendung gestartet und geladen.

Alle Online-Befehle wie die folgenden deaktivieren den sicheren Betrieb:

- Laden
- Online-Change
- Breakpoint setzen
- Werte schreiben
- Werte forcen
- Trace
- Einzelschritt
- Start/Stopp
- Ablaufkontrolle

Variablen-Monitoring im Online-Modus deaktiviert den sicheren Betrieb nicht.

4.4.2.4 Vorgehensweise für die Anwendungserstellung

Die Erstellung der Anwendung muss nach den Richtlinien der relevanten Sicherheitsnormen, z. B. IEC 61508 für funktionale Sicherheit, IEC 61511 für funktionale Sicherheit in der Prozessautomation und ISO 13849-1 und IEC 62061 für funktionale Sicherheit in Maschinen, erfolgen. Zusätzlich zur umfassenden Dokumentation von Anforderungen, Architektur und Modul-Schnittstellen betrifft dies auch einen vollständigen Funktionstest sämtlicher Teile der Sicherheitsanwendung. Für diesen Test muss die Maschine in ihrem bzw. der Prozess in seinem endgültigen Zustand sein, d. h. einschließlich der mechanischen, elektrischen und elektronischen Komponenten, Sensoren, Aktoren und der Software. Das Testen in einer speziellen Testumgebung, z. B. unter Verwendung eines Debuggers, erleichtert das Bestehen des Abschlusstests, kann aber nicht als Ersatz verwendet werden.

4.4.2.5 Einstellungen

Tab. 10: Die folgenden Systemeinstellungen sind erforderlich:

Einstellen	Wert
Konstanten ersetzen	Ausgewählt unter *„Projekt ➔ Optionen ➔ Übersetzungsoptionen“*
Aktionen verschatten Programme	Ausgewählt unter *„Projekt ➔ Optionen ➔ Übersetzungsoptionen“*

4.4.2.6 Klassifikation

Generell können die meisten Sprachen für Sicherheitsanwendungen verwendet werden. Für bestimmte Sprachen, für die ein erhöhtes Fehlerpotenzial bei der Erstellung der Anwendung besteht, gilt dies nur in beschränktem Maße, und die Durchführung einer zusätzlichen Fehlerprävention wird unbedingt empfohlen. Diese Maßnahmen sind zusammen mit der entsprechenden Sprache angegeben.

4.4.3 Sprachenspezifische Programmierrichtlinien

4.4.3.1 Sicherheitsbezogene Einschränkungen für Entwickler

Bei der Entwicklung von Sicherheitsanwendungen mit AC500-S Programming Tool gelten bestimmte Einschränkungen, die durch organisatorische Maßnahmen sichergestellt werden müssen. Diese sind wie folgt:

- Für Sicherheitsanwendungen sind AC500-S Programming Tool-Visualisierungen nur zu Anzeigezwecken gestattet. Das Verändern von Werten mit Menübefehlen (z. B. „Werte schreiben“ *Kapitel 4.4.2.3 „Steuerungsspezifische Anwendungshinweise“ auf Seite 197*) versetzt das Laufzeitsystem in den nicht sicheren Modus, möglicherweise ohne den Anwender zu informieren.

4.4.3.2 Sprache

Von den IEC 61131-3-Sprachen, die in AC500-S Programming Tool implementiert sind, sind „Strukturierter Text“ (ST), „Funktionsbausteinsprache“ (FUP) und „Kontaktplan“ (KOP) für Sicherheitsanwendungen zugelassen.

4.4.3.3 Task-System

Aufgrund der schlechten Testmöglichkeiten in Sicherheitsanwendungen sollte Multitasking auf ein Minimum beschränkt werden. Für eine mit AC500-S Programming Tool erstellte Anwendung bedeutet das Folgendes:

- Die gesamte Anwendung bestehend aus sicherheitsgerichteten und nicht sicherheitsgerichteten Teilen sollte aus dem Programm „PLC_PRG“ aufgerufen werden. Für eine gute Strukturierung des Programms sollte in „PLC_PRG“ keine Logikverarbeitung programmiert werden. Zuweisungen, Programmaufrufe, Funktionen und Funktionsbausteine sind zulässig.
- Die steuerungsspezifischen Optionen zur Überwachung der Gesamt-Ausführungszeit müssen aktiviert werden und deutlich unter der Fehlertoleranzzeit liegen.

4.4.3.4 Variablendeklarationen

Die folgenden Variablentypen und -attribute aus IEC 61131-3 sind für Sicherheitsanwendungen geeignet:

Schlüsselwort	Beschreibung	Geeignet (Ja / In begrenztem Maße / Nein) (Kommentar)
VAR	Lokale Bausteinvariable	Ja
VAR_INPUT	Baustein-Eingangsparameter	Ja
VAR_OUTPUT	Baustein-Ausgangsparameter	Ja

Schlüsselwort	Beschreibung	Geeignet (Ja / In begrenztem Maße / Nein) (Kommentar)
VAR_IN_OUT	Baustein-Referenzparameter	In begrenztem Maße. (Zur Wiedergabe des Nebeneffekts sollte der Parameter ein Präfix haben. Noch besser wäre stattdessen die Verwendung eines Ein- und Ausgangsparameters.)
VAR_GLOBAL	Globale Variable	Ja. (Es wird dringend empfohlen, globale Variablen mit einem Präfix, z. B. „G_“ oder „GS_“ zu versehen (sicherheitsgerichtete Variablen).)
VAR_EXTERNAL	Deklaration globaler Variablen, die im Baustein verwendet werden	Ja
AT	Zuweisung der Variablenadresse	In begrenztem Umfang *Kapitel 4.4.3.5 „Direkte Adressen“ auf Seite 199*
CONSTANT	Deklaration als Konstante (kein Schreibzugriff möglich)	Ja. (Es wird empfohlen, jede Konstante explizit zu deklarieren.)
RETAIN	Der Variablenwert wird nach dem Ausschalten erhalten	Nein, nicht unterstützt
PERSISTENT	Der Variablenwert wird nach dem erneuten Laden erhalten	Nein, nicht unterstützt

Für bessere Lesbarkeit sollten die folgenden Regeln bei der Deklaration der Variablen befolgt werden:

- Nur ein Baustein einer Deklarationsart (z. B. VAR, VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR_GLOBAL und Kombinationen mit CONSTANT) je Komponente
- Nur eine Variablendeklaration pro Zeile mit informativem Kommentar

  Schlecht:

```
VAR
   A, B, C: BOOL; (* mehrere Variablen *)
END_VAR
```

  Gut:

```
VAR
   A: BOOL; (* erste Variable *)
   B: BOOL; (* zweite Variable *)
   C: BOOL; (* dritte Variable *)
END_VAR
```

- Lokale Variablen (VAR) sollten stets einen abweichenden Namen haben. Die Verdeckung globaler Variablen durch lokale Variablen sollte vermieden werden.

4.4.3.5 Direkte Adressen

Die folgenden Regeln sind bei der Verwendung von Adressen zum Erstellen von Sicherheitsanwendungen zu beachten:

- Keine Anwendung von Adressen direkt im Programmcode. Jede verwendete Adresse muss mit „AT“ einer Variablen bei der Deklaration zugewiesen werden. Außerdem wird empfohlen, Ein-/Ausgangsvariablen durch ein Präfix zu identifizieren und diese zusammen in einer einzigen Variablenliste zu definieren.
- Die Verwendung von Merkeradressen (%M) sollte aufgrund der Fehleranfälligkeit der Zuweisung und des fehlenden Zwecks (Speicher wird für Variablen automatisch zugewiesen) auf ein Minimum beschränkt werden.

- Eine multiple Adresszuweisung sollte wegen der Verschleierung der Nebeneffekte vermieden werden. Für Wort- und Bit-weisen Zugriff ist eine Variable für das Wort definiert; zugegriffen wird über den Bitzugang <variable>.<bit number>.
- Keine Adressdeklarationen innerhalb von Programmen, Funktionsbausteinen, Funktionen und Datenstrukturen.

4.4.3.6 Datentypen

Die folgenden Datentypen aus AC500-S Programming Tool sind für die Erstellung von Sicherheitsanwendungen zugelassen:

Tab. 11: Einfacher Datentyp

Schlüsselwort	Geeignet (Ja / In begrenztem Maße / Nein) (Kommentar)
BOOL	Ja
BYTE, SINT, USINT	Ja
WORD, INT, UINT	Ja
DWORD, DINT, UDINT	Ja
TIME, TOD, DATE, DT	Ja
STRING	In begrenztem Maße. (Technisch möglich, allerdings aufgrund fehlender sicherheitsgerichteter E/A-Geräte wenig sinnvoll.)
REAL	In begrenztem Maße. (Fehleranfällig durch Rundungsfehler; deshalb keine Abfrage mit EQ-Operator; prüfen Sie auf ungültige Operationen wie Division durch Null, Quadratwurzel oder Logarithmus einer negativen Zahl.)

Tab. 12: Komplexe Datentypen

Schlüsselwort	Geeignet (Ja / In begrenztem Maße / Nein) (Kommentar)
ARRAY	In begrenztem Maße. (Nur mit expliziter Bereichsüberprüfung, ansonsten zu fehleranfällig.)
STRUCT	Ja
Listing-Typen	Ja
Unterbereichstypen	Ja
POINTER	In begrenztem Maße. (Empfohlene Maßnahmen: keine Zeigerarithmetik, Bereichsüberprüfung, Neuzuweisung eines Zeigerwerts beim Start jedes Zyklus.)

Die folgenden Regeln müssen bei der Verwendung von komplexen Datentypen beachtet werden:

- Bei komplexen Datentypen wird die Typendeklaration empfohlen.
- Vor jedem Zugriff auf ein Array muss eine explizite Bereichsüberprüfung des Index durchgeführt werden. Bei einer Verletzung, die nicht durch die Anwendung erklärt werden kann, sollte das Steuerungssystem in einen sicheren Zustand überführt werden.

GEFAHR!
Ein Speicherzugriff über POINTER (z. B. ADR-Funktion) ist fehleranfällig und wird grundsätzlich nicht empfohlen. Bei einer Verwendung in Sicherheitsanwendungen liegt die Verantwortung für die korrekte Verwendung dieser und der damit verbundenen Funktionen bei der Organisation und bei Personen, die diese Funktionen in der AC500-S-Sicherheitssteuerung verwenden.

4.4.3.7 Bausteine

Sämtliche Bausteintypen aus IEC 61131-3 sind für Sicherheitsanwendungen geeignet:

- PROGRAM
- FUNCTION
- FUNCTION_BLOCK

Die folgenden Programmierrichtlinien müssen für Bausteine beachtet werden:

- Funktionen und Funktionsbausteine dürfen die globalen Anwendungszustände nicht beeinflussen. Dies kann durch Schreibzugriff auf globale Daten und das Aufrufen von Systemkomponenten erreicht werden.
- Ein expliziter Parametertransfer wird für den Aufruf von Programmen und Funktionsbausteinen bevorzugt.
  Schlecht:
  ```
  Inst.Param1 := 7;
  Inst.Param2 := 3;
  Inst();
  X := (Inst.Out1 AND A) OR B;
  ```
  Gut:
  ```
  Inst(Param1 := 7, Param2 := 3, Out => Result);
  X := (Result AND A) OR B;
  ```
- Sämtliche Eingangsparameter sollten für einen Aufruf zugewiesen sein.

4.4.3.8 Bibliotheken

Externe Bibliotheken, die vom Hersteller des Steuerungssystems zugelassen, d. h. in die Firmware des Steuerungssystems implementiert wurden, können für Sicherheitsanwendungen verwendet werden.

Von den Standard-Bibliotheken in AC500-S Programming Tool sind nur die folgenden zugelassen:

Bibliothek	Beschreibung	Version (Datum)
Safety_Standard.lib (früher: Standard.lib)	IEC 61131-3-Standardfunktionen: ● Timer ● Zähler ● Trigger ● Flip-Flops ● String-Verarbeitung	ab 2.3 (04.10.2005)

Anwenderbibliotheken, die vom Hersteller des Steuerungssystems oder dem Endanwender erstellt wurden, können verwendet werden. Beim Einfügen einer Bibliothek muss geprüft werden, ob die ausgewählte Bibliothek tatsächlich eingefügt wurde. Die entsprechende Information wird beim Einfügen der Bibliothek angezeigt.

4.4.3.9 Ausdrücke

4.4.3.9.1 Allgemeines

Die folgenden Regeln sind bei der Programmierung von Ausdrücken für Sicherheitsanwendungen zu beachten:

- Das Mischen von verschiedenen Datentypen in einem Ausdruck ist zu vermeiden. Wenn das Mischen unbedingt erforderlich ist, sollte stattdessen die explizite Typenkonvertierung verwendet werden.
- Die Komplexität der Ausdrücke sollte durch die folgenden Maßnahmen auf ein Minimum gebracht werden:
  - Begrenzung der Verschachtelung (z. B. nicht mehr als 3 Verschachtelungsebenen) pro Ausdruck
  - Nicht mehr als 10 Operatoren und 10 Operanden pro Ausdruck
  - Keine Anwendung von Ausdrücken in Array-Indizes bei Array-Zugriff
  - Keine Anwendung von Ausdrücken in Parametern für Funktionen, Funktionsbausteinen und Programmen

4.4.3.9.2 Konstanten

Für eine transparentere Semantik sollten Konstanten entweder explizit deklariert oder mit der expliziten Typisierung verknüpft werden.

Schlecht:

```
VAR
   size: REAL;
   diameter: REAL;
END_VAR
size:= diameter * 3.14;
```

Gut:

```
VAR CONSTANT
    PI: REAL := 3.14;
END_VAR
VAR
    size: REAL;
    diameter: REAL;
END_VAR
size:= diameter * PI;
```

Auch gut:

```
VAR
   size: REAL;
   diameter: REAL;
END_VAR
size:= diameter * REAL#3.14;
```

4.4.3.9.3 Zuweisungen

Die folgenden Programmierrichtlinien müssen für Zuweisungen beachtet werden:

- Nur eine Zuweisung ist für jede Anweisung gestattet. Die Zuweisung von komplexen Ausdrücken in AC500-S Programming Tool darf für Sicherheitsanwendungen nicht verwendet werden.
  Schlecht:
  ```
  Res1 := Res2 := FunCall(1, C := D, 3);
  ```
  Gut:
  ```
  C := D;
  Res2 := FunCall(1, C, 3);
  Res1 := Res2;
  ```
- Die implizite Konvertierung von vorzeichenlosen, vorzeichenbehafteten und Bitstring-Typen in AC500-S Programming Tool und die Erweiterung von kurzen auf längere Typen während der Zuweisung sollten nicht verwendet werden. Stattdessen sollte die explizite Konvertierung verwendet werden.

4.4.3.9.4 Klammern

Durch die Definition von Prioritäten für Operatoren ist jeder Ausdruck auch ohne Verwendung von Klammern eindeutig definiert. Um jedoch Fehler zu vermeiden und die Lesbarkeit zu verbessern, wird die Verwendung von Klammern empfohlen (außer in bekannten Fällen wie Multiplikation und Division vor Addition und Subtraktion).

Schlecht:

```
X := A < B AND NOT A > C + D OR E;
```

Gut:

```
X := (A < B) AND NOT(A > (C + D)) OR E;
```

4.4.3.9.5 Bitzugriff

Bitzugriff (<variable>.<bit number>) ist für Sicherheitsanwendungen zugelassen und sollte statt der regelmäßig verwendeten multiplen Adresszuweisung verwendet werden.

Schlecht:

```
VAR_GLOBAL
   Flags AT %QW12: WORD;
   Enable AT %QX12.0: BOOL;
END_VAR
Flags := 0;
Enable := TRUE;
```

Gut:

```
VAR CONSTANT
   EnableBit: INT := 0;
END_VAR
VAR
   Flags AT %QW12: WORD;
END_VAR
Flags := 0;
Flags.EnableBit := TRUE;
```

4.4.3.9.6 Konvertierungen

Für die Zuweisung und gemischte Typen sollten keine impliziten Typenkonvertierungen verwendet werden, sondern nur explizite.

Schlecht:

```
VAR
   A: BYTE;
   B: INT;
   C: DWORD;
END_VAR
C := A + B;
```

Gut:

```
VAR
   A: BYTE;
   B: INT;
   C: DWORD;
END_VAR
C := INT_TO_DWORD(B + BYTE_TO_INT(A));
```

4.4.3.10 Operatoren

Die folgende Tabelle zeigt, welche Operatoren für Sicherheitsanwendungen geeignet sind.

Schlüsselwort	Geeignet (Ja / In begrenztem Maße / Nein) (Kommentar)
AND, OR, NOT, XOR	Ja
+, -, *, /, MOD	Ja. (Division sollte einen expliziten Test für Division durch 0 enthalten)
=, <>, >, >=, <, <=	Ja
SQRT, SIN, COS, TAN, ASIN, ACOS, ATAN, LOG, LN, EXPT, EXP	In begrenztem Maße. (Fehleranfällig durch Rundungsfehler.)
MIN, MAX, LIMIT	Ja
MUX, SEL	Ja. (Bitte beachten Sie, dass nicht ausgewählte Zweige nicht ausgeführt werden. Dies kann zu Problemen führen, wenn Funktionen verwendet werden, die Systembibliotheken aufrufen.)
TIME	Ja
ADR	In begrenztem Maße. (Für ZEIGER erforderlich, die in begrenztem Maße eingesetzt werden können.)
INDEXOF	In begrenztem Maße. (Nur als Parameter für Funktionen des Laufzeitsystems. Die verwendete Funktion sollte als unabhängige Task betrachtet werden.)
SIZEOF	Ja
ROL, ROR, SHR, SHL	Ja

4.4.3.11 Sprachenkonstrukte

Die folgenden Steuerelemente der Sprache ST sind für Sicherheitsanwendungen geeignet:

Schlüsselwort	Geeignet (Ja / In begrenztem Maße / Nein) (Kommentar)
IF	Ja
CASE	Ja
FOR	Ja
WHILE	In begrenztem Maße. (Bestätigung, dass keine Endlosschleife vorliegt, ist erforderlich.)
REPEAT	In begrenztem Maße. (Bestätigung, dass keine Endlosschleife vorliegt, ist erforderlich.)
EXIT	In begrenztem Maße. (Verlässt eine Schleife unverzüglich. Eine Schleife sollte nur durch die Endbedingung verlassen werden.)
RETURN	In begrenztem Maße. (Verlässt eine Subroutine unverzüglich. Eine Subroutine sollte nur verlassen werden, nachdem alle Anweisungen verarbeitet worden sind.)

4.4.4 Allgemeine Programmierrichtlinien

Zusätzlich zu sprachenspezifischen Richtlinien sollten Fehler durch die Beachtung allgemeiner Richtlinien vermieden werden. Diese Richtlinien sind hier in keiner besonderen Reihenfolge aufgeführt:

- Wenige Zustände

  Zustände in der Form von Variablen, die ihren Wert über einen Steuerungszyklus hinaus behalten, erschweren die Prüffreundlichkeit einer Anwendung. Das kann mit den folgenden Maßnahmen verhindert werden:
  - Vermeidung von Zuständen, wo immer möglich
  - Eine Zustandsvariable sollte nur einmal pro Zyklus beschrieben werden. Dies erleichtert das Finden von Fehlern, wenn ein Zustand einen ungültigen Wert hat.
  - Wenn ein Zustand aus mehreren Variablen besteht, sollte er in einen Funktionsbaustein eingebettet werden. Zustandsübergänge sollte nur vom Aufrufen eines Bausteins betroffen sein.
- Keine Warnungen

  Eine Sicherheitsanwendung darf keine Compiler-Warnungen generieren!
- Begrenzte Anzahl von Zeilen pro Baustein (500)

  Für eine bessere Transparenz sollte ein Baustein nicht mehr als 500 Zeilen haben.
- Begrenzte Anzahl von Zeichen pro Zeile (150)

  Für eine bessere Transparenz sollte eine Zeile nicht mehr als 150 Zeichen haben.
- Keine Wiederverwendung von Variablen

  Jede Variable sollte nur für einen Zweck eingesetzt werden. Eine Verwendung in einem anderen Kontext, auch wenn der vorherige Zweck nicht länger wichtig ist, enthält ein bedeutendes Fehlerpotenzial, insbesondere im Rahmen von Änderungen.
- Variablen so lokal wie nötig

  Variablen, die nur in einem Baustein beschrieben sind, müssen lokal deklariert werden. Die einzige Ausnahme betrifft Variablen, die mit Adressen verknüpft sind. Diese sollten zur Vermeidung mehrfacher Zuweisungen global deklariert werden.
- Nur ein Zugriff auf Ausgang

  Für Zustände sollten Ausgänge nur an einer Stelle im Programm beschrieben werden.
- Kein Zugriff auf globale Variablen aus Funktionen und Funktionsbausteinen

  Eine Funktion sollte keine Nebeneffekte haben und ein Funktionsbaustein sollte nur den Zustand seiner eigenen Instanz ändern. Funktionen und Funktionsbausteine sollten deshalb keinen Zugriff auf globale Variablen haben.

4.4.5 Sicherheitsgerichtete und nicht sicherheitsgerichtete Teile der Anwendung

Für sehr komplexe Anwendungen wird empfohlen, alle Teile der Sicherheitsanwendung auf ein separates Steuerungssystem zu übertragen. Wenn das nicht möglich ist, sollten diese Teile der Anwendung durch die folgenden Maßnahmen getrennt werden:

- Bausteine (Programme, Funktionsbausteine und Funktionen) sind entweder sicherheitsgerichtet oder nicht. Sämtliche sicherheitsgerichteten Bausteine sollten ein bestimmtes Präfix aufweisen (z. B. „S_“).
- Das Aufrufen von nicht sicherheitsgerichteten Bausteinen in Sicherheitsbausteinen ist nicht erlaubt. Dies muss mit der Funktion „Aufrufbaum ausgeben“ überprüft werden.
- Das Aufrufen von Sicherheitsbausteinen in nicht sicherheitsgerichteten Bausteinen ist in begrenztem Maße erlaubt. Dies muss mit der Funktion „Aufrufbaum ausgeben“ überprüft werden.
- Globale Variablen sind sicherheitsgerichtet oder nicht. Sämtliche Sicherheitsvariablen sollten ein bestimmtes Präfix haben (z. B. „S_“). Sämtliche Sicherheitsvariablen werden in speziellen Variablenlisten definiert, die auch mit einem Präfix versehen sind.
- Schreibzugriff auf Sicherheitsvariablen von nicht sicherheitsgerichteten Bausteinen aus ist nicht erlaubt. Dies muss mit der Funktion „Querverweisliste ausgeben“ überprüft werden.
- Schreibzugriff auf nicht sicherheitsgerichtete Variablen von Sicherheitsbausteinen aus ist nicht erlaubt. Dies muss mit der Funktion „Querverweisliste ausgeben“ überprüft werden.
- In den nicht sicherheitsgerichteten Teilen sollten auch die folgenden Maßnahmen beachtet werden:
  - Begrenzte Anwendung von Zeigern
  - Bereichsüberprüfung der Indizes vor Schreibzugriff auf Felder (ARRAY)
  - Keine multiple Adresszuweisung

4.5 Sicherheitscodeanalyse-Tool

Anstatt die Sicherheitsprogrammierrichtlinien manuell zu überprüfen, kann ein Großteil der Sicherheitsregeln mit dem ABB-Softwaretool „AC500-S Safety Code Analysis“ (SCA) geprüft werden.

Eine detaillierte Beschreibung der Verwendung des SCA-Tools von ABB ist unter *www.abb.com/plc* und im ABB-Hilfesystem zu finden. Das Tool AC500-S SCA kann kostenlos über *www.abb.com/plc* heruntergeladen werden.

Bestimmte Regeln müssen jedoch weiterhin manuell überprüft werden ↳ *Tab. 13 „Manuell zu überprüfende Programmierrichtlinien“ auf Seite 206*. Sie werden vom AC500-S SCA Tool in der Sicherheitsanwendung nicht erkannt.

Tab. 13: Manuell zu überprüfende Programmierrichtlinien

Regel für manuelle Überprüfung in AC500-S Programming Tool	Kommentare (relevant für AC500-S)
Überprüfen Sie, ob der Watchdog aktiviert wurde. Prüfen Sie, ob die Watchdog-Zeit deutlich kürzer ist als die Antwortzeit auf Prozessfehler.	Verwenden Sie eine spezielle Bibliotheken-POE SF_WDOG_TIME_SET ↳ *Kapitel 4.6.7.3 „SF_WDOG_TIME_SET“ auf Seite 346*
Vergewissern Sie sich, dass es nur eine Task gibt.	AC500-S unterstützt nur eine Task; deshalb muss dies nicht überprüft werden.
Prüfen Sie, ob nur Bibliotheken verwendet werden, die für Sicherheitsanwendungen zugelassen sind.	Diese Regeln sind enthalten in ↳ *Kapitel 6.2 „Checkliste für die Erstellung von Sicherheitsprogrammen“ auf Seite 374*.
Prüfen Sie für jede POE, ob es keine unnötigen Zustandsvariablen gibt.	
Prüfen Sie, ob das Folgende für alle Funktionsbausteine eingehalten wird: Wenn mehr als eine Variable zum Speichern der Zustandsinfo verwendet wird, nutzen Sie für diese Variablen ihren eigenen Funktionsbaustein und rufen Sie ihn nur für Zustandsänderungen auf.	

Regel für manuelle Überprüfung in AC500-S Programming Tool	Kommentare (relevant für AC500-S)
Prüfen Sie, ob vom Compiler beim Kompilieren der Anwendung keine Fehler oder Warnungen angezeigt werden.	
Prüfen Sie für jede POE, ob die Variablen nicht später mit einer anderen Bedeutung wiederverwendet werden.	
Prüfen Sie, ob die Namen der Sicherheits-POEs mit „S_“ beginnen. Prüfen Sie, ob die Namen der nicht sicherheitsgerichteten POEs nicht mit „S_“ beginnen.	Diese Regeln müssen nur überprüft werden, wenn Sie sowohl Sicherheits- als auch nicht sicherheitsgerichtete Funktionen in Ihre Sicherheits-CPU AC500-S einbauen möchten. In typischen Anwendungen mit AC500-S ist dies nicht der Fall, da die nicht sicherheitsgerichteten Funktionen in Standard-CPUs realisiert werden.
Prüfen Sie, ob die Namen der Sicherheitsvariablen mit „S_“ beginnen. Prüfen Sie, ob die Namen der globalen Sicherheitsvariablen mit „GS_“ beginnen. Prüfen Sie, ob die Namen der Sicherheitseingänge mit „IS_“ beginnen. Prüfen Sie, ob die Namen der Sicherheitsausgänge mit „OS_“ beginnen. Prüfen Sie, ob die Namen der nicht sicherheitsgerichteten Variablen nicht mit „S_“, „GS_“, „IS_“ oder „OS_“ beginnen.	
Prüfen Sie, ob die Namen der Listen mit globalen Variablen, die nicht sicherheitsgerichtete Variablen enthalten, nicht mit „S_“ beginnen.	
Prüfen Sie, ob die Namen der Listen mit globalen Sicherheitsvariablen mit „S_“ beginnen.	
Prüfen Sie für jede nicht sicherheitsgerichtete POE, ob sie nicht in eine Sicherheitsvariable schreibt.	

4.6 AC500-S-Bibliotheken

4.6.1 Übersicht

Die folgenden Sicherheitsbibliotheken sind vom TÜV SÜD zertifiziert und dürfen mit der AC500-S-Sicherheitssteuerung verwendet werden.

Tab. 14: Sicherheitsbibliotheken

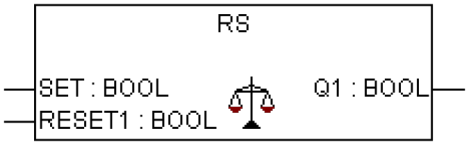
Bibliotheksname / Version	CRC der Bibliothek	Beschreibung
Safety_Standard.lib Version 2.3	fd5d3581	Standard-Sicherheitsfunktionen von AC500-S-Sicherheits-CPUs
Safety_SysLibTime.lib Version 2.4.0.6	672b8325	Interne Zeitsystem-Bibliothek **(Nur für interne Verwendung!)**
SafetyBase_PROFIsafe_LV210_AC500_V22.lib Version 2.1.0	8069df7b	PROFIsafe F-Host und Sicherheits-E/A-Basisfunktionen *Tab. 122 „Versionshistorie der Bibliothek SafetyBase_PROFIsafe“ auf Seite 461*
SafetyBlocks_PLCopen_AC500_v22.lib Version 1.0.0	b6e0bc60	PLCopen Safety-Bibliothek
SafetyDeviceExt_LV100_PROFIsafe_AC500_V27.lib Version 1.0.0	2eadeae9	PROFIsafe F-Device-Funktion an der Sicherheits-CPU

Bibliotheksname / Version	CRC der Bibliothek	Beschreibung
SafetyExt2_LV110_AC500_V27.lib Version 1.1.0	aa3be9be	Sicherheitsfunktionen für die Sicherheits-CPU: ● Auslösung eines Safe Stop ● Lesen des konfigurierten, maximalen Spannungsabfallwertes ● Lesen der Bootprojekt-CRC ● Spezielle Funktionen für benutzerdefinierte CRC Dies sind Zusatzfunktionen zu den in SafetyExt_AC500_V22.lib verfügbaren Funktionen. *Tab. 125 „Versionshistorie der Bibliothek SafetyExt2“ auf Seite 462*
SafetyExt_AC500_V22.lib Version 1.0.0	72a88162	Sicherheitsfunktionen für die Zyklus-, Unter- und Überspannungsüberwachung der Sicherheits-CPU, Datenaustausch mit der Standard-CPU, Nutzerdatenspeicherung im Flash-Speicher usw.
SafetyUtil_CoDeSys_AC500_V22.lib Version 1.0.0	6b29c54	Interne Sicherheitsfunktionen der Sicherheits-CPU **(Nur für interne Verwendung!)**
SysLibCallback.lib Version 2.4.0.6	62ad210d	Interne Sicherheitsbibliothek (nicht in der Bibliotheksverwaltung angezeigt) **(Nur für interne Verwendung!)**
Target_AC500_V22.lib Version 3.4.0.6	8daa436	Interne AC500-Bibliothek (nicht in der Bibliotheksverwaltung angezeigt) **(Nur für interne Verwendung!)**

4.6.2 Safety_Standard.lib

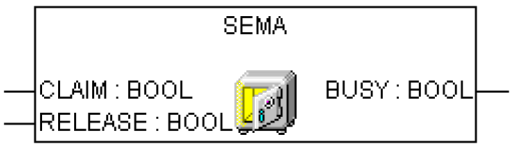
Die Standard-POEs der Safety_Standard.lib in werden nur kurz beschrieben. Detailliertere Informationen über Standardfunktionen finden Sie unter *[3]*.

RS



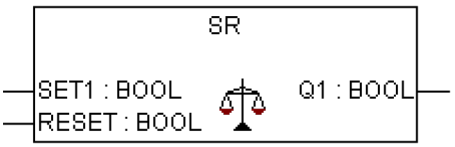
Bistabile Funktion, RESET dominant.
Q1 = NOT RESET1 AND (SET OR Q1)

SEMA



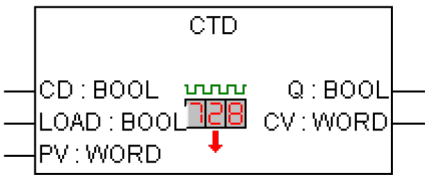
Software-Semaphor. Kann unterbrochen werden!
BUSY ist TRUE, sofern ein Aufruf mit CLAIM = TRUE vorliegt, aber kein Aufruf mit RELEASE = TRUE.
CLAIM = TRUE setzt BUSY = TRUE;
RELEASE = TRUE setzt BUSY = FALSE;

SR



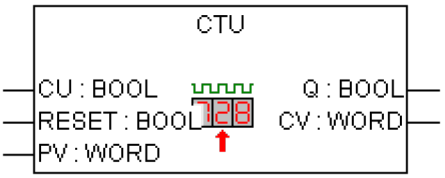
Bistabile Funktion, SET dominant.
Q1 = SET1 OR (NOT RESET AND Q1)

CTD



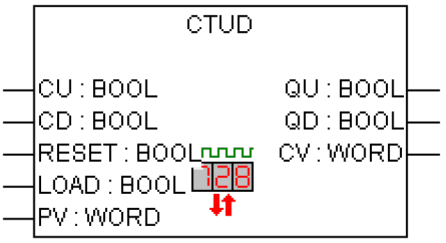
Abwärtszähler.
CV wird um 1 verringert bei steigender Flanke an CD.
Q ist TRUE, wenn CV 0 erreicht.

CTU



Aufwärtszähler.
CV wird um 1 erhöht bei steigender Flanke an CU.
Q ist TRUE, wenn CV PV erreicht.

CTUD

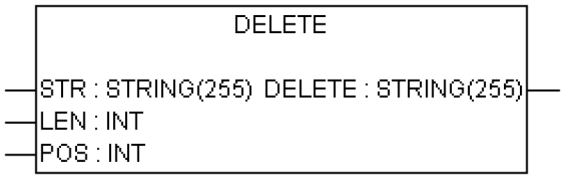


Auf-/Abwärtszähler
CV wird um 1 erhöht bei steigender Flanke an CU.
CV wird um 1 verringert bei steigender Flanke an CD.
QU ist TRUE, wenn Zähler PV ist.
QD ist TRUE, wenn Zähler 0 ist.

CONCAT

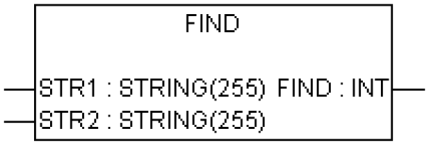
Verknüpfung von zwei Strings.

DELETE



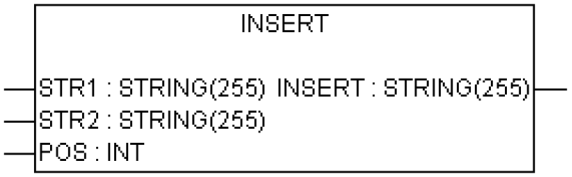
Löscht LEN Zeichen von STR, beginnend ab der POS-ten Zeichenposition.
POS = 1 ist das 1. Zeichen.

FIND



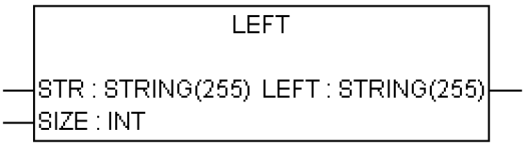
Findet die Zeichenposition zu Beginn des ersten Auftretens von STR2 in STR1.
Wenn kein STR1 gefunden wird, ist das Ergebnis 0.

INSERT



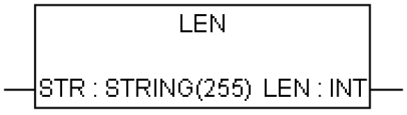
Fügt STR2 in STR1 nach POS-ten Zeichenposition ein.
POS = 0 – Einfügen vor dem ersten Zeichen.
POS = 1 – Einfügen nach dem ersten Zeichen.

LEFT



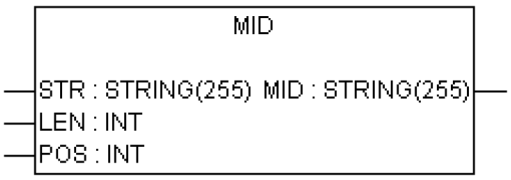
Gibt die ersten SIZE Zeichen von STR zurück.

LEN



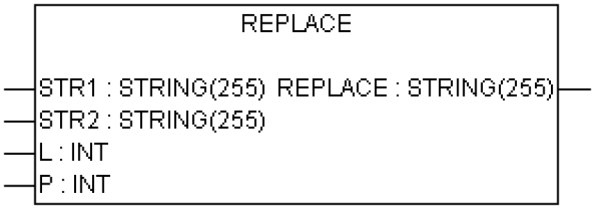
Stringlängen-Funktion.
Gibt die Anzahl der Zeichen in STR zurück.

MID



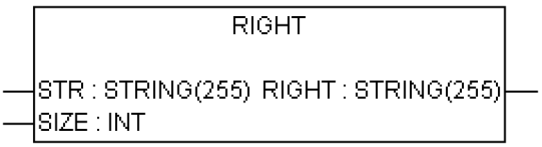
Gibt LEN Zeichen von STR zurück, beginnend ab der POS-ten Zeichenposition.
POS = 1 ist das 1. Zeichen.

REPLACE



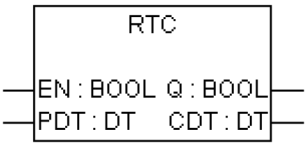
Ersetzt L Zeichen von STR1 durch STR2,
beginnend ab der POS-ten Zeichenposition; gibt einen neuen String zurück.
POS = 1 ist das 1. Zeichen.

RIGHT



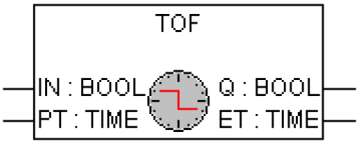
Gibt die letzten SIZE Zeichen von STR zurück.

RTC



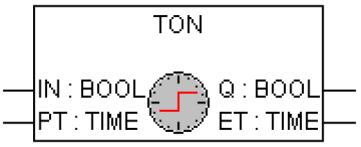
Setzt CDT auf PDT bei steigender Flanke an EN und beginnt mit Erhöhung von CDT.
Bei EN = FALSE, CDT auf DT#1970-01-01-00-00:00

TOF



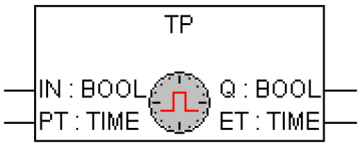
Ausschaltverzögerung.
Q ist FALSE, PT Millisekunden nach fallender Flanke an IN.

TON



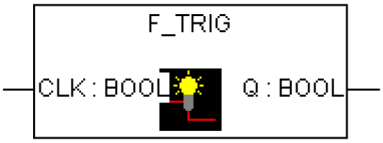
Einschaltverzögerung.
Q ist TRUE, PT Millisekunden nach steigender Flanke an IN.

TP



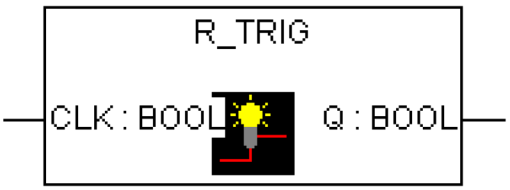
Impulsgeber.
Q gibt High-Signal mit Länge PT bei jeder steigenden Flanke an IN.

F_TRIG



Erkennung fallender Flanken.

R_TRIG



Erkennung steigender Flanken.

4.6.3 SafetyBase_PROFIsafe_LV210_AC500_V22.lib

Diese Bibliothek enthält eine PROFIsafe-Stack-Implementierung (durch POE PROFISAFES-TACK); diese ist eine Hauptkomponente des F-Host.

HINWEIS!

Berücksichtigen Sie bei der Aktualisierung dieser Bibliothek in bestehenden Projekten Folgendes:

Die Verwendung der Bibliotheksversion V2.1.0 (oder höher) resultiert in einer höheren Datenspeicherlast für jedes instanziierte F-Submodul, verglichen mit älteren Versionen der Bibliothek, beispielsweise V2.0.0.

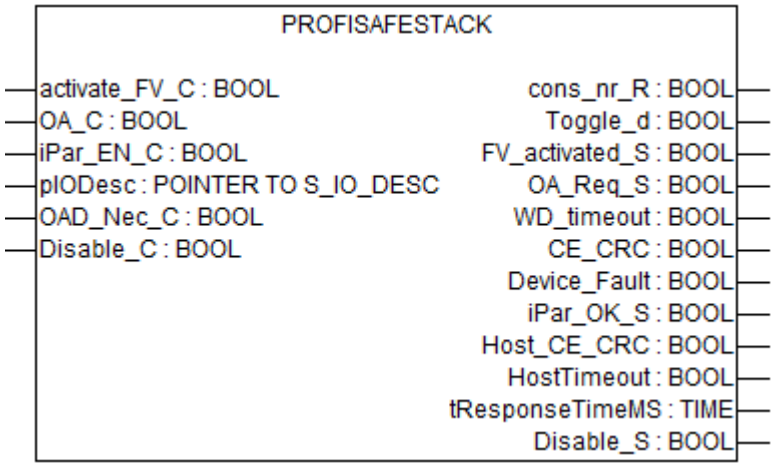
HINWEIS!

Betrifft nur die PROFIsafe-Kommunikation gemäß der PROFIsafe Protokollversion V2.4:

Loopback-Prüfung mit Bit 7 im Status-/Kontrollbyte des PROFIsafe-Telegramms ist implementiert; dies bedeutet, dass Endanwender nicht überlegen müssen, ob systematische Loopback-Konfigurationsfehler vermieden wurden (siehe *www.profisafe.net* für weitere Details).

GEFAHR!

In 100 Stunden ist max. ein Kommunikationsfehler (Ausgangssignale von CE_CRC oder Host_CE_CRC werden TRUE) zulässig, der vom Bediener mit dem Eingangssignal OA_C quittiert wird, ohne dass das verantwortliche Sicherheitspersonal kontaktiert werden muss (weitere Details unter *www.profisafe.net*).



Dieser Funktionsbaustein repräsentiert eine PROFIsafe F-Host-Instanz zur Kontrolle und Überwachung des Zustands eines bestimmten F-Device (Sicherheits-E/A etc.) *[2]*.

Unterstützte Features (in Bezug auf die GSDML-Definitionen der F-Devices):

- „Kurze“ Prozessdaten-Frames gemäß PROFIsafe V2.4 Protokollspezifikation (max. 12 Bytes)
- „Kurze“ Prozessdaten-Frames gemäß PROFIsafe V2.6 Protokollspezifikation (max. 13 Bytes)
- „Lange“ Prozessdaten-Frames gemäß PROFIsafe V2.6 Protokollspezifikation (max. 123 Bytes)
- RIOforFA Profil *Kapitel 4.6.3.1 „RIOforFA Profil“ auf Seite 216*
- Feature „Reaktion auf Device_Fault“ *Kapitel 4.6.3.2 „Feature „Reaktion auf Device_Fault““ auf Seite 217*
- Feature „F-(Sub)Moduledeaktivieren“ *Kapitel 4.6.3.3 „Feature „F-(Sub)Module deaktivieren““ auf Seite 217*

> **HINWEIS!**
> Die beiden Features „Reaktion auf Device_Fault“ and „F-(Sub)Moduledeaktivieren“ können gleichzeitig genutzt werden.

Tab. 15: FB-Name: PROFISAFESTACK

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
activate_FV_C	BOOL	FALSE	Befehl (= TRUE) zum Aktivieren von Failsafe-Werten im F-Device oder (= FALSE) für normalen F-Device-Betrieb
OA_C	BOOL	FALSE	Befehl (= TRUE) für Bedienerquittierung und Wiederaufnahme der Sicherheitsfunktion durch F-Device
iPar_EN_C	BOOL	FALSE	Ist diese Variable TRUE, kann ein Sicherheitsprogramm ein F-Device in einen Modus schalten, in dem es iParameter akzeptiert. Dieser Modus wird von Sicherheits-E/A-Modulen der Serie AC500-S (DI581-S, DX581-S, AI581-S und Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4) nicht unterstützt.
pIODesc	POINTER	NULL	Interner Eingangsparameter **(nur für interne Verwendung!)**
OAD_Nec_C	BOOL	FALSE	*Kapitel 4.6.3.2 „Feature „Reaktion auf Device_Fault““ auf Seite 217*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
Disable_C	BOOL	FALSE	*Kapitel 4.6.3.3 „Feature „F-(Sub)Module deaktivieren““ auf Seite 217*
VAR_OUTPUT			
cons_nr_R	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Er wird gesetzt, wenn das F-Device seinen Zähler für „consecutive number“ in der PROFIsafe-Kommunikation zurückgesetzt hat *[2]*.
Toggle_d	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Dies ist ein gerätebasiertes Toggle-Bit, das einen Trigger zur Erhöhung der virtuellen „consecutive number“ im F-Host anzeigt *[2]*.
FV_activated_S	BOOL	FALSE	Bei Eingabegeräten zeigt diese Variable an, dass der Treiber bei TRUE für jeden Eingangswert Failsafe-Werte „0“ an das F-Host-Programm liefert. Bei Ausgabegeräten zeigt diese Variable an, dass jeder Ausgang bei TRUE auf Failsafe-Werte „0“ gesetzt wird (Standardverhalten) oder dass bei einem F-Ausgang gerätespezifische Werte durch das Signal „activate_FV“ kontrolliert werden *[2]*.
OA_Req_S	BOOL	FALSE	Diese Variable zeigt eine Quittieranforderung vor Wiederaufnahme der Sicherheitsfunktion an. Falls der F-Host-Treiber oder das F-Device einen Kommunikationsfehler oder F-Device-Fehler feststellt, werden Failsafe-Werte aktiviert. Der F-Device-Treiber setzt dann die Variable OA_Req_S (= TRUE), sobald der Fehler behoben wurde und die Quittierung durch den Bediener möglich ist. Nach Quittierung (OA_C = TRUE) setzt der F-Device-Treiber die Anforderungsvariable OA_Req_S (= FALSE) zurück *[2]*.
WD_timeout	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Er ist TRUE, wenn das F-Device einen Kommunikationsfehler erkennt, d. h. wenn die Watchdog-Zeit des F-Device überschritten wird *[2]*.
CE_CRC	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Er wird gesetzt, wenn das F-Device einen Kommunikationsfehler erkennt, d. h. wenn die „consecutive number“ falsch ist (erkannt über CRC2-Fehler im V2-Modus) oder die Datenintegrität verletzt wird (CRC-Fehler) *[2]*.
Device_Fault	BOOL	FALSE	Dieser Parameter wird TRUE, wenn es eine Störung im F-Device gibt (z. B. Unter- oder Überspannung) *[2]*. Wenn das Profil RIOforFA aktiv ist (F_Passivation = 1), ist Device_Fault immer FALSE *Kapitel 4.6.3.1 „RIOforFA Profil“ auf Seite 216*.
iPar_OK_S	BOOL	FALSE	Dieser Parameter wird TRUE, wenn dem F-Device neue Parameterwerte zugewiesen werden *[2]*.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
Host_CE_CRC	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Dieser Parameter wird auf TRUE gesetzt, wenn ein Kommunikationsfehler (CRC-Fehler beim F-Host) vorliegt.
HostTimeout	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Dieser Parameter wird auf TRUE gesetzt, wenn ein Kommunikationsfehler (Zeitüberschreitung beim F-Host) vorliegt.
tResponseTimeMS	TIME	16#0000	**Dieser Parameter ist nur für Debugging bestimmt.** Er repräsentiert die aktuelle Antwortzeit des F-Device in ms. Dieser Wert muss kleiner als der definierte Parameter F_WD_Time für das jeweilige F-Device sein. Wenn dies nicht der Fall ist, wird das F-Device passiviert.
Disable_S	BOOL	FALSE	*Kapitel 4.6.3.3 „Feature „F-(Sub)Module deaktivieren““ auf Seite 217*

Die FB-Instanzen werden für alle F-Devices automatisch generiert; sie liegen im Sicherheitsprojekt unter *„„Ressourcen ➔ Globale Variablen ➔ PROFIsafe“* (Abb. 92, Seite 216). Diese FB-Instanzen sind normale globale Variablen; Endanwender können aus ihren Sicherheitsprogrammen auf sie zugreifen.

GEFAHR!

Unbeabsichtigtes Verhalten vermeiden

Nur gültig, wenn der Eingang OAD_Nec_C = FALSE ist.

Um unerwünschtes Verhalten wie einen unerwünschten Neustart von PROFIsafe-Geräten von Drittanbietern zu vermeiden, sollte der Beschreibung des PROFIsafe Bits Device_Fault im Sicherheitshandbuch für solche Geräte besondere Beachtung geschenkt werden.

Es wird dringend empfohlen, das Bit Device_Fault von PROFIsafe-Aktoren von Drittanbietern wie Ventilen etc. laufend zu überwachen, um deren unerwünschten Neustart, z. B. nach einem Stromausfall, zu vermeiden. Wird Device_Fault = 1 für solche Geräte erkannt, muss die Sicherheitsanwendung das Modul mit activate_FV_C = 1 passivieren. Die Neustarterlaubnis (activate_FV_C = 0) muss in der Sicherheitsanwendung mit der Funktionalität ähnlich FB SF_OutControl bearbeitet werden *Kapitel 4.6.4.17 „SF_OutControl“ auf Seite 319*.

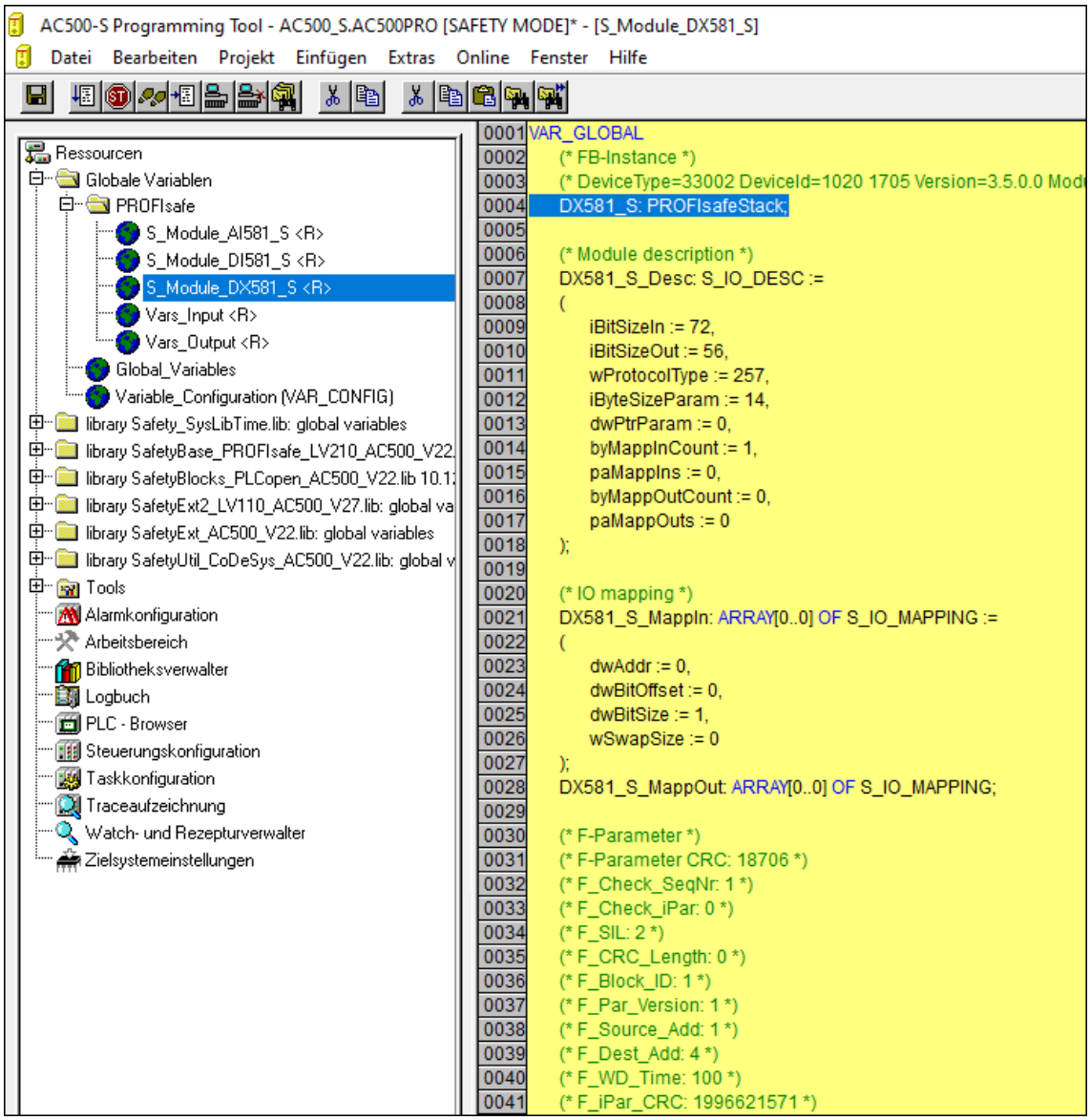


Abb. 92: FB-Instanzen für F-Devices

Beachten Sie, dass die Bibliothek SafetyBase_PROFIsafe_LV200_AC500_V22.lib auch eine Reihe interner POEs (GetWord, MappingIn, MappingOut und SMemCpy) zur Sicherheits-E/A-Abwicklung enthält. **Diese POEs sind nur für interne Verwendung!**

4.6.3.1 RIOforFA Profil

Das Profil RIOforFA verarbeitet kanalgranulare Fehler mit „Status-Bits“, die zusätzlich zu den E/A-Prozesswerten im Prozessdaten-Frame übertragen werden, wie in *[12]* vorgeschlagen.

Voraussetzung: Der F-Parameter F_Passivation ist auf 1 = „Channel“ gesetzt, was nur akzeptiert wird, wenn der F-Parameter F_CRC_Seed = 1 ein PROFIsafe V2.6 F-Device angibt.

RIOforFA ist nur nutzbar, wenn vom F-Device durch definierte Einträge in der GSDML-Datei angeboten (Definition zusätzlicher Status-Bits für jeden E/A-Kanal in Verbindung mit den oben genannten F-Parametern).

Der Vorteil besteht darin, dass Fehler unmittelbar auf der Kanalebene signalisiert werden, auf der die Sicherheitsanwendung reagieren kann. Wenn RIOforFA aktiv ist, belegt der F-Host sein Ausgangssignal Device_Fault nicht, da es im PROFIsafe Statusbyte nicht vorhanden ist (immer FALSE). Wenn mindestens ein Kanalfehler beseitigt ist, zeigt der F-Host dies durch Setzen des Ausgangssignals OA_Req_S an. Nach dem Quittieren (OA_C = TRUE) setzt das F-Device das entsprechende Kanal-Status-Bit auf TRUE. Dadurch wird angegeben, dass der Kanal erneut einen gültigen Wert ausgibt.

HINWEIS!

Nach einem PROFIsafe-Fehler und einer Kanalpassivierung sind zwei Flanken False → True auf OA_C erforderlich: Die erste quittiert den PROFIsafe-Fehler, die zweite die Kanalpassivierung.

Die Anwendung kann erkennen, dass am Ausgang FV_activated_S ein PROFIsafe-Fehler auftrat.

4.6.3.2 Feature „Reaktion auf Device_Fault“

Gemäß *[2], Anhang C.1*

☑ Vorbedingung: RIOforFA-Profil ist nicht aktiviert (F-Parameter F_Passivation = 0, symbolischer Wert „Device/Module“).

▷ Setzen Sie den F-Host-Eingang OAD_Nec_C = TRUE (OperatorAcknowledgeDevicefault_Necessary).

⇨ „Reaktion auf Device_Fault“ ist aktiviert.

Verhalten bei OAD_Nec_C = TRUE („Reaktion auf Device_Fault“ ist aktiviert):

Wenn der F-Host oder ein F-Device einen F-(Sub)Modul-Fehler (Device_Fault = TRUE) erkennt, werden Failsafe-Werte aktiviert, bis ein Fehler gelöscht und quittiert ist. Der F-Host setzt dann den Ausgang OA_Req_S (= TRUE), sobald der Fehler beseitigt wurde und ein Quittieren durch den Bediener möglich ist. Nach dem Quittieren (OA_C = TRUE) setzt der F-Host-Treiber die Anforderungsvariable OA_Req_S (= FALSE) zurück.

Verhalten bei OAD_Nec_C = FALSE („Reaktion auf Device_Fault“ ist nicht aktiviert):

Wenn der F-Host oder ein F-Device einen F-(Sub)Modul-Fehler (Device_Fault = TRUE) erkennt, werden Failsafe-Werte aktiviert, bis ein Fehler gelöscht ist. Eine Quittierung ist nicht erforderlich. *„Unbeabsichtigtes Verhalten vermeiden“ auf Seite 215*

4.6.3.3 Feature „F-(Sub)Module deaktivieren“

Gemäß *[2], Anhang C.2*.

F-Devices, die aus Gründen der Energieeffizienz oder für einen Geräteaustausch heruntergefahren werden sollten, erfordern eine F-Host-Erweiterung, die es ermöglicht, Host-Timeout-/CRC-Fehler zu ignorieren.

Nachdem die Anwendung Disable_C = TRUE setzt, muss der F-Host für dieses F-Device Failsafe-Werte nutzen.

Durch Disable_S = TRUE wird der Anwendung signalisiert, dass der F-Host nun Failsafe-Werte nutzt. Während Disable_S = TRUE werden Timeout/CRC-Fehler vom F-Device ignoriert.

4.6.4 SafetyBlocks_PLCopen_AC500_v22.lib

Eine Liste unterstützter POEs für PLCopen Safety wird in den folgenden Abschnitten vorgestellt. Die entwickelten POEs für PLCopen Safety basieren auf *[8]*.

HINWEIS!

Die in den folgenden Abschnitten angegebenen Normen dienen nur der Information:

- EN 954-1:1996
- IEC 60204-1 Ed. 5.0:2003
- IEC 61496-1:2004
- IEC 62046/Ed.1:2005
- ISO 12100-2:2003
- Maschinenrichtlinie 98/37/EG, Anhang I
- EN 418:1992
- EN 574:1996
- EN 1088:1995
- EN 953:1997

Verwenden Sie für die Zertifizierung der funktionalen Sicherheit die neuesten Normen zur funktionalen Sicherheit *Kapitel 1.8 „Anwendbare Normen“ auf Seite 13*.

4.6.4.1 Einführung

Allgemeine Parameter und Diagnosecodes für POEs von PLCopen Safety werden unten angeführt.

Tab. 16: Allgemeine Eingangsparameter

Name	Typ	Beschreibung
Activate	BOOL	Variable oder Konstante. Aktivierung des Funktionsbausteins. Der Anfangswert ist FALSE. Dieser Parameter kann mit der Variablen verbunden werden, die den Zustand (aktiv oder nicht aktiv) des relevanten Sicherheitsgerätes anzeigt. Dadurch wird keine irrelevante Diagnoseinfo generiert, sobald ein Gerät deaktiviert wird. Bei FALSE werden alle Ausgangsvariablen auf ihren Anfangswert gesetzt. Wenn kein Gerät angeschlossen ist, muss ein statisches TRUE-Signal zugewiesen werden.
S_StartReset	BOOL	Variable oder Konstante. Anfangswert FALSE: Manuelles Rücksetzen, wenn PES gestartet wird (Warm- oder Kaltstart). TRUE: Automatisches Rücksetzen, wenn PES gestartet wird (Warm- oder Kaltstart). Diese Funktion sollte nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht. Deshalb erfordert die Verwendung des automatischen Schaltkreis-Resets der Funktionsbausteine die Implementierung weiterer Maßnahmen auf System- oder Anwendungsebene, um sicherzustellen, dass keine unerwarteten oder unbeabsichtigten Starts auftreten.

Name	Typ	Beschreibung
S_AutoReset	BOOL	Variable oder Konstante. Anfangswert FALSE: Manuelles Rücksetzen, wenn ein Not-Halt-Taster losgelassen wird. TRUE: Automatisches Rücksetzen, wenn ein Not-Halt-Taster losgelassen wird. Diese Funktion sollte nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht. Deshalb erfordert die Verwendung des automatischen Schaltkreis-Resets der Funktionsbausteine die Implementierung weiterer Maßnahmen auf System- oder Anwendungsebene, um sicherzustellen, dass keine unerwarteten oder unbeabsichtigten Starts auftreten.
Reset	BOOL	Variable. Der Anfangswert ist FALSE. Abhängig von der Funktion kann dieser Eingang für verschiedene Zwecke verwendet werden: ● Zurücksetzen der von DiagCode angezeigten Zustandsmaschine-, Kombifehler- und Zustandsmeldungen, nachdem die Fehlerursache behoben wurde. Dieses Verhalten ist als Fehler-Reset ausgelegt. ● Manuelles Rücksetzen einer „Wiederanlaufsperre“ durch den Bediener (siehe EN 954-1). Dieses Verhalten ist als funktionaler Reset ausgelegt. ● Zusätzliche FB-spezifische Reset-Funktionen. Diese Funktion ist nur bei einem Signalwechsel von FALSE auf TRUE aktiv. Ein statisches TRUE-Signal löst keine weiteren Aktionen aus, kann aber in manchen Funktionsbausteinen als Fehler gewertet werden. Die entsprechende Bedeutung muss in jedem Funktionsbaustein beschrieben werden.

Tab. 17: Allgemeine Ausgabeparameter

Name	Typ	Beschreibung
Ready	BOOL	TRUE gibt an, dass der Funktionsbaustein aktiviert wurde und die Ausgangsergebnisse gültig sind (wie die „POWER“-LED eines Sicherheitsrelais). Bei FALSE ist der Funktionsbaustein nicht aktiv und das Programm wird nicht ausgeführt. Hilfreich im Debug-Modus oder zur (De-)Aktivierung zusätzlicher Funktionsbausteine sowie zur weiteren Verarbeitung im funktionsbasierten Programm.
Error	BOOL	Fehleranzeiger (wie die „K1/K2“-LED eines Sicherheitsrelais). TRUE zeigt an, dass ein Fehler aufgetreten und der Funktionsbaustein in einem Fehlerzustand ist. Der relevante Fehlerzustand wird am DiagCode-Ausgang gespiegelt. Bei FALSE gibt es keinen Fehler und der Funktionsbaustein ist in einem anderen Zustand. Dies wird auch durch DiagCode gespiegelt (d. h. DiagCode muss im selben Zyklus wie der Zustandswechsel gesetzt werden). Hilfreich im Debug-Modus sowie zur weiteren Verarbeitung im funktionsbasierten Programm.
DiagCode	WORD	Diagnoseregister. Alle Zustände des Funktionsbausteins (aktiv, nicht aktiv und Fehler) werden durch dieses Register dargestellt. Diese Information wird in Hexadezimalformat kodiert, um mehr als 16 Codes darzustellen. Es wird nur ein konsistenter Code auf einmal dargestellt. Bei multiplen Fehlern zeigt der DiagCode-Ausgang den ersten erkannten Fehler an. *Tab. 18 „Allgemeine Bereiche für Diagnosecodes“ auf Seite 220* *Tab. 19 „System- oder gerätespezifische Codes“ auf Seite 221* *Tab. 20 „Allgemeine Diagnosecodes“ auf Seite 221* Hilfreich im Debug-Modus sowie zur weiteren Verarbeitung im funktionsbasierten Programm.

Ein transparentes und eindeutiges Diagnosekonzept ist die Basis aller Funktionsbausteine. So wird sichergestellt, dass dem Anwender als DiagCode eindeutige Diagnoseinformationen zur Verfügung stehen, unabhängig von der jeweiligen Implementierung des Herstellers. Liegt kein Fehler vor, wird der interne Zustand des Funktionsbausteins angezeigt (Zustandsmaschine). Ein Fehler wird durch einen Binärausgang (Fehler) angezeigt. Detaillierte Informationen über interne und externe Fehler des Funktionsbausteins werden von DiagCode zur Verfügung gestellt. Der Funktionsbaustein muss über verschiedene Reset-Eingänge zurückgesetzt werden.

Ein Hersteller kann zusätzliche Schnittstellen über Funktionsbausteine mit herstellerspezifischen Diagnoseinformationen zur Verfügung stellen.

Tab. 18: Allgemeine Bereiche für Diagnosecodes

DiagCode	Beschreibung
$0000\_0000\_0000\_0000_{bin}$	Der Funktionsbaustein wurde nicht aktiviert, oder die Sicherheits-CPU wurde gestoppt.
$10xx\_xxxx\_xxxx\_xxxx_{bin}$	Zeigt, dass sich der aktivierte Funktionsbaustein im Betriebszustand ohne Fehler befindet. X = FB-spezifischer Code.
$11xx\_xxxx\_xxxx\_xxxx_{bin}$	Zeigt, dass für den aktivierten Funktionsbaustein ein Fehler vorliegt. X = FB-spezifischer Code.

Tab. 19: System- oder gerätespezifische Codes

DiagCode	Beschreibung
$0xxx\_xxxx\_xxxx\_xxxx_{bin}$	X = System- oder gerätespezifische Meldung. Diese enthält die Diagnoseinformation für System oder Geräte und wird direkt auf den DiagCode-Ausgang abgebildet. (Anmerkung: 0000_{hex} ist reserviert)

Tab. 20: Allgemeine Diagnosecodes

DiagCode	Beschreibung
$0000\_0000\_0000\_0000_{bin}$ 0000_{hex}	Der Funktionsbaustein wurde nicht aktiviert. Dieser Code zeigt Leerlauf an. Nachfolgend ein allgemeines Beispiel einer E/A-Einstellung: Activate = FALSE S_In = FALSE oder TRUE Ready = FALSE Error = FALSE S_Out = FALSE
$0111\_1111\_1111\_1111_{bin}$ $7FFF_{hex}$	Der Wert 16#7FFF am DiagCode-Ausgang der PLCopen Safety-Funktionsbausteine zeigt einen internen Fehler an. Wenden Sie sich an den technischen Support von ABB. **Hinweis:** Dies ist ein durch die Sicherheitssteuerung AC500-S definierter herstellerspezifischer Wert.
$1000\_0000\_0000\_0000_{bin}$ 8000_{hex}	Der Funktionsbaustein wurde ohne Fehler oder einen anderen Zustand aktiviert, der den Sicherheitsausgang auf FALSE setzt. Dies ist der Standard-Betriebszustand, wobei der Sicherheitsausgang S_Out im Normalbetrieb TRUE ist. Nachfolgend ein allgemeines Beispiel einer E/A-Einstellung: Activate = TRUE S_In = TRUE Ready = TRUE Error = FALSE S_Out = TRUE
$1000\_0000\_0000\_0001_{bin}$ 8001_{hex}	Der Funktionsbaustein hat eine Aktivierung erkannt und wird jetzt aktiviert, aber der Sicherheitsausgang S_Out ist FALSE. Dieser Code zeigt den Init-Zustand der Betriebsart an. Nachfolgend ein allgemeines Beispiel einer E/A-Einstellung: Activate = TRUE S_In = FALSE oder TRUE Ready = TRUE Error = FALSE S_Out = FALSE

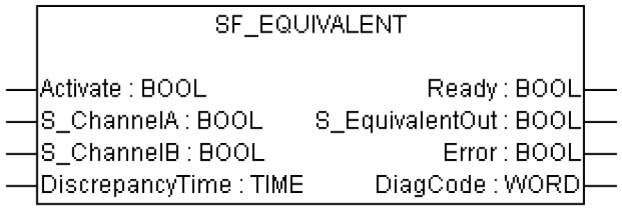
DiagCode	Beschreibung
$1000\_0000\_0000\_0010_{bin}$ 8002_{hex}	Der aktivierte Funktionsbaustein erkennt eine Sicherheitsanforderung, z. B. S_In = FALSE. Der Sicherheitsausgang ist deaktiviert. Dies ist ein Betriebszustand, in dem der Sicherheitsausgang S_Out FALSE ist. Nachfolgend ein allgemeines Beispiel einer E/A-Einstellung: Activate = TRUE S_In = FALSE Ready = TRUE Error = FALSE S_Out = FALSE
$1000\_0000\_0000\_0011_{bin}$ 8003_{hex}	Der Sicherheitsausgang des aktivierten Funktionsbausteins wurde durch die Sicherheitsanforderung deaktiviert. Die Sicherheitsanforderung wird jetzt zurückgezogen, aber der Sicherheitsausgang bleibt FALSE, bis ein Reset-Zustand erkannt wird. Dies ist ein Betriebszustand, in dem der Sicherheitsausgang S_Out FALSE ist. Nachfolgend ein allgemeines Beispiel einer E/A-Einstellung: Activate = TRUE S_In = FALSE => TRUE (danach statisches TRUE) Ready = TRUE Error = FALSE S_Out = FALSE

Hinweis: Wenn es mehrere Betriebszustände gibt, in denen der Sicherheitsausgang TRUE ist, wird die nächste verfügbare DiagCode-Nummer den nachfolgenden Zuständen zugewiesen.

4.6.4.2 SF_Equivalent

Normen	Anforderungen
EN 954-1:1996	6.2 Allgemeine Sicherheitsleitlinien, Ruhestrom 6.2 Fehlererkennung für Kategorie 3 und 4

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.

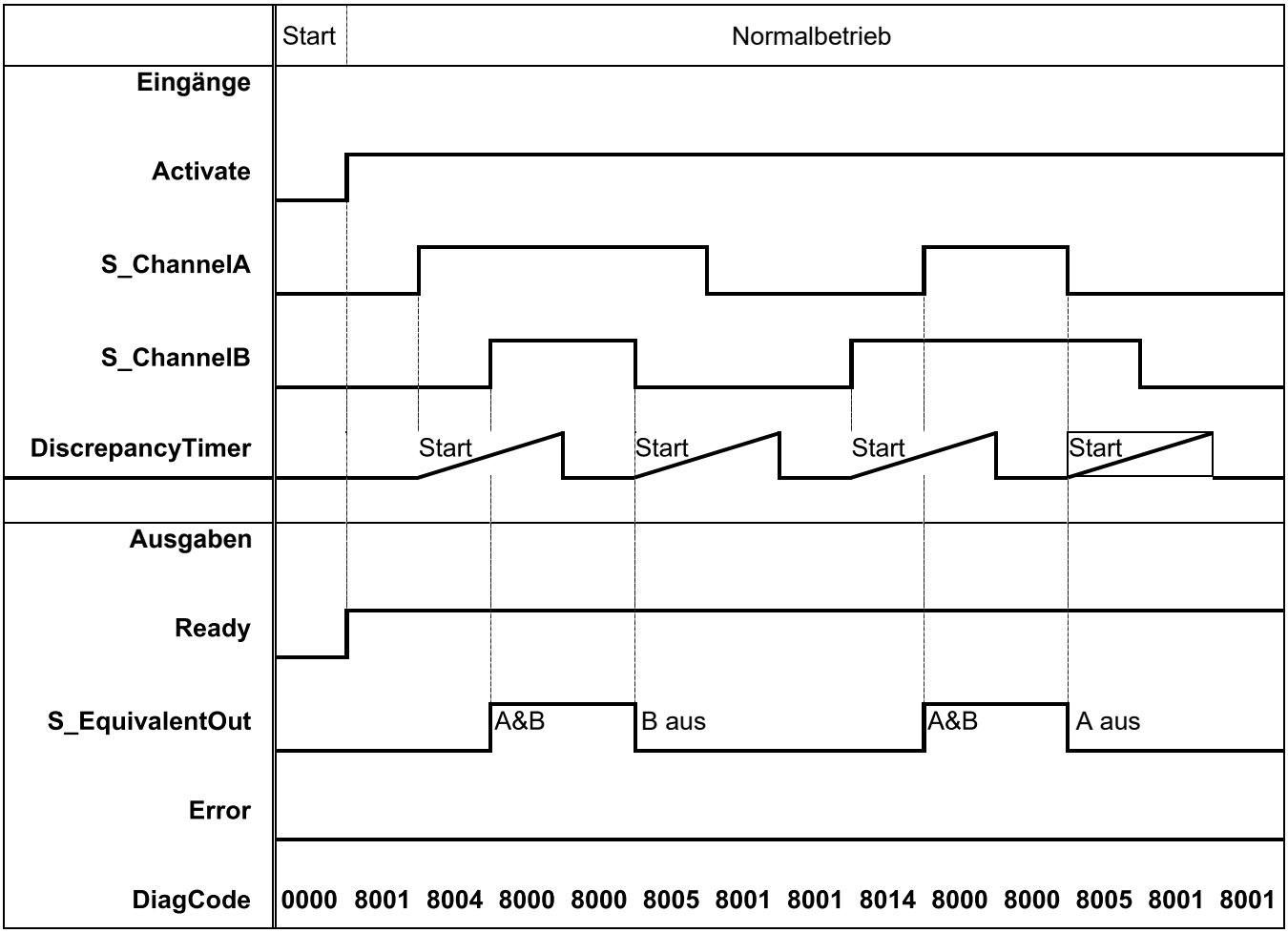


Dieser Funktionsbaustein konvertiert zwei gleiche BOOL-Eingänge (beide NO oder NC) in einen BOOL-Ausgang, einschließlich Diskrepanzzeit-Überwachung. Dieser Funktionsbaustein sollte nicht alleine verwendet werden, da er nicht über eine Wiederanlaufsperre verfügt. Er ist erforderlich, um den Ausgang mit anderen Sicherheits-Funktionsbausteinen zu verbinden.

Tab. 21: FB-Name: SF_Equivalent

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_ChannelA	BOOL	FALSE	Variable. Eingang A für logische Verbindung. FALSE: Kontakt A offen TRUE: Kontakt A geschlossen
S_ChannelB	BOOL	FALSE	Variable. Eingang B für logische Verbindung. FALSE: Kontakt B offen TRUE: Kontakt B geschlossen
DiscrepancyTime	TIME	T#0ms	Konstante. Maximale Überwachungszeit für den Diskrepanzzustand beider Eingänge.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_EquivalentOut	BOOL	FALSE	Sicherheitsausgang FALSE: Mindestens ein Eingangssignal = FALSE oder Zustandsänderung außerhalb der Überwachungszeit. TRUE: Beide Eingangssignale „active“ und Zustandsänderung innerhalb der Überwachungszeit.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typische Zeitdiagramme



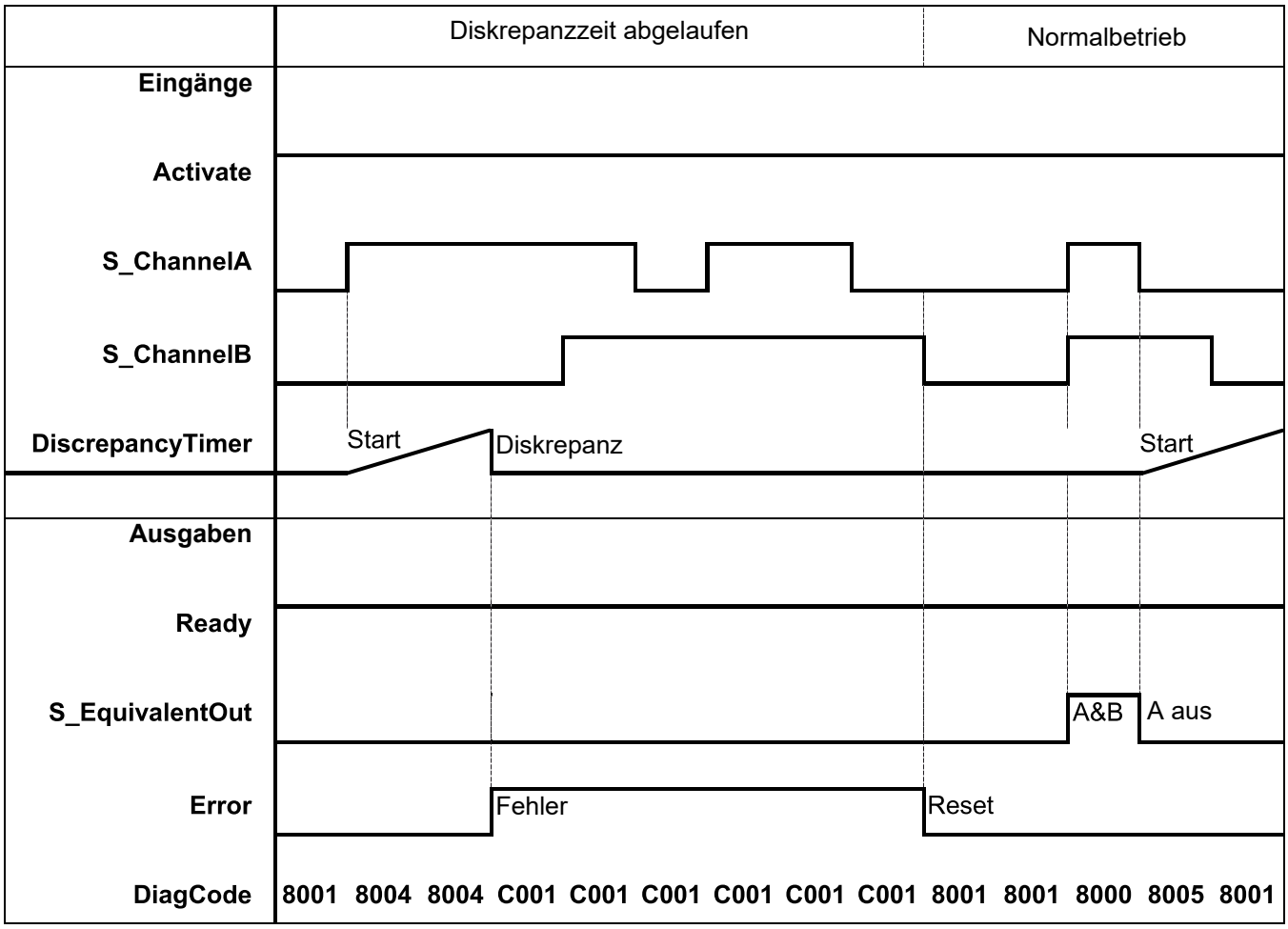


Abb. 93: Typisches Zeitdiagramm für SF_Equivalent

Dieser Funktionsbaustein überwacht die Diskrepanzzeit zwischen Kanal A und B beim Schalten auf TRUE bzw. FALSE.

Verhalten im Fehlerfall

S_EquivalentOut wird auf FALSE gesetzt. Error-Ausgang wird auf TRUE gesetzt. DiagCode zeigt die Fehlerzustände an. Es gibt keinen separaten RESET-Eingang zum Rücksetzen eines Fehlers. Wenn an den Eingängen ein Fehler auftritt, müssen neue Eingangssignale mit korrektem S_EquivalentOut den Fehlermerker zurücksetzen können. (Beispiel: Wenn ein Schaltelement fehlerhaft ist und ausgetauscht wird, führt das erneute Verwenden des Schaltelements zu korrekten Ausgangswerten.)

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 22: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler 1	Diskrepanzzeit im Zustand 8004 abgelaufen. Ready = TRUE S_EquivalentOut = FALSE Error = TRUE
C002	Fehler 2	Diskrepanzzeit im Zustand 8014 abgelaufen. Ready = TRUE S_EquivalentOut = FALSE Error = TRUE
C003	Fehler 3	Diskrepanzzeit im Zustand 8005 abgelaufen. Ready = TRUE S_EquivalentOut = FALSE Error = TRUE

Tab. 23: FB-spezifische Zustandscodes (kein Fehler):

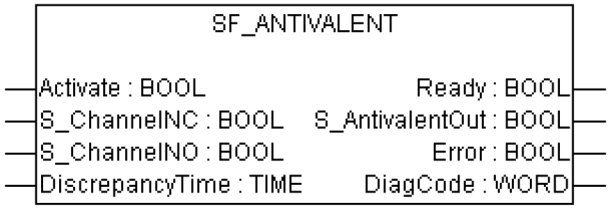
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_EquivalentOut = FALSE Error = FALSE
8001	Init	Der Funktionsbaustein hat eine Aktivierung erkannt und wird jetzt aktiviert. Ready = TRUE S_EquivalentOut = FALSE Error = FALSE
8000	Sicherheits-ausgang aktiviert	Die Eingänge werden im Äquivalenzmodus TRUE. Ready = TRUE S_EquivalentOut = TRUE Error = FALSE
8004	Warten auf Kanal B	Kanal A wurde auf TRUE gesetzt – warten auf Kanal B; Diskrepanz-Timer gestartet. Ready = TRUE S_EquivalentOut = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8014	Warten auf Kanal A	Kanal B wurde auf TRUE gesetzt – warten auf Kanal A; Diskrepanz-Timer gestartet. Ready = TRUE S_EquivalentOut = FALSE Error = FALSE
8005	Warten auf „Active“	Ein Kanal wurde auf FALSE gesetzt; warten, dass der zweite Kanal auch auf FALSE schaltet; Diskrepanz-Timer gestartet. Ready = TRUE S_EquivalentOut = FALSE Error = FALSE

4.6.4.3 SF_Antivalent

Normen	Anforderungen
EN 954-1:1996	6.2 Allgemeine Sicherheitsleitlinien, Ruhestrom 6.2 Fehlererkennung für Kategorie 3 und 4

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein konvertiert zwei antivalente BOOL-Eingänge (NO/NC-Paar) in einen BOOL-Ausgang, einschließlich Diskrepanzzeit-Überwachung. Dieser Funktionsbaustein sollte nicht alleine verwendet werden, da er nicht über eine Wiederanlaufsperre verfügt. Er ist erforderlich, um den Ausgang mit anderen Sicherheits-Funktionsbausteinen zu verbinden.

