



SMART
SAFETY.

Handbuch

Protokolle

PROFINET, PROFIsafe,
PROFIBUS DP



Alle in diesem Handbuch genannten HIMA Produkte sind mit dem Warenzeichen geschützt. Dies gilt ebenfalls, soweit nicht anders vermerkt, für weitere genannte Hersteller und deren Produkte.

HIQuad®, HIQuad®X, HIMax®, HIMatrix®, SILworX®, XMR®, HICore® und FlexSILon® sind eingetragene Warenzeichen der HIMA Paul Hildebrandt GmbH.

Alle technischen Angaben und Hinweise in diesem Handbuch wurden mit größter Sorgfalt erarbeitet und unter Einschaltung wirksamer Kontrollmaßnahmen zusammengestellt. Bei Fragen bitte direkt an HIMA wenden. Für Anregungen, z. B. welche Informationen noch in das Handbuch aufgenommen werden sollen, ist HIMA dankbar.

Technische Änderungen vorbehalten. Ferner behält sich HIMA vor, Aktualisierungen des schriftlichen Materials ohne vorherige Ankündigungen vorzunehmen.

Alle aktuellen Handbücher können über die E-Mail-Adresse documentation@hima.com angefragt werden.

© Copyright 2020, HIMA Paul Hildebrandt GmbH

Alle Rechte vorbehalten.

Kontakt

HIMA Paul Hildebrandt GmbH

Postfach 1261

68777 Brühl

Tel.: +49 6202 709-0

Fax: +49 6202 709-107

E-Mail: info@hima.com

Revisions-index	Änderungen	Art der Änderung	
		technisch	redaktionell
11.00	Neu erstellt für SILworX V11		
11.01	Aktualisierte Ausgabe zu SILworX V12 Geändert: Kapitel 3.6.2	X	X

Inhaltsverzeichnis

1	Einleitung	7
1.1	Aufbau und Gebrauch des Handbuchs	7
1.2	Zielgruppe	8
1.3	Darstellungskonventionen	9
1.3.1	Sicherheitshinweise	9
1.3.2	Gebrauchshinweise	10
1.4	Safety Lifecycle Services	11
2	Sicherheit	12
2.1	Bestimmungsgemäßer Einsatz	12
2.2	Restrisiken	12
2.3	Sicherheitsvorkehrungen	12
2.4	Notfallinformationen	12
2.5	Automation Security bei HIMA Systemen	12
3	PROFINET IO	14
3.1	PROFINET IO-Funktionsbausteine	14
3.2	Steuerung des Consumer/Provider Status (IOxS)	15
3.2.1	Steuervariablen im HIMA Controller	15
3.2.2	Steuervariablen im HIMA DO Device	15
3.2.3	Steuervariablen im HIMA DI Device	15
3.3	PROFIsafe	16
3.3.1	PROFIsafe bei HIMA Steuerungen	16
3.3.2	PROFIsafe-Control-Byte und Status-Byte	17
3.3.3	F_WD_Time (PROFIsafe-Watchdog-Zeit)	17
3.3.4	SFRT (Safety Function Response Time)	19
3.4	Auflagen für den sicheren Betrieb von PROFIsafe	21
3.4.1	Adressierung	21
3.4.2	Netzwerkaspekte	22
3.5	PROFINET IO-Controller und PROFIsafe-F-Host	23
3.5.1	Systemanforderung PROFINET IO-Controller	23
3.5.2	PROFINET IO-Controller und PROFIsafe-Host Eigenschaften	23
3.6	PROFINET IO/ PROFIsafe Beispiel	24
3.6.1	Anlegen des PROFINET IO-Controller in SILworX	25
3.6.2	Anlegen der GSDML-Bibliothek in SILworX	25
3.6.3	Konfiguration des Device im PROFINET IO-Controller	26
3.6.4	Konfiguration des Device Access Point (DAP) Moduls	27
3.6.5	Konfiguration der PROFINET IO-Device Module	27
3.6.6	Konfiguration der PROFIsafe-IO-Device-Module	28
3.6.7	Ermitteln der PROFINET IO-Devices im Netzwerk	29
3.7	Menüfunktionen des PROFINET IO-Controllers	30
3.7.1	Beispiel für Strukturbaum des PROFINET IO-Controllers	30
3.7.2	PROFINET IO-Controller	30
3.7.3	PROFINET IO-Device (im Controller)	32
3.7.4	DAP Modul (Device Access Point Modul)	32
3.7.5	Input/Output PROFINET IO-Module	34
3.7.6	Submodul Input	35

3.7.7	Submodul Output	40
3.7.8	Submodul Inputs und Outputs	42
3.7.9	Application Relation (Eigenschaften)	44
3.7.10	Alarm CR (Eigenschaften)	45
3.7.11	Input CR (Eigenschaften)	46
3.8	PROFINET IO-Device	49
3.8.1	Systemanforderung	49
3.8.2	PROFINET IO-Device Eigenschaften	49
3.9	PROFINET IO/PROFIsafe Beispiel	50
3.9.1	Konfiguration des PROFINET IO-Device in SILworX	50
3.9.2	Konfiguration des PROFINET IO-Device Eingangsmoduls	51
3.9.3	Konfiguration des PROFIsafe-Device Ausgangsmoduls	51
3.9.4	Verifikation der PROFINET IO-Device Konfiguration	51
3.10	Menüfunktionen des PROFINET IO-Device	53
3.10.1	Menüfunktion Eigenschaften	53
3.10.2	PROFINET IO-Module	54
3.10.3	HIMA PROFIsafe-Module	55
3.10.4	PROFINet IO und PROFIsafe-Modul	56
4	PROFIBUS DP	59
4.1	HIMA PROFIBUS DP Master	60
4.1.1	Anlegen eines HIMA PROFIBUS DP Master	60
4.2	PROFIBUS DP Beispiel	61
4.2.1	PROFIBUS DP Slave Konfigurieren	61
4.2.2	PROFIBUS DP Master Konfigurieren	63
4.3	Menüfunktionen des PROFIBUS DP Master	70
4.3.1	Edit	70
4.3.2	Menüfunktion Eigenschaften	71
4.4	Die PROFIBUS DP Buszugriffsverfahren	76
4.4.1	Master/Slave-Protokoll	76
4.4.2	Token-Protokoll	76
4.4.3	Token-Umlaufzeit (Ttr)	76
4.4.4	Token-Umlaufzeit Ttr berechnen	77
4.5	Isochroner PROFIBUS DP Zyklus (ab DP-V2)	79
4.5.1	Isochron Mode (ab DP-V2)	80
4.5.2	Isochron Mode Sync (ab DP-V2)	80
4.5.3	Isochrone Mode Freeze (ab DP-V2)	80
4.6	Menüfunktionen PROFIBUS DP Slave (im Master)	81
4.6.1	Anlegen eines PROFIBUS DP Slave (im Master)	81
4.6.2	Edit	81
4.6.3	Eigenschaften	82
4.7	GSD-Datei einlesen	86
4.8	User-Parameter bearbeiten	87
4.9	PROFIBUS Funktionsbausteine	89
4.9.1	Funktionsbaustein MSTAT	90
4.9.2	Funktionsbaustein RALRM	95
4.9.3	Funktionsbaustein RDIAG	99
4.9.4	Funktionsbaustein RDREC	103
4.9.5	Funktionsbaustein SLACT	106
4.9.6	Funktionsbaustein WRREC	109

4.10	PROFIBUS Hilfsfunktionsbausteine	112
4.10.1	Hilfsfunktionsbaustein ACTIVE	112
4.10.2	Hilfsfunktionsbaustein ALARM	113
4.10.3	Hilfsfunktionsbaustein DEID	114
4.10.4	Hilfsfunktionsbaustein ID	115
4.10.5	Hilfsfunktionsbaustein NSLOT	116
4.10.6	Hilfsfunktionsbaustein SLOT	116
4.10.7	Hilfsfunktionsbaustein STDDIAG	117
4.11	Fehlercodes der Funktionsbausteine	119
4.12	Control Panel (PROFIBUS DP Master)	120
4.12.1	Kontextmenü (PROFIBUS Master)	120
4.12.2	Kontextmenü (PROFIBUS DP Slave)	120
4.12.3	Anzeigefeld (PROFIBUS Master)	121
4.12.4	Zustand des PROFIBUS DP Master	122
4.12.5	Verhalten des PROFIBUS DP Master	122
4.12.6	Funktion der FBx LED beim PROFIBUS DP Master	123
4.12.7	Funktion der FAULT LED beim PROFIBUS DP Master (Nur HIMax)	123
4.13	HIMA PROFIBUS DP Slave	124
4.13.1	Anlegen eines HIMA PROFIBUS DP Slave	124
4.14	Menüfunktionen PROFIBUS DP Slave	125
4.14.1	Edit	125
4.14.2	Eigenschaften	126
4.15	Control Panel (PROFIBUS DP Slave)	128
4.15.1	Anzeigefeld (PROFIBUS DP Slave)	128
4.16	Funktion der FBx LED beim PROFIBUS DP Slave	129
4.17	Funktion der FAULT LED beim PROFIBUS DP Slave (Nur HIMax)	129
5	Allgemein	130
5.1	Maximale Kommunikationszeitscheibe	130
5.1.1	Ermitteln der maximalen Dauer der Kommunikationszeitscheibe	130
5.2	Lastbegrenzung	130
5.3	Konfiguration der Funktionsbausteine	131
5.3.1	Beschaffung der Funktionsbausteinbibliotheken	131
5.3.2	Konfiguration der Funktionsbausteine im Anwenderprogramm	131
5.3.3	Konfiguration der Funktionsbausteine im Strukturbaum von SILworX	132
	Anhang	133
	Glossar	133
	Abbildungsverzeichnis	134
	Tabellenverzeichnis	135

1 Einleitung

Dieses Handbuch beschreibt die Eigenschaften und die Konfiguration der PROFINET, PROFIsafe und PROFIBUS DP Protokolle für die sicherheitsbezogenen HIMA Steuerungssysteme mit dem Programmierwerkzeug SILworX.

Die Kenntnis von Vorschriften und das technisch einwandfreie Umsetzen der in diesem Handbuch enthaltenen Hinweise durch qualifiziertes Personal sind Voraussetzung für die Planung, Projektierung, Programmierung, Installation, Inbetriebnahme, Betrieb und Instandhaltung der HIMA Steuerungen.

Bei nicht qualifizierten Eingriffen in die Geräte, bei Abschalten oder Umgehen (Bypass) von Sicherheitsfunktionen oder bei Nichtbeachtung von Hinweisen dieses Handbuchs (und dadurch verursachten Störungen oder Beeinträchtigungen von Sicherheitsfunktionen) können schwere Personen-, Sach- oder Umweltschäden eintreten, für die HIMA keine Haftung übernehmen kann.

HIMA Automatisierungsgeräte werden unter Beachtung der einschlägigen Sicherheitsnormen entwickelt, gefertigt und geprüft. Nur für die in den Beschreibungen vorgesehenen Einsatzfälle mit den spezifizierten Umgebungsbedingungen verwenden.

1.1 Aufbau und Gebrauch des Handbuchs

Das Handbuch enthält die folgenden Hauptkapitel:

- Einleitung
- Sicherheit
- Produktbeschreibung
- PROFINET
- PROFIsafe
- PROFIBUS DP

Zusätzlich sind die folgenden Dokumente zu beachten:

Name	Inhalt	Dokumenten-Nr.
HIMax Systemhandbuch	Hardware-Beschreibung HIMax System	HI 801 000 D
HIMax Sicherheitshandbuch	Sicherheitsfunktionen HIMax Systems	HI 801 002 D
HIMatrix Sicherheitshandbuch	Sicherheitsfunktionen HIMatrix Systems	HI 800 022 D
HIMatrix Kompakt Systemhandbuch	Hardware-Beschreibung HIMatrix Kompakt System	HI 800 140 D
HIMatrix Modular Systemhandbuch	Hardware-Beschreibung HIMatrix Modular System F 60	HI 800 190 D
HIQuad X Systemhandbuch	Hardware-Beschreibung HIQuad X System	HI 803 210 D
HIQuad X Sicherheitshandbuch	Sicherheitsfunktionen HIQuad X System	HI 803 208 D
Automation Security Handbuch	Beschreibung von Automation Security Aspekten bei HIMA Systemen	HI 801 372 D
SILworX Erste Schritte	Einführung in SILworX	HI 801 102 D

Tabelle 1: Zusätzlich geltende Handbücher

Alle aktuellen Handbücher können über die E-Mail-Adresse documentation@hima.com angefragt werden. Für registrierte Kunden stellt HIMA die Dokumentationen im Download-Bereich <https://www.hima.com/de/downloads/> zur Verfügung.

1.2 Zielgruppe

Dieses Dokument wendet sich an Planer, Projektoren, Programmierer und Personen, die zur Inbetriebnahme, zur Wartung und zum Betreiben von Automatisierungsanlagen berechtigt sind. Vorausgesetzt werden spezielle Kenntnisse auf dem Gebiet der sicherheitsbezogenen Automatisierungssysteme.

1.3 Darstellungskonventionen

Zur besseren Lesbarkeit und zur Verdeutlichung gelten in diesem Dokument folgende Schreibweisen:

Fett	Hervorhebung wichtiger Textteile. Bezeichnungen von Schaltflächen, Menüpunkten und Registern im Programmierwerkzeug, die angeklickt werden können.
<i>Kursiv</i>	Parameter und Systemvariablen, Referenzen.
<code>Courier</code>	Wörtliche Benutzereingaben.
RUN	Bezeichnungen von Betriebszuständen (Großbuchstaben).
Kap. 1.2.3	Querverweise sind Hyperlinks, auch wenn sie nicht besonders gekennzeichnet sind. Im elektronischen Dokument (PDF): Wird der Mauszeiger auf einen Hyperlink positioniert, verändert er seine Gestalt. Bei einem Klick springt das Dokument zur betreffenden Stelle.

Sicherheits- und Gebrauchshinweise sind besonders gekennzeichnet.

1.3.1 Sicherheitshinweise

Um ein möglichst geringes Risiko zu gewährleisten, sind die Sicherheitshinweise unbedingt zu befolgen.

Die Sicherheitshinweise im Dokument sind wie folgt dargestellt.

- Signalwort: Warnung, Vorsicht, Hinweis.
- Art und Quelle des Risikos.
- Folgen bei Nichtbeachtung.
- Vermeidung des Risikos.

Die Bedeutung der Signalworte ist:

- Warnung: Bei Missachtung droht schwere Körperverletzung bis Tod.
- Vorsicht: Bei Missachtung droht leichte Körperverletzung.
- Hinweis: Bei Missachtung droht Sachschaden.

SIGNALWORT



Art und Quelle des Risikos!
Folgen bei Nichtbeachtung.
Vermeidung des Risikos.

HINWEIS



Art und Quelle des Schadens!
Vermeidung des Schadens.

1.3.2 Gebrauchshinweise

Zusatzinformationen sind nach folgendem Beispiel aufgebaut:

i

An dieser Stelle steht der Text der Zusatzinformation.

Nützliche Tipps und Tricks erscheinen in der Form:

TIPP

An dieser Stelle steht der Text des Tipps.

1.4 Safety Lifecycle Services

HIMA unterstützt Sie in allen Phasen des Sicherheitslebenszyklus einer Anlage: Von der Planung, der Projektierung, über die Inbetriebnahme, bis zur Aufrechterhaltung der Sicherheit.

Für Informationen und Fragen zu unseren Produkten, zu Funktionaler Sicherheit und zu Automation Security stehen Ihnen die Experten des HIMA Support zur Verfügung.

Für die geforderte Qualifizierung gemäß Sicherheitsstandards führt HIMA produkt- oder kundenspezifische Seminare in eigenen Trainingszentren oder bei Ihnen vor Ort durch. Das aktuelle Seminarangebot zu Funktionaler Sicherheit, Automation Security und zu HIMA Produkten finden Sie auf der HIMA Webseite.

Safety Lifecycle Services:

Onsite+ / Vor-Ort-Engineering	In enger Abstimmung mit Ihnen führt HIMA vor Ort Änderungen oder Erweiterungen durch.
Startup+ / Vorbeugende Wartung	HIMA ist verantwortlich für die Planung und Durchführung der vorbeugenden Wartung. Wartungsarbeiten erfolgen gemäß der Herstellervorgabe und werden für den Kunden dokumentiert.
Lifecycle+ / Lifecycle-Management	Im Rahmen des Lifecycle-Managements analysiert HIMA den aktuellen Status aller installierten Systeme und erstellt konkrete Empfehlungen für Wartung, Upgrade und Migration.
Hotline+ / 24-h-Hotline	HIMA Sicherheitsingenieure stehen Ihnen für Problemlösung rund um die Uhr telefonisch zur Verfügung.
Standby+ / 24-h-Rufbereitschaft	Fehler, die nicht telefonisch gelöst werden können, werden von HIMA Spezialisten innerhalb vertraglich festgelegter Zeitfenster bearbeitet.
Logistic+/ 24-h-Ersatzteilservice	HIMA hält notwendige Ersatzteile vor und garantiert eine schnelle und langfristige Verfügbarkeit.

Ansprechpartner:

Safety Lifecycle Services	https://www.hima.com/de/unternehmen/ansprechpartner-weltweit/
Technischer Support	https://www.hima.com/de/produkte-services/support/
Seminarangebot	https://www.hima.com/de/produkte-services/seminarangebot/

2 Sicherheit

Sicherheitsinformationen, Hinweise und Anweisungen in diesem Dokument unbedingt lesen. Das Produkt nur unter Beachtung aller Richtlinien und Sicherheitsrichtlinien einsetzen.

Dieses Produkt wird mit SELV oder PELV betrieben. Vom Produkt selbst geht kein Risiko aus. Einsatz im Ex-Bereich nur mit zusätzlichen Maßnahmen erlaubt.

2.1 Bestimmungsgemäßer Einsatz

Für den Einsatz von HIMA Steuerungen, sind die jeweiligen Bedingungen einzuhalten, siehe zusätzlich geltende Handbücher in Tabelle 1.

2.2 Restrisiken

Von einem HIMA System selbst geht kein Risiko aus.

Restrisiken können ausgehen von:

- Fehlern in der Projektierung.
- Fehlern im Anwenderprogramm.
- Fehlern in der Verdrahtung.

2.3 Sicherheitsvorkehrungen

Am Einsatzort geltende Sicherheitsbestimmungen beachten und vorgeschriebene Schutzausrüstung tragen.

2.4 Notfallinformationen

Ein HIMA System ist Teil der Sicherheitstechnik einer Anlage. Der Ausfall einer Steuerung bringt die Anlage in den sicheren Zustand.

Im Notfall ist jeder Eingriff, der die Sicherheitsfunktion des HIMA Systems verhindert, verboten.

2.5 Automation Security bei HIMA Systemen

Industrielle Steuerungen müssen gegen IT-typische Problemquellen geschützt werden. Diese Problemquellen sind:

- Angreifer innerhalb und außerhalb der Kundenanlage
- Bedienungsfehler
- Software-Fehler

Die Anforderungen der Sicherheits- und Anwendungsnormen bezüglich des Schutzes vor Manipulationen sind zu beachten. Die Autorisierung von Personal und die notwendigen Schutzmaßnahmen unterliegen der Verantwortung des Betreibers.

WARNUNG



Personenschaden durch unbefugte Manipulation an der Steuerung möglich!

Die Steuerung ist gegen unbefugte Zugriffe zu schützen!

Beispielsweise:

- die Standardeinstellungen für Login und Passwort ändern.
- physischen Zugang zur Steuerung und zum PADT kontrollieren!

Sorgfältige Planung sollte die zu ergreifenden Maßnahmen nennen. Nach erfolgter Risikoanalyse sind die benötigten Maßnahmen zu ergreifen. Solche Maßnahmen sind beispielsweise:

- Sinnvolle Einteilung von Benutzergruppen.
- Gepflegte Netzwerkpläne helfen sicherzustellen, dass secure Netzwerke dauerhaft von öffentlichen Netzwerken getrennt sind und, falls nötig, nur ein definierter Übergang (z. B. über eine Firewall oder eine DMZ) besteht.
- Verwendung geeigneter Passwörter.

Ein regelmäßiges Review (z. B. jährlich) der Security-Maßnahmen ist ratsam.

Die für eine Anlage geeignete Umsetzung der benötigten Maßnahmen liegt in der Verantwortung des Anwenders!

Weitere Einzelheiten siehe HIMA Automation Security Handbuch HI 801 372 D.

3 PROFINET IO

PROFINET IO ist das auf Ethernet-Technologie basierte Übertragungsprotokoll der PROFIBUS Nutzerorganisation für die Automatisierung.

Wie bei PROFIBUS DP werden die dezentralen Feldgeräte bei PROFINET IO über eine Gerätebeschreibung (GSDML-Datei) in SILworX eingebunden.

Der PROFINET IO-Controller entspricht der Conformance Class A und unterstützt die Non-Real-Time-Kommunikation (NRT) und die Real-Time-Kommunikation (RT) zu den PROFINET IO-Devices. Die Real-Time Kommunikation wird für den zeitkritischen Datenaustausch und die Non-Real Kommunikation wird für nicht zeitkritische Vorgänge (z. B. azyklisches Lesen/Schreiben) verwendet.

Eine redundante PROFINET IO-Verbindung kann nur durch die Konfiguration eines zweiten PROFINET IO-Controller/Device und Anpassungen im Anwenderprogramm erreicht werden.

3.1 PROFINET IO-Funktionsbausteine

Zum azyklischen Datenaustausch stehen Ihnen in SILworX die funktional gleichen Funktionsbausteine wie bei PROFIBUS DP zur Verfügung.

Folgende PROFINET IO-Funktionsbausteine stehen zur Verfügung:

Funktionsbaustein	Beschreibung der Funktion
MSTAT 4.9.1	Zustand des Controllers durch das Anwenderprogramm steuern
RALRM 4.9.2	Alarmmeldungen der Devices lesen
RDREC 4.9.4	Datensätze der Devices lesen
SLACT 4.9.5	Zustand der Devices durch das Anwenderprogramm steuern
WRREC 4.9.6	Datensätze der Devices schreiben

Tabelle 2: Übersicht PROFINET IO-Funktionsbausteine

Die PROFINET IO-Funktionsbausteine werden wie die PROFIBUS DP Funktionsbausteine parametrieren, siehe ab Kapitel 4.9.

3.2 Steuerung des Consumer/Provider Status (IOxS)

Mit denen in diesem Kapitel beschriebenen Systemvariablen kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) gesteuert werden. Ist keine Steuerung durch das Anwenderprogramm gewünscht, so müssen die Ausgangsvariablen eines Geräts mit einer Konstanten mit dem Wert TRUE belegt werden. Dann werden die Status auf GOOD gesetzt, sobald das Kommunikationsmodul gültige Prozesswerte vom Prozessmodul erhalten hat.

Das folgende Bild zeigt den Austausch der Systemvariablen zwischen dem HIMA Controller und jeweils einem DO Device und einem DI Device.

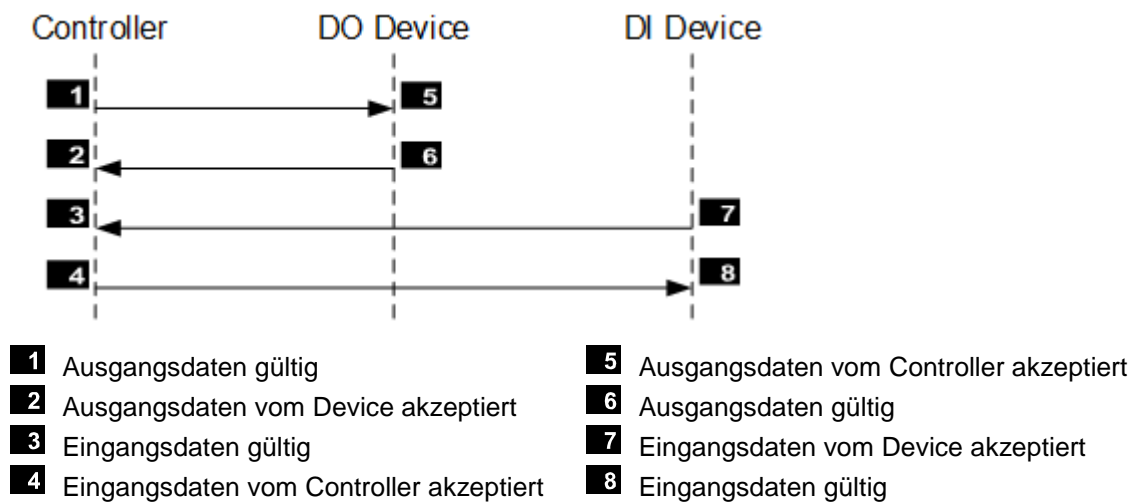


Bild 1: Steuerung des Consumer/Provider Status (IOxS)

3.2.1 Steuervariablen im HIMA Controller

Mit den Ausgangsvariablen *Ausgangsdaten gültig* **1** und *Eingangsdaten vom Controller akzeptiert* **4** kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) gesteuert werden.

Mit den Eingangsvariablen *Ausgangsdaten vom Device akzeptiert* **2** und *Eingangsdaten gültig* **3** kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) ausgelesen werden.

3.2.2 Steuervariablen im HIMA DO Device

Mit der Ausgangsvariable *Ausgangsdaten gültig* **6** kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) gesteuert werden.

Mit der Eingangsvariable *Ausgangsdaten vom Controller akzeptiert* **5** kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) ausgelesen werden.

3.2.3 Steuervariablen im HIMA DI Device

Mit der Ausgangsvariable *Eingangsdaten vom Device akzeptiert* **7** kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) gesteuert werden.

Mit der Eingangsvariable *Eingangsdaten gültig* **8** kann über das Anwenderprogramm der Consumer/Provider Status (IOxS) ausgelesen werden.

3.3 PROFIsafe

Die PROFIsafe-Spezifikation der PNO wird als bekannt vorausgesetzt!

PROFIsafe benutzt das PROFINET Protokoll zur Übertragung von sicherheitsbezogenen Daten bis SIL 3 auf Basis der Ethernet-Technologie.

Das PROFIsafe-Protokoll ist dem PROFINET Protokoll überlagert und enthält die sicheren Nutzdaten, sowie die Informationen zur Datensicherung. Die sicheren PROFIsafe-Daten werden gemeinsam mit nicht sicherheitsrelevanten PROFINET-Daten über das unterlagerte PROFINET Protokoll übertragen.

PROFIsafe verwendet „unsichere Datenübertragungskanäle“ (Ethernet) ähnlich dem Black-Channel-Prinzip für die Übertragung von sicheren Daten. Auf diesem Weg tauschen der F-Host und das F-Device die sicheren PROFIsafe-Daten aus.

3.3.1 PROFIsafe bei HIMA Steuerungen

Nach der PROFIsafe-Spezifikation versendet der F-Host solange wiederholte Nachrichtenpakete, bis das F-Device eine Empfangsbestätigung an den F-Host zurückgibt. Erst dann wird vom F-Host ein neues Nachrichtenpaket an das F-Device gesendet.

Über jedes wiederholte PROFIsafe-Nachrichtenpaket wird der momentane Wert des Prozesswerts übertragen. So kann es vorkommen, dass das gleiche Prozesssignal in den wiederholten Nachrichtenpaketen unterschiedliche Werte hat.

PROFIsafe ist in HIMA Geräten empfangsseitig so ausgelegt, dass Prozesswerte nur einmalig beim ersten Empfang eines Nachrichtenpakets übernommen werden.

Die Prozesswerte der wiederholten Nachrichtenpakete (mit gleicher fortlaufender Nummer des Nachrichtenpakets) werden verworfen.

Bei einem Verbindungsverlust nehmen nach Ablauf der F_WD_Time , die PROFIsafe-Prozesswert-Variablen den Initialwert ein.

Damit ein bestimmter Prozesswert auf der Gegenseite (F-Host/F-Device) Empfangen wird, muss der Prozesswert für mindestens die folgende Zeit unverändert anstehen:

$$2 * F_WD_Time + F_WD_Time2$$

Das PROFIsafe-System muss vom Anwender derart parametrierung werden, dass die $SFRT$ (Safety Function Response Time) für die jeweiligen Sicherheitsfunktionen geeignet ist.

Berechnungsformeln, siehe Kapitel 3.3.4.

i

Für PROFIsafe konformes Verhalten müssen folgende Bedingungen erfüllt sein:

- Die Initialwerte der Prozesswert-Variablen müssen mit "0" konfiguriert sein
- Der Parameter *AutoAwaitFParamsOnConnLoss* muss deaktiviert sein, siehe Kapitel 3.10.

Die Standardeinstellungen in SILworX entsprechen dem PROFIsafe konformen Verhalten!

3.3.2 PROFIsafe-Control-Byte und Status-Byte

Die beiden Systemvariablen Control-Byte und Status-Byte sind in jedem PROFIsafe-Submodul enthalten und werden bei der Kommunikation zwischen F-Host und F-Device ausgetauscht, siehe Kapitel 3.7.6.3 und Kapitel 3.10.4.

Das PROFIsafe-Control-Byte wird im F-Host beschrieben und im F-Device gelesen.
Das PROFIsafe-Status-Byte wird im F-Device beschrieben und im F-Host gelesen.

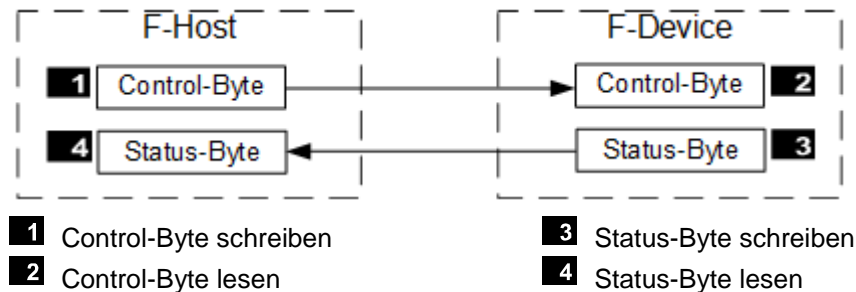


Bild 2: PROFIsafe-Control-Byte und Status-Byte

i

Die Systemvariablen Control-Byte und Status-Byte haben bei HIMA zusätzliche Funktionen, die von der PROFIsafe-Spezifikation abweichen, siehe Tabelle 12 und Tabelle 29.

3.3.3 F_WD_Time (PROFIsafe-Watchdog-Zeit)

Für eine funktionierende PROFIsafe-Verbindung zwischen einem HIMA F-Host und einem F-Device gilt die folgende Ungleichung:

$$F_WD_Time > 3 * CPU\text{-}Zykluszeit * Anzahl\text{-}Kommunikationszeitscheiben + 2 * PROFINet\text{-}Controller\text{-}Produktionsabstand^{1)} + 1 * DAT (F\text{-}Device Acknowledgement Time) + 2 * interne\ Buszeit\ des\ F\text{-}Devices + 2 * PROFINet\text{-}Device\text{-}Produktionsabstand^{1)} + 2 * Ethernet\text{-}Delay$$

¹⁾PROFINet-Controller- und PROFINet-Device-Produktionsabstand sind im allgemeinen gleich und werden folgendermaßen berechnet: $Reduktionsfaktor * SendClockFaktor * 31,25 \mu s$

i

DAT (F-Device Acknowledgement Time) und die *interne Buszeit des Devices* sind aus der Gerätebeschreibung des F-Device-Herstellers zu entnehmen!

Bei HIMax und HIMatrix F-Devices ist $DAT = DAT_{out} = DAT_{in} = 2 * WDZ\text{-}CPU$

3.3.3.1 Anmerkungen zu F_WD_Time (PROFIsafe-Watchdog-Zeit)

1. DAT (F-Device Acknowledgement Time) ist die Zeit, die ein F-Device benötigt, um auf eine empfangene PROFIsafe-Nachricht mit einer Antwort zu reagieren. Mit F-Device ist die sichere Einheit (bei HIMA das CPU-Modul) gemeint, die den F-Device-Stack ausführt. Insbesondere bei modularen Systemen/Geräten sind hier die Zeiten der nicht sicherheitsbezogenen Funktionen/Komponenten nicht enthalten. Diese Definition von DAT weicht von der PROFIsafe-Spezifikation V2.5c Kapitel 9.3.3. in den folgenden Punkten ab:
 - DAT enthält nicht die Zeiten für den internen Bus des F-Devices.
 - DAT enthält nicht den Anteil des PROFINet-Device-Produktionsabstands.
 - DAT enthält nicht die Verzögerungen, wie z. B. Filter der Eingangs-/Ausgangswerte, Verzögerungen der Ausgangs-/Eingangsphysik, etc.
 - DAT steht für DATin (Eingang) oder DATout (Ausgang), je nach Verbindung.
 - für alle Zeiten sind die jeweiligen Maxima zu verwenden.
2. Interne Buszeit des F-Devices ist bei HIMatrix und bei HIMax die *(max. Anzahl Kommunikationszeitscheiben - 1) * WDZ-CPU*.
3. Voraussetzung: das F-Device arbeitet zyklisch und dessen DAT ist $DAT = 2 * \text{max. Zyklus}$
 - F-Device arbeitet **nicht mit** *Kommunikationszeitscheiben*.
Wenn
 $HIMA\ CPU\text{-Zykluszeit} * \text{Anzahl-Kommunikationszeitscheiben} < F\text{-Device-Zykluszeit}$ ist,
dann muss für *HIMA CPU-Zykluszeit* der *F_WD_Time* Berechnung noch
 $\Delta = F\text{-Device-Zykluszeit} - HIMA\ CPU\text{-Zykluszeit} * \text{Anzahl-Kommunikationszeitscheiben}$
addiert werden.
 - F-Device arbeitet **mit** *Kommunikationszeitscheiben*.
Wenn
 $(HIMA\ CPU\text{-Zykluszeit} * \text{Anzahl-Kommunikationszeitscheiben}) < F\text{-Device-Zykluszeit} * \text{Anzahl-F-Device-Kommunikationszeitscheiben}$
ist, dann muss für *HIMA CPU-Zykluszeit* der *F_WD_Time* Berechnung noch
 $\Delta = (F\text{-Device-Zykluszeit} * \text{Anzahl-Device-Kommunikationszeitscheiben}) - (HIMA\ CPU\text{-Zykluszeit} * \text{Anzahl-Kommunikationszeitscheiben})$
addiert werden.

3.3.4 SFRT (Safety Function Response Time)

3.3.4.1 Berechnung der SFRT zwischen einem F-Device und einem HIMA F-Host

Die zulässige SFRT für eine PROFIsafe-Verbindung zwischen einem F-Device und einem HIMA F-Host mit lokaler Ausgabe wird wie folgt berechnet:

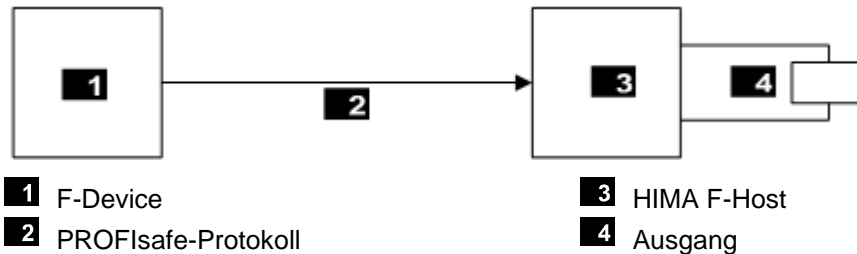


Bild 3: Reaktionszeit zwischen einem F-Device und einem HIMA F-Host

$$\text{SFRT} \leq \text{MaxDataAgeIn} + 2 * \text{F_WD_TIME} + \text{MaxDataAgeOut} + T_u$$

Anmerkungen:

Bei HIMax/HIMatrix und lokaler Verwendung der Daten, d. h. wenn sie nicht auf einer HIMax IO ausgegeben werden, kann die Fehlertoleranzzeit des CPU-Moduls durch $2 * \text{WDZ-CPU}$ ersetzt werden. Siehe auch *Anmerkung zu SFRT Berechnungen* im Kapitel 3.3.4.3.

3.3.4.2 Berechnung der SFRT mit F-Devices und einem HIMA F-Host

Die zulässige SFRT für eine PROFIsafe-Verbindung zwischen einem HIMA F-Host und einem F-Device mit lokaler Ausgabe wird wie folgt berechnet:

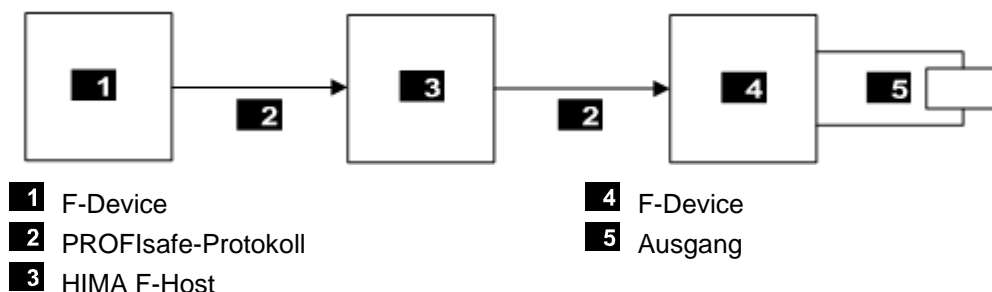


Bild 4: Reaktionszeit mit einem HIMA F-Host und zwei F-Devices

$$\text{SFRT} \leq \text{MaxDataAgeIn} + 2 * \text{F_WD_TIME}(\text{input}) + 2 * \text{WDZ-CPU} + 2 * \text{F_WD_TIME}(\text{output}) + \text{MaxDataAgeOut} + T_u$$

3.3.4.3 Anmerkung zu SFRT (Safety Function Response Time) Berechnungen

1. Definition **SFRT** gemäß IEC61784-3 Ed.2.
2. Alle etwaige zusätzliche Verzögerungen im Anwenderprogramm (z. B. durch Funktionsbausteine TOF, TON, etc.) oder Module/Baugruppen (z. B. durch Ausgangsfilter, Eingangsfilter, Relais, etc.) müssen addiert werden.
3. **MaxDataAgeIn** ist das maximale Alter eines an einem physikalischen Eingang eingelesenen Prozesswertes, den ein F-Device in eine PROFIsafe-Nachricht einfügt. Jedoch nur der Anteil, der nicht schon in DATin enthalten ist.

i

Bei HIMatrix ist *MaxDataAgeIn* mit 0 ms anzusetzen.

Bei einem HIMax System ist *MaxDataAgeIn* wie folgt anzusetzen:

- Für physikalische Eingänge bis zu $FTZ-CPU^{1)} - 2 * WDZ-CPU - DATin$ groß.
 - Für vom Anwenderprogramm gebildete Daten mit 0 ms.
-

4. **MaxDataAgeOut**

Ist die Worst-Case-Reaktionszeit eines F-Output-Devices oder F-Hosts für

- a) Ausgabe empfangener Prozesswerte an einem physikalischen Ausgang,
- b) Ansteuern der physikalischen Ausgänge nach Ablauf von *F_WD_Time* und
- c) Deaktivieren der physikalischen Ausgänge bei Geräteausfall.

