



WAGO-I/O-SYSTEM 750
WAGO-I/O-IPC-C6
758-874/000-112
CANopen-Master, CODESYS 2.3

Version 2.4.0



© 2014 by WAGO Kontakttechnik GmbH & Co. KG
Alle Rechte vorbehalten.

WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

Tel.: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-Mail: info@wago.com

Web: <http://www.wago.com>

Technischer Support

Tel.: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 55

E-Mail: support@wago.com

Es wurden alle erdenklichen Maßnahmen getroffen, um die Richtigkeit und Vollständigkeit der vorliegenden Dokumentation zu gewährleisten. Da sich Fehler, trotz aller Sorgfalt, nie vollständig vermeiden lassen, sind wir für Hinweise und Anregungen jederzeit dankbar.

E-Mail: documentation@wago.com

Wir weisen darauf hin, dass die im Handbuch verwendeten Soft- und Hardware-Bezeichnungen und Markennamen der jeweiligen Firmen im Allgemeinen einem Warenzeichenschutz, Markenzeichenschutz oder patentrechtlichem Schutz unterliegen.

Inhaltsverzeichnis

1	Hinweise zu dieser Dokumentation	10
1.1	Gültigkeitsbereich	10
1.2	Urheberschutz	10
1.3	Symbole.....	11
1.4	Darstellung der Zahlensysteme.....	12
1.5	Schriftkonventionen	12
2	Wichtige Erläuterungen	13
2.1	Rechtliche Grundlagen.....	13
2.1.1	Änderungsvorbehalt	13
2.1.2	Personalqualifikation.....	13
2.1.3	Bestimmungsgemäße Verwendung.....	14
2.1.4	Technischer Zustand der Geräte.....	14
2.2	Sicherheitshinweise.....	15
2.3	Sicherheitseinrichtungen.....	16
2.4	Hinweise zum Betrieb.....	16
2.5	Spezielle Einsatzbestimmungen für ETHERNET-Geräte	16
3	Lieferumfang.....	17
4	Gerätebeschreibung.....	18
4.1	Übersicht der physikalischen Schnittstellen.....	20
4.2	Anzeigeelemente	22
4.3	Bedienelemente	23
4.4	Batterie	25
4.5	Bedruckung	26
4.6	Technische Daten	27
4.6.1	Gerätedaten.....	27
4.6.2	Systemdaten.....	28
4.6.3	Versorgung	28
4.6.4	Kommunikation.....	29
4.6.5	Schutz und Sicherheit.....	29
4.6.6	Laufzeitsystem.....	29
4.6.7	Umgebungsbedingungen	29
4.6.8	Anschlusstechnik.....	30
4.7	Normen und Richtlinien.....	30
4.7.1	Elektromagnetische Verträglichkeit.....	30
4.8	Zulassungen.....	31
5	Beschreibung der I/O-IPC-Schnittstellen	32
5.1	ETHERNET-Schnittstellen (X8, X9).....	32
5.2	Schnittstelle für die Elektronikversorgung (X4).....	33
5.3	CANopen-Schnittstelle (X3).....	34
5.4	Integrierte Ein- und Ausgänge (X5).....	35
5.5	USB-Schnittstellen (X10, X11)	38
5.6	Serielle RS-232-Schnittstelle (X6).....	39
5.7	DVI-I-Schnittstelle (X7)	40
6	Montage und Demontage des I/O-IPC	42

6.1	Hinweise zur Montage/Demontage.....	42
6.2	Benötigtes Zubehör für die Montage	43
6.3	Zulässige Einbaurichtungen des I/O-IPC.....	43
6.4	Befestigung des I/O-IPC auf einer Tragschiene.....	44
6.5	Anstecken der Busklemmen an den I/O-IPC	45
6.6	Demontage des I/O-IPC	47
6.6.1	Entfernen der Leitungen	47
6.6.2	Demontage des I/O-IPC von der Tragschiene.....	48
7	Versorgungsspannung anschließen	50
7.1	Hinweise.....	50
7.2	Benötigtes Zubehör	51
7.3	Einspeisung bei Verwendung der Potentialeinspeiseklemme 750-602 ..	52
7.4	Einspeisung bei Verwendung der Filterklemme 750-626.....	56
7.5	Sensor- und Aktorleitung an die Busklemmen anschließen	58
8	In Betrieb nehmen.....	59
8.1	Einschalten des I/O-IPC	59
8.2	Ermitteln der IP-Adresse des Host-PC.....	60
8.3	Einstellen einer IP-Adresse	61
8.3.1	Zuweisen einer IP-Adresse mittels BootP	61
8.3.2	Ändern einer IP-Adresse mittels „IPC Configuration Tool“	65
8.4	Test der Netzwerkverbindung	68
8.5	Ausschalten/Neustart.....	69
9	Konfigurieren	70
9.1	Konfiguration mittels Web-based Management (WBM).....	70
9.1.1	Benutzerverwaltung des WBM	72
9.1.2	Seite „Information“	73
9.1.3	Seite „CODESYS“	74
9.1.4	Seite „TCP/IP“.....	75
9.1.5	Seite „ETHERNET“	76
9.1.6	Seite „NTP“	76
9.1.7	Seite „Clock“	77
9.1.8	Seite „Users“	78
9.1.9	Seite „HMI Settings“	79
9.1.10	Seite „Administration“	81
9.1.11	Seite „Package Server“	82
9.1.12	Seite „Mass Storage“	84
9.1.13	Seite „Downloads“	84
9.1.14	Seite „Port“	85
9.1.15	Seite „MODBUS“	85
9.1.16	Seite „SNMP“	86
9.1.17	I/O Configuration	88
9.1.18	Seite „WebVisu“	88
9.2	Konfiguration mit einem Terminalprogramm.....	89
9.3	Konfiguration mit Touchscreen/Monitor und USB-Tastatur	90
10	MODBUS/TCP.....	91
10.1	Prozessdaten des MODBUS-Servers	92
10.2	Zugriff auf das Prozessabbild über MODBUS	92
10.2.1	Registerdienste	93

10.2.2	Bitdienste	94
10.3	Konfigurationsregister	95
10.4	Adressierungsbeispiel	96
11	Laufzeitumgebung CODESYS 2.3	98
11.1	Prozessabbilder	98
11.1.1	Prozessabbild für die am I/O-IPC angeschlossenen Busklemmen.....	98
11.1.2	Prozessabbild für die am Feldbus angeschlossenen Slaves.....	98
11.2	Schreibweise logischer Adressen	99
11.3	Zugriff auf die Prozessabbilder der Ein- und Ausgangsdaten über CODESYS 2.3.....	100
11.4	Adressierungsbeispiel	103
11.5	Installieren des Programmiersystems CODESYS 2.3	104
11.6	Das erste Programm mit CODESYS 2.3.....	104
11.6.1	Starten Sie das Programmiersystem CODESYS.....	104
11.6.2	Anlegen eines Projekts und Auswahl des Zielsystems.....	105
11.6.3	Anlegen der Steuerungskonfiguration.....	107
11.6.4	Editieren des Programmbausteins	111
11.6.5	SPS-Programm in den I/O-IPC laden und ausführen (ETHERNET).....	113
11.6.6	SPS-Programm in den I/O-IPC laden und ausführen (RS-232).....	115
11.6.7	Boot-Projekt erzeugen	117
11.7	Anlegen von Task-Prioritäten	118
11.7.1	Zyklische Task-Prioritäten	120
11.7.2	Freilaufende Tasks.....	121
11.8	Systemereignisse	122
11.9	Klemmenbussynchronisation	124
11.9.1	Fall 1: CODESYS-Task-Intervall kleiner als Klemmenbuszyklus eingestellt.....	124
11.9.2	Fall 2: CODESYS-Task-Intervall kleiner als doppelter Klemmenbuszyklus	125
11.9.3	Fall 3: CODESYS-Task-Intervall größer als doppelter Klemmenbuszyklus	126
11.9.4	Fall 4: CODESYS-Task-Intervall größer als 10 ms.....	127
11.10	CODESYS-Visualisierung.....	128
11.10.1	Einbinden von Schriften	130
11.10.2	Grenzen der CODESYS-Visualisierung.....	131
11.10.3	Beseitigung von Störungen der CODESYS-Web-Visualisierung....	133
11.10.4	Häufig gestellte Fragen zur CODESYS-Web-Visualisierung.....	134
12	CANopen-Master in CODESYS 2.3.....	136
12.1	Steuerungskonfiguration des CANopen-I/O-IPC	136
12.2	Einstellmöglichkeiten der Steuerungskonfiguration	140
12.2.1	CANopen-Master (I/O-IPC).....	140
12.2.2	CANopen-Slaves	142
12.3	Zugriff auf die CANopen-Prozessdaten.....	150
12.4	SPS-Programm in den I/O-IPC laden	153
12.5	Diagnose des Feldbuskopplers.....	154
12.5.1	DiagGetBusState() und DiagGetState()	154
12.5.2	Erstellen von Diagnosefunktionen in CODESYS 2.3	155
12.5.3	Aufruf des Diagnosebausteins.....	157

12.5.4	Durchführen der Busdiagnose mittels DiagGetBusState().....	158
12.5.5	Durchführen der Teilnehmerdiagnose mittels DiagGetState().....	160
12.5.6	Auswerten der CANopen-Diagnose (Emergency-Nachrichten)	161
12.6	Datenaustausch von einfachen CAN-Teilnehmern mit dem I/O-IPC...	166
13	C-Funktionen als CODESYS-Bibliothek einbinden	168
13.1	Beispiel zum Einbinden einer dynamischen Library	168
13.1.1	Linux-Shared-Library entwickeln und übersetzen	168
13.1.2	Beschreibungsdatei für das CODESYS-Laufzeitsystem erzeugen ..	169
13.1.3	Library und INI-Datei kopieren und das CODESYS-Laufzeitsystem neu starten	170
13.1.4	Eine IEC-Library erzeugen.....	171
13.1.5	Bibliothek im CODESYS-Projekt einbinden	173
13.2	Besonderheiten	176
13.2.1	Datentypen.....	176
13.2.2	Strukturen	177
13.2.3	Parameterübergabe per Referenz oder per Value	178
13.3	Weitere Anwendungen	178
14	Betriebssystem.....	179
14.1	Verwendeter Linux-Kernel	179
14.2	Grand Unified Bootloader (GRUB)	180
14.3	Startablauf von Linux.....	181
14.4	Linux-Konsole.....	182
14.4.1	Zugriff auf die Linux-Konsole	182
14.4.1.1	Zugriff über Telnet	184
14.4.1.2	Zugriff über RS-232-Schnittstelle und Terminalprogramm	185
14.4.1.3	Zugriff über Tastatur und Monitor (DVI- und USB-Schnittstelle)	186
14.4.2	Installierte Anwendungen.....	187
14.4.3	Aufbau des Dateisystems	188
14.4.4	Installierte Shell (BASH)	191
14.4.5	Busybox und andere Hilfsprogramme.....	191
14.5	Treiber für spezielle Hardwareteile.....	193
14.6	Einbinden eines USB-Druckers	193
14.7	Installierte Dienste der ETHERNET-Schnittstelle.....	194
14.7.1	Telnet-Server (telnetd).....	194
14.7.2	FTP-Server (pure-ftpd).....	195
14.7.3	NFS-Server	196
14.7.4	FTP-Client	196
14.7.5	Webserver (lighttpd).....	197
14.7.6	NTP-Client	197
14.7.7	NFS-Client.....	198
14.7.8	SNMP-Agent	198
15	Diagnose.....	200
15.1	Betriebs- und Statusmeldungen.....	200
15.2	Diagnosemeldungen (I/O-LED).....	204
15.2.1	Ablauf der Blinksequenz	205
15.2.2	Beispiel einer Diagnosemeldung mittels Blinkcode.....	206
15.2.3	Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung...	207

16	Service	212
16.1	Austausch der Batterie	212
16.2	Entsorgung	214
17	Einsatz in explosionsgefährdeten Bereichen	215
17.1	Beispielhafter Aufbau der Kennzeichnung	216
17.1.1	Kennzeichnung für Europa gemäß ATEX und IEC-Ex	216
17.1.2	Kennzeichnung für Amerika gemäß NEC 500.....	221
17.2	Errichtungsbestimmungen.....	222
17.2.1	Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 07 ATEX 554086 X).....	223
17.2.2	Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 12 ATEX 106032 X).....	224
17.2.3	Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEX TUN 09.0001 X).....	225
17.2.4	Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEX TUN 12.0039 X).....	226
17.2.5	ANSI/ISA 12.12.01	227
18	Busklemmen	228
18.1	Übersicht	228
18.2	Aufbau der Prozessdaten für MODBUS/TCP.....	229
18.2.1	Digitaleingangsklemmen.....	230
18.2.1.1	1-Kanal-Digitaleingangsklemmen mit Diagnose	230
18.2.1.2	2-Kanal-Digitaleingangsklemmen.....	230
18.2.1.3	2-Kanal-Digitaleingangsklemmen mit Diagnose	230
18.2.1.4	2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten.....	231
18.2.1.5	4-Kanal-Digitaleingangsklemmen.....	231
18.2.1.6	8-Kanal-Digitaleingangsklemmen.....	231
18.2.1.7	8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten.....	232
18.2.2	16-Kanal-Digitaleingangsklemmen.....	232
18.2.2.1	Digitalausgangsklemmen.....	233
18.2.2.2	1-Kanal-Digitalausgangsklemmen mit Eingangsdaten.....	233
18.2.2.3	2-Kanal-Digitalausgangsklemmen	233
18.2.2.4	2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten.....	234
18.2.2.5	4-Kanal-Digitalausgangsklemmen	235
18.2.2.6	4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten.....	235
18.2.2.7	8-Kanal-Digitalausgangsklemmen	235
18.2.2.8	8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten.....	236
18.2.2.9	16-Kanal-Digitalausgangsklemmen	236
18.2.2.10	8-Kanal-Digitaleingangsklemmen/-Digitalausgangsklemmen....	237
18.2.3	Analogeingangsklemmen	238
18.2.3.1	1-Kanal-Analogeingangsklemmen	238
18.2.3.2	2-Kanal-Analogeingangsklemmen	238
18.2.3.3	4-Kanal-Analogeingangsklemmen	239
18.2.3.4	3-Phasen-Leistungsmessklemme.....	239

18.2.3.5	8-Kanal-Analogeingangsklemmen	240
18.2.4	Analogausgangsklemmen	241
18.2.4.1	2-Kanal-Analogausgangsklemmen	241
18.2.4.2	4-Kanal-Analogausgangsklemmen	241
18.2.4.3	8-Kanal-Analogausgangsklemmen	242
18.2.5	Sonderklemmen	243
18.2.5.1	Zählerklemmen	243
18.2.5.2	Pulsweitenklemmen	245
18.2.5.3	Serielle Schnittstellen mit alternativem Datenformat	245
18.2.5.4	Serielle Schnittstellen mit Standard-Datenformat	246
18.2.5.5	Datenaustauschklemmen	246
18.2.5.6	SSI-Geber-Interface-Busklemmen	246
18.2.5.7	Weg- und Winkelmessung	247
18.2.5.8	DC-Drive Controller	249
18.2.5.9	Steppercontroller	250
18.2.5.10	RTC-Modul	251
18.2.5.11	DALI/DSI-Masterklemme	251
18.2.5.12	DALI-Multi-Master-Klemme	252
18.2.5.13	LON [®] -FTT-Klemme	254
18.2.5.14	Funkreceiver EnOcean	254
18.2.5.15	MP-Bus-Masterklemme	254
18.2.5.16	Bluetooth [®] RF-Transceiver	255
18.2.5.17	Schwingstärke/Wälzlagerüberwachung VIB I/O	256
18.2.5.18	KNX/EIB/TP1-Klemme	256
18.2.5.19	AS-Interface-Masterklemme	257
18.2.6	Systemklemmen	258
18.2.6.1	Systemklemmen mit Diagnose	258
18.3	Mailboxklemmen	260
19	Anhang	261
19.1	WagoConfigToolLIB.lib	261
19.1.1	Aufrufe zur Bibliothek „WagoConfigToolLIB.lib“	263
19.2	WagoLibNetSnmplib	284
19.2.1	snmpRegisterCustomOID_INT32()	285
19.2.2	snmpRegisterCustomOID_STRING()	286
19.2.3	snmpRegisterCustomOID_UINT32()	287
19.2.4	snmpGetValueCustomOID_INT32()	288
19.2.5	snmpGetValueCustomOID_STRING()	289
19.2.6	snmpGetValueCustomOID_UINT32()	290
19.2.7	snmpSetValueCustomOID_INT32()	291
19.2.8	snmpSetValueCustomOID_STRING()	292
19.2.9	snmpSetValueCustomOID_UINT32()	293
19.2.10	Rückmeldungen	294
19.2.11	Beispielprogramm „Test.pro“	295
19.3	WAGO_CANopen_02.lib	298
19.3.1	CIA405_GET_KERNEL_STATE	299
19.3.2	CIA405_GET_LOCAL_NODE_ID	301
19.3.3	CIA405_RECV_EMCY	303
19.3.4	CIA405_RECV_EMCY_DEV	305
19.3.5	CIA405_GET_STATE	308
19.3.6	CIA405_NMT	310

19.3.7	CIA405_SDO_READ4	312
19.3.8	CIA405_SDO_READxx	315
19.3.9	CIA405_SDO_WRITE4	318
19.3.10	CIA405_SDO_WRITExx	321
19.3.11	NMT_GUARD_ERROR	324
19.3.12	NMT_GUARD_ERROR_DEV	326
19.3.13	CANOPEN_VERSION	328
19.3.14	CIA405_DEVICE	329
19.3.15	CIA405_SDO_ERROR	329
19.3.16	CIA405_EMCY_ERROR	330
19.3.17	CIA405_STATE	331
19.3.18	CIA405_TRANSITION_STATE	332
19.3.19	CANOPEN_KERNEL_ERROR	333
19.4	WAGO_CANLayer2_01.lib	334
19.4.1	CAN_LAYER2_VERSION	335
19.4.2	CAN_ERROR_INFO	336
19.4.3	CAN_RX_11BIT_FRAME	338
19.4.4	CAN_RX_29BIT_FRAME	342
19.4.5	CAN_TX_11BIT_FRAME	346
19.4.6	CAN_TX_29BIT_FRAME	348
19.4.7	CAN_LAYER2_FRAME_ERROR	350
19.5	mod_com.lib	352
19.6	SerComm.lib	352
19.7	WagoLibTerminalDiag.lib	352
19.8	WagoLibKBUS.lib	352
19.9	SysLibCom.lib	352
19.10	SysLibFile, SysLibDir, SysLibFileAsync	352
	Abbildungsverzeichnis	354
	Tabellenverzeichnis	358

1 Hinweise zu dieser Dokumentation

ACHTUNG**Betriebsanleitung lesen!**

Zur Vermeidung von Personen- und Sachschäden installieren und betreiben Sie den I/O-IPC nur nach Anweisungen dieser Betriebsanleitung und der Systembeschreibung 750-xxx. Ferner beachten Sie unbedingt die Hinweise im Kapitel „Sicherheit“.

ACHTUNG**Örtliche Bestimmungen beachten!**

Zur Einbindung der 750-Komponenten in Ihre Maschine oder Anlage sind bei allen Tätigkeiten die jeweils gültigen und anwendbaren Normen, Vorschriften und Richtlinien zu beachten.

ACHTUNG**Versorgungsauslegung des WAGO-I/O-SYSTEMs 750!**

Ergänzend zu dieser Betriebsanleitung benötigen Sie die Systembeschreibung „Projektierungshinweise“, die unter www.wago.com erhältlich ist. Dort erhalten Sie unter anderem wichtige Informationen zu Potentialtrennung, Systemversorgung und Einspeisungsvorschriften.

Hinweis**Dokumentation aufbewahren!**

Diese Dokumentation ist Teil des Produkts. Bewahren Sie deshalb die Dokumentation während der gesamten Lebensdauer des Gerätes auf. Geben Sie die Dokumentation an jeden nachfolgenden Besitzer oder Benutzer des Gerätes weiter. Stellen Sie darüber hinaus sicher, dass gegebenenfalls jede erhaltene Ergänzung in die Dokumentation mit aufgenommen wird.

1.1 Gültigkeitsbereich

Die vorliegende Dokumentation gilt für den WAGO-I/O-IPC-C6, 758-874/000-112 der Serie WAGO-I/O-SYSTEM 750.

1.2 Urheberrecht

Diese Dokumentation, einschließlich aller darin befindlichen Abbildungen, ist urheberrechtlich geschützt. Jede Weiterverwendung dieser Dokumentation, die von den urheberrechtlichen Bestimmungen abweicht, ist nicht gestattet. Die Reproduktion, Übersetzung in andere Sprachen sowie die elektronische und fototechnische Archivierung und Veränderung bedarf der schriftlichen Genehmigung der WAGO Kontakttechnik GmbH & Co. KG, Minden. Zuwiderhandlungen ziehen einen Schadenersatzanspruch nach sich.

1.3 Symbole

GEFAHR



Warnung vor Personenschäden!

Kennzeichnet eine unmittelbare Gefährdung mit hohem Risiko, die Tod oder schwere Körperverletzung zur Folge haben wird, wenn sie nicht vermieden wird.

GEFAHR



Warnung vor Personenschäden durch elektrischen Strom!

Kennzeichnet eine unmittelbare Gefährdung mit hohem Risiko, die Tod oder schwere Körperverletzung zur Folge haben wird, wenn sie nicht vermieden wird.

WARNUNG



Warnung vor Personenschäden!

Kennzeichnet eine mögliche Gefährdung mit mittlerem Risiko, die Tod oder (schwere) Körperverletzung zur Folge haben kann, wenn sie nicht vermieden wird.

VORSICHT



Warnung vor Personenschäden!

Kennzeichnet eine mögliche Gefährdung mit geringem Risiko, die leichte oder mittlere Körperverletzung zur Folge haben könnte, wenn sie nicht vermieden wird.

ACHTUNG



Warnung vor Sachschäden!

Kennzeichnet eine mögliche Gefährdung, die Sachschaden zur Folge haben könnte, wenn sie nicht vermieden wird.

ESD



Warnung vor Sachschäden durch elektrostatische Aufladung!

Kennzeichnet eine mögliche Gefährdung, die Sachschaden zur Folge haben könnte, wenn sie nicht vermieden wird.

Hinweis



Wichtiger Hinweis!

Kennzeichnet eine mögliche Fehlfunktion, die aber keinen Sachschaden zur Folge hat, wenn sie nicht vermieden wird.

Information



Weitere Information

Weist auf weitere Informationen hin, die kein wesentlicher Bestandteil dieser Dokumentation sind (z. B. Internet).

1.4 Darstellung der Zahlensysteme

Tabelle 1: Darstellungen der Zahlensysteme

Zahlensystem	Beispiel	Bemerkung
Dezimal	100	Normale Schreibweise
Hexadezimal	0x64	C-Notation
Binär	'100' '0110.0100'	In Hochkomma, Nibble durch Punkt getrennt

1.5 Schriftkonventionen

Tabelle 2: Schriftkonventionen

Schriftart	Bedeutung
<i>kursiv</i>	Namen von Pfaden und Dateien werden kursiv dargestellt z. B.: <i>C:\Programme\WAGO-I/O-CHECK</i>
Menü	Menüpunkte werden fett dargestellt z. B.: Speichern
>	Ein „Größer als“- Zeichen zwischen zwei Namen bedeutet die Auswahl eines Menüpunktes aus einem Menü z. B.: Datei > Neu
Eingabe	Bezeichnungen von Eingabe- oder Auswahlfeldern werden fett dargestellt z. B.: Messbereichsanfang
„Wert“	Eingabe- oder Auswahlwerte werden in Anführungszeichen dargestellt z. B.: Geben Sie unter Messbereichsanfang den Wert „4 mA“ ein.
[Button]	Schaltflächenbeschriftungen in Dialogen werden fett dargestellt und in eckige Klammern eingefasst z. B.: [Eingabe]
[Taste]	Tastenbeschriftungen auf der Tastatur werden fett dargestellt und in eckige Klammern eingefasst z. B.: [F5]

2 Wichtige Erläuterungen

Dieses Kapitel beinhaltet ausschließlich eine Zusammenfassung der wichtigsten Sicherheitsbestimmungen und Hinweise. Diese werden in den einzelnen Kapiteln wieder aufgenommen. Zum Schutz vor Personenschäden und zur Vorbeugung von Sachschäden an Geräten ist es notwendig, die Sicherheitsrichtlinien sorgfältig zu lesen und einzuhalten.

2.1 Rechtliche Grundlagen

2.1.1 Änderungsvorbehalt

Die WAGO Kontakttechnik GmbH & Co. KG behält sich Änderungen, die dem technischen Fortschritt dienen, vor. Alle Rechte für den Fall der Patenterteilung oder des Gebrauchsmusterschutzes sind der WAGO Kontakttechnik GmbH & Co. KG vorbehalten. Fremdprodukte werden stets ohne Vermerk auf Patentrechte genannt. Die Existenz solcher Rechte ist daher nicht auszuschließen.

2.1.2 Personalqualifikation

Sämtliche Arbeitsschritte, die an dem I/O-IPC durchgeführt werden, dürfen nur von Elektrofachkräften mit ausreichenden Kenntnissen im Bereich der Automatisierungstechnik vorgenommen werden. Diese müssen mit den aktuellen Normen und Richtlinien für I/O-IPC und Automatisierungsumfeld vertraut sein.

Alle Eingriffe in die Steuerung sind stets von Fachkräften mit ausreichenden Kenntnissen in der SPS-Programmierung durchzuführen. Ferner sind detaillierte Kenntnisse des Betriebssystems Linux erforderlich.

2.1.3 Bestimmungsgemäße Verwendung

Der I/O-IPC dient ausschließlich zur Steuerung von Automatisierungsaufgaben. Er darf dabei nicht zur Übertragung und Verarbeitung von sicherheitsrelevanten Informationen genutzt werden, d. h., beispielsweise dürfen keine Not-Aus-Einrichtungen an diesem betrieben werden.

Der I/O-IPC ist ein Gerät der Klasse A und kann in Wohngebieten Funkstörungen verursachen. Ist dies der Fall, dürfen Sie den I/O-IPC nur nach Maßnahmen zur Reduzierung der Störaussendung einsetzen.

Der I/O-IPC ist für eine Betriebsumgebung vorgesehen, die keine höheren Anforderungen als die Schutzart IP20 an den I/O-IPC stellen darf.

Am I/O-IPC lassen sich bis zu 64 Busklemmen der Serie 750/753 anschließen. Mit der WAGO-Klemmenbusverlängerung (optional) ist die Nutzung von bis zu 250 Busklemmen möglich. Dabei sind folgende Systemgrenzen zu beachten:

- Die Gesamtlänge der Busklemmen hinter dem I/O-IPC darf einschließlich der Endklemme maximal 780 mm betragen.
- Die maximale Größe des Prozessabbilds für die Ein- und Ausgangsdaten darf jeweils 500 Byte nicht überschreiten.

Andere Anwendungen als die in dieser Anleitung beschriebenen sind nicht zulässig.

2.1.4 Technischer Zustand der Geräte

Die Geräte werden ab Werk für den jeweiligen Anwendungsfall mit einer festen Hard- und Software-Konfiguration ausgeliefert. Alle Veränderungen an der Hard- oder Software sowie der nicht bestimmungsgemäße Gebrauch der Komponenten bewirken den Haftungsausschluss der WAGO Kontakttechnik GmbH & Co. KG.

Wünsche an eine abgewandelte bzw. neue Hard- oder Software-Konfiguration richten Sie bitte an die WAGO Kontakttechnik GmbH & Co. KG.

2.2 Sicherheitshinweise

Zur Vermeidung von **Personenschäden** lesen und beachten Sie folgende Sicherheitshinweise, bevor Sie den I/O-IPC und die Busklemmen der Serie 750/753 verwenden.

GEFAHR



Elektrische Spannung!

Betreiben Sie den I/O-IPC ausschließlich mit 24 V DC PELV- oder SELV-Spannungsquellen („Protective Extra Low Voltage“/„Safety Extra Low Voltage“). Bei Nichtbeachtung besteht die Gefahr eines Stromschlags.

- Während des Betriebs können hohe Oberflächentemperaturen am Gehäuse des I/O-IPC auftreten. War der I/O-IPC in Betrieb, lassen Sie ihn abkühlen, bevor Sie diesen berühren.
- Beachten Sie bei der Montage, Inbetriebnahme, Wartung und Störbehebung die für Ihre Anlage zutreffenden Unfallverhütungsvorschriften.
- Beachten Sie für jede Tätigkeit die entsprechende Personenqualifikation im Kapitel „Personalqualifikation“.
- Lesen und beachten Sie die Betriebsanleitungen der WAGO-Busklemmen, die Sie am I/O-IPC anschließen.

Zur Vermeidung von Sachschäden lesen und beachten Sie folgende Hinweise:

- Die 750-Komponenten dürfen nicht mit Substanzen in Kontakt kommen, die kriechende und isolierende Eigenschaften besitzen. Andernfalls müssen Sie für die Geräte Zusatzmaßnahmen ergreifen wie den Einbau in ein Gehäuse, das gegen die oben genannten Substanzeigenschaften resistent ist.
- Die 750-Komponenten enthalten elektronische Bauelemente, die bei elektrostatischer Entladung zerstört werden können (siehe auch Kapitel tech. Daten). Beim Umgang mit den Komponenten achten Sie auf gute Erdung der Umgebung (Personen, Arbeitsplatz und Verpackung). Berühren Sie keine elektrisch leitenden Bauteile, z. B. Datenkontakte und Leiterplatten.
- Halten Sie genügend Abstand zu elektromagnetischen Störquellen ein (Frequenzumrichter, Motoren, ...), um eine hohe Störfestigkeit elektromagnetische Störstrahlungen zu erzielen. Verwenden Sie an den erforderlichen Stellen ausschließlich geschirmte Leitungen. Beachten Sie dazu die entsprechenden Normen für EMV-gerechte Installationen.
- Tauschen Sie defekte oder beschädigte 750-Komponenten aus, da es andernfalls zu Funktionsstörungen kommen kann.
- Achten Sie beim Verlegen sämtlicher Leitungen darauf, dass Sie diese nicht in Scherbereichen von beweglichen Anlagenteilen verlegen.

2.3 Sicherheitseinrichtungen

Alle 750-Komponenten entsprechen der Schutzart IP20. Unter anderem besteht daraus ein vollständiger Berührungsschutz vor elektrischen Spannungen und Strömen.

2.4 Hinweise zum Betrieb

Zur Einbindung der 750-Komponenten in Ihre Maschine oder Anlage sind bei allen Tätigkeiten die jeweils gültigen und anwendbaren Normen, Vorschriften und Richtlinien zu beachten. Darüber hinaus müssen die Not-Aus-Einrichtungen in allen Betriebsarten wirksam bleiben.

Zum Schutz vor elektromagnetischen Störungen

- schließen Sie Ihre Anlage an Schutz Erde (PE) an und
- stellen Sie sicher, dass die Leitungsführung und die Installation der Versorgungs- und Signalleitungen korrekt sind.

Folgende Maßnahmen zur 24-V-Versorgung müssen vorhanden sein:

- Äußerer Blitzschutz an Gebäuden
- Innerer Blitzschutz der Versorgungs- und Signalleitungen
- Sichere elektrische Trennung der Kleinspannung 24 V DC durch PELV-Spannungsquellen (Protective Extra Low Voltage) oder SELV-Spannungsquellen (Safety Extra Low Voltage)

2.5 Spezielle Einsatzbestimmungen für ETHERNET-Geräte

Wo nicht speziell beschrieben, sind ETHERNET-Geräte für den Einsatz in lokalen Netzwerken bestimmt. Beachten Sie folgende Hinweise, wenn Sie ETHERNET-Geräte in Ihrer Anlage einsetzen:

- Verbinden Sie Steuerungskomponenten und Steuerungsnetzwerke nicht mit einem offenen Netzwerk wie dem Internet oder einem Büronetzwerk. WAGO empfiehlt, Steuerungskomponenten und Steuerungsnetzwerke hinter einer Firewall anzubringen.
- Beschränken Sie den physikalischen und elektronischen Zugang zu sämtlichen Automatisierungskomponenten auf einen autorisierten Personenkreis.

- Ändern Sie vor der ersten Inbetriebnahme unbedingt die standardmäßig eingestellten Passwörter! Sie verringern so das Risiko, dass Unbefugte Zugriff auf Ihr System erhalten.
- Ändern Sie regelmäßig die verwendeten Passwörter! Sie verringern so das Risiko, dass Unbefugte Zugriff auf Ihr System erhalten.
- Ist ein Fernzugriff auf Steuerungskomponenten und Steuerungsnetzwerke erforderlich, sollte ein „Virtual Private Network“ (VPN) genutzt werden.
- Führen Sie regelmäßig eine Bedrohungsanalyse durch. So können Sie prüfen, ob die getroffenen Maßnahmen Ihrem Schutzbedürfnis entsprechen.
- Wenden Sie in der sicherheitsgerichteten Gestaltung Ihrer Anlage „Defense-in-depth“-Mechanismen an, um den Zugriff und die Kontrolle auf individuelle Produkte und Netzwerke einzuschränken.

3 Lieferumfang

Zum Lieferumfang des I/O-IPC gehören folgende Komponenten:

- Potentialeinspeiseklemme 750-602
(ab HW-Version 11 nicht mehr im Lieferumfang enthalten)
- Buchse für den Anschluss der Versorgungsspannung
- Schutzkappen

4 Gerätebeschreibung

Bei dem Automatisierungsgerät I/O-IPC handelt es sich um einen Industrie-PC, der die Steuerungsaufgaben einer SPS/PLC erledigen kann. Er ist zur Montage auf einer Hutschiene geeignet und zeichnet sich durch verschiedene Schnittstellen aus.

Am I/O-IPC können Sie alle verfügbaren Busklemmen des WAGO-I/O-SYSTEMs 750/753 anschließen. Dadurch kann er analoge und digitale Signale aus dem Automatisierungsumfeld intern verarbeiten oder über eine der vorhandenen Schnittstellen anderen Geräten zur Verfügung stellen.

Mit den zwei eigenständigen ETHERNET-Schnittstellen sind Übertragungsraten von 10 Mbit/s oder 100 Mbit/s im Halbduplex- oder Vollduplexbetrieb möglich.

Für den Datenaustausch stehen implementierte MODBUS/TCP-, -UDP- und -RTU-Server sowie ein CANopen-Master zur Verfügung.

Automatisierungsaufgaben lassen sich in allen IEC-61131-3 kompatiblen Sprachen mit dem Programmiersystem CODESYS 2.3 (WAGO-I/O-PRO CAA) realisieren. Die Implementierung der CODESYS-Taskarbeit ist für Linux mit Echtzeiterweiterungen optimiert, um die maximale Leistung für Automatisierungsaufgaben bereitzustellen. Zur Visualisierung stehen neben der Entwicklungsumgebung auch die CODESYS-Target-Visualisierung und die Web-Visualisierung zur Verfügung.

Die Feldbuskonfiguration ist auch mit der Steuerungskonfiguration von CODESYS 2.3 möglich.

Der I/O-IPC stellt 256 MB Programm- und Datenspeicher und 1023 kB Remanentspeicher bereit.

Über Funktionsbausteine können sowohl Clients als auch Server für TCP oder UDP programmiert werden.

Zur Konfiguration von Anwenderprogrammen dient unter anderem das Web-based Management (WBM). Dort sind unter anderem Informationen über die Konfiguration und den Status des I/O-IPC bereits als dynamische HTML-Seiten im I/O-IPC gespeichert. Bei Bedarf können Sie diese über einen Internet-Browser auslesen. Darüber hinaus lassen sich über ein implementiertes Dateisystem auch eigene HTML-Seiten hinterlegen oder Programme direkt aufrufen.

Die im Auslieferungszustand installierte Firmware basiert auf Linux mit speziellen Echtzeiterweiterungen des RT-Preempt-Patches. Zudem sind neben verschiedenen Hilfsprogrammen folgende Anwenderprogramme auf dem I/O-IPC installiert:

- ein SNMP-Server/Client

- ein Telnet-Server
- ein FTP-Server (unterstützt zeitgleich zehn FTP-Verbindungen)
- ein NTP-Client
- ein BootP- und DHCP-Daemon
- die CODESYS-Laufzeitumgebung.

Hinweis



Speicherkarte ist nicht im Lieferumfang enthalten!

Beachten Sie, der I/O-IPC wird ohne Speicherkarte ausgeliefert. Für die Nutzung einer Speicherkarte müssen Sie diese separat dazu bestellen.

Der I/O-IPC kann auch ohne Speicherkartenerweiterung betrieben werden, die Verwendung einer Speicherkarte ist optional.

Hinweis



Nur empfohlene Speicherkarte verwenden!

Setzen Sie ausschließlich die von WAGO erhältliche Speicherkarte CF (Art.-Nr. 758-879/000-000) ein, da diese für industrielle Anwendungen unter erschwerten Umweltbedingungen und für den Einsatz im I/O-IPC spezifiziert ist.

Die Kompatibilität zu anderen im Handel erhältlichen Speichermedien kann nicht gewährleistet werden.

4.1 Übersicht der physikalischen Schnittstellen

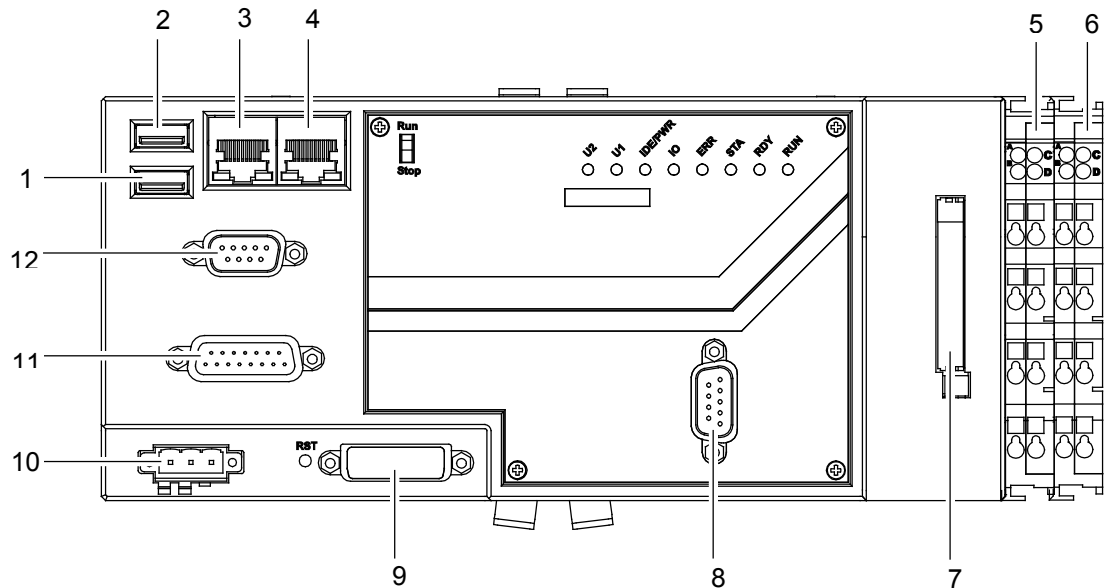


Abbildung 1: Übersicht der Schnittstellen

ACHTUNG Klemmenbus-Schnittstelle



Die Klemmenbus-Schnittstelle (Position 5) darf nicht demontiert werden. Diese Busklemme ist Bestandteil des I/O-IPCs.

Tabelle 3: Legende zur Abbildung „Übersicht der physikalischen Schnittstellen“

Position	Beschreibung	Funktion
1	USB-Schnittstelle (X10), Typ A	Zum Anschluss von USB-Geräten, wie z. B. Tastatur, USB-Speicher usw.
2	USB-Schnittstelle (X11), Typ A	
3	ETHERNET-Schnittstelle (X8), RJ-45	
4	ETHERNET -Schnittstelle (X9), RJ-45	Schnittstellen zum Anschluss des I/O-IPC an ein ETHERNET-Netzwerk.
5	Klemmenbusschnittstelle	Datenaustausch über den Klemmenbus.
6	Potentialeinspeiseklemme 750-602	Anschluss der Feldversorgung (Sensoren/Aktoren).
7	Steckplatz für CF-Karte	Steckplatz für CF-Karten, Typ I und II.
8	CANopen-Schnittstelle (X3), 9-poliger D-Sub-Stecker	Schnittstelle zum Anschluss des I/O-IPC an ein CAN-Netzwerk.

9	DVI-Schnittstelle (X7), D-Sub-Buchse, 24+5	Zum Anschluss eines digitalen oder analogen Monitors. Zum Anschluss eines Monitors über ein VGA-Kabel wird ein „DVI zu VGA Adapter“ benötigt.
10	Schnittstelle für die Elektronikversorgung (X4)	Anschluss für die 24-V-Elektronikversorgung des I/O-IPC. Der Anschluss ist gegen Verpolung geschützt.
11	Integrierte Ein- und Ausgänge (X5), 15-polige D-Sub-Buchse	Schnittstelle zum Anschluss von direkten digitalen Signalgebern.
12	Serielle RS-232-Schnittstelle (X6), 9-poliger D-Sub-Stecker	Physikalischer Anschluss an die Linux-Konsole, MODBUS/RTU, IO-Check oder CODESYS. Diese Schnittstelle ist über das Web-based Management oder über die Linux-Konsole konfigurierbar.

4.2 Anzeigeelemente

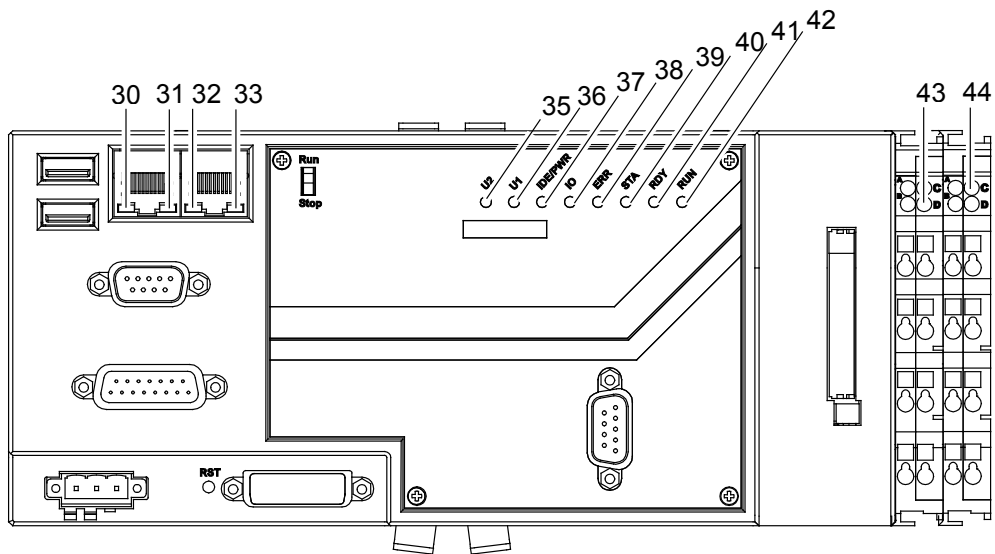


Abbildung 2: Kennzeichnung der LEDs

Tabelle 4: Legende zur Abbildung „Anzeigeelemente“

Position	LED	Farbe	Erläuterung
30	ACT/LNK	gelb/aus	Status zur Netzwerkverbindung der ETHERNET-Schnittstelle X8.
31	Speed	grün/aus	Status zur Übertragungsgeschwindigkeit der ETHERNET-Schnittstelle X8.
32	ACT/LNK	gelb/aus	Status zur Netzwerkverbindung der ETHERNET-Schnittstelle X9.
33	Speed	grün/aus	Status zur Übertragungsgeschwindigkeit der ETHERNET-Schnittstelle X9.
-	-	-	-
35	U2	grün/rot/ gelb/aus	Vom Anwender frei programmierbare Leuchtanzeige (zwei ansteuerbare LEDs).
36	U1	grün/rot/ gelb/ aus	Vom Anwender frei programmierbare Leuchtanzeige. (zwei ansteuerbare LEDs).
37	IDE/PWR	grün/rot	Status der Versorgungsspannung, des internen Flash-Speichers sowie der CF-Karte.
38	IO	grün/rot	Anzeige der Störmeldungen durch einen Blinkcode.
39	ERR	rot	Feldbusstatus
40	STA	gelb	
41	RDY	gelb	
42	RUN	grün	
43	LED D	aus	Diese LED wird nicht verwendet.
44	LED C	grün/aus	Status zur 24-V-Versorgungsspannung mittels Potentialeinspeiseklemme 750-602.

Detaillierte Informationen zu den LEDs erhalten Sie ab Kapitel „LED-Signalisierung“.

4.3 Bedienelemente

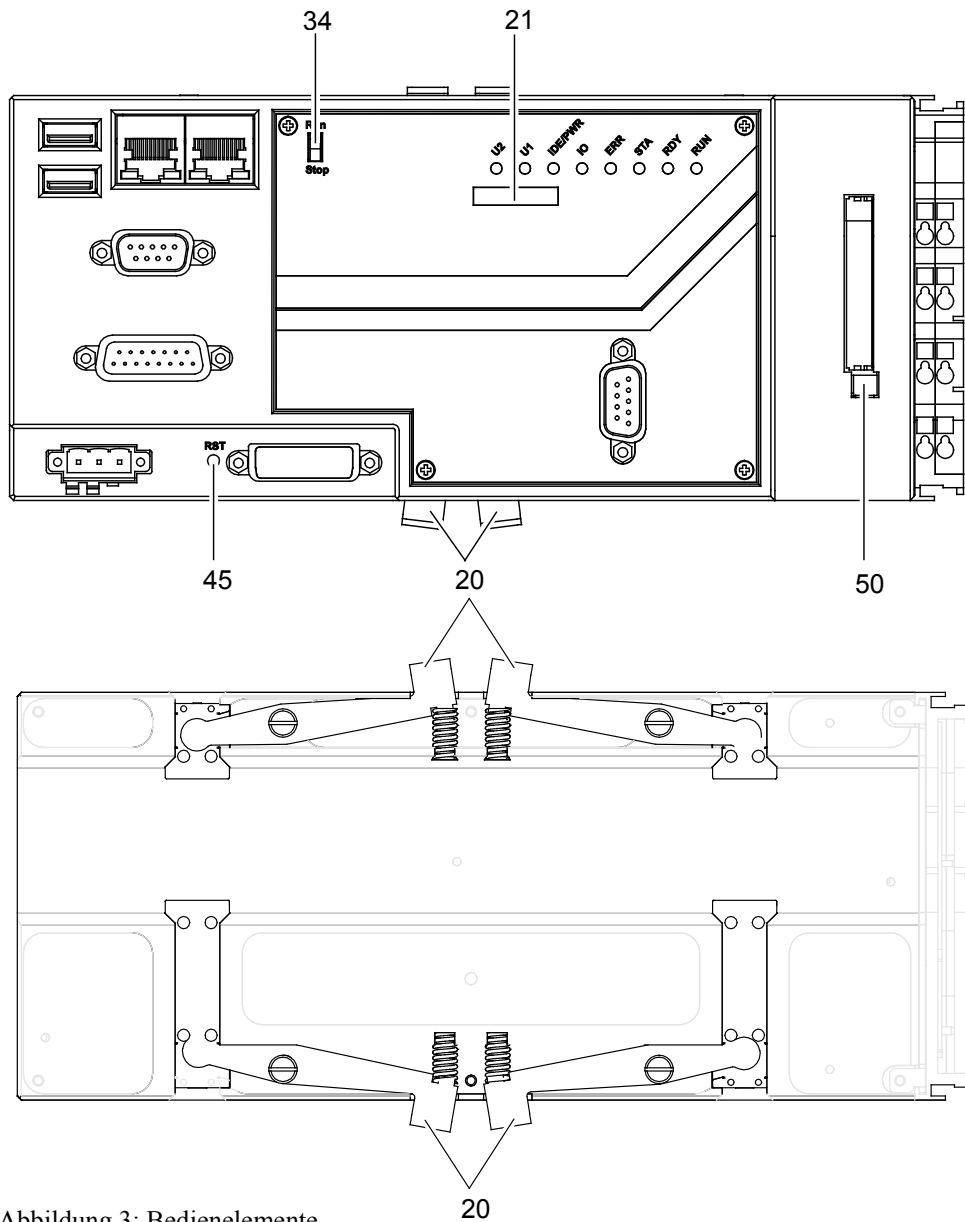


Abbildung 3: Bedienelemente

Tabelle 5: Legende zur Abbildung „Bedienelemente“

Position	Bedienelement	Erläuterung
20	Tragschienenbefestigung	Zur Befestigung des I/O-IPC an einer geerdeten Tragschiene.
21	Beschriftungsstreifen	Zur 4-stelligen Kennzeichnung des I/O-IPC durch das WAGO-Schnellbezeichnungssystem Mini-WSB.
34	Run/Stopp-Schalter	Run: automatischer Start des Boot-Projekts (CODESYS) bei Start des I/O-IPC bzw. beim Starten des SPS-Programms. Stopp: stoppen des SPS-Programms.
45	Reset-Taste	Zur Durchführung eines Neustarts des I/O-IPC.
50	Entriegelung	Zum Entfernen der CF-Karte die Entriegelung reindrücken.

4.4 Batterie

Die 3-V-Batterie (52) vom Typ CR2025 (LiMnO₂, ca.165 mAh) befindet sich unter verschraubten Abdeckung (51).

Die Batterie hält im Falle eines Stromausfalls die Spannungsversorgung für die Echtzeituhr (RTC) und den flüchtigen Speicher (SRAM) mit den CODESYS-Retain-Variablen aufrecht. Weitere Informationen erhalten Sie im Kapitel „Wartung“.