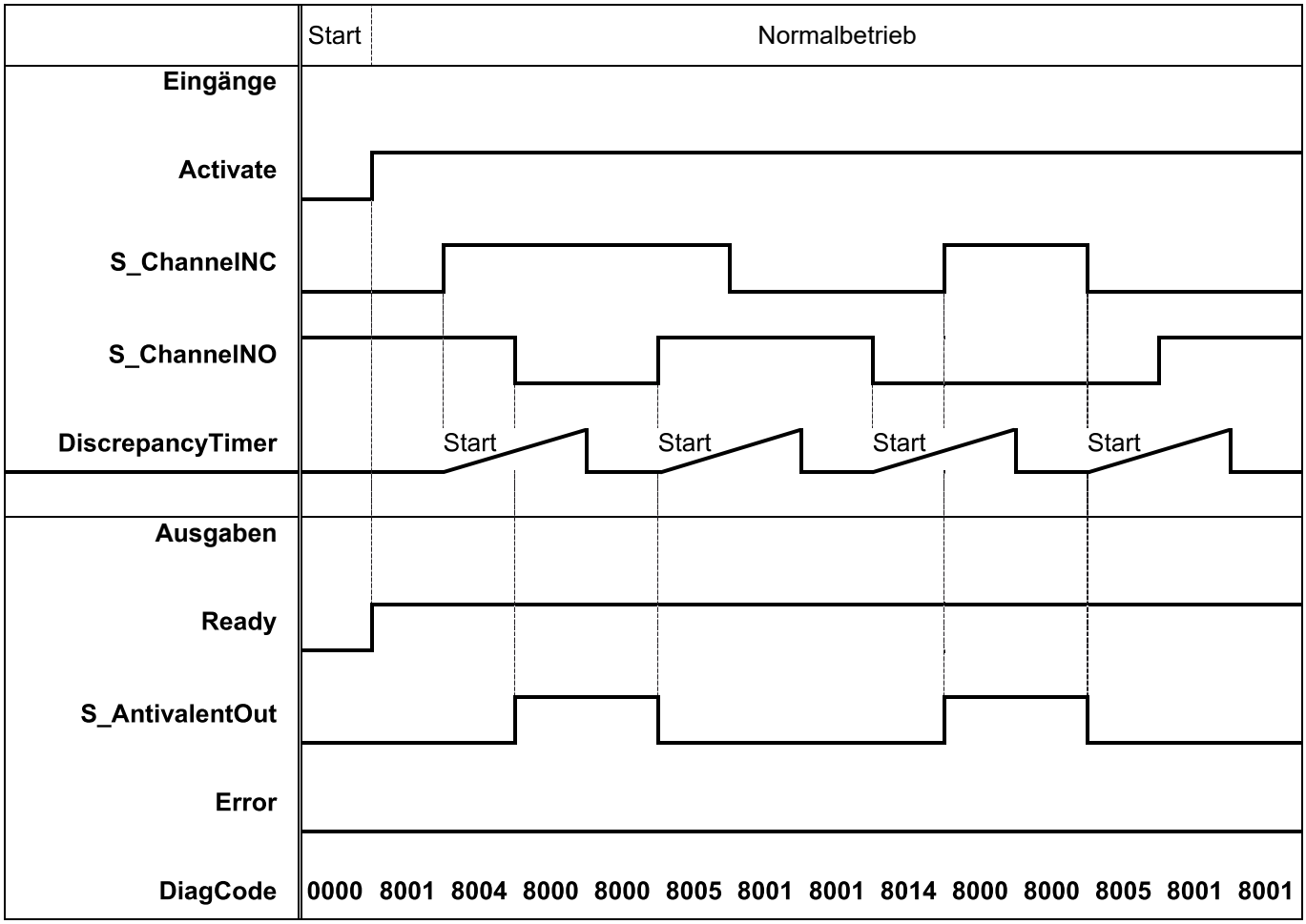
Tab. 24: FB-Name: SF_Antivalent

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_ChannelNC	BOOL	FALSE	Variable. NC bedeutet „Normally Closed“ (Öffner). Eingang für NC-Anschluss. FALSE: Kontakt NC offen TRUE: Kontakt NC geschlossen

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_ChannelNO	BOOL	TRUE	Variable. NO bedeutet „Normally Open“ (Schließer). Eingang für NO-Anschluss. FALSE: Kontakt NO offen TRUE: Kontakt NO geschlossen
DiscrepancyTime	TIME	T#0ms	Konstante. Maximale Überwachungszeit für den Diskrepanzzustand beider Eingänge.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_AntivalentOut	BOOL	FALSE	Sicherheitsausgang FALSE: Mindestens ein Eingangssignal „not active“ oder Zustandsänderung außerhalb der Überwachungszeit. TRUE: Beide Eingangssignale „active“ und Zustandsänderung innerhalb der Überwachungszeit.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweise: „Antivalent“ bedeutet, dass die beiden Eingänge im Normalbetrieb gegensätzliche Zustände aufweisen. Dies wird mitunter auch als „komplementär“ oder „nicht äquivalent“ bezeichnet.

Typische Zeitdiagramme



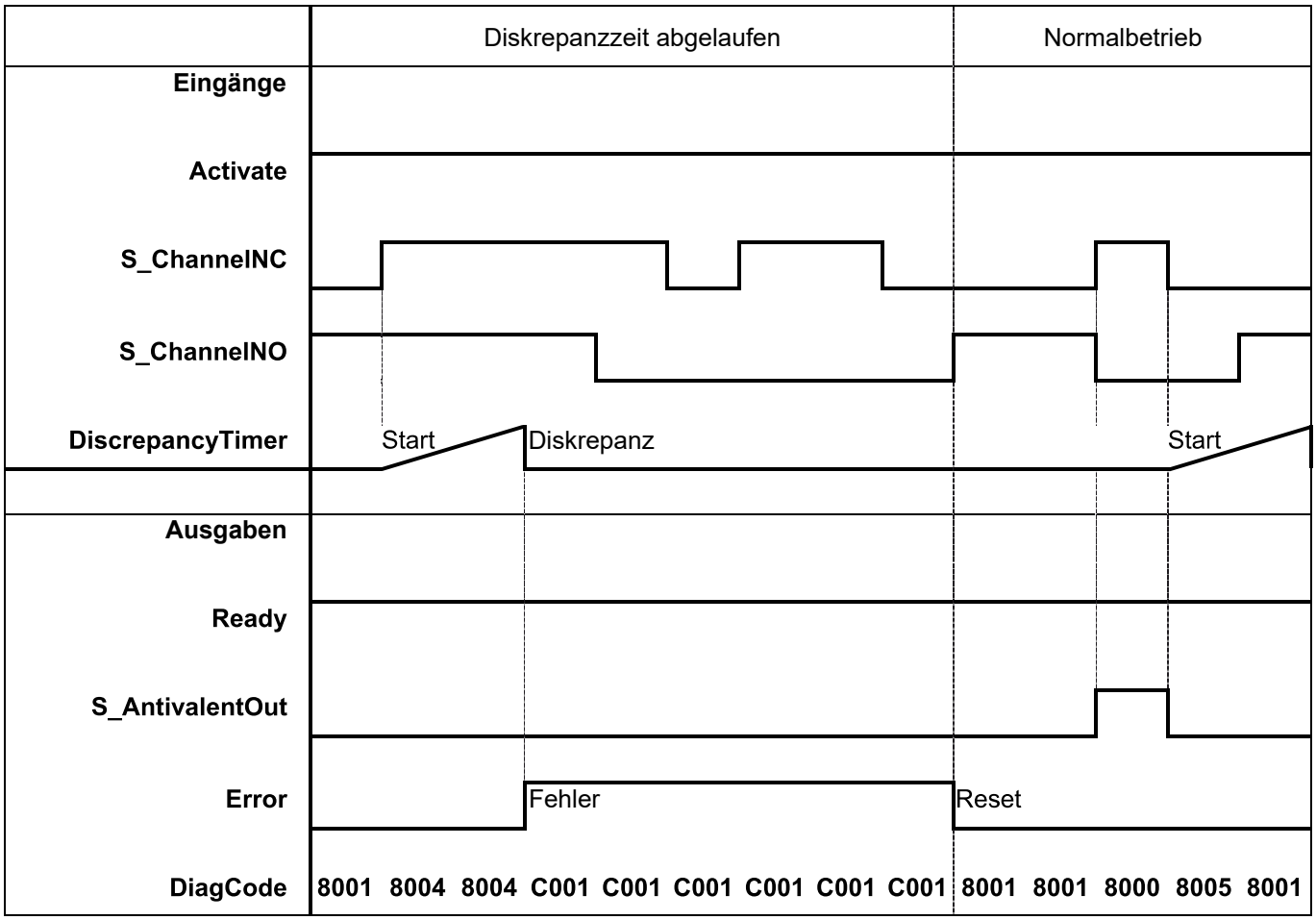


Abb. 94: Typisches Zeitdiagramm für SF_Antivalent

Dieser Funktionsbaustein überwacht die Diskrepanzzeit zwischen Kanal NO und Kanal NC.

Verhalten im Fehlerfall

Der Ausgang S_AntivalentOut wird FALSE. Error-Ausgang wird auf TRUE gesetzt. DiagCode zeigt die Fehlerzustände an.

Es gibt keinen separaten RESET-Eingang zum Rücksetzen eines Fehlers. Wenn an den Eingängen ein Fehler auftritt, müssen neue Eingangssignale mit korrektem Wert den Fehlermerker zurücksetzen können. (Beispiel: Wenn ein Schaltelement fehlerhaft ist und ausgetauscht wird, führt das erneute Verwenden des Schaltelements zu korrekten Ausgangswerten.)

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 25: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler 1	Diskrepanzzeit im Zustand 8004 abgelaufen. Ready = TRUE S_AntivalentOut = FALSE Error = TRUE
C002	Fehler 2	Diskrepanzzeit im Zustand 8014 abgelaufen. Ready = TRUE S_AntivalentOut = FALSE Error = TRUE
C003	Fehler 3	Diskrepanzzeit im Zustand 8005 abgelaufen. Ready = TRUE S_AntivalentOut = FALSE Error = TRUE

Tab. 26: FB-spezifische Zustandscodes (kein Fehler):

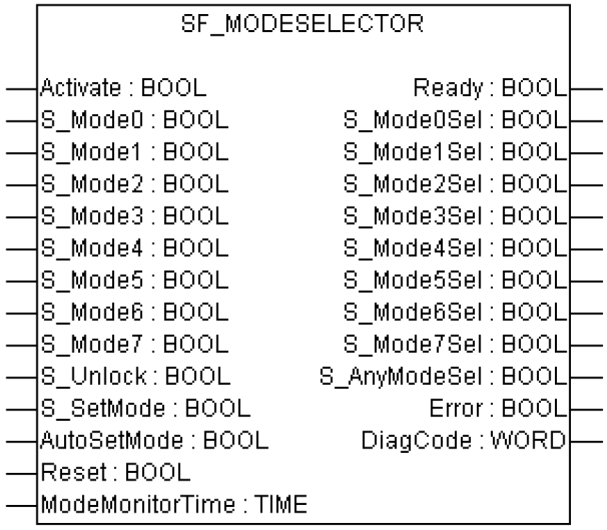
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_AntivalentOut = FALSE Error = FALSE
8001	Init	Der Funktionsbaustein hat eine Aktivierung erkannt und wird jetzt aktiviert. Ready = TRUE S_AntivalentOut = FALSE Error = FALSE
8000	Sicherheits-ausgang aktiviert	Die Eingänge werden im Antivalenzmodus „Active“. Ready = TRUE S_AntivalentOut = TRUE Error = FALSE
8004	Warten auf NO	Der Kanal NC wurde auf TRUE gesetzt; warten, dass der Kanal NO auf FALSE schaltet; Diskrepanz-Timer gestartet. Ready = TRUE S_AntivalentOut = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8014	Warten auf NC	Der Kanal NO wurde auf FALSE gesetzt; warten, dass der Kanal NC auf TRUE schaltet; Diskrepanz-Timer gestartet. Ready = TRUE S_AntivalentOut = FALSE Error = FALSE
8005	Warten auf „Active“	Ein Kanal wurde auf „inactive“ gesetzt; warten, dass der zweite Kanal auch auf „inactive“ gesetzt wird. Ready = TRUE S_AntivalentOut = FALSE Error = FALSE

4.6.4.4 SF_ModeSelector

Normen	Anforderungen
Maschinenrichtlinie 98/37/EG, Anhang I	1.2.3. Ingangsetzen … Das Ingangsetzen einer Maschine darf nur durch absichtliches Betätigen eines hierfür vorgesehenen Stellteils möglich sein. … Dies gilt auch: … – für eine wesentliche Änderung des Betriebszustands … 1.2.5 … in jeder Stellung verriegelbaren Betriebsartenwählschalter. Jede Stellung des Wählschalters darf nur einer Steuerungs- oder Betriebsart entsprechen …
EN ISO 12100-2:2003	4.11.10 Wahl der Steuerungs- und Betriebsart … so muss sie mit einem in jeder Stellung abschließbaren Betriebsartenwahlschalter ausgestattet sein. Jede Stellung des Wahlschalters muss deutlich erkennbar sein und darf nur einer Steuerungs- oder Betriebsart entsprechen …
IEC 60204-1, Ed. 5.0:2003	9.2.3 Betriebsarten … … Sofern durch eine Betriebsartenwahl eine gefahrbringende Situation entstehen kann, muss eine solche Wahl durch geeignete Mittel verhindert werden (z. B. Schlüsselschalter, Zugangscode). Die Betriebsartenwahl selbst darf die Maschine nicht in Betrieb setzen. Eine separate Handlung des Bedieners muss erforderlich sein … Eine Anzeige der ausgewählten Betriebsart ist zur Verfügung zu stellen …
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein wählt die Betriebsart des Systems, z. B. manuell, automatisch, halbautomatisch usw.

Tab. 27: FB-Name: SF_ModeSelector

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_Mode0	BOOL	FALSE	Variable oder Konstante. Eingang 0 von Betriebsartenschalter FALSE: Betriebsart 0 wird vom Bediener nicht angefordert. TRUE: Betriebsart 0 wird vom Bediener angefordert.
S_Mode1	BOOL	FALSE	Variable oder Konstante. Eingang 1 von Betriebsartenschalter FALSE: Betriebsart 1 wird vom Bediener nicht angefordert. TRUE: Betriebsart 1 wird vom Bediener angefordert.
S_Mode2	BOOL	FALSE	Variable oder Konstante. Eingang 2 von Betriebsartenschalter FALSE: Betriebsart 2 wird vom Bediener nicht angefordert. TRUE: Betriebsart 2 wird vom Bediener angefordert.
S_Mode3	BOOL	FALSE	Variable oder Konstante. Eingang 3 von Betriebsartenschalter FALSE: Betriebsart 3 wird vom Bediener nicht angefordert. TRUE: Betriebsart 3 wird vom Bediener angefordert.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_Mode4	BOOL	FALSE	Variable oder Konstante. Eingang 4 von Betriebsartenschalter FALSE: Betriebsart 4 wird vom Bediener nicht angefordert. TRUE: Betriebsart 4 wird vom Bediener angefordert.
S_Mode5	BOOL	FALSE	Variable oder Konstante. Eingang 5 von Betriebsartenschalter FALSE: Betriebsart 5 wird vom Bediener nicht angefordert. TRUE: Betriebsart 5 wird vom Bediener angefordert.
S_Mode6	BOOL	FALSE	Variable oder Konstante. Eingang 6 von Betriebsartenschalter FALSE: Betriebsart 6 wird vom Bediener nicht angefordert. TRUE: Betriebsart 6 wird vom Bediener angefordert.
S_Mode7	BOOL	FALSE	Variable oder Konstante. Eingang 7 von Betriebsartenschalter FALSE: Betriebsart 7 wird vom Bediener nicht angefordert. TRUE: Betriebsart 7 wird vom Bediener angefordert.
S_Unlock	BOOL	FALSE	Variable oder Konstante. Sperrt die ausgewählte Betriebsart FALSE: Der Ist-Wert am Ausgang S_ModeXSel wird gesperrt; deshalb führt eine Veränderung an einem der Eingänge S_ModeX auch bei einer steigenden Flanke von SetMode nicht zu einer Veränderung des Ausgangs S_ModeXSel. TRUE: Der ausgewählte S_ModeXSel ist nicht gesperrt, eine Veränderung der Betriebsart ist möglich.
S_SetMode	BOOL	FALSE	Variable (oder Konstante FALSE, wenn AutoSetMode = TRUE) Setzt die ausgewählte Betriebsart Der Bediener quittiert die Wahl der Betriebsart. Jede Veränderung zu einem neuen S_ModeX = TRUE führt zu S_AnyModeSel/S_ModeXSel = FALSE; nur eine steigende SetMode-Flanke führt dann zu einem neuen S_ModeXSel = TRUE.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
AutoSetMode	BOOL	FALSE	Konstante. Parametrierung der Quittierung der Betriebsarten-wahl FALSE: Eine Änderung der Betriebsart muss vom Bediener über SetMode quittiert werden. TRUE: Eine gültige Änderung des Eingangs S_ModeX zu einem anderen S_ModeX führt auto-matisch zu einer Änderung von S_ModeXSel, ohne dass der Bediener dies durch SetMode quittieren muss (solange dies nicht durch S_Unlock gesperrt ist).
Reset	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
ModeMonitorTime	TIME	T#0	Konstante. Max. zulässige Zeit für das Ändern des Auswahlein-gangs.
VAR_OUTPUT			
Ready	BOOL	FALSE	↳ *Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_Mode0Sel	BOOL	FALSE	Gibt an, dass Betriebsart 0 gewählt und quittiert wurde. FALSE: Betriebsart 0 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 0 wurde ausgewählt oder ist aktiv.
S_Mode1Sel	BOOL	FALSE	Gibt an, dass Betriebsart 1 gewählt und quittiert wurde. FALSE: Betriebsart 1 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 1 wurde ausgewählt oder ist aktiv.
S_Mode2Sel	BOOL	FALSE	Gibt an, dass Betriebsart 2 gewählt und quittiert wurde. FALSE: Betriebsart 2 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 2 wurde ausgewählt oder ist aktiv.
S_Mode3Sel	BOOL	FALSE	Gibt an, dass Betriebsart 3 gewählt und quittiert wurde. FALSE: Betriebsart 3 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 3 wurde ausgewählt oder ist aktiv.
S_Mode4Sel	BOOL	FALSE	Gibt an, dass Betriebsart 4 gewählt und quittiert wurde. FALSE: Betriebsart 4 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 4 wurde ausgewählt oder ist aktiv.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_Mode5Sel	BOOL	FALSE	Gibt an, dass Betriebsart 5 gewählt und quittiert wurde. FALSE: Betriebsart 5 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 5 wurde ausgewählt oder ist aktiv.
S_Mode6Sel	BOOL	FALSE	Gibt an, dass Betriebsart 6 gewählt und quittiert wurde. FALSE: Betriebsart 6 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 6 wurde ausgewählt oder ist aktiv.
S_Mode7Sel	BOOL	FALSE	Gibt an, dass Betriebsart 7 gewählt und quittiert wurde. FALSE: Betriebsart 7 wurde nicht ausgewählt oder ist nicht aktiv. TRUE: Betriebsart 7 wurde ausgewählt oder ist aktiv.
S_AnyModeSel	BOOL	FALSE	Gibt an, dass einer der 8 Modi ausgewählt und quittiert wurde. FALSE: Kein S_ModeX ausgewählt. TRUE: Einer der 8 S_ModeX ist ausgewählt und aktiv.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweis: Das X in den Parametern „S_ModeX“ oder „S_ModeXSel“ ist ein Platzhalter für die Zahlen 0 bis 7.

Typische Zeitdiagramme

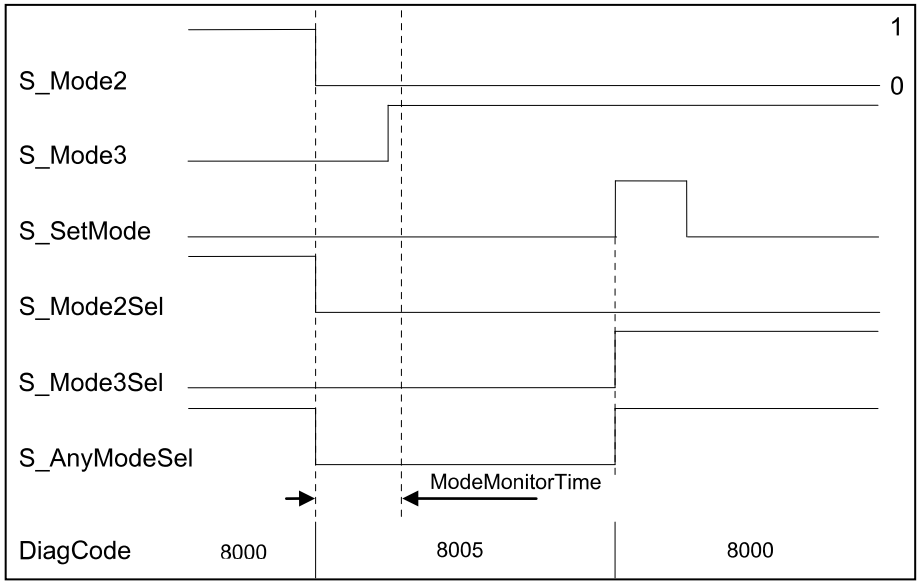


Abb. 95: Zeitdiagramm für SF_ModeSelector, gültige Änderung am Betriebsarten-Eingang mit Quittierung

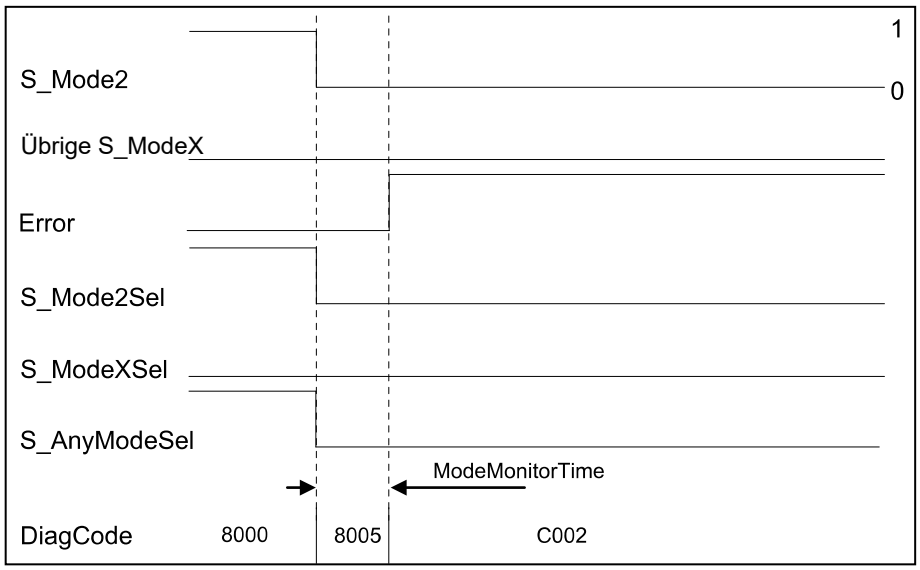


Abb. 96: Zeitdiagramm für SF_ModeSelector, Fehlerbedingung 2 an Betriebsarten-Eingängen

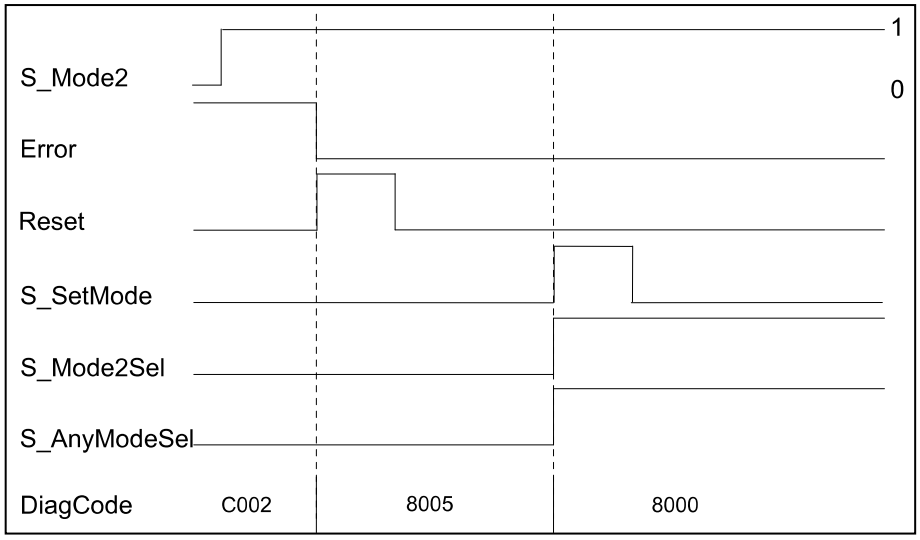


Abb. 97: Zeitdiagramm für SF_ModeSelector, Rücksetzen der Fehlerbedingung

Der Funktionsbaustein erkennt, wenn keiner der Betriebsarten-Eingänge gewählt wurde. Diese ungültige Bedingung wird nach Ablauf von ModeMonitorTime erkannt:

- Die Zeit startet mit jeder fallenden Flanke eines über S_ModeX geschalteten Betriebsarten-Eingangs neu
- Nach der Aktivierung des Funktionsbausteins liegt Zustand ModeChanged vor

Im Gegensatz dazu erkennt der Funktionsbaustein direkt, ob mehr als ein Betriebsarten-Eingang S_ModeX zur selben Zeit gewählt wurde.

Eine statische Rücksetzbedingung wird erkannt, wenn der Funktionsbaustein entweder im Fehlerzustand C001 oder C002 ist.

Verhalten im Fehlerfall

Bei einem Fehler werden die Ausgänge S_ModeXSel und S_AnyModeSel in den sicheren Zustand FALSE geschaltet. Der Ausgang DiagCode zeigt den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Ein Fehler muss mit einer steigenden Flanke des Eingangs Reset BOOL quittiert werden. Der Funktionsbaustein wechselt vom Zustand Error in den Zustand ModeChanged.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 28: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Error Kurzschluss	Der Funktionsbaustein hat erkannt, dass zwei oder mehr S_ModeX auf TRUE gesetzt sind, z. B. durch einen Kurzschluss. Ready = TRUE Error = TRUE S_AnyModeSel = FALSE Alle S_ModeXSel = FALSE
C002	Error Drahtbruch	Der Funktionsbaustein hat erkannt, dass alle S_ModeX auf FALSE gesetzt sind: Die Zeit nach einer fallenden S_ModeX-Flanke überschreitet ModeMonitorTime, z. B. durch Drahtbruch. Ready = TRUE Error = TRUE S_AnyModeSel = FALSE Alle S_ModeXSel = FALSE
C003	Fehler-Reset 1	Statisches Reset-Signal im Zustand C001. Ready = TRUE Error = TRUE S_AnyModeSel = FALSE Alle S_ModeXSel = FALSE
C004	Fehler-Reset 2	Statisches Reset-Signal im Zustand C002. Ready = TRUE Error = TRUE S_AnyModeSel = FALSE Alle S_ModeXSel = FALSE

Tab. 29: FB-spezifische Zustandscodes (kein Fehler):

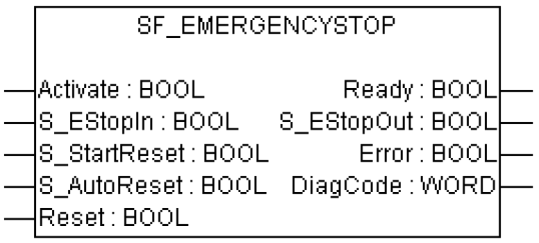
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE Error = FALSE S_AnyModeSel = FALSE Alle S_ModeXSel = FALSE
8005	Modus-Änderung	Zustand nach Aktivierung oder bei Änderung von S_ModeX (sofern nicht gesperrt) oder nach dem Rücksetzen eines Fehler-zustands. Ready = TRUE Error = FALSE S_AnyModeSel = FALSE Alle S_ModeXSel = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8000	Modusaus-wahl	Gültige Modusauswahl, aber noch nicht gesperrt. Ready = TRUE Error = FALSE S_AnyModeSel = TRUE S_ModeXSel = Auswahl X ist TRUE, andere sind FALSE.
8004	Modus gesperrt	Gültige Modusauswahl ist gesperrt. Ready = TRUE Error = FALSE S_AnyModeSel = TRUE S_ModeXSel = Auswahl X ist TRUE, andere sind FALSE.

4.6.4.5 SF_EmergencyStop

Normen	Anforderungen
EN 418:1992	3. Definitionen 4.1.12 … Das Rücksetzen des Steuergerätes selbst darf keinen Wiederanlaufbefehl auslösen.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4 Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen
IEC 60204-1, Ed. 5.0:2003	9.2.2. Stoppfunktionen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein ist ein sicherheitsrelevanter Funktionsbaustein für die Überwachung eines Not-Halt-Tasters. Dieser Funktionsbaustein kann für Not-Halt-Abschaltung (Not-Halt der Kategorie 0) oder — mit zusätzlicher Unterstützung der Peripherie — als Not-Halt (Not-Halt der Kategorien 1 oder 2) verwendet werden.

Tab. 30: FB-Name: SF_EmergencyStop

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_EStopIn	BOOL	FALSE	Eingang mit Sicherheitsanforderung. Variable. FALSE: Anforderung von sicherheitsgerichteter Antwort (z. B. Not-Halt-Taster betätigt). TRUE: Keine Anforderung von sicherheitsgerichteter Antwort (z. B. Not-Halt-Taster nicht betätigt).
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_EStopOut	BOOL	FALSE	Ausgang für die sicherheitsgerichtete Antwort. FALSE: Sicherheitsausgang deaktiviert. Anforderung von sicherheitsgerichteter Antwort (z. B. Not-Halt-Taster betätigt, Rücksetzen erforderlich oder interne Fehler liegen vor). TRUE: Sicherheitsausgang aktiviert. Keine Anforderung von sicherheitsgerichteter Antwort (z. B. Not-Halt-Taster nicht betätigt, keine internen Fehler).
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweis: Die folgenden in der Norm EN 418:1992 definierten Anforderungen müssen vom Anwender erfüllt werden (beim nachfolgenden Text handelt es sich um eine Übersetzung aus dem englischen Original der Norm):

- 4.1.4 Nach der Aktivierung des Stellglieds muss die Not-Aus-Ausrüstung in einer Weise funktionieren, dass die Gefahr abgewendet oder automatisch auf bestmögliche Art und Weise verringert wird.
- 4.1.7 Der Not-Halt-Befehl muss Vorrang vor allen anderen Befehlen haben.
- 4.1.12 Ein Rücksetzen des Steuergerätes darf nur als Folge einer manuellen Handlung am Steuergerät selbst möglich sein … Es darf nicht möglich sein, die Maschine neu zu starten, bevor alle betätigten Steuergeräte manuell, individuell und absichtlich zurückgesetzt wurden.

Typische Zeitdiagramme

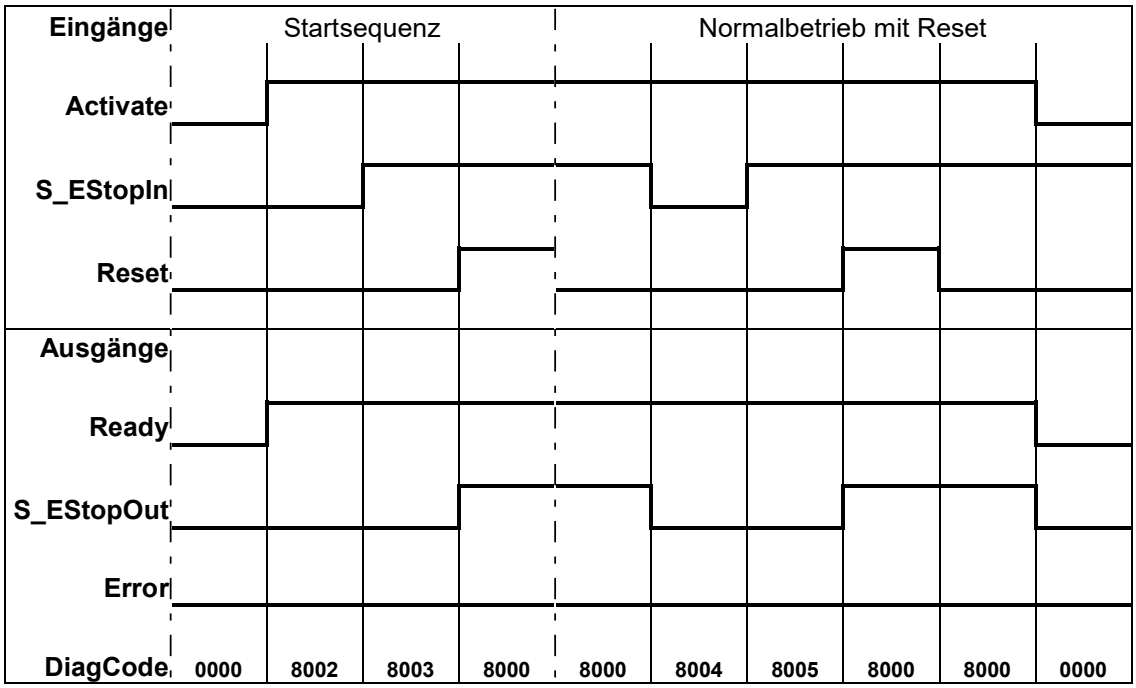


Abb. 98: Zeitdiagramm für SF_EmergencyStop: S_StartReset = FALSE; S_AutoReset = FALSE; Start, Reset, Normalbetrieb, Sicherheitsanforderung, Neustart

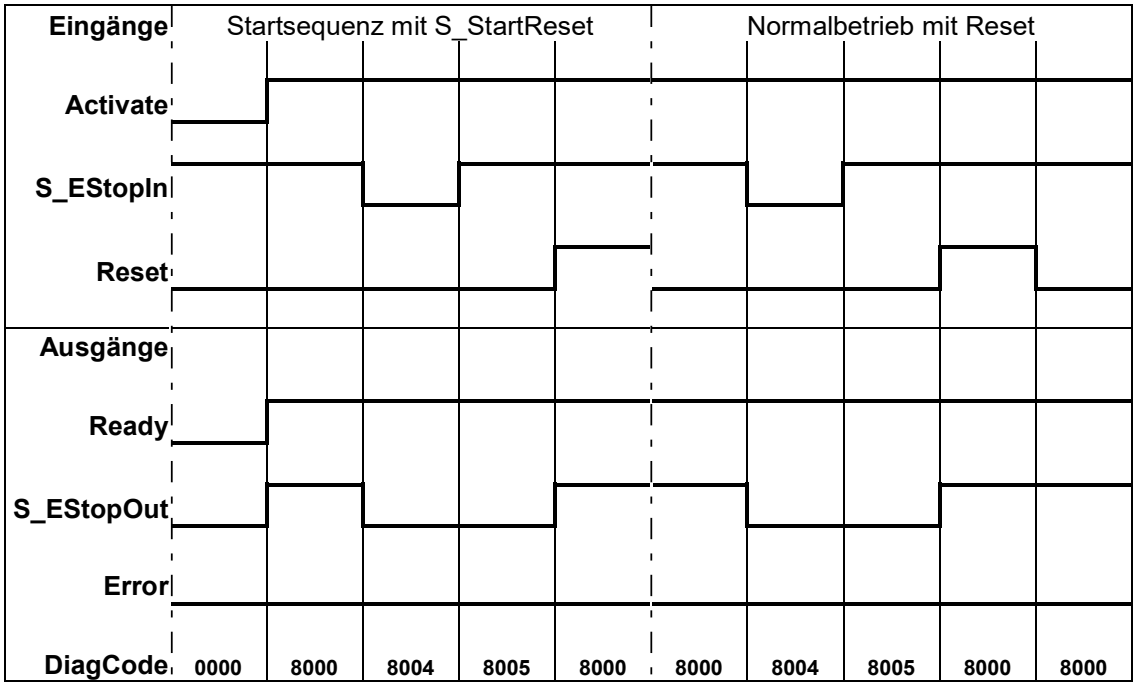


Abb. 99: Zeitdiagramm für SF_EmergencyStop: S_StartReset = TRUE; S_AutoReset = FALSE; Start, Normalbetrieb, Sicherheitsanforderung, Neustart

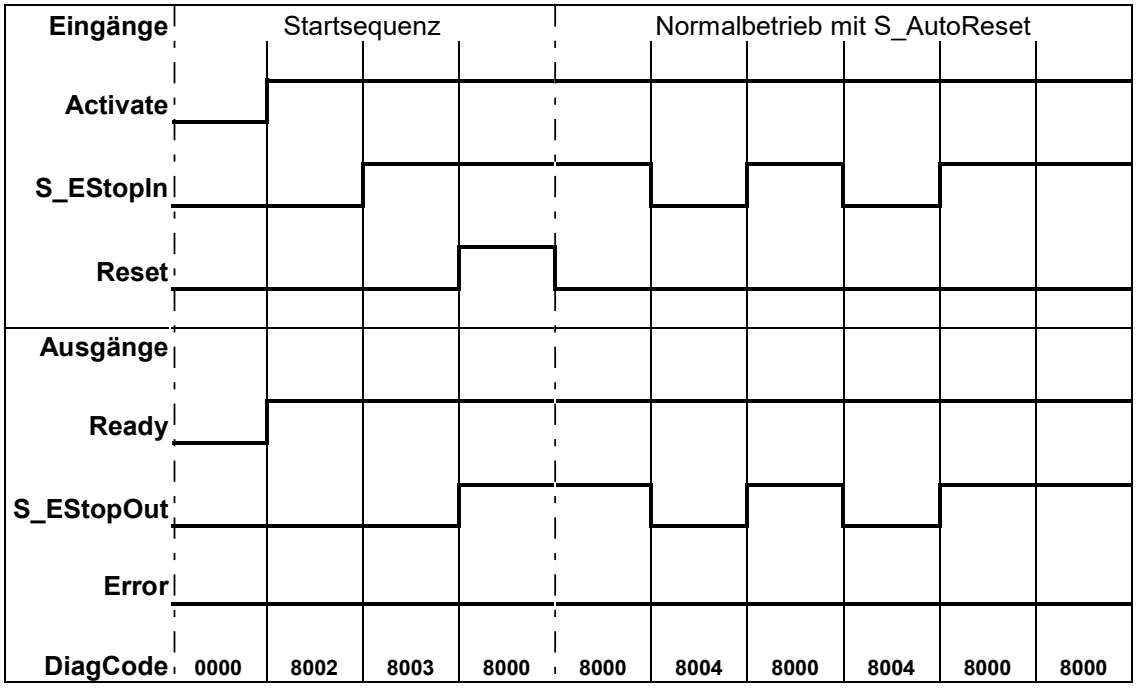


Abb. 100: Zeitdiagramm für SF_EmergencyStop: S_StartReset = FALSE; S_AutoReset = TRUE; Start, Normalbetrieb, Sicherheitsanforderung, Neustart

Der Funktionsbaustein erkennt ein statisches TRUE-Signal am RESET-Eingang.

Verhalten im Fehlerfall

S_EStopOut wird FALSE gesetzt. Bei einem statischen TRUE-Signal am RESET-Eingang zeigt der Ausgang DiagCode den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Zum Verlassen der Fehlerzustände muss Reset auf FALSE gesetzt werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 31: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Reset ist TRUE beim Warten auf S_EStopIn = TRUE. Ready = TRUE S_EStopOut = FALSE Error = TRUE
C002	Fehler-Reset 2	Reset ist TRUE beim Warten auf S_EStopIn = TRUE. Ready = TRUE S_EStopOut = FALSE Error = TRUE

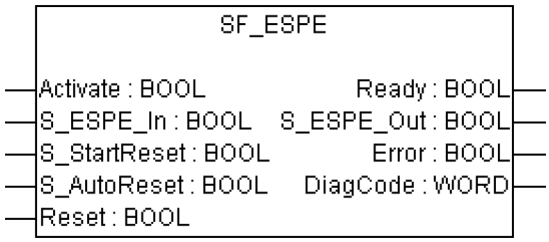
Tab. 32: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_EStopOut = FALSE Error = FALSE
8001	Init	Aktivierung ist TRUE. Der Funktionsbaustein wurde aktiviert. Prüfen Sie, ob S_StartReset erforderlich ist. Ready = TRUE S_EStopOut = FALSE Error = FALSE
8002	Warten auf S_EstopIn 1	Aktivierung ist TRUE. Prüfen Sie, ob Reset FALSE ist und warten Sie auf S_EStopIn = TRUE. Ready = TRUE S_EStopOut = FALSE Error = FALSE
8003	Warten auf Reset 1	Aktivierung ist TRUE. S_EStopIn = TRUE. Warten auf steigende Flanke von Reset. Ready = TRUE S_EStopOut = FALSE Error = FALSE
8004	Warten auf S_EstopIn 2	Aktivierung ist TRUE. Sicherheitsanforderung erkannt. Prüfen Sie, ob Reset FALSE ist und warten Sie auf S_EStopIn = TRUE. Ready = TRUE S_EStopOut = FALSE Error = FALSE
8005	Warten auf Reset 2	Aktivierung ist TRUE. S_EStopIn = TRUE. S_AutoReset prüfen oder auf steigende Flanke von Reset warten. Ready = TRUE S_EStopOut = FALSE Error = FALSE
8000	Sicherheits-ausgang aktiviert	Aktivierung ist TRUE. S_EStopIn = TRUE. Funktionsweise mit S_EStopOut = TRUE. Ready = TRUE S_EStopOut = TRUE Error = FALSE

4.6.4.6 SF_ESPE

Normen	Anforderungen
EN IEC 61496-1:2004	A.5.1 Anlaufsperre: Die Anlaufsperre muss verhindern, dass das (die) OSSD(s) in den EIN-Zustand geht (gehen), wenn die elektrische Versorgung eingeschaltet wird oder unterbrochen und wiederhergestellt wird. A.5.2: Ein Ausfall in der Anlaufsperre, der dazu führt, dass sie in den EIN-Zustand geht oder darin verbleibt, muss veranlassen, dass die BWS in den Verriegelungszustand geht oder dort verbleibt. A.6.1 Wiederanlaufsperre: … Der gesperrte Zustand muss beibehalten werden, bis die Wiederanlaufsperre manuell zurückgesetzt wird. Jedoch darf es nicht möglich sein, die Wiederanlaufsperre zurückzusetzen, während der Sensorteil aktiviert ist.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein ist ein sicherheitsrelevanter Funktionsbaustein für die Überwachung berührungslos wirkender Schutzeinrichtungen (BWS). Seine Funktion ist dieselbe wie bei SF_EmergencyStop. Das Ausgangssignal S_ESPE_Out wird FALSE, sobald der Eingang S_ESPE_In auf FALSE gesetzt wird. Das Ausgangssignal S_ESPE_Out wird nur TRUE, wenn der Eingang S_ESPE_In auf TRUE gesetzt wird und ein Reset durchgeführt wird. Die Reset-Aktivierung hängt von den definierten Eingängen S_StartReset, S_AutoReset und Reset ab.

Bei S_AutoReset = TRUE erfolgt eine automatische Quittierung.

Bei S_AutoReset = FALSE muss eine steigende Flanke am RESET-Eingang verwendet werden, um die Freigabe zu quittieren.

Bei S_StartReset = TRUE erfolgt eine automatische Quittierung, wenn PES zum 1. Mal gestartet wird.

Bei S_StartReset = FALSE muss eine steigende Flanke am RESET-Eingang verwendet werden, um die Freigabe zu quittieren.

Die Eingänge S_StartReset und S_AutoReset dürfen nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Die berührungslos wirkende Schutzeinrichtung (BWS) muss gemäß den Produktnormen EN IEC 61496-1, -2 und -3 und den erforderlichen Kategorien laut EN 954-1 gewählt werden.

Tab. 33: FB-Name: SF_ESPE

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↪ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_ESPE_In	BOOL	FALSE	Eingang mit Sicherheitsanforderung. Variable. FALSE: BWS betätigt, Anforderung von sicherheitsgerichteter Antwort. TRUE: BWS nicht betätigt, keine Anforderung von sicherheitsgerichteter Antwort. Die Sicherheitssteuerung muss fähig sein, eine sehr kurze Unterbrechung des Sensors zu erkennen (laut 61496-1: min. 80 ms), wenn die berührungslos wirkende Schutzeinrichtung als Auslöser in Anwendungen verwendet wird.
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_ESPE_Out	BOOL	FALSE	Ausgang für die sicherheitsgerichtete Antwort. FALSE: Sicherheitsausgang deaktiviert. Anforderung von sicherheitsgerichteter Antwort (z. B. Rücksetzen erforderlich oder interne Fehler liegen vor). TRUE: Sicherheitsausgang aktiviert. Keine Anforderung von sicherheitsgerichteter Antwort.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typische Zeitdiagramme

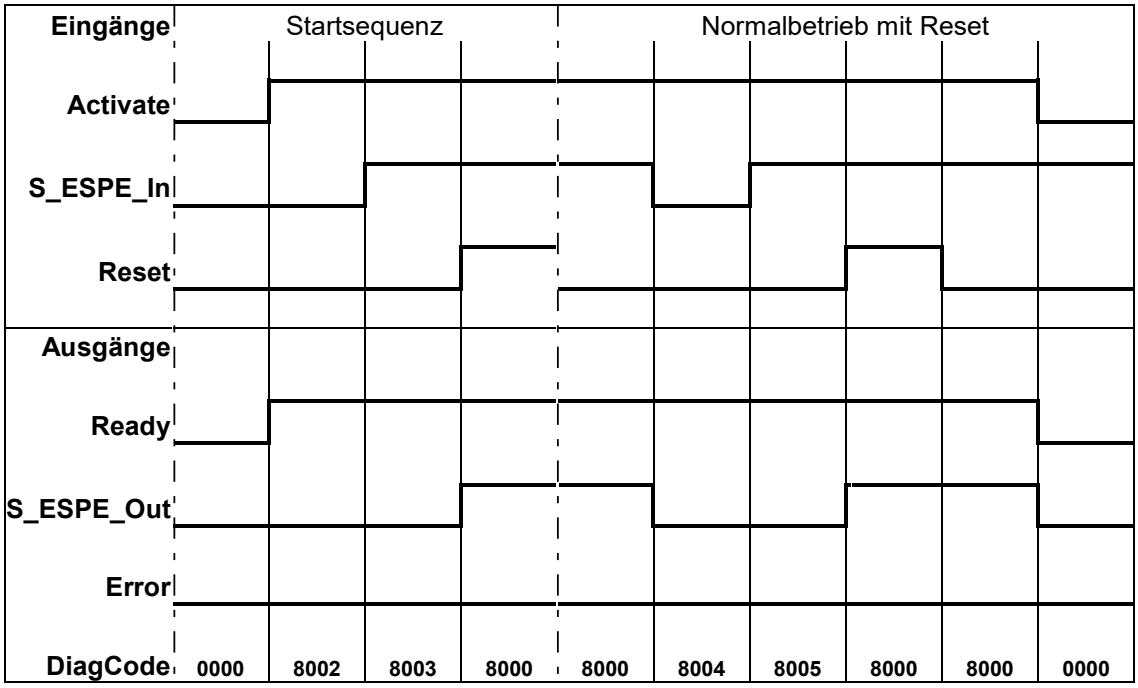


Abb. 101: Zeitdiagramm für SF_ESPE: S_StartReset = FALSE; S_AutoReset = FALSE; Start, Reset, Normalbetrieb, Sicherheitsanforderung, Neustart

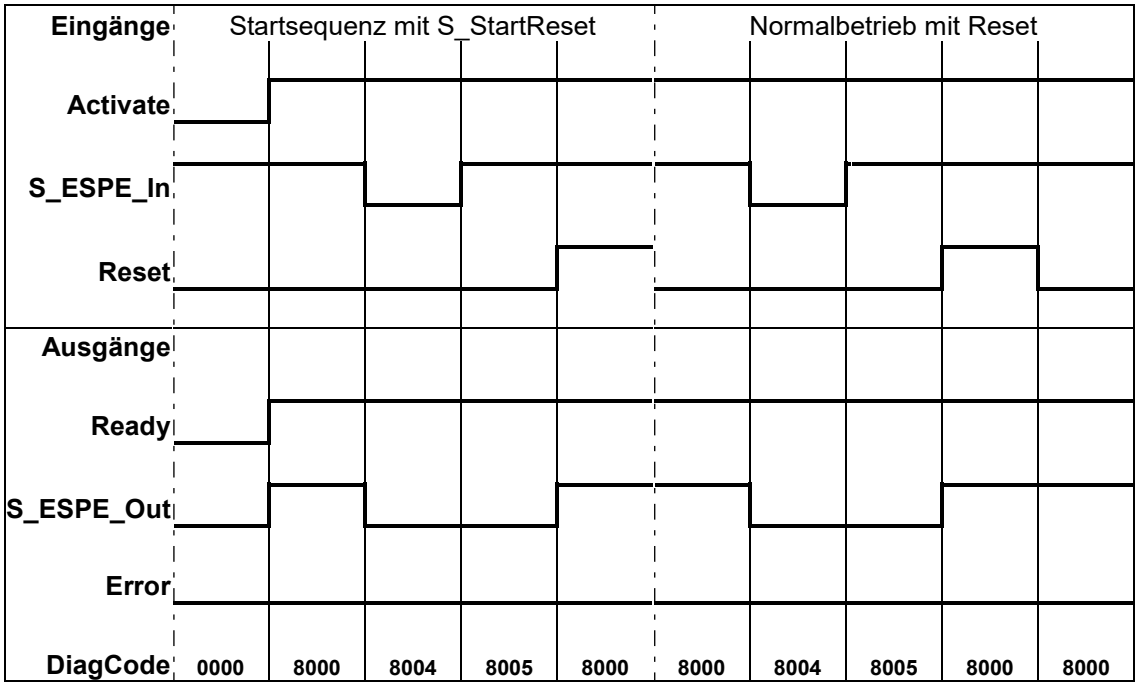


Abb. 102: Zeitdiagramm für SF_ESPE: S_StartReset = TRUE; S_AutoReset = FALSE; Start, Normalbetrieb, Sicherheitsanforderung, Neustart

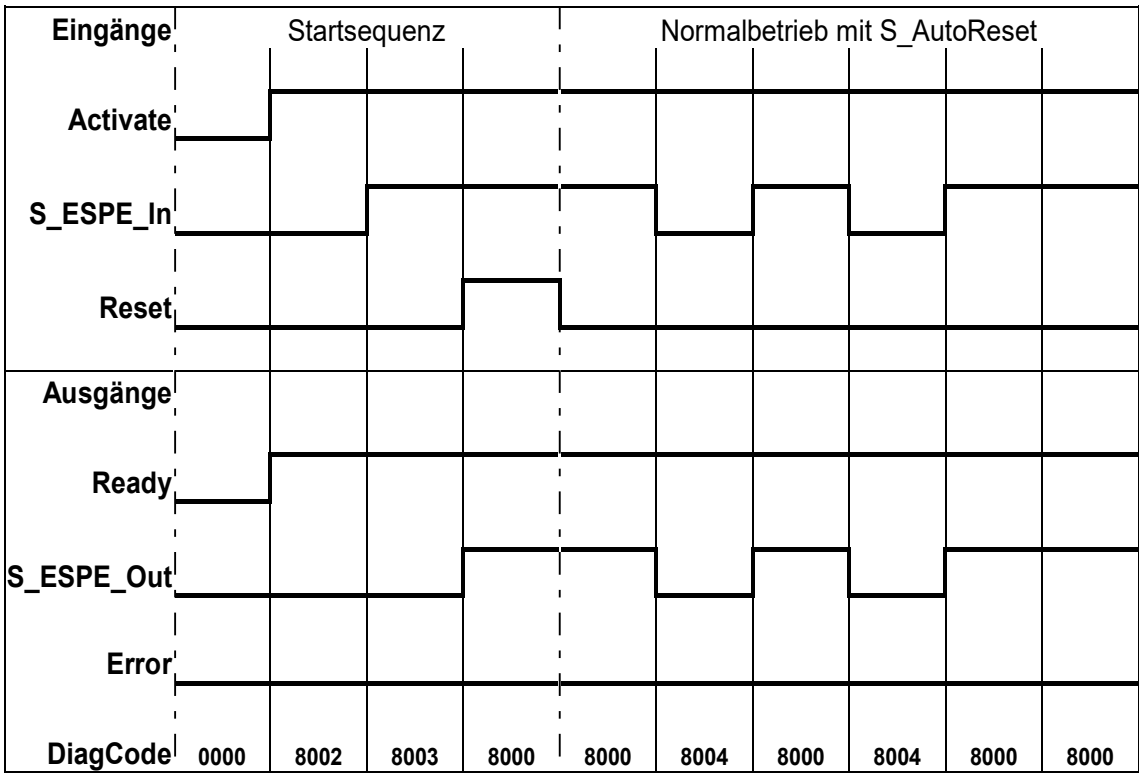


Abb. 103: Zeitdiagramm für SF_ESPE: S_StartReset = FALSE; S_AutoReset = TRUE; Start, Normalbetrieb, Sicherheitsanforderung, Neustart

Der Funktionsbaustein erkennt ein statisches TRUE-Signal am RESET-Eingang.

Verhalten im Fehlerfall

S_ESPE_Out wird auf FALSE gesetzt. Bei einem statischen TRUE-Signal am RESET-Eingang zeigt der Ausgang DiagCode den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Zum Verlassen der Fehlerzustände muss Reset auf FALSE gesetzt werden.

Fehler- und Zustandscodes des Funktions-bausteins

Tab. 34: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Reset ist TRUE beim Warten auf S_ESPEIn = TRUE. Ready = TRUE S_ESPE_Out = FALSE Error = TRUE
C002	Fehler-Reset 2	Reset ist TRUE beim Warten auf S_ESPEIn = TRUE. Ready = TRUE S_ESPE_Out = FALSE Error = TRUE

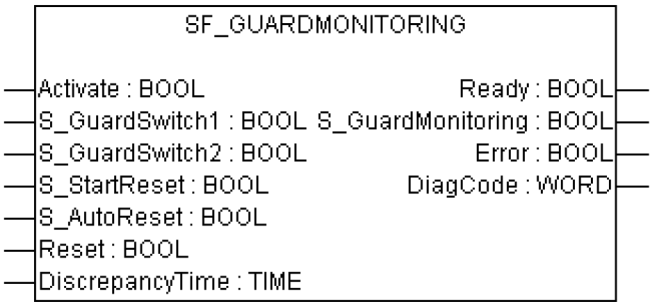
Tab. 35: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_ESPE_Out = FALSE Error = FALSE
8001	Init	Aktivierung ist TRUE. Der Funktionsbaustein wurde aktiviert. Prüfen Sie, ob S_StartReset erforderlich ist. Ready = TRUE S_ESPE_Out = FALSE Error = FALSE
8002	Warten auf S_ESPE_In 1	Aktivierung ist TRUE. Prüfen, ob Reset FALSE ist, und auf S_ESPEIn = TRUE warten. Ready = TRUE S_ESPE_Out = FALSE Error = FALSE
8003	Warten auf Reset 1	Aktivierung ist TRUE. S_ESPE_In = TRUE. Warten auf steigende Flanke von Reset. Ready = TRUE S_ESPE_Out = FALSE Error = FALSE
8004	Warten auf S_ESPE_In 2	Aktivierung ist TRUE. Sicherheitsanforderung erkannt. Prüfen, ob Reset FALSE ist, und auf S_ESPEIn = TRUE warten. Ready = TRUE S_ESPE_Out = FALSE Error = FALSE
8005	Warten auf Reset 2	Aktivierung ist TRUE. S_ESPE_In = TRUE. S_AutoReset prüfen oder auf steigende Flanke von Reset warten. Ready = TRUE S_ESPE_Out = FALSE Error = FALSE
8000	Sicherheits-ausgang aktiviert	Aktivierung ist TRUE. S_ESPE_In = TRUE. Funktionsweise mit S_ESPEOut = TRUE. Ready = TRUE S_ESPE_Out = TRUE Error = FALSE

4.6.4.7 SF_GuardMonitoring

Normen	Anforderungen
EN 953:1997	3.3.3 Trennende Schutzeinrichtung mit Startfunktion ● Die von der Schutzeinrichtung „abgedeckten“ gefährlichen Maschinenfunktionen können nicht ausgeführt werden, solange die Schutzeinrichtung geöffnet ist; ● Das Schließen der Schutzeinrichtung ermöglicht den Betrieb der gefährlichen Maschinenfunktion(en).
EN 1088:1995	3.2 Verriegelnde Schutzeinrichtung ● Die von der Schutzeinrichtung „abgedeckten“ gefährlichen Maschinenfunktionen können nicht ausgeführt werden, solange die Schutzeinrichtung geöffnet ist; ● Wenn die Schutzeinrichtung während der Ausführung der gefährlichen Maschinenfunktionen geöffnet wird, führt dies zur Ausgabe eines Stoppbefehls; ● Wenn die Schutzeinrichtung geschlossen ist, können die von der Schutzeinrichtung „abgedeckten“ gefährlichen Maschinenfunktionen ausgeführt werden, wobei das Schließen der Schutzeinrichtung selbst nicht deren Betrieb einleitet;
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4 Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein überwacht die relevante Schutzeinrichtung. Es gibt zwei unabhängige Eingangsparameter für zwei Schaltelemente an der Schutzeinrichtung, gekoppelt mit einer Zeitdifferenz (MonitoringTime) zum Schließen der Schutzeinrichtung.

Der Funktionsbaustein erfordert bei Schutzeinrichtungen mit zwei Schaltelementen zwei Eingänge, die die Position der Schutzeinrichtung anzeigen (laut EN 1088): die Eingänge DiscrepancyTime und Reset. Wenn die Schutzeinrichtung nur über ein Schaltelement verfügt, können die Eingänge S_GuardSwitch1 und S_GuardSwitch2 überbrückt werden. Die Überwachungszeit ist die maximale Zeit, die für eine Antwort beider Schaltelemente nach Schließen der Schutzeinrichtung erforderlich ist. Die Eingänge Reset, S_StartReset und S_AutoReset legen fest, wie der Funktionsbaustein zurückgesetzt wird, nachdem die Schutzeinrichtung geöffnet wurde.

Beim Öffnen der Schutzeinrichtung sollten die beiden Eingänge S_GuardSwitch1 und S_GuardSwitch2 auf FALSE schalten. Der Ausgang S_GuardMonitoring schaltet auf FALSE, sobald eines der Schaltelemente auf FALSE gesetzt wird. Beim Schließen der Schutzeinrichtung sollten die beiden Eingänge S_GuardSwitch1 und S_GuardSwitch2 auf TRUE schalten.

Dieser Funktionsbaustein überwacht die Symmetrie des Schaltverhaltens der zwei Schaltelemente. Der Ausgang S_GuardMonitoring bleibt FALSE, wenn nur einer der Kontakte das Öffnen/Schließen beendet hat.

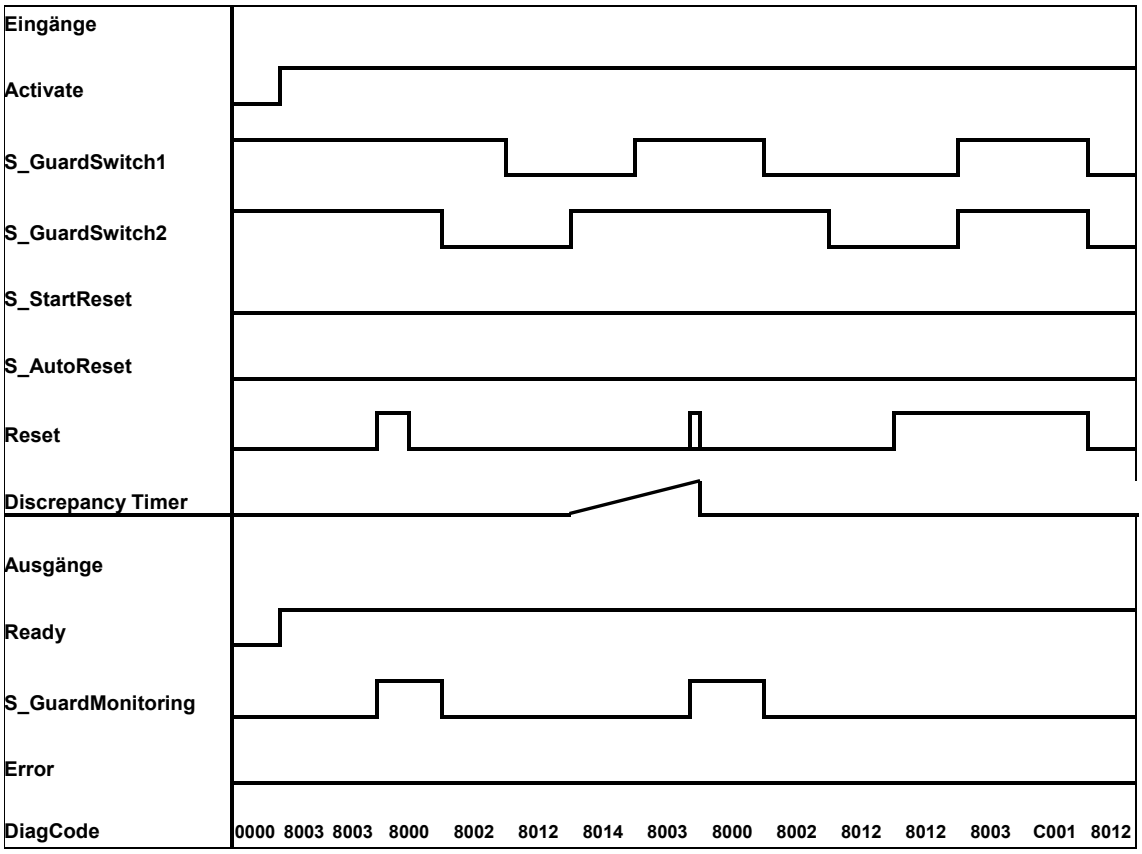
Das Verhalten des Ausgangs S_GuardMonitoring hängt von der Zeitdifferenz zwischen den schaltenden Eingängen ab. Die Diskrepanzzeit wird überwacht, sobald der Wert der beiden Eingänge S_GuardSwitch1 und S_GuardSwitch2 sich unterscheidet. Wenn DiscrepancyTime abgelaufen ist, aber die Eingänge sich weiterhin unterscheiden, bleibt der Ausgang S_GuardMonitoring FALSE. Wenn der zweite Eingang (S_GuardSwitch1 bzw. S_GuardSwitch2) innerhalb des für DiscrepancyTime festgelegten Wertes auf TRUE schaltet, wird der Ausgang S_GuardMonitoring nach Quittierung auch auf TRUE gesetzt.

Die Eingänge S_StartReset und S_AutoReset dürfen nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 36: FB-Name: SF_GuardMonitoring

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_GuardSwitch1	BOOL	FALSE	Variable. Eingang des Schaltelements 1 der Schutzeinrichtung. FALSE: Schutzeinrichtung ist offen. TRUE: Schutzeinrichtung ist geschlossen.
S_GuardSwitch2	BOOL	FALSE	Variable. Eingang des Schaltelements 2 der Schutzeinrichtung. FALSE: Schutzeinrichtung ist offen. TRUE: Schutzeinrichtung ist geschlossen.
DiscrepancyTime	TIME	T#0ms	Konstante. Konfiguriert die überwachte Synchronzeit zwischen S_GuardSwitch1 und S_GuardSwitch2.
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218* – nur Konstante
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218* – nur Konstante
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_GuardMonitoring	BOOL	FALSE	Ausgang, der den Zustand der Schutzeinrichtung angibt. FALSE: Schutzeinrichtung ist nicht aktiv. TRUE: Beide S_GuardSwitches sind TRUE, kein Fehler und Quittierung. Schutzeinrichtung ist aktiv.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typische Zeitdiagramme



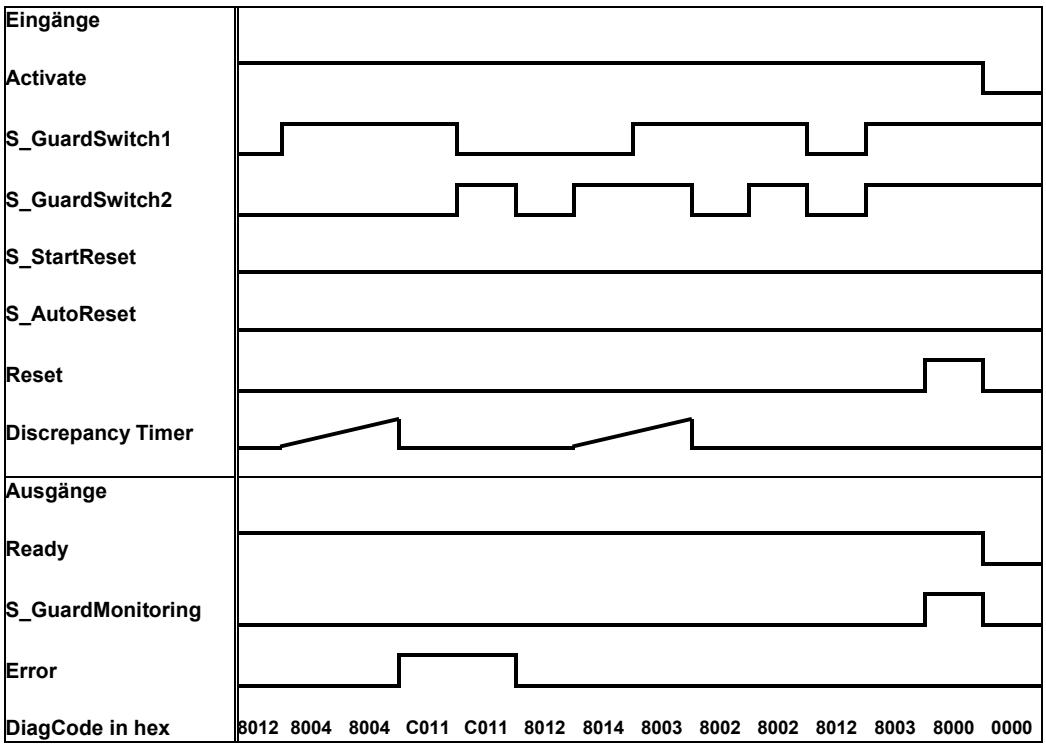


Abb. 104: Zeitdiagramme für SF_GuardMonitoring

Externe Signale: Der mechanische Aufbau kombiniert ein öffnendes und schließendes Schaltelement laut EN 954 (Schutzeinrichtung mit zwei Schaltelementen). Die Diskrepanzzeit für die Zeitspanne zwischen der mechanischen Reaktion der beiden Schaltelemente laut EN 954 (gilt als Erkennung eines „Anwendungsfehlers“, d. h. durch die Anwendung generiert) wird überwacht.

Ein Fehler wird festgestellt, wenn die Zeitspanne zwischen erstem S_GuardSwitch1/S_GuardSwitch2-Eingang und dem zweiten länger ist als der Wert für den DiscrepancyTime-Eingang. Der Fehlerausgang wird auf TRUE gesetzt.

Der Funktionsbaustein erkennt ein statisches TRUE-Signal am RESET-Eingang.

Verhalten im Fehlerfall und bei Reset

Der Ausgang S_GuardMonitoring wird auf FALSE gesetzt. Wenn die zwei Eingänge S_GuardSwitch1 und S_GuardSwitch2 überbrückt sind, wird kein Fehler festgestellt. Um den Zustand „Fehler-Reset“ zu verlassen, muss der RESET-Eingang auf FALSE gesetzt werden. Um die Diskrepanzzeit-Fehler zu verlassen, müssen die beiden Eingänge S_GuardSwitch1 und 2 beide FALSE sein.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 37: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset	Statisches Reset im Zustand 8003 erkannt. Ready = TRUE S_GuardMonitoring = FALSE Error = TRUE
C011	Diskrepanzzeit-Fehler 1	DiscrepancyTime abgelaufen im Zustand 8004. Ready = TRUE S_GuardMonitoring = FALSE Error = TRUE
C012	Diskrepanzzeit-Fehler 2	DiscrepancyTime abgelaufen im Zustand 8014. Ready = TRUE S_GuardMonitoring = FALSE Error = TRUE

Tab. 38: FB-spezifische Zustandscodes (kein Fehler):

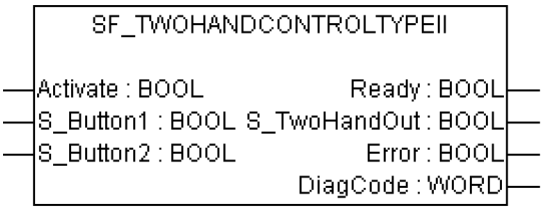
DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_GuardMonitoring = FALSE Error = FALSE
8000	Normal	Schutzeinrichtung geschlossen und sicherer Zustand quittiert. Ready = TRUE S_GuardMonitoring = TRUE Error = FALSE
8001	Init	Der Funktionsbaustein wurde aktiviert. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8002	Anforderung für offene Schutzein-richtung	Komplette Schaltfolge erforderlich. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE
8003	Warten auf Reset	Warten auf steigende Flanke an Reset. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE
8012	Schutzein-richtung offen	Schutzeinrichtung komplett geöffnet. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE
8004	Warten auf GuardS-witch2	S_GuardSwitch1 wurde auf TRUE gesetzt – warten auf S_GuardSwitch2; Diskrepanz-Timer gestartet. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE
8014	Warten auf GuardS-witch1	S_GuardSwitch2 wurde auf TRUE gesetzt – warten auf S_GuardSwitch1; Diskrepanz-Timer gestartet. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE
8005	Schutzein-richtung geschlossen	Schutzeinrichtung ist geschlossen. Warten auf Reset, wenn S_AutoReset = FALSE. Ready = TRUE S_GuardMonitoring = FALSE Error = FALSE

4.6.4.8 SF_TwoHandControlTypeII

Normen	Anforderungen
EN 574:1996	Satz 4, Tabelle 1, Typ II. 5.1 Verwendung beider Hände / gleichzeitige Betätigung. 5.2 Zusammenhang zwischen Ausgangs- und Eingangssignalen. 5.3 Abschluss des Ausgangssignals. 5.6 Wiederaufnahme des Ausgangssignals. 6.3 Anwendung von DIN EN 954-1 Kategorie 3 (kann nur mit Öffnern und Schließern zusammen mit antivalenter Verarbeitung umgesetzt werden)
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein bietet die Funktionalität der Zweihandbedienung (siehe EN 574, Abschnitt 4, Typ II).

Dieser Funktionsbaustein bietet die Funktionalität der Zweihandbedienung (siehe EN 574, Abschnitt 4, Typ II). Wenn S_Button1 und S_Button2 in einer korrekten Reihenfolge auf TRUE gesetzt werden, wird der Ausgang S_TwoHandOut auch auf TRUE gesetzt. Der Funktionsbaustein kontrolliert auch das Loslassen der beiden Taster, bevor der Ausgang S_TwoHandOut erneut auf TRUE gesetzt wird.

Tab. 39: FB-Name: SF_TwoHandControlTypeII

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_Button1	BOOL	FALSE	Variable. Eingang von Taster 1 (für Kategorie 3 oder 4: zwei antivalente Kontakte) FALSE: Taster 1 losgelassen. TRUE: Taster 1 betätigt.
S_Button2	BOOL	FALSE	Variable. Eingang von Taster 2 (für Kategorie 3 oder 4: zwei antivalente Kontakte) FALSE: Taster 2 losgelassen. TRUE: Taster 2 betätigt.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_TwoHandOut	BOOL	FALSE	Sicherheitsgerichtetes Ausgangssignal. FALSE: Keine korrekte Zweihandbedienung. TRUE: Die Eingänge S_Button1 und S_Button2 sind TRUE und es liegt kein Fehler vor. Korrekte Zweihandbedienung.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweise: Kein RESET-Eingang oder Error-Ausgang erforderlich, weil kein Test an beiden Schaltelementen durchgeführt werden kann.

Typisches Zeitdiagramm

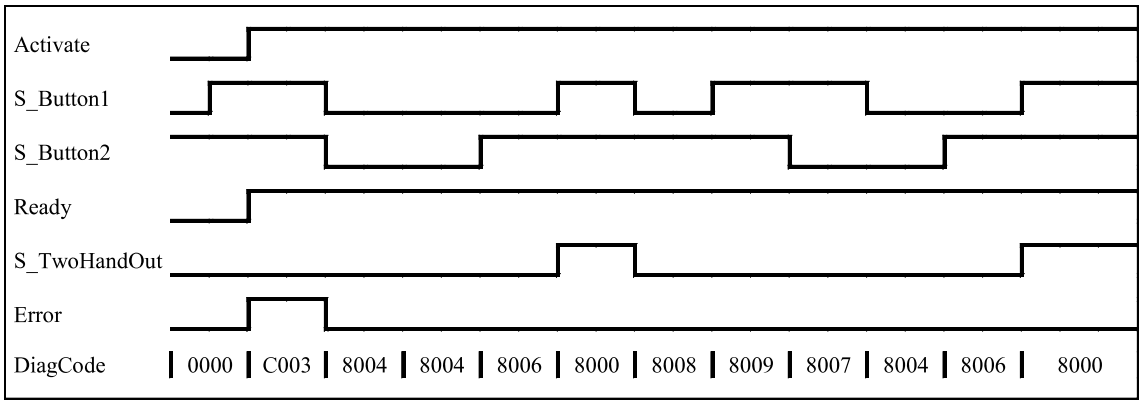


Abb. 105: Zeitdiagramm für SF_TwoHandControlTypeII

Nach Aktivierung des Funktionsbausteins wird jeder Taster, für den es ein TRUE-Signal gibt, als ungültige Einstellung des Eingangs erkannt, was zu einem Fehler führt.

Verhalten im Fehlerfall

Bei einem Fehler wird der Ausgang S_TwoHandOut auf FALSE gesetzt und bleibt in diesem sicheren Zustand.

Der Fehlerzustand ist beendet, wenn beide Taster losgelassen (auf FALSE gesetzt) werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 40: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler B1	S_Button1 war TRUE bei Aktivierung des Funktionsbausteins. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE
C002	Fehler B2	S_Button2 war TRUE bei Aktivierung des Funktionsbausteins. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE
C003	Fehler B1&B2	Die Signale an S_Button1 und S_Button2 waren TRUE bei Aktivierung des Funktionsbausteins. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE

Tab. 41: FB-spezifische Zustandscodes (kein Fehler):

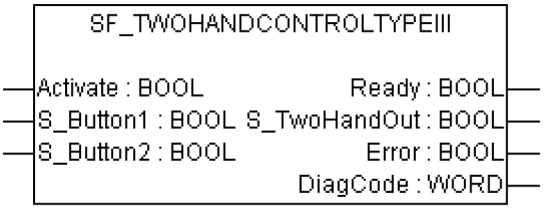
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE Error = FALSE S_TwoHandOut = FALSE
8000	Taster betätigt	Beide Taster korrekt betätigt. Der Sicherheitsausgang wird aktiviert. Ready = TRUE Error = FALSE S_TwoHandOut = TRUE
8001	Init	Der Funktionsbaustein ist aktiv, aber im Zustand Init. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8004	Taster losgelassen	Kein Taster betätigt. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8005	Taster 1 betätigt	Nur Taster 1 ist betätigt. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8006	Taster 2 betätigt	Nur Taster 2 ist betätigt. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8007	Taster 2 losgelassen	Der Sicherheitsausgang wurde aktiviert und wieder deaktiviert. FALSE bei S_Button1 und S_Button2 wurde nicht erreicht nach dem Deaktivieren des Sicherheitsausgangs. In diesem Zustand ist S_Button1 TRUE und S_Button2 FALSE nach dem Deaktivieren des Sicherheitsausgangs. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8008	Taster 1 losgelassen	Der Sicherheitsausgang wurde aktiviert und wieder deaktiviert. FALSE bei S_Button1 und S_Button2 wurde nicht erreicht nach dem Deaktivieren des Sicherheitsausgangs. In diesem Zustand ist S_Button1 FALSE und S_Button2 TRUE nach dem Deaktivieren des Sicherheitsausgangs. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8009	Verriegelt Aus	Der Sicherheitsausgang wurde aktiviert und wieder deaktiviert. FALSE bei S_Button1 und S_Button2 wurde nicht erreicht nach dem Deaktivieren des Sicherheitsausgangs. In diesem Zustand ist S_Button1 TRUE und S_Button2 TRUE nach dem Deaktivieren des Sicherheitsausgangs. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8019	Verriegelt Ein	Nicht korrekte Betätigung der Taster. Warten auf das Loslassen beider Taster. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE

4.6.4.9 SF_TwoHandControlTypeIII

Normen	Anforderungen
EN 574:1996	Satz 4, Tabelle 1, Typ III A; B; C. 5.1 Verwendung beider Hände / gleichzeitige Betätigung. 5.2 Zusammenhang zwischen Ausgangs- und Eingangssignalen. 5.3 Abschluss des Ausgangssignals. 5.6 Wiederaufnahme des Ausgangssignals. 5.7 Synchrone Betätigung. 6.2 Anwendung von DIN EN 954-1 Kategorie 1. 6.3 Anwendung von DIN EN 954-1 Kategorie 3 (Kann nur mit Öffnern und Schließern zusammen mit antivalenter Verarbeitung umgesetzt werden) 6.4 Anwendung von DIN EN 954-1 Kategorie 4 (Kann nur mit Öffnern und Schließern zusammen mit antivalenter Verarbeitung umgesetzt werden)
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein bietet die Funktionalität der Zweihandbedienung (siehe EN 574, Abschnitt 4, Typ III. Der fest definierte Zeitunterschied ist 500 ms).

Dieser Funktionsbaustein bietet die Funktionalität der Zweihandbedienung (siehe EN 574, Abschnitt 4, Typ III). Wenn S_Button1 und S_Button2 in einer korrekten Reihenfolge innerhalb von 500 ms auf TRUE gesetzt werden, wird der Ausgang S_TwoHandOut auch auf TRUE gesetzt. Der Funktionsbaustein kontrolliert auch das Loslassen der beiden Taster, bevor der Ausgang S_TwoHandOut erneut auf TRUE gesetzt wird.

Tab. 42: FB-Name: SF_TwoHandControlTypeIII

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_Button1	BOOL	FALSE	Variable. Eingang von Taster 1 (für Kategorie 3 oder 4: zwei antivalente Kontakte) FALSE: Taster 1 losgelassen. TRUE: Taster 1 betätigt.
S_Button2	BOOL	FALSE	Variable. Eingang von Taster 2 (für Kategorie 3 oder 4: zwei antivalente Kontakte) FALSE: Taster 2 losgelassen. TRUE: Taster 2 betätigt.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_TwoHandOut	BOOL	FALSE	Sicherheitsgerichtetes Ausgangssignal. FALSE: Keine korrekte Zweihandbedienung. TRUE: Die Eingänge S_Button1 und S_Button2 wechselten innerhalb von 500 ms von FALSE auf TRUE und kein Fehler lag vor. Die Zweihandbedienung erfolgte korrekt.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweise: Kein RESET-Eingang oder Error-Ausgang erforderlich, weil kein Test an beiden Schaltelementen durchgeführt werden kann.

Typisches Zeitdiagramm

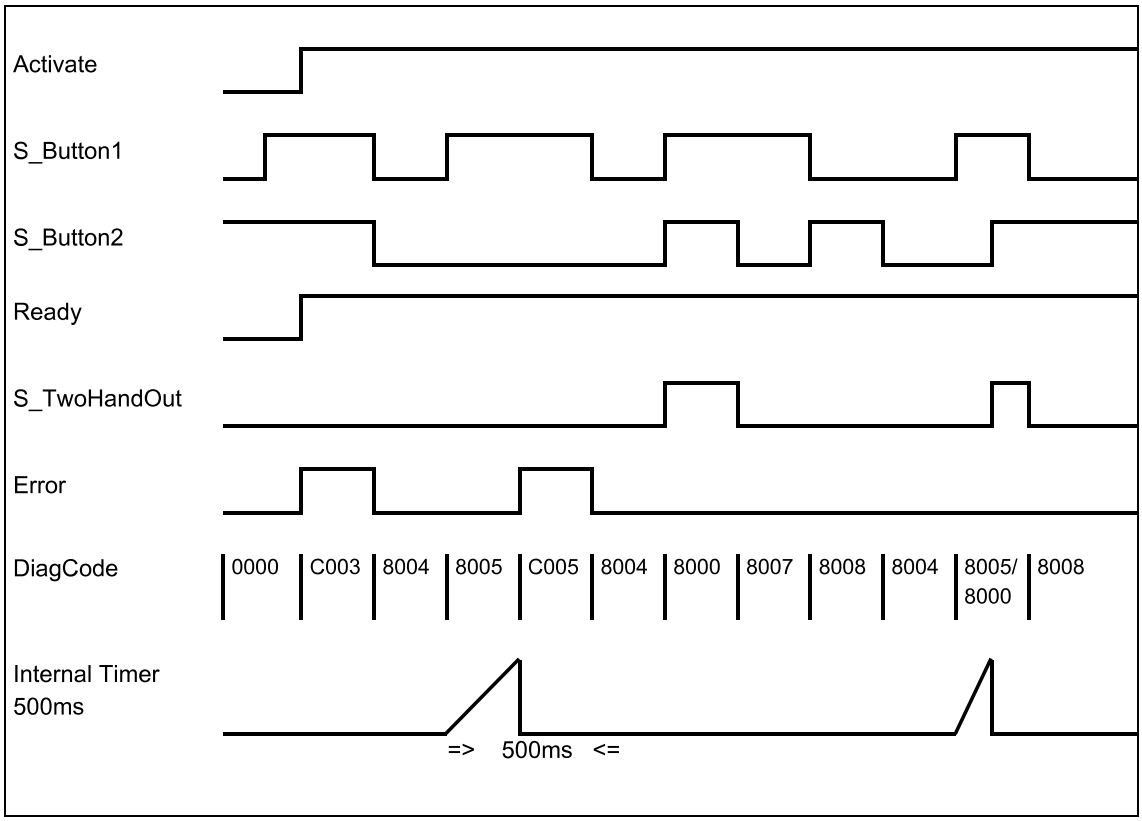


Abb. 106: Zeitdiagramm für SF_TwoHandControlTypeIII

Nach Aktivierung des Funktionsbausteins wird jeder Taster, für den es ein TRUE-Signal gibt, als ungültige Einstellung des Eingangs erkannt, was zu einem Fehler führt. Der Funktionsbaustein erkennt, wenn der Unterschied zwischen den Eingangssignalen mehr als 500 ms beträgt.