- Bei HIMatrix ist $MaxDataAgeOut = 3 * WDZ-CPU$ (für phys. Ausgänge)
- Bei HIMax ist $MaxDataAgeOut = WDZ-CPU + FTZ-CPU^{1)}$ (für phys. Ausgänge)
- Bei Ablauf der *F_WD_TIME* reagieren HIMax und HIMatrix ohne Fehler spätestens nach $2 * WDZ-CPU$.
 - Wenn der F-Host/F-Device (bei HIMA CPU-Modul) unmittelbar vor dieser Reaktion ausfällt, so werden bei HIMatrix nach *WDZ-CPU* die Ausgänge energielos. Bei HIMax geschieht dies spätestens nach $FTZ-CPU^{1)} - WDZ-CPU$, da das Ausgangs-Modul schon eine *WDZ-CPU* lang keine Nachricht bekommen hat.
 - Nimmt man nur **einen** Fehler an, so kann $1 * WDZ-CPU$ von *MaxDataAgeOut* für HIMatrix und HIMax subtrahiert werden.

5. **Tu** ist das Minimum von *DATin*, *DATout*, *WDZ-CPU*. Theoretisch kann man für *DATin* und *DATout* jeweils die Hälfte ansetzen, dazu muss der Hersteller jedoch spezifiziert haben, mit welcher Ungenauigkeit das Device die *F_WD_Time* überwacht. Falls das Device zyklisch arbeitet, dann ist die

$DAT = 2 * \max. \text{Zyklus des Devices}$.

Bei HIMA HIMatrix und HIMax F-Devices ist $DATout = DATin = 2 * WDZ-CPU$.

6. **F_WD_TIME**, siehe Kapitel 3.3.3.

¹⁾ Prozess-Sicherheitszeit des CPU-Moduls

3.4 Auflagen für den sicheren Betrieb von PROFIsafe

3.4.1 Adressierung

Das PROFIsafe-Netzwerk entspricht bei HIMA dem PROFINET-Ethernet-Netzwerk, über welches die PROFIsafe-Nachrichten übertragen werden können. Als Netzwerk ist hier ein logisches Netzwerk gemeint, das sich auch über mehrere physikalische Teilnetze erstrecken kann.

Eine geeignete Netzwerktrennung für PROFIsafe liegt vor, wenn es nicht möglich ist, dass PROFIsafe-Nachrichten die Netzwerktrennung überwinden können.

Dies ist zum Beispiel dann der Fall, wenn ein Router auf IP-Basis verwendet wird und die Netzwerke an unterschiedlichen Ethernet-Schnittstellen des Routers angeschlossen sind.

Eine Netzwerktrennung für PROFIsafe ist nicht gegeben, wenn zum Beispiel die Netzwerke über Ein-Port-Router, Switches, Hubs oder Ethernet-Bridges verbunden werden.

Auch wenn managebare Switches verwendet werden und die PROFIsafe-Netzwerke zum Beispiel durch Port-Based-VLANs getrennt sind, sollte dennoch eine Eineindeutigkeit der Adressierung angestrebt werden. Dies verhindert, dass zum Beispiel bei Wartungsarbeiten oder Erweiterungen, versehentlich Verbindungen zwischen PROFIsafe-Netzwerken hergestellt werden.

Folgende Punkte zur Adressierung von PROFIsafe-Geräten sind einzuhalten:

- Es muss sichergestellt werden, dass die F-Adressen der PROFIsafe-Geräte/Module in einem PROFIsafe-Netzwerk eineindeutig sind.
- Zur Adressierungssicherheit wird weiterhin empfohlen, die F-Adressen auch bei getrennten PROFIsafe-Netzwerken eineindeutig zu wählen.
- Bei der Inbetriebnahme und bei Änderungen der Sicherheitsfunktionen muss überprüft werden, dass die Sicherheitsfunktion über PROFIsafe hinweg die richtigen Ein- und Ausgänge, der richtigen PROFIsafe-Geräte nutzt.
- Die PROFIsafe-F-Module sind so zu konfigurieren (z. B. durch Vergabe einer geeigneten F-Adresse oder F_WD_Time), dass in einem PROFIsafe-Netzwerk F-Module gleicher Input- und gleicher Output-Datenlänge unterschiedliche CRC1-Signaturen haben. Der Anwender kann den CRC1 in SILworX ablesen.

Anmerkung:

Dies ist dann auf jeden Fall für die betroffenen F-Module gewährleistet, wenn in einem PROFIsafe-Netzwerk nur ein F-Host genutzt wird und die F-Parameter der F-Module in einem PROFIsafe-Netzwerk sich nur in der F-Adresse unterscheiden.

Damit nicht versehentlich eine gleiche CRC1-Signatur erzeugt wird, muss darauf geachtet werden, dass zum Beispiel die Parameter *F_WD_Time*, *F_Prm_Flag1/2* für alle F-Module gleich sind und die F-Module keinen i-Par-CRC verwenden.

Risiko von PROFIsafe-Geräten mit gleicher Input- und Output-Datenlänge

PROFIsafe-Geräte dürfen nur betrieben werden, wenn die F-INPUT-Datenlänge ungleich der F-OUTPUT-Datenlänge derselben PROFIsafe-Verbindung ist.

Andernfalls könnten Adressierungsfehler der Standardkomponenten und/oder der Standardübertragungstechnik von PROFIsafe unentdeckt bleiben und zu sicherheitsbezogenem Fehlverhalten führen.

Bei HIMA F-Modulen, die als Teil der HIMA Steuerung konfiguriert werden, müssen die Längen von F-Input und F-Output unterschiedlich gewählt werden. Um das Risiko einer sicherheitsbezogenen Fehlfunktion vorzubeugen, nur F-Input oder F-Output Module verwenden. Keine F-Input/Output Module verwenden!

3.4.2 Netzwerkaspekte

Das für die Übertragung von PROFI-safe-Nachrichten verwendete Netzwerk muss eine ausreichende Verfügbarkeit und Übertragungsqualität aufweisen.

i

Erkennt PROFI-safe eine verminderte Übertragungsqualität, die von den Standardübertragungseinrichtungen (Ethernet) nicht erkannt wird, so erfolgt die Sicherheitsreaktion.

Nach einer Sicherheitsreaktion auf Grund verminderter Übertragungsqualität, müssen die Probleme beseitigt werden, um wieder eine ausreichende Übertragungsqualität zu gewährleisten. Erst nach den dazu erforderlichen Maßnahmen darf die Quittierung für den Wiederanlauf von PROFI-safe ausgelöst werden. Diese Quittierung erfolgt durch das Signal Operator-Acknowledge oder Reset.

WARNUNG



Operator-Acknowledge und Reset dürfen nur genutzt werden, wenn keine gefährlichen Zustände mehr existieren.

Ein PROFI-safe-Netzwerk muss vor unzulässigen Eingriffen (DoS, Hacker, ...) geschützt werden. Die Maßnahmen sind mit der überwachenden Behörde abzustimmen. Dies hat besondere Relevanz, wenn Wireless-Techniken zur Übertragung eingesetzt werden. Für weitere Informationen, siehe PROFI-safe Specification V2.5c Tabelle 23 und Tabelle 24.

Verfügbarkeit gegenüber eingefügten Nachrichten

Nachrichtenpakete können z. B. durch Netzwerkkomponenten wie Switches gespeichert und zu einem späteren Zeitpunkt eingefügt (gesendet) werden¹⁾. Diese Nachrichtenpakete führen zu einer Abschaltung, wenn diese älter (siehe Consecutive Number Tabelle 12 und Tabelle 29) als das zuletzt vom PROFI-safe-Gerät empfangene Nachrichtenpaket ist.

¹⁾Fehlerannahme für Sicherheitsbezogene Kommunikation

3.5 PROFINET IO-Controller und PROFIsafe-F-Host

Dieses Kapitel beschreibt die Eigenschaften des PROFINET IO-Controller und PROFIsafe-F-Host sowie die Menüfunktionen und Dialoge in SILworX, die zur Konfiguration des PROFINET IO-Controllers und PROFIsafe-F-Host benötigt werden.

3.5.1 Systemanforderung PROFINET IO-Controller

Benötigte Ausstattung und Systemanforderung:

Element	Beschreibung
Steuerung	HIMax mit X-COM 01 Modul HIMatrix
CPU-Modul	Die Ethernet-Schnittstellen des Prozessormoduls können für PROFINET IO nicht verwendet werden.
COM-Modul	Ethernet 10/100BaseT.
Aktivierung	Die Freischaltung erfolgt per Software-Freischaltcode, siehe Kommunikationshandbuch HI 801 100 D.

Tabelle 3: Systemanforderung und Ausstattung für PROFINET IO-Controller

3.5.2 PROFINET IO-Controller und PROFIsafe-Host Eigenschaften

Eigenschaften	Beschreibung
Sicherheitsbezogen	PROFINET Nein PROFIsafe Ja
Übertragungsrate	100 Mbit/s voll duplex
Transportweg	Ethernet-Schnittstellen der COM-Module Verwendete Ethernet-Schnittstellen simultan auch für andere Protokolle nutzbar.
Konformitätsklasse	Der PROFINET IO-Controller entspricht den Anforderungen der Conformance Class A.
Real Time Class	RT-Klasse 1
Max. Anzahl PROFINET IO-Controller	Für jedes COM-Modul kann ein PROFINET IO-Controller konfiguriert werden.
Max. Anzahl PROFINET IO-Devices Applikationsbeziehungen (ARs)	Ein PROFINET IO-Controller kann mit max. 64 PROFINET IO-Devices eine Applikationsbeziehung (AR) aufbauen.
Anzahl (CRs pro AR) Kommunikationsbeziehungen	Standard: 1 Input CR, 1 Output CR, 1 Alarm CR
Max. Prozessdatenlänge einer Kommunikationsbeziehung (CR)	Output: max. 1440 Bytes Input: max. 1440 Bytes
Sendetakt	Möglich über die Einstellung der <i>Reduction Rate</i> auf der Device Ebene.
Die folgenden Eigenschaften gelten für PROFIsafe	
Max. Anzahl F-Hosts (HIMax)	1024
Max. Anzahl F-Hosts (HIMatrix)	512
Max. Prozessdatenlänge einer Kommunikationsbeziehung (CR)	Output: max. 123 Bytes Nutzdaten + 5 Bytes ¹⁾ Input: max. 123 Bytes Nutzdaten + 5 Bytes ¹⁾
Max. Größe Nutzdaten HIMax HIMatrix	1024 x 123 Bytes = 125 952 Bytes 512 x 123 Bytes = 62 976 Bytes
¹⁾ 5 Bytes Verwaltungsdaten (Status/Control Bytes und CRC)	

Tabelle 4: Eigenschaften PROFINET IO-Controller

3.6 **PROFINET IO/ PROFIsafe Beispiel**

In diesem Beispiel wird beschrieben, wie auf einer HIMax Steuerung ein PROFINET IO-Controller konfiguriert wird, der eine Verbindung zu einem PROFINET IO-Device hat.

Das PROFINET IO-Device ist mit einem PROFINET IO-Modul und einem PROFIsafe-Modul bestückt. Im PROFINET IO-Controller muss das PROFINET IO-Device genau so konfiguriert werden, wie es tatsächlich aufgebaut ist.

3.6.1 Anlegen des PROFINET IO-Controller in SILworX

Anlegen eines neuen PROFINET IO-Controllers

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle** wählen.
2. Im Kontextmenü von Protokolle **Neu, PROFINET IO-Controller** wählen, um einen neuen PROFINET IO-Controller hinzuzufügen.
3. Im Kontextmenü vom PROFINET IO-Controller **Eigenschaften** wählen.
4. Im Feld **Name** den Gerätenamen des Controllers eintragen.
5. **COM-Modul** wählen.

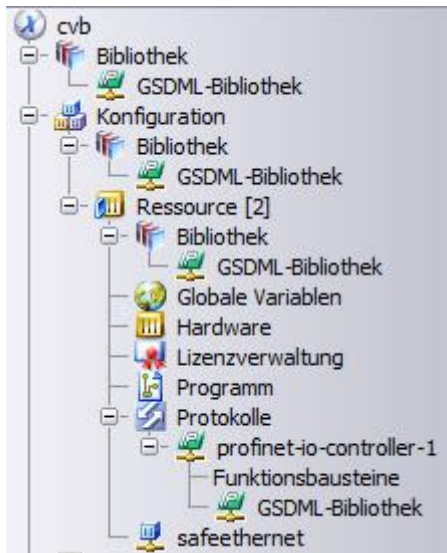


Bild 5: PROFINET IO-Controller im HIMax Strukturbaum

3.6.2 Anlegen der GSDML-Bibliothek in SILworX

Eine GSDML-Bibliothek dient der Verwaltung von GSDML-Dateien, welche innerhalb eines PROFINET IO-Controllers weiterverwendet werden können.

Eine GSDML-Bibliothek kann auf Projektebene, Konfigurationsebene, Ressourcenebene oder direkt unterhalb des PROFINET IO-Controllers angelegt werden, siehe Bild 5.

Wird die GSDML-Bibliothek auf Projektebene, Konfigurationsebene oder Ressourcenebene angelegt, kann diese von allen PROFINET IO-Controllern die darunter liegen verwendet werden.

Anlegen der GSDML-Bibliothek in SILworX auf Konfigurationsebene

1. Ist im Projekt noch keine Bibliothek angelegt, dann im Strukturbaum Rechtsklick auf **Projekt** und **Neu, Bibliothek** wählen.
2. Im Strukturbaum Rechtsklick auf **Projekt, Bibliothek** und **Neu, GSDML-Bibliothek** wählen.
☒ Eine neue GSDML-Bibliothek wird angelegt.

Einlesen der GSDML-Datei aus einer externen Datenquelle (z. B. DVD, USB-Stick)

1. Im Strukturbaum **Projekt, Bibliothek, GSDML-Bibliothek** wählen.
2. Rechtsklick auf **GSDML-Bibliothek** und im Kontextmenü **Neu** wählen, um den Dialog GSDML-Datei hinzufügen zu öffnen.
3. Schaltfläche ... betätigen und die benötigten GSDML-Dateien aus dem Speicherort wählen.
4. Mit **OK** bestätigen.

3.6.3 Konfiguration des Device im PROFINET IO-Controller

Anlegen eines HiMax PROFINET IO-Devices im PROFINET IO-Controller

1. Im Kontextmenü des PROFINET IO-Controller **Neu**, **PROFINET IO-Device** wählen.

Laden der GSDML-Datei für ein neues PROFINET IO-Device

1. Im Strukturbaum **Konfiguration**, **Ressource**, **Protokolle**, **PROFINET IO-Controller**, **PROFINET IO-Device** wählen.
2. Im Kontextmenü **Eigenschaften** wählen und das Register **Parameter** öffnen.
3. Im Feld **Name** den Gerätenamen des PROFINET IO-Devices eintragen.
4. IP-Adresse des PROFINET IO-Device in das Feld **IP-Adresse** eintragen.
5. Im Dropdown-Menü **GSDML-Datei**, die zum PROFINET IO-Device zugehörige GSDML-Bibliotheksdatei wählen und **Eigenschaften** schließen.

Auswahl des Device-Access-Point (DAP) für das PROFINET IO-Device

1. Im Strukturbaum **Protokolle**, **PROFINET IO-Controller**, **PROFINET IO-Device**, **DAP-Modul** wählen.
2. Im Kontextmenü **Edit** wählen und den passenden *Device-Access-Point (DAP)*-Datensatz zum PROFINET IO-Device wählen.

i

Die GSDML-Datei beinhaltet oft mehrere *Device-Access-Points (DAP)* eines Herstellers.

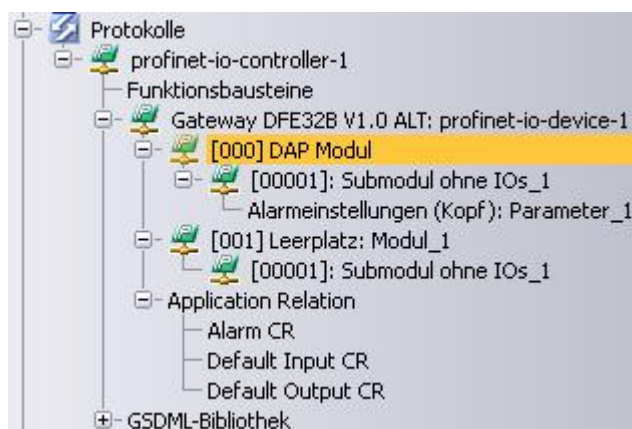


Bild 6: Device-Access-Point (DAP) für das PROFINET IO-Device

Konfigurieren der Modul-Steckplätze

1. Im Strukturbaum **Protokolle**, **PROFINET IO-Device** öffnen.
2. Im Kontextmenü von PROFINET IO-Device **Neu** wählen, um die Modul-Liste zu öffnen.
3. Für das PROFINET IO-Device die passenden Module aus der Modul-Liste wählen und mit **Modul(e) einfügen** bestätigen.

Nummerieren der PROFINET IO-Device-Module

Das **Device-Access-Point (DAP) Modul** hat den festen Steckplatz 0. Alle weiteren PROFINET IO-Device-Module müssen nummeriert werden.

1. Im Kontextmenü von PROFINET IO-Device-Modul **Eigenschaften** wählen.

2. Im Feld **Steckplatz** die Steckplätze der Device-Module so eintragen, wie das PROFINET IO-Device tatsächlich aufgebaut ist.
 3. Diese Schritte für weitere **PROFINET IO-Device-Module** wiederholen.
- Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei.

Konfigurieren der Application-Relation

1. Im Strukturbau **PROFINET IO-Device, Application-Relation** öffnen.
2. Rechtsklicken auf **Default-Input-CR** und im Kontextmenü **Eigenschaften** wählen.
3. Parameter Reduktionsfaktor anpassen, z. B. auf 4 setzen.
4. Rechtsklicken auf **Default-Output-CR** und im Kontextmenü **Eigenschaften** wählen.
5. Parameter Reduktionsfaktor anpassen, z. B. auf 4 setzen.

3.6.4 Konfiguration des Device Access Point (DAP) Moduls

Konfigurieren des Device Access Points (DAP) Moduls

1. **[000] DAP Modul, [xxxxx] DAP Submodul** wählen.
2. Rechtsklicken auf **[xxxxx] DAP Submodul** und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Systemvariablen** wählen, und die Ausgangsvariable *Eingangsdaten vom Controller akzeptiert* mit einer globalen Variablen mit dem Initialwert TRUE belegen, wenn eine Steuerung des Consumer/Provider Status mit spezieller Anwenderprogramm-Logik nicht gewünscht ist.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

Einstellung der Kopf-Parameter z. B. für einen DAP mit Alarmeinstellungen.

Einstellen der Kopf-Parameter des Device Access Points (DAP) für das PROFINET IO-Device

1. Im Strukturbau **Protokolle, PROFINET IO-Controller, PROFINET IO-Device, DAP Modul, [xxxxx] DAP Submodul, Alarmeinstellungen (Kopf): Parameter** wählen.
2. Im Kontextmenü **Eigenschaften** wählen.
3. Im Feld **Name** den Parameternamen der Kopf-Parameter eintragen.
4. Über die Schaltfläche **Edit** wird ein Dialog geöffnet, in dem die Einstellungen zu Schnittstellen und Diagnose/Alarmer bearbeitet werden können.

3.6.5 Konfiguration der PROFINET IO-Device Module



Die Summe der Variablen in Byte muss mit der Größe des jeweiligen Moduls in Byte übereinstimmen.

Konfigurieren des PROFINET IO-Device Moduls

1. **[001] PROFINET IO-Device Modul, [xxxxx] PROFINET IO-Device Submodul** wählen.
2. Rechtsklicken auf **[xxxxx] Submodul** und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Eingangssignale** ziehen.

5. Rechtsklick auf eine leere Stelle im Bereich **Eingangssignale** und im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.
6. Im Dialog **Edit** das Register **Systemvariablen** wählen, und die Ausgangsvariablen *Eingangsdaten vom Controller akzeptiert* und *Ausgangsdaten gültig* mit einer globalen Variablen mit dem Initialwert TRUE belegen, wenn eine Steuerung des Consumer/Provider Status nicht gewünscht ist.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

3.6.6 Konfiguration der PROFIsafe-IO-Device-Module



Die Summe der Variablen in Byte muss mit der Größe des jeweiligen Moduls in Byte übereinstimmen.

Konfigurieren der PROFIsafe-IO-Device-Module

1. **[001] PROFIsafe-IO-Device-Modul, [xxxxx] PROFIsafe-IO-Device-Submodul** wählen.
2. Rechtsklicken auf **[xxxxx] Submodul** und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Eingangssignale** ziehen.
5. Rechtsklick auf eine leere Stelle im Bereich **Eingangssignale** und im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.
6. Im Dialog **Edit** das Register **Systemvariablen** wählen, und die Ausgangsvariable *Eingangsdaten vom Controller akzeptiert* und *Ausgangsdaten gültig* mit einer globalen Variablen mit dem Initialwert TRUE belegen, wenn eine Steuerung des Consumer/Provider Status nicht gewünscht ist.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

Konfigurieren der F-Parameter

1. **[001] PROFIsafe-IO-Device-Modul, [xxxxx] PROFIsafe-IO-Device-Submodul, F-Parameter** wählen.
2. Rechtsklicken auf **F-Parameter** und im Kontextmenü **Eigenschaften** wählen.
3. Folgende Parameter einstellen:
 - F_Dest_Add: Zieladresse des Device Module einstellen
 - F_WD_Time: Watchdogzeit der Verbindung zu diesem Device Module

PROFINET IO-Konfiguration verifizieren

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFINET IO-Controller** öffnen.
2. Rechtsklick auf **PROFINET IO-Controller** und im Kontextmenü **Verifikation** wählen.
3. Einträge im Logbuch sorgfältig überprüfen, gegebenenfalls Fehler korrigieren.



Die Ressource neu compilieren und in die Steuerungen laden, damit die Konfiguration für die PROFINET IO-Kommunikation wirksam wird.

3.6.7 Ermitteln der PROFINET IO-Devices im Netzwerk

Auffinden des PROFINET IO-Devices im Ethernet Netzwerk

1. Modul-Login auf das Kommunikationsmodul mit dem **PROFINET IO-Controller** ausführen.
2. Im Strukturbaum der Online-Ansicht
PROFINET IO-Controller, PROFINET IO-Netzwerkteilnehmer wählen.
3. Im Kontextmenü **PROFINET-Netzwerkteilnehmer ermitteln** wählen.
 - ☒ Alle PROFINET-Devices im Netzwerk dieses PROFINET IO-Controllers werden aufgelistet.

Konfiguration des PROFINET IO-Devices in der Online-Ansicht

1. In der Geräteliste, Rechtsklick auf das zu konfigurierende PROFINET IO-Device, um die Einstellungen zu ändern.
2. Mit der Kontextmenüfunktion **Setze PROFINET IO-Gerätenamen**, den Gerätenamen setzen.
 - ☒ Sicherstellen, dass der PROFINET IO-Geräteiname mit der Projektierung übereinstimmt. (Es sind nur Kleinbuchstaben erlaubt!)
3. Mit der Kontextmenüfunktion **Netzwerkeinstellungen** die IP-Adresse, Subnet Mask und den Gateway setzen.



Der PROFINET IO-Gerätenamen und die Netzwerkeinstellungen des PROFINET IO-Device müssen im PROFINET IO-Controller eingetragen sein, sonst ist eine Kommunikation nicht möglich.

3.7 Menüfunktionen des PROFINET IO-Controllers

3.7.1 Beispiel für Strukturbaum des PROFINET IO-Controllers

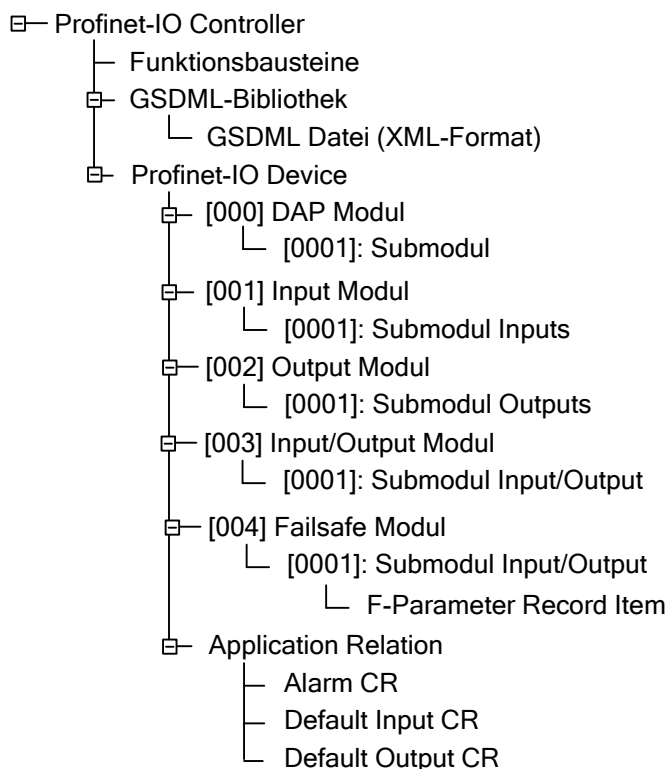


Bild 7: Strukturbaum des PROFINET IO-Controllers

3.7.2 PROFINET IO-Controller

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des PROFINET IO-Controller öffnet den Dialog **Eigenschaften**. Das Dialogfenster enthält die folgenden Register:

3.7.2.1 Register PROFINET IO (Eigenschaften)

Element	Beschreibung
Typ	PROFINET IO-Controller
Name	Gerätename des PROFINET IO-Controller
Aktualisierungsintervall der Prozessdaten [ms]	Aktualisierungszeit in Millisekunden, mit der die Daten des Protokolls zwischen COM und CPU ausgetauscht werden. Ist die <i>Refresh Rate</i> Null oder kleiner als die Zykluszeit der Steuerung, dann erfolgt der Datenaustausch so schnell wie möglich. Wertebereich: 4 ... (2 ³¹ -1) Standardwert: 0
Prozessdaten-Konsistenz erzwingen	Aktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM innerhalb eines Zyklus der CPU. Deaktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM, verteilt über mehrere CPU Zyklen zu je 1100 Byte pro Datenrichtung. Damit kann eventuell auch die Zykluszeit der Steuerung reduziert werden. Standardwert: Aktiviert
Modul	Auswahl des COM-Moduls auf dem dieses Protokoll abgearbeitet wird.

Element	Beschreibung
Max. μ P-Budget aktivieren	Aktiviert: Limit des μ P-Budget aus dem Feld <i>Max. μP-Budget in [%]</i> übernehmen. Deaktiviert: Kein Limit des μ P-Budget, für dieses Protokoll verwenden.
Max. μ P-Budget in [%]	Maximales μ P-Budget des Moduls, welches bei der Abarbeitung des Protokolls produziert werden darf. Wertebereich: 1 ... 100% Standardwert: 30%
RPC Port Server	Remote Procedure Call Port Wertebereich: 1024 ... 65535 Standardwert: 49152 RPC Port Server und RPC Port Client dürfen nicht identisch sein!
RPC Port Client	Remote Procedure Call Port Wertebereich: 1024 ... 65535 Standardwert: 49153 RPC Port Server und RPC Port Client dürfen nicht identisch sein!
F_Source_Add	Adresse des Controllers (F-Host). Der Anwender muss eindeutige PROFI-safe Controller/Device-Adressen in einem PROFI-safe-Netzwerk verwenden. <i>siehe auch IEC61784-3-3 V2.5c Kapitel 9.7</i>

Tabelle 5: Register PROFINET IO (Eigenschaften)

3.7.3 PROFINET IO-Device (im Controller)

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des PROFINET IO-Device öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Register Parameter (Eigenschaften)

Element	Beschreibung
Name	Gerätename des PROFINET IO-Device
IP-Adresse	IP-Adresse des Kommunikationspartners. Wertebereich: 0.0.0.0 ... 255.255.255.255 Standardwert: 192.168.0.99 Keine belegten IP-Adressen verwenden, siehe Kommunikationshandbuch 801 100 D Kapitel <i>Verwendete Netzwerkports für Ethernet-Kommunikation</i> .
Subnet Mask	Subnet Mask des adressierten Subnetzes in dem sich das Device befindet. Wertebereich: 0.0.0.0 ... 255.255.255.255 Standardwert: 255.255.255.0
GSDML-Datei	GSDML ist die Abkürzung für Generic Station Description Markup Language und ist eine XML-basierte Beschreibungssprache. Die GSDML-Datei enthält die Stammdaten des PROFINET Device

Tabelle 6: Register Parameter (Eigenschaften)

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Herstellername*, *Gerätebeschreibung* oder *unterstützte Faktoren*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.4 DAP Modul (Device Access Point Modul)

Innerhalb eines PROFINET Devices wird immer ein DAP Modul für die Busanschaltung angelegt. Dieses DAP Modul sollte nicht gelöscht werden.

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des DAP Moduls öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Register Parameter (Eigenschaften)

Element	Beschreibung
Name	Name für das DAP Modul
Steckplatz	Nicht änderbar. Standardwert: 0

Tabelle 7: Register Parameter (Eigenschaften)

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Modulkennung*, *Hardware-/Software Version*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.4.1 DAP Submodul (Eigenschaften)

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des DAP Submoduls öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Register Parameter

Element	Beschreibung
Name	Name des Input Submoduls
Substeckplatz	Standardwert: 1
IO Data CR, Eingänge	Auswahl der Kommunikationsbeziehung (Communication Relation CR) in der die Eingänge des Submoduls übertragen werden sollen. - Keine - Default Input CR
Eingangsdaten vom Controller akzeptiert	Auswahl der Kommunikationsbeziehung (Communication Relation CR) in welcher der IO Consumer Status (CS) des Submoduls übertragen werden soll. - Keine - Default Output CR

Tabelle 8: Register Parameter

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Submodul Kennung*, *Datenlängen*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.4.2 DAP Submodul (Edit)

Die Menüfunktion **Edit** aus dem Kontextmenü der Input Submodule öffnet den Dialog **Edit**, der die folgenden Register enthält:

Register Systemvariablen

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFINET IO-Submoduls im Anwenderprogramm auszuwerten bzw. zu steuern.

Element	Beschreibung
Eingangsdaten gültig	TRUE Eingangsdaten gültig GOOD
	FALSE Eingangsdaten ungültig BAD
Eingangsdaten vom Controller akzeptiert	TRUE Eingangsdaten gültig GOOD
	FALSE Eingangsdaten ungültig BAD

Tabelle 9: Register Systemvariablen



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

3.7.4.3 Kopf-Parameter

Manche Devices enthalten sogenannte Kopf-Parameter in denen z. B. Diagnose, Alarm und Schnittstellen aktiviert/deaktiviert werden können.

3.7.5 Input/Output PROFINET IO-Module

Ein Input/Output PROFINET IO-Modul kann mehrere Submodule haben. H1Max PROFINET IO-Controller haben in jedem Input/Output PROFINET IO-Modul ein Submodul.

In den PROFINET IO-Input Modulen werden die Eingangsvariablen des H1Max PROFINET IO-Controllers eingetragen, die vom PROFINET IO-Device gesendet werden.

In den PROFINET IO-Output Modulen werden die Ausgangsvariablen des H1Max PROFINET IO-Controllers eingetragen, die zum PROFINET IO-Device gesendet werden.

Anlegen des benötigten PROFINET IO-Moduls:

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFINET IO-Device** öffnen.
2. Im Kontextmenü von PROFINET IO-Device **Neu** wählen.
3. Passende Module wählen.

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü der Input/Output PROFINET IO-Module öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

3.7.5.1 Register Parameter

Element	Beschreibung
Name	Name des Input/Output PROFINET IO-Moduls
Steckplatz	0 ... 32767 Standardwert: 1

Tabelle 10: Register Parameter des I/O PROFINET IO-Moduls

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Modulkennung*, *Hardware-/Software Version*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.6 Submodul Input

Mit den Parametern der Submodule werden die Kommunikationsbeziehungen und das Verhalten der Module bei Verbindungsunterbrechung eingestellt.

3.7.6.1 Submodul Input (Eigenschaften)

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü der *Input Submodule* öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Register Parameter

Element	Beschreibung				
Name	Name des Input Submoduls.				
Substeckplatz	Nicht änderbar für HIMax PROFINET IO-Controller. Standardwert: 1				
IO Data CR, Eingänge	Auswahl der Kommunikationsbeziehung (Communication Relation CR) in der die Eingänge des Submoduls übertragen werden sollen. - Keine - Default Input CR				
Eingangsdaten vom Controller akzeptiert	Auswahl der Kommunikationsbeziehung (Communication Relation CR) in welcher der IO Consumer Status (CS) des Submoduls übertragen werden soll. - Keine - Default Output CR				
Shared Input	Nicht anwendbar				
Input-Werte bei IO-CR Unterbrechung	Verhalten der Input Variablen dieses PROFINET IO-Submoduls bei Verbindungsunterbrechung. <table border="1"> <tr> <td>Letzten Wert behalten</td><td>Die Input Variablen werden auf dem momentanen Wert eingefroren und bis zur erneuten Verbindungsaufnahme verwendet.</td></tr> <tr> <td>Initialwerte annehmen</td><td>Für die Input Variablen werden die Initialdaten verwendet.</td></tr> </table>	Letzten Wert behalten	Die Input Variablen werden auf dem momentanen Wert eingefroren und bis zur erneuten Verbindungsaufnahme verwendet.	Initialwerte annehmen	Für die Input Variablen werden die Initialdaten verwendet.
Letzten Wert behalten	Die Input Variablen werden auf dem momentanen Wert eingefroren und bis zur erneuten Verbindungsaufnahme verwendet.				
Initialwerte annehmen	Für die Input Variablen werden die Initialdaten verwendet.				

Tabelle 11: Register Parameter

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Submodulkennung*, *Hardware-/Software Version* oder *Datenlängen*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.6.2 Submodul Input (Edit)

Die Menüfunktion **Edit** aus dem Kontextmenü der Input Submodule öffnet den Dialog **Edit**. Das Dialogfenster enthält die folgenden Register:

Register Systemvariablen

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFINET IO-Submoduls im Anwenderprogramm auszuwerten.

Element	Beschreibung	
Eingangsdaten gültig	TRUE	Eingangsdaten gültig GOOD
	FALSE	Eingangsdaten ungültig BAD
Eingangsdaten vom Controller akzeptiert	TRUE	Eingangsdaten gültig GOOD
	FALSE	Eingangsdaten ungültig BAD
Die folgenden Parameter sind nur für PROFIsafe-Module verfügbar:		
Ausgangsdaten gültig	TRUE	Ausgangsdaten gültig GOOD
	FALSE	Ausgangsdaten ungültig BAD
Ausgangsdaten vom Device akzeptiert	TRUE	Ausgangsdaten gültig GOOD
	FALSE	Ausgangsdaten ungültig BAD



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

Element	Beschreibung
PROFIsafe Control	<p>PROFIsafe sendet vom Controller mit jeder Nachricht das PROFIsafe-Control-Byte an das Device, welches im Anwenderprogramm gesetzt werden kann. Siehe auch Kapitel 3.3.2.</p> <p>Bit 0 iPar_EN_C: Um neue iParameter in das F-Device zu laden muss iPar_EN_C auf TRUE gesetzt werden um das F-Device zu entriegeln. Solange iPar_EN_C auf TRUE gesetzt ist, werden sichere Werte "0" zwischen F-Host und F-Device ausgetauscht.</p> <p>Bit 1 OA_C: Operator Acknowledge. Bit muss nach einem PROFIsafe-Fehler (z. B. CRC-Fehler oder Timeout) für mindestens einen PROFIsafe-Zyklus auf TRUE gesetzt werden. Soll eine PROFIsafe-Verbindung (neu) gestartet werden, darf die Freigabe (Operator-Acknowledge) erst dann geben, wenn keine gefährlichen Zustände existieren.</p> <p>Bit 2 Reserved</p> <p>Bit 3 Reserved</p> <p>Bit 4 Activate_FV_C: FALSE: Prozesswerte werden zwischen F-Host und F-Device ausgetauscht. TRUE: Sichere Werte "0" werden zwischen F-Host und F-Device ausgetauscht.</p> <p>Bit 5 Reserved</p> <p>Bit 6 Reserved</p> <p>Bit 7 Reset_Comm: Reset der PROFIsafe-Kommunikation, der Protokoll-Stack wird in den initialen Zustand versetzt. Bit muss für solange auf TRUE gesetzt werden, bis vom PROFIsafe-Status <i>Bit 2 Reset_Comm</i> der der Wert TRUE zurückgelesen wird.</p>
PROFIsafe RoundTrip Time last	<p>Dies ist für einen F-Host die Zeit, zwischen dem Absenden einer Datennachricht (mit Consecutive Number N) und dem Empfang des zugehörigen Acknowledgments (mit Consecutive Number N), gemessen in Millisekunden.</p>

Element	Beschreibung
PROFIsafe Status	<p>PROFIsafe empfängt auf dem Host in jeder Nachricht das PROFIsafe-Status-Byte, welches im Anwenderprogramm ausgewertet werden kann.</p> <p>Bit 0 iPar_OK_S TRUE: Neue iParameter erhalten FALSE: Keine Änderung</p> <p>Bit 1 OA_Req_S Operator Acknowledge Requested.</p> <p>Bit 2 Reset_Comm ist der Rückgelesene Wert Reset_Comm aus dem Host-Control Byte. Dieses Bit signalisiert, ob der Reset_Comm angekommen ist.</p> <p>Bit 3 FV_activated_S</p> <p>Bit 4 Toggle_h</p> <p>Bit 5 Device_Fault TRUE: Das F-Device hat einen Device-Fault gemeldet. FALSE: Das F-Device hat keinen Device-Fault gemeldet.</p> <p>Bit 6 WD_timeout TRUE: Entweder hat das F-Device einen WD-Timeout gemeldet oder auf dem F-Host ist der Host-Timeout abgelaufen. FALSE: Weder auf dem F-Device noch auf dem F-Host ist ein Timeout aufgetreten.</p> <p>Bit 7 CRC TRUE: Entweder hat das F-Device einen CRC-Fehler gemeldet oder auf dem F-Host ist ein CRC-Fehler aufgetreten. FALSE: Weder auf dem F-Device noch auf dem F-Host ist ein CRC Fehler aufgetreten.</p>

Tabelle 12: Register Systemvariablen

Im Register **Prozessvariablen** werden die Eingangsvariablen eingetragen.

3.7.6.3 F-Parameter von Submodul Input (Nur für PROFIsafe-Module)

PROFIsafe-F-Devices benötigen für den sicheren Prozessdaten-Austausch die genormten F-Parameter. Das F-Device nimmt die Kommunikation erst auf, wenn gültige F-Parameter eingestellt wurden. Die ausgegrauten Parameter können nicht bearbeitet werden und werden teilweise von der GSDML-Datei vorgegeben oder automatisch berechnet.

Element	Beschreibung
Name	Name des Moduls
Index	Index des Moduls
F_Par_Version	Nur V2-Mode wird unterstützt. V1-Mode wird abgelehnt. Wird über die GSDML-Datei bestimmt.
F_Source_Add	Die F_Source_Adresse des F-Hosts muss innerhalb des PROFIsafe-Netzwerkes eindeutig sein! Wertebereich: 1 ... 65534
F_Dest_Add	Die F_Destination_Adresse des F-Devices muss innerhalb des PROFIsafe-Netzwerkes eindeutig sein! Wertebereich: 1 ... 65534
F_WD_Time	Watchdogtime Wertebereich: 1 ... 65534 ms
F_iPar_CRC	F_iPar_CRC des F-Devices in dieses Feld eingeben.
F_SIL	Anzeige des SIL-Levels 0 - SIL1 1 - SIL2 2 - SIL3 3 - NoSIL Wird über die GSDML-Datei bestimmt.
F_Check_iPar	Anzeige des iParameter CRC Wird über die GSDML-Datei bestimmt.
F_Block_ID	Struktur der F-Parameter Wird über die GSDML-Datei bestimmt.
F_CRC_Length	Gibt an, ob der 3 Byte CRC oder der 4 Byte CRC verwendet wird. Wird über die GSDML-Datei bestimmt.
F_Par_CRC	Anzeige des F-Parameter CRC (CRC1) Wird aus den aktuellen F-Parametern berechnet

Tabelle 13: F-Parameter von Submodul Input (Eigenschaften)

3.7.7 Submodul Output

Mit den Parametern der Submodule werden die Kommunikationsbeziehungen und das Verhalten der Module bei Verbindungsunterbrechung eingestellt.