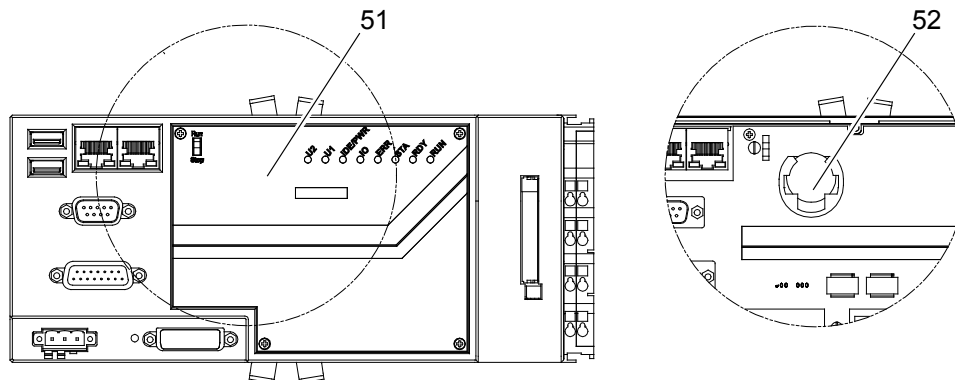


Abbildung 4: Batterie

4.5 Bedruckung

Seitlich des I/O-IPC befindet sich ein Etikett mit folgenden Informationen:

- A: Bezeichnung des I/O-IPC
- B: Bestellnummer des I/O-IPC
- C: Seriennummer des Gerätes
- D: Hardwarestand bei Auslieferung
- E: Firmwarestand bei Auslieferung
- F, G: MAC-Adressen für die ETHERNET-Schnittstellen X8 und X9.
Die MAC-Adressen dienen zur Identifikation und zur Adressierung von ETHERNET-Geräten. Jede MAC-Adresse kommt weltweit nur einmal vor.
- H: Zulassungen für den I/O-IPC
- I: Hersteller
- J: ATEX-/IECEx-Kennzeichnung mit Ex-Logo

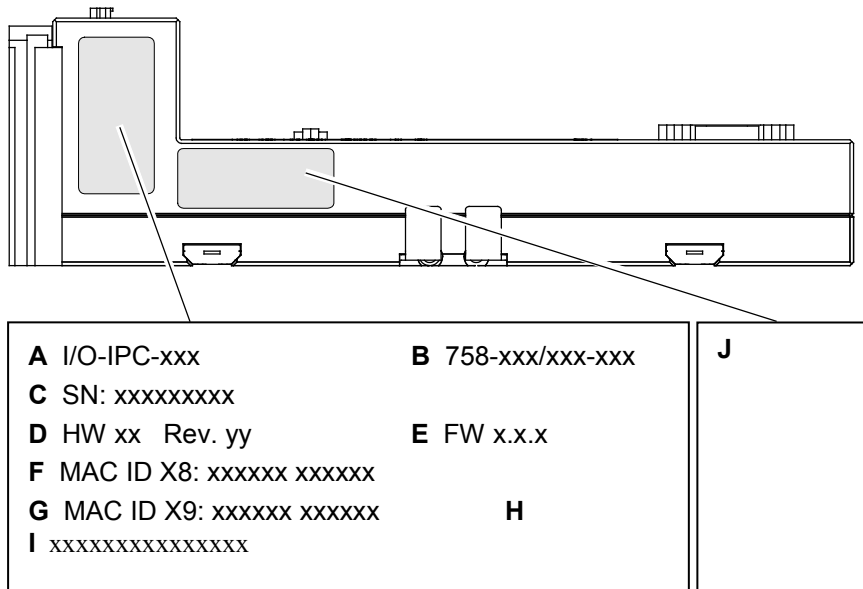


Abbildung 5: Seitliche Beschriftung auf dem I/O-IPC

4.6 Technische Daten

4.6.1 Gerätedaten

Tabelle 6: Technische Daten Gerät

Breite	236 mm
Höhe	70 mm Höhe ab Oberkante Tragschiene
Tiefe	100 mm
Gehäusematerial	Aluminium
Montageart	TS 35
Gewicht	Ca. 1100 g

4.6.2 Systemdaten

Tabelle 7: Technische Daten – Systemdaten

Anzahl der anschließbaren Busklemmen am I/O-IPC	64 Stück Mit Klemmenbusverlängerung (optional) bis zu 250 Stück.
Eingangsprozessabbild, max.	500 Byte
Ausgangsprozessabbild, max.	500 Byte
CPU	Celeron® M, 600 MHz (17 W)
Bios	Insyde
Unterstützte Bildschirmauflösungen	640x480, 256 Farben 800x600, 256 Farben 1024x768, 256 Farben 1280x1024, 256 Farben 640x480, 16 bit 800x600, 16 bit 1024x768, 16 bit 1280x1024, 16 bit 640x480, 32 bit 800x600, 32 bit 1024x768, 32 bit 1280x1024, 32 bit
Speichererweiterung	CompactFlash, Typ I/II
Betriebssystem	Linux 2.6 mit RT-Preempt Patch
Hauptspeicher (RAM)	256 MB
Interner Speicher (Flash)	512 MB
Remanentspeicher (Retain)	1023 kB
Grafik	DVI, 1280x1024; LCD/Panel link
Anwenderspezifische Echtzeitunterstützung	1024 kB PLC-SRAM mit Batterie-Backup; NMI-Timer

4.6.3 Versorgung

Tabelle 8: Technische Daten – Versorgung

Spannungsversorgung	DC 24 V (-25 % ... + 30 %)
Eingangsstrom	770 mA
Summenstrom für Busklemmen	1000 mA

4.6.4 Kommunikation

Tabelle 9: Technische Daten – Kommunikation

LAN	2 x10Base-T/100Base-TX
I/O-Schnittstelle, seriell	Ein 9-poliger D-Sub-Stecker gemäß EIA RS-232
I/O-Schnittstelle, USB	Zwei USB-Schnittstellen gemäß Spezifikation 2.0
I/O-Schnittstelle, USB	Zwei USB-Schnittstellen gemäß Spezifikation 2.0
Feldbus	CANopen, Master

4.6.5 Schutz und Sicherheit

Tabelle 10: Technische Daten – Schutz und Sicherheit

Schutzart	IP20 gemäß EN 60529
-----------	---------------------

4.6.6 Laufzeitsystem

Tabelle 11: Technische Daten – Laufzeitsystem

Programmierung	CODESYS 2.3 (WAGO-I/O-PRO CAA)
IEC 61131-3	AWL, KOP, FUP, ST, AS

4.6.7 Umgebungsbedingungen

Tabelle 12: Technische Daten – Umgebungsbedingungen

Vibrationsfestigkeit	Gemäß IEC 60068-2-6
Feuchtigkeit	5 – 95 % ohne Betauung
Lagertemperatur	-10 °C ... +85 °C
Betriebstemperatur	0 °C ... +55 °C

4.6.8 Anschlussstechnik

Tabelle 13: Technische Daten – Anschlussstechnik

Anschlussstechnik	CAGE CLAMP®
Querschnitte	0,08 mm ² – 2,5 mm ² /AWG 28 – 14
Abisolierlänge	8 – 9 mm/0,33 Inch

4.7 Normen und Richtlinien

4.7.1 Elektromagnetische Verträglichkeit

Tabelle 14: Technische Daten – Elektromagnetische Verträglichkeit

Grenzwerte für Störaussendung und Betrieb	Gemäß DIN EN 61000-6-4
Grenzwerte für Störfestigkeit und Betrieb	Gemäß DIN EN 61000-6-2

4.8 Zulassungen


Information



Weitere Informationen zu Zulassungen


Detaillierte Hinweise zu den Zulassungen können Sie dem Dokument „Übersicht Zulassungen **WAGO-I/O-SYSTEM 750**“ entnehmen. Dieses finden Sie im Internet unter: www.wago.com → Service → Downloads → Zusätzliche Dokumentation und Information für Automatisierungsprodukte → WAGO-I/O-SYSTEM 750 → Systembeschreibung.

Folgende Zulassungen wurden für den I/O-IPC 758-874/000-112 erteilt:

 Konformitätskennzeichnung

 cUL_{US} UL508

Folgende Ex-Zulassungen wurden für den I/O-IPC 758-874/000-112 erteilt:

 TÜV 07 ATEX 554086 X
I M2 Ex d I
II 3 G Ex nA IIC T4
II 3 D Ex tD A22 IP6X T135°C
IECEX TUN 09.0001 X
Ex d I
Ex nA IIC T4
Ex tD A22 IP6X T135°C

5 Beschreibung der I/O-IPC-Schnittstellen

5.1 ETHERNET-Schnittstellen (X8, X9)

Die beiden ETHERNET-Schnittstellen vom Typ RJ-45 basieren auf dem Übertragungsstandard 10/100BASE-T. Dadurch ermöglichen sie, in Abhängigkeit des verwendeten ETHERNET-Netzwerks, einen Datenaustausch mit einer Übertragungsrate von jeweils 10 Mbit/s oder 100 Mbit/s im Halbduplex- sowie Vollduplexbetrieb.

Die LEDs „ACT/LNK“ und „Speed“ der beiden Ethernet-Schnittstellen geben den aktuellen Betriebsstatus an:

Tabelle 15: ACT/LNK- und Speed-LED

LED	Farbe/Staus	Ursache
ACT/LNK	gelb	Verbindung zum Ethernet-Netzwerk vorhanden
	gelb blinkend	Datenaustausch wird durchgeführt.
Speed	aus	Übertragungsgeschwindigkeit 10 Mbit/s
	grün	Übertragungsgeschwindigkeit 100 Mbit/s

Zum Anschluss des I/O-IPC über Ethernet an einen PC haben Sie folgende Möglichkeiten:

- Direkt, mithilfe eines Cross-over-Kabels
 - Über einen Switch oder Hub in Verbindung mit einem Patchkabel
- Die nachfolgende Tabelle gibt Ihnen Auskunft zu der Anschlussbelegung der Ethernet-Schnittstellen:

Tabelle 16: Ethernet-Schnittstellen: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
 <p>Abbildung 6: RJ-45 CP</p>	1	TD+
	2	TD-
	3	RD+
	4	Nicht belegt
	5	Nicht belegt
	6	RD-
	7	Nicht belegt
	8	Nicht belegt

5.2 Schnittstelle für die Elektronikversorgung (X4)

Über diese Schnittstelle speisen Sie die Elektronikversorgung für den I/O-IPC und für die am Klemmenbus angeschlossenen Busklemmen ein.

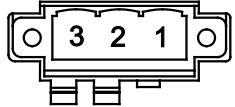
GEFAHR



Elektrische Spannung!

Betreiben Sie den I/O-IPC ausschließlich mit 24 V DC PELV- (Protective Extra Low Voltage) oder SELV-Spannungsquellen (Safety Extra Low Voltage). Bei Nichtbeachtung besteht die Gefahr eines Stromschlags.

Tabelle 17: Schnittstelle für die Elektronikversorgung: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
 Abbildung 7: Elektronikversorgung (X4)	1	V_IN (+)
	2	GND (-)
	3	Schirm (optional)

5.3 CANopen-Schnittstelle (X3)

Der Feldbus dient zur Kommunikation zwischen dem I/O-IPC und den daran angeschlossenen CANopen-Feldbuskopplern (Slaves).

ESD




Offene Schnittstelle!

Wenn Sie die Schnittstelle nicht benötigen, verschließen Sie diese mit der mitgelieferten Schutzkappe, um mögliche Schäden durch elektrostatische Entladung zu vermeiden.

Die nachfolgende Tabelle gibt Ihnen Auskunft zur Belegung der CANopen-Schnittstelle.

Tabelle 18: CANopen-Schnittstelle: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
 <p>Abbildung 8: Schnittstelle X3</p>	1	NC
	2	CAN-
	3	CAN_GND
	4	NC
	5	NC
	6	NC
	7	CAN+
	8	NC
	9	CAN_+5V

5.4 Integrierte Ein- und Ausgänge (X5)

Die 12-polige D-Sub-Buchse stellt zwei integrierte digitale Eingänge und zwei Ausgänge zur Verfügung. Diese dient zum direkten Anschluss von Sensoren oder Aktoren.

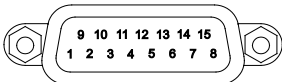
Hinweis



Für integrierte Ein-/Ausgänge nur geschirmte Leitungen verwenden!
Beachten Sie bei der Verwendung der integrierten Ein- und Ausgänge, dass diese nicht den Anforderungen der IEC-61131-2 genügen und nur der Anschluss von geschirmten Leitungen zulässig ist.

Die nachfolgende Tabelle gibt Ihnen Auskunft zur Belegung der integrierten Ein- und Ausgänge:

Tabelle 19: Digitale Ein- und Ausgänge: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
	1	DIN0+
	2	DIN1+
	3	DOUT0+
	4	DOUT1+
	5	Nicht belegt
	6	Nicht belegt
	7	Nicht belegt
	8	Nicht belegt
	9	DIN0-
	10	DIN1-
	11	DOUT0-
	12	DOUT1-
	13	Nicht belegt
	14	Nicht belegt
	15	Nicht belegt

Digitale Eingänge

Die zwei digitalen Eingänge sind autark vom Klemmenbus. Dadurch werden digitale Signale auch bei einer Störung des Klemmenbusses verarbeitet.

Spannungsbereich	Low: -3 V ... +5 V High: +11 V ... +30 V (+24-V-Standard)
Max. Strom pro Kanal	5 mA
Kanäle	2
Eingangsimpedanz	Min. 1,5 k Ω Max. 6 k Ω bei 30 V
Merkmale	Optokoppler, 2 kV Tiefpassfilter, 10 kHz Strombegrenzung Überspannungsschutz Verpolungsschutz

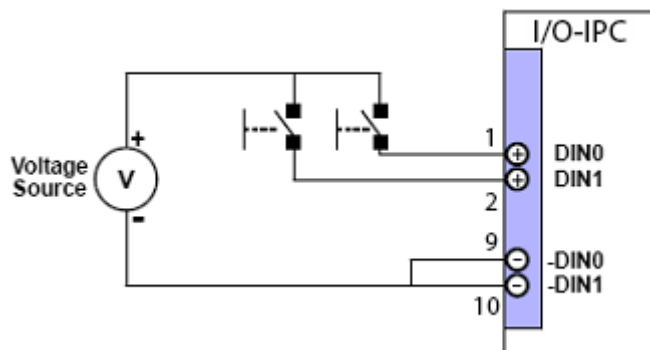


Abbildung 9: Anschluss der integrierten Eingänge

Digitale Ausgänge

ACHTUNG



Höchste Strombelastbarkeit der Ausgangskanäle: 0,1 A!

Beachten Sie für die digitalen Ausgangskanäle die maximale Strombelastbarkeit von 0,1 A. Eine Erhöhung des Stroms führt zur Überhitzung des Ausgangstreibers und zu Schäden am I/O-IPC.

Die zwei digitalen Ausgänge sind autark vom Klemmenbus. Dadurch werden digitale Signale auch bei einer Störung des Klemmenbusses verarbeitet.

Externe Spannungsquelle, max.	24 V DC
Spannungsbereich	Abhängig von externer Schaltung
Max. Strom pro Kanal	0,1 A (typ.)
Kanäle	2
Merkmale	Optokoppler 2 kV

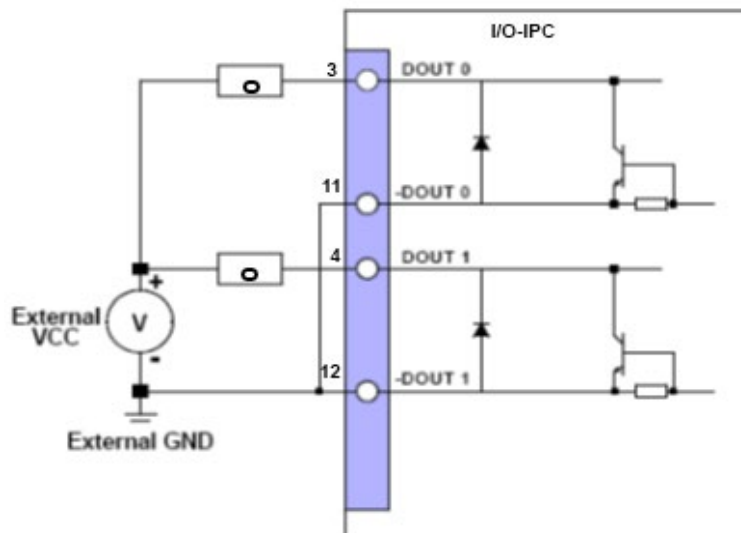


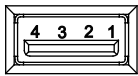
Abbildung 10: Anschluss der integrierten Ausgänge

5.5 USB-Schnittstellen (X10, X11)

Die USB-Schnittstellen dienen dem Anschluss von USB-Geräten, wie zum Beispiel USB-Speichern. Wenn das angeschlossene USB-Gerät nicht verwendet wird, können Sie dieses zu jeder Zeit entfernen. Bei einem angeschlossenen USB-Speicher ist darauf zu achten, dass Sie die geöffneten Dateien schließen, bevor Sie den USB-Speicher entfernen.

Die nachfolgende Tabelle gibt Ihnen Auskunft zu der Belegung der USB-Schnittstellen:

Tabelle 20: USB-Schnittstellen: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
 Abbildung 11: USB-Schnittstelle	1	USB_VCC1
	2	USB_N1
	3	USB_P1
	4	USB_GND

ACHTUNG **Betrieb einer externen Festplatte ist nicht möglich!**



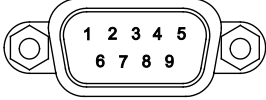
Der Betrieb einer externen Festplatte mit USB-Anschluss ist am I/O-IPC nicht möglich.

Es kann nicht sichergestellt werden, dass die externe Festplatte über den USB-Anschluss mit dem erforderlichen Strom versorgt wird. Dadurch können interne Schäden an den Geräten auftreten.

5.6 Serielle RS-232-Schnittstelle (X6)

Die nachfolgende Tabelle gibt Ihnen Auskunft zu der Belegung der RS-232-Schnittstelle:

Tabelle 21: RS-232-Schnittstelle: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
 <p>Abbildung 12: RS-232-Schnittstelle</p>	1	DCD1
	2	RXD1
	3	TXD1
	4	DTR1
	5	GND
	6	DSR1
	7	RTS1
	8	CTS1
	9	RI1



ESD

Offene Schnittstelle!

Wenn Sie die Schnittstelle nicht benötigen, verschließen Sie diese mit der mitgelieferten Schutzkappe, um mögliche Schäden durch elektrostatische Entladung zu vermeiden.

Darüber können Sie mit dieser Schnittstelle folgende Anwendungen und Dienste nutzen:

- I/O-Check
- MODBUS-RTU
- CODESYS 2.3
- Linux-Konsole
 - Deutsche Tastaturbelegung
 - Englische Tastaturbelegung

Es kann immer nur einer der Anwendungen oder Dienste auf die RS-232-Schnittstelle zugreifen. Diese eindeutige Zuweisung lässt sich u. a. mit dem Web-based Management durchführen. Siehe dazu Kapitel „Seite Administration“.

Hinweis



Systemstart

Während des Systemstarts dürfen angeschlossene Geräte keine Daten an die RS-232-Schnittstelle senden, da andernfalls die Firmware nicht startet.

Wenn dieses jedoch nicht auszuschließen ist, kommentieren Sie in der Datei menu.lst (/boot/grub/menu.lst) den Parameter „serial“ aus:

```
01 #serial --unit=0 --speed=115200
```

```
02 terminal --timeout=2 console
```

Der Bootloader reagiert dann nicht auf die an der RS-232-Schnittstelle eingegebenen Daten.

5.7 DVI-I-Schnittstelle (X7)

Diese Schnittstelle überträgt analoge sowie digitale Signale und eignet sich zum Anschluss an digitalen Monitoren.

Die DVI-I-Schnittstelle überträgt auch analoge Bild-Signale, sodass der Anschluss von CRT-VGA-Monitoren unter Verwendung eines DVI-to-VGA-Adapters möglich ist.

ESD



Offene Schnittstelle!

Wenn Sie die Schnittstelle nicht benötigen, verschließen Sie diese mit der mitgelieferten Schutzkappe, um mögliche Schäden durch elektrostatische Entladung zu vermeiden.

Die nachfolgende Tabelle gibt Ihnen Auskunft zu der Belegung der DVI-I-Schnittstelle:

Tabelle 22: DVI-I-Schnittstelle: Anschlussbelegung

Anschluss	Kontakt	Beschreibung
	1	TXD2-
	2	TXD2+
	3	GND
	4	Nicht belegt
	5	Nicht belegt
	6	DDCCLK
	7	DDCDATA
	8	CRT_VSY
	C1	CRT_R
	C4	CRT_HSY
	9	TXD1-
	10	TXD1+
	11	GND
	12	Nicht belegt
	13	Nicht belegt
	14	VCC_DVI
	15	GND
	16	Nicht belegt
	C2	CRT_G
	C5	GND
	17	TXD0-
	18	TXD0+
	19	GND
	20	Nicht belegt
	21	Nicht belegt
	22	GND
	23	TXCP
	24	TXCN
	C3	CRT_B
	C5A	GND

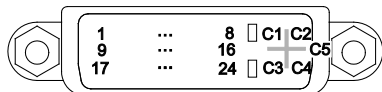


Abbildung 13: DVI-Schnittstelle

6 Montage und Demontage des I/O-IPC

Der I/O-IPC hat eine erhöhte Wärmeentwicklung. Die überschüssige Wärme wird über die passive Wärmeabführung (Aluminium-Unterseite des I/O-IPC und Tragschiene) abgeführt. Insofern ist die Montage als wärmeleitende Verbindung immer auf einer Tragschiene durchzuführen.

6.1 Hinweise zur Montage/Demontage

Nachfolgende Hinweise sind stets zu beachten:

ACHTUNG



Belüftung des Einbauorts

Beim Einbau des I/O-IPC ist seitlich und nach oben mindestens ein Freiraum von 40 mm zu lassen, um eine ausreichende Wärmeabführung zu erreichen. Am Einbauort darf die Umgebungstemperatur während des Betriebs +55 °C nicht übersteigen.

ESD



Offene Schnittstelle!

Nicht benötigte Schnittstellen des I/O-IPC sind durch die mitgelieferten Schutzkappen zu verschließen, um mögliche Schäden durch elektrostatische Entladung zu vermeiden.

- Wählen Sie eine ausreichend stabile Tragschiene aus und nutzen Sie ggf. für diese mehrere Montagepunkte (alle 20 cm), um ein Durchbiegen und Verdrehen der Tragschiene durch den I/O-IPC zu verhindern.
- Verwenden Sie bei Nutzung von Tragschienen mit einer Höhe von 7,5 mm flache Nieten oder Senkkopfschrauben. Andernfalls lassen sich der I/O-IPC und die daran angeschlossenen Busklemmen nicht korrekt auf die Tragschiene montieren.
- Achten Sie bei der Montage darauf, dass Sie die physikalischen Schnittstellen nicht verschmutzen. Dies kann zu Beschädigung und Korrosion der Kontakte führen.
- Um eine Beschädigung des I/O-IPC zu vermeiden, montieren Sie ihn nicht in Scherbereichen von beweglichen Anlagen- oder Maschinenteilen.
- Sorgen Sie für einen angemessenen Potenzialausgleich in Ihrer Anlage.
- Die Tragschienen sind mit dem geerdeten Einbauort leitend zu verschrauben.

6.2 Benötigtes Zubehör für die Montage

Zur Montage des I/O-IPC benötigen Sie

- gelochte oder ungelochte Tragschienen nach EN 60715 sowie
- eine Endklemme 750-600.

6.3 Zulässige Einbaurichtungen des I/O-IPC

Der I/O-IPC ist waagrecht oder senkrecht auf eine Tragschiene zu montieren, die eine wärmeleitende Verbindung zum Befestigungsort hat. Bei der senkrechten Montage sind geeignete Maßnahmen zu treffen, wie beispielsweise eine Abrutschsicherung (B), damit der I/O-IPC bei Vibrations- und Schockbelastungen nicht herunterrutscht.

Hinweis



Wärmeabführung

Um eine gute Wärmeabführung des I/O-IPC zu erreichen, empfehlen wir die Montage A 1 in Abb. 10.

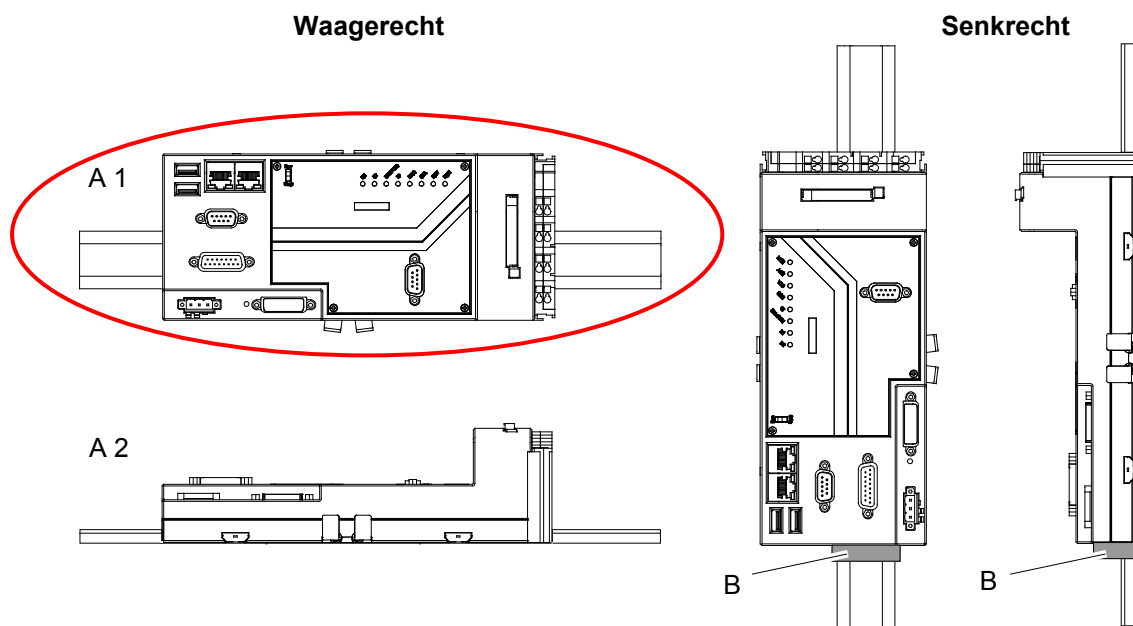


Abbildung 14: Einbaurichtungen des I/O-IPC; empfohlene Einbaurichtung (A 1)

6.4 Befestigung des I/O-IPC auf einer Tragschiene

Für die Montage des I/O-IPC auf einer Tragschiene befinden sich an der Unterseite vier Klemmhebel, die den I/O-IPC auf der Tragschiene festhalten.

Hinweis



Befestigung der Tragschienen

Wenn Sie Tragschienen mit einer Höhe von 7,5 mm einsetzen, dann verwenden Sie flache Nieten oder abgesenkte Schrauben zur Befestigung der Tragschiene. Andernfalls lassen sich der I/O-IPC und die daran angeschlossenen Busklemmen nicht korrekt auf die Tragschiene montieren.

Zur Montage gehen Sie wie nachfolgend beschrieben vor:

1. Trennen Sie denjenigen Anlagenteil von der Stromversorgung, an dem Sie den I/O-IPC montieren.
2. Zum Aufsetzen auf die Tragschiene (25) drücken Sie gleichzeitig die vier Hebel der Tragschienenbefestigung (20) zusammen. Drücken Sie den I/O-IPC auf die Tragschiene und lassen Sie die Hebel los.
3. Kontrollieren Sie den festen Sitz des I/O-IPC auf der Tragschiene. Alle vier Klemmkeile (23) müssen hinter der Tragschienenkante eingerastet sein.
4. Ist eine Seite nicht eingerastet, pressen Sie die entsprechenden Hebel zusammen, drücken die nicht eingerastete Seite auf die Tragschiene und lassen danach die Hebel wieder los.

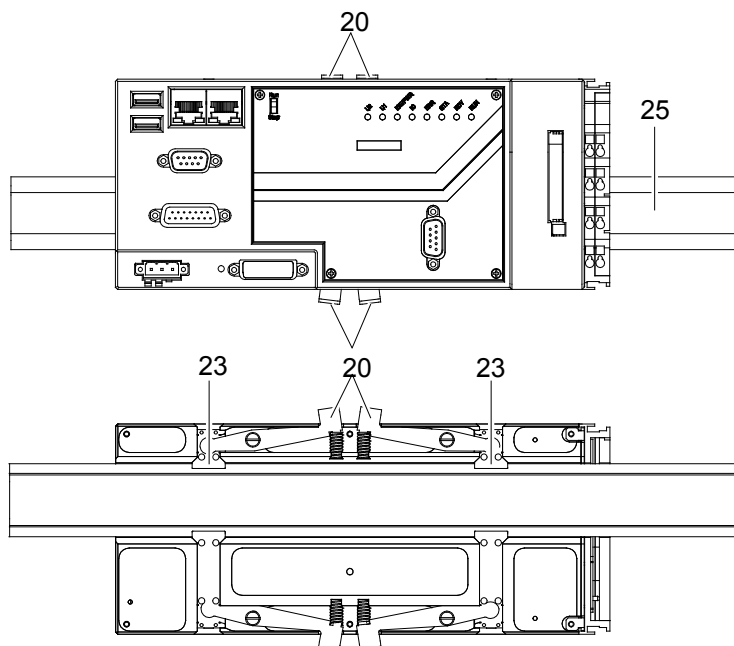


Abbildung 15: Befestigung des I/O-IPC auf einer Tragschiene

6.5 Anstecken der Busklemmen an den I/O-IPC

Nach der Montage des I/O-IPC auf einer Tragschiene befestigen Sie die für Ihre Anwendung erforderlichen Busklemmen am I/O-IPC. Sie können an diese bis zu 64 Busklemmen der Serie 750/753 anstecken. Die Anzahl ist dabei abhängig von der Gesamtlänge der angesteckten Busklemmen. Diese darf maximal 780 mm einschließlich der Endklemme betragen und die maximale Größe des Prozessabbaus für die Ein- und Ausgangsdaten darf jeweils 500 Byte nicht überschreiten.

Beispiel zur Gesamtlänge:

Haben die einzelnen Busklemmen eine Breite von 12 mm, sind 64 Stück steckbar, bei einer Breite von 24 mm jedoch nur noch 32 Busklemmen.

Mit der optionalen WAGO-Klemmenbusverlängerung (bestehend aus Kopplerklemme 750-628 und Endklemme 750-627) ist es möglich, bis zu 250 Busklemmen zu nutzen. Hierbei gelten dieselben Einschränkungen wie für die Verwendung von 64 Busklemmen.

Information



Verwendung der WAGO-Klemmenbusverlängerung

Informationen zur Verwendung der WAGO-Klemmenbusverlängerung erhalten Sie in den Dokumentationen 750-627 und 750-628, die auf der WAGO-Internetseite erhältlich sind.

Information



Weiterführende Informationen und Anwendungshinweise zur Verwendung der WAGO-Busklemmen

Weiterführende Informationen und Anwendungshinweise zur Verwendung der WAGO-Busklemmen erhalten Sie in der Systembeschreibung des WAGO-I/O-SYSTEMS 750/753, den dazugehörigen Handbüchern und Datenblättern unter www.wago.com.

Zum Anstecken der Busklemmen gehen Sie wie nachfolgend beschrieben vor:

1. Trennen Sie denjenigen Anlagenteil von der Stromversorgung, an dem Sie den I/O-IPC montieren.
2. Stecken Sie die Busklemmen jeweils mit der Nut (71) in die Feder (70) der vorherigen.

GEFAHR**Elektrische Spannung!**

Bei Verwendung der 120/230-V-Busklemmen beachten Sie die Sicherheitshinweise im dazugehörigen Handbuch. Bei Nichtbeachtung besteht die Gefahr eines Stromschlags.

ACHTUNG**Höchste Strombelastbarkeit der Leistungskontakte: 10 A!**

Die maximale Strombelastbarkeit der Leistungskontakte der Busklemmen darf 10 A nicht überschreiten. Eine Erhöhung des Stroms kann zur Überhitzung der Kontakte und zu Schäden an den Busklemmen führen.

3. Stecken Sie als letztes die Endklemme an.

Durch das Anrasten einer weiteren Busklemme wird die Versorgungsspannung für die Sensoren und Aktoren automatisch über die Leistungskontakte weitergeleitet. Voraussetzung dafür ist, dass Sie Busklemmen verwenden, die über beidseitige Leistungskontakte verfügen.

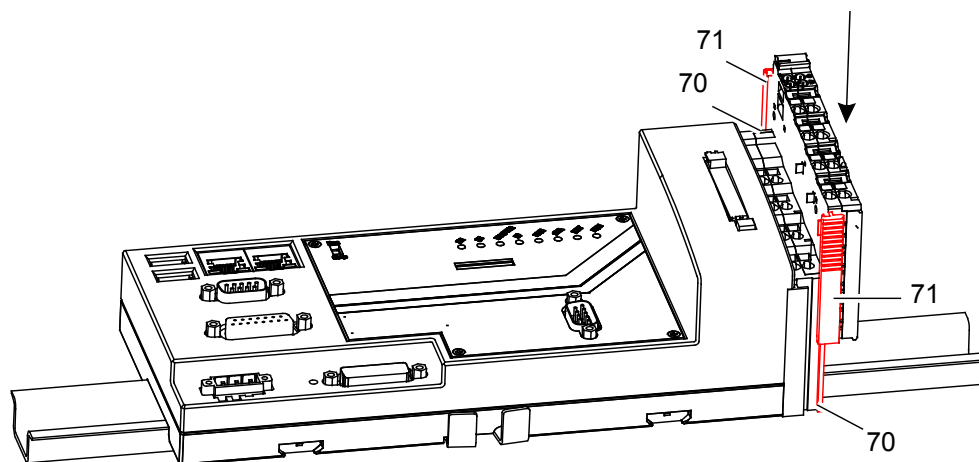


Abbildung 16: Anstecken einer Busklemme an der Klemmenbusschnittstelle des I/O-IPC

6.6 Demontage des I/O-IPC

Zum Austauschen eines I/O-IPC, z. B. bei einem Variantenwechsel, gehen Sie wie in den nachfolgenden Kapiteln beschrieben vor.

VORSICHT Heiße Oberfläche!



Während des Betriebs können hohe Oberflächentemperaturen am Gehäuse des I/O-IPC auftreten. War der I/O-IPC in Betrieb, lassen Sie ihn abkühlen, bevor Sie diesen berühren.

6.6.1 Entfernen der Leitungen

Zum Entfernen der Leitungen vom I/O-IPC gehen Sie folgendermaßen vor:

1. Trennen Sie denjenigen Anlagenteil von der Stromversorgung, an dem Sie den I/O-IPC demontieren.
2. Lösen Sie die Arretierungsschrauben an den Steckverbindern der Datenkabel und ziehen Sie anschließend die Steckverbinder von den Schnittstellen des I/O-IPC ab.
3. Entfernen Sie ggf. die Leitungen der ersten, an der Klemmenbusschnittstelle angeschlossenen Busklemme (A).

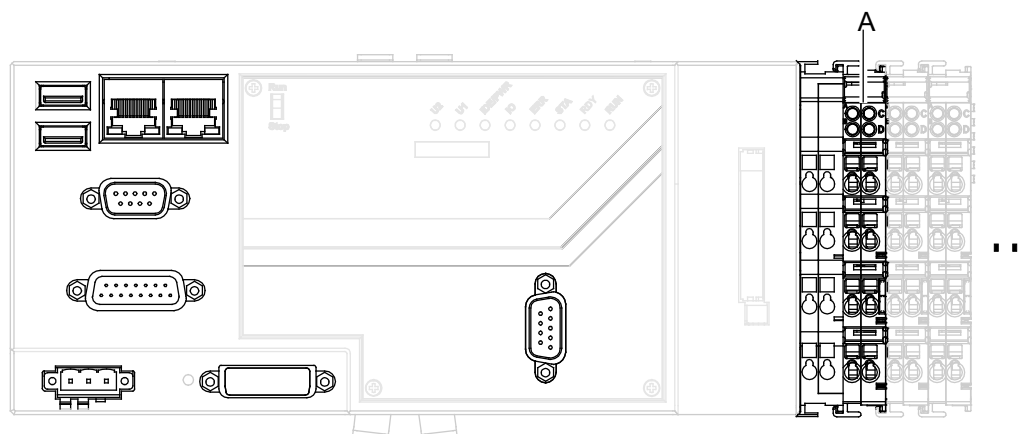


Abbildung 17: Schnittstellen des I/O-IPC

6.6.2 Demontage des I/O-IPC von der Tragschiene

Gehen Sie zur Demontage des I/O-IPC von der Tragschiene wie nachfolgend beschrieben vor:

1. Trennen Sie denjenigen Anlagenteil von der Stromversorgung, an dem Sie den I/O-IPC montieren.
2. Zum Abnehmen des I/O-IPC von der Tragschiene entfernen Sie die erste angeschlossene Busklemme. Ziehen Sie diese dazu an der orangefarbenen Entriegelungslasche (24) von der Tragschiene ab (A und B).
3. Zum Abnehmen des I/O-IPC von der Tragschiene drücken Sie gleichzeitig mit beiden Händen die vier Hebel der Tragschienenbefestigung (20) bis zum Anschlag zusammen und ziehen den I/O-IPC ab (C).

ACHTUNG



Klemmenbus-Schnittstelle

Die Klemmenbus-Schnittstelle (Position 5) darf nicht demontiert werden. Diese Busklemme ist Bestandteil des I/O-IPCs.

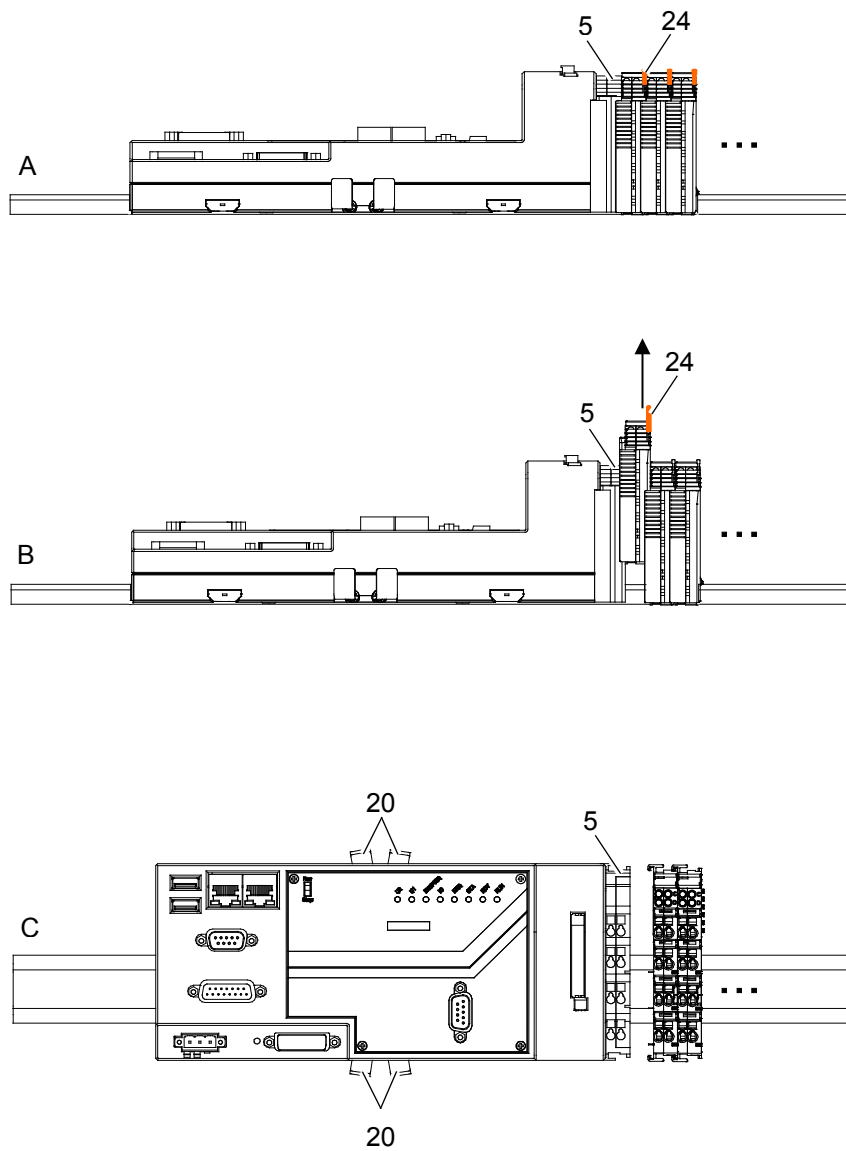


Abbildung 18: I/O-IPC von der Tragschiene entfernen

7 Versorgungsspannung anschließen

Je nach Anwendungsbereich des I/O-IPC ist die Versorgungsspannung wie folgt anzuschließen:

Hinweis



Lieferumfang

Die Potentialeinspeiseklemme 750-602 ist ab HW-Version 11 nicht mehr im Lieferumfang enthalten.

Tabelle 23: Verwendung von 750-602/626 in Abhängigkeit des I/O-IPC-Anwendungsbereichs

Anwendungsbereich	Filterklemme	Anschluss
Leitungslänge < 3 m	750-602	Siehe Kapitel „Einspeisung bei Verwendung der Potential-einspeiseklemme 750-602“
Leitungslänge > 3 m	750-626	Siehe Kapitel „Einspeisung bei Verwendung der Filterklemme 750-626“

7.1 Hinweise

GEFAHR



Elektrische Spannung!

Betreiben Sie den I/O-IPC ausschließlich mit 24 V DC PELV- (Protective Extra Low Voltage) oder SELV-Spannungsquellen (Safety Extra Low Voltage). Bei Nichtbeachtung besteht die Gefahr eines Stromschlags.

Hinweis



Unterbrechung der Versorgungsspannung

Bei einer Unterbrechung der Versorgungsspannung von mehr als 1 ms wird der I/O-IPC zurückgesetzt und automatisch ein Neustart des I/O-IPC durchgeführt.

- Zur Sicherstellung der galvanischen Trennung ist die Verwendung von jeweils einem Netzteil für den Anschluss der Elektronik- und Feldversorgung erforderlich.
- Schließen Sie die Leitungen für die Versorgungsspannung nur im spannungsfreien Zustand an.
- Halten Sie mit den Leitungen der Versorgungsspannung genügend Abstand zu elektromagnetischen Störquellen ein, um eine hohe Störfestigkeit der 750-Komponenten gegen elektromagnetische Störstrahlungen zu erzielen.
- Achten Sie beim Verlegen sämtlicher Leitungen darauf, dass Sie diese nicht in Scherbereichen von beweglichen Maschinenteilen verlegen.
- Achten Sie auf die korrekte Auslegung des Potenzialausgleichs.

7.2 Benötigtes Zubehör

Zum Anschluss der Versorgungsspannung an den I/O-IPC benötigen Sie ggf. die Filterklemme 750-626. Diese können Sie unter www.wago.com bestellen. Das zum Anschluss der Versorgungsspannung benötigte Zubehör (z. B. einzelne Leitungen) und Werkzeug ist von Ihnen bereitzustellen.

7.3 Einspeisung bei Verwendung der Potentialeinspeiseklemme 750-602

Hinweis



Lieferumfang

Die Potentialeinspeiseklemme 750-602 ist ab der HW-Version 11 nicht mehr im Lieferumfang enthalten.

Hinweis



Länge der Leitung für die Versorgungsspannung

Die Leitung für die Versorgungsspannung darf bei dieser Einspeisevariante eine Länge von 3 m zwischen Spannungsquelle und I/O-IPC nicht überschreiten. Falls eine längere Leitung benötigt wird, dürfen Sie die Versorgungsspannung nur wie in Kapitel „Einspeisung bei Verwendung der 750-626“ beschrieben anschließen.

ACHTUNG



Korrekten Leiterquerschnitt beachten!

Verwenden Sie ausschließlich für den Anschluss X4 und für die CAGE-CLAMP®-Anschlüsse der Potentialeinspeiseklemme Leiterquerschnitte von 0,08 mm² ... 2,5 mm² (AWG 28 ... 14).

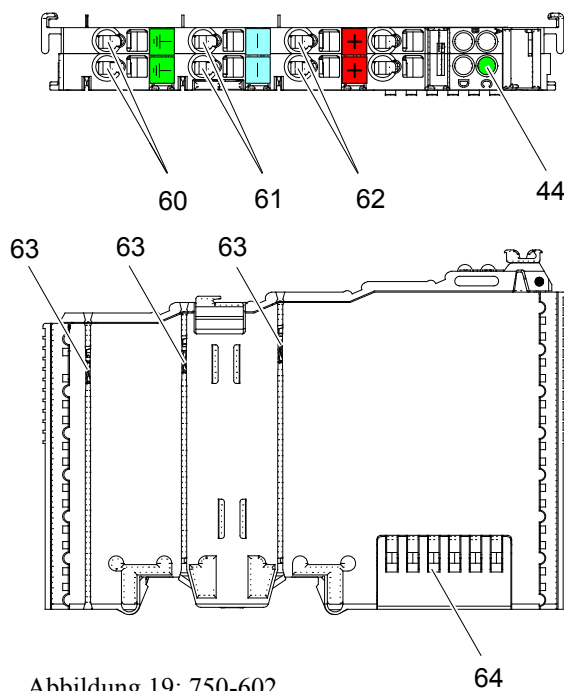


Abbildung 19: 750-602

Tabelle 24: Anschlüsse, Kontakte und LEDs der Einspeiseklemme

Position	LED/ Bedienelement	Farbe/ Status	Bedeutung
60	Schutzleiter	-	Anschluss von Schutzerde
61	0 V DC	-	Masse (GND) der Versorgungsspannung
62	Feldversorgung, 24 V DC	-	24-V-Versorgungsspannung für die Sensoren/Aktoren. Der Anschluss der Versorgungsspannung ist gegen Verpolung geschützt. Zur Einspeisung anderer Feldpotentiale, z. B. AC 230 V, stehen dafür vorgesehene Potentialeinspeiseklemmen zur Verfügung. Detaillierte Informationen darüber erhalten Sie in den Handbüchern der Einspeiseklemmen und in der Systembeschreibung 750-xxxx.
63	Leistungskontakte	-	Weiterführung der feldseitigen Versorgungsspannung zu den angeschlossenen Busklemmen.
64	Datenkontakte	-	Diese stellen die Versorgungsspannung (5 V, 1 A) für die Elektronik der am Klemmenbus angeschlossenen Busklemmen zur Verfügung. Die Spannungsversorgung erfolgt durch den I/O-IPC (Anschluss X4).
44	LED	Grün/aus	Die LED leuchtet, wenn die Feldversorgung vorhanden (siehe Pos. 62) ist. Andernfalls ist die LED aus.

Vorbereitung zum Anschluss der Versorgungsspannung:

- Sie haben jeweils zwei Versorgungsleitungen an zwei Spannungsquellen von +24 V DC und 0 V DC angeschlossen – bei ausgeschalteter Spannungsversorgung.
- Sie haben die mitgelieferte Buchse für den Anschluss X4 an die Versorgungsleitung folgendermaßen angeschlossen:

Tabelle 25: Anschluss für die Elektronikversorgung: Anschlussbelegung

Anschluss X4	Kontakt	Beschreibung
 <p>Abbildung 20: Elektronikversorgung (X4)</p>	1	V_IN (+)
	2	GND (-)
	3	Schirm (optional)

Zum Anschluss der Elektronikversorgung des I/O-IPC und der Feldversorgung für die angeschlossenen Busklemmen, Sensoren und Aktoren gehen Sie folgendermaßen vor:

1. Trennen Sie denjenigen Anlagenteil von der Stromversorgung, an dem Sie den I/O-IPC montieren.
2. Schließen Sie die Elektronikversorgung an den I/O-IPC an, indem Sie die mit der Versorgungsleitung verbundene Buchse auf den Anschluss X4 (10) stecken. Sehen Sie eine Absicherung von 1,6 A vor.
3. Sichern Sie anschließend die Buchse mittels der dazugehörigen Schrauben.
4. Schließen die Feldversorgung an die Potentialeinspeiseklemme 750-602 (6) an, indem Sie gemäß der Abbildung 24 V mit „+“ und 0 V mit „-“ verbinden. Sehen Sie eine Absicherung von 10 A vor.