Verhalten im Fehlerfall

Bei einem Fehler wird der Ausgang S_TwoHandOut auf FALSE gesetzt und bleibt in diesem sicheren Zustand.

Der Fehlerzustand ist beendet, wenn beide Taster losgelassen (auf FALSE gesetzt) werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 43: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler 1 B1	S_Button1 war TRUE bei Aktivierung des Funktionsbausteins. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE
C002	Fehler 1 B2	S_Button2 war TRUE bei Aktivierung des Funktionsbausteins. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE
C003	Fehler 1 B1&B2	Die Signale an S_Button1 und S_Button2 waren TRUE bei Aktivierung des Funktionsbausteins. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C004	Fehler 2 B1	S_Button1 war FALSE und S_Button2 war TRUE nach 500 ms in Zustand 8005. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE
C005	Fehler 2 B2	S_Button1 war TRUE und S_Button2 war FALSE nach 500 ms in Zustand 8005. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE
C006	Fehler 2 B1&B2	S_Button1 war TRUE und S_Button2 war TRUE nach 500 ms in Zustand 8005 oder 8006. Dieser Zustand ist nur möglich, wenn die Zustände der Eingänge (S_Button1 und S_Button2) simultan von divergent auf konvergent (beide TRUE) schalten und wenn der Timer (500 ms) im gleichen Zyklus abgelaufen ist. Ready = TRUE Error = TRUE S_TwoHandOut = FALSE

Tab. 44: FB-spezifische Zustandscodes (kein Fehler):

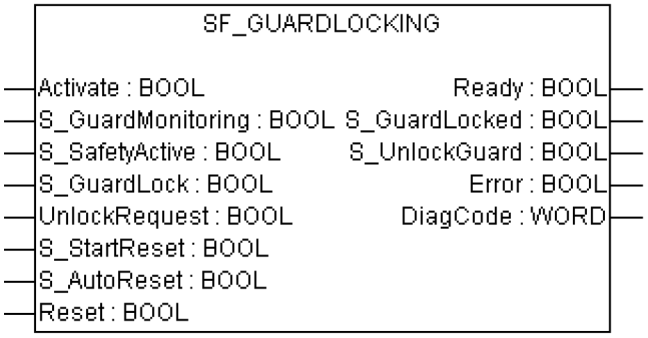
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE Error = FALSE S_TwoHandOut = FALSE
8000	Taster betätigt	Beide Taster korrekt betätigt. Der Sicherheitsausgang wird aktiviert. Ready = TRUE Error = FALSE S_TwoHandOut = TRUE
8001	Init	Der Funktionsbaustein ist aktiv, aber im Zustand Init. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8004	Taster losgelassen	Kein Taster betätigt. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8005	Taster 1 betätigt	Nur Taster 1 ist betätigt. Überwachungs-Timer starten. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8006	Taster 2 betätigt	Nur Taster 2 ist betätigt. Überwachungs-Timer starten. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8007	Taster 2 los-gelassen	Der Sicherheitsausgang wurde aktiviert und wieder deaktiviert. FALSE bei S_Button1 und S_Button2 wurde nicht erreicht nach dem Deaktivieren des Sicherheitsausgangs. In diesem Zustand ist S_Button1 TRUE und S_Button2 FALSE nach dem Deaktivieren des Sicherheitsausgangs. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8008	Taster 1 los-gelassen	Der Sicherheitsausgang wurde aktiviert und wieder deaktiviert. FALSE bei S_Button1 und S_Button2 wurde nicht erreicht nach dem Deaktivieren des Sicherheitsausgangs. In diesem Zustand ist S_Button1 FALSE und S_Button2 TRUE nach dem Deaktivieren des Sicherheitsausgangs. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8009	Verriegelt Aus	Der Sicherheitsausgang wurde aktiviert und wieder deaktiviert. FALSE bei S_Button1 und S_Button2 wurde nicht erreicht nach dem Deaktivieren des Sicherheitsausgangs. In diesem Zustand ist S_Button1 TRUE und S_Button2 TRUE nach dem Deaktivieren des Sicherheitsausgangs. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE
8019	Verriegelt Ein	Nicht korrekte Betätigung der Taster. Warten auf das Loslassen beider Taster. Ready = TRUE Error = FALSE S_TwoHandOut = FALSE

4.6.4.10 SF_GuardLocking

Normen	Anforderungen
EN 953:1997	3.3.3 Trennende Schutzeinrichtung mit Startfunktion ● Die von der Schutzeinrichtung „abgedeckten“ gefährlichen Maschinenfunktionen können nicht ausgeführt werden, solange die Schutzeinrichtung geöffnet ist; ● Das Schließen der Schutzeinrichtung ermöglicht den Betrieb der gefährlichen Maschinenfunktion(en).
EN 1088:1995	3.3 Definition: Verriegelnde Schutzeinrichtung mit Zuhaltung ● Die von der Schutzeinrichtung „abgedeckten“ gefährlichen Maschinenfunktionen können nicht ausgeführt werden, solange die Schutzeinrichtung nicht geschlossen und verriegelt ist; ● Die Schutzeinrichtung bleibt geschlossen, bis keine Verletzungsgefahr durch die gefährlichen Maschinenfunktionen mehr besteht; ● Wenn die Schutzeinrichtung geschlossen und verriegelt ist, können die von der Schutzeinrichtung „abgedeckten“ gefährlichen Maschinenfunktionen ausgeführt werden, wobei das Schließen der Schutzeinrichtung selbst nicht deren Betrieb einleitet. 4.2.2 – Verriegelnde Schutzeinrichtung mit Zuhaltung Bedingtes Entriegeln („Four-State Interlocking“), siehe Abb. 3 b2 in der Norm)
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein kontrolliert einen Eingang zu einem Gefahrenbereich mit einer verriegelten trennenden Schutzeinrichtung mit Zuhaltung („Four-State Interlocking“).

Die Funktion kontrolliert die Zuhaltung und überwacht die Position der Schutzeinrichtung und der Verriegelung. Dieser Funktionsbaustein kann mit einem mechanischen Sperrschalter verwendet werden.

Der Bediener möchte Zugang zum Gefahrenbereich erhalten. Die Schutzeinrichtung kann nur entriegelt werden, wenn der gefährliche Bereich in einem sicheren Zustand ist. Die Schutzeinrichtung kann verriegelt werden, wenn sie geschlossen ist. Die Maschine kann gestartet werden, wenn die Schutzeinrichtung geschlossen und verriegelt ist. Eine offene oder nicht verriegelte Schutzeinrichtung wird bei einer sicherheitskritischen Situation erkannt.

Die Eingänge S_StartReset und S_AutoReset dürfen nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 45: FB-Name: SF_GuardLocking

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_GuardMonitoring	BOOL	FALSE	Variable. Überwacht die Verriegelung der Schutzeinrichtung. FALSE: Schutzeinrichtung ist offen. TRUE: Schutzeinrichtung ist geschlossen.
S_SafetyActive	BOOL	FALSE	Variable. Zustand des Gefahrenbereichs (EDM), z. B. basierend auf Geschwindigkeitsüberwachung oder Verzögerung nach sicherer Abschaltung. FALSE: Maschine im nicht sicheren Zustand. TRUE: Maschine im sicheren Zustand.
S_GuardLock	BOOL	FALSE	Variable. Zustand der mechanischen Verriegelung. FALSE: Schutzeinrichtung ist nicht verriegelt. TRUE: Schutzeinrichtung ist verriegelt.
UnlockRequest	BOOL	FALSE	Variable. Bedienereingriff – Entriegelung der Schutzeinrichtung wurde angefordert. FALSE: Keine Anforderung. TRUE: Anforderung.
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218* Wird auch verwendet, um eine erneute Verriegelung der Schutzeinrichtung anzufordern. Die Qualität des Signals muss dem für ein manuelles Rücksetzgerät entsprechen (siehe EN 954-1, Kapitel 5.4)
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_GuardLocked	BOOL	FALSE	Schnittstelle zum Gefahrenbereich, in dem die gefährlichen Bewegungen gestoppt werden sollen. FALSE: Kein sicherer Zustand. TRUE: Sicherer Zustand.
S_UnlockGuard	BOOL	FALSE	Signal zur Entriegelung der Schutzeinrichtung. FALSE: Schutzeinrichtung schließen. TRUE: Schutzeinrichtung entriegeln.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typisches Zeitdiagramm

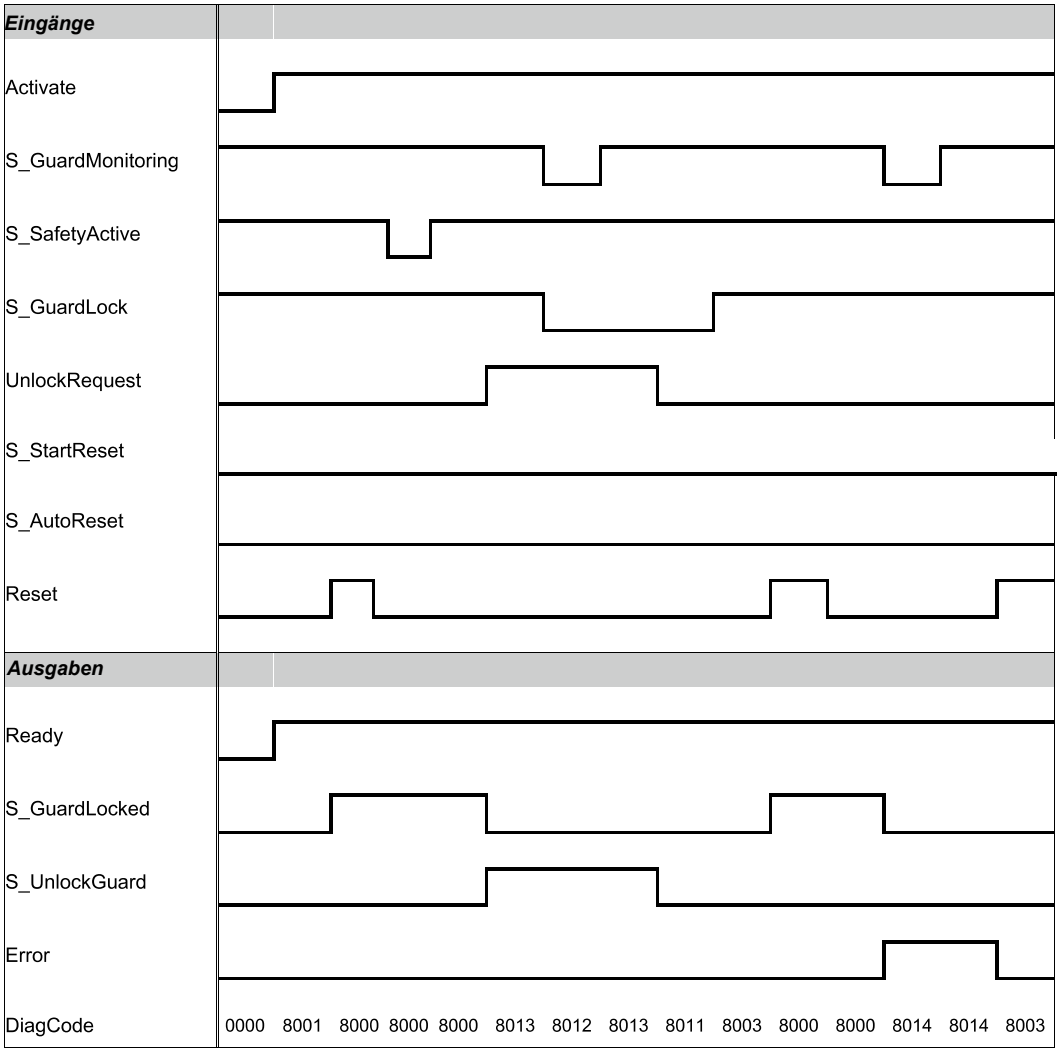


Abb. 107: Zeitdiagramm für SF_GuardLocking

Statische Signale wurden an Reset erkannt. Fehler wurden an den Schaltelementen der Schutzeinrichtung erkannt.

Verhalten im Fehlerfall

Bei einem Fehler werden die Ausgänge S_GuardLocked und S_UnlockGuard auf FALSE gesetzt; der Ausgang DiagCode zeigt den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Ein Fehler muss mit einer steigenden Flanke am RESET-Eingang quittiert werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 46: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Statisches Reset im Zustand 8001 erkannt. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = TRUE
C002	Fehler-Reset 2	Statisches Reset im Zustand C004 erkannt. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = TRUE
C003	Fehler-Reset 3	Statisches Reset im Zustand 8011 erkannt. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = TRUE
C004	Keine Sicherheit	Keine Sicherheit mehr; Schutzeinrichtung offen oder nicht verriegelt. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = TRUE

Tab. 47: FB-spezifische Zustandscodes (kein Fehler):

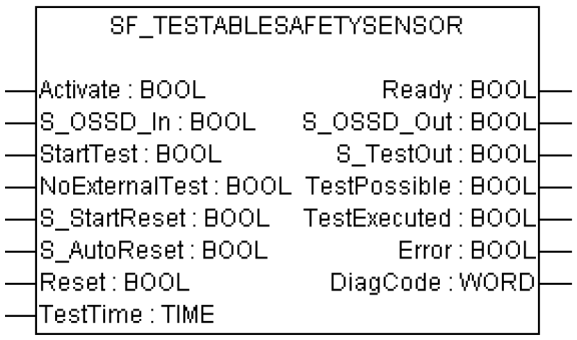
DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = FALSE
8000	Schutzeinrichtung geschlossen und verriegelt	Schutzeinrichtung ist verriegelt. Ready = TRUE S_GuardLocked = TRUE S_UnlockGuard = FALSE Error = FALSE
8001	Init	Der Funktionsbaustein wurde aktiviert und initialisiert. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8003	Warten auf Reset	Die Tür ist geschlossen und verriegelt; warten auf Reset des Bedieners. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = FALSE
8011	Warten auf Bediener	Warten auf Anforderung des Bedieners für Entriegelung oder Reset. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = FALSE
8012	Schutzein-richtung offen und entriegelt	Verriegelung wurde gelöst und Schutzeinrichtung ist offen. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = TRUE Error = FALSE
8013	Schutzein-richtung geschlossen aber entrie-gelt	Verriegelung wurde gelöst, aber Schutzeinrichtung ist geschlossen. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = TRUE Error = FALSE
8014	Rückgabe von Sicher-heit	Rückgabe des Signals S_SafetyActive, warten auf Bedienerquit-tierung. Ready = TRUE S_GuardLocked = FALSE S_UnlockGuard = FALSE Error = FALSE

4.6.4.11 SF_TestableSafetySensor

Normen	Anforderungen
IEC 61496-1:2004	4.2.2.3 Spezielle Anforderungen für BWS vom Typ 2 Eine BWS vom Typ 2 muss über Einrichtungen für regelmäßige Prüfungen auf Gefahren verfügen (z. B. Verlust der Erkennungsfunktion, Antwortzeit über Maximalwert). Ein einzelner Fehler, der zum Verlust der Erkennungsfunktion oder einer Erhöhung der Antwortzeit über den Maximalwert führt oder eine oder mehrere der OSSDs daran hindert, in den Zustand AUS überzugehen, muss infolge der nächsten regelmäßigen Prüfung zu einem gesperrten Zustand führen. Sofern die regelmäßige Prüfung von einem externen sicherheitsgerichteten Steuerungssystem (z. B. einer Maschine) initiiert werden soll, muss die BWS mit geeigneten Eingängen (z. B. Klemmen) ausgestattet werden. Die Dauer der regelmäßigen Prüfung darf die Sicherheitsfunktion nicht beeinträchtigen. Hinweis: Wenn die BWS vom Typ 2 als Auslösevorrichtung verwendet werden soll (z. B. als Umgebungswächter) und die Dauer der regelmäßigen Prüfung 150 ms überschreitet, kann eine Person die Erkennungszone passieren, ohne erkannt zu werden. In diesem Fall sollte eine Wiederanlaufsperre integriert werden. Wenn die regelmäßige Prüfung automatisch initiiert wird, muss die korrekte Funktion der regelmäßigen Prüfung überwacht werden und ein einzelner Fehler in den Teilen zur Implementierung der Überwachungsfunktion muss erkannt werden. Im Falle eines Fehlers muss/müssen der/die OSSD(s) einen Befehl zum Übergang in den Zustand AUS erhalten. Wenn eine oder mehrere OSSDs nicht in den Zustand AUS übergehen, muss ein gesperrter Zustand initiiert werden.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Dieser Funktionsbaustein erkennt zum Beispiel, ob eine Erkennung durch die Abtasteinrichtung nicht länger möglich ist, eine Überschreitung der festgelegten Antwortzeit und ein statisches EIN-Signal im Einzelkanal-Sensorsystem. Er kann für externe überprüfbare Sicherheitssensoren (BWS: berührungslos wirkende Schutzeinrichtung, z. B. Lichtschranke) verwendet werden.

Tab. 48: FB-Name: SF_TestableSafetySensor

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_OSSD_In	BOOL	FALSE	Variable. Status des Sensorausgangs, z. B. Lichtvorhang. FALSE: Der Sicherheitssensor ist im Testzustand, oder es liegt eine Anforderung einer sicherheitsgerichteten Antwort vor. TRUE: Sensor ist im Zustand für Normalbetrieb.
StartTest	BOOL	FALSE	Variable. Eingang zum Starten des Sensortests. Setzt S_TestOut und startet die interne Zeitüberwachungsfunktion im Funktionsbaustein. FALSE: Kein Test angefordert. TRUE: Test angefordert.
NoExternalTest	BOOL	FALSE	Konstante. Gibt an, ob der externe manuelle Sensortest unterstützt wird. FALSE: Der externe manuelle Sensortest wird unterstützt. Ein automatischer Test nach einem fehlerhaften automatischen Sensortest ist erst nach einer kompletten manuellen Sensor-Schaltsequenz wieder möglich. TRUE: Der externe manuelle Sensortest wird nicht unterstützt. Ein automatischer Test ist nach einem fehlerhaften automatischen Sensortest ohne komplette manuelle Sensor-Schaltsequenz wieder möglich.
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
TestTime	TIME	T#10ms	Konstante. Bereich: 0 … 150 ms. Testzeit des Sicherheitssensors.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_OSSD_Out	BOOL	FALSE	Sicherheitsausgang, der den Zustand der berührungslos wirkenden Schutzeinrichtung angibt. FALSE: Am Sensor liegt eine Anforderung einer sicherheitsgerichteten Aktion oder ein Testfehler vor. TRUE: Am Sensor liegt keine Anforderung einer sicherheitsgerichteten Aktion und kein Testfehler vor.
S_TestOut	BOOL	TRUE	Gekoppelt mit dem Testeingang des Sensors. FALSE: Test-Anforderung. TRUE: Kein Test angefordert.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
TestPossible	BOOL	FALSE	Rückmeldung an den Prozess. FALSE: Ein automatischer Sensortest ist nicht möglich. TRUE: Ein automatischer Sensortest ist möglich.
TestExecuted	BOOL	FALSE	Eine positive Signalflanke zeigt die erfolgreiche Ausführung des automatischen Sensortests an. FALSE: – Ein automatischer Sensortest wurde noch nicht durchgeführt. – Ein automatischer Sensortest ist aktiv. – Ein automatischer Sensortest war fehlerhaft. TRUE: Ein Sensortest wurde erfolgreich durchgeführt.
Error	BOOL	FALSE	↳ *Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	↳ *Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typisches Zeitdiagramm

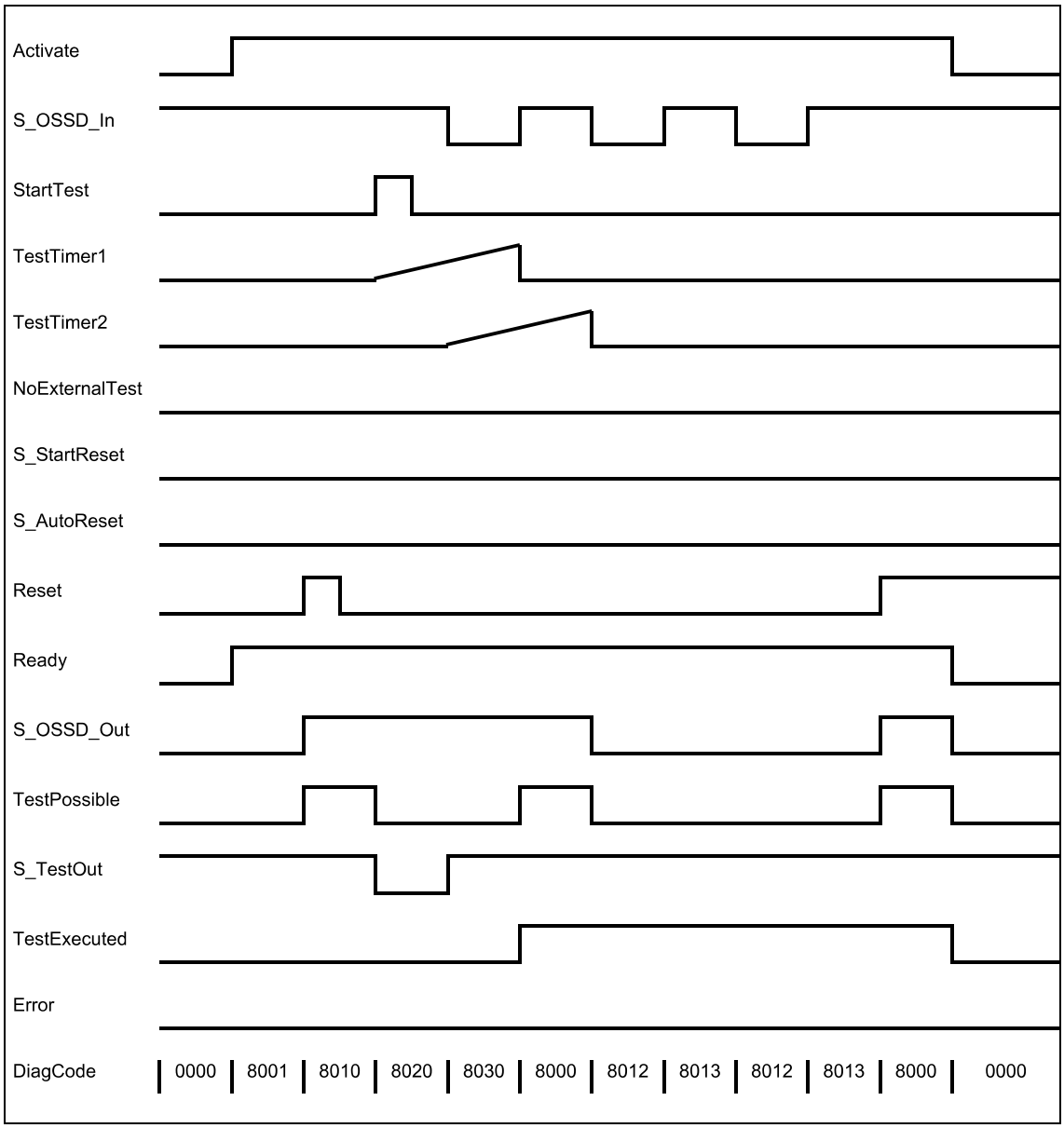


Abb. 108: Zeitdiagramm für SF_TestableSafetySensor

Die folgenden Situationen führen zu einem Übergang in den Fehlerzustand:

- Testzeit überschritten ohne verzögertes Sensor-Feedback.
- Test ohne Sensorsignal-Feedback.
- Ungültiges statisches Reset-Signal im Prozess.
- Plausibilitätsprüfung der eingestellten Überwachungszeit.

Bei einem Fehler wird der Ausgang S_OSSD_Out auf FALSE gesetzt und bleibt in diesem sicheren Zustand.

Sobald der Fehler behoben wurde und der Sensor aktiviert ist (S_OSSD_In = TRUE), setzt ein Reset den Fehlerzustand zurück und den Ausgang S_OSSD_Out auf TRUE.

Bei S_AutoReset = FALSE ist eine steigende Flanke an Reset erforderlich.

Nachdem S_OSSD_In auf TRUE gesetzt wurde, kann die optionale Anlaufsperre durch eine steigende Flanke am RESET-Eingang zurückgesetzt werden.

Nach Aktivierung des Bausteins kann die optionale Anlaufsperre durch eine steigende Flanke am RESET-Eingang zurückgesetzt werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 49: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C000	Parameterfehler	Ungültiger Wert des TestTime-Parameters. Werte zwischen 0 ms und 150 ms sind möglich. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C001	Fehler-Reset 1	Nach Aktivierung des Funktionsbausteins wurde eine statische Reset-Bedingung erkannt. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C002	Fehler-Reset 2	Statische Reset-Bedingung im Zustand 8003. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C003	Fehler-Reset 3	Statische Reset-Bedingung im Zustand C010. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C004	Fehler-Reset 4	Statische Reset-Bedingung im Zustand C020. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C005	Fehler-Reset 5	Statische Reset-Bedingung im Zustand 8006. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C006	Fehler-Reset 6	Statische Reset-Bedingung im Zustand C000. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C007	Fehler-Reset 7	Statische Reset-Bedingung im Zustand 8013. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = TRUE Error = TRUE
C010	Testfehler 1	Testzeit im Zustand 8020 abgelaufen. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE
C020	Testfehler 2	Testzeit im Zustand 8030 abgelaufen. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = TRUE

Tab. 50: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8001	Init	Der Funktionsbaustein hat eine Aktivierung erkannt. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8002	BWS unter-brochen 1	Der Funktionsbaustein hat eine Sicherheitsanforderung erkannt. Das Schaltelement wurde noch nicht automatisch getestet. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8003	Warten auf Reset 1	Warten auf steigende Flanke von Reset nach Zustand 8002. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8004	Externer Funktions-test	Der automatische Sensortest war fehlerhaft. Ein externer manueller Sensortest ist erforderlich. Die Unterstützung für den erforderlichen externen manuellen Sensortest wurde im Funktionsbaustein aktiviert (NoExternalTest = FALSE). Eine negative Signalflanke ist am Sensor erforderlich. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8005	BWS unter-brochen – externer Test	Der automatische Sensortest war fehlerhaft. Ein externer manueller Sensortest ist erforderlich. Die Unterstützung für den erforderlichen externen manuellen Sensortest wurde im Funktionsbaustein aktiviert (NoExternalTest = FALSE). Ein TRUE-Signal ist am Sensor erforderlich. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8006	Ende externer Test	Der automatische Sensortest war fehlerhaft. Ein externer manueller Sensortest ist erforderlich. Die Unterstützung für den erforderlichen externen manuellen Sensortest wurde im Funktionsbaustein aktiviert (NoExternalTest = FALSE). Der externe manuelle Sensortest ist beendet. Der Funktionsbaustein hat einen vollständigen Sensor-Schalt-zyklus festgestellt (extern überwacht). Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE

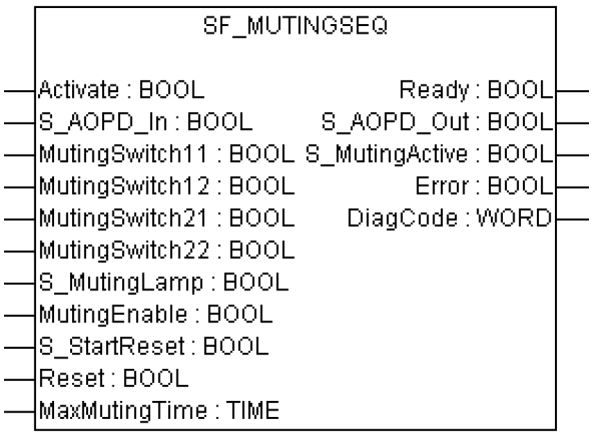
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8010	BWS frei – kein Test	Der Funktionsbaustein hat keine Sicherheitsanforderung erkannt. Der Sensor wurde nicht automatisch getestet. Ready = TRUE S_OSSD_Out = TRUE S_TestOut = TRUE TestPossible = TRUE TestExecuted = FALSE Error = FALSE
8020	Testanfrage	Der automatische Sensortest ist aktiv. Der Test-Timer wird zum ersten Mal gestartet. Das Gebersignal des Sensors wird vom Funktionsbaustein deaktiviert. Das Signal des Empfängers muss dem Signal des Gebers folgen. Ready = TRUE S_OSSD_Out = TRUE S_TestOut = FALSE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8030	Test aktiv	Der automatische Sensortest ist aktiv. Der Test-Timer wird zum zweiten Mal gestartet. Das Gebersignal des Sensors wird vom Funktionsbaustein aktiviert. Das Signal des Empfängers muss dem Signal des Gebers folgen. Ready = TRUE S_OSSD_Out = TRUE S_TestOut = TRUE TestPossible = FALSE TestExecuted = FALSE Error = FALSE
8000	BWS frei – Test OK	Der Funktionsbaustein hat keine Sicherheitsanforderung erkannt. Der Sensor wurde automatisch getestet. Ready = TRUE S_OSSD_Out = TRUE S_TestOut = TRUE TestPossible = TRUE TestExecuted = TRUE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8012	BWS unter-brochen 2	Der Funktionsbaustein hat eine Sicherheitsanforderung erkannt. Das Schaltelement wurde automatisch getestet. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = TRUE Error = FALSE
8013	Warten auf Reset 2	Warten auf steigende Flanke von Reset nach Zustand 8012. Ready = TRUE S_OSSD_Out = FALSE S_TestOut = TRUE TestPossible = FALSE TestExecuted = TRUE Error = FALSE

4.6.4.12 SF_MutingSeq

Normen	Anforderungen
IEC 61496-1:2004	A.7 Muting A.7.1.2 Es müssen mindestens zwei unabhängige und fest verdrahtete Muting-Signalquellen zur Initiierung der Funktion zur Verfügung stehen. Muting darf nicht möglich sein, wenn sich die OSSDs bereits im Zustand AUS befinden. A.7.1.3 Die Muting-Funktion darf nur von der korrekten Sequenz und/oder vom korrekten Timing der Muting-Signale initiiert werden. Sollten widersprüchliche Muting-Signale auftreten, darf die BWS keinen Muting-Zustand zulassen. A.7.1.4 Es müssen mindestens zwei unabhängige und fest verdrahtete Muting-Signalquellen zum Stoppen der Funktion zur Verfügung stehen. Die Muting-Funktion muss stoppen, wenn das erste dieser Muting-Signale seinen Zustand ändert. Die Deaktivierung der Muting-Funktion darf nicht nur auf der Freigabe der BWS beruhen. A.7.1.5 Die Muting-Signale sollten beim Muting durchgehend vorhanden sein. Wenn die Signale nicht durchgehend vorhanden sind, muss eine fehlerhafte Reihenfolge und/oder der Ablauf einer voreingestellten Zeitbegrenzung entweder einen gesperrten Zustand oder eine Wiederanlaufsperre auslösen. A.7.4 Anzeige: Ein Mute-Statussignal oder eine Statusanzeige muss zur Verfügung gestellt werden (in einigen Anwendungen ist ein Anzeigesignal für Muting erforderlich).
IEC 62046/ Ed. 1:2005	5.5.1: … eine Anzeige für den Aktivitätsstatus der Muting-Funktion kann erforderlich sein. Die Muting-Funktion muss automatisch initiiert und terminiert werden … Falsche Signale, Sequenzen oder Timing der Muting-Sensoren oder Signale dürfen keinen Muting-Zustand zulassen. Eine Initiierung der Muting-Funktion darf in den folgenden Fällen nicht möglich sein: ● Die Schutzausrüstungs-OSSDs befinden sich im Zustand AUS; ● Die Schutzausrüstung befindet sich im Sperrzustand. ● Initiierung der Muting-Funktion durch zwei oder mehrere unabhängige Muting-Sensoren, sodass ein einzelner Fehler keinen Mute-Zustand auslösen kann; ● Terminierung der Muting-Funktion durch zwei oder mehrere unabhängige Muting-Sensoren, sodass die Deaktivierung eines Sensors die Muting-Funktion beendet; ● Verwendung von Timing- und Sequenzsteuerung für Muting-Sensoren, um einen korrekten Muting-Betrieb sicherzustellen; 5.5.3: Die folgenden Maßnahmen sind zu berücksichtigen: … ● Die Begrenzung des Muting-Zustandes auf einen bestimmten Zeitraum, der für den Transport des Materials durch die Erkennungszone ausreichend ist. Wenn diese Zeit überschritten wird, sollte die Muting-Funktion beendet und alle gefährlichen Bewegungen sollten gestoppt werden. Anhang F.3 Vier Lichtschranken – Ablaufsteuerung: (siehe auch Abb. F.3.1 und Tabelle F.1 in der Norm) Die Initiierung der Muting-Funktion hängt von der Überwachung der korrekten Aktivierungsreihenfolge der Muting-Sensoren ab. Wenn beispielsweise im Muting-Zustand S2 (in diesem Dokument MS_12) deaktiviert wird, bevor S3 (in diesem Dokument MS_21) aktiviert wird, führt dies zum Abbruch des Muting. Anhang F.5: Methoden zur Vermeidung von Manipulationen der Muting-Funktion: … Verwenden Sie einen vom Steuerungssystem der Maschine erzeugten Aktivierungsbefehl, der die Muting-Funktion nur dann auslöst, wenn dies im Maschinenzyklus erforderlich ist.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Muting ist die gewollte Unterdrückung der Sicherheitsfunktion (z. B. Lichtschranken). In diesem Funktionsbaustein wird sequenzielles Muting mit vier Muting-Sensoren spezifiziert.

Muting ist die gewollte Unterdrückung der Sicherheitsfunktion. Dies ist z. B. erforderlich, wenn Material in den Gefahrenbereich transportiert wird, ohne dass die Maschine gestoppt werden soll. Muting wird durch Muting-Sensoren ausgelöst. Die Verwendung von zwei oder vier Muting-Sensoren und die korrekte Integration in die Produktionssequenz müssen sicherstellen, dass niemand den Gefahrenbereich betreten kann, während der Lichtvorhang deaktiviert ist. Als Muting-Sensoren können Näherungsschalter, Lichtschranken, Grenzwertschalter usw. verwendet werden; sie müssen nicht zwingend „failsafe“ sein. Aktives Muting muss durch Kontrollleuchten angezeigt werden.

Es gibt sequenzielles und paralleles Muting. In diesem Funktionsbaustein wird sequenzielles Muting mit vier Muting-Sensoren verwendet; eine Erklärung für die Vorwärtsbewegung beim Transport ist unten angegeben. Der Funktionsbaustein kann in beide Richtungen verwendet werden: vorwärts und rückwärts. Muting sollte durch die Prozesssteuerung über das Signal MutingEnable aktiviert werden, um eine Manipulation zu vermeiden. Wenn das Signal MutingEnable nicht vorhanden ist, muss dieser Eingang auf TRUE gesetzt werden.

Die Eingangsparameter des Funktionsbausteins umfassen die Signale der vier Muting-Sensoren (MutingSwitch11 … MutingSwitch22) sowie das OSSD-Signal der aktiven optoelektronischen Schutzeinrichtung (S_AOPD_In).

Der Eingang S_StartReset darf nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 51: FB-Name: SF_MutingSeq

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AOPD_In	BOOL	FALSE	Variable. OSSD-Signal der AOPD. FALSE: Schutzfeld unterbrochen. TRUE: Schutzfeld nicht unterbrochen.
MutingSwitch11	BOOL	FALSE	Variable. Zustand des Muting-Sensors 11. FALSE: Muting-Sensor 11 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 11.

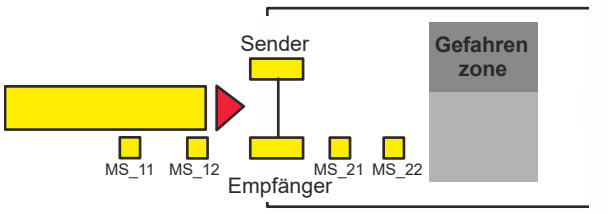
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
MutingSwitch12	BOOL	FALSE	Variable. Zustand des Muting-Sensors 12. FALSE: Muting-Sensor 12 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 12.
MutingSwitch21	BOOL	FALSE	Variable. Zustand des Muting-Sensors 21. FALSE: Muting-Sensor 21 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 21.
MutingSwitch22	BOOL	FALSE	Variable. Zustand des Muting-Sensors 22. FALSE: Muting-Sensor 22 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 22.
S_MutingLamp	BOOL	FALSE	Variable oder Konstante. Zeigt den Betrieb der Muting-Lampe. FALSE: Ausfall der Muting-Lampe. TRUE: Kein Ausfall der Muting-Lampe.
MutingEnable	BOOL	FALSE	Variable oder Konstante. Befehl des Steuerungssystems, der die Muting-Funktion auslöst, sobald dies im Maschinenzyklus erforderlich ist. Nach dem Start der Muting-Funktion kann dieses Signal ausgeschaltet werden. FALSE: Muting nicht aktiviert TRUE: Muting-Funktion aktiviert
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
MaxMutingTime	TIME	T#0s	Konstante 0 .. 10 min; Maximale Zeit für das Beenden der Muting-Sequenz; der Timer startet, wenn der erste Muting-Sensor betätigt wird.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_AOPD_Out	BOOL	FALSE	Sicherheitsausgang, der den Status der Schutzeinrichtung im Muting-Zustand anzeigt. FALSE: Schutzfeld der aktiven optoelektronischen Schutzeinrichtung unterbrochen und Muting nicht aktiv. TRUE: Schutzfeld der aktiven optoelektronischen Schutzeinrichtung nicht unterbrochen oder Muting aktiv.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_MutingActive	BOOL	FALSE	Zeigt den Muting-Zustand. FALSE: Muting nicht aktiv. TRUE: Muting aktiv.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweis: Ein Kurzschluss der Muting-Sensor-Signale oder ein Fehler der funktionalen Anwendung bei der Signalübertragung wird von diesem Funktionsbaustein nicht erkannt, aber als fehlerhafte Muting-Sequenz interpretiert. Dies sollte jedoch nicht zu einem ungewollten Muting führen. Anwender sollten dies in die Risikoanalyse aufnehmen.

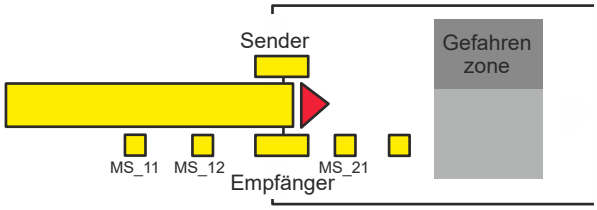
Beispiel für SF_MutingSeq in Vorwärtsbewegung mit vier Sensoren

1



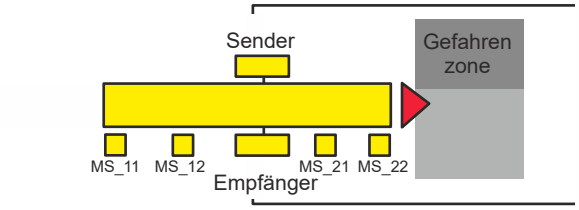
Wenn das Produkt den Muting-Sensor MutingSwitch12 (MS_12) nach MutingSwitch11 (MS_11) aktiviert, ist der Muting-Modus aktiviert.

2



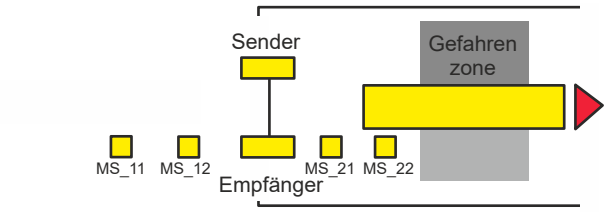
Muting bleibt so lange aktiv, wie MutingSwitch11 (MS_11) und MutingSwitch12 (MS_12) vom Produkt aktiviert werden. Das Produkt kann den Lichtvorhang passieren, ohne dass die Maschine stoppt.

3



Bevor die Muting-Sensoren MutingSwitch11 (MS_11) und MutingSwitch12 (MS_12) deaktiviert werden, müssen die Muting-Sensoren MutingSwitch21 (MS_21) und MutingSwitch22 (MS_22) aktiviert werden. Dies stellt sicher, dass der Muting-Modus aktiv bleibt.

4



Der Muting-Modus ist beendet, wenn nur der Muting-Sensor MutingSwitch22 (MS_22) vom Produkt aktiviert wird.

Bedingungen für Muting

Vorwärtsbewegung

Muting-Bedingung 1 (bis Zustand 8011) (MS_11 ist das erste bestätigte Schaltelement am Eingang). Timer MaxMutingTime wird gestartet:

MutingEnable AND (R_TRIG at MS_11 AND NOT MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 2 (von Zustand 8011 bis Zustand 8012) (MS_12 ist das zweite bestätigte Schaltelement am Eingang):

MutingEnable AND (MS_11 AND R_TRIG at MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 3 (von Zustand 8012 bis Zustand 8000) (MS_21 ist das erste freigegebene Schaltelement am Ausgang). Timer MaxMutingTime wird gestoppt:

NOT MS_11 AND NOT MS_12 AND F_TRIG at MS_21 AND MS_22

Rückwärtsbewegung

Muting-Bedingung 11 (bis Zustand 8122) (MS_22 ist das erste betätigte Schaltelement am Eingang). Timer MaxMutingTime wird gestartet:

MutingEnable AND (NOT MS_11 AND NOT MS_12 AND NOT MS_21 AND R_TRIG at MS_22)

Muting-Bedingung 12 (von Zustand 8122 bis Zustand 8112) (MS_21 ist das zweite bestätigte Schaltelement am Eingang):

MutingEnable AND (NOT MS_11 AND NOT MS_12 AND R_TRIG at MS_21 AND MS_22)

Muting-Bedingung 13 (MS_12 ist das erste freigegebene Schaltelement am Ausgang). Timer MaxMutingTime wird gestoppt:

MS_11 AND F_TRIG at MS_12 AND NOT MS_21 AND NOT MS_22

Spezifizierung falscher Muting-Sequenzen:

In Zustand 8000 - (NOT MutingEnable AND R_TRIG at MS_11) OR (NOT MutingEnable AND R_TRIG at MS_22) OR (MS_12 OR MS_21) OR (MS_11 AND MS_22)

In Zustand 8011 - NOT MutingEnable OR NOT MS_11 OR MS_21 OR MS_22

In Zustand 8012 - R_TRIG at MS_11 OR R_TRIG at MS_12 OR F_TRIG at MS_22

In Zustand 8122 - NOT MutingEnable OR MS_11 OR MS_12 OR NOT MS_22

In Zustand 8112 - F_TRIG at MS_11 OR R_TRIG at MS_21 OR R_TRIG at MS_22

Typisches Zeitdiagramm

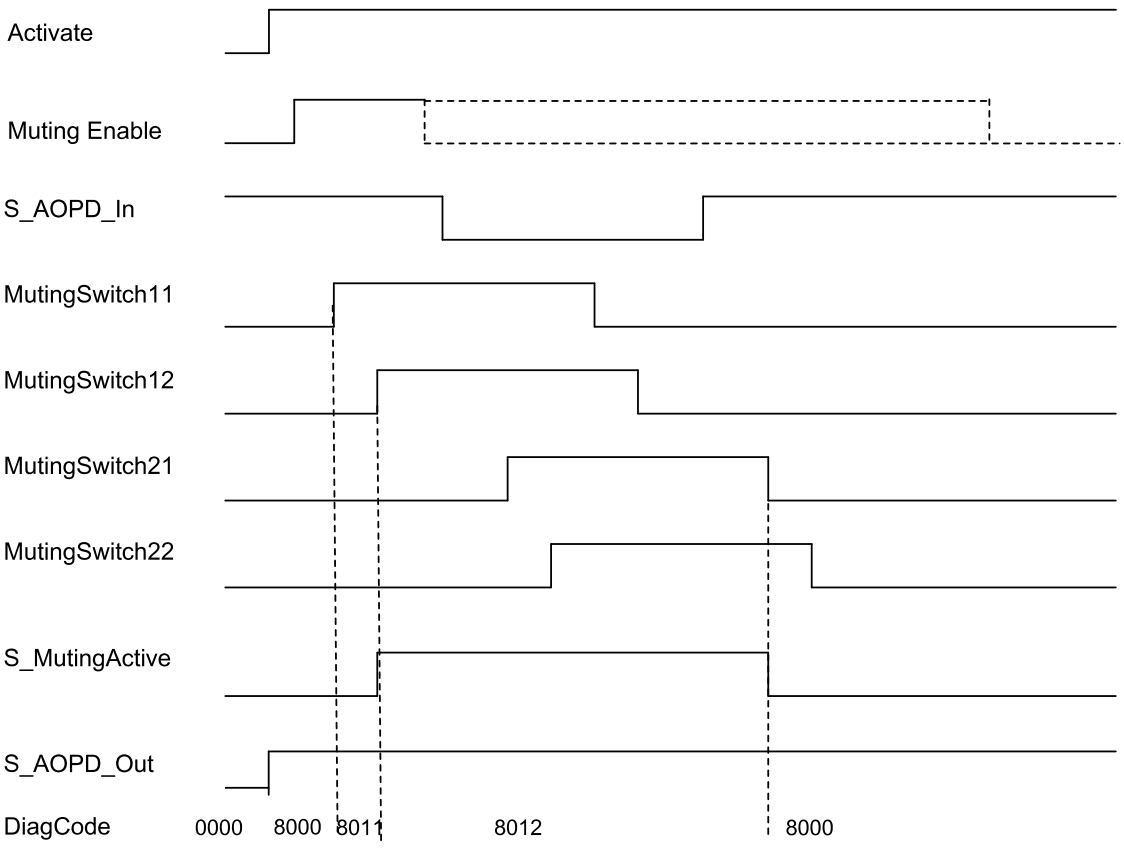


Abb. 109: Zeitdiagramm für SF_MutingSeq mit S_StartReset = TRUE

Der Funktionsbaustein erkennt die folgenden Fehlerbedingungen:

- Die Muting-Sensoren MutingSwitch11, MutingSwitch12, MutingSwitch21 und MutingSwitch22 werden in der falschen Reihenfolge aktiviert.
- Die Muting-Sequenz startet, ohne von MutingEnable aktiviert worden zu sein.

- Eine fehlerhafte Muting-Lampe wird von S_MutingLamp = FALSE angezeigt.
- Eine statische Reset-Bedingung.
- Der Wert für MaxMutingTime ist kleiner als T#0s oder größer als T#10min.
- Die Muting-Funktion (S_MutingActive = TRUE) überschreitet die maximale Muting-Zeit MaxMutingTime.

Verhalten im Fehlerfall

Bei einem Fehler werden die Ausgänge S_AOPD_Out und S_MutingActive auf FALSE gesetzt. Der Ausgang DiagCode zeigt den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Ein Neustart ist erst möglich, wenn die Fehler behoben wurden und der sichere Zustand vom Bediener mit Reset quittiert wurde.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 52: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Nach Aktivierung des Funktionsbausteins wurde eine statische Reset-Bedingung erkannt. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C002	Fehler-Reset 2	Statische Reset-Bedingung im Zustand 8003. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C003	Fehler bei Muting-Lampe	Fehlerhafte Muting-Lampe. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
CYx4	Fehler in Muting-Sequenz	Fehler in der Muting-Sequenz in den Zuständen 8000, 8011, 8012, 8112 oder 8122. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE Y = Zustand in der Sequenz (2 Zustände für vorwärts und 2 Zustände für rückwärts). C0x4 = Fehler in Zustand 8000 C1x4 = Fehler in Zustand Vorwärts 8011 C2x4 = Fehler in Zustand Vorwärts 8012 C3x4 = Fehler in Zustand Rückwärts 8122 C4x4 = Fehler in Zustand Rückwärts 8112 CFx4 = MutingEnable fehlt x = Zustand der Sensoren, als der Fehler auftrat (4 Bits: LSB = MS_11; MS_12; MS_21; MSB = MS_22).
C005	Parameter-fehler	MaxMutingTime außerhalb des gültigen Bereichs. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C006	Fehler in MaxMuting-Timer	Zeitfehler: Die aktive Muting-Zeit (bei S_MutingActive = TRUE) übersteigt MaxMutingTime. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

Tab. 53: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8000	AOPD frei	Muting ist nicht aktiv, keine Sicherheitsanforderung der aktiven optoelektronischen Schutzeinrichtung. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE

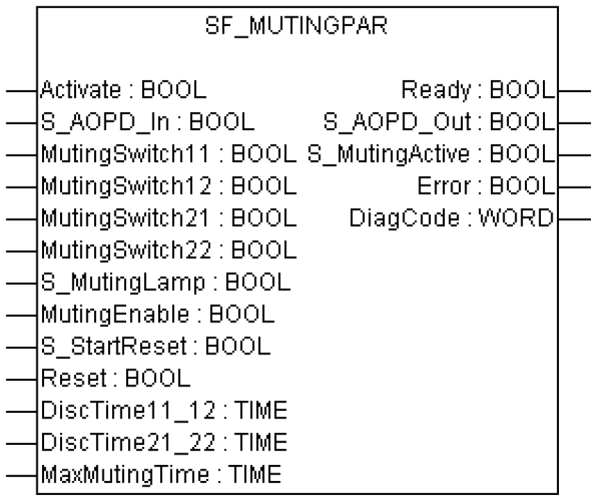
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8001	Init	Der Funktionsbaustein wurde aktiviert. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8002	Sicherheits-anforderung – AOPD	Sicherheitsanforderung von aktiver optoelektronischer Schutzein-richtung erkannt, Muting ist nicht aktiv. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8003	Warten auf Reset	Sicherheitsanforderung oder Fehler wurde erkannt und behoben. Bedienerquittierung durch Reset erforderlich. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8005	Sicher	Sicherheitsfunktion aktiviert. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8011	Muting vor-wärts – Start	Muting vorwärts; Sequenz ist in Startphase, keine Sicherheitsan-forderung. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8012	Muting vor-wärts aktiv	Muting vorwärts, Sequenz ist aktiv. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8112	Muting rück-wärts aktiv	Muting rückwärts, Sequenz ist aktiv. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE
8122	Muting rück-wärts – Start	Muting rückwärts; Sequenz ist in Startphase, keine Sicherheitsan-forderung. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE

4.6.4.13 SF_MutingPar

Normen	Anforderungen
IEC 61496-1:2004	A.7 Muting A.7.1.2 Es müssen mindestens zwei unabhängige und fest verdrahtete Muting-Signalquellen zur Initiierung der Funktion zur Verfügung stehen. Muting darf nicht möglich sein, wenn sich die OSSDs bereits im Zustand AUS befinden. A.7.1.3 Die Muting-Funktion darf nur von der korrekten Sequenz und/oder vom korrekten Timing der Muting-Signale initiiert werden. Sollten widersprüchliche Muting-Signale auftreten, darf die BWS keinen Muting-Zustand zulassen. A.7.1.4 Es müssen mindestens zwei unabhängige und fest verdrahtete Muting-Signalquellen zum Stoppen der Funktion zur Verfügung stehen. Die Muting-Funktion muss stoppen, wenn das erste dieser Muting-Signale seinen Zustand ändert. Die Deaktivierung der Muting-Funktion darf nicht nur auf der Freigabe der BWS beruhen. A.7.1.5 Die Muting-Signale sollten beim Muting durchgehend vorhanden sein. Wenn die Signale nicht durchgehend vorhanden sind, muss eine fehlerhafte Reihenfolge und/oder der Ablauf einer voreingestellten Zeitbegrenzung entweder einen gesperrten Zustand oder eine Wiederanlaufsperre auslösen. A.7.4 Anzeige: Ein Mute-Statussignal oder eine Statusanzeige muss bereitgestellt werden (in einigen Anwendungen ist ein Anzeigesignal für Muting erforderlich).
IEC 62046/ Ed. 1:2005	5.5.1: … eine Anzeige für den Aktivitätsstatus der Muting-Funktion kann erforderlich sein. Die Muting-Funktion muss automatisch initiiert und terminiert werden … Falsche Signale, Sequenzen oder Timing der Muting-Sensoren oder Signale dürfen keinen Muting-Zustand zulassen. Eine Initiierung der Muting-Funktion darf in den folgenden Fällen nicht möglich sein: ● Die Schutzausrüstungs-OSSDs befinden sich im Zustand AUS; ● Die Schutzausrüstung befindet sich im Sperrzustand; ● Initiierung der Muting-Funktion durch zwei oder mehrere unabhängige Muting-Sensoren, sodass ein einzelner Fehler keinen Mute-Zustand auslösen kann; ● Terminierung der Muting-Funktion durch zwei oder mehrere unabhängige Muting-Sensoren, sodass die Deaktivierung eines Sensors die Muting-Funktion beendet; ● Verwendung von Timing- und Sequenzsteuerung für Muting-Sensoren, um einen korrekten Muting-Betrieb sicherzustellen; 5.5.3: Die folgenden Maßnahmen sind zu berücksichtigen: … ● Die Begrenzung des Muting-Zustandes auf einen bestimmten Zeitraum, der für den Transport des Materials durch die Erkennungszone ausreichend ist. Wenn diese Zeit überschritten wird, sollten die Muting-Funktion beendet und alle gefährlichen Bewegungen gestoppt werden. Anhang F.2 Vier Lichtschranken – Ablaufsteuerung: (siehe auch Abb. F.2.4 in der Norm): Die Überwachung der Muting-Funktion basiert auf der Zeitbegrenzung zwischen dem Auslösen der Sensoren S1 (in diesem Dokument MS_11) und S2 (in diesem Dokument MS_12) sowie zwischen dem Auslösen der Sensoren S3 (in diesem Dokument MS_21) und S4 (in diesem Dokument MS_22). Eine maximale Dauer von 4 s wird empfohlen. Die Muting-Funktion wird von den beiden Sensoren S1 und S2 initiiert und von den beiden Sensoren S3 und S4 aufrechterhalten. Das bedeutet, dass für einen bestimmten Zeitraum alle vier Sensoren aktiviert sind. Die Muting-Funktion wird beendet, wenn S3 oder S4 deaktiviert wird. Anhang F.5: Methoden zur Vermeidung von Manipulationen der Muting-Funktion: … Verwenden Sie einen vom Steuerungssystem der Maschine erzeugten Aktivierungsbefehl, der die Muting-Funktion nur dann auslöst, wenn dies im Maschinenzyklus erforderlich ist.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Muting ist die gewollte Unterdrückung der Sicherheitsfunktion. In diesem Funktionsbaustein wird paralleles Muting mit vier Muting-Sensoren spezifiziert.

Dies ist z. B. erforderlich, wenn Material in den Gefahrenbereich transportiert wird, ohne dass die Maschine gestoppt werden soll. Muting wird durch Muting-Sensoren ausgelöst. Die Verwendung von zwei oder vier Muting-Sensoren und die korrekte Integration in die Produktionssequenz müssen sicherstellen, dass niemand den Gefahrenbereich betreten kann, während der Lichtvorhang deaktiviert ist. Als Muting-Sensoren können Näherungsschalter, Lichtschranken, Grenzwertschalter usw. verwendet werden; sie müssen nicht zwingend „failsafe“ sein. Aktives Muting muss durch Kontrollleuchten angezeigt werden.

Es gibt sequenzielles und paralleles Muting. In diesem Funktionsbaustein wird paralleles Muting mit vier Muting-Sensoren verwendet; eine Erklärung wird unten angegeben. Der Funktionsbaustein kann in beide Richtungen verwendet werden: vorwärts und rückwärts. Muting sollte durch die Prozesssteuerung über das Signal MutingEnable aktiviert werden, um eine Manipulation zu vermeiden.

Die Eingangsparameter des Funktionsbausteins umfassen die Signale der vier Muting-Sensoren (MutingSwitch11 … MutingSwitch22), das OSSD-Signal der aktiven optoelektronischen Schutzeinrichtung (S_AOPD_In) sowie drei parametrierbare Zeiten (DiscTime11_12, DiscTime21_22 und MaxMutingTime).

Der Eingang S_StartReset darf nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 54: FB-Name: SF_MutingPar

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AOPD_In	BOOL	FALSE	Variable. OSSD-Signal der AOPD. FALSE: Schutzfeld unterbrochen. TRUE: Schutzfeld nicht unterbrochen.
MutingSwitch11	BOOL	FALSE	Variable. Zustand des Muting-Sensors 11. FALSE: Muting-Sensor 11 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 11.

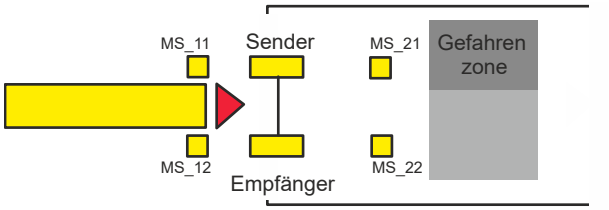
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
MutingSwitch12	BOOL	FALSE	Variable. Zustand des Muting-Sensors 12. FALSE: Muting-Sensor 12 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 12.
MutingSwitch21	BOOL	FALSE	Variable. Zustand des Muting-Sensors 21. FALSE: Muting-Sensor 21 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 21.
MutingSwitch22	BOOL	FALSE	Variable. Zustand des Muting-Sensors 22. FALSE: Muting-Sensor 22 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 22.
S_MutingLamp	BOOL	FALSE	Variable oder Konstante. Zeigt den Betrieb der Muting-Lampe. FALSE: Ausfall der Muting-Lampe. TRUE: Kein Ausfall der Muting-Lampe.
MutingEnable	BOOL	FALSE	Variable oder Konstante. Befehl des Steuerungssystems, der die Muting-Funktion auslöst, sobald dies im Maschinenzyklus erforderlich ist. Nach dem Start der Muting-Funktion kann dieses Signal ausgeschaltet werden. FALSE: Muting nicht aktiviert TRUE: Muting-Funktion aktiviert
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
DiscTime11_12	TIME	T#0s	Konstante 0..4 s; Maximale Diskrepanzzeit für MutingSwitch11 und MutingSwitch12.
DiscTime21_22	TIME	T#0s	Konstante 0..4 s; Maximale Diskrepanzzeit für MutingSwitch21 und MutingSwitch22.
MaxMutingTime	TIME	T#0s	Konstante 0..10 min; Maximale Zeit für das Beenden der Muting-Sequenz; der Timer startet, wenn der erste Muting-Sensor betätigt wird.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_AOPD_Out	BOOL	FALSE	Sicherheitsausgang, der den Status der Schutzeinrichtung im Muting-Zustand anzeigt. FALSE: Schutzfeld der aktiven optoelektronischen Schutzeinrichtung unterbrochen und Muting nicht aktiv. TRUE: Schutzfeld der aktiven optoelektronischen Schutzeinrichtung nicht unterbrochen oder Muting aktiv.
S_MutingActive	BOOL	FALSE	Zeigt den Muting-Zustand. FALSE: Muting nicht aktiv. TRUE: Muting aktiv.
Error	BOOL	FALSE	↳ *Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	↳ *Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweis: Ein Kurzschluss der Muting-Sensor-Signale oder ein Fehler der funktionalen Anwendung bei der Signalübertragung wird von diesem Funktionsbaustein nicht erkannt. Dies sollte jedoch nicht zu einem ungewollten Muting führen. Anwender sollten dies in die Risikoanalyse aufnehmen.

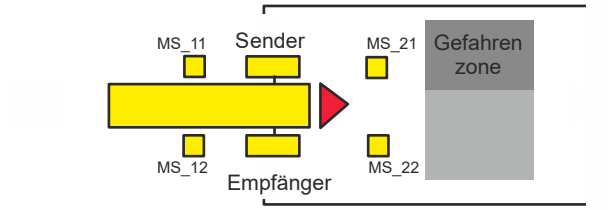
Beispiel für SF_MutingPar in Vorwärtsbewegung mit vier Sensoren

1



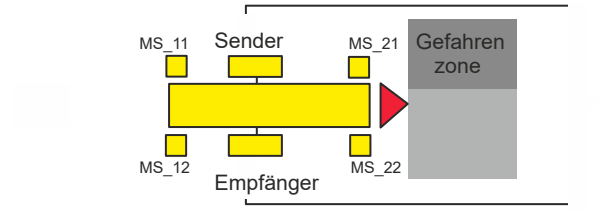
Wenn das Produkt die Muting-Sensoren MutingSwitch11 (MS_11) und MutingSwitch12 (MS_12) innerhalb der Zeit DiscTime11_12 aktiviert, wird der Muting-Modus aktiviert (S_MutingActive = TRUE).

2



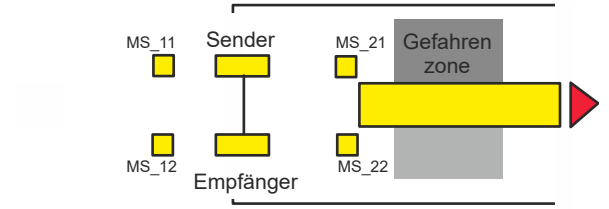
Muting bleibt so lange aktiv, wie MutingSwitch11 (MS_11) und MutingSwitch12 (MS_12) vom Produkt aktiviert werden. Das Produkt kann den Lichtvorhang passieren, ohne dass die Maschine stoppt.

3



Bevor die Muting-Sensoren MutingSwitch11 (MS_11) und MutingSwitch12 (MS_12) deaktiviert werden, müssen die Muting-Sensoren MutingSwitch21 (MS_21) und MutingSwitch22 (MS_22) aktiviert werden. Dies stellt sicher, dass der Muting-Modus aktiv bleibt. Die Zeitdiskrepanz zwischen dem Schalten von MutingSwitch21 und MutingSwitch22 wird von DiscTime21_22 überwacht.

4



Der Muting-Modus wird beendet, wenn entweder Muting-Sensor MutingSwitch21 (MS_21) oder MutingSwitch22 (MS_22) vom Produkt deaktiviert wird. Die maximale Zeit für einen aktiven Muting-Modus ist MaxMutingTime.

Bedingungen für Muting

Vorwärtsbewegung

Muting-Bedingung 1 (bis Zustand 8011) (MS_11 ist das erste bestätigte Schaltelement am Eingang). Timer MaxMutingTime und DiscTime11_12 wird gestartet:

MutingEnable AND (R_TRIG at MS_11 AND NOT MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 1 (bis Zustand 8311) (MS_12 ist das erste bestätigte Schaltelement am Eingang). Timer MaxMutingTime und DiscTime11_12 wird gestartet:

MutingEnable AND (NOT MS_11 AND R_TRIG at MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 2 (von Zustand 8011) (MS_12 ist das zweite bestätigte Schaltelement am Eingang): Timer DiscTime11_12 wird gestoppt:

MutingEnable AND (MS_11 AND R_TRIG at MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 2 (von Zustand 8311) (MS_11 ist das zweite bestätigte Schaltelement am Eingang): Timer DiscTime11_12 wird gestoppt:

MutingEnable AND (R_TRIG at MS_11 AND MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 3 (beide Schaltelemente am Eingang werden im selben Zyklus betätigt). Timer MaxMutingTime wird gestartet:

MutingEnable AND (R_TRIG at MS_11 AND R_TRIG at MS_12 AND NOT MS_21 AND NOT MS_22)

Muting-Bedingung 4 (alle Schaltelemente betätigt): MS_11 AND MS_12 AND MS_21 AND MS_22

Muting-Bedingung 24 (bis Zustand 8014) (MS_21 ist das erste betätigte Schaltelement am Ausgang). Timer DiscTime21_22 wird gestartet:

MS_11 AND MS_12 AND R_TRIG at MS_21 AND NOT MS_22

Muting-Bedingung 24 (bis Zustand 8314) (MS_22 ist das erste betätigte Schaltelement am Ausgang). Timer DiscTime21_22 wird gestartet:

MS_11 AND MS_12 AND NOT MS_21 AND R_TRIG at MS_22

Muting-Bedingung 25 (von Zustand 8014) (MS_22 ist das zweite betätigte Schaltelement am Ausgang). Timer DiscTime21_22 wird gestoppt:

MS_11 AND MS_12 AND MS_21 AND R_TRIG at MS_22

Muting-Bedingung 25 (von Zustand 8314) (MS_21 ist das zweite betätigte Schaltelement am Ausgang). Timer DiscTime21_22 wird gestoppt:

MS_11 AND MS_12 AND R_TRIG at MS_21 AND MS_22

Muting-Bedingung 5 (eines der Schaltelemente am Ausgang wird freigegeben). Timer MaxMutingTime wird gestoppt:

NOT MS_11 AND NOT MS_12 AND (F_TRIG at MS_21 OR F_TRIG at MS_22)

Rückwärtsbewegung

Muting-Bedingung 11 (bis Zustand 8122) (MS_21 ist das erste betätigte Schaltelement am Eingang). Timer MaxMutingTime und DiscTime21_22 wird gestartet:

MutingEnable AND (NOT MS_22 AND R_TRIG at MS_21 AND NOT MS_11 AND NOT MS_12)

Muting-Bedingung 11 (bis Zustand 8422) (MS_22 ist das erste betätigte Schaltelement am Eingang). Timer MaxMutingTime und DiscTime21_22 wird gestartet:

MutingEnable AND (R_TRIG at MS_22 AND NOT MS_21 AND NOT MS_11 AND NOT MS_12)

Muting-Bedingung 12 (von Zustand 8122) (MS_22 ist das zweite betätigte Schaltelement am Eingang). Timer DiscTime21_22 wird gestoppt:

MutingEnable AND (MS_21 AND R_TRIG at MS_22 AND NOT MS_11 AND NOT MS_12)

Muting-Bedingung 12 (von Zustand 8422) (MS_21 ist das zweite bestätigte Schaltelement am Eingang). Timer DiscTime21_22 wird gestoppt:

MutingEnable AND (R_TRIG at MS_21 AND MS_22 AND NOT MS_11 AND NOT MS_12)

Muting-Bedingung 13 (beide Schaltelemente am Eingang werden im selben Zyklus betätigt). Timer MaxMutingTime wird gestartet:

MutingEnable AND (R_TRIG at MS_21 AND R_TRIG at MS_22 AND NOT MS_11 AND NOT MS_12)

Muting-Bedingung 14 (alle Schaltelemente betätigt): MS_11 AND MS_12 AND MS_21 AND MS_22

Muting-Bedingung 44 (bis Zustand 8114) (MS_11 ist das erste betätigte Schaltelement am Ausgang). Timer DiscTime11_12 wird gestartet:

MS_21 AND MS_22 AND R_TRIG at MS_11 AND NOT MS_12

Muting-Bedingung 44 (bis Zustand 8414) (MS_12 ist das erste betätigte Schaltelement am Ausgang). Timer DiscTime11_12 wird gestartet:

MS_21 AND MS_22 AND NOT MS_11 AND R_TRIG at MS_12

Muting-Bedingung 45 (von Zustand 8114) (MS_12 ist das zweite betätigte Schaltelement am Ausgang). Timer DiscTime11_12 wird gestoppt:

MS_21 AND MS_22 AND MS_11 AND R_TRIG at MS_12

Muting-Bedingung 45 (von Zustand 8414) (MS_11 ist das zweite betätigte Schaltelement am Ausgang). Timer DiscTime11_12 wird gestoppt:

MS_21 AND MS_22 AND R_TRIG at MS_11 AND MS_12

Muting-Bedingung 15 (eines der Schaltelemente am Ausgang wird freigegeben). Timer MaxMutingTime wird gestoppt:

NOT MS_21 AND NOT MS_22 AND (F_TRIG at MS_11 OR F_TRIG at MS_12)

Falsche Muting-Sequenzen:

Zustand 8000 - (MutingEnable = FALSE, wenn Muting-Sequenz startet) OR
((MS_11 OR MS_12) AND (MS_21 OR MS_22)) OR
(R_TRIG at MS_11 AND MS_12 AND NOT R_TRIG at MS_12) OR
(R_TRIG at MS_12 AND MS_11 AND NOT R_TRIG at MS_11) OR
(R_TRIG at MS_21 AND MS_22 AND NOT R_TRIG at MS_22) OR
(R_TRIG at MS_22 AND MS_21 AND NOT R_TRIG at MS_21) OR
((MS_11 AND NOT R_TRIG at MS_11) AND (MS_12 AND NOT R_TRIG at MS_12)) OR
((MS_21 AND NOT R_TRIG at MS_21) AND (MS_22 AND NOT R_TRIG at MS_22))

Zustand 8011 - NOT MutingEnable OR NOT MS_11 OR MS_21 OR MS_22

Zustand 8311 - NOT MutingEnable OR NOT MS_12 OR MS_21 OR MS_22

Zustand 8012 - NOT MS_11 OR NOT MS_12

Zustand 8021 - R_TRIG at MS_11 OR R_TRIG at MS_12 OR R_TRIG at MS_21 OR R_TRIG at MS_22

Zustand 8014 - NOT MS_11 OR NOT MS_12 OR NOT MS_21

Zustand 8314 - NOT MS_11 OR NOT MS_12 OR NOT MS_22

Zustand 8122 - NOT MutingEnable OR MS_11 OR MS_12 OR NOT MS_21

Zustand 8422 - NOT MutingEnable OR MS_11 OR MS_12 OR NOT MS_22

Zustand 8121 - NOT MS_21 OR NOT MS_22

Zustand 8112 - R_TRIG at MS_11 OR R_TRIG at MS_12 OR R_TRIG at MS_21 OR R_TRIG at MS_22

Zustand 8114 - NOT MS_21 OR NOT MS_22 OR NOT MS_11

Zustand 8414 - NOT MS_21 OR NOT MS_22 OR NOT MS_12

Typisches Zeitdiagramm

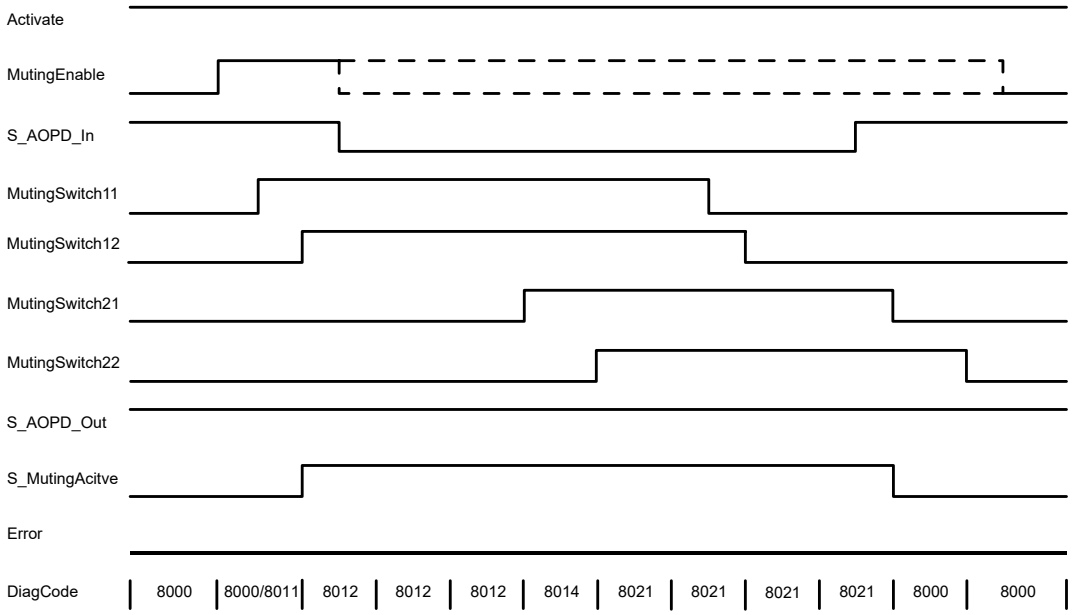


Abb. 110: Zeitdiagramm für SF_MutingPar

Der Funktionsbaustein erkennt die folgenden Fehlerbedingungen:

- Die Werte für DiscTime11_12 und DiscTime21_22 sind kleiner als T#0s oder größer als T#4s.
- Der Wert für MaxMutingTime ist kleiner als T#0s oder größer als T#10min.
- Die Diskrepanzzeit für die Sensorpaare MutingSwitch11/MutingSwitch12 oder MutingSwitch21/MutingSwitch22 wurde überschritten.
- Die Muting-Funktion (S_MutingActive = TRUE) überschreitet die maximale Muting-Zeit MaxMutingTime.
- Die Muting-Sensoren MutingSwitch11, MutingSwitch12, MutingSwitch21 und MutingSwitch22 werden in der falschen Reihenfolge aktiviert.
- Die Muting-Sequenz startet, ohne von MutingEnable aktiviert worden zu sein.
- Eine fehlerhafte Muting-Lampe wird von S_MutingLamp = FALSE angezeigt.
- Eine statische Reset-Bedingung wurde im Zustand 8001 und 8003 erkannt.

Verhalten im Fehlerfall

Bei einem Fehler werden die Ausgänge S_AOPD_Out und S_MutingActive auf FALSE gesetzt. Der Ausgang DiagCode zeigt den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Ein Neustart ist erst möglich, wenn die Fehler behoben wurden und der sichere Zustand vom Bediener mit Reset quittiert wurde.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 55: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Nach Aktivierung des Funktionsbausteins im Zustand 8001 wurde eine statische Reset-Bedingung erkannt. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C002	Fehler-Reset 2	Statische Reset-Bedingung im Zustand 8003. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C003	Fehler bei Muting-Lampe	Fehlerhafte Muting-Lampe. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
CYx4	Fehler in Muting-Sequenz	Fehler in der Muting-Sequenz in den Zuständen 8000, 8011, 8311, 8012, 8021, 8014, 8314, 8122, 8422, 8121, 8112, 8114 oder 8414. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE Y = Zustand in der Sequenz (6 Zustände für vorwärts und 6 Zustände für rückwärts). C0x4 = Fehler in Zustand 8000 C1x4 = Fehler in Zustand Vorwärts 8011 C2x4 = Fehler in Zustand Vorwärts 8311 C3x4 = Fehler in Zustand Vorwärts 8012 C4x4 = Fehler in Zustand Vorwärts 8014 C5x4 = Fehler in Zustand Vorwärts 8314 C6x4 = Fehler in Zustand Vorwärts 8021 C7x4 = Fehler in Zustand Rückwärts 8122 C8x4 = Fehler in Zustand Rückwärts 8422 C9x4 = Fehler in Zustand Rückwärts 8121 CAx4 = Fehler in Zustand Rückwärts 8114 CBx4 = Fehler in Zustand Rückwärts 8414 CCx4 = Fehler in Zustand Rückwärts 8112 … CFx4 = MutingEnable fehlt x = Zustand der Sensoren, als der Fehler auftrat (4 Bits: LSB = MS_11; MS_12; MS_21; MSB = MS_22).
C005	Parameter-fehler	Wert für DiscTime11_12, DiscTime21_22 oder MaxMutingTime außerhalb des gültigen Bereichs. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C006	Fehler in MaxMuting-Timer	Zeitfehler: Die aktive Muting-Zeit (bei S_MutingActive = TRUE) übersteigt MaxMutingTime. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C007	Fehler in Timer MS11_12	Zeitfehler: Diskrepanzzeit für das Schalten von MutingSwitch11 und MutingSwitch12 > DiscTime11_12. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C008	Fehler in Timer MS21_22	Zeitfehler: Diskrepanzzeit für das Schalten von MutingSwitch21 und MutingSwitch22 > DiscTime21_22. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

Tab. 56: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8000	AOPD frei	Muting ist nicht aktiv, keine Sicherheitsanforderung der aktiven optoelektronischen Schutzeinrichtung. Wenn die Timer von nachfolgendem Muting noch laufen, werden sie gestoppt. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8001	Init	Der Funktionsbaustein wurde aktiviert. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8002	Sicherheits-anforderung – AOPD	Sicherheitsanforderung von aktiver optoelektronischer Schutzeinrichtung erkannt, Muting ist nicht aktiv. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8003	Warten auf Reset	Sicherheitsanforderung oder Fehler wurde erkannt und behoben. Bedienerquittierung durch Reset erforderlich. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8005	Sicher	Sicherheitsfunktion aktiviert. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8011	Muting vor-wärts – Start 1	Muting vorwärts; Sequenz ist in Startphase nach steigender Flanke von MutingSwitch11. Überwachung von DiscTime11_12 ist aktiviert. Überwachung von MaxMutingTime ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8311	Muting vor-wärts – Start 2	Muting vorwärts; Sequenz ist in Startphase nach steigender Flanke von MutingSwitch12. Überwachung von DiscTime11_12 ist aktiviert. Überwachung von MaxMutingTime ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8012	Muting vor-wärts aktiv 1	Muting vorwärts, Sequenz ist aktiv: – nachdem eine steigende Flanke am zweiten Schaltelement am Eingang, MutingSwitch12 oder MutingSwitch11 erkannt wurde. – wenn sowohl MutingSwitch11 als auch MutingSwitch12 im selben Zyklus betätigt wurden. Überwachung von DiscTime11_12 wurde gestoppt. Überwachung von MaxMutingTime ist aktiviert, wenn der Übergang direkt vom Zustand 8000 kam. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE
8014	Muting vor-wärts – Schritt 1	Muting vorwärts, Sequenz ist aktiv. MutingSwitch21 ist das erste betätigte Schaltelement am Ausgang. Überwachung von Disc-Time21_22 wurde gestartet. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE

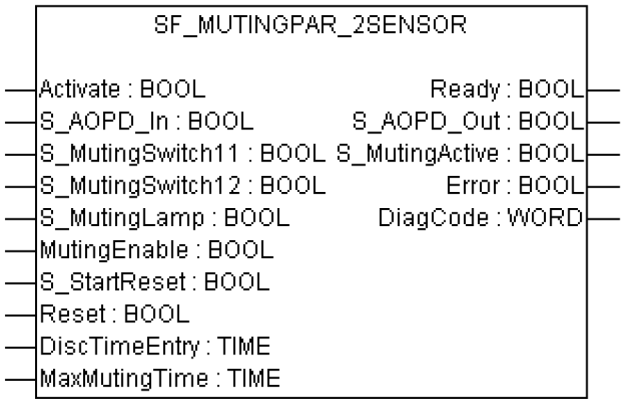
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8314	Muting vor-wärts – Schritt 2	Muting vorwärts, Sequenz ist aktiv. MutingSwitch22 ist das erste betätigte Schaltelement am Ausgang. Überwachung von Disc-Time21_22 wurde gestartet. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE
8021	Muting vor-wärts aktiv 2	Muting vorwärts, Sequenz ist noch aktiv. MutingSwitch21 und MutingSwitch22 wurden beide betätigt, die Überwachung von DiscTime21_22 wurde gestoppt. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE
8122	Muting rück-wärts – Start 1	Muting rückwärts; Sequenz ist in Startphase nach steigender Flanke von MutingSwitch21. Überwachung von DiscTime21_22 ist aktiviert. Überwachung von MaxMutingTime ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8422	Muting rück-wärts – Start 2	Muting rückwärts; Sequenz ist in Startphase nach steigender Flanke von MutingSwitch22. Überwachung von DiscTime21_22 ist aktiviert. Überwachung von MaxMutingTime ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8121	Muting rück-wärts aktiv 1	Muting rückwärts, Sequenz ist aktiv: – nachdem eine steigende Flanke am zweiten Schaltelement am Eingang, MutingSwitch21 oder MutingSwitch22 erkannt wurde. – wenn sowohl MutingSwitch21 als auch MutingSwitch22 im selben Zyklus betätigt wurden. Überwachung von DiscTime21_22 wurde gestoppt. Überwachung von MaxMutingTime ist aktiviert, wenn der Übergang direkt vom Zustand 8000 kam. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8114	Muting rück-wärts – Schritt 1	Muting rückwärts, Sequenz ist aktiv. MutingSwitch11 ist das erste betätigte Schaltelement am Ausgang. Überwachung von Disc-Time11_12 wurde gestartet. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE
8414	Muting rück-wärts – Schritt 2	Muting rückwärts, Sequenz ist aktiv. MutingSwitch12 ist das erste betätigte Schaltelement am Ausgang. Überwachung von Disc-Time11_12 wurde gestartet. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE
8112	Muting rück-wärts aktiv 2	Muting rückwärts, Sequenz ist noch aktiv. Beide Schaltelemente am Ausgang, MutingSwitch11 und MutingSwitch12, wurden betätigt, die Überwachung von DiscTime11_12 wurde gestoppt. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE

4.6.4.14 SF_MutingPar_2Sensor

Normen	Anforderungen
IEC 61496-1:2004	A.7 Muting A.7.1.2 Es müssen mindestens zwei unabhängige und fest verdrahtete Muting-Signalquellen zur Initiierung der Funktion zur Verfügung stehen. Muting darf nicht möglich sein, wenn sich die OSSDs bereits im Zustand AUS befinden. A.7.1.3 Die Muting-Funktion darf nur von der korrekten Sequenz und/oder vom korrekten Timing der Muting-Signale initiiert werden. Sollten widersprüchliche Muting-Signale auftreten, darf die BWS keinen Muting-Zustand zulassen. A.7.1.4 Es müssen mindestens zwei unabhängige und fest verdrahtete Muting-Signalquellen zum Stoppen der Funktion zur Verfügung stehen. Die Muting-Funktion muss stoppen, wenn das erste dieser Muting-Signale seinen Zustand ändert. Die Deaktivierung der Muting-Funktion darf nicht nur auf der Freigabe der BWS beruhen. A.7.1.5 Die Muting-Signale sollten beim Muting durchgehend vorhanden sein. Wenn die Signale nicht durchgehend vorhanden sind, muss eine fehlerhafte Reihenfolge und/oder der Ablauf einer voreingestellten Zeitbegrenzung entweder einen gesperrten Zustand oder eine Wiederanlaufsperre auslösen. A.7.4 Anzeige: Ein Mute-Statussignal oder eine Statusanzeige muss zur Verfügung gestellt werden (in einigen Anwendungen ist ein Anzeigesignal für Muting erforderlich).
IEC 62046/ Ed. 1:2005	5.5.1: … eine Anzeige für den Aktivitätsstatus der Muting-Funktion kann erforderlich sein. Die Muting-Funktion muss automatisch initiiert und terminiert werden … Falsche Signale, Sequenzen oder Timing der Muting-Sensoren oder Signale dürfen keinen Muting-Zustand zulassen. Eine Initiierung der Muting-Funktion darf in den folgenden Fällen nicht möglich sein: ● Die Schutzausrüstungs-OSSDs befinden sich im Zustand AUS; ● Die Schutzausrüstung befindet sich im Sperrzustand; ● Initiierung der Muting-Funktion durch zwei oder mehrere unabhängige Muting-Sensoren, sodass ein einzelner Fehler keinen Mute-Zustand auslösen kann; ● Terminierung der Muting-Funktion durch zwei oder mehrere unabhängige Muting-Sensoren, sodass die Deaktivierung eines Sensors die Muting-Funktion beendet; ● Verwendung von Timing- und Sequenzsteuerung für Muting-Sensoren, um einen korrekten Muting-Betrieb sicherzustellen; 5.5.3: Die folgenden Maßnahmen sind zu berücksichtigen … ● Die Begrenzung des Muting-Zustandes auf einen bestimmten Zeitraum, der für den Transport des Materials durch die Erkennungszone ausreichend ist. Wenn diese Zeit überschritten wird, sollte die Muting-Funktion beendet und alle gefährlichen Bewegungen sollten gestoppt werden. Anhang F.7 Zwei Sensoren – Gekreuzte Lichtschranken (siehe auch Abb. F.7.2 und F.7.3 in der Norm) Die Muting-Funktion sollte nur initiiert werden, wenn die beiden Lichtschranken innerhalb eines Zeitfensters von 4 Sekunden aktiviert werden. Die Muting-Funktion sollte deaktiviert werden, sobald eine der beiden Lichtschranken der Muting-Sensoren nicht mehr aktiviert ist. Ein überwachter Timer zur Begrenzung der Muting-Funktion auf die minimale praktikable Zeit ist erforderlich. Anhang F.5: Methoden zur Vermeidung von Manipulationen der Muting-Funktion: … Verwenden Sie einen vom Steuerungssystem der Maschine erzeugten Aktivierungsbefehl, der die Muting-Funktion nur dann auslöst, wenn dies im Maschinenzyklus erforderlich ist.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Muting ist die gewollte Unterdrückung der Sicherheitsfunktion. In diesem Funktionsbaustein wird paralleles Muting mit zwei Muting-Sensoren spezifiziert.