3.7.7.1 Submodul Output (Eigenschaften)

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü der *Output Submodule* öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Register Parameter

Element	Beschreibung
Name	Name des Output Submoduls
Substeckplatz	Nicht änderbar für H1Max PROFINET IO-Controller. Standardwert: 1
IO Data CR, Ausgänge	Auswahl der Kommunikationsbeziehung (Communication Relation CR), in der die Ausgänge des Submoduls übertragen werden sollen. - keine - Default Input CR
Ausgangsdaten vom Device akzeptiert	Auswahl der Kommunikationsbeziehung (Communication Relation CR) in welcher der IO Consumer Status (CS) des Submoduls übertragen werden soll. - keine - Default Output CR

Tabelle 14: Register Parameter

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Submodulkennung*, *Hardware-/Software Version* oder *Datenlänge*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.7.2 Submodul Output (Edit)

Die Menüfunktion **Edit** aus dem Kontextmenü des Submoduls Output öffnet den Dialog **Edit**, der die folgenden Register enthält:

Register Systemvariablen

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFINET IO-Submoduls im Anwenderprogramm auszuwerten.

Element	Beschreibung	
Ausgangsdaten gültig	TRUE	Ausgangsdaten gültig GOOD
	FALSE	Ausgangsdaten ungültig BAD
Ausgangsdaten vom Device akzeptiert	TRUE	Ausgangsdaten gültig GOOD
	FALSE	Ausgangsdaten ungültig BAD
Die folgenden Parameter sind nur für PROFI-safe-Module verfügbar:		
Eingangsdaten gültig	TRUE	Eingangsdaten gültig GOOD
	FALSE	Eingangsdaten ungültig BAD
Eingangsdaten vom Controller akzeptiert	TRUE	Eingangsdaten gültig GOOD
	FALSE	Eingangsdaten ungültig BAD
Weitere Parameter für PROFI-safe-Module, siehe Tabelle 12.		

Tabelle 15: Register Systemvariablen

i

Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

Im Register **Prozessvariablen** werden die Ausgangsvariablen eingetragen

3.7.7.3 F-Parameter von Submodul Output (Nur für PROFI-safe-Module)

Beschreibung der F-Parameter, siehe Kapitel 3.7.6.3.

3.7.8 Submodul Inputs und Outputs

Mit den Parametern der Submodule werden die Kommunikationsbeziehungen und das Verhalten der Module bei Verbindungsunterbrechung eingestellt.

3.7.8.1 Submodul Inputs und Outputs (Eigenschaften)

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü der *Input und Output Submodule* öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Register Parameter

Element	Beschreibung
Name	Name des Input/Output Submoduls
Substeckplatz	Nicht änderbar für HIMax PROFINET IO-Controller. Standardwert: 1
IO Data CR, Eingänge	Auswahl der Kommunikationsbeziehung (Communication Relation CR), in der die Eingänge des Submoduls übertragen werden sollen. - keine - Default Input CR
IO Data CR, Ausgänge	Auswahl der Kommunikationsbeziehung (Communication Relation CR), in der die Ausgänge des Submoduls übertragen werden sollen. - keine - Default Output CR
Eingangsdaten vom Controller akzeptiert	Auswahl der Kommunikationsbeziehung (Communication Relation CR), in welcher der IO Consumer Status (CS) des Submoduls übertragen werden soll. - keine - Default Output CR
Ausgangsdaten vom Device akzeptiert	Auswahl der Kommunikationsbeziehung (Communication Relation CR), in welcher der IO Consumer Status (CS) des Submoduls übertragen werden soll. - keine - Default Input CR
Input-Werte bei IO-CR Unterbrechung	- Letzten Wert behalten - Initialwerte annehmen

Tabelle 16: Register Parameter

Register Modell und Features

Die Register **Modell** und **Features** zeigen weitere Einzelheiten der GSDML-Datei, wie zum Beispiel *Submodulkennung*, *Hardware-/Software Version* oder *Datenlänge*. Diese Daten dienen als unterstützende Information zur Parametrierung des Geräts und können nicht geändert werden.

3.7.8.2 Submodul Inputs und Outputs (Edit)

Die Menüfunktion **Edit** aus dem Kontextmenü der Submodule Input/Output öffnet den Dialog **Edit**, der die folgenden Register enthält:

Register Systemvariablen

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFINET IO-Submoduls im Anwenderprogramm auszuwerten.

Element	Beschreibung	
Ausgangsdaten gültig	TRUE	Ausgangsdaten gültig GOOD
	FALSE	Ausgangsdaten ungültig BAD
Ausgangsdaten vom Device akzeptiert	TRUE	Ausgangsdaten gültig GOOD
	FALSE	Ausgangsdaten ungültig BAD
Eingangsdaten gültig	TRUE	Eingangsdaten gültig GOOD
	FALSE	Eingangsdaten ungültig BAD
Eingangsdaten vom Controller akzeptiert	TRUE	Eingangsdaten gültig GOOD
	FALSE	Eingangsdaten ungültig BAD
Parameter für PROFIsafe-Module, siehe Tabelle 12.		

Tabelle 17: Register Systemvariablen

Im Register **Prozessvariablen** werden die Eingangsvariablen und Ausgangsvariablen im jeweiligen Bereich eingetragen



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

3.7.8.3 F-Parameter von Submodul Inputs/Outputs (Nur für PROFIsafe-Module)

Beschreibung der F-Parameter, siehe Kapitel 3.7.6.3.

3.7.9 Application Relation (Eigenschaften)

Eine Applikationsbeziehung (AR: Application Relation) ist ein logisches Gebilde zum Datenaustausch zwischen Controller und Device. Die Datenübertragung innerhalb der Applikationsbeziehung findet in diesem Beispiel (siehe Bild 8) über die Standard Kommunikationsbeziehungen (Alarm CR, Default Input CR und Default Output CR) statt. Diese Kommunikationsbeziehungen sind bereits standardmäßig in den Input und Output Modulen eingestellt.

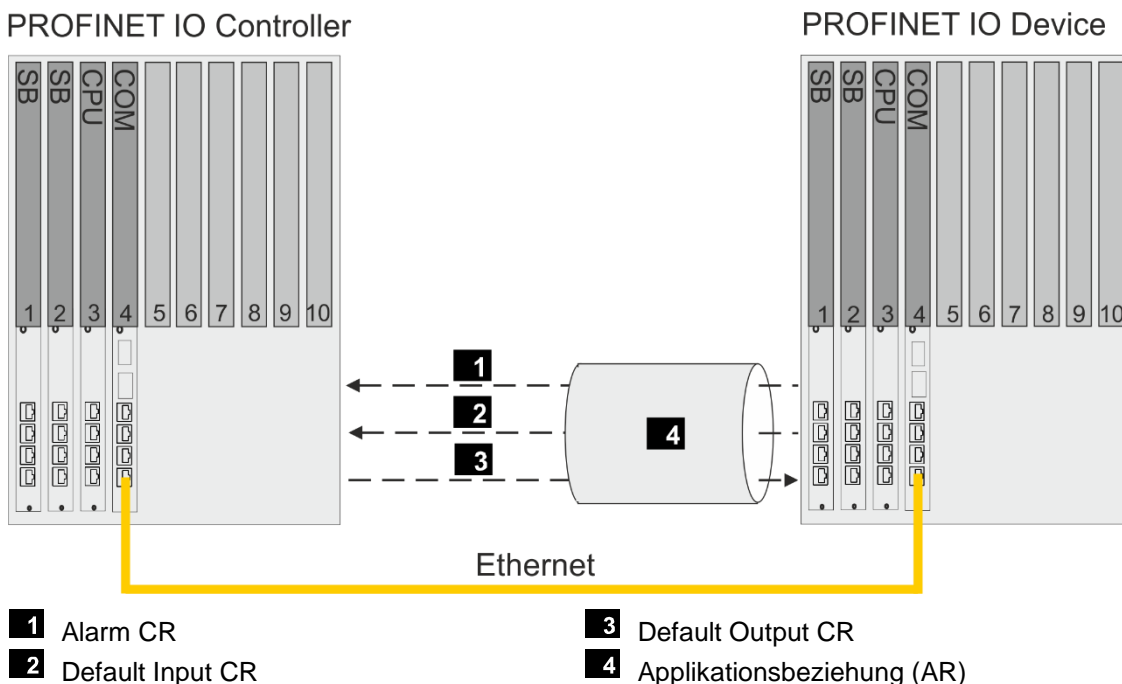


Bild 8: Kommunikation über PROFINET/PROFIsafe

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü von *Application Relation* öffnet den Dialog **Eigenschaften**.

Element	Beschreibung
Name	Nicht änderbar
AR UUID	Kennzahl zur eindeutigen identifizierung der Applikationsbeziehung (AR). Nicht änderbar
Verbindungsaufbau Timeout Faktor	Mit diesem Parameter wird die Zeit berechnet, welche aus Sicht eines PROFINET IO-Devices während des Verbindungsaufbaus zwischen dem Senden der Antwort auf den Connect-Request und dem Empfang eines neuen Requests vom PROFINET IO-Controller maximal verstreichen darf. Wertebereich: 1 ... 1000 (x 100 ms) Standardwert: 600
Supervisor darf die AR übernehmen	Definition, ob ein PROFINET IO-Supervisor die Applikationsbeziehung (AR) übernehmen darf. 0 Nicht erlaubt 1 Erlaubt Standardwert: 0

Tabelle 18: Application Relation (Eigenschaften)

3.7.10 Alarm CR (Eigenschaften)

Innerhalb einer Applikationsbeziehung können mehrere Kommunikationsbeziehungen (CRs: Communication Relations) aufgebaut werden.

Über die Kommunikationsbeziehung Alarm CR überträgt ein PROFINET IO-Device Alarme an den PROFINET IO-Controller.

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü von Application Relation öffnet den Dialog **Eigenschaften**, der die folgenden Register enthält:

Element	Beschreibung
Name	Nicht änderbar
VLAN-ID, hochpriore Alarme	Jedem Virtuellen LAN (VLAN) wird zur Abschottung eine eindeutige Nummer zugeordnet. Ein Gerät, das zum VLAN mit der ID=1 gehört, kann mit jedem anderen Gerät im gleichen VLAN kommunizieren, nicht jedoch mit einem Gerät in einem anderen VLAN mit ID=2, 3, ... Wertebereich, siehe auch IEC-61158-6: 0x000 Kein VLAN 0x001 Standard VLAN 0x002 ... Siehe IEEE 802.1 Q 0xFFFF Standardwert: 0
VLAN-ID, niederpriore Alarme	Beschreibung, siehe VLAN-ID, hochpriore Alarme Standardwert: 0
Alarm Priorität	Benutzer Priorität Die vom Benutzer vergebene verwenden Priorität wird verwendet. Benutzer Priorität Die vom Benutzer vergebene ignorieren Priorität wird ignoriert. Der generierte Alarm ist immer niedrigprior.
Alarm Sendeversuche	Maximale Anzahl an Sendeversuchen des Devices, falls der Controller nicht antwortet. Wertebereich 3 ... 15 Standardwert: 10
Alarm Timeout Faktor	Mit dem RTA Timeout Factor wird die Timeoutzeit des Devices berechnet, welche nach dem Versenden eines RTA_Data(Alarm)-Frames und dem Erhalt des RTA_ACK-Frames maximal verstreichen darf. RTA Timeout = RTA Timeout Factor x 100 ms Wertebereich: 1 ... 65535 Standardwert: 5

Tabelle 19: Alarm CR (Eigenschaften)

3.7.11 Input CR (Eigenschaften)

Über die Kommunikationsbeziehung Input CR überträgt ein PROFINET IO-Device Variablen zum PROFINET IO-Controller.

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü von Default Input CR öffnet den Dialog **Eigenschaften**. Das Dialogfenster enthält die folgenden Parameter:

Element	Beschreibung
Name	Beliebiger, eindeutiger Name für ein Input CR Das Default Input CR ist nicht änderbar
Typ	1 (Nicht änderbar)
Send-Clock-Faktor	Der Send-Clock-Faktor bestimmt den Send Clock der zyklischen Datenübertragung einer IO-CR. Send Clock = Send-Clock-Faktor x 31,25 µs Wertebereich: 1 ... 128 Standardwert: 32
Reduktions-Faktor	Der Reduktions-Faktor ermöglicht die Untersetzung der tatsächlichen Zykluszeit des Sendens der Daten einer IO-CR zur Send Clock. Die tatsächliche Zykluszeit der Daten errechnet sich aus: Send Cycle = Reduktions Faktor x Send Clock Wertebereich: 1 ... 16384 Standardwert: 32 (abhängig vom Gerät)
Watchdog-Faktor	Mit dem Watchdog-Faktor wird diejenige Zeit berechnet, welche aus Sicht eines IO-CR-Consumers zwischen dem Empfang zweier Frames maximal verstreichen darf: Watchdog Time = Watchdog Factor x Send Cycle Wertebereich: 1 ... 7680 Standardwert: 3
VLAN-ID	Jedem Virtuellen LAN (VLAN) wird zur Abschottung eine eindeutige Nummer zugeordnet. Ein Gerät, das zum VLAN mit der ID=1 gehört, kann mit jedem anderen Gerät im gleichen VLAN kommunizieren, nicht jedoch mit einem Gerät in einem anderen VLAN mit ID=2, 3, ... Wertebereich, siehe auch IEC-61158-6: 0x000 Kein VLAN 0x001 Standard VLAN 0x002 ... Siehe IEEE 802.1 Q 0xFFFF Standardwert: 0

Tabelle 20: Input CR (Eigenschaften)

3.7.11.1 Input CR (Edit)

Die Menüfunktion **Edit** aus dem Kontextmenü von Default Input CR öffnet den Dialog **Systemvariablen**, und enthält die folgende Systemvariable:

Element	Beschreibung	
Daten Status Input CR	Wert	Beschreibung
	0	State Primary beschreibt bei redundanten Verbindungen den führenden Kanal. 1 = Primary 0 = Backup Bei Mono Verbindungen gilt: 1 = Verbunden 0 = nicht Verbunden
	1	Nicht verwendet
	2	Data Valid Invalid wird in der Hochlaufphase gesetzt oder wenn die Applikation nicht in der Lage ist, Fehler über den IOPS zu melden. 1 = Valid 0 = Invalid
	3	Nicht verwendet
	4	Process State Hat nur informativen Charakter, tatsächliche Gültigkeit der Daten wird über IOPS gemeldet. 1 = Run 0 = Stop
	5	Problem Indicator Bei Problem detected werden Details über Diagnosedaten der Alarm-CR zur Verfügung gestellt. 1 = Regular operation 0 = Problem detected
	6	Nicht verwendet
	7	Nicht verwendet

Tabelle 21: Input CR (Edit)

3.7.11.2 Output CR (Eigenschaften)

Innerhalb einer Applikationsbeziehung können mehrere Kommunikationsbeziehungen (CRs: Communication Relations) aufgebaut werden.

Über die Kommunikationsbeziehung Output CR überträgt der PROFINET IO-Controller Variablen zum PROFINET IO-Device.

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü von Output CR öffnet den Dialog **Eigenschaften**, der die folgenden Parameter enthält:

Element	Beschreibung
Name	Beliebiger, eindeutiger Name, für ein OUTPUT CR. Das Default Output CR ist nicht änderbar.
Typ	2 (Nicht änderbar)
Send-Clock-Faktor	Der Send-Clock-Faktor bestimmt den Send Clock der zyklischen Datenübertragung einer IO-CR. Send Clock = Send-Clock-Faktor x 31,25 µs Wertebereich: 1 ... 128 Standardwert: 32
Reduktions-Faktor	Zur Einstellung der Übertragungshäufigkeit. Der Reduktions-Faktor ermöglicht die Untersetzung der tatsächlichen Zykluszeit des Sendens der Daten einer IO-CR. Die tatsächliche Zykluszeit der Daten errechnet sich aus: Send Cycle = Reduktions Faktor x Send Clock Wertebereich: 1 ... 16384 Standardwert: 32
Watchdog-Faktor	Mit dem Watchdog-Faktor wird diejenige Zeit berechnet, welche aus Sicht eines IO-CR-Consumers zwischen dem Empfang zweier Frames maximal verstreichen darf: Watchdog Time = Watchdog Faktor x Send Cycle Wertebereich: 1 ... 7680 Standardwert: 3
VLAN-ID	Jedem Virtuellen LAN (VLAN) wird zur Abschottung eine eindeutige Nummer zugeordnet. Ein Gerät, das zum VLAN mit der ID=1 gehört, kann mit jedem anderen Gerät im gleichen VLAN kommunizieren, nicht jedoch mit einem Gerät in einem anderen VLAN mit ID=2, 3, ... Wertebereich, siehe auch IEC-61158-6: 0x000 Kein VLAN 0x001 Standard VLAN 0x002 ... Siehe IEEE 802.1 Q 0xFFFF Standardwert: 0

Tabelle 22: Output CR (Eigenschaften)

3.8 PROFINET IO-Device

Dieses Kapitel beschreibt die Eigenschaften des PROFINET IO-Device sowie die Menüfunktionen und Dialoge in SILworX, die zur Konfiguration des PROFINET IO-Controllers benötigt werden.

3.8.1 Systemanforderung

Benötigte Ausstattung und Systemanforderung:

Element	Beschreibung
Steuerung	HIMax mit COM-Modul HIMatrix (Für HIMatrix F*02 nicht anwendbar)
CPU-Modul	Die Ethernet-Schnittstellen des Prozessormoduls können für PROFINET IO nicht verwendet werden.
COM-Modul	Ethernet 10/100BaseT
Aktivierung	Die Freischaltung erfolgt per Software-Freischaltcode, siehe Kommunikationshandbuch HI 801 100 D.

Tabelle 23: Systemanforderung und Ausstattung für PROFINET IO-Controller

3.8.2 PROFINET IO-Device Eigenschaften

Element	Beschreibung
Sicherheitsbezogen	PROFINET: Nein PROFIsafe: Ja
Übertragungsrate	100 Mbit/s voll duplex
Transportweg	Ethernet-Schnittstellen der COM-Module Verwendete Ethernet-Schnittstellen simultan auch für andere Protokolle nutzbar.
Konformitätsklasse	Das PROFINET IO-Device entspricht den Anforderungen der Conformance Class A.
Real Time Class	RT-Klasse 1
Max. Anzahl PROFINET IO-Devices	Für jedes COM-Modul kann ein PROFINET IO-Device konfiguriert werden.
Max. Anzahl Applikationsbeziehungen (ARs) zum PROFINET IO-Controller	Ein PROFINET IO-Device kann max. 5 Applikationsbeziehungen (ARs) zu einem PROFINET IO-Controllern aufbauen.
Max. Anzahl (CRs pro AR) Kommunikationsbeziehungen	Standard: 1 Input, 1 Output, 1 Alarm
Max. Prozessdatenlänge aller konfigurierten PROFINET IO-Module	Output: max. 1440 Bytes Input: max. 1440 Bytes
Priorisierung der Daten	Möglich über die Einstellung der <i>Reduction Rate</i> auf der Device Ebene.
Die folgenden Eigenschaften gelten für PROFIsafe	
Max. Anzahl F-Devices pro COM-Modul (HIMax und HIMatrix)	63
Max. Prozessdatenlänge einer Kommunikationsbeziehung (CR)	Output: max. 123 Bytes Nutzdaten + 5 Bytes ¹⁾ Input: max. 123 Bytes Nutzdaten + 5 Bytes ¹⁾
Max. Größe Nutzdaten HIMax: HIMatrix	1024 x 123 Bytes = 125 952 Bytes 512 x 123 Bytes = 62 976 Bytes
¹⁾ 5 Bytes Verwaltungsdaten (Status/Control Bytes und CRC)	

Tabelle 24: Eigenschaften PROFINET IO-Controller

3.9 PROFINET IO/PROFIsafe Beispiel

Das folgende Kapitel beschreibt die Konfiguration des PROFINET IO/ PROFIsafe-Device.

3.9.1 Konfiguration des PROFINET IO-Device in SILworX

Anlegen eines neuen PROFINETPROFINET IO-IO Devices

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle** öffnen.
2. Im Kontextmenü von Protokolle **Neu, PROFINET IO-Device** wählen, um ein neues PROFINET IO-Device hinzuzufügen.
3. Im Kontextmenü vom PROFINET IO-Device **Eigenschaften** wählen.
4. Im Feld **Name** den Gerätenamen des PROFINET IO-Devices eintragen.
5. **COM Modul** wählen.

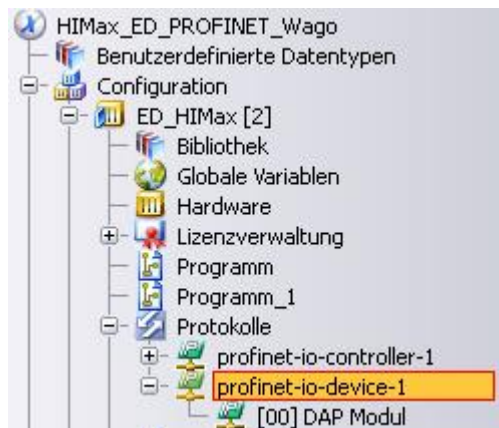


Bild 9: PROFINET IO-Device im HIMAx Strukturbaum

Anlegen der benötigten PROFINET IO-Module

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFINET IO-Device** öffnen.
2. Im Kontextmenü von PROFINET IO-Device **Neu** wählen.
3. Für dieses Beispiel die folgenden passenden Module wählen.

PROFINET IO/ PROFIsafe-Modul	Steckplatz
In 1 Byte	1
Safe Out 1 Byte	2

Nummerieren der PROFINET IO-Device Module

1. Rechtsklick auf das erste **PROFINET IO-Device Modul** und im Kontextmenü **Eigenschaften** wählen.
2. Im Feld **Steckplatz** eine **1** eintragen.
3. Diesen Schritt für weitere **PROFINET IO-Device Module** wiederholen und fortlaufend nummerieren.

i

Module so **nummerieren**, wie das PROFINET IO-Device tatsächlich aufgebaut ist.

Folgender Schritt ist nur für die PROFIsafe-Module notwendig!

4. Im Feld **PROFIsafe F_Destination_Address**, die Adresse des PROFIsafe-Moduls eintragen.

3.9.2 Konfiguration des PROFINET IO-Device Eingangsmoduls



Die Summe der Variablen in Byte muss mit der Größe des jeweiligen Moduls in Byte übereinstimmen.

Eingangsmodul [01] In 1 Byte konfigurieren

1. Im PROFINET IO-Device das Eingangsmodul **[01] In 1 Byte** wählen.
2. Rechtsklicken auf **[01] In 1 Byte** und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Eingangssignale** ziehen.
5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Eingangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.
7. Im Dialog **Edit** das Register **Systemvariablen** wählen, und die Ausgangsvariable *Eingangsdaten vom Device akzeptiert* mit einer globalen Variablen mit dem Initialwert TRUE belegen, wenn eine Steuerung des Consumer/Provider Status nicht gewünscht ist.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

3.9.3 Konfiguration des PROFIsafe-Device Ausgangsmoduls

Ausgangsmodul [02] Safe Out 1 Byte konfigurieren

1. Im PROFIsafe-Device das Ausgangsmodul **[02] Safe Out 1 Byte** wählen.
2. Rechtsklicken auf **[02] Safe Out 1 Byte** und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Ausgangssignale** ziehen.
5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Ausgangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.
7. Im Dialog **Edit** das Register **Systemvariablen** wählen, und die Ausgangsvariablen *Ausgangsvariable gültig* und *Eingangsdaten vom Device akzeptiert* mit einer globalen Variablen mit dem Initialwert TRUE belegen, wenn eine Steuerung des Consumer/Provider Status nicht gewünscht ist.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

3.9.4 Verifikation der PROFINET IO-Device Konfiguration

PROFINET IO-Device Konfiguration verifizieren

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFINET IO-Device** selektieren.
2. Rechtsklick und im Kontextmenü **Verifikation** wählen.
3. Einträge im Logbuch sorgfältig überprüfen, gegebenenfalls korrigieren.

i

Die Konfiguration des PROFINET IO-Device mit dem Anwenderprogramm der PROFINET IO-Device Ressource neu compilieren und in die Steuerungen laden, damit diese für die PROFINET IO-Kommunikation wirksam werden.

3.10 Menüfunktionen des PROFINET IO-Device

3.10.1 Menüfunktion Eigenschaften

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des PROFINET IO-Device öffnet den Dialog **Eigenschaften**.

Element	Beschreibung
Typ	PROFINET IO-Device
Name	Beliebiger, eindeutiger Name, für ein PROFINET IO-Device
Aktualisierungsintervall der Prozessdaten [ms]	Aktualisierungsintervall in Millisekunden, mit der die Daten des Protokolls zwischen COM und CPU ausgetauscht werden. Ist das Aktualisierungsintervall Null oder kleiner als die Zykluszeit der Steuerung, dann erfolgt der Datenaustausch so schnell wie möglich. Wertebereich: 4 ... (2 ³¹ -1) Standardwert: 0
Prozessdaten-Konsistenz erzwingen	Aktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM innerhalb eines Zyklus der CPU. Deaktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM, verteilt über mehrere CPU Zyklen zu je 1100 Byte pro Datenrichtung. Damit kann eventuell auch die Zykluszeit der Steuerung reduziert werden. Standardwert: Aktiviert
Modul	Auswahl des COM-Moduls auf dem dieses Protokoll abgearbeitet wird.
Max. µP-Budget aktivieren	Aktiviert: Limit des µP-Budget aus dem Feld <i>Max. µP-Budget in [%]</i> übernehmen. Deaktiviert: Kein Limit des µP-Budget, für dieses Protokoll verwenden.
Max. µP-Budget in [%]	Maximales µP-Budget des Moduls, welches bei der Abarbeitung des Protokolls produziert werden darf. Wertebereich: 1 ... 100% Standardwert: 30%
RPC Port Server	Remote Procedure Call Port Wertebereich: 1024 ... 65535 Standardwert: 49152 RPC Port Server und RPC Port Client dürfen nicht identisch sein!
RPC Port Client	Remote Procedure Call Port Wertebereich: 1024 ... 65535 Standardwert: 49153 RPC Port Server und RPC Port Client dürfen nicht identisch sein!
AutoAwait-FParams-OnConnLoss	Dieser Parameter wird nur für PROFI-safe-Module verwendet. Nach jedem Verbindungsverlust zum F-Host müssen die F-Parameter erneut in das F-Device geladen werden. Um die Inbetriebnahme von PROFI-safe zu erleichtern kann dieser Parameter aktiviert werden. Danach übernimmt das F-Device bei Neustart oder bei Verbindungsverlust automatisch die erforderlichen F-Parameter. Nach der Inbetriebnahme muss dieser Parameter für PROFI-safe konformes Verhalten zwingend deaktiviert werden. Aktiviert: Das F-Device geht automatisch in den Zustand <i>Warten auf F-Parameter</i> . Deaktiviert: Das F-Device muss vom Anwender manuell über ein Online-Kommando in den Zustand <i>Warten auf F-Parameter</i> gebracht werden. Standardwert: Deaktiviert

Tabelle 25: Allgemeine Eigenschaften des PROFINET IO-Device

3.10.2 PROFINET IO-Module

Die folgenden PROFINET IO-Module stehen im PROFINET IO-Device zur Verfügung.

PROFINET IO-Modul	Max. Größe Eingangsvariablen	Max. Größe Ausgangsvariablen
In 1 Byte	1 Byte	
In 2 Bytes	2 Bytes	
In 4 Bytes	4 Bytes	
In 8 Bytes	8 Bytes	
In 16 Bytes	16 Bytes	
In 32 Bytes	32 Bytes	
In 64 Bytes	64 Bytes	
In 128 Bytes	128 Bytes	
In 256 Bytes	256 Bytes	
In 512 Bytes	512 Bytes	
In 1024 Bytes	1024 Bytes	
In-Out 1 Byte	1 Byte	1 Byte
In-Out 2 Bytes	2 Bytes	2 Bytes
In-Out 4 Bytes	4 Bytes	4 Bytes
In-Out 8 Bytes	8 Bytes	8 Bytes
In-Out 16 Bytes	16 Bytes	16 Bytes
In-Out 32 Bytes	32 Bytes	32 Bytes
In-Out 64 Bytes	64 Bytes	64 Bytes
In-Out 128 Bytes	128 Bytes	128 Bytes
In-Out 256 Bytes	256 Bytes	256 Bytes
In-Out 512 Bytes	512 Bytes	512 Bytes
Out 1 Byte		1 Byte
Out 2 Bytes		2 Bytes
Out 4 Bytes		4 Bytes
Out 8 Bytes		8 Bytes
Out 16 Bytes		16 Bytes
Out 32 Bytes		32 Bytes
Out 64 Bytes		64 Bytes
Out 128 Bytes		128 Bytes
Out 256 Bytes		256 Bytes
Out 512 Bytes		512 Bytes
Out 1024 Bytes		1024 Bytes

Tabelle 26: PROFINET IO-Module

3.10.3 HIMA PROFIsafe-Module

Die folgenden PROFIsafe-Module stehen im PROFINET IO-Device zur Verfügung.

PROFIsafe-Modul	Max. Größe Eingangsvariablen	Max. Größe Ausgangsvariablen
Safe In 1 Byte	1 Byte	
Safe In 2 Bytes	2 Bytes	
Safe In 4 Bytes	4 Bytes	
Safe In 8 Bytes	8 Bytes	
Safe In 16 Bytes	16 Bytes	
Safe In 32 Bytes	32 Bytes	
Safe In 64 Bytes	64 Bytes	
Safe In 123 Bytes	123 Bytes	
Safe In-Out 1 Byte	1 Byte	1 Byte
Safe In-Out 2 Bytes	2 Bytes	2 Bytes
Safe In-Out 4 Bytes	4 Bytes	4 Bytes
Safe In-Out 8 Bytes	8 Bytes	8 Bytes
Safe In-Out 16 Bytes	16 Bytes	16 Bytes
Safe In-Out 32 Bytes	32 Bytes	32 Bytes
Safe In-Out 64 Bytes	64 Bytes	64 Bytes
Safe In-Out 123 Bytes	123 Bytes	123 Bytes
Safe Out 1 Byte		1 Byte
Safe Out 2 Bytes		2 Bytes
Safe Out 4 Bytes		4 Bytes
Safe Out 8 Bytes		8 Bytes
Safe Out 16 Bytes		16 Bytes
Safe Out 32 Bytes		32 Bytes
Safe Out 64 Bytes		64 Bytes
Safe Out 123 Bytes		123 Bytes

Tabelle 27: PROFIsafe-Module

Anlegen eines PROFINet oder PROFIsafe-Moduls

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFINET IO-Device** öffnen.
2. Im Kontextmenü von PROFINET IO-Device **Module einfügen** wählen.
3. Passendes Modul wählen, um die erforderlichen Prozessdaten zu transportieren.
4. Rechtsklick auf das ausgewählte Modul und **Edit** wählen.
 - Im Register **Prozessvariablen** die Eingangs- und/oder Ausgangsvariablen einfügen
 - Im Register **Systemvariablen** die Ausgangsvariablen *Eingangsdaten vom Controller akzeptiert* und *Ausgangsdaten gültig* mit einer globalen Variablen mit dem Initialwert TRUE belegen, wenn eine Steuerung des Consumer/Provider Status nicht gewünscht ist.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

Die folgende Einstellung ist nur für PROFIsafe-Module notwendig.

- Im Register **Eigenschaften** den *Steckplatz* und die *F_Destination_Address* eingeben.

3.10.4 PROFINet IO und PROFIsafe-Modul

Mit den Parametern der Module werden die Kommunikationsbeziehungen und das Verhalten der Module bei Verbindungsunterbrechung eingestellt.

Eigenschaften

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü der Module öffnet den Dialog **Eigenschaften**. Das Dialogfenster enthält die folgenden Register:

Element	Beschreibung
Name	Name des Device Moduls
Steckplatz	0 ... 32767
Modul Kennung	Eindeutige Zahl
Substeckplatz	Anzahl Substeckplätze
Prozessdatenverhalten	Wert der Prozessdaten nach einem Verbindungsabbruch - Letzte gültige Prozessdaten behalten - Initialdaten annehmen
Länge der IO-Input Daten	1 ... 123
Länge der IO-Output Daten	1 ... 123
PROFIsafe F_Destination_Address	Die F_Destination_Adresse des F-Devices muss innerhalb des PROFIsafe-Netzwerkes eindeutig sein! Wertebereich: 1 ... 65534

Tabelle 28: Allgemeine Eigenschaften des Device Moduls

Edit

Die Menüfunktion **Edit** aus dem Kontextmenü der Submodule öffnet den Dialog **Edit**.

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des Submoduls im Anwenderprogramm auszuwerten.



Diese Systemvariablen können zur Steuerung des Consumer/Provider Status verwendet werden, siehe Kapitel 3.2.

Element	Beschreibung
Ausgangsdaten gültig	TRUE Ausgangsdaten gültig GOOD
	FALSE Ausgangsdaten ungültig BAD
Ausgangsdaten vom Controller akzeptiert	TRUE Ausgangsdaten gültig GOOD
	FALSE Ausgangsdaten ungültig BAD
Eingangsdaten gültig	TRUE Eingangsdaten gültig GOOD
	FALSE Eingangsdaten ungültig BAD
Eingangsdaten vom Device akzeptiert	TRUE Eingangsdaten gültig GOOD
	FALSE Eingangsdaten ungültig BAD

Element	Beschreibung
Die folgenden Variablen werden nur für PROFIsafe-Module verwendet:	
PROFIsafe Control	<p>Das vom Controller gesendete PROFIsafe-Control-Byte kann im Device ausgelesen werden, siehe auch Kapitel 3.3.2.</p> <p>Bit 0 iPar_EN_DC Freigabe vom Controller entriegelt das Device um neue iParameter in das Device zu laden.</p> <p>Bit 1 OA_Req_DC Operator Acknowledge aus dem Host-Control Byte.</p> <p>Bit 2 Reset_Comm ist der Rückgelesene Wert Reset_Comm aus dem Host-Control Byte</p> <p>Bit 3 activate_FV_DC FALSE: Prozesswerte werden zwischen F-Host und F-Device ausgetauscht. TRUE: Failsafe Values "0" werden zwischen F-Host und F-Device ausgetauscht.</p> <p>Bit 4 Toggle_d Toggle Bit des F-Device</p> <p>Bit 5 Cons_nr_R Die Consecutive Number wird immer dann übernommen, wenn das Toggle_d Bit eine Änderung zwischen zwei aufeinander folgenden Control Bytes aufweist, d. h. dies ist unabhängig des Auftretens eines Fehlers.</p> <p>Bit 6 F_ParamValid TRUE: F-Parametrierung ist erfolgt FALSE: sonst</p> <p>Bit 7 F_Param_ConfiguredTwice TRUE: Das F-Device ist mindestens zweimal mit unterschiedlichen F-Parameter parametriert worden FALSE: sonst</p>
PROFIsafe F_iPar_CRC	<p>iParameter sind individuelle oder technologie-spezifische F-Device parameter. Der iPar_CRC ergibt sich aus der Konfiguration des F-Device.</p> <hr/> <p>i Es liegt in der Verantwortung des Benutzers nach der iParametrierung den richtigen iPar_CRC einzustellen und dann in den scharfen Betrieb zu wechseln.</p> <hr/>
PROFIsafe F_SIL	<p>0 SIL1</p> <p>1 SIL2</p> <p>2 SIL3</p> <p>3 Kein SIL</p>

Element	Beschreibung
PROFIsafe RoundTrip Time last	PROFIsafe muss auf einem F-Host die RoundTripTimeLast bestimmen. Dies ist für einen F-Host die Zeit, die zwischen dem Absenden einer Datennachricht (mit Consecutive Number N) und dem Empfang des zugehörigen Acknowledgments (mit Consecutive Number N) vergeht, gemessen in Millisekunden.
PROFIsafe Status	<p>PROFIsafe sendet vom Device mit jeder Nachricht das PROFIsafe-Status-Byte an den Controller, welches im Device-Anwenderprogramm gesetzt werden kann. Siehe auch Kapitel 3.3.2.</p> <p>Bit 0 iPar_OK_DS Neue iParameter erhalten.</p> <p>Bit 1 Device_Fault_DS TRUE: Device-Fehler FALSE: Kein Device-Fehler Wird erst ab dem PROFIsafe-Zustand <i>21 Await Message</i> berücksichtigt.</p> <p>Bit 2 Reserved</p> <p>Bit 3 Reserved</p> <p>Bit 4 FV_activated_DS Fail-safe value aktiviert</p> <p>Bit 5 Reserved</p> <p>Bit 6 Reserved</p> <p>Bit 7 Reset_Comm Protokoll-Stack wird in den initialen Zustand überführt.</p>

Tabelle 29: Edit Dialog Submodul

Im Register **Prozessvariablen** werden die Eingangsvariablen eingetragen.

4 PROFIBUS DP

PROFIBUS DP ist ein internationaler, offener Feldbusstandard und wird überall dort eingesetzt, wo eine schnelle Reaktionszeit bei vornehmlich kleinen Datenmengen gefordert wird.

Der HIMA PROFIBUS DP Master und der HIMA PROFIBUS DP Slave erfüllen die Kriterien der europäischen Norm EN 50170 [7] und der weltweit verbindlichen Norm IEC 61158 für PROFIBUS DP.

Der HIMA PROFIBUS DP Master kann zyklisch und azyklisch Daten mit den PROFIBUS DP Slaves austauschen.

Zum azyklischen Datenaustausch stehen Ihnen in SILworX verschiedene Funktionsbausteine zur Verfügung. Mit diesen Funktionsbausteinen können der HIMA PROFIBUS DP Master und die PROFIBUS DP Slaves optimal den Erfordernissen des Projekts angepasst werden.

Eine redundante PROFIBUS DP Verbindung kann nur durch die Konfiguration eines zweiten PROFIBUS DP Master/Slave und Anpassungen im Anwenderprogramm erreicht werden.

- PROFIBUS DP Master (Siehe Kapitel 4.1)
- PROFIBUS DP Slave (Siehe Kapitel 4.13)

4.1 HIMA PROFIBUS DP Master

Dieses Kapitel beschreibt die Eigenschaften des HIMA PROFIBUS DP Master sowie die Menüfunktionen und Dialoge in SILworX, die zur Konfiguration des HIMA PROFIBUS DP Master benötigt werden.

Benötigte Ausstattung und Systemanforderungen:

Element	Beschreibung
HIMA Steuerung	HIMax mit COM-Modul HIMatrix ab CPU BS V7 und COM BS V12 HIQuad X: Nicht anwendbar
COM-Modul	Das COM-Modul muss an der verwendeten seriellen Feldbus-Schnittstelle (FB1 oder FB2) mit einem optionalen HIMA PROFIBUS DP Master Submodul ausgerüstet sein.
Aktivierung	Freischaltung durch Feldbus-Submodul, siehe Kommunikationshandbuch HI 801 100 D.