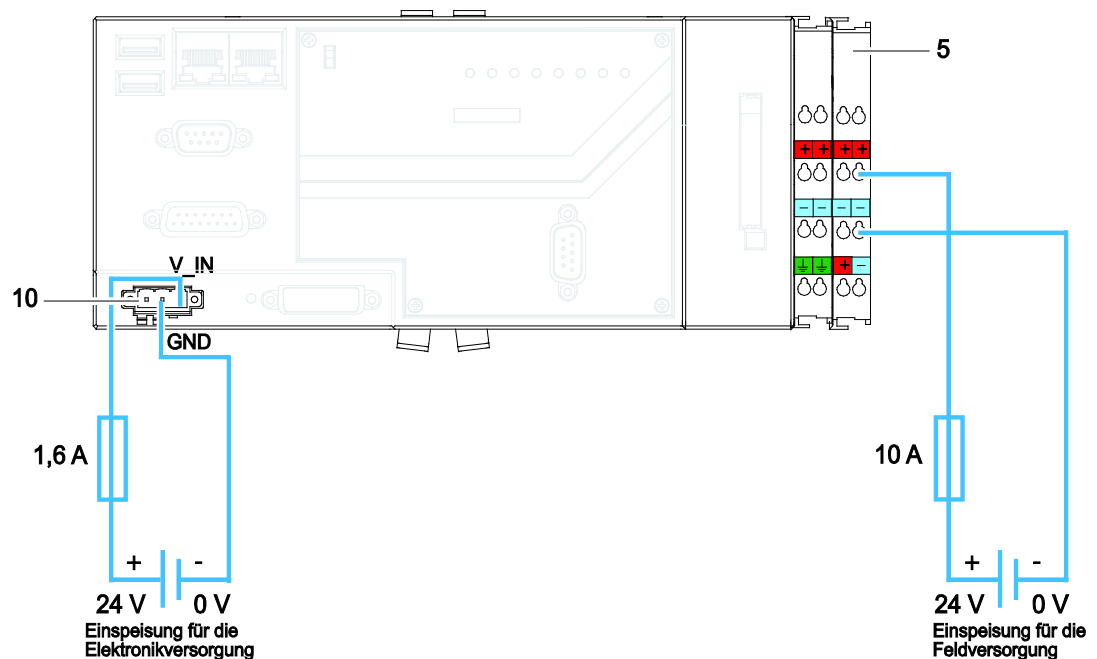


Abbildung 21: Einspeisung bei Verwendung 750-602 bis HW10 (mit Feldbus)

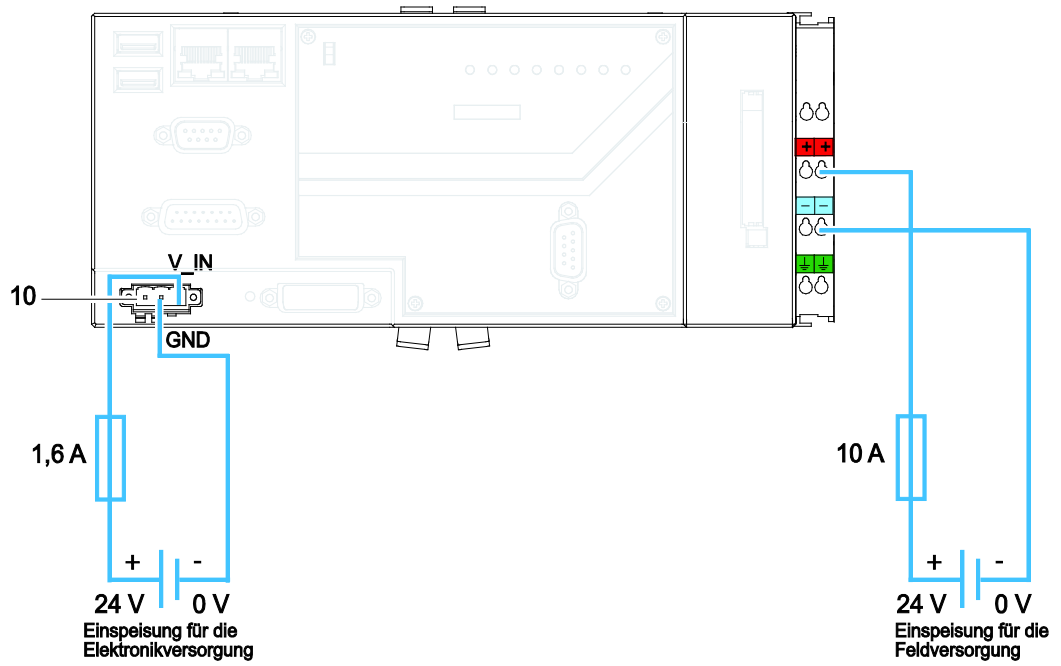


Abbildung 22: Einspeisung bei Verwendung 750-602 ab HW11 (mit Feldbus)

7.4 Einspeisung bei Verwendung der Filterklemme 750-626

Sie benötigen die Filterklemme 750-626, sobald eine Versorgungsleitung die Länge von 3 m zwischen Spannungsquelle und I/O-IPC überschreitet.

ACHTUNG Strombelastbarkeit beachten!



Für diese Einspeisevariante benötigen Sie die Filterklemme 750-626 **ab HW-Version 4**. Nur diese ist für die höhere Strombelastung des I/O-IPC ausgelegt. Die Filterklemme erhalten Sie unter www.wago.com.

ACHTUNG Isolationsspannung beachten!



Bei Verwendung der Filterklemme 750-626 verringert sich die Isolationsspannung der Feld- und Elektronikversorgung gegen PE auf 50 V.

ACHTUNG Korrekten Leiterquerschnitt beachten!



Verwenden Sie ausschließlich für die CAGE-CLAMP®-Anschlüsse der Filterklemme Leiterquerschnitte von 0,08 mm² ... 2,5 mm² (AWG 28 ... 14).

Vorbereitung zum Anschluss der Versorgungsspannung:

- Sie haben jeweils zwei Versorgungsleitungen an zwei Spannungsquellen von +24 V DC und 0 V DC angeschlossen – bei ausgeschalteter Spannungsversorgung.
- Sie haben die mitgelieferte Buchse für den Anschluss X4 an die Versorgungsleitung folgendermaßen angeschlossen:

Tabelle 26: Anschluss für die Elektronikversorgung: Anschlussbelegung

Anschluss X4	Kontakt	Beschreibung
	1	V_IN (+)
	2	GND (-)
	3	Schirm (optional)

Zum Anschluss der Elektronikversorgung des I/O-IPC und der Feldversorgung für die angeschlossenen Busklemmen, Sensoren und Aktoren gehen Sie folgendermaßen vor:

1. Trennen Sie denjenigen Anlagenteil von der Stromversorgung, an dem Sie den I/O-IPC montieren.
2. Schließen Sie die Elektronikversorgung für den I/O-IPC an die Filterklemme 750-626 (6) an, indem Sie gemäß der Abbildung 24 V mit „+“ und 0 V mit „-“ verbinden. Sehen Sie eine Absicherung von 1,6 A vor.
3. Leiten Sie die Elektronikversorgung von der Filterklemme (6) zum I/O-IPC, indem Sie die mit der Versorgungsleitung verbundene Buchse auf den Anschluss X4 (10) stecken.
4. Schließen Sie die Feldversorgung an die Filterklemme 750-626 (6) an, indem Sie gemäß der Abbildung 24 V mit „+“ und 0 V mit „-“ verbinden. Sehen Sie eine Absicherung von 10 A vor.

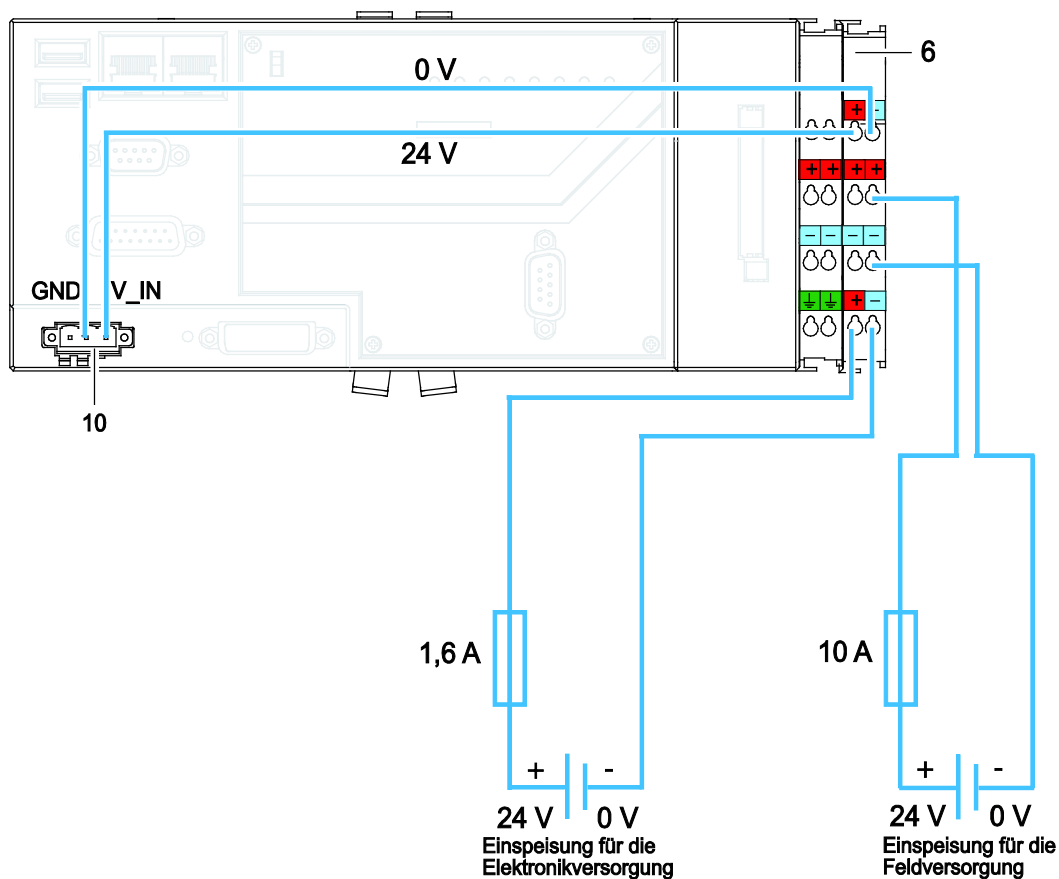


Abbildung 23: Einspeisung bei Verwendung 750-626 (mit Feldbus)

7.5 Sensor- und Aktorleitung an die Busklemmen anschließen

Weiterführende Informationen und Anwendungshinweise zur Verdrahtung einzelner WAGO-Busklemmen erhalten Sie in der Systembeschreibung des WAGO-I/O-SYSTEMs 750, den dazugehörigen Handbüchern und Datenblättern unter www.wago.com.

8 In Betrieb nehmen

8.1 Einschalten des I/O-IPC

Überprüfen Sie vor Einschalten des I/O-IPC, dass Sie

- den I/O-IPC ordnungsgemäß montiert haben (siehe Kapitel „Montage und Demontage des I/O-IPC“),
- alle benötigten Datenleitungen (siehe Kapitel „Beschreibung der I/O-IPC-Schnittstellen“) an die entsprechenden Schnittstellen angeschlossen und mit den an den Steckverbindern vorhandenen Arretierungsschrauben befestigt haben,
- die Elektronik- und Feldversorgung angeschlossen haben (siehe Kapitel „Versorgungsspannung anschließen“),
- die Endklemme (750-600) befestigt haben (siehe Kapitel „Anstecken der Busklemmen an den I/O-IPC“),
- einen angemessenen Potenzialausgleich an Ihrer Maschine/Anlage durchgeführt haben (siehe Systembeschreibung 750-xxx) und
- die Schirmung ordnungsgemäß durchgeführt haben (siehe Systembeschreibung 750-xxx).

WARNUNG



Warnung vor Personenschäden!

Bei Einsatz der I/O-IPC-Varianten 758-874/xxx, 758-875/xxx und 758-876/xxx in explosionsgefährdeten Bereichen beachten Sie unbedingt das Kapitel „Einsatz in explosionsgefährdeten Bereichen“.

Zum Einschalten des I/O-IPC und der daran angeschlossenen Busklemmen schalten Sie an Ihrem Netzteil die Versorgungsspannung ein. Nach der Initialisierungsphase startet das Betriebssystem Linux und anschließend das Programmiersystem CODESYS 2.3. Nach einem fehlerfreien Systemstart blinkt/leuchtet die I/O-LED des I/O-IPC grün.

Wenn Sie Ihre bestehende Firmware-Version aktualisieren, kann dies – abhängig von der Version – einige Minuten in Anspruch nehmen. Bitte warten Sie solange, bis das Betriebssystem erneut gestartet ist.

Hinweis



Keine Busklemmen entfernen

Während des Betriebs dürfen Sie keine Busklemmen entfernen oder hinzufügen, da dies ansonsten eine Störung des I/O-IPC und/oder der angeschlossenen Busklemmen zur Folge hat.

8.2 Ermitteln der IP-Adresse des Host-PC

Damit der Host-PC (z. B. Notebook) mit dem I/O-IPC über das ETHERNET-Netzwerk kommunizieren kann, müssen sich beide im gleichen Subnetz befinden.

Zum Ermitteln der IP-Adresse des Host-PC (mit Betriebssystem MS-Windows) mittels der MS-DOS-Eingabeaufforderung gehen Sie folgendermaßen vor:

1. Klicken Sie auf „Start“ und wählen „Ausführen“.
2. Geben Sie den Befehl `cmd` ein und drücken die Entertaste.
Es öffnet sich die Eingabeaufforderung.
3. Geben Sie den Befehl `ipconfig` ein und drücken die Entertaste.
4. Es erscheinen die IP-Adresse, Subnetzmaske und das Standard-Gateway mit den dazugehörigen Parametern.

8.3 Einstellen einer IP-Adresse

Im Auslieferungszustand des I/O-IPC sind für die ETHERNET-Schnittstellen X8 und X9 folgende IP-Adressierungen vergeben:

Tabelle 27: Voreingestellte IP-Adressierungen der Ethernet-Schnittstellen

Ethernet-Schnittstelle	Voreinstellung
X8	Dynamische Vergabe der IP-Adresse mittels Bootstrap-Protokoll (BootP)
X9	Feste IP-Adresse 192.168.2.17 .

Damit ein PC und der I/O-IPC miteinander kommunizieren können, passen Sie mit dem Web-based Management oder mit dem „IPC Configuration Tool“ die IP-Adressierung an Ihre Systemstruktur an (siehe Kapitel „Konfiguration“).

Beispiel zum Einbinden des I/O-IPC (192.168.2.17) in ein bestehendes Netzwerk:

Wenn die IP-Adresse Ihres Host-PC 192.168.1.2 lautet, dann muss sich der I/O-IPC im selben Subnetz befinden. Das heißt, bei der Netzmaske **255.255.255.0** müssen die ersten drei Stellen des I/O-IPC mit denen Ihres PC übereinstimmen. Daraus ergibt sich für den I/O-IPC folgender Adressraum:

Tabelle 28: Netzmaske 255.255.255.0

Host-PC	Subnetzadressraum für den I/O-IPC
192.168.1. 2	192.168.1.2 ... 192.168.1.254

Hinweis



IP-Adressen

Sie dürfen nicht IP-Adressen für beide ETHERNET-Schnittstellen im selben Subnetz vergeben. Dieses darf auch nicht vom BootP- oder DHCP-Server geschehen.

8.3.1 Zuweisen einer IP-Adresse mittels BootP

Der I/O-IPC kann seine IP-Adresse dynamisch (DHCP/BootP) von einem Server beziehen oder mit einer statischen IP-Adresse versehen sein. Im Gegenteil zu festen IP-Adressen werden dynamisch zugewiesene nicht remanent gespeichert. Daher ist bei jedem Neustart des I/O-IPC die Anwesenheit eines BootP- oder DHCP-Servers erforderlich.

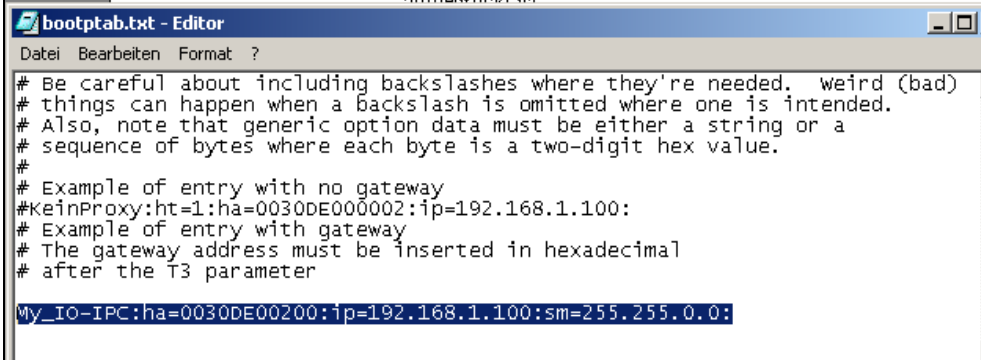
Das Zuweisen der IP-Adresse mittels BootP ist hier exemplarisch an dem WAGO-BootP-Server erläutert.

Voraussetzung:

Der WAGO-BootP-Server ist auf Ihrem PC installiert. Dieser ist unter www.wago.com erhältlich.

WAGO-BootP-Server konfigurieren

1. Notieren Sie sich die MAC-Adressen der ETHERNET-Schnittstellen des I/O-IPC. Diese finden Sie auf dem seitlichen Etikett des I/O-IPC. Siehe dazu Kapitel „Seitliche Beschriftung“.
2. Starten Sie Ihren PC.
3. Klicken Sie auf die „Start“-Schaltfläche und starten Sie den WAGO-BootP-Server unter Programme > **WAGO Software** > **WAGO BOOTP Server**.
4. Öffnen Sie Konfigurationsdatei, indem Sie im BootP-Server auf die Schaltfläche **[Edit BootPtab]** klicken. In der Konfigurationsdatei ordnen Sie die MAC-Adressen je einer IP-Adresse aus demselben Netzwerk zu.
5. Klicken Sie dazu in die markierte Zeile der Konfigurationsdatei:



```

# Be careful about including backslashes where they're needed. weird (bad)
# things can happen when a backslash is omitted where one is intended.
# Also, note that generic option data must be either a string or a
# sequence of bytes where each byte is a two-digit hex value.
#
# Example of entry with no gateway
#keinProxy:ht=1:ha=0030DE000002:ip=192.168.1.100:
# Example of entry with gateway
# The gateway address must be inserted in hexadecimal
# after the T3 parameter
My_IO-IPC:ha=0030DE00200:ip=192.168.1.100:sm=255.255.0.0:

```

Abbildung 24: Konfigurationszeile in der Konfigurationsdatei

6. Ersetzen Sie die aus zwölf Zeichen bestehende MAC-Adresse der ersten ETHERNET-Schnittstelle hinter **:ha=** mit der, die auf dem seitlichen Etikett des I/O-IPC aufgedruckt ist.
7. Geben Sie eine IP-Adresse nach **ip=** ein. In diesem Beispiel ist dies 192.168.1.100.
8. Zur Adressierung der zweiten ETHERNET-Schnittstelle fügen Sie eine weitere Zeile mit der entsprechenden Zuordnung in der Datei bootptab.txt ein. Wiederholen Sie dazu die Handlungsschritte 5 bis 7.
9. Speichern Sie die neuen Einstellungen in der Datei bootptab.txt. Klicken Sie dazu in das Menü „Datei“ und wählen Sie „Sichern“.
10. Schließen Sie den Editor.

Tabelle 29: Erläuterungen der Konfigurationszeile

Parameter	Beschreibung
Node_1	Name des I/O-IPC mit den Busklemmen. Dieser ist frei wählbar.
ht=1	Hardwaretyp des Netzwerks. Dieser lautet für ETHERNET 1.
ha=0030DE000200	MAC-Adresse einer ETHERNET-Schnittstelle.
ip= 192.168.1.2 ip= 192.168.1.100	IP-Adresse für den I/O-IPC, die sich im selben Netzwerk befindet wie auch der Host-PC.
gw=192.168.1.1	IP-Adresse für das Gateway. Für ein lokales Netzwerk brauchen Sie kein Gateway anzugeben.
Sm=255.255.255.0	Subnetzmaske des Subnetzes, zu dem der I/O-IPC gehören soll.

IP-Adresse mittels des WAGO-BootP-Servers vergeben

1. Zum Starten des BootP-Servers klicken Sie im geöffneten BootP-Dialogfenster auf die Schaltfläche **[Start]**. Diverse Meldungen werden im BootP-Dialogfenster angezeigt. Die Fehlermeldungen weisen darauf hin, dass einige Dienste (z. B. Port 67, Port 68) im Betriebssystem nicht definiert worden sind. Diese Fehlermeldung brauchen Sie nicht zu beachten.

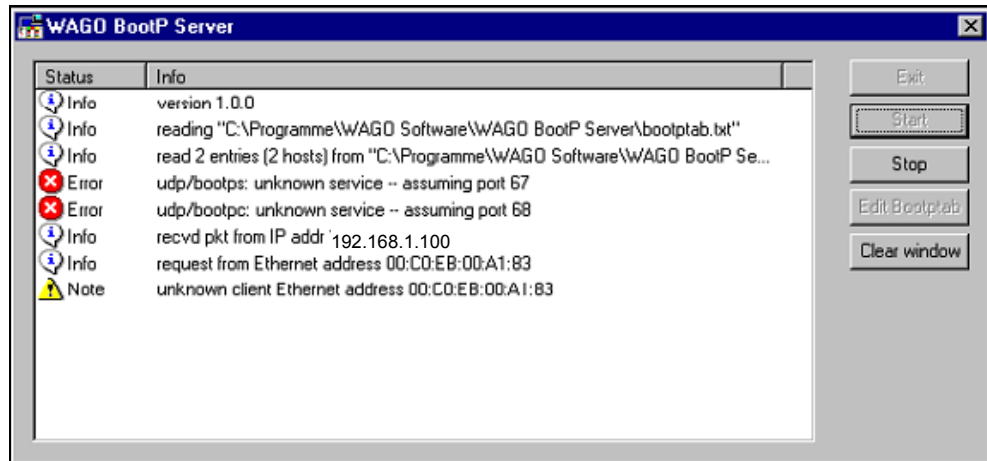


Abbildung 25: Dialogfenster des WAGO-BootP-Servers mit Nachrichten

2. Führen Sie einen Neustart des I/O-IPC durch, indem Sie die Versorgungsspannung des I/O-IPC aus- und anschließend wieder einschalten oder Sie die Reset-Taste drücken. Es erscheint eine Anfrage (request) vom I/O-IPC. Der BootP-Server antwortet, dass die IP-Adresse akzeptiert wurde (keine Fehler). Die IP-Adresse ist nun vorübergehend im I/O-IPC vorhanden, aber nicht remanent gespeichert. Bei einem Neustart versucht der I/O-IPC wieder eine neue IP-Adresse vom BootP-Server zu bekommen.
3. Klicken Sie auf die Schaltfläche **[Stop]** und danach auf die Schaltfläche **[Exit]**, um den BootP-Server zu schließen.
4. Um die IP-Adresse dauerhaft im I/O-IPC zu speichern, wählen Sie im Web-based Management auf der Seite „TCP/IP“ die Option „Static IP“ aus.

8.3.2 Ändern einer IP-Adresse mittels „IPC Configuration Tool“

Über das auf der Linux-Konsole erreichbare IPC-Configuration-Tool können Sie u. a. den ETHERNET-Schnittstellen X8 und X9 eine neue IP-Adresse zuweisen. Weitere Informationen zum IPC-Configuration-Tool erhalten Sie im Kapitel „Konfiguration“.

Vorbereitung:

Sie haben einen Monitor an der DVI-I-Schnittstelle und eine Tastatur an der USB-Schnittstelle des I/O-IPC angeschlossen. Siehe dazu Kapitel „Zugriff über Tastatur und Monitor (DVI- und USB-Schnittstelle)“.

1. Öffnen Sie über die Tastenkombination **[Alt] + [F3]** die dritte Konsole des I/O-IPC, auf der sich das IPC-Configuration-Tool befindet (Abbildungen können von der tatsächlichen abweichen).

```
=====
WAGO IPC Configuration Tool
=====
Main Menu
-----
0. QUIT
1. Information
2. TCP/IP
3. Ethernet
4. NTP
5. Clock
6. WBM Users
7. HMI Settings
8. Administration
9. Downloads
10. Port
11. Modbus
12. Webserver
-----
Select an entry or q to quit
=====
```

Abbildung 26: Startbild des WAGO-IPC-Configuration-Tools

2. Wählen Sie über die Tastatur (Pfeiltasten oder Nummernblock) den Eintrag **TCP/IP** aus und drücken Sie die **[Enter]**-Taste.

```
=====
WAGO IPC-Configuration-Tool
=====
Main Menu
-----
0. QUIT
1. Information
2. TCP/IP
3. NTP
4. Clock
5. WBM Users
6. Administration
7. Port
8. Modbus
-----
Select an entry or q to quit
=====
```

Abbildung 27: TCP/IP

3. Zum Ändern der Ethernet-Schnittstelle X8 wählen Sie **TCP/IP Configuration eth0** oder **TCP/IP Configuration eth1** für die Ethernet-Schnittstelle X9. Drücken Sie anschließend die **[Enter]**-Taste.

In diesem Beispiel wird die Ethernet-Schnittstelle X8 zur Änderung der voreingestellten IP-Adresse ausgewählt:

```
=====
WAGO IPC-Configuration-Tool
=====
TCP/IP
-----
0. Back to Main Menu
1. Hostname
2. Default Gateway
3. DNS Server
4. TCP/IP Configuration eth0 (X8)
5. TCP/IP Configuration eth1 (X9)
-----
Select an entry or q to quit
=====
```

Abbildung 28: TCP/IP-Configuration eth0 (X8)

4. Wählen Sie **IP-Address** aus und drücken Sie die **[Enter]**-Taste.

```
=====
WAGO IPC-Configuration-Tool
=====
TCP/IP Configuration eth0 (X8)
-----
0. Back to TCP/IP Menu
1. State.....enabled
2. Type of IP-Address-Configuration...static
3. IP-Address.....192.168.1.17
4. Subnet Mask.....255.255.255.0
-----
Select an entry or q to quit
=====
```

Abbildung 29: IP-Address

5. Geben Sie die neue IP-Adresse für die ausgewählte Ethernet-Schnittstelle ein und bestätigen Sie diese mittels **[OK]**. Wollen Sie ohne eine Änderung ins Hauptmenü zurückkehren, wählen Sie **[Abort]**.

```
=====
WAGO IPC-Configuration-Tool
=====
Change IP-Address eth0 (X8)
-----

Enter new Address:
+-----+
|192.168.1.17 |
+-----+

< OK >   <Abort>

-----
OK: confirm value, Abort: quit without changes
=====
```

Abbildung 30: Enter new address

Hinweis



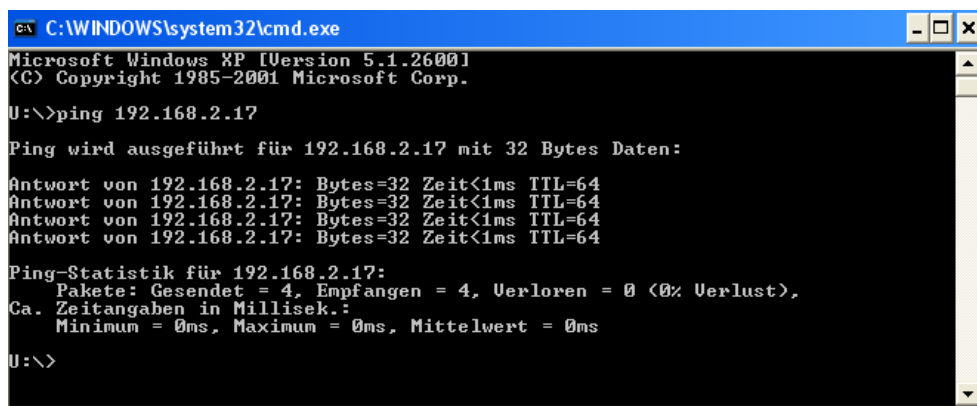
Vergabe der IP-Adressen

Sie dürfen nicht IP-Adressen für beide ETHERNET-Schnittstellen im selben Subnetz vergeben. Das darf auch nicht vom BootP- oder DHCP-Server geschehen.

8.4 Test der Netzwerkverbindung

Um zu überprüfen, ob Sie den I/O-IPC unter der von Ihnen vergebenen IP-Adresse im Netzwerk erreichen, führen Sie den Netzwerkdienst „ping“ durch. Öffnen Sie dazu unter MS-Windows die Eingabeaufforderung, indem Sie auf die „Start“-Schaltfläche klicken und Sie **Programme > Ausführen** wählen. Geben Sie im „Ausführen“-Dialog `cmd` ein und klicken Sie auf **[OK]**.

1. Geben Sie im DOS-Fenster den Befehl `ping` und die IP-Adresse des I/O-IPC ein: Beispiel: `ping 192.168.2.17`.
2. Drücken Sie die Entertaste. Ihr PC sendet eine Anfrage, die vom I/O-IPC beantwortet wird. Die Antwort erscheint im DOS-Fenster. Wenn die Fehlermeldung „Timeout“ erscheint, hat der I/O-IPC sich nicht ordnungsgemäß gemeldet. Überprüfen Sie bitte Ihre Netzwerkeinstellung.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

U:\>ping 192.168.2.17

Ping wird ausgeführt für 192.168.2.17 mit 32 Bytes Daten:

Antwort von 192.168.2.17: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.2.17: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.2.17: Bytes=32 Zeit<1ms TTL=64
Antwort von 192.168.2.17: Bytes=32 Zeit<1ms TTL=64

Ping-Statistik für 192.168.2.17:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0 (0% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 0ms, Mittelwert = 0ms