Muting ist die gewollte Unterdrückung der Sicherheitsfunktion. Dies ist z. B. erforderlich, wenn Material in den Gefahrenbereich transportiert wird, ohne dass die Maschine gestoppt werden soll. Muting wird durch Muting-Sensoren ausgelöst. Die Verwendung von zwei Muting-Sensoren und die korrekte Integration in die Produktionssequenz müssen sicherstellen, dass niemand den Gefahrenbereich betreten kann, während der Lichtvorhang deaktiviert ist. Als Muting-Sensoren können Drucktaster, Näherungsschalter, Lichtschranken, Grenzwertschalter usw. verwendet werden; sie müssen nicht zwingend „failsafe“ sein. Aktives Muting muss durch Kontrollleuchten angezeigt werden.

Es gibt sequenzielles und paralleles Muting. In diesem Funktionsbaustein wird paralleles Muting mit zwei Muting-Sensoren verwendet; eine Erklärung ist unten angegeben. Das Positionieren der Sensoren erfolgt laut Anhang F.7 von IEC 62046, 2005 ↳ *„Beispiel für SF_MutingPar_2Sensor mit zwei Reflexionslichtschranken“ auf Seite 304*. Der Funktionsbaustein kann in beide Richtungen verwendet werden: vorwärts und rückwärts. Die aktuelle Richtung kann jedoch nicht identifiziert werden. Muting sollte durch die Prozesssteuerung über das Signal MutingEnable aktiviert werden, um eine Manipulation zu vermeiden.

Die Eingangsparameter des Funktionsbausteins umfassen die Signale der zwei Muting-Sensoren (S_MutingSwitch11 und S_MutingSwitch12), das OSSD-Signal der berührungslos wirkenden Schutzeinrichtung „AOPD“ (S_AOPD_In) sowie zwei parametrierbare Zeiten (DiscTimeEntry und MaxMutingTime).

Der Eingang S_StartReset darf nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 57: FB-Name: SF_MutingPar_2Sensor

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AOPD_In	BOOL	FALSE	Variable. OSSD-Signal der AOPD. FALSE: Schutzfeld unterbrochen. TRUE: Schutzfeld nicht unterbrochen.
S_MutingSwitch11	BOOL	FALSE	Variable. Zustand des Muting-Sensors 11. FALSE: Muting-Sensor 11 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 11.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_MutingSwitch12	BOOL	FALSE	Variable. Zustand des Muting-Sensors 12. FALSE: Muting-Sensor 12 nicht betätigt. TRUE: Das Werkstück betätigt den Muting-Sensor 12.
S_MutingLamp	BOOL	FALSE	Variable oder Konstante. Zeigt den Betrieb der Muting-Lampe. FALSE: Ausfall der Muting-Lampe. TRUE: Kein Ausfall der Muting-Lampe.
MutingEnable	BOOL	FALSE	Variable oder Konstante. Befehl des Steuerungssystems, der die Muting-Funktion auslöst, sobald dies im Maschinenzyklus erforderlich ist. Nach dem Start der Muting-Funktion kann dieses Signal ausgeschaltet werden. FALSE: Muting nicht aktiviert TRUE: Muting-Funktion aktiviert
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
DiscTimeEntry	TIME	T#0s	Konstante 0..4 s; Maximale Diskrepanzzeit für S_MutingSwitch11 und S_MutingSwitch12 am Muting-Eingang.
MaxMutingTime	TIME	T#0s	Konstante 0..10 min; Maximale Zeit für das Beenden der Muting-Sequenz; der Timer startet, wenn der erste Muting-Sensor betätigt wird.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_AOPD_Out	BOOL	FALSE	Sicherheitsausgang, der den Status der Schutzeinrichtung im Muting-Zustand anzeigt. FALSE: Schutzfeld der aktiven optoelektronischen Schutzeinrichtung unterbrochen und Muting nicht aktiv. TRUE: Schutzfeld der aktiven optoelektronischen Schutzeinrichtung nicht unterbrochen oder Muting aktiv.
S_MutingActive	BOOL	FALSE	Zeigt den Muting-Zustand. FALSE: Muting nicht aktiv. TRUE: Muting aktiv.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Hinweis: Die Leitungskontrolle der Muting-Sensor-Signale muss im Sicherheitskreis aktiv sein.

Beispiel für SF_MutingPar_2 Sensor mit zwei Reflexionslicht-schranken

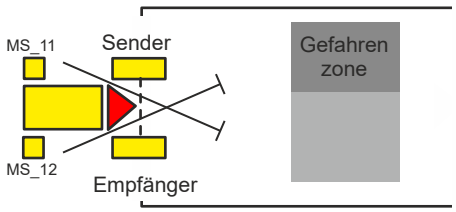


Abb. 111: Beispiel für SF_MutingPar_2Sensor

Wenn Reflexionslichtschranken als Muting-Sensoren verwendet werden, sind diese normalerweise diagonal angeordnet. Im Allgemeinen erfordert diese Anordnung der Reflexionslichtschranken als Muting-Sensoren nur zwei Lichtschranken und nur S_MutingSwitch11 (MS_11) und S_MutingSwitch12 (MS_12) sind zugeordnet.

Bedingungen für Muting

Muting-Bedingung 1 (bis Zustand 8011) (MS_11 ist das erste bestätigte Schaltelement am Eingang). Timer DiscTimeEntry und MaxMutingTime werden gestartet:

MutingEnable AND R_TRIG at MS_11 AND NOT MS_12

Muting-Bedingung 2 (bis Zustand 8311) (MS_12 ist das erste bestätigte Schaltelement am Eingang). Timer DiscTimeEntry und MaxMutingTime werden gestartet:

MutingEnable AND NOT MS_11 AND R_TRIG at MS_12

Muting-Bedingung 3 (von Zustand 8011 bis Zustand 8012) (MS_12 ist das zweite bestätigte Schaltelement am Eingang):

Timer DiscTimeEntry wird gestoppt:

MutingEnable AND MS_11 AND R_TRIG at MS_12

Muting-Bedingung 4 (von Zustand 8311 bis Zustand 8012) (MS_11 ist das zweite bestätigte Schaltelement am Eingang):

Timer DiscTimeEntry wird gestoppt:

MutingEnable AND R_TRIG at MS_11 AND MS_12

Muting-Bedingung 5 (von Zustand 8000 bis Zustand 8012) (beide Schaltelemente werden im selben Zyklus betätigt): Timer MaxMutingTime wird gestartet:

MutingEnable AND R_TRIG at MS_11 AND R_TRIG at MS_12

Muting-Bedingung 6 (von Zustand 8012 bis Zustand 8000) (beide Schaltelemente werden im selben Zyklus freigegeben oder MS_11 und MS_12 werden nacheinander freigegeben). Timer MaxMutingTime wird gestoppt: NOT MS_11 OR NOT MS_12

Falsche Muting-Sequenzen

Zustand 8000 - (R_TRIG at MS_11 AND MS_12 AND NOT R_TRIG at MS_12) OR
(R_TRIG at MS_12 AND MS_11 AND NOT R_TRIG at MS_11) OR
((MS_11 AND NOT R_TRIG at MS_11) AND (MS_12 AND NOT R_TRIG at MS_12)) OR
(NOT MutingEnable AND R_TRIG at MS_11) OR
(NOT MutingEnable AND R_TRIG at MS_12)

Zustand 8011 - NOT MutingEnable OR NOT MS_11

Zustand 8311 - NOT MutingEnable OR NOT MS_12

Zustand 8012 - Alle möglichen Übergänge zulässig

Typisches Zeitdiagramm

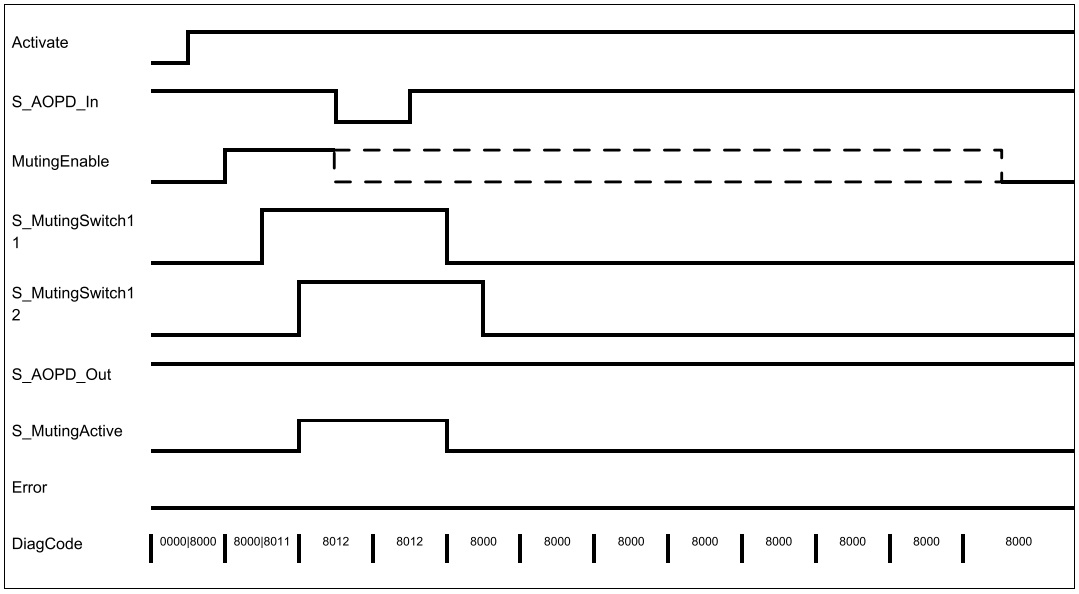


Abb. 112: Zeitdiagramm für SF_MutingPar_2Sensor (S_StartReset = TRUE, Reset = FALSE, S_MutingLamp = TRUE)

Der Funktionsbaustein erkennt die folgenden Fehlerbedingungen:

- Der Wert für DiscTimeEntry ist kleiner als T#0s oder größer als T#4s.
- Der Wert für MaxMutingTime ist kleiner als T#0s oder größer als T#10min.
- Die Diskrepanzzeit für das Sensorpaar S_MutingSwitch11/S_MutingSwitch12 wurde überschritten.
- Die Muting-Funktion (S_MutingActive = TRUE) überschreitet die maximale Muting-Zeit MaxMutingTime.
- Die Muting-Sensoren S_MutingSwitch11 und S_MutingSwitch12 werden in der falschen Reihenfolge aktiviert.
- Die Muting-Sequenz startet, ohne von MutingEnable aktiviert worden zu sein.
- Statische Muting-Sensor-Signale.
- Eine fehlerhafte Muting-Lampe wird von S_MutingLamp = FALSE angezeigt.
- Eine statische Reset-Bedingung wurde im Zustand 8001 und 8003 erkannt.

Verhalten im Fehlerfall

Bei einem Fehler werden die Ausgänge S_AOPD_Out und S_MutingActive auf FALSE gesetzt. Der Ausgang DiagCode zeigt den relevanten Fehlercode an und der Fehlerausgang wird auf TRUE gesetzt.

Ein Neustart ist erst möglich, wenn die Fehler behoben wurden und der sichere Zustand vom Bediener mit Reset quittiert wurde.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 58: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Nach Aktivierung des Funktionsbausteins im Zustand 8001 wurde eine statische Reset-Bedingung erkannt. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C002	Fehler-Reset 2	Statische Reset-Bedingung im Zustand 8003. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C003	Fehler bei Muting-Lampe	Fehlerhafte Muting-Lampe. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
CYx4	Fehler in Muting-Sequenz	Fehler in der Muting-Sequenz in den Zuständen 8000, 8011, 8311. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE Y = Zustand in der Sequenz C0x4 = Fehler in Zustand 8000 C1x4 = Fehler in Zustand 8011 C2x4 = Fehler in Zustand 8311 CFx4 = MutingEnable fehlt x = Zustand der Sensoren, als der Fehler auftrat (4 Bits: LSB = MS_11; neben LSB = MS_12).
C005	Parameter-fehler	Werte für DiscTimeEntry oder MaxMutingTime außerhalb des gültigen Bereichs. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C006	Fehler in MaxMuting-Timer	Zeitfehler: Die aktive Muting-Zeit (bei S_MutingActive = TRUE) übersteigt MaxMutingTime. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE
C007	Fehler in Timer am Eingang	Zeitfehler: Diskrepanzzeit für das Schalten von S_MutingSwitch11 und S_MutingSwitch12 von FALSE auf TRUE > DiscTimeEntry. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = TRUE

Tab. 59: FB-spezifische Zustandscodes (kein Fehler):

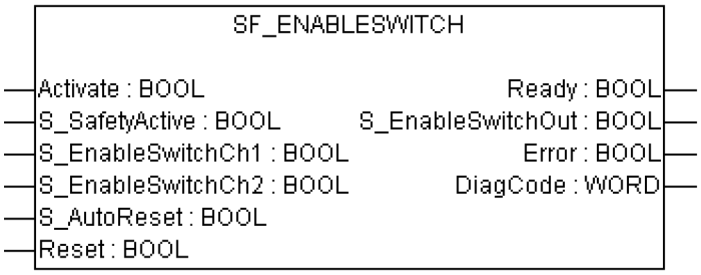
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8000	AOPD frei	Muting ist nicht aktiv, keine Sicherheitsanforderung der aktiven optoelektronischen Schutzeinrichtung. Wenn die Timer von nachfolgendem Muting noch laufen, werden sie gestoppt. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8001	Init	Der Funktionsbaustein wurde aktiviert. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8002	Sicherheits-anforderung – AOPD	Sicherheitsanforderung von aktiver optoelektronischer Schutzeinrichtung erkannt, Muting ist nicht aktiv. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8003	Warten auf Reset	Sicherheitsanforderung oder Fehler wurde erkannt und behoben. Bedienerquittierung durch Reset erforderlich. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8005	Sicher	Sicherheitsfunktion aktiviert. Ready = TRUE S_AOPD_Out = FALSE S_MutingActive = FALSE Error = FALSE
8011	Muting Start 1	Die Muting-Sequenz ist in Startphase nach steigender Flanke von S_MutingSwitch11. Überwachung von DiscTimeEntry ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8311	Muting Start 2	Die Muting-Sequenz ist in Startphase nach steigender Flanke von S_MutingSwitch12. Überwachung von DiscTimeEntry ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = FALSE Error = FALSE
8012	Muting aktiv	Muting-Sequenz ist aktiv: – nachdem eine steigende Flanke am zweiten Schaltelement, S_MutingSwitch12 oder S_MutingSwitch11 erkannt wurde. – wenn sowohl S_MutingSwitch11 als auch S_MutingSwitch12 im selben Zyklus betätigt wurden. Überwachung von DiscTimeEntry wurde gestoppt. Überwachung von MaxMutingTime ist aktiviert. Ready = TRUE S_AOPD_Out = TRUE S_MutingActive = TRUE Error = FALSE

4.6.4.15 SF_EnableSwitch

Normen	Anforderungen
IEC 60204-1, Ed. 5.0:2003	9.2.6.3: Steuerungsfreigabe (siehe auch 10.9 unten) ist eine manuell aktivierte Steuerungsfunktion-Verriegelung mit den folgenden Eigenschaften: ● ermöglicht bei Aktivierung die Initiierung des Maschinenbetriebs über ein separates Start-Bedienelement und ● initiiert eine Haltfunktion und verhindert ein Starten des Maschinenbetriebs, wenn sie deaktiviert ist. Die Steuerungsfreigabe muss so eingerichtet sein, dass die Möglichkeiten, diese zu umgehen, minimiert werden. Dies kann z. B. erreicht werden, wenn das Steuerungsfreigabegerät vor dem erneuten Start des Maschinenbetriebs deaktiviert werden muss. Es darf nicht möglich sein, die Freigabefunktion mit einfachen Mitteln zu umgehen. 10.9: Wenn ein Steuerungsfreigabegerät als Teil eines Systems mitgeliefert wird, muss es die Steuerungsfreigabe anzeigen und den Betrieb im aktivierten Zustand in nur einer Position zulassen. In jeder anderen Position muss der Betrieb gestoppt bzw. verhindert werden. Es sind Steuerungsfreigabegeräte mit den folgenden Eigenschaften auszuwählen: … ● für Ausführungen mit drei Positionen: – Position 1: Aus-Funktion des Schalters (Aktor wird nicht betrieben); – Position 2: Freigabefunktion (Aktor wird in Mittelstellung betrieben); – Position 3: Aus-Funktion (Aktor wird hinter seiner Mittelstellung betrieben); ● Bei der Rückkehr von Position 3 zu Position 2 ist die Freigabefunktion nicht aktiviert.
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Der Funktionsbaustein SF_EnableSwitch wertet die Signale eines Freigabeschalters mit drei Positionen aus.

Der Funktionsbaustein SF_EnableSwitch unterstützt die Aufhebung von Sicherheitsfunktionen (EN 60204 Abschnitt 9.2.4) mit Freigabeschaltern (EN 60204 Abschnitt 9.2.5.8), wenn die relevante Betriebsart ausgewählt und aktiv ist. Die relevante Betriebsart (Begrenzung der Geschwindigkeit oder Antriebskraft, Begrenzung des Bewegungsbereichs) muss außerhalb des Funktionsbausteins SF_EnableSwitch gewählt werden.

Der Funktionsbaustein SF_EnableSwitch wertet die Signale eines Freigabeschalters mit drei Positionen aus (EN 60204 Abschnitt 9.2.5.8).

Die Eingangsparameter S_EnableSwitchCh1 und S_EnableSwitchCh2 verarbeiten die folgenden Signalstufen der Kontakte E1 bis E4:

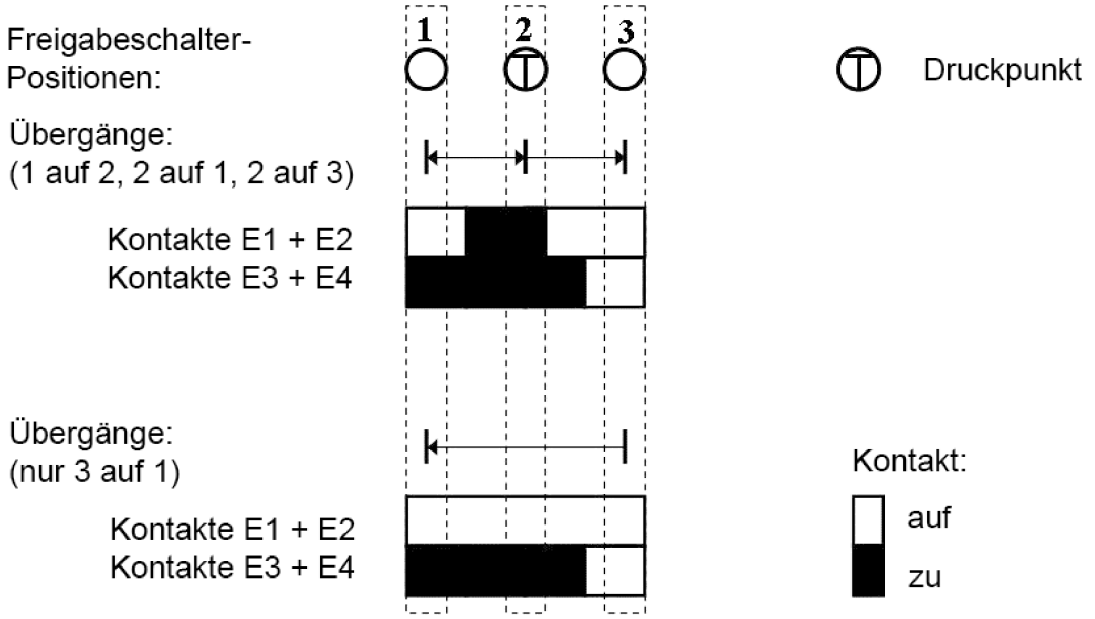


Abb. 113: Stellungen der Schaltelemente

Das Signal von E1+E2 muss mit dem Parameter S_EnableSwitchCh1 verbunden sein. Das Signal von E3+E4 muss mit dem Parameter S_EnableSwitchCh2 verbunden sein. Die Stellung des Freigabeschalters wird mit dieser Signalsequenz im Funktionsbaustein erkannt.

Der Übergang von Position 2 auf 3 kann von dem hier gezeigten abweichen.

Die Schaltrichtung (Position 1 => Position 2/Position 3 => Position 2) kann im Funktionsbaustein mit der definierten Signalsequenz der Kontakte des Freigabeschalters erkannt werden. Die Aufhebung der Sicherheitsfunktion kann nur durch den Funktionsbaustein nach einer Bewegung von Position 1 auf Position 2 aktiviert werden. Andere Schaltrichtungen oder Positionen dürfen nicht zur Aufhebung der Sicherheitsfunktion verwendet werden. Diese Maßnahme erfüllt die Forderungen von EN 60204 Abschnitt 9.2.5.8.

Um die Forderungen von EN 60204 Abschnitt 9.2.4 zu erfüllen, müssen die Anwender geeignete Schaltelemente verwenden. Außerdem müssen die Anwender sicherstellen, dass die relevante Betriebsart (EN 60204 Abschnitt 9.2.3) in der Anwendung ausgewählt wurde (der automatische Betrieb muss in dieser Betriebsart durch geeignete Maßnahmen deaktiviert werden).

Die Betriebsart wird normalerweise durch einen Betriebsartenwahlschalter zusammen mit den Funktionsbausteinen SF_ModeSelector und SF_SafeRequest oder SF_SafelyLimitedSpeed festgelegt.

Der Funktionsbaustein SF_EnableSwitch verarbeitet die Bestätigung des Sicherheitsmodus über den Parameter S_SafetyActive. Wird der Sicherheitsmodus ohne Bestätigung in einer Anwendung implementiert, wird ein statisches TRUE-Signal mit dem Parameter S_SafetyActive verbunden.

Der Eingang S_AutoReset darf nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 60: FB-Name: SF_EnableSwitch

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↬ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_SafetyActive	BOOL	FALSE	Variable oder Konstante. Bestätigung des Sicherheitsmodus (Begrenzung der Geschwindigkeit oder Antriebskraft, Begrenzung des Bewegungsbereichs). FALSE: Sicherheitsmodus ist nicht aktiv. TRUE: Sicherheitsmodus ist aktiv.
S_EnableSwitchCh1	BOOL	FALSE	Variable. Signale der Kontakte E1 und E2 sind mit dem Freigabeschalter verbunden. FALSE: Die verbundenen Schaltelemente sind offen. TRUE: Die verbundenen Schaltelemente sind geschlossen.
S_EnableSwitchCh2	BOOL	FALSE	Variable. Signale der Kontakte E3 und E4 sind mit dem Freigabeschalter verbunden. FALSE: Die verbundenen Schaltelemente sind offen. TRUE: Die verbundenen Schaltelemente sind geschlossen.
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_EnableSwitchOut	BOOL	FALSE	Sicherheitsausgang: Zeigt die Aufhebung der Schutzeinrichtung an. FALSE: Aufhebung der Sicherheitsfunktion deaktivieren. TRUE: Aufhebung der Sicherheitsfunktion aktivieren.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typische Zeitdiagramme

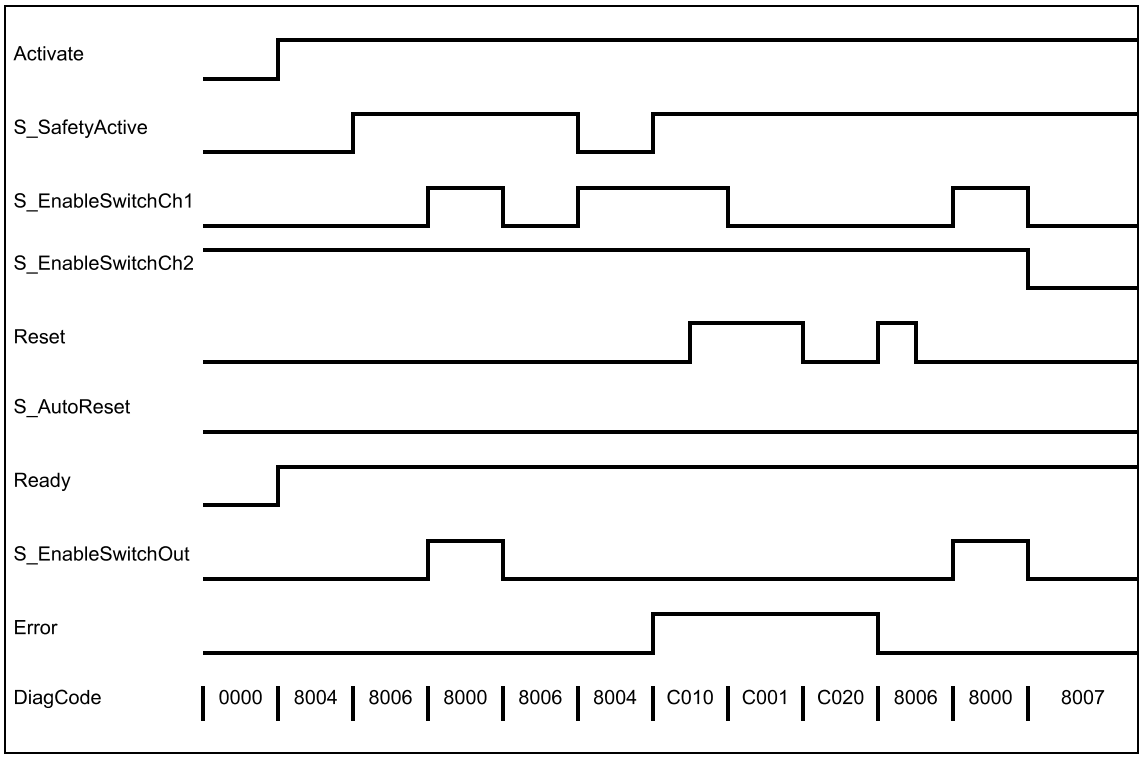


Abb. 114: Zeitdiagramm für SF_EnableSwitch: S_AutoReset = FALSE

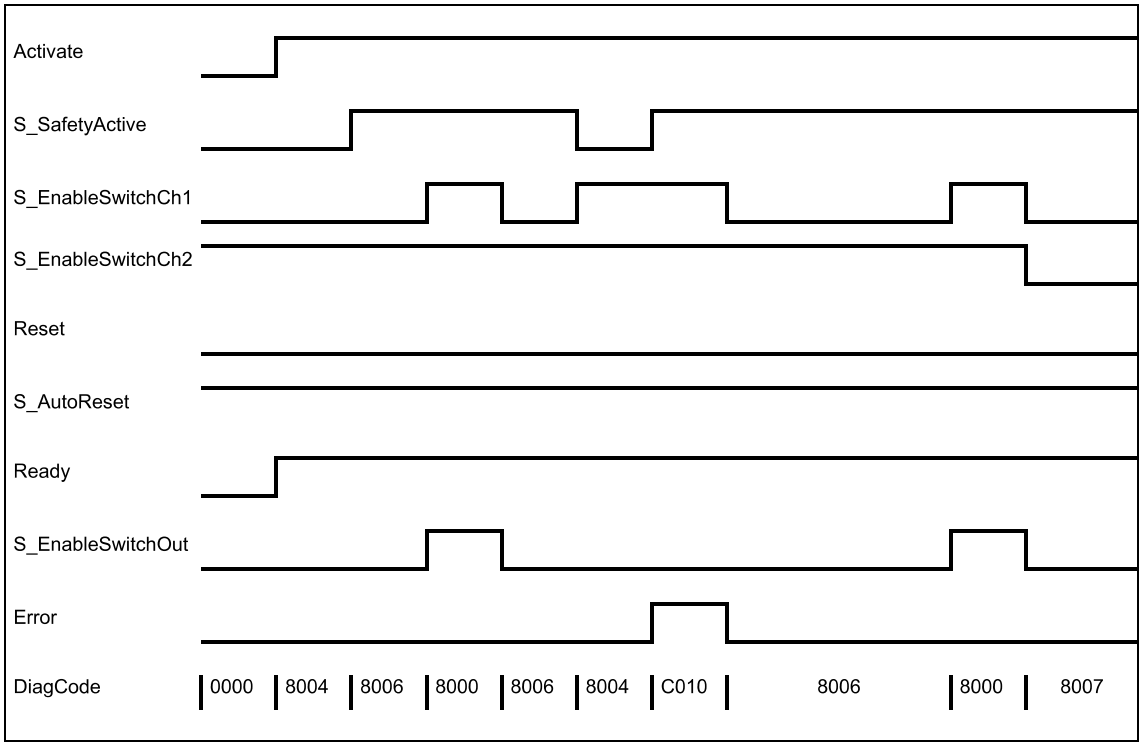


Abb. 115: Zeitdiagramm für SF_EnableSwitch: S_AutoReset = TRUE

Die folgenden Situationen führen zu einem Übergang in den Fehlerzustand:

- Ungültiges statisches Reset-Signal im Prozess.
- Ungültige Stellungen der Schaltelemente.

Verhalten im Fehlerfall

Bei einem Fehler wird der sichere Ausgang S_EnableSwitchOut auf FALSE gesetzt und bleibt in diesem sicheren Zustand.

Im Unterschied zu anderen Funktionsbausteinen kann der Zustand Fehler-Reset durch die Bedingung Reset = FALSE oder zusätzlich durch das Signal S_SafetyActive = FALSE verlassen werden.

Sobald der Fehler behoben wurde, muss der Freigabeschalter in der im Prozess spezifizierten Ausgangsposition sein, bevor der Ausgang S_EnableSwitchOut mit diesem Schaltelement auf TRUE gesetzt werden kann. Bei S_AutoReset = FALSE ist eine steigende Flanke an Reset erforderlich.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 61: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Statisches Reset-Signal im Zustand C020. Ready = TRUE S_EnableSwitchOut = FALSE Error = TRUE
C002	Fehler-Reset 2	Statisches Reset-Signal im Zustand C040. Ready = TRUE S_EnableSwitchOut = FALSE Error = TRUE
C010	Betriebsfehler 1	Der Freigabeschalter ist während der Aktivierung von S_SafetyActive nicht in Position 1. Ready = TRUE S_EnableSwitchOut = FALSE Error = TRUE
C020	Betriebsfehler 2	Freigabeschalter in Position 1 nach C010. Ready = TRUE S_EnableSwitchOut = FALSE Error = TRUE
C030	Betriebsfehler 3	Freigabeschalter in Position 2 nach Position 3. Ready = TRUE S_EnableSwitchOut = FALSE Error = TRUE
C040	Betriebsfehler 4	Freigabeschalter nicht in Position 2 nach C030. Ready = TRUE S_EnableSwitchOut = FALSE Error = TRUE

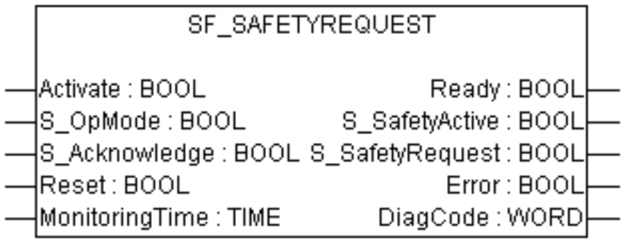
Tab. 62: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_EnableSwitchOut = FALSE Error = FALSE
8004	Grund-Betriebsart	Der Sicherheitsmodus ist nicht aktiv. Ready = TRUE S_EnableSwitchOut = FALSE Error = FALSE
8005	Sichere Betriebsart	Der Sicherheitsmodus ist aktiv. Ready = TRUE S_EnableSwitchOut = FALSE Error = FALSE
8006	Position 1	Der Sicherheitsmodus ist aktiv und der Freigabeschalter ist in Position 1. Ready = TRUE S_EnableSwitchOut = FALSE Error = FALSE
8007	Position 3	Der Sicherheitsmodus ist aktiv und der Freigabeschalter ist in Position 3. Ready = TRUE S_EnableSwitchOut = FALSE Error = FALSE
8000	Position 2	Der Sicherheitsmodus ist aktiv und der Freigabeschalter ist in Position 2. Ready = TRUE S_EnableSwitchOut = TRUE Error = FALSE

4.6.4.16 SF_SafetyRequest

Normen	Anforderungen
IEC 60204-1, Ed. 5.0:2003	9.2.4 Aufhebung von Sicherheitsfunktionen und/oder Schutzmaßnahmen Wenn eine Aufhebung von Sicherheitsfunktionen und/oder Schutzmaßnahmen erforderlich ist (z. B. für Einstellungs- oder Wartungszwecke), muss der Schutz wie folgt sichergestellt werden: ● Deaktivierung aller anderen Betriebs- und Steuerungsarten und ● andere relevante Vorgehensweisen (siehe 4.11.9 in ISO 12100-2:2003), die folgende Punkte beinhalten können: – Begrenzung der Geschwindigkeit oder Antriebskraft; – Begrenzung des Bewegungsbereichs;
EN 954-1:1996	5.4 Manuelles Rücksetzen
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Der Funktionsbaustein repräsentiert die Schnittstelle zwischen Anwenderprogramm und Systemumgebung.

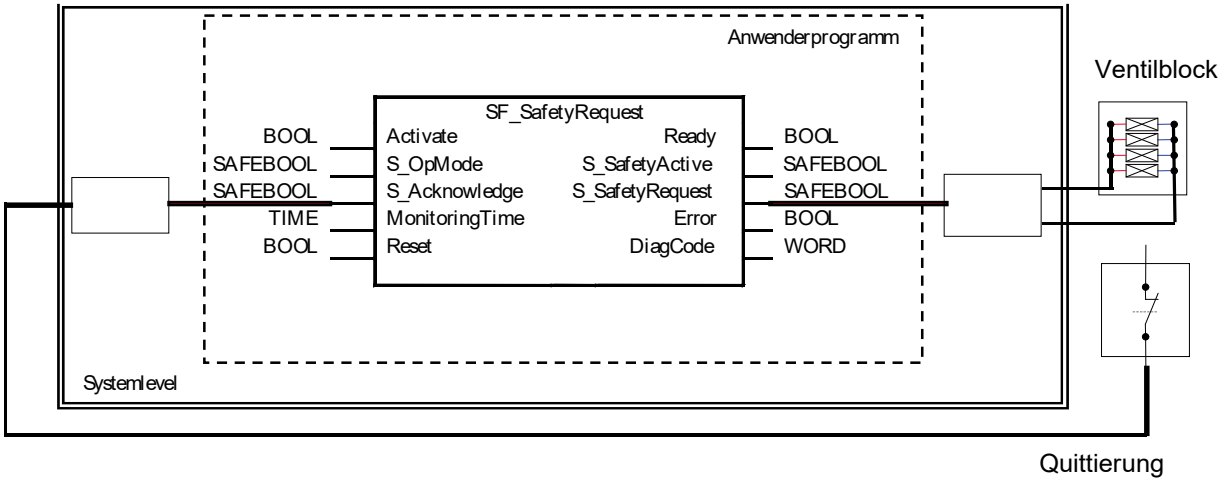


Abb. 116: Beispiel für SF_SafetyRequest

Dieser Funktionsbaustein ist die Schnittstelle zu einem allgemeinen Aktor, z. B. einem Sicherheitsantrieb oder Sicherheitsventil; durch ihn wird der Aktor in einen sicheren Zustand gebracht.

Dieser Funktionsbaustein ist die Schnittstelle zwischen Sicherheitssystem und einem allgemeinen Aktor. Dies bedeutet, dass die Sicherheitsfunktionen des Aktors im Anwendungsprogramm verfügbar sind. Es gibt jedoch nur zwei Binärsignale, um den sicheren Zustand des allgemeinen Aktors zu kontrollieren, d. h. eines zum Anfordern der Bestätigung und eines zum Empfang dieser Bestätigung.

Die Sicherheitsfunktion wird vom Aktor zur Verfügung gestellt. Deshalb initiiert der Funktionsbaustein nur die Anforderung, überwacht sie und setzt den Ausgang, wenn der Aktor den sicheren Zustand quittiert. Dies wird vom Ausgang S_SafetyActive angezeigt.

Dieser Funktionsbaustein definiert keine allgemeinen Aktor-spezifischen Parameter. Sie sollten beim allgemeinen Aktor selbst spezifiziert werden. Er schaltet den allgemeinen Aktor von der Betriebsart in den sicheren Zustand.

Tab. 63: FB-Name: SF_SafetyRequest

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_OpMode	BOOL	FALSE	Variable. Angeforderter Modus eines allgemeinen Sicherheits-aktors. FALSE: Der Sicherheitsmodus ist angefordert. TRUE: Die Betriebsart ist angefordert.
S_Acknowledge	BOOL	FALSE	Variable. Bestätigung des allgemeinen Aktors, wenn dieser im sicheren Zustand ist. FALSE: Betriebsart (nicht sicher). TRUE: Sicherheitsmodus.
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
MonitoringTime	TIME	T#0s	Konstante. Überwachung der Antwortzeit zwischen Anforderung der Sicherheitsfunktion (S_OpMode auf FALSE) und Quittierung durch den Aktor (S_Acknowledge schaltet auf TRUE).
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_SafetyActive	BOOL	FALSE	Bestätigung des sicheren Zustands. FALSE: Kein sicherer Zustand. TRUE: Sicherer Zustand.
S_SafetyRequest	BOOL	FALSE	Anforderung, den Aktor in einen sicheren Zustand zu bringen. FALSE: Der sichere Zustand ist angefordert. TRUE: Kein sicherer Zustand.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typisches Zeitdiagramm

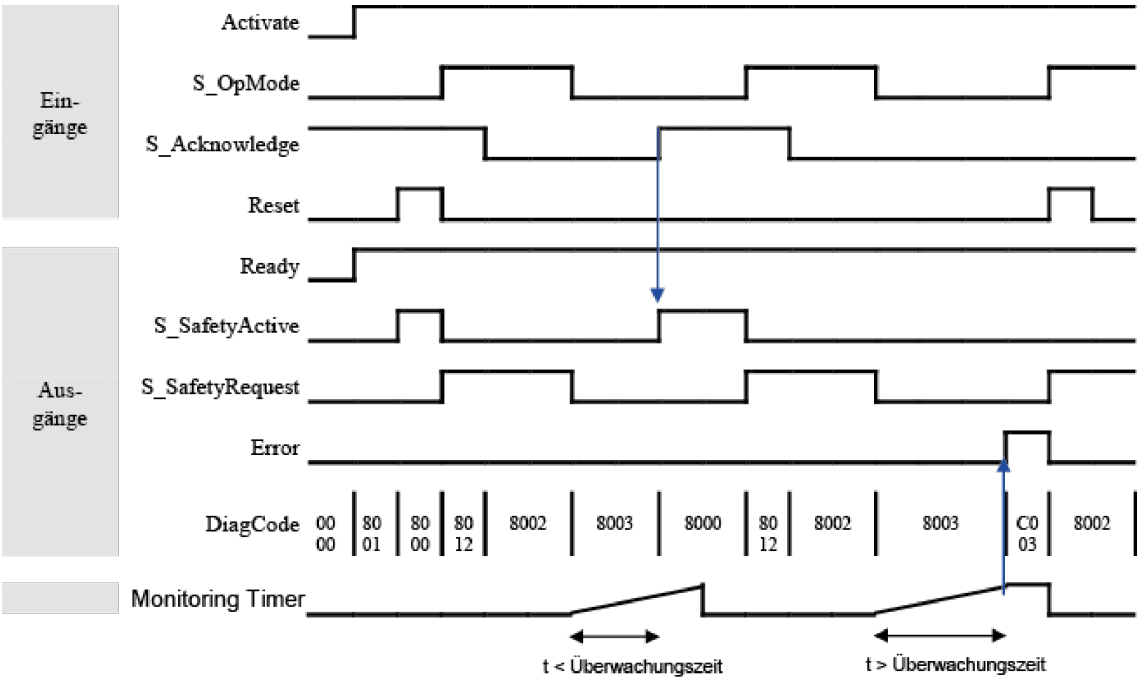


Abb. 117: Zeitdiagramm für SF_SafetyRequest

Der Funktionsbaustein erkennt, wenn der Aktor nicht innerhalb der Überwachungszeit in den sicheren Zustand geht.

Der Funktionsbaustein erkennt, ob das Quittiersignal verloren ging, während die Anforderung noch aktiv war.

Der Funktionsbaustein erkennt ein statisches Reset-Signal.

Externe Funktionsbausteinfehler: Es gibt keine externen Fehler, da der allgemeine Aktor keine Fehlerbits/-information bereitstellt.

Verhalten im Fehlerfall

Bei einem Fehler wird der Ausgang S_SafetyActive auf FALSE gesetzt.

Ein Fehler muss mit einer steigenden Flanke am RESET-Eingang quittiert werden. Um mit dem Funktionsbaustein nach dem Rücksetzen fortzufahren, muss die Anforderung S_OpMode auf TRUE gesetzt werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 64: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C002	Keine Quittierung	Das Quittiersignal ging im sicheren Zustand verloren. Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = TRUE
C003	Überwachungszeit abgelaufen	Die Anforderung S_OpMode konnte nicht innerhalb der Überwachungszeit beendet werden. Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C004	Fehler-Reset 2	Statisches Reset im Zustand C002 (Quittierung verloren). Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = TRUE
C005	Fehler-Reset 3	Statisches Reset im Zustand C003 (MonitoringTime abgelaufen). Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = TRUE

Tab. 65: FB-spezifische Zustandscodes (kein Fehler):

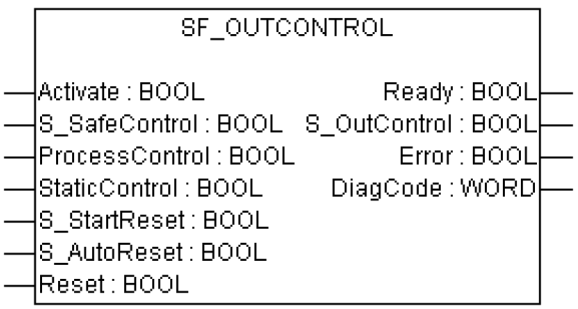
DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = FALSE
8000	Sicherheits-modus	Der Aktor ist in einem sicheren Modus. Ready = TRUE S_SafetyActive = TRUE S_SafetyRequest = FALSE Error = FALSE
8001	Init	Zustand, nachdem Activate auf TRUE gesetzt wurde, oder nach einer steigenden Flanke an Reset. Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = FALSE
8002	Betriebsart	Betriebsart ohne Quittierung des Sicherheitsmodus Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = TRUE Error = FALSE
8012	Warte auf Bestätigung der Betriebsart	Betriebsart mit Quittierung des Sicherheitsmodus Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = TRUE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8003	Warte auf Bestätigung	Warten auf Bestätigung des Antriebs (Systemschnittstelle). Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = FALSE
8005	Warten auf Betriebsart	Der Fehler wurde behoben. S_OpMode muss jedoch erst auf TRUE gesetzt werden, bevor der Funktionsbaustein initialisiert werden kann. Ready = TRUE S_SafetyActive = FALSE S_SafetyRequest = FALSE Error = FALSE

4.6.4.17 SF_OutControl

Normen	Anforderungen
IEC 60204-1, Ed. 5.0:2003	9.2.2: Stoppfunktionen: Stoppfunktionskategorien; Kategorie 0 – Stopp durch unmittelbares Trennen der Stromversorgung von den Aktoren der Maschine (d. h. ein unkontrollierter Stopp …) 9.2.5.2: Start: Start des Betriebs darf nur möglich sein, wenn alle relevanten Sicherheitsfunktionen und/oder Schutzmaßnahmen mit Ausnahme der in 9.2.4 beschriebenen Bedingungen vorhanden und betriebsbereit sind. Um einen Start in korrekter Reihenfolge sicherzustellen, sind geeignete Verriegelungen vorzusehen.
EN 954-1:1996	5.2: Stoppfunktion; ein von Schutzvorrichtungen initiierter Stopp muss die Maschine in einen sicheren Zustand versetzen … und muss gegenüber einem Stopp aus betrieblichen Gründen Priorität haben. 5.5: Start und Neustart; automatischer Neustart nur, wenn das Vorliegen einer gefährlichen Situation ausgeschlossen ist. 5.11: Schwankungen der zugeführten Energie; für den Fall, dass die Energieversorgung ausfällt, sind Ausgänge zur Erhaltung eines sicheren Zustands zur Verfügung zu stellen bzw. zu initiieren.
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen
EN 954-1:1996	5.4 Manuelles Rücksetzen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Kontrolle eines Sicherheitsausgangs mit einem Signal von einer funktionalen Anwendung und einem Sicherheitssignal mit optionaler Anlaufsperre.

Der Funktionsbaustein SF_OutControl ist ein Ausgangstreiber für einen Sicherheitsausgang.

Der Sicherheitsausgang wird über S_OutControl mithilfe eines Signals von der funktionalen Anwendung (ProcessControl zur Prozesskontrolle) und eines Signals von der Sicherheitsanwendung (S_SafeControl zur Kontrolle der Sicherheitsfunktion) kontrolliert.

Optionale Bedingungen für Prozesskontrolle (ProcessControl):

- Ein zusätzlicher Funktionsstart (ProcessControl FALSE => TRUE) ist nach Blockaktivierung oder Feedback vom sicheren Signal (S_SafeControl) erforderlich. Ein statisches TRUE-Signal an ProcessControl setzt S_OutControl nicht auf TRUE.
- Ein zusätzlicher Funktionsstart (ProcessControl FALSE => TRUE) ist nach Blockaktivierung oder Feedback vom sicheren Signal (S_SafeControl) nicht erforderlich. Ein statisches TRUE-Signal an ProcessControl setzt S_OutControl auf TRUE, wenn die anderen Bedingungen erfüllt wurden.

Optionale Anlaufsperre:

- Anlaufsperre nach Aktivierung des Funktionsbausteins.
- Anlaufsperre nach Unterbrechung der Schutzeinrichtung.

Die Eingänge StaticControl, S_StartReset und S_AutoReset dürfen nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 66: FB-Name: SF_OutControl

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_SafeControl	BOOL	FALSE	Variable. Kontrollsignal des vorhergehenden Sicherheits-Funktionsbausteins. Typische Funktionsbaustein-Signale aus der Bibliothek (z. B. SF_EStop, SF_GuardMonitoring, SF_TwoHandControlTypeII usw.). FALSE: Die vorhergehenden Sicherheits-Funktionsbausteine sind in einem sicheren Zustand. TRUE: Die vorhergehenden Sicherheits-Funktionsbausteine aktivieren die Sicherheitskontrolle.
ProcessControl	BOOL	FALSE	Variable oder Konstante. Kontrollsignal der funktionalen Anwendung. FALSE: Setzen von S_OutControl auf FALSE angefordert. TRUE: Setzen von S_OutControl auf TRUE angefordert.
StaticControl	BOOL	FALSE	Konstante. Optionale Bedingungen für Prozesskontrolle. FALSE: Dynamische Veränderung an Process-Control (FALSE => TRUE) erforderlich nach Blockaktivierung oder Auslösen der Sicherheitsfunktion. Start von Zusatzfunktion erforderlich. TRUE: Keine dynamische Veränderung an Process-Control (FALSE => TRUE) erforderlich nach Blockaktivierung oder Auslösen der Sicherheitsfunktion.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
S_StartReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_AutoReset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	*Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_OutControl	BOOL	FALSE	Steuert die angeschlossenen Aktoren. FALSE: Angeschlossene Aktoren deaktivieren. TRUE: Angeschlossene Aktoren aktivieren.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typische Zeitdiagramme

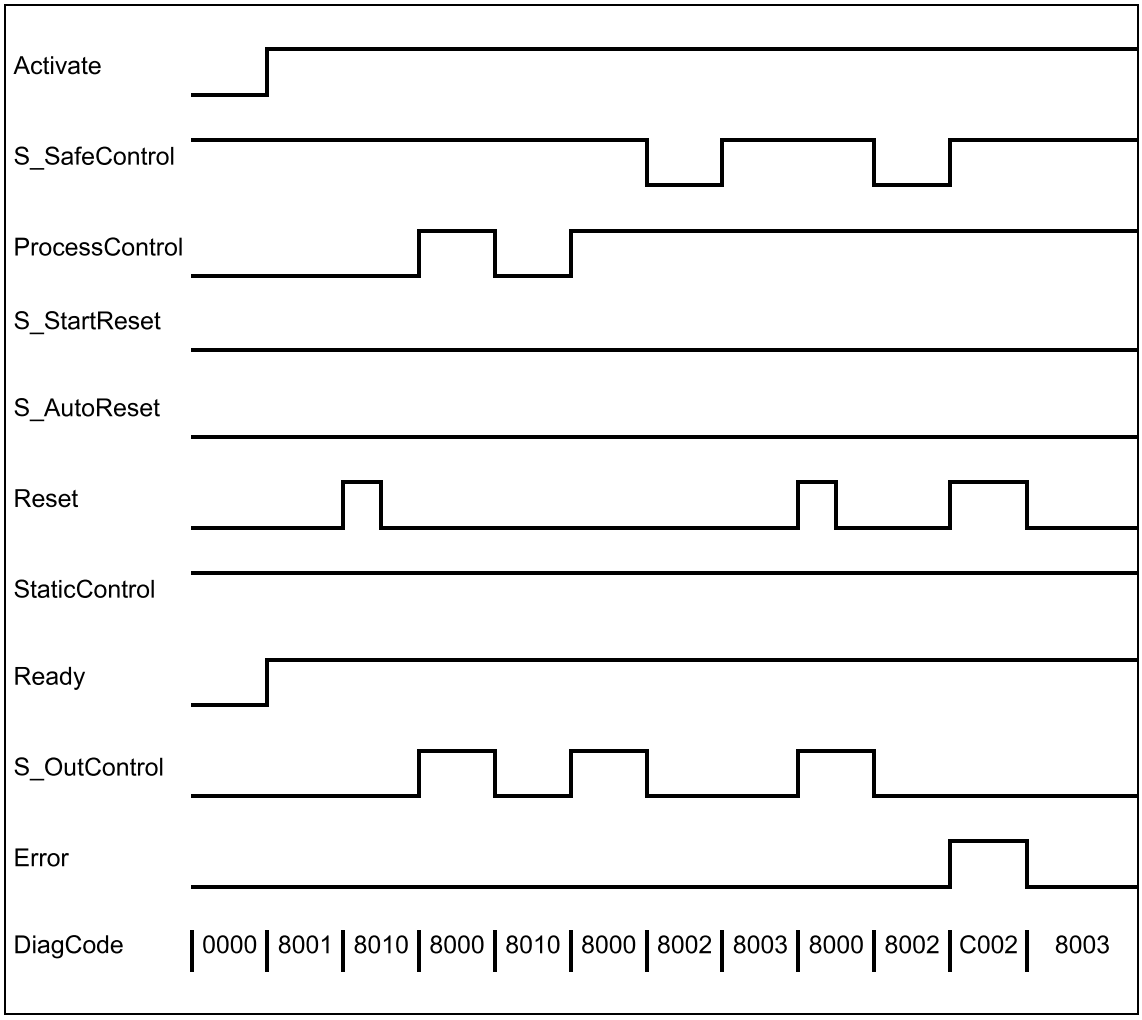


Abb. 118: Zeitdiagramm für SF_OutControl: S_StartReset = FALSE

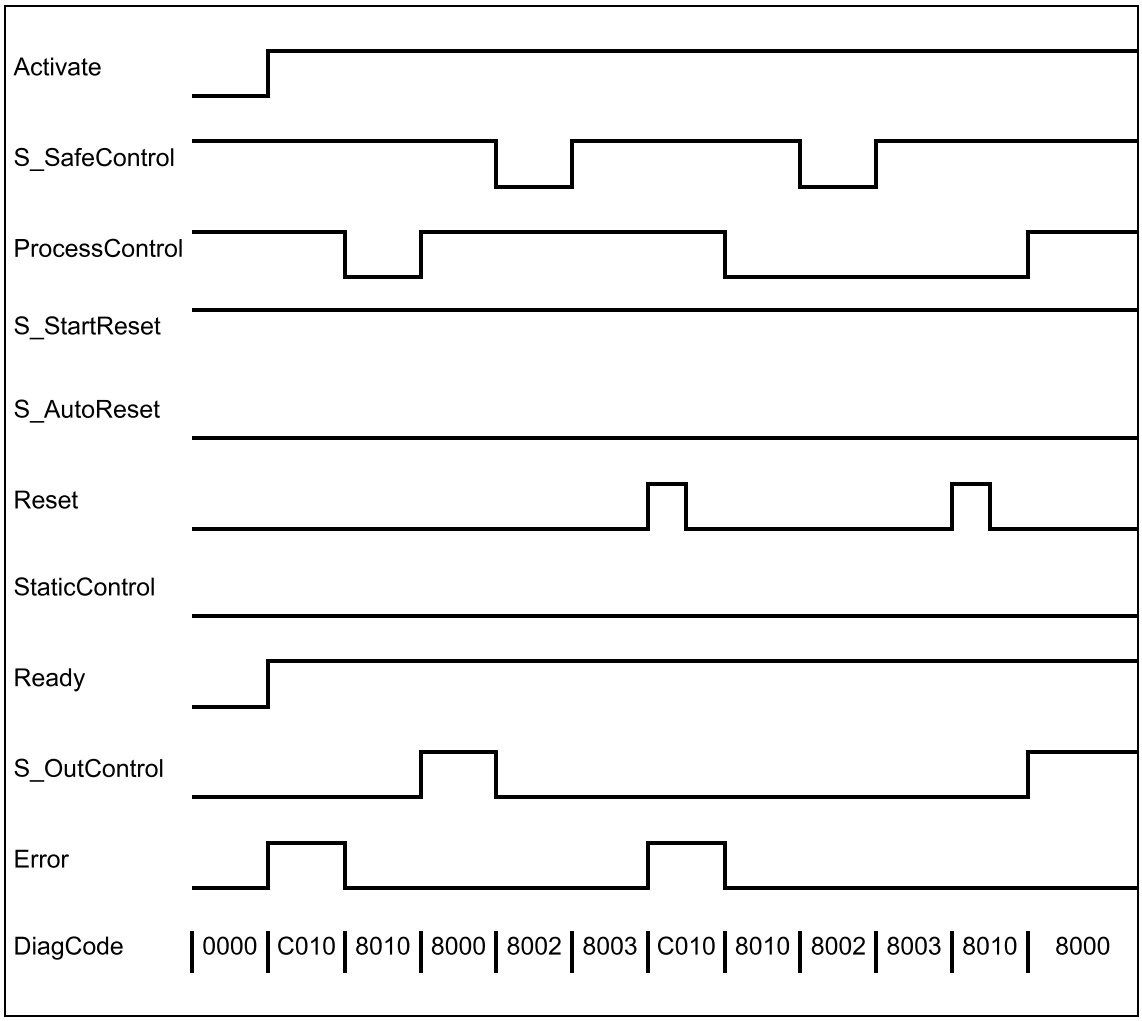


Abb. 119: Zeitdiagramm für SF_OutControl: S_StartReset = TRUE

Die folgenden Situationen führen zu einem Übergang in den Fehlerzustand:

- Ungültiges statisches Reset-Signal im Prozess.
- Ungültiges statisches ProcessControl-Signal.
- ProcessControl und Reset sind aufgrund eines Programmierfehlers falsch verbunden.

Verhalten im Fehlerfall

Bei einem Fehler wird der Ausgang S_OutControl auf FALSE gesetzt und bleibt in diesem sicheren Zustand.

Um den Zustand Reset, Initialisierung oder Verriegelungsfehler zu verlassen, muss der RESET-Eingang auf FALSE gesetzt werden. Um den Zustand Steuerungsfehler zu verlassen, muss der Eingang ProcessControl auf FALSE gesetzt werden.

Nachdem S_SafeControl auf TRUE gesetzt wurde, kann die optionale Anlaufsperre durch eine steigende Flanke am RESET-Eingang zurückgesetzt werden.

Nach Aktivierung des Bausteins kann die optionale Anlaufsperre durch eine steigende Flanke am RESET-Eingang zurückgesetzt werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 67: FB-spezifische Fehlercodes

DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Statisches Reset-Signal im Zustand 8001. Ready = TRUE S_OutControl = FALSE Error = TRUE
C002	Fehler-Reset 2	Statisches Reset-Signal im Zustand 8003. Ready = TRUE S_OutControl = FALSE Error = TRUE
C010	Steuerungsfehler	Statisches Signal an ProcessControl in Zustand 8010. Ready = TRUE S_OutControl = FALSE Error = TRUE
C111	Initialisierungsfehler	Gleichzeitig steigende Flanke an Reset und ProcessControl im Zustand 8001. Ready = TRUE S_OutControl = FALSE Error = TRUE
C211	Verriegelungsfehler	Gleichzeitig steigende Flanke an Reset und ProcessControl im Zustand 8003. Ready = TRUE S_OutControl = FALSE Error = TRUE

Tab. 68: FB-spezifische Zustandscodes (kein Fehler):

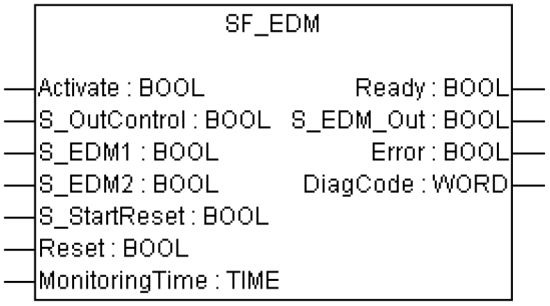
DiagCode	Zustandsname	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_OutControl = FALSE Error = FALSE
8001	Init	Anlaufsperre nach Aktivierung des Funktionsbausteins ist aktiv. Reset erforderlich. Ready = TRUE S_OutControl = FALSE Error = FALSE
8002	Sicher	Sicherheitsfunktion ausgelöst. Ready = TRUE S_OutControl = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8003	Sperre	Anlaufsperre nach Sicherheitsfunktion ist aktiv. Reset erforderlich. Ready = TRUE S_OutControl = FALSE Error = FALSE
8010	Ausgangs-deaktivie-rung	Die Prozesssteuerung ist nicht aktiv. Ready = TRUE S_OutControl = FALSE Error = FALSE
8000	Ausgangsak-tivierung	Die Prozesssteuerung ist aktiv und Sicherheit ist aktiviert. Ready = TRUE S_OutControl = TRUE Error = FALSE

4.6.4.18 SF_EDM

Normen	Anforderungen
IEC 60204-1, Ed. 5.0:2003	Abschnitt 9.2.2: Stoppfunktionskategorien; Kategorie 0
EN 954-1:1996	5.2: Stoppfunktion; ein von Schutzvorrichtungen initiierter Stopp muss die Maschine in einen sicheren Zustand versetzen. 6.2: Spezifikation von Kategorien: Fehlererkennung (des Aktors, z. B. Drahtbrüche)
ISO 12100-2:2003	4.11.4: Wiederingangsetzen nach Ausfall der Energieversorgung/spontanes Wiederanlaufen
EN 954-1:1996	5.4 Manuelles Rücksetzen

Hinweis: Der Text in der obigen Tabelle ist eine Übersetzung aus dem englischen Original der jeweiligen Norm.



Externe Geräteüberwachung (EDM): Der Funktionsbaustein kontrolliert den Sicherheitsausgang und überwacht kontrollierte Aktoren, z. B. nachfolgende Schaltelemente.

Der Funktionsbaustein SF_EDM kontrolliert einen Sicherheitsausgang und überwacht kontrollierte Aktoren.

Dieser Funktionsbaustein überwacht den Ausgangszustand der Aktoren über Feedback-Signale (S_EDM1 und S_EDM2), bevor die Aktoren vom Funktionsbaustein aktiviert werden.

Der Funktionsbaustein überwacht den Schaltzustand der Aktoren (MonitoringTime), nachdem die Aktoren vom Funktionsbaustein aktiviert wurden.

Zwei einzelne Feedback-Signale müssen für eine genaue Diagnose der angeschlossenen Aktoren verwendet werden. Ein gemeinsames Feedback-Signal der zwei angeschlossenen Aktoren muss für eine beschränkte, doch einfache, Diagnosefunktion der angeschlossenen Aktoren verwendet werden. Dabei muss der Anwender dieses gemeinsame Signal mit den beiden Parametern S_EDM1 und S_EDM2 verbinden. S_EDM1 und S_EDM2 werden dann vom selben Signal kontrolliert.

Die Schaltelemente, die in der Sicherheitsfunktion verwendet werden, müssen der in der Risikoanalyse festgelegten Kategorie entsprechen (EN 954-1).

Optionale Anlaufsperre:

- Anlaufsperre bei Aktivierung des Funktionsbausteins.

Der Eingang S_StartReset darf nur aktiviert werden, wenn sichergestellt ist, dass vom PES-Start keine Gefahr ausgeht.

Tab. 69: FB-Name: SF_EDM

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Activate	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
S_OutControl	BOOL	FALSE	Variable. Kontrollsignal der vorhergehenden Sicherheits-Funktionsbausteine. Typische Funktionsbaustein-Signale aus der Bibliothek (z. B. SF_OutControl, SF_TwoHandControlTypeII usw.). FALSE: Sicherheitsausgang (S_EDM_Out) deaktivieren. TRUE: Sicherheitsausgang (S_EDM_Out) aktivieren.
S_EDM1	BOOL	FALSE	Variable. Feedback-Signal des ersten angeschlossenen Aktors. FALSE: Schaltzustand des ersten angeschlossenen Aktors. TRUE: Ausgangszustand des ersten angeschlossenen Aktors.
S_EDM2	BOOL	FALSE	Variable. Feedback-Signal des zweiten angeschlossenen Aktors. Wenn nur ein Signal in der Anwendung verwendet wird, muss der Anwender eine graphische Verbindung der Parameter S_EDM1 und S_EDM2 herstellen. S_EDM1 und S_EDM2 werden dann vom selben Signal kontrolliert. FALSE: Schaltzustand des zweiten angeschlossenen Aktors. TRUE: Ausgangszustand des zweiten angeschlossenen Aktors.
S_StartReset	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*
Reset	BOOL	FALSE	↳ *Tab. 16 „Allgemeine Eingangsparameter“ auf Seite 218*

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
MonitoringTime	TIME	#0ms	Konstante. Max. Antwortzeit der angeschlossenen und überwachten Aktoren.
VAR_OUTPUT			
Ready	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
S_EDM_Out	BOOL	FALSE	Steuert den Aktor. Das Ergebnis wird vom Feedback-Signal S_EDMx überwacht. FALSE: Angeschlossene Aktoren deaktivieren. TRUE: Angeschlossene Aktoren aktivieren.
Error	BOOL	FALSE	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*
DiagCode	WORD	16#0000	*Tab. 17 „Allgemeine Ausgabeparameter“ auf Seite 220*

Typische Zeitdiagramme

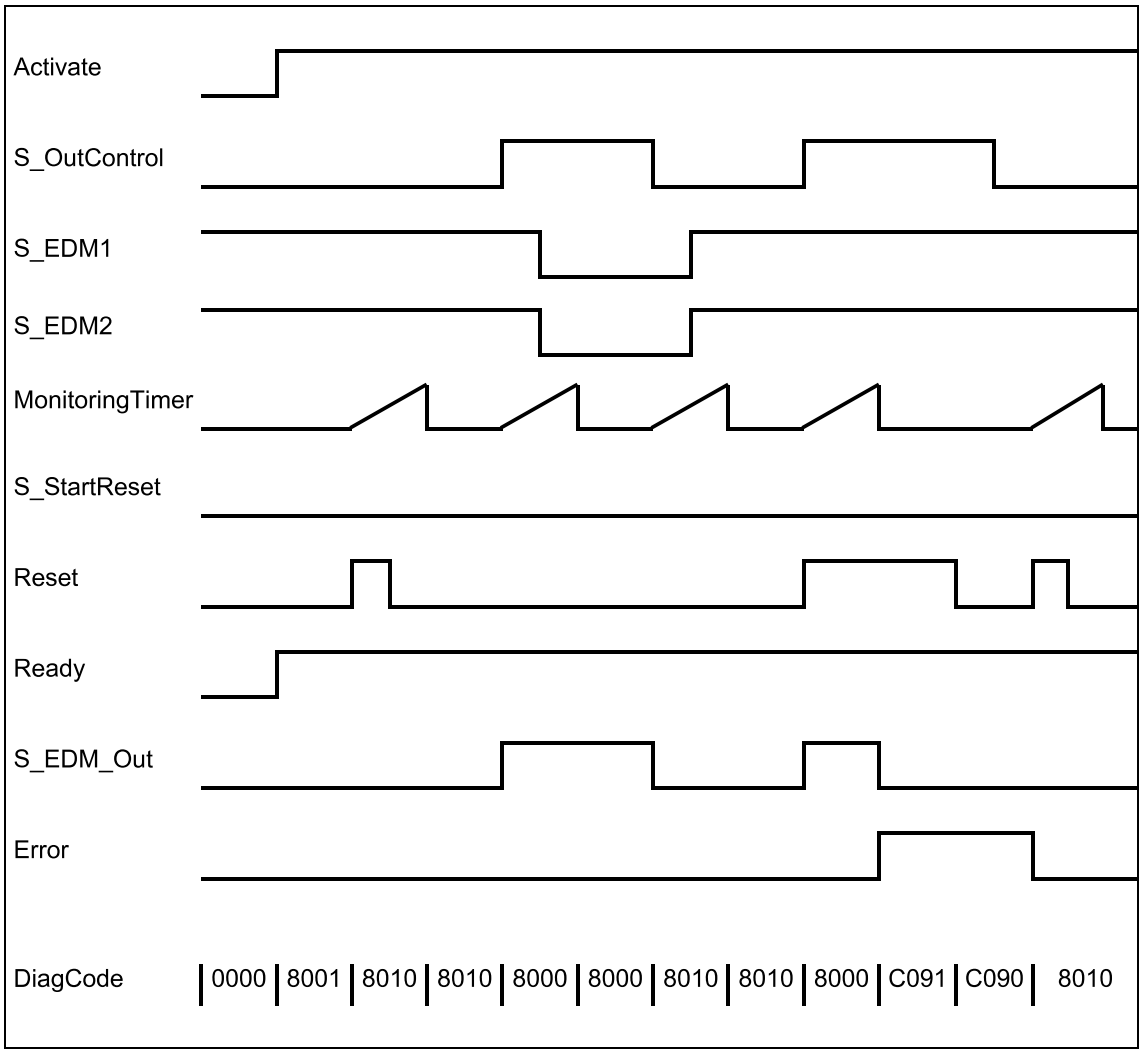


Abb. 120: Zeitdiagramme für SF_EDM: S_StartReset = FALSE

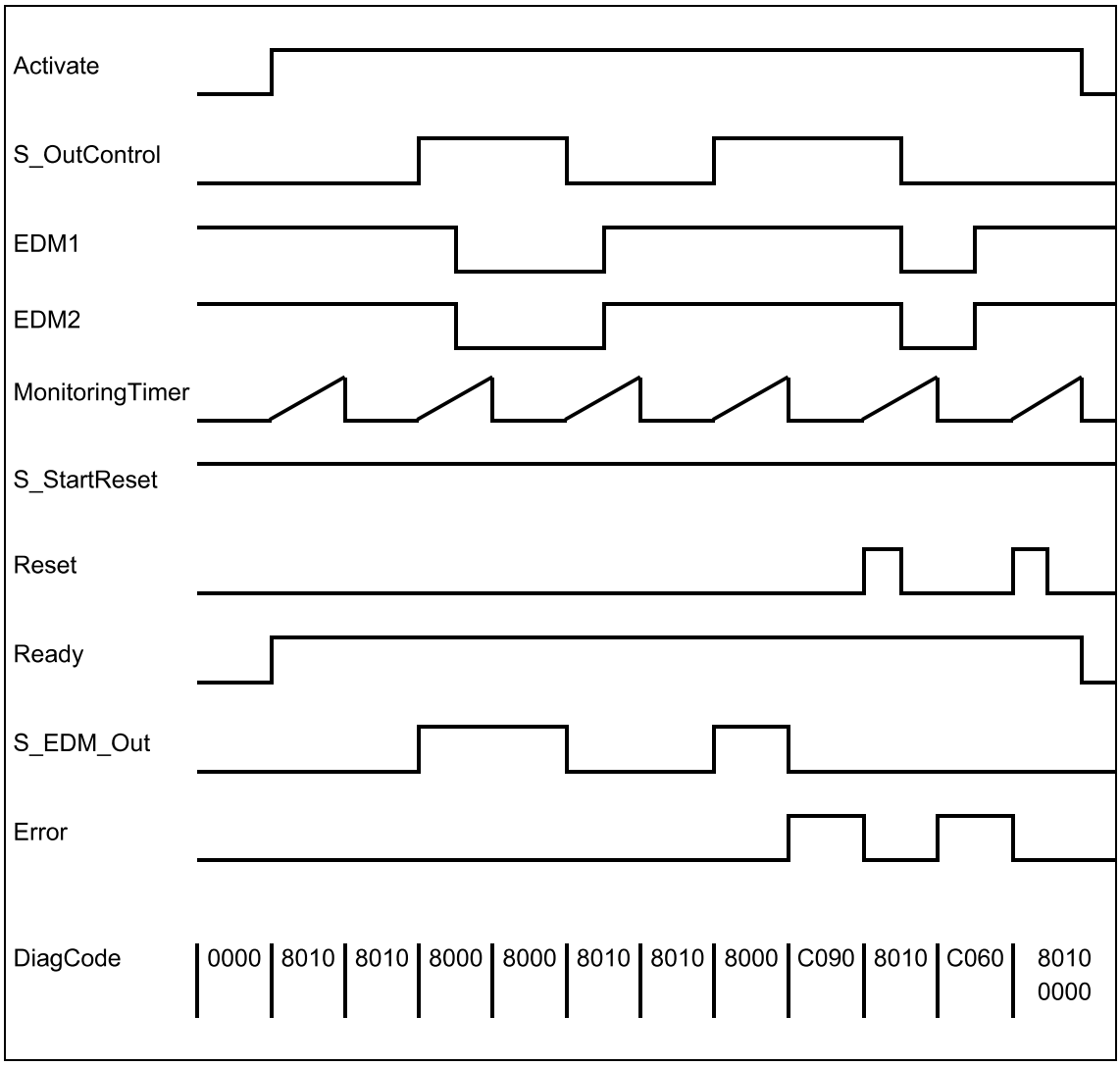


Abb. 121: Zeitdiagramme für SF_EDM: S_StartReset = TRUE

Die folgenden Situationen führen zu einem Übergang in den Fehlerzustand:

- Ungültiges statisches Reset-Signal im Prozess.
- Ungültiges EDM-Signal im Prozess.
- S_OutControl und Reset sind aufgrund eines Programmierfehlers falsch verbunden.

Verhalten im Fehlerfall

Bei Fehlerzuständen sind die Ausgänge wie folgt:

- Bei einem Fehler wird der Ausgang S_EDM_Out auf FALSE gesetzt und bleibt in diesem sicheren Zustand.
- Eine EDM-Fehlermeldung muss immer mit einer steigenden Flanke an Reset zurückgesetzt werden.
- Eine Reset-Fehlermeldung kann durch das Setzen von Reset auf FALSE zurückgesetzt werden.

Nach Aktivierung des Bausteins kann die optionale Anlaufsperre durch eine steigende Flanke am RESET-Eingang zurückgesetzt werden.

Fehler- und Zustandscodes des Funktionsbausteins

Tab. 70: FB-spezifische Fehlercodes

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C001	Fehler-Reset 1	Statisches Reset-Signal im Zustand 8001. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C011	Fehler-Reset 21	Statisches Reset-Signal oder gleiche Signale an EDM1 und Reset (steigende Flanke gleichzeitig an Reset und EDM1) in Zustand C010. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C021	Fehler-Reset 22	Statisches Reset-Signal oder gleiche Signale an EDM2 und Reset (steigende Flanke gleichzeitig an Reset und EDM2) in Zustand C020. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C031	Fehler-Reset 23	Statisches Reset-Signal oder gleiche Signale an EDM1, EDM2 und Reset (steigende Flanke gleichzeitig an Reset, EDM1 und EDM2) in Zustand C030. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C041	Fehler-Reset 31	Statisches Reset-Signal oder gleiche Signale an EDM1 und Reset (steigende Flanke gleichzeitig an Reset und EDM1) in Zustand C040. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C051	Fehler-Reset 32	Statisches Reset-Signal oder gleiche Signale an EDM2 und Reset (steigende Flanke gleichzeitig an Reset und EDM2) in Zustand C050. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C061	Fehler-Reset 33	Statisches Reset-Signal oder gleiche Signale an EDM1, EDM2 und Reset (steigende Flanke gleichzeitig an Reset, EDM1 und EDM2) in Zustand C060. Ready = TRUE S_EDM_Out = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C071	Fehler-Reset 41	Statisches Reset-Signal im Zustand C070. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C081	Fehler-Reset 42	Statisches Reset-Signal im Zustand C080. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C091	Fehler-Reset 43	Statisches Reset-Signal im Zustand C090. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C010	EDM-Fehler 11	Das Signal an EDM1 ist nicht gültig im Ausgangszustand des Aktors. Im Zustand 8010 ist das EDM1-Signal FALSE beim Aktivieren von S_OutControl. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C020	EDM-Fehler 12	Das Signal an EDM2 ist nicht gültig im Ausgangszustand des Aktors. Im Zustand 8010 ist das EDM2-Signal FALSE beim Aktivieren von S_OutControl. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C030	EDM-Fehler 13	Die Signale an EDM1 und EDM2 sind nicht gültig im Ausgangszustand der Aktoren. Im Zustand 8010 sind die EDM1- und EDM2-Signale FALSE beim Aktivieren von S_OutControl. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C040	EDM-Fehler 21	Das Signal an EDM1 ist nicht gültig im Ausgangszustand des Aktors. Im Zustand 8010 ist das EDM1-Signal FALSE und die Überwachungszeit ist abgelaufen. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C050	EDM-Fehler 22	Das Signal an EDM2 ist nicht gültig im Ausgangszustand des Aktors. Im Zustand 8010 ist das EDM2-Signal FALSE und die Überwachungszeit ist abgelaufen. Ready = TRUE S_EDM_Out = FALSE Error = TRUE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
C060	EDM-Fehler 23	Die Signale an EDM1 und EDM2 sind nicht gültig im Ausgangszustand der Aktoren. Im Zustand 8010 sind die EDM1- und EDM2-Signale FALSE und die Überwachungszeit ist abgelaufen. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C070	EDM-Fehler 31	Das Signal an EDM1 ist nicht gültig im Schaltzustand des Aktors. Im Zustand 8000 ist das EDM1-Signal TRUE und die Überwachungszeit ist abgelaufen. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C080	EDM-Fehler 32	Das Signal an EDM2 ist nicht gültig im Schaltzustand des Aktors. Im Zustand 8000 ist das EDM2-Signal TRUE und die Überwachungszeit ist abgelaufen. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C090	EDM-Fehler 33	Die Signale an EDM1 und EDM2 sind nicht gültig im Schaltzustand des Aktors. Im Zustand 8000 sind die EDM1- und EDM2-Signale TRUE und die Überwachungszeit ist abgelaufen. Ready = TRUE S_EDM_Out = FALSE Error = TRUE
C111	Initialisierungsfehler	Ähnliche Signale an S_OutControl und Reset (R_TRIG im selben Zyklus) erkannt (eventuell Programmierfehler) Ready = TRUE S_EDM_Out = FALSE Error = TRUE

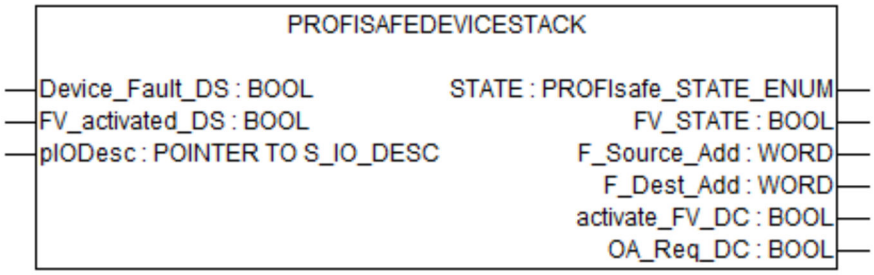
Tab. 71: FB-spezifische Zustandscodes (kein Fehler):

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
0000	Leerlauf	Der Funktionsbaustein ist nicht aktiv (Ausgangszustand). Ready = FALSE S_EDM_Out = FALSE Error = FALSE
8001	Init	Anlaufsperre nach Aktivierung des Funktionsbausteins ist aktiv. Reset erforderlich. Ready = TRUE S_EDM_Out = FALSE Error = FALSE

DiagCode	Zustands-name	Zustandsbeschreibung und Einstellung des Ausgangs
8010	Ausgangs-deaktivie-rung	EDM-Kontrolle ist nicht aktiv. Der Timer startet, wenn der Zustand erreicht wird. Ready = TRUE S_EDM_Out = FALSE Error = FALSE
8000	Ausgangsak-tivierung	EDM-Kontrolle ist aktiv. Der Timer startet, wenn der Zustand erreicht wird. Ready = TRUE S_EDM_Out = TRUE Error = FALSE

4.6.5 SafetyDeviceExt_LV100_PROFIsafe_AC500_V27.lib

Diese Bibliothek enthält eine PROFIsafe F-Device-Stack-Implementierung (durch POE PROFI-SAFEDEVICESTACK); diese ist eine Hauptkomponente des F-Device.