Tabelle 30: Benötigte Ausstattung und Systemanforderungen

i

HIMA empfiehlt für HIMax, PROFIBUS DP über die Feldbus-Schnittstelle FB1 (Übertragungsrate maximal 12 MBit) zu betreiben. Über die Feldbus-Schnittstelle FB2 ist eine maximale Übertragungsrate von 1,5 MBit zugelassen.

PROFIBUS DP Master Eigenschaften:

Element	Beschreibung
Typ des HIMA PROFIBUS DP Master	DP-V1 Master Klasse 1 mit zusätzlichen DP-V2 Funktionen
Übertragungsrate	9,6 kbit/s ... 12 Mbit/s
Busadresse	0 ... 125
Max. Anzahl PROFIBUS DP Master	Pro HIMax COM-Modul oder HIMatrix F30, F35, F60 können zwei PROFIBUS DP Master konfiguriert werden. Pro HIMatrix F20 nur ein PROFIBUS DP Master.
Max. Anzahl PROFIBUS DP Slaves	Es können bis zu 122 Slaves pro Ressource (in allen Master Protokollinstanzen) konfiguriert werden. Hierbei besteht die Restriktion, dass maximal 31 Slaves an ein Bussegment ohne Repeater angeschlossen werden können.
Max. Prozessdatenlänge zu einem Slave	DP-Output: max. 244 Bytes DP-Input: max. 244 Bytes

Tabelle 31: Eigenschaften PROFIBUS DP Master

Nach der Norm sind insgesamt drei Repeater zulässig, so dass maximal 122 Busteilnehmer pro serielle Schnittstelle eines Masters möglich sind.

4.1.1 Anlegen eines HIMA PROFIBUS DP Master

Anlegen eines neuen HIMA PROFIBUS DP Masters

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle** öffnen.
2. Im Kontextmenü von Protokolle **Neu, PROFIBUS DP Master** wählen, um einen neuen PROFIBUS DP Master hinzuzufügen.
3. Im Kontextmenü vom PROFIBUS DP Master **Eigenschaften, Allgemein** wählen.
4. **Modul** und die **Schnittstelle** wählen.

4.2 PROFIBUS DP Beispiel

In diesem Beispiel tauscht ein HIMA PROFIBUS DP Master Variablen mit einem HIMA PROFIBUS DP Slave aus.

Hier wird gezeigt, wie ein HIMA PROFIBUS DP Master und ein HIMA PROFIBUS DP Slave angelegt und parametrieren werden.

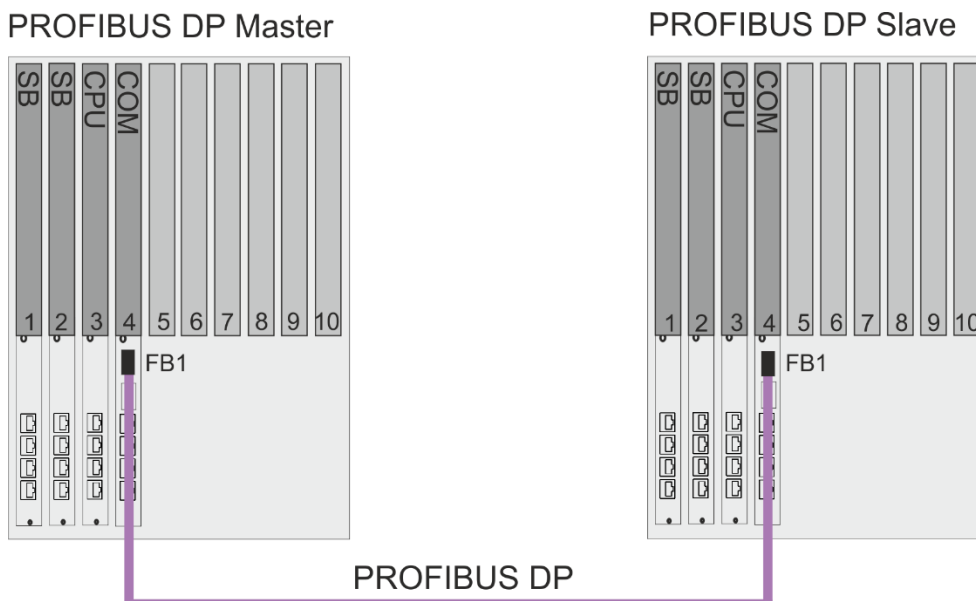


Bild 10: Kommunikation über PROFIBUS DP

Die COM-Module der beiden HIMA Steuerungen müssen an der Feldbus-Schnittstelle 1 mit dem jeweiligen PROFIBUS DP Submodul ausgestattet sein, siehe Kommunikationshandbuch HI 801 100 D.

Folgende globalen Variablen müssen für dieses Beispiel in SILworX angelegt werden:

Globale Variable	Typ
PB_Slave_Master1	UINT
PB_Slave_Master2	DWORD
PB_Slave_Master3	DWORD
PB_Slave_Master4	BYTE
PB_Master_Slave1	UINT
PB_Master_Slave2	BYTE

4.2.1 PROFIBUS DP Slave Konfigurieren

Konfiguration des PROFIBUS DP Slave.

Anlegen eines neuen HIMA PROFIBUS DP Slave

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle** öffnen.
2. Im Kontextmenü von Protokolle **Neu, PROFIBUS DP Slave** wählen um einen neuen PROFIBUS DP Slave hinzuzufügen.
3. Im Kontextmenü vom PROFIBUS DP Slave **Edit** wählen.
4. Im Register **Eigenschaften** das **COM-Modul** und die **Schnittstelle** (z. B. FB1) wählen.

Variablen Zuordnung im HIMA PROFIBUS DP Slave:

1. Im Kontextmenü des PROFIBUS DP Slave **Edit** wählen.
2. Im Dialog **Edit** das Register **Prozessvariablen** wählen.

i

Die Startadresse der HIMA PROFIBUS DP Slave Ein- und Ausgangsvariablen beginnt immer bei 0. Erwartet der PROFIBUS DP Master (eines anderen Herstellers) eine höhere Startadresse, müssen Dummy-Variablen vor den Nutzvariablen eingefügt werden.

Ausgänge im HIMA PROFIBUS DP Slave

Name	Typ	Offset	Globale Variable
PB_Slave_Master1	UINT	0	PB_Slave_Master1
PB_Slave_Master2	DWORD	2	PB_Slave_Master2
PB_Slave_Master3	DWORD	6	PB_Slave_Master3
PB_Slave_Master4	BYTE	10	PB_Slave_Master4

Tabelle 32: Ausgänge HIMA PROFIBUS DP Slave

1. In der Objektauswahl die globalen Variablen zum Versenden per Drag&Drop in den Bereich **Ausgangsvariablen** ziehen.

i

Die Ausgangsvariablen des HIMA PROFIBUS DP Slave bestehen in diesem Beispiel aus **4 Variablen** mit insgesamt **11 Bytes**. Die Ausgangsvariable mit dem niedrigsten Offset hat die Startadresse **0**.

2. Kontextmenü mit Rechtsklick auf eine leere Stelle im Bereich **Ausgangsvariablen** öffnen.
3. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

Eingänge im HIMA PROFIBUS DP Slave

Name	Typ	Offset	Globale Variable
PB_Master_Slave1	UINT	0	PB_Master_Slave1
PB_Master_Slave2	BYTE	2	PB_Master_Slave2

Tabelle 33: Eingänge HIMA PROFIBUS DP Slave

1. In der Objektauswahl die globalen Variablen zum Empfangen per Drag&Drop in den Bereich **Eingangsvariablen** ziehen.

i

Die Eingangsvariablen des HIMA PROFIBUS DP Slave bestehen in diesem Beispiel aus **2 Variablen** mit insgesamt **3 Bytes**. Die Eingangsvariable mit dem niedrigsten Offset hat die Startadresse **0**.

2. Kontextmenü mit Rechtsklick auf eine leere Stelle im Bereich **Eingangsvariablen** öffnen.
3. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

PROFIBUS DP Slave Konfiguration verifizieren

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Slave** selektieren.
2. Rechtsklick und im Kontextmenü **Verifikation** wählen.
3. Einträge im Logbuch sorgfältig überprüfen, gegebenenfalls korrigieren.



Die Konfiguration des PROFIBUS DP Slave muss mit dem Anwenderprogramm der PROFIBUS DP Slave-Ressource neu compiliert und in die Steuerungen geladen werden, damit diese für die PROFIBUS DP Kommunikation wirksam werden.

4.2.2 PROFIBUS DP Master Konfigurieren

Anlegen eines neuen HIMA PROFIBUS DP Master

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle** öffnen.
2. Im Kontextmenü von Protokolle **Neu, PROFIBUS DP Master** wählen um einen neuen PROFIBUS DP Master hinzuzufügen.
3. Im Kontextmenü vom PROFIBUS DP Master **Eigenschaften, Allgemein** wählen.
4. Im Register **Allgemein** das **COM-Modul** und die **Schnittstelle** (z. B. FB1) wählen.



Die folgenden Schritte dienen der Konfiguration des HIMA PROFIBUS DP Slave im HIMA PROFIBUS DP Master.

Anlegen eines HIMA PROFIBUS DP Slave im PROFIBUS DP Master

1. Im Kontextmenü des PROFIBUS DP Master **Neu, PROFIBUS DP Slave** wählen.

GSD-Datei für den neuen PROFIBUS DP Slave einlesen

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, PROFIBUS Slave** öffnen.
2. Im Kontextmenü vom PROFIBUS DP Master **GSD-Datei einlesen** wählen und die zum PROFIBUS Slave zugehörige GSD-Datei wählen (z. B. hax100ea.gsd).



Die GSD-Dateien für HIMA Steuerungen ist auf der HIMA Webseite www.hima.com zu finden.

4.2.2.1 Anlegen der HiMax PROFIBUS DP Module

Im PROFIBUS DP Master muss die Anzahl der tatsächlich zu übertragenden Bytes konfiguriert werden. Dies durch Auswahl von *Modulen* erreicht, bis die physikalische Konfiguration des Slaves erreicht ist.

i

Es ist nicht von Bedeutung, wie viele Module verwendet werden, um auf die erforderliche Anzahl an Bytes zu kommen, so lange die Anzahl von maximal 32 Modulen nicht überschritten wird.

Um die Konfiguration des PROFIBUS DP Master nicht unnötig zu erschweren, sollte die Zahl der gewählten Module möglichst klein gehalten werden.

Anlegen der benötigten PROFIBUS DP Module

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, PROFIBUS Slave** öffnen.
2. In der Menüleiste **PROFIBUS DP Master, Module einfügen** wählen.
3. Für dieses Beispiel die folgenden passenden Module wählen, um vom PROFIBUS DP Slave **11 Bytes** zu Empfangen und **3 Bytes** zu senden.

Nummerieren der PROFIBUS DP Module

1. Rechtsklick auf das erste **PROFIBUS DP Modul** und im Kontextmenü **Eigenschaften** wählen.
2. **0** im Feld **Steckplatz** eintragen.
3. Diesen Schritt für weitere **PROFIBUS DP Modul** wiederholen und fortlaufend nummerieren.

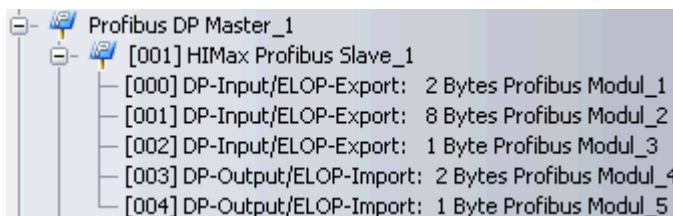


Bild 11: HiMax PROFIBUS DP Slave mit Modulen

i

Die HiMax PROFIBUS DP Module bei **0** beginnend in aufsteigender Reihenfolge und ohne Lücken **nummerieren**.

Die Reihenfolge der PROFIBUS DP Module ist dabei für die Funktion nicht von Bedeutung. Zur besseren Übersicht sollten jedoch die DP-Input-Module und die DP-Output-Module geordnet angelegt werden.

4.2.2.2 Konfiguration der Eingangs-/Ausgangsmodule

i

Die Summe der Variablen in Byte muss mit der Größe des jeweiligen Moduls in Byte übereinstimmen.

Eingangsmodul [000] DP-Input/ELOP-Export: 2 Bytes konfigurieren

1. Im PROFIBUS DP Slave das Eingangsmodul **[000] DP-Input/ELOP-Export: 2 Bytes** wählen.
2. Rechtsklicken auf Eingangsmodul und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Eingangssignale** des Eingangsmoduls **[000] DP-Input/ELOP-Export: 2 Bytes** ziehen.

Name	Typ	Offset	Globale Variable
PB_Slave_Master1	UINT	0	PB_Slave_Master1

Tabelle 34: Variablen Eingangsmodul [000] DP-Input/ELOP-Export: 2 Bytes

5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Eingangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

Eingangsmodul [001] DP-Input/ELOP-Export: 8 Bytes konfigurieren

1. Im PROFIBUS DP Slave das Eingangsmodul **[001] DP-Input/ELOP-Export: 8 Bytes** wählen.
2. Rechtsklicken auf Eingangsmodul und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Eingangssignale** des Eingangsmoduls **[001] DP-Input/ELOP-Export: 8 Bytes** ziehen.

Name	Typ	Offset	Globale Variable
PB_Slave_Master2	DWORD	0	PB_Slave_Master2
PB_Slave_Master3	DWORD	4	PB_Slave_Master3

Tabelle 35: Variablen Eingangsmodul [001] DP-Input/ELOP-Export: 8 Bytes

5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Eingangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

Eingangsmodul [002] DP-Input/ELOP-Export: 1 Byte konfigurieren

1. Im PROFIBUS DP Slave das Eingangsmodul **[002] DP-Input/ELOP-Export: 1 Byte** wählen.
2. Rechtsklicken auf Eingangsmodul und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Eingangssignale** des Eingangsmoduls **[002] DP-Input/ELOP-Export: 1 Byte** ziehen.

Name	Typ	Offset	Globale Variable
PB_Slave_Master4	BYTE	0	PB_Slave_Master4

Tabelle 36: Variablen Eingangsmodul [002] DP-Input/ELOP-Export: 1 Byte

5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Eingangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

Ausgangsmodul [003] DP-Output/ELOP-Import 2 Bytes konfigurieren

1. Im PROFIBUS DP Slave das Ausgangsmodul **[003] DP-Output/ELOP-Import 2 Bytes** wählen.
2. Rechtsklicken auf Ausgangsmodul und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Ausgangssignale** des Ausgangsmoduls **[003] DP-Output/ELOP-Import 2 Bytes** ziehen.

Name	Typ	Offset	Globale Variable
PB_Master_Slave1	UINT	0	PB_Master_Slave1

Tabelle 37: Variablen Ausgangsmodul [003] DP-Output/ELOP-Import: 2 Bytes

5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Ausgangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

Ausgangsmodul [004] DP-Output/ELOP-Import 1 Byte konfigurieren

1. Im PROFIBUS DP Slave das Ausgangsmodul **[004] DP-Output/ELOP-Import 1 Byte** wählen.
2. Rechtsklicken auf Ausgangsmodul und im Kontextmenü **Edit** wählen.
3. Im Dialog **Edit** das Register **Prozessvariablen** wählen.
4. In der Objektauswahl die passende Variable wählen und per Drag&Drop in den Bereich **Ausgangssignale** des Ausgangsmoduls **[004] DP-Output/ELOP-Import 1 Byte** ziehen.

Name	Typ	Offset	Globale Variable
PB_Master_Slave2	BYTE	0	PB_Master_Slave2

Tabelle 38: Variablen Ausgangsmodul [004] DP-Output/ELOP-Import: 1 Byte

5. Kontextmenü durch einen Rechtsklick auf eine leere Stelle im Bereich **Ausgangssignale** öffnen.
6. Im Kontextmenü **Neue Offsets** wählen, damit die Offsets der Variablen neu generiert werden.

4.2.2.3 Anlegen der Benutzerdaten im PROFIBUS DP Master

Anlegen der Benutzerdaten im PROFIBUS DP Master

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, PROFIBUS DP Slave** öffnen.
2. Rechtsklick auf **PROFIBUS DP Slave** und im Kontextmenü **Eigenschaften** wählen.
3. Register **Daten** wählen und die Schaltfläche **Edit** neben den Benutzerdaten anklicken.
In dem 32 Byte langen Benutzerdatenfeld werden die *Startadresse* und die *Anzahl der Variablen der Blöcke* definiert, siehe auch Kapitel 4.8.
4. Für dieses Beispiel müssen die folgenden Benutzerdaten eingetragen werden:
4, damit **4 Variablen** vom PROFIBUS DP Master Empfangen werden.

2 damit **2 Variablen** vom PROFIBUS DP Master gesendet werden.
Die Startadresse des Eingangs- und Ausgangsblocks beginnt jeweils bei **0**.

Bild 12: Benutzerdatenfeld

PROFIBUS DP Slave Konfiguration verifizieren

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master** selektieren.
2. Rechtsklick und im Kontextmenü **Verifikation** wählen.
3. Einträge im Logbuch sorgfältig überprüfen, gegebenenfalls korrigieren.

F	Datum/Zeit	Stufe	
2	07.09.2007 11:39:31.718	Fehler	Bitte wählen sie eine Schnittstelle für den Profibus DP Master
3	07.09.2007 11:39:31.718	Info	Verifikation beendet. Warnungen: 0, Fehler: 2.

Fehler: Verifikation beendet. Warnungen: 0, Fehler: 2.

Bild 13: Dialogfenster Verifikation

i

Die Konfiguration des PROFIBUS DP Master muss mit dem Anwenderprogramm der PROFIBUS DP Master Ressource neu kompiliert und in die Steuerung geladen werden, damit sie für die PROFIBUS DP Kommunikation wirksam werden.

4.2.2.4 Optimieren der PROFIBUS DP Parameter

Mit den Standardwerten der PROFIBUS Parameter ist eine PROFIBUS Kommunikation in den meisten Fällen problemlos möglich. Durch eine weitere Optimierung der Einstellungen kann jedoch ein schnellerer Datenaustausch und eine bessere Fehlererkennung erreicht werden.

Ermitteln der tatsächlichen Target Rotation Time TTR [ms]

1. Control Panel der HIMax PROFIBUS DP Master öffnen.
2. Im Strukturbaum des Control Panels **PROFIBUS DP Master** wählen und die tatsächliche **Target Rotation Time TTR [ms]** ablesen. Diesen Wert notieren.

Ermitteln der benötigten Parameter vom PROFIBUS DP Slave

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, PROFIBUS DP Slave** öffnen.
2. Rechtsklick auf HIMax **PROFIBUS Slave** und **Eigenschaften** wählen.
3. Register **Features** wählen und Parameter **Min. Slave Interval MSI [ms]** für diesen PROFIBUS DP Slave ablesen. Diesen Wert notieren.
4. Register **Übertragungsraten** wählen und Parameter **Max. Tsdr** für die verwendete Übertragungsrate ablesen. Diesen Wert notieren.

Ermittelten Parameter eintragen

1. Rechtsklick auf **PROFIBUS Master** und im Kontextmenü **Eigenschaften**.
2. Register **Zeiten** wählen.

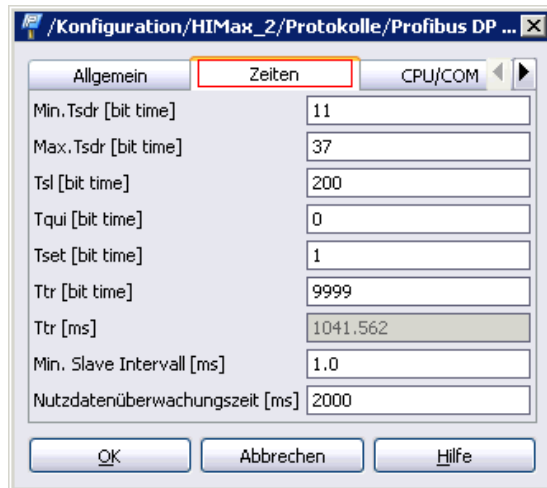


Bild 14: Eigenschaften des PROFIBUS DP Master

3. Abgelesene **Max. Tsdr** in **bit Time** umrechnen und eintragen.
4. Abgelesene **Target Rotation Time TTR [ms]** in **bit Time** umrechnen und mit zuzüglich 1/3 Sicherheitszuschlag in das Feld **Target Rotation Time TTR [ms]** eintragen.
5. Abgelesene **Min. Slave Interval MSI [ms]** eintragen.

i

Sind mehrere Slaves konfiguriert, muss der höchste Wert von MaxTsdr [bit time] und der größte von Min. Slave Intervall [ms] verwendet werden.

6. Die Nutzdatenüberwachungszeit [ms] ist $\geq 6 \cdot Ttr$ [ms] einzustellen.

Watchdog-Zeit für PROFIBUS DP Slave eintragen

1. Im Strukturbaum **Konfiguration**, **Ressource**, **Protokolle**, **PROFIBUS DP Master**, **PROFIBUS DP Slave** öffnen.
2. Rechtsklick auf HIMax **PROFIBUS Slave** und **Eigenschaften** wählen.

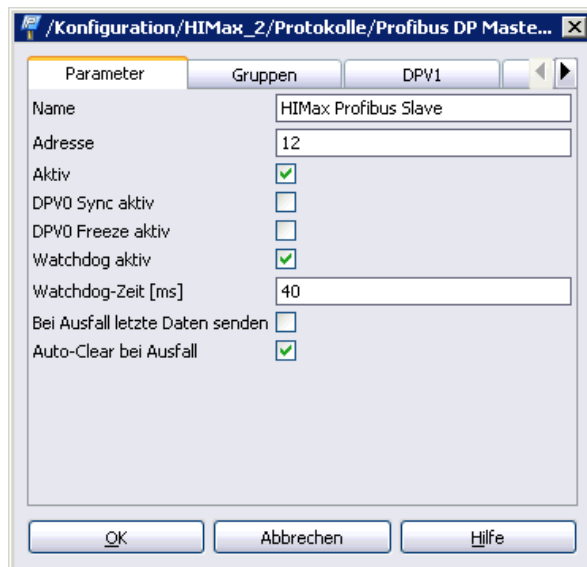


Bild 15: Eigenschaften des PROFIBUS DP Slave

3. Register **Parameter** wählen und im Kontrollkästchen **Watchdog aktiv** Häkchen setzen.
4. Watchdog-Zeit [ms] $\geq 6 \cdot T_{tr}$ [ms] im Feld **Watchdog-Zeit [ms]** eintragen.

i

Die Konfiguration des PROFIBUS DP Master und Slave muss mit dem Anwenderprogramm der PROFIBUS DP Master und Slave-Ressourcen neu compiliert und in die Steuerungen geladen werden, damit sie für die PROFIBUS DP Kommunikation wirksam werden.

4.3 Menüfunktionen des PROFIBUS DP Master

4.3.1 Edit

Die Menüfunktion **Edit** aus dem Kontextmenü des PROFIBUS DP Master öffnet den Dialog **Edit**.

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFIBUS DP Master im Anwenderprogramm auszuwerten.

Element	Beschreibung																
Anzahl Fehler	Anzahl Fehler seit Reset der Statistik.																
Baud-Rate	Baudrate (bit/s), mit welcher der Bus betrieben wird.																
Bus Fehler	<p>Tritt ein Busfehler auf, wird in der Systemvariable <i>Bus Fehler</i> ein Fehlercode gesetzt. Ein Fehlercode bleibt solange anliegen, bis der Busfehler behoben ist.</p> <table border="1"> <thead> <tr> <th>Code</th><th>Bedeutung</th></tr> </thead> <tbody> <tr> <td>0</td><td>OK, kein Busfehler.</td></tr> <tr> <td>1</td><td>Adressfehler: Adresse des Masters auf dem Bus bereits vorhanden.</td></tr> <tr> <td>2</td><td>Busstörung Störung auf dem Bus registriert, (z. B. Bus nicht richtig abgeschlossen, mehrere Teilnehmer senden gleichzeitig).</td></tr> <tr> <td>3</td><td>Protokollfehler Fehlerhaft codiertes Paket empfangen.</td></tr> <tr> <td>4</td><td>Hardwarefehler Hardware hat Fehler gemeldet, z. B. bei zu kurz eingestellten Zeiten.</td></tr> <tr> <td>5</td><td>Unbekannter Fehler Master hat Zustand aus unbekanntem Grund gewechselt.</td></tr> <tr> <td>6</td><td>Controller Reset Controller-Chip wird bei schweren Busstörungen zurückgesetzt.</td></tr> </tbody> </table> <p>Um die Statusvariable <i>Bus Fehler</i> im Anwenderprogramm auszuwerten, muss diese mit einer Variablen verbunden werden.</p>	Code	Bedeutung	0	OK, kein Busfehler.	1	Adressfehler: Adresse des Masters auf dem Bus bereits vorhanden.	2	Busstörung Störung auf dem Bus registriert, (z. B. Bus nicht richtig abgeschlossen, mehrere Teilnehmer senden gleichzeitig).	3	Protokollfehler Fehlerhaft codiertes Paket empfangen.	4	Hardwarefehler Hardware hat Fehler gemeldet, z. B. bei zu kurz eingestellten Zeiten.	5	Unbekannter Fehler Master hat Zustand aus unbekanntem Grund gewechselt.	6	Controller Reset Controller-Chip wird bei schweren Busstörungen zurückgesetzt.
Code	Bedeutung																
0	OK, kein Busfehler.																
1	Adressfehler: Adresse des Masters auf dem Bus bereits vorhanden.																
2	Busstörung Störung auf dem Bus registriert, (z. B. Bus nicht richtig abgeschlossen, mehrere Teilnehmer senden gleichzeitig).																
3	Protokollfehler Fehlerhaft codiertes Paket empfangen.																
4	Hardwarefehler Hardware hat Fehler gemeldet, z. B. bei zu kurz eingestellten Zeiten.																
5	Unbekannter Fehler Master hat Zustand aus unbekanntem Grund gewechselt.																
6	Controller Reset Controller-Chip wird bei schweren Busstörungen zurückgesetzt.																
Durchschnittliche Zykluszeit	Gemessene durchschnittliche Buszykluszeit in Millisekunden.																
Letzte Zykluszeit	Gemessene Buszykluszeit in Millisekunden.																
Master-Status	<p>Zeigt den momentanen Protokollzustand an.</p> <p>0: OFFLINE 1: STOP 2: CLEAR 3: OPERATE</p> <p>Um die Statusvariable <i>Master-Status</i> im Anwenderprogramm auszuwerten, muss dieses mit einer Variablen verbunden werden.</p>																
Maximale Zykluszeit	Gemessene maximale Buszykluszeit in Millisekunden.																
Min. Slave-Intervall	Gemessenes minimales Slave-Intervall eines diesem Master zugeordneten Slaves.																
Minimale Zykluszeit	Gemessene minimale Buszykluszeit in Millisekunden.																
Target rotation time	Projektierte Token-Umlaufzeit.																

Tabelle 39: Systemvariablen des PROFIBUS DP Master

4.3.2 Menüfunktion Eigenschaften

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des PROFIBUS DP Master öffnet den Dialog **Eigenschaften**.

Das Dialogfenster enthält die folgenden Register:

4.3.2.1 Register Allgemein

Element	Beschreibung
Typ	PROFIBUS DP Master.
Name	Beliebiger, eindeutiger Name, für einen PROFIBUS DP Master.
Modul	Auswahl des COM-Moduls auf dem dieses Protokoll abgearbeitet wird.
Max. μ P-Budget aktivieren	Aktiviert: Limit des μ P-Budgets aus dem Feld <i>Max. μP-Budget in [%]</i> übernehmen. Deaktiviert: Kein Limit des μ P-Budgets für dieses Protokoll verwenden.
Max. μ P-Budget in [%]	Maximales μ P-Budget des Moduls, welches bei der Abarbeitung des Protokolls produziert werden darf. Wertebereich: 1 ... 100% Standardwert: 30%
Verhalten bei CPU/COM Verbindungsverlust	Bei Verbindungsverlust des Prozessormoduls zum Kommunikationsmodul werden in Abhängigkeit dieses Parameters die Eingangsvariablen entweder initialisiert oder unverändert im Prozessormodul verwendet. (z. B. wenn Kommunikationsmodul bei laufender Kommunikation gezogen wird). Soll ein Projekt von kleiner SILworX V3 konvertiert werden, muss dieser Wert auf "Letzten Wert beibehalten" gesetzt sein, um den CRC nicht zu ändern. Für HIMatrix Steuerungen kleiner CPU BS V8 muss dieser Wert immer auf "Letzten Wert beibehalten" gesetzt sein. Initialdaten annehmen Eingangsvariablen werden auf die Initialwerte zurückgesetzt. Letzten Wert beibehalten Eingangsvariablen behalten den letzten Wert.
Adresse	Stationsadresse des Masters. Die Stationsadresse des Masters darf auf dem Bus nur einmal vorhanden sein. Wertebereich: 0 ... 125 Standardwert: 0
Schnittstelle	COM-Schnittstelle, die für den Master benutzt werden soll. Wertebereich: FB1, FB2

Element	Beschreibung																																												
Baudrate	<p>Baudrate (Bit/s), mit welcher der Bus betrieben wird.</p> <p>Mögliche Werte:</p> <table><tr><th>Wert</th><th>Baudrate</th><th>FB1</th><th>FB2</th></tr><tr><td>9600</td><td>9,6 kbit/s</td><td>X</td><td>X</td></tr><tr><td>19200</td><td>19,2 kbit/s</td><td>X</td><td>X</td></tr><tr><td>45450</td><td>45,45 kbit/s</td><td>X</td><td>X</td></tr><tr><td>93750</td><td>93,75 kbit/s</td><td>X</td><td>X</td></tr><tr><td>187500</td><td>187,5 kbit/s</td><td>X</td><td>X</td></tr><tr><td>500000</td><td>500 kbit/s</td><td>X</td><td>X</td></tr><tr><td>1500000</td><td>1,5 Mbit/s</td><td>X</td><td>X</td></tr><tr><td>3000000</td><td>3 Mbit/s</td><td>X</td><td>-</td></tr><tr><td>6000000</td><td>6 Mbit/s</td><td>X</td><td>-</td></tr><tr><td>12000000</td><td>12 Mbit/s</td><td>X</td><td>-</td></tr></table> <p>Standardwert: 9,6 kbit/s</p>	Wert	Baudrate	FB1	FB2	9600	9,6 kbit/s	X	X	19200	19,2 kbit/s	X	X	45450	45,45 kbit/s	X	X	93750	93,75 kbit/s	X	X	187500	187,5 kbit/s	X	X	500000	500 kbit/s	X	X	1500000	1,5 Mbit/s	X	X	3000000	3 Mbit/s	X	-	6000000	6 Mbit/s	X	-	12000000	12 Mbit/s	X	-
Wert	Baudrate	FB1	FB2																																										
9600	9,6 kbit/s	X	X																																										
19200	19,2 kbit/s	X	X																																										
45450	45,45 kbit/s	X	X																																										
93750	93,75 kbit/s	X	X																																										
187500	187,5 kbit/s	X	X																																										
500000	500 kbit/s	X	X																																										
1500000	1,5 Mbit/s	X	X																																										
3000000	3 Mbit/s	X	-																																										
6000000	6 Mbit/s	X	-																																										
12000000	12 Mbit/s	X	-																																										

Tabelle 40: Allgemeine Eigenschaften des PROFIBUS DP Master

4.3.2.2 Register Zeiten

Element	Beschreibung
MinTsdr [bit time]	Min. Station Delay Time Zeit, die ein PROFIBUS DP Slave mindestens warten muss, bevor er antworten darf. Wertebereich: 11 ... 1023 Standardwert: 11
MaxTsdr [bit time]	Max. Station Delay Time Zeit, die ein PROFIBUS DP Slave maximal benötigen darf, um zu antworten. Max Tsdr \geq Tsdr (des angeschlossenen Slaves mit der höchsten Tsdr). Die MaxTsdr-Werte der Slaves werden aus den GSD-Dateien gelesen und befinden sich im Dialog Eigenschaften der Slaves im Register Übertragungsraten . Wertebereich: 37 ... 65535 Standardwert: 100
Tsl [bit time]	Slot Time Maximale Zeitspanne, in welcher der Master auf eine Antwort des Slaves wartet. $Tsl > MaxTsdr + 2 \cdot Tset + Tqui + 13$ Wertebereich: 37 ... 16383 Standardwert: 200
Tqui [bit time]	Quiet Time for Modulator (Modulatorausklingzeit) Zeit, die ein Teilnehmer für das Umschalten von Senden auf Empfangen benötigen darf. Wertebereich: 0 ... 493 Standardwert: 0
Tset [bit time]	Setup Time Reaktionszeit auf ein Ereignis. Wertebereich: 1 ... 494 Standardwert: 1
Ttr [bit time]	Projektierte Zeit für einen Token-Umlauf Die maximale zur Verfügung gestellte Zeit für einen Token-Umlauf. Eine untere Abschätzung der Ttr erhält man durch eine Berechnung, siehe Kapitel 4.4.4. Wertebereich: 256 ... 16777215 Standardwert: 9999
Ttr [ms]	Tatsächliche Token-Umlaufzeit in ms
Min. Slave Intervall [ms]	Mindestzeit, die zwischen zwei zyklischen Abfragen eines Slaves verstreichen muss. Diese wird vom Master eingehalten und auf keinen Fall unterschritten. Der PROFIBUS DP Zyklus kann sich jedoch verlängern, wenn der Isochron Mode inaktiv ist und der Anteil der azyklischen Telegramme in einem Zyklus ansteigt. Der Wert für <i>Min. Slave Intervall</i> des Slaves wird aus der GSD-Datei gelesen und befindet sich im Dialog Eigenschaften des Slaves im Register Features . Im Isochron Mode gibt <i>Min. Slave Intervall</i> die Zeit des isochronen Zyklus vor. Der Isochron Mode wird aktiviert wenn Isochron Mode Sync oder Freeze aktiviert sind. Siehe auch Aktualisierungszeit zwischen CPU und COM (Register CPU/COM). Wertebereich: 0 ... 6553.5 (Schrittweite 0.1 ms) Standardwert: 1.0

Element	Beschreibung
Nutzdaten- überwachungszeit [ms]	Zeitspanne, innerhalb welcher der Master seinen momentanen Zustand auf dem Bus mitteilen muss. Richtwert: Nutzdatenüberwachungszeit = WDZ des Slaves Wertebereich: 0 ... 655350 (Schrittweite 10 ms) Standardwert: 2000

Tabelle 41: Register Zeiten im Eigenschaftendialog des PROFIBUS DP Master

4.3.2.3 Register CPU/COM

Die Vorgabewerte für die Parameter sorgen für den schnellstmöglichen Datenaustausch der PROFIBUS DP Daten zwischen dem COM-Modul (COM) und dem CPU-Modul (CPU) in der HIMax Steuerung. Diese Parameter sollten nur dann geändert werden, wenn eine Reduzierung der COM- und/oder CPU-Auslastung für eine Anwendung erforderlich ist und der Prozess dies zulässt.



Die Änderung der Parameter wird nur für den erfahrenen Programmierer empfohlen. Eine Erhöhung der COM- und CPU-Aktualisierungszeit bedeutet auch, dass die tatsächliche Aktualisierungszeit der PROFIBUS DP Daten erhöht wird. Zeitanforderungen der Anlage prüfen.

Den Parameter *Min. Slave Intervall [ms]* beachten, der die Aktualisierungszeit der PROFIBUS DP Daten vom/zum PROFIBUS DP Slave festlegt. Dieser kann entsprechend der COM/CPU-Aktualisierungszeit erhöht werden.

Element	Beschreibung
Aktualisierungs- intervall der Prozessdaten [ms]	Aktualisierungszeit in Millisekunden, mit der die Daten des Protokolls zwischen COM und CPU ausgetauscht werden. Ist die <i>Refresh Rate</i> Null oder kleiner als die Zykluszeit der Steuerung, dann erfolgt der Datenaustausch so schnell wie möglich. Wertebereich: 0 ... ($2^{31}-1$) Standardwert: 0
Prozessdaten- Konsistenz erzwingen	Aktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM innerhalb eines Zyklus der CPU. Deaktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM, verteilt über mehrere CPU Zyklen zu je 1100 Byte pro Datenrichtung. Damit kann eventuell auch die Zykluszeit der Steuerung reduziert werden. Standardwert: Aktiviert

Tabelle 42: Register CPU/COM im Eigenschaftendialog des PROFIBUS DP Master

4.3.2.4 Register Sonstige

Element	Beschreibung
Max. Anz. Sendewdh.	Maximale Anzahl an Sendewiederholungen eines Masters, wenn ein Slave nicht antwortet. Wertebereich: 0 ... 7 Standardwert: 1
Highest Active Address	Highest Station Address (HSA) Höchste zu erwartende Stationsadresse eines Masters. Master, mit Stationsadressen jenseits von HSA, werden nicht in den Token-Ring aufgenommen. Wertebereich: 0 ... 125 Standardwert: 125
Isochron Mode Sync	Der Isochron Mode Sync ermöglicht eine takt synchrone Regelung in Master und Slave und ein zeitgleiches Aktivieren der physikalischen Ausgänge mehrerer Slaves. Ist der Isochron Mode Sync aktiv, dann sendet der Master den Steuerbefehl „Sync“ als Broadcast-Telegramm an alle Slaves. Sobald Slaves, die den Isochron Mode unterstützen, den Steuerbefehl „Sync“ erhalten, schalten sie die Daten aus dem Anwenderprogramm zeitgleich auf die physikalischen Ausgänge. Die Werte der physikalischen Ausgänge bleiben bis zum nächsten Sync-Befehl eingefroren. Die Zykluszeit wird durch das „Min. Slave Intervall“ vorgegeben. Bedingung: $T_{tr} < \text{Min. Slave Intervall}$ Standardwert: Deaktiviert
Isochron Mode Freeze	Der Isochron Mode Freeze ermöglicht eine zeitgleiche Übernahme der Eingangsdaten mehrerer Slaves. Ist der Isochron Mode Freeze aktiv, sendet der Master den Steuerbefehl „Freeze“ als Broadcast-Telegramm an alle Slaves. Sobald die Slaves, die den Isochron Mode unterstützen, den Steuerbefehl „Freeze“ erhalten, werden die Variablen der physikalischen Eingänge auf dem momentanen Wert eingefroren. Die Werte können dann vom Master gelesen werden. Erst nach einem weiteren Steuerbefehl „Freeze“ werden die Eingangsdaten aktualisiert. Die Zykluszeit wird durch das „Min. Slave Intervall“ vorgegeben. Bedingung: $T_{tr} < \text{Min. Slave Intervall}$ Standardwert: Deaktiviert
Auto-Clear bei Fehler	Der Master geht in den Zustand CLEAR, wenn ein Slave ausfällt bei dem <i>Auto-Clear bei Ausfall</i> gesetzt ist. Standardwert: Deaktiviert
Zeitmaster	Der Master ist auch Zeitmaster und versendet die Systemzeit periodisch über den Bus. Standardwert: Deaktiviert
Clock Sync Interval [ms]	Uhr-Synchronisations-Intervall. Zeitspanne, innerhalb welcher der Zeitmaster die Systemzeit auf dem Bus versendet. Wertebereich: 0 ... 655350 (Schrittweite 10 ms) Standardwert: 0 (kein Zeitmaster)

Tabelle 43: Sonstige Eigenschaften des PROFIBUS DP Master

4.4 Die PROFIBUS DP Buszugriffsverfahren

Das Buszugriffsverfahren stellt jedem Busteilnehmer ein definiertes Zeitfenster zur Verfügung, in dem der Busteilnehmer seine Kommunikationsaufgabe erfüllen muss.