U:\>
```

Abbildung 31: Beispiel eines Funktionstests

3. Haben Sie den Test erfolgreich durchgeführt, dann schließen Sie das DOS-Fenster.

8.5 Ausschalten/Neustart

VORSICHT Heiße Oberfläche!



Während des Betriebs können hohe Oberflächentemperaturen am Gehäuse des I/O-IPC auftreten. War der I/O-IPC in Betrieb, lassen Sie ihn abkühlen, bevor Sie diesen berühren.

Um den I/O-IPC auszuschalten, gehen Sie folgendermaßen vor:

1. Beenden Sie die laufende Software und das Betriebssystem.
2. Schalten Sie den I/O-IPC aus, indem Sie die Versorgungsspannung abschalten oder die Buchse der Versorgungsspannung vom Anschluss X4 (10) abziehen.

Um einen Neustart des I/O-IPC durchzuführen, drücken Sie den Reset-Taster (45) oder schalten den I/O-IPC aus und anschließend wieder ein.

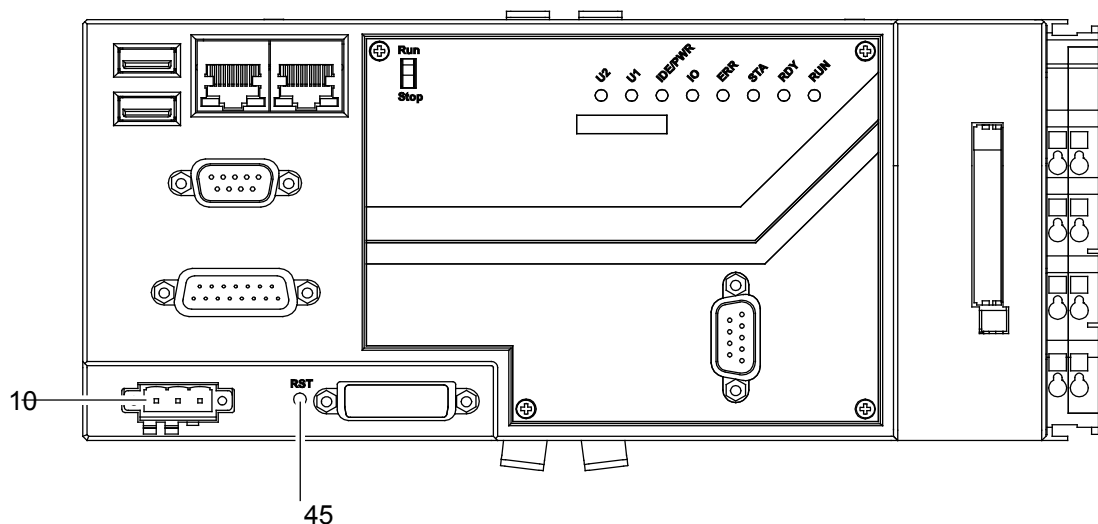


Abbildung 32: Ausschalten/Neustart des I/O-IPC

9 Konfigurieren

Zur Konfiguration des I/O-IPC stehen Ihnen folgende Wege zur Verfügung:

- Zugriff über den PC mittels Internet-Browser auf das Web-based Management (Kapitel „Konfiguration mittels Web-based Management (WBM)“)
- Zugriff über den PC mittels eines Terminalprogramms (über ETHERNET und/oder RS-232-Schnittstelle) auf das „IPC Configuration Tool“ (Kapitel „Konfiguration mit einem Terminalprogramm“)
- Zugriff über den I/O-IPC mittels Touchscreen/Monitor und USB-Tastatur auf das „IPC Configuration Tool“ (Kapitel „Konfiguration mit Touchscreen/Monitor und USB-Tastatur“)
- Zugriff über das SPS-Programm CODESYS mittels der WagoConfigToolLIB.lib (Kapitel „Aufrufe zur Bibliothek „WagoConfigToolLIB.lib““)

Das „IPC Configuration Tool“ stellt dieselben Parameter zur Konfiguration des I/O-IPC zur Verfügung wie das WBM. Die Erläuterungen zu den Parametern entnehmen Sie bitte ab Kapitel „Seite Information“.

9.1 Konfiguration mittels Web-based Management (WBM)

Die implementierten HTML-Seiten (im Folgenden kurz: Seiten) des Web-based Managements dienen zur Konfiguration des I/O-IPC. Für den Zugriff auf das WBM über einen Internet-Browser gehen Sie folgendermaßen vor:

1. Verbinden Sie den I/O-IPC über die ETHERNET-Schnittstelle **X9** mit dem ETHERNET -Netzwerk.
2. Um auf die Seiten zuzugreifen, geben Sie in die Adresszeile Ihres Internet-Browsers die voreingestellte IP-Adresse **192.168.2.17** ein. Beachten Sie, dass sich PC und I/O-IPC im selben Subnetz befinden müssen (siehe dazu Kapitel „Einstellen einer IP-Adresse“).

Wenn Sie einen BootP-Server auf Ihrem PC installiert haben und über BootP auf das WBM zugreifen möchten, nutzen Sie die andere Schnittstelle. Detaillierte Informationen dazu erhalten Sie im Kapitel „Zuweisen einer IP-Adresse mittels BootP“.

Hinweis



Startseite der I/O-IPC

Zeigt der I/O-IPC nicht die Startseite an, vergewissern Sie sich, dass die Einstellungen Ihres Internet-Browsers das Umgehen des Proxyservers für lokale Adressen gestattet. Ferner kontrollieren Sie, ob sich Ihr PC im gleichen Subnetz befindet wie der I/O-IPC.

Einige Seiten des WBM sind passwortgeschützt. Wählen Sie erstmalig einen Eintrag aus der Navigationsleiste, erscheint die Passwortabfrage:

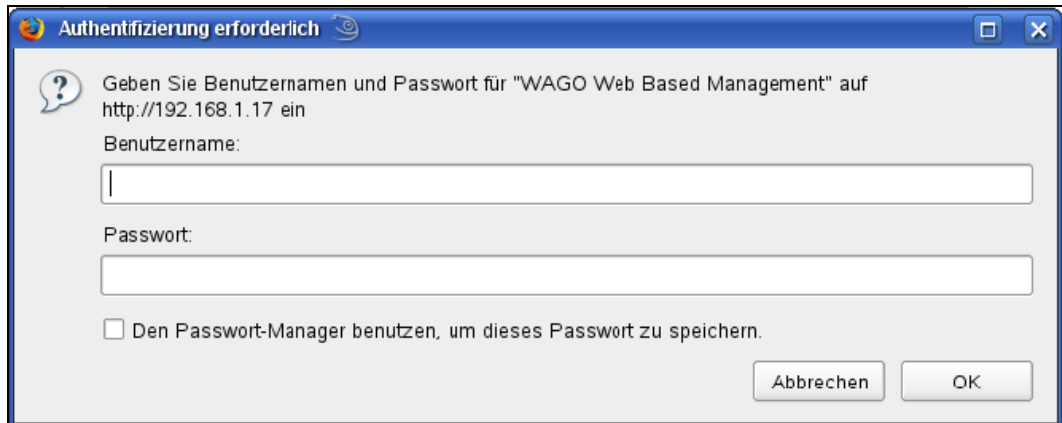


Abbildung 33: Authentifizierung eingeben

9.1.1 Benutzerverwaltung des WBM

Um Einstellungen nur durch einen ausgewählten Personenkreis zu erlauben, begrenzen Sie über die Benutzerverwaltung den Zugriff auf die Funktionen des WBM.

ACHTUNG **Passwörter**



Ändern Sie die Passwörter entsprechend Ihren Vorstellungen ab, da die Standardpasswörter in dieser Betriebsanleitung dokumentiert sind. Somit liefern diese keinen hinreichenden Schutz. Siehe dazu Kapitel „Seite „Users““.

Tabelle 30: Benutzereinstellungen im Auslieferungszustand

Benutzer	Passwort
user	user
admin	wago

Für die Seiten des WBM sieht der Zugriff folgendermaßen aus:

Tabelle 31: Zugriffsrechte für die WBM-Seiten

Internetseite des WBM	Benutzer
Information	-
CODESYS	
TCP/IP	user oder admin
ETHERNET	user oder admin
NTP	user oder admin
Clock	user oder admin
Users	admin
HMI Settings	
Administration	
Package Server	
Mass Storage	
Downloads	
Port	user oder admin
Modbus	user oder admin
SNMP	user oder admin
I/O Configuration	-
WebVisu	

9.1.2 Seite „Information“

Nach Eingabe der IP-Adresse erscheint die Startseite „Information“ des Web-based Managements. Die Seite liefert Informationen zum I/O-IPC und zum ETHERNET-Netzwerk.

Navigation

- Information
- CoDeSys
- TCP/IP
- Ethernet
- NTP
- Clock
- Users
- HMI Settings
- Administration
- Package Server
- Mass Storage
- Downloads
- Port
- Modbus
- SNMP
- I/O-Configuration
- WebVisu

Status Information

Coupler Details

Order Number	0758-0874-0000-0111
Processor Type	Intel(R) Celeron(R) M processor 600MHz
Fieldbus Type	Profibus-Master
License Information	Codesys-Runtime-License
Firmware Revision	01.02.08(00)
KBus FW Revision	01.03.11(00)
CoDeSys Webserver Version	1.1.9.10

Network Details Eth0 (X8)

State	disabled
Mac Address	00:80:82:62:06:F8
IP Address	
Subnet Mask	

Network Details Eth1 (X9)

State	enabled
Mac Address	00:80:82:62:06:F9
IP Address	192.168.1.17
Subnet Mask	255.255.255.0

WAGO Kontakttechnik GmbH & Co KG • Hansastrasse 27 • D-32429 Minden • www.wago.com

Abbildung 34: Seite „Information“ (Beispiel)

Die nachfolgende Tabelle erläutert die auf der Seite aufgeführten Parameter:

Tabelle 32: Beschreibung der Parameter der Seite „Information“

Coupler Details	
Order Number	Bestellnummer des I/O-IPC.
Processor Type	CPU-Typ des I/O-IPC
Fieldbus Type	Feldbustyp des I/O-IPC
Firmware Revision	Firmwarestand
Licence Information	Anzeige, dass das Laufzeitsystem CODESYS vorhanden ist.
Kbus FW Revision	Firmwarestand des Klemmenbus-Controllers
Network Details Eth0 (X8)/Eth1 (X9)	
State	Status der ETHERNET-Schnittstelle (aktiviert/deaktiviert).
Mac Address	MAC-Adresse, die zur Identifikation und Adressierung des I/O-IPC dient.
IP Address	Aktuelle IP-Adresse des I/O-IPC.
Subnet Mask	Aktuelle Subnetzmaske des I/O-IPC.

9.1.3 Seite „CODESYS“

Auf der Seite „CODESYS“ erhalten alle Informationen zu dem in CODESYS erstellten SPS-Programm.

Tabelle 33: Beschreibung der Parameter der Seite „CODESYS“

Project Details	
Date	Anzeige von Projektinformationen, die der Programmierer im SPS-Programm eingetragen hat (in CODESYS unter Projekt > Projektinformation...). Die Informationen erscheinen nur bei einem ausgeführten SPS-Programm. Unter „Description“ werden bis zu 1024 Zeichen lange Beschreibungstexte dargestellt.
Title	
Version	
Author	
Description	
CODESYS State	
State	STOP: SPS-Programm wird nicht ausgeführt RUN: SPS-Programm wird ausgeführt
Tasks	
Bei Ausführung eines SPS-Programms erscheinen für jeden Task folgende Informationen:	
<ul style="list-style-type: none"> - Number of tasks - Cycle count - Cycletime (msec) - Cycletime min (msec) - Cycletime max (msec) - Cycletime avg (msec) - Status - Mode - Priority - Interval (msec) - Event - Function pointer - Function index 	

9.1.4 Seite „TCP/IP“

Auf der Seite „TCP/IP“ können Sie die Parameter für die ETHERNET-Konfiguration verändern. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche [SUBMIT].

Tabelle 34: Beschreibung der Parameter der Seite „TCP/IP“

Common Configuration Data	
Hostname	Wenn Sie die dynamische Zuweisung einer IP-Adresse über DHCP ausgewählt haben, tragen Sie hier den Hostnamen ihres PC ein, damit Sie Ihren PC im ETHERNET-Netzwerk erreichen.
Default Gateway	
None	Hier wählen Sie die Schnittstelle aus, die sie als Standard-Gateway nutzen möchten. Der I/O-IPC verwendet das Standard-Gateway, wenn die Zieladresse außerhalb des eigenen Netzwerks liegt.
X8	
X9	
Value	Hier stellen Sie die Adresse des Standard-Gateways ein.
DNS-Server	
Domain Name	Hier stellen Sie den Domainnamen ein.
DNS-Server 1, 2, ...	Hier stellen Sie die Adresse des DNS-Servers ein. [CHANGE]: Übernehmen der DNS-Adresse. [DELETE]: Löschen der DNS-Adresse (Zeile wird entfernt).
Add DNS-Server	Hier fügen Sie weitere DNS-Adressen hinzu.
TCP/IP Configuration Eth0/Eth1	
State	
Enabled	Hier aktivieren Sie die entsprechende ETHERNET-Schnittstelle.
Disabled	Hier deaktivieren Sie die entsprechende ETHERNET-Schnittstelle.
Type of IP Address Configuration	
Static IP	Hier wählen Sie aus, ob Sie eine statische oder dynamische IP-Adressierung verwenden möchten. Static IP: Statische IP-Adressierung DHCP und BootP: Dynamische IP-Adressierung
DHCP	
BootP	
Configuration Data	
IP Address	Hier geben Sie eine statische IP-Adresse ein. Diese ist aktiv, wenn im Feld „Type of Address Configuration“ „Static IP“ aktiviert ist.
Subnet Mask	Hier geben Sie Subnetzmaske ein. Diese ist aktiv, wenn im Feld „Type of Address Configuration“ „Static IP“ aktiviert ist.

9.1.5 Seite „ETHERNET“

Auf der Seite „ETHERNET“ konfigurieren Sie die Übertragungsgeschwindigkeit und das Kommunikationsverfahren der ETHERNET-Schnittstelle des I/O-IPC. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche **[SUBMIT]**.

Tabelle 35: Beschreibung der Parameter der Seite „ETHERNET“

Transmission Mode Eth0 und Eth1	
Autonegotiation on	Bei aktivierter Funktion stellen zwei miteinander verbundenen ETHERNET-Schnittstellen (z. B. von Computer und I/O-IPC) selbständig untereinander die maximal mögliche Übertragungsgeschwindigkeit und das Duplex-Verfahren her.
Settings	Hier wählen Sie feste Übertragungsgeschwindigkeit und Kommunikationsverfahren aus: 10 Bit oder 100 Mbit Halbduplex (Halbduplex: Informationen können nur gesendet oder empfangen werden) 10 MBit oder 100 Mbit Vollduplex (Vollduplex: Informationen können gleichzeitig gesendet und empfangen werden)

9.1.6 Seite „NTP“

Auf der Seite „NTP“ konfigurieren Sie die NTP-Parameter zur Einstellung der Uhrzeit. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche **[SUBMIT]**.

Tabelle 36: Beschreibung der Parameter der Seite „NTP“

Configuration Data	
Enabled	Hier aktivieren/deaktivieren Sie die Aktualisierung der Uhrzeit.
Disabled	
Port	Hier geben Sie die Port-Nummer für den NTP-Zugriff ein (Grundeinstellung: 123)
Time Server	Hier geben Sie die IP-Adresse des Time-Servers ein.
Update Time (sec)	Hier legen Sie den Abfragezyklus des Time-Servers fest.

9.1.7 Seite „Clock“

Auf der Seite „Clock“ konfigurieren Sie die Echtzeituhr. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche [CHANGE].

Hinweis



Eingabe der Zeitzone

Ihre Eingaben zur Zeitzone sind erst nach einem Neustart/Reset des I/O-IPC aktiv.

Tabelle 37: Beschreibung der Parameter der Seite „Clock“

Time and Date	
Date on device local	Hier stellen Sie das Datum ein.
Time on device local	Hier stellen Sie die lokale Uhrzeit ein.
Time on device UTC	Hier stellen Sie die GMT-Zeit ein.
12-Hour-Format	Umschaltung zwischen 12h- und 24h-Darstellung der Uhrzeit.
Timezone	
Select	<p>Hier wählen Sie die für Ihr Land zutreffende Zeitzone aus. Grundeinstellung:</p> <p>AST/ADT: „Atlantic Standard Time“, Halifax EST/EDT: „Eastern Standard Time“, New York, Toronto CST/CDT: „Central Standard Time“, Chicago, Winnipeg MST/MDT: „Mountain Standard Time“, Denver, Edmonton PST/PDT: „Pacific Standard Time“, Los Angeles, Whitehouse GMT/BST: „Greenwich Main Time“, GB, P, IRL, IS, ... CET/CEST: „Central European Time“, B, DK, D, F, I, CRO, NL, ... EET/EEST: „East European Time“, BUL, FI, GR, TR, ... CST: „China Standard Time“ JST: „Japan/Korea Standard Time“</p>
Edit TZ-String	<p>Für nicht über den Parameter „Select“ auswählbare Zeitzone geben Sie hier die für Sie zutreffende Zeitzone ein. Eine Übersicht aller Zeitzone erhalten Sie unter http://home.tiscali.nl/~t876506/TZworld.html Informationen dazu, wie Sie den TZ-String in Linux editieren, erhalten Sie unter http://www.minix-vmd.org/pub/Minix-vmd/1.7.0/wwwman/man5/TZ.5.html</p>

9.1.8 Seite „Users“

Auf der Seite „Users“ ändern Sie die Passwörter der Benutzer **admin** und **user**. Sie müssen dazu als Benutzer **admin** angemeldet sein. Eine Übersicht der Passwörter finden Sie im Kapitel „Benutzerverwaltung des WBM“. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche **[SUBMIT]**.

ACHTUNG Passwörter



Ändern Sie die Passwörter entsprechend Ihren Vorstellungen ab, da die Standardpasswörter in dieser Betriebsanleitung dokumentiert sind. Somit liefern diese keinen hinreichenden Schutz.

Tabelle 38: Beschreibung der Parameter der Seite „Users“

Configuration Data	
Select User	Hier wählen Sie den Benutzer (user oder admin) aus, für den Sie ein neues Passwort vergeben wollen.
New Password	Hier geben Sie das neue Passwort für den unter „Select User“ ausgewählten Benutzer ein.
Confirm Password	Hier geben Sie zur Kontrolle das neue Passwort erneut ein.

9.1.9 Seite „HMI Settings“




Auf der Seite „HMI Settings“ ändern Sie die grafische Auflösung für die DVI-I-Schnittstelle, konfigurieren den Touchscreen oder Monitor und wählen zwischen englisch- oder deutschsprachiger Tastaturbelegung.

Zum Speichern aller auf der Seite durchgeführten Einstellungen, klicken Sie auf die Schaltfläche **[SUBMIT]**. Die Einstellungen sind erst nach einem Neustart oder Reset des I/O-IPC aktiv.

Tabelle 39: Beschreibung der Parameter Screensaver und Cleanmode der Seite „HMI Settings“

Screensaver	
Display	<p>Zeigt an, ob das Display gerade ein- oder ausgeschaltet ist. aus: rotes Kreuz und Text „off“ an: grüner Haken und Text „on“</p> <p>Das Display kann entweder durch den Screensaver (nach entsprechender Wartezeit), oder explizit durch den Benutzer ausgeschaltet werden.</p> <p>Auf der Webseite wird der aktuelle Wert zyklisch alle zwei Sekunden abgefragt und angezeigt (im Configtool wird nicht zyklisch, sondern nur einmalig abgefragt).</p> <p>Mit der Schaltfläche [Switch on] oder [Switch off] können Sie das Display ein- oder ausschalten. Diese Änderung wirkt sich jeweils sofort aus.</p>
Screensaver	<p>Zeigt den Status der Screensaver-Funktionalität an. enabled: grüner Haken und Text „enabled“ disabled: rotes Kreuz und Text „disabled“</p> <p>Bei eingeschaltetem Screensaver (enabled), wird das Display des Touchscreens nach der konfigurierten Wartezeit ausgeschaltet und jeweils nach Touchscreen- oder Tastatur-Eingabe wieder angeschaltet. Bei ausgeschaltetem Screensaver (disabled), kann das Display nur explizit durch den Benutzer ein- oder ausgeschaltet werden (siehe Punkt „Display“).</p> <p>Mit der Schaltfläche [Enable] oder [Disable] können Sie die Screensaver-Funktionalität sofort aktivieren oder deaktivieren.</p> <p>Dies wirkt sich nicht auf den aktuellen Zustand des Displays aus – auch wenn der Screensaver aktiviert wird, ändert sich der Zustand des Displays erst nach Ablauf der Wartezeit, bzw. Druck auf Touchscreen/Tastatur.</p>
Wait time (sec)	<p>Der Parameter „Wait time“ wird nur ausgewertet, wenn der Parameter „Screensaver“ den Status „enabled“ hat. Ist dies der Fall, so gibt der Wert die Zeit in Sekunden an, nach deren Ablauf der Bildschirmschoner aktiviert wird.</p> <p>Sie können die Wait time ändern, indem Sie den Wert im Input-Feld ändern und auf die Schaltfläche [Change] klicken. Die Änderungen wirken sich sofort auf die nächste Aktivierung des Screensavers aus.</p>


Tabelle 39: Beschreibung der Parameter Screensaver und Cleanmode der Seite „HMI Settings“

Cleanmode					
Actual State	<p>Zeigt an, ob der Cleanmode aktuell aktiv oder nicht aktiv ist aktiv: grüner Haken und Text „on“ nicht aktiv: rotes Kreuz und Text „off“</p> <p>Bei aktiviertem Cleanmode werden Berührungen am Touchscreen ignoriert. Der Cleanmode kann durch den Benutzer wieder ausgeschaltet werden; spätestens wird er nach der eingestellten Timeout-Zeit automatisch ausgeschaltet (und der Touchscreen damit wieder benutzbar).</p> <p>Auf der Webseite wird der aktuelle Wert zyklisch alle zwei Sekunden abgefragt und angezeigt (im Configtool wird nicht zyklisch, sondern nur einmalig abgefragt).</p> <p>Mit der Schaltfläche [Switch on] oder [Switch off] können Sie den Cleanmode aktivieren oder der aktive Cleanmode unterbrochen werden. Diese Eingaben wirken sich sofort aus.</p>				
Timeout (sec)	<p>Hier wird die Timeout-Zeit des Cleanmodes angezeigt, d.h. die Zeit, in der bei aktiviertem Cleanmode Berührungen am Touchscreen ignoriert werden.</p> <p>Sie können die Timeout-Zeit ändern, indem Sie den Wert im Input-Feld ändern und auf die Schaltfläche [Change] klicken. Die Änderungen wirken sich sofort auf die nächste Aktivierung des Cleanmodes aus (aber nicht bei einem bereits aktivierten Cleanmode).</p>				
VGA-Configuration					
<p>Hier wählen Sie für den verwendeten Monitor die grafische Auflösung für die DVI-I-Schnittstelle aus. Um Ihre Eingaben zu übernehmen, klicken Sie auf die Schaltfläche [SUBMIT]. Wird auf dem Monitor/Touchscreen nur ein dunkles Bild angezeigt, stimmt die gewählte Auflösung nicht mit der des Monitors/Touchscreen überein. Wählen Sie eine andere Auflösung aus.</p>					
Show mouse pointer	Hierüber blenden Sie den Mauszeiger auf dem Monitor aus oder ein.				
Touchscreen Configuration					
Device-Name	<p>Hier wählen Sie einen über USB angeschlossenen Touchscreen (mouse dev) aus. Der I/O-IPC unterstützt keine seriell anzuschließende Touchscreens.</p> <p>Ist ein ausgewählter Touchscreen nicht mehr angeschlossen, erscheint für diesen der Hinweis „(not available)“.</p>				
Driver-Name	Hier wählen Sie einen Gerätetreiber für den zuvor ausgewählten Touchscreen aus.				
Execute calibration of touchscreen at next start	<p>Wenn Sie diese Option aktivieren, erscheint beim nächsten Start des I/O-IPC vor Ausführung des SPS-Programms eine Oberfläche zum Kalibrieren des Touchscreens.</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">Hinweis</td> <td>Touchscreen</td> </tr> <tr> <td style="text-align: center;"></td> <td>Ist kein Touchscreen am I/O-IPC angeschlossen, muss diese Option deaktiviert bleiben, da andernfalls das SPS-Programm nicht startet.</td> </tr> </table>	Hinweis	Touchscreen		Ist kein Touchscreen am I/O-IPC angeschlossen, muss diese Option deaktiviert bleiben, da andernfalls das SPS-Programm nicht startet.
Hinweis	Touchscreen				
	Ist kein Touchscreen am I/O-IPC angeschlossen, muss diese Option deaktiviert bleiben, da andernfalls das SPS-Programm nicht startet.				
Keyboard Layout					
German	Hier wählen Sie zwischen englisch- oder deutschsprachiger Tastaturbelegung der Linux-Konsole aus.				
English					

9.1.10 Seite „Administration“

Über die Seite „Administration“ speichern Sie sämtliche durchgeführten Einstellungen: auf einer CF-Karte, im internen Speicher des I/O-IPC oder auf einem USB-Speicher.

Tabelle 40: Beschreibung der Parameter der Seite „Administration“

Create bootable image from active partition (Internal Flash)	
Select destination	Wählen Sie das Speichermedium aus, worauf Sie das Image kopieren möchten. Entweder von der CF-Karte auf den intern Flash-Speicher oder auf einen USB-Speicher oder umgekehrt. Über die Schaltfläche [Start Copy] starten Sie den Kopiervorgang.
Configuration of Serial Interface	
CODESYS Debug	Hier wählen Sie den Dienst für die serielle Schnittstelle aus. Zum Übernehmen der Einstellung klicken Sie auf die Schaltfläche [SUBMIT] .
IO-Check	
MODBUS RTU	
Linux Console	<p>WARNUNG! Aktivierung der Betriebsart „Control-Mode“ in WAGO-IO-CHECK!</p> <p></p> <p>Bei Verwendung von WAGO-IO-CHECK können Sie in der Betriebsart „Control-Mode“ - unabhängig davon, ob Feldbus oder SPS-Funktionalitäten aktiviert oder deaktiviert sind - Prozessdaten und Parameter überschreiben. Dadurch können Maschinenkomponenten in einen gefährlichen Zustand versetzt sowie Personal und Maschine gefährdet werden.</p> <p>Vor Änderung der Parameter und vor Änderung von Prozessdaten bringen Sie die Maschinenkomponenten in einen definierten und sicheren Zustand und schalten Sie die übergeordnete Steuerung aus. Vergewissern Sie sich vor Inbetriebnahme, dass sich kein Personal im Gefahrenbereich der Maschinenkomponenten aufhält.</p>
Free Port (CODESYS Libs)	Die serielle Schnittstelle steht für eigene Anwendungen zur Verfügung, wie beispielsweise CODESYS-Projekte mit Verwendung der Bibliothek SysLibCom.lib oder SerComm.lib).
File system Check	
Select Device	Hierüber führen Sie eine Prüfung des Dateisystems für eines aus der Liste gewählte Gerät durch. Zum Starten der Prüfung klicken Sie auf die Schaltfläche [Start Check] . Wurde bei der Prüfung ein Problem erkannt, wird eine entsprechende Fehlermeldung oben auf der Seite über „Configuration of Serial Interface“ angezeigt („Error while filecheck. If more ...“).
Start Backup System	
Start Backup System	Hierüber schalten Sie den Bootloader um, damit er beim nächsten Neustart die andere Systempartition startet. Hier befindet sich noch die ältere Version der Firmware, wenn Sie vorher über die Seite „Firmware Update“ die System-Firmware wiederhergestellt haben.
Reboot IPC	
Über die Schaltfläche [Start Reboot] führen Sie einen Neustart des I/O-IPC durch.	


9.1.11 Seite „Package Server“

Über die Seite „Package Server“ kopieren Sie von der aktuellen Partition des I/O-IPC die Firmware als „Backup-Pakete“ auf Geräte, die am I/O-IPC angeschlossen sind. Ein „Update-Paket“ kann alle oder folgende der einzelne Bestandteile enthalten: Das CODESYS-Projekt, I/O-IPC-Einstellungen sowie das Dateisystem.

Tabelle 41: Beschreibung der Parameter der Seite „Package Server“

<p>Backup Hierüber können Sie Backups von der aktiven Partition Ihres IPC auf ein ausgewähltes Ziel veranlassen. In der Tabelle visualisiert ein Pfeil die Richtung des Backups – von der aktiven Partition werden die Pakete auf das ausgewählte Ziel gesichert.</p> <p>Die Erstellung der Backup-Dateien nimmt auf dem IPC einige Zeit in Anspruch. Nach dem Klicken der Schaltfläche [SUBMIT] erscheint eine Fortschrittsanzeige.</p>	
Active Partition	Hier wird Ihnen die aktuell aktive Partition Ihres IPC angezeigt. Die Backups können nur von dieser ausgeführt werden.
Packages	<p>Hier können Sie per Checkbox die Pakete auswählen, für die Sie ein Backup anlegen möchten:</p> <p>All: Auswahl aller „Backup-Pakete“ CODESYS Project: CODESYS-Projekt Settings: alle Einstellungen die u. a. über das WBM vorgenommen werden können, wie z.B. IP-Adressen, ETHERNET-Settings, HMI-Konfiguration, System: die komplette System-Partition</p>
Destination	<p>Hier können Sie das Ziel auswählen, auf das Ihre Backups abgelegt werden sollen. Dafür werden Ihnen alle derzeit angeschlossenen Speichermedien angezeigt, wie z.B. die CF-Karte, der interne Flash-Speicher und diverse USB-Speicher. Das derzeit aktive Medium (von dem aus das Backup erstellt wird) erscheint nicht in dieser Liste.</p> <p>Zusätzlich enthält die Liste den Punkt „Network“. Wenn dieser ausgewählt wird, erfolgt die Ablage der Backup-Dateien auf Ihrem Computer durch einen Download über das Netzwerk: Es kann dabei nur ein Paket in einem Durchgang übertragen werden. Sollte der Menüpunkt „Network“ angewählt werden, sind die Auswahl mehrerer Pakete und die Checkbox „All“ nicht möglich. Nach dem Klicken der Schaltfläche [SUBMIT] und der Erstellung der Backup-Datei wird Ihnen Ihr Browser das Browser-typische Download-Fenster anzeigen, um die Datei in dem Dateisystem Ihres Rechners zu speichern. Entfernen Sie beim Speichern möglichst keine Text-Teile des vorgeschlagenen Dateinamens, sondern fügen etwas hinzu (Nummern o.ä.): Die Packages können im ersten Schritt (nur) über den Dateinamen zugeordnet werden.</p> <p>Das Backup der Pakete „CODESYS“ und „System“ erzeugt komprimierte Speicher-Abbilder kompletter Partitionen. Das Settings-Backup hingegen wird mit Hilfe der Configtools erstellt und in einer Text-Datei abgelegt. Diese können Sie mit einem normalen Text-Editor betrachten und bei Bedarf auch von Hand ändern. Dafür sind allerdings genaue Kenntnisse darüber nötig, in welchem Format die einzelnen Configtools die Parameterwerte ablegen, bzw. benötigen.</p>


Tabelle 41: Beschreibung der Parameter der Seite „Package Server“

<p>Activate auto update feature</p>	<p>Wenn Sie Pakete mit dieser Funktion erstellen, werden die Pakete automatisch beim Starten des Geräts von dem Speichermedium (CF-Karte/USB-Speicher) auf das Gerät kopiert.</p> <hr/> <p>Hinweis  Sicherheitsabfrage Es erfolgt keine Sicherheitsabfrage beim Booten. Alle Daten auf dem Speichermedium (CF-Karte/USB-Speicher) werden automatisch auf das System kopiert: alte Daten/Programme werden überspielt.</p>
<p>Restore</p>	
<p>Beim Restore werden die zuvor abgelegten Backups auf dem aktiven Medium des IPC eingespielt. Von der ausgewählten Quelle werden die Pakete in die aktive Partition eingespielt. Nach dem Restore der „System“- , „Settings“ oder „CODESYS“-Pakete wird der IPC automatisch neu gestartet.</p> <p>Der Restore des Systems bietet gegenüber den anderen Paketen noch die Besonderheit, dass hierfür eine extra Partition zur Verfügung steht. Die Backup-Dateien werden dort hin kopiert, und erst nach dem automatischen Neustart des IPC mit umgeschaltetem Bootloader aktiv. Dadurch steht die alte Version des Systems (solange die Funktion „Restore“ nicht ausgeführt wird) noch zur Verfügung und kann über die Funktion „Start Backup System“ auf der WBM-Seite „Administration“ wieder aktiviert werden.</p>	
<p>Source</p>	<p>Hier werden Ihnen alle als Quelle von Backup-Dateien möglichen Speicher-Medien angezeigt. Die derzeit aktive Partition – also das Ziel des Restores - erscheint nicht in der Liste, sondern wird ganz rechts in der Tabelle unter „active partition“ angezeigt.</p>
<p>Packages</p>	<p>In dieser Liste können Sie über Checkboxes auswählen, welche(s) Paket(e) sie wieder herstellen möchten. Um alle Pakete anzuwählen, ist die Checkbox „All“ vorhanden.</p> <p>„Network“ ist auch beim Restore ein Sonderfall: hier sendet der Browser die Backup-Dateien von Ihrem Computer als Upload zum IPC. Sie müssen selber festlegen, welche Dateien dafür verwendet werden sollen. Wird „Network“ ausgewählt, erscheinen deshalb zusätzliche Eingabefelder für die Dateinamen jeder Backup-Datei. Nach einem Klick auf den Schaltfläche [Browse...] wird das Datei-Auswahl-Menü Ihres Browsers geöffnet und Sie können die gewünschte Datei auswählen. Anschließend wird überprüft, ob der Name der angegebenen Datei zu den Namenskonventionen passt, die der IPC beim Download vorgegeben hat. Ist dies nicht der Fall, wird eine Warnmeldung ausgegeben. Der Upload der Datei ist trotzdem möglich, aber um die Sicherheit an dieser Stelle zu erhöhen ist es – wie beim Punkt Backup erwähnt - sinnvoll, bereits beim Download die eigentlichen Dateinamen nicht zu verändern. Die Warnmeldung erscheint natürlich auch, wenn sie versehentlich eine vollkommen unpassende Datei ausgewählt haben, oder eine Backup-Datei, die nicht zu einem anderen Paket gehört. Bitte überprüfen Sie dies sorgfältig, bevor sie trotz Warnung die „Restore“-Funktion ausführen.</p> <p>Bei allen anderen Auswahlmöglichkeiten im Menü „Source“, wird die Webseite selbständig ermitteln, welche auf dem ausgewählten Source-Medium zur Verfügung stehen. Dies nimmt einige Zeit in Anspruch, weshalb Ihnen währenddessen (in der Box für die Paket-Auswahl), eine entsprechende Meldung angezeigt wird. Dieser Vorgang ist auch beim ersten Aufruf der Webseite nötig. Danach werden die Checkboxes der nicht vorhandenen Pakete ausgegraut.</p>
<p>Active Partition</p>	<p>Hier wird Ihnen das aktuell aktive Medium angezeigt. Dieses ist immer das Ziel des Restore.</p>

9.1.12 Seite „Mass Storage“

Auf der Seite „Mass Storage“ bekommen Sie Informationen über die für den IPC verfügbaren Massenspeicher-Medien und können diesbezüglich unterschiedlich Aktionen ausführen lassen. Für jedes verfügbare Medium wird eine eigene Tabelle angezeigt.

Tabelle 42: Beschreibung der Parameter der Seite „Mass Storage“

Mass Storage	
Speichermedium/aktive Partition	In der Tabellenüberschrift wird das Speichermedium angegeben, also „Internal Flash“, „CF Card“, „USB1“, etc. Falls dieses Medium gerade aktiv ist, wird dahinter der Text „Active Partition“ ausgegeben.
Device	Der Name des Devices im Dateisystem des Betriebssystems.
Volume name	Hier erscheint der Name des Speichermediums.
Bootflag	Eine Grafik und ein entsprechender Text zeigen an, ob der Speicher „Bootable“ oder „Not bootable“ ist. Über die Schaltfläche [SET/RESET BOOTFLAG] rechts in der Tabellenzeile können Sie das Bootflag dementsprechend setzen oder rücksetzen. Der interne Flash-Speicher muss immer bootfähig bleiben; die Schaltfläche wird hier deshalb nicht angezeigt.
FAT format Medium	Über die Schaltfläche [START FORMATTING] starten Sie die Formatierung des Mediums mit (genau) einer Partition im FAT32-Format. <div style="display: flex; align-items: center;"> <div style="text-align: center; margin-right: 10px;">  </div> <div> <p>ACHTUNG Löschen vorhandener Daten Beim Formatieren werden die vorhandenen Daten gelöscht. Die gerade aktive Partition und der interne Flash dürfen nicht formatiert werden. Der Button wird hier deshalb nicht angezeigt.</p> </div> </div>
Volume name	Diesen Namen erhält die neu formatierte Partition Ihres Mediums. Linux benutzt diesen Namen z.B., um die Partition später einzubinden.

Hinweis



Begrenzte Anzahl von USB-Sticks

Es können maximal 8 am I/O-IPC angeschlossene USB-Sticks dargestellt und bearbeitet werden. Alle weiteren USB-Sticks werden als „Unknown Medium“ gelistet/dargestellt und können nicht vom System verwendet werden.

9.1.13 Seite „Downloads“

Auf der Seite können Sie aktuelle I/O-IPC-Firmware, Feldbussoftware, Programmlizenzen und Update-Scripte über die Schaltfläche **[Browse...]** im Dateisystem des PC suchen und über **[Download]** in den I/O-IPC importieren. Über die Schaltfläche **[Activate]** aktivieren Sie die neuen Daten im I/O-IPC.

9.1.14 Seite „Port“

Auf der Seite zur Protokoll-Konfiguration wählen Sie die Protokolle aus, die Sie zur Kommunikation verwenden möchten. Sie haben die Wahl zwischen den folgenden Protokollen:

- Telnet
Bei Verwendung der Linux-Konsole über ETHERNET
- CODESYS-Webserver
Zur Verwendung der CODESYS-Web-Visualisierung
- FTP
Zum Transfer von Dateien
- CODESYS
Für den Zugriff auf CODESYS

Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche **[SUBMIT]**.

9.1.15 Seite „MODBUS“

Auf der Seite ändern Sie die MODBUS-Einstellungen. Wählen Sie, ob Sie MODBUS-UDP und/oder MODBUS-TCP als Protokoll zum Prozessdatenaustausch verwenden möchten. Ferner stellen Sie auf dieser Seite auch das Time-out (MODBUS-TCP) ein. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche **[SUBMIT]**.

Tabelle 43: Beschreibung der Parameter der Seite „MODBUS“

MODBUS UDP	
Enabled/Disabled	Hier aktivieren oder deaktivieren Sie das MODBUS-UDP-Protokoll.
MODBUS TCP	
Enabled/Disabled	Hier aktivieren oder deaktivieren Sie das MODBUS-TCP-Protokoll.
Timeout (msec)	Hier stellen Sie die Zeitspanne der MODBUS-TCP-Verbindung ein, nach der die Verbindung bei einer Unterbrechung der Kommunikation automatisch beendet wird.
MODBUS RTU	
State	Anzeige der aktuellen MODBUS-Verbindung, die auf Seite „Administration“ ausgewählt wurde.
Node-ID	Auswahl einer MODBUS-Node-ID im Bereich 1 – 247.
Timeout (msec)	Hier stellen Sie die Zeitspanne der MODBUS-RTU-Verbindung ein, nach der die Verbindung bei einer Unterbrechung der Kommunikation automatisch beendet wird.
Baudrate	Hier wählen Sie die Übertragungsgeschwindigkeit der seriellen Schnittstelle aus.
Databits	Auswahl der zu übertragenden Databits.
Parity	Aktivieren/deaktivieren der Übertragungsfehlererkennung.
Stop-Bits	Hier wählen Sie die Anzahl der Stoppbits aus.
Flow-Control	Hier stellen Sie die Flusskontrolle für Hardware und Software.

9.1.16 Seite „SNMP“

Auf der Seite zur SNMP-Konfiguration verändern Sie Parameter für das „Simple Network Management Protocol“. Zum Übernehmen Ihrer Eingaben klicken Sie auf die Schaltfläche **[SUBMIT]** bzw. zum Löschen auf **[DELETE]**.

Im Header eines SNMP-Paketes muss jeweils der „Community Name“ übertragen werden. Durch ihn wird eine Art Zugriffsrecht vergeben. Da dieser jedoch im Klartext übertragen wird, gilt SNMP in seinen ersten Versionen (v1 und v2c) als ein äußerst unsicheres Protokoll.

Der IPC unterstützt SNMP in den Versionen v1, v2c und v3. v2c bietet gegenüber v1 ein paar zusätzliche Funktionen und mit v3 wurde die Verschlüsselung und eine verbesserte Authentifizierung eingeführt. Bei SNMP-v3 ist der Nachrichtenaustausch an Benutzer gebunden.

Tabelle 44: Beschreibung der Parameter der Seite „SNMP“

General SNMP Configuration	
Hier werden allgemeine SNMP-Informationsdaten konfiguriert.	
Name of device	Gerätename (sysName)
Description	Gerätebeschreibung (sysDescription)
Physical location	Standort des Gerätes (sysLocation)
Contact	E-Mail-Kontaktadresse (sysContact)
SNMP v1/v2c Manager Configuration	
Protocol enabled	Hier de-/aktivieren Sie das SNMP-Protokoll für v1/v2c.
Local Community Name	Hier geben Sie den Community-Namen für die SNMP-Manager-Konfiguration an. Über den Community-Namen können Beziehungen zwischen SNMP-Manager und -Agenten eingerichtet werden, die jeweils als Community bezeichnet werden und die Identifizierung sowie den Zugriff zwischen den SNMP-Teilnehmern steuern. Der Community-Name darf max. 32 Zeichen lang sein und keine Leerzeichen enthalten. Um das SNMP-Protokoll verwenden zu können, muss immer ein gültiger Community-Name angegeben sein. Standard ist "public".
SNMP v1/v2c Trap Receiver Configuration	
Hier wird Ihnen eine Liste mit den Daten aller konfigurierten „Trap Receiver“ für v1/v2c angezeigt. Die Anzahl der Receiver ist im I/O-IPC prinzipiell nicht begrenzt. Sie haben die Möglichkeit, die einzelnen konfigurierten Receiver über [DELETE] zu löschen. Am Ende der Liste finden Sie ein Formular, mit dem ein kompletter neuer Trap-Receiver angelegt werden kann. Die einzelnen Konfigurationsdaten der bereits bestehenden Receiver können nicht geändert werden.	
IP Address	IP-Adresse des Trap-Empfängers (Management-Station).
Community Name	Hier geben Sie den Community-Namen für die Trap-Receiver-Konfiguration an. Der Community-Name kann durch den Trap-Empfänger ausgewertet werden. Der Community-Name darf max. 32 Zeichen lang sein und keine Leerzeichen enthalten.
Version	SNMP-Version, über welche die Traps gesendet werden sollen: v1 oder v2c (Traps über v3 werden in einem gesonderten Formular konfiguriert).

Tabelle 44: Beschreibung der Parameter der Seite „SNMP“

SNMP v3 Configuration	
<p>Hier wird Ihnen eine Liste aller konfigurierten v3-User angezeigt. Die Anzahl der User ist im I/O-IPC prinzipiell nicht begrenzt. Sie haben die Möglichkeit, die einzelnen User über [DELETE] zu löschen. Am Ende der Liste finden Sie ein Formular, mit dem ein neuer User angelegt werden kann. Die einzelnen Konfigurationsdaten der bereits bestehenden User können nicht geändert werden.</p>	
Security Authentication Name	<p>Benutzer-Name. Dieser muss eindeutig sein; ein bereits vorhandener Benutzer-Name wird bei der Neu-Eingabe nicht akzeptiert. Der Security-Authentication-Name darf max. 32 Zeichen lang sein und keine Leerzeichen enthalten..</p>
Authentication Type	<p>Authentifizierungs-Typ für die SNMP-v3-Pakete. Mögliche Werte sind: - keine Authentifizierung benutzen („None“) - Message Digest 5 („MD5“) - Secure Hash Algorithm („SHA“)</p>
Authentication Key (min. 8 char.)	<p>Schlüssel-String für die Authentifizierung. Der Authentication-Key darf mind. 8 und max. 32 Zeichen lang sein und keine Leerzeichen enthalten..</p>
Privacy	<p>Verschlüsselungs-Algorithmus für die SNMP-Nachricht. Mögliche Werte sind: - keine Verschlüsselung („None“) - Data Encryption Standard („DES“) - Advanced Encryption Standard („AES“)</p>
Privacy Key (min. 8 char.)	<p>Schlüssel-String für die Verschlüsselung („Privacy“) der SNMP-Nachricht. Wird hier nichts angegeben, wird automatisch der „Authentication Key“ verwendet. Der Privacy-Key darf mind. 8 und max. 32 Zeichen lang sein und keine Leerzeichen enthalten..</p>
Notification Receiver IP	<p>IP-Adresse eines Trap-Empfängers für v3-Traps. Falls für diesen User keine v3-Traps gesendet werden sollen, bleibt das Feld leer.</p>

9.1.17 I/O Configuration

Auf der Seite wird die am I/O-IPC angeschlossene Busklemmenkonfiguration mit den Prozesswerten der einzelnen Busklemmen angezeigt.

Tabelle 45: Beschreibung der Parameter der Seite „I/O Configuration“

I/O configuration and values	
Pos	Position der am I/O-IPC angeschlossenen Busklemme. Passive Busklemmen erscheinen nicht im WBM (z. B. 750-600, -602, ...).
Module	Artikelnummer der Busklemme oder Kurzbezeichnung.
Type	Beschreibung, um welche Busklemme es sich handelt (8DI, 4AO, ...).
Channel	Angabe der Klemmenposition und Kanalnummer der Busklemme.
Values	Anzeige der Prozessdaten zum Zeitpunkt der letzten Aktualisierung der Seite „I/O Configuration“. Zur Anzeige der aktuellen Prozessdaten aktualisieren Sie die Seite.

9.1.18 Seite „WebVisu“

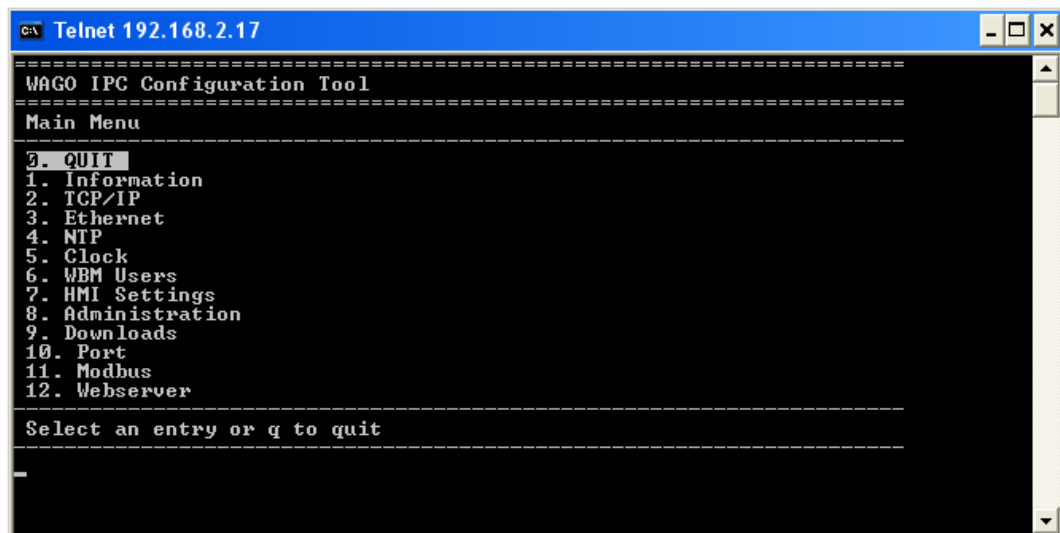
Auf der Seite „WebVisu“ wählen Sie aus, ob bei Eingabe der IP-Adresse des I/O-IPC das WBM oder die CODESYS-Web-Visualisierung erscheinen soll. Zum Speichern aller auf der Seite durchgeführten Einstellungen klicken Sie auf die Schaltfläche **[SUBMIT]**. Die Einstellungen sind erst nach einem Neustart oder Reset des I/O-IPC aktiv. Zur Anzeige der CODESYS-Web-Visualisierung aktualisieren Sie zudem den Internet-Browser.

Um wieder in das WBM zu gelangen, geben zusätzlich zu der IP-Adresse die Port-Nummer „:8080“ an: <http://192.168.2.17:8080> (Socket-Adresse).

Weitere Informationen zu der CODESYS-Web-Visualisierung erhalten Sie im gleichnamigen Kapitel.

9.2 Konfiguration mit einem Terminalprogramm

Sie können den I/O-IPC sowohl über ETHERNET mittels Telnet als auch über die Linux-Konsole mittels der RS-232-Schnittstelle über das IPC-Configuration-Tool konfigurieren. Zum Aufruf des IPC-Configuration-Tools melden Sie sich bei beiden Varianten an der Linux-Konsole an und geben den Befehl *ipconfig* ein. Siehe dazu Kapitel „Zugriff über Telnet“ und Kapitel „Zugriff über RS-232-Schnittstelle und Terminalprogramm“.



```

c:\ Telnet 192.168.2.17
=====
WAGO IPC Configuration Tool
=====
Main Menu
=====
0. QUIT
1. Information
2. TCP/IP
3. Ethernet
4. NTP
5. Clock
6. WBM Users
7. HMI Settings
8. Administration
9. Downloads
10. Port
11. Modbus
12. Webserver
=====
Select an entry or q to quit
=====

```

Abbildung 35: Zugriff auf das IPC-Configuration-Tool mittels Telnet

9.3 Konfiguration mit Touchscreen/Monitor und USB-Tastatur

Vorbereitung:

Sie haben einen Monitor an der DVI-I-Schnittstelle und eine Tastatur an der USB-Schnittstelle des I/O-IPC angeschlossen. Siehe dazu Kapitel „Zugriff über Tastatur und Monitor (DVI- und USB-Schnittstelle)“.

Mittels der Tastenkombination [Alt] + [F3] öffnen Sie die 3. Konsole des I/O-IPC, auf der sich das IPC-Configuration-Tool befindet. Diese Linux-Konsole kann nicht beendet werden. Demnach besitzt „QUIT“ in der Navigationsleiste keine Funktion.