Tab. 72: FB-Name: PROFISAFEDEVICESTACK

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
Device_Fault_DS	BOOL	FALSE	Fehler im Gerät. Mit diesem Parameter kann die Anwendung den F-Host über eine Störung informieren. Ist Device_Fault_DS gesetzt, setzt der Masterstack FV_activated = 1 im Steuerbyte.
FV_activated_DS	BOOL	FALSE	Failsafe-Werte sind aktiviert. Dies ermöglicht der Anwendung, den F-Host darüber zu informieren, dass sie Failsafe-Werte verwendet. Dies wird intern vom PROFIsafe-Device-Stack gesetzt, wenn sich das SM560-S-FD-1 / SM560-S-FD-4 im Zustand DEBUG STOP befindet.
pIODesc	POINTER	NULL	Interner Eingangsparameter. **(Nur für interne Verwendung!)**
VAR_OUTPUT			
STATE	PROFIsafe_STATE_ENUM	PROFIsafe_STATE_INIT	Dieser Parameter gibt den aktuellen Zustand des PROFIsafe-Device-Stack zurück. Der Anwender kann beispielsweise herausfinden, warum der aktuell übertragene F-Parametersatz nicht akzeptiert wurde ↳ *Tab. 73 „Zustände des PROFIsafe F-Device“ auf Seite 334*.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
FV_STATE	BOOL	TRUE	Bei TRUE zeigt dieser Parameter, dass der Device-Stack dem F-Host-Programm für jeden Eingangswert den Failsafe-Wert „0“ liefert. Anderenfalls werden Prozesswerte geliefert.
F_Source_Add	WORD	0	Dieser Parameter stellt die F-Quellenadresse dar, die vom F-Host mit den F-Parametern an dieses F-Device übertragen wurde.
F_Dest_Add	WORD	0	Dieser Parameter legt die F-Zieladresse fest, die mit der Einstellung der Schalteradresse von SM560-S-FD-1 / SM560-S-FD-4 und der Formel für die F-Zieladressen übereinstimmen muss *Tab. 9 „F-Parameter von AC500-S-Sicherheitsmodulen“ auf Seite 153*.
activate_FV_DC	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Bei TRUE zeigt dieser Parameter dem F-Device, dass FV verwendet werden sollen.
OA_Req_DC	BOOL	FALSE	**Dieser Parameter ist nur für Debugging bestimmt.** Mit TRUE fordert der F-Host für das F-Device eine Bedienerquittierung von der F-Host-Sicherheitsanwendung an. Im Falle eines Fehlers (Watchdog-Zeitüberschreitung oder CRC usw.) werden die Failsafe-Werte aktiviert. Liegt der Fehler nicht mehr vor (die Kommunikation mit dem Modul wurde wiederhergestellt) und ist eine Bedienerquittierung möglich, dann setzt der F-Host-Treiber OA_Req_S = TRUE. Wenn die F-Host-Anwendung OA_C = TRUE setzt, wird OA_Req_S auf FALSE zurückgesetzt und der normale Betrieb wieder aufgenommen.

HINWEIS!
Da die F-Device-Instanzen iParameter nicht unterstützen, kann der Funktionsbaustein nicht das Bit iPar_OK_S im Statusbyte setzen oder das Bit iPar_EN_C vom PROFIsafe-Steuerbyte lesen.

Die PROFIsafe F-Device-Instanzen starten nach dem Einschalten asynchron. F-Parameter werden in das PROFINET IO-Device (CM589-PNIO oder CM589-PNIO-4) vom entsprechenden F-Host / PROFINET IO-Controller geschrieben. Diese F-Parameter werden dann über die Standard-CPU an die SM560-S-FD-1 / SM560-S-FD-4 übertragen, die sie zur Parametrierung der F-Device-Instanz verwendet.

Wird die Parametrierung wiederholt, müssen die F-Device-Instanzen zur Laufzeit neu initialisiert werden. F-Parameter werden nur von den AC500-Kommunikationsmodulen und der Standard-CPU übertragen und sind durch F_Par_CRC vor Übertragungsfehlern geschützt.

Die F-Quellenadresse einer F-Device-Instanz wird beim Betrieb vom F-Host mit dem Parameter F_Source_Add in F-Parameter gesetzt. Bei SM560-S-FD-1 / SM560-S-FD-4 wird zusätzlich zu den normalen Prüfungen des F-Device-Stack geprüft, ob sich die F-Quellenadresse einer F-Device-Instanz mit den F-Quellenadressen des eigenen F-Host überschneidet. Liegt eine Überschneidung vor, wird ein Fehler für die neu parametrierte F-Device-Instanz gesetzt.

Sobald die F-Device-Instanz konfiguriert ist, wird weiter geprüft, ob die vom F-Host gemeldeten F-Quellenadressen gültig sind. Falls nicht, wird ein Fehler gesetzt und das Bootprojekt wird nicht geladen.

Der F-Device-Stack kann mit dem Statusbyte folgende Fehler an den F-Host melden:

- Device_Fault: Störung im Gerät. Dieser Fehler kann von der Anwendung durch den Merker Device_Fault_DS im Funktionsbaustein PROFISAFEDEVICESTACK ausgelöst werden.
- CE_CRC (Kommunikationsfehler): CRC-Fehler oder falsche „consecutive number“. Dieser Fehler wird vom Stack automatisch ausgelöst.
- WD_timeout (Watchdog-Zeitüberschreitung): Kein gültiges PROFIsafe-Telegramm innerhalb der F_WD_Time empfangen. Dieser Fehler wird vom Stack automatisch ausgelöst.
- FV_activated_S (Failsafe-Werte sind aktiviert): Zeigt dem F-Host, dass FV verwendet werden. Dies kann auch mit dem Merker FV_activated_DS von der F-Device-Anwendung gesetzt werden.

Der F-Host kann ebenfalls Kommunikationsfehler erkennen (Watchdog-Zeitüberschreitung, CRC-Fehler oder falsche „consecutive number“). Die Anwendung hinter dem entsprechenden F-Device kann über diese Fehler mit dem Merker activate_FV_DC = TRUE der Instanz PROFISAFEDEVICESTACK informiert werden und entsprechend reagieren.

Die Anwendung kann die Ausgangsvariable „STATE“ verwenden, um Informationen über den aktuellen Zustand der F-Device-Instanz zu erhalten.

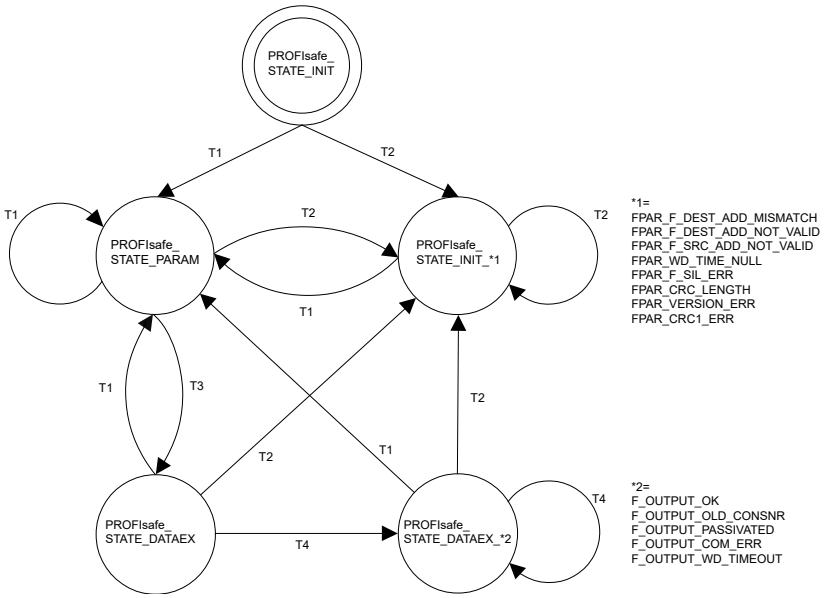


Abb. 122: Zustandsdiagramm des PROFIsafe F-Device

T1 Gute F-Parameter empfangen
T2 Schlechte F-Parameter empfangen
T3 F-Host-Grenze nicht erreicht
T4 Meldung verarbeitet

Die Zustandsübergänge T1 und T2 werden sofort ausgeführt, wenn neue F-Parameter für die F-Device-Instanz übertragen wurden. Ist die Grenze der F-Quellenadresse für die SM560-S-FD-1 (max. 1 F-Quellenadresse) / SM560-S-FD-4 (max. 4 unterschiedliche F-Quellenadressen) noch nicht erreicht, schaltet der Übergang T3 sofort. Ist die Grenze der F-Quellenadresse erreicht, müssen aktive F-Device-Instanzen (Zustände PROFIsafe_STATE_DATAEX) eines F-Host vom Übergang T1 oder T2 gestoppt werden.

Die nachfolgende Tabelle zeigt die Bedeutung jedes Zustands:

Tab. 73: Zustände des PROFIsafe F-Device

Wert des STATE-Ausgangs an PROFIsafe F-Device-Stack-Instanz	**Bedeutung**
PROFIsafe_STATE_INIT	Zustand nach der Initialisierung von F-Device-Instanzen.
PROFIsafe_STATE_FPAR_F_DEST_ADD_MISMATCH	Parametrierungsfehler: Die F-Quellenadresse passt nicht zum gegebenen Wert, der auf dem Wert des Adress-Drehschalters an der Sicherheits-CPU SM560-S-FD-1 / SM560-S-FD-4 basiert. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 28
PROFIsafe_STATE_FPAR_F_DEST_ADD_NOT_VALID	Parametrierungsfehler: F-Zieladresse ungültig. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 1
PROFIsafe_STATE_FPAR_F_SRC_ADD_NOT_VALID	Parametrierungsfehler: Die F-Quellenadresse ist ungültig oder überschneidet sich mit F-Quellenadressen der F-Host-Instanzen. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 2
PROFIsafe_STATE_FPAR_WD_TIME_NULL	Parametrierungsfehler: Watchdog-Zeit auf Null eingestellt. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 11
PROFIsafe_STATE_FPAR_F_SIL_ERR	Parametrierungsfehler: Angefordertes SIL zu hoch. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 10
PROFIsafe_STATE_FPAR_CRC_LENGTH	Parametrierungsfehler: Erforderliche CRC-Länge passt nicht zur Datenlänge. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 42
PROFIsafe_STATE_FPAR_VERSION_ERR	Parametrierungsfehler: PROFIsafe-Versionsfehler Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 40
PROFIsafe_STATE_FPAR_CRC1_ERR	Parametrierungsfehler: CRC-Fehler in F-Parametern. Siehe auch Diagnose ↳ *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422*. Modul 28, Fehler 19

Wert des STATE-Ausgangs an PROFIsafe F-Device-Stack-Instanz	Bedeutung
PROFIsafe_STATE_PARAM	F-Host-Begrenzungsfehler: F-Parameter akzeptiert, aber das F-Device tauscht aufgrund der F-Host-Begrenzung keine Daten aus. Es liegt keine Diagnosemeldung vor. Ggf. muss eine individuelle AC500-Diagnosemeldung erzeugt werden.
PROFIsafe_STATE_DATAEX	F-Parameter werden akzeptiert, die F-Device-Instanz kann Prozessdaten austauschen.
PROFIsafe_STATE_DATAEX_F_OUTPUT_OK	Das PROFIsafe-Ausgangstelegramm für den F-Host ist gültig.
PROFIsafe_STATE_ DATAEX_F_OUTPUT_OLD_CONSNR	Das PROFIsafe-Ausgangstelegramm für den F-Host ist gültig mit einer alten „consecutive number“.
PROFIsafe_STATE_ DATAEX_F_OUTPUT_PASSIVATED	Ein Kommunikationsfehler wurde erkannt oder der F-Host sendet „activate_FV“ im PROFIsafe-Steuerbyte. Ggf. muss eine individuelle AC500-Diagnosemeldung von der Anwendung erzeugt werden (wenn PROFIsafe_STATE_ DATAEX_F_OUTPUT_PASSIVATED am STATE-Ausgang der F-Device-Stack-Instanz erkannt wird).
PROFIsafe_STATE_DATAEX_F_OUTPUT_COM_ERR	PROFIsafe-Fehler: Ein CRC-Fehler im PROFIsafe-Ausgangstelegramm wird erkannt. Ggf. muss eine individuelle AC500-Diagnosemeldung von der Anwendung erzeugt werden (wenn PROFIsafe_STATE_DATAEX_F_OUTPUT_COM_ERR am STATE-Ausgang der F-Device-Stack-Instanz erkannt wird).
PROFIsafe_STATE_ DATAEX_F_OUTPUT_WD_TIMEOUT	PROFIsafe-Fehler: Watchdog-Zeitüberschreitung erkannt. Ggf. muss eine individuelle AC500-Diagnosemeldung von der Anwendung erzeugt werden (wenn PROFIsafe_ STATE_DATAEX_F_OUTPUT_WD_TIMEOUT am STATE-Ausgang der F-Device-Stack-Instanz erkannt wird).

4.6.6 SafetyExt2_LV110_AC500_V27.lib

Die Bibliothek SafetyExt2_LV110_AC500_V27.lib enthält die folgenden POEs:

Systembefehle

- SF_SAFE_STOP (Auslösung des SAFE STOP an der Sicherheits-CPU)

Systeminformationen

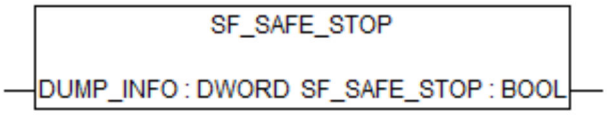
- SF_MAX_POWER_DIP_GET_CFG (gibt die konfigurierte Anzahl von Neustarts nach Spannungsabfall in der Sicherheits-CPU an)
- SF_BOOTPROJECT_CRC (gibt die Bootprojekt-CRC an)

Spezielle Funktionen für benutzerdefinierte CRC

- SF_CRC_INIT (Initialisierung von CRC-Berechnungstabellen für ein anwenderdefiniertes CRC-Polynom)
- SF_CRC_INPUT (Start der CRC-Berechnung für einen Datenblock)
- SF_CRC_FINISH (Rückgabe des CRC-Wertes und Neuinitialisierung für die nächste CRC-Berechnung)

4.6.6.1 SF_SAFE_STOP

Die Funktion SF_SAFE_STOP ermöglicht es dem Anwender, die Sicherheits-CPU direkt in den Zustand SAFE STOP zu versetzen.



Tab. 74: FB-Name: SF_SAFE_STOP

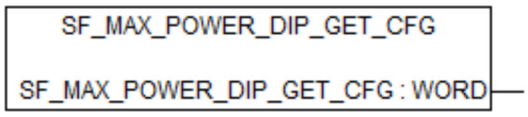
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
DUMP_INFO	DWORD	16#00000000	Der Wert DUMP_INFO wird in den Speicherauszug geschrieben, sodass der Anwender zusammen mit dem Team des ABB-Supports herausfinden kann, an welcher Stelle in der Sicherheitsanwendung der Zustand SAFE STOP ausgelöst wurde.
VAR_OUTPUT			
SF_SAFE_STOP	BOOL	FALSE	Der Ausgang wird nicht verwendet und ist nur verfügbar, da Funktionen mit einem Rückgabewert definiert werden müssen. Die Anwendung kann den Ausgang nicht auswerten, da die Sicherheits-CPU in den sicheren Zustand schaltet.

Aufruf in ST

```
SF_SAFE_STOP(DUMP_INFO:=16#B5006BB1);
```

4.6.6.2 SF_MAX_POWER_DIP_GET_CFG

Die Funktion SF_MAX_POWER_DIP_GET_CFG gibt den konfigurierten, maximalen Spannungsabfallwert der Sicherheits-CPU zurück ↳ *Kapitel 4.6.7.2 „SF_MAX_POWER_DIP_SET“ auf Seite 344* ↳ *Kapitel 4.6.7.6 „SF_MAX_POWER_DIP_GET“ auf Seite 349*.



Tab. 75: FB-Name: SF_MAX_POWER_DIP_GET_CFG

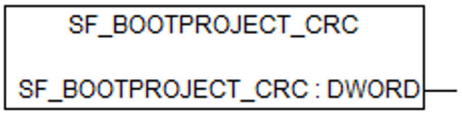
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_OUTPUT			
SF_MAX_POWER_DIP_GET_CFG	WORD	16#0000	Konfigurierte max. Zahl der tolerierten Spannungsabfälle (Unter-/Überspannungsfehler).

Aufruf in ST

```
MAX_POWER_DIPS_CFG := SF_MAX_POWER_DIP_GET_CFG();
```

4.6.6.3 SF_BOOTPROJECT_CRC

Die Funktion SF_BOOTPROJECT_CRC gibt die CRC des Bootprojekts zurück, die im Flash-Speicher war, als die Sicherheits-CPU gestartet wurde (entspricht der Bootprojekt-CRC, die im AC500-S Programming Tool unter dem Menüpunkt *„Online ➔ Bootprojekt in SPS prüfen“* angezeigt wird).



Tab. 76: FB-Name: SF_BOOTPROJECT_CRC

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_OUTPUT			
SF_BOOTPRO-JECT_CRC	DWORD	16#00000000	CRC des Bootprojekts im Flash-Speicher, als die Sicherheits-CPU gestartet wurde.

Aufruf in ST

```
BOOTPROJECT_CRC := SF_BOOTPROJECT_CRC();
```

4.6.6.4 Spezielle Funktionen für benutzerdefinierte CRC

Die Funktionen SF_CRC_INIT, SF_CRC_INPUT und SF_CRC_FINISH bieten eine CRC-Berechnung für einen anwenderdefinierten Datenblock durch ein anwenderdefiniertes CRC-Polynom, z. B. FSoE (Functional Safety over EtherCAT) oder CRC8. Die anwenderdefinierten Daten und die berechnete CRC können sowohl für das Senden der anwenderdefinierten Daten mit der berechneten CRC als auch für das Empfangen der anwenderdefinierten Daten mit der CRC verwendet werden (dann wird die berechnete CRC für den Vergleich mit dem empfangenen CRC-Wert genutzt), wenn für die anwenderdefinierte Sicherheitskommunikation ein azyklischer nicht sicherer Datenaustausch oder ein zyklischer nicht sicherer Datenaustausch genutzt wird (weitere Informationen erhalten Sie beim technischen Support von ABB).

↳ *Anhang B.5 „Datenaustausch zwischen Sicherheits-CPU und AC500 V2-Standard-CPU“ auf Seite 430*
↳ *Anhang C.5 „Datenaustausch zwischen Sicherheits-CPU und AC500 V3-Standard-CPU“ auf Seite 448*

Die CRC-Berechnungstabellen existieren auf nur einem der zwei Mikroprozessoren der Sicherheits-CPU, um eine 1oo2-Sicherheitsarchitektur für die Verarbeitung von Sicherheitstelegrammen zu implementieren. Der gleiche Mechanismus wird für die PROFIsafe-Kommunikation genutzt. Dieser Mechanismus ermöglicht das Erreichen des Safety Integrity Level SIL 3 (IEC 61508 und IEC 62061) und PL e (ISO 13849-1) für den Datenaustausch mittels azyklischem nicht sicherem Datenaustausch oder zyklischem nicht sicherem Datenaustausch.

Damit wird dem Anwender die Möglichkeit gegeben, für die sichere Kommunikation verschiedene Protokolle wie FSoE mit unterschiedlichen CRC-Polynomen (falls erforderlich) zu nutzen und parallel bis zu 8 unterschiedliche CRC-Operationen für die sichere Kommunikation zu verwalten (wobei jeder über den Funktionseingang CRC_SLOT identifiziert wird).

Unter Verwendung der bereitgestellten Funktionen müssen in der Sicherheitsanwendung für die anwenderdefinierte CRC-Berechnung drei Phasen implementiert werden.

Phase 1: CRC-Initialisierung

1. Voraussetzung für eine anwenderdefinierte CRC-Berechnung ist deren Konfiguration durch den Anwender. Wenn geplant ist, dass mehr als eine anwenderdefinierte sichere Kommunikation in der Sicherheitsanwendung verwendet wird, müssen Sie für jede geplante sichere Kommunikation zur Initialisierung ihrer CRC-Berechnungen einmal SF_CRC_INIT aufrufen.

   ⇨ Durch das Aufrufen von SF_CRC_INIT wird für ein anwenderdefiniertes CRC-Polynom (über die Eingabe POLYNOM) mit seiner CRC-Länge (Eingabe BITS) für die ausgewählte sichere Kommunikation (identifiziert über die Eingabe CRC_SLOT) die CRC-Berechnungstabelle aufgebaut.

   Nur eine Initialisierung einer ausgewählten CRC-Berechnungstabelle ist zulässig. Eine Neuinitialisierung nach der Initialisierung führt zu einem Fehler ↳ *Kapitel 4.6.6.4.1 „SF_CRC_INIT“ auf Seite 339*.

2. Weitere Konfigurationseinstellungen müssen über weitere Eingaben am Funktionsbaustein vorgenommen werden.

Phase 2: CRC-Berechnung

▷ Rufen Sie SF_CRC_INPUT auf, um einen CRC-Wert (vorher konfiguriert über SF_CRC_INIT) über einen Anwenderdatenblock für die ausgewählte sichere Kommunikation (beschrieben durch den Eingang CRC_SLOT) zu berechnen.

Sie müssen SF_CRC_INPUT für jede sichere Kommunikation separat mit dem korrekten CRC_SLOT-Eingangswert aufrufen.

⇨ Die Berechnung erfolgt in einem Verarbeitungszyklus der CPU (wobei eine große Datenmenge zu einer Verlängerung der CPU-Zykluszeit führen kann) ↳ *Kapitel 4.6.6.4.2 „SF_CRC_INPUT“ auf Seite 340*.

Phase 3: Finalisieren der CRC-Berechnung

1. Rufen Sie SF_CRC_FINISH auf, um den berechneten CRC-Wert für jede ausgewählte sichere Kommunikation (beschrieben durch den Eingang CRC_SLOT) zu erhalten.

   Sie müssen SF_CRC_FINISH für jede sichere Kommunikation separat zusammen mit dem korrekten CRC_SLOT-Eingabewert aufrufen.

   ⇨ Die Funktion gibt den vorher mit SF_CRC_INPUT berechneten CRC-Wert zurück und bereitet den nächsten CRC-Berechnungszyklus vor.

2. Im Falle der Empfangsrichtung der Sicherheitskommunikation muss der berechnete Wert, der von SF_CRC_FINISH zurückgegeben wird, gegen den empfangenen CRC-Wert validiert werden ↳ *Kapitel 4.6.6.4.3 „SF_CRC_FINISH“ auf Seite 342*.

HINWEIS!
Die Nutzung dieser Funktionen erfordert detaillierte Kenntnisse über den Umgang mit CRC-geschützten Daten in Protokollen zur sicheren Kommunikation. Ferner ist es erforderlich, die Funktionen in einer richtigen Weise aufzurufen, da nicht alle Fehlerszenarien explizit erkennbar sind. Weitere Informationen und Anweisungen zur Implementierung des Anwendungsprogramms finden Sie in ↳ *Kapitel 4.6.6.4.4 „Leitlinien zur Anwendung“ auf Seite 343*.

GEFAHR!
Bei der Berechnung des CRC-Werts für einen empfangenen Datenblock darf das Anwenderprogramm die CRC-Daten im Datenblock nicht enthalten. Das ist erforderlich, um zu verhindern, dass das CRC-Ergebnis von "0" immer berechnet wird, was zu einem unerwarteten CRC-Berechnungsergebnis von "0" führen würde. Es ist zwingend erforderlich, den berechneten CRC aus SF_CRC_FINISH gegen den empfangenen CRC-Wert zu validieren.

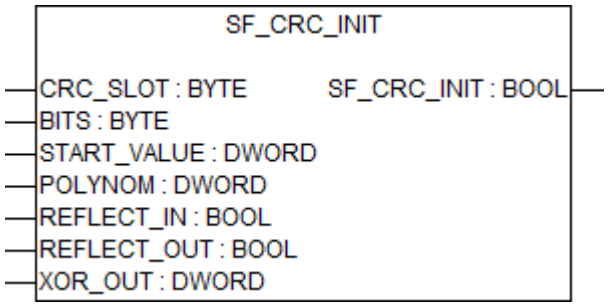
4.6.6.4.1 SF_CRC_INIT

Die Funktion SF_CRC_INIT initialisiert die CRC-Berechnungstabelle und nimmt weitere Einstellungen vor für die CRC-Berechnung der benutzten sicheren Kommunikation, die über den Eingang CRC_SLOT festgelegt wird.

Aus folgenden Gründen darf diese Funktion für jede sichere Kommunikation und den entsprechenden CRC-Slot nur einmal aufgerufen werden:

- Für eine optimierte Laufzeitberechnung werden interne Tabellen erstellt, was Verarbeitungszeit erfordert.
- Weitere Aufrufe (nach der erfolgreichen Konfiguration) geben FALSE zurück; eine Neukonfiguration wird zurückgewiesen, da, wie vorgesehen, die frühere erfolgreiche Initialisierung unverändert bleibt.

Die Funktion SF_CRC_INIT muss für jede genutzte sichere Kommunikation aufgerufen werden, die über den Eingang CRC_SLOT festgelegt wurde.



Tab. 77: FB-Name: SF_CRC_INIT

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
CRC_SLOT	BYTE	0	Identifiziert die CRC für die betreffende sichere Kommunikation mit dem entsprechenden CRC-Slotwert, für den die CRC-Initialisierung konfiguriert ist. Zulässige Werte: 0..7
BITS	BYTE	0	Definiert die zu verwendende CRC-Bitgröße (abhängig vom Grad des genutzten Polynoms). Zulässige Werte: 1 … 32
START_VALUE	DWORD	16#00000000	Definiert den Anfangswert der CRC-Berechnung. Er hängt von der Spezifikation des Protokolls für die sichere Kommunikation ab. Alle Werte sind zulässig.
POLYNOM	DWORD	16#00000000	CRC-Polynom (repräsentiert durch den Hexadezimalwert der betreffenden CRC-Gleichung). Alle Werte außer 0 sind zulässig. Der Wert 0 führt bei einem anschließenden Ausführen der Funktion SF_CRC_FINISH zu einem SAFE STOP der Sicherheits-CPU.
REFLECT_IN	BOOL	FALSE	Definiert, ob eingegebene Daten bitweise gedreht werden sollen oder nicht. FALSE: Keine bitweise Drehung TRUE: Bitweise Drehung

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
REFLECT_OUT	BOOL	FALSE	Definiert, ob der CRC-Wert bitweise gedreht werden soll oder nicht. FALSE: Keine bitweise Drehung TRUE: Bitweise Drehung
XOR_OUT	DWORD	16#00000000	Definiert den Operanden für die bitweise XOR-Verknüpfung mit dem CRC-Wert, der später am Ausgang unter Verwendung der Funktion SF_CRC_FINISH ausgegeben wird. Alle Werte sind zulässig.
VAR_OUTPUT			
SF_CRC_INIT	BOOL	TRUE	Ergebnis der Initialisierung der CRC-Berechnung für die sichere Kommunikation und dem entsprechenden CRC-Slot. TRUE: Initialisierung der CRC-Berechnung erfolgreich. FALSE: Fehler bei der Initialisierung der CRC-Berechnung. Mögliche Gründe: ● CRC_SLOT ungültig (> 7) ● BITS ungültig (nicht im Bereich 1 … 32) ● CRC_SLOT bereits erfolgreich initialisiert.

Aufruf in ST

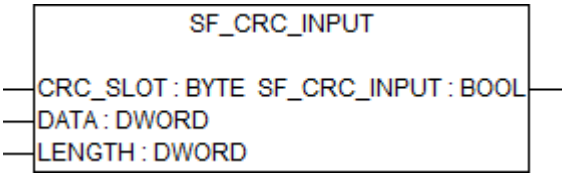
```
SF_CRC_INIT_Slot1 := SF_CRC_INIT(CRC_SLOT1,
                                 BITS_SLOT1,
                                 START_VALUE_SLOT1,
                                 POLYNOM_SLOT1,
                                 REFLECT_IN_SLOT1,
                                 REFLECT_OUT_SLOT1,
                                 XOR_OUT_SLOT1);
```

4.6.6.4.2 SF_CRC_INPUT

Die Funktion SF_CRC_INPUT führt die CRC-Berechnung über einen bestimmten benutzerdefinierten Datenblock (adressiert über Zeiger am Eingang DATA) mit einer bestimmten Länge (über den Eingang LENGTH) für die betreffende, über den Eingang CRC_SLOT festgelegte sichere Kommunikation aus. Die CRC-Berechnung wird nur auf einem Mikroprozessor vorgenommen (Nutzung der 1oo2-Sicherheitsarchitektur auf der AC500-S-Sicherheits-CPU), doch das Ergebnis der CRC-Berechnung ist auf beiden Mikroprozessoren der Sicherheits-CPU verfügbar.

Für die CRC-Berechnung sind zwei Optionen möglich:

- Berechnung in einem Verarbeitungszyklus:
  Das bedeutet, dass die Berechnung durch Setzen des Eingangs DATA auf die Basisadresse des Datenpuffers und Setzen des Eingangs LENGTH auf die Größe des gesamten Datenpuffers vorgenommen wird.
- Sequenzierte Berechnung:
  Das bedeutet, dass die Berechnung in mehrere Verarbeitungszyklen geteilt wird. Dies könnte aus bestimmten Gründen im Zusammenhang mit dem Protokoll der sicheren Kommunikation erforderlich sein. Gestartet wird die Sequenz durch Setzen des Eingangs DATA auf die Basisadresse des Datenpuffers und Setzen des Eingangs LENGTH auf einen Teil der Größe des Datenpuffers (den ersten Teil des gesamten Puffers). Die nächste Sequenz wird durch Setzen der Eingangs DATA auf [Basisadresse des Datenpuffers + Länge in der vorhergehenden Sequenz] und Setzen des Eingangs LENGTH auf einen nächsten Teil der Größe des Datenpuffers vorgenommen. Infolgedessen kann die CRC-Berechnung Byte für Byte sequenziert werden. Wichtig ist, dass nach dem Ausführen der gesamten CRC-Berechnungssequenz über den gesamten Datenblock die Funktion SF_CRC_FINISH nur einmal aufgerufen wird.



Tab. 78: FB-Name: SF_CRC_INPUT

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
CRC_SLOT	BYTE	0	Identifiziert den CRC-Slot der sicheren Kommunikation, für welche die CRC-Berechnung ausgeführt wird. Zulässige Werte: 0...7
DATA	DWORD	16#00000000	Speicherstartadresse als Zeiger (über ADR-Operator) des Datenblocks, für den die CRC berechnet wird. Zulässige Werte: Muss innerhalb des Benutzerspeicherbereichs (in Kombination mit der Eingabe LENGTH) liegen
LENGTH	WORD	16#0000	Länge des Datenblocks (basierend auf dem Eingang DATA), für den die CRC berechnet wird. Zulässige Werte: Muss innerhalb des Benutzerspeicherbereichs (in Kombination mit dem Eingang DATA) liegen
VAR_OUTPUT			
SF_CRC_INPUT	BOOL	FALSE	Ergebnis der Funktion SF_CRC_INPUT. TRUE: CRC-Berechnung erfolgreich. FALSE: Fehler in der CRC-Berechnung, mögliche Gründe: ● CRC_SLOT ungültig (> 7) ● CRC-Polynom ungültig („0“) ● DATA und/oder LENGTH ungültig (Datenpuffer außerhalb des zulässigen Benutzerspeicherbereichs) ● Ausgewählter CRC_SLOT nicht erfolgreich initialisiert.

```
SF_CRC_INPUT_Slot1 := SF_CRC_INPUT(CRC_SLOT1,
                                   ADR(DATA_SLOT1),
                                   LENGTH_SLOT1);
```

4.6.6.4.3 SF_CRC_FINISH

Die Funktion SF_CRC_FINISH gibt den berechneten CRC-Wert zurück und initialisiert die CRC-Berechnung für die ausgewählte sichere Kommunikation, die über den Eingang CRC_SLOT festgelegt wurde, neu.

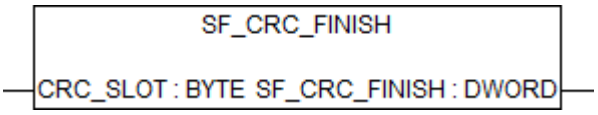
HINWEIS!

SF_CRC_FINISH darf nach der CRC-Berechnung mit SF_CRC_INPUT und **vor** dem Starten eines neuen CRC-Berechnungszyklus **nur einmal aufgerufen werden**.

Folgen eines Nichtaufrufens der Funktion SF_CRC_FINISH (nach der aktuellen CRC-Berechnung und vor der nächsten CRC-Berechnung):

- Der neu berechnete CRC-Wert ist in der Sicherheitsanwendung nicht verfügbar.
- Die erforderliche Neuinitialisierung für die nächste CRC-Berechnung fehlt, weshalb beim nächsten Aufrufen der Funktion SF_CRC_INPUT ein unerwartetes Ergebnis zurückgegeben wird.

Wenn Funktion SF_CRC_FINISH (nach der aktuellen CRC-Berechnung und vor der nächsten CRC-Berechnung) mehr als einmal aufgerufen wird, gibt nur das erste Aufrufen den gültigen berechneten CRC-Wert zurück. Durch folgende Aufrufe werden ungültige CRC-Werte zurückgegeben.



Tab. 79: FB-Name: SF_CRC_FINISH

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
CRC_SLOT	BYTE	0	Identifiziert den CRC-Slot der sicheren Kommunikation, für welche der CRC-Wert zurückgegeben wird. Zulässige Werte: 0...7
VAR_OUTPUT			
SF_CRC_FINISH	DWORD	16#00000000	Ergebnis der CRC-Berechnung

HINWEIS!

Unter den folgenden Fehlerbedingungen leitet SF_CRC_FINISH zum Schutz vor nicht behebbaren Fehlersituationen, die möglicherweise durch eine fehlerhafte Anwendung verursacht werden (nicht am Rückgabewert der Funktion ablesbar, da der berechnete CRC-Wert keinerlei Wertbeschränkungen unterliegt) einen SAFE STOP ein.

- CRC_SLOT ungültig (> 7)
- CRC_SLOT mit CRC-Polynom „0“ konfiguriert
- CRC_SLOT überhaupt nicht konfiguriert oder nicht erfolgreich konfiguriert

Aufruf in ST

```
SF_CRC_FINISH_Slot1 := SF_CRC_FINISH(CRC_SLOT1);
```

4.6.6.4.4 Leitlinien zur Anwendung

Zur Vermeidung des Risikos ungültiger CRC-Werte oder eines unerwünschten SAFE STOP der Sicherheits-CPU empfehlen wir die Einhaltung der folgenden Leitlinien.

Leitlinie zur Vorbereitung

- Analysieren Sie die Spezifikation des Protokolls für die sichere Kommunikation, die Sie realisieren möchten. Definieren Sie die zur Konfiguration Ihrer CRC-Funktionalität erforderlichen Konfigurationswerte. Dies betrifft alle Eingangswerte der Funktion SF_CRC_INIT.

Leitlinie zur Implementierung

- Stellen Sie sicher, dass Sie SF_CRC_INIT **nur einmal** und mit einem von 0 verschiedenen Polynom aufrufen.
  Nur wenn die Funktion SF_CRC_INIT **TRUE** zurückgibt, erlauben Sie weitere Aufrufe der Funktionen SF_CRC_INPUT und SF_CRC_FINSH.
- Stellen Sie sicher, dass SF_CRC_INPUT/SF_CRC_FINISH mit der korrekten Eingabe CRC_SLOT aufrufen.
- Stellen Sie sicher, dass Sie SF_CRC_INPUT im Rahmen der zulässigen Speicherkapazität der Sicherheitsanwendung aufrufen, beispielsweise unter Verwendung der Funktionen ADR und SIZEOF.
- Stellen Sie sicher, dass Sie SF_CRC_FINISH **exakt einmal** nach Abschluss der CRC-Berechnung für den betreffenden Benutzerdatenblock und vor dem nächsten CRC-Berechnungszyklus aufrufen.
- Stellen Sie sicher, dass Sie immer den empfangenen CRC-Wert aus der CRC-Berechnung ausschließen, und vergleichen Sie den empfangenen CRC-Wert mit dem berechneten CRC-Wert (Ausgabe der Funktion SF_CRC_FINISH). Bei einer Abweichung weisen Sie die empfangenen Daten zurück. Akzeptieren Sie nur Daten mit erfolgreicher CRC-Validierung.

4.6.7 SafetyExt_AC500_V22.lib

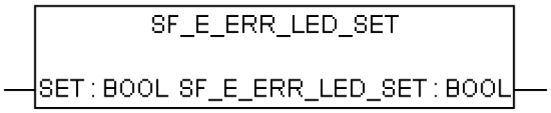
Die Bibliothek SafetyExt_AC500_V22.lib enthält die folgenden POEs:

- Systembefehle
  - SF_E_ERR_LED_SET (setzt die LED E-ERR (EIN oder AUS))
  - SF_MAX_POWER_DIP_SET (setzt die max. Anzahl Neustarts nach Spannungsabfall in der Sicherheits-CPU)
  - SF_WDOG_TIME_SET (setzt die maximal zulässige Zykluszeit der Sicherheits-CPU)
  - SF_APPL_MEASURE_BEGIN (diese Funktion definiert den Startpunkt der Zeitprofilerstellung)
  - SF_APPL_MEASURE_END (diese Funktion definiert den Endpunkt der Zeitprofilerstellung)
- Systeminformationen
  - SF_MAX_POWER_DIP_GET (gibt die aktuelle Anzahl von Neustarts nach Spannungsabfall in der Sicherheits-CPU an)
  - SF_SAFETY_MODE (gibt an, ob die Sicherheits-CPU im DEBUG- oder SAFETY-Modus ist)
  - SF_SM5XX_OWN_ADR (gibt den Wert der Drehschalter-Adresse der Sicherheits-CPU an)
  - SF_RTS_INFO (Gibt die Firmwareversion der Sicherheits-CPU an. Die Version ist eine Dezimale im Binärcode, 16#10 bedeutet z. B. Version 1.0.)

- Datenspeicherung
  - SF_FLASH_DEL (Dieser Funktionsbaustein löscht ein Datensegment im Flash-Speicher. Sämtliche Daten in diesem Datensegment werden gelöscht.)
  - SF_FLASH_READ (Der Funktionsbaustein liest einen Datensatz aus einem Datensegment des Flash-Speichers und legt diesen Datensatz ab dem durch die Sicherheits-CPU definierten Anfangsmerker ab.)
  - SF_FLASH_WRITE (Dieser Funktionsbaustein schreibt Daten in ein Datensegment im Flash-Speicher.)
- Azyklischer nicht sicherer Datenaustausch
  - SF_DPRAM_PM5XX_S_REC (Empfang von Daten der Standard-CPU)
  - SF_DPRAM_PM5XX_S_SEND (Versand von Daten an die Standard-CPU)

HINWEIS!
Um einen azyklischen nicht sicheren Datenaustausch zwischen Sicherheits- und Standard-CPU herzustellen, ist die Verwendung dedizierter Funktionsbausteine für die Standard-CPU erforderlich *Anhang B.5.1 „Azyklischer nicht sicherer Datenaustausch“ auf Seite 431* *Anhang C.5.1 „Azyklischer nicht sicherer Datenaustausch“ auf Seite 449*.

4.6.7.1 SF_E_ERR_LED_SET



Setzt den Zustand der LED E-ERR (EIN = TRUE oder AUS = FALSE)

Die LED E-ERR wird direkt im Zyklus der Sicherheits-CPU gesetzt. Der Zustand bleibt unverändert, bis er explizit mit Aufrufen von SF_E_ERR_LED_SET geändert wird.

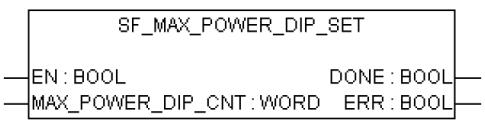
Tab. 80: FUN-Name: SF_E_ERR_LED_SET

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
SET	BOOL	FALSE	FALSE = LED E-ERR ist AUS, TRUE = LED E-ERR ist EIN
VAR_OUTPUT			
SF_E_ERR_LED_SET	BOOL	FALSE	FALSE = LED E-ERR ist AUS, TRUE = LED E-ERR ist EIN

Aufruf in ST

```
SF_E_ERR_LED_SET_Value := SF_E_ERR_LED_SET(SF_E_ERR_LED_SET_Set);
```

4.6.7.2 SF_MAX_POWER_DIP_SET



Legt die max. Anzahl von Spannungsabfällen in der Sicherheits-CPU SM560-S fest

Der Funktionsbaustein SF_MAX_POWER_DIP_SET ermöglicht den Anwendern das Einstellen des Neustartverhaltens der Sicherheits-CPU nach Phasen der Unterbrechung der Stromversorgung seitens der Standard-CPU von weniger als 1,5 s („Spannungseinbruch“). Um die wiederholte Erkennung eines Spannungseinbruchs an der Sicherheits-CPU zu vermeiden, stellen Sie sicher, dass die Phase der Stromunterbrechung des Power Cycle mindestens 1,5 s andauert, ehe die Stromversorgung wiederhergestellt wird.

Für einen erfolgreichen Neustart der Sicherheits-CPU im Modus RUN (Sicherheitsmodus) nach der Erkennung eines Spannungseinbruchs müssen Sie entsprechend dem Neustartverfahren vorgehen. Zu Vermeidung eines unkontrollierten Verhaltens nach einem Spannungseinbruch sind ein oder zwei Power Cycles erforderlich.

Ohne Verwendung des Funktionsbausteins SF_MAX_POWER_DIP_SET müssen nach einem Spannungseinbruch zwei Power Cycles (oder ein Neustartbefehl) ausgeführt werden.

Alternativ können Sie die Neustartsteuerung auch mit dem Funktionsbaustein SF_MAX_POWER_DIP_SET konfigurieren. Definieren Sie eine Anzahl tolerierter Spannungseinbrüche durch Eingabe von MAX_POWER_DIP_CNT. Für die definierte Anzahl von Spannungseinbrüchen wird ein Neustart mit nur einem Power Cycle (oder Neustartbefehl) akzeptiert.

Die Anzahl der Spannungseinbrüche wird in der Sicherheits-CPU gezählt (die aktuelle Anzahl kann über den Funktionsbaustein SF_MAX_POWER_DIP_GET *Kapitel 4.6.7.6 „SF_MAX_POWER_DIP_GET“ auf Seite 349* abgefragt werden) und mit der Zahl, die vor dem Starten des Sicherheitsprogramms verfügbar ist (die konfigurierte Anzahl kann über den Funktionsbaustein SF_MAX_POWER_DIP_GET_CFG *Kapitel 4.6.6.2 „SF_MAX_POWER_DIP_GET_CFG“ auf Seite 336* aufgerufen werden), verglichen. Sofern die gezählte Anzahl nicht höher als die konfigurierte Anzahl ist, ist für das Neustarten der Sicherheits-CPU nur ein Power Cycle (oder Neustartbefehl) erforderlich. Falls die gezählte Anzahl höher als der konfigurierte Wert ist, sind für das Neustarten der Sicherheits-CPU zwei Power Cycles (oder Neustartbefehle) erforderlich. Der Zähler kann durch erneutes Aufrufen des Funktionsbausteins SF_MAX_POWER_DIP_SET zurückgesetzt werden.

Es darf nur eine Funktionsbausteininstanz im Sicherheitsprogramm verwendet werden; anderenfalls wird eine Warnung angezeigt.

HINWEIS!

Bei jedem Aufruf des Funktionsbausteins SF_MAX_POWER_DIP_SET mit EN-Übergang von FALSE zu TRUE wird der interne Spannungsabfall-Zähler zurückgesetzt; d. h. er beginnt wieder mit 0. Deshalb ist es sinnvoll, den Funktionsbaustein SF_MAX_POWER_DIP_SET nur einmal im Sicherheitsprogramm mit EN-Übergang von FALSE zu TRUE als Einzelparametrierung der Spannungsabfall-Funktionalität aufzurufen.

Wenn Sie dieser Empfehlung nicht folgen, wird der Zählerwert für Neustarts nach Spannungseinbrüchen in der Sicherheits-CPU für den Funktionsbaustein SF_MAX_POWER_DIP_GET bei jedem Aufruf des Funktionsbausteins SF_MAX_POWER_DIP_SET mit EN-Übergang von FALSE zu TRUE durch das sicherheitsgerichtete Anwendungsprogramm auf „0“ zurückgesetzt.

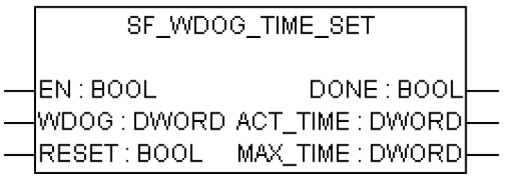
Tab. 81: FB-Name: SF_MAX_POWER_DIP_SET

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Der Baustein wird zum Speichern des Wertes MAX_POWER_DIP_CNT im Flash-Speicher durch Übergang des Eingangs EN von FALSE zu TRUE aktiviert. Der Baustein bleibt aktiv und ignoriert Veränderungen am Eingang EN, bis der Ausgang DONE auch TRUE ist. Der Wert MAX_POWER_DIP_CNT kann nur im Flash-Speicher gespeichert werden, wenn der Übergang von FALSE zu TRUE am Eingang EN ausgelöst wird.
MAX_POWER_DIP_CNT	WORD	16#0000	Maximale Anzahl tolerierter Neustarts der Sicherheits-CPU mit nur einem Power Cycle (oder Neustartbefehl) nach Fehlern durch Spannungseinbruch.
VAR_OUTPUT			
DONE	BOOL	FALSE	Am Ausgang DONE wird angezeigt, dass der Set-Vorgang abgeschlossen ist (siehe auch Ausgang ERR).
ERR	BOOL	FALSE	Bei TRUE ist ein Fehler während des Set-Vorgangs aufgetreten (Speichern des Wertes MAX_POWER_DIP_CNT in den Flash-Speicher).

Aufruf in ST

```
SF_MAX_POWER_DIP_SET (EN := SF_MAX_POWER_DIP_SET_EN,
MAX_POWER_DIP_CNT := SF_MAX_POWER_DIP_SET_MAX_POWER_DIP_CNT,
DONE => SF_MAX_POWER_DIP_SET_DONE, ERR => SF_MAX_POWER_DIP_SET_ERR);
```

4.6.7.3 SF_WDOG_TIME_SET



Setzt die maximal zulässige Zykluszeit der Sicherheits-CPU

Mit dem Funktionsbaustein SF_WDOG_TIME_SET hat der Anwender die Möglichkeit, die Zykluszeit zu überwachen. Der Funktionsbaustein muss vom Anwender im ersten Zyklus aufgerufen werden. Zur Aktualisierung der Ausgänge ACT_TIME und MAX_TIME muss der Funktionsbaustein in jedem Zyklus aufgerufen werden. Ist der Funktionsbaustein in der Anwendung nicht vorhanden, wechseln die Sicherheits-CPU und das Anwendungsprogramm nach dem ersten Zyklus in den Zustand SAFE STOP. Die Watchdog-Zeit wird vor der Ausgabe der PROFIsafe-Telegramme überwacht.

Bei einer Überschreitung der Zykluszeit wird eine Fehlermeldung ausgegeben und die Sicherheits-CPU geht in den Zustand SAFE STOP. Sinnvolle Werte sind größer als die typische Laufzeit der Sicherheits-CPU und halb so groß wie oder kleiner als F_WD_Time der Sicherheits-E/A-Module.

Es darf nur eine Funktionsbausteininstanz im Sicherheitsprogramm verwendet werden; anderenfalls wird eine Warnung angezeigt.

HINWEIS!
Die Überwachung der Zyklusdauer erfolgt nur im Modus RUN (sicher).

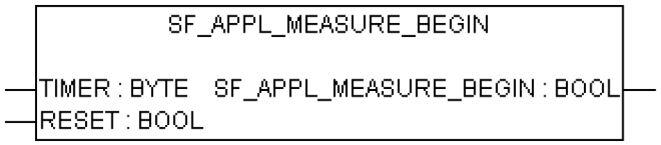
Tab. 82: FB-Name: SF_WDOG_TIME_SET

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Der Funktionsbaustein wird über den Eingang EN aktiviert (EN = TRUE) oder deaktiviert (EN = FALSE). Wenn der Baustein aktiv ist, sind die aktuellen Werte an den Ausgängen verfügbar.
WDOG	DWORD	16#00000000	Watchdog-Zeit in ms. Der max. zulässige Wert ist 1000. Bei WDOG > 1000 geht die Sicherheits-CPU in den Zustand SAFE STOP.
RESET	BOOL	FALSE	TRUE setzt MAX_TIME auf 0.
VAR_OUTPUT			
DONE	BOOL	FALSE	Am Ausgang DONE wird angezeigt, dass der Set-Vorgang abgeschlossen ist.
ACT_TIME	DWORD	16#00000000	Tatsächliche Zyklusdauer der Sicherheits-CPU in ms
MAX_TIME	DWORD	16#00000000	Max. überwachte Zyklusdauer der Sicherheits-CPU in ms

Aufruf in ST

```
SF_WDOG_TIME_SET (EN := SF_WDOG_TIME_SET_EN,
WDOG := SF_WDOG_TIME_SET_WDOG,
RESET := SF_WDOG_TIME_SET_RESET,
DONE => SF_WDOG_TIME_SET_DONE,
ACT_TIME => SF_WDOG_TIME_SET,
MAX_TIME => SF_WDOG_TIME_SET_MAX_TIME);
```

4.6.7.4 SF_APPL_MEASURE_BEGIN



Definiert den Startpunkt von Zeitprofilen

Diese Funktion definiert den Startpunkt der Zeitprofilerstellung im Sicherheitsprogramm und ist zusammen mit der Funktion SF_APPL_MEASURE_END zu verwenden. Die Ergebnisse der Zeitprofilerstellung können nur mit dem SPS-Browser-Befehl „applinfo“ angezeigt und nicht im Sicherheitsprogramm verwendet werden.

Die Zeit zwischen dem Aufrufen von SF_APPL_MEASURE_BEGIN und SF_APPL_MEASURE_END im Sicherheitsprogramm wird gemessen (auch innerhalb eines Sicherheits-CPU-Zyklus) und in dem durch den Eingangsparameter TIMER identifizierten Timer gespeichert.

HINWEIS!
Die Funktion SF_APPL_MEASURE_BEGIN wurde nur für das Messen kurzer Zeitintervalle entwickelt; bei Zeitintervallen von ca. 10 Minuten sind die Ergebnisse ungültig.

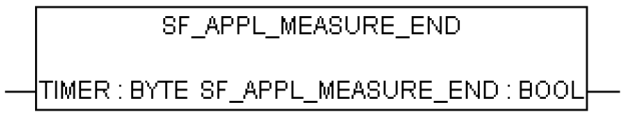
Tab. 83: FUN-Name: SF_APPL_MEASURE_BEGIN

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
TIMER	BYTE	16#00	Timer-Identifikation. Der zulässige Bereich reicht von 0 bis 31.
RESET	BOOL	FALSE	Bei TRUE werden die MAX und MIN Ergebnisse der Zeitprofilerstellung gelöscht. Anderenfalls werden die beobachteten Werte beibehalten.
VAR_OUTPUT			
SF_APPL_MEASURE_BEGIN	BOOL	FALSE	Der zurückgegebene Wert ist TRUE, wenn der TIMER-Wert im zulässigen Bereich (0 … 31) liegt; anderenfalls FALSE.

Aufruf in ST

```
SF_APPL_MEASURE_BEGIN_VALUE :=
SF_APPL_MEASURE_BEGIN(SF_APPL_MEASURE_BEGIN_TIMER,
SF_APPL_MEASURE_BEGIN_RESET);
...
...
SF_APPL_MEASURE_END_VALUE :=
SF_APPL_MEASURE_END(SF_APPL_MEASURE_END_TIMER);
```

4.6.7.5 SF_APPL_MEASURE_END



Definiert den Endpunkt von Zeitprofilen

Diese Funktion definiert den Endpunkt der Zeitprofilerstellung im Sicherheitsprogramm und ist zusammen mit der Funktion SF_APPL_MEASURE_BEGIN zu verwenden. Die Ergebnisse der Zeitprofilerstellung können nur mit dem SPS-Browserbefehl „applinfo“ angezeigt und nicht im Sicherheitsprogramm verwendet werden.

Die Zeit zwischen dem Aufrufen von SF_APPL_MEASURE_BEGIN und SF_APPL_MEASURE_END im Sicherheitsprogramm wird gemessen und in dem durch den Eingangsparameter TIMER identifizierten Timer gespeichert.

HINWEIS!
Die Funktion SF_APPL_MEASURE_END wurde nur für das Messen kurzer Zeitintervalle entwickelt; bei Zeitintervallen von ca. 10 Minuten sind die Ergebnisse ungültig.

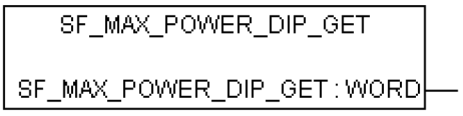
Tab. 84: FUN-Name: SF_APPL_MEASURE_END

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
TIMER	BYTE	16#00	Timer-Identifikation. Der zulässige Bereich reicht von 0 bis 31.
VAR_OUTPUT			
SF_APPL_MEASURE_END	BOOL	FALSE	Der zurückgegebene Wert ist TRUE, wenn der TIMER-Wert im zulässigen Bereich (0 … 31) liegt; anderenfalls FALSE.

Aufruf in ST

```
SF_APPL_MEASURE_BEGIN_VALUE :=
SF_APPL_MEASURE_BEGIN(SF_APPL_MEASURE_BEGIN_TIMER,
SF_APPL_MEASURE_BEGIN_RESET);
...
...
SF_APPL_MEASURE_END_VALUE :=
SF_APPL_MEASURE_END(SF_APPL_MEASURE_END_TIMER);
```

4.6.7.6 SF_MAX_POWER_DIP_GET



Gibt die aktuelle Anzahl von Neustarts nach einem Spannungsabfall in der Sicherheits-CPU an

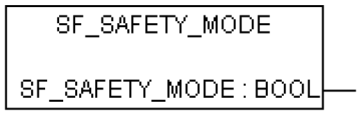
Tab. 85: FUN-Name: SF_MAX_POWER_DIP_GET

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_OUTPUT			
SF_MAX_POWER_DIP_GET	WORD	16#0000	Istwert des Zählers für Spannungsabfallfehler.

Aufruf in ST

```
SF_MAX_POWER_DIP_GET_Value := SF_MAX_POWER_DIP_GET();
```

4.6.7.7 SF_SAFETY_MODE



Ausgelesener Wert gibt an, ob sich die Sicherheits-CPU im Modus DEBUG RUN (nicht sicher), DEBUG STOP (nicht sicher) oder im Modus RUN (Sicherheitsmodus) befindet.

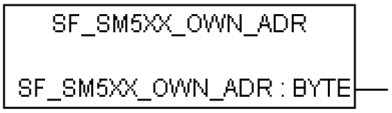
Tab. 86: FUN-Name: SF_SAFETY_MODE

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_OUTPUT			
SF_SAFETY_MODE	BOOL	FALSE	Modus der Sicherheits-CPU: ● FALSE: Der Modus DEBUG RUN (nicht sicher) oder DEBUG STOP (nicht sicher) ist aktiv. ● TRUE: Der Modus RUN (Sicherheitsmodus) ist aktiv.

Aufruf in ST

```
SF_SAFETY_MODE_Value := SF_SAFETY_MODE();
```

4.6.7.8 SF_SM5XX_OWN_ADR



Gibt den Wert der Drehschalter-Adresse der Sicherheits-CPU an

Nur der während des Starts der Sicherheits-CPU SM560-S gesetzte Wert wird gelesen. Weitere Änderungen der Drehschalter-Adresse werden ignoriert.

> **HINWEIS!**
> Ungeachtet der Tatsache, dass die SF_SM5XX_OWN_ADR-Funktion eine Sicherheits-POE ist, ist der Adresswert des Hardwareschalters ein nicht sicherer Wert und es sind zusätzliche Maßnahmen nötig, um die funktionellen sicherheitsbezogenen Anforderungen zu erfüllen.

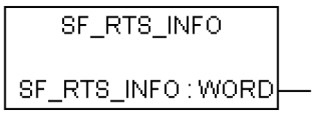
Tab. 87: FUN-Name: SF_SM5XX_OWN_ADR

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_OUTPUT			
SF_SM5XX_OWN_ADR	BYTE	16#00	Wert der Drehschalter-Adresse der Sicherheits-CPU, der während des Starts gesetzt war.

Aufruf in ST

```
SF_SM5XX_OWN_ADR_Value := SF_SM5XX_OWN_ADR();
```

4.6.7.9 SF_RTS_INFO



Anzeige der Firmwareversion der Sicherheits-CPU

Diese Funktion gibt die Firmwareversion der Sicherheits-CPU an. Die Version ist eine Dezimale im Binärcode, 16#10 bedeutet z. B. Version 1.0.

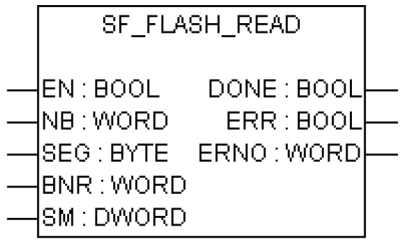
Tab. 88: FUN-Name: SF_RTS_INFO

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_OUTPUT			
SF_RTS_INFO	WORD	16#0000	Firmwareversion der Sicherheits-CPU. Das obere BYTE des Eintrags stellt die Hauptversion, das untere BYTE die Unterversion des Laufzeitsystems dar. Beispiel: RTS_VERSION = 16#0110 ➔ V01.1.0

Aufruf in ST

```
SF_RTS_INFO_Value := SF_RTS_INFO();
```

4.6.7.10 SF_FLASH_READ



Lesen der Nutzerdaten aus dem Flash-Speicher

Der Funktionsbaustein liest einen Datensatz aus einem Datensegment im Flash-Speicher und legt diesen Datensatz ab dem am Eingang SM projektierten Anfangsmerker ab. Die Daten des Datensatzes wurden durch den Funktionsbaustein SF_FLASH_WRITE im Flash-Speicher abgelegt.

HINWEIS!

Der Zugriff auf den Flash-Speicher ist nur mit den Funktionsbausteinen SF_FLASH_WRITE, SF_FLASH_DEL und SF_FLASH_READ möglich.

Es werden NB-Bausteine ab dem Baustein BNR im Segment SEG gelesen und ab der Adresse SM abgelegt.

Pro Baustein werden 32 Binär-Daten oder 16 Wort-Daten oder 8 Doppelwort-Daten gelesen.

Ein Baustein enthält 38 Bytes:

– 32 Bytes Daten
– 4 Bytes für CRC-Prüfsumme
– 1 Byte als „beschrieben“-Kennung
– 1 Byte für Ausrichtung

↳ *Tab. 90 „Struktur eines Segments mit Nutzerdaten im Flash-Speicher“ auf Seite 354*

Mit einer FALSE/TRUE-Flanke am Eingang EN wird der einmalige Lesevorgang eines Datensatzes ausgelöst. Wenn beim Lesen der Daten kein Fehler aufgetreten ist, wird der Ausgang DONE auf TRUE und die Ausgänge ERR und ERNO auf FALSE gesetzt. Der Datensatz wird ab dem projektierten Anfangsmerker SM abgelegt.

Die Ablage des Datensatzes kann mehrere CPU-Zyklen dauern.

Tritt beim Lesen ein Fehler auf, werden DONE und ERR auf TRUE gesetzt und die Daten ab SM sind gleich 0. Die Art des Fehlers wird am Ausgang ERNO signalisiert.

HINWEIS!

Dieser Funktionsbaustein wird durch eine positive Flanke der Eingangsvariable EN aktiviert. Während des Zyklus, in dem der Funktionsbaustein feststellt, dass die Operation abgeschlossen ist (Ausgang DONE = TRUE), setzt er die Ausgangsvariablen nur für einen Zyklus. Wenn der Funktionsbaustein erneut aufgerufen wird, setzt er die Ausgangsvariablen sofort zurück.

Tab. 89: FB-Name: SF_FLASH_READ

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Aktivierung des Funktionsbausteins mit positiver Flanke Es gilt: ● EN = FALSE/TRUE-Flanke: Es wird der einmalige Lesevorgang des Datensatzes durchgeführt. ● EN = TRUE: Der Funktionsbaustein wird nicht verarbeitet, d. h. er verändert seine Ausgänge nicht mehr.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
NB	WORD	16#0000	Anzahl der Datensatz-Bausteine (Dezimal 1 … 1724) Am Eingang NB wird die Anzahl der Bausteine des Datensatzes angegeben. Pro Baustein werden 32 Byte-Daten oder 16 Wort-Daten oder 8 Doppelwort-Daten gelesen. Gültige Werte: 1 … 1724 Beispiel: ● SM = ADR(%MW0.0) und NB = 1: Ablage der Daten von %MW0.0 bis %MW0.15 (1 Baustein = 16 Wort-Daten) ● SM = ADR(%MW0.0) und NB = 2: Ablage der Daten von %MW0.0 bis %MW0.31 (2 Bausteine = 32 Wort-Daten)
SEG	BYTE	16#00	ID-Nummer des Datensegments (16#01 oder 16#02)
BNR	WORD	16#0000	Nummer des Startbausteins im Datensegment im Flash-Speicher (Dezimal 0 … 1723)
SM	DWORD	16#00000000	Zieladresse für den gelesenen Datensatz (Adresse der ersten Variable, ab der die Daten abgelegt sind)
VAR_OUTPUT			
DONE	BOOL	FALSE	Der Lesevorgang ist abgeschlossen (DONE = TRUE) Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Lesevorgang abgeschlossen. Der Datensatz wurde ab dem projektierten Eingang SM abgelegt. ● DONE = TRUE und ERR = TRUE: Beim Lesevorgang ist ein Fehler aufgetreten. Der Ausgang ERNO signalisiert die Fehlernummer.
ERR	BOOL	FALSE	Fehler aufgetreten (Datensegment konnte nicht gelesen werden) Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Ist ein Fehler aufgetreten, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer *[3]* Am Ausgang ERNO wird eine Fehlernummer ausgegeben. Dieser Ausgang muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Da das Anwenderprogramm der Sicherheits-CPU mit höherer Priorität ausgeführt wird, kann SF_FLASH_READ relativ viel Zeit beanspruchen. Am Ausgang ERNO wird angezeigt, dass der Funktionsbaustein die Verarbeitung angestoßen hat (0x0FFF = BUSY). Während dieser Phase sind ERR=FALSE und DONE=FALSE.

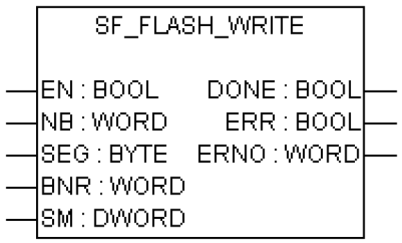
Tab. 90: Struktur eines Segments mit Nutzerdaten im Flash-Speicher

Byte:		**1 | 2**	**3 | 4**	**5 | 6**	**…**	**29 | 30**	**31 | 32**	**33 … 36**	**37**	**38**
Byte-offset	**Bau-stein-Nr.**	**Wort 1**	**Wort 2**	**Wort 3**	**…**	**Wort 15**	**Wort 16**	**CRC**	**Beschri eben-Ken-nung**	**Aus-rich-tung**
0	0									
38	1									
76	2									
…	…									
65436	1722									
65474	1723									

Aufruf in ST

```
READ_FLASH(EN := EN_FLASH_READ,
NB := NB_FLASH_READ,
SEG := SEG_FLASH_READ,
BNR := BNR_FLASH_READ,
SM := SM_FLASH_READ,
DONE => DONE_FLASH_READ,
ERR => ERR_FLASH_READ,
ERNO => ERNO_FLASH_READ);
```

4.6.7.11 SF_FLASH_WRITE



Schreiben von Nutzerdaten in den Flash-Speicher

Der Funktionsbaustein schreibt einen Datensatz in ein Datensegment im Flash-Speicher. Dazu stehen in der Sicherheits-CPU zwei Datensegmente zur Verfügung. Ein Löschvorgang (Funktionsbaustein SF_FLASH_DEL) löscht immer ein komplettes Datensegment. Ein Datensegment besteht aus 1724 Bausteinen (0 … 1723). Jeder Baustein besteht aus 38 Bytes. Die Anzahl der Schreibzyklen auf den Flash-Speicher ist begrenzt. Das Löschen vom Flash-Speicher ist auch ein „Schreiben“-Vorgang.

Nach einem Löschvorgang kann jeder dieser 1724 Bausteine eines Datensegments nur einmal Daten aufnehmen. Soll ein Baustein, der Daten enthält, mit neuen Daten überschrieben werden, muss das gesamte Datensegment vorher gelöscht werden. Dadurch gehen alle Daten in diesem Datensegment verloren.

Es werden NB-Bausteine ab der Adresse SM gelesen und im Segment SEG ab Baustein BNR abgelegt.

Pro Baustein werden 32 Binär-Daten oder 16 Wort-Daten oder 8 Doppelwort-Daten gelesen.

Ein Baustein enthält 38 Bytes:

- 32 Bytes Daten
- 4 Bytes für CRC-Prüfsumme

- 1 Byte als „beschrieben“-Kennung
- 1 Byte für Ausrichtung

Tab. 90 „Struktur eines Segments mit Nutzerdaten im Flash-Speicher“ auf Seite 354

Wird der Schreibvorgang eines Datensatzes gestartet (FALSE/TRUE-Flanke am Eingang EN), dann dürfen die Daten des Datensatzes bis zur Beendigung des Schreibvorgangs (DONE = TRUE) nicht mehr verändert werden. Die Ablage des Datensatzes im Flash-Speicher kann mehrere Zyklen der Sicherheits-CPU dauern.

Mit einer FALSE/TRUE-Flanke am Eingang EN wird der einmalige Schreibvorgang des Datensatzes ausgelöst. Bis zur Beendigung der Ablage (DONE = TRUE) wird der Eingang EN nicht mehr ausgewertet.

Nach Beendigung des Schreibvorgangs werden die Funktionsbaustein-Ausgänge DONE, ERR und ERNO aktualisiert. Bei DONE = TRUE und ERR = FALSE war die Sicherung erfolgreich. Sind DONE = TRUE und ERR = TRUE, ist ein Fehler aufgetreten. Die Art des Fehlers wird am Ausgang ERNO signalisiert.

Eine erneute FALSE/TRUE-Flanke am Eingang EN startet einen neuen Schreibvorgang. Da ohne vorheriges Löschen des Datensegments keine neuen Daten in Bausteine, die bereits Daten enthalten, geschrieben werden können, muss beim nächsten Schreibvorgang der Eingang BNR auf den nächsten freien Baustein zeigen.

HINWEIS!
Dieser Funktionsbaustein wird durch eine positive Flanke der Eingangsvariable EN aktiviert. Während des Zyklus, in dem der Funktionsbaustein feststellt, dass die Operation abgeschlossen ist (Ausgang DONE = TRUE), setzt er die Ausgangsvariablen nur für einen Zyklus. Wenn der Funktionsbaustein erneut aufgerufen wird, setzt er die Ausgangsvariablen sofort zurück.

Tab. 91: FB-Name: SF_FLASH_WRITE

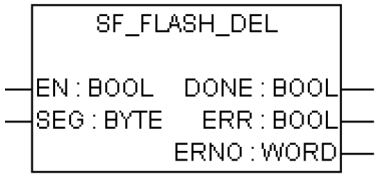
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Aktivierung des Funktionsbausteins mit positiver Flanke Es gilt: ● EN = FALSE/TRUE-Flanke: Es wird der einmalige Lesevorgang des Datensatzes durchgeführt. ● EN = TRUE: Der Funktionsbaustein wird nicht verarbeitet, d. h. er verändert seine Ausgänge nicht mehr.
NB	WORD	16#0000	Anzahl der Datensatz-Bausteine (Dezimal 1 … 1724) Am Eingang NB wird die Anzahl der Bausteine des Datensatzes angegeben. Pro Baustein werden 32 Byte-Daten oder 16 Wort-Daten oder 8 Doppelwort-Daten gelesen. Gültige Werte: 1 … 1724 Beispiel: – SM = ADR(%MW0.0) und NB = 1: Ablage der Daten von %MW0.0 bis %MW0.15 (1 Baustein = 16 Wort-Daten) – SM = ADR(%MW0.0) und NB = 2: Ablage der Daten von %MW0.0 bis %MW0.31 (2 Bausteine = 32 Wort-Daten)
SEG	BYTE	16#00	ID-Nummer des Datensegments (16#01 oder 16#02)

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
BNR	WORD	16#0000	Nummer des Startbausteins im Datensegment im Flash-Speicher (Dezimal 0 … 1723)
SM	DWORD	16#00000000	Startadresse (Adresse der ersten Variable, ab der die Daten in den Flash-Speicher geschrieben werden) Am Eingang SM wird mittels ADR-Operator die Adresse der ersten Variable des Datensatzes angegeben. Wird der Schreibvorgang eines Datensatzes gestartet (FALSE/TRUE-Flanke am Eingang EN), dann dürfen die Daten des Datensatzes bis zur Beendigung des Schreibvorgangs (DONE = TRUE) nicht mehr verändert werden.
VAR_OUTPUT			
DONE	BOOL	FALSE	Der Schreibvorgang ist abgeschlossen (DONE = TRUE) Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Schreibvorgang abgeschlossen. Der Datensatz wurde im Flash-Speicher abgelegt. ● DONE = TRUE und ERR = TRUE: Beim Schreibvorgang ist ein Fehler aufgetreten. Der Ausgang ERNO signalisiert die Fehlernummer.
ERR	BOOL	FALSE	Fehler aufgetreten (Datensegment konnte nicht geschrieben werden) Am Ausgang ERR wird angezeigt, ob beim Schreibvorgang ein Fehler aufgetreten ist. Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Ist ein Fehler aufgetreten, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer ↳ *[3]* Am Ausgang ERNO wird eine Fehlernummer ausgegeben. Dieser Ausgang muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Da das Anwenderprogramm der Sicherheitssteuerung mit höherer Priorität ausgeführt wird, kann der Vorgang SF_FLASH_WRITE relativ viel Zeit beanspruchen. Am Ausgang ERNO wird dann angezeigt, dass der Funktionsbaustein die Verarbeitung angestoßen hat (0x0FFF = BUSY). Während dieser Phase sind ERR=FALSE und DONE=FALSE.

Aufruf in ST

```
WRITE_FLASH(EN := EN_FLASH_WRITE,
NB := NB_FLASH_WRITE,
SEG := SEG_FLASH_WRITE,
BNR := BNR_FLASH_WRITE,
SM := SM_FLASH_WRITE,
DONE => DONE_FLASH_WRITE,
ERR => ERR_FLASH_WRITE,
ERNO => ERNO_FLASH_WRITE);
```

4.6.7.12 SF_FLASH_DEL



Ausgewähltes Segment aus dem Flash-Speicher löschen

Dieser Funktionsbaustein löscht ein ausgewähltes Segment der Nutzerdaten aus dem Flash-Speicher.

Der Eingang SEG legt das Datensegment im Flash-Speicher fest. In der Sicherheits-CPU sind zwei Segmente (1 und 2) mit je 64 kB (inkl. CRC, Merker und Ausrichtung) für den Anwender reserviert. Das Löschen eines Datensegments im Flash-Speicher kann mehrere SPS-Zyklen dauern.

Mit einer FALSE/TRUE-Flanke am Eingang EN wird der einmalige Löschvorgang des Datensegments ausgelöst. Bis zur Beendigung des Löschvorganges (DONE = TRUE) wird der Eingang EN nicht mehr ausgewertet.

Nach Beendigung des Löschvorganges werden alle Funktionsbaustein-Ausgänge aktualisiert. Bei DONE = TRUE und ERR = FALSE war das Löschen erfolgreich. Sind die Ausgänge DONE = TRUE und ERR = TRUE, dann konnte das Datensegment nicht gelöscht werden.

> **HINWEIS!**
> Dieser Funktionsbaustein wird durch eine positive Flanke der Eingangsvariable EN aktiviert. Während des Zyklus, in dem der Funktionsbaustein feststellt, dass die Operation abgeschlossen ist (Ausgang DONE = TRUE), setzt er die Ausgangsvariablen nur für einen Zyklus. Wenn der Funktionsbaustein erneut aufgerufen wird, setzt er die Ausgangsvariablen sofort zurück.

Tab. 92: FB-Name: SF_FLASH_DEL

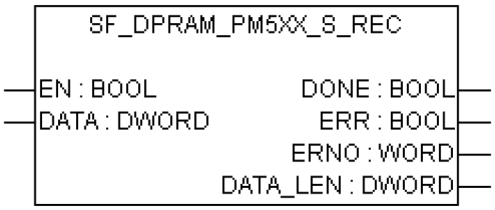
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Aktivierung des Funktionsbausteins mit positiver Flanke Es wird der einmalige Löschvorgang des Datensegments ausgelöst. Bis zur Beendigung des Löschvorganges (DONE = TRUE) wird der Eingang EN nicht mehr ausgewertet. EN = TRUE: Der Funktionsbaustein wird nicht verarbeitet, d. h. er verändert seine Ausgänge nicht mehr. Dies gilt nicht während eines Löschvorganges.

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
SEG	BYTE	16#00	ID-Nummer des Datensegments (16#01 oder 16#02)
VAR_OUTPUT			
DONE	BOOL	FALSE	Der Löschvorgang ist abgeschlossen (DONE = TRUE) Am Ausgang DONE wird angezeigt, dass der Löschvorgang des Datensegmentes abgeschlossen ist. Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Der Löschvorgang ist abgeschlossen. Das Datensegment wurde gelöscht. ● DONE = TRUE und ERR = TRUE: Beim Löschvorgang ist ein Fehler aufgetreten. Das Datensegment konnte nicht gelöscht werden.
ERR	BOOL	FALSE	Fehler aufgetreten (Datensegment konnte nicht gelöscht werden) Am Ausgang ERR wird angezeigt, ob beim Löschvorgang ein Fehler aufgetreten ist. Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Kann das Datensegment nicht gelöscht werden, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer ↳ *[3]* Am Ausgang ERNO wird eine Fehlernummer ausgegeben. Dieser Ausgang muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Da das Anwenderprogramm der Sicherheits-CPU mit höherer Priorität ausgeführt wird, kann SF_FLASH_DEL relativ viel Zeit beanspruchen. Am Ausgang ERNO wird angezeigt, dass der Funktionsbaustein die Verarbeitung angestoßen hat (0x0FFF = BUSY). Während dieser Phase sind ERR=FALSE und DONE=FALSE.

Aufruf in ST

```
DEL_FLASH(EN := EN_FLASH_DEL,
SEG := SEG_FLASH_DEL,
DONE => DONE_FLASH_DEL,
ERR => ERR_FLASH_DEL,
ERNO => ERNO_FLASH_DEL);
```

4.6.7.13 SF_DPRAM_PM5XX_S_REC



Lesen der Daten aus der Standard-CPU in die Sicherheitsanwendung der Sicherheits-CPU

GEFAHR!

Es wird nicht empfohlen, Datenwerte von der Standard-CPU auf die Sicherheits-CPU zu übertragen. Hierbei müssen die Endanwender zusätzliche prozessspezifische Validierungsverfahren in ihrem Sicherheitsprogramm definieren, um die Korrektheit der übertragenen nicht sicheren Daten zu überprüfen, wenn sie diese nicht sicheren Werte für Sicherheitsfunktionen verwenden möchten.

Datenwerte von der Sicherheits-CPU auf die Standard-CPU zu übertragen, z. B. für Diagnose und spätere Darstellung auf Bedienpanels, ist kein Problem.

GEFAHR!

Wenn der Funktionsbaustein SF_DPRAM_PM5XX_S_REC zum Empfangen von Sicherheitsdaten von der Sicherheits-CPU verwendet wird, sind die funktionalen sicherheitsbezogenen Anforderungen für SIL 3 (IEC 61508 und IEC 62061) und PL e (ISO 13849-1) für empfangene Daten nicht erfüllt (unabhängig vom verwendeten applikativen Sicherheitskommunikationsprofil), da in der Sicherheits-CPU nur ein Mikroprozessor (keine 1oo2-Sicherheitsarchitektur im Hintergrund) für die Empfangsrichtung zuständig ist.

Wenden Sie sich an den technischen Support von ABB, um Informationen zum Erreichen von SIL 3 und PL e zu erhalten.

Über den Funktionsbaustein SF_DPRAM_PM5XX_S_REC werden Daten von der Standard-CPU empfangen. Diese Daten werden im Speicherbereich abgelegt (DATA, Speicheradresse für die Empfangsdaten über ADR-Operator). Die Aktivierung des Funktionsbausteins erfolgt durch ein TRUE-Signal an Eingang EN. Der Baustein ist solange aktiv, bis Eingang EN = FALSE wird. An Ausgang DATA_LEN wird die Länge der empfangenen Daten in Byte ausgegeben. Ein erfolgreicher Datenempfang wird durch DONE=TRUE und ERR=FALSE signalisiert. Wurde bei der Verarbeitung des Funktionsbausteins ein Fehler festgestellt, wird er an den Ausgängen ERR und ERNO angezeigt.

HINWEIS!

Der Empfang mit dem Funktionsbaustein SF_DPRAM_SM5XX_S_REC ist nicht flankengetriggert. Der Eingang EN ist also dauerhaft auf TRUE zu setzen, solange Daten empfangen werden sollen.

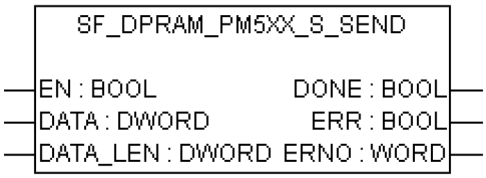
Tab. 93: FB-Name: SF_DPRAM_PM5XX_S_REC

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Die Verarbeitung dieses Funktionsbausteins wird vom Eingang EN gesteuert. Der Funktionsbaustein ist aktiv, wenn EN = TRUE ist. Der Empfang von Daten wird durch den Ausgang DONE signalisiert.
DATA	DWORD	16#00000000	Am Eingang DATA wird die Adresse der Variable angegeben, in die die Nutzerdaten kopiert werden sollen. Die an DATA spezifizierte Adresse muss zu einer Variablen vom Typ ARRAY oder STRUCT gehören. Speicherbereichsüberschneidungen vermeiden, indem die Größe der Variablen an die maximal zu erwartenden Daten angepasst wird.
VAR_OUTPUT			
DONE	BOOL	FALSE	Am Ausgang DONE wird der Empfang der Daten angezeigt. Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Der Empfangsvorgang ist abgeschlossen. Ein Datensatz wurde korrekt empfangen. ● DONE = TRUE und ERR = TRUE: Beim Empfangsvorgang ist ein Fehler aufgetreten. Die Fehlernummer wird am Ausgang ERNO ausgegeben.
ERR	BOOL	FALSE	Am Ausgang ERR wird angezeigt, ob beim Empfangsvorgang ein Fehler aufgetreten ist. Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Ist ein Fehler aufgetreten beim Empfang, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer ↳ *[3]* Am Ausgang ERNO wird eine Fehlerkennung ausgegeben, wenn an einem Eingang ein ungültiger Wert angegeben wurde oder während der Verarbeitung des Auftrags ein Fehler aufgetreten ist. ERNO muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Der an ERNO ausgegebene Wert ist nur gültig, wenn DONE = TRUE und ERR = TRUE ist.
DATA_LEN	DWORD	16#00000000	An Ausgang DATA_LEN wird die Länge der empfangenen Daten in Byte ausgegeben (max. 84). Der an DATA_LEN ausgegebene Wert ist nur gültig, wenn DONE = TRUE ist.