4.4.1 Master/Slave-Protokoll

Die Buszuteilung zwischen einem PROFIBUS DP Master und einem PROFIBUS DP Slave wird über das Master/Slave-Verfahren sichergestellt.

Ein aktiver PROFIBUS DP Master kommuniziert mit passiven PROFIBUS DP Slaves.

Der PROFIBUS DP Master, der das Token besitzt, hat die Sendeberechtigung und kann mit dem ihm zugewiesenen PROFIBUS DP Slaves kommunizieren. Der Master teilt einem Slave den Bus für bestimmte Zeit zu, innerhalb welcher der Slave antworten muss.

4.4.2 Token-Protokoll

Die Buszuteilung zwischen Automatisierungsgeräten (Master Klasse 1) und/oder Programmiergeräten (Master Klasse 2) wird über Token-Passing sichergestellt.

Alle PROFIBUS DP Master, die gemeinsam an einem Bus angeschlossen sind, bilden einen Token-Ring. Der aktive PROFIBUS DP Master, der im Besitz des Tokens ist, übernimmt in dieser Zeit die Masterfunktion am Bus.

Die PROFIBUS DP Master werden im Token-Ring nach aufsteigenden Stationsadressen geordnet und das Token wird in dieser Reihenfolge bis zum PROFIBUS DP Master mit der höchsten Stationsadresse weitergegeben.

Dieser gibt das Token an den Master mit der niedrigsten Stationsadresse weiter, um den Token-Ring zu schließen.

Die Token-Umlaufzeit entspricht dem einmaligen Umlauf des Token über alle PROFIBUS DP Master. Die Token-Umlaufzeit T_{tr} ist die maximal erlaubte Zeit für einen Token-Umlauf.

4.4.3 Token-Umlaufzeit (T_{tr})

Richtwerte für verschiedene Übertragungsraten.

Bei der Konfiguration des PROFIBUS DP Master beachten, dass ein Teil der Parameter im Register **Zeiten** von der im Register **Allgemein** eingestellten Baudrate abhängt. Für die erste (initial) Konfiguration die in der folgenden Tabelle angegebenen Richtwerte verwenden. In einem späteren Schritt werden die Werte optimiert.

	9,6k	19,2k	45,45k	93,75k	187,5k	500k	1,5M	3M	6M	12M
MinTsdr	11	11	11	11	11	11	11	11	11	11
MaxTsdr	60	60	400	60	60	100	150	250	450	800
Tsl bit time	100	100	640	100	100	200	300	400	600	1000
Tqui bit time	0	0	0	0	0	0	0	3	6	9
Tset bit time	1	1	95	1	1	1	1	4	8	16

Tabelle 44: Richtwerte für Token-Umlaufzeit bei verschiedenen Übertragungsraten

Alle Zeitangaben sind in Tbit angegeben ($1\text{Tbit} = 1/[\text{bit/s}]$).

MinTsdr ist mindestens 11 Tbit lang, da ein Zeichen aus 11 Bits (1 Startbit, 1 Stoppbit, 1 Paritätsbit, 8 Datenbits) besteht.

Übertragungszeit für ein Zeichen

Baudrate	Tbit bit = $1/\text{Baudrate}$	Zeit
9600 bit/s	$1 / 9600 = 104,166 \mu\text{s}$	$11 \cdot 104,166 \mu\text{s} = 114,583 \text{ ms}$
6 Mbit/s	$1 / 6 \cdot 10^6 = 166,667 \text{ ns}$	$11 \cdot 166,667 \text{ ns} = 1,833 \mu\text{s}$

Tabelle 45: Übertragungszeit für ein Zeichen bei verschiedenen Übertragungsraten

4.4.4 Token-Umlaufzeit T_{tr} berechnen

Minimale Token-Umlaufzeit T_{tr} wie folgt berechnen:

$$T_{tr_{min}} = n * (198 + T_1 + T_2) + b * 11 + 242 + T_1 + T_2 + T_{sl}$$

Element	Bedeutung
n	Anzahl aktiver Slaves.
b	Anzahl E/A-Datenbytes der aktiven Slaves (Input plus Output).
T ₀	$35 + 2 * T_{set} + T_{qui}$
T ₁	Wenn $T_0 < MinT_{sdr}$: $T_1 = MinT_{sdr}$ Wenn $T_0 > MinT_{sdr}$: $T_1 = T_0$
T ₂	Wenn $T_0 < MaxT_{sdr}$: $T_2 = MaxT_{sdr}$ Wenn $T_0 > MaxT_{sdr}$: $T_2 = T_0$
T _{sl}	Slot Time Maximale Zeitspanne, in welcher der Master auf eine Antwort des Slaves wartet.
198	Zwei mal Telegrammkopf des Telegramms mit variabler Länge (für Request und Response).
242	Global_Control, FDL_Status_Req und Token-Weitergabe.

Tabelle 46: Elemente zur Berechnung der Token-Umlaufzeit

i

Diese Abschätzung der Token-Umlaufzeit T_{tr} gilt nur, wenn nur ein Master am Bus betrieben wird, keine Sendungen wiederholt werden müssen und keine azyklischen Daten übertragen werden.

Auf keinen Fall eine kleinere T_{tr} einstellen als mit obiger Formel berechnet, da sonst eine fehlerfreie Funktion nicht mehr garantiert werden kann. HIMA empfiehlt, das Doppelte oder Dreifache des berechneten Wertes einzustellen.

Beispiel zur Berechnung der Token-Umlaufzeit T_{tr}

Folgende Konfiguration ist gegeben:

5 aktive Slaves

(n = 5)

20 E/A-Datenbytes pro Slave

(b = 100)

Die folgenden Zeitkonstanten für eine Übertragungsrate von 6 Mbit/s wurden aus der Tabelle 46 entnommen:

- $MinT_{sdr} = 11 T_{bit}$
- $MaxT_{sdr} = 450 T_{bit}$
- $T_{sl} \text{ bit time} = 600 T_{bit}$
- $T_{qui} \text{ bit time} = 6 T_{bit}$
- $T_{set} \text{ bit time} = 8 T_{bit}$

$$T_0 = 35 + 2 * T_{set} + T_{qui}$$

$$T_0 = 35 + 2 * 8 + 6$$

$$\mathbf{T_0 = 57 T_{bit}}$$

Da $T_0 > MinT_{sdr}$: **$T_1 = T_0 = 57 T_{bit}$**

Da $T_0 < \text{MaxT}_{\text{sdr}}$: **$T_2 = \text{MaxT}_{\text{sdr}} = 450 T_{\text{bit}}$**

Die ermittelten Werte in die Formel für die minimale Token-Umlaufzeit einsetzen:

$$T_{\text{tr min}} = n * (198 + T_1 + T_2) + b * 11 + 242 + T_1 + T_2 + T_{\text{sl}}$$

$$T_{\text{tr min}} = 5 (198 + 57 + 450) + 100 * 11 + 242 + 57 + 450 + 600$$

$$T_{\text{tr min}} [T_{\text{bit}}] = 5974 T_{\text{bit}}$$

Ergebnis:

$$T_{\text{tr min}} [\mu\text{s}] = 5974 T_{\text{bit}} * 166,67 \text{ ns} = 995,68 \mu\text{s}$$

i

Ttr wird bei der Eingabe im Dialogfenster geprüft.

Ist der eingetragene Wert *Ttr* kleiner als der von SILworX errechnete Wert, erfolgt eine Fehlermeldung im Logbuch. Zusätzlich wird ein Mindestwert für *Ttr* vorgeschlagen.

Sind *Isochron Mode Sync* oder *Isochron Mode Freeze* ausgewählt, wird die Zykluszeit vom Parameter *MinSlaveInterval* vorgegeben. Die *Ttr* muss dann auf jeden Fall kleiner als das *Min. Slave Intervall* sein.

Das Nichteinhalten dieser Bedingung im Iso-Mode führt zu einer Fehlermeldung.

4.5 Isochroner PROFIBUS DP Zyklus (ab DP-V2)

Der PROFIBUS DP Zyklus besteht hier aus einem festen, zyklischen und einem ereignisbedingten, azyklischen Telegrammteil.

Der azyklische Telegrammteil in einem PROFIBUS DP Zyklus kann diesen entsprechend verlängern, was in bestimmten Anwendungen, wie z. B. in der Antriebstechnik, unerwünscht ist.

Um eine konstante Zykluszeit (t_{const}) zu erreichen, wird im Master der Isochron-Mode aktiviert, bei dem der Parameter *Min. Slave Intervall [ms]* die konstante Zykluszeit (t_{const}) vorgibt. Der so parametrisierte isochrone PROFIBUS DP Zyklus besitzt eine Taktgenauigkeit mit einer Abweichung von < 10 ms.

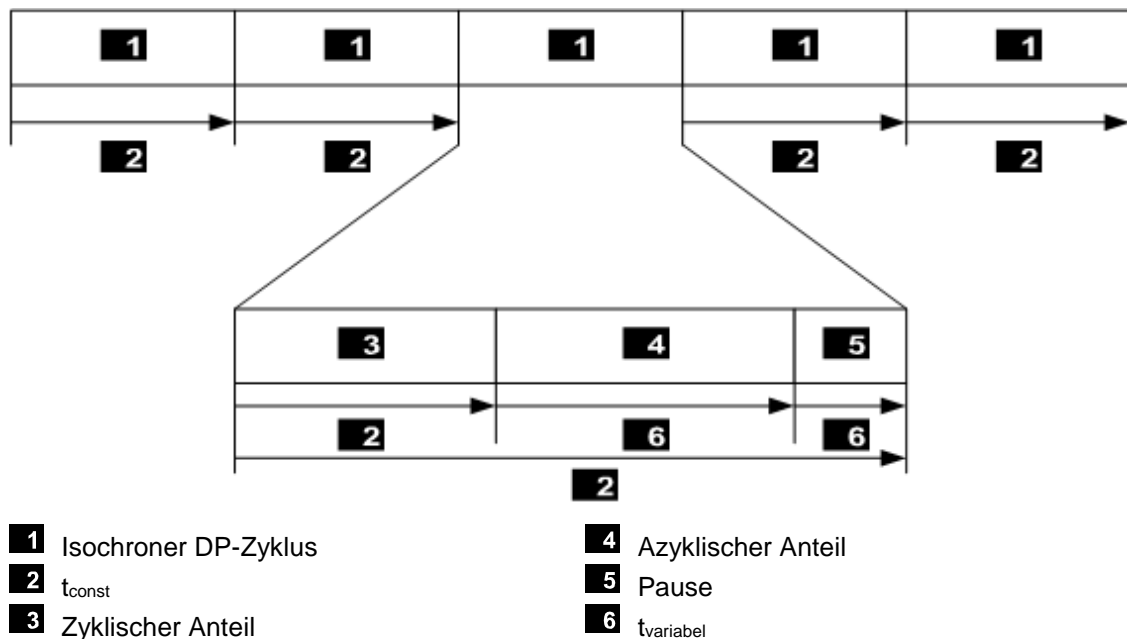


Bild 16: Isochroner PROFIBUS DP Zyklus

Um den zyklischen Anteil zu ermitteln, muss der Anwender die minimale Token-Umlaufzeit berechnen.

Zusätzlich muss ein ausreichend großes Zeitintervall (typischerweise zwei- bis dreimal minimale Token-Umlaufzeit T_{tr}) für den azyklischen Anteil reserviert werden. Wird die reservierte Zeit nicht benötigt, wird eine Pause vor dem nächsten Zyklus eingelegt, um die Zykluszeit konstant zu halten. (siehe auch Kapitel 4.4.3, Token-Umlaufzeit T_{tr}).

i

Der Master wird über *Min. Slave Intervall [ms]* mit der vom Anwender ermittelten DP-Zykluszeit parametrisiert.

Damit der *Isochron Mode* wirksam ist, muss mindestens einer der beiden Parameter *Isochron Mode Sync* oder *Isochron Mode Freeze* im Master aktiviert werden.

An dem Bus darf dann nur ein Master im isochronen Mode betrieben werden. Weitere Master sind nicht zulässig.

4.5.1 Isochron Mode (ab DP-V2)

Diese Funktion ermöglicht eine taktsynchrone Regelung in Master und Slave, unabhängig von der Belastung des Busses. Der Buszyklus wird mit einer Taktabweichungen von < 10 ms synchronisiert. Damit können hochgenaue Positionierungsvorgänge realisiert werden.

i

Die Vorteile des *Isochron-Mode* können auch eingeschränkt von Slaves (DP-V0-Slaves) genutzt werden, die den *Isochron-Mode* nicht unterstützen. Man aktiviert dazu bei den *Slaves Sync* und/oder *Freeze* und ordnet sie der Gruppe 8 zu.

Typischerweise verwendet man den *Sync*- und *Freeze* -Mode gleichzeitig.

4.5.2 Isochron Mode Sync (ab DP-V2)

Der *Isochron Mode Sync* ermöglicht eine taktsynchrone Regelung in Master und Slave, und ein zeitgleiches aktivieren der Ausgänge mehrerer Slaves.

4.5.3 Isochrone Mode Freeze (ab DP-V2)

Der *Isochron Mode Freeze* ermöglicht eine zeitgleiche Übernahme der Eingangsdaten mehrerer Slaves.

4.6 Menüfunktionen PROFIBUS DP Slave (im Master)

4.6.1 Anlegen eines PROFIBUS DP Slave (im Master)

Im HIMA PROFIBUS DP Master einen PROFIBUS DP Slave anlegen

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, Profibus DP Master** öffnen.
2. Im Kontextmenü von PROFIBUS DP Master **Neu, PROFIBUS Slave** wählen, um einen neuen PROFIBUS Slave hinzuzufügen.

4.6.2 Edit

Die Menüfunktion **Edit** aus dem Kontextmenü des PROFIBUS DP Slave öffnet den Dialog **Systemvariablen**.

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFIBUS DP Slave im Anwenderprogramm auszuwerten.

Element	Beschreibung	
Activation control	Wechsel von 0 nach 1 deaktiviert den Slave. Wechsel von 1 nach 0 aktiviert einen zuvor deaktivierten Slave. Aktiviert = 0 Deaktiviert = 1	
PNO Ident Nummer	Von der PROFIBUS DP Nutzerorganisation e.V. zugeteilte 16-Bit-Nummer, die ein Produkt (Feldgerät) eindeutig kennzeichnet.	
Standard-Diagnose	Über die Standard-Diagnose teilt der Slave dem Master seinen aktuellen Zustand mit. Diese Variable enthält immer die zuletzt empfangene Standard-Diagnose. Die Parameter entsprechen dem Diagnosetelegramm gemäß IEC 61158.	
Verbindungszähler	Wird mit jeder neuen Verbindung inkrementiert. Zählt ab Reset des Zählers	
Verbindungszustand	Wert	Beschreibung
	0	Deaktiviert: Die Parametersätze für diese Slaves werden geladen, die Slaves jedoch vollkommen ignoriert. Die Input-Daten werden auf die Initialwerte gesetzt, auf dem Bus ist keinerlei Aktivität bzgl. dieser Slaves sichtbar.
	1	Inaktiv (nicht verbunden): Ist ein Slave nicht (mehr) erreichbar, werden die Input-Daten auf die Initialwerte gesetzt. Für jeden Slave können die folgenden Optionen gewählt werden: <ul style="list-style-type: none"> ▪ Master sendet weiterhin Output-Daten zum Slave oder. ▪ Master versucht, den Slave neu zu parametrieren.
	2	Aktiv (verbunden): Slaves tauschen E/A-Daten mit der CPU aus.
Zähler Slave-Alarme	Anzahl bisher gelieferter Alarme. Zählt ab Reset des Zählers.	
Zähler Standard-Diagnose	Anzahl bisher gelieferter Diagnosemeldungen. Zählt ab Reset des Zählers.	

Tabelle 47: Systemvariablen des PROFIBUS DP Slave

4.6.3 Eigenschaften

Die Menüfunktion **Eigenschaften** aus dem Kontextmenü des PROFIBUS DP Slave öffnet den Dialog **Eigenschaften**. Das Dialogfenster enthält die folgenden Register:

4.6.3.1 Register Parameter

Element	Beschreibung
Name	Name des Slave
Adresse	Adresse des Slave Wertebereich: 0 ... 125 Standardwert: 0
Aktiv	Zustand des Slave Nur ein aktiver Slave kann mit einem PROFIBUS DP Master kommunizieren. Standardwert: Aktiviert
DPV0 Sync aktiv	Der <i>Sync Mode</i> ermöglicht ein zeitgleiches Aktivieren der Ausgänge mehrerer DP-V0-Slaves. Achtung: Bei DP-V2-Slaves, die im <i>Isochron Mode Sync</i> arbeiten, muss dieses Feld deaktiviert sein. Standardwert: Deaktiviert
DPV0 Freeze aktiv	Der <i>Freeze Mode</i> ermöglicht eine zeitgleiche Übernahme der Eingangsdaten mehrerer DP-V0-Slaves. Achtung: Bei DP-V2-Slaves, die im <i>Isochron Mode Freeze</i> arbeiten, muss dieses Feld deaktiviert sein. Standardwert: Deaktiviert
Watchdog aktiv	Wenn ausgewählt, erkennt der Slave den Ausfall eines Masters und geht in einen sicheren Zustand. Standardwert: Deaktiviert
Watchdog-Zeit [ms]	Kontrollkästchen Watchdog aktiv muss aktiviert sein. Wenn innerhalb dieser Zeitspanne kein Datenaustausch zwischen Master und Slave stattgefunden hat, schaltet sich der Slave ab und setzt alle DP-Output Daten auf ihren Initialwert zurück. 0 = Deaktiviert Richtwert: Watchdog-Zeit des Slaves > 6 * Ttr Wertebereich: 0 ... 65535 Standardwert: 0
Bei Ausfall letzte Daten senden	Aktiviert: Sendet im Fehlerfall weiter Daten auch ohne Bestätigung des Slaves. Deaktiviert: Verbindung wird im Fehlerfall abgebaut und neu aufgebaut. Standardwert: Deaktiviert
Auto-Clear bei Ausfall	Parameter ist nur wirksam, wenn auch im Master <i>Auto-Clear bei Fehler</i> aktiviert ist! Aktiviert: Auto-Clear Funktion für diesen Slave aktiviert. Der Master wechselt automatisch vom Zustand OPERATE nach CLEAR, wenn mit einem Auto-Clear Slave kein Datenaustausch mehr möglich ist. Sobald alle Auto-Clear Slaves wieder aktiv sind, wechselt der Master automatisch in den Zustand OPERATE. Deaktiviert: Auto-Clear Funktion für diesen Slave deaktiviert. Standardwert: Aktiviert

Tabelle 48: Register Parameter des PROFIBUS DP Slave

4.6.3.2 Register Gruppen

In diesem Register können die Slaves in verschiedenen Gruppen organisiert werden. Die GlobalControl-Kommandos *Sync* und *Freeze* können dann gezielt eine oder mehrere Gruppen ansprechen.

Element	Beschreibung	
Mitglied in Gruppe 1	Mitglied in Gruppe 1	Standardwert: Deaktiviert
Mitglied in Gruppe 2	Mitglied in Gruppe 2	
Mitglied in Gruppe 3	Mitglied in Gruppe 3	
Mitglied in Gruppe 4	Mitglied in Gruppe 4	
Mitglied in Gruppe 5	Mitglied in Gruppe 5	
Mitglied in Gruppe 6	Mitglied in Gruppe 6	
Mitglied in Gruppe 7	Mitglied in Gruppe 7	
Mitglied in Gruppe 8	Mitglied in Gruppe 8	

Tabelle 49: Register Gruppen im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.3 Register DP-V1

In diesem Register befinden sich Parameter, die erst ab DP-V1 definiert sind. Bei DP-V0-Slaves kann hier nichts ausgewählt werden. Welche Parameter von dem Slave unterstützt werden, erkennt man in der Spalte **Unterstützt**.

Element	Beschreibung
DP-V1	Wenn der DP-V1-Modus nicht aktiviert ist, können auch die anderen DP-V1-Features nicht genutzt werden. Der Slave verhält sich dann wie ein DP-V0-Slave. Eventuell müssen dann auch die Parametrierdaten geändert werden (siehe Handbuch des Slaves). Standardwert: Deaktiviert
Failsafe	Wenn dieser Modus aktiviert ist, sendet der Master im Zustand CLEAR keine Nullen als Ausgangsdaten zum Slave, sondern ein leeres Datenpaket (Failsafe-Paket). Der Slave erkennt daran, dass er jetzt die sicheren Ausgangsdaten (die nicht notwendigerweise alle Null sind) auf die Ausgänge legen soll. Standardwert: Deaktiviert
Isochron Modus	Diese Funktion ermöglicht eine taktsynchrone Regelung in Master und Slave unabhängig von der Belastung des Busses. Der Buszyklus wird mit einer Taktabweichung von < 1ms synchronisiert. Damit können hochgenaue Positionierungsvorgänge realisiert werden. Standardwert: Deaktiviert
Publisher aktiv	Diese Funktion wird für den Slave-Querverkehr benötigt. Dies ermöglicht die direkte und zeitsparende Kommunikation zwischen den Slaves via Broadcast ohne Umwege über den Master. Standardwert: Deaktiviert
Prm Block Struct. Supp.	Der Slave unterstützt strukturierte Parametrierdaten (Nur Lesen). Standardwert: Deaktiviert
Check. Cfg.-Mode	Reduzierte Konfigurationskontrolle, wenn Check Cfg.-Mode aktiviert ist, dann kann der Slave ohne die komplette Konfiguration betrieben werden. Für die Inbetriebnahme sollte dieses Feld deaktiviert werden. Standardwert: Deaktiviert

Tabelle 50: Register DP-V1 im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.4 Register Alarme

Auf dieser Seite können Alarme aktiviert werden. Das geht jedoch nur bei DP-V1-Slaves, wenn DP-V1 aktiviert ist und der Slave Alarme unterstützt. Welche Alarme unterstützt werden, erkennt man an dem Häkchen in der Spalte **Unterstützt**. Wird ein Alarm vorgeschrieben, erkennt man dies in der Spalte **Verlangt**.

Element	Beschreibung	
Update-Alarm	Alarm, wenn Parameter eines Moduls geändert.	Standardwert: Deaktiviert
Status-Alarm	Alarm, wenn Zustand eines Moduls geändert.	
Vendor-Alarm	Herstellerspezifischer Alarm.	
Diagnose-Alarm	Alarm bei bestimmten Ereignissen wie Kurzschluss, Übertemperatur etc. an einem Modul.	
Prozess-Alarm	Alarm bei wichtigen Ereignissen im Prozess.	
Stecken/Ziehen-Alarm	Alarm bei Ziehen oder Stecken eines Moduls.	

Tabelle 51: Register Alarme im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.5 Register Daten

In diesem Register befinden sich Informationen über die unterstützten Datenlängen, sowie über die Benutzerdaten (erweiterte Parametrierdaten).

Element	Beschreibung
Max. Input Len	Maximale Länge der Eingangsdaten.
Max. Output Len	Maximale Länge der Ausgangsdaten.
Max. Data Len	Maximale Gesamtlänge der Ein- und Ausgangsdaten.
Benutzerdatengröße	Länge der Benutzerdaten.
Benutzerdaten	Parametrierdaten. Das Editieren empfiehlt sich nicht hier durchzuführen. Komfortabler geht es mit dem Dialog <i>User Parameter bearbeiten</i> Kapitel 4.8.
Max. Diag. Data Len	Maximale Länge der Diagnosedaten, die der Slave sendet.

Tabelle 52: Register Daten im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.6 Register Modell

Auf dieser Seite befinden sich verschiedene Informationen, die selbsterklärend sind:

Element	Beschreibung
Modell	Herstellerbezeichnung des PROFIBUS DP Slave.
Hersteller	Hersteller des Feldgerätes.
Ident. Nummer	Slave-Kennung der PROFIBUS Nutzerorganisation (PNO).
Revision	Ausgabestand des PROFIBUS DP Slave.
Hardware-Release	Hardware-Ausgabestand des PROFIBUS DP Slave.
Software-Release	Software-Ausgabestand des PROFIBUS DP Slave.
GSD-Dateiname	Dateiname der GSD-Datei.
Infotext	Zusätzliche Info zum PROFIBUS DP Slave.

Tabelle 53: Register Modell im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.7 Register Features

Element	Beschreibung
Modularstation	TRUE: Modularstation FALSE: Kompaktstation
Erste parametrierbare Slotnummer	Nummerierung der Module (Slots) muss mit dieser Nummer beginnen und fortlaufend erfolgen.
Max. Module	Maximale Anzahl an Modulen, die eine modulare Station aufnehmen kann.
Unterstützung für Set Slave Add	Slave unterstützt dynamische Adressvergabe.
Min. Slave Intervall [ms]	Die Mindestzeit, die zwischen zwei zyklischen Aufrufen des Slaves verstreichen muss.
Diag. Update	Anzahl der Pollzyklen, bis die Diagnose des Slaves den aktuellen Zustand widerspiegelt.
Unterstützung für WDBase1ms	Slave unterstützt 1ms als Zeitbasis für die Watchdog.
Unterstützung für DPV0 Sync	Slave unterstützt DP-V0 Sync.
Unterstützung für DPV0 Freeze	Slave unterstützt DP-V0 Freeze.
DPV1 Datentypen	Slave unterstützt die DP-V1-Datentypen.
Extra Alarm SAP	Slave unterstützt SAP 50 zur Alarmbestätigung.
Anzahl paralleler, aktiver Alarmer	Gibt an, wie viele aktive Alarmer der Slave gleichzeitig bearbeiten kann. Null bedeutet ein Alarm von jedem Typ.

Tabelle 54: Register Features im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.8 Register Übertragungsraten

In diesem Register befinden sich die Übertragungsraten, die der Slave unterstützt, sowie jeweils die zugehörige *MaxTsdr*.

MaxTsdr ist die Zeit, innerhalb welcher der Slave spätestens auf eine Anforderung vom Master antworten muss. Der Wertebereich ist abhängig vom Slave und der Übertragungsgeschwindigkeit und liegt zwischen 15 und 800 Tbit.

Element	Beschreibung
9,6k	MaxTsdr = 60
19,2k	MaxTsdr = 60
31,25k	Wird nicht unterstützt.
45,45k	MaxTsdr = 60
93,75k	MaxTsdr = 60
187,5k	MaxTsdr = 60
500k	MaxTsdr = 70
1,5M	MaxTsdr = 75
3M	MaxTsdr = 90
6M	MaxTsdr = 100
12M	MaxTsdr = 120

Tabelle 55: Register Übertragungsraten im Eigenschaftendialog des PROFIBUS DP Slave

4.6.3.9 Register Azyklisch

In diesem Register befinden sich einige Parameter für die azyklische Datenübertragung:

Element	Beschreibung
C1 Read/Write-Unterstützung	Slave unterstützt azyklische Datenübertragung.
C1 Read/Write-notwendig	Slave erfordert azyklische Datenübertragung.
C1 Datengröße [Byte]	Maximale Länge eines azyklischen Datenpaketes.
C1 Response Timeout [ms]	Timeout für azyklische Datenübertragung.

Tabelle 56: Register Azyklisch im Eigenschaftendialog des PROFIBUS DP Slave

4.7 GSD-Datei einlesen

Die GSD-Datei enthält die Daten für die Parametrierung des PROFIBUS DP Slave.

GSD-Datei für den neuen PROFIBUS DP Slave einlesen

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, PROFIBUS Slave** öffnen.
2. Im Kontextmenü vom PROFIBUS DP Master **GSD-Datei einlesen** wählen und die zum PROFIBUS Slave zugehörige GSD-Datei (z. B. hax100ea.gsd) wählen.

i

Die GSD-Dateien für HIMA Steuerungen sind auf der HIMA Webseite www.hima.com zu finden.

Für die Richtigkeit der GSD-Datei ist der Hersteller des jeweiligen Feldgerätes verantwortlich.

In den GSD-Dateien für HIMax (hax100ea.gsd) und HIMatrix (hix100ea.gsd) sind die folgenden Module enthalten:

PROFIBUS DP Master Eingangs Module	Typ	Anzahl
DP-Input/ELOP-Export	Byte	1
DP-Input/ELOP-Export	Bytes	2
DP-Input/ELOP-Export	Bytes	4
DP-Input/ELOP-Export	Bytes	8
DP-Input/ELOP-Export	Bytes	16
DP-Input/ELOP-Export	Word	1
DP-Input/ELOP-Export	Words	2
DP-Input/ELOP-Export	Words	4
DP-Input/ELOP-Export	Words	8
DP-Input/ELOP-Export	Words	16
PROFIBUS DP Master Ausgangs Module	Typ	Anzahl
DP-Output/ELOP-Import	Byte	1
DP-Output/ELOP-Import	Bytes	2
DP-Output/ELOP-Import	Bytes	4
DP-Output/ELOP-Import	Bytes	8
DP-Output/ELOP-Import	Bytes	16
DP-Output/ELOP-Import	Word	1
DP-Output/ELOP-Import	Words	2
DP-Output/ELOP-Import	Words	4
DP-Output/ELOP-Import	Words	8
DP-Output/ELOP-Import	Words	16

Tabelle 57: GSD-Datei des HIMA PROFIBUS DP Slave

4.8 User-Parameter bearbeiten

In dem Benutzerdatenfeld werden die **Startadresse** und die **Anzahl der Variablen** der Blöcke definiert.

Zusätzlich muss der PROFIBUS DP Master auch für die Anzahl der tatsächlich zu übertragenden Bytes konfiguriert werden. Dies geschieht durch Auswahl der PROFIBUS DP Module aus der GSD-Datei des PROFIBUS DP Slave (siehe auch Kapitel 4.2.2).

Öffnen des Dialogs User-Parameter bearbeiten

1. Im Strukturbaum **Konfiguration/Ressource/Protokolle PROFIBUS DP Master** öffnen.
2. Rechtsklick auf **PROFIBUS Slave** und **Eigenschaften** wählen.
3. Auf Register **Daten** klicken und **Schaltfläche...** neben den Benutzerdaten wählen.

i

Der Aufbau des Dialogs **User Parameter bearbeiten** wird von der GSD-Datei des Slaves festgelegt.

Aufbau des 32 Byte-Benutzerdatenfelds

Das 32 Byte-Benutzerdatenfeld ist wie folgt aufgebaut:

Die 32 Bytes sind in **8 Blöcke gruppiert**, mit jeweils **4 Bytes per Block**.

Die Blöcke 1 ... 4 definieren welche und wie viele Variablen der PROFIBUS DP Master vom PROFIBUS DP Slave empfängt.

Die Blöcke 5 ... 8 definieren welche und wie viele Variablen der PROFIBUS DP Master an den PROFIBUS DP Slave sendet.

Die ersten beiden Bytes eines jeden Blocks spezifizieren die Startadresse für die erste zu lesende oder zu schreibende Variable eines Blocks.

Die letzten beiden Bytes eines jeden Blocks spezifizieren die Anzahl der Variablen, die Empfangen oder gesendet werden sollen.

Konfiguration der Benutzerdaten in verschiedenen Blöcken

Normalerweise ist es nicht notwendig, die Variablen (Benutzerdaten) auf verschiedene Blöcke zu verteilen. Es ist ausreichend nur den jeweils ersten Variablenblock der Eingangs- und Ausgangsvariablen zu definieren und die Daten *en bloc* zu lesen und zu schreiben.

In Anwendungen, in denen es jedoch erforderlich ist, nur ausgewählte Variablen zu lesen und zu schreiben, können bis zu je vier Variablenblöcke für die Ausgangs- und Eingangsvariablen definiert werden.

Beispiel

Der PROFIBUS DP Master sendet und empfängt die folgenden Variablen vom PROFIBUS DP Slave:

1. Block: **4** Eingangsvariablen ab der Startadresse **0**.
2. Block: **6** Eingangsvariablen ab der Startadresse **50**.
4. Block: **9** Eingangsvariablen ab der Startadresse **100**.
5. Block: **2** Ausgangsvariablen ab der Startadresse **10**.

Konfiguration der Benutzerdaten im PROFIBUS DP Master

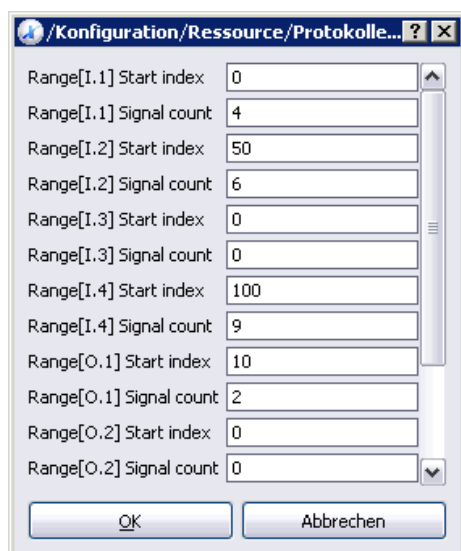
Master Import/Slave Export	Startadresse	Anz. Variable
1. Block (Byte 0 ... 3)	0,0	0,4
2. Block (Byte 4 ... 7)	0,50	0,6
3. Block (Byte 8 ... 11)	0,0	0,0
4. Block (Byte 12 ... 15)	0,100	0,9

Tabelle 58: Beispiel: Blöcke 1 ... 4 des Benutzerdatenfeldes

Master Export/Slave Import	Startadresse	Anz. Variable
5. Block (Byte 16 ... 19)	0,10	0,2
6. Block (Byte 20 ... 23)	0,0	0,0
7. Block (Byte 24 ... 27)	0,0	0,0
8. Block (Byte 28 ... 31)	0,0	0,0

Tabelle 59: Beispiel: Blöcke 1 ... 4 des Benutzerdatenfeldes

Dialog *User Parameter bearbeiten* eines HIMatrix oder HIMax PROFIBUS DP Slaves.

Bild 17: Dialog *User Parameter bearbeiten*

4.9 PROFIBUS Funktionsbausteine

Mit den PROFIBUS Funktionsbausteinen können der HIMA PROFIBUS DP Master und die ihm zugeordneten PROFIBUS DP Slaves optimal den Erfordernissen des Projekts angepasst werden.

Die Funktionsbausteine werden im Anwenderprogramm parametrierbar, sodass die Funktionen des Masters und der Slaves (Alarmer, Diagnosedaten, Zustände) im Anwenderprogramm gesetzt und gelesen werden können.

i

Funktionsbausteine werden nur für spezielle Anwendungen benötigt. Für den normalen zyklischen Datenverkehr zwischen Master und Slave sind diese Funktionsbausteine nicht erforderlich!

Prinzipielle Konfiguration der PROFIBUS DP Funktionsbausteine, siehe Kapitel 5.3.

Folgende Funktionsbausteine stehen zur Verfügung:

Funktionsbaustein	Beschreibung der Funktion	Geeignet ab DP Leistungsstufe
MSTAT 4.9.1	Zustand des Master durch das Anwenderprogramm steuern.	DP-V0
RALRM 4.9.2	Alarmmeldungen der Slaves lesen.	DP-V1
RDIAG 4.9.3	Diagnosemeldungen der Slaves lesen.	DP-V0
RDREC 4.9.4	Azyklische Datensätze der Slaves lesen.	DP-V1
SLACT 4.9.5	Zustand der Slaves durch das Anwenderprogramm steuern.	DP-V0
WRREC 4.9.6	Azyklische Datensätze der Slaves schreiben.	DP-V1

Tabelle 60: Übersicht PROFIBUS DP Funktionsbausteine

i

HIMA PROFIBUS DP Master arbeiten mit der Leistungsstufe DP-V1.

HIMA PROFIBUS DP Slaves arbeiten mit der Leistungsstufe DP-V0.

Dadurch können nicht alle Funktionsbausteine des HIMA PROFIBUS DP Masters einen HIMA PROFIBUS DP Slave steuern.

4.9.1 Funktionsbaustein MSTAT

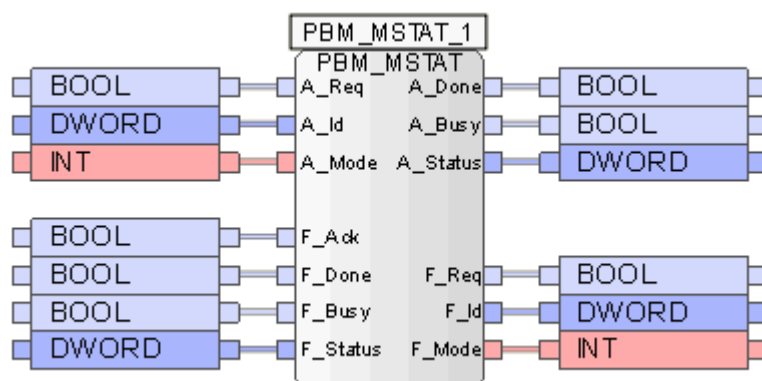


Bild 18: Funktionsbaustein MSTAT

Durch den Funktionsbaustein **MSTAT** (ab DP-V0) kann der PROFIBUS DP Master vom Anwenderprogramm gesteuert werden. Somit ist es möglich, den PROFIBUS DP Master durch einen mechanischen Schalter an einem physikalischen Eingang oder durch einen Timer in einen der folgenden Betriebszustände zu setzen:

- 0: OFFLINE
- 1: STOP
- 2: CLEAR
- 3: OPERATE

Darüber hinaus ermöglicht der Funktionsbaustein **MSTAT** alternernativ die Ansteuerung der PROFIBUS Slaves mit den Zusatzfunktionen FREEZE, UNFREEZE, SYNC, UNSYNC. Sowie die Auswahl der Gruppenzugehörigkeit der anzusteuernenden Slaves.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix A

Über diese Ein- und Ausgänge kann der Funktionsbaustein mit Hilfe des Anwenderprogramms gesteuert und ausgewertet werden. Das Präfix "A" steht für "Application".

A-Eingänge	Beschreibung	Typ
A_Req	Positive Flanke startet den Baustein.	BOOL
A_Id	Master ID (nicht genutzt).	DWORD
A_Mode	PROFIBUS DP Master Betriebsarten In folgende Betriebsarten kann der PROFIBUS DP Master gesetzt werden. Niederwertiges Byte setzt die Betriebsarten. (höherwertiges Byte=0): 0x0000 - OFFLINE 0x0001 - STOP 0x0002 - CLEAR 0x0003 - OPERATE	INT

A-Eingänge	Beschreibung	Typ
	<p>PROFIBUS Slave Zusatzfunktionen</p> <p>Alternativ können Zusatzfunktionen für die an diesen PROFIBUS Master angeschlossenen PROFIBUS Slaves gesetzt werden.</p> <p>Höherwertiges Byte setzt die Zusatzfunktionen:</p> <p>0x04XX - Unfreeze 0x08XX - Freeze 0x10XX - Unsync 0x20XX - Sync</p> <p>Niederwertiges Byte beschreibt die Gruppe(n), für welche die Zusatzfunktionen gelten:</p> <p>0xFF00 - für alle Slaves 0xFF01 - Gruppe 1 0xFF02 - Gruppe 2 0xFF04 - Gruppe 3 ... 0xFF80 - Gruppe 8</p> <p>Beispiel: Um Sync+Freeze an alle Gruppen zu schicken, muss Mode auf 0x28ff gesetzt werden.</p>	

Tabelle 61: A-Eingänge Funktionsbaustein MSTAT

Programmierbeispiel zum Setzen des Eingangs A_Mode

Aus Gründen der Übersichtlichkeit wird ein Hilfsfunktionsbaustein zur Aufbereitung der Eingangsvariablen **A_Mode** des Funktionsbausteins **MSTAT** empfohlen.