```
=====
WAGO IPC Configuration Tool
=====
Main Menu
-----
0. QUIT
1. Information
2. TCP/IP
3. Ethernet
4. NTP
5. Clock
6. WBM Users
7. HMI Settings
8. Administration
9. Downloads
10. Port
11. Modbus
12. Webserver
-----
Select an entry or q to quit
=====
```

Abbildung 36: Startbild des „IPC Configuration Tool“

10 MODBUS/TCP

Das modulare Konzept der Serie 750 ermöglicht es, bis zu 250 (mit Klemmenbusverlängerung) Busklemmen an den I/O-IPC anzuschließen. Dieser variable Aufbau sowie die große Anzahl verschiedener Busklemmen verhindert jedoch eine statische Zuordnung von Ein- und Ausgangsdaten auf feste MODBUS-Adressen. Einzige Ausnahme sind die „digitalen“ MODBUS-Dienste. Bei ihnen ist die MODBUS-Adresse identisch mit der Kanalnummer, d. h., den 47ten digitalen Eingang findet man immer an MODBUS-Adresse „46“.

Durch das Hinzufügen oder Entfernen von Busklemmen verändert sich der Aufbau der Prozessabbilder und damit auch die MODBUS-Adressen der einzelnen Kanäle der Busklemmen.

Die MODBUS-Kommunikation wird mittels Dienstaufrufen durchgeführt. Dazu sendet der MODBUS-Master (Client) ein Request-Telegramm an Port 502 des MODBUS-Slaves (Server). Der MODBUS-Slave liefert das Ergebnis des Dienstaufwurfes in einem Response-Telegramm an den MODBUS-Master zurück. Die wesentlichsten Elemente eines MODBUS/TCP-Telegramms sind:

Tabelle 46: Elemente eines MODBUS/TCP-Telegramms

Begriff	Beschreibung
UnitID	Kennzeichnung, welches Gerät angesprochen werden soll (<FF)
FunctionCode (FC)	Dienstkennung: Lese- oder Schreiboperation auf Bits oder Words
Address	Startadresse der Operation
Count	Dienstabhängig die Anzahl Bits oder Words
[Data]	Prozessdaten

Die Dienstkennung bzw. der „FunctionCode“ (FC) bestimmt zunächst, ob es sich um eine Lese- oder Schreiboperation handelt. Zusätzlich bestimmt sie den Grunddatentyp, auf den die Operation angewendet werden soll. Damit ist auch die Bedeutung der Parameter „Address“ und „Count“ abhängig vom Funktionscode. So kann „address :=3“ für ein Bit oder ein Word im Ein- oder Ausgangsprozessabbild stehen.

Das MODBUS/TCP-Protokoll basiert im Wesentlichen auf den folgenden Grunddatentypen:

Tabelle 47: MODBUS-Grunddatentypen


Datentyp	Länge	Beschreibung
Discrete Inputs	1 Bit	Digitale Eingänge
Coils	1 Bit	Digitale Ausgänge
Input Register	16 Bit	Analoge Eingangsdaten
Holding Register	16 Bit	Analoge Ausgangsdaten

Für jeden Grunddatentyp sind ein oder mehrere „FunctionCodes“ definiert.

10.1 Prozessdaten des MODBUS-Servers

Über die Word-Dienste des MODBUS-Servers erreichen Sie den ersten analogen Aus- bzw. Eingang oder den digitalen, wenn kein analoger Ausgang vorhanden ist.

Eine Besonderheit beim Zugriff über MODBUS ist, dass Sie mit den „digitalen“ MODBUS-Diensten an der I/O-IPC-Adresse 0 immer auf den ersten digitalen Ausgang bzw. Eingang des Klemmbusprozessabbildes zugreifen, obwohl digitale und analoge Prozessdaten des I/O-IPC und der Busklemmen in einem Prozessabbild zusammengefasst sind. Informationen zur Länge der Prozessdaten erhalten Sie im Kapitel „Aufbau der Prozessdaten für MODBUS/TCP“.

WARNUNG **Aktivierung der Betriebsart „Control-Mode“ in WAGO-IO-CHECK!**
 Bei Verwendung von WAGO-IO-CHECK können Sie in der Betriebsart „Control-Mode“ - unabhängig davon, ob Feldbus oder SPS-Funktionalitäten aktiviert oder deaktiviert sind - Prozessdaten und Parameter überschreiben. Dadurch können Maschinenkomponenten in einen gefährlichen Zustand versetzt sowie Personal und Maschine gefährdet werden.

Vor Änderung der Parameter und vor Änderung von Prozessdaten bringen Sie die Maschinenkomponenten in einen definierten und sicheren Zustand und schalten Sie die übergeordnete Steuerung aus. Vergewissern Sie sich vor Inbetriebnahme, dass sich kein Personal im Gefahrenbereich der Maschinenkomponenten aufhält.

10.2 Zugriff auf das Prozessabbild über MODBUS

Die folgende Tabelle beschreibt die MODBUS-Funktionscodes, mit denen Sie auf die Adressbereiche des Prozessabbilds für die am Klemmenbus angeschlossenen Ein- und Ausgänge zugreifen:

Tabelle 48: MODBUS-Funktionscodes

FC	Name	Beschreibung
FC1	Read coils	Rücklesen mehrerer digitaler Ausgangswerte
FC2	Read inputs discrete	Lesen mehrerer digitaler Eingangswerte
FC3	Read holding registers	Lesen mehrerer analoger Ausgangswerte
FC4	Read input registers	Lesen mehrerer analoger Eingangswerte
FC5	Write coil	Schreiben eines einzelnen digitalen Ausgangswerts
FC6	Write single register	Schreiben eines einzelnen analogen Ausgangswerts
FC15	Force multiple coils	Schreiben mehrerer digitaler Ausgangswerte
FC16	Write multiple registers	Schreiben mehrerer analoger Ausgangswerte
FC23	Read/write multiple registers	Schreib- und Leseoperation auf analoge Ein- und Ausgangswerte

10.2.1 Registerdienste

Mit den Registerdiensten ermitteln oder verändern Sie die Zustände von Analogein- und -ausgangsklemmen für die folgenden Adressbereiche:

Tabelle 49: Lesen von Analogeingangsklemmen mittels FC3, FC4, FC23

MODBUS-Adresse	Adressen in CODESYS	Beschreibung
0x0000 – 0x00FF (0 – 255)	%IW0 ... %IW255	Lesen analoger oder digitaler Eingangswerte. Physikalischer Adressraum der Eingangsdaten von 256 Wörtern.
0x100 – 0x1FF (256 – 511)	%QW256 bis %QW511	Lesen der PFC-Variablen
0x1000 – 0x2FFF (4096 – 12287)	Siehe MODBUS-Konfigurationsregister	MODBUS-Konfigurationsregister
0x3000 – 0x3FFF (12288 – 16384) oder einstellbar bis max. 0xFFFF (65534)	%MW0 ... %MW4095 %MW0 ... %MW53247	Retain-Speicher (8 kB) Nichtflüchtige SPS-Variablen Retain-Speicher (24 kB) Nichtflüchtige SPS-Variablen

Tabelle 50: Schreiben von Analogausgangsklemmen mittels FC6, FC16, FC23

MODBUS-Adresse	Adressen in CODESYS	Beschreibung
0x0000 – 0x00FF (0 – 255)	%QW0 ... %QW255	Schreiben analoger oder digitaler Ausgangswerte. Physikalischer Adressraum der Ausgangsdaten von 256 Wörtern.
0x100 – 0x1FF (256 – 511)	%IW256 bis %IW511	Schreiben der PFC-Variablen
0x1000 – 0x2FFF (4096 – 12287)	Siehe MODBUS-Konfigurationsregister	MODBUS-Konfigurationsregister
0x3000 – 0x3FFF (12288 – 16384) oder einstellbar bis max. 0xFFFF (65534)	%MW0 ... %MW4095 %MW0 ... %MW53247	Retain-Speicher (8 kB) Nichtflüchtige SPS-Variablen Retain-Speicher (24 kB) Nichtflüchtige SPS-Variablen

10.2.2 Bitdienste

Mit den digitalen Bitdiensten ermitteln oder verändern Sie die Zustände von Digitalein- und -ausgangsklemmen für die folgenden Adressbereiche:

Tabelle 51: Lesen von Digitaleingangsklemmen mittels FC1, FC2

MODBUS-Adresse	Adressen in CODESYS	Beschreibung
0x0000 – 0x01FF (0 – 511)	%IX 0.0 ... %IX 32.15 + Offsetwert	Eingangsprozessabbild Die bitweise Adressierung bei MODBUS beginnt bei der ersten Digitaleingangsklemme am Klemmenbus. Bei Verwendung von Analogeingangsklemmen wird der Adressbereich, welcher durch diese belegt wird, von der Adressierung übersprungen (Offsetwert).
0x0400 – 0x0401 (1024 – 1025)	%IX2300.0 ... %IX2300.1	Integrierte digitale Eingänge
0x3000 – 0x7FFF (12288 – 32750)	%MX0.0 ... %MX1279.15	Retain-Speicher (8 kB) Nichtflüchtige SPS-Variablen

Tabelle 52: Schreiben von Digitalausgangsklemmen mittels FC5, FC15

MODBUS-Adresse	Adressen in CODESYS	Beschreibung
0x0000 – 0x01FF (0 – 511)	%QX 0.0 ... %QX 32.15 + Offsetwert	512 digitale Ausgangsdaten. Die bitweise Adressierung bei MODBUS beginnt bei der ersten Digitalausgangsklemme am Klemmenbus. Bei Verwendung von Analogausgangsklemmen wird der Adressbereich, welcher durch diese belegt wird, von der Adressierung übersprungen (Offsetwert).
0x0400 – 0x0401 (1024 – 1025)	%QX2300.0 ... %QX2003.1	Integrierte digitale Ausgänge
0x3000 – 0x7FFF (12288 – 32750)	%MX0.0 ... %MX1023.15	Retain-Speicher (8 kB) Nichtflüchtige SPS-Variablen

10.3 Konfigurationsregister

Mittels der MODBUS-Konfigurationsregister können Sie den I/O-IPC konfigurieren und Informationen über diesen auslesen.

Tabelle 53: Konfigurationsregister

MODBUS-Adresse	Länge [Word]	Zugriff	Beschreibung
0x1031 (4145)	3	Read	MAC-Adresse der ETHERNET-Schnittstelle X8.
0x1034 (4148)	3	Read	MAC-Adresse der ETHERNET-Schnittstelle X9.
0x1030 (4144)	1	Read/write	Hier stellen Sie die Zeitspanne der MODBUS/TCP-Verbindung ein, nach der die Verbindung bei einer Unterbrechung der Kommunikation automatisch beendet wird.

10.4 Adressierungsbeispiel

Folgendes Adressierungsbeispiel verdeutlicht den Zugriff auf das Prozessabbild:

Tabelle 54: Anordnung der Busklemmen für das Adressierungsbeispiel

I/O-IPC	750-400	750-554	750-402	750-504	750-454	750-650	750-468	750-600
	1	2	3	4	5	6	7	8

Tabelle 36: Adressierungsbeispiel

Busklemme		Eingangsdaten		Ausgangsdaten		Beschreibung
Typ	C*	FC3, FC4	FC1, FC2	FC6	FC5	
750-400	1	0008	00000			2DI, 24 V, 3 ms: 1. Digitaleingangsklemme mit einer Datenbreite von 2 Bit. Da die Analogeingangsklemmen bereits die ersten 8 Wörter des Eingangsprozessabbilds besetzen, belegen die 2 Bit die niederwertigsten Bits des 8. Wortes.
	2		00001			
750-554	1			00000		2AO, 4 – 20 mA: 1. Analogausgangsklemme mit einer Datenbreite von 2 Wörtern. Diese belegt die ersten 2 Wörter im Ausgangsprozessabbild.
	2			00001		
750-402	1	0008	00002			4DI, 24 V: 2. Digitaleingangsklemme mit einer Datenbreite von 4 Bit. Diese werden zu den 2 Bit der 750-400 hinzugefügt und in das 8. Wort des Eingangsprozessabbilds abgelegt.
	2		00003			
	3		00004			
	4		00005			
750-504	1			00004	0000	4DO, 24 V: 1. Digitalausgangsklemme mit einer Datenbreite von 4 Bit. Da die Analogausgangsklemme bereits die ersten 4 Wörter des Ausgangsprozessabbilds besetzt, belegen die 4 Bit die niederwertigsten Bits des 4. Wortes.
	2				0001	
	3				0002	
	4				0003	
750-454	1	0000				2AI, 4 – 20 mA: 1. Analogeingangsklemme mit einer Datenbreite von 2 Wörtern. Diese belegt die ersten 2 Wörter im Eingangsprozessabbild.
	2	0001				
750-650	1	0002		00002		RS232, C 9600/8/N/1: Die serielle Schnittstellenklemme ist eine Analogeingangs- und -ausgangsklemme, die sich sowohl im Eingangsprozessabbild als auch im Ausgangsprozessabbild mit je 2 Wörtern darstellt.
		0003		00003		
750-468	1	0004				4AI, 0 – 10 V S.E: 2. Analogeingangsklemme mit einer Datenbreite von 4 Wörtern. Da die Analogeingangs- und -ausgangsklemmen 750-454 und 750-650 bereits die ersten 4 Wörter des Eingangsprozessabbilds belegen, werden die 4 Wörter dieser Busklemme hinter den der anderen hinzugefügt.
	2	0005				
	3	0006				
	4	0007				

Tabelle 36: Adressierungsbeispiel

Busklemme		Eingangsdaten		Ausgangsdaten		Beschreibung
Typ	C*	FC3, FC4	FC1, FC2	FC6	FC5	
750-600						Endklemme Die passive Endklemme 750-600 überträgt keine Daten.

Analogein- und -ausgangsklemmen
 Digitalein- und -ausgangsklemmen

*C: Nummer des Ein-/Ausgangs

11 Laufzeitumgebung CODESYS 2.3

11.1 Prozessabbilder

Ein Prozessabbild ist ein Speicherbereich, in dem die Prozessdaten in einer definierten Reihenfolge abgelegt sind. Es setzt sich zusammen aus den am Klemmenbus angeschlossenen Busklemmen, den PFC-Variablen, dem Merkerbereich und den am Feldbus angeschlossenen Slaves.

Der Zugriff auf die Prozessabbilder durch MODBUS und CODESYS unterscheidet sich. Informationen zum Zugriff auf die Prozessdaten der am I/O-IPC angeschlossenen Busklemmen und Slaves erhalten Sie in den Kapiteln „Zugriff auf das Prozessabbild über MODBUS/TCP“ und „Zugriff auf die Prozessabbilder der Ein- und Ausgangsdaten über CODESYS 2.3“.

11.1.1 Prozessabbild für die am I/O-IPC angeschlossenen Busklemmen

Nach Inbetriebnahme des I/O-IPC ermittelt dieser automatisch alle angeschlossenen Busklemmen. Der I/O-IPC erstellt daraus ein Prozessabbild, unterteilt in einen Bereich von jeweils maximal 500 Byte für die Ein- und Ausgangsdaten.

Im Prozessabbild werden zuerst die analogen Ein- und Ausgangsdaten wortweise abgelegt. Im Anschluss folgen die zu Wörtern zusammengefassten Bits der digitalen Ein- und Ausgangsdaten.

Hinweis



Datenbreite einer Busklemme

Die Datenbreite einer Busklemme kann zwischen 0 und 48 Byte betragen. Detaillierte Informationen zur jeweiligen Prozessdatenbreite einzelner Busklemmen entnehmen Sie Kapitel „Aufbau der Prozessdaten für MODBUS/TCP“.

Hinweis



Prozessdaten der Busklemmen

Überprüfen Sie die Prozessdaten der Busklemmen, wenn Sie diese an dem I/O-IPC hinzufügen oder entfernen: Durch die Änderung der Busklemmentopologie ergibt sich eine Verschiebung des Prozessabbilds, da sich die Adressen der Prozessdaten ändern.

11.1.2 Prozessabbild für die am Feldbus angeschlossenen Slaves

Am I/O-IPC lassen sich bis zu 126 Slaves anschließen. Der I/O-IPC kann Eingangsdaten bis zu einer Größe von 3584 Byte von den Slaves empfangen und 3584 Byte Ausgangsdaten an die Slaves senden. Beachten Sie dazu die Netzwerk-Richtlinien für den verwendeten Feldbusses. Die Feldbuskonfiguration führen Sie mit der CODESYS-Steuerungskonfiguration durch. Siehe dazu Kapitel „Anlegen der Steuerungskonfiguration“.

11.2 Schreibweise logischer Adressen

Den Zugriff auf individuelle Speicherelemente gemäß IEC 61131-3 ist nur durch folgende Zeichen möglich:

Tabelle 55: Schreibweise logischer Adressen

Position	Zeichen	Bezeichnung	Anmerkungen
1	%	Startet absolute Adresse	-
2	I	Eingang	
	Q	Ausgang	
	M	Merker	
3	X	Einzelbit	Datenbreite
	B	Byte (8 Bits)	
	W	Wort (16 Bits)	
	D	Doppelwort (32 Bits)	
4		Adresse	

Nachfolgend zwei Beispiele:

Adressierung wortweise %QW27 (28. Wort)
Adressierung bitweise %IX1.9 (10. Bit im Wort 2)

Geben Sie die Zeichenfolge der absoluten Adresse ohne Leerstellen ein. Das erste Bit eines Wortes hat die Adresse 0.

11.3 Zugriff auf die Prozessabbilder der Ein- und Ausgangsdaten über CODESYS 2.3

Die folgende Tabelle beschreibt die Möglichkeiten, mit denen Sie auf die Adressbereiche des Prozessabbilds für die am Klemmenbus und am Feldbus (nur bei I/O-IPCs mit Feldbusanschluss) angeschlossenen Ein- und Ausgänge zugreifen:

Tabelle 56: Zugriff auf die Prozessabbilder der Ein- und Ausgangsdaten über CODESYS

Speicherbereich	Beschreibung	Zugriff über MODBUS-TCP	Zugriff über die SPS	Logischer Adressbereich
Eingangsprozessabbild	Abbild der lokalen Eingangsklemmen (Klemmenbus, Busklemme 1 bis 64*) im RAM.	Lesen	Lesen	Wort %IW 0 bis %IW255
				Byte %IB 0 bis %IB511
Ausgangsprozessabbild	Abbild der lokalen Ausgangsklemmen (Klemmenbus, Busklemme 1 bis 64*) im RAM.	Schreiben	Lesen/ Schreiben	Wort %QW 0 bis %QW255
				Byte %QB 0 bis %QB511
Eingangsprozessabbild der SPS	Abbild der SPS-Eingangsvariablen im RAM. Der Zugriff mit MODBUS/TCP ist möglich.	Lesen/ Schreiben	Lesen	Wort %IW256 bis %IW511
				Byte %IB 512 bis %IB 1023
Ausgangsprozessabbild der SPS	Abbild der SPS-Ausgangsvariablen im RAM. Der Zugriff mit MODBUS/TCP ist möglich.	Lesen	Lesen/ Schreiben	Wort %QW256 bis %QW511
				Byte %QB 512 bis %QB 1023
Integrierter digitaler Eingang	Abbild der digitalen I/O-IPC-Eingangsbits 0,1 im RAM.	-	Lesen	Bit %IX 2300.0 bis %IX 2300.1
Integrierter digitaler Ausgang	Abbild der digitalen I/O-IPC-Ausgangsbits 0,1 im RAM.	-	Lesen/ Schreiben	Bit %QX 2300.0 bis %QX 2300.1
Feldbus-Eingangsva- riablen**	Eingangsvariablen des konfigurierten Feldbusses im RAM.	-	Lesen	Wort %IW2400 bis %IW31750
				Byte %IB4800 bis %IB65535
Feldbus-Ausgangs- va- riablen**	Ausgangsvariablen des konfigurierten Feldbusses im RAM.	-	Schreiben/ Lesen	Wort %QW2400 bis %QW31750
				Byte %QB4800 bis %QB65535

Tabelle 56: Zugriff auf die Prozessabbilder der Ein- und Ausgangsdaten über CODESYS

Speicherbereich	Beschreibung	Zugriff über MODBUS-TCP	Zugriff über die SPS	Logischer Adressbereich
Merker-Variablen	8 kB remanenter Speicher im SRAM. Deklariert mit „AT %M<Adresse>“. Erweiterbar auf 24 kB.	Lesen/ Schreiben	Lesen/ Schreiben	Wort %MW0 bis %MW 4095
				Byte %MB0 bis %MB8190 (%MW12287)
Retain-Variablen	Symbolisch adressierbarer Retain-Speicher im SRAM: 1023 kB	-	Lesen/ Schreiben	-

* Mit der WAGO-Klemmenbusverlängerung ist die Nutzung von bis zu 250 Busklemmen möglich.

** Nur bei I/O-IPCs mit Feldbusanschluss

Die Gesamtgröße des Speichers für die Merker- und Retain-Variablen beträgt 1023 kB. Verwenden Sie eine bitorientierte Adressierung, beachten Sie, dass die Basisadresse wortbasierend ist. Die Bits befinden sich im Bereich 0 bis15.

WARNUNG Aktivierung der Betriebsart „Control-Mode“ in WAGO-IO-CHECK!



Bei Verwendung von WAGO-IO-CHECK können Sie in der Betriebsart „Control-Mode“ - unabhängig davon, ob Feldbus oder SPS-Funktionalitäten aktiviert oder deaktiviert sind - Prozessdaten und Parameter überschreiben. Dadurch können Maschinenkomponenten in einen gefährlichen Zustand versetzt sowie Personal und Maschine gefährdet werden. Vor Änderung der Parameter und vor Änderung von Prozessdaten bringen Sie die Maschinenkomponenten in einen definierten und sicheren Zustand und schalten Sie die übergeordnete Steuerung aus. Vergewissern Sie sich vor Inbetriebnahme, dass sich kein Personal im Gefahrenbereich der Maschinenkomponenten aufhält.

Anpassen des remanenten Speicherbereichs

Beim Anlegen eines Projekts erscheint ein Konfigurationsfenster zur Auswahl des Zielsystems (siehe Kapitel „Anlegen eines Projekts und Auswahl des Zielsystems“).

1. Zum Anpassen des remanenten Speicherbereichs klicken Sie im Konfigurationsfenster „Zielsystem Einstellungen“ auf der Registerkarte „Speicheraufteilung“.
2. Tragen Sie in das Feld „Memory“ und „Retain“ folgende Werte ein:

- **Remanenter Speicherbereich von 8 kB**
Memory: 16#2000 (8 kB)
Retain: 16#FDF00 (1015 kB)
Summe: 16#FFF00 (1023 kB)
- **Remanenter Speicherbereich von 16 kB**
Memory: 16#4000 (16 kB)
Retain: 16#FBF00 (1007 kB)
Summe: 16#FFF00 (1023 kB)
- **Remanenter Speicherbereich von 24 kB**
Memory: 16#6000 (24 kB)
Retain: 16#F9F00 (999 kB)
Summe: 16#FFF00 (1023 kB)

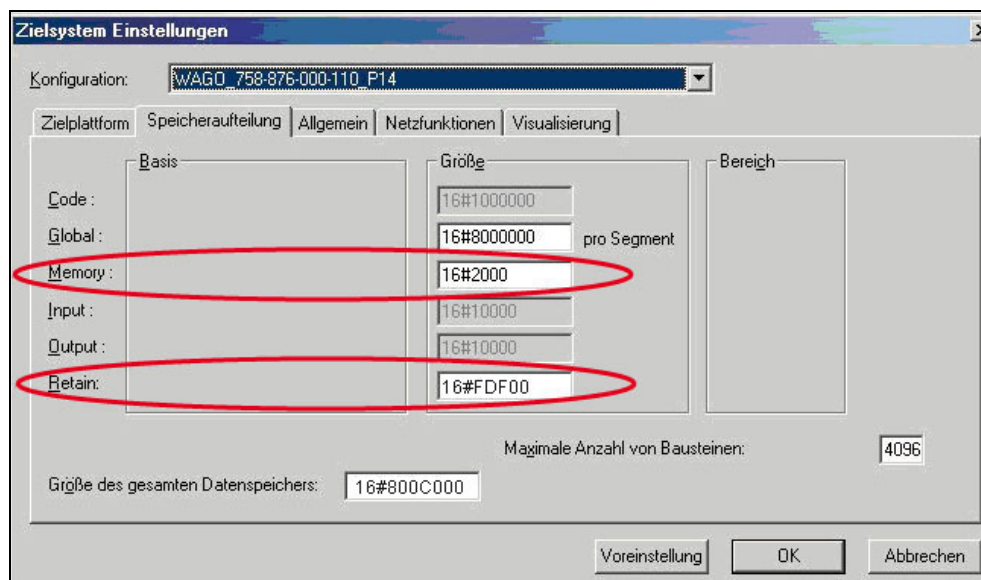


Abbildung 37: Anpassen des remanenten Speicherbereichs

11.4 Adressierungsbeispiel

Folgendes Adressierungsbeispiel verdeutlicht den Zugriff auf das Prozessabbild:

Tabelle 57: Anordnung der Busklemmen für das Adressierungsbeispiel

I/O-IPC	750-400	750-554	750-402	750-504	750-454	750-650	750-468	750-600
	1	2	3	4	5	6	7	8

Tabelle 36: Adressierungsbeispiel

Busklemme	Eingangsdaten	Ausgangsdaten	Beschreibung
Typ	C*		
750-400	1	%IX8.0	2DI, 24 V, 3 ms: 1. Digitaleingangsklemme mit einer Datenbreite von 2 Bit. Da die Analogeingangsklemmen bereits die ersten 8 Wörter des Eingangsprozessabbilds besetzen, belegen die 2 Bit die niederwertigsten Bits des 8. Wortes.
	2	%IX8.1	
750-554	1	%QW0	2AO, 4 – 20 mA: 1. Analogausgangsklemme mit einer Datenbreite von 2 Wörtern. Diese belegt die ersten 2 Wörter im Ausgangsprozessabbild.
	2	%QW1	
750-402	1	%IX8.2	4DI, 24 V: 2. Digitaleingangsklemme mit einer Datenbreite von 4 Bit. Diese werden zu den 2 Bit der 750-400 hinzugefügt und in das 8. Wort des Eingangsprozessabbilds abgelegt.
	2	%IX8.3	
	3	%IX8.4	
	4	%IX8.5	
750-504	1	%QX4.0	4DO, 24 V: 1. Digitalausgangsklemme mit einer Datenbreite von 4 Bit. Da die Analogausgangsklemme bereits die ersten 4 Wörter des Ausgangsprozessabbilds besetzt, belegen die 4 Bit die niederwertigsten Bits des 4. Wortes.
	2	%QX4.1	
	3	%QX4.2	
	4	%QX4.3	
750-454	1	%IW0	2AI, 4 – 20 mA: 1. Analogeingangsklemme mit einer Datenbreite von 2 Wörtern. Diese belegt die ersten 2 Wörter im Eingangsprozessabbild.
	2	%IW1	
750-650	1	%IW2	RS232, C 9600/8/N/1: Die serielle Schnittstellenklemme ist eine Analogeingangs- und -ausgangsklemme, die sich sowohl im Eingangsprozessabbild als auch im Ausgangsprozessabbild mit je 2 Wörtern darstellt.
		%IW3	
		%QW2	
		%QW3	
750-468	1	%IW4	4AI, 0 – 10 V S.E.: 2. Analogeingangsklemme mit einer Datenbreite von 4 Wörtern. Da die Analogein- und -ausgangsklemmen 750-454 und 750-650 bereits die ersten 4 Wörter des Eingangsprozessabbilds belegen, werden die 4 Wörter dieser Busklemme hinter den der anderen hinzugefügt.
	2	%IW5	
	3	%IW6	
	4	%IW7	

Tabelle 36: Adressierungsbeispiel

Busklemme		Eingangsdaten		Ausgangsdaten		Beschreibung
Typ	C*					
750-600						Endklemme Die passive Endklemme 750-600 überträgt keine Daten.

Analogein- und -ausgangsklemmen

Digitalein- und -ausgangsklemmen

*C: Nummer des Ein-/Ausgangs

11.5 Installieren des Programmiersystems CODESYS 2.3

Die Installation von CODESYS umfasst zusätzlich die WAGO-Targetfiles. Diese beinhalten alle gerätespezifischen Informationen für die WAGO-Produktserien 750/758.

Gehen Sie wie nachfolgend beschrieben vor, um die Programmiersoftware CODESYS 2.3 auf dem I/O-IPC zu installieren.

1. Legen Sie die CD-ROM „WAGO-I/O-PRO CAA“ in Ihr Computerlaufwerk ein.
2. Zur Installation des Programmiersystems folgen Sie den Anweisungen, die auf Ihrem Bildschirm erscheinen. Bei erfolgreicher Installation erscheint das CODESYS-Piktogramm auf Ihrem Desktop.

11.6 Das erste Programm mit CODESYS 2.3

Dieses Kapitel erläutert anhand eines Beispiels die relevanten Schritte, die Sie zur Erstellung eines CODESYS-Projekts benötigen. Es dient als Schnellstartanleitung und beinhaltet nicht den vollen Funktionsumfang von CODESYS 2.3.

Information Weitere Informationen



Eine detaillierte Beschreibung des vollen Funktionsumfangs entnehmen Sie bitte dem Handbuch „Handbuch für die SPS-Programmierung mit CODESYS 2.3“ auf der CD „WAGO-I/O-PRO CAA“ (759-911).

11.6.1 Starten Sie das Programmiersystem CODESYS

Starten Sie CODESYS durch einen Doppelklick auf das CODESYS-Piktogramm auf Ihrem Desktop oder über das über das Startmenü Ihres Betriebssystems.

Klicken Sie dazu auf die Schaltfläche „Start“ und wählen **Programme > WAGO Software > CODESYS for Automation Alliance > CODESYS V2.3**.

11.6.2 Anlegen eines Projekts und Auswahl des Zielsystems

1. Klicken Sie in der Menüleiste auf **Datei** und wählen Sie **Neu**. Es öffnet sich das Fenster „Zielsystem Einstellung“. Hier sind alle verfügbaren Zielsysteme aufgelistet, die sich mit CODESYS 2.3 programmieren lassen.
2. Öffnen Sie das Auswahlfeld des Fensters „Zielsystem Einstellung“ und wählen Sie den von Ihnen verwendeten I/O-IPC aus. In diesem Beispiel ist es der I/O-IPC vom Typ PROFIBUS-Master „WAGO_758-876-000-111“.
3. Klicken Sie auf die Schaltfläche **[OK]**. Es öffnet sich das Konfigurationsfenster „Zielsystem Einstellungen“.

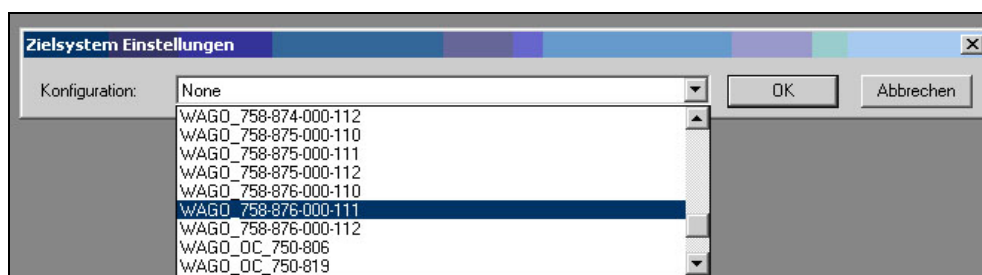


Abbildung 38: Zielsystem-Einstellungen (1)

4. Zum Übernehmen der Standardkonfiguration für den I/O-IPC klicken Sie auf die Schaltfläche **[OK]**. Es öffnet sich das Fenster „Neuer Baustein“.

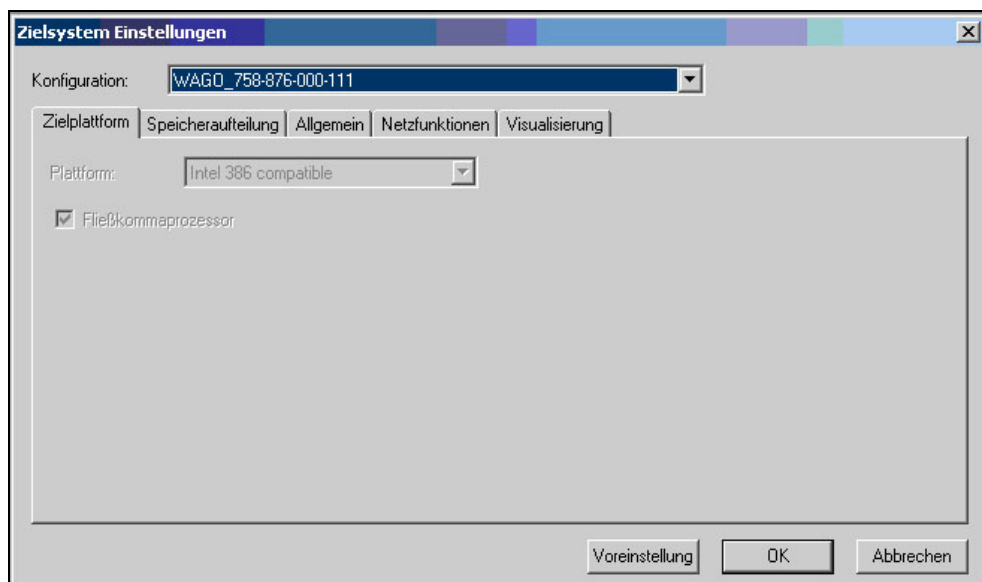


Abbildung 39: Zielsystem-Einstellungen (2)

5. Legen Sie im Fenster „Neuer Baustein“ ein Programmbaustein an. In diesem Beispiel wird ein neuer Baustein „PLC_PRG“ in der Programmiersprache „ST“ angelegt.
6. Klicken Sie auf [OK], um das Projekt zu erzeugen. Es öffnet sich die Programmieroberfläche.

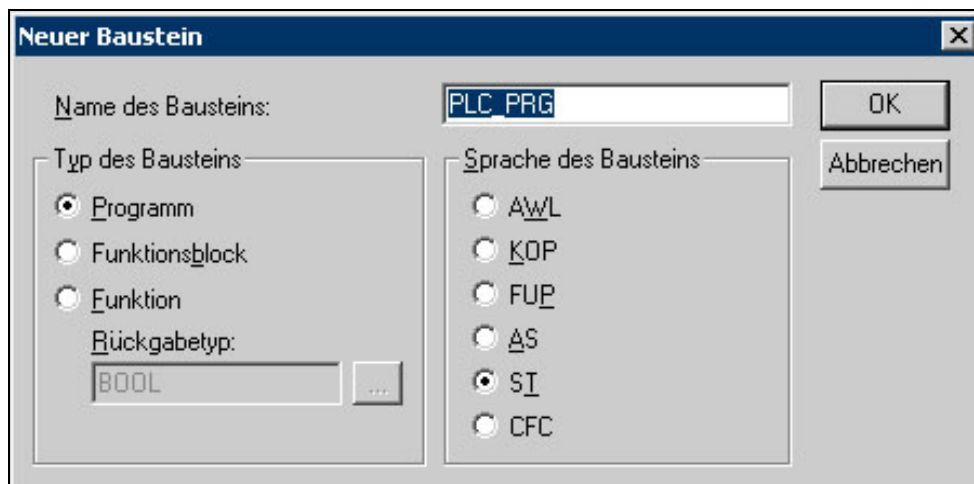


Abbildung 40: Anlegen eines neuen Bausteins

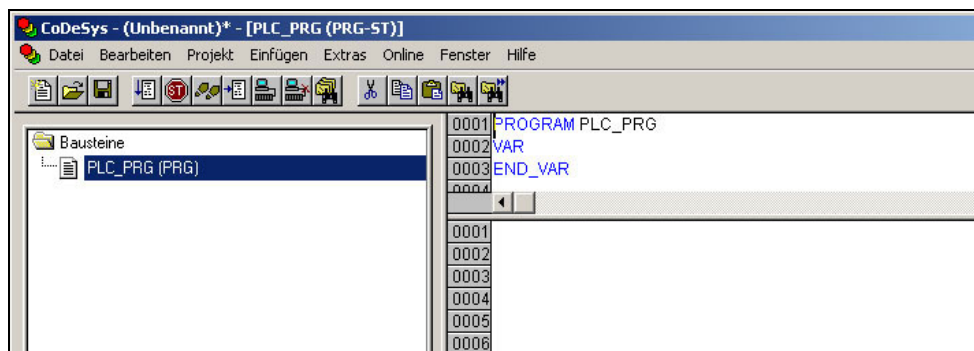


Abbildung 41: Programmieroberfläche mit dem Programmbaustein PLC_PRG

11.6.3 Anlegen der Steuerungskonfiguration

Hinweis



Vorgehensweise bei Anlegen der Steuerungskonfiguration

Verwenden Sie einen I/O-IPC mit einem Feldbusanschluss, dann gehen Sie zum Anlegen der Steuerungskonfiguration vor, wie im Kapitel „CANopen-Master in CODESYS 2.3“ bzw. „PROFIBUS-Master in CODESYS 2.3“ beschrieben.

Bei einer I/O-IPC-Variante ohne Feldbusanschluss gehen Sie entsprechend diesem Kapitel vor.

Die Steuerungskonfiguration dient dazu, den I/O-IPC mit den daran angeschlossenen Busklemmen zu konfigurieren und Variablen zu deklarieren, um auf die Ein- oder Ausgänge der Busklemmen zuzugreifen. Gehen Sie dazu folgendermaßen vor:

1. Klicken Sie auf der Registerkarte „Ressourcen“.

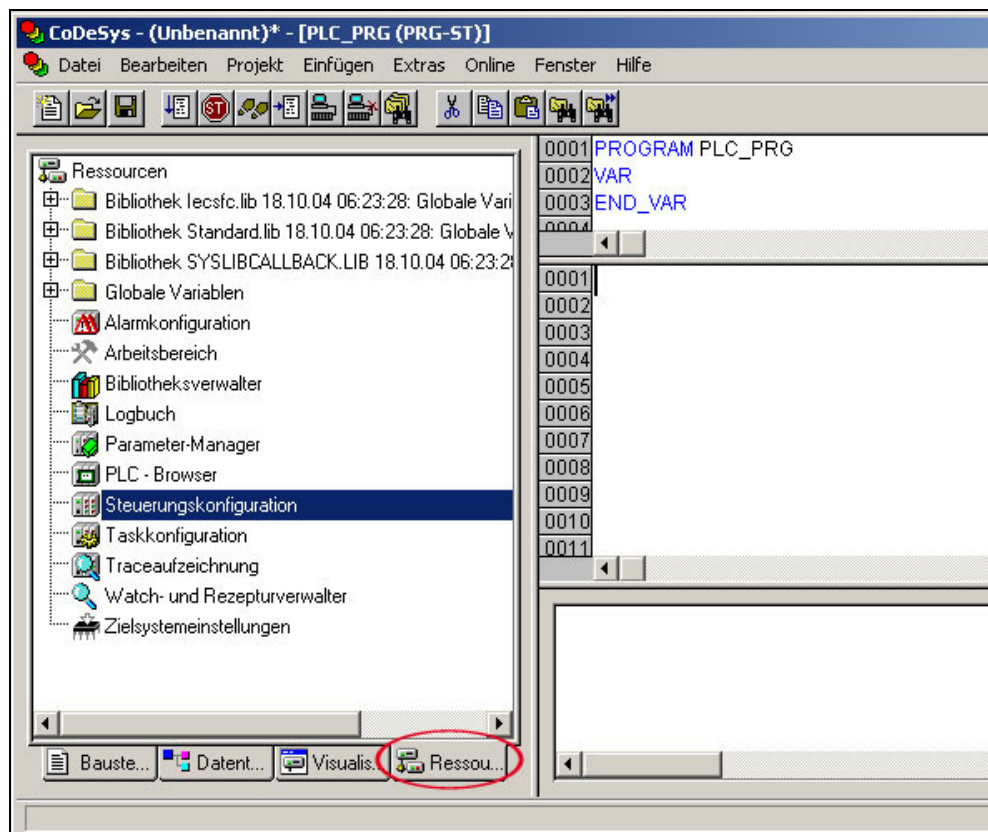


Abbildung 42: Registerkarte „Ressourcen“

2. Klicken Sie im linken Fenster mit einem Doppelklick auf „Steuerungskonfiguration“. Es öffnet sich die Steuerungskonfiguration des I/O-IPC.
3. Klicken Sie mit der rechten Maustaste auf den Eintrag „K-Bus[Fix]“ und wählen Sie im Kontextmenü „Bearbeiten“.

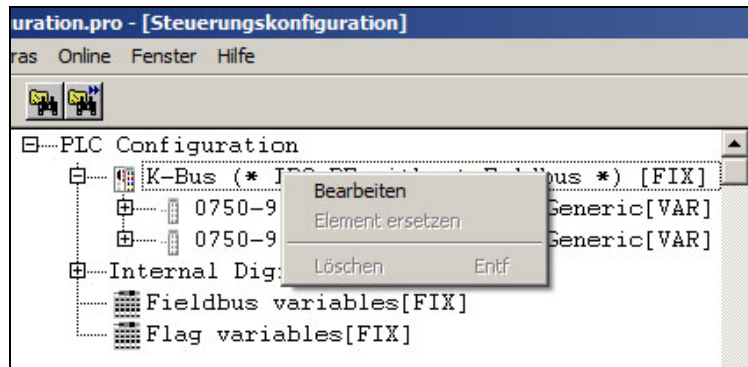


Abbildung 43: Steuerungskonfiguration: Bearbeiten

4. Es öffnet sich der Dialog „Konfiguration“.

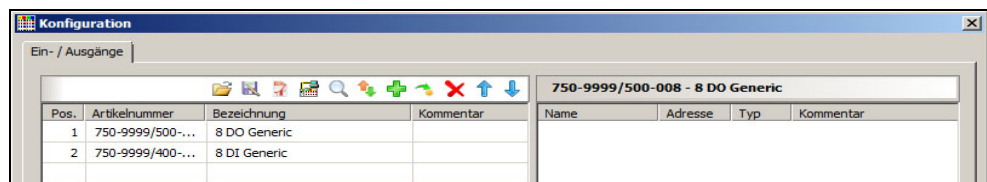


Abbildung 44: Konfiguration

5. Um die Konfiguration vorzunehmen oder zu ändern, können Sie mit der Schaltfläche **[Hinzufügen]** neue Busklemmen hinzufügen.

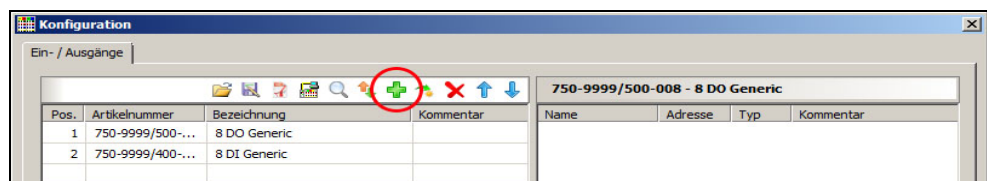


Abbildung 45: Schaltfläche „Busklemmen hinzufügen“

6. Im neu erscheinenden Fenster „Modulauswahl“ können Sie nun die gewünschten Module auswählen.

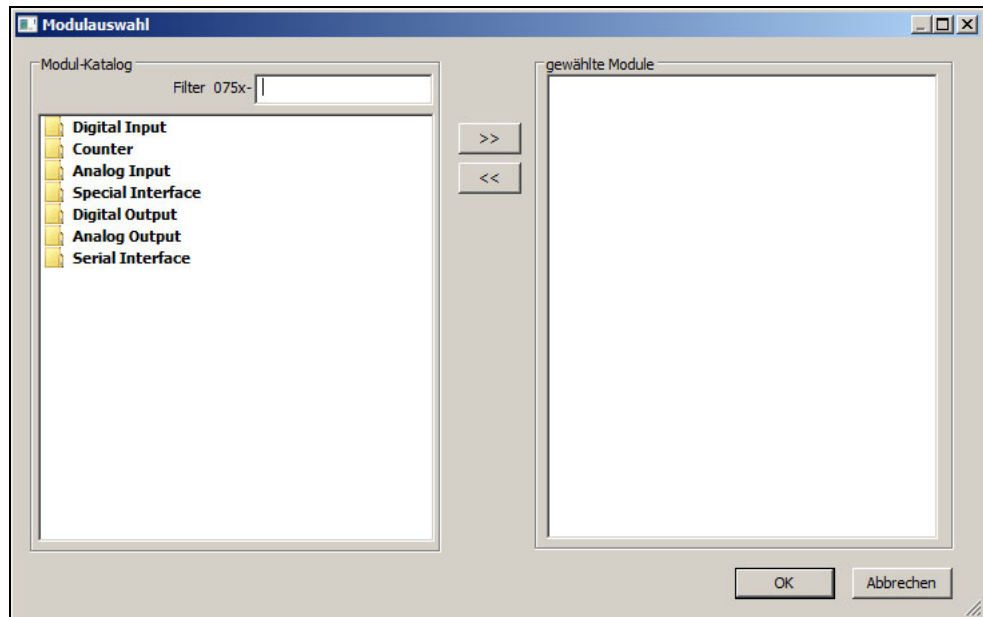


Abbildung 46: Fenster „Modulwahl“

- Die Position einer Busklemme verändern Sie, indem Sie diese markieren und mittels der Pfeil-Tasten am rechten Rand des Fensters nach oben oder nach unten verschieben.

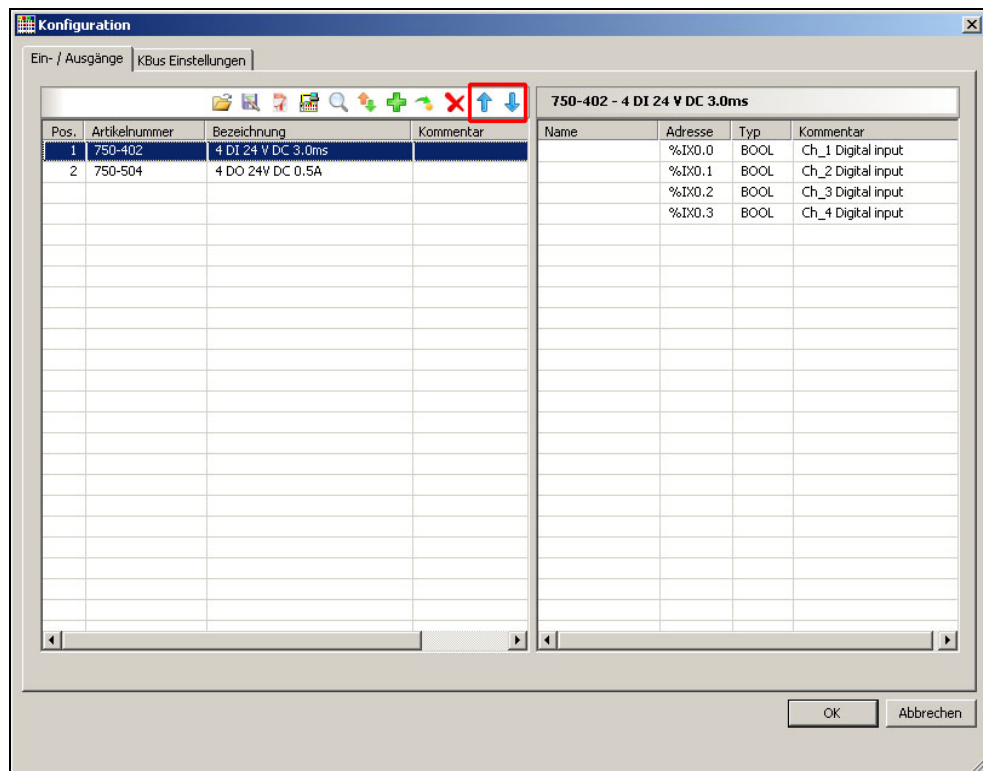


Abbildung 47: I/O-Konfigurator mit eingetragenen Busklemmen

8. Im rechten Teil des Konfigurationsfensters werden die einzelnen Ein- bzw. Ausgänge der jeweils ausgewählten Busklemme angezeigt. Hier können Sie in der Spalte „Name“ für jeden Ein- und Ausgang eine eigene Variable deklarieren. z. B. „Ausgang_1“, „Ausgang_2“, „Eingang_1“, „Eingang_2“.

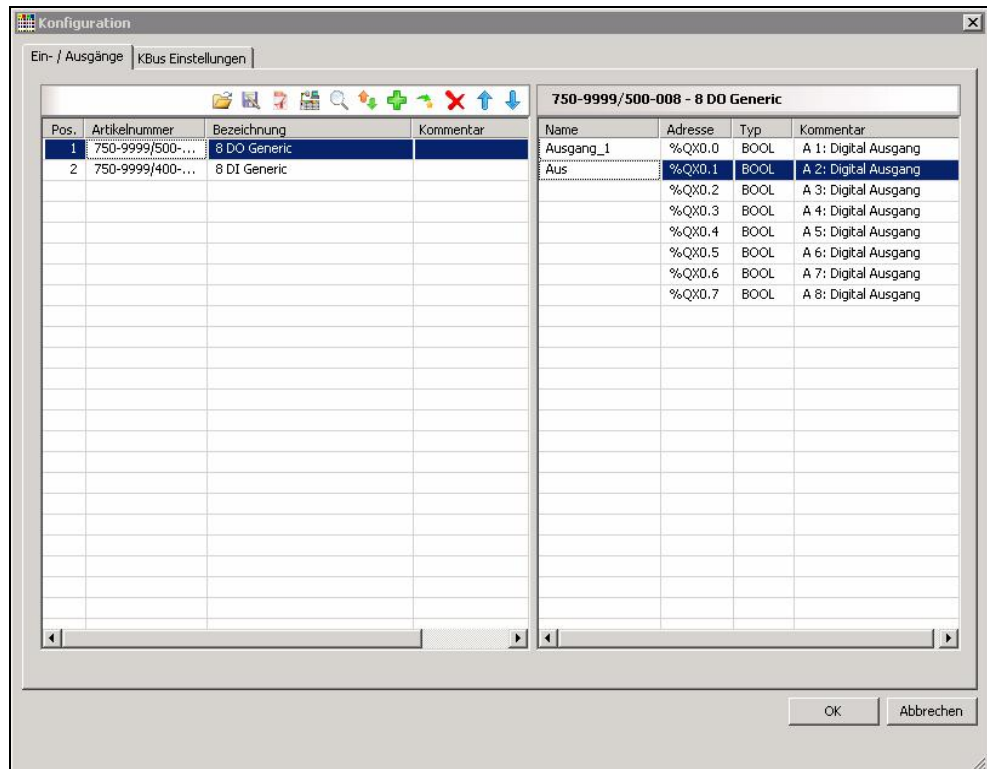


Abbildung 48: Variablendeklaration

9. Zum Beenden des I/O-Konfigurators klicken Sie auf **[OK]**.
10. In der Steuerungskonfiguration erscheinen unter „K-Bus[FIX]“ die eingefügten Busklemmen mit den dazugehörigen festen Adressen und die ggf. vorher eingestellten Variablennamen.

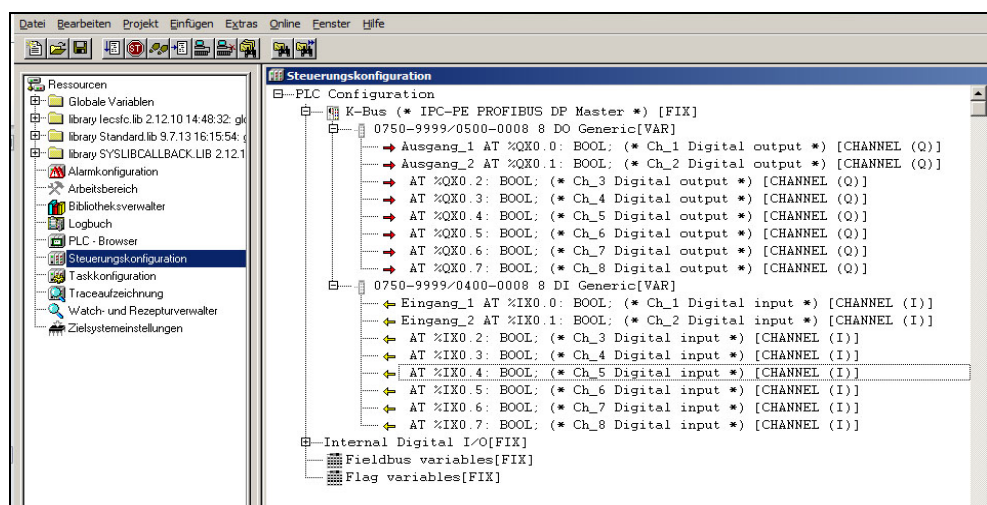


Abbildung 49: Steuerungskonfiguration: Busklemmen mit den dazugehörigen Adressen

11.6.4 Editieren des Programmbausteins

Zum Editieren des Programmbausteins PLC_PRG wechseln Sie auf der Registerkarte „Baustein“ und klicken Sie mit einem Doppelklick auf den Programmbaustein PLC_PRG.

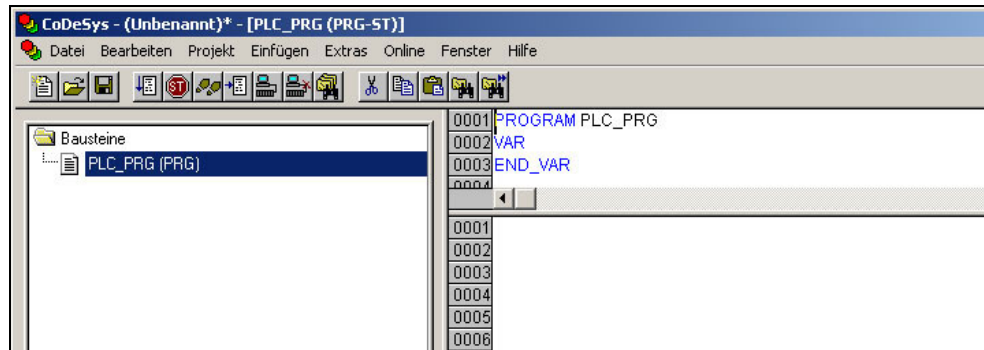


Abbildung 50: Programmbaustein

Folgendes Beispiel soll das Editieren des Programmbausteins verdeutlichen. Dazu wird ein Eingang einem Ausgang zugewiesen:

1. Drücken Sie **[F2]**, um die Eingabehilfe zu öffnen, oder Sie klicken auf die rechte Maustaste und wählen aus dem Kontextmenü „Eingabehilfe“.

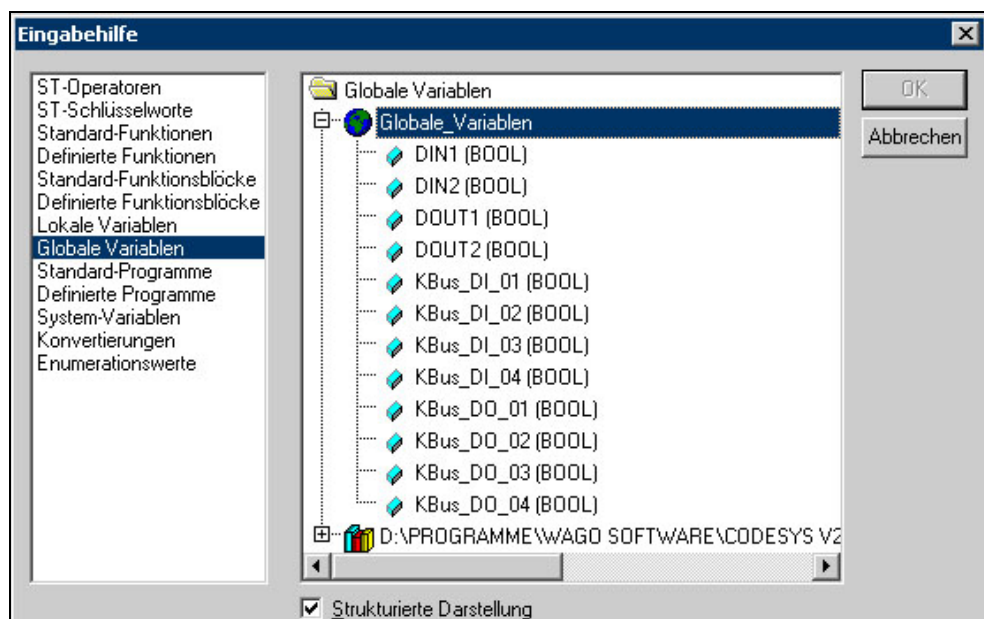


Abbildung 51: Eingabehilfe zur Auswahl der Variablen

2. Selektieren Sie unter „Globale Variablen“ die zuvor deklarierte Variable „K-Bus_Do_01“ und klicken Sie zum Einfügen dieser auf **[OK]**.
3. Geben Sie hinter dem Variablennamen die Zuweisung:= ein.
4. Wiederholen Sie Schritt 2 für die Variable „K-Bus_DI_01“.

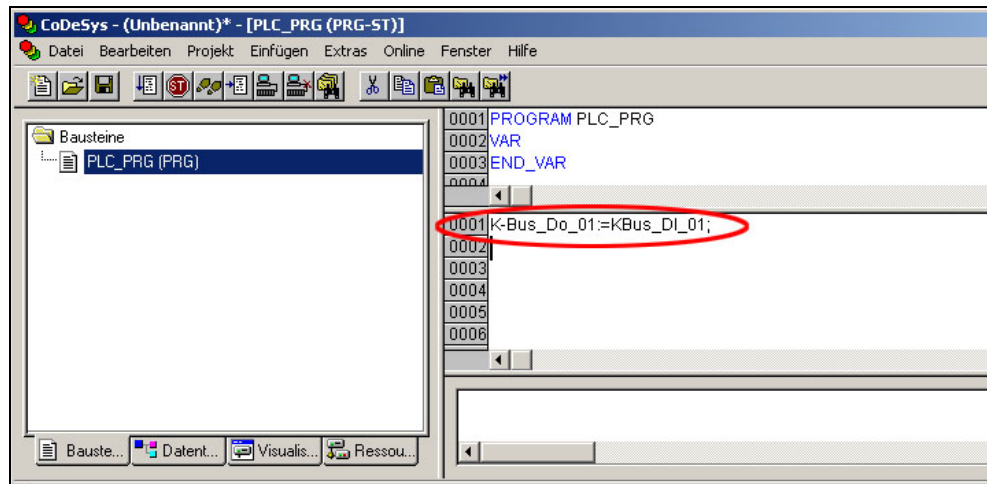


Abbildung 52: Beispiel einer Zuweisung

5. Zum Kompilieren klicken Sie in der Menüleiste auf **Projekt > Alles Übersetzen**.

11.6.5 SPS-Programm in den I/O-IPC laden und ausführen (ETHERNET)

Voraussetzung:

Die Simulation ist deaktiviert (**Online > Simulation**).

Sie haben den PC über ein ETHERNET-Kabel (RJ-45) mit der ETHERNET-Schnittstelle des I/O-IPC verbunden. Siehe dazu Kapitel „ETHERNET-Schnittstellen (X8, X9)“.

1. Klicken Sie in der Menüleiste auf **Online** und wählen Sie **Kommunikationsparameter ...**. Es öffnet sich das Fenster „Kommunikationsparameter“.
2. Zum Auswählen einer Kommunikationsverbindung klicken Sie im Fenster „Kommunikationsparameter“ auf **[Neu]**. Es öffnet sich das Fenster zum Anlegen einer Kommunikationsverbindung.

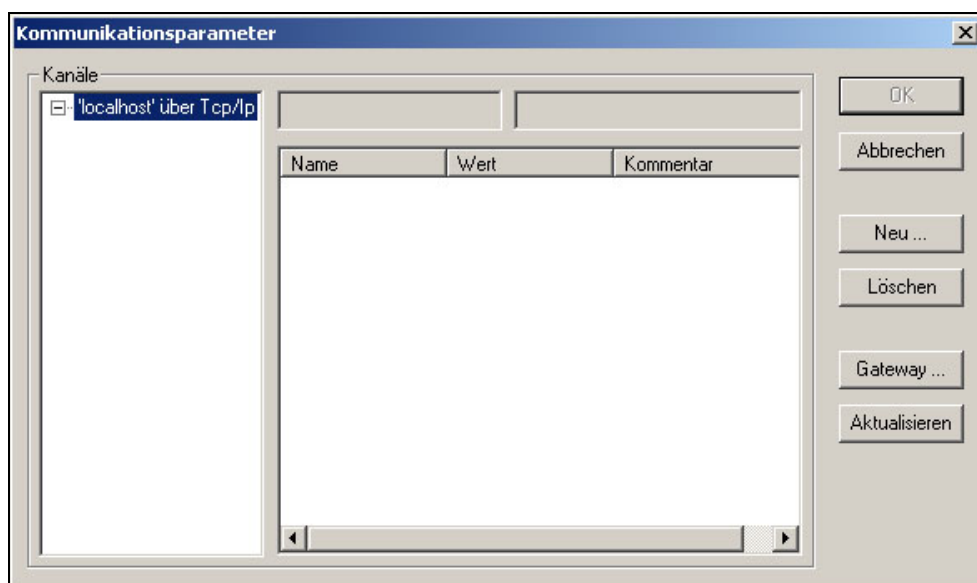


Abbildung 53: Anlegen einer Kommunikationsverbindung 1

3. Geben Sie im Feld „Name“ eine beliebige Bezeichnung für Ihren I/O-IPC ein und klicken Sie auf „Tcp/Ip...“. Klicken Sie anschließend auf **[OK]**.

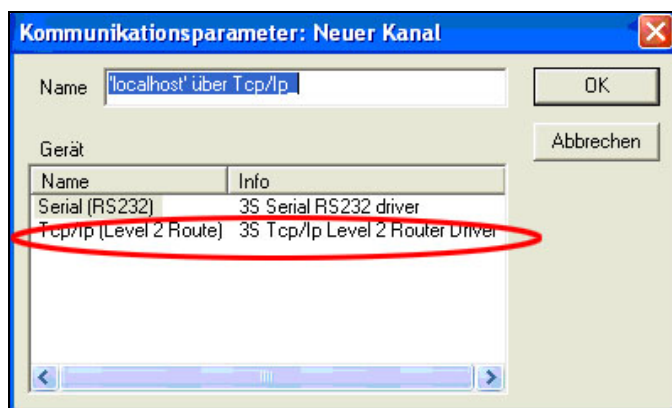


Abbildung 54: Anlegen einer Kommunikationsverbindung 2

4. Tragen Sie innerhalb des Fensters „Kommunikationsparameter“ im Feld „Address“ **die IP-Adresse Ihres I/O-IPC ein** und ändern Sie den Wert unter „Port“ auf **1200**.
Drücken Sie anschließend die Eingabetaste auf ihrer PC-Tastatur. Zum Schließen des Fensters klicken Sie in diesem auf **[OK]**.
Zum Auswählen eines bereits angelegten I/O-IPC selektieren Sie diesen im linken Fenster und klicken Sie anschließend auf **[OK]**.

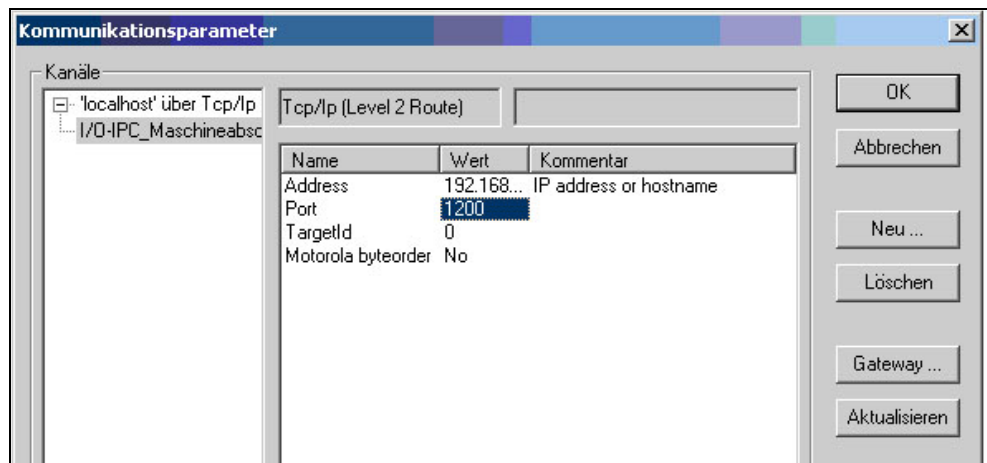


Abbildung 55: Anlegen einer Kommunikationsverbindung 3

5. Übertragen Sie das SPS-Programm, indem Sie in der Menüleiste auf **Online** klicken und **Einloggen** wählen.
6. Vergewissern Sie sich, dass sich der Run/Stop-Schalter des I/O-IPC in Position „Run“ befindet.
7. Starten Sie das SPS-Programm, indem Sie in der Menüleiste auf **Online > Start** klicken.

11.6.6 SPS-Programm in den I/O-IPC laden und ausführen (RS-232)

Voraussetzung:

Die Simulation ist deaktiviert (**Online > Simulation**).

Sie haben den PC über ein Nullmodemkabel mit der seriellen Schnittstelle des I/O-IPC verbunden. Siehe dazu Kapitel „Zugriff über RS-232-Schnittstelle und Terminalprogramm“.

1. Wählen Sie im WBM oder über das IPC-Configuration-Tool für die Schnittstelle RS-232 CODESYS aus. Siehe dazu Kapitel „Seite Administration“.
2. Klicken Sie in der Menüleiste auf **Online** und wählen Sie **Kommunikationsparameter ...**. Es öffnet sich das Fenster „Kommunikationsparameter“.
3. Zum Auswählen einer Kommunikationsverbindung klicken Sie im Fenster „Kommunikationsparameter“ auf **[Neu]**. Es öffnet sich das Fenster zum Anlegen einer Kommunikationsverbindung.

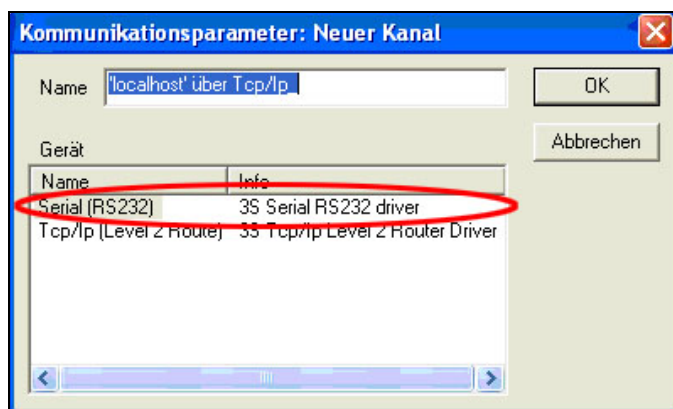


Abbildung 56: Anlegen einer Kommunikationsverbindung (RS-232) 1

4. Geben Sie im Feld „Name“ eine beliebige Bezeichnung für Ihren I/O-IPC ein und klicken Sie auf „Serial (RS232)“. Klicken Sie anschließend auf **[OK]**.
5. Klicken Sie im Fenster „Kommunikationsparameter“ auf **[Gateway]** und wählen Sie für die Kommunikationsverbindung den Parameter „lokal“ aus. Zum Schließen der Fenster klicken Sie auf **[OK]**.

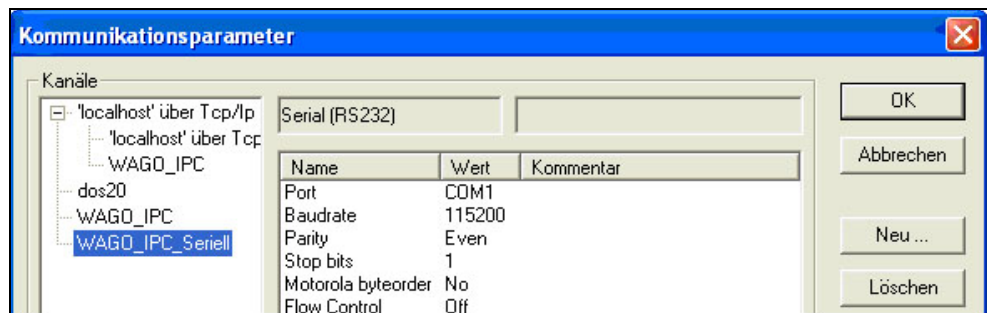


Abbildung 57: Anlegen einer Kommunikationsverbindung (RS-232) 2

6. Geben Sie folgende Kommunikationsparameter der RS-232-Schnittstelle ein:
 - Baudrate: 115200 bit/s
 - Parity: Even
 - Stop Bits: 1
 - Motorola Byteorder: No
 - Flow Control: Off
7. Übertragen Sie das SPS-Programm, indem Sie in der Menüleiste auf **Online** klicken und **Einloggen** wählen.
8. Vergewissern Sie sich, dass sich der Run/Stopp-Schalter des I/O-IPC in Position „Run“ befindet.
9. Starten Sie das SPS-Programm, indem Sie in der Menüleiste auf **Online > Start** klicken.

11.6.7 Boot-Projekt erzeugen

Damit nach einem Neustart des I/O-IPC das SPS-Programm wieder automatisch startet, erzeugen Sie ein Boot-Projekt. Wählen Sie dazu in der Menüleiste **Online** > **Bootprojekt erzeugen**. Beachten Sie, dass Sie noch in CODESYS angemeldet („eingeloggt“) sind.

Hinweis



Boot-Projekt automatisch laden

Darüber hinaus können Sie das Boot-Projekt automatisch beim Start des I/O-IPC laden. Klicken Sie auf die Registerkarte „Ressourcen“ und öffnen Sie die „Zielsystemeinstellungen“. Wählen Sie die Karteireiter „Allgemein“ aus und wählen „Bootprojekt automatisch laden“.

Wenn ein Bootprojekt (DEFAULT.PRG.und DEFAULT.CHK) unter */home/codesys* vorhanden ist und der Schalter „Run/Stop“ des I/O-IPC auf „Run“ steht, beginnt der I/O-IPC automatisch mit der Abarbeitung des SPS-Programms. Steht dieser auf „Stop“, wird das SPS-Programm nicht gestartet.

Wenn ein SPS-Programm im I/O-IPC läuft, startet ein SPS-Task mit dem Lesen der Feldbusdaten (nur bei I/O-IPCs mit Feldbusanschluss), der Daten der integrierten Ein- und Ausgänge und der Busklemmen. Die im SPS-Programm geänderten Ausgangsdaten werden nach Abarbeitung der SPS-Task aktualisiert. Ein Wechsel der Betriebsart („Stop/Run“) wird nur am Ende eines SPS-Tasks durchgeführt. Die Zykluszeit umfasst die Zeit vom Start des SPS-Programms bis zum nächsten Start. Wird eine größere Schleife innerhalb eines SPS-Programms programmiert, verlängert sich die Task-Zeit entsprechend. Die Eingänge und Ausgänge werden während der Abarbeitung nicht aktualisiert. Diese Aktualisierungen finden nur am Ende eines SPS-Tasks statt.

11.7 Anlegen von Task-Prioritäten

Mit der Task-Konfiguration stellen Sie das Zeitverhalten und die Priorität einzelner Programmbausteine ein.

Hinweis



Watchdog

In einem Anwenderprogramm ohne Task-Konfiguration gibt es keinen Watchdog, der die Zykluszeit des Anwenderprogramms (PLC_PRG) überwacht.

Legen Sie eine Task-Konfiguration folgendermaßen an:

1. Zum Öffnen der Task-Konfiguration klicken Sie mit einem Doppelklick auf „Taskkonfiguration“ in der Spalte „Ressource“.

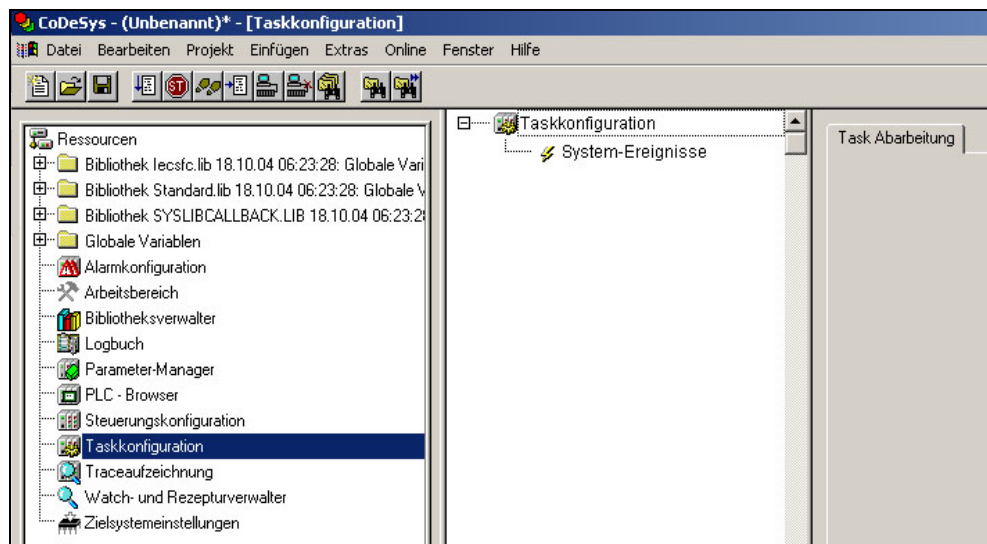


Abbildung 58: Task-Konfiguration

2. Zum Anlegen einer Task-Konfiguration klicken Sie mit der rechten Maustaste auf „Taskkonfiguration“ und wählen im Kontextmenü „Task anhängen“.

- Um dem Task einen neuen Namen zuzuweisen (z. B. PLC_Prog), klicken Sie auf „Neue Task“. Wählen Sie anschließend den Typ des Tasks aus. In diesem Beispiel ist dies der Typ „Zyklisch“.

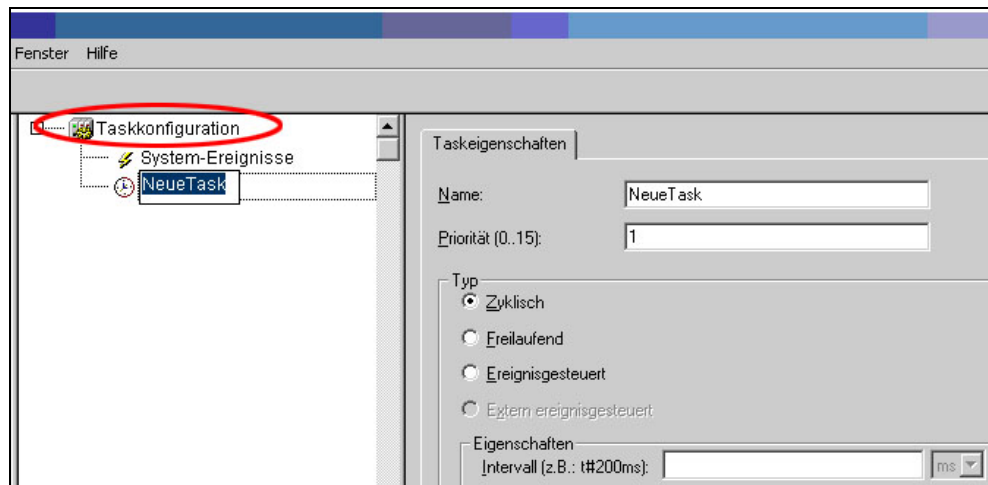


Abbildung 59: Task-Namen ändern 1

- Fügen Sie den zuvor erstellten Programmbaustein PLC_PRG ein (siehe Kapitel „Editieren des Programmbausteins“). Klicken Sie dazu mit der rechten Maustaste auf das „Uhr“-Symbol und wählen im Kontextmenü „Programmaufruf anhängen“. Anschließend klicken Sie auf die Schaltfläche [...] und auf [OK].

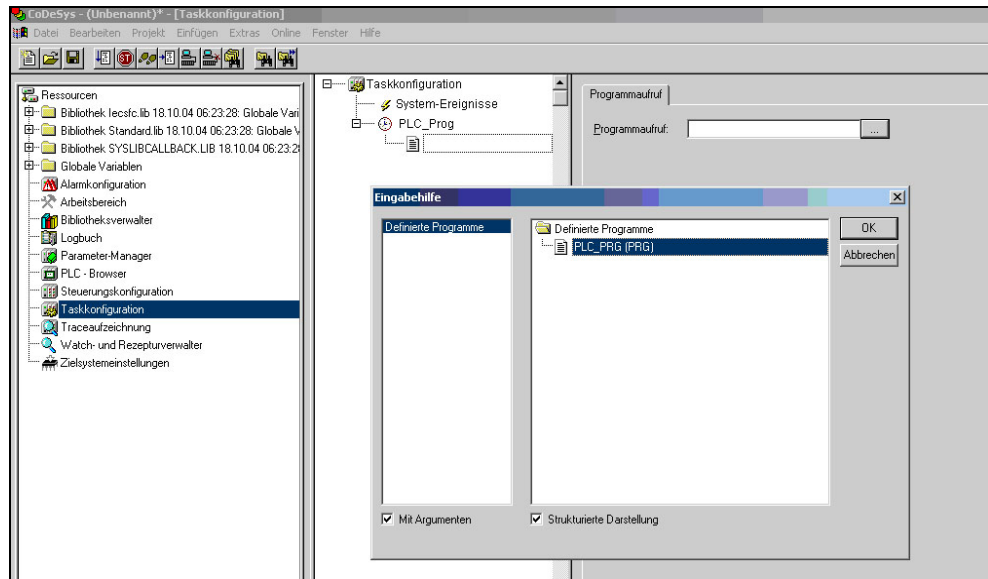


Abbildung 60: Aufruf zum Anhängen des Programmbausteins

- Kompilieren Sie das Beispielpogramm, indem Sie in der Menüleiste **Projekt > Übersetzen** wählen.

11.7.1 Zyklische Task-Prioritäten

Für jeden Task können Sie eine Priorität vergeben, um die Reihenfolge der Abarbeitung der Tasks festzulegen.

Alle Tasks, die auf das Prozessabbild des Klemmenbusses zugreifen, werden mit diesem synchronisiert. Das bedeutet, dass die Tasks mit dem Zugriff auf das Prozessabbild des Klemmenbusses solange warten, bis mindestens ein korrekt abgeschlossener Klemmenbuszyklus ausgeführt wurde.

Tritt am Klemmenbus ein Fehler auf (z. B. defekte Busklemme), werden die Tasks, die auf das Prozessabbild des Klemmenbusses zugreifen, nicht mehr ausgeführt. Die Tasks können erst wieder abgearbeitet werden, wenn ihnen neue Eingangsdaten zur Verfügung stehen.

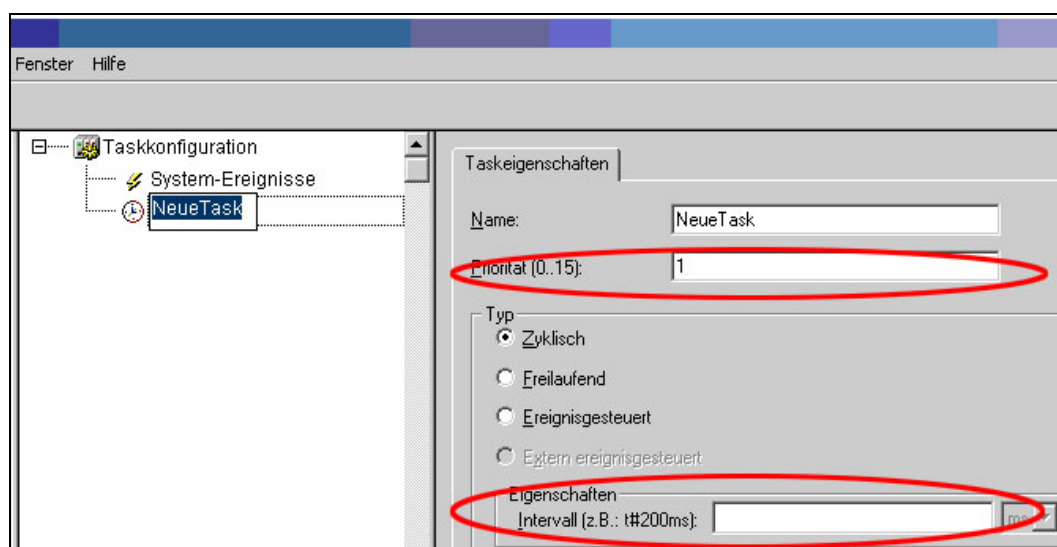


Abbildung 61: Task-Namen ändern 2

Hinweis



Reihenfolge der Task-Abarbeitung

Die unten stehenden Prioritäten geben nicht die Reihenfolgen der Task-Abarbeitung an. Die Tasks starten in beliebiger Reihenfolge.

Priorität 0 – 5:

Als Tasks mit den höchsten Prioritäten 0 – 5 sollten wichtige Rechenoperationen und hochsynchrone Zugriffe auf das Prozessabbild der Busklemmen ausgeführt werden. Die Tasks werden voll prioritätsgesteuert abgearbeitet und entsprechen den LinuxRT-Prioritäten -68 bis -63.

Priorität 6 – 10:

Als Tasks mit den mittleren Prioritäten 6 – 10 sollten Echtzeitzugriffe wie beispielsweise auf ETHERNET, Dateisystem, Feldbusdaten (nur bei I/O-IPCs mit Feldbusanschluss) und RS-232-Schnittstelle ausgeführt werden. Die Tasks werden voll prioritätsgesteuert abgearbeitet und entsprechen den LinuxRT-Prioritäten -50 bis -46.

Priorität 11 – 15:

Als Tasks mit den niedrigsten Prioritäten 11 – 15 sollten Anwendungen wie beispielsweise lang andauernde Rechenoperationen sowie nicht echtzeitrelevante Zugriffe auf Klemmenbus, ETHERNET, Dateisystem, Feldbusdaten (nur bei I/O-IPCs mit Feldbusanschluss) und RS-232-Schnittstelle ausgeführt werden. Die Programme innerhalb einer Priorität haben keinen Prioritätsunterschied, da jedem Task vom Betriebssystem die gleiche Rechenzeit zugewiesen wird („Completely Fair Scheduler“-Verfahren).

Hinweis



Freilaufende Tasks

Wenn Sie keine Task-Konfiguration vornehmen, wird das Programm PLC_PRG mit der niedrigsten Priorität zyklisch alle 10 ms ausgeführt. Die Laufzeit der „Freilaufenden Task“ wird nicht durch einen CODESYS-Watchdog überwacht.

11.7.2 Freilaufende Tasks

Bei Verwendung von „Freilaufenden Tasks“ ist das Eingabefeld „Priorität (0 ... 15)“ in der unten stehenden Abbildung ohne Funktion, da diese im Betriebssystem die niedrigste Priorität besitzen.

Bei Verwendung mehrerer „Freilaufender Tasks“ übernimmt das Betriebssystem deren Verwaltung und weist jedem die gleiche Rechenzeit zu, da sich „Freilaufende Tasks“ in der Priorität nicht unterscheiden.

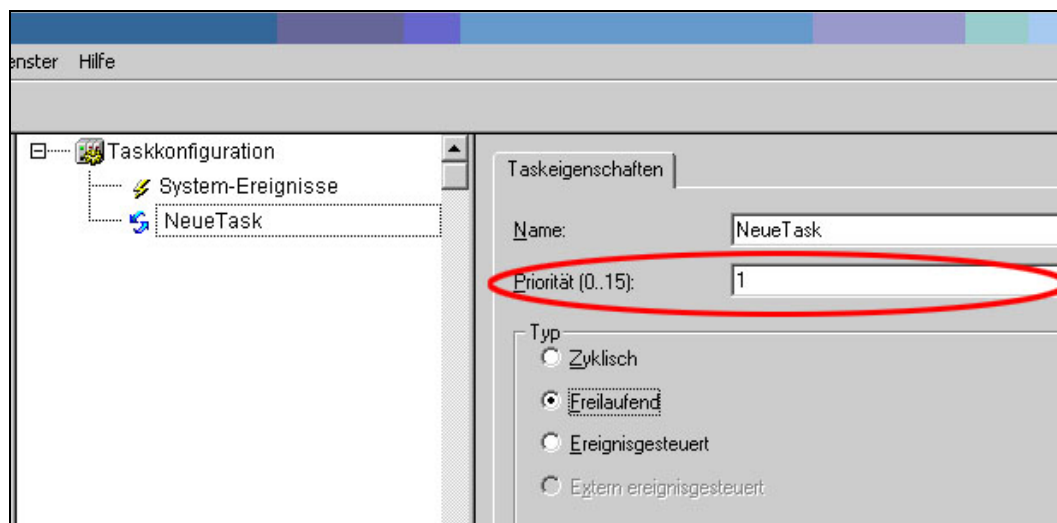


Abbildung 62: Freilaufende Tasks

11.8 Systemereignisse

In der CODESYS-Taskkonfiguration können neben zyklischen Tasks auch Event-Tasks verwendet werden. Diese Tasks werden bei bestimmten Ereignissen im Gerät aufgerufen.

Die Events können Sie im folgenden Dialog aktivieren und ein aufzurufendes Programm eintragen:

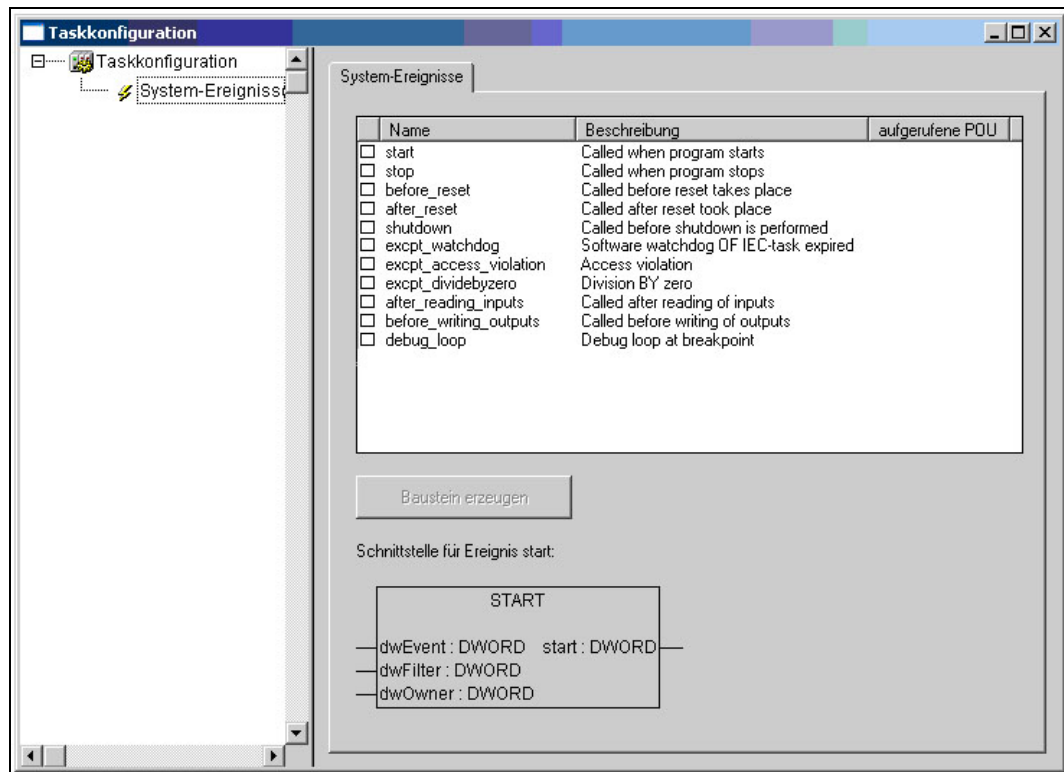


Abbildung 63: Systemereignisse

Die folgenden Events können aktiviert werden:

Tabelle 58: Events

Name	Beschreibung
start	Der Event wird unmittelbar nach dem Start des Anwenderprogramms aufgerufen.
stop	Der Event wird unmittelbar nach dem Stoppen des Anwenderprogramms aufgerufen.
before_reset	Der Event wird unmittelbar vor dem Rücksetzen des Anwenderprogramms aufgerufen.
after_reset	Der Event wird unmittelbar nach dem Rücksetzen des Anwenderprogramms aufgerufen.
shutdown	Der Event wird unmittelbar vor dem Runterfahren des Systems aufgerufen.
excpt_watchdog	Der Event wird aufgerufen, wenn ein Task-Watchdog erkannt wurde.
excpt_access_violation	Der Event wird aufgerufen, wenn ein Speicherzugriffsfehler auf einen ungültigen Speicherbereich erkannt wurde. (falscher Pointer, ungültiger Array-Index, ungültiger Dateideskriptor)
excpt_dividebyzero	Der Event wird aufgerufen, wenn eine Division durch Null erkannt wurde.
after_reading_inputs	Dieser Event wird unabhängig vom Anwenderprogramm nach dem Lesen aller Eingänge ausgelöst.
before_writing_outputs	Dieser Event wird vor dem Schreiben aller Ausgänge unabhängig vom Anwenderprogramm ausgelöst.
debug_loop	Dieser Event wird bei jedem Task-Aufruf ausgelöst, wenn in dieser Task ein Breakpoint erreicht wurde und dadurch die Abarbeitung dieser Task blockiert ist.

11.9 Klemmenbussynchronisation

Der Klemmenbuszyklus und der CODESYS-Task-Zyklus werden automatisch optimal synchronisiert: Abhängig von der Anzahl der gesteckten Busklemmen und dem schnellsten eingestellten CODESYS-Task-Zyklus des I/O-IPC. Dabei können die im Folgenden beschriebenen Synchronisierungsfälle auftreten.

Mit dem CODESYS-Task sind in diesem Kapitel nur Tasks innerhalb von CODESYS gemeint, die einen Zugriff auf den Klemmenbus enthalten. Tasks die nicht auf den Klemmenbus zugreifen, werden nicht wie im Folgenden beschrieben synchronisiert. Siehe dazu Kapitel „Anlegen von Task-Prioritäten“.

11.9.1 Fall 1: CODESYS-Task-Intervall kleiner als Klemmenbuszyklus eingestellt

Die Ausführung der CODESYS-Task wird mit der Zykluszeit des Klemmenbusses synchronisiert.

Der CODESYS-Task wird parallel zum Klemmenbuszyklus abgearbeitet. Das CODESYS-Task-Intervall wird auf die Klemmenbuszykluszeit verlängert. Das ist notwendig, damit jede CODESYS-Task mit neuen Eingangsdaten vom Klemmenbus startet und nach jeder CODESYS-Task die Ausgangswerte an den Klemmen auch gesetzt werden.

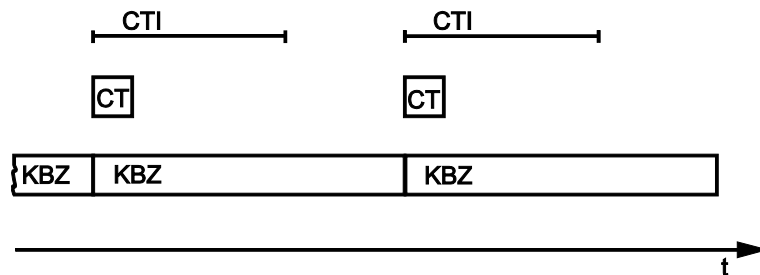


Abbildung 64: Klemmenbussynchronisation 01

CTI: CODESYS-Task-Intervall
 CT: CODESYS-Task, der auf die I/O-Klemmen des Klemmenbusses zugreift
 KBZ: Klemmenbuszyklus

Beispiel:

CODESYS-Task-Intervall (CTI): 100 μ s

Klemmenbuszyklus (KBZ): 350 μ s

Ergebnis: Anpassung des CODESYS-Task-Intervalls an den Klemmenbuszyklus: 350 μ s.

11.9.2 Fall 2: CODESYS-Task-Intervall kleiner als doppelter Klemmenbuszyklus

Die Ausführung des Klemmenbusses wird mit dem eingestellten CODESYS-Task-Intervall synchronisiert.

Am Ende des CODESYS-Tasks startet der Klemmenbuszyklus, der synchron zur schnellsten CODESYS-Task bearbeitet wird. So wird sichergestellt, dass bei Start jedes CODESYS-Tasks aktuelle Eingangsdaten vom Klemmenbus bereitstehen und die Ausgangswerte jedes CODESYS-Tasks an den Klemmen auch ausgegeben werden.

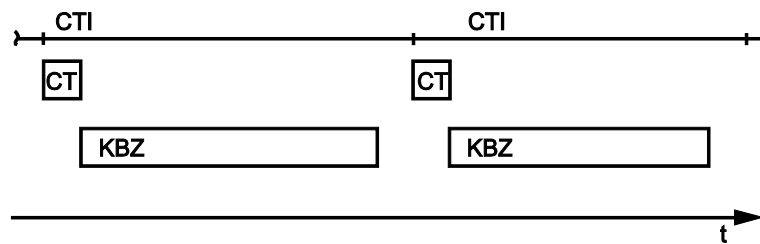


Abbildung 65: Klemmenbussynchronisation 02

CTI: CODESYS-Task-Intervall

CT: CODESYS-Task, der auf die I/O-Klemmen des Klemmenbusses zugreift

KBZ: Klemmenbuszyklus

Beispiel:

CODESYS-Task-Intervall (CTI): 500 μ s

Klemmenbuszyklus (KBZ): 350 μ s

Ergebnis: Ausführung des Klemmenbuszyklus alle 500 μ s.

11.9.3 Fall 3: CODESYS-Task-Intervall größer als doppelter Klemmenbuszyklus

Die IO-Daten des Klemmenbusses werden einmal vor dem CODESYS-Task und einmal nach dem CODESYS-Task aktualisiert.

Vor der Abarbeitung des CODESYS-Tasks wird der Klemmenbuszyklus ausgeführt, der die aktuellen Eingangsdaten für den CODESYS-Task zur Verfügung stellt. Nach Ausführung des CODESYS-Tasks wird ein weiterer Klemmenbuszyklus gestartet, der die Ausgangsdaten an den Klemmen zur Verfügung stellt.

So wird sichergestellt, dass bei Start jedes CODESYS-Tasks die aktuellen Eingangsdaten vom Klemmenbus bereitstehen und die Ausgangswerte jedes CODESYS-Tasks schnell an den Klemmen ausgegeben werden. Es wird dabei die Verarbeitung von Klemmenbuszyklen vermieden, die unnötig viel Rechenzeit der CPU verwenden würden.

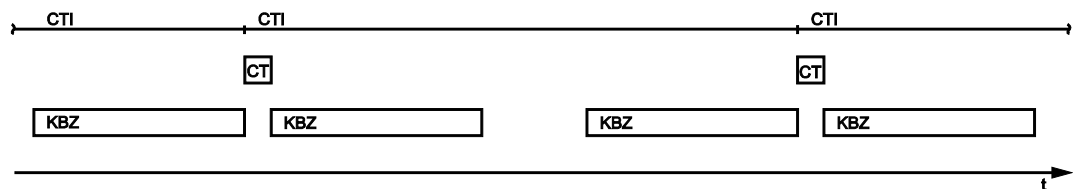


Abbildung 66: Klemmenbussynchronisation 03

CTI: CODESYS-Task-Intervall

CT: CODESYS-Task, der auf die I/O-Klemmen des Klemmenbusses zugreift

KBZ: Klemmenbuszyklus

Beispiel:

CODESYS-Task-Intervall (CTI): 2000 μ s

Klemmenbuszyklus (KBZ): 350 μ s

Ergebnis: Ausführung des Klemmenbuszyklus 350 μ s vor der CODESYS-Task und einmal direkt nach der CODESYS-Task.

11.9.4 Fall 4: CODESYS-Task-Intervall größer als 10 ms

Die Synchronisierung erfolgt wie im Fall 3, jedoch würden die Ausgangsklemmen nach 150 ms ohne Klemmenbuszyklus in ihren Default-Zustand zurückgesetzt. Dieses wird dadurch vermieden, dass nach mindestens 10 ms auf jeden Fall ein Klemmenbuszyklus ausgeführt wird.

Die IO-Daten des Klemmenbusses werden einmal vor dem CODESYS-Task und einmal nach dem CODESYS-Task aktualisiert und zusätzlich wird alle 10 ms ein weiterer Klemmenbuszyklus ausgeführt.

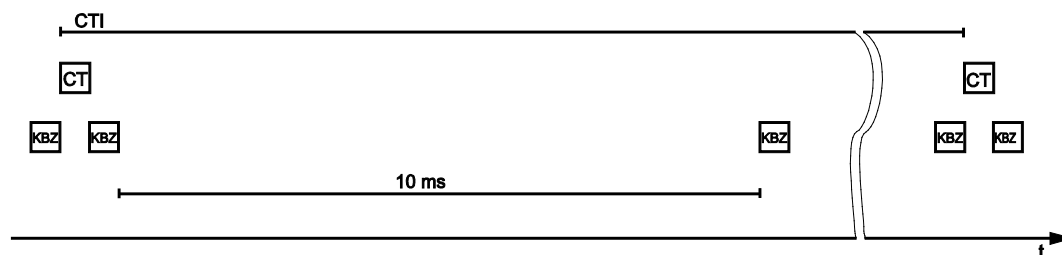


Abbildung 67: Klemmenbussynchronisation 04

CTI: CODESYS-Task-Intervall
 CT: CODESYS-Task, der auf die I/O-Klemmen des Klemmenbusses zugreift
 KBZ: Klemmenbuszyklus

Beispiel:

CODESYS-Task-Intervall (CTI): 150000 μ s

Klemmenbuszyklus (KBZ): 350 μ s

Ergebnis: Ausführung des Klemmenbuszyklus 350 μ s vor der CODESYS-Task, einmal direkt nach der CODESYS-Task und 10 ms nach dem letzten Klemmenbuszyklus.

11.10 CODESYS-Visualisierung