Aufruf in ST

```
PM5xxRec (EN := PM5xxRec_EN,
DATA := ADR(PM5xxRec_DATA),
DONE => PM5xxRec_DONE,
ERR => PM5xxRec_ERR,
ERNO => PM5xxRec_ERNO,
DATA_LEN => PM5xxRec_DATA_LEN);
```

4.6.7.14 SF_DPRAM_PM5XX_S_SEND



Senden von Daten von der Sicherheits-CPU an die Standard-CPU

Über den Funktionsbaustein SF_DPRAM_PM5XX_S_SEND werden Daten an die Standard-CPU gesendet. Diese Daten werden im Speicherbereich bereitgestellt (DATA, Speicheradresse für die Sendedaten über ADR-Operator). Der Funktionsbaustein wird mit einem TRUE-Signal (FALSE/TRUE-Flanke) am Eingang EN aktiviert. Am Eingang DATA_LEN wird die Länge der zu sendenden Daten in Byte angegeben. Ein erfolgreicher Sendevorgang wird durch DONE=TRUE und ERR=FALSE signalisiert. Wurde bei der Verarbeitung des Funktionsbausteins ein Fehler festgestellt, wird er an den Ausgängen ERR und ERNO angezeigt.

GEFAHR!

Wenn der Funktionsbaustein SF_DPRAM_PM5XX_S_SEND zum Senden von Sicherheitsdaten von der Sicherheits-CPU an die Standard-CPU verwendet wird, sind die funktionalen sicherheitsbezogenen Anforderungen für SIL 3 (IEC 61508 und IEC 62061) und PL e (ISO 13849-1) für gesendete Daten nicht erfüllt (unabhängig vom verwendeten applikativen Sicherheitskommunikationsprofil), da in der Sicherheits-CPU nur ein Mikroprozessor (keine 1oo2-Sicherheitsarchitektur im Hintergrund) für die Senderichtung zuständig ist.

Wenden Sie sich an den technischen Support von ABB, um Informationen zum Erreichen von SIL 3 und PL e zu erhalten.

HINWEIS!

Das Senden von Daten mit dem Funktionsbaustein SF_DPRAM_PM5XX_S_SEND ist flankengetriggert, d. h. jeder Sendevorgang wird durch eine FALSE-TRUE-Flanke am Eingang EN ausgelöst.

HINWEIS!

Dieser Funktionsbaustein wird durch eine positive Flanke der Eingangsvariable EN aktiviert. Während des Zyklus, in dem der Funktionsbaustein feststellt, dass die Operation abgeschlossen ist (Ausgang DONE = TRUE), setzt er die Ausgangsvariablen nur für einen Zyklus. Wenn der Funktionsbaustein erneut aufgerufen wird, setzt er die Ausgangsvariablen sofort zurück.

Tab. 94: FB-Name: SF_DPRAM_PM5XX_S_SEND

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Freigabe der Funktionsbausteinverarbeitung Die Verarbeitung dieses Funktionsbausteins wird vom Eingang EN gesteuert. Die Datenübertragung wird durch eine FALSE/TRUE-Flanke angestoßen. Das Senden von Daten wird durch den Ausgang DONE signalisiert

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
DATA	DWORD	16#00000000	Am Eingang DATA wird die Adresse der Variablen angegeben, in die die Anwenderdaten kopiert werden sollen. Die an DATA spezifizierte Adresse muss zu einer Variablen vom Typ ARRAY oder STRUCT gehören. Speicherbereichsüberschneidungen vermeiden, indem die Größe der Variablen an die maximal zu erwartenden Daten angepasst wird.
DATA_LEN	DWORD	16#00000000	Am Eingang DATA_LEN wird die Länge der zu sendenden Daten in Byte angegeben. Die maximale Anzahl ist 84.
VAR_OUTPUT			
DONE	BOOL	FALSE	Am Ausgang DONE wird der Versand der Daten angezeigt. Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Der Sendevorgang ist abgeschlossen. Es wurde ein Datensatz korrekt gesendet. ● DONE = TRUE und ERR = TRUE: Beim Sendevorgang ist ein Fehler aufgetreten. Die Fehlernummer wird am Ausgang ERNO ausgegeben.
ERR	BOOL	FALSE	Am Ausgang ERR wird angezeigt, ob beim Sendevorgang ein Fehler aufgetreten ist. Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Ist ein Fehler aufgetreten beim Versand, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer ↬ *[3]* Am Ausgang ERNO wird eine Fehlerkennung ausgegeben, wenn an einem Eingang ein ungültiger Wert angegeben wurde oder während der Verarbeitung des Auftrags ein Fehler aufgetreten ist. ERNO muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Der an ERNO ausgegebene Wert ist nur gültig, wenn DONE = TRUE und ERR = TRUE ist.

Aufruf in ST

```
PM5xxSend (EN := PM5xxSend_EN,
DATA := ADR(PM5xxSend_DATA),
DATA_LEN := PM5xxSend_DATA_LEN,
DONE => PM5xxSend_DONE,
ERR => PM5xxSend_ERR,
ERNO => PM5xxSend_ERNO);
```

5 Sicherheitszeiten

5.1 Übersicht

Fehler im System können zu gefährlichen Betriebszuständen führen. Potenzielle Fehler werden durch Selbsttests der Sicherheitsmodule im Hintergrund erkannt. Dadurch werden definierte Reaktionen auf die Fehler in den Sicherheitsmodulen ausgelöst, um die fehlerhaften Module in einen sicheren Zustand zu bringen. In diesem Kapitel werden verschiedene Sicherheitszeiten für AC500-S-Sicherheitsmodule und die Sicherheitssteuerung eines AC500-S-Systems aufgeführt.

5.2 Fehlerreaktionszeit

Die Fehlerreaktionszeit ist die maximale Zeit zwischen dem Auftreten des Fehlers im System und dem Auslösen der vordefinierten Fehlerreaktionen. Die unten aufgeführte Tabelle enthält einen Überblick der längsten Fehlerreaktionszeiten der AC500-S-Sicherheitsmodule.

Tab. 95: Fehlerreaktionszeiten der AC500-S-Sicherheitsmodule

Module	**Fehlerreaktionszeit**	
	Interne Fehler (z. B. RAM-Fehler)	**Externe Fehler (z. B. falsche Verkabelung)**
AC500-S-Sicherheits-CPUs	< 24 h	Nicht zutreffend
Sicherheits-E/A DI581-S	< 24 h	< 1,9 s
Sicherheits-E/A DX581-S	< 24 h	< 0,5 s
Sicherheits-E/A AI581-S	< 24 h	< 0,8 s

Sollten Sie weitere Einzelheiten zu den Fehlerreaktionszeiten benötigen, wenden Sie sich bei Bedarf an den technischen Support von ABB.

5.3 Antwortzeit der Sicherheitsfunktion (= Safety Function Response Time)

Die Antwortzeit der Sicherheitsfunktion (SFRT) ist die Zeit, innerhalb der die Sicherheitssteuerung AC500-S im normalen Modus RUN reagieren muss, nachdem ein Fehler im System aufgetreten ist.

Auf Anwendungsseite ist SFRT die maximale Zeit, in der das Sicherheitssystem auf die Veränderung der Eingangssignale oder Modulausfälle antworten muss.

SFRT ist eine der wichtigsten Zeiten im Bereich Sicherheit, da sie in zeitkritischen Sicherheitsanwendungen (z. B. Pressen) verwendet wird, um zum Schutz von Menschen vor den potenziell gefährlichen Maschinenteilen einen angemessenen Abstand für einen Lichtvorhang oder einen anderen Sicherheitssensor zu definieren.

SFRT kann für PROFIsafe-Geräte basierend auf *[7]* definiert werden als:

Gleichung 1: $SFRT = TWCDT + längstes\ \Delta T\_WD$

wobei:

- TWCDT (Gesamt-Worst-Case-Verzögerungszeit) ist die maximale Zeit für die Übertragung eines Eingangssignals im AC500-S-System bis zur Reaktion des Ausgangs im Worst-Case (alle Komponenten erfordern die maximale Zeit);
- Längstes ΔT_WD ist die längste Zeitdifferenz, die zwischen Ablauf der Worst-Case-Verzögerungszeit bis zum Ansprechen der Watchdog-Zeit entsteht. Im Sicherheitskontext muss zur Identifizierung von SFRT ein potenzieller Einzelfehler innerhalb der an einer Signalübertragung beteiligten Sicherheitsmodule in Betracht gezogen werden. Es ist ausreichend, nur einen Einzelfehler zu berücksichtigen ↳ *[7]*.

In Abb. 123, Abb. 124 und Abb. 125 wird SFRT im Detail erläutert. Im Modell in Abb. 123 und Abb. 124 werden die Phasen Lesen des Eingangssignals, sicherer Datentransfer, sichere Logikverarbeitung, sicherer Datentransfer und sichere Signalausgabe angeführt. Im Modell in Abb. 125 wird die sichere Kommunikation von CPU zu CPU dargestellt, die die Phasen sichere Logikverarbeitung, sicherer Datentransfer und sichere Logikverarbeitung beinhaltet.

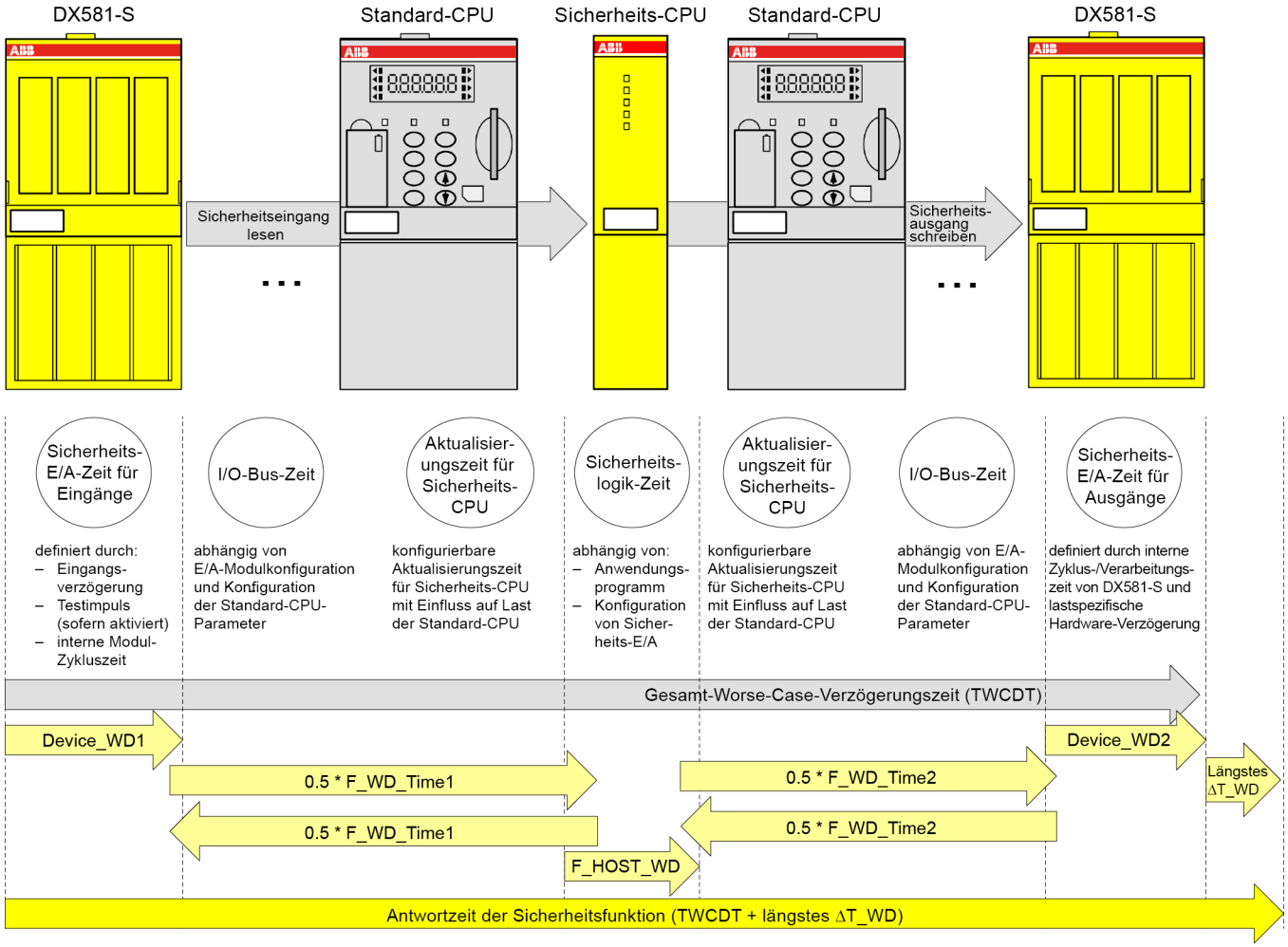


Abb. 123: SFRT in einem AC500-S-System ohne PROFINET-Komponenten

Alle Begriffe in dieser Abbildung werden ↳ *auf Seite 366* erläutert.

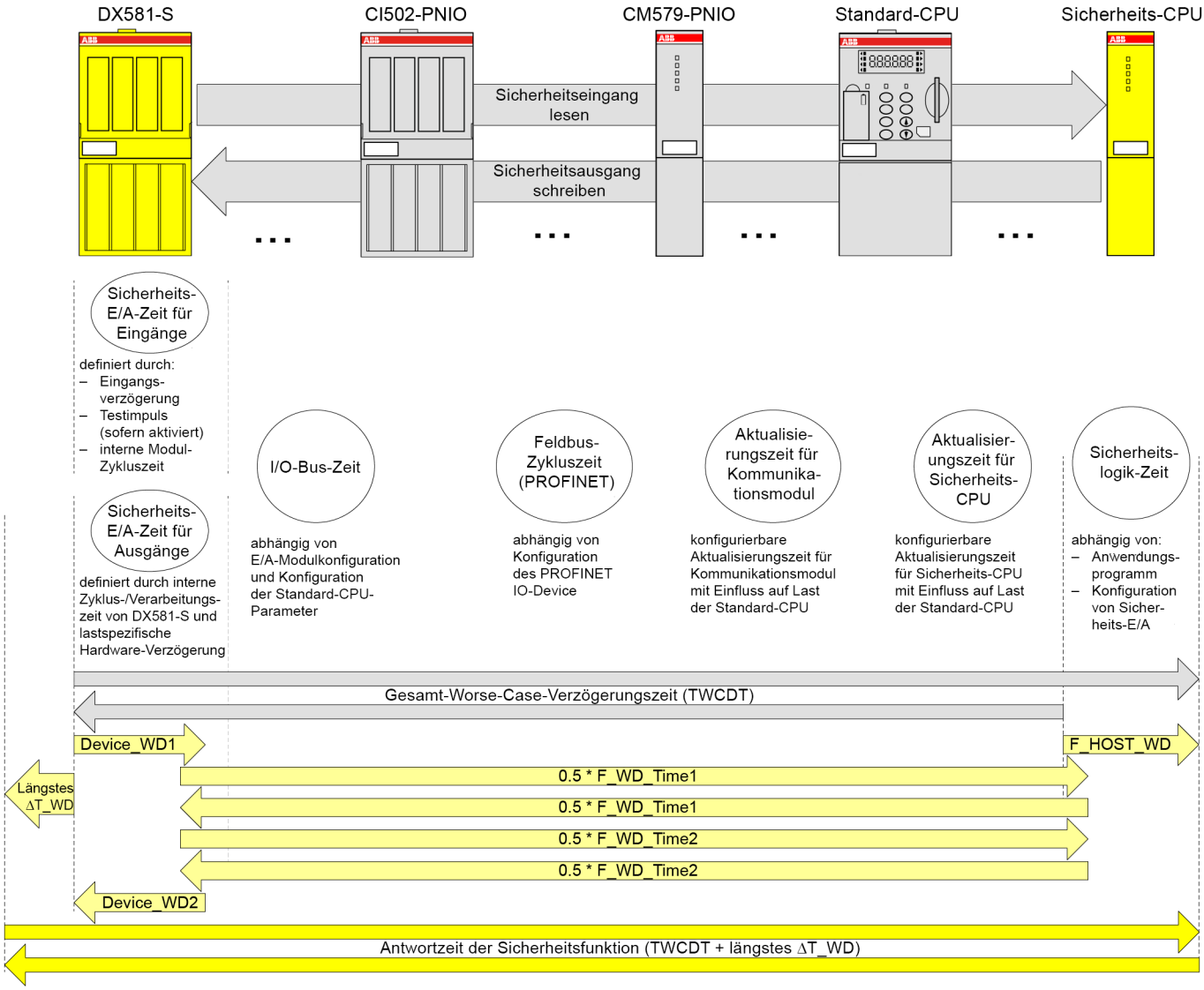


Abb. 124: SFRT in einem AC500-S-System mit PROFINET-Komponenten und Sicherheits-E/A-Modulen

Alle Begriffe in dieser Abbildung werden *auf Seite 366* erläutert.

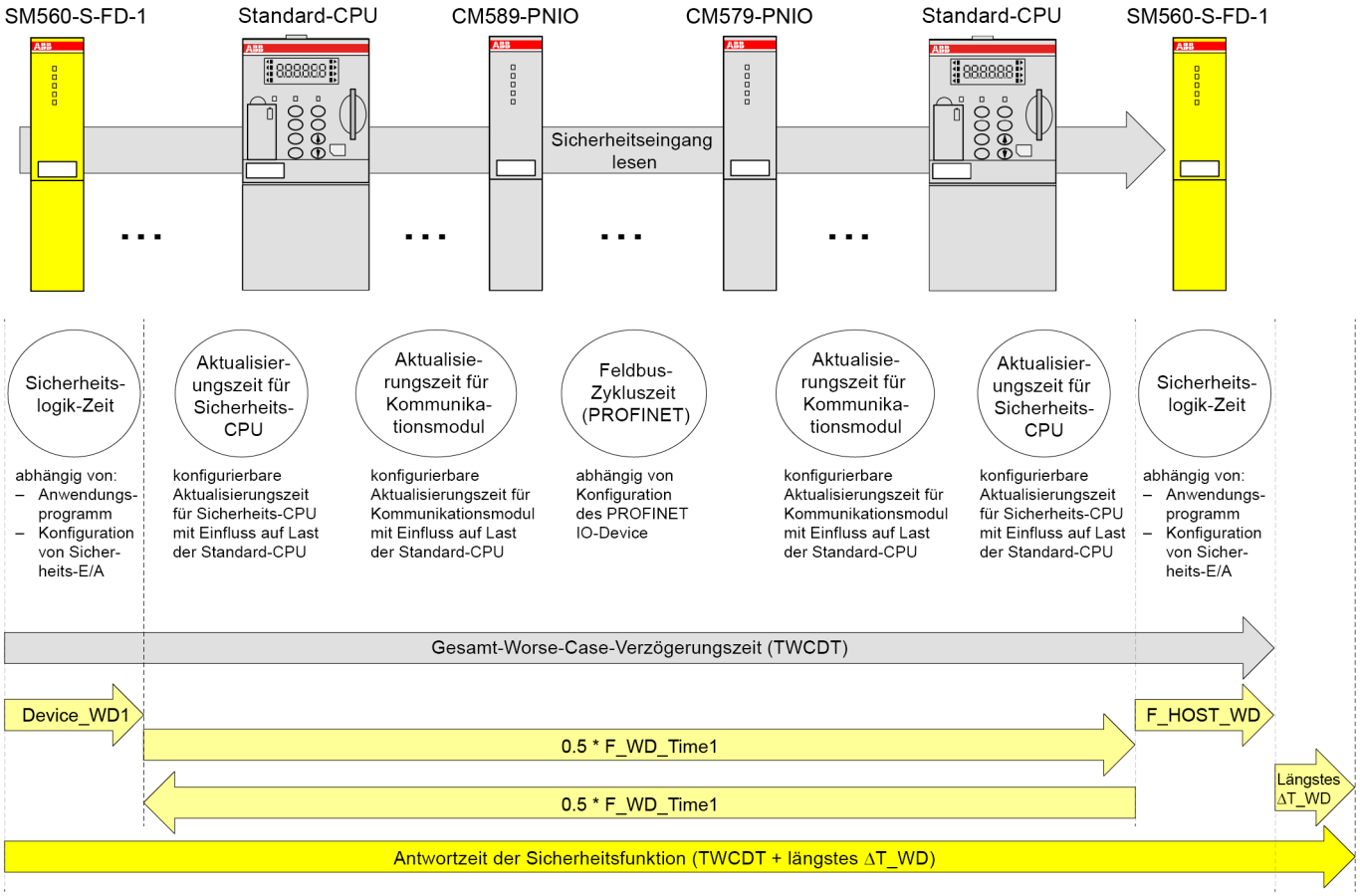


Abb. 125: SFRT in einem AC500-S-System mit PROFINET-Komponenten und sicherer Kommunikation von CPU zu CPU (Beispiel: SM560-S-FD-1 zu SM560-S)

Alle Begriffe in dieser Abbildung werden *auf Seite 366* erläutert.

Erläuterung von Begriffen im Zusammenhang mit SFRT

Die folgenden Begriffe werden (in alphabetischer Reihenfolge) in Abb. 123, Abb. 124 und Abb. 125 definiert:

- **Device_WD1 (Sicherheits-E/A-Zeit für Eingänge)** ist eine interne Watchdog-Zeit des Eingabegerätes in ms; sie umfasst:
  - Eingangsverzögerung (variierbar als Parameter; nicht bei sicheren Analogeingängen, die stattdessen eine interne Worst-Case-Eingangsverzögerung von 67,5 ms aufweisen);
  - Genauigkeit der Eingangsverzögerung *Tab. 4 „Genauigkeit der Eingangsverzögerung für DI581-S“ auf Seite 72* *Tab. 6 „Genauigkeit der Eingangsverzögerung für DX581-S“ auf Seite 99*
  - Low-Level-Testimpuls (festgesetzt auf 1 ms und optional (nur wenn Testimpulse verwendet werden); nicht bei sicheren Analogeingängen);
  - 2 × interne Zyklusdauer (fest; AI581-S ➔ 4,5 ms, DX581-S ➔ 5,5 ms und DI581-S ➔ 6,5 ms);
- **Device_WD2 (Sicherheits-E/A-Zeit für Ausgänge)** ist eine interne Watchdog-Zeit des Ausgabegerätes in ms; sie umfasst:
  - Interne Zyklusdauer des Sicherheitsausgabegerätes (fest; DX581-S ➔ 5,5 ms);
  - Verarbeitungszeit am Ausgang von DX581-S (fest 1,5 ms);
  - Hardwareverzögerung (abhängig vom Strom, z. B. ~1 ms (747 µs bei 5 mA) und maximal 4 ms bei einem maximalen Ausgangsstrom von 500 mA). Genauere Werte erhalten Sie vom technischen Support von ABB.
- **F_HOST_WD (Sicherheitslogikzeit)** entspricht dem Dreifachen der Watchdog-Zeit des Sicherheitsanwendungszyklus. Die Watchdog-Zeit des Sicherheitsanwendungszyklus ist unter Verwendung von POU SF_WDOG_TIME_SET konfigurierbar. Die Watchdog-Zeit des Sicherheitsanwendungszyklus hängt von der Zahl der F-Devices, dem Sicherheitsprogramm und der Systemkonfiguration ab.

- **F_WD_Time1** und **F_WD_Time2**: Die Summe entspricht der Gesamt-Datenübertragungszeit über den „Black-Channel“. Sie umfasst unterschiedliche „Black Channel“-Komponenten, z. B. Feldbus-Zykluszeit (PROFINET), I/O-Bus-Zeit und Aktualisierungszeit für Sicherheits-CPU (konfigurierbar als Parameter) und Kommunikationsmodul.
- Die **Feldbus-Zykluszeit (PROFINET)** hängt von den Kommunikationseinstellungen für das PROFINET IO-Device ab, an dem das Sicherheits-E/A-Modul angebracht ist. Die Zykluszeit ergibt sich aus der Multiplikation von zwei Parametern des PROFINET IO-Device.
  – *„Send clock“*, z. B. für CI501-PNIO und CI502-PNIO: 1 ms, 2 ms oder 4 ms
  – *„Reduction ratio“*, z. B. für CI501-PNIO und CI502-PNIO: 1, 2, 4, 8, 16 … 512

  Diese Werte können in Abhängigkeit von den definierten PROFINET-Parametern für dieses PROFINET-Modul ausgewählt werden.

- Die konfigurierbare **Aktualisierungszeit für Sicherheits-CPU und Kommunikationsmodule** beschreibt die Datenübertragungszeit über den Kommunikationsmodul-Bus.
  - Mit AC500 V2-Standard-CPU:

    Die Aktualisierungszeit kann sowohl für die Sicherheits-CPU als auch für die Kommunikationsmodule innerhalb eines Bereichs von 0 … 20000 ms konfiguriert werden.
  - Mit AC500 V3-Standard-CPU:

    Die Aktualisierungszeit für die Sicherheits-CPU kann innerhalb eines Bereichs von 1 … 20000 ms konfiguriert werden.

    Die Aktualisierungszeit für Kommunikationsmodule bezieht sich auf die Einstellungen für PROFINET IO-Controller (CM579-PNIO) und PROFINET IO-Device (CM589-PNIO). Sie ist definiert durch die Kommunikationsmoduleinstellung *„Buszyklus-Task“*, z. B. in der Registerkarte *„PROFINET-IO-Controller E/A-Abbild“*. Weiterführende Informationen: ↯ *„„Buszyklus-Task““ auf Seite 446*
- Die **I/O-Bus-Zeit** beschreibt die Datenübertragungszeit über den I/O-Bus für die Kommunikation zwischen der Standard-CPU und den zugehörigen lokalen I/O-Bus-Modulen sowie für die Kommunikation zwischen den Kommunikationsschnittstellen-Modulen und den zugehörigen lokalen I/O-Bus-Modulen.
  - Mit AC500 V2-Standard-CPU:

    Die I/O-Bus-Zykluszeit weist keinen festen vordefinierten Zykluswert auf. Sie wird unabhängig von den Einstellungen der Standard-CPU durch die Anzahl und den Typ der konfigurierten E/A-Module definiert. Die I/O-Bus-Zeit umfasst die folgenden Werte:

    – I/O-Bus-Master-Zyklus: 2 ms (2 Zyklen, je 1 ms)

    – I/O-Bus-Zykluszeit: In der Regel 2 … 5 ms (2 Zyklen, je 1 … 2,5 ms)

    Insgesamt beträgt der typische Bereich für die I/O-Bus-Zeit 4 … 7 ms.
  - Mit AC500 V3-Standard-CPU:

    Der I/O-Bus wird mit einer definierten Zykluszeit betrieben. Diese I/O-Bus-Zykluszeit bezieht sich auf die Einstellung *„Buszyklus-Task“* der Standard-CPU in der Registerkarte *„I/O-Bus E/A-Abbild“*. Weiterführende Informationen finden Sie unter: ↯ *„„Buszyklus-Task““ auf Seite 446*.

    Eine grundlegende Definition der I/O-Bus-Zykluszeiten wird für die Standard-CPU unter der Einstellung *„Buszyklus-Task“* in der Registerkarte *„SPS-Einstellungen“* vorgenommen.

    Beispiel für eine Einstellung mit Zuordnung zu einer Task mit einer Zykluszeit von 2 ms (und kürzer als die definierte Aktualisierungszeit für die Sicherheits-CPU):

    – Ergebnis für I/O-Bus-Master-Zyklus: 2 ms = 2 Zyklen, je 1 ms

    – Ergebnis für I/O-Bus-Zykluszeit: In der Regel 4 … 5 ms = 2 Zyklen, je 2 … 2,5 ms (wenn die konfigurierte Task-Zykluszeit für die I/O-Bus-Konstellation nicht ausreicht, kann die I/O-Bus-Zykluszeit auf maximal 2,5 ms verlängert werden)

    Insgesamt beträgt die I/O-Bus-Zeit für dieses Beispiel 6 … 7 ms.

    Weiterführende Informationen finden Sie unter ↯ *„„Buszyklus-Task““ auf Seite 446*, z. B. für I/O-Bus.
  - Mit Kommunikationsschnittstellen-Modul CI50x-PNIO:

    Die I/O-Bus-Zykluszeit weist keinen festen vordefinierten Zykluswert auf. Sie wird unabhängig von den Einstellungen des Kommunikationsschnittstellen-Moduls durch die Anzahl und den Typ der konfigurierten E/A-Module definiert. Die I/O-Bus-Zeit umfasst die folgenden Werte:

    – I/O-Bus-Master-Zyklus: 2 ms (2 Zyklen, je 1 ms)

    – I/O-Bus-Zykluszeit: In der Regel 4 … 7 ms (2 Zyklen, je 2 … 3,5 ms)

    Insgesamt beträgt der typische Bereich für die I/O-Bus-Zeit 6 … 9 ms.

Unten sind einige Beispiele zur Berechnung von SFRT-Werten in den vorgestellten AC500-S-Systemkonfigurationen angegeben. Bei der Berechnung der SFRT wird folgender Ansatz auf der Basis von ↯ *[2]* und ↯ *[7]* angewendet:

Gleichung 2: $SFRT = Device\_WD1 + 0{,}5 * F\_WD\_Time1 + F\_Host\_WD + 0{,}5 * F\_WD\_Time2 + Device\_WD2 + längstes\ \Delta T\_WD$

GEFAHR!

Eingangsverzögerung, Genauigkeit der Eingangsverzögerung und Testimpuls-Low-Phase sind für AI581-S nicht erforderlich. Jedoch sollte für AI581-S die für den Worst-Case festgelegte interne Eingangsverzögerung von 67,5 ms verwendet werden.

GEFAHR!

Die Genauigkeit der Eingangsverzögerung muss unter den folgenden Annahmen berechnet werden:

- Wird nicht für sichere Analogeingänge verwendet.
- Wenn für den entsprechenden sicherheitsgerichteten Digitaleingang keine Testimpulse konfiguriert wurden, kann die Genauigkeit der Eingangsverzögerung berechnet werden als 1 % der eingestellten Eingangsverzögerung (die Genauigkeit der Eingangsverzögerung muss jedoch mindestens 0,5 ms sein!).
- Wenn für den entsprechenden sicherheitsgerichteten Digitaleingang Testimpulse konfiguriert wurden, können in Abhängigkeit vom Modultyp (DI581-S oder DX581-S) und vom gesetzten Wert für die Eingangsverzögerung die folgenden Werte für die Genauigkeit der Eingangsverzögerung bei der Berechnung der SFRT verwendet werden: *Tab. 4 „Genauigkeit der Eingangsverzögerung für DI581-S“ auf Seite 72* *Tab. 6 „Genauigkeit der Eingangsverzögerung für DX581-S“ auf Seite 99*

HINWEIS!

↳ *Gleichung 2, Seite 368* wurde für die Berechnung von SFRT aus folgenden Gründen gewählt:

- Device_WD1 und Device_WD2 als Worst-Case-Verzögerungszeiten für Sicherheits-E/As können wie unter Abb. 123 und Abb. 124 dargelegt definiert werden.
- Für die Berechnung der Worst-Case-Verzögerungszeit für „Black Channel"-Komponenten (siehe AC500-Standardmodule in Abb. 123 und Abb. 124) wird empfohlen, stattdessen den halben Wert von F_WD_Time1 und F_WD_Time2 zu verwenden. F_WD_Time1 und F_WD_Time2 können empirisch für die AC500-Systemkonfiguration bestimmt werden, indem man die Werte für tResponseTimeMS für gegebene Sicherheits-E/As in der Sicherheitsanwendung zurückverfolgt. Verwenden Sie die PROFIsafe-Instanz für den gegebenen Sicherheits-E/A ↳ *Kapitel 4.6.3 „Safety-Base_PROFIsafe_LV210_AC500_V22.lib" auf Seite 212*. F_WD_Time1 und F_WD_Time2 sollten etwa 30 % höher als der Worst-Case-Wert für tResponseTimeMS des gegebenen Sicherheits-E/A gesetzt werden.
- Es wird empfohlen, die Zeit F_Host_WD statt der Worst-Case-Verzögerungszeit der Sicherheits-CPU SM560-S zu nehmen. F_Host_WD wird berechnet als drei Mal der Wert, der mithilfe der POEs SF_WDOG_TIME_SET gesetzt wird. Der korrekte Wert für SF_WDOG_TIME_SET kann empirisch bestimmt werden, indem man den Ausgang MAX_TIME derselben POE in einem Testlauf verfolgt. Der Wert für SF_WDOG_TIME_SET sollte ca. 30 % höher als der Worst-Case-Wert (MAX_TIME) sein, der in der Sicherheitsanwendung beobachtet wurde, um mögliche Verfügbarkeitsprobleme durch das Auslösen des Watchdogs der Sicherheits-CPU SM560-S zu vermeiden.
- F_WD_Time1 und F_WD_Time2 sind die einzigen potenziellen Kandidaten für Längstes ΔT_WD, da F_Host_WD, Device_WD1 und Device_WD2 bereits gleich der Worst-Case-Verzögerungszeit sind. Somit ist

Längstes ΔT_WD = Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2)

HINWEIS!

Bessere SFRT-Werte als mit ↳ *Gleichung 2, Seite 368* erhält man mit einer detaillierten technischen Analyse. Wenden Sie sich an den technischen Support von ABB für weitere Details.

HINWEIS!

Die Werte F_WD_Time1 und F_WD_Time2 müssen mindestens doppelt so groß sein wie die mit SF_WDOG_TIME_SET eingestellte Zeit, um ungewollte Systemstopps aufgrund des Ablaufens des PROFIsafe-Watchdogs zu vermeiden.

GEFAHR!

AC500-S-Sicherheits-E/A-Module erfüllen die Anforderung der IEC 61131 zum Überbrücken einer möglichen Unterspannung mit einer Dauer von bis zu 10 ms. Während dieser Unterspannungsphase von bis zu 10 ms liefern die AC500-S-Sicherheits-E/A-Module den letzten Prozesswert, der vor der Erkennung der Unterspannung gültig war, für die sicherheitsgerichteten Analogeingangskanäle im AI581-S und für die sicherheitsgerichteten Digitaleingänge/-ausgänge in den Modulen DI581-S und DX581-S.

Wenn die Unterspannungsphase länger als 10 ms andauert, werden die E/A-Module passiviert ↳ *Kapitel 3.2.3 „Unterspannung / Überspannung“ auf Seite 68*.

Wenn häufig Unterspannungen mit einer Dauer von < 10 ms in der Sicherheitsanwendung auftreten, müssen Sie 10 ms für das AI581-S-Modul in der Berechnung der SFRT hinzufügen, um eine Überbrückungsphase (wie oben beschrieben) zu berücksichtigen. In der Regel geht man davon aus, dass Unterspannungen mit einer Dauer von < 10 ms in der Spannungsversorgung des Sicherheitssystems eher selten und daher mit geringer Wahrscheinlichkeit auftreten, sodass diese in der SFRT-Berechnung außer Acht gelassen werden können.

Basierend auf Abb. 123, Abb. 124 und Abb. 125 können die folgenden SFRT-Beispielwerte für einige typische AC500-S-Konfigurationen durch die Nutzung von ↳ *Gleichung 2, Seite 368* berechnet werden:

Ohne PROFINET (DI581-S ➔SM560-S ➔ DX581-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + 0,5 * F_WD_Time2 + Device_WD2 + Längstes ΔT_WD = 14,5 + 10 + 6 + 10 + 8 + 10 = 58,5 ms

wobei:

- Device_WD1 = 1 ms + 0,5 ms + 2 x 6,5 ms = 14,5 ms (ohne Testimpulse)
- F_WD_Time1 = 20 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- F_WD_Time2 = 20 ms
- Device_WD2 = 8 ms (Ausgangsstrom = ~ 5 mA)
- Längstes ΔT_WD = Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2) = 10 ms

Ohne PROFINET (DX581-S ➔ SM560-S ➔ DX581-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + 0,5 * F_WD_Time2 + Device_WD2 + Längstes ΔT_WD = 12,5 + 10 + 6 + 10 + 8 + 10 = 56,5 ms

wobei:

- Device_WD1 = 1 ms + 0,5 ms + 2 x 5,5 ms = 12,5 ms (ohne Testimpulse)
- F_WD_Time1 = 20 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- F_WD_Time2 = 20 ms
- Device_WD2 = 8 ms (Ausgangsstrom = ~ 5 mA)
- Längstes ΔT_WD = Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2) = 10 ms

Ohne PROFINET (AI581-S ➔ SM560-S ➔ DX581-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + 0,5 * F_WD_Time2 + Device_WD2 + Längstes ΔT_WD = 76,5 + 10 + 6 + 10 + 8 +10 = 120,5 ms

wobei:

- Device_WD1 = 2 x 4,5 ms + 67,5 ms = 76,5 ms
- F_WD_Time1 = 20 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- F_WD_Time2 = 20 ms
- Device_WD2 = 8 ms (Ausgangsstrom = ~ 5 mA)
- Längstes ΔT_WD = Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2) = 10 ms

Mit PROFINET (DI581-S ➔ SM560-S ➔ DX581-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + 0,5 * F_WD_Time2 + Device_WD2 + Längstes ΔT_WD = 14,5 + 15 + 6 + 15 + 8 + 15 = 73,5 ms

wobei:

- Device_WD1 = 1 ms + 0,5 ms + 2 x 6,5 ms = 14,5 ms (ohne Testimpulse)
- F_WD_Time1 = 30 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- F_WD_Time2 = 30 ms
- Device_WD2 = 8 ms (Ausgangsstrom = ~ 5 mA)
- Längstes ΔT_WD = Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2) = 15 ms

Mit PROFINET (DX581-S ➔ SM560-S ➔ DX581-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + 0,5 * F_WD_Time2 + Device_WD2 + Längstes ΔT_WD = 12,5 + 15 + 6 + 15 + 8 + 15 = 71,5 ms

wobei:

- Device_WD1 = 1 ms + 0,5 ms + 2 x 5,5 ms = 12,5 ms (ohne Testimpulse)
- F_WD_Time1 = 30 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- F_WD_Time2 = 30 ms
- Device_WD2 = 8 ms (Ausgangsstrom = ~ 5 mA)
- Längstes ΔT_WD = (Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2) = 15 ms

Mit PROFINET (AI581-S ➔ SM560-S ➔ DX581-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + 0,5 * F_WD_Time2 + Device_WD2 + Längstes ΔT_WD = 76,5 + 15 + 6 + 15 + 8 + 15 = 135,5 ms

wobei:

- Device_WD1 = 2 x 4,5 ms + 67,5 ms = 76,5 ms
- F_WD_Time1 = 30 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- F_WD_Time2 = 30 ms
- Device_WD2 = 8 ms (Ausgangsstrom = ~ 5 mA)
- Längstes ΔT_WD = Max (0,5 * F_WD_Time1; 0,5 * F_WD_Time2) = 15 ms

Mit PROFINET (SM560-S-FD-1 ➔ SM560-S)

SFRT = Device_WD1 + 0,5 * F_WD_Time1 + F_Host_WD + Längstes ΔT_WD = 9 + 25 + 6 + 25 = 65 ms

wobei:

- Device_WD1 = 3 x 3 ms (SF_WDOG_TIME_SET Zeit) = 9 ms
- F_WD_Time1 = 50 ms
- F_Host_WD = 3 x 2 ms (SF_WDOG_TIME_SET Zeit) = 6 ms
- Längstes ΔT_WD = 0,5 * F_WD_Time1 = 25 ms

HINWEIS!

SFRT-Berechnung für solche Fälle wie SM560-S-FD-4 ➔ SM560-S, SM560-S ➔ SM560-S-FD-1, SM560-S ➔ SM560-S-FD-4 usw. kann wie unter Abb. 125 gezeigt berechnet werden.

GEFAHR!

Fehler in der SFRT-Berechnung können Tod oder schwere Verletzung von Personen zur Folge haben, insbesondere in Anwendungen mit Pressen, Roboterzellen usw.

HINWEIS!

Die Tasks mit hoher Priorität der Standard-CPU, die Teil des „Black Channel“ für sichere Kommunikation sind, können die TWCDT für die Sicherheitssteuerung AC500-S beeinflussen.

6 Checkliste für die Inbetriebnahme der AC500-S

6.1 Übersicht

Alle Anwender der Sicherheitssteuerung AC500-S müssen die Punkte aus den Checklisten im diesem Kapitel für die Inbetriebnahme der Serie AC500-S berücksichtigen und in ihren Endberichten dokumentieren.

Die in den Checklisten aufgeführten Punkte betreffen nur die wichtigsten Aspekte in Bezug auf die Sicherheitssteuerung AC500-S. Dies bedeutet, dass Anwender die Checklisten für AC500-S um weitere Aspekte, die für ihre Sicherheitsanwendungen wichtig sind, ergänzen können.

6.2 Checkliste für die Erstellung von Sicherheitsprogrammen

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
1.	Prüfen Sie, ob für alle Sicherheitsfunktionen nur Sicherheitssignale verwendet werden.		
2.	Prüfen Sie, ob sowohl das Projekt der Sicherheitsanwendung in die Sicherheits-CPU als auch das relevante Projekt der Standardanwendung in die Standard-CPU geladen ist. Prüfen Sie, ob die Programme aus dem RAM-Speicher in den Flash-Speicher gesichert wurden, d. h. „Bootprojekt erzeugen“ wurde durchgeführt.		
3.	Bis Automation Builder 2.2.x: Prüfen Sie, ob die F-Parameter für alle Sicherheits-E/As und andere F-Devices, die im F-Parameter-Editor gesetzt wurden, dieselben sind wie jene, die aufgeführt sind in AC500-S Programming Tool: *„Globale Variablen ➔ PROFIsafe“ Kapitel 4 „Konfiguration und Programmierung“ auf Seite 144*. Automation Builder 2.3.x (und höher): Prüfen Sie, ob ein gültiger SVT-Bericht für das Anwendungsprojekt vorhanden ist.		
4.	Der F-Host auf der Sicherheits-CPU kann ggf. mehr als eine F_Source_Add verarbeiten, z. B. für das Koppeln von PROFIsafe Master – Master aus verschiedenen Netzwerken. Prüfen Sie, ob die Einstellungen der F_Source_Add für verschiedene F-Devices der Sicherheitsanwendung eindeutig sind. *Anmerkung:* *Die Regel „F_Source_Add <> F_Dest_Add für F-Device“ wird automatisch vom Automation Builder geprüft.*		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
5.	iParameter validieren. Dafür gibt es zwei Optionen: A) Überprüfen Sie mithilfe von für diese Parameter geeigneten Funktionsvalidierungstests, ob alle iParameter (Eingangsverzögerung, Kanalkonfiguration usw.) für alle Sicherheits-E/As und anderen F-Devices mit einem gegebenen F_iPar_CRC-Wert korrekt sind (weitere Informationen erhalten Sie beim technischen Support von ABB). oder B) Verwenden Sie ein spezielles Verifizierungsverfahren (siehe ↳ *Kapitel 6.5 „Verifizierung einer sicheren iParameter-Einstellung in den AC500-S-Sicherheits-E/As“ auf Seite 382*) zur Validierung jedes iParameter. Führen Sie anschließend nur Tests zur Validierung der Funktionssicherheit Ihrer Anwendung durch (es ist nicht notwendig, jeden einzelnen iParameter-Wert zu überprüfen). Ein Bericht, in dem bestätigt wird, dass sämtliche iParameter wie in ↳ *Kapitel 6.5 „Verifizierung einer sicheren iParameter-Einstellung in den AC500-S-Sicherheits-E/As“ auf Seite 382* beschrieben überprüft wurden, muss erstellt werden. Stellen Sie sicher, dass alle F_iPar_CRC > 0 sind.		
6.	Überprüfen Sie, ob die Sicherheitsprogrammierrichtlinien im Programm der Sicherheitsanwendung korrekt verwendet wurden ↳ *Kapitel 4.4 „Sicherheitsprogrammierrichtlinien“ auf Seite 196*.		
7.	Alle Signale vom nicht sicherheitsgerichteten Anwenderprogramm der Standard-CPU, die im Sicherheitsprogramm der Sicherheit-CPU evaluiert werden, müssen auch im Ausdruck des Sicherheitsprogramms erscheinen.		
8.	Wurde das Sicherheitsprogramm von einer Person überprüft, die an der Erstellung nicht beteiligt war?		
9.	Wurde das Ergebnis der Überprüfung des Sicherheitsprogramms dokumentiert und freigegeben (mit Datum und Unterschrift)?		
10.	Wurde ein Backup des vollständigen Sicherheits- (siehe Hinweis unten) und Standardprojekts erstellt, bevor Programme in die Sicherheits- und die Standard-CPU geladen wurden? *Hinweis:* ● *Stellen Sie sicher, dass Dateiname, Änderungsdatum, Titel, Autor, Version, Beschreibung und CRC des Sicherheits-Bootprojekts in einem Backup dokumentiert sind.* ● *Keine weiteren Veränderungen sind an Sicherheitsteilen im Automation Builder-Projekt und in AC500-S Programming Tool erlaubt. Wenn trotzdem Veränderungen vorgenommen werden, führt dies zu einer neuen CRC des Sicherheits-Bootprojekts. In diesem Fall muss diese Checkliste erneut von Beginn an durchgearbeitet werden.*		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
11.	Prüfen Sie über den Menüpunkt *„Online ➔ Prüfe Bootprojekt der Steuerung“*, ob das Offline-Sicherheitsprojekt von AC500-S Programming Tool und das Bootprojekt in der Sicherheits-CPU identisch sind (Dateiname, Änderungsdatum, Titel, Autor, Version, Beschreibung und CRC).		
12.	Wenn Fließkommaoperationen verwendet werden, überprüfen Sie, ob die Regeln aus Abschnitt *Kapitel 3.1.2.2 „Fließkommaoperationen“ auf Seite 39* berücksichtigt wurden und nicht zu unsicheren Zuständen in der Sicherheitsanwendung führen.		
13.	Prüfen Sie, ob die POE SF_WDOG_TIME_SET einmal im Sicherheitsprogramm aufgerufen wird und die Watchdog-Zeit korrekt gewählt wurde.		
14.	Prüfen Sie, ob ein Passwort für die Sicherheits-CPU definiert wurde, um nicht autorisierten Zugang zu den Daten zu verhindern.		
15.	Prüfen Sie, ob nur autorisiertes Personal *„Schreibzugang“* zu den Parametereinstellungen der Sicherheitsmodule und Programmen in Automation Builder und AC500-S Programming Tool hat.		
16.	Prüfen Sie, ob der korrekte Wert zur Überwachung der Spannungsversorgung mit POE SF_MAX_POWER_DIP_SET gesetzt wurde, damit das System bei Unter- oder Überspannung richtig funktioniert.		
17.	Prüfen Sie, ob die POE SF_SAFETY_MODE im Programm der Sicherheitsanwendung korrekt verwendet wird, um eine ungewollte Ausführung des Programms im DEBUG-Modus (nicht sicher) zu verhindern.		
18.	Prüfen Sie, ob keine Profilversionsänderung oder die Funktionen *„Gerät aktualisieren …“*, Export / Import, Kopieren und Einfügen und Archivieren im Automation Builder auf die Sicherheitsmodule angewandt wurden, nachdem das Projekt validiert wurde. Wenn die oben erwähnten Funktionen verwendet wurden und dies zu einem Sicherheits-Bootprojekt mit neuer CRC führte, muss ein kompletter Funktionstest sämtlicher Teile der Sicherheitsanwendung durchgeführt werden. Für diesen Test muss die Maschine in ihrem endgültigen Zustand sein, d. h. einschließlich der mechanischen, elektrischen und elektronischen Komponenten, Sensoren, Aktoren und der Software.		
19.	Überprüfen Sie mithilfe der Bibliotheks-CRC (gezeigt in AC500-S Programming Tool), ob nur zertifizierte Sicherheitsbibliotheken mit korrekten CRCs (siehe *Kapitel 4.6.1 „Übersicht“ auf Seite 207*) in dem betreffenden Sicherheitsprojekt zur Ausführung der Sicherheitsfunktionen verwendet werden. Alle anderen anwenderspezifischen Bibliotheken müssen von den Endanwendern separat validiert werden, damit deren Eignung für eine bestimmte Sicherheitsanwendung bestätigt wird.		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
20.	Stellen Sie sicher, dass interne POEs von SafetyUtil_CoDeSys_AC500_V22.lib und interne Aktionen von SafetyBase_PROFIsafe_LV210_AC500_V22.lib (oder ältere Versionen) nicht vom Endanwenderprogramm, das vom PLC_PRG im Hauptpfad startet, aufgerufen werden.		
21.	Stellen Sie sicher, dass in AC500-S Programming Tool alle drei Systemereignisse (*„CallbackInit“*, *„CallbackReadInputs“* und *„CallbackWriteOutputs“*) in *„Ressourcen ➔ Task-Konfiguration ➔ System-Ereignisse“* ausgewählt sind.		
22.	Wenn der Inhalt des Flash-Speichers (Funktionsbausteine SF_FLASH_READ bzw. SF_FLASH_WRITE werden in der Sicherheitsanwendung aufgerufen) für Sicherheitsfunktionen in der Sicherheitsanwendung verwendet wird, müssen entsprechende Verfahren zur Validierung des Flash-Speicher-Inhalts (z. B. geeignete Sicherheitsanwendungs-CRC über gespeicherte Sicherheitsdaten) implementiert werden, um die Datenintegrität der Sicherheitsanwendung sicherzustellen.		
23.	Prüfen Sie: ● ob die symbolischen Variablen konfigurierter F-Devices ordnungsgemäß zugeordnet sind, und ● ob die gelieferten Sicherheitsdaten in Ihrer Sicherheitsanwendung korrekt dargestellt sind. Das heißt, ob Datentypen, die mehr als ein Byte erfordern (wie Unsigned16, Unsigned32, Integer16, Integer32 oder Float32) in PROFIsafe-Daten verwendet werden. *Hinweis:* *Die Bytefolge in PROFIsafe-Datentypen hängt von der verwendeten Byte-Reihenfolge (Endianwert) des PROFIsafe-Gerätes und dem ausgewählten AC500-CPU-Typ ab. (AC500 V2-Standard-CPU unterstützt Big Endian. AC500 V3-Standard-CPU unterstützt Little Endian.)*		
24.	Wenn Sie den zyklischen nicht sicheren Datenaustausch nutzen, stellen Sie sicher, dass beim Senden von Daten über den zyklischen nicht sicheren Datenaustausch nur die Sicherheitsfunktionen mit bis zu SIL 2 (IEC 61508 und IEC 62061) und PL d (ISO 13849-1) angestoßen werden. *Hinweis:* *Wenn der zyklische nicht sichere Datenaustausch zum Senden oder Empfangen von sicherheitskritischen Daten verwendet wird, sind die sicherheitsbezogenen Anforderungen für SIL 3 (IEC 61508 und IEC 62061) und PL e (ISO 13849-1) für gesendete oder empfangene Daten nicht erfüllt (unabhängig vom verwendeten applikativen Sicherheitskommunikationsprofil), da in der Sicherheits-CPU nur ein Mikroprozessor (keine 1oo2-Sicherheitsarchitektur im Hintergrund) für die Sende- und Empfangsrichtung zuständig ist.* *Wenden Sie sich an den technischen Support von ABB, um Informationen zum Erreichen von SIL 3 und PL e zu erhalten.*		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
25.	Wenn Sie den zyklischen nicht sicheren Datenaustausch nutzen, prüfen Sie, ob beim zyklischen nicht sicheren Datenaustausch die Namen der Variablen, die für die Sicherheits-CPU angelegt werden, nicht mit „S_“, „GS_“, „IS_“ oder „OS_“ beginnen.		
Prüfer: Maschine/Applikation <ID>: Unterschrift: Datum:			

6.3 Checkliste für Konfiguration und Verkabelung

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
1.	Sind sämtliche Sicherheits-Ein-/Ausgangssignale korrekt konfiguriert worden und die Ausgangssignale an die physischen Ausgangskanäle angeschlossen?		
2.	Prüfen Sie, ob die Drehschalter-Adressen 0xF0 … 0xFF der Sicherheits-CPU nicht zur Identifizierung der Sicherheits-CPU verwendet werden (z. B. PROFIsafe-Adressen).		
3.	Prüfen Sie, ob spezielle organisatorische Abläufe (z. B. beschränkter Zugriff auf den Schaltschrank, in dem sich die Sicherheits-CPU befindet) am Standort des Endanwenders definiert werden, um eine ungewollte Firmware- und/oder Bootcode-Aktualisierung der Sicherheits-CPU mit einer SD-Karte zu verhindern.		
4.	Prüfen Sie, ob die korrekten Parametereinstellungen der Standard-CPU für die Sicherheitsanwendung verwendet werden ↳ *Anhang B.3 „Konfiguration der AC500 V2-Standard-CPU-Parameter“ auf Seite 427* ↳ *Anhang C.3 „Konfiguration der AC500 V3-Standard-CPU-Parameter“ auf Seite 445*.		
5.	Prüfen Sie, ob die erforderliche Antwortzeit der Sicherheitsfunktion Ihrer Sicherheitsanwendung mit den aktuellen Einstellungen der Sicherheitssteuerung AC500-S möglich ist und ob Sie die SFRT wie in ↳ *Kapitel 5.3 „Antwortzeit der Sicherheitsfunktion (= Safety Function Response Time)“ auf Seite 363* beschrieben berechnet haben.		
6.	Prüfen Sie, ob keiner der Sicherheits-Ausgangskanäle eine Konfiguration mit dem Parameter „Erkennung“ = AUS hat, was die Sicherheitsdiagnose für solch einen Sicherheits-Ausgangskanal verringert. Wenn diese Konfiguration verwendet wird, geben Sie im Bereich „Anmerkung“ dieser Checkliste Ihre Gründe an und begründen Sie, warum die erforderlichen SIL- und PL-Werte für die Anwendung mit solch einer Konfiguration erreicht werden können.		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
7.	Prüfen Sie Folgendes: ● Adresseinstellung ist korrekt. ● Belegung der Signaleingänge ist vollständig. ● Belegung der Signalausgänge ist vollständig. ● Belegung nicht benutzter Eingänge ist vollständig. ● Alle Klemmenblöcke sind gesteckt.		
8.	Prüfen Sie, ob die korrekten Firmware-Versionen für abhängige Standardkomponenten ↳ *Anhang B.1 „Kompatibilität mit AC500 V2-Standard-CPU“ auf Seite 415* ↳ *Anhang C.1 „Kompatibilität mit AC500 V3-Standard-CPU“ auf Seite 437* verwendet werden. Wenden Sie sich bei Bedarf an den technischen Support von ABB.		
9.	Prüfen Sie, ob nur eine Sicherheits-CPU an der Standard-CPU angebracht ist. Die Verwendung von mehr als einer Sicherheits-CPU an einer Standard-CPU ist nicht erlaubt.		
10.	Prüfen Sie, ob das korrekte Sicherheits-Bootprojekt auf die richtige Sicherheits-CPU AC500-S geladen wird, beispielsweise durch organisatorische Verfahren oder Fehlerausschluss (nur eine Sicherheits-CPU ist in der Maschine verfügbar). Beispiele für organisatorische Verfahren sind: ● Wird ein Engineering-PC verwendet und gibt es mehr als eine Sicherheits-CPU, dann stellen Sie sicher, dass nur eine und zwar die richtige Sicherheits-CPU für den Engineering-PC erreichbar ist, wenn ein bestimmtes Sicherheits-Bootprojekt auf die Sicherheits-CPU übertragen wird. ● Wird eine SD-Karte verwendet und gibt es mehr als eine Sicherheits-CPU, dann bezeichnen Sie eindeutig jede Sicherheits-CPU und SD-Karte mit einer geeigneten ID-Kennzeichnung auf Etiketten auf jeder Sicherheits-CPU und SD-Karte. Diese ID-Kennzeichnungen auf Etiketten müssen eine gut lesbare, eindeutige Bezeichnung jedes Objektes bieten, um klare Regeln für die Beziehungen „SD-Karte mit gegebenem Sicherheits-Bootprojekt – Sicherheits-CPU“ zu schaffen.		
11.	Prüfen Sie, ob die folgenden Regeln für die sichere Kommunikation von CPU zu CPU mit den CPUs SM560-S-FD-1 und SM560-S-FD-4 korrekt angewandt wurden: ● Im gleichen Codename-Space muss F_Dest_Add eindeutig sein (Abb. 6, Seite 45). ● Im gleichen Codename-Space darf F_Source_Add nicht in anderen F-Hosts wiederverwendet werden. Im gleichen F-Host ist eine Wiederverwendung für mehrere F-Host-Treiber erlaubt. ● Im gleichen Codename-Space darf F_Dest_Add nicht als F_Source_Add verwendet werden und umgekehrt.		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
12.	Wenn SM560-S-FD-1 oder SM560-S-FD-4 verwendet wird, stellen Sie sicher, dass F-Submodule („12 Byte In/Out (PROFIsafe V2.4)“ / „8 Byte and 2 Int In/Out (PROFIsafe V2.4)“ / „12 Byte In/Out (PROFIsafe V2.6)“ / „123 Byte In/Out (PROFIsafe V2.6)“) ordnungsgemäß mit Mastersystemen verbunden sind.		
13.	Prüfen Sie, ob nicht nur die Codenamen, sondern auch F_Dest_Add in PROFIsafe-Netzwerken eindeutig sind, wenn nur F_Dest_Add vom F-Device geprüft wird.		
Prüfer: Maschine/Applikation <ID>: Unterschrift: Datum:			

6.4 Checkliste für Betrieb, Instandhaltung und Reparatur

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
1.	Stellen Sie sicher, dass die Sicherheitsmodule in ihre Positionen im Modulträger für die Sicherheits-CPU oder Klemmenblock für Sicherheits-E/As richtig eingesteckt sind und dass ein einwandfreier Kontakt zwischen Klemmen und Sicherheitsmodulen besteht.		
2	Prüfen Sie, ob geeignete Maßnahmen zur Temperaturüberwachung (z. B. könnten Temperatursensoren im Schaltschrank positioniert werden und mit den Sicherheits-Analogeingangskanälen des AI581-S verbunden werden) in den Schaltschrank implementiert wurden, in dem sich die AC500-S-Sicherheitsmodule befinden, wenn der Betriebstemperaturbereich für die Sicherheitssteuerung AC500-S nicht garantiert werden kann. *Hinweis:* *Sicherheitsgerichtete Digitalausgänge der DX581-S-Module verfügen über einen eingebauten Übertemperaturschutz und liefern immer Failsafe-„0“-Werte bei Übertemperatur.*		
3.	Stellen Sie sicher, dass die folgende Regel laut PROFIsafe-Norm (siehe *www.profisafe.net* für weitere Details) in der Analyse der Sicherheitsanwendung berücksichtigt wurde: ● Maximal 10 Kommunikationsverbindungen (d. h. PROFIsafe-Verbindungen von einem gegebenen Sicherheitseingang zu einem gegebenen Sicherheitsausgang) je Sicherheitsfunktion sind für eine mittlere Wahrscheinlichkeit eines gefährlichen Ausfalls von 10^{-9}/h (SIL 3) zulässig. Bei mehr als 10 Kommunikationsverbindungen je Sicherheitsfunktion steigt die Wahrscheinlichkeit eines gefährlichen Ausfalls auf 10^{-10}/h pro zusätzliche Kommunikationsverbindung. Dementsprechend sind bei SIL 2 maximal 100 Kommunikationsverbindungen zulässig.		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
4.	Stellen Sie sicher, dass alle Netzwerkgeräte, die zusammen mit der Sicherheitssteuerung AC500-S verwendet werden, die Forderungen der IEC 61010 oder IEC 61131-2 (z. B. PELV) erfüllen. Single-Port-Router sind zur Trennung von Sicherheitsinseln nicht zulässig. Weitere Details finden Sie unter *[2]*.		
5	Vor dem Einsatz einer Sicherheitsanwendung mit PROFIsafe, insbesondere bei der Verwendung von Wireless-Komponenten, muss eine Überprüfung eventueller gefährlicher Bedrohungen, wie Abhöraktionen oder Datenmanipulation, durchgeführt werden (weitere Details siehe *[10]*). Prüfen Sie, ob ausreichende Sicherheitsstufen durch die Festlegung von Sicherheitsbereichen mit Sicherheitsschleusen eingerichtet wurden. Gibt es keine Bedrohung, sind keine Sicherheitsmaßnahmen erforderlich. *Hinweis:* *Bis jetzt wurden zwei mögliche Gefahrenquellen für Anwendungen mit Wireless-Komponenten identifiziert [2]:* ● *Beabsichtigte Änderungen der Parameter der F-Devices und des Sicherheitsprogramms;* ● *Angriffe auf die zyklische Kommunikation, z. B. Simulation der Sicherheitskommunikation.*		
6.	Ein kompletter Funktionstest sämtlicher Teile der Sicherheitsanwendung muss durchgeführt werden. Für diesen Test muss die Maschine in ihrem endgültigen Zustand sein, d. h. einschließlich der mechanischen, elektrischen und elektronischen Komponenten, Sensoren, Aktoren und der Software.		
7.	Überprüfen Sie, ob klare Verfahren für Betrieb, Instandhaltung und Reparatur der Sicherheitsanwendung definiert wurden (Organisation, Verantwortlichkeiten, Ersatzteile, Projektdaten-Backup usw.). Hinweis: ● Ein Neustart des Sicherheits-Steuerkreises ist nur zulässig, wenn kein gefährlicher Prozesszustand vorliegt, und nach einer Bedienerquittierung (OA_C). Weitere Details finden Sie unter *[2]*.		
8.	Prüfen Sie, ob ein korrekter elektrischer Kontakt zwischen Sicherheits-E/A-Modulen (AI581-S, DI581-S und DX581-S) und TU582-S-Klemmenblöcken besteht. Befolgen Sie die Montageanleitung für die Sicherheits-E/A-Module *„Montage von DI581-S“ auf Seite 76* *„Montage von DX581-S“ auf Seite 104* *„Montage von AI581-S“ auf Seite 124*.		
9.	Stellen Sie sicher, dass die durchschnittliche Betriebstemperatur der sich in Betrieb befindlichen Sicherheitsmodule (AC500-S und AC500-S-XC) + 40 °C nicht überschreitet (z. B. könnten zur Temperaturüberwachung Temperatursensoren im Schaltschrank positioniert werden und mit den Sicherheits-Analogeingangskanälen des AI581-S verbunden werden).		

Nr.	Zu prüfender Punkt	Erfüllt (ja/nein)?	Kommentar
10.	Prüfen Sie, ob kein automatischer Reboot der Standard-CPU im nicht sicherheitsgerichteten Programm programmiert wurde. Ein automatischer Reboot der Standard-CPU würde zu einem automatischen Neustart der Sicherheits-CPU führen, da sie direkt mit der Standard-CPU verbunden ist. Solch ein automatischer Neustart der Sicherheits-CPU ist in bestimmten Sicherheitsanwendungen nicht zulässig.		
Prüfer: Maschine/Applikation <ID>: Unterschrift: Datum:			

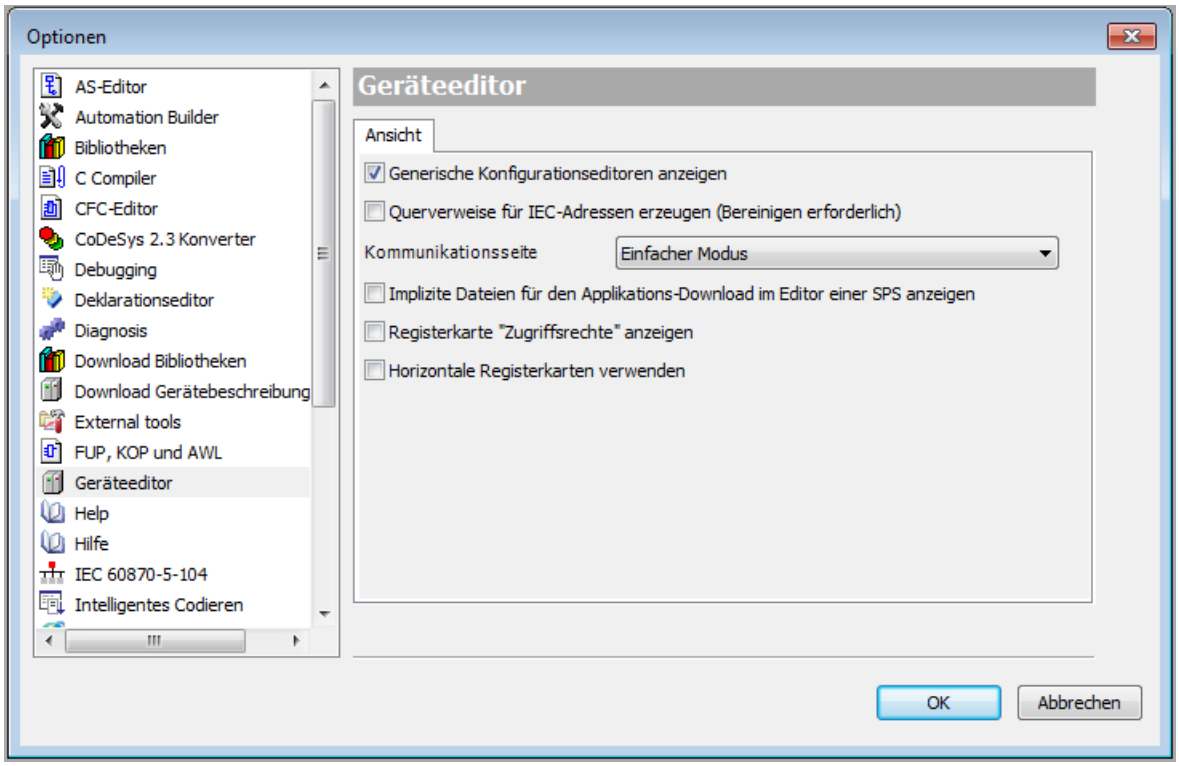
6.5 Verifizierung einer sicheren iParameter-Einstellung in den AC500-S-Sicherheits-E/As

Diese Verifizierung muss vor der Inbetriebnahme der endgültigen Sicherheitsanwendung und vor relevanten Validierungstests für die Bestätigung, dass F_iPar_CRC für einen korrekten iParameter-Satz berechnet wurde, durchgeführt werden.

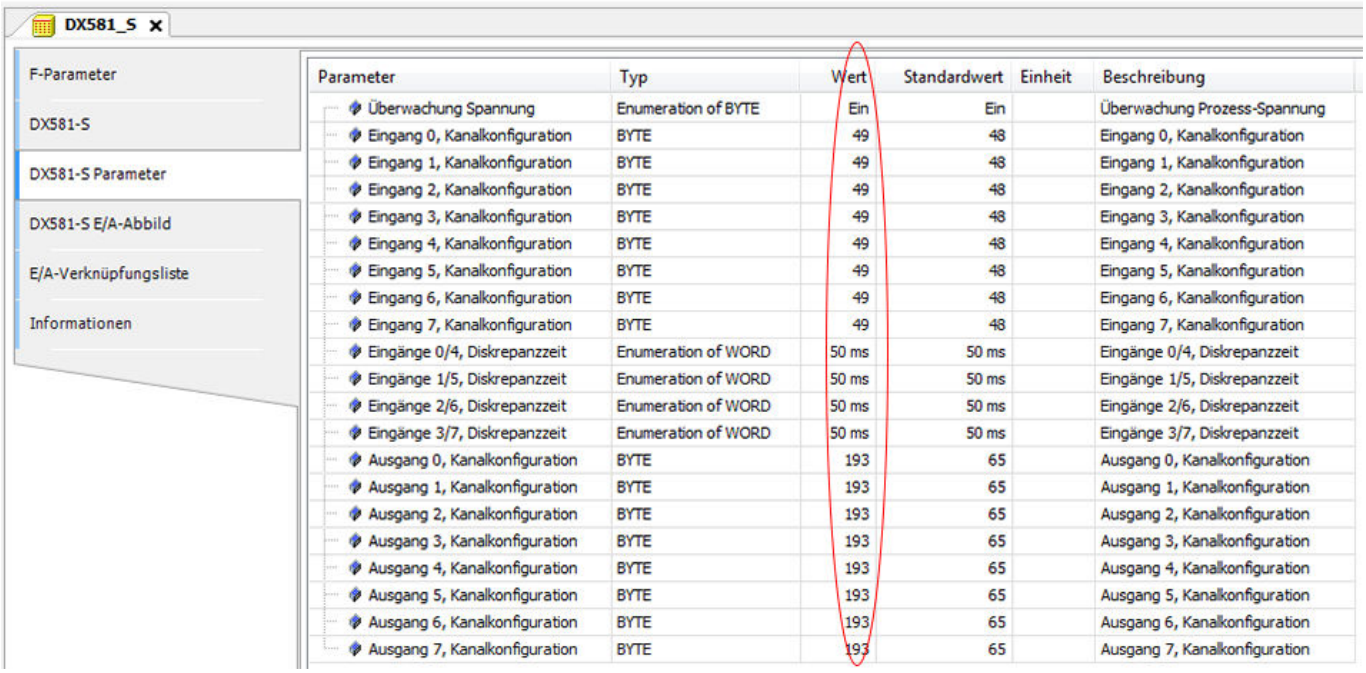
6.5.1 Ablauf des Verifizierungsverfahrens

Personal:
- Sicherheits-Anwendungstechniker für die AC500-S-Sicherheitssteuerung

1. Navigieren Sie im Automation Builder zu *„Tools ➔ Optionen…“*. Aktivieren Sie *„Generische Gerätekonfigurationsansichten anzeigen“* und instanziieren Sie einen bestimmten Typ Sicherheits-E/A-Modul (AI581-S, DI581-S oder DX581-S) in der Strukturansicht von Automation Builder (DX581-S wird als Beispiel verwendet):



2. Öffnen Sie die Registerkarte für die iParameter-Einstellungen des entsprechenden Moduls (*„DX581-S“*, *„DI581-S“* oder *„AI581-S“*) und geben Sie geeignete iParameter-Werte ein (z. B. *„Testimpuls“*, *„Eingangsverzögerung“* usw.)
3. Überprüfen Sie anhand der technischen Spezifikationen für die Sicherheitsanwendung, ob alle iParameter für alle Sicherheits-E/A-Kanäle korrekt gesetzt wurden.
4. Öffnen Sie die Registerkarte *„F-Parameter“* und klicken Sie auf *[Berechnen]*. Kopieren Sie den berechneten F_iPar_CRC-Wert aus dem Feld *„Prüfsumme iParameter“* in das Feld *„F_iPar_CRC“* des F-Parameter-Editors.
5. Öffnen Sie die Registerkarte *„<Name des Sicherheits-E/A-Moduls>-Parameter“* und überprüfen Sie anhand einer Gegenprobe gemäß ↳ *Kapitel 6.5.2 „ Verifizierungstabellen für iParameter-Einstellungen bei AC500-S-Sicherheits-E/As“ auf Seite 384*, ob die zuvor in Schritt 2 vorgenommenen iParameter-Einstellungen dieselben sind wie die, die in der Spalte *„Wert“* für die entsprechenden Kanäle angegeben sind (verwenden Sie ↳ *Kapitel 6.5.2 „ Verifizierungstabellen für iParameter-Einstellungen bei AC500-S-Sicherheits-E/As“ auf Seite 384* zur Umrechnung der Integerwerte in echte Parameterwerte).



Parameter	Typ	Wert	Standardwert	Einheit	Beschreibung
Überwachung Spannung	Enumeration of BYTE	Ein	Ein		Überwachung Prozess-Spannung
Eingang 0, Kanalkonfiguration	BYTE	49	48		Eingang 0, Kanalkonfiguration
Eingang 1, Kanalkonfiguration	BYTE	49	48		Eingang 1, Kanalkonfiguration
Eingang 2, Kanalkonfiguration	BYTE	49	48		Eingang 2, Kanalkonfiguration
Eingang 3, Kanalkonfiguration	BYTE	49	48		Eingang 3, Kanalkonfiguration
Eingang 4, Kanalkonfiguration	BYTE	49	48		Eingang 4, Kanalkonfiguration
Eingang 5, Kanalkonfiguration	BYTE	49	48		Eingang 5, Kanalkonfiguration
Eingang 6, Kanalkonfiguration	BYTE	49	48		Eingang 6, Kanalkonfiguration
Eingang 7, Kanalkonfiguration	BYTE	49	48		Eingang 7, Kanalkonfiguration
Eingänge 0/4, Diskrepanzzeit	Enumeration of WORD	50 ms	50 ms		Eingänge 0/4, Diskrepanzzeit
Eingänge 1/5, Diskrepanzzeit	Enumeration of WORD	50 ms	50 ms		Eingänge 1/5, Diskrepanzzeit
Eingänge 2/6, Diskrepanzzeit	Enumeration of WORD	50 ms	50 ms		Eingänge 2/6, Diskrepanzzeit
Eingänge 3/7, Diskrepanzzeit	Enumeration of WORD	50 ms	50 ms		Eingänge 3/7, Diskrepanzzeit
Ausgang 0, Kanalkonfiguration	BYTE	193	65		Ausgang 0, Kanalkonfiguration
Ausgang 1, Kanalkonfiguration	BYTE	193	65		Ausgang 1, Kanalkonfiguration
Ausgang 2, Kanalkonfiguration	BYTE	193	65		Ausgang 2, Kanalkonfiguration
Ausgang 3, Kanalkonfiguration	BYTE	193	65		Ausgang 3, Kanalkonfiguration
Ausgang 4, Kanalkonfiguration	BYTE	193	65		Ausgang 4, Kanalkonfiguration
Ausgang 5, Kanalkonfiguration	BYTE	193	65		Ausgang 5, Kanalkonfiguration
Ausgang 6, Kanalkonfiguration	BYTE	193	65		Ausgang 6, Kanalkonfiguration
Ausgang 7, Kanalkonfiguration	BYTE	193	65		Ausgang 7, Kanalkonfiguration

6. Öffnen Sie die Registerkarte *„F-Parameter“* und klicken Sie erneut auf *[Berechnen]*, auch wenn der vorherige Wert noch vorhanden ist. Vergleichen Sie die Werte im Feld *„Prüfsumme iParameter“* und im Feld F_iPar_CRC des F-Parameter-Editors; sie müssen gleich sein.

   ⇨ Wenn die F_iPar_CRC-Werte dieselben sind, wurde das Verifizierungsverfahren für die iParameter-Einstellungen des AC500-S-Sicherheits-E/A-Moduls **erfolgreich abgeschlossen.**

Wichtig!

- Wenn in den Schritten 1 … 6 Fehler auftreten (F_iPar_CRC oder iParameter sind nicht gleich), muss der gesamte Prozess erneut durchgeführt werden. Wenn nach dieser Wiederholung immer noch Inkonsistenzen vorliegen, wenden Sie sich an den technischen Support von ABB.
- Wenn iParameter-Werte wie in den Schritten 1 … 6 beschrieben verifiziert werden, können Sie diese iParameter-Kombination mit F_iPar_CRC für weitere Module desselben Typs verwenden, ohne das oben beschriebene Verifizierungsverfahren wiederholen zu müssen.

6.5.2 Verifizierungstabellen für iParameter-Einstellungen bei AC500-S-Sicherheits-E/As

Die Anweisungen unten liefern die Basis für eine Gegenprobe der Werte, die für die iParameter in den Registerkarten *„AI581-S“*, *„DI581-S“* und *„DX581-S“* angegeben wurden.

6.5.2.1 Tabellen für AI581-S-Sicherheits-E/A

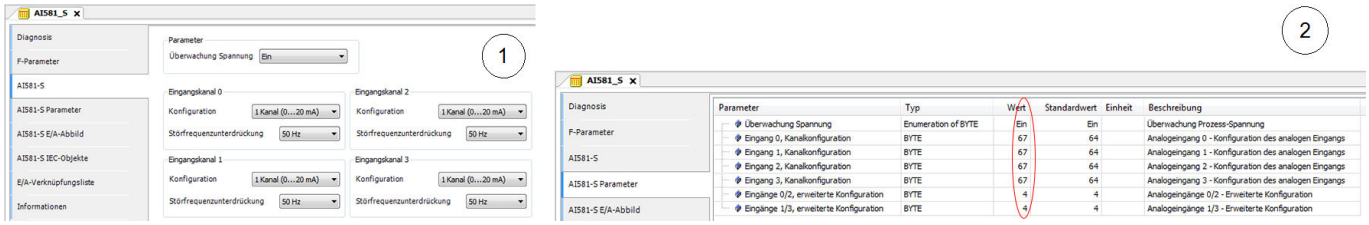


Abb. 126: Die Registerkarte „AI581-S-Parameter“ ist eine Rückleseansicht für iParameter, die in der Registerkarte „AI581-S“ eingestellt wurden.

1 Registerkarte *„AI581-S“*
2 Registerkarte *„AI581-S-Parameter“*

1. Stellen Sie sicher, dass der Parameter *„Überwachung Spannung“* auf den Registerkarten *„AI581-S“* und *„AI581-S-Parameter“* den gleichen Wert hat („Ein“ bzw. „Aus“).
2. Berechnen Sie in der Registerkarte *„AI581-S“* für *„Eingangskanal 0“* das dezimale Äquivalent **(Dec_InputChannel0)** wie folgt:

   Dec_InputChannel0 = Konfiguration_Wert + Störfrequenzunterdrückung_Wert

   wobei:

   Konfiguration_Wert:

   0 ➔ Nicht belegt

   3 ➔ 1 Kanal (0 … 20 mA)

   4 ➔ 1 Kanal (4 … 20 mA)

   5 ➔ 2 Kanal (4 … 20 mA)

   Störfrequenzunterdrückung_Wert:

   0 ➔ Keine

   64 ➔ 50 Hz

   128 ➔ 60 Hz

   Vergleichen Sie den berechneten Wert für **Dec_InputChannel0** mit **„Eingang 0, Kanalkonfiguration”.** Sie müssen gleich sein.

   Wenn sie nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.
3. Wiederholen Sie Schritt 2 für die übrigen Analogeingangskanäle (Eingang 1, 2 und 3).

4. Berechnen Sie in der Registerkarte *„AI581-S“* für **„Analogeingänge 0/2 – erweiterte Konfiguration“** das dezimale Äquivalent **(Dec_ExtConf0_2)** wie folgt:

   Dec_ExtConf0_2 = Toleranzbereich_Wert + Min_Max_Wert

   wobei:

   Toleranzbereich_Wert:

   4 ➔ 4 %

   5 ➔ 5 %

   6 ➔ 6 %

   7 ➔ 7 %

   8 ➔ 8 %

   9 ➔ 9 %

   10 ➔ 10 %

   11 ➔ 11 %

   12 ➔ 12 %

   Min_Max_Wert:

   0 ➔ Min.

   128 ➔ Max.

   Vergleichen Sie den berechneten Wert für **Dec_ExtConf0_2** mit **„Analogeingänge 0/2 – erweiterte Konfiguration“.** Sie müssen gleich sein.

   Wenn sie nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.

5. Wiederholen Sie Schritt 4 für „Analogeingänge 1/3 – erweiterte Konfiguration“.

6.5.2.2 Tabellen für DI581-S-Sicherheits-E/A

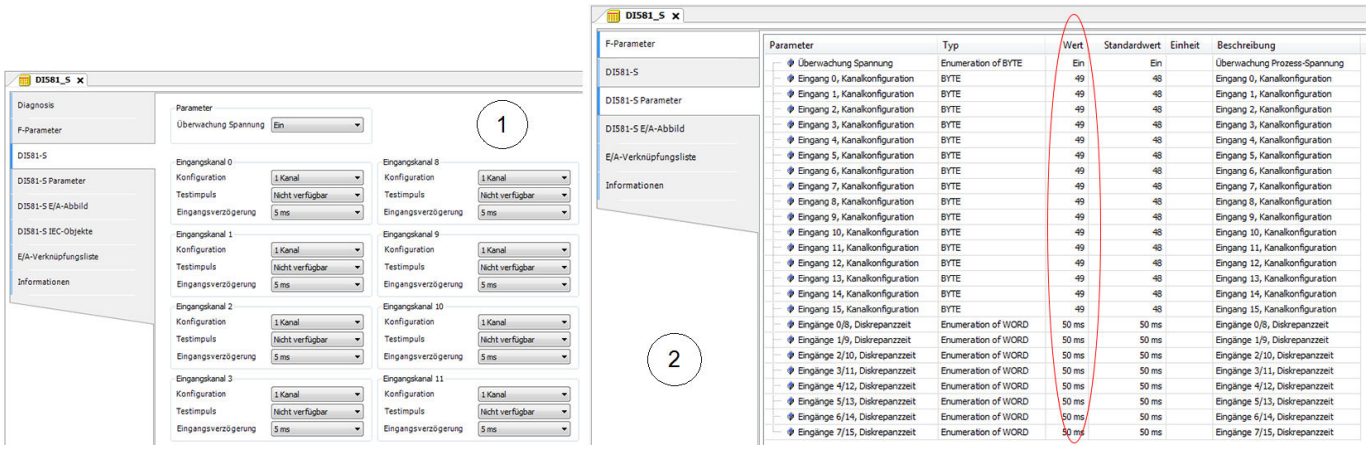


Abb. 127: Die Registerkarte „DI581-S-Parameter“ ist eine Rückleseansicht für iParameter, die in der Registerkarte „DI581-S“ eingestellt wurden.