Die folgende Bilder zeigen ein Programmierbeispiel, wie dieser Hilfsfunktionsbaustein im Anwenderprogramm realisiert werden kann.

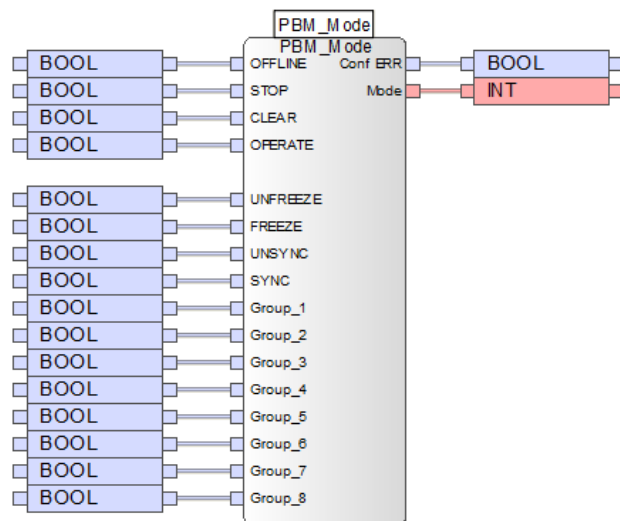


Bild 19: Programmierbeispiel zum Setzen des Eingangs A_Mode

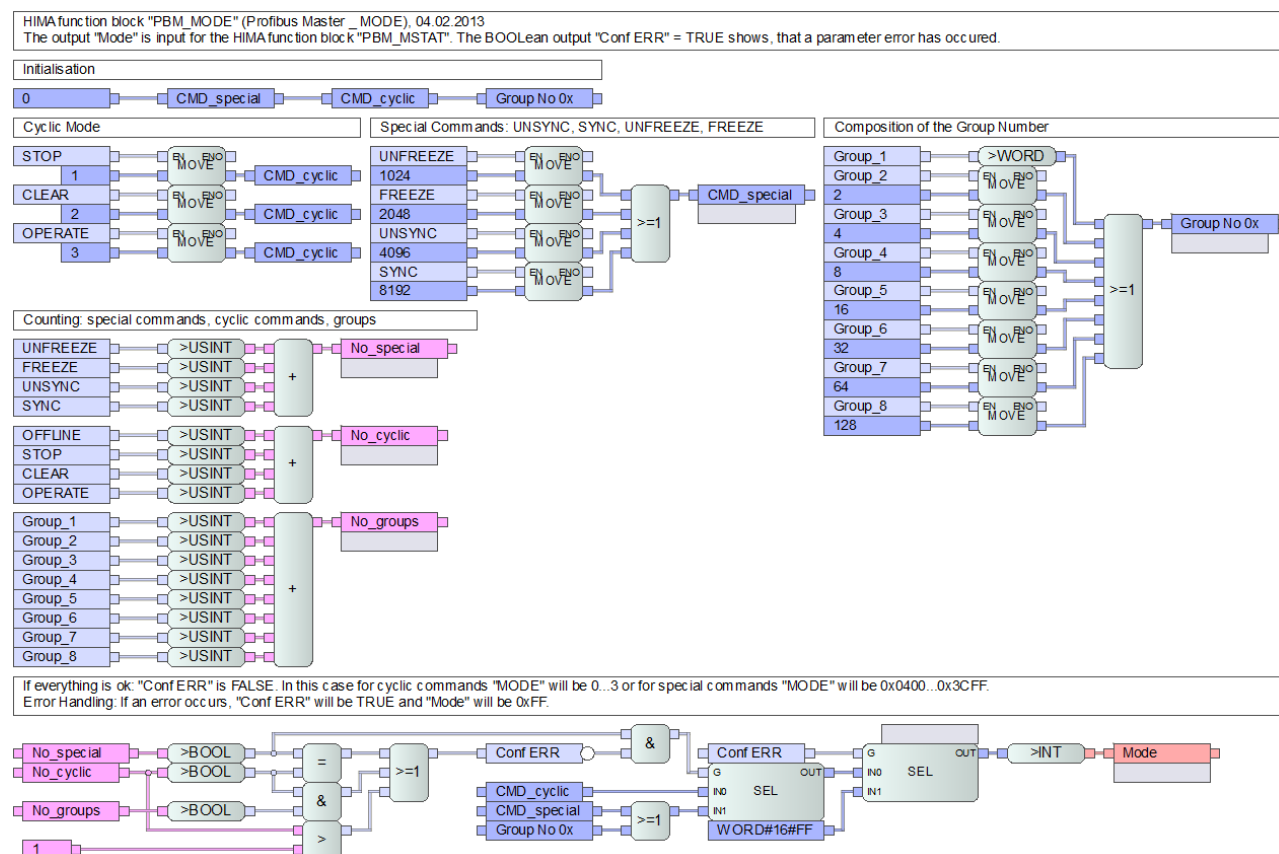


Bild 20: Programmervorschlag für Logik des Hilfsfunktionsbausteins PBM_Mode

A-Ausgänge	Beschreibung	Typ
A_Done	TRUE: Der PROFIBUS DP Master wurde in den am Eingang A_Mode definierten Zustand gesetzt.	BOOL
A_Busy	TRUE: Das Setzen des PROFIBUS DP Master ist noch nicht beendet.	BOOL
A_Status	Status oder Fehlercode, siehe Kapitel 4.11.	DWORD

Tabelle 62: A-Ausgänge Funktionsbaustein MSTAT

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix F

Diese Ein- und Ausgänge des Funktionsbausteins stellen die Verbindung zum Funktionsbaustein MSTAT im Strukturbaum her. Das Präfix "F" steht für "Field".

- i** Die Verbindung des Funktionsbausteins MSTAT im Strukturbaum (im Ordner Funktionsbausteine) mit dem Funktionsbaustein MSTAT (im Anwenderprogramm) erfolgt über gemeinsame Variablen. Diese müssen zuvor im Variableneditor erstellt werden.

Die *F-Eingänge* des Funktionsbausteins **MSTAT** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Ausgänge des Funktionsbausteins **MSTAT** im Strukturbaum verbunden werden.

F-Eingänge	Typ
F_ACK	BOOL
F_DONE	BOOL
F_BUSY	BOOL
F_STATUS	DWORD

Tabelle 63: F-Eingänge Funktionsbaustein MSTAT

Die *F-Ausgänge* des Funktionsbausteins **MSTAT** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Eingänge des Funktionsbausteins **MSTAT** im Strukturbaum verbunden werden.

F-Ausgänge	Typ
F_REQ	BOOL
F_ID	DWORD
F_MODE	INT

Tabelle 64: F-Ausgänge Funktionsbaustein MSTAT

Erstellen des Funktionsbausteins MSTAT im Strukturbaum

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, Funktionsbausteine, Neu** öffnen.
2. Funktionsbaustein **MSTAT** wählen.
3. Rechtsklick auf Funktionsbaustein **MSTAT** und **Edit** wählen.
☒ Variablenzuweisung zum Funktionsbaustein wird geöffnet.

Die Eingänge des Funktionsbausteins **MSTAT** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Ausgänge* des Funktionsbausteins **MSTAT** im Anwenderprogramm verbunden wurden.

Eingänge	Typ
M_ID	DWORD
MODE	INT
REQ	BOOL

Tabelle 65: Eingangs-Systemvariablen

Die Ausgänge des Funktionsbausteins **MSTAT** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Eingänge* des Funktionsbausteins **MSTAT** im Anwenderprogramm verbunden wurden.

Ausgänge	Typ
ACK	BOOL
BUSY	BOOL
DONE	BOOL
STATUS	DWORD

Tabelle 66: Ausgangs-Systemvariablen

Bedienung des Funktionsbausteins MSTAT

1. Im Anwenderprogramm den Eingang *A_Mode* auf den gewünschten Zustand setzen.
Wird *A_Mode* nicht gesetzt, wird nach Schritt 2 ein Fehlercode am Ausgang *A_Status* ausgegeben und der PROFIBUS DP Master wird nicht gesetzt.
2. Im Anwenderprogramm den Eingang *A_Req* auf TRUE setzen.

i

Der Funktionsbaustein reagiert auf einen positiven Flankenwechsel an *A_Req*.

- ☒ Der Ausgang *A_Busy* ist TRUE, bis der MSTAT-Befehl abgearbeitet ist. Danach wechseln die Ausgänge *A_Busy* auf FALSE und *A_Done* auf TRUE.
-

i

Konnte der vorgegebene Mode nicht gesetzt werden, wird ein Fehlercode am Ausgang *A_Status* ausgegeben.

Der aktuelle Mode des Masters kann der Variablen Master-Status entnommen werden (siehe Kapitel 4.10).

4.9.2 Funktionsbaustein RALRM

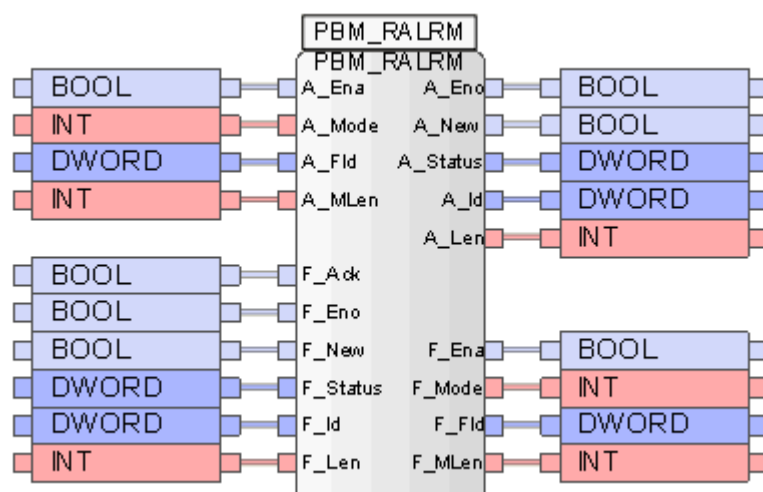


Bild 21: Funktionsbaustein RALRM

Der Funktionsbaustein **RALRM** (ab DP-V1) dient zur Auswertung von Alarmen.

Alarmer sind eine spezielle Form von Diagnosemeldungen, die vorrangig behandelt werden. Alarmer melden der Anwendung, wichtige Ereignisse, die Reaktionen seitens der Anwendung erfordern (z. B. ein WRREC). Dies ist jedoch herstellerabhängig und kann dem Gerätehandbuch des PROFIBUS DP Slaves entnommen werden.

Solange der Funktionsbaustein **RALRM** aktiv ist, wartet dieser auf Alarmermeldungen der Slaves. Wird ein Alarm empfangen, wird der Ausgang **A_NEW** für mindestens einen Zyklus auf TRUE geschaltet und die Alarmerdaten können per Alarmtelegramm ausgelesen werden. Vor dem nächsten Alarm geht **A_NEW** für mindesten einen Zyklus auf FALSE. Alle Alarmer werden implizit bestätigt. Es geht kein Alarm verloren.

Das Anwenderprogramm ist bei Verwendung mehrerer Funktionsbausteine **RALRM** so anzulegen, dass immer nur ein Funktionsbaustein **RALRM** aktiv ist.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek ziehen in das Anwenderprogramm, siehe auch Kapitel 5.3.

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix A

Über diese Ein- und Ausgänge kann der Funktionsbaustein mit Hilfe des Anwenderprogramms gesteuert und ausgewertet werden. Das Präfix "A" steht für "Application".

A-Eingänge	Beschreibung	Typ
A_Ena	Mit TRUE wird der Funktionsbaustein freigegeben.	BOOL
A_Mode	Nicht genutzt.	INT
A_Fld	Nicht genutzt.	DWORD
A_MLen	Maximal erwartete Länge der zu empfangenden Alarmerdaten in Bytes.	INT

Tabelle 67: A-Eingänge Funktionsbaustein RALRM

A-Ausgänge	Beschreibung	Typ
A_Eno	TRUE: Funktionsbaustein aktiv. FALSE: Funktionsbaustein nicht aktiv.	BOOL
A_New	TRUE: Neuer Alarm wurde empfangen. FALSE: Kein neuer Alarm.	BOOL
A_Status	Status oder Fehlercode, siehe Kapitel 4.11.	DWORD
A_Id	Identifikationsnummer des Alarm auslösenden Slave.	DWORD
A_Len	Länge der empfangenen Alarmdaten in Bytes.	INT

Tabelle 68: A-Ausgänge Funktionsbaustein RALRM

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix F

Diese Ein- und Ausgänge des Funktionsbausteins stellen die Verbindung zum Funktionsbaustein RALRM im Strukturbaum her. Das Präfix "F" steht für "Field".

i

Die Verbindung des Funktionsbausteins **RALRM** im Strukturbaum (im Ordner Funktionsbausteine) mit dem Funktionsbaustein **RALRM** (im Anwenderprogramm) erfolgt über gemeinsame Variablen. Diese müssen zuvor im Variableneditor erstellt werden.

Die *F-Eingänge* des Funktionsbausteins **RALRM** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Ausgänge des Funktionsbausteins **RALRM** im Strukturbaum verbunden werden.

F-Eingänge	Typ
F_Ack	BOOL
F_Eno	BOOL
F_New	BOOL
F_Status	DWORD
F_Id	DWORD
F_Len	INT

Tabelle 69: F-Eingänge Funktionsbaustein RALRM

Die *F-Ausgänge* des Funktionsbausteins **RALRM** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Eingänge des Funktionsbausteins **RALRM** im Strukturbaum verbunden werden.

F-Ausgänge	Typ
F_Ena	BOOL
F_Mode	INT
F_FId	DWORD
F_MLen	INT

Tabelle 70: F-Ausgänge Funktionsbaustein RALRM

Erstellen des Funktionsbausteins RALRM im Strukturbaum

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, Funktionsbausteine, Neu** öffnen.
2. Funktionsbaustein **RALRM** wählen.
3. Rechtsklick auf Funktionsbaustein **RALRM** und **Edit** wählen.
☒ Variablenzuweisung zum Funktionsbaustein wird geöffnet.

Die Eingänge des Funktionsbausteins **RALRM** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Ausgänge* des Funktionsbausteins **RALRM** im Anwenderprogramm verbunden wurden.

Eingänge	Typ
EN	BOOL
F_ID	DWORD
MLEN	INT
MODE	INT

Tabelle 71: Eingangs-Systemvariablen

Die Ausgänge des Funktionsbausteins **RALRM** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Eingänge* des Funktionsbausteins **RALRM** im Anwenderprogramm verbunden wurden.

Ausgänge	Typ
ACK	BOOL
ENO	BOOL
ID	DWORD
LEN	INT
NEW	BOOL
STATUS	DWORD

Tabelle 72: Ausgangs-Systemvariablen

Im Register „Prozessvariablen“ des Funktionsbausteins **RALRM** im Strukturbaum sind Variablen zu definieren, deren Struktur zu den Alarmdaten passt. Werden keine Variablen definiert, können Alarmdaten zwar angefordert, aber nicht gelesen werden.

Eine Alarmmeldung enthält mindestens vier Bytes. Die ersten vier Bytes der Alarmmeldung enthalten die Standard-Alarmdaten.

Zur Dekodierung der Standard-Alarme stellt HIMA den Hilfsfunktionsbaustein **ALARM** (siehe Kapitel 4.10) bereit.

-
- i** Enthält ein Alarmtelegramm mehr Bytes als im Register „Daten“ definiert wurden, wird nur die Anzahl der definierten Bytes übernommen. Der Rest wird abgeschnitten.
-

Alarmdaten	Beschreibung
Byte 0	Länge der Alarmmeldung in Byte (4 ... 126)
Byte 1	Kennung für den Alarmtyp 1: Diagnosealarm 2: Prozessalarm 3: Ziehenalarm 4: Steckenalarm 5: Statusalarm 6: Updatealarm 31: Ausfall einer Erweiterung eines Master oder Slaves. 32 ... 126: Herstellerspezifisch: Die Bedeutung muss der Herstellerbeschreibung des Geräts entnommen werden.
Byte 2	Steckplatznummer der Alarm auslösenden Komponente.
Byte 3	0: Keine weitere Information. 1: Ankommender Alarm, Slot gestört. 2: Ausgehender Alarm, Slot nicht mehr gestört. 3: Ausgehender Alarm, Slot weiterhin gestört.
Byte 4 ... 126	Die Bedeutung muss der Herstellerbeschreibung des Geräts entnommen werden.

Tabelle 73: Alarmdaten

i

Die Struktur der Standard-Alarme (Bytes 0...3) ist normiert und für alle Hersteller identisch. Für die herstellerspezifisch genutzten Bytes 4...126, siehe Gerätehandbuch des PROFIBUS DP Slave.

Geräte nach dem DP-V0-Standard unterstützen keine Alarmtelegramme.

Bedienung des Funktionsbausteins RALRM

1. Im Anwenderprogramm am Eingang *A_Mlen* die Anzahl der maximal zu erwartenden Alarmdaten in Bytes definieren. Während des Betriebs kann *A_Mlen* nicht geändert werden.
 2. Im Anwenderprogramm den Eingang *A_Ena* auf TRUE setzen.
-

i

Im Gegensatz zu den anderen Funktionsbausteinen, ist der Funktionsbaustein **RALRM** nur aktiv, solange der Eingang *A_Ena* TRUE ist.

Wurde der Baustein erfolgreich gestartet, dann geht der Ausgang *A_Eno* auf TRUE. Konnte der Baustein nicht gestartet werden, wird ein Fehlercode am Ausgang *A_Status* ausgegeben.

Trifft ein neuer Alarm ein, geht der Ausgang *A_New* für mindestens einen Zyklus auf TRUE. Für diese Zeit enthalten die Ausgänge die Alarmdaten des Alarm auslösenden Slaves, die ausgewertet werden können.

Danach geht der Ausgang *A_New* wieder für mindestens einen Zyklus auf FALSE. Die Ausgänge *A_ID* und *A_Len* werden auf Null zurückgesetzt, bevor die nächste Alarmmeldung empfangen und ausgewertet werden kann.

4.9.3 Funktionsbaustein RDIAG

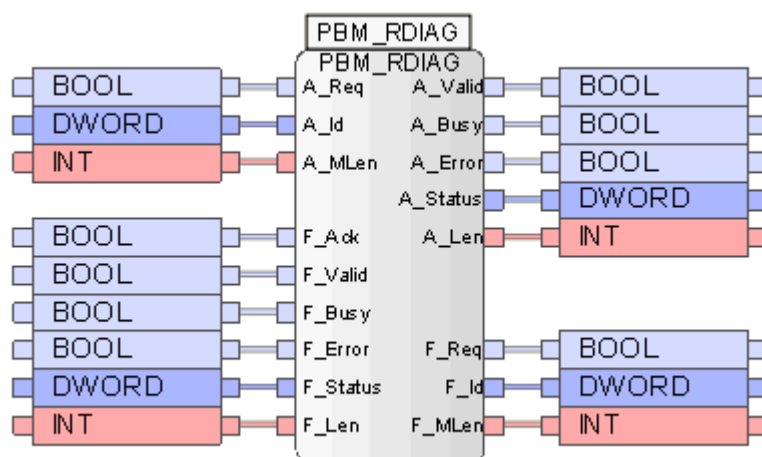


Bild 22: Funktionsbaustein RDIAG

Der Funktionsbaustein **RDIAG** (ab DP-V0) dient zum Auslesen der aktuellen Diagnosemeldung (6 Byte ... 240 Bytes) eines Slave.

Im HIMA PROFIBUS DP Master dürfen beliebig viele **RDIAG**-Bausteine gleichzeitig aktiv sein.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix A

Über diese Ein- und Ausgänge kann der Funktionsbaustein mit Hilfe des Anwenderprogramms gesteuert und ausgewertet werden. Das Präfix "A" steht für "Application".

A-Eingänge	Beschreibung	Typ
A_Req	Positive Flanke startet die Anforderung einer Diagnosemeldung.	BOOL
A_ID	Identifikationsnummer des Slave, siehe Kapitel 4.10.	DWORD
A_MLen	Maximal erwartete Länge der zu lesenden Diagnosemeldung in Bytes.	INT

Tabelle 74: A-Eingänge Funktionsbaustein RDIAG

A-Ausgänge	Beschreibung	Typ
A_Valid	Eine neue Diagnosemeldung wurde Empfangen und ist gültig.	BOOL
A_Busy	TRUE: Das Lesen ist noch nicht beendet.	BOOL
A_Error	TRUE: Beim Lesen trat ein Fehler auf.	BOOL
A_Status	Status oder Fehlercode, siehe Kapitel 4.11.	DWORD
A_Len	Länge der gelesenen Diagnosedaten in Bytes.	INT

Tabelle 75: A-Ausgänge Funktionsbaustein RDIAG

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix F

Diese Ein- und Ausgänge des Funktionsbausteins stellen die Verbindung zum Funktionsbaustein **RDIAG** im Strukturbaum her. Das Präfix "F" steht für "Field".

i

Die Verbindung des Funktionsbausteins **RDIAG** im Strukturbaum (im Ordner Funktionsbausteine) mit dem Funktionsbaustein **RDIAG** (im Anwenderprogramm) erfolgt über gemeinsame Variablen. Diese müssen zuvor im Variableneditor erstellt werden.

Die *F-Eingänge* des Funktionsbausteins **RDIAG** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Ausgänge des Funktionsbausteins **RDIAG** im Strukturbaum verbunden werden.

F-Eingänge	Typ
F_Ack	BOOL
F_Valid	BOOL
F_Busy	BOOL
F_Error	BOOL
F_Status	DWORD
F_Len	INT

Tabelle 76: F-Eingänge Funktionsbaustein RDIAG

Die *F-Ausgänge* des Funktionsbausteins **RDIAG** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Eingänge des Funktionsbausteins **RDIAG** im Strukturbaum verbunden werden.

F-Ausgänge	Typ
F_Req	BOOL
F_Id	DWORD
F_MLen	INT

Tabelle 77: F-Ausgänge Funktionsbaustein RDIAG

Erstellen des Funktionsbausteins RDIAG im Strukturbaum

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, Funktionsbausteine, Neu** öffnen.
2. Funktionsbaustein **RDIAG** wählen.
3. Rechtsklick auf Funktionsbaustein **RDIAG** und **Edit** wählen.
☒ Variablenzuweisung zum Funktionsbaustein wird geöffnet.

Die Eingänge des Funktionsbausteins **RDIAG** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Ausgänge* des Funktionsbausteins **RDIAG** im Anwenderprogramm verbunden wurden.

Eingänge	Typ
ID	DWORD
MLEN	INT
REQ	BOOL

Tabelle 78: Eingangs-Systemvariablen

Die Ausgänge des Funktionsbausteins **RDIAG** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Eingänge* des Funktionsbausteins **RDIAG** im Anwenderprogramm verbunden wurden.

Ausgänge	Typ
ACK	BOOL
BUSY	BOOL
ERROR	BOOL
LEN	INT
Status	DWORD
VALID	BOOL

Tabelle 79: Ausgangs-Systemvariablen

Diagnosedaten

Im Register **Daten** sind Variablen zu definieren, deren Struktur zu den Diagnosedaten passen muss. Eine Diagnosemeldung enthält mindestens sechs Bytes und maximal 240 Bytes. Die ersten vier Bytes der Diagnosemeldung enthalten die Standard-Diagnose.

Zur Dekodierung der Standard-Diagnose stellt HIMA den Hilfsfunktionsbaustein **STDDIAG** bereit, siehe Kapitel 4.10.

i

Enthält ein Diagnosetelegramm mehr Bytes als im Register „Daten“ definiert wurden, wird nur die Anzahl der definierten Bytes übernommen. Der Rest wird abgeschnitten

Diagnosedaten	Beschreibung
Byte 0	Byte 0 ... 3 enthalten die Standard-Diagnose. Die Standard-Diagnose kann als Variable vom Typ DWORD mit dem Hilfsfunktionsbaustein STDDIAG dekodiert werden.
Byte 1	
Byte 2	
Byte 3	Busadresse des Masters, dem ein Slave zugeordnet ist.
Byte 4	High-Byte (Herstellerkennung).
Byte 5	Low-Byte (Herstellerkennung).
Byte 6 ... 240	Die Bedeutung muss der Herstellerbeschreibung des Geräts entnommen werden.

Tabelle 80: Diagnosedaten

i

Die HIMA Slaves liefern ein Diagnosetelegramm von sechs Bytes Länge. Die Bedeutung der Bytes ist standardisiert.

Für Slaves anderer Hersteller sind nur die ersten sechs Bytes funktionell identisch.

Für weitere Informationen über das Diagnosetelegramm, siehe Herstellerbeschreibung des Slaves.

Bedienung des Funktionsbausteins RDIAG

1. Im Anwenderprogramm die Slave-Adresse an den Eingang *A_Id* setzen.
2. Im Anwenderprogramm am Eingang *A_Mlen* die Anzahl der maximal zu erwartenden Diagnosedaten in Bytes definieren.
3. Im Anwenderprogramm den Eingang *A_Req* auf TRUE setzen.

i

Der Funktionsbaustein reagiert auf einen positiven Flankenwechsel an *A_Req*.

Der Ausgang *A_Busy* ist TRUE, bis die Diagnoseanforderung abgearbeitet ist. Danach wechseln die Ausgänge *A_Busy* auf FALSE und *A_Valid* oder *A_Error* auf TRUE.

Ist das Diagnosetelegramm gültig, geht der Ausgang *A_Valid* auf TRUE. Die Diagnosedaten können über die im Register *Daten* definierten Variablen ausgewertet werden. Der Ausgang *A_Len* enthält die Anzahl der Diagnosedaten in Bytes, die tatsächlich ausgelesen wurden.

Konnte das Diagnosetelegramm nicht erfolgreich gelesen werden, dann wechselt der Ausgang *A_Error* auf TRUE und ein Fehlercode wird am Ausgang *A_Status* ausgegeben.

4.9.4 Funktionsbaustein RDREC

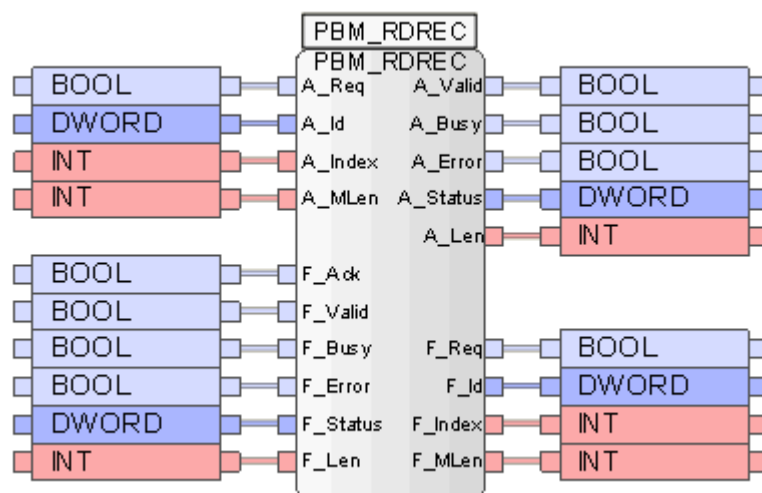


Bild 23: Funktionsbaustein RDREC

Der Funktionsbaustein **RDREC** dient zum azyklischen Lesen eines am Eingang *A_Index* adressierten Datensatzes von einem Slave. Welche Daten gelesen werden können, muss der Betriebsanleitung des Slaves entnommen werden.

Diese Funktion ist erst ab DP-V1 definiert und optional.

Im HIMA PROFIBUS DP Master können gleichzeitig bis zu 32 **RDREC**- und/oder **WRREC**-Bausteine aktiv sein.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix A

Über diese Ein- und Ausgänge kann der Funktionsbaustein mit Hilfe des Anwenderprogramms gesteuert und ausgewertet werden. Das Präfix "A" steht für "Application".

A-Eingänge	Beschreibung	Typ
A_Req	Positive Flanke startet die Anforderung zum Lesen.	BOOL
A_Id	Identifikationsnummer des Slave, siehe Kapitel 4.10.	DWORD
A_Index	Datensatznummer des zu lesenden Datensatzes. Die Bedeutung muss der Herstellerbeschreibung des Geräts entnommen werden.	INT
A_MLen	Maximale Länge der zu lesenden Daten in Bytes.	INT

Tabelle 81: A-Eingänge Funktionsbaustein RDREC

A-Ausgänge	Beschreibung	Typ
A_Valid	Ein neuer Datensatz wurde Empfangen und ist gültig.	BOOL
A_Busy	TRUE: Der Lesevorgang ist noch nicht beendet.	BOOL
A_Error	TRUE: Ein Fehler ist aufgetreten. FALSE: Kein Fehler.	BOOL
A_Status	Status oder Fehlercode, siehe Kapitel 4.11.	DWORD
A_Len	Länge der gelesenen Datensatzinformation in Bytes.	INT

Tabelle 82: A-Ausgänge Funktionsbaustein RDREC

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix F

Diese Ein- und Ausgänge des Funktionsbausteins stellen die Verbindung zum Funktionsbaustein **RDREC** im Strukturbaum her. Das Präfix "F" steht für "Field".

i

Die Verbindung des Funktionsbausteins **RDREC** im Strukturbaum (im Ordner Funktionsbausteine) mit dem Funktionsbaustein **RDREC** (im Anwenderprogramm) erfolgt über gemeinsame Variablen. Diese müssen zuvor im Variableneditor erstellt werden.

Die *F-Eingänge* des Funktionsbausteins **RDREC** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Ausgänge des Funktionsbausteins **RDREC** im Strukturbaum verbunden werden.

F-Eingänge	Typ
F_Ack	BOOL
F_Valid	BOOL
F_Busy	BOOL
F_Error	BOOL
F_Status	DWORD
F_Len	INT

Tabelle 83: F-Eingänge Funktionsbaustein RDREC

Die *F-Ausgänge* des Funktionsbausteins **RDREC** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Eingänge des Funktionsbausteins **RDREC** im Strukturbaum verbunden werden.

F-Ausgänge	Typ
F_Req	BOOL
F_Id	DWORD
F_Index	INT
F_Mlen	INT

Tabelle 84: F-Ausgänge Funktionsbaustein RDREC

Erstellen des Funktionsbausteins RDREC im Strukturbaum

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, Funktionsbausteine, Neu** öffnen.
2. Funktionsbaustein **RDREC** wählen.
3. Rechtsklick auf Funktionsbaustein **RDREC** und **Edit** wählen.
☒ Variablenzuweisung zum Funktionsbaustein wird geöffnet.

Die Eingänge des Funktionsbausteins **RDREC** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Ausgänge* des Funktionsbausteins **RDREC** im Anwenderprogramm verbunden wurden.

Eingänge	Typ
ID	DWORD
INDEX	INT
MLen	INT
REQ	BOOL

Tabelle 85: Eingangs-Systemvariablen

Die Ausgänge des Funktionsbausteins **RDREC** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Eingänge* des Funktionsbausteins **RDREC** im Anwenderprogramm verbunden wurden.

Ausgänge	Typ
ACK	BOOL
BUSY	BOOL
ERROR	BOOL
LEN	INT
STATUS	DWORD
VALID	BOOL

Tabelle 86: Ausgangs-Systemvariablen

Daten	Beschreibung
Es sind keine Variablen vorgegeben	Im Register <i>Prozessvariablen</i> kann eine beliebige Datenstruktur definiert werden, die allerdings auf die Struktur des Datensatzes passen muss. Die Struktur des Datensatzes muss aus der Bedienungsanleitung vom Hersteller des Slaves entnommen werden.

Tabelle 87: Daten

Bedienung des Funktionsbausteins RDREC

1. Im Anwenderprogramm die Slave-Adresse an den Eingang *A_Id* setzen.
2. Im Anwenderprogramm den Slave-spezifischen Index für den Datensatz (Handbuch des Herstellers) am Eingang *A_Index* setzen.
3. Im Anwenderprogramm die Länge des zu lesenden Datensatzes am Eingang *A_Len* setzen.
4. Im Anwenderprogramm den Eingang *A_Req* auf TRUE setzen.

i

Der Funktionsbaustein reagiert auf einen positiven Flankenwechsel an *A_Req*.

Der Ausgang *A_Busy* ist TRUE, bis die Datensatzanforderung abgearbeitet ist. Danach wechseln die Ausgänge *A_Busy* auf FALSE und *A_Valid* oder *A_Error* auf TRUE.

Ist der Datensatz gültig, geht der Ausgang *A_Valid* auf TRUE. Der Datensatz kann über die im Register *Daten* definierten Variablen ausgewertet werden. Der Ausgang *A_Len* enthält die tatsächliche Länge des ausgelesenen Datensatzes.

Konnte der Datensatz nicht erfolgreich gelesen werden, dann wechselt der Ausgang *A_Error* auf TRUE und ein Fehlercode wird am Ausgang *A_Status* ausgegeben.

4.9.5 Funktionsbaustein SLACT

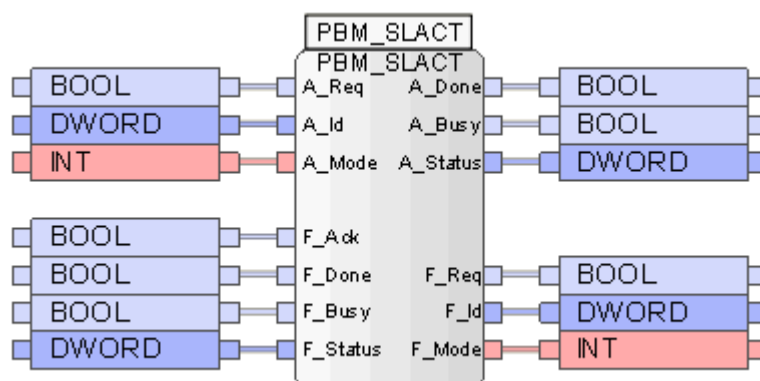


Bild 24: Funktionsbaustein SLACT

Der Funktionsbaustein **SLACT** (ab DP-V0) dient zum Aktivieren und Deaktivieren eines Slave aus dem Anwenderprogramm des PROFIBUS DP Master. Somit ist es möglich, den Slave durch einen mechanischen Schalter an einem physikalischen Eingang des PROFIBUS DP Master, oder durch einen Timer in einen der folgenden Zustände zu setzen:

- $\neq 0$: Aktiv
- $= 0$: Nicht aktiv

Das Anwenderprogramm ist bei Verwendung mehrerer Funktionsbausteine **SLACT** so anzulegen, dass immer nur ein Funktionsbaustein **SLACT** aktiv ist.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix A

Über diese Ein- und Ausgänge kann der Funktionsbaustein mit Hilfe des Anwenderprogramms gesteuert und ausgewertet werden. Das Präfix "A" steht für "Application".

A-Eingänge	Beschreibung	Typ
A_Req	Positive Flanke startet den Baustein.	BOOL
A_Id	Identifikationsnummer des Slave, siehe Kapitel 4.10.	DWORD
A_Mode	Zustand, in den der PROFIBUS DP Slave gesetzt werden soll: $\neq 0$: Aktiv (Verbunden). $= 0$: Nicht aktiv (Deaktiviert).	INT

Tabelle 88: A-Eingänge Funktionsbaustein SLACT

A-Ausgänge	Beschreibung	Typ
A_Done	TRUE: Der PROFIBUS DP Slave wurde in den am Eingang „A_Mode“ definierten Zustand gesetzt.	BOOL
A_Busy	TRUE: Das Setzen des PROFIBUS DP Slave ist noch nicht beendet.	BOOL
A_Status	Status oder Fehlercode, siehe Kapitel 4.11.	DWORD

Tabelle 89: A-Ausgänge Funktionsbaustein SLACT

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix F

Diese Ein- und Ausgänge des Funktionsbausteins stellen die Verbindung zum Funktionsbaustein **SLACT** im Strukturbaum her. Das Präfix "F" steht für "Field".

i

Die Verbindung des Funktionsbausteins **SLACT** im Strukturbaum (im Ordner Funktionsbausteine) mit dem Funktionsbaustein **SLACT** (im Anwenderprogramm) erfolgt über gemeinsame Variablen. Diese müssen zuvor im Variableneditor erstellt werden.

Die *F-Eingänge* des Funktionsbausteins **SLACT** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Ausgänge des Funktionsbausteins **SLACT** im Strukturbaum verbunden werden.

F-Eingänge	Typ
F_Ack	BOOL
F_Done	BOOL
F_Busy	BOOL
F_Status	DWORD

Tabelle 90: F-Eingänge Funktionsbaustein SLACT

Die *F-Ausgänge* des Funktionsbausteins **SLACT** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Eingänge des Funktionsbausteins **SLACT** im Strukturbaum verbunden werden.

F-Ausgänge	Typ
F_Req	BOOL
F_Id	DWORD
F_Mode	INT

Tabelle 91: F-Ausgänge Funktionsbaustein SLACT

Erstellen des Funktionsbausteins SLACT im Strukturbaum

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, Funktionsbausteine, Neu** öffnen.
2. Funktionsbaustein **SLACT** wählen.
3. Rechtsklick auf Funktionsbaustein **SLACT** und **Edit** wählen.
☒ Variablenzuweisung zum Funktionsbaustein wird geöffnet.

Die Eingänge des Funktionsbausteins **SLACT** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Ausgänge* des Funktionsbausteins **SLACT** im Anwenderprogramm verbunden wurden.

Eingänge	Typ
ID	DWORD
MODE	INT
REQ	BOOL

Tabelle 92: Eingangs-Systemvariablen

Die Ausgänge des Funktionsbausteins **SLACT** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Eingänge* des Funktionsbausteins **SLACT** im Anwenderprogramm verbunden wurden.

Ausgänge	Typ
ACK	BOOL
BUSY	BOOL
DONE	BOOL
STATUS	DWORD

Tabelle 93: Ausgangs-Systemvariablen

Bedienung des Funktionsbausteins SLACT

1. Im Anwenderprogramm den Eingang *A_Mode* auf den gewünschten Zustand setzen.
2. Im Anwenderprogramm den Identifier mit der Slave-Adresse am Eingang *A_Id* setzen.
3. Im Anwenderprogramm den Eingang *A_Req* auf TRUE setzen.

i

Der Funktionsbaustein reagiert auf einen positiven Flankenwechsel an *A_Req*.

Der Ausgang *A_Busy* ist TRUE, bis der *SLACT*-Befehl abgearbeitet ist. Danach wechseln die Ausgänge *A_Busy* auf FALSE und *A_Done* auf TRUE.

Am Ausgang *A_Status* wird der Slave-Mode ausgegeben, wenn der Slave-Mode gesetzt werden konnte.

Am Ausgang *A_Status* wird ein Fehlercode ausgegeben, wenn der Slave-Mode nicht gesetzt werden konnte.

4.9.6 Funktionsbaustein WRREC

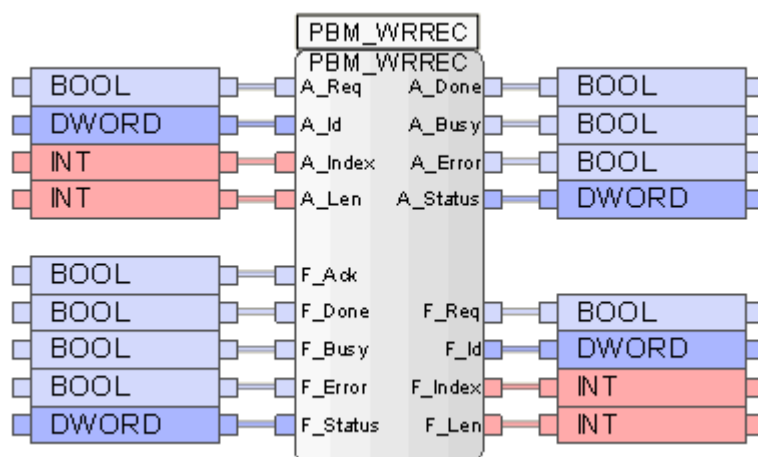


Bild 25: Funktionsbaustein WRREC

Der Funktionsbaustein **WRREC** (ab DP-V1) dient zum azyklischen Schreiben eines mit *A_Index* adressierten Datensatzes an einen Slave. Welche Daten geschrieben werden können, muss der Betriebsanleitung des Slaves entnommen werden.