Die CODESYS-Web-Visualisierung basiert auf der Java-Technologie. Alle Java-Programme benötigen eine Java-Laufzeitumgebung (JRE), die auf dem Host-PC zusammen mit einem Internet-Browser installiert sein muss. Ein Applet wird im Dateisystem eines Webservers abgelegt und über eine HTML-Einstiegsseite für Internet-Browser zugänglich gemacht.

Alle Visualisierungsvarianten (HMI, Web-Visualisierung und Target-Visualisierung) erstellen Sie mit dem grafischen Editor von CODESYS. Über das Fenster „Zielsystemeinstellung“ wählen Sie die Visualisierungsvarianten aus. Aus den Informationen wird für jede dieser Seiten eine Beschreibungsdatei im XML-Format erzeugt. Sie finden diese Dateien im Installationspfad von CODESYS im Unterordner „*visu*“. Dort liegen auch die HTML-Startseite „webvisu.htm“, das Java-Archiv „webvisu.jar“ in dem das Applet (webvisu.class) komprimiert gespeichert ist.

Nach dem Erstellen einer Visualisierung sind zu deren Ausführung noch nachfolgende Schritte notwendig:

1. Klicken Sie auf die Registerkarte „Ressourcen“ und öffnen Sie die „Zielsystemeinstellungen“. Wählen Sie aus, ob Sie sich die Visualisierung als „Web-Visualisierung“ über einen Internet-Browser und/oder als „Target-Visualisierung“ über einen an der DVI-I-Schnittstelle angeschlossenen Monitor anzeigen lassen wollen.

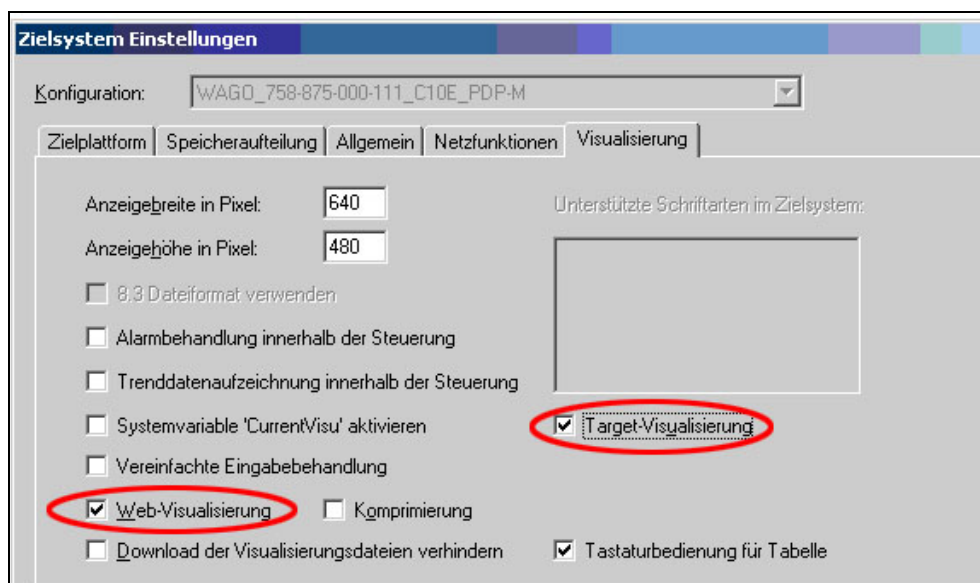


Abbildung 68: Auswahl der Visualisierungsvariante in der Zielsystemeinstellung

- Erzeugen Sie eine Startseite für die Visualisierung. Klicken Sie in der Registerkarte „Visualisierung“ mit der rechten Maustaste auf den Ordner „Visualisierung“. Wählen Sie im Kontextmenü **Objekt einfügen ...** Es öffnet sich der Dialog „Neue Visualisierung“.

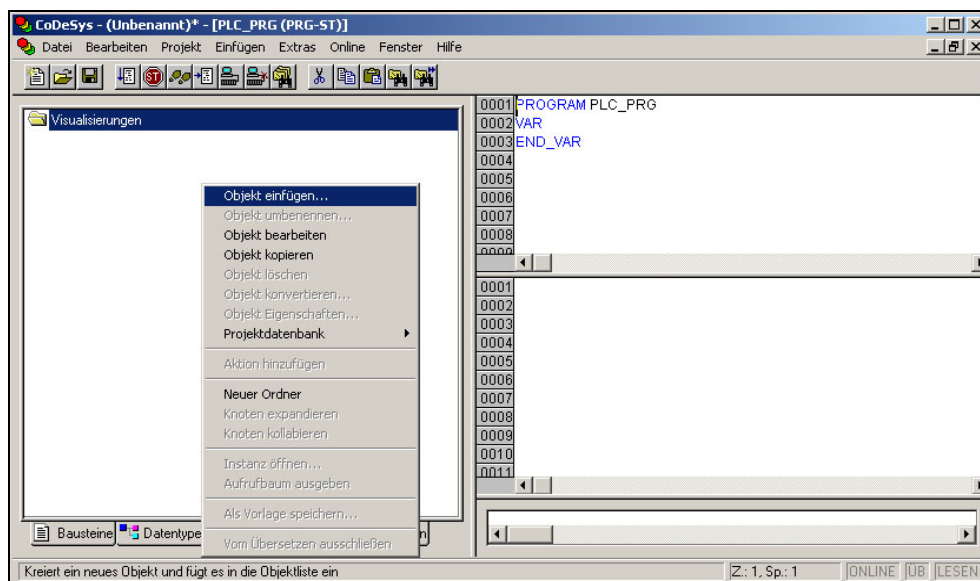


Abbildung 69: Erzeugern der Startvisualisierung PLC_VISU

- Geben Sie im Dialog „Neue Visualisierung“ für die Startvisualisierung den Namen **PLC_VISU** ein. Beim Systemstart erscheint dann diese Seite als Startseite.

Wenn Sie das SPS-Programm in den I/O-IPC übertragen (**Online > Einloggen**) und gestartet haben (**Online > Start**), wird die Target-Visualisierung automatisch angezeigt. Zur Anzeige der Web-Visualisierung geben Sie in die Adresszeile des verwendeten Internet-Browsers folgende URL ein:

http://<IP-Adresse des I/O-IPC>:8080/webvisu.htm.

Ferner können Sie sich auch über das WBM die Web-Visualisierung anzeigen lassen (siehe Kapitel „Seite `WebVisu`“).

Hinweis



WAGO-Startbildschirm bei Target-Visualisierung

Bei der Target-Visualisierung wird der WAGO-Startbildschirm nur angezeigt, wenn kein CODESYS-Projekt geladen ist (ab FW09).

Hinweis



Weitere Informationen

Weitere Informationen (FAQ) zur CODESYS-Web-Visualisierung erhalten Sie im Kapitel „Häufig gestellte Fragen zur CODESYS-Web-Visualisierung“ und in der Online-Hilfe von CODESYS 2.3.

11.10.1 Einbinden von Schriften

Für die CODESYS-Target-Visualisierung stehen im Auslieferungszustand die TrueType-Schriftarten Arial und Courier zur Verfügung.

Zusätzlich können Sie auch beliebige TrueType-Schriften (*.ttf) einbinden. Dabei sind ggf. die Lizenzbedingungen der verwendeten Schriften zu berücksichtigen. Freie Schriftarten stehen u. a. kostenlos im Internet bereit:

[ftp://microwindows.censoft.com/
pub/microwindows/microwindows-fonts-0.90.tar.gz](ftp://microwindows.censoft.com/pub/microwindows/microwindows-fonts-0.90.tar.gz)

Für die Verwendung dieser Fonts sind nachfolgende Schritte vorzunehmen:

- Die Grafik-Bibliothek der CODESYS-Target-Visualisierung greift auf ein Verzeichnis im Dateisystem des WAGO-I/O-IPC zurück, in welches Sie diese Schriftarten ablegen müssen. Dieses Verzeichnis ist von Ihnen erstellen. Der einfachste Weg hierzu ist über einen FTP-Zugang vom PC aus (siehe Kapitel „FTP-Server (pure-ftpd)“)
- Nach dem Anmelden befindet Sie sich im Ordner */home* des I/O-IPC-Dateisystems. Kopieren Sie die Fonts in das Verzeichnis */home/codesys/fontz/truetype*.
- Die Extension der gewünschten Schriftart muss immer ***.ttf** lauten.
- Der heruntergeladene Font-Name auf dem I/O-IPC muss immer in direktem Zusammenhang mit dem angezeigten Namen in der Windows-Auswahlbox hängen (z. B. Times New Roman.ttf). Dabei sind auch hier wieder die Groß- und Kleinschreibungen, aber auch Leerzeichen zu beachten. Andernfalls wird stattdessen automatisch eine Ersatzschriftart verwendet.

Tabelle 59: Namenskonvention für Fonts (Beispiel)

CODESYS-Auswahlbox	Font-Name für Target-Visualisierung
Arial	Arial.ttf Arialb.ttf (Bold) Ariali.ttf (Italic) Arialz.ttf (Bold Italic)
Courier	Courier.ttf Courierb.ttf (Bold) Courieri.ttf (Italic) Courierz.ttf (Bold Italic)

11.10.2 Grenzen der CODESYS-Visualisierung

Der I/O-IPC unterstützt die in CODESYS integrierten Visualisierungsvarianten „TargetVisu“ und „WebVisu“. Abhängig von der Variante ergeben sich technologische Einschränkungen.

Hinweis



Darstellung eines ActiveX-Elementes in der „Target-Visu“

Die Darstellung eines ActiveX-Elementes in der „Target-Visu“ ist nicht möglich.

Verschiedene Optionen der komplexen Visualisierungsobjekte „Alarm“ und „Trend“ sind ausschließlich in der „HMI“ verfügbar. Dieses gilt z. B. für das Versenden von E-Mails als Reaktion auf einen Alarm oder für die Navigation

Die Web-Visualisierung auf dem I/O-IPC wird im Vergleich zur „HMI“ in wesentlich engeren physikalischen Grenzen ausgeführt. Kann die „HMI“ auf die nahezu unbeschränkten Ressourcen eines Desktop-PC zurückgreifen, ist beim Einsatz der Web-Visualisierung auf folgende Einschränkungen zu achten:

Anpassung an das Dateisystem

Die Gesamtgröße von SPS-Programm, Visualisierungsdateien, Bitmaps, Log-Dateien, Konfigurationsdateien usw. muss in das Dateisystem passen.

Der Prozessdatenspeicher

Die Web-Visualisierung verwendet ein eigenes Protokoll für den Austausch von Prozessdaten zwischen Applet und Steuerung.

Der I/O-IPC überträgt die Prozessdaten ASCII-codiert. Als Trennzeichen zwischen zwei Prozesswerten dient das Pipe-Zeichen („|“). Damit ist der Platzbedarf einer Prozessdatenvariablen im Prozessdatenspeicher nicht nur abhängig vom Datentyp, sondern zusätzlich vom Prozesswert selbst. So belegt eine Variable vom Type „WORD“ zwischen einem Byte für die Werte 0 bis 9 und fünf Bytes für Werte ab 10000. Das gewählte Format (ASCII + |) erlaubt lediglich eine grobe Abschätzung des Platzbedarfes für die einzelnen Prozessdaten im Prozessdatenpuffer. Wird die Größe der ASCII-codierten Prozessdaten überschritten, arbeitet die Web-Visualisierung nicht mehr erwartungsgemäß.

Gleichzeitige Verbindungen

Das Laufzeitsystem unterstützt maximal 97 (Anzahl der CODESYS-Tasks im PLC-Programm) gleichzeitige TCP/IP Verbindungen auf Port 1200.

Die Rechnerleistung/Prozessorzeit

Der I/O-IPC basiert auf einem Echtzeitbetriebssystem. Dabei unterbrechen oder verdrängen hochpriorige Prozesse, wie zum Beispiel das SPS-Programm, niederpriorige Prozesse. Der Webserver, der für die Web-Visualisierung zuständig ist, zählt zu einem solch niederpriorigen Prozess.

Hinweis



Prozessorzeit

Achten Sie bei der Task-Konfiguration darauf, dass für alle Prozesse genügend Prozessorzeit zur Verfügung steht

Die Netzwerkbelastung

Die CPU des I/O-IPC ist sowohl für die Abarbeitung des SPS-Programms als auch für die Abwicklung des Netzwerkverkehrs zuständig. Die ETHERNET-Kommunikation verlangt, dass jedes empfangene Telegramm, unabhängig davon ob es für den I/O-IPC bestimmt ist oder nicht, bearbeitet wird.

Eine deutliche Reduzierung der Netzwerkbelastung ist durch die Verwendung eines Switches statt eines Hubs erreichbar.

Gegen Broadcast-Telegramme ist jedoch keine Maßnahme auf dem I/O-IPC vorhanden. Diese lassen sich nur beim Sender eindämmen oder mit konfigurierbaren Switches eindämmen, die über eine Broadcast-Limitierung verfügen. Ein Netzwerkmonitor wie z. B. „wireshark“ (www.wireshark.com) verschafft einen Überblick über die aktuelle Auslastung in ihrem Netzwerk.

11.10.3 Beseitigung von Störungen der CODESYS-Web-Visualisierung

Treten bei der Verwendung mit der CODESYS-Web-Visualisierung Probleme auf, versuchen Sie bitte zuerst mittels der nachfolgenden Tabelle eine Lösung zu finden. Lassen sich die Probleme nicht beheben, kontaktieren Sie bitte den WAGO-Support.

Tabelle 60: Fehler und deren Abhilfe

Fehler	Abhilfe
Internet Explorer meldet „APPLET NOT INITIATED“	Schließen Sie alle Fenster des Internet Explorers und starten Sie ihn erneut. Sollte der Fehler weiterhin auftreten, deutet dies auf eine fehlende oder zerstörte Datei hin. Überprüfen Sie mittels FTP, ob das Java-Archiv „webvisu.jar“ vollständig im Ordner „/PLC“ des I/O-IPC vorhanden ist. Die Originaldatei finden Sie im Installationspfad von CODESYS (üblich unter <i>C:\Programme\WAGO Software\CODESYS V2.3\Visu\webvisu.jar</i>). Ersetzen Sie gegebenenfalls die beschädigte Datei mittels FTP oder erzwingen Sie in CODESYS mit Alles bereinigen > Alles übersetzen > Einloggen den Download aller Dateien.
Web-Visualisierung wird nicht angezeigt	Haben Sie die JRE installiert? Prüfen Sie die Einstellungen der Firewall, z. B. ob der Port 8080 freigegeben ist.
Web-Visualisierung „friert“ ein. Web-Visualisierung bleibt nach längerer Zeit stehen.	Die Aufrufintervalle in der Task-Konfiguration sind zu klein gewählt. Dadurch bekommt der Webserver des I/O-IPC, der mit einer niedrigen Priorität ausgeführt wird, nicht genügend oder keine Rechenzeit. Sollte keine (explizite) Task-Konfiguration angelegt worden sein, wird (implizit) das PLC_PRG als „Freilaufender Task“ mit der Prio 1 ausgeführt. Dies lässt dem Webserver zu wenig Rechenzeit. Legen Sie bei Verwendung der Web-Visualisierung immer eine Task-Konfiguration an. Dabei sollte das Aufrufintervall die dreifache mittlere Ausführungszeit nicht unterschreiten. Achten Sie bei der Ermittlung der Ausführungszeit darauf, dass das SPS-Programm „eingeschwungen“ ist.
Web-Visualisierung lässt sich nicht in den I/O-IPC laden	Möglicherweise passen nicht alle Dateien in das Dateisystem des I/O-IPC. Löschen Sie nicht benötigte Daten (z. B. mittels FTP).
Bitmap wird nicht angezeigt	Enthält der Name einer Bilddatei Umlaute, so kann der Webserver diesen Bildnamen nicht interpretieren.
Java-Konsole meldet: „Class not found“	Die JRE findet im Java-Archiv „WebVisu.jar“ nicht den Einsprungspunkt für die Klasse „webvisu.class“. Vermutlich ist das Java-Archiv unvollständig. Löschen Sie die „WebVisu.jar“ aus dem Java-Cache und oder deaktivieren Sie den Cache. In diesem Fall wird das Archiv (Applet) neu vom I/O-IPC angefordert. Sollte das Problem weiter bestehen, laden Sie das Projekt erneut in den I/O-IPC.
Web-Visualisierung wird statisch angezeigt, alle Prozesswerte zeigen „0“	Ursache ist, dass die Prozessdatenkommunikation fehlschlägt. Wird die Web-Visualisierung über einen Proxy-Server betrieben, so ist neben dem eigentlichen HTTP-Proxy für den Prozessdatenaustausch zusätzlich ein SOCKS-Proxy erforderlich.

11.10.4 Häufig gestellte Fragen zur CODESYS-Web-Visualisierung

Wie lässt sich das Applet für spezielle Bildschirmauflösungen optimieren?

Um die Web-Visualisierung für einen PDA oder eines Touchpanels mit fester Auflösung zu optimieren, empfiehlt sich folgendes Vorgehen:

Geben Sie in den „Zielsystemeinstellungen“ auf der Registerkarte „Visualisierung“ die Höhe und Breite des Zeichenbereiches in „Pixel“ an. Bei der Erstellung von Visualisierung wird dann der später sichtbare Bereich grau hinterlegt. Die tatsächliche Größe des Zeichenbereiches der Web-Visualisierung wird jedoch durch die Attribute „Height“ und „Width“ des Tags HTML-APPLET in der Datei „webvisu.htm definiert. Passen Sie auch diese Parameter an die vorliegende Auflösung an.

Welche Java-Ausführungsumgebung sollte ich verwenden?

Empfohlen wird die Verwendung der Java2-Standard-Edition in der Version 1.5.0 (J2SE1.5.0_06) oder höher. Diese ist kostenfrei unter www.sun.com verfügbar. Getestet wurde auch Microsofts MSJVM3810. Des Weiteren stehen für PDAs Laufzeitumgebungen anderer Hersteller zur Verfügung (JamaicaVM, CrEme, ...). Zu beachten ist, dass sich diese Lösungen bei der Web-Visualisierung in Bezug auf den Leistungsumfang (z. B. Stabilität) anders verhalten können, als die oben genannten.

Sollte der Java-Cache verwendet werden?

Hier gibt es kein Ja oder Nein. Nach einer Standardinstallation ist der Cache aktiviert. Bei aktiviertem Cache legt das JRE verwendete Applets und Java-Archive in diesem ab. Für den zweiten Aufruf der Web-Visualisierung verkürzt sich dessen Startzeit deutlich, da das ca. 250 kB große Applet nicht erneut über das Netzwerk geladen werden muss, sondern schon im Cache bereitliegt. Dies ist besonders bei langsamen Netzwerkverbindungen interessant.

Hinweis:

Durch Netzwerkstörungen kann es vorkommen, dass die Java-Archive nicht vollständig in den Cache übertragen werden. In diesem Fall ist der Cache manuell zu leeren oder zu deaktivieren.

Warum kann das Visualisierungselement „TREND“ in der Web-Visualisierung nur „Online“ arbeiten?

Für die Visualisierungsprojekte sind folgende Einstellungen zu wählen:

Registerkarte **Ressourcen > Zielsystemeinstellungen**.

Aktivieren Sie „Web-Visualisierung“ und „Trenddatenaufzeichnung innerhalb der Steuerung“. Andernfalls werden die Trenddaten auf der Festplatte des CODESYS-Entwicklungsrechners gespeichert. Dies macht eine permanente Verbindung zwischen I/O-IPC und dem CODESYS-Gateway erforderlich. Eine Unterbrechung dieser Verbindung kann zu unvorhersehbaren Verhalten des I/O-IPC führen.

Im Konfigurationsdialog TREND kann zwischen den Betriebsarten „Online“ und „Historie“ gewählt werden. Der I/O-IPC unterstützt für Visualisierungsprojekte nur die Betriebsart „Online“, da es keine Möglichkeit gibt, die maximale Größe (Quota) der Trenddateien (*.trd) zu konfigurieren. Ein unkontrolliertes Anwachsen der Trenddateien kann zu einem unvorhersehbaren Verhalten des I/O-IPC führen.

In den häufigsten Fällen ist die Verwendung des Visualisierungselement „HISTOGRAM“ die bessere Wahl, da hier die volle Kontrolle über Zeitpunkt und Anzahl der Messungen und damit dem benötigtem Speicherplatz besteht.

Was ist bei der Verwendung des Visualisierungselements „ALARMTABELLE“ in der Web-Visualisierung zu beachten?

Der Status dieser Visualisierungskomponente wird am besten mit „Add-On“ beschrieben, womit eine kostenlose Zugabe gemeint ist, für die keinerlei Garantien gewährt werden.

Für die Visualisierungsprojekte sind folgende Einstellungen zu wählen:

Registerkarte **Ressourcen > Zielsystemeinstellungen**.

Aktivieren Sie „Web-Visualisierung“ (Haken setzen) und „Alarmbehandlung innerhalb der Steuerung“. Andernfalls werden die Alarmdaten auf dem CODESYS-Entwicklungsrechner bearbeitet. Dies macht eine permanente Verbindung zwischen I/O-IPC und dem CODESYS-Gateway erforderlich. Eine Unterbrechung dieser Verbindung kann zu unvorhersehbaren Verhalten des I/O-IPC führen.

12 CANopen-Master in CODESYS 2.3

12.1 Steuerungskonfiguration des CANopen-I/O-IPC

Hinweis



Aufruf der Adressen oder symbolischen Namen der Ein- und Ausgänge
Rufen Sie Adressen oder symbolische Namen der Ein- und Ausgänge explizit auf, da sonst das Prozessabbild nicht aktualisiert wird. Alternativ können Sie auch an den Speicheradressen IB%4800 oder QB%4800 ein Array von max. 240 Byte anlegen. Dieses Array ist im SPS-Programm aufzurufen.

Bevor eine Applikation auf das angeschlossene CAN-Netzwerk Zugriff hat, müssen Sie dieses in CODESYS konfigurieren:

1. Zum Einfügen des CANopen-I/O-IPC in die Steuerungskonfiguration klicken Sie mit der rechten Maustaste auf „PLC-Configuration“ und wählen Sie „CanMaster anhängen“.

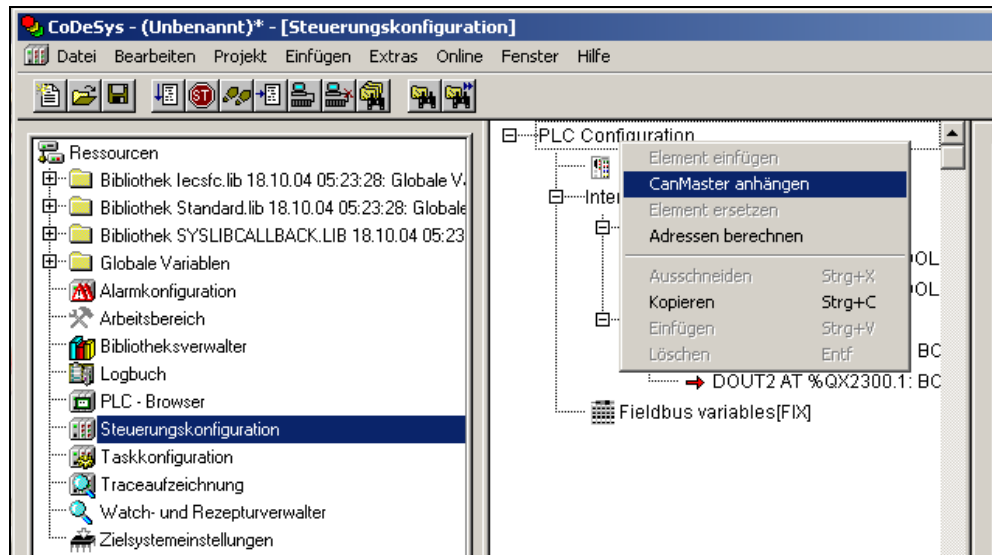


Abbildung 70: Anhängen des CANopen-Masters

2. Aktivieren Sie im Karteireiter „CAN Parameter“ das Kontrollfeld „DSP301, V4.01 ...“ zur Unterstützung modulare Slaves und einiger zusätzlichen Erweiterungen bezüglich der Normen DSP301 V3.01 und DSP306.

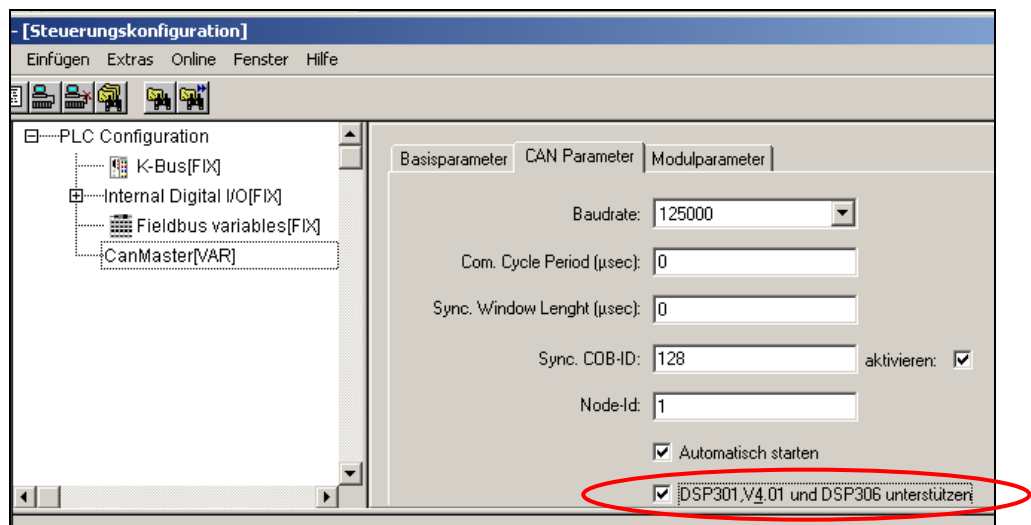


Abbildung 71: DSP301 und DSP306 aktivieren

3. Zum Auswählen eines (oder mehrerer) CANopen-Slaves klicken Sie mit der rechten Maustaste auf den I/O-IPC (CANMaster[VAR]) und wählen Sie „Unterelemente anhängen“. In diesem Beispiel wurde der 750-337 als Slave gewählt.

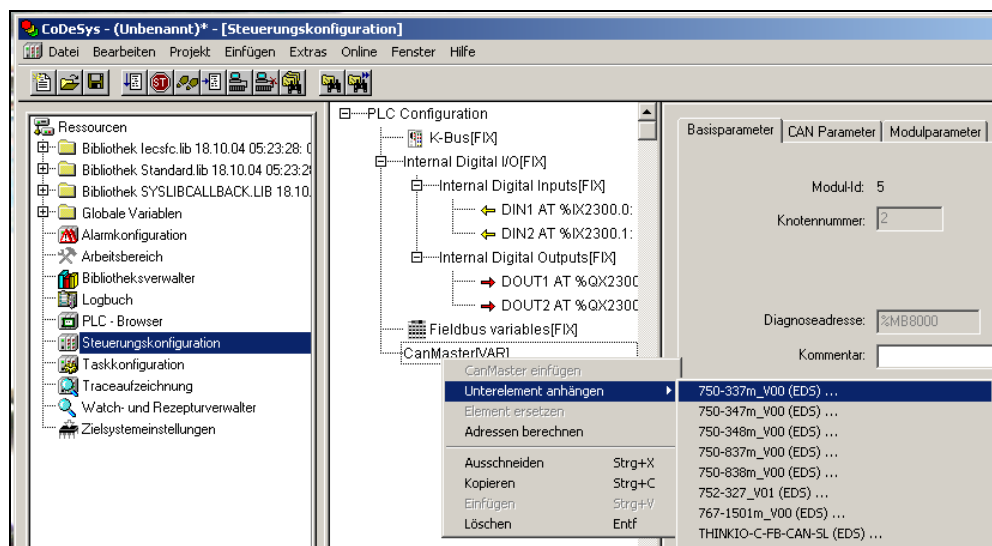


Abbildung 72: Anhängen der CANopen-Slaves

Hinweis



EDS-Dateien

Die EDS-Dateien aktueller Komponenten des WAGO-I/O-SYSTEMs sind in den Target-Files für den WAGO-I/O-IPC integriert. Für die Anbindung WAGO-fremder Geräte benötigen Sie die zugehörigen EDS-Dateien. Klicken Sie dazu in der Menüleiste auf **Extras > Konfigurationsdatei hinzufügen**.

4. Klicken Sie auf den Karteireiter „CAN-Modulwahl“. Übernehmen Sie die Topologie der am Slave angeschlossenen Busklemmen (von der Endklemme zum Koppler) in die Steuerkonfiguration. Fügen Sie dazu über **[Hinzufügen]** die entsprechenden Busklemmen in das rechte Fenster („Ausgewählte Module“) ein.
5. Mittels **[Entfernen]** löschen Sie ggf. falsch eingefügte Busklemmen aus dem rechten Fenster heraus. Dazu muss die Busklemme markiert sein.

Im folgenden Beispiel sind eine Digitaleingangsklemme und eine Digitalausgangsklemme am Slave angeschlossen.

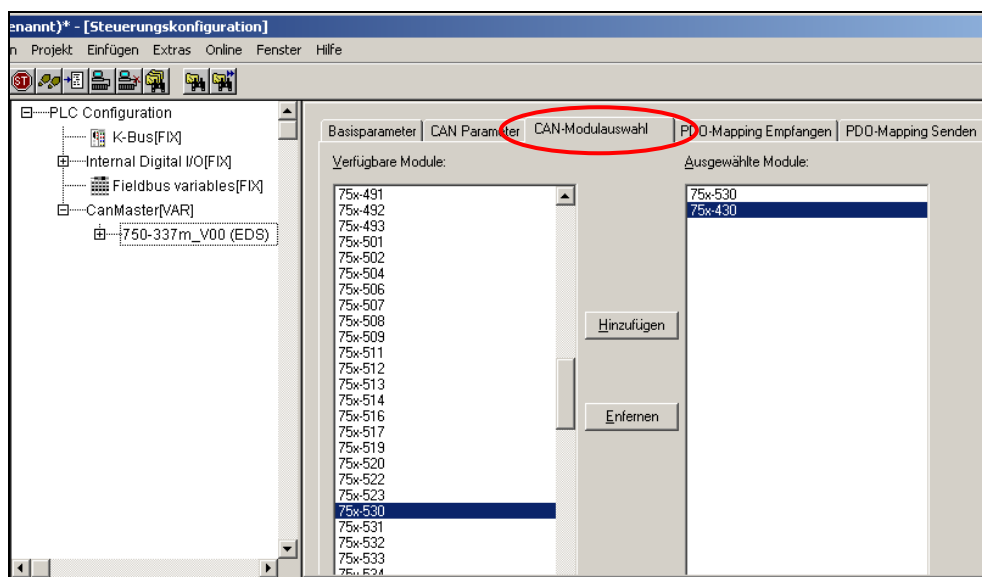


Abbildung 73: Busklemmen auswählen

Hinweis



Busklemmen, die keine Prozessdaten liefern

Busklemmen, die keine Prozessdaten liefern (z. B. Einspeiseklemme, Endklemme) werden bei der Konfiguration nicht berücksichtigt und erscheinen demnach auch nicht in der Auswahlliste der EDS-Datei.

6. In Abhängigkeit der für den CANopen-Slave ausgewählten Busklemmen erscheinen im Karteireiter „PDO-Mapping Empfangen“ und „PDO-Mapping Senden“ die entsprechenden CANopen-Objekte. Über die Registerkarten verändern Sie das in der EDS-Datei beschriebene „Default-Mapping“.

7. Klicken Sie auf den Karteireiter „CAN Parameter“, um die Stationsadresse des Slaves anzupassen. Tragen Sie im Feld „Node ID“ die Stationsadresse ein, die der des Slaves entspricht. Stimmen die beiden Stationsadressen nicht überein, ist das Herstellen der Kommunikationsverbindung nicht möglich.
8. Tragen Sie den Gerätetyp in das Feld „Device Type“ ein. Dieser ist abhängig vom Typ (digital/analog) der gesteckten Busklemmen am Slave. Für das verwendete Geräteprofil DS401 ergeben sich 15 verschiedene Gerätetypen, die bei Änderungen an der Slave-Topologie anzupassen sind:

1	0x10191	Digitale Eingänge
2	0x20191	Digitale Ausgänge
3	0x30191	Digitale Ein- und Ausgänge
4	0x40191	Analoge Eingänge
5	0x50191	Digitale Eingänge und analoge Eingänge
6	0x60191	Digitale Ausgänge und analoge Eingänge
7	0x70191	Digitale Ein- und Ausgänge und analoge Eingänge
8	0x80191	Analoge Ausgänge
9	0x90191	Digitale Eingänge und analoge Ausgänge
10	0xA0191	Digitale Ausgänge und analoge Ausgänge
11	0xB0191	Digitale Ein- und Ausgänge und analoge Ausgänge
12	0xC0191	Analoge Ein- und Ausgänge
13	0xD0191	Digitale Eingänge und analoge Ein- und Ausgänge
14	0xE0191	Digitale Ausgänge und analoge Ein- und Ausgänge
15	0xF0191	Digitale Ein- und Ausgänge und analoge Ein- und Ausgänge

Hinweis



Weitere Informationen

Weitere Informationen zum Gerätetyp (Objekt 0x1000) erhalten Sie in den Handbüchern der CANopen-Feldbuskoppler.

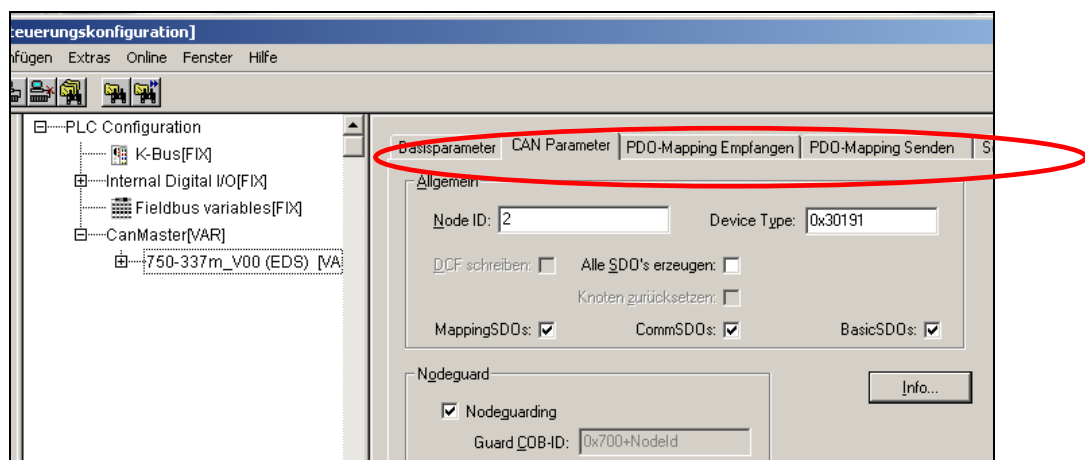


Abbildung 74: Node-ID einstellen

12.2 Einstellmöglichkeiten der Steuerungskonfiguration

12.2.1 CANopen-Master (I/O-IPC)

Über die Karteireiter für den I/O-IPC definieren Sie die globalen Einstellungen und Überwachungsparameter für den CANopen-Master.

Basisparameter

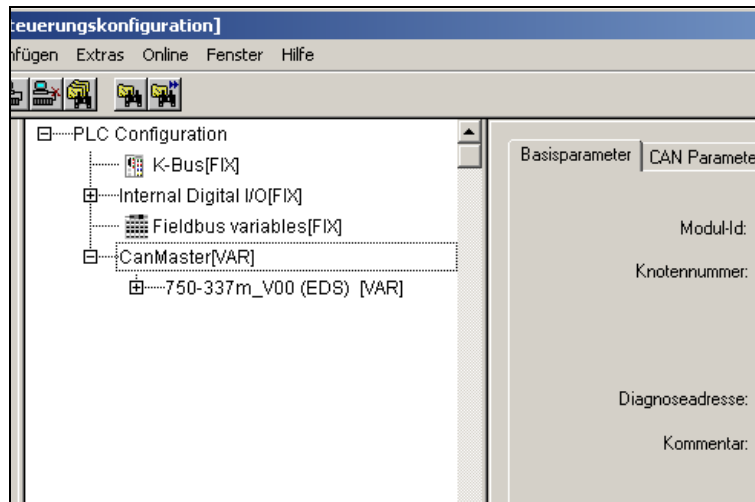


Abbildung 75: Karteireiter „Basisparameter“

Die nachfolgende Tabelle erläutert die auf dem Karteireiter aufgeführten Parameter:

Tabelle 61: Beschreibung der Basisparameter

Basisparameter	
Modul-ID	Parameter, die das Laufzeitsystem CODESYS nutzt.
Knotennummer	
Diagnoseadresse	
Kommentar	Eingabefeld für einen Kommentar.

CAN-Parameter

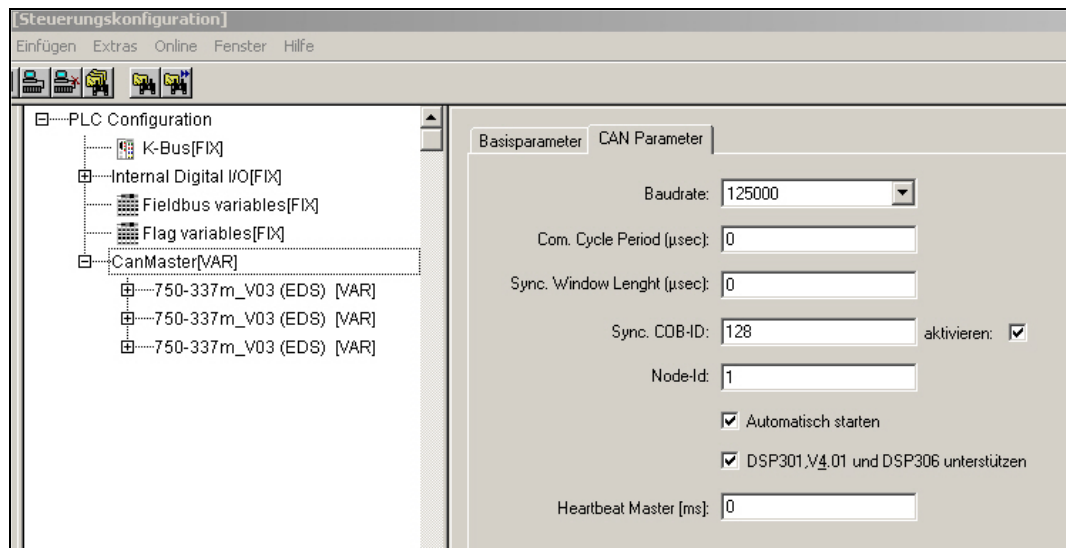


Abbildung 76: Karteireiter „CAN Parameter“

Die nachfolgende Tabelle erläutert die auf dem Karteireiter aufgeführten Parameter:

Tabelle 62: Beschreibung der CAN-Parameter

CAN Parameter	
Baudrate	Hier wählen Sie gewünschte Baudrate aus, die für die Übertragung im CAN-Bus gelten soll (Voreinstellung: 125000 Baud).
Com. Cycle Period (µsec)	Hier geben Sie das Zeitintervall (in µsec) ein, in dem die Synchronisationsnachricht vom I/O-IPC versendet wird. Kleinstes Zeitintervall: 1000 µsec
Sync. Window Length (µsec)	Zurzeit nicht implementiert.
Sync. COB-ID	Hier aktivieren oder deaktivieren Sie das Senden von Synchronisationsnachrichten des I/O-IPC. Diese Funktion ist für den Slave nur bei Verwendung des Übertragungstyps „azyklisch-synchron“ aktiviert. Voreinstellung: COB-ID 128 (0x80). Siehe dazu Tabelle „Beschreibung der Parameter“.
Node-ID	Stationsadresse (Node-ID) des I/O-IPC am CAN-Bus.
Automatisch starten	Wenn Sie dieses Kontrollfeld aktivieren, baut der I/O-IPC die Kommunikationsverbindung zu den CANopen-Slaves auf (über Broadcast-Nachricht „Start remoteNode“). Wenn Sie dieses Kontrollfeld deaktivieren, wird die Broadcast-Nachricht nicht gesendet. Über den Modulparameter „EnableNMTStartNode“ (siehe „Modulparameter“) können Sie CANopen-Slaves einzeln starten.
DSP301 ... unterstützen	Wenn Sie dieses Kontrollfeld aktivieren, unterstützt die Steuerungskonfiguration modulare CAN-Slaves sowie einige zusätzliche Erweiterungen bezüglich der Normen DSP301 V3.01 und DSP306.
Heartbeat Master (ms)	Ist die Option „Heartbeat Erzeugung“ aktiviert, sendet das CAN Device in den bei „Heartbeat Producer Time“ angegebenen ms-Abständen Heartbeats aus. Die Heartbeat Verbrauch ist derzeit nicht implementiert.

12.2.2 CANopen-Slaves

Über die im Folgenden beschriebenen Karteireiter definieren Sie das Verhalten von CANopen-Slaves.

Basisparameter

Abbildung 77: CANopen-Slaves Karteireiter „Basisparameter“

Die nachfolgende Tabelle erläutert die auf dem Karteireiter aufgeführten Parameter, die automatisch durch die Zielsystemeinstellungen (siehe Kapitel „Anlegen eines Projekts und Auswahl des Zielsystems“) vergeben werden:

Tabelle 63: Beschreibung der Basisparameter

Basisparameter	
Modul-ID:	Kennung des Slaves.
Knotennummer:	In der CODESYS-Laufzeitumgebung verwendete Knotennummer des Slaves.
Eingabeadresse:	Startadresse der Eingangsdaten: Der Adressraum beginnt immer bei %IB 4800 und wird automatisch vergeben. Sie können die Startadresse auch erhöhen.
Ausgabeadresse:	Startadresse der Ausgangsdaten: Der Adressraum beginnt immer bei %QB 4800 und wird automatisch vergeben. Sie können die Startadresse auch erhöhen.
Diagnoseadresse:	Speicherbereich für interne Diagnosebearbeitung.
Kommentar:	Eingabefeld für einen Kommentar.

CAN-Parameter

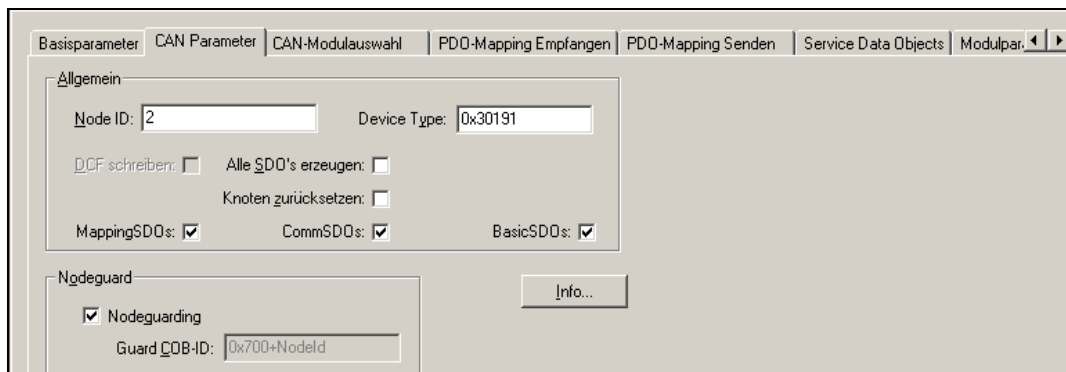


Abbildung 78: Karteireiter „CAN Parameter“ 1

Die nachfolgende Tabelle erläutert die auf dem Karteireiter aufgeführten Parameter:

Tabelle 64: Beschreibung der CAN-Parameter

CAN Parameter	
Allgemein	
Note ID:	Die Node-ID (1-126) ist die Stationsadresse, unter welcher der I/O-IPC mit dem Slave im CAN-Netzwerk kommuniziert.
Device Type:	Tragen Sie hier den Gerätetyp ein. Dieser ist abhängig vom Typ (digital/analog) der gesteckten Busklemmen am Slave. Für das Geräteprofil DS401 ergeben sich 15 verschiedene Gerätetyp-Werte, die bei Änderungen an der Slave-Topologie anzupassen sind (Objekt 0x1000). Siehe dazu Kapitel „Steuerungskonfiguration des CANopen-I/O-IPC“.
DCF schreiben	Zurzeit nicht implementiert.
Alle SDO's erzeugen:	Bei aktiviertem Kontrollfeld werden für alle Objekte aus der EDS-Datei SDOs erzeugt. Zusätzlich sind die entsprechenden Optionen zu aktivieren. Sollen z.B. die Nodeguarding-Objekte geschrieben werden, ist auch das Kontrollfeld bei der Option „Nodeguarding“ zu aktivieren. Ist das Kontrollfeld deaktiviert, werden nur für die Objekte SDO erzeugt, bei denen die Defaultwerte aus der EDS-Datei abweichen.
Knoten zurücksetzen	Wenn Sie diese Option aktivieren, wird der Slave durch den I/O-IPC zurückgesetzt (erhält einen „Reset-Node“), bevor die Konfiguration an den Slave gesendet wird. Diese Funktion ist zurzeit nicht implementiert.
MappingSDOs:	Hier aktivieren oder deaktivieren Sie jeden der drei SDO-Bereiche der Slave-Konfiguration.
CommSDOs:	MappingSDOs: Objekte 0x1600 – 1620 Objekte 1A00 – 1A20
BasicSDOs:	CommSDOs: Objekte 0x1400 – 1420 Objekte 1800 – 1820 BasicsSDOs: Objekte 0x100C – 1017

The screenshot shows the 'CAN Parameters' configuration dialog box. It is organized into four main sections, each with a title bar and a set of controls:

- Node guard:** Contains a checked checkbox for 'Nodeguarding'. Below it are three input fields: 'Guard COB-ID' with the value '0x700+NodeId', 'Guard time (ms)' with the value '0', and 'Life time factor' with the value '0'.
- Heartbeat settings:** Contains two checked checkboxes: 'Activate heartbeat generation' and 'Activate heartbeat consumer'. Between them is an input field for 'Heartbeat producer time' with the value '0' and the unit 'ms'.
- Emergency telegram:** Contains a checked checkbox for 'Emergency' and an input field for 'COB-ID' with the value '\$NodeID+0x80'.
- Communication Cycle:** Contains an unchecked checkbox for 'Cycle'.

An 'Info...' button is located in the top right corner of the dialog.

Abbildung 79: Karteireiter „CAN Parameter“ 2

Tabelle 65: Beschreibung der Parameter

CAN Parameter	
Nodeguard	
Nodeguarding: Guard Time (ms): Life Time Factor:	<p>Bei aktiviertem Nodeguarding überwacht der Slave den I/O-IPC auf eine mögliche Unterbrechung der Feldbuskommunikation. Unter „Guard Time“ geben Sie das Intervall an, indem der I/O-IPC die „Confirmation“ des Slaves erwartet.</p> <p>Im Feld „Life Time Factor“ (≥ 3) stellen Sie den Multiplikationswert für die „Guard Time“ ein. Ist die sich aus „Guard Time“ x „Life Time Factor“ ergebende Zeit („Node Life Time“) abgelaufen, wird der WAGO-Slave in den vordefinierten Zustand gebracht.</p>
Heartbeat-Einstellungen	
Ist die Option „Heartbeat Erzeugung“ aktiviert, sendet das CAN Device in den bei „Heartbeat Producer Time“ angegebenen ms-Abständen Heartbeats aus. Die Heartbeat Verbrauch ist derzeit nicht implementiert.	
Emergency-Telegramm	
Emergency	<p>Wenn Sie dieses Kontrollfeld aktivieren, sendet der Slave Fehler- und Statusmeldungen, die als Emergency-Nachrichten an die Diagnoseadresse im Merkerbereich hinterlegt werden. Mittels der „BusDiag.lib“ lesen Sie diese Fehler- und Statusmeldungen aus.</p> <p>Wenn Sie dieses Kontrollfeld deaktivieren, wird das SDO 0x1014 nicht zum Slave übertragen. Die Defaulteinstellung des Slave ist damit weiterhin gültig.</p>
COB-ID	Voreingestellt: Node-ID + 0x80
Info ...	
„FileInfo“ und „DeviceInfo“ der EDS-Datei	
Communication Cycle	
Zurzeit nicht implementiert.	

PDO-Mapping empfangen/senden

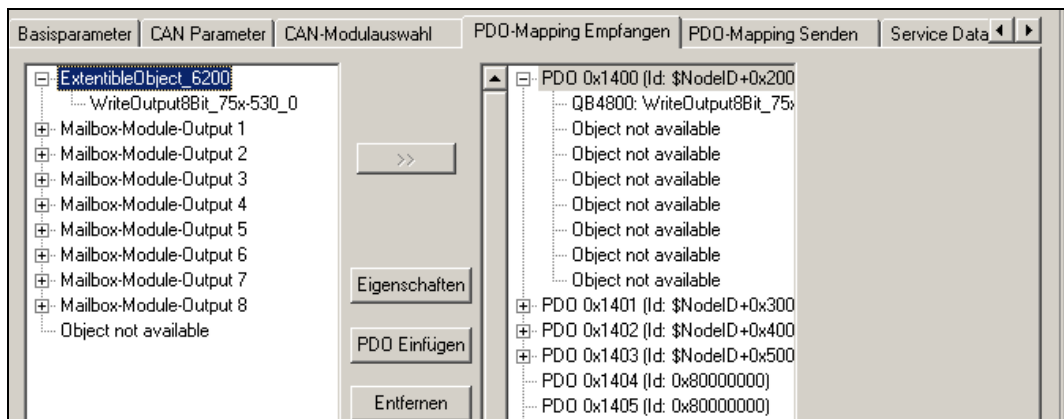


Abbildung 80: Karteireiter „PDO-Mapping empfangen“

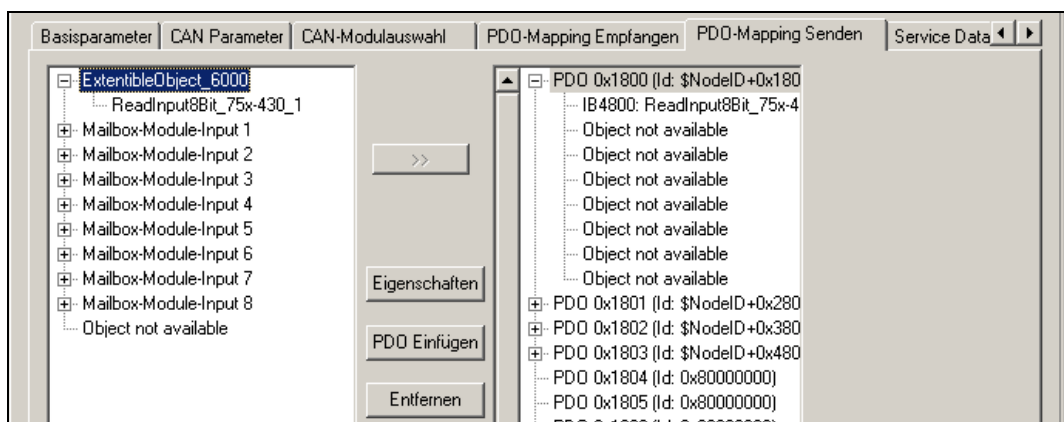


Abbildung 81: Karteireiter „PDO-Mapping senden“

Die nachfolgende Tabelle erläutert die auf dem Karteireiter aufgeführten Parameter:

Tabelle 66: Beschreibung zum PDO-Mapping empfangen und senden

PDO-Mapping empfangen	
	In Abhängigkeit der für den CANopen-Slave ausgewählten Busklemmen erscheinen im Karteireiter „PDO-Mapping empfangen“ (I/O-IPC → Slave) und „PDO-Mapping senden“ (Slave → I/O-IPC) die entsprechenden CANopen-Objekte. Über diese Registerkarten verändern Sie das in der EDS-Datei beschriebene „Default-Mapping“.
PDO einfügen	Über die Schaltfläche [PDO Einfügen] passen Sie die PDO an die Busklemmentopologie an. Es öffnet sich das PDO-Eigenschaften-Fenster, über das Sie dem PDO mit bestimmten Eigenschaften versehen. Weitere Informationen dazu erhalten Sie unter „Eigenschaften“.
	Um einen der 32 PDOs ein Objekt aus dem linken Fenster zuzuordnen, markieren Sie sowohl das entsprechende Objekt als auch das entsprechende PDO und drücken Sie dann auf [>>]. Daraufhin wird das Objekt unterhalb des PDOs im rechten Fenster eingehängt. Die ersten 64 digitalen und die ersten 12 analogen Ein- und Ausgänge werden automatisch den PDOs 1 – 4 zugeordnet.
Entfernen	Über die Schaltfläche [Entfernen] löschen Sie den augenblicklich im rechten Fenster markierten Eintrag aus der Konfiguration.
Eigenschaften	Es öffnet sich das Dialogfenster zu den PDO-Eigenschaften

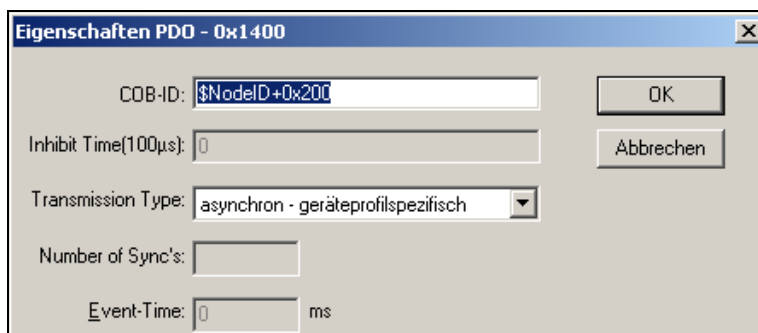


Abbildung 82: PDO-Eigenschaften-Fenster

Die nachfolgende Tabelle erläutert die im PDO-Eigenschaften-Fenster aufgeführten Parameter:

Tabelle 67: Beschreibung des PDO-Eigenschaften-Fensters

PDO-Mapping Empfangen	
COB-ID:	CAN-Identifizier
Inhibit Time (100 µs):	Hier stellen Sie zur Reduzierung der Kommunikationsaufkommen die Zeitspanne eines PDO ein, die vergehen muss, bis dieses erneut gesendet werden kann.
Transmission Type:	<p>Hierüber wählen Sie den Übertragungsmodus für das PDO aus:</p> <p>azyklisch-synchron: (Übertragungstyp 0) Das PDO wird synchron, aber nicht periodisch übertragen. Für Empfang-PDOs werden die Übertragungstypen 0 – 240 gleich behandelt.</p> <p>zyklisch-synchron: (Übertragungstyp 1 – 240) Das PDO wird synchron übertragen, wobei „Number of Syncs“ die Anzahl der Synchronisationsnachrichten angibt, die zwischen zwei Übertragungen des PDOs liegen. Für Empfang-PDOs werden die Übertragungstypen 0 – 240 gleich behandelt (nicht implementiert).</p> <p>synchron - nur RTR: (Übertragungstyp 252) Das PDO wird nach einer Synchronisationsnachricht aktualisiert, aber nicht versendet. Übertragen wird es nur auf eine explizite Anfrage „Remote Transmission Request“ (nicht implementiert).</p> <p>asynchron - nur RTR: (Übertragungstyp 253) Das PDO wird nur auf eine explizite Anfrage „Remote Transmission Request“ aktualisiert und übertragen (nicht implementiert).</p> <p>asynchron-herstellerspezifisch: (Übertragungstyp 254) Das PDO wird nur nach bestimmten Ereignissen übertragen.</p> <p>asynchron-geräteprofilsspezifisch: (Übertragungstyp 255) Das PDO wird nur nach bestimmten Ereignissen übertragen.</p>

Tabelle 67: Beschreibung des PDO-Eigenschaften-Fensters

PDO-Mapping Empfangen	
Number of Syncs	Abhängig vom „Transmission Type“ ist dieses Feld editierbar zur Eingabe der Anzahl der Synchronisationsnachrichten 1 – 240
Event-Time	Event-Time: Abhängig vom „Transmission Type“ geben Sie hier die Zeitspanne (in ms) an, die zwischen zwei Übertragungen des PDOs liegen soll.

Service Data Object

Index	Name	Wert	Typ	Default
4300sub2	Mailbox		Octed S...	
4300sub3	Process Data 1		Unsigne...	
4300sub4	Process Data 2		Unsigne...	
4300sub5	Process Data 3		Unsigne...	
4300sub6	Process Data 4		Unsigne...	
4300sub7	Process Data 5		Unsigne...	
4300sub8	Process Data 6		Unsigne...	
4300sub9	Process Data 7		Unsigne...	
4300suba	Process Data 8		Unsigne...	
4300subb	Process Data 9		Unsigne...	
4300subc	Process Data 10		Unsigne...	
4300subd	Process Data 11		Unsigne...	
4300sube	Process Data 12		Unsigne...	
4300subf	Process Data 13		Unsigne...	
4300su...	Process Data 14		Unsigne...	
4300su...	Process Data 15		Unsigne...	
4300su...	Process Data 16		Unsigne...	
4300su...	Process Data 17		Unsigne...	
4300su...	Process Data 18		Unsigne...	
4300su...	Process Data 19		Unsigne...	
4300su...	Process Data 20		Unsigne...	
4300su...	Process Data 21		Unsigne...	
4300su...	Process Data 22		Unsigne...	
4300su...	Process Data 23		Unsigne...	
4300su...	Process Data 24		Unsigne...	

Abbildung 83: Karteireiter „Service Data Objects“

Hier werden alle Objekte der EDS-Datei aufgelistet, die im Bereich von Index 0x2000 bis 0x9FFF liegen und beschreibbar sind.

Zu jedem Objekt sind Index, Name, Wert, Typ und Default angegeben.

Der Wert der Objekte kann verändert werden. Markieren Sie dazu das entsprechende Feld in der Spalte „Wert“ und überschreiben Sie diesen mit Ihrer Eingabe und drücken Sie anschließend die **[Enter-Taste]**. Bei der Initialisierung des CAN-Busses werden die eingestellten Werte in Form von SDO an die Slaves übertragen.

Modulparameter

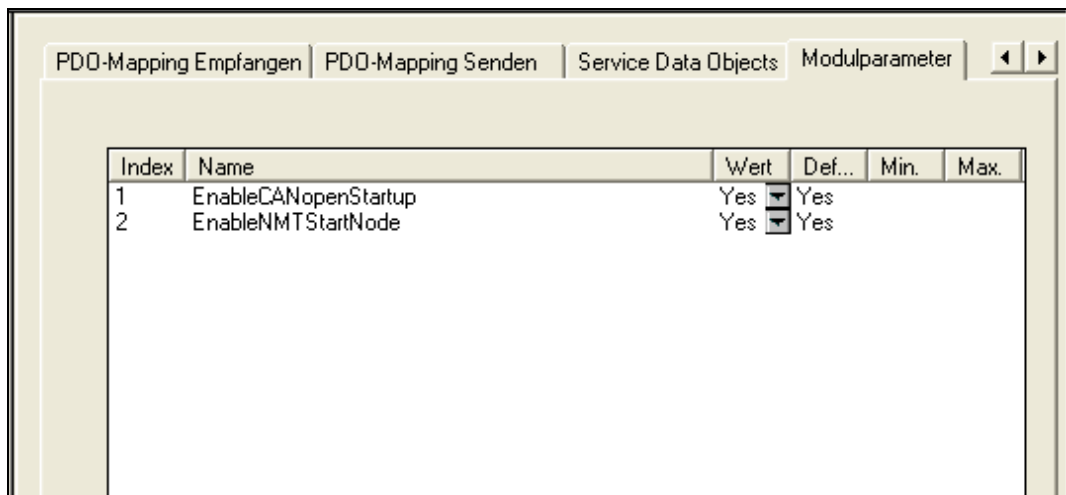


Abbildung 84: Karteireiter „Modulparameter“

Tabelle 68: Beschreibung der Modulparameter (Slave)

PDO-Mapping Empfangen	
EnableCANopenStartup	<p>Yes (Voreinstellung): Während der Hochlaufphase des CANopen-Netzwerks werden sämtliche Basic-SDO-Frames an den ausgewählten CANopen-Slave gesendet.</p> <p>No: In dieser Einstellung werden keine SDO-Frames an das CAN-Layer-2-Gerät gesendet.</p>
EnableNMTStartNode	<p>Yes (Voreinstellung): Während der Hochlaufphase des CANopen-Netzwerks wird das NMT-Kommando „Start remote node“ an den ausgewählten CANopen-Slave gesendet (Kommunikationsverbindung wird aufgebaut).</p> <p>No: In dieser Einstellung wird kein Startkommando übertragen. Der CANopen-Slave kann später zu einem beliebigen Zeitpunkt mit dem Kommando „Start remote node“ gestartet werden.</p> <p>Hinweis: Deaktivieren Sie dazu den Parameter „Automatisch starten“.</p>

12.3 Zugriff auf die CANopen-Prozessdaten

In der Steuerungskonfiguration erscheinen unter dem Slave „WAGO 750-337“ die eingefügten Busklemmen mit den dazugehörigen festen Adressen (siehe Abb.). Sie können für die Ein- und Ausgänge dieser Busklemmen eigene Variablen deklarieren.

Zur Variablendeklaration gehen Sie folgendermaßen vor:

1. Klicken Sie auf den Karteireiter „Ressourcen“ und öffnen Sie die Steuerungskonfiguration.
2. Klicken Sie dazu mit einem Doppelklick auf den Eintrag „AT“ (neben dem Pfeil) und geben Sie einen Variablennamen ein. In diesem Beispiel wurden die Namen „CAN_Input_Byte“ und „CAN_Output_Byte“ vergeben.

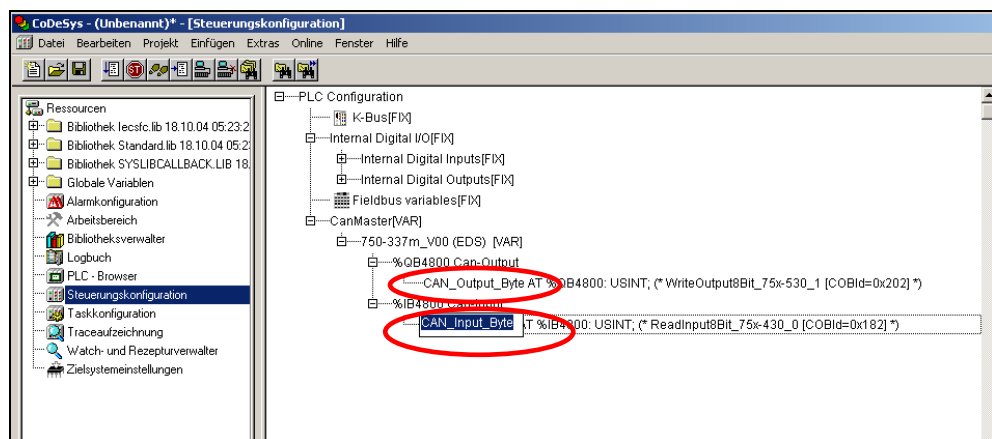


Abbildung 85: Steuerungskonfiguration: Busklemmen mit den dazugehörigen Adressen

Folgendes Beispiel in der Programmiersprache „Strukturierter Text“ (ST) soll den Zugriff auf die Variablen verdeutlichen. Dazu wird ein Eingang einem Ausgang zugewiesen:

1. Wechseln Sie auf den Karteireiter „Baustein“ und klicken Sie mit einem Doppelklick auf den Programmbaustein PLC_PRG.

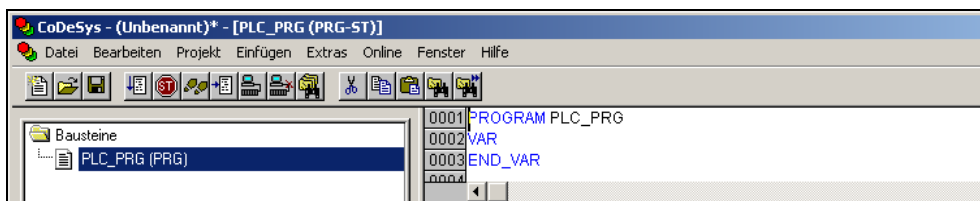


Abbildung 86: PLC_PRG

2. Drücken Sie die Taste **[F2]**, um die Eingabehilfe zu öffnen, oder Sie klicken auf die rechte Maustaste und wählen aus dem Kontextmenü **Eingabehilfe**.
3. Selektieren Sie unter „Globale Variablen“ die zuvor deklariert Variable „CAN_Output_Byte“ und klicken Sie zum Einfügen dieser auf **[OK]**.

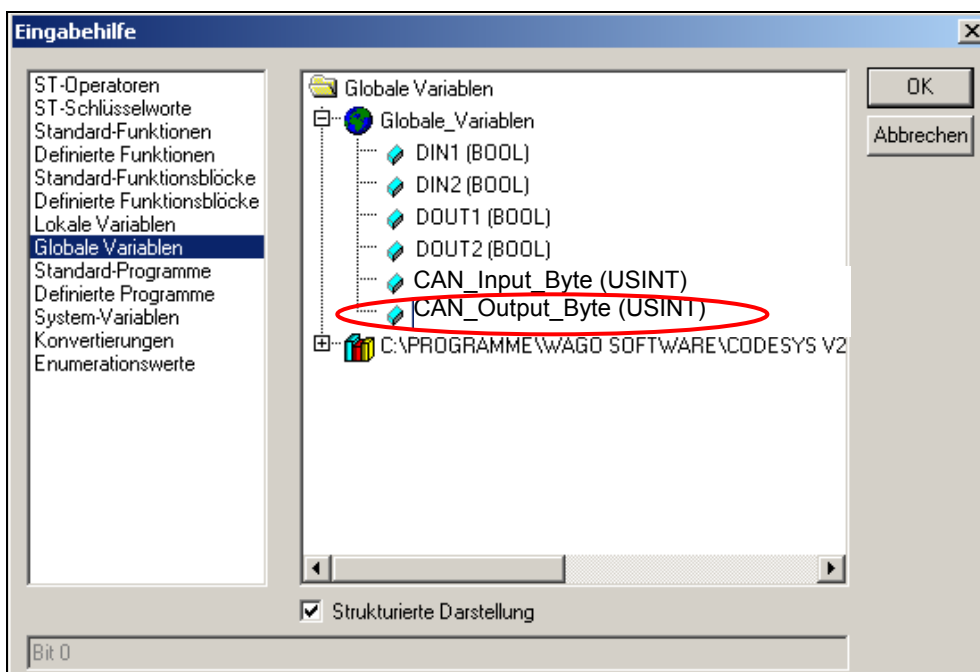


Abbildung 87: Eingabehilfe zur Auswahl der Variablen

4. Geben Sie hinter dem Variablennamen „CAN_Output_Byte“ für den Bitzugriff **.0** ein.
5. Geben Sie hinter Bitzugriff **.0** die Zuweisung **:=** ein.
6. Wiederholen Sie Schritt 3 für die Variable „CAN_Input_Byte“.

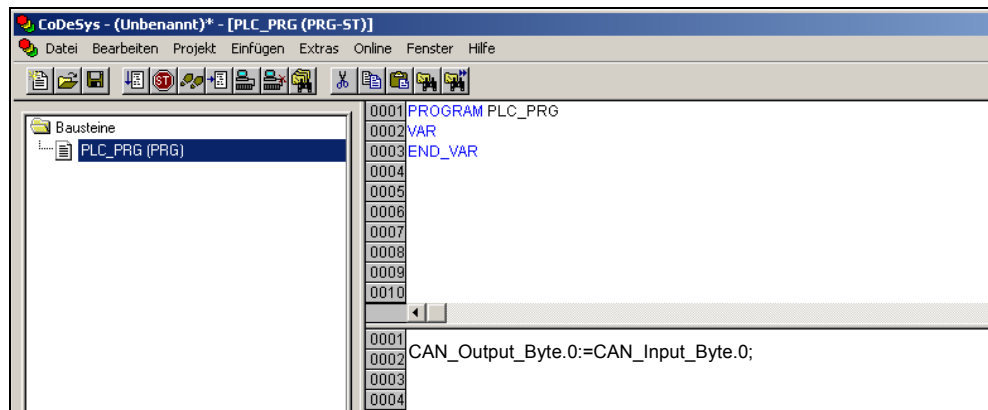


Abbildung 88: Beispiel einer Zuweisung der zuvor angelegten Variablen

7. Zum Kompilieren klicken Sie in der Menüleiste auf **Projekt > Alles Übersetzen**.

12.4 SPS-Programm in den I/O-IPC laden

Zum Übertragen des SPS-Programms in den I/O-IPC klicken Sie der Menüleiste auf **Online > Einloggen**. Der CAN-Bus startet automatisch beim Download der Steuerungskonfiguration in den I/O-IPC. Siehe dazu auch Kapitel „SPS-Programm in den I/O-IPC laden und ausführen (ETHERNET)“.

12.5 Diagnose des Feldbuskopplers

Dieses Kapitel setzt gute Kenntnisse über das Programmierwerkzeug CODESYS voraus. Es erläutert ausschließlich die Vorgehensweise zur Erstellung einer Diagnose anhand eines Beispiels für den Feldbus-Master.

Voraussetzungen für eine Diagnose in den Feldbus-Netzwerken sind konfigurierte Slaves, z. B. ein Feldbuskoppler oder Feldbuscontroller.

12.5.1 DiagGetBusState() und DiagGetState()

Für die Auswertung der Diagnose benötigen Sie folgende Funktionsbausteine aus der Bibliothek BusDiag.lib:

- **DiagGetBusState()** für die Busdiagnose
Dieser Funktionsbaustein liefert Ihnen allgemeine Informationen über jeden angeschlossenen Slave (z. B. Anzahl der Slaves).
- **DiagGetState()** für die Teilnehmerdiagnose
Dieser Funktionsbaustein liefert Ihnen detaillierte Informationen zu jedem Slave (z. B. Informationen über Diagnosen).

12.5.2 Erstellen von Diagnosefunktionen in CODESYS 2.3

Um eine Busdiagnose und eine Teilnehmerdiagnose der Slaves durchzuführen, ist die Bibliothek BusDiag.lib in CODESYS einzubinden. In dieser sind die für die Diagnose benötigten Funktionsbausteine DiagGetBusState() für die Busdiagnose und DiagGetState() für die Teilnehmerdiagnose enthalten.

Binden Sie die Bibliothek BusDiag.lib wie folgt in CODESYS ein:

1. Klicken Sie auf den Karteireiter „Ressourcen“.

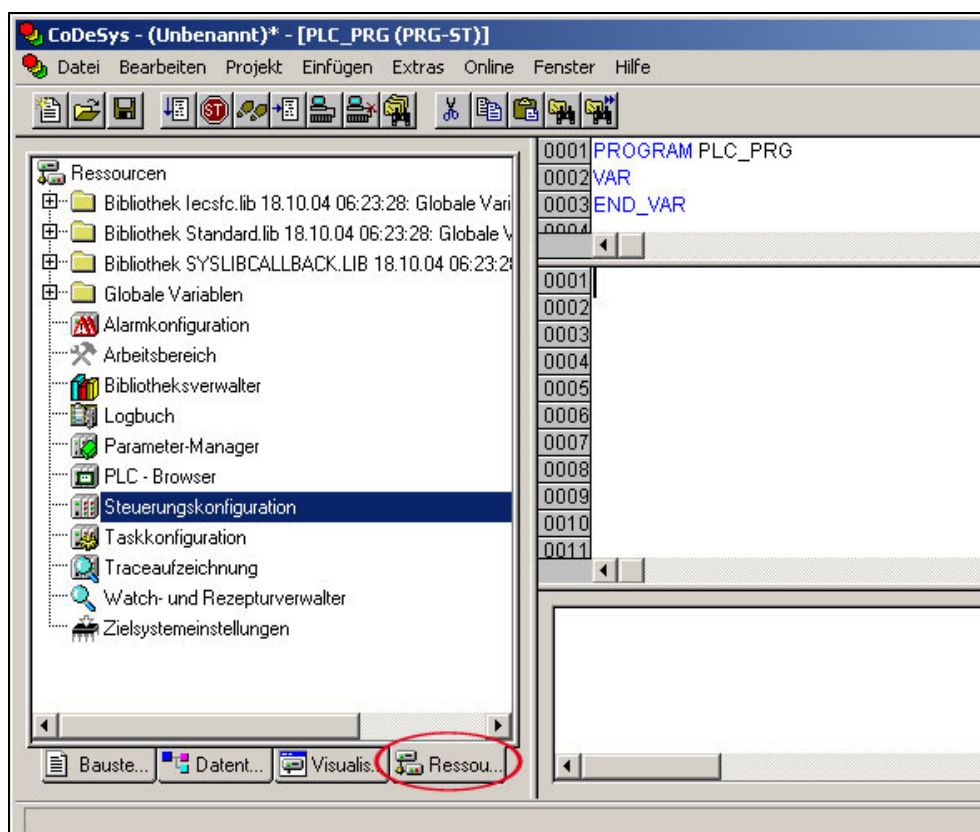


Abbildung 89: Karteireiter „Ressourcen“

2. Klicken Sie in der linken Spalte mit einem Doppelklick auf „Bibliotheksverwalter“.

3. Klicken Sie in der Menüleiste auf **Einfügen > Weitere Bibliothek**. Es öffnet sich der „Öffnen“-Dialog. Selektieren Sie die BusDiag.lib und klicken auf **[Öffnen]**, um diese in das Projekt einzufügen.

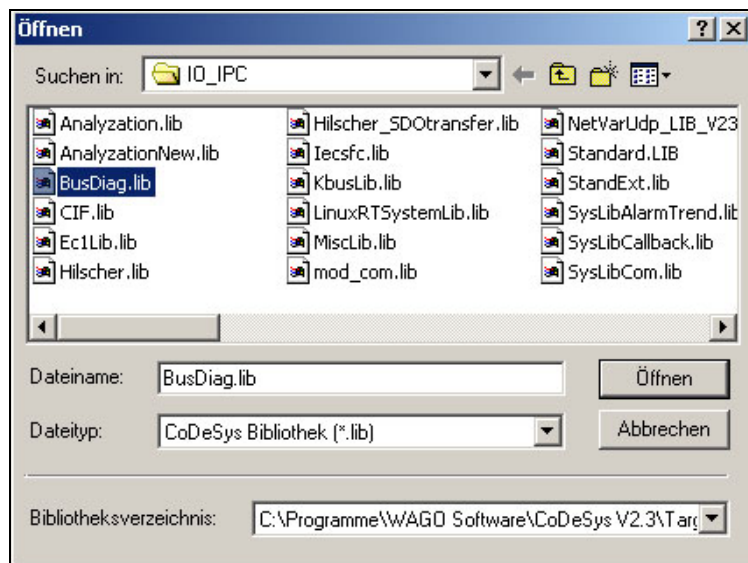


Abbildung 90: Dialog „Öffnen“

4. Klicken Sie in der Menüleiste auf das Symbol „Baustein“.

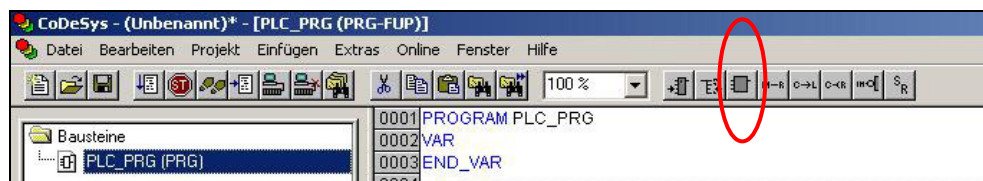


Abbildung 91: Baustein-Symbol in der Menüleiste; Programmiersprache FUP

5. Drücken Sie die Taste **[F2]** auf Ihrer Tastatur. Es öffnet sich der Dialog „Eingabehilfe“. Klicken Sie auf die Option „Standard-Funktionsblöcke“ und wählen Sie den Funktionsbaustein DiagGetBusState().
6. Erstellen Sie eine Instanz des Funktionsbausteins DiagGetBusState(). Geben Sie dazu einen Namen oberhalb des Funktionsbausteins ein. In diesem Beispiel ist dies „GeneralBusInformation“.

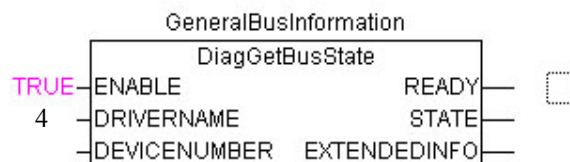


Abbildung 92: Instanz des Funktionsbausteins DiagGetBusState() in FUP

7. Rufen Sie den Funktionsbaustein DiagGetBusState() für die Diagnose der Slaves aus der Bibliothek BusDiag.lib auf.

8. Erstellen Sie eine Instanz des Funktionsbausteins DiagGetState(). In unserem Beispiel ist dies „DiagnoseKnoten“.

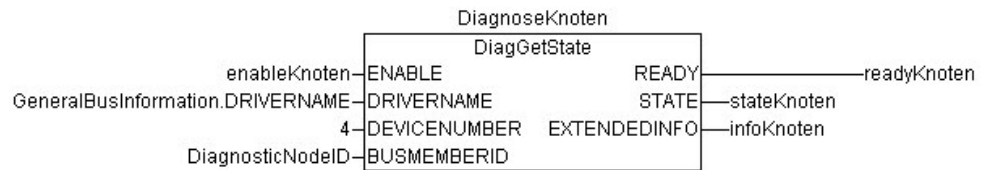


Abbildung 93: Funktionsbaustein DiagGetState() in FUP

Während des Programmablaufs werden in diesem Beispiel beide Funktionsblöcke aufgerufen. Um die Zykluszeiten nicht während des Programmablaufs zu verlängern, setzen Sie den Eingang „ENABLE“ von DiagGetState() erst dann auf „TRUE“, wenn Sie eine Diagnose durchführen.

12.5.3 Aufruf des Diagnosebausteins

Rufen Sie den Funktionsbaustein wie im nachfolgenden Bild dargestellt auf.

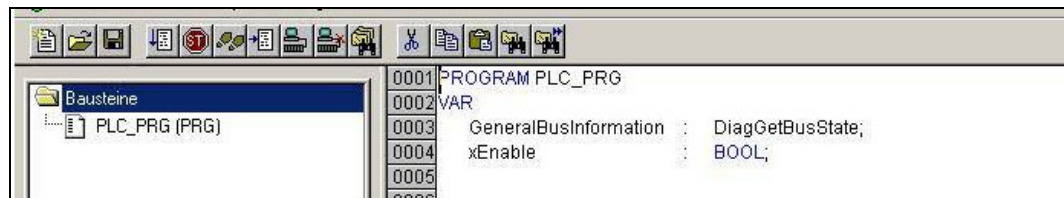


Abbildung 94: Offline-Ansicht des Variablenfensters in CODESYS

12.5.4 Durchführen der Busdiagnose mittels DiagGetBusState()

Zum Durchführen einer Busdiagnose gehen Sie folgendermaßen vor:

1. Melden Sie sich in CODESYS an. Klicken Sie dazu in der Menüleiste auf **Online > Einloggen**. Nun zeigt das Variablenfenster die Informationen über die Variablen an (Online-Ansicht).
2. Zum Starten des SPS-Programms klicken Sie in der Menüleiste auf **Online > Start**. Durch das Starten wird der Funktionsbaustein DiagGetBusState() aufgerufen und die Diagnoseinformation in das Array EXTENDEDINFO ausgegeben.

In der Online-Ansicht des Variablenfensters gibt das Array EXTENDEDINFO Auskunft über den Zustand der Slaves. Für jeden Slave ist ein Eintrag im Array reserviert. Die Slave-Adresse ist dem Array-Index zugeordnet. In diesem Beispiel sind es die Slaves mit den Stationsadressen 2 und 5, die Diagnoseinformation bereithalten.

Hinweis



Anzeige der Diagnoseinformationen

Die Diagnoseinformationen werden nur für die Dauer eines Programmzyklus angezeigt. Sollen die Diagnoseinformationen länger verfügbar sein, ist ein entsprechendes Programm zu schreiben.

Abbildung 95: Online-Ansicht des Variablenfensters (oberes Fenster) in FUP

- Die Binärdarstellung erleichtert die Auswertung der einzelnen Diagnosebits. Sie können sich die Diagnoseinformationen des Arrays EXTENDEDINFO in der Binärdarstellung anzeigen lassen. Klicken Sie dazu mit der rechten Maustaste in das Variablenfenster und wählen Sie **binär**.

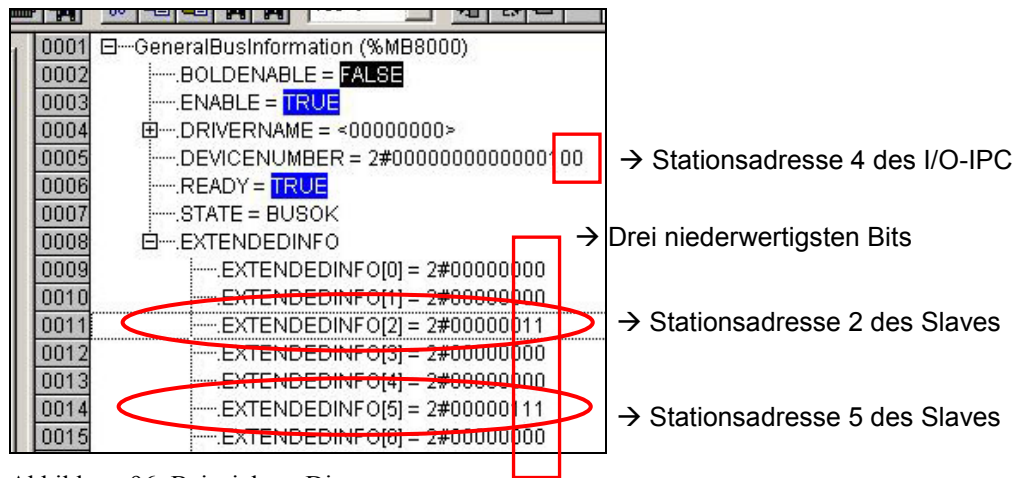


Abbildung 96: Beispiel zur Diagnose

- Vergleichen Sie die drei niederwertigsten Bits der Diagnoseinformation der Slaves mit den Stationsadressen 2 und 5 mit den Bits aus der folgenden Tabelle:

Tabelle 69: Bits der Diagnoseinformation

2. Bit		1. Bit		0. Bit	
1	0	1	0	1	0
Es stehen Diagnoseinformationen am Slave bereit.	Es stehen keine Diagnoseinformationen am Slave bereit.	Slave ist aktiv.	Slave ist inaktiv.	Slave projiziert.	Slave nicht projiziert.

- Der Slave mit der Stationsadresse 2 liefert den Wert 011. Dieser bedeutet, dass der Slave projiziert und aktiv ist.
- Der Slave mit der Stationsadresse 5 liefert den Wert 111. Dieser bedeutet, dass der Slave projiziert und aktiv ist sowie Diagnoseinformationen bereithält. Zum Auswerten dieser Diagnoseinformationen ist die Teilnehmerdiagnose durchzuführen. Siehe dazu das Kapitel „Durchführen der Teilnehmerdiagnose mittels DiagGetState()“.

Hinweis**Diagnoseinformationen**

Wenn `READY = TRUE`, dann gibt `STATE` durch einen der folgenden Werte Auskunft über den aktuellen Busstatus:

BUSOK: alle konfigurierten Slaves befinden sich im Datenaustausch mit dem DP-Master.

BUSFAULT: einer oder mehrere konfigurierte Slaves befinden sich nicht im Datenaustausch mit dem DP-Master

BUSNOTCOMMUNICATION: alle konfigurierten Slaves befinden sich nicht im Datenaustausch mit dem DP-Master.

12.5.5 Durchführen der Teilnehmerdiagnose mittels `DiagGetState()`

Hat die Busdiagnose ergeben, dass eine Busklemme eine Diagnoseinformation bereitstellt, dann nehmen Sie am entsprechenden Slave eine Teilnehmerdiagnose vor. Gehen Sie dazu folgendermaßen vor:

1. Rufen Sie den Funktionsbaustein `DiagGetState()` auf, indem Sie den Eingang `ENABLE` auf „True“ setzen.
2. Geben Sie an der Eingangsvariablen `BUSMEMBERID` den Slave an, an dem eine Diagnoseinformation anliegt. In unserem Beispiel ist es der Slave mit der Feldbusadresse 5.

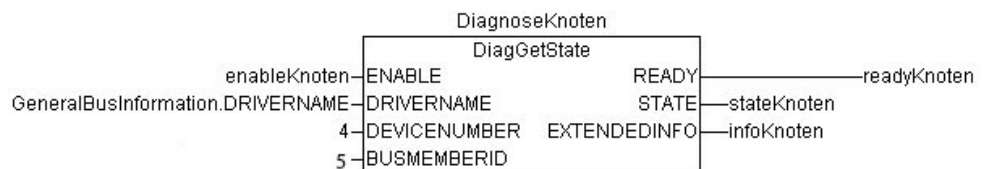


Abbildung 97: Diagnoseaufruf `DiagGetState()`

- **DRIVERNAME:**
Der Eingangsparameter `DRIVERNAME` wird über die Instanzdaten des Bausteins `DiagGetBusState` vorgegeben.
- **DEVICENUMBER:**
Die `DEVICENUMBER` muss beim I/O-IPC stets 4 lauten.

12.5.6 Auswerten der CANopen-Diagnose (Emergency-Nachrichten)

Die Arrayelemente [0] bis [15] in der unten stehenden Abbildung enthalten die CANopen-Statusinformationen in Bytes. Ab Arrayelement [16] sind die Emergency-Nachrichten der Slaves abgelegt.