1 Registerkarte *„DI581-S“*
2 Registerkarte *„DI581-S-Parameter“*

1. Stellen Sie sicher, dass der Parameter *„Überwachung Spannung“* auf den Registerkarten *„DI581-S“* und *„DI581-S-Parameter“* den gleichen Wert hat („Ein“ bzw. „Aus“).

2. Berechnen Sie in der Registerkarte *„DI581-S“* für *„Eingangskanal 0“* das dezimale Äquivalent **(Dec_InputChannel0)** wie folgt:

   Dec_InputChannel0 = Konfiguration_Wert + Testimpuls_Wert + Eingangsverzögerung_Wert

   wobei:

   Konfiguration_Wert:

   0 ➔ Nicht belegt

   1 ➔ 1 Kanal

   2 ➔ 2-Kanal äquivalent

   3 ➔ 2 Kanal antivalent

   Testimpuls_Wert:

   0 ➔ deaktiviert

   8 ➔ aktiviert

   Eingangsverzögerung_Wert:

   16 ➔ 1 ms

   32 ➔ 2 ms

   48 ➔ 5 ms

   64 ➔ 10 ms

   80 ➔ 15 ms

   96 ➔ 30 ms

   112 ➔ 50 ms

   128 ➔ 100 ms

   144 ➔ 200 ms

   160 ➔ 500 ms

   Vergleichen Sie den berechneten Wert für **Dec_InputChannel0** mit **„Eingang 0, Kanalkonfiguration”.** Sie müssen gleich sein.

   Wenn sie nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.

3. Wiederholen Sie Schritt 2 für die übrigen Digitaleingangskanäle (Eingang 1, Eingang 2, … Eingang 15).

4. Stellen Sie sicher, dass der Parameter *„2-Kanalkonfiguration 0/8“* in der Registerkarte *„DI581-S“* den gleichen Wert wie *„Eingänge 0/8, Diskrepanzzeit“* in der Registerkarte *„DI581-S-Parameter“* hat.

   Wenn die Werte nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.

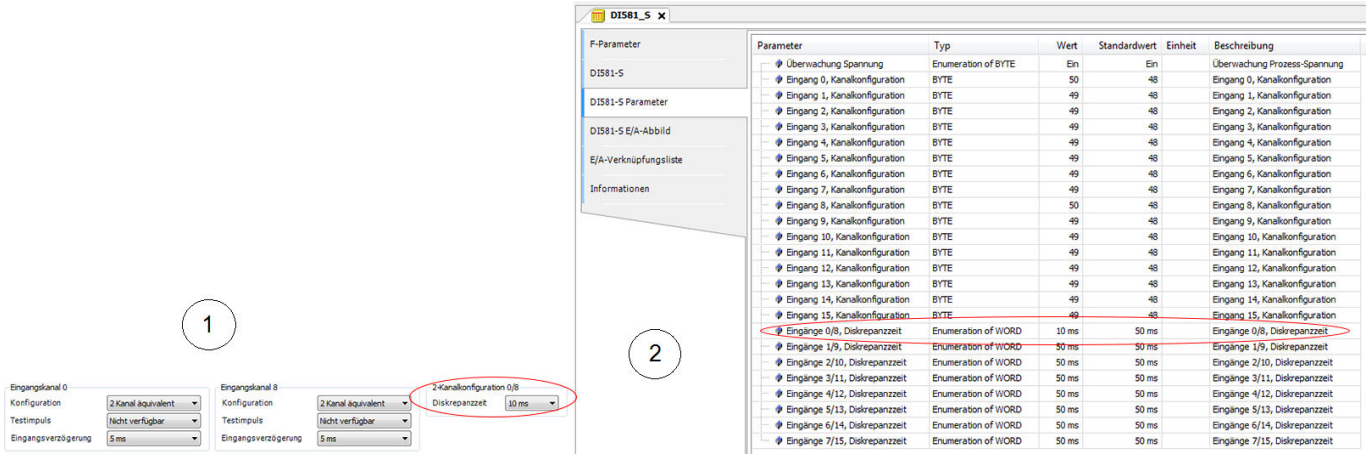


Abb. 128: Vergleichen Sie die Registerkarten „DI581-S“ und „DI581-S-Parameter“.

1 Parameter *„2-Kanalkonfiguration 0/8“* in Registerkarte *„DI581-S“*
2 Parameter *„Eingänge 0/8, Diskrepanzzeit“* in Registerkarte *„DI581-S-Parameter“*

5. Wiederholen Sie Schritt 4 für die übrigen Kanalkombinationen:

   - Eingänge 1/9, Diskrepanzzeit
   - Eingänge 2/10, Diskrepanzzeit
   - Eingänge 3/11, Diskrepanzzeit
   - Eingänge 4/12, Diskrepanzzeit
   - Eingänge 5/13, Diskrepanzzeit
   - Eingänge 6/14, Diskrepanzzeit
   - Eingänge 7/15, Diskrepanzzeit

6.5.2.3 Tabellen für DX581-S-Sicherheits-E/A

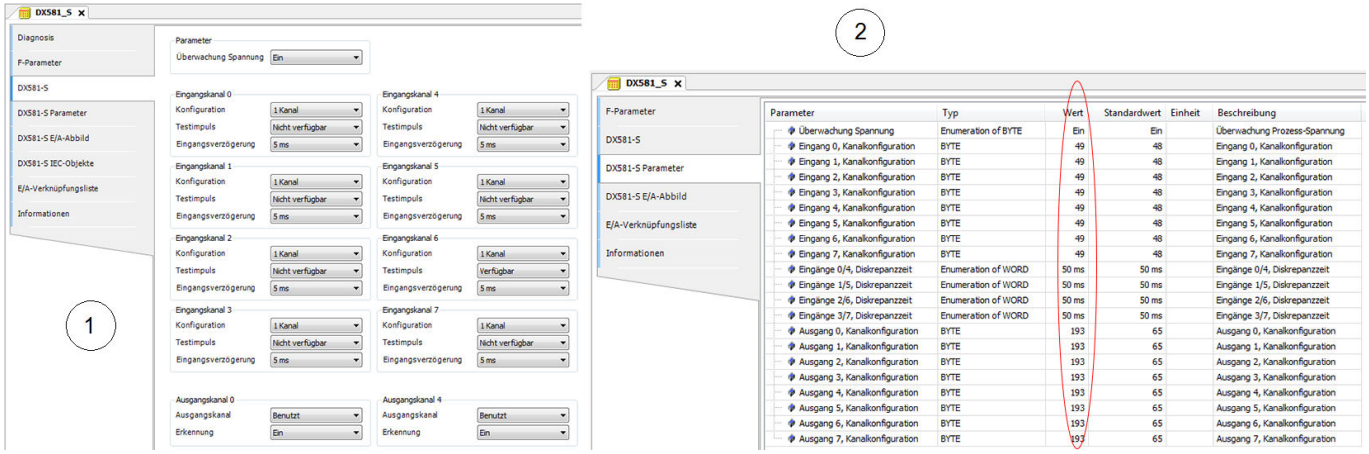


Abb. 129: Die Registerkarte „DX581-S-Parameter“ ist eine Rückleseansicht für iParameter, die in der Registerkarte „DX581-S“ eingestellt wurden.

1 Registerkarte *„DX581-S“*
2 Registerkarte *„DX581-S-Parameter“*

1. Stellen Sie sicher, dass der Parameter *„Überwachung Spannung“* auf den Registerkarten *„DX581-S“* und *„DX581-S-Parameter“* den gleichen Wert hat („Ein“ bzw. „Aus“).

2. Berechnen Sie in der Registerkarte *„DX581-S“* für *„Eingangskanal 0“* das dezimale Äquivalent **(Dec_InputChannel0)** wie folgt:

   Dec_InputChannel0 = Konfiguration_Wert + Testimpuls_Wert + Eingangsverzögerung_Wert

   wobei:

   Konfiguration_Wert:

   0 ➔ Nicht belegt

   1 ➔ 1 Kanal

   2 ➔ 2-Kanal äquivalent

   3 ➔ 2 Kanal antivalent

   Testimpuls_Wert:

   0 ➔ deaktiviert

   8 ➔ aktiviert

   Eingangsverzögerung_Wert:

   16 ➔ 1 ms

   32 ➔ 2 ms

   48 ➔ 5 ms

   64 ➔ 10 ms

   80 ➔ 15 ms

   96 ➔ 30 ms

   112 ➔ 50 ms

   128 ➔ 100 ms

   144 ➔ 200 ms

   160 ➔ 500 ms

   Vergleichen Sie den berechneten Wert für **Dec_InputChannel0** mit **„Eingang 0, Kanalkonfiguration”.** Sie müssen gleich sein.

   Wenn sie nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.

3. Wiederholen Sie Schritt 2 für die übrigen Digitaleingangskanäle (Eingang 1, Eingang 2, … Eingang 7).

4. Stellen Sie sicher, dass der Parameter *„2-Kanalkonfiguration 0/4“* in der Registerkarte *„DX581-S“* den gleichen Wert wie *„Eingänge 0/4, Diskrepanzzeit“* in der Registerkarte *„DX581-S-Parameter“*.

   Wenn die Werte nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.

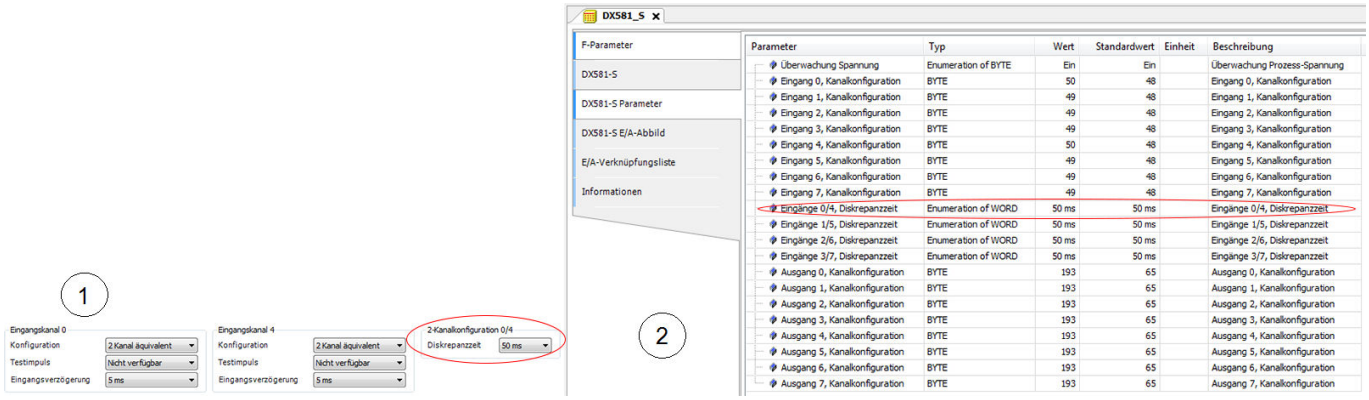


Abb. 130: Vergleichen Sie die Registerkarten „DX581-S“ und „DX581-S-Parameter“.

1 Parameter *„2-Kanalkonfiguration 0/4“* in Registerkarte *„DX581-S“*
2 Parameter *„Eingänge 0/4, Diskrepanzzeit“* in Registerkarte *„DX581-S-Parameter“*

5. Wiederholen Sie Schritt 4 für die übrigen Eingangskanalkombinationen:
   - Eingänge 1/5, Diskrepanzzeit
   - Eingänge 2/6, Diskrepanzzeit
   - Eingänge 3/7, Diskrepanzzeit
6. Berechnen Sie in der Registerkarte *„DX581-S“* für **„Ausgangskanal 0“** das dezimale Äquivalent (Dec_OutputChannel0) wie folgt:

   Dec_OutputChannel0 = Erkennung_Wert + Ausgangskanal_Wert + 1

   wobei:

   Erkennung_Wert:

   0 ➔ Aus

   64 ➔ Ein

   Ausgangskanal_Wert:

   0 ➔ Nicht belegt

   128 ➔ Belegt

   Vergleichen Sie den berechneten Wert für **Dec_OutputChannel0** mit **„Ausgang 0, Kanalkonfiguration“.** Sie müssen gleich sein.

   Wenn sie nicht gleich sind, stoppen Sie den Prozess, passen Sie die Konfiguration an und vergleichen Sie erneut.

   Wenn nach dem zweiten Versuch immer noch ein Unterschied zwischen diesen Werten besteht, stoppen Sie die Verifizierung und wenden Sie sich an den technischen Support von ABB.
7. Wiederholen Sie Schritt 6 für die übrigen Digitalausgangskanäle (Kanal 1, Kanal 2, … Kanal 7).

7 Beispiele für Sicherheitsanwendungen

7.1 Übersicht

In diesem Kapitel werden Anwendungsbeispiele vorgestellt, in denen POEs von PLCopen Safety verwendet werden. Das Hauptziel dabei ist, zu erklären, wie diese POEs von PLCopen Safety in typischen Sicherheitsanwendungen verwendet werden können. Beispiele werden mit Genehmigung der PLCopen-Organisation laut *[6]* zitiert.

Initialisierungsverfahren für PROFIsafe-Startverhalten und AC500-S-spezifische POEs sind in diesen Beispielen nicht aufgeführt, müssen aber in das endgültige Sicherheitsprogramm mit aufgenommen werden.

Als Beispiel für die Verwendung von Sicherheitsfunktionen wird die folgende Fertigungsanlage verwendet. Die unten beschriebenen Funktionsbausteine von PLCopen können für die einfache Erstellung eines Sicherheitsprogramms für diese Fertigungsanlage verwendet werden.

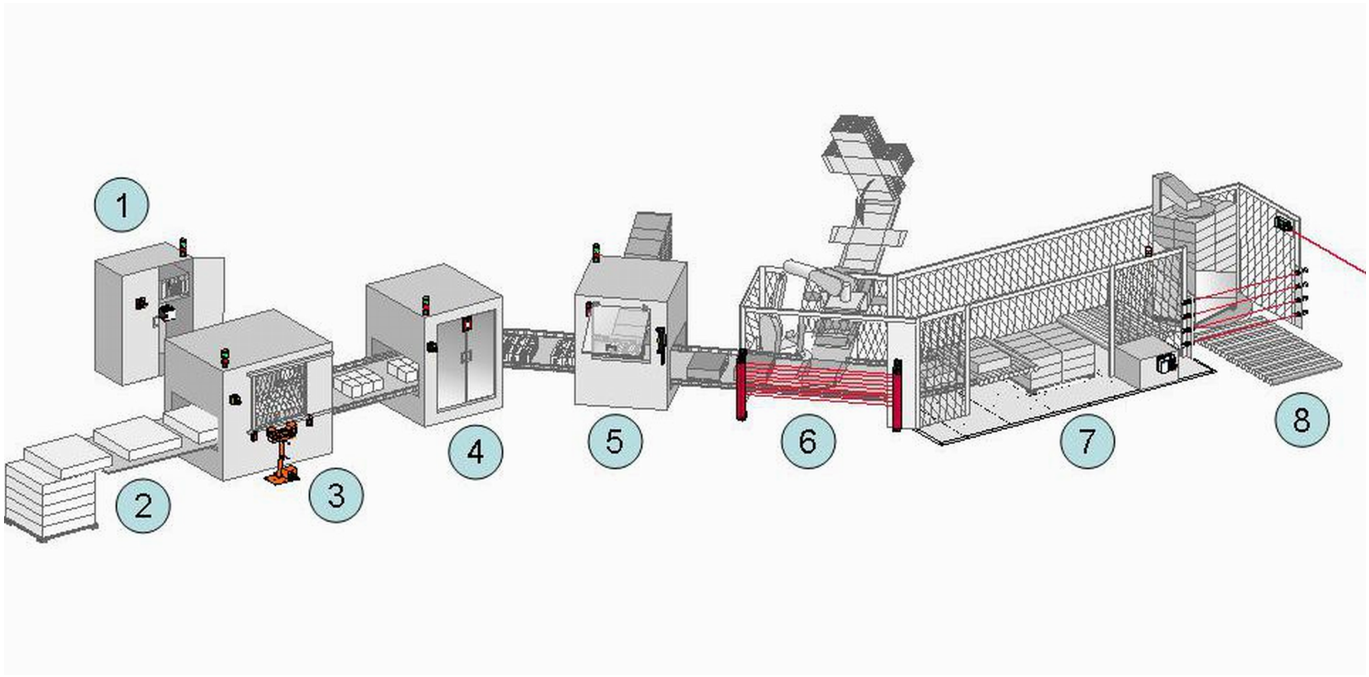


Abb. 131: Beispiel der Sicherheitsfunktionen in einer Fertigungslinie

1 Zentraler Schaltschrank mit dem Sicherheitsteil des Steuerungssystems, in dem die sicherheitsgerichteten Funktionsbausteine laufen.
2 Materialzuführung. In diesem Teil werden dafür keine speziellen Sicherheitsfunktionen verwendet. Man könnte jedoch Muting für eine Trennung zwischen Produkt und Personen einsetzen.
3 Schneiden des Materials. Für die manuelle Steuerung gibt es eine Zweihandbedienung als Sicherheitsfunktion (steht vor der Maschine) zusammen mit einem Zweifach-Tür-Überwachungssystem (befestigt an der Tür der Maschine).
4 Automatische Druckeinheit mit Türüberwachung als Sicherheitsfunktion für Servicearbeiten (an der Tür der Maschine).
5 Erste Kartoniermaschine mit Türüberwachung als Sicherheitsfunktion für Servicearbeiten (an der Tür der Maschine). Manchmal ist ein manueller Betrieb erforderlich. In diesem Fall kann der Bediener die Maschine mit reduzierter Geschwindigkeit betreiben; diese wird mit einem Freigabeschalter gesteuert, welche beim Loslassen die Maschine zu einem sicheren Halt bringt.
6 Zweite Kartoniermaschine mit berührungslos wirkender Schutzeinrichtung (BWS). In diesem Fall handelt es sich hierbei um einen Lichtvorhang.
7 Palettierfunktion, durch Sicherheitsmatten geschützt. Diese Funktionalität könnte mit der BWS-Sicherheitsfunktion gekoppelt werden.
8 Folienverpackungseinheit für die palettierten Produkte mit einem Ausgang der Fertigungsanlage. Dieser Bereich wird von mehreren kombinierten Lichtschranken zusammen mit einer BWS-Sicherheitsfunktion geschützt.

Zusätzlich verfügt jede Einheit über einen Not-Halt-Taster.

7.2 Beispiel 1: Diagnosekonzept

Dieses Beispiel zeigt die Verwendung des Diagnosekonzepts mit einer Reihenschaltung der Funktionsbaustein-Parameter „Activate“ und „Ready“ (eventuell mit einer Vorabbewertung von Hardwarefehlern). In den anderen Beispielen werden die Diagnoseverbindungen nicht aufgeführt ↳ *Kapitel 7.3 „Beispiel 2: Muting“ auf Seite 396* ↳ *Kapitel 7.4 „Beispiel 3: Zweihandschaltung“ auf Seite 400*.

Die Sicherheitsfunktion stoppt einen Antrieb gemäß Stoppkategorie 1 aus IEC 60204-1, wenn ein Not-Halt-Taster gedrückt oder ein Lichtvorhang unterbrochen wird. Die äquivalente Überwachung der 2 Steckverbinder des Not-Halt-Tasters erfolgt in der Sicherheitsanwendung.

In diesem Beispiel werden beide Optionen einer Eingangsevaluierung gezeigt:

- über einen intelligenten Sicherheitseingang
- über den äquivalenten Funktionsbaustein

7.2.1 Funktionsbeschreibung der Sicherheitsfunktionen

In diesem Beispiel werden die folgenden Sicherheitsfunktionen verwendet:

- Das Drücken des Not-Halt-Tasters (über SF_EmergencyStop) oder die Unterbrechung der Lichtschranke im Lichtvorhang (über SF_ESPE) stoppt den Antrieb gemäß Stoppkategorie 1.
- Das Anhalten des elektrischen Antriebs innerhalb einer vordefinierten Zeit wird überwacht (über SF_SafeStop1).
- Der sichere Zustand des Antriebs wird von der Variable S_Stopped angezeigt, die mit der funktionalen Anwendung verbunden ist.
- Nach Anhalten über den Not-Halt-Taster ist ein manuelles Rücksetzen erforderlich (über SF_EmergencyStop).
- Wenn eine Überschreitung der Überwachungszeit (über SF_SafeStop1) erkannt wird, ist vor dem Rücksetzen eine manuelle Fehlerquittierung erforderlich.
- Die 2-kanaligen Kontakte des Not-Halt-Tasters werden überwacht. Ein Fehler wird erkannt, wenn beide Eingänge nicht denselben Zustand aufweisen, nachdem die Diskrepanzzeit abgelaufen ist (über SF_Equivalent).
- Der Funktionsstopp in diesem Beispiel wird als sicherer Halt durch die funktionale Anwendung ausgeführt. Eine Wiederanlaufsperre ist für diesen Halt nicht notwendig.

7.2.2 Graphische Übersicht der Schnittstelle der Sicherheitsanwendung

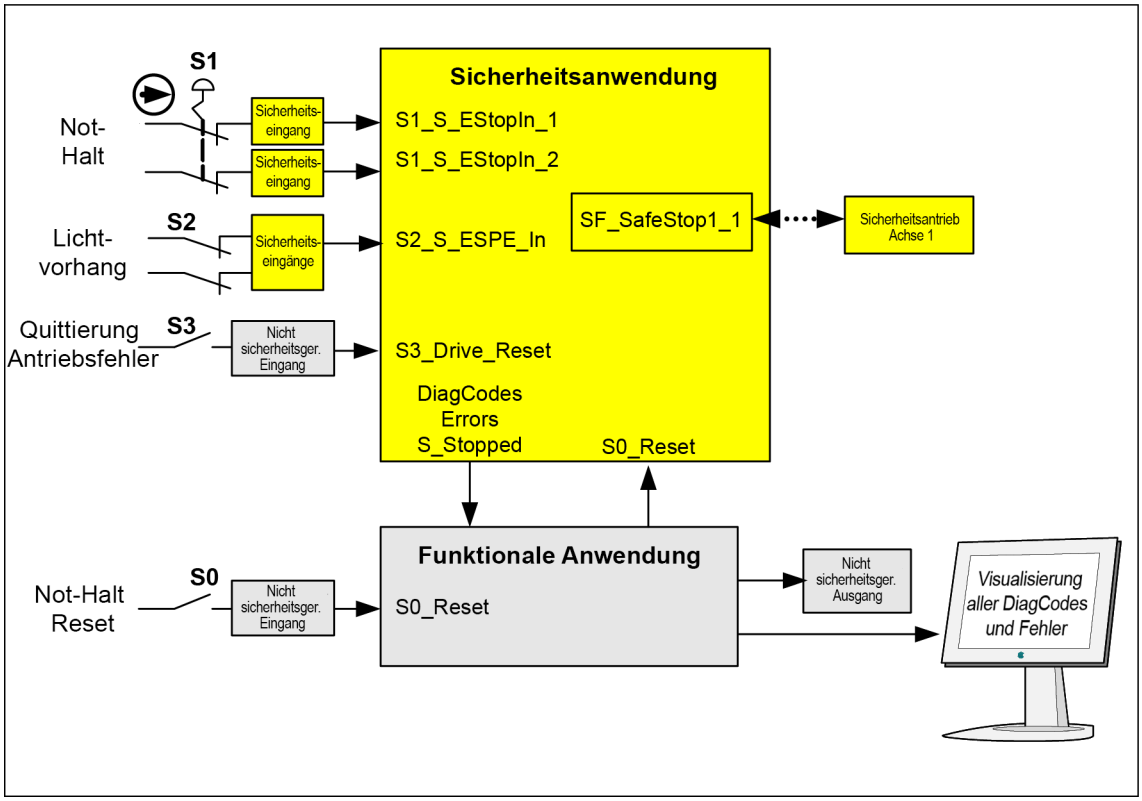


Abb. 132: Graphische Übersicht des Beispiels mit Not-Halt

Das Symbol steht für eine Zwangsöffnung (siehe IEC 60947-5-1).

7.2.3 Deklaration der verwendeten Variablen

Tab. 96: Eingänge

Name	Datentyp	Beschreibung
S1_S_EstopIn_1	BOOL	Not-Halt-Kanal 1
S1_S_EstopIn_2	BOOL	Not-Halt-Kanal 2
S2_ESPE_In	BOOL	Lichtvorhangssignal
S0_Reset	BOOL	Rücksetzen des Not-Halts und der BWS
S3_Drive_Reset	BOOL	Rücksetzen des Antriebsfehlers

Tab. 97: Ausgänge

Name	Datentyp	Beschreibung
S_Stopped	BOOL	Anzeige des sicheren Halts des Antriebs
Errors	BOOL	Repräsentiert alle Fehler des verwendeten Funktionsbausteins (an die funktionale Anwendung)
DiagCodes	WORD	Repräsentiert alle Diagnosecodes des verwendeten Funktionsbausteins (an die funktionale Anwendung)

Tab. 98: Versteckte Schnittstelle der Funktionsbausteininstanzen in Richtung Antrieb (herstellerspezifisch)

Name	Beschreibung
SF_SafeStop1_1	Verbindung mit Antrieb 1

Tab. 99: Lokale Variable

Name	Datentyp	Beschreibung
S_EStopOut	BOOL	Not-Halt-Anforderung
InputDevice1_active	BOOL	Zustand des relevanten Eingabegerätes laut System
InputDevice2_active	BOOL	Zustand des relevanten Eingabegerätes laut System

7.2.4 Programmbeispiel

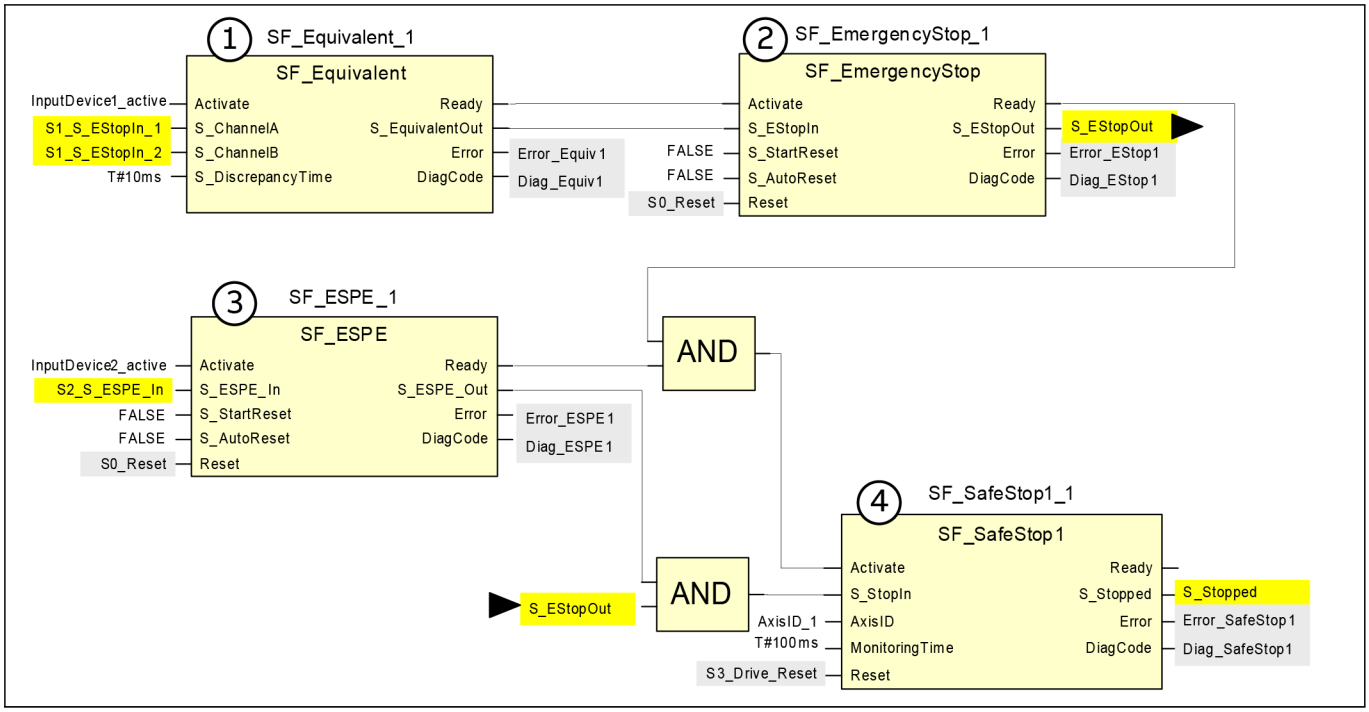


Abb. 133: Programmbeispiel – Not-Halt mit sicherem Halt und Äquivalenzüberwachung

1 2-Kanal-Überwachung: Der Funktionsbaustein SF_Equivalent gibt ein einzelnes BOOL-Signal aus den zwei getrennten Signalen der Not-Halt-Kanäle aus. Die Diskrepanzzeit wird konstant auf 10 ms gesetzt.

2 Not-Halt mit Wiederanlauf-Sperre: Der Funktionsbaustein SF_EmergencyStop verarbeitet die Not-Halt-Bedingung. Nach der Not-Halt-Anforderung und Einschalten wird der Sicherheitsausgang nur nach manuellem Neustart freigegeben. Dies erfolgt, nachdem die Eingänge S_AutoReset und S_StartReset auf FALSE gesetzt wurden.

3 ESPE: Der Funktionsbaustein SF_ESPE verarbeitet die Schnittstelle zu den Lichtvorhängen. Nach Eindringen in das geschützte Feld und Einschalten wird der Sicherheitsausgang nur nach manuellem Neustart freigegeben. Dies erfolgt, nachdem die Eingänge S_StartReset und S_AutoReset auf FALSE gesetzt wurden.

4 Verarbeitung der Anforderung SAFE STOP 1: Der Funktionsbaustein SF_SafeStop1 verarbeitet die SAFE STOP 1-Anforderung für AxisID_1 und überwacht, dass die Achse der Anforderung innerhalb der vordefinierten Überwachungszeit von 100 ms entspricht. Jeder Fehler der Achse muss quittiert werden durch ein manuelles Antriebs-Reset-Signal.

7.2.5 Weitere Hinweise

Dieses Beispiel verwendet verschiedene Reset-Signale zum Quittieren des Not-Halts sowie zum Quittieren bei Überschreitung der Überwachungszeit des Antriebs. Wenn die Sicherheitsanforderung der Anwendung die Quittierung beider Vorfälle mit demselben Signalgeber erlaubt, kann das identische Signal aus der funktionalen Anwendung verwendet werden, um die Funktionsbausteine SF_EmergencyStop_1 und SF_SafeStop1_1 zurückzusetzen.

Informationen über das Diagnosekonzept

Die Vorstellung des Diagnosekonzepts ist rein informativ. Für die Sicherheitsfunktion müssen die dedizierten Sicherheits-E/As verwendet werden.

Reihenschaltung von „Activate“ und „Ready“

Die Verbindung des „Ready“-Ausgangs mit dem „Activate“-Eingang des folgenden Funktionsbausteins stellt sicher, dass keine irrelevante Diagnoseinformation generiert wird, wenn das Gerät deaktiviert ist. Die Reihenschaltung von „Activate“ und „Ready“ vermeidet nachfolgende Fehlermeldungen verbundener Funktionsbausteine.

Vorabbewertung von Hardwarefehlern

Wenn das Zielsystem ein Fehlersignal unterstützt, das den Zustand (aktiv oder nicht aktiv) des relevanten Sicherheitsgerätes darstellt, z. B. InputDevice_active, kann dieses Signal zur Deaktivierung der Sicherheitsfunktionsbausteine verwendet werden. Dadurch wird keine irrelevante Diagnoseinfo generiert, sobald ein Gerät deaktiviert wird. Wenn solch ein Fehlersignal nicht durch das Zielsystem bereitgestellt wird, muss dem „Activate“-Eingang ein statisches TRUE-Signal zugewiesen werden.

Bewertung der Diagnoseinformationen

Die Fehlersignale und DiagCodes jedes Sicherheitsfunktionsbausteins werden in die Standardanwendung übertragen. Diagnoseinformationen können dann von einer angeschlossenen Visualisierung verarbeitet und angezeigt werden. Es gibt verschiedene Möglichkeiten zur Bewertung der Diagnoseinformation:

- Übertragen Sie diese Werte in die Visualisierung und führen Sie die Diagnosebewertung dort durch.
- Führen Sie die Diagnosebewertung in der Standardlogik durch und übertragen Sie die Ergebnisse in die Visualisierung.

Aufgrund der verschiedenen Möglichkeiten und Unterschiede im Zielsystem zur Verarbeitung der Diagnose wird hier kein spezielles Beispiel gezeigt. Eine weitere Bearbeitung der Diagnose könnte wie folgt aussehen:

- Anzeige des Fehlerzustands für jeden Sicherheitsfunktionsbaustein
- Fehlerübersicht, die mit den funktionsbausteinspezifischen Fehleranzeigen verknüpft ist
- Erkennung und Anzeige des letzten Fehlers des in der Sicherheitsanwendung verwendeten Sicherheitsfunktionsbausteins

Information über die verwendeten Funktionsbaustein-Parameter

Funktionsbaustein	Eingang	Konstanter Wert	Beschreibung
SF_Equivalent_1	S_Discrepancy-Time	10 ms	Maximale Überwachungszeit für den Diskrepanzzustand beider Eingänge.
SF_Emergen-cyStop_1	S_StartReset	FALSE	Manuelles Rücksetzen, wenn PES gestartet wird (Warm- oder Kaltstart).
	S_AutoReset	FALSE	Manuelles Rücksetzen, wenn ein Not-Halt-Taster losgelassen wird.
SF_SafeStop1_1	AxisID	AxisID_1	Antriebsadresse, herstellerspezifischer Wert

Funktionsbaustein	Eingang	Konstanter Wert	Beschreibung
	MonitoringTime	100 ms	Zeit, bis der Antrieb gestoppt werden soll.
SF_ESPE	S_StartReset	FALSE	Manuelles Rücksetzen, wenn PES gestartet wird (Warm- oder Kaltstart).
	S_AutoReset	FALSE	Manuelles Rücksetzen, nachdem die Situation, die zur Sicherheitsanforderung führte, behoben wurde.

7.3 Beispiel 2: Muting

Dieses Beispiel beschreibt die Sicherheitsfunktionen für die Absicherung der Fertigungszelle. Objekte werden durch eine durch einen Lichtvorhang geschützte Eingangsschleuse transportiert. Dieser Lichtvorhang kann nur für den Materialtransport in die Zelle überbrückt werden. Der Bediener kann die Zelle durch eine Sicherheitstür betreten. Der Prozess in der Zelle wird durch die funktionale Anwendung gesteuert und durch den Sicherheitskreis freigegeben. Bei einer Sicherheitsanforderung oder einem Fehler werden alle gefährlichen Bewegungen gemäß Stoppkategorie 0 gestoppt.

7.3.1 Funktionsbeschreibung der Sicherheitsfunktionen

Alle gefährlichen Bewegungen werden gestoppt, wenn

- eine Tür geöffnet wird,
- ein Fehler auftritt (z. B. ungültige Muting-Sequenz)
- ein im Muting nicht vorgesehener Lichtvorhang unterbrochen wird (z. B. durch eine Person)
- ein Not-Halt-Taster gedrückt wird.

Durch das Drücken eines Not-Halt-Tasters kann der Bediener auch alle gefährlichen Bewegungen in Stoppkategorie 0 stoppen (über SF_EmergencyStop und folgende Funktionsbausteine).

Bei der Unterbrechung eines im Muting nicht vorgesehenen Lichtvorhangs stoppen alle gefährlichen Bewegungen. In dieser Anwendung wird ein Lichtvorhang Typ 2 verwendet, der eine Überprüfung durch den Funktionsbaustein SF_TestableSafetySensor erfordert.

Für die beschriebene Muting-Funktion werden vier Muting-Sensoren nacheinander verwendet (über SF_MutingSeq). Zusätzlich wird die Muting-Phase durch eine Lampe angezeigt, die überwacht wird (auch über SF_MutingSeq).

Eine zusätzliche Tür für Wartungszwecke wird über einen Türschalter überwacht (über SF_GuardMonitoring).

Durch Quittierungstaster muss der Bediener die erkannte Anforderung der Sicherheitsfunktionen und Fehler quittieren.

Der Anfangs- und Betriebszustand des angeschlossenen Aktors wird über eine externe Geräteüberwachung überprüft. Wenn ein Fehler erkannt wird, kann die Steuerung nicht in Betrieb gehen (über SF_EDM).

Der Prozess und die damit verbundenen Bewegungen in der Fertigungszelle werden von der funktionalen Anwendung gesteuert. Innerhalb der Sicherheitsanwendung wird diese Steuerung durch den oben beschriebenen Sicherheitskreis (über SF_OutControl) freigegeben; sie treibt dann den Aktor über den Sicherheitsausgang an.

7.3.2 Graphische Übersicht der Schnittstelle der Sicherheitsanwendung

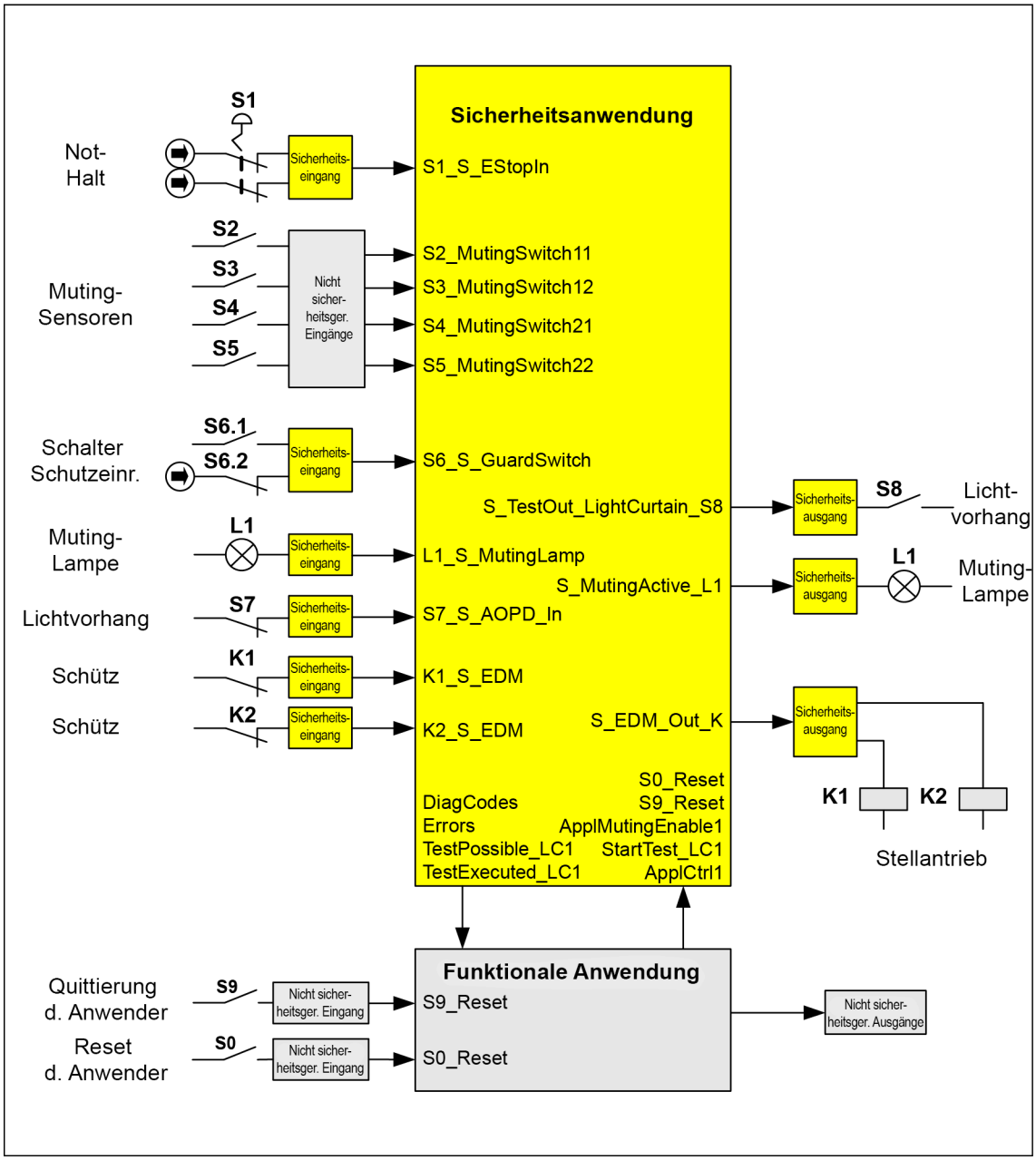


Abb. 134: Graphische Übersicht für den beispielhaften Zugangsschutz an einer Materialschleuse

7.3.3 Deklaration der verwendeten Variablen

Tab. 100: Eingänge

Name	Datentyp	Beschreibung
S1_S_EStopIn	BOOL	Not-Halt-Taster S1
S2_MutingSwitch11	BOOL	Muting-Sensor S2
S3_MutingSwitch12	BOOL	Muting-Sensor S3
S4_MutingSwitch21	BOOL	Muting-Sensor S4
S5_MutingSwitch22	BOOL	Muting-Sensor S5
S6_S_GuardSwitch	BOOL	Türschalter S6 mit zwei Kontakten

Name	Datentyp	Beschreibung
L1_S_MutingLamp	BOOL	Überwachungssignal L1 der Muting-Lampe
S7_S_AOPD_In	BOOL	OSSD vom Lichtvorhang S7
K1_S_EDM	BOOL	Feedback vom externen Gerät K1 (Stellantrieb)
K2_S_EDM	BOOL	Feedback vom externen Gerät K2 (Stellantrieb)
S9_Reset	BOOL	Rücksetzen der Sicherheitsanforderung durch den Anwender S9
S0_Reset	BOOL	Rücksetzen des Fehlers durch den Anwender über S0 (aus der funktionalen Anwendung)
ApplCtrl1	BOOL	Signal, das den Aktor steuert; aktiviert vom Sicherheitskreis (aus der funktionalen Anwendung)
StartTest_LC1	BOOL	Signal S7, das den Test des Lichtvorhangs startet (aus der funktionalen Anwendung)
ApplMutingEnable1	BOOL	Signal, das den Start der Muting-Sequenz aktiviert (aus der funktionalen Anwendung)

Tab. 101: Ausgänge

Name	Datentyp	Beschreibung
S_EDM_Out_K	BOOL	Steuert den Aktor über K1 und K2
S_MutingActive_L1	BOOL	Steuert die Muting-Lampe L1
S_TestOut_LightCurtain_S8	BOOL	Testausgang für Lichtvorhang S8
Errors	BOOL	Repräsentiert alle Fehler des verwendeten Funktionsbausteins (an die funktionale Anwendung)
DiagCodes	WORD	Repräsentiert alle Diagnosecodes des verwendeten Funktionsbausteins (an die funktionale Anwendung)
TestPossible_LC1	BOOL	Zeigt der funktionalen Anwendung an, dass ein automatischer Sensortest des Lichtvorhangs möglich ist.
TestExecuted_LC1	BOOL	Zeigt der funktionalen Anwendung die erfolgreiche Durchführung des automatischen Sensortests des Lichtvorhangs an.

Tab. 102: Lokale Variablen

Name	Datentyp	Beschreibung
S_SafeControl	BOOL	Zeigt den Zustand der Schutzeinrichtungen an (TRUE = Sicherheitsmodus aktiviert)

7.3.4 Programmbeispiel

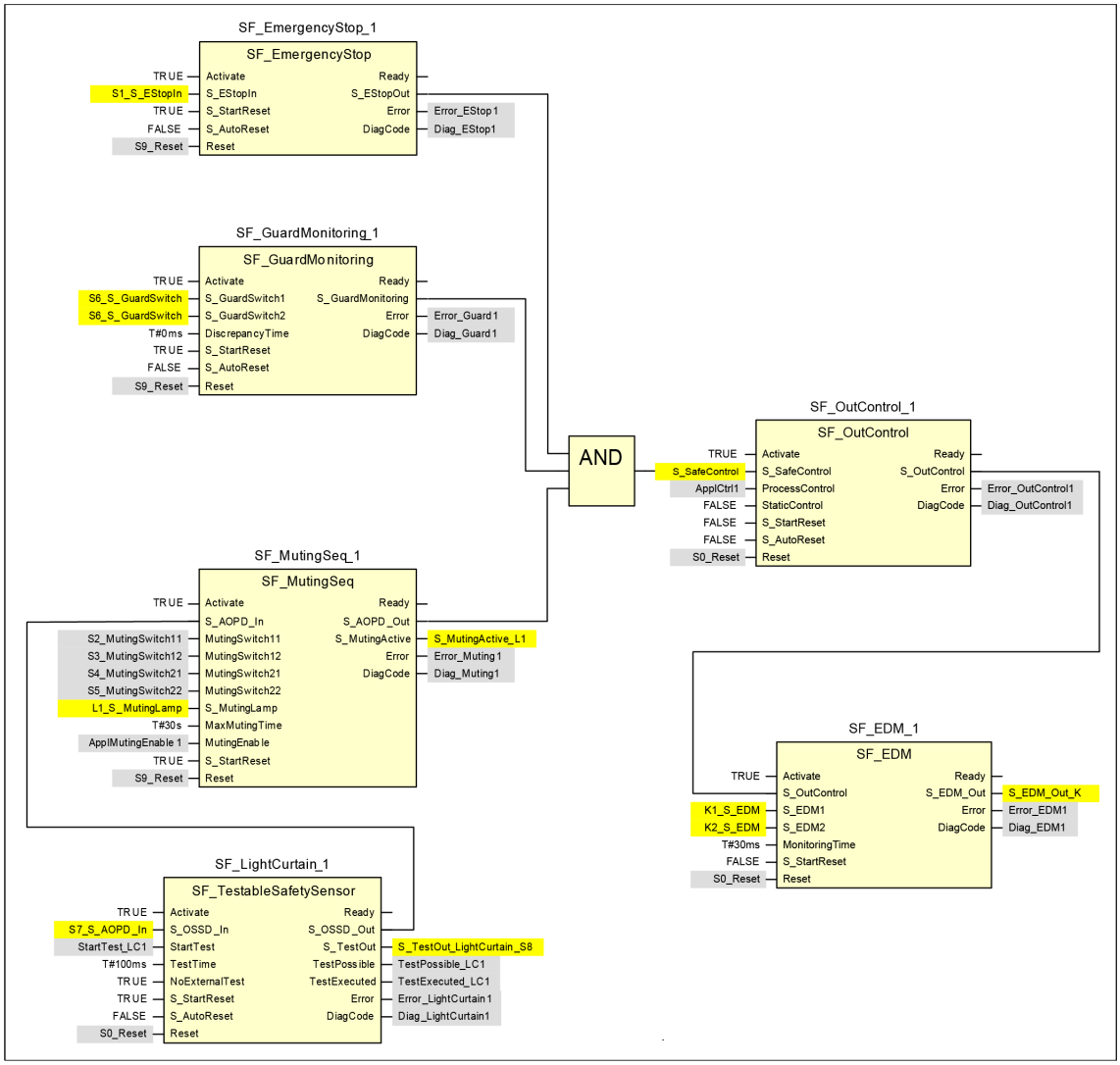


Abb. 135: Zugangsschutz an einer Materialschleuse

7.3.5 Weitere Hinweise

In diesem Beispiel sind die zwei Kontakte des Schaltelements der Schutzeinrichtung mit dem Sicherheitseingabemodul verbunden, welches die Fehlerüberwachung übernimmt. Das resultierende BOOL-Signal ist mit den zwei Eingangskanälen von SF_GuardMonitoring_1 verbunden.

Die Abfrage der Diagnoseinformation wurde in diesem Beispiel nicht behandelt. Hierzu siehe ↳ *Kapitel 7.2.5 „Weitere Hinweise“ auf Seite 395*. Der Eingangsparameter Activate für die dynamische Aktivierung wurde auf TRUE gesetzt. In einer Anwendung kann dafür jedoch eine Variable verwendet werden.

Information über die verwendeten Funktionsbaustein-Parameter

Funktionsbaustein	Eingang	Konstanter Wert	Beschreibung
SF_EmergencyStop_1	S_StartReset	TRUE	Automatisches Rücksetzen zulässig, wenn PES gestartet wird
	S_AutoReset	FALSE	Kein automatisches Rücksetzen; Reset/Quittierung vom Anwender erforderlich
SF_GuardMonitoring_1	S_StartReset	TRUE	Automatisches Rücksetzen zulässig, wenn PES gestartet wird
	S_AutoReset	FALSE	Kein automatisches Rücksetzen; Reset/Quittierung vom Anwender erforderlich
	DiscrepancyTime	T#0ms	Die Diskrepanzzeit zwischen den beiden Sicherheitseingängen S_GuardSwitchX wird nicht überwacht, weil sie identisch sind und das Eingabegerät ein BOOL-Signal von den Schaltelementen liefert.
SF_MutingSeq_1	S_StartReset	TRUE	Automatisches Rücksetzen zulässig, wenn PES gestartet wird
	MaxMutingTime	T#30s	Die maximale Muting-Zeit (30 s) wird überwacht.
SF_LightCurtain_1	S_StartReset	TRUE	Automatisches Rücksetzen zulässig, wenn PES gestartet wird
	S_AutoReset	FALSE	Kein automatisches Rücksetzen; Reset/Quittierung vom Anwender erforderlich
	TestTime	T#100ms	Die maximale Testzeit (100 ms) wird überwacht.
	NoExternalTest	TRUE	Der externe manuelle Sensortest wird nicht unterstützt.
SF_OutControl_1	S_StartReset	FALSE	Kein automatisches Rücksetzen zulässig, wenn PES gestartet wird
	S_AutoReset	FALSE	Kein automatisches Rücksetzen; Reset/Quittierung vom Anwender erforderlich
	StaticControl	FALSE	Eine dynamische Veränderung des Signals ApplCtrl1 (steigende Flanke) ist nach Blockaktivierung oder einer ausgelösten Sicherheitsfunktion erforderlich (S_SafeControl = FALSE).
SF_EDM_Contactor_1	S_StartReset	FALSE	Kein automatisches Rücksetzen zulässig, wenn PES gestartet wird
	MonitoringTime	T#30ms	Die maximale Antwortzeit (30 ms) der beiden Feedback-Signale S_EDM1 und S_EDM2 wird überwacht.

7.4 Beispiel 3: Zweihandschaltung

Dieses Beispiel beschreibt eine Maschine, bei der eine Zweihandschaltung die gefährliche Bewegung startet, solange beide Drucktaster der Zweihandschaltung gedrückt sind und der Prozess ein Freigabesignal liefert.

Die gefährliche Bewegung wird durch das Schließen der zwei aufeinanderfolgenden Schaltelemente initiiert, die über einen Rückführkreis überwacht werden.

7.4.1 Funktionsbeschreibung der Sicherheitsfunktionen

In diesem Beispiel werden die folgenden Sicherheitsfunktionen verwendet:

- Durch das Drücken eines Not-Halt-Tasters müssen alle gefährlichen Bewegungen gestoppt werden (über SF_EmergencyStop). Der Not-Halt hat die höchste Priorität. Nach dem Loslassen des Not-Halt-Tasters ist ein Rücksetzen über S0_Reset erforderlich.
- Wenn beide Drucktaster der Zweihandschaltung gedrückt werden, wird der Sicherheitsausgang aktiviert. Das Loslassen der Drucktaster deaktiviert den Sicherheitsausgang und stoppt die gefährliche Bewegung über die Schaltelemente K1 und K2 (über SF_TwoHandControlTypeII).
- Die Anfangs- und Betriebszustände der angeschlossenen Schaltelemente K1 und K2 werden überwacht, und wenn ein Fehler erkannt wird, ist der Sicherheitsausgang nicht betriebsbereit (über SF_EDM).
- Nach dem Aktivieren der Sicherheits- oder funktionalen Anwendung oder nach einem Not-Halt muss die Zweihandschaltung losgelassen und wieder betätigt werden, um den Sicherheitsausgang wieder zu aktivieren (über SF_OutControl). Um dies für den Neustart der funktionalen Anwendung zu garantieren, ist das Prozesssignal der funktionalen Anwendung mit dem „Activate“-Eingang des Zweihand-Funktionsbausteins THC_S2_S3 verbunden (wenn die Anwendung neu gestartet wird, während die Zweihandschaltung aktiviert ist, geht der Funktionsbaustein in den Zustand C003 und signalisiert einen Fehler, dass beide Drucktaster bei Aktivierung gedrückt sind, was einen Neustart verbietet).

In diesem Beispiel gibt es nur eine Betriebsart.

7.4.2 Graphische Übersicht der Schnittstelle der Sicherheitsanwendung

Die Sicherheitseingänge für die Zweihandschaltung (S2_S_Switch1 und S3_S_Switch2) sind mit der Zweihandschaltung Typ II verbunden.

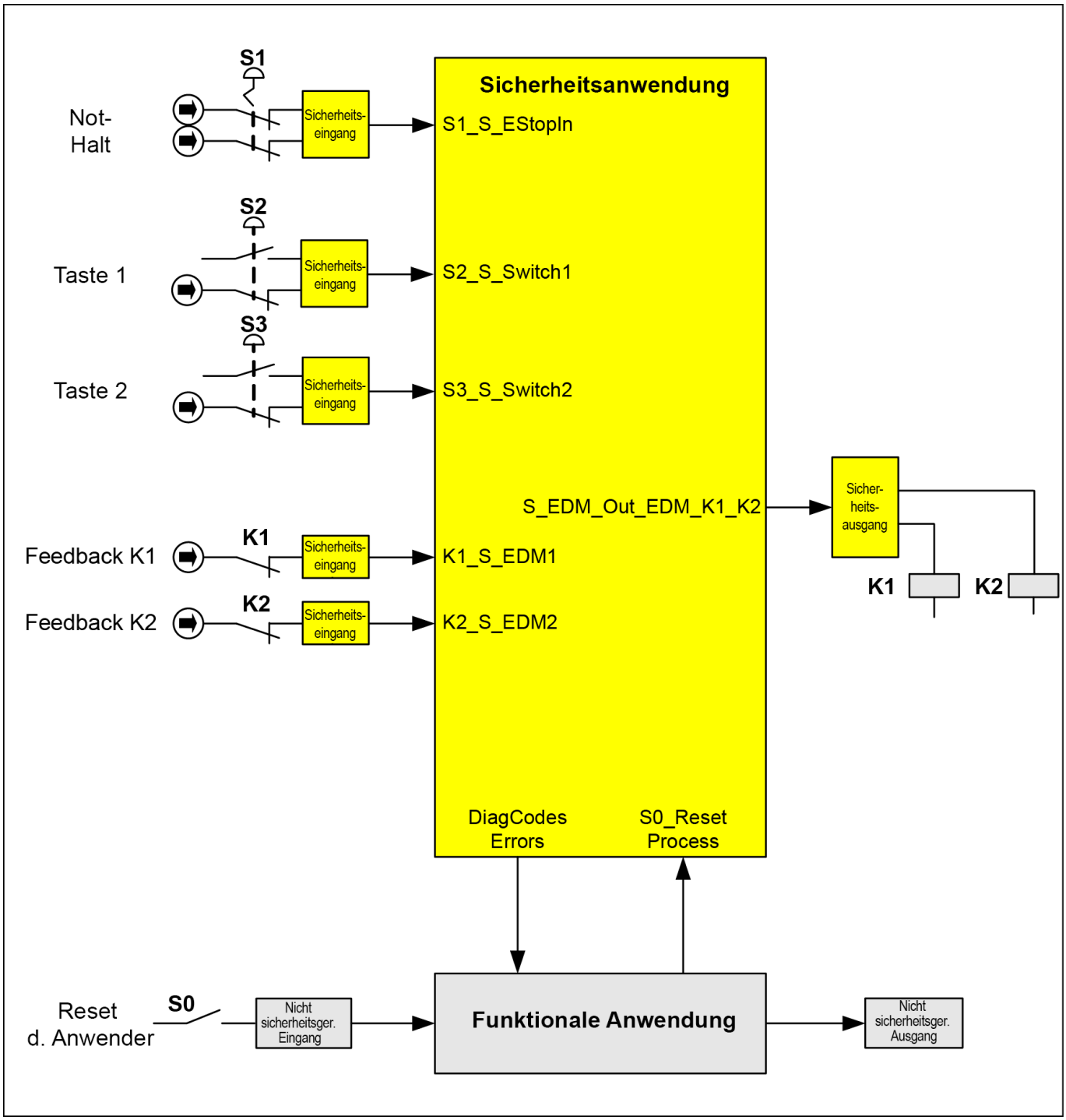


Abb. 136: Graphische Übersicht der beispielhaften Zweihandschaltung mit EDM

7.4.3 Deklaration der verwendeten Variablen

Tab. 103: Eingänge

Name	Datentyp	Beschreibung
S1_S_EStopIn	BOOL	Not-Halt-Taster S1
S2_S_Switch1	BOOL	Schaltelement S2 für Drucktaster 1 der Zweihandschaltung
S3_S_Switch2	BOOL	Schaltelement S3 für Drucktaster 2 der Zweihandschaltung
K1_S_EDM1	BOOL	Feedback vom externen Gerät K1 (Stellantrieb)

Name	Datentyp	Beschreibung
K2_S_EDM2	BOOL	Feedback vom externen Gerät K2 (Stellantrieb)
S0_Reset	BOOL	Rücksetzen durch den Anwender über Schaltelement S0 (aus der funktionalen Anwendung)
Vorgang	BOOL	Aktivieren der Bewegung durch den Prozess (aus der funktionalen Anwendung)

Tab. 104: Ausgänge

Name	Datentyp	Beschreibung
S_EDM_Out_EDM_K1_K2	BOOL	Steuert den Aktor über K1 und K2
Errors	BOOL	Repräsentiert alle Fehler des verwendeten Funktionsbausteins (an die funktionale Anwendung)
DiagCodes	WORD	Repräsentiert alle Diagnosecodes des verwendeten Funktionsbausteins (an die funktionale Anwendung)

7.4.4 Programmbeispiel

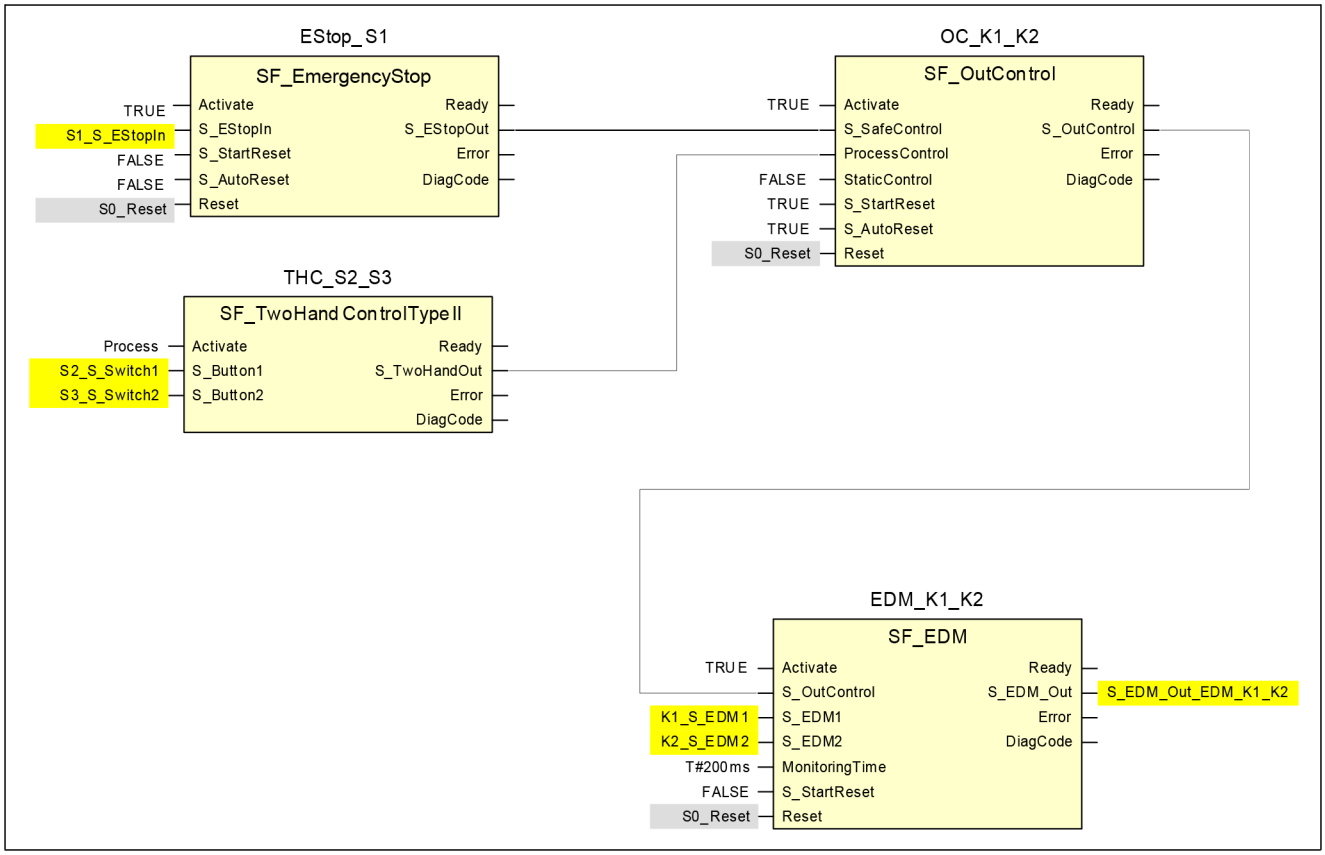


Abb. 137: Anwendungsprogramm der Zweihandschaltung mit EDM

7.4.5 Weitere Hinweise

Dieses Beispiel kann auch mit SF_TwoHandControlTypeIII verwendet werden.

Die Abfrage der Diagnoseinformation wurde in diesem Beispiel nicht behandelt. Hierzu siehe ↳ *Kapitel 7.2.5 „Weitere Hinweise“ auf Seite 395*. Der Eingang „Activate“ wurde auf TRUE gesetzt. In einer Anwendung kann dafür jedoch eine Variable verwendet werden.

Information über die verwendeten Funktionsbaustein-Parameter

Funktionsbaustein	Eingang	Konstanter Wert	Beschreibung
EStop_S1	S_StartReset	FALSE	Kein automatisches Rücksetzen, wenn PES gestartet wird
	S_AutoReset	FALSE	Kein automatisches Rücksetzen; Reset/Quittierung vom Anwender erforderlich
OC_K1_K2	S_StartReset	TRUE	Automatisches Rücksetzen zulässig, wenn PES gestartet wird
	S_AutoReset	TRUE	Automatisches Rücksetzen; kein Reset/ keine Quittierung vom Anwender erforderlich
	StaticControl	FALSE	Eine dynamische Veränderung des Signals Appl_Control (steigende Flanke) ist nach Blockaktivierung oder einer ausgelösten Sicherheitsfunktion erforderlich (S_SafeControl = FALSE).
EDM_K1_K2	S_StartReset	FALSE	Kein automatisches Rücksetzen, wenn PES gestartet wird
	MonitoringTime	T#200ms	Die maximale Antwortzeit (200 ms) der beiden Feedback-Signale S_EDM1 und S_EDM2 wird überwacht.

8 Index

G

I

K

L

M

P

R

S

T

U

V

W

Z

Anhang

A Systemdaten für AC500-S-XC

A.1 Umgebungsbedingungen

Prozess- und Versorgungs-spannungen

Angabe	Wert	Einheit
Prozess- und Versorgungsspannung (-25 %, +30 % inklusive Restwelligkeit)	24	V DC
Absolute Grenzwerte inklusive Restwelligkeit	18 … 31,2	V
Restwelligkeit	< 10	%
Verpolschutz	Ja	
Zulässige Unterbrechungen der Gleichstromversorgung	< 10	ms
Zeit zwischen 2 Unterbrechungen, PS2	> 1	s

GEFAHR!

Das Überschreiten der zulässigen Prozess- oder Versorgungsspannung (< -35 V DC bzw. > +35 V DC) kann zu irreparablen Schäden am System führen.

GEFAHR!

Zur Versorgung der Module müssen Netzteile gemäß PELV- oder SELV-Spezifikationen verwendet werden.

HINWEIS!

Die Kriech- und Luftstrecken entsprechen der Überspannungskategorie II, Verschmutzungsgrad 2.

Temperatur

Angabe	Wert	Einheit
Betriebstemperatur*	-40 … +70	°C
Betriebstemperatur (vertikale Montage des Moduls, Ausgangslast beschränkt auf 50 % pro Gruppe)	-40 … +40	°C
Lagerungstemperatur	-40 … +85	°C
Transporttemperatur	-40 … +85	°C

* +60 … +70 °C mit den folgenden Leistungsreduzierungen:

- Modulträger: maximal 2 Kommunikationsmodule zulässig
- Digitaleingänge: maximale Anzahl simultan geschalteter Digitaleingänge auf Eingangskanäle auf 50 % je Gruppe beschränkt (z. B. 8 Kanäle => 4 Kanäle)
- Digitalausgänge: maximaler Ausgangsstrom (alle Kanäle zusammen) beschränkt auf 50 % je Gruppe (z. B. 4 A => 2 A)
- Analogeingänge: Keine Einschränkungen

GEFAHR!

Bei der Berechnung der Sicherheitswerte wird von der durchschnittlichen Temperatur ausgegangen. Die durchschnittliche Temperatur für den erweiterten Temperaturbereich (-40 °C … +70 °C) sowie den Normaltemperaturbereich (0 °C … +60 °C) ist auf +40 °C definiert.

Stellen Sie sicher, dass die durchschnittliche Betriebstemperatur für in Betrieb befindliche AC500-S- und AC500-S-XC-Module +40 °C nicht überschreitet.

Feuchte

Angabe	Wert	Einheit
Relative Feuchte mit Betauung (Betrieb/Lagerung)	100	%

Luftdruck

Angabe	Wert	Einheit
Betriebsluftdruck	1080 … 620	hPa
Betriebshöhe	-1000 … 4000	m
Reduktion der Betriebstemperatur bei Luftdrücken von < 795 hPa (oder > 2000 m über NN)	10 (z. B. +70 °C bis +60 °C)	K

Beständigkeit gegenüber Korrosivgasen

Angabe	Wert
Betrieb: nach ISA S71.04.1985 Harsh-Gruppe A, G3/GX IEC 60721-3-3 3C2 / 3C3	Ja

Beständigkeit gegenüber Salznebel

Angabe	Wert
Betrieb: nur horizontale Montage, nach IEC 60068-2-52 Schweregrad: 1	Ja

Elektromagnetische Verträglichkeit

Angabe	Wert
Abgestrahlte Emission (Funkstörung) nach CISPR 16-2-3	Ja
Leitungsgeführte Emission (Funkstörung) nach CISPR 16-2-1, CISPR 16-1-2	Ja
Elektrostatische Entladung (ESD) nach IEC 61000-4-2, Zone B, Kriterium B	Ja
Schnelle vorübergehende Störspannungen (Burst) nach IEC 61000-4-4, Zone B, Kriterium B	Ja
Energiereiche vorübergehende Störspannungen (Surge) nach IEC 61000-4-5, Zone B, Kriterium B	Ja
Einfluss von Störstrahlung nach IEC 61000-4-3, Zone B, Kriterium A	Ja
Einfluss von leitungsgeführten Störungen nach IEC 61000-4-6, Zone B, Kriterium A	Ja
Einfluss von Netzfrequenz-Magnetfeldern nach IEC 61000-4-8, Zone B, Kriterium A	Ja

HINWEIS!

Zur Vorbeugung von Störungen wird empfohlen, dass sich das Bedienpersonal vor dem Anfassen der Kommunikations-Steckverbinder entlädt oder andere geeignete Maßnahmen trifft, um die Auswirkungen von elektrostatischer Entladung zu reduzieren.

HINWEIS!

In nicht verwendete Anschlüsse für Kommunikationsmodule auf den Modulträgern müssen Dummy-Kommunikationsmodule TA524 eingesteckt werden. E/A-Busanschlüsse dürfen während des Betriebs nicht berührt werden.

A.2 Mechanische Daten

Angabe	Wert
Anschlusstechnik	Federzugklemmen
Schutzart	IP 20
Vibrationsfestigkeit nach IEC 61131-2, IEC 60068-2-6, IEC 60068-2-64	Ja
Stoßfestigkeit nach IEC 60068-2-27	Ja
Horizontale Einbaulage	Ja
Vertikale Einbaulage (keine Anwendung in Umgebungen mit Salznebel)	Ja

Montage auf Hutschiene nach IEC 60715

Angabe	Wert	Einheit
Hutschienentyp	35	mm
Tiefe Hutschienentyp	7,5 oder 15	mm

Montage mit Schrauben

Angabe	Wert	Einheit
Schraubendurchmesser	4	mm
Anzugs-Drehmoment	1,2	Nm

A.3 Umweltprüfungen

Lagerung	IEC 60068-2-1 Prüfverfahren Ab: Kältefestigkeitsprüfung -40 °C / 16 h IEC 60068-2-2 Prüfverfahren Bb: Trockene Wärmefestigkeitsprüfung +85 °C / 16 h
Feuchte	IEC 60068-2-30 Prüfverfahren Dd: Zyklisch (12 h / 12 h) Feuchte Wärme +55 °C, 93 % relative Feuchte / +25 °C, 95 % relative Feuchte, 6 Zyklen IEC 60068-2-78, Feuchte Wärme, konstant: +40 °C, 93 % relative Feuchte, 240 h
Isolationsprüfung	IEC 61131-2
Vibrationsfestigkeit	IEC 61131-2 / IEC 60068-2-6: 5 Hz … 500 Hz, 2 g (mit in Standard-CPU eingesteckter SD-Speicherkarte) IEC 60068-2-64: 5 Hz … 500 Hz, 4 g RMS
Stoßfestigkeit	IEC 60068-2-27: alle 3 Achsen 15 g, 11 ms, Halbsinus

Elektromagnetische Verträglichkeit

Elektrostatische Entladung (ESD)

Angabe	Wert	Einheit
Elektrostatische Spannung bei Luftentladung	8	kV
Elektrostatische Spannung bei Kontaktentladung	6	kV

Schnelle vorübergehende Störspannungen (Burst)

Angabe	Wert	Einheit
Spannungsversorgungseinheiten (DC)	4	kV
Digitale Ein-/Ausgänge (24 V DC)	2	kV
Analoge Ein-/Ausgänge	2	kV
Geschirmte Kommunikationsleitungen	2	kV
E/A-Versorgung (DC Ausgang)	2	kV

Energiereiche vorübergehende Störspannungen (Surge) – Gleichtakt (CM)

Angabe	Wert	Einheit
Spannungsversorgungseinheiten (DC)	1	kV
Digitale Ein-/Ausgänge (24 V DC)	1	kV
Analoge Ein-/Ausgänge	1	kV
Geschirmte Kommunikationsleitungen	1	kV
E/A-Versorgung (DC Ausgang)	0,5	kV

Energiereiche vorübergehende Störspannungen (Surge) – Gegentakt (DM)

Angabe	Wert	Einheit
Spannungsversorgungseinheiten (DC)	0,5	kV
Digitale Ein-/Ausgänge (24 V DC)	0,5	kV
Analoge Ein-/Ausgänge	0,5	kV
E/A-Versorgung (DC Ausgang)	0,5	kV

Angabe	Wert	Einheit
Einfluss von Störstrahlung: Test-Feldstärke	10	V/m

Angabe	Wert	Einheit
Einfluss von leitungsgeführten Störungen: Prüfspannung	10	V
Netzfrequenz-Magnetfelder bei 30 A/m	50 und 60	Hz

HINWEIS!

Extreme Umweltbedingungen und relevante Anforderungen für verwendete Standard-CPUs und Standard-E/A-Module der Produktfamilie AC500-XC müssen berücksichtigt werden *[3]*.

B Verwendung von Sicherheits-CPU mit AC500 V2-Standard-CPU PM5xx

B.1 Kompatibilität mit AC500 V2-Standard-CPU

Alle Kompatibilitätsinformationen sind für normale sowie für XC-Geräte gültig.

Tab. 105: Kompatibilität für Sicherheits-CPU mit AC500 Standard-CPU AC500 V2

Sicherheits-CPU	**SM560-S**	**SM560-S-FD-1, SM560-S-FD-4**
Firmware-Version von Sicherheits-CPU	Beliebig	Ab V2.0.0
Standard-CPU	Jede V2-CPU, mit Ausnahme von AC500-eCo-CPUs	Jede V2-CPU, mit Ausnahme von AC500-eCo-CPUs
Firmware-Version von Standard-CPU	Ab V2.2.1	Ab V2.7
Version der Engineering Suite Automation Builder	1.0 oder höher	2.1 oder höher
Version der Engineering Suite Control Builder Plus	Ab V2.2.1	Nicht kompatibel

Tab. 106: Kompatibilität für AC500-S mit Standardkomponenten mit Ausnahme von CPUs

Komponente	**SM560-S**	**SM560-S-FD-1, SM560-S-FD-4**
Firmware-Version von Kommunikationsmodul CM579-PNIO	Ab V2.6.5.1	Ab V2.6.5.1
Firmware-Version von Kommunikationsmodul CM589-PNIO(-4)	Nicht zutreffend	Ab V1.6.2.20
Firmware-Version von Kommunikationsschnittstellen-Modul CI501-PNIO, CI502-PNIO, CI504-PNIO, CI506-PNIO	Ab V3.2.0	Ab V3.2.0

B.2 Fehlermeldungen mit AC500 Standard-CPU V2

HINWEIS!

Die Fehlermeldungen der Sicherheits-CPU werden im Diagnose-Stack der Standard-CPU zusammengefasst.

Mit den Befehlen `diagreset`, `diagack all`, `diagack x`, `diagshow all` und `diagshow x` im nicht sicherheitsgerichteten SPS-Browser können Sie verschiedene Fehlermeldungen in einem AC500-System auflisten und verarbeiten, einschließlich der Meldungen von der Sicherheits-CPU. Ausführliche Informationen zu diesen Befehlen finden Sie unter *[3]*.

Bei Verwendung der IO-Device-Kommunikationsmodule CM589-PNIO oder CM589-PNIO-4 können auch PROFINET-Diagnosemeldungen für F-Devices an SM560-S-FD-1 und SM560-S-FD-4 erzeugt werden *Tab. 108 „Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4 “ auf Seite 422* *Tab. 109 „Abbild der AC500/AC500-S-Fehler auf PROFINET-Kanalfehler“ auf Seite 423*.

B.2.1 Fehlermeldungen für Sicherheits-CPUs

Die Fehler werden so angezeigt, wie sie im Automation Builder dargestellt sind.

Tab. 107: Häufige Fehlermeldungen für die Sicherheits-CPUs SM560-S / SM560-S-FD-1 / SM560-S-FD-4

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E2	1 … 4	255	30	1	0	Operation beendet.	Adressschaltereinstellung der Sicherheitssteuerung ändern oder SD-Karte aus nicht sicherer Steuerung entnehmen. Sicherheitssteuerung neu starten. Falls dieser Fehler immer noch besteht, Sicherheitssteuerung austauschen.
E2	1 … 4	255	30	1	1	Falsche Nutzerdaten	Löschen Sie die Nutzerdaten von der Sicherheitssteuerung. Starten Sie die Sicherheitssteuerung neu und schreiben Sie die Nutzerdaten nochmals.
E2	1 … 4	255	30	1	2	Interner PROFIsafe-Initialisierungsfehler	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus. Wenden Sie sich an den technischen Support von ABB.
E2	1 … 4	255	30	1	12	Flash-Lesefehler	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus. Wenden Sie sich an den technischen Support von ABB.
E2	1 … 4	255	30	1	18	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
E2	1 … 4	255	30	1	28	Download-Fehler Bootprojekt	Laden Sie das Bootprojekt erneut. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
E2	1 … 4	255	30	1	40	Falsche Firmwareversion	Aktualisieren Sie die Firmware der Sicherheitssteuerung. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
E2	1 … 4	255	30	1	43	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E2	1 … 4	255	30	1	48	Über- oder Unterspannung erkannt	Starten Sie die Sicherheits-steuerung neu. Überprüfen Sie, ob in den Einstellungen der Sicherheitssteuerung ein Fehler in der Spannungs-versorgung vorliegt. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheits-steuerung aus.
E2	1 … 4	255	30	1	52	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	0	Anwenderpro-gramm hat einen sicheren Stopp ausgelöst	Prüfen Sie das Anwenderpro-gramm.
E2	1 … 4	255	30	2	1	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	2	Interner PROFIsafe-Fehler	Starten Sie die Sicherheits-steuerung neu. Wenn der Fehler weiterhin besteht, tau-schen Sie die Sicherheits-steuerung aus. Wenden Sie sich an den technischen Sup-port von ABB.
E2	1 … 4	255	30	2	3	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	10	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	13	Flash-Schreib-fehler	Starten Sie die Sicherheits-steuerung neu. Wenn der Fehler weiterhin besteht, tau-schen Sie die Sicherheits-steuerung aus. Wenden Sie sich an den technischen Sup-port von ABB.
E2	1 … 4	255	30	2	17	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	18	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E2	1 … 4	255	30	2	19	Prüfsummen-fehler in Sicher-heitssteuerung.	Starten Sie die Sicherheits-steuerung neu. Wenn der Fehler weiterhin besteht, tau-schen Sie die Sicherheits-steuerung aus.
E2	1 … 4	255	30	2	25	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	37	Zykluszeitfehler in Sicherheits-steuerung	Überprüfen Sie die Watchdog-Zeit der Sicher-heitssteuerung.
E2	1 … 4	255	30	2	38	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	42	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	43	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	52	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	2	54	Interner Fehler	Wenden Sie sich an den tech-nischen Support von ABB. Ersetzen Sie die Sicherheits-steuerung.
E2	1 … 4	255	30	3	30	PROFIsafe-Konfigurations-fehler	Überprüfen Sie die F-Para-meter-Konfiguration des E/A-Moduls und laden Sie das Bootprojekt erneut.
E2	9	1 … 4	1	0	17	Zugriffstest fehl-geschlagen.	Überprüfen Sie die Adres-seinstellungen für die Schalter der Sicherheitssteu-erung. Starten Sie die Sicher-heitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheits-steuerung aus.
E2	9	1 … 4	1	0	43	Interner Fehler	Überprüfen Sie die Adres-seinstellungen für die Schalter der Sicherheitssteu-erung. Starten Sie die Sicher-heitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheits-steuerung aus.

Fehlerschweregrad	Komponente oder Schnittstelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E2	9	1 … 4	31	0	43	Interner Fehler	Tauschen Sie das Modul aus.
E3	1 … 4	255	30	1	26	Fehler in den Konfigurationsdaten, die Sicherheitssteuerung kann die Konfigurationsdaten nicht lesen.	Erzeugen Sie neue Konfigurationsdaten
E3	1 … 4	255	30	1	27	Fehler in den Konfigurationsdaten, die Sicherheitssteuerung kann die Konfigurationsdaten nicht lesen.	Erzeugen Sie ein Bootprojekt.
E4	1 … 4	255	30	1	0	Vorgang beendet	Ändern Sie die Adresseinstellungen für den Schalter der Sicherheitssteuerung oder entfernen Sie die SD-Karte aus der nicht sicherheitsgerichteten SPS. Sicherheitssteuerung neu starten. Falls dieser Fehler immer noch besteht, Sicherheitssteuerung austauschen.
E4	1 … 4	255	30	1	4	Bootprojekt nicht geladen, max. Spannungseinbruch erreicht	Starten Sie die Sicherheitssteuerung neu.
E4	1 … 4	255	30	1	8	Spannungseinbruchdaten fehlen oder korrupt. Standard-Spannungseinbruchdaten wurden von der Sicherheitssteuerung im Flash-Speicher gespeichert.	Warnung
E4	1 … 4	255	30	1	19	Prüfsummenfehler in der Konfiguration der Sicherheitssteuerung	Erzeugen Sie ein neues Bootprojekt und starten Sie die Sicherheitssteuerung neu.
E4	1 … 4	255	30	2	13	Flash-Schreibfehler (Produktionsdaten)	Warnung

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E4	1 … 4	255	30	2	26	Keine oder falsche Konfigurationsdaten von der PM5x, Zustand RUN nicht möglich	Erzeugen Sie ein korrektes Bootprojekt für die PM5x.
E4	1 … 4	255	30	2	39	Mehr als eine Instanz von SF_WDOG_TIME_SET oder SF_MAX_POWER_DIP_SET	Warnung
E4	1 … 4	255	30	4	13	Flash-Schreibfehler (Bootprojekt)	Warnung
E4	1 … 4	255	30	5	13	Flash-Schreibfehler (Bootcode)	Warnung
E4	1 … 4	255	30	6	13	Flash-Schreibfehler (Firmware)	Warnung
E4	1 … 4	255	30	7	13	Flash-Schreibfehler (Passwort)	Warnung
E4	1 … 4	255	30	8	13	Flash-Schreibfehler (Nutzerdaten)	Warnung
E4	1 … 4	255	30	9	13	Flash-Schreibfehler (Nutzerdaten)	Warnung
E4	1 … 4	255	30	10	13	Flash-Schreibfehler (intern)	Warnung
E4	1 … 4	255	30	11	13	Flash-Schreibfehler (intern)	Warnung
E4	1 … 4	255	30	12	13	Flash-Schreibfehler (intern)	Warnung

Tab. 108: Spezifische Fehlermeldungen für die Sicherheits-CPUs SM560-S-FD-1 / SM560-S-FD-4

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E2	1 … 4	255	28	0 … 31	43	Interner PROFIsafe F-Device-Fehler	Starten Sie die Sicherheits-steuerung neu. Wenn der Fehler weiterhin besteht, tau-schen Sie die Sicherheits-steuerung aus. Wenden Sie sich an den technischen Sup-port von ABB.
E3	1 … 4	255	28	0 … 31	1	Sicherheitsziel-adresse nicht gültig (F_Dest_Add)	Überprüfen Sie die Konfigu-ration der Sicherheitssteue-rung oder die Einstellung der Schalteradresse. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
E3	1 … 4	255	28	0 … 31	2	Sicherheits-quelladresse nicht gültig (F_Source_Add)	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung.
E3	1 … 4	255	28	0 … 31	10	Parameter „F_SIL“ übertrifft SIL einer spezi-ellen Geräteanwendung	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung.
E3	1 … 4	255	28	0 … 31	11	Wert der Sicher-heits-Watchdog-Zeit ist 0 ms (F_WD_Time)	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung.
E3	1 … 4	255	28	0 … 31	19	Fehler CRC1	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung. Wenn der Fehler weiterhin besteht, wenden Sie sich an den technischen Support von ABB.
E3	1 … 4	255	28	0 … 31	28	Diskrepanz der Sicherheitsziel-adresse (F_Dest_Add)	Überprüfen Sie die Konfigu-ration der Sicherheitssteue-rung oder die Einstellung der Schalteradresse. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
E3	1 … 4	255	28	0 … 31	42	Parameter „F_CRC_Length “ passt nicht zu den erzeugten Werten	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung.
E3	1 … 4	255	28	0 … 31	40	Version des F-Parameter-Satzes nicht korrekt	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung.