Im HIMA PROFIBUS DP Master können gleichzeitig bis zu 32 **RDREC**- und/oder **WRREC**-Bausteine aktiv sein.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix A

Über diese Ein- und Ausgänge kann der Funktionsbaustein mit Hilfe des Anwenderprogramms gesteuert und ausgewertet werden. Das Präfix "A" steht für "Application".

A-Eingänge	Beschreibung	Typ
A_Req	Positive Flanke startet die Anforderung zum Schreiben eines Datensatzes.	BOOL
A_Id	Identifikationsnummer des Slaves, siehe Kapitel 4.10.	DWORD
A_Index	Datensatznummer des zu schreibenden Datensatzes. Die Bedeutung muss der Herstellerbeschreibung des Geräts entnommen werden.	INT
A_Len	Länge des zu schreibenden Datensatzes in Bytes.	INT

Tabelle 94: A-Eingänge Funktionsbaustein WRREC

A-Ausgänge	Beschreibung	Typ
A_Done	TRUE: Funktionsbaustein hat den Schreibvorgang beendet.	BOOL
A_Busy	TRUE: Funktionsbaustein hat den Schreibvorgang noch nicht beendet.	BOOL
A_Error	TRUE: Beim Schreibvorgang trat ein Fehler auf.	BOOL
A_Status	Status oder Fehlercode, siehe Kapitel 4.11.	DWORD

Tabelle 95: A-Ausgänge Funktionsbaustein WRREC

Ein- und Ausgänge des Funktionsbausteins mit dem Präfix F

Diese Ein- und Ausgänge des Funktionsbausteins stellen die Verbindung zum Funktionsbaustein **WRREC** im Strukturbaum her. Das Präfix "F" steht für "Field".

i

Die Verbindung des Funktionsbausteins **WRREC** im Strukturbaum (im Ordner Funktionsbausteine) mit dem Funktionsbaustein **WRREC** (im Anwenderprogramm) erfolgt über gemeinsame Variablen. Diese müssen zuvor im Variableneditor erstellt werden.

Die *F-Eingänge* des Funktionsbausteins **WRREC** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Ausgänge des Funktionsbausteins **WRREC** im Strukturbaum verbunden werden.

F-Eingänge	Typ
F_Ack	BOOL
F_Done	BOOL
F_Busy	BOOL
F_Error	BOOL
F_Status	DWORD

Tabelle 96: F-Eingänge Funktionsbaustein **WRREC**

Die *F-Ausgänge* des Funktionsbausteins **WRREC** im Anwenderprogramm mit den gleichen Variablen verbinden, mit denen später auch die Eingänge des Funktionsbausteins **WRREC** im Strukturbaum verbunden werden.

F-Ausgänge	Typ
F_Req	BOOL
F_Id	DWORD
F_Index	INT
F_Len	INT

Tabelle 97: F-Ausgänge Funktionsbaustein **WRREC**

Erstellen des Funktionsbausteins **WRREC** im Strukturbaum

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master, Funktionsbausteine, Neu** öffnen.
2. Funktionsbaustein **WRREC** wählen.
3. Rechtsklick auf Funktionsbaustein **WRREC** und **Edit** wählen.
 - ☒ Variablenzuweisung zum Funktionsbaustein wird geöffnet.

Die Eingänge des Funktionsbausteins **WRREC** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Ausgänge* des Funktionsbausteins **WRREC** im Anwenderprogramm verbunden wurden.

Eingänge	Typ
ID	DWORD
INDEX	INT
LEN	INT
REQ	BOOL

Tabelle 98: Eingangs-Systemvariablen

Die Ausgänge des Funktionsbausteins **WRREC** im Strukturbaum mit den gleichen Variablen verbinden, mit denen zuvor auch die *F-Eingänge* des Funktionsbausteins **WRREC** im Anwenderprogramm verbunden wurden.

Ausgänge	Typ
ACK	BOOL
BUSY	BOOL
DONE	BOOL
ERROR	BOOL
STATUS	DWORD

Tabelle 99: Ausgangs-Systemvariablen

Daten	Beschreibung
Es sind keine Variablen vorgegeben	Im Register <i>Prozessvariablen</i> kann eine beliebige Datenstruktur definiert werden, die allerdings auf die Struktur des Datensatzes passen muss. Die Struktur des Datensatzes muss aus der Bedienungsanleitung vom Hersteller des Slaves entnommen werden.

Tabelle 100: Daten

Für die Bedienung des Funktionsbausteins **WRREC** sind die folgenden Schritte erforderlich

1. Im Anwenderprogramm die Slave-Adresse an den Eingang *A_Id* setzen.
2. Im Anwenderprogramm den Slave-spezifischen Index für den Datensatz (Handbuch des Herstellers) am Eingang *A_Index* setzen.
3. Im Anwenderprogramm die Länge des zu schreibenden Datensatzes am Eingang *A_Len* setzen.
4. Im Anwenderprogramm den Datensatz, wie im Register „Daten“ definiert, einstellen.
5. Im Anwenderprogramm den Eingang *A_Req* auf TRUE setzen.

i

Der Funktionsbaustein reagiert auf einen positiven Flankenwechsel an *A_Req*.

Der Ausgang *A_Busy* ist TRUE, bis der Datensatz geschrieben ist. Danach wechseln die Ausgänge *A_Busy* auf FALSE und *A_Done* auf TRUE.

Konnte der Datensatz nicht erfolgreich geschrieben werden, dann wechselt der Ausgang *A_Error* auf TRUE und ein Fehlercode wird am Ausgang *A_Status* ausgegeben.

4.10 PROFIBUS Hilfsfunktionsbausteine

Die Hilfsfunktionsbausteine werden zur Parametrierung und Auswertung der Ein- und Ausgänge der Funktionsbausteine verwendet.

Folgende Hilfsfunktionsbausteine stehen zur Verfügung:

Hilfsfunktionsbausteine	Beschreibung der Funktion
ACTIVE (siehe Kap. 4.10.1)	Ist der Slave Aktiv oder Inaktiv.
ALARM (siehe Kap. 4.10.2)	Dekodieren der Alarmedaten.
DEID (siehe Kap. 4.10.3)	Identifikationsnummer dekodieren.
ID (siehe Kap. 4.10.4)	Die Funktion ID generiert aus vier Bytes einen Identifier.
NSLOT (siehe Kap. 4.10.5)	Fortlaufende Identifikationsnummer für die Slots erstellen.
SLOT (siehe Kap. 4.10.6)	SLOT Identifikationsnummer mit Slot-Nummer erstellen.
STDDIAG (siehe Kap. 4.10.7)	Standard-Diagnose eines Slaves dekodieren.
LATCH	Wird nur innerhalb anderer Funktionsbausteine verwendet.
PIG	Wird nur innerhalb anderer Funktionsbausteine verwendet.
PIGII	Wird nur innerhalb anderer Funktionsbausteine verwendet.

Tabelle 101: Übersicht Hilfsfunktionsbausteine

4.10.1 Hilfsfunktionsbaustein ACTIVE

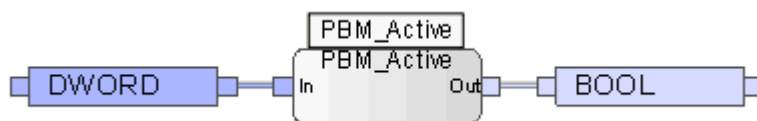


Bild 26: Hilfsfunktionsbaustein ACTIVE

Der Hilfsfunktionsbaustein ACTIVE ermittelt aus der Standard-Diagnose eines PROFIBUS DP Slaves, ob der Slave gerade aktiv oder inaktiv ist.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
IN	Standard-Diagnose des Slave.	DWORD

Tabelle 102: Eingänge Hilfsfunktionsbaustein ACTIVE

Ausgänge	Beschreibung	Typ
OUT	TRUE: Slave ist aktiv. FALSE: Slave ist inaktiv.	BOOL

Tabelle 103: Ausgänge Hilfsfunktionsbaustein ACTIVE

4.10.2 Hilfsfunktionsbaustein ALARM

(Dekodieren der Alarmdaten)

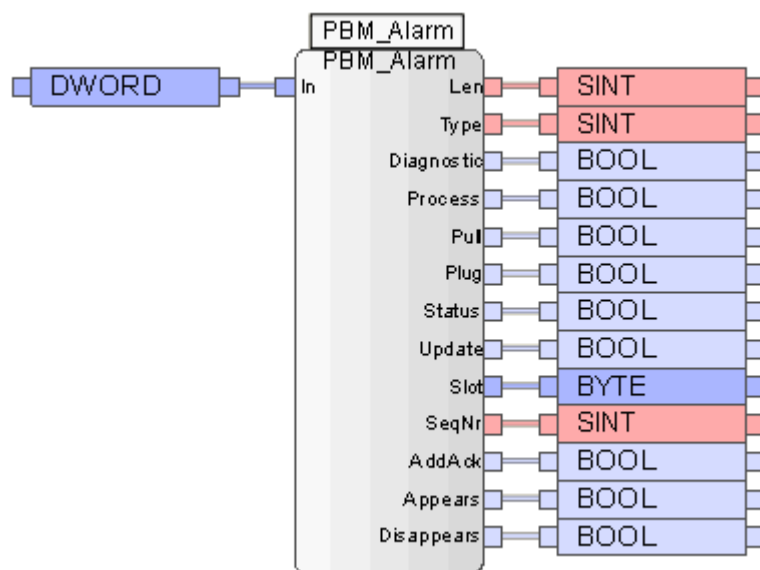


Bild 27: Hilfsfunktionsbaustein ALARM

Der Hilfsfunktionsbaustein **ALARM** dekodiert die Standard-Alarmdaten eines PROFIBUS DP Slaves.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
IN	Standardalarm	DWORD

Tabelle 104: Eingänge Hilfsfunktionsbaustein ALARM

Ausgang	Beschreibung	Typ
Len	Länge der gesamten Alarmmeldung.	SINT
Type	1: Diagnosealarm 2: Prozessalarm 3: Ziehenalarm 4: Steckenalarm 5: Statusalarm 6: Updatealarm Andere Nummern sind entweder reserviert oder herstellerspezifisch. Die Bedeutung muss der Herstellerbeschreibung des Geräts entnommen werden.	SINT
Diagnostic	TRUE = Diagnosealarm	BOOL
Process	TRUE = Prozessalarm	BOOL
Pull	TRUE = Modul wurde gezogen.	BOOL
Plug	TRUE = Modul wurde wieder gesteckt.	BOOL
Status	TRUE = Status-Alarm	BOOL
Update	TRUE = Update-Alarm	BOOL
Slot	Alarmauslösendes Modul.	BYTE
SeqNr	Alarm-Sequenznummer.	SINT

Ausgang	Beschreibung	Typ																							
AddAck	TRUE bedeutet, dass der Slave, der diesen Alarm ausgelöst hat, eine zusätzliche Bestätigung durch die Anwendung erwartet. Welche genau, muss dem Slave-Handbuch des Herstellers entnommen werden.	BOOL																							
Appears Disappears	<table border="1"> <thead> <tr> <th>Ausgang</th><th>Wert</th><th>Beschreibung</th></tr> </thead> <tbody> <tr> <td>Appears</td><td>TRUE</td><td rowspan="2">Sind beide FALSE, dann ist bis zu diesem Zeitpunkt kein Fehler aufgetreten.</td></tr> <tr> <td>Disappears</td><td>FALSE</td></tr> <tr> <td>Appears</td><td>TRUE</td><td rowspan="2">Ein Fehler ist aufgetreten und steht noch an.</td></tr> <tr> <td>Disappears</td><td>FALSE</td></tr> <tr> <td>Appears</td><td>TRUE</td><td rowspan="2">Ein Fehler war aufgetreten und verschwindet gerade.</td></tr> <tr> <td>Disappears</td><td>FALSE</td></tr> <tr> <td>Appears</td><td>TRUE</td><td rowspan="2">Sind beide TRUE, dann verschwindet der Fehler, der Slave ist aber weiterhin gestört.</td></tr> <tr> <td>Disappears</td><td>FALSE</td></tr> </tbody> </table>	Ausgang	Wert	Beschreibung	Appears	TRUE	Sind beide FALSE, dann ist bis zu diesem Zeitpunkt kein Fehler aufgetreten.	Disappears	FALSE	Appears	TRUE	Ein Fehler ist aufgetreten und steht noch an.	Disappears	FALSE	Appears	TRUE	Ein Fehler war aufgetreten und verschwindet gerade.	Disappears	FALSE	Appears	TRUE	Sind beide TRUE, dann verschwindet der Fehler, der Slave ist aber weiterhin gestört.	Disappears	FALSE	BOOL
Ausgang	Wert	Beschreibung																							
Appears	TRUE	Sind beide FALSE, dann ist bis zu diesem Zeitpunkt kein Fehler aufgetreten.																							
Disappears	FALSE																								
Appears	TRUE	Ein Fehler ist aufgetreten und steht noch an.																							
Disappears	FALSE																								
Appears	TRUE	Ein Fehler war aufgetreten und verschwindet gerade.																							
Disappears	FALSE																								
Appears	TRUE	Sind beide TRUE, dann verschwindet der Fehler, der Slave ist aber weiterhin gestört.																							
Disappears	FALSE																								

Tabelle 105: Ausgänge Hilfsfunktionsbaustein ALARM

4.10.3 Hilfsfunktionsbaustein DEID (Identifikationsnummer dekodieren)

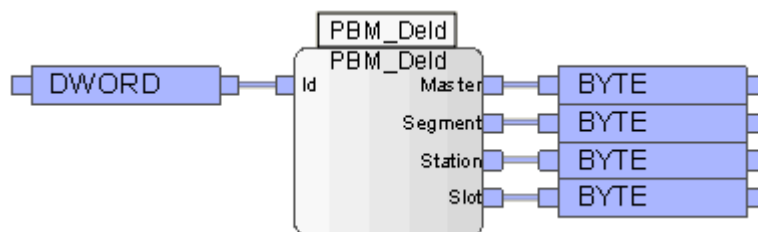


Bild 28: Hilfsfunktionsbaustein DEID

Der Hilfsfunktionsbaustein **DEID** dekodiert die Identifikationsnummer und zerlegt diese in ihre vier Bestandteile.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
Id	Identifikationsnummer des Slaves.	DWORD

Tabelle 106: Eingänge Hilfsfunktionsbaustein DEID

Ausgänge	Beschreibung	Typ
Master	Busadresse des Master.	BYTE
Segment	Segment	BYTE
Station	Busadresse des Slave.	BYTE
Slot	Slot- oder Modul Nummer.	BYTE

Tabelle 107: Ausgänge Hilfsfunktionsbaustein DEID

4.10.4 Hilfsfunktionsbaustein ID
(Identifikationsnummer generieren)

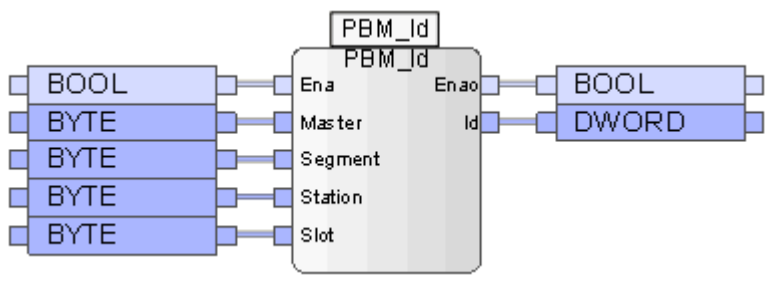


Bild 29: Hilfsfunktionsbaustein ID

Der Hilfsfunktionsbaustein **ID** generiert aus vier Bytes einen Identifier (Identifikationsnummer), die von anderen Hilfsfunktionsbausteinen genutzt wird.

i Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
Ena	Nicht genutzt.	BOOL
Master	Busadresse	BYTE
Segment	Segment	BYTE
Station	Busadresse des Slave.	BYTE
Slot	Slot- oder Modul Nummer.	BYTE

Tabelle 108: Eingänge Hilfsfunktionsbaustein ID

Ausgänge	Beschreibung	Typ
Enao	Nicht genutzt.	BOOL
Id	Identifikationsnummer des Slaves.	DWORD

Tabelle 109: Ausgänge Hilfsfunktionsbaustein ID

4.10.5 Hilfsfunktionsbaustein NSLOT

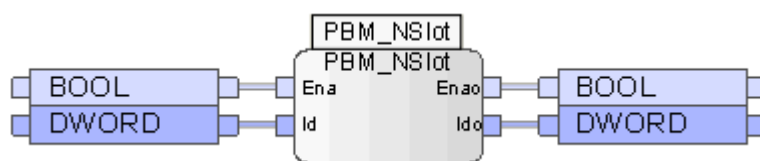


Bild 30: Hilfsfunktionsbaustein NSLOT

Der Hilfsfunktionsbaustein **NSLOT** generiert aus einem Identifier einen neuen Identifier, der den nächsten Slot im gleichen Slave adressiert. Ena muss auf TRUE gesetzt werden, damit der Hilfsfunktionsbaustein läuft.

Enao ist TRUE, wenn das Ergebnis am Ausgang Ido gültig ist.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
Ena	Solange TRUE anliegt, läuft der Baustein.	BOOL
Id	Identifikationsnummer des Slaves.	DWORD

Tabelle 110: Eingänge Hilfsbaustein NSLOT

Ausgänge	Beschreibung	Typ
Enao	TRUE = Ergebnis gültig. FALSE = Keine weiteren Slot-Nummern.	BOOL
Ido	Identifikationsnummer des Slaves.	DWORD

Tabelle 111: Ausgänge Hilfsbaustein NSLOT

4.10.6 Hilfsfunktionsbaustein SLOT

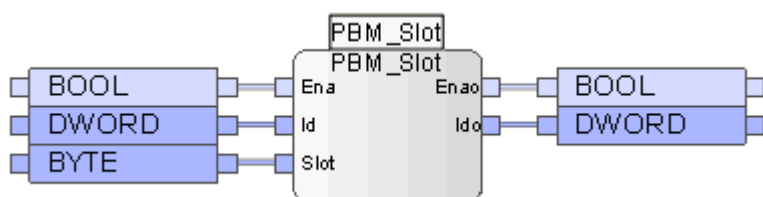


Bild 31: Hilfsfunktionsbaustein SLOT

Der Hilfsfunktionsbaustein **SLOT** generiert aus einem Identifier und einer Slot-Nummer einen neuen Identifier, der den gleichen Slave adressiert, wie der alte Identifier, jedoch mit der neuen Slot-Nummer.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
Ena	Nicht genutzt.	BOOL
Id	Logische Adresse der Slave-Komponente (Slave-ID und Slot-Nummer).	DWORD
Slot	Neue Slot- oder Modul Nummer.	BYTE

Tabelle 112: Eingänge Hilfsbaustein SLOT

Ausgänge	Beschreibung	Typ
Enao	Nicht genutzt.	BOOL
Ido	Identifikationsnummer des Slaves.	DWORD

Tabelle 113: Ausgänge Hilfsbaustein SLOT

4.10.7 Hilfsfunktionsbaustein STDDIAG

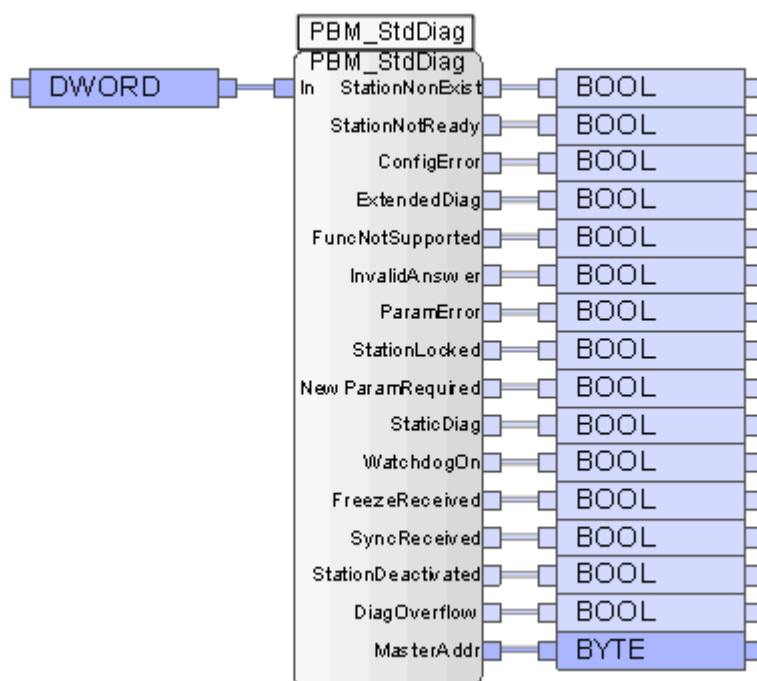


Bild 32: Hilfsfunktionsbaustein STDDIAG

Der Hilfsfunktionsbaustein Standard-Diagnose **STDDIAG** dekodiert die Standard-Diagnose eines PROFIBUS DP Slaves.

Die Ausgänge vom Typ BOOL des Funktionsbaustein **STDDIAG** sind TRUE, wenn das dazugehörige Bit in der Standard-Diagnose gesetzt ist.

i

Zur Konfiguration den Funktionsbaustein per Drag&Drop aus der Bausteinbibliothek in das Anwenderprogramm ziehen, siehe auch Kapitel 5.3.

Eingänge	Beschreibung	Typ
IN	Standard-Diagnose des Slave.	DWORD

Tabelle 114: Eingänge Hilfsbaustein STDDIAG

Ausgänge	Beschreibung	Typ
StationNonExist	Slave existiert nicht.	BOOL
StationNotReady	Slave nicht bereit.	BOOL
ConfigError	Konfigurationsfehler.	BOOL
ExtendedDiag	Erweiterte Diagnose folgt.	BOOL
FuncNotSupported	Funktion nicht unterstützt.	BOOL
InvalidAnswer	Ungültige Antwort vom Slave.	BOOL
ParamError	Parametrierfehler.	BOOL
StationLocked	Slave von anderem Master gesperrt.	BOOL
NewParamRequired	Neue Parametrierdaten erforderlich.	BOOL
StaticDiag	Statische Diagnose.	BOOL
WatchdogOn	Watchdog aktiviert.	BOOL
FreezeReceived	Freeze-Kommando erhalten.	BOOL
SyncReceived	Sync-Kommando erhalten.	BOOL
StationDeactivated	Slave wurde deaktiviert.	BOOL
DiagOverflow	Diagnose überlauf.	BOOL
MasterAddr	Busadresse des Masters.	BYTE

Tabelle 115: Ausgänge Hilfsbaustein STDDIAG

Standard-Diagnose des PROFIBUS DP Slave auslesen

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle, PROFIBUS DP Master**.
2. Rechtsklicken auf **PROFIBUS Slave** und **Edit** wählen.
3. Globale Variable vom Typ DWORD auf das Feld *Standard-Diagnose* ziehen.
4. Diese globale Variable mit dem Eingang des Funktionsbausteins **STDDIAG** verbinden.

4.11 Fehlercodes der Funktionsbausteine

Wenn ein Funktionsbaustein ein Kommando nicht korrekt ausführen konnte, wird am Ausgang **A_Status** ein Fehlercode ausgegeben. Die Bedeutung des Fehlercodes kann der folgenden Tabelle entnommen werden:

Fehlercode	Symbol	Beschreibung
16#40800800	TEMP_NOT_AVAIL	Dienst steht vorübergehend nicht zur Verfügung.
16#40801000	INVALID_PARA	Ungültiger Parameter.
16#40801100	WRONG_STATE	Slave unterstützt kein DP-V1.
16#40808000	FATAL_ERR	Fataler Programmfehler.
16#40808100	BAD_CONFIG	Konfigurationsfehler im Datenbereich.
16#40808200	PLC_STOPPED	Steuerung wurde gestoppt.
16#4080A000	READ_ERR	Fehler beim Lesen eines Records.
16#4080A100	WRITE_ERR	Fehler beim Schreiben eines Records.
16#4080A200	MODULE_FAILURE	Fehler nicht näher spezifizierbar.
16#4080B000	INVALID_INDEX	Index ist ungültig.
16#4080B100	WRITE_LENGTH	Falsche Länge beim Schreiben.
16#4080B200	INVALID_SLOT	Slot-Nummer ist ungültig.
16#4080B300	TYPE_CONFLICT	Falscher Typ.
16#4080B400	INVALID_AREA	Falscher Lese- oder Schreibbereich.
16#4080B500	STATE_CONFLICT	Master im falschen Zustand.
16#4080B600	ACCESS_DENIED	Slave nicht aktiv (oder ähnliches).
16#4080B700	INVALID_RANGE	Falscher Lese- oder Schreibbereich.
16#4080B800	INVALID_PARAMETER	Falscher Parameterwert.
16#4080B900	INVALID_TYPE	Falscher Parametertyp.
16#4080C300	NO_RESOURCE	Slave nicht vorhanden.
16#4080BA00	BAD_VALUE	Ungültiger Wert.
16#4080BB00	BUS_ERROR	Busfehler.
16#4080BC00	INVALID_SLAVE	Ungültige Slave-Id.
16#4080BD00	TIMEOUT	Timeout aufgetreten.
16#4080C000	READ_CONSTRAIN	Lesebeschränkung.
16#4080C100	WRITE_CONSTRAIN	Schreibbeschränkung.
16#4080C200	BUSY	Ein Baustein dieser Art ist bereits aktiv.
16#4080C300	NO_RESOURCE	Slave nicht aktiv.

Tabelle 116: Fehlercodes Funktionsbausteine

4.12 Control Panel (PROFIBUS DP Master)

Im Control Panel kann der Anwender die Einstellungen des PROFIBUS DP Master überprüfen und steuern. Zudem werden aktuelle Statusinformationen (z. B. Zykluszeit, Bus-Zustand usw.) des Masters und der zugehörigen Slaves angezeigt.

Öffnen des Control Panels zur Überwachung des PROFIBUS DP Master

1. Im Strukturbaum **Hardware** und im Kontextmenü **Online** wählen.
2. Im **System-Login**, Zugangsdaten eingeben um die Online Ansicht der Hardware zu öffnen.
3. Doppelklick auf **COM-Modul** und im Strukturbaum **PROFIBUS DP Master** wählen.

4.12.1 Kontextmenü (PROFIBUS Master)

Aus dem Kontextmenü des selektierten PROFIBUS DP Master können die folgenden Kommandos gewählt werden:

Offline

Schaltet den selektierten PROFIBUS DP Master aus. Ist der Master ausgeschaltet, erfolgen keine Aktivitäten.

Stop

Stoppt den selektierten PROFIBUS DP Master. Der PROFIBUS DP Master nimmt weiterhin am Token-Protokoll teil, sendet aber keine Daten an die Slaves.

Clear

Durch Betätigen der Schaltfläche CLEAR wird der selektierte PROFIBUS DP Master in einen sichereren Zustand gesetzt und tauscht nun sichere Daten mit den Slaves aus. Die Ausgangsdaten, die zu den Slaves gesendet werden, enthalten nur Nullen. FailSafe-Slaves erhalten FailSafe-Telegramme, die keine Daten enthalten. Die Eingangsdaten von den Slaves werden vom PROFIBUS DP Master ignoriert und stattdessen Initialwerte im Anwenderprogramm verwendet.

Operate

Startet den selektierten PROFIBUS DP Master. Der PROFIBUS DP Master tauscht zyklisch E/A-Daten mit den Slaves aus.

Zurücksetzen Statistik

Die Schaltfläche *zurücksetzen Statistik* setzt die statistischen Daten (Zykluszeit min, max usw.) auf null zurück.

4.12.2 Kontextmenü (PROFIBUS DP Slave)

Aus dem Kontextmenü des selektierte PROFIBUS DP Slave können die folgenden Kommandos gewählt werden:

Aktivieren

Aktiviert den selektierten Slave, der mit dem PROFIBUS DP Master nun Daten austauschen kann.

Deaktivieren

Deaktiviert den selektierten Slave. Die Kommunikation wird beendet.

4.12.3 Anzeigefeld (PROFIBUS Master)

In dem Anzeigefeld werden die folgenden Werte des selektierten PROFIBUS DP Master angezeigt.

Element	Beschreibung
Name	Name des PROFIBUS DP Master.
Master-Zustand	Zeigt den momentanen Protokollzustand an, siehe Kapitel 4.12.4. 0 = OFFLINE 1 = STOP 2 = CLEAR 3 = OPERATE 100 = UNDEFINED
Bus-Zustand	Fehlercode Busfehler 0 ... 6: 0 = OK 1 = Adressfehler: die Adresse des Masters ist auf dem Bus bereits vorhanden. 2 = Busstörung: Es wurde eine Störung auf dem Bus registriert, z. B. Bus nicht richtig abgeschlossen, mehrere Teilnehmer senden gleichzeitig. 3 = Protokollfehler: Ein fehlerhaft codiertes Paket wurde empfangen. 4 = Hardwarefehler: Die Hardware hat einen Fehler gemeldet, z. B. bei zu knapp eingestellten Zeiten. 5 = Unbekannter Fehler: Master hat Zustand aus unbekanntem Grund gewechselt. 6 = Controller Reset: Bei schweren Busstörungen geht mitunter der Controller-Chip in einen undefinierten Zustand und wird zurückgesetzt. Der Fehlercode bleibt solange anliegen, bis der Busfehler behoben ist.
Feldbus-Schnittstelle	FB1, FB2
µP-Last (projektiert) [%]	Anzeige der projektierten Auslastung des COM-Moduls für dieses Protokoll.
µP-Last (tatsächliche) [%]	Tatsächliche Auslastung des COM-Moduls für dieses Protokoll.
Baudrate [bps]	Baudrate des Masters. Der Master kann mit allen Baudraten, die im Standard spezifiziert sind kommunizieren. Zykluszeiten sind bis zu einer Untergrenze von 2 ms möglich.
Feldbusadresse	Busadresse des Master (0 ... 125).
PNO-IdentNo	Von der PROFIBUS DP Nutzerorganisation e.V. zugeteilte 16-Bit-Nummer, die ein Produkt (Feldgerät) eindeutig kennzeichnet.
Anzahl BusFehler	Anzahl der Busfehler bisher.
MSI [ms]	Min. Slave Interval in ms, Auflösung 0.1 ms.
TTR [ms]	Target Rotation Time in ms, Auflösung 0.1 ms.

Element	Beschreibung
Letzte Zykluszeit [ms]	Letzte PROFIBUS DP Zykluszeit [ms].
Minimale Zykluszeit [ms]	Minimale PROFIBUS DP Zykluszeit [ms].
Mittlere Zykluszeit [ms]	Mittlere PROFIBUS DP Zykluszeit [ms].
Maximale Zykluszeit [ms]	Maximale PROFIBUS DP Zykluszeit [ms].

Tabelle 117: Anzeigenfeld PROFIBUS Master

4.12.4 Zustand des PROFIBUS DP Master

Der Zustand des Masters wird im Anzeigenfeld des Control Panel angezeigt und kann mit der Statusvariable Master Verbindungszustand im Anwenderprogramm ausgewertet werden.

Master Zustand	Master Zustand
OFFLINE	Der Master ist ausgeschaltet und es erfolgen keine Busaktivitäten.
STOP	Der Master nimmt am Token-Protokoll teil, sendet aber keine Daten an die Slaves.
CLEAR	Der Master ist im sicheren Zustand und tauscht Daten mit den Slaves aus. <ul style="list-style-type: none"> Die Ausgangsdaten, die zu den Slaves gesendet werden, enthalten nur Nullen. Die FailSafe-Slaves erhalten FailSafe-Telegramme (diese enthalten keine Daten). Die Eingangsdaten von den Slaves werden ignoriert und stattdessen werden Initialwerte verwendet.
OPERATE	Der Master ist im Arbeitsmodus und tauscht zyklisch E/A-Daten mit den Slaves aus.
UNDEFINED	Firmware-Update des PROFIBUS DP Master Moduls läuft gerade.

Tabelle 118: PROFIBUS DP Master Zustand

4.12.5 Verhalten des PROFIBUS DP Master

Verhalten des PROFIBUS DP Master in Abhängigkeit vom Betriebszustand der Steuerung.

Zustand der Steuerung	Verhalten des HIMA PROFIBUS DP Master
STOPP ¹⁾	Steuerung in STOPP, dann ist der Master im Zustand OFFLINE.
RUN	Steuerung in RUN, der Master versucht in den Zustand OPERATE zu gelangen.
STOPP	Steuerung in STOPP, der Master geht in den Zustand CLEAR. Ist der Master bereits in STOPP oder OFFLINE, bleibt er in diesem Zustand.
¹⁾ Nach Einschalten der Steuerung oder nach Laden der Konfiguration.	

Tabelle 119: PROFIBUS DP Master Verhalten

4.12.6 Funktion der FBx LED beim PROFIBUS DP Master

Den Zustand des PROFIBUS DP Protokolls signalisiert die FBx LED der zugeordneten Feldbus-Schnittstelle. Die Zustände der FBx LED sind in der folgenden Tabelle dargestellt:

FBx LED	Beschreibung
AUS	Keine oder ungültige Konfiguration des PROFIBUS DP Master.
Blinkt im 2 Sekundentakt	Gültige Konfiguration. Der PROFIBUS DP Master befindet sich im Zustand OFFLINE oder STOPP.
AN	Der PROFIBUS DP Master befindet sich im Zustand OPERATE oder CLEAR, wenn alle aktivierten Slaves verbunden sind.
Blinkt im Sekundentakt	Mindestens ein Slave ist ausgefallen.

Tabelle 120: LED FBx

4.12.7 Funktion der FAULT LED beim PROFIBUS DP Master (Nur HIMax)

Die Störung des PROFIBUS DP Protokolls signalisiert die FAULT LED der zugeordneten Feldbus-Schnittstelle. Die Zustände der FAULT LED sind in der folgenden Tabelle dargestellt:

FAULT LED	Farbe	Beschreibung
AUS	Rot	PROFIBUS DP Protokoll ohne Störung.
Blinkt	Rot	Protokoll gestört. <ul style="list-style-type: none"> Mindestens ein Slave ist ausgefallen. Busfehler erkannt. Überschreitung des Rechenzeitbudgets. Tritt länger als 5 Sekunden kein Fehlerereignis auf, so wechselt die Anzeige in den Zustand <i>Protokoll ohne Störung</i> .

Tabelle 121: FAULT FBx

4.13 HIMA PROFIBUS DP Slave

Dieses Kapitel beschreibt die Eigenschaften des HIMA PROFIBUS DP Slave und die Menüfunktionen in SILworX, die zur Konfiguration des HIMA PROFIBUS DP Slave benötigt werden.

Benötigte Ausstattung und Systemanforderungen

Element	Beschreibung
HIMA Steuerung	HIMax mit X-COM 01 Modul HIQuad X mit F-COM 01 Modul HIMatrix ab CPU BS V7 und COM BS V12
COM-Modul	Das COM-Modul muss an der verwendeten seriellen Feldbus-Schnittstelle (FB1 oder FB2) mit einem optionalen HIMA PROFIBUS DP Slave Submodul ausgerüstet sein. Schnittstellenbelegung, siehe Kommunikationshandbuch HI 801 100 D.
Aktivierung	Freischaltung durch Aufsteckmodul, siehe Kommunikationshandbuch 801 100 D.

Tabelle 122: Ausstattung und Systemanforderungen HIMA PROFIBUS DP Slave

i

HIMA empfiehlt für HIMax, PROFIBUS DP über die Feldbus-Schnittstelle FB1 (Übertragungsrate maximal 12 MBit) zu betreiben. Über die Feldbus-Schnittstelle FB2 ist eine maximale Übertragungsrate von 1,5 MBit zugelassen.

PROFIBUS DP Slave Eigenschaften

Element	Beschreibung	
Typ des HIMA PROFIBUS DP Slaves	DP-V0	
Übertragungsrate	9,6 kbit/s ... 12 Mbit/s	
Busadresse	0 ... 125	
Max. Anzahl Slaves	Es kann für jedes COM-Modul ein HIMA PROFIBUS DP Slave konfiguriert werden.	
Prozessdatenmenge eines HIMA PROFIBUS DP Slaves	HIMax und HIMatrix	DP-Output: max. 192 Bytes DP-Input: max. 240 Bytes Insgesamt jedoch: max. 256 Bytes
	HIQuad X	DP-Output: max. 244 Bytes DP-Input: max. 244 Bytes Insgesamt: max. 488 Bytes
Protokoll Watchdog	Ist die COM im Zustand RUN und die Verbindung zum PROFIBUS DP Master geht verloren, dann wird dies vom DP Slave nach Ablauf des Watchdog-Timeouts (muss im Master parametrisiert werden) erkannt. In diesem Fall werden die DP-Output-Daten (Eingangsdaten aus Sicht der Ressource) auf ihren Initialwert zurückgesetzt und das <i>Daten gültig</i> -Flag (Statusvariable des DP Slave Protokolls) auf FALSE gesetzt.	

Tabelle 123: Eigenschaften HIMA PROFIBUS DP Slave

4.13.1 Anlegen eines HIMA PROFIBUS DP Slave

Einen neuen HIMA PROFIBUS DP Slave anlegen

1. Im Strukturbaum **Konfiguration**, **Ressource**, **Protokolle** öffnen.
2. Im Kontextmenü von Protokolle **Neu**, **PROFIBUS DP Slave** wählen um einen neuen PROFIBUS DP Slave hinzuzufügen.
3. Im Kontextmenü vom PROFIBUS DP Slave **Edit** wählen.

4. Im Register **Eigenschaften** das **Modul** und die **Schnittstelle** wählen.

4.14 Menüfunktionen PROFIBUS DP Slave

4.14.1 Edit

Das Dialogfenster **Edit** des PROFIBUS DP Master enthält die folgenden Register.

Prozessvariablen

Im Register **Prozessvariablen** werden die Variablen zum Empfangen und Senden angelegt.

Eingangsvariablen

Die Variablen, die von dieser Steuerung Empfangen werden sollen, werden im Bereich *Eingangssignale* eingetragen.

Im Bereich *Eingangssignale* können beliebige Variablen angelegt werden. Die Offsets und Typen der Variablen müssen allerdings identisch mit den Offsets und den Typen der Sendevariablen des Kommunikationspartners sein.

Ausgangsvariablen

Die Variablen für den zyklischen Datenaustausch, die von dieser Steuerung gesendet werden sollen, werden im Bereich *Ausgangssignale* eingetragen.

Im Bereich *Ausgangssignale* können beliebige Variablen angelegt werden. Die Offsets und Typen der Variablen müssen allerdings identisch mit den Offsets und den Typen der Empfangsvariablen des Kommunikationspartners sein.

Systemvariablen

Im Register **Systemvariablen** wird festgelegt, welche Variablen in die Steuerung eingelesen werden sollen.

Das Register **Systemvariablen** stellt die folgenden Systemvariablen bereit, die es erlauben, den Zustand des PROFIBUS DP Slave im Anwenderprogramm auszuwerten.