```
16  B- EXTENDEDINFO
17  ..... EXTENDEDINFO[0] = 2#00000000
18  ..... EXTENDEDINFO[1] = 2#00000000    → Byte 1
19  ..... EXTENDEDINFO[2] = 2#00000000    → Byte 2
20  ..... EXTENDEDINFO[3] = 2#00000000
21  ..... EXTENDEDINFO[4] = 2#00000000    → ...
22  ..... EXTENDEDINFO[5] = 2#00011100
23  ..... EXTENDEDINFO[6] = 2#00000101
24  ..... EXTENDEDINFO[7] = 2#00000001
25  BYTE EXTENDEDINFO[8] = 2#00001000
26  ..... EXTENDEDINFO[9] = 2#00001100
27  ..... EXTENDEDINFO[10] = 2#00000000
28  ..... EXTENDEDINFO[11] = 2#00000001
29  ..... EXTENDEDINFO[12] = 2#10110111    Bereich der CANopen-
30  ..... EXTENDEDINFO[13] = 2#01010100    Statusinformationen
31  ..... EXTENDEDINFO[14] = 2#01001001
32  ..... EXTENDEDINFO[15] = 2#00001000
33  ..... EXTENDEDINFO[16] = 2#00000000
34  ..... EXTENDEDINFO[17] = 2#00000000
35  ..... EXTENDEDINFO[18] = 2#00000000
36  ..... EXTENDEDINFO[19] = 2#00000000
37  ..... EXTENDEDINFO[20] = 2#00000000
38  ..... EXTENDEDINFO[21] = 2#00000000
39  ..... EXTENDEDINFO[22] = 2#00000000
40  ..... EXTENDEDINFO[23] = 2#00000111
41  ..... EXTENDEDINFO[24] = 2#10100000
42  ..... EXTENDEDINFO[25] = 2#00000000
43  ..... EXTENDEDINFO[26] = 2#00000000
44  ..... EXTENDEDINFO[27] = 2#00000000
45  ..... EXTENDEDINFO[28] = 2#00000000
46  ..... EXTENDEDINFO[29] = 2#00000000
47  ..... EXTENDEDINFO[30] = 2#10000100    Bereich für die Emergency-
48  ..... EXTENDEDINFO[31] = 2#01000000    Nachrichten der Slaves
49  ..... EXTENDEDINFO[32] = 2#10101000
50  ..... EXTENDEDINFO[33] = 2#10000100
51  ..... EXTENDEDINFO[34] = 2#01000001
52  ..... EXTENDEDINFO[35] = 2#10101000
53  ..... EXTENDEDINFO[36] = 2#00000000
```

Abbildung 98: Online-Ansicht des Arrays EXTENDEDINFO in der Binärdarstellung

Die CANopen-Statusinformationen und die Emergency-Nachrichten der Slaves sind auf den nachfolgenden Seiten beschrieben.

Beschreibung der Diagnoseinformation des Bausteins DiagGetState.EXTENDEDINFO für CANopen

Die folgende Tabelle beschreibt den Inhalt des Arrays EXTENDEDINFO[0-23].

EXTENDEDINFO[0]:	Slave-Adresse
EXTENDEDINFO[1]:	0 (ungenutzt)
EXTENDEDINFO[2]:	0 (ungenutzt)
EXTENDEDINFO[3]:	0 (ungenutzt)
EXTENDEDINFO[4]:	0 (ungenutzt)
EXTENDEDINFO[5]:	48, Länge der Diagnosestruktur
EXTENDEDINFO[6]:	10 (ungenutzt)
EXTENDEDINFO[7]:	1 (ungenutzt)
EXTENDEDINFO[8]:	Verbindungsstatus des Slaves Bit 1: Slave antwortet nicht. Bit 2: Fehlerspeicher ist voll. Bit 3: Slave ist falsch parametrier Bit 4: „Node guarding“-Protokoll ist aktiviert. Bit 5: Reserviert Bit 6: Reserviert Bit 7: Reserviert Bit 8: Slave ist deaktiviert
EXTENDEDINFO[9 – 12]:	Device Type Profilnummer und Zusatzinformationen des Slaves. Diese vier Bytes werden während des Hochfahrens des Slaves über das Objekt 0x1000 ausgelesen.
EXTENDEDINFO[13]:	Stationsstatus Rückmeldung des Slave-Status vom „Node Guarding“-Protokoll. Falls das „Node guarding“-Protokoll für diesen Slave aktiv ist, wird der Status im Arrayelement EXTENDEDINFO[13] ausgegeben: 1 Disconnected (Stationsausfall) 2 Connecting 3 Preparing 4 Prepared 5 Operational (Operational Mode) ... 127 Pre-Operational (Pre-Operationaln Mode und Stopped)
EXTENDEDINFO[14]:	Actual error In diesem Byte werden die Online-Error-Information des Slaves angezeigt. Siehe dazu die Tabelle auf der nachfolgenden Seite.
EXTENDEDINFO[15]	Emcy_Entries Dieses Byte beinhaltet die Anzahl der gespeicherten Emergency-Nachrichten des nachfolgenden Datenbereichs.

EXTENDEDINFO[16 – 23]	<p>Emcy_Data</p> <p>Im EXTENDEDINFO[16 – 23] sind die Emergency-Nachrichten gespeichert.</p> <p>Eine Emergency-Nachricht besteht aus dem „Error Code“, „Error Register“ und „Additional Code“:</p> <p>EXTENDEDINFO[16]: Error Code LSB EXTENDEDINFO[17]: Error Code MSB EXTENDEDINFO[18]: Error Register EXTENDEDINFO[19]: Additional Code EXTENDEDINFO[20]: Additional Code EXTENDEDINFO[21]: Additional Code EXTENDEDINFO[22]: Additional Code EXTENDEDINFO[23]: Additional Code</p> <p>Eine ausführliche Beschreibung der Emergency-Nachrichten entnehmen Sie der Handbücher zu den CANopen-Slaves (z. B. 750-337, Kapitel „Emergency“).</p>
-----------------------	--

Actual error:

Tabelle 70: Actual error

Err event	Ursache	Bereich	Abhilfe
30	Das „Node Guarding“ wird nicht in der eingestellten Zeit beantwortet: Der I/O-IPC kann nicht mit dem Slave kommunizieren.	Slave	Überprüfen Sie die Verkabelung der Versorgungsspannung und des Feldbusses auf Beschädigung. Erhöhen Sie die „Node Guarding“-Zeit.
31	Slave befindet sich nicht mehr im Operational-Zustand	Slave	Führen Sie einen Hardware-Reset durch. Drücken Sie dazu am I/O-IPC den Reset-Taster (46) oder schalten Sie dessen Versorgungsspannung aus und anschließend wieder ein. Führen Sie den NMT-Service „Reset_Node“ aus. Führen Sie einen Neustart des SPS-Programms durch.
32	Sequenzfehler im „Guarding“-Protocol	Slave	Führen Sie einen Hardware-Reset durch. Drücken Sie dazu am I/O-IPC den Reset-Taster (46) oder schalten Sie dessen Versorgungsspannung aus und anschließend wieder ein. Führen Sie einen Neustart des SPS-Programms durch.
33	-	-	-
34	Der Slave kann keine Konfigurationstelegramme beantworten.	Slave	Überprüfen Sie die Verkabelung der Versorgungsspannung und des Feldbusses auf Beschädigung. Der Slave muss sich im Operational-Zustand befinden.
35	Das eingestellte Profil „DSP301, 306“ des I/O-IPC stimmt nicht mit dem des Slaves überein.	Steuerungs-konfiguration	Deaktivieren Sie im Karteireiter „CAN Parameter“ das Kontrollfeld „DSP301, V4.01 ...“.

Tabelle 70: Actual error

Err_event	Ursache	Bereich	Abhilfe
36	Der „Device Type“ des Slaves stimmt nicht mit den gesteckten Busklemmen überein.	Steuerungs- konfiguration	Tragen Sie den korrekten „Device Type“ auf dem Karteireiter „CAN Parameter“ ein. Dieser ist abhängig vom Typ (digital/analog) der gesteckten Busklemmen am Slave. Siehe dazu Kapitel „Steuerungskonfiguration des CANopen-I/O-IPC“.
37	Ungültige SDO-Antwort erhalten	Slave	Slave entspricht nicht dem CANopen-Profil.
38	SDO-Datenlänge < 8 Byte	Slave	Slave entspricht nicht dem CANopen-Profil.

12.6 Datenaustausch von einfachen CAN-Teilnehmern mit dem I/O-IPC

Zur Vereinfachung der Steuerungskonfiguration beim Einbinden von CAN-Layer-2-Geräten wurde die auf das Wesentliche reduzierte EDS-Datei „Generic CAN-Device“ erstellt. Die EDS-Datei enthält 16 Sende- und Empfangs-PDOs, welche jeweils mit 8x1byte-Einträgen belegt sind. Sie müssen nur noch die CANopen typischen Konfigurations- und Überwachungstelegramme für diesen Teilnehmer deaktivieren.

Sie können die Steuerungskonfiguration auch mit einer beliebigen EDS-Datei für CANopen durchführen.

1. Zum Einfügen des CANopen-I/O-IPC in die Steuerungskonfiguration klicken Sie mit der rechten Maustaste auf „PLC-Configuration“ und wählen Sie „CanMaster anhängen“.

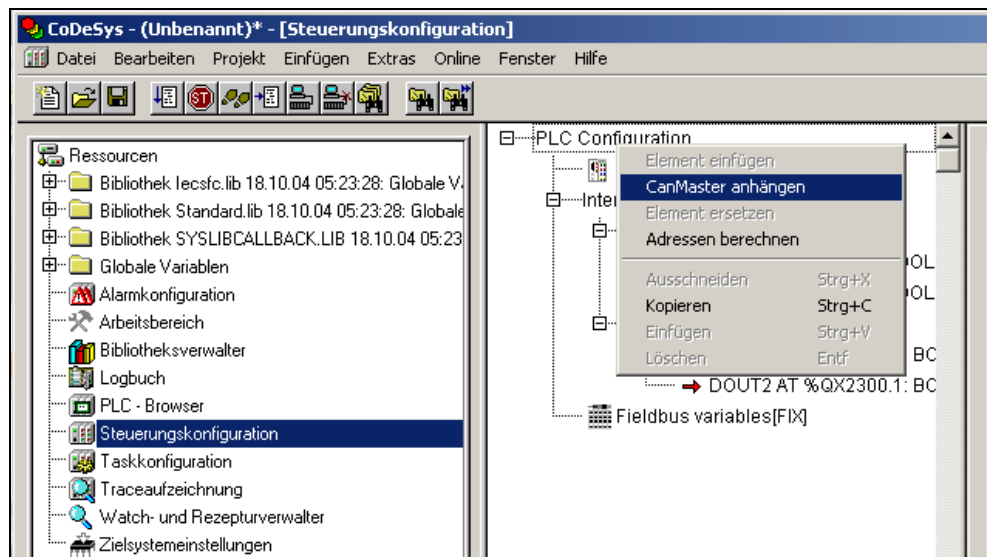


Abbildung 99: Anhängen des CANopen-Masters

2. Wählen Sie im Karteireiter „CAN Parameter“ die gewünschte Baudrate aus.

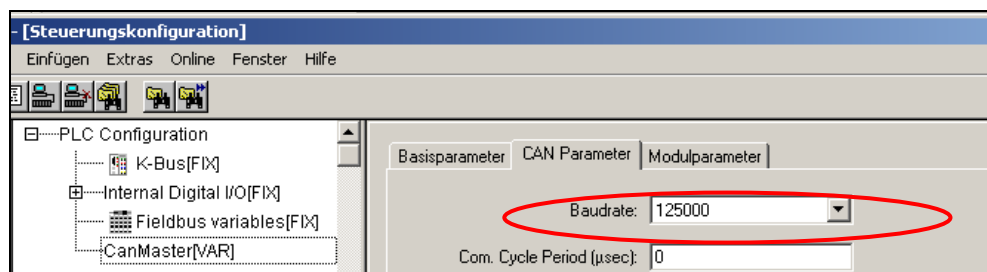


Abbildung 100: Baudrate einstellen

3. Zum Einfügen des Slaves klicken Sie mit der rechten Maustaste auf den I/O-IPC (CANMaster[VAR]) und wählen Sie **Unterelemente anhängen > Generic CAN-Device**.

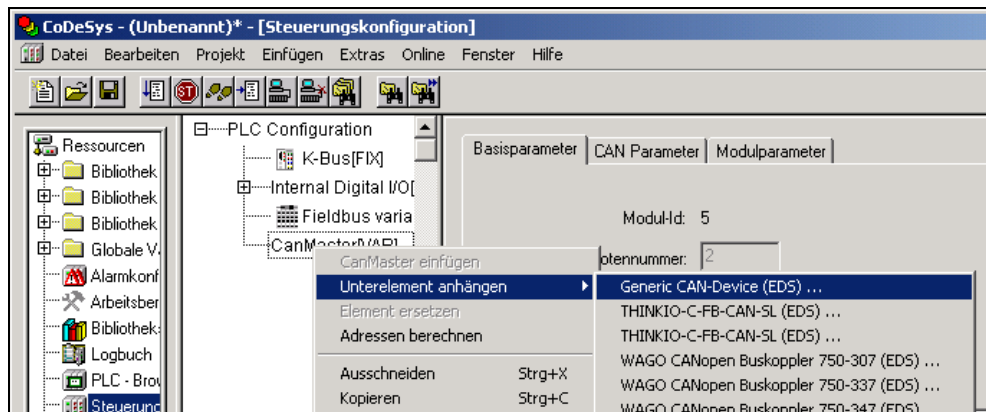


Abbildung 101: EDS-Datei „Generic CAN-Device“

4. Öffnen Sie den Karteireiter „Modulparameter“ des Slaves. Zur Kommunikation mit einfachen CAN-Layer-2-Geräten deaktivieren Sie über „EnableCANopenStartup“ (= „No“) die „BasicSDO“ (0x1000, 0x1005, 0x100C, 0x100D, 0x1014).

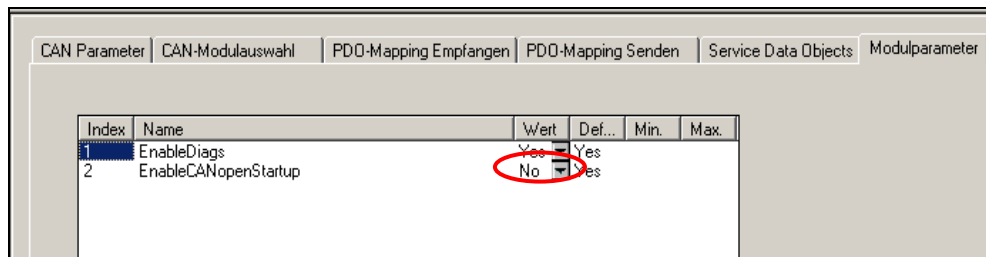


Abbildung 102: Karteireiter „Modulparameter“ CAN

5. Öffnen Sie den Karteireiter „CAN Parameter“ des Slaves. Deaktivieren Sie die Parameter „CommSDO“ und „MappingSDO“.

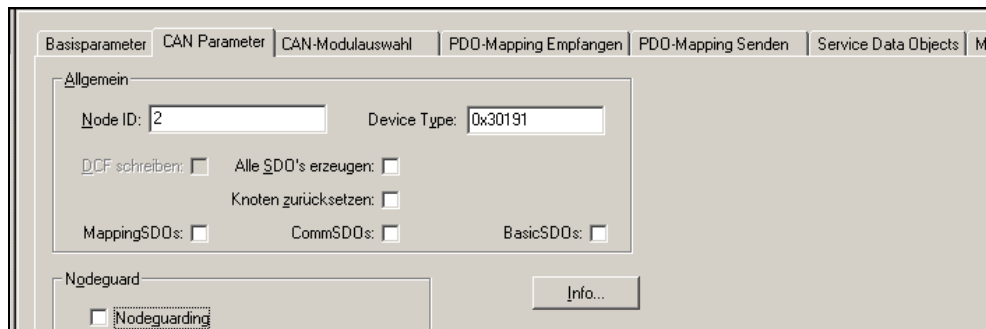


Abbildung 103: Karteireiter „CAN Parameter“ CAN

6. Zum Konfigurieren der CAN-Frames für CAN-Layer-2-Geräte siehe Kapitel „CANopen-Slaves“ > PDO-Mapping empfangen/senden.

13 C-Funktionen als CODESYS-Bibliothek einbinden

Zur Verwendung beliebiger C- oder auch Linux-Funktionalitäten in CODESYS steht die im Folgenden beschriebene Import-Schnittstelle zur Verfügung. Über diese können Sie Linux-Libraries dynamisch in das CODESYS-Laufzeitsystem laden und verwenden.

13.1 Beispiel zum Einbinden einer dynamischen Library

Die folgenden Kapitel erläutern Ihnen anhand eines Beispiels die Vorgehensweise zur Einbindung einer dynamischen Library mittels einer C-Funktion in CODESYS.

13.1.1 Linux-Shared-Library entwickeln und übersetzen

Bevor Sie C-Funktionen innerhalb eines CODESYS-Programms verwenden können, müssen Sie eine Linux-Shared-Library entwickeln und übersetzen. Gehen Sie dazu folgendermaßen vor:

1. Erstellen Sie eine Datei mit den benötigten Funktionen. In diesem Beispiel wurde dazu die Datei „libmytest.c“ mit einer Funktion „MyTestFunction“ und den Datentypen „unsigned-short“ erstellt.

```
#include <stdio.h>
unsigned short MyTestFunction(unsigned short value)
{
    return value+=2;
}
```

Abbildung 104: Datei „libmytest.c“

2. Kompilieren und Linken Sie die Datei, indem Sie folgenden Befehl in die Linux-Konsole eingeben:

```
gcc libmytest.c -Wall -shared -o libmytest.so
```

In Linux ist beim Aufrufen des Compilers „gcc“ der Parameter „-shared“ zu verwenden.

Bei einer fehlerfreien Kompilierung der Datei wird mit dem Parameter „-shared“ eine dynamische Library „libmytest.so“ erzeugt, welche die C-Funktion „MyTestFunction“ enthält.

Hinweis



Dateinamen der Linux-Libraries

Die Dateinamen der Linux-Libraries müssen mit lib beginnen.

13.1.2 Beschreibungsdatei für das CODESYS-Laufzeitsystem erzeugen

Erzeugen Sie eine Beschreibungsdatei mit dem Namen „extlibs.ini“, um die C-Funktionen dem CODESYS-Laufzeitsystem bekannt zu machen.

1. Damit das Laufzeitsystem beim Starten die Linux-Library erkennt und korrekt einbinden kann, müssen Sie eine INI-Datei erstellen, die alle dynamisch zu ladenden Library-Namen und deren Funktionsnamen enthält. Für das Beispiel sieht die INI-Datei folgendermaßen aus:

```
[EXT_LIB_LIST]
1=mytest

[mytest]
1=MyTestFunction
```

Abbildung 105: Datei „extlibs.ini“

2. Wenn Sie weitere Libraries unter dem Eintrag EXT_LIB_LIST hinzufügen möchten, fügen Sie unter dem entsprechenden Library-Tag einen fortlaufenden Index, gefolgt vom Library-Namen (z. B. „2 = CSV-File“), ein. Die Angabe der Übergabe- und Rückgabe-Parameter der Funktionen ist an dieser Stelle nicht nötig.

Die entsprechend erweiterte Datei würde folgendermaßen aussehen:

```
[EXT_LIB_LIST]
1=mytest
2=CSV-File

[mytest]
1=MyTestFunction

[CSV-File]
1=ReadCSVString
2=WriteCSVString
```

Abbildung 106: Datei „extlibs.ini“

13.1.3 Library und INI-Datei kopieren und das CODESYS-Laufzeitsystem neu starten

Zum Einbinden der Library und der INI-Datei in das CODESYS-Laufzeitsystem gehen Sie wie folgt vor:

1. Kopieren Sie die beiden neu erstellten Dateien (libmytest.so und extlibs.ini) auf den I/O-IPC. Verwenden Sie dazu einen USB-Speicher, eine CF-Karte, FTP oder NFS.
2. Kopieren Sie die Datei extlibs.ini in das Verzeichnis */home/codesys*.
3. Kopieren Sie die Library in das Verzeichnis */lib* oder */usr/lib*.
Ferner gibt es auch die Möglichkeit, die neue Library an eine beliebige Stelle im I/O-IPC zu kopieren. Passen Sie dazu die Umgebungsvariable `LD_LIBRARY_PATH` in der verwendeten Linux-Konsole vor jedem neuen Starten von CODESYS an, z. B.:

```
env LD_LIBRARY_PATH=/home/codesys ./plclinux_rt
```

4. Bevor sich ein Neustart des CODESYS-Laufzeitsystems durchführen lässt, geben Sie folgenden Befehl in die Linux-Konsole ein:

```
ps A
```

5. Ermitteln Sie aus der angezeigten Liste die PID des Programms „plclinux_rt“
6. Beenden Sie das CODESYS-Laufzeitsystem, indem Sie den Befehl `kill <PID>` mit der zuvor ermittelten PID eingeben, z. B.
7. Zum Neustart des Laufzeitsystems geben Sie folgenden Befehl in die Linux-Konsole ein:

```
kill 2069.
```

```
plclinux_rt &.
```

Bei einer Unstimmigkeit zwischen INI-Datei und der Library wird beim Starten des CODESYS-Laufzeitsystems eine Fehlermeldung auf der Linux-Konsole angezeigt.

Hinweis



CODESYS-Laufzeitsystem

Das CODESYS-Laufzeitsystem muss durch einen Benutzer mit superuser-Rechten neu gestartet werden.

Hinweis



Änderung der Library

Ändern Sie nicht die Library, während diese vom CODESYS-Laufzeitsystem verwendet wird, da andernfalls Zugriffsverletzungen auftreten können.

13.1.4 Eine IEC-Library erzeugen

Um die hinzugefügte Library innerhalb von CODESYS als Bibliotheksfunktionen verwenden zu können, sind die Funktionsprototypen in einer externen CODESYS-Bibliothek anzulegen. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie ein neues CODESYS-Projekt, indem Sie in der Menüleiste **Datei > Neu** wählen.
2. Wählen Sie im Fenster „Zielsystem Einstellung“ „None“ aus und klicken Sie auf die Schaltfläche **[OK]**.

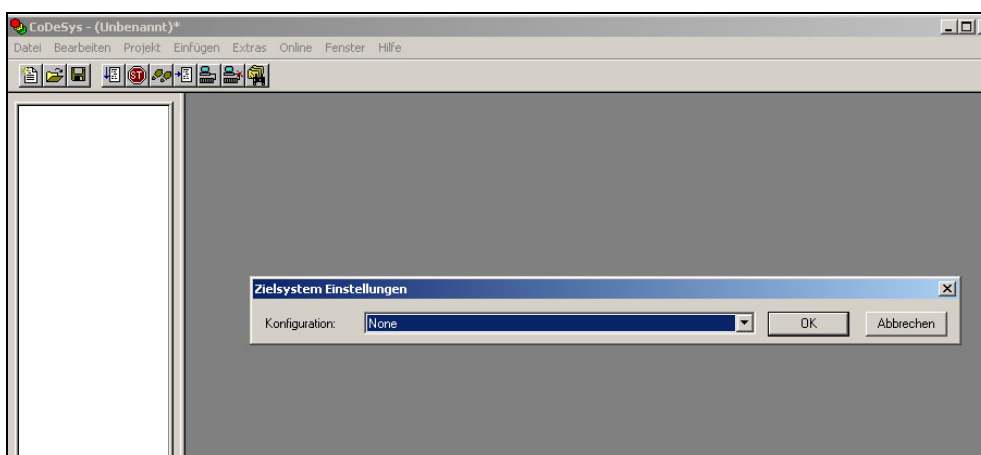


Abbildung 107: Fenster „Zielsystem Einstellungen“

3. Nehmen Sie die im Fenster „Neuer Baustein“ aufgeführten Einstellungen vor (siehe Abb.). Der Name des Bausteins muss mit dem der zuvor erstellten C-Datei übereinstimmen. Beachten Sie dabei die Groß- und Kleinschreibung.

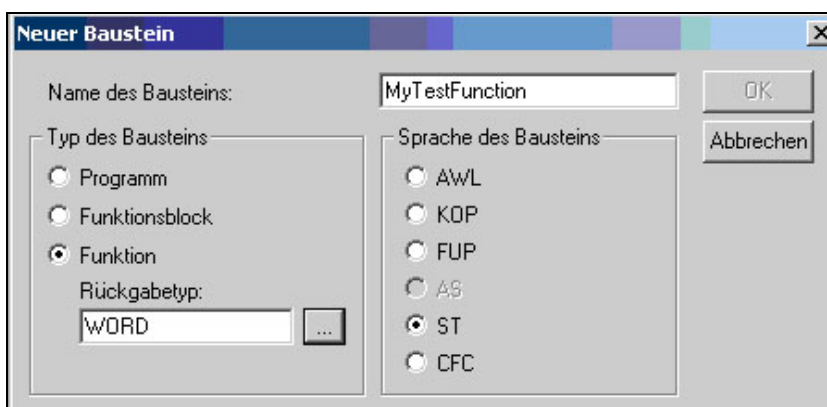


Abbildung 108: Fenster „Neuer Baustein“

- Anschließend definieren Sie den Eingangsparameter mit *value : WORD*; und fügen Sie im Programmteil der Funktion (unteres Fenster) ein Semikolon ein. Andernfalls tritt ein CODESYS-Fehler auf.

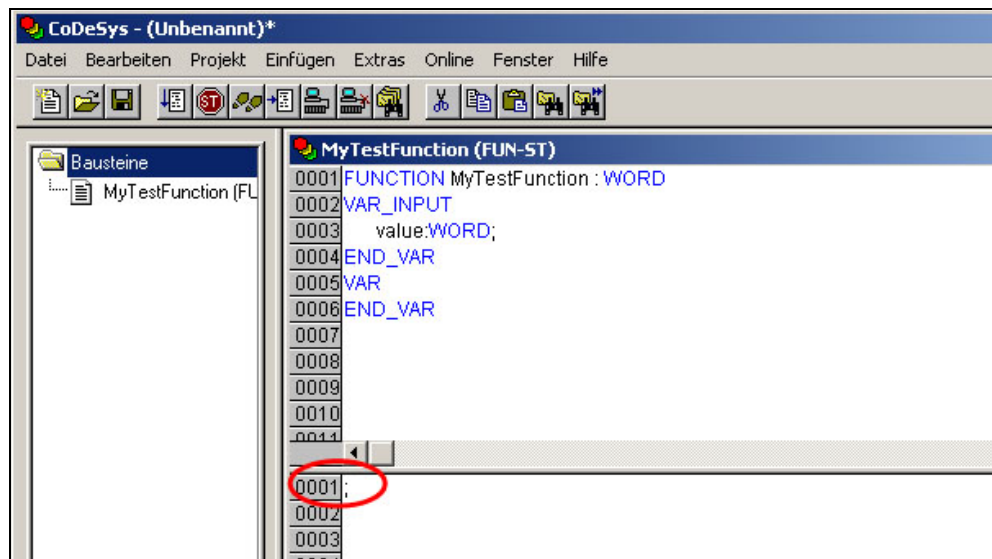


Abbildung 109: Fenster „MyTestFunction“

- Wählen Sie in der Menüleiste **Datei > Speichern unter**. Geben Sie als Dateinamen „mytest.lib“ ein, wählen Sie den Dateityp „Externe Bibliothek“ aus und klicken Sie auf **[Speichern]**.

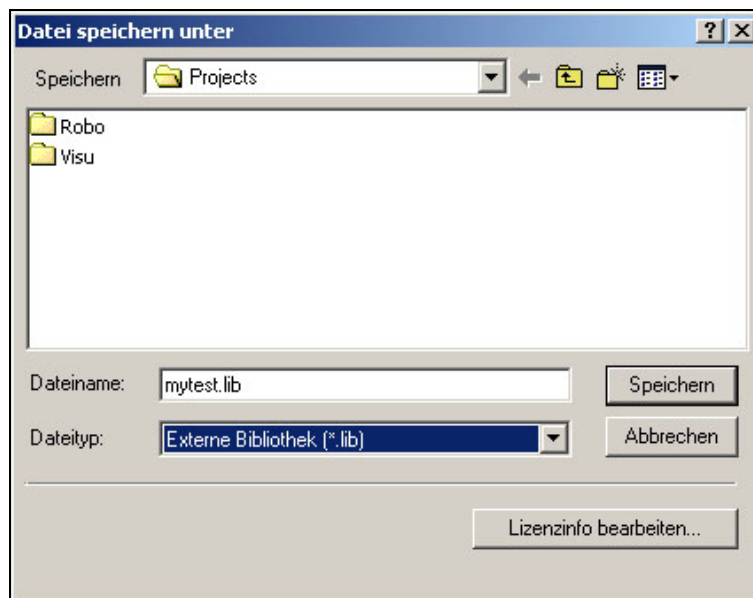


Abbildung 110: Fenster „Datei speichern unter“

Sind mehrere Funktionen in der Library enthalten, lassen sich diese Funktionen hier auch einbinden. Dabei sind auch Funktionen mit mehreren Übergabeparametern möglich.

13.1.5 Bibliothek im CODESYS-Projekt einbinden

Um die zuvor erstellte Bibliothek mytest.lib in CODESYS einzubinden, gehen Sie folgendermaßen vor:

1. Klicken Sie in der Menüleiste auf **Datei** und wählen Sie **Neu**.
2. Öffnen Sie das Auswahlfeld der „Zielsystem Einstellung“ und wählen Sie den von Ihnen verwendeten I/O-IPC. In diesem Beispiel ist es der 758-876-111.

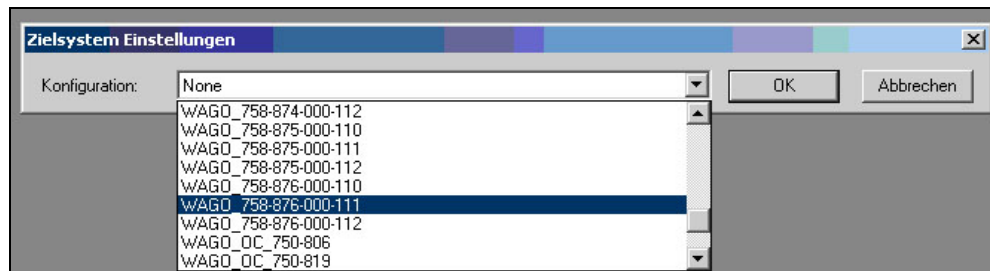


Abbildung 111: Zielsystem-Einstellungen (1)

3. Klicken Sie auf die Schaltfläche **[OK]**. Es öffnet sich das Fenster „Zielsystem Einstellungen“.
4. Klicken Sie im Fenster „Zielsystem Einstellungen“ auf die Schaltfläche **[OK]**.

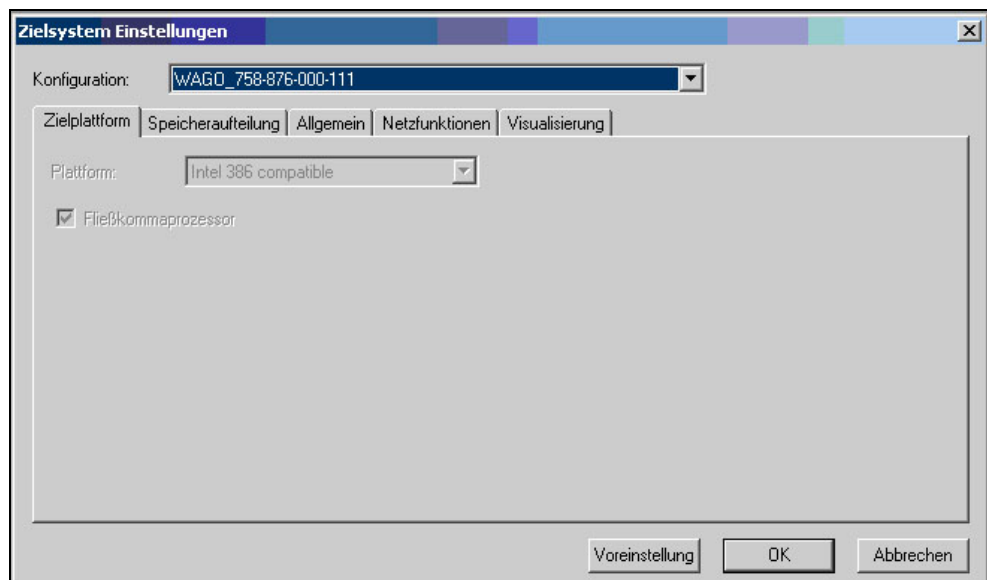


Abbildung 112: Zielsystem-Einstellungen (2)

5. Klicken Sie im Fenster „Neuer Baustein“ auf **[OK]**.

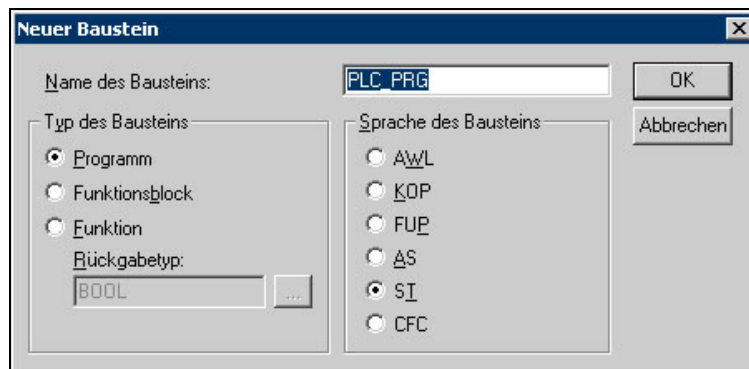


Abbildung 113: Fenster „Neuer Baustein“

6. Klicken Sie auf den Karteireiter „Ressourcen“.

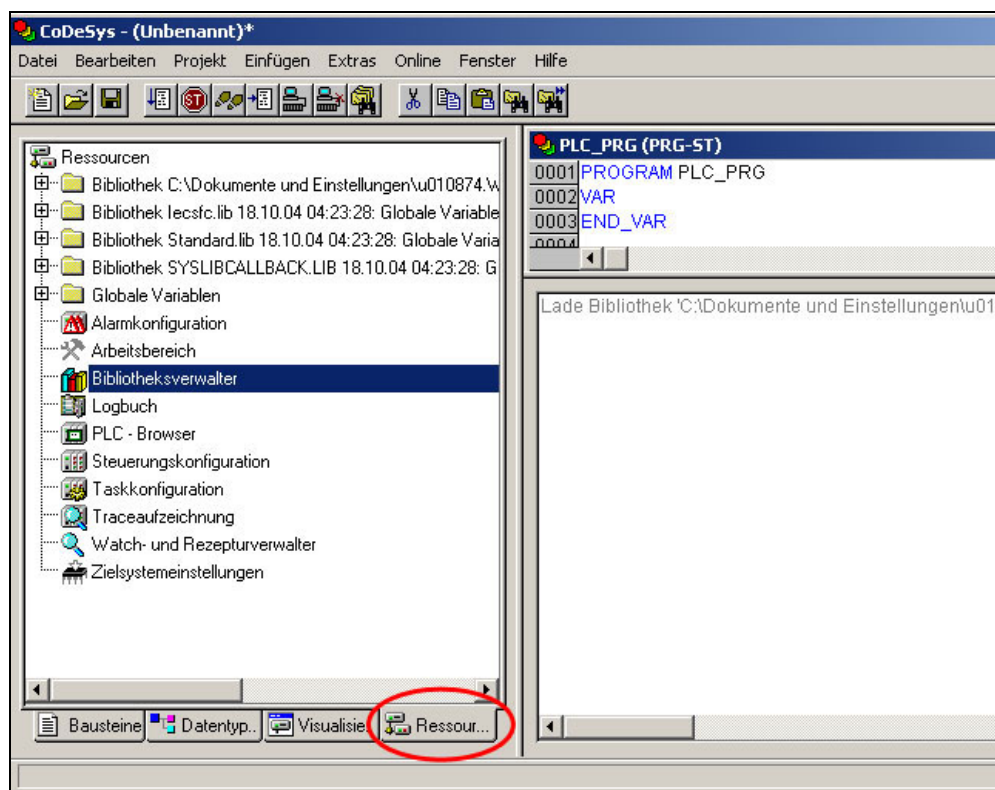


Abbildung 114: Karteireiter „Ressourcen“

7. Klicken Sie im linken Fenster mit einem Doppelklick auf „Bibliotheksverwaltung“.
8. Klicken Sie in der Menüleiste auf **Einfügen** > **Weitere Bibliothek** und wählen Sie mytest.lib.
9. Klicken Sie auf den Karteireiter „Bausteine“.
10. Anschließend rufen Sie die Funktion in CODESYS wie folgt auf:

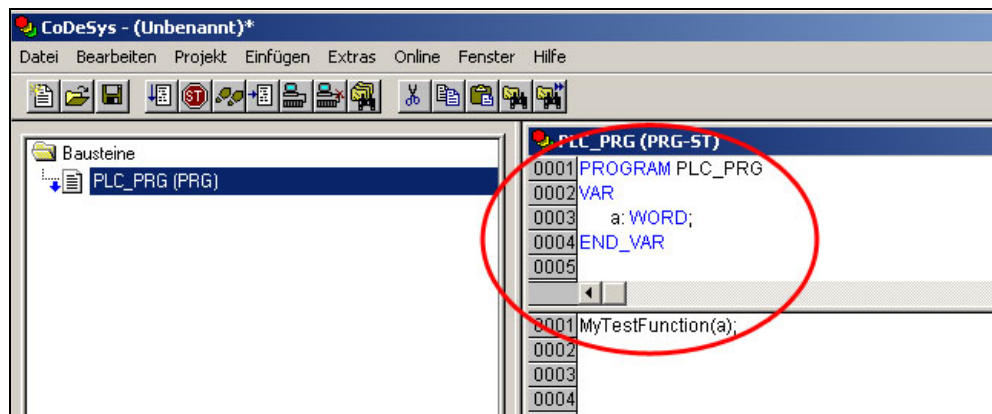


Abbildung 13: Fenster „PLC_PRG(PRG)“

13.2 Besonderheiten

13.2.1 Datentypen

Alle CODESYS-Datentypen können als Übergabeparameter verwendet werden. Dabei werden die CODESYS-Datentypen in C wie folgt interpretiert:

Tabelle 71: Datentypen

CODESYS	C/C++
BOOL	char
BYTE	char
WORD	unsigned short
DWORD	unsigned int
LWORD	unsigned long
SINT	signed char
USINT	unsigned char
INT	short
UINT	unsigned short
DINT	int
UDINT	unsigned int
LINT	long int
ULINT	unsigned long int
REAL	float
LREAL	double
STRING	char[]

13.2.2 Strukturen

Auch Strukturen lassen sich übergeben. Dabei ist es wichtig, dass die Datentypen exakt eingehalten werden. Zudem sind die Strukturen zwingend mit dem Attribut „packed“ zu definieren. Somit würde die folgende CODESYS-Struktur

```
TYPE t_teststruct :  
STRUCT  
    a : BYTE;    b : WORD;  
    c : INT;  
    d : DWORD;  
    e : REAL;  
    f : POINTER TO STRING;  
END_STRUCT  
END_TYPE
```

Abbildung 115: Datei „Beispiel.lib“

in C wie folgt aussehen:

```
struct t_teststruct{           // Codesys-Style:  
    char a;                    // BYTE  
    unsigned short b;         // WORD  
    signed short c;           // INT  
    unsigned int d;           // DWORD  
    float e;                  // REAL  
    char *f;                  // POINTER TO STRING  
} __attribute__((packed));
```

Abbildung 116: Datei „Beispiel.h“

Zu beachten sind in der Datei „Beispiel.h“ die Datentypen und das Attribut „packed“.

Des Weiteren ist für jede in CODESYS erstellte Struktur eine init-Funktion in der Library zu erstellen. Für die Datei „