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E3	1 … 4	255	30	1	17	Sicherheitsquel-ladressen können nicht geprüft werden	Prüfen Sie die PROFIsafe F-Host-Bibliotheksversion (2.0.0 oder höher). Wenn der Fehler weiterhin besteht, wenden Sie sich an den technischen Sup-port von ABB.
E3	1 … 4	255	30	1	54	PROFIsafe-Regeln für F_Dest_Add werden verletzt	Vergleichen Sie die Kon-figuration der Sicherheits-steuerung oder die Einstel-lung des Adressschalters mit den PROFIsafe-Konfigu-rationsregeln für F_Dest_Add. Starten Sie die Sicherheits-steuerung neu. Wenn der Fehler weiterhin besteht, wenden Sie sich an den tech-nischen Support von ABB.
E3	1…4	255	28	0…31	26	F_Block_ID nicht unterstützt	Überprüfen Sie die Konfigura-tion der Sicherheitssteuerung.
E3	1…4	255	28	0…31	20	Übertragungs-fehler: Daten inkonsistent (CRC2-Fehler)	Konfiguration und Verkabe-lung prüfen
E3	1…4	255	28	0…31	25	Übertragungs-fehler: Zeitüber-schreitung (F_WD_Time oder F_WD_Time_2 verstrichen)	Konfiguration der Sicherheits-steuerung prüfen.

Tab. 109: Abbild der AC500/AC500-S-Fehler auf PROFINET-Kanalfehler

AC500/AC500-S-Fehler	PROFINET-Kanalfehlertyp	PROFINET-Diagnoseinformationen
28	64	Diskrepanz der Sicherheitszieladresse (F_Dest_Add)
1	65	Sicherheitszieladresse nicht gültig (F_Dest_Add)
2	66	Sicherheitsquelladresse nicht gültig (F_Source_Add)
11	67	Wert der Sicherheits-Watchdog-Zeit ist 0 ms (F_WD_Time)
10	68	Parameter „F_SIL“ übertrifft SIL einer spe-ziellen Geräteanwendung
42	69	Parameter „F_CRC_Length“ passt nicht zu den erzeugten Werten
40	70	Version des F-Parameter-Satzes nicht kor-rekt
19	71	Fehler CRC1

AC500/AC500-S-Fehler	PROFINET-Kanalfehlertyp	PROFINET-Diagnoseinformationen
26	76	F_Block_ID nicht unterstützt
20	77	Übertragungsfehler: Daten inkonsistent (CRC2-Fehler)
25	78	Übertragungsfehler: Zeitüberschreitung (F_WD_Time oder F_WD_Time_2 verstrichen)

B.2.2 Fehlermeldungen für Sicherheits-E/A-Module

Tab. 110: Fehlermeldungen für Sicherheits-E/A-Module (Kanal- oder Modulreintegration möglich)

Fehler-schwe-regrad	Kompo-nente oder Schnitt-stelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E3	14	1..10	0	0..15	3	Diskrepanzzeit abgelaufen	Überprüfen Sie Diskrepanz-zeit, Kanal-Verdrahtung und Sensor.
E3	14	1..10	0	0..15	12	Fehler Testim-puls	Überprüfen Sie Verdrahtung und Sensor.
E3	14	1..10	0	0..15	13	Fehler Kanal-übersprechen Testimpuls	Überprüfen Sie Verdrahtung und Sensor. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus. Wenden Sie sich an den tech-nischen Support von ABB.
E3	14	1..10	0	0..15	25	Fehler. Kanal reagiert nicht mehr.	Überprüfen Sie die Verdrah-tung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
E3	14	1..10	0	0..15	28	Fehler Kanal-übersprechen	Überprüfen Sie die Verdrah-tung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
E3	14	1..10	1	0..3	4	Messwertüber-schreitung am E/A-Modul	Überprüfen Sie die Kanal-Verdrahtung und die Sensor-Spannungsversorgung.
E3	14	1..10	1	0..3	7	Messwertunter-schreitung am Ein-/Ausgang	Überprüfen Sie die Kanal-Verdrahtung und die Sensor-Spannungsversorgung.
E3	14	1..10	1	0..3	55	Differenz der Kanalwerte zu groß	Passen Sie die Toleranz-fenster für Kanäle an. Überprüfen Sie die Kanal-Verdrahtung und die Sensor-Konfiguration.
E3	14	1..10	2	0..7	13	Fehler Kanal-rücklesen	Überprüfen Sie die Verdrah-tung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
E3	14	1..10	2	0..7	18	Fehler Kanal-übersprechen	Überprüfen Sie die Verdrah-tung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
E3	14	1..10	31	31	10	Prozessspan-nung zu hoch	Überprüfen Sie die Prozess-spannung.

Fehlerschweregrad	Komponente oder Schnittstelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E3	14	1..10	31	31	11	Prozessspannung zu niedrig	Überprüfen Sie die Prozessspannung.
E3	14	1..10	31	31	20	PROFIsafe-Kommunikationsfehler	Starten Sie das E/A-Modul neu. Wenn der Fehler weiterhin besteht, wenden Sie sich an den technischen Support von ABB.
E3	14	1..10	31	31	25	PROFIsafe-Timeout des PROFIsafe-Watchdog	Starten Sie das E/A-Modul neu. Wenn der Fehler weiterhin besteht, verlängern Sie die PROFIsafe-Watchdog-Zeit.
E3	14	1..10	31	31	43	Interner Fehler im Gerät	Tauschen Sie das E/A-Modul aus.

Tab. 111: Fehlermeldungen der Sicherheits-E/A-Module (Kanal- oder Modulreintegration nicht möglich)

Fehlerschweregrad	Komponente oder Schnittstelle	Gerät	Module	Kanal	Error	Fehlertext	Abhilfe
E3	14	1..10	31	31	18	Plausibilitätsprüfung fehlgeschlagen (iParameter)	Überprüfen Sie die Konfiguration.
E3	14	1..10	31	31	19	Prüfsummenfehler im E/A-Modul	Überprüfen Sie die Sicherheitskonfiguration und CRCs für I- und F-Parameter.
E3	14	1..10	31	31	26	Parameterfehler	Überprüfen Sie Master oder Konfiguration.
E3	14	1..10	31	31	28	Konfiguration für F-Parameter stimmt nicht mit dem Wert des Adressschalters überein.	Überprüfen Sie die Konfiguration für F-Parameter des E/A-Moduls und den Wert des Adressschalters.

B.3 Konfiguration der AC500 V2-Standard-CPU-Parameter

Die folgenden Parameter der Standard-CPU-Konfiguration beeinflussen das Gesamtverhalten der Sicherheits- und der Standard-CPU.

- *„Verhalten der Ausgänge bei Stopp“*
- *„Stopp bei Fehlerklasse“*
- *„Warmstart“* nach Fehler mit Schweregrad 2

Die Einstellungen für diese Parameter beeinträchtigen nicht die Systemsicherheit.

„Verhalten der Ausgänge bei Stopp“

Wert *„False in Hardware und Onlineanzeige“* (Default)

Wird die Standard-CPU gestoppt, wird die Ausführung des Anwendungsprogramms auf der Sicherheits-CPU gestoppt. Die Übertragung von Ausgangswerten der Sicherheits-CPU durch eine Standard-CPU in Sicherheitstelegrammen wird ebenfalls gestoppt. Keine gültigen PROFIsafe Sicherheitstelegramme erreichen die Sicherheits-E/A-Module und andere F-Devices. Diese werden passiviert, sobald die Watchdog-Zeit abläuft.

Wert *„False in Hardware und akt. Zustand in Onlineanzeige“*

Wenn die Standard-CPU gestoppt wird, wird auch die Übertragung von Ausgangswerten der Sicherheits-CPU in PROFIsafe Sicherheitstelegrammen gestoppt. Der Hardware-Status der Kommunikationsschnittstelle der Sicherheits-CPU wird „0“. Auf der Online-Anzeige werden die letzten gültigen Werte vom letzten Programmzyklus der Sicherheitsanwendung angezeigt. Aufgrund der gestoppten Übertragung von Werten an die Kommunikationsschnittstelle der Sicherheits-CPU können keine gültigen PROFIsafe Sicherheitstelegramme die Sicherheits-E/A-Module und andere F-Devices erreichen. Diese werden passiviert, sobald die Watchdog-Zeit abläuft.

Wert *„Akt. Zustand in Hardware und Onlineanzeige“*

Wenn die Standard-CPU gestoppt wird, läuft die Sicherheits-CPU weiter. Die Ausgangswerte der Sicherheits-CPU in PROFIsafe Sicherheitstelegrammen werden weiter von der Standard-CPU übertragen. Der Hardware-Status der Kommunikationsschnittstelle der Sicherheits-CPU und die Werte der Online-Anzeige bleiben intakt. Sicherheits-E/A-Module und andere F-Devices können Sicherheitstelegramme von der Sicherheits-CPU empfangen. Der Betrieb des sicherheitsbezogenen Teils wird durch einen Stopp der Standard-CPU nicht beeinflusst.

„Stopp bei Fehlerklasse“

Wert *„E2“* (Standard)

Bei einem Fehler mit Schweregrad 1 oder 2 werden die Standard-CPU, alle zugehörigen Kommunikationsmodule und die Sicherheits-CPU gestoppt. PROFIsafe F-Host und F-Device-Stacks laufen auf der Sicherheits-CPU mit Failsafe-Werten weiter.

Wert *„E3“*

Bei einem Fehler mit Schweregrad 1, 2 oder 3 werden die Standard-CPU, alle zugehörigen Kommunikationsmodule und die Sicherheits-CPU gestoppt. PROFIsafe F-Host und F-Device-Stacks laufen auf der Sicherheits-CPU mit Failsafe-Werten weiter.

Wert *„E4“*

Bei einem Fehler mit Schweregrad 1, 2, 3 oder 4 werden die Standard-CPU, alle zugehörigen Kommunikationsmodule und die Sicherheits-CPU gestoppt. PROFIsafe F-Host und F-Device-Stacks laufen auf der Sicherheits-CPU mit Failsafe-Werten weiter.

„Warmstart“ **Wert *„Aus“* (Standard)**

Bei einem Fehler mit Schweregrad 2 erfolgt kein Warmstart der Standard-CPU, aller zugehörigen Kommunikationsmodule und der Sicherheits-CPU.

Werte *„Ein nach E2-Fehler“*, *„Ein nach kurzem Spannungseinbruch“*, *„Ein nach E2-Fehler oder kurzem Spannungseinbruch“*

Bei einem Fehler mit Schweregrad 2 oder nach einem kurzen Spannungseinbruch erfolgt ein Warmstart der Standard-CPU, aller zugehörigen Kommunikationsmodule und der Sicherheits-CPU. Nach dem Neustart der Sicherheits-CPU können die dezentralen Sicherheits-E/A-Module reintegriert werden, z. B. mit dem PROFIsafe F-Device-Reintegrationsschema ↬ *[2]*.

B.4 SPS-Befehle AC500 V2-Standard-CPU

Die folgenden SPS-Browser-Befehle (falls von der Firmware der aktuellen Standard-CPU unterstützt) von der Standard-CPU ändern den Zustand der Sicherheits-CPU:

- `reboot`
  Startet die Standard-CPU und somit auch die Sicherheits-CPU neu.
- `resetprgorg`
  Versetzt die Standard-CPU und die Sicherheits-CPU in ihren Originalzustand (sämtliche Variablen, Flash-Speicher usw. erhalten wieder die Initialwerte). Die Sicherheits-CPU ändert ihren Zustand von RUN zu SAFE STOP (nicht sicher).
- `stopprg`, `resetprg`, `resetprgcold` und Menüpunkte *„Online ➔ Zurücksetzen (kalt, Original)“*
  Zwingen die Sicherheits-CPU, den Modus RUN (Sicherheitsmodus) zu verlassen und in den Modus DEBUG STOP (nicht sicher) zu wechseln.
- `startprg`
  Zwingt die Sicherheits-CPU, den Modus DEBUG STOP (nicht sicher) zu verlassen und in den Modus DEBUG RUN (nicht sicher) zu wechseln. Wenn sich die Sicherheits-CPU bereits im Modus RUN (Sicherheitsmodus) oder DEBUG RUN (nicht sicher) befindet, hat dieser SPS-Browser-Befehl keinen Einfluss auf die Sicherheits-CPU.

B.5 Datenaustausch zwischen Sicherheits-CPU und AC500 V2-Standard-CPU

Optionen für den Datenaustausch zwischen Sicherheits-CPU und AC500 V2-Standard-CPU:

- Azyklischer nicht sicherer Datenaustausch: mehrere Zyklen der Sicherheits-CPU für die Übertragung der Daten erforderlich, max. 84 Bytes in jede Richtung ↳ *Anhang B.5.1 „Azyklischer nicht sicherer Datenaustausch“ auf Seite 431*
- Zyklischer nicht sicherer Datenaustausch: max. 3 Zyklen der Sicherheits-CPU für die Übertragung der Daten erforderlich, max. 2 kB in jede Richtung ↳ *Anhang B.5.2 „Zyklischer nicht sicherer Datenaustausch“ auf Seite 436*

GEFAHR!

Es wird nicht empfohlen, Datenwerte von der Standard-CPU auf die Sicherheits-CPU zu übertragen. Hierbei müssen die Endanwender zusätzliche prozessspezifische Validierungsverfahren in ihrem Sicherheitsprogramm definieren, um die Korrektheit der übertragenen nicht sicheren Daten zu überprüfen, wenn sie diese nicht sicheren Werte für Sicherheitsfunktionen verwenden möchten.

Datenwerte von der Sicherheits-CPU auf die Standard-CPU zu übertragen, z. B. für Diagnose und spätere Darstellung auf Bedienpanels, ist kein Problem.

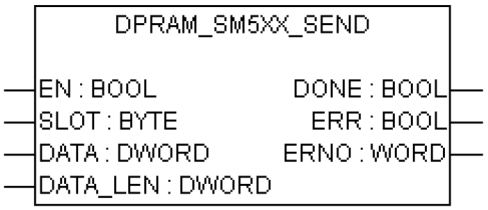
B.5.1 Azyklischer nicht sicherer Datenaustausch

Der azyklische nicht sichere Datenaustausch ist standardmäßig in der Programmierumgebung für die Sicherheits-CPU und die Standard-CPU verfügbar.

Verwenden Sie auf der Sicherheits-CPU die Funktionsbausteine SF_DPRAM_PM5XX_S_REC und SF_DPRAM_PM5XX_S_SEND ↳ *Kapitel 4.6.7.13 „SF_DPRAM_PM5XX_S_REC“ auf Seite 359* ↳ *Kapitel 4.6.7.14 „SF_DPRAM_PM5XX_S_SEND“ auf Seite 361*.

Verwenden Sie auf der Standard-CPU die Funktionsbausteine DPRAM_SM5XX_SEND and DPRAM_SM5XX_REC ↳ *Anhang B.5.1.1 „ DPRAM_SM5XX_SEND“ auf Seite 432* ↳ *Anhang B.5.1.2 „ DPRAM_SM5XX_REC“ auf Seite 434*.

B.5.1.1 DPRAM_SM5XX_SEND



Der Funktionsbaustein DPRAM_SM5XX_S_SEND sendet Daten zur Sicherheits-CPU.

Über den Funktionsbaustein DPRAM_SM5XX_SEND werden Daten an die Sicherheits-CPU gesendet. Diese Daten werden im Speicherbereich bereitgestellt (DATA, Speicheradresse für die Sendedaten über ADR-Operator). Der Funktionsbaustein wird mit einem TRUE-Signal (Flanke „0“ ➔ „1“) am Eingang EN aktiviert. Die Steckplatznummer der Sicherheits-CPU wird am Eingang SLOT eingestellt. Am Eingang DATA_LEN wird die Länge der zu sendenden Daten in Byte angegeben. Ein erfolgreicher Sendevorgang wird durch DONE=TRUE und ERR=FALSE signalisiert. Wurde bei der Verarbeitung des Funktionsbausteins ein Fehler festgestellt, wird er an den Ausgängen ERR und ERNO angezeigt.

Hinweis: Das Senden von Daten mit dem Funktionsbaustein DPRAM_SM5XX_SEND ist flankengetriggert, d. h. jeder Sendevorgang wird durch eine FALSE-TRUE-Flanke am Eingang EN ausgelöst.

Tab. 112: FB-Name: DPRAM_SM5XX_SEND

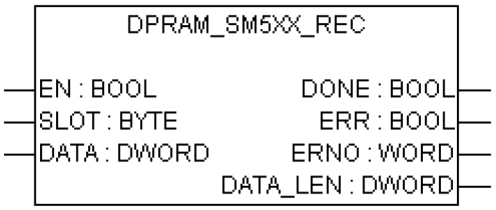
Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Freigabe der Funktionsbausteinverarbeitung Die Verarbeitung dieses Funktionsbausteins wird vom Eingang EN gesteuert. Die Datenübertragung wird durch eine FALSE/TRUE-Flanke angestoßen. Das Senden von Daten wird durch den Ausgang DONE signalisiert
SLOT	BYTE	16#00	Steckplatznummer (Modulnummer) Am Eingang SLOT wird der Steckplatz (Modulnummer) ausgewählt, zu dem die Daten gesendet werden sollen. Die externen Steckplätze sind von rechts nach links durchnummeriert und beginnen mit der Nummer 1.
DATA	DWORD	16#00000000	Speicheradresse für die Sendedaten über ADR-Operator Am Eingang DATA wird die Adresse der Variablen angegeben, in die die Anwenderdaten kopiert werden sollen. Die an DATA spezifizierte Adresse muss zu einer Variablen vom Typ ARRAY oder STRUCT gehören. Hinweis: Speicherbereichsüberschneidungen vermeiden, indem die Größe der Variablen an die maximal zu erwartenden Daten angepasst wird.
DATA_LEN	WORD	16#0000	Länge der zu sendenden Daten ab Adresse DATA in Byte, max. 84 Am Eingang DATA_LEN wird die Länge der zu sendenden Daten in Byte angegeben. Die maximale Anzahl ist 84.
VAR_OUTPUT			

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
DONE	BOOL	FALSE	Die Daten wurden gesendet. Am Ausgang DONE wird angezeigt, dass Daten gesendet wurden. Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Der Sendevorgang ist abgeschlossen. Es wurde ein Datensatz korrekt gesendet. ● DONE = TRUE und ERR = TRUE: Beim Sendevorgang ist ein Fehler aufgetreten. Die Fehlernummer wird am Ausgang ERNO ausgegeben.
ERR	BOOL	FALSE	Fehlermeldung des Funktionsbausteins Am Ausgang ERR wird angezeigt, ob beim Sendevorgang ein Fehler aufgetreten ist. Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Ist ein Fehler aufgetreten beim Versand, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer ↳ *[3]* Am Ausgang ERNO wird eine Fehlerkennung ausgegeben, wenn an einem Eingang ein ungültiger Wert angegeben wurde oder während der Verarbeitung des Auftrags ein Fehler aufgetreten ist. ERNO muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Der an ERNO ausgegebene Wert ist nur gültig, wenn DONE = TRUE und ERR = TRUE ist. Die Kodierung der Fehlermeldungen am Ausgang ERNO wird am Anfang der Funktionsbausteinbeschreibung erläutert.

Aufruf in ST

```
SM5xxSend (EN := SM5xxSend_EN,
SLOT := SM5xxSend_SLOT,
DATA := ADR(SM5xxSend_DATA),
DATA_LEN := SM5xxSend_DATA_LEN,
DONE => SM5xxSend_DONE,
ERR => SM5xxSend_ERR,
ERNO => SM5xxSend_ERNO);
```

B.5.1.2 DPRAM_SM5XX_REC



Der Funktionsbaustein DPRAM_SM5XX_S_REC empfängt Daten von der Sicherheits-CPU

Über den Funktionsbaustein DPRAM_SM5XX_REC werden Daten von der Sicherheits-CPU empfangen. Die Daten werden im Speicherbereich abgelegt (DATA, Speicheradresse für die Empfangsdaten über ADR-Operator). Die Aktivierung des Funktionsbausteins erfolgt durch ein TRUE-Signal an Eingang EN. Der Baustein ist solange aktiv, bis Eingang EN = FALSE wird. Die Steckplatznummer der Sicherheits-CPU wird am Eingang SLOT eingestellt. An Ausgang DATA_LEN wird die Länge der empfangenen Daten in Byte ausgegeben. Ein erfolgreicher Empfangsvorgang wird durch DONE=TRUE und ERR=FALSE signalisiert. Wurde bei der Verarbeitung des Funktionsbausteins ein Fehler festgestellt, wird er an den Ausgängen ERR und ERNO angezeigt.

Hinweis: Der Empfang mit dem Funktionsbaustein DPRAM_SM5XX_REC ist nicht flankengetriggert. Der Eingang EN ist also dauerhaft auf TRUE zu setzen, solange Daten empfangen werden sollen.

Tab. 113: FB-Name: DPRAM_SM5XX_REC

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
VAR_INPUT			
EN	BOOL	FALSE	Freigabe der Funktionsbausteinverarbeitung Die Verarbeitung dieses Funktionsbausteins wird vom Eingang EN gesteuert. Der Funktionsbaustein ist aktiv, wenn EN = TRUE ist. Der Empfang von Daten wird durch den Ausgang DONE signalisiert.
SLOT	BYTE	16#00	Steckplatznummer (Modulnummer) Am Eingang SLOT wird der Steckplatz (Modulnummer) ausgewählt, von dem die Daten ausgelesen werden sollen. Die externen Steckplätze sind von rechts nach links durchnummeriert und beginnen mit der Nummer 1.
DATA	DWORD	16#00000000	Speicheradresse für die Empfangsdaten über ADR-Operator Am Eingang DATA wird die Adresse der Variablen angegeben, in die die Anwenderdaten kopiert werden sollen. Die an DATA spezifizierte Adresse muss zu einer Variablen vom Typ ARRAY oder STRUCT gehören. Hinweis: Speicherbereichsüberschneidungen vermeiden, indem die Größe der Variablen an die maximal zu erwartenden Daten angepasst wird.
VAR_OUTPUT			

Name	Datentyp	Initialwert	Beschreibung, Parameterwerte
DONE	BOOL	FALSE	Die Daten wurden empfangen. Am Ausgang DONE wird der Empfang der Daten angezeigt. Der Ausgang muss immer im Zusammenhang mit dem Ausgang ERR betrachtet werden. Es gilt: ● DONE = TRUE und ERR = FALSE: Der Empfangsvorgang ist abgeschlossen. Ein Datensatz wurde korrekt empfangen. ● DONE = TRUE und ERR = TRUE: Beim Empfangsvorgang ist ein Fehler aufgetreten. Die Fehlernummer wird am Ausgang ERNO ausgegeben.
ERR	BOOL	FALSE	Fehlermeldung des Funktionsbausteins Am Ausgang ERR wird angezeigt, ob beim Empfangsvorgang ein Fehler aufgetreten ist. Dieser Ausgang muss immer zusammen mit dem Ausgang DONE ausgewertet werden. Ist bei der Verarbeitung des Funktionsbausteins ein Fehler aufgetreten, so gilt: DONE = TRUE und ERR = TRUE. Der Ausgang ERNO signalisiert die Fehlernummer.
ERNO	WORD	16#0000	Fehlernummer *[3]* Am Ausgang ERNO wird eine Fehlerkennung ausgegeben, wenn an einem Eingang ein ungültiger Wert angegeben wurde oder während der Verarbeitung des Auftrags ein Fehler aufgetreten ist. ERNO muss immer im Zusammenhang mit den Ausgängen DONE und ERR betrachtet werden. Der an ERNO ausgegebene Wert ist nur gültig, wenn DONE = TRUE und ERR = TRUE ist. Die Kodierung der Fehlermeldungen am Ausgang ERNO wird am Anfang der Funktionsbausteinbeschreibung erläutert.
DATA_LEN	WORD	16#0000	Länge der Daten in Byte An Ausgang DATA_LEN wird die Länge der empfangenen Daten in Byte ausgegeben. DATA_LEN ist nur gültig, wenn DONE = TRUE ist.

Aufruf in ST

```
SM5xxRec (EN := SM5xxRec_EN,
SLOT := SM5xxRec_SLOT,
DATA := ADR(SM5xxRec_DATA),
DONE => SM5xxRec_DONE,
ERR => SM5xxRec_ERR,
ERNO => SM5xxRec_ERNO,
DATA_LEN => SM5xxRec_DATA_LEN);
```

B.5.2 Zyklischer nicht sicherer Datenaustausch

Verwenden Sie im Automation Builder die Registerkarte *„Konfiguration Datenaustausch“* der Sicherheits-CPU, um die Funktion für den zyklischen nicht sicheren Datenaustausch zu konfigurieren. Diese ermöglicht den Datenaustausch zwischen der Sicherheits-CPU und der Standard-CPU für eine schnelle zyklische Kommunikation und die Übertragung einer großen Datenmenge per DPRAM. Für die meisten Sicherheitsanwendungen wird diese Funktion nicht benötigt und sollte auch nicht verwendet werden. Standardmäßig ist das Ankreuzfeld *„Zyklischer nicht sicherer Datenaustausch“* nicht markiert. Wenn Sie die Funktion dennoch benötigen, finden Sie Informationen zur Verwendung der Funktion für den zyklischen nicht sicheren Datenaustausch in der Beschreibung unter *www.abb.com/plc – Dokumentnr. 33ADR025195M0202*.

Der zyklische nicht sichere Datenaustausch mit AC500 V2-Standard-CPUs wird von Automation Builder 1.0.1 unterstützt.

C Verwendung von Sicherheits-CPU mit AC500 V3-Standard-CPU PM56xx

C.1 Kompatibilität mit AC500 V3-Standard-CPU

Alle Kompatibilitätsinformationen sind für normale sowie für XC-Geräte gültig.

Tab. 114: Kompatibilität für Sicherheits-CPU mit AC500 Standard-CPU V3

Sicherheits-CPU	**SM560-S**	**SM560-S-FD-1, SM560-S-FD-4**
Firmware-Version von Sicherheits-CPU	Beliebig	Beliebig
Standard-CPU	Jede V3 CPU, mit Ausnahme von AC500-eCo-CPUs	In Vorbereitung
Firmware-Version von Standard-CPU	Ab V3.3.0	
Version der Engineering Suite Automation Builder	Ab 2.3.0	

Tab. 115: Kompatibilität für AC500-S mit Standardkomponenten mit Ausnahme von CPUs

Komponente	**SM560-S**	**SM560-S-FD-1, SM560-S-FD-4**
Firmware-Version von Kommunikationsmodul CM579-PNIO	Ab V2.8.6.21	Ab V2.8.6.21
Firmware-Version von Kommunikationsmodul CM589-PNIO(-4)	Nicht zutreffend	In Vorbereitung
Firmware-Version von Kommunikationsschnittstellen-Modul CI501-PNIO, CI502-PNIO, CI504-PNIO, CI506-PNIO	Ab V3.2.0	Ab V3.2.0

C.2 Fehlermeldungen mit AC500 V3-Standard-CPUs

C.2.1 Fehlermeldungen für Sicherheits-CPUs

Die Fehler werden so angezeigt, wie sie im Automation Builder dargestellt sind. In AC500-S Programming Tool werden Fehler ähnlich wie Fehlermeldungen von AC500 V2-Standard-CPUs angezeigt.

Tab. 116: Fehlermeldungen für Sicherheits-CPUs

Schweregrad	Fehlercode	Beschreibung	Abhilfe
2	8235	Interner Fehler	Tauschen Sie das Modul aus.
2	8448	Vorgang beendet	Ändern Sie die Adresseinstellungen für die Schalter der Sicherheitssteuerung oder entfernen Sie die SD-Karte aus der nicht sicherheitsgerichteten SPS. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
2	8449	Falsche Nutzerdaten	Löschen Sie die Nutzerdaten von der Sicherheitssteuerung. Starten Sie die Sicherheitssteuerung neu und schreiben Sie die Nutzerdaten nochmals.
2	8450	Interner PROFIsafe-Initialisierungsfehler	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus. Wenden Sie sich an den technischen Support von ABB.
2	8460	Flash-Lesefehler	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus. Wenden Sie sich an den technischen Support von ABB.
2	8466	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8476	Download-Fehler Bootprojekt	Laden Sie das Bootprojekt erneut. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
2	8488	Falsche Firmwareversion	Aktualisieren Sie die Firmware der Sicherheitssteuerung. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
2	8491	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.

Schweregrad	Fehlercode	Beschreibung	Abhilfe
2	8496	Über- oder Unterspannung erkannt	Starten Sie die Sicherheitssteuerung neu. Überprüfen Sie, ob in den Einstellungen der Sicherheitssteuerung ein Fehler in der Spannungsversorgung vorliegt. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
2	8500	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8704	Anwenderprogramm hat einen sicheren Stopp ausgelöst	Prüfen Sie das Anwenderprogramm.
2	8705	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8706	Interner PROFIsafe-Fehler	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus. Wenden Sie sich an den technischen Support von ABB.
2	8707	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8714	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8717	Flash-Schreibfehler	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus. Wenden Sie sich an den technischen Support von ABB.
2	8721	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8722	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8723	Prüfsummenfehler in Sicherheitssteuerung	Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
2	8729	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.

Schweregrad	Fehlercode	Beschreibung	Abhilfe
2	8741	Zykluszeitfehler in Sicherheitssteuerung	Überprüfen Sie die Watchdog-Zeit der Sicherheitssteuerung.
2	8742	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8746	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8747	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8756	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8758	Interner Fehler	Wenden Sie sich an den technischen Support von ABB. Ersetzen Sie die Sicherheitssteuerung.
2	8990	PROFIsafe-Konfigurationsfehler	Überprüfen Sie die F-Parameter-Konfiguration des E/A-Moduls und laden Sie das Bootprojekt erneut.
3	12561	Sicherheitsquelladressen können nicht geprüft werden	Prüfen Sie die PROFIsafe F-Host-Bibliotheksversion (2.0.0 oder höher). Wenn der Fehler weiterhin besteht, wenden Sie sich an den technischen Support von ABB.
3	12570	Fehler in den Konfigurationsdaten, die Sicherheitssteuerung hat die Konfigurationsdaten nicht akzeptiert, z. B. Abweichungen zwischen den Konfigurationen der Sicherheits- und Standard-SPS.	Erzeugen Sie nochmals neue Konfigurationsdaten für die Sicherheits- und Standard-SPS, erstellen Sie Bootprojekte neu und laden Sie sie nochmals auf die Sicherheits- und Standard-SPS herunter.
3	12571	Fehler in den Konfigurationsdaten, die Sicherheitssteuerung kann die Konfigurationsdaten nicht lesen.	Erzeugen Sie ein Bootprojekt.
3	12598	PROFIsafe-Regeln für F_Dest_Add werden verletzt	Vergleichen Sie die Konfiguration der Sicherheitssteuerung oder die Einstellung des Adressschalters mit den PROFIsafe-Konfigurationsregeln für F_Dest_Add. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, wenden Sie sich an den technischen Support von ABB.

Schweregrad	Fehlercode	Beschreibung	Abhilfe
3	32770	Watchdog-Fehler des Kopplers	
3	32771	Falsche Firmware-Version des Kommunikationsmoduls	Aktualisieren Sie die Firmware.
3	32772	Initialisierung des Sicherheitsmoduls am Steckplatz fehlgeschlagen. Mehr als ein Sicherheitsmodul eingesteckt.	Entfernen Sie dieses Modul vom betreffenden Steckplatz oder, wenn nur ein Sicherheitsmodul eingesteckt und das Modul defekt ist, tauschen Sie dieses Modul aus.
3	32774	Ungültige Konfigurationsdaten	Überprüfen Sie die Konfiguration.
3	32775	Sicherheitsmodul nicht gefunden	Überprüfen Sie die Konfiguration. An der Sicherheitssteuerung: Überprüfen Sie die Adresseinstellungen für die Schalter der Sicherheitssteuerung. Starten Sie die Sicherheitssteuerung neu. Wenn der Fehler weiterhin besteht, tauschen Sie die Sicherheitssteuerung aus.
3	32776	Sicherheitsmodul weist falschen Typ auf	Überprüfen Sie die Konfiguration.
4	16640	Reservierte Adresseinstellung.	Warnung
4	16644	Bootprojekt nicht geladen, max. Spannungseinbruch erreicht	Starten Sie die Sicherheitssteuerung neu.
4	16648	Spannungseinbruchdaten fehlen oder korrupt. Standard-Spannungseinbruchdaten wurden von der Sicherheitssteuerung im Flash-Speicher gespeichert.	Warnung
4	16659	CRC-Fehler des Bootprojekts	Erzeugen Sie ein neues Bootprojekt und starten Sie die Sicherheitssteuerung neu.
4	16909	Flash-Schreibfehler (Produktionsdaten)	Warnung
4	16935	Mehr als eine Instanz von SF_WDOG_TIME_SET oder SF_MAX_POWER_DIP_SET	Warnung
4	16922	Keine oder falsche Konfigurationsdaten von der PM5x, Zustand RUN nicht möglich	Erzeugen Sie ein korrektes Bootprojekt für die PM5x.
4	17421	Flash-Schreibfehler (Bootprojekt)	Warnung
4	17677	Flash-Schreibfehler (Bootcode)	Warnung

Schweregrad	Fehlercode	Beschreibung	Abhilfe
4	17933	Flash-Schreibfehler (Firmware)	Warnung
4	18189	Flash-Schreibfehler (Passwort)	Warnung
4	18445	Flash-Schreibfehler (Nutzerdaten)	Warnung
4	18701	Flash-Schreibfehler (Nutzerdaten)	Warnung
4	18957	Flash-Schreibfehler (intern)	Warnung
4	19213	Flash-Schreibfehler (intern)	Warnung
4	19469	Flash-Schreibfehler (intern)	Warnung
4	32777	Programm wegen Konfigurationsfehler nicht gestartet.	Überprüfen Sie die Konfiguration.
4	32778	Programm nicht gestartet, in Sicherheitsmodul wird keine Anwendung ausgeführt	Überprüfen Sie die Konfiguration, laden Sie die Sicherheitsanwendung auf das Sicherheitsmodul.

C.2.2 Fehlermeldungen für Sicherheits-E/A-Module

Tab. 117: Fehlermeldungen für Sicherheits-E/A-Module (Kanal- oder Modulreintegration möglich)

Schweregrad	Fehlercode	Beschreibung	Abhilfe
3	3	Diskrepanzzeit abgelaufen	Überprüfen Sie Diskrepanzzeit, Kanal-Verdrahtung und Sensor.
3	12	Fehler Testimpuls	Überprüfen Sie Verdrahtung und Sensor.
3	13	Fehler Kanalübersprechen Testimpuls	Überprüfen Sie Verdrahtung und Sensor. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus. Wenden Sie sich an den technischen Support von ABB.
3	25	Fehler. Kanal reagiert nicht mehr.	Überprüfen Sie die Verdrahtung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
3	28	Fehler Kanalübersprechen	Überprüfen Sie die Verdrahtung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
3	260	Messwertüberschreitung am E/A-Modul	Überprüfen Sie die Kanal-Verdrahtung und die Sensor-Spannungsversorgung.
3	263	Messwertunterschreitung am Ein-/Ausgang	Überprüfen Sie die Kanal-Verdrahtung und die Sensor-Spannungsversorgung.
3	311	Differenz der Kanalwerte zu groß	Passen Sie die Toleranzfenster für Kanäle an. Überprüfen Sie die Kanal-Verdrahtung und die Sensor-Konfiguration.
3	525	Fehler Kanalrücklesen	Überprüfen Sie die Verdrahtung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
3	530	Fehler Kanalübersprechen	Überprüfen Sie die Verdrahtung des E/A-Moduls. Starten Sie ggf. das E/A-Modul neu. Wenn der Fehler weiterhin besteht, tauschen Sie das E/A-Modul aus.
3	16138	Prozessspannung zu hoch	Überprüfen Sie die Prozess-spannung.
3	16139	Prozessspannung zu niedrig	Überprüfen Sie die Prozess-spannung.

Schweregrad	Fehlercode	Beschreibung	Abhilfe
3	16148	PROFIsafe-Kommunikationsfehler	Starten Sie das E/A-Modul neu. Wenn der Fehler weiterhin besteht, wenden Sie sich an den technischen Support von ABB.
3	16153	Timeout des PROFIsafe-Watchdog	Starten Sie das E/A-Modul neu. Wenn der Fehler weiterhin besteht, verlängern Sie die PROFIsafe-Watchdog-Zeit.
3	16171	Interner Fehler im Gerät	Tauschen Sie das E/A-Modul aus.

Tab. 118: Fehlermeldungen der Sicherheits-E/A-Module (Kanal- oder Modulreintegration nicht möglich)

Schweregrad	Fehlercode	Beschreibung	Abhilfe
3	16146	Plausibilitätsprüfung fehlgeschlagen (iParameter)	Überprüfen Sie die Konfiguration.
3	16147	Prüfsummenfehler im E/A-Modul	Überprüfen Sie die Sicherheitskonfiguration und CRCs für I- und F-Parameter.
3	16154	Parameterwert	Überprüfen Sie Master oder Konfiguration.
3	16156	Konfiguration für F-Parameter stimmt nicht mit dem Wert des Adressschalters überein.	Überprüfen Sie die Konfiguration für F-Parameter des E/A-Moduls und den Wert des Adressschalters.

C.3 Konfiguration der AC500 V3-Standard-CPU-Parameter

Wenn die Standard-CPU gestoppt wird, wechselt die Sicherheits-CPU in den Zustand DEBUG STOP (nicht sicher) (Abb. 12 auf Seite 51) und die Sicherheits-E/A-Module wechseln umgehend in den Zustand RUN (Modulpassivierung mit einem Befehl) (Abb. 15 auf Seite 60).

Wenn die Sicherheits-CPU zu einem späteren Zeitpunkt in den Zustand DEBUG RUN (nicht sicher) wechselt, z. B. nachdem die Standard-CPU wieder in den Zustand RUN geschaltet wurde, wechseln die Sicherheits-E/A-Module umgehend in den Zustand RUN (OK) (Abb. 15 auf Seite 60) und liefern gültige Prozesswerte an die Sicherheits-CPU, ohne dass eine Reintegration erforderlich ist.

HINWEIS!

Das beschriebene Verhalten mit AC500 V3-Standard-CPUs unterscheidet sich vom Verhalten mit AC500 V2-Standard-CPUs. Wenn Sie mit AC500 V2-Standard-CPUs vertraut sind, müssen Sie sich der folgenden Unterschiede bewusst sein:

Wenn die AC500 V2-Standard-CPU gestoppt wird, wechselt die Sicherheits-CPU in den Zustand DEBUG STOP (nicht sicher) und die **Sicherheits-E/A-Module wechseln in den Zustand RUN (Modulpassivierung)** (Abb. 15 auf Seite 60).

Wenn die Sicherheits-CPU in den Zustand DEBUG RUN (nicht sicher) wechselt, müssen die **Sicherheits-E/As zunächst wieder integriert werden**, indem der Zustand RUN (Anforderung der Quittierung durch Anwender) durchlaufen wird (Abb. 15 auf Seite 60). Erst dann liefern sie die aktuellen gültigen Prozessausgänge an die Sicherheits-CPU.

Die folgenden Einstellungen der Konfiguration des Standardmoduls AC500 beeinflussen das Gesamtverhalten der Sicherheits- und der Standard-CPUs.

Einstellungen für Standard-CPU im Automation Builder:

- Registerkarte *„SPS-Einstellungen“*
  - *„Buszyklus-Task“*
- Registerkarte *„CPU-Parameter Parameter“*
  - *„Stopp bei Fehlerklasse“*
- Registerkarte *„I/O-Bus E/A-Abbild“*
  - *„Buszyklus-Task“*

Einstellungen für das Kommunikationsmodul im Automation Builder:

- Registerkarte *„PROFINET IO-Controller E/A-Abbild“* / *„PROFINET IO-Device E/A-Abbild“*
  - *„Buszyklus-Task“*

Die Einstellungen für diese Parameter beeinträchtigen nicht die Systemsicherheit.

„Buszyklus-Task“

Wir empfehlen dringend, die AC500 Benutzerdokumentation ↳ *[3]* zu diesem Thema zu lesen, um den Parameter *„Buszyklus-Task“* im Zusammenhang mit den oben aufgeführten Einstellungen und Abhängigkeiten von anderen Parametern besser zu verstehen.

Diese Einstellungen müssen sorgfältig bedacht werden. Einerseits, um Überlastungs-Szenarios an der Standard-CPU zu vermeiden. Andererseits, um die SFRT nicht zu überschreiten.

Eine einfache Möglichkeit zum Einrichten des Buszyklus

1. In der Registerkarte *„SPS-Einstellungen“* können Sie durch Zuordnung einer *„Buszyklus-Task“* eine globale Buszykluszeit einstellen.
2. Behalten Sie die Standardwerte für die Buszyklus-Tasks für I/O-Bus und Kommunikationsmodule bei.

Bei diesen Einstellungen werden die beiden Buszykluszeiten für I/O-Bus und Kommunikationsmodule von der Standard-CPU mit der Zykluszeit der zugeordneten Task gesteuert (in der Registerkarte *„SPS Einstellungen“*).

HINWEIS!

Der Wert des Sicherheits-CPU-Parameters *„Aktualisierungszyklus-Zeit“* ist die begrenzende Buszykluszeit für I/O-Bus und Kommunikationsmodule. Wenn höhere Werte für die Buszyklus-Tasks für I/O-Bus und Kommunikationsmodule zugeordnet sind, werden diese auf den geringeren Wert von *„Aktualisierungszyklus-Zeit“* begrenzt. Wenn geringere Werte für die Buszyklus-Tasks für I/O-Bus und Kommunikationsmodule zugeordnet sind, werden diese unverändert beibehalten.

HINWEIS!

Die Zykluszeiten für I/O-Bus und Kommunikationsmodule können sich auf die SFRT Ihres Systems auswirken ↳ *Kapitel 5.3 „Antwortzeit der Sicherheitsfunktion (= Safety Function Response Time)“ auf Seite 363*.

„Stopp bei Fehlerklasse“

Parameter in Registerkarte *„CPU-Parameter Parameter“* der Standard-CPU.

Wert *„Diagnose von mindestens Fehlerklasse 2“* (Standard)

Bei einem Fehler mit Schweregrad 1 oder 2 werden die Standard-CPU und die Sicherheits-CPU gestoppt. Sofern auf der betreffenden Sicherheits-CPU vorhanden, laufen PROFIsafe F-Host und F-Device-Stacks auf der Sicherheits-CPU mit Failsafe-Werten weiter.

Wert *„Diagnose von mindestens Fehlerklasse 3“*

Bei einem Fehler mit Schweregrad 1, 2 oder 3 werden die Standard-CPU und die Sicherheits-CPU gestoppt. Sofern auf der betreffenden Sicherheits-CPU vorhanden, laufen PROFIsafe F-Device und F-Device-Stacks auf der Sicherheits-CPU mit Failsafe-Werten weiter.

Wert *„Diagnose von mindestens Fehlerklasse 4“*

Bei einem Fehler mit Schweregrad 1, 2, 3 oder 4 werden die Standard-CPU und die Sicherheits-CPU gestoppt. Sofern auf der betreffenden Sicherheits-CPU vorhanden, laufen PROFIsafe F-Device und F-Device-Stacks auf der Sicherheits-CPU mit Failsafe-Werten weiter.

C.4 SPS-Befehle AC500 V3-Standard-CPU

Die folgenden SPS-Shell-Befehle (falls von der Firmware der aktuellen Standard-CPU unterstützt) von der Standard-CPU ändern den Zustand der Sicherheits-CPU:

- `reboot`
  Startet die Standard-CPU und somit auch die Sicherheits-CPU neu.
- `stopprg`, `resetprg`, `resetprgcold`
  Zwingen die Sicherheits-CPU, den Modus RUN (Sicherheitsmodus) zu verlassen und in den Modus DEBUG STOP (nicht sicher) zu wechseln.
- `startprg`
  Zwingt die Sicherheits-CPU, den Modus DEBUG STOP (nicht sicher) zu verlassen und in den Modus DEBUG RUN (nicht sicher) zu wechseln. Wenn sich die Sicherheits-CPU bereits im Modus RUN (Sicherheitsmodus) oder DEBUG RUN (nicht sicher) befindet, hat dieser SPS-Shell-Befehl keinen Einfluss auf die Sicherheits-CPU.

HINWEIS!
Die Fehlermeldungen der Sicherheits-CPU werden im Diagnosesystem der Standard-CPU zusammengefasst. Informationen zur Bearbeitung und Verwendung der Diagnosefunktionen der Standard-CPU finden Sie unter *[3]*.

C.5 Datenaustausch zwischen Sicherheits-CPU und AC500 V3-Standard-CPU

Optionen für den Datenaustausch zwischen Sicherheits-CPU und AC500 V3-Standard-CPU:

- Azyklischer nicht sicherer Datenaustausch: mehrere Zyklen der Sicherheits-CPU für die Übertragung der Daten erforderlich, max. 84 Bytes in jede Richtung ↳ *Anhang C.5.1 „Azyklischer nicht sicherer Datenaustausch“ auf Seite 449*
- Zyklischer nicht sicherer Datenaustausch: max. 3 Zyklen der Sicherheits-CPU für die Übertragung der Daten erforderlich, max. 2 kB in jede Richtung ↳ *Anhang C.5.2 „Zyklischer nicht sicherer Datenaustausch“ auf Seite 450*

GEFAHR!

Es wird nicht empfohlen, Datenwerte von der Standard-CPU auf die Sicherheits-CPU zu übertragen. Hierbei müssen die Endanwender zusätzliche prozessspezifische Validierungsverfahren in ihrem Sicherheitsprogramm definieren, um die Korrektheit der übertragenen nicht sicheren Daten zu überprüfen, wenn sie diese nicht sicheren Werte für Sicherheitsfunktionen verwenden möchten.

Datenwerte von der Sicherheits-CPU auf die Standard-CPU zu übertragen, z. B. für Diagnose und spätere Darstellung auf Bedienpanels, ist kein Problem.

C.5.1 Azyklischer nicht sicherer Datenaustausch

Verwenden Sie auf der Sicherheits-CPU die Funktionsbausteine SF_DPRAM_PM5XX_S_REC und SF_DPRAM_PM5XX_S_SEND ↳ *Kapitel 4.6.7.13 „SF_DPRAM_PM5XX_S_REC“ auf Seite 359* ↳ *Kapitel 4.6.7.14 „SF_DPRAM_PM5XX_S_SEND“ auf Seite 361*.

Verwenden Sie auf der Standard-CPU die Funktionsbausteine Sm560Send und Sm560Rec. Die Funktionsbausteine sind in der Bibliothek SM560Safety enthalten. Ausführliche Informationen finden Sie im Automation Builder unter Bibliotheksverwalter.

HINWEIS!

Übertragene Daten werden vertauscht

Der Datenaustausch erfolgt Byte für Byte, was dazu führt, dass für alle Datentypen, die größer als 1 Byte sind, im Zielsystem ein Daten-Byte-Tausch erfolgt.

Ursache hierfür sind die unterschiedlichen endianischen Systeme in der Sicherheits-CPU und Standard-CPU.

C.5.2 Zyklischer nicht sicherer Datenaustausch

GEFAHR!

Wenn der zyklische nicht sichere Datenaustausch zum Empfangen oder Senden von Sicherheitsdaten von der oder zur Sicherheits-CPU verwendet wird, sind die funktionalen sicherheitsbezogenen Anforderungen für SIL 3 (IEC 61508 und IEC 62061) und PL e (ISO 13849-1) für empfangene und gesendete Daten nicht erfüllt (unabhängig vom verwendeten applikativen Sicherheitskommunikationsprofil), da in der Sicherheits-CPU nur ein Mikroprozessor (keine 1oo2-Sicherheitsarchitektur im Hintergrund) für die Sende- und Empfangsrichtung zuständig ist.

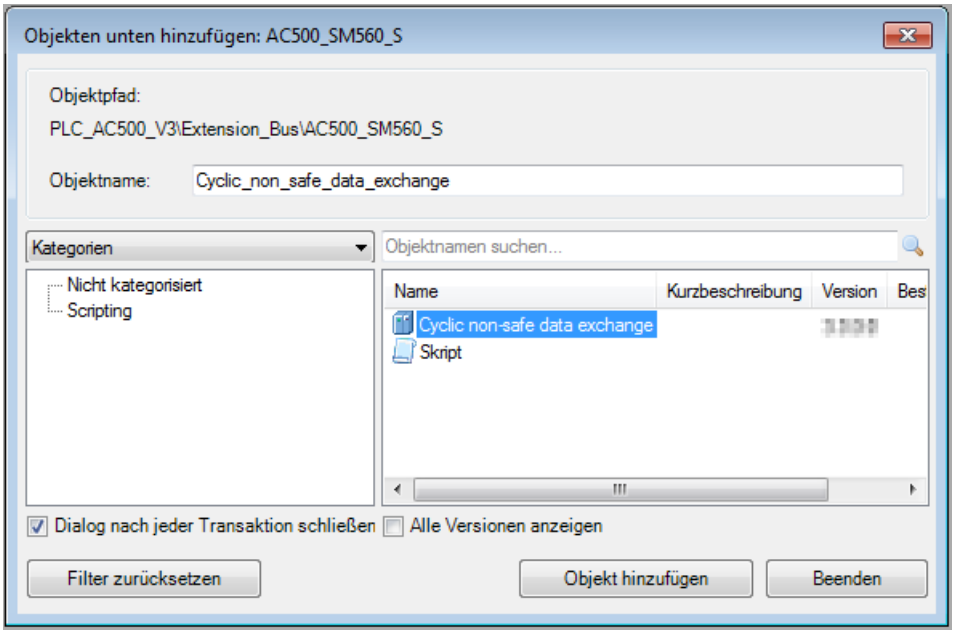
Wenden Sie sich an den technischen Support von ABB, um Informationen zum Erreichen von SIL 3 und PL e zu erhalten.

GEFAHR!

Durch eine entsprechende Konfiguration der Automation Builder-Benutzerverwaltung muss sichergestellt werden, dass ausschließlich Anwender der Sicherheitsgruppe zur Implementierung des zyklischen nicht sicheren Datenaustauschs berechtigt sind.

Zyklischen nicht sicheren Datenaustausch verwenden

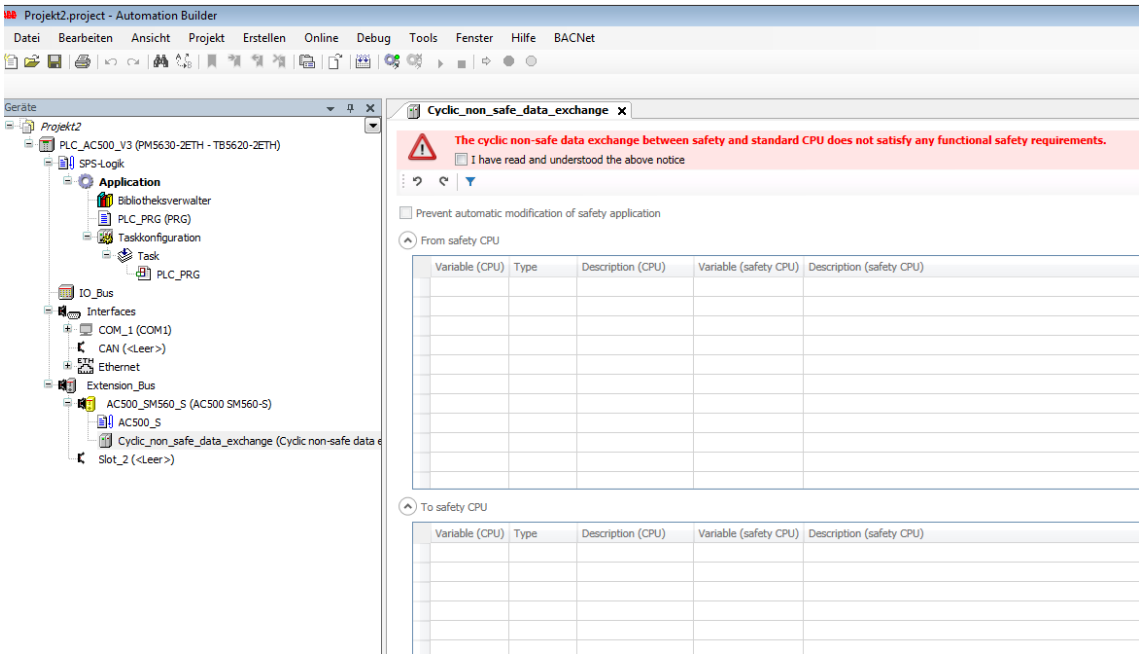
1. Führen Sie einen Rechtsklick auf den Knoten der Sicherheits-CPU aus und wählen Sie *„Objekt hinzufügen“*.
2. Wählen Sie *„Zyklischer nicht sicherer Datenaustausch“*.



⇨ Die Instanz für den zyklischen nicht sicheren Datenaustausch wird dem Knoten der Sicherheits-CPU hinzugefügt.

3. Doppelklicken Sie auf die Instanz *„Zyklischer nicht sicherer Datenaustausch“*.

   ⇨ Eine Warnung darüber, dass bei der Verwendung des zyklischen nicht sicheren Datenaustauschs die Sicherheitsanforderungen nicht erfüllt werden, wird angezeigt.



4. Lesen Sie die Warnung aufmerksam und bestätigen Sie sie.

   Wenn Sie keine Bestätigung vornehmen, können Sie keine Variablen definieren und daher auch nicht den Datenaustausch verwenden.

5. Details zum Ankreuzfeld *„Automatische Änderung der Sicherheitsanwendung verhindern“* finden Sie unter ↬ *Anhang C.5.2.1 „Migration von AC500 V2 in AC500 V3 (Kompatibilitätsmodus)“ auf Seite 455*.

6. Definieren Sie Variablen in den Tabellen. Eine ausführliche Beschreibung der Definition von Variablen finden Sie unter ↬ *„Variablen definieren“ auf Seite 452*.

   Tabelle *„Von Sicherheits-CPU“*: Variablen, die von der Sicherheits-CPU geschrieben und von der Standard-CPU gelesen werden sollen.

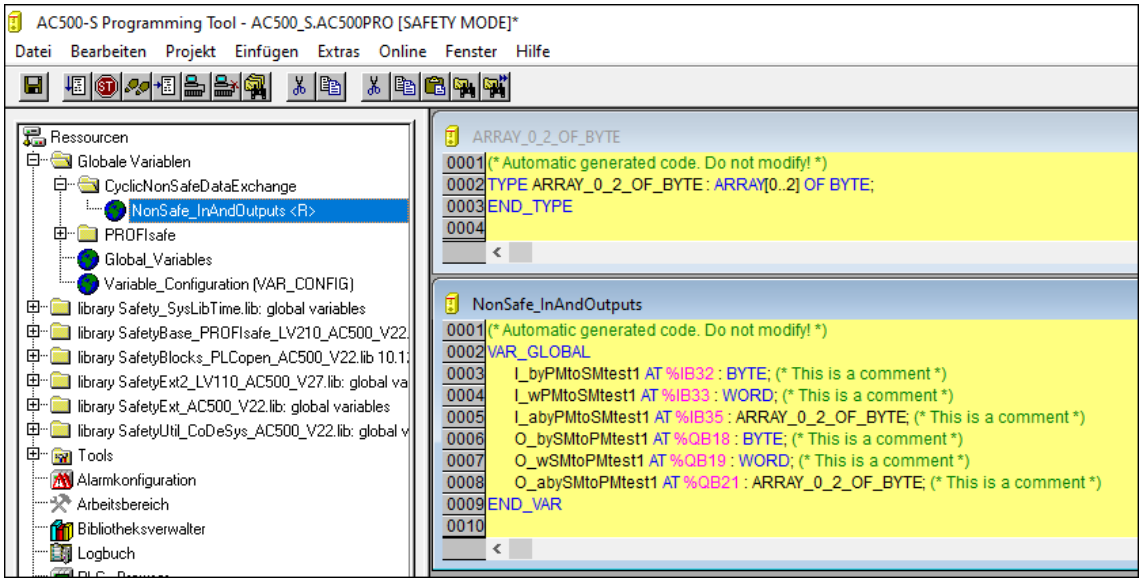
   Tabelle *„An Sicherheits-CPU“*: Variablen, die von der Standard-CPU geschrieben und von der Sicherheits-CPU gelesen werden sollen.

7. Übersetzen Sie die Standardanwendung im Automation Builder. Dies muss nach jeder Änderung für den zyklischen nicht sicheren Datenaustausch vorgenommen werden, z. B. nachdem neue Variablen hinzugefügt oder vorhandene Variablen aktualisiert wurden.

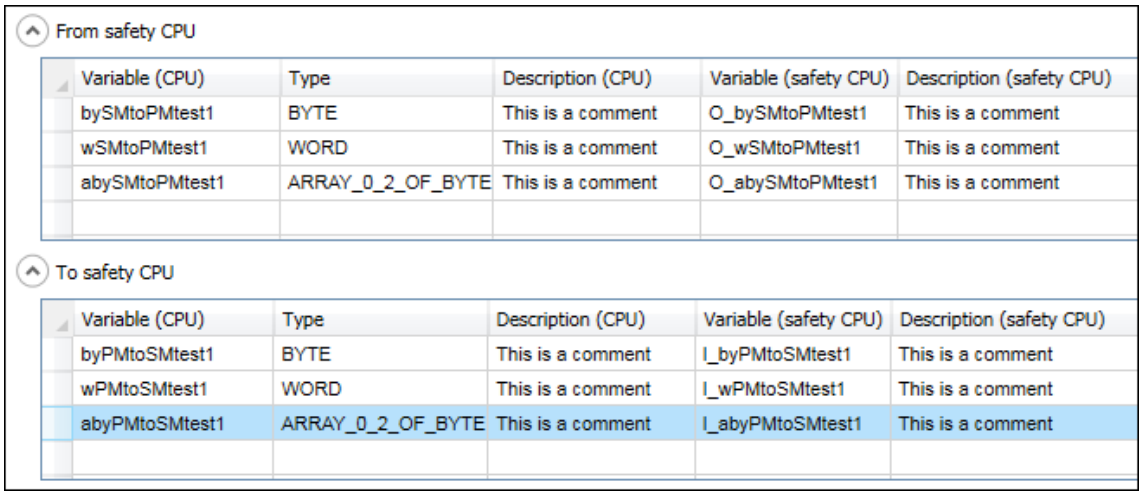
   ⇨ Die Variablen werden angelegt und können von der Standardanwendung verwendet werden.

8. Führen Sie einen Rechtsklick auf den Knoten der Sicherheitsanwendung (*„AC500_S“*) aus und wählen Sie *„Sicherheits-Konfigurationsdaten erzeugen“*. Dies muss nach jeder Änderung für den zyklischen nicht sicheren Datenaustausch vorgenommen werden, z. B. nachdem neue Variablen hinzugefügt oder vorhandene Variablen aktualisiert wurden.

   ⇨ Die Variablen werden erstellt und können in AC500-S Programming Tool verwendet werden.



Variablen definieren



Variable (CPU)	Variablenname für die Standardanwendung
Typ	Variablentyp für Standard- und Sicherheitsanwendung
Beschreibung (CPU)	Variablenbeschreibung für Standardanwendung
Variable (Sicherheits-CPU)	Variablenname für Sicherheitsanwendung
Beschreibung (Sicherheits-CPU)	Variablenbeschreibung für Sicherheitsanwendung

▷ Fügen Sie eine Variable für eine Standardanwendung in die letzte leere Zeile ein.

⇨ Der entsprechende Variablenname und die zugehörige Beschreibung für die Sicherheits-CPU werden automatisch hinzugefügt. Bei Bedarf können Sie sie unabhängig vom Namen und von der Beschreibung der nicht sicherheitsgerichteten Variable anpassen.

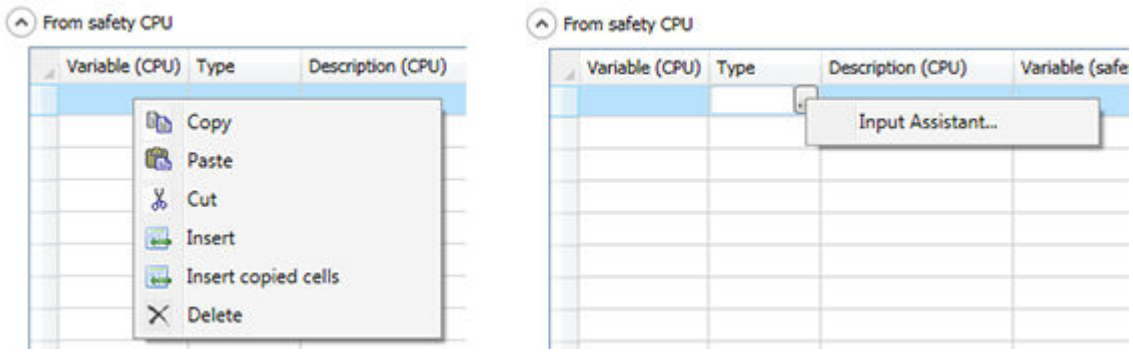
Um sie erneut zu synchronisieren, ändern sie manuell die betreffenden Einträge, die identisch sein sollen (d. h. die Variablennamen haben dieselbe Schreibweise). Die automatische Synchronisation ist wieder aktiv.

Unterstützte Datentypen:

- Standard-Datentypen, z. B. BYTE, WORD, INT
- Array-Datentypen
- Dateneinheitstypen (DUTs)
  DUT-Objekte werden automatisch während des Schritts *„Sicherheits-Konfigurationsdaten erzeugen“* in AC500-S Programming Tool erzeugt.
- Eine Kombination der obigen Datentypen

Unterstützte Funktionen beim Hinzufügen von Variablen:

- Variablen über das Kontextmenü und die standardmäßigen Windows-Tastenkombinationen ausschneiden, kopieren, einfügen, löschen und hinzufügen.
- Massendatenänderung, z. B. Variablen aus *.csv*-Datei kopieren und in .csv-Datei einfügen.
- Filter für jede Spalte.
- Änderungen rückgängig machen und wiederherstellen.
- *„Eingabehilfe“* für Variablennamen und -typ ↳ *[3]*.



HINWEIS!
Da die Variablennamen sowohl für die Sicherheitsanwendung als auch für die Standardanwendung erzeugt werden, wird die Verwendung von Variablennamen empfohlen, die die Übertragungsrichtung eindeutig beschreiben, z. B. „PMtoSM“ und „SMtoPM“ oder „toSM“ und „fromSM“.

GEFAHR!
Um die Sicherheitsprogrammierrichtlinien ↳ *Kapitel 4.4 „Sicherheitsprogrammierrichtlinien“ auf Seite 196* zu erfüllen, müssen Sie die folgenden Regeln beachten:

- Verwenden Sie die Präfixe „I_“ (Standardeingänge für die Sicherheits-CPU) und „O_“ (Standardausgänge für die Sicherheits-CPU) für die Variablennamen der Sicherheits-CPU. Der zyklische nicht sichere Datenaustausch ist nicht sicher. Verwenden Sie daher keine Sicherheitspräfixe ↳ *Kapitel 4.5 „Sicherheitscodeanalyse-Tool“ auf Seite 206*.
- Fügen Sie eine Beschreibung mit mindestens 10 Zeichen für jede Variable hinzu.

HINWEIS!
Bei der Verwendung des zyklischen nicht sicheren Datenaustauschs können Änderungen an der nicht sicherheitsgerichteten Programmierumgebung zu einer neuen Bootprojekt-CRC führen.

HINWEIS!

Der zyklische nicht sichere Datenaustausch verwendet den Speicher gemeinsam mit den PROFIsafe-Prozessdaten (z. B. Sicherheitseingänge und -ausgänge) der konfigurierten Sicherheits-E/A-Module und ist auf 2048 Bytes für jede Richtung begrenzt.

Automation Builder überprüft die Größe nicht bei der Definition der Variablen, jedoch während des Vorgangs *„Sicherheits-Konfigurationsdaten erzeugen“*.

HINWEIS!

Die Verwendung des zyklischen nicht sicheren Datenaustauschs beeinflusst die Zykluszeit der Standard-CPU. Beispielsweise kann der Datenaustausch mit granularen Variablen eine wesentliche Last für die Standard-CPU verursachen.

C.5.2.1 Migration von AC500 V2 in AC500 V3 (Kompatibilitätsmodus)

Sie können ein vorhandenes Automation Builder-Projekt mit AC500-V2-Standard-CPUs und Sicherheits-CPUs mit zyklischem nicht sicheren Datenaustausch in ein Projekt mit AC500-V3-Standard-CPUs migrieren. Wenn Sie die Sicherheitsanwendung nicht ändern möchten, markieren Sie das Ankreuzfeld *„Automatische Änderung der Sicherheitsanwendung verhindern“*. Wenn das Ankreuzfeld markiert ist, werden keine Variablenzuordnungen zwischen Sicherheits- und Standard-CPU vorgenommen.

In AC500-S Programming Tool werden weder ein Ordner mit der Bezeichnung *„CyclicNonSafeDataExchange“* noch die entsprechenden globalen Variablen generiert. Die Sicherheitsanwendung bleibt unverändert. Auf der Sicherheits-CPU erfolgt der Datenaustausch mit der Standard-CPU mit spezifischen Funktionsbausteinen. Weitere Informationen finden Sie in der entsprechenden Beschreibung, die unter *www.abb.com/plc – Dokumentnr. 3ADR025195M0202* verfügbar ist.

Auf der Standard-CPU erfolgt der Datenaustausch mit der Sicherheits-CPU über die in den Tabellen *„Von Sicherheits-CPU“* und *„An Sicherheits-CPU“* definierten Variablen.

HINWEIS!

Verwenden Sie bei Anwendung des Kompatibilitätsmodus ↳ *Anhang C.5.2.1 „Migration von AC500 V2 in AC500 V3 (Kompatibilitätsmodus)“ auf Seite 455* die Checkliste für den zyklischen nicht sicheren Datenaustausch mit AC500 V2 ↳ *Anhang B.5.2 „Zyklischer nicht sicherer Datenaustausch“ auf Seite 436*.

C.5.2.2 Fehlerbehebung

HINWEIS!

Wenn Sie den Kompatibilitätsmodus *Anhang C.5.2.1 „Migration von AC500 V2 in AC500 V3 (Kompatibilitätsmodus)“ auf Seite 455* verwenden, finden Sie weitere Informationen im Abschnitt zur Fehlerbehebung für den zyklischen nicht sicheren Datenaustausch mit AC500 V2 *Anhang B.5.2 „Zyklischer nicht sicherer Datenaustausch“ auf Seite 436*.

ID	Verhalten	Mögliche Ursache	Abhilfe
1.	Zyklische nicht sichere Variablen nicht aktualisiert auf Sicherheits- und/oder Standard-CPU.	Konfiguration wurde nicht aktualisiert.	Bereinigen und übersetzen Sie die Standard-CPU-Anwendung (neu). Erzeugen Sie Sicherheits-Konfigurationsdaten. Prüfen Sie auf Fehlermeldungen. Melden Sie sich an der Standard- und Sicherheits-CPU an und laden Sie die Anwendungen. Erzeugen Sie neue Bootprojekte für die Sicherheits-CPU und Standard-CPU.
2.	Zykluszeit der Sicherheits-CPU zu hoch für die jeweilige Anwendung.	Die Menge der zyklischen nicht sicheren Daten ist zu groß.	Überprüfen Sie, ob die konfigurierten Variablen für den betreffenden Anwendungsfall wirklich notwendig sind. Verringern Sie die Anzahl von Variablen, um die Leistung zu erhöhen.
3.	Verwendung der Variable nicht möglich, da sie nicht definiert oder nicht in der Eingabehilfe aufgelistet ist.	Konfiguration wurde nicht aktualisiert.	Bereinigen und übersetzen Sie die Standard-CPU-Anwendung (neu). Erzeugen Sie Sicherheits-Konfigurationsdaten. Prüfen Sie auf Fehlermeldungen.
4.	Die verwendete Größe einer Variable ist größer als erwartet.	In bestimmten Fällen sind ein oder mehr Füll-Bytes erforderlich, um das Daten-Alignment zu erreichen. Dies erfolgt im Automation Builder automatisch.	Nehmen Sie eine Reorganisation der Variablen in den verwendeten DUTs vor. Verwenden Sie nach Möglichkeit zuerst den größten Datentyp. Schlechtes Beispiel: ● VAR0 : BYTE ● VAR1 : DWORD ● VAR2 : BYTE ● VAR3 : WORD Gutes Beispiel: ● VAR1 : DWORD ● VAR3 : WORD ● VAR0 : BYTE ● VAR2 : BYTE
5.	Übersetzungsfehler.	Inkonsistente interne Daten.	Bereinigen Sie die Standardanwendung und übersetzen Sie sie erneut.

ID	Verhalten	Mögliche Ursache	Abhilfe
6.	Variable in Tabelle nicht hinzugefügt.	Fehlende oder falsche Werte für die Variablendefinition.	Geben Sie mindestens den Variablennamen *„Variable (CPU)“* und den entsprechenden Typ ein. Diese Werte sind obligatorisch.
7.	Fehlermeldung „… kein gültiges Zuordnungsziel“	Variable in der falschen Tabelle definiert.	Stellen Sie sicher, dass die in der Tabelle *„Von Sicherheits-CPU“* definierten Werte von der Sicherheits-CPU geschrieben werden und nur von der Standard-CPU gelesen werden können. Die in der Tabelle *„An Sicherheits-CPU“* definierten Werte werden von der Standard-CPU geschrieben und können nur von der Sicherheits-CPU gelesen werden.
8.	Fehlermeldung über Speicherüberlauf.	Der zyklische nicht sichere Datenaustausch verwendet den Speicher gemeinsam mit den PROFIsafe-Prozessdaten (z. B. Sicherheitseingänge und -ausgänge) der konfigurierten Sicherheits-E/A-Module und ist auf insgesamt 2048 Bytes für jede Richtung begrenzt. Automation Builder überprüft die Größe nicht bei der Definition der Variablen, jedoch während des Vorgangs *„Sicherheits-Konfigurationsdaten erzeugen“*.	Verringern Sie die Größe für den zyklischen nicht sicheren Datenaustausch und führen Sie erneut *„Sicherheits-Konfigurationsdaten erzeugen“* aus.

Wenn das Problem weiter besteht, wenden Sie sich an den technischen Support von ABB.

D Versionsinformationen

Jede freigegebene Sicherheits-CPU-Firmware ist mit allen früher freigegebenen Firmware-Versionen abwärtskompatibel. An bestehenden Sicherheitsprojekten sind keine Änderungen erforderlich (bestehende Bootprojekte können beibehalten werden).

Wenn Sie die neuen Funktionalitäten der neuesten Sicherheits-CPU-Firmware nutzen möchten, müssen Sie die neueste freigegebene Version von Automation Builder nutzen. Dadurch ist sichergestellt, dass Sie mit den neuesten Sicherheitsbibliotheken arbeiten.

D.1 Kompatibilität mit PROFIsafe Profilen

Tab. 119: Kompatibilität der Sicherheitsanwendungen mit PROFIsafe-Profil F-Host

PROFIsafe Profil	Automation Builder	Bibliothek SafetyBase_PROFIsafe	Firmware-Version von Sicherheits-CPU	Sicherheits-CPU
F-Host V2.4	Ab V1.0.0	Ab V1.0.1	Ab V1.0.0	SM560-S, SM560-S-FD-1, SM560-S-FD-4
F-Host V2.6	Ab V2.5.0	Ab V2.1.0	Ab V2.2.0	SM560-S, SM560-S-FD-1, SM560-S-FD-4

Tab. 120: Kompatibilität der Sicherheitsanwendungen mit PROFIsafe-Profil F-Device

PROFIsafe Profil	Automation Builder	Bibliothek SafetyDeviceExt	Bibliothek SafetyBase_PROFIsafe	Firmware-Version von Sicherheits-CPU	Sicherheits-CPU
F-Device V2.4	Ab V2.1.0	Ab V1.0.0	Ab V2.0.0	Ab V2.0.0	SM560-S-FD-1, SM560-S-FD-4
F-Device V2.6	Ab V2.5.0	Ab V1.0.0	Ab V2.0.0	Ab V2.2.0	SM560-S-FD-1, SM560-S-FD-4

D.2 Versionshistorie der Sicherheits-CPU-Firmware

Tab. 121: Versionshistorie der Sicherheits-CPU-Firmware

Firmware-Version von Sicherheits-CPU	Beschreibung der Version / Änderungen	Versionsdatum
V2.2.0	Erweiterungen für Kompatibilität mit PROFIsafe V2.6: Unterstützung von F-Devices mit PROFIsafe-V2.6-Kompatibilität FD-Varianten: Erweiterung um 2 neue F-Submodule mit PROFIsafe-V2.6-Kompatibilität: 12 Bytes Sicherheitsprozessdaten, 123 Bytes Sicherheitsprozessdaten Erweiterung um neue Sicherheitsfunktionen, die eine SIL3-kompatible Übertragung sicherheitsrelevanter Daten mit Hilfe der Mechanismen zum azyklischen/zyklischen nicht sicheren Datenaustausch (SF_CRC_INIT, SF_CRC_INPUT, SF_CRC_FINISH) ermöglichen. Voraussetzungen (verfügbar mit Automation Builder ab V2.3.0): Nutzung der neuen Sicherheitsbibliothek SafetyBase_PROFIsafe_LV210_AC500_V22.lib Nutzung der neuen Sicherheitsbibliothek SafetyExt2_LV110_AC500_V27.lib	2021
V2.1.0	Wartungs-Update, keine funktionalen Veränderungen	2019
V2.0.0	Einführung zweier neuer Sicherheits-CPU-Varianten mit F-Device-Funktionalität: Unterstützt werden die neuen Varianten SM560-S-FD-1 und SM560-S-FD-4 mit F-Device-Funktionalität Erweiterung um neue Sicherheitsfunktionen (SF_SAFE_STOP, SF_BOOTPROJECT_CRC, SF_MAX_POWER_DIP_GET_CFG). Voraussetzungen (verfügbar mit Automation Builder ab V2.1.0): Nutzung der neuen Sicherheitsbibliothek SafetyBase_PROFIsafe_LV200_AC500_V22.lib Nutzung der neuen Sicherheitsbibliothek SafetyDeviceExt_LV100_PROFIsafe_AC500_V27.lib Unterstützung der neuen Sicherheitsbibliothek SafetyExt2_LV100_AC500_V27.lib	2018
V1.0.0	Erste Freigabeversion für SM560-S	2012

D.3 Versionshistorie von Sicherheitsbibliotheken

Alte Versionen dürfen NICHT für neue AC500-S-Anwendungsprojekte verwendet werden
↳ *Kapitel 4.6.1 „Übersicht“ auf Seite 207*.

Bibliotheken, die ausschließlich zur internen Nutzung bestimmt sind, sind nicht in der Versionshistorie aufgelistet.

Tab. 122: Versionshistorie der Bibliothek SafetyBase_PROFIsafe

Version der Sicherheitsbibliothek	Beschreibung der Version / Änderungen	Voraussetzungen	Versionsdatum
V2.1.0	SafetyBase_PROFIsafe_LV210_AC500_V22.lib ● Erweiterungen für Kompatibilität mit PROFIsafe V2.6 ● Konfigurierbare Zeitüberschreitung beim Start für PROFIsafe-Kommunikation ● Unterstützung von 32-bit-Datentypen für F-Device-Prozesssignale	Automation Builder 2.5.0 mit Sicherheits-CPU-Firmware V2.2.0	2021
V2.0.0	SafetyBase_PROFIsafe_LV200_AC500_V22.lib ● Erweiterung zur Unterstützung von F-Device V2.4 in den neuen Varianten SM560-S-FD-1/ SM560-S-FD-4) ● CRC der Bibliothek: 1d881052	Automation Builder 2.1.0 mit Sicherheits-CPU-Firmware V2.0.0	2018
V1.0.1	SafetyBase_PROFIsafe_AC500_V22_Ext.lib ● Wartungsupdate (CRC-Berechnung korrigiert für 0-Telegramme) ● CRC der Bibliothek: f34d9a48	Automation Builder 1.0.0 mit Sicherheits-CPU-Firmware V1.0.0	2017
V1.0.0	SafetyBase_PROFIsafe_AC500_V22.lib ● Erste Freigabeversion (F-Host-Unterstützung für PROFIsafe V2.4 F-Devices)	Automation Builder 1.0.0 mit Sicherheits-CPU-Firmware V1.0.0	2012

Tab. 123: Versionshistorie der Bibliothek SafetyBlocks_PLCopen

Version der Sicherheitsbibliothek	Beschreibung der Version / Änderungen	Voraussetzungen	Versionsdatum
V1.0.0	SafetyBlocks_PLCopen_AC500_V22.lib ● Erste Freigabeversion	Automation Builder 1.0.0 mit Sicherheits-CPU-Firmware V1.0.0	2012

Tab. 124: Versionshistorie der Bibliothek SafetyDeviceExt

Version der Sicherheitsbibliothek	Beschreibung der Version / Änderungen	Voraussetzungen	Versionsdatum
V1.0.0	SafetyDeviceExt_LV100_PROFIsafe_AC500_V27.lib ● Erste Freigabeversion (F-Device-Unterstützung in den neuen Varianten SM560-S-FD-1/ SM560-S-FD-4)	Automation Builder 2.1.0 mit Sicherheits-CPU-Firmware V2.0.0	2018

Tab. 125: Versionshistorie der Bibliothek SafetyExt2

Version der Sicherheitsbibliothek	Beschreibung der Version / Änderungen	Voraussetzungen	Versionsdatum
V1.1.0	SafetyExt2_LV110_AC500_V27.lib ● Erweiterung um zusätzliche Funktionsbausteine (SF_CRC_INIT, SF_CRC_INPUT, SF_CRC_FINISH)	Automation Builder 2.3.0 mit Sicherheits-CPU-Firmware V2.2.0	2021
V1.0.0	SafetyExt2_LV100_AC500_V27.lib ● Erste Freigabeversion ● CRC der Bibliothek: f3eb2fbc	Automation Builder 2.1.0 mit Sicherheits-CPU-Firmware V2.0.0	2018

Tab. 126: Versionshistorie der Bibliothek SafetyExt

Version der Sicherheitsbibliothek	Beschreibung der Version / Änderungen	Voraussetzungen	Versionsdatum
V1.0.0	SafetyExt_AC500_V22.lib ● Erste Freigabeversion	Automation Builder 1.0.0 mit Sicherheits-CPU-Firmware V1.0.0	2012

ABB AG
Eppelheimer Str. 82
69123 Heidelberg, Deutschland
Telefon: +49 (0)6221 701 1444
Telefax: +49 (0)6221 701 1382
E-Mail: plc.support@de.abb.com
new.abb.com/plc

3ADR025091M0110, 14, de_DE