Element	Beschreibung
Aktuelle Baudrate	Baudrate, mit der das PROFIBUS DP Slave Protokoll aktuell arbeitet.
Daten gültig	<p>Ist die Statusvariable <i>Daten gültig</i> auf TRUE gesetzt, dann hat der Slave gültige Import-Daten des Masters empfangen. Die Statusvariable wird auf FALSE gesetzt wenn die Watchdog-Zeit beim Slave abgelaufen ist.</p> <p>Standardwert: FALSE</p> <p>Hinweise:</p> <ul style="list-style-type: none"> ▪ Eine mit der Systemvariablen <i>Daten gültig</i> belegte globale Variable darf nicht den Initialwert TRUE haben. ▪ Wurde der Watchdog des Slaves durch den Master nicht aktiviert und die Verbindung geht verloren, so behält die Statusvariable <i>Daten gültig</i> den Wert TRUE, da der PROFIBUS DP Slave keine Möglichkeit hat, den Verbindungsverlust zu erkennen. Dieser Umstand ist bei der Verwendung dieser Variablen unbedingt zu beachten!

Element	Beschreibung
Fehlercode	Ist in dem PROFIBUS DP Slave Protokoll ein Fehler aufgetreten, so wird dies in dieser Variablen übertragen. Es wird jeweils der aktuell aufgetretene Fehler angezeigt. Mögliche Werte (hexadezimal) sind: 0x00: Kein Fehler. 0xE1: Falsche Parametrierung durch den PROFIBUS DP Master. 0xD2: Falsche Konfigurierung durch den PROFIBUS DP Master. Standardwert: 0x00
Master-Adresse	Dies ist die Adresse des PROFIBUS DP Master, der den eigenen PROFIBUS DP Slave parametrieren und konfigurieren hat. Mögliche Werte (dezimal) sind: 0-125: Adresse des Masters. 255: Slave ist aktuell keinem Master zugeordnet. Standardwert: 255
Protokollzustand	Beschreibt den Zustand des PROFIBUS DP Slave Protokolls. Mögliche Werte (hexadezimal) sind: 0xE1: Die Steuerung ist vom Bus getrennt bzw. nicht aktiv. 0xD2: Die Steuerung wartet auf eine Konfigurierung durch den PROFIBUS DP Master. 0xC3: Die Steuerung tauscht zyklisch Daten mit dem PROFIBUS DP Master aus. Standardwert: 0xE1
Slave-Adresse	Dies ist die Adresse des PROFIBUS DP Slave der Steuerung. Diese Adresse wurde zuvor vom Anwender durch das PADT konfiguriert. Mögliche Werte (dezimal) sind: 0-125: Adresse des Slaves
Watchdog-Zeit	Im Master parametrisierte Watchdog-Zeit in Millisekunden, siehe Kapitel 4.6.3.

Tabelle 124: Systemvariablen PROFIBUS DP Slave

4.14.2 Eigenschaften

Das Register **Eigenschaften** des HIMA PROFIBUS DP Slave enthält die folgenden Parameter zur Konfiguration des PROFIBUS DP Slave.

Die Vorgabewerte der Parameter *In einem Zyklus* und *Aktualisierungsintervall der Prozessdaten [ms]* sorgen für einen schnellen Datenaustausch der PROFIBUS DP Daten zwischen dem COM-Modul (COM) und der PROFIBUS DP Slave Hardware der HIMA Steuerung.

Diese Parameter sollten nur dann geändert werden, wenn eine Reduzierung der COM-Auslastung für eine Anwendung erforderlich ist und der Prozess dies zulässt.

i

Die Änderung der Parameter wird nur dem erfahrenen Programmierer empfohlen.
Eine Erhöhung der Aktualisierungszeit der COM/ PROFIBUS DP Hardware bedeutet auch, dass die tatsächliche Aktualisierungszeit der PROFIBUS DP Daten erhöht wird. Die Zeitanforderungen der Anlage sind zu prüfen.

Mit dem Parameter **Min. Slave Intervall [ms]**, wird die minimale Aktualisierungszeit der PROFIBUS DP Daten zwischen PROFIBUS DP Master und PROFIBUS DP Slave festgelegt, siehe Kapitel 4.3.2 im Register Zeiten.

Element	Beschreibung																																												
Typ	PROFIBUS DP Slave.																																												
Name	Name für den PROFIBUS DP Slave.																																												
Modul	Auswahl des COM-Moduls, auf dem dieses Protokoll abgearbeitet wird.																																												
Max. µP-Budget aktivieren	Aktiviert : Limit des µP-Budget aus dem Feld Max. µP-Budget in [%] übernehmen. Deaktiviert: Kein Limit des µP-Budget für dieses Protokoll verwenden.																																												
Max. µP-Budget in [%]	Maximales µP-Budget des Moduls, welches bei der Abarbeitung des Protokolls produziert werden darf. Wertebereich: 1 ... 100% Standardwert: 30%																																												
Verhalten bei CPU/COM Verbindungsverlust	Bei Verbindungsverlust des Prozessormoduls zum Kommunikationsmodul werden in Abhängigkeit dieses Parameters die Eingangsvariablen entweder initialisiert oder unverändert im Prozessormodul verwendet. (z. B. wenn Kommunikationsmodul bei laufender Kommunikation gezogen wird). Soll ein Projekt von kleiner SILworX V3 konvertiert werden, muss dieser Wert auf "Letzten Wert beibehalten" gesetzt sein, um den CRC nicht zu ändern. Für HIMatrix Steuerungen kleiner CPU BS V8 muss dieser Wert immer auf "Letzten Wert beibehalten" gesetzt sein. <table><tr><td>Initialdaten annehmen</td><td>Eingangsvariablen werden auf die Initialwerte zurückgesetzt.</td></tr><tr><td>Letzten Wert beibehalten</td><td>Eingangsvariablen behalten den letzten Wert.</td></tr></table>	Initialdaten annehmen	Eingangsvariablen werden auf die Initialwerte zurückgesetzt.	Letzten Wert beibehalten	Eingangsvariablen behalten den letzten Wert.																																								
Initialdaten annehmen	Eingangsvariablen werden auf die Initialwerte zurückgesetzt.																																												
Letzten Wert beibehalten	Eingangsvariablen behalten den letzten Wert.																																												
Stationsadresse	Stationsadresse des Slaves. Die Stationsadresse des Slaves darf auf dem Bus nur einmal vorhanden sein. Wertebereich: 1 ... 125 Standardwert: 0																																												
Schnittstelle	Feldbus-Schnittstelle, die für den PROFIBUS DP Slave benutzt werden soll. Wertebereich: fb1, fb2 Standardwert: Keine																																												
Baudrate [bps]	Baudrate mit welcher der Bus betrieben wird. Mögliche Werte: <table><tr><th>Wert</th><th>Baudrate</th><th>FB1</th><th>FB2</th></tr><tr><td>9600</td><td>9,6 kbit/s</td><td>X</td><td>X</td></tr><tr><td>19200</td><td>16,2 kbit/s</td><td>X</td><td>X</td></tr><tr><td>45450</td><td>45,45 kbit/s</td><td>X</td><td>X</td></tr><tr><td>93750</td><td>93,75 kbit/s</td><td>X</td><td>X</td></tr><tr><td>187500</td><td>187,5 kbit/s</td><td>X</td><td>X</td></tr><tr><td>500000</td><td>500 kbit/s</td><td>X</td><td>X</td></tr><tr><td>1500000</td><td>1,5 Mbit/s</td><td>X</td><td>X</td></tr><tr><td>3000000</td><td>3 Mbit/s</td><td>X</td><td>-</td></tr><tr><td>6000000</td><td>6 Mbit/s</td><td>X</td><td>-</td></tr><tr><td>12000000</td><td>12 Mbit/s</td><td>X</td><td>-</td></tr></table>	Wert	Baudrate	FB1	FB2	9600	9,6 kbit/s	X	X	19200	16,2 kbit/s	X	X	45450	45,45 kbit/s	X	X	93750	93,75 kbit/s	X	X	187500	187,5 kbit/s	X	X	500000	500 kbit/s	X	X	1500000	1,5 Mbit/s	X	X	3000000	3 Mbit/s	X	-	6000000	6 Mbit/s	X	-	12000000	12 Mbit/s	X	-
Wert	Baudrate	FB1	FB2																																										
9600	9,6 kbit/s	X	X																																										
19200	16,2 kbit/s	X	X																																										
45450	45,45 kbit/s	X	X																																										
93750	93,75 kbit/s	X	X																																										
187500	187,5 kbit/s	X	X																																										
500000	500 kbit/s	X	X																																										
1500000	1,5 Mbit/s	X	X																																										
3000000	3 Mbit/s	X	-																																										
6000000	6 Mbit/s	X	-																																										
12000000	12 Mbit/s	X	-																																										

Element	Beschreibung
Aktualisierungsintervall der Prozessdaten [ms]	Aktualisierungszeit in Millisekunden, mit der die Daten des Protokolls zwischen COM und der PROFIBUS DP Slave Hardware ausgetauscht werden. Wertebereich: 4 ... 1000 Standardwert: 10
Prozessdaten-Konsistenz erzwingen	Aktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM innerhalb eines Zyklus der CPU. Deaktiviert: Transfer der gesamten Daten des Protokolls von der CPU zur COM, verteilt über mehrere CPU Zyklen zu je 1100 Byte pro Datenrichtung. Damit kann eventuell auch die Zykluszeit der Steuerung reduziert werden. Standardwert: Aktiviert

Tabelle 125: Slave Eigenschaften: Register Allgemein

4.15 Control Panel (PROFIBUS DP Slave)

Im Control Panel kann der Anwender die Einstellungen des PROFIBUS DP Slave überprüfen und steuern. Zudem werden aktuelle Statusinformationen (z. B. Zykluszeit, Bus-Zustand usw.) des Slaves angezeigt.

Öffnen des Control Panels zur Überwachung des PROFIBUS DP Slave

1. Im Strukturbaum **Hardware** und im Kontextmeü **Online** wählen.
2. Im **System-Login**, Zugangsdaten eingeben um die Online Ansicht der Hardware zu öffnen.
3. Doppelklick auf **COM-Modul** und im Strukturbaum **PROFIBUS DP Slave** wählen.

4.15.1 Anzeigefeld (PROFIBUS DP Slave)

In dem Anzeigefeld werden die folgenden Werte des selektierten PROFIBUS DP Slave angezeigt.

Element	Beschreibung
Name	Name des PROFIBUS DP Slave.
Feldbus-Schnittstelle	Zugeordnete Feldbus-Schnittstelle des Slave.
Protokoll-Zustand	Verbindungszustand 0 = Deaktiviert, 1 = Inaktiv (versucht Verbindung aufzunehmen), 2 = Verbunden
Fehler-Zustand	Siehe Kapitel 4.14.1.
Timeout	Im Master parametrisierte Watchdog-Zeit in Millisekunden. Siehe Kapitel 4.6.3
Watchdog-Zeit [ms]	Wird im Master eingestellt. Siehe Kapitel 4.6.3.
Feldbusadresse	Siehe Kapitel 4.14.2.
Masteradresse	Adresse des PROFIBUS DP Master.
Baudrate [bps]	Aktuelle Baudrate. Siehe Kapitel 4.14.2.
µP-Budget (projektiert) [%]	Anzeige der projektierten Auslastung des COM-Moduls für dieses Protokoll.
µP-Budget (tatsächliche) [%]	Tatsächliche Auslastung des COM-Moduls für dieses Protokoll.

Tabelle 126: Anzeigefeld (PROFIBUS DP Slave)

4.16 Funktion der FBx LED beim PROFIBUS DP Slave

Den Zustand des PROFIBUS DP Protokolls signalisiert die FBx LED der zugeordneten Feldbus-Schnittstelle. Die Zustände der FBx LED sind in der folgenden Tabelle dargestellt:

FBx LED	Farbe	Beschreibung
AUS	Gelb	Das PROFIBUS DP Protokoll ist nicht aktiv! Das bedeutet, dass die Steuerung im Zustand STOPP ist oder es ist kein PROFIBUS DP Slave konfiguriert.
Blinkt im 2 Sekunden Takt	Gelb	Kein Datenaustausch! Der PROFIBUS DP Slave ist konfiguriert und bereit.
AN	Gelb	Das PROFIBUS DP Protokoll ist aktiv und befindet sich im Datenaustausch mit dem PROFIBUS DP Master.

Tabelle 127: LED FBx

4.17 Funktion der FAULT LED beim PROFIBUS DP Slave (Nur HIMax)

Die Störung des PROFIBUS DP Protokolls signalisiert die FAULT LED der zugeordneten Feldbus-Schnittstelle. Die Zustände der FAULT LED sind in der folgenden Tabelle dargestellt:

FAULT LED	Farbe	Beschreibung
AUS	Rot	PROFIBUS DP Protokoll ohne Störung.
Blinkt	Rot	Protokoll gestört. <ul style="list-style-type: none"> Die Konfigurationen des PROFIBUS DP Masters und/oder des PROFIBUS DP Slaves sind fehlerhaft. Überschreitung des Rechenzeitbudgets. Tritt länger als 5 Sekunden kein Fehlerereignis auf, so wechselt die Anzeige in den Zustand <i>Protokoll ohne Störung</i> .

Tabelle 128: FAULT FBx

5 Allgemein

In diesem Kapitel sind Parameter gesammelt, die für alle Kommunikationsprotokolle relevant sind.

5.1 Maximale Kommunikationszeitscheibe

Die maximale Kommunikationszeitscheibe ist die zugeteilte Zeit in Millisekunden (ms) pro CPU-Zyklus, innerhalb der das Prozessormodul die Kommunikationsaufgaben abarbeitet. Wenn die Protokollverarbeitung innerhalb der Dauer einer Kommunikationszeitscheibe nicht beendet werden konnte, führt die CPU dennoch die sicherheitsrelevanten Überwachungen für alle Protokolle in einem CPU-Zyklus aus.

i

Wenn nicht alle in einem CPU-Zyklus anstehenden Kommunikationsaufgaben ausgeführt werden können, erfolgt die komplette Übertragung der Kommunikationsdaten über mehrere CPU-Zyklen. Die Anzahl der Kommunikationszeitscheiben ist dann größer 1.

Für die Berechnungen der zulässigen maximalen Reaktionszeiten gilt die Bedingung, dass die Anzahl der Kommunikationszeitscheiben genau 1 ist.

5.1.1 Ermitteln der maximalen Dauer der Kommunikationszeitscheibe

Für eine erste Abschätzung der maximalen Dauer der Kommunikationszeitscheibe müssen die folgenden Zeiten aufsummiert und das Ergebnis in den Systemparameter *Max. Kom.-Zeitscheibe [ms]* in den Eigenschaften der Ressource eingetragen werden:

- Pro COM-Modul 3 ms.
- Pro redundante **safeethernet** Verbindung 1 ms.
- Pro nicht redundante **safeethernet** Verbindung 0,5 ms.
- Pro KByte Nutzdaten bei nichtsicheren Protokollen (z. B. Modbus) 1 ms.

HIMA empfiehlt, den abgeschätzten Wert *Max. Kom.-Zeitscheibe [ms]* mit dem im Control Panel angezeigten Wert zu vergleichen und gegebenenfalls in den Eigenschaften der Ressource zu korrigieren. Dies kann z. B. in einem FAT (Factory Acceptance Test) oder SAT (Site Acceptance Test) durchgeführt werden.

Ermitteln der tatsächlichen Dauer der maximalen Kommunikationszeitscheibe

1. Das HIMA System unter voller Last betreiben (FAT, SAT):
Alle Kommunikationsprotokolle sind in Betrieb (**safeethernet** und Standardprotokolle).
2. Das **Control Panel** öffnen und im Strukturbaum das Verzeichnis **Kom.-Zeitscheibe** wählen.
3. Anzeige *Maximale Kom.-Zeitscheibe Dauer pro Zyklus [ms]* auszulesen.
4. Anzeige *Maximale Anzahl benötigter Kom.-Zeitscheibe Zyklen* auszulesen.

Die Dauer der Kommunikationszeitscheibe ist so hoch einzustellen, dass der CPU-Zyklus die vom Prozess vorgegebene Watchdog-Zeit nicht überschreiten kann, wenn er die eingestellte Kommunikationszeitscheibe ausnutzt.

5.2 Lastbegrenzung

Für jedes Kommunikationsprotokoll kann ein Rechenzeitbudget in % (*μP-Budget*) vorgegeben werden. So kann die verfügbare Rechenzeit zwischen den konfigurierten Protokollen verteilt werden. Die Summe der Rechenzeitbudgets aller parametrisierten Kommunikationsprotokolle eines CPU- oder COM-Moduls darf nicht größer als 100 % sein.

Die festgelegten Rechenzeitbudgets der einzelnen Kommunikationsprotokolle werden überwacht. Hat ein Kommunikationsprotokoll sein Rechenzeitbudget erreicht oder überschritten und es steht keine zusätzliche Rechenzeit als Reserve zur Verfügung, so wird das Kommunikationsprotokoll nicht komplett abgearbeitet.

Wenn noch genügend zusätzliche Rechenzeit vorhanden ist, wird diese verwendet, um ein Kommunikationsprotokoll, das sein Rechenzeitbudget erreicht oder überschritten hat noch abzuarbeiten. Dadurch kann es vorkommen, dass ein Kommunikationsprotokoll tatsächlich ein höheres Rechenzeitbudget verwendet als ihm zugeteilt wurde.

Eventuell werden über 100 % Rechenzeitbudget online angezeigt. Dies ist kein Fehler, das Rechenzeitbudget über 100 % ist die zusätzlich verwendete Rechenzeit.

i

Das zusätzliche Rechenzeitbudget ist keinesfalls eine Zusicherung für ein bestimmtes Kommunikationsprotokoll und kann jederzeit vom System zurückgenommen werden.

5.3 Konfiguration der Funktionsbausteine

Die Feldbus-Protokolle und die zugehörigen Funktionsbausteine laufen auf dem COM-Modul des HIMA Systems. Daher müssen diese Funktionsbausteine im SILworX Strukturbaum unter **Konfiguration, Ressource, Protokolle...** angelegt werden.

Um diese Funktionsbausteine auf dem COM-Modul zu steuern, können im Anwenderprogramm von SILworX Funktionsbausteine angelegt werden (siehe Kapitel 5.3.1), die wie Standard-Funktionsbausteine verwendet werden können.

Die Verbindung der Funktionsbausteine im Anwenderprogramm von SILworX mit den entsprechenden Funktionsbausteinen im Strukturbaum von SILworX erfolgt über gemeinsame Variablen. Diese müssen zuvor vom Anwender im Variablen-Editor erstellt werden.

5.3.1 Beschaffung der Funktionsbausteinbibliotheken

Die Funktionsbausteinbibliotheken für PROFIBUS DP und TCP Send/Receive müssen über die Funktion *Wiederherstellen...* (Kontextmenü des Projekts) dem Projekt hinzugefügt werden.

Die Funktionsbausteinbibliothek ist auf Anfrage über den HIMA Support erhältlich.

5.3.2 Konfiguration der Funktionsbausteine im Anwenderprogramm

Die benötigten Funktionsbausteine können per Drag&Drop in das Anwenderprogramm kopiert werden. Die Eingänge und Ausgänge sind nach der Beschreibung des jeweiligen Funktionsbausteins zu konfigurieren.

Oberer Teil des Funktionsbausteins

Der obere Teil des Funktionsbausteins entspricht der Benutzerschnittstelle, über die der Funktionsbaustein vom Anwenderprogramm gesteuert wird.

Hier werden die Variablen verbunden, die im Anwenderprogramm verwendet werden. Das Präfix "A" steht für "Applikation".

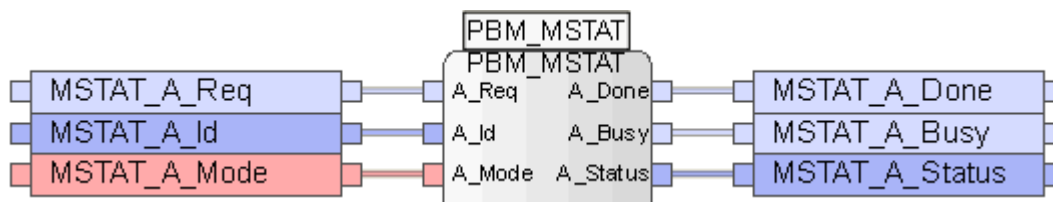


Bild 33: Funktionsbaustein PNM_MSTST (oberer Teil)

Unterer Teil des Funktionsbausteins

Der untere Teil des Funktionsbausteins stellt die Verbindung zum Funktionsbaustein (im Strukturbaum von SILworX) dar.

Hier werden die Variablen verbunden, die mit dem Funktionsbaustein im Strukturbaum von SILworX verbunden werden müssen. Das Präfix "F" steht für "Field".

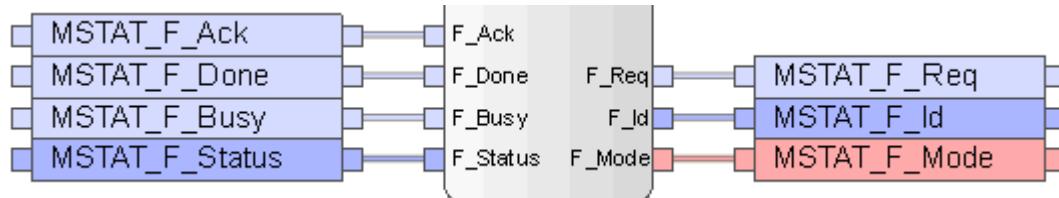


Bild 34: Funktionsbaustein PNM_MSTST (unterer Teil)

5.3.3 Konfiguration der Funktionsbausteine im Strukturbaum von SILworX

Den Funktionsbaustein im Strukturbaum von SILworX anlegen:

1. Im Strukturbaum **Konfiguration, Ressource, Protokolle**, z. B. **PROFIBUS Master** wählen.
2. Rechtsklick auf **Funktionsbausteine** und **Neu** wählen.
3. Den passenden Funktionsbaustein (im Strukturbaum von SILworX) auswählen.

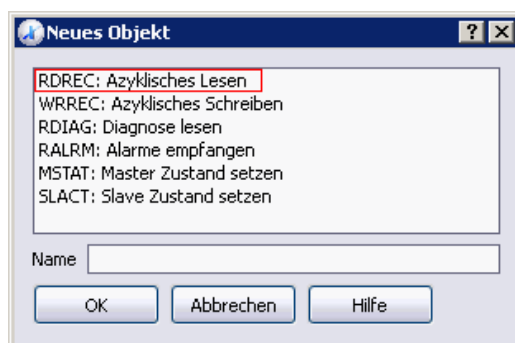


Bild 35: Auswahl Funktionsbausteine

Die Eingänge des Funktionsbausteins (Häkchen in Spalte Eingangsvariable) müssen mit den gleichen Variablen verbunden werden, die mit den *F-Ausgängen* des Funktionsbausteins im Anwenderprogramm verbunden sind.

Die Ausgänge des Funktionsbausteins (kein Häkchen in Spalte Eingangsvariable) müssen mit den gleichen Variablen verbunden werden, die mit den *F-Eingängen* des Funktionsbausteins im Anwenderprogramm verbunden sind.

Systemvariablen					
F	Name	Datentyp	Transfer-Operation	Eingangsvariable	Globale Variable
1	ACK	BOOL		5 <input checked="" type="checkbox"/>	MSTAT_F_Ack
2	BUSY	BOOL		5 <input checked="" type="checkbox"/>	MSTAT_F_Busy
3	DONE	BOOL		5 <input checked="" type="checkbox"/>	MSTAT_F_Done
4	REQ	BOOL		5 <input type="checkbox"/>	MSTAT_F_Req
5	M_ID	DWORD		5 <input type="checkbox"/>	MSTAT_F_Id
6	STATUS	DWORD		5 <input checked="" type="checkbox"/>	MSTAT_F_Status
7	MODE	INT		5 <input type="checkbox"/>	MSTAT_F_Mode

Bild 36: Systemvariablen des Funktionsbausteins MSTAT

Anhang

Glossar

Begriff	Beschreibung
ARP	Address Resolution Protocol: Netzwerkprotokoll zur Zuordnung von Netzwerkadressen zu Hardwareadressen.
Bit-Variable	Variable, die bitweise adressiert wird.
CENELEC	Comité Européen de Normalisation Électrotechnique (Europäisches Komitee für elektrotechnische Normung)
Connector Board	Anschlusskarte für HIMax Modul.
COM	Kommunikationsmodul
CPU	Prozessormodul
CRC	Cyclic Redundancy Check, Prüfsumme
Dataview	Einer Dataview sind die Globalen Variablen für Eingangs- und Ausgangsdaten für den Zugriff durch Modbus-Quellen zugeordnet.
EN	Europäische Normen
Exportbereich	Als Exportbereich wird die Prozessdatenmenge bezeichnet, die vom System (aus einem Anwenderprogramm, HW-Eingang oder einem anderen Protokoll) geschrieben und vom Modbus-Master gelesen werden kann.
FB	Feldbus
FBS	Funktionsbausteinsprache
ICMP	Internet Control Message Protocol: Netzwerkprotokoll für Status- und Fehlermeldungen.
IEC	Internationale Normen für die Elektrotechnik.
Importbereich	Als Importbereich wird die Prozessdatenmenge bezeichnet, die vom Modbus-Master geschrieben wird und als Eingangsdaten für das System (in einem Anwenderprogramm, HW-Ausgang oder einem anderen Protokoll) verwendet werden kann.
KE	Kommunikationsendpunkt
MAC-Adresse	Hardware-Adresse eines Netzwerkanschlusses (Media Access Control).
NSIP	Nicht-sicherheitsbezogenes Protokoll.
PADT	Programming and Debugging Tool (nach IEC 61131-3), PC mit SILworX.
PE	Schutzerde
PELV	Protective Extra Low Voltage: Funktionskleinspannung mit sicherer Trennung.
PES	Programmierbares Elektronisches System
R	Read
Rack-ID	Identifikation eines Basisträgers (Nummer).
rückwirkungsfrei	Es seien zwei Eingangsschaltungen an dieselbe Quelle (z. B. Transmitter) angeschlossen. Dann wird eine Eingangsschaltung „rückwirkungsfrei“ genannt, wenn sie die Signale der anderen Eingangsschaltung nicht verfälscht.
R/W	Read/Write
Register-Variable	Variable, die wortweise adressiert wird.
SB	Systembusmodul
SFF	Safe Failure Fraction, Anteil der sicher beherrschbaren Fehler.
SIF	Sicherheitstechnische Funktion
SIL	Safety Integrity Level (nach IEC 61508)
SILworX	Programmiersoftware für HIMax, HIQuad X und HIMatrix.
SIP	Sicherheitsbezogenes Protokoll
SNTP	Simple Network Time Protocol (RFC 1769)
SRS	System.Rack.Slot
SW	Software

Begriff	Beschreibung
TMO	Timeout
W	Write
WD	Watchdog
WDZ	Watchdog-Zeit

Abbildungsverzeichnis

Bild 1:	Steuerung des Consumer/Provider Status (IOxS)	15
Bild 2:	PROFIsafe-Control-Byte und Status-Byte	17
Bild 3:	Reaktionszeit zwischen einem F-Device und einem HIMA F-Host	19
Bild 4:	Reaktionszeit mit einem HIMA F-Host und zwei F-Devices	19
Bild 5:	PROFINET IO-Controller im HIMax Strukturbaum	25
Bild 6:	Device-Access-Point (DAP) für das PROFINET IO-Device	26
Bild 7:	Strukturbaum des PROFINET IO-Controllers	30
Bild 8:	Kommunikation über PROFINET/PROFIsafe	44
Bild 9:	PROFINET IO-Device im HIMax Strukturbaum	50
Bild 10:	Kommunikation über PROFIBUS DP	61
Bild 11:	HIMax PROFIBUS DP Slave mit Modulen	64
Bild 12:	Benutzerdatenfeld	67
Bild 13:	Dialogfenster Verifikation	67
Bild 14:	Eigenschaften des PROFIBUS DP Master	68
Bild 15:	Eigenschaften des PROFIBUS DP Slave	69
Bild 16:	Isochroner PROFIBUS DP Zyklus	79
Bild 17:	Dialog <i>User Parameter bearbeiten</i>	88
Bild 18:	Funktionsbaustein MSTAT	90
Bild 19:	Programmierbeispiel zum Setzen des Eingangs A_Mode	91
Bild 20:	Programmiervorschlag für Logik des Hilfsfunktionsbausteins PBM_Mode	92
Bild 21:	Funktionsbaustein RALRM	95
Bild 22:	Funktionsbaustein RDIAG	99
Bild 23:	Funktionsbaustein RDREC	103
Bild 24:	Funktionsbaustein SLACT	106
Bild 25:	Funktionsbaustein WRREC	109
Bild 26:	Hilfsfunktionsbaustein ACTIVE	112
Bild 27:	Hilfsfunktionsbaustein ALARM	113
Bild 28:	Hilfsfunktionsbaustein DEID	114
Bild 29:	Hilfsfunktionsbaustein ID	115
Bild 30:	Hilfsfunktionsbaustein NSLOT	116
Bild 31:	Hilfsfunktionsbaustein SLOT	116
Bild 32:	Hilfsfunktionsbaustein STDDIAG	117
Bild 33:	Funktionsbaustein PNM_MSTST (oberer Teil)	131

Bild 34:	Funktionsbaustein PNM_MSTST (unterer Teil)	132
Bild 35:	Auswahl Funktionsbausteine	132
Bild 36:	Systemvariablen des Funktionsbausteins MSTAT	132

Tabellenverzeichnis

Tabelle 1:	Zusätzlich geltende Handbücher	7
Tabelle 2:	Übersicht PROFINET IO-Funktionsbausteine	14
Tabelle 3:	Systemanforderung und Ausstattung für PROFINET IO-Controller	23
Tabelle 4:	Eigenschaften PROFINET IO-Controller	23
Tabelle 5:	Register PROFINET IO (Eigenschaften)	31
Tabelle 6:	Register Parameter (Eigenschaften)	32
Tabelle 7:	Register Parameter (Eigenschaften)	32
Tabelle 8:	Register Parameter	33
Tabelle 9:	Register Systemvariablen	33
Tabelle 10:	Register Parameter des I/O PROFINET IO-Moduls	34
Tabelle 11:	Register Parameter	35
Tabelle 12:	Register Systemvariablen	38
Tabelle 13:	F-Parameter von Submodul Input (Eigenschaften)	39
Tabelle 14:	Register Parameter	40
Tabelle 15:	Register Systemvariablen	41
Tabelle 16:	Register Parameter	42
Tabelle 17:	Register Systemvariablen	43
Tabelle 18:	Application Relation (Eigenschaften)	44
Tabelle 19:	Alarm CR (Eigenschaften)	45
Tabelle 20:	Input CR (Eigenschaften)	46
Tabelle 21:	Input CR (Edit)	47
Tabelle 22:	Output CR (Eigenschaften)	48
Tabelle 23:	Systemanforderung und Ausstattung für PROFINET IO-Controller	49
Tabelle 24:	Eigenschaften PROFINET IO-Controller	49
Tabelle 25:	Allgemeine Eigenschaften des PROFINET IO-Device	53
Tabelle 26:	PROFINET IO-Module	54
Tabelle 27:	PROFIsafe-Module	55
Tabelle 28:	Allgemeine Eigenschaften des Device Moduls	56
Tabelle 29:	Edit Dialog Submodul	58
Tabelle 30:	Benötigte Ausstattung und Systemanforderungen	60
Tabelle 31:	Eigenschaften PROFIBUS DP Master	60
Tabelle 32:	Ausgänge HIMA PROFIBUS DP Slave	62
Tabelle 33:	Eingänge HIMA PROFIBUS DP Slave	62
Tabelle 34:	Variablen Eingangsmodul [000] DP-Input/ELOP-Export: 2 Bytes	65

Tabelle 35: Variablen Eingangsmodul [001] DP-Input/ELOP-Export: 8 Bytes	65
Tabelle 36: Variablen Eingangsmodul [002] DP-Input/ELOP-Export: 1 Byte	65
Tabelle 37: Variablen Ausgangsmodul [003] DP-Output/ELOP-Import: 2 Bytes	66
Tabelle 38: Variablen Ausgangsmodul [004] DP-Output/ELOP-Import: 1 Byte	66
Tabelle 39: Systemvariablen des PROFIBUS DP Master	70
Tabelle 40: Allgemeine Eigenschaften des PROFIBUS DP Master	72
Tabelle 41: Register Zeiten im Eigenschaftendialog des PROFIBUS DP Master	74
Tabelle 42: Register CPU/COM im Eigenschaftendialog des PROFIBUS DP Master	74
Tabelle 43: Sonstige Eigenschaften des PROFIBUS DP Master	75
Tabelle 44: Richtwerte für Token-Umlaufzeit bei verschiedenen Übertragungsraten	76
Tabelle 45: Übertragungszeit für ein Zeichen bei verschiedenen Übertragungsraten	76
Tabelle 46: Elemente zur Berechnung der Token-Umlaufzeit	77
Tabelle 47: Systemvariablen des PROFIBUS DP Slave	81
Tabelle 48: Register Parameter des PROFIBUS DP Slave	82
Tabelle 49: Register Gruppen im Eigenschaftendialog des PROFIBUS DP Slave	83
Tabelle 50: Register DP-V1 im Eigenschaftendialog des PROFIBUS DP Slave	83
Tabelle 51: Register Alarme im Eigenschaftendialog des PROFIBUS DP Slave	84
Tabelle 52: Register Daten im Eigenschaftendialog des PROFIBUS DP Slave	84
Tabelle 53: Register Modell im Eigenschaftendialog des PROFIBUS DP Slave	84
Tabelle 54: Register Features im Eigenschaftendialog des PROFIBUS DP Slave	85
Tabelle 55: Register Übertragungsraten im Eigenschaftendialog des PROFIBUS DP Slave	85
Tabelle 56: Register Azyklisch im Eigenschaftendialog des PROFIBUS DP Slave	86
Tabelle 57: GSD-Datei des HIMA PROFIBUS DP Slave	86
Tabelle 58: Beispiel: Blöcke 1 ... 4 des Benutzerdatenfeldes	88
Tabelle 59: Beispiel: Blöcke 1 ... 4 des Benutzerdatenfeldes	88
Tabelle 60: Übersicht PROFIBUS DP Funktionsbausteine	89
Tabelle 61: A-Eingänge Funktionsbaustein MSTAT	91
Tabelle 62: A-Ausgänge Funktionsbaustein MSTAT	92
Tabelle 63: F-Eingänge Funktionsbaustein MSTAT	93
Tabelle 64: F-Ausgänge Funktionsbaustein MSTAT	93
Tabelle 65: Eingangs-Systemvariablen	93
Tabelle 66: Ausgangs-Systemvariablen	93
Tabelle 67: A-Eingänge Funktionsbaustein RALRM	95
Tabelle 68: A-Ausgänge Funktionsbaustein RALRM	96
Tabelle 69: F-Eingänge Funktionsbaustein RALRM	96
Tabelle 70: F-Ausgänge Funktionsbaustein RALRM	96
Tabelle 71: Eingangs-Systemvariablen	97
Tabelle 72: Ausgangs-Systemvariablen	97
Tabelle 73: Alarmdaten	98
Tabelle 74: A-Eingänge Funktionsbaustein RDIAG	99

Tabelle 75: A-Ausgänge Funktionsbaustein RDIAG	99
Tabelle 76: F-Eingänge Funktionsbaustein RDIAG	100
Tabelle 77: F-Ausgänge Funktionsbaustein RDIAG	100
Tabelle 78: Eingangs-Systemvariablen	100
Tabelle 79: Ausgangs-Systemvariablen	101
Tabelle 80: Diagnosedaten	101
Tabelle 81: A-Eingänge Funktionsbaustein RDREC	103
Tabelle 82: A-Ausgänge Funktionsbaustein RDREC	103
Tabelle 83: F-Eingänge Funktionsbaustein RDREC	104
Tabelle 84: F-Ausgänge Funktionsbaustein RDREC	104
Tabelle 85: Eingangs-Systemvariablen	104
Tabelle 86: Ausgangs-Systemvariablen	105
Tabelle 87: Daten	105
Tabelle 88: A-Eingänge Funktionsbaustein SLACT	106
Tabelle 89: A-Ausgänge Funktionsbaustein SLACT	106
Tabelle 90: F-Eingänge Funktionsbaustein SLACT	107
Tabelle 91: F-Ausgänge Funktionsbaustein SLACT	107
Tabelle 92: Eingangs-Systemvariablen	107
Tabelle 93: Ausgangs-Systemvariablen	108
Tabelle 94: A-Eingänge Funktionsbaustein WRREC	109
Tabelle 95: A-Ausgänge Funktionsbaustein WRREC	109
Tabelle 96: F-Eingänge Funktionsbaustein WRREC	110
Tabelle 97: F-Ausgänge Funktionsbaustein WRREC	110
Tabelle 98: Eingangs-Systemvariablen	110
Tabelle 99: Ausgangs-Systemvariablen	111
Tabelle 100: Daten	111
Tabelle 101: Übersicht Hilfsfunktionsbausteine	112
Tabelle 102: Eingänge Hilfsfunktionsbaustein ACTIVE	112
Tabelle 103: Ausgänge Hilfsfunktionsbaustein ACTIVE	112
Tabelle 104: Eingänge Hilfsfunktionsbaustein ALARM	113
Tabelle 105: Ausgänge Hilfsfunktionsbaustein ALARM	114
Tabelle 106: Eingänge Hilfsfunktionsbaustein DEID	114
Tabelle 107: Ausgänge Hilfsfunktionsbaustein DEID	114
Tabelle 108: Eingänge Hilfsfunktionsbaustein ID	115
Tabelle 109: Ausgänge Hilfsfunktionsbaustein ID	115
Tabelle 110: Eingänge Hilfsbaustein NSLOT	116
Tabelle 111: Ausgänge Hilfsbaustein NSLOT	116
Tabelle 112: Eingänge Hilfsbaustein SLOT	117
Tabelle 113: Ausgänge Hilfsbaustein SLOT	117
Tabelle 114: Eingänge Hilfsbaustein STDDIAG	117

Allgemein		Protokoll
Tabelle 115:	Ausgänge Hilfsbaustein STDDIAG	118
Tabelle 116:	Fehlercodes Funktionsbausteine	119
Tabelle 117:	Anzeigenfeld PROFIBUS Master	122
Tabelle 118:	PROFIBUS DP Master Zustand	122
Tabelle 119:	PROFIBUS DP Master Verhalten	122
Tabelle 120:	LED FBx	123
Tabelle 121:	FAULT FBx	123
Tabelle 122:	Ausstattung und Systemanforderungen HIMA PROFIBUS DP Slave	124
Tabelle 123:	Eigenschaften HIMA PROFIBUS DP Slave	124
Tabelle 124:	Systemvariablen PROFIBUS DP Slave	126
Tabelle 125:	Slave Eigenschaften: Register Allgemein	128
Tabelle 126:	Anzeigenfeld (PROFIBUS DP Slave)	128
Tabelle 127:	LED FBx	129
Tabelle 128:	FAULT FBx	129

Für weitere Informationen kontaktieren Sie:

HIMA Paul Hildebrandt GmbH

Albert-Bassermann-Str. 28
68782 Brühl, Germany

Telefon +49 6202 709-0
Fax +49 6202 709-107
E-Mail info@hima.com

Erfahren Sie online mehr über HIMA Lösungen:



www.hima.com/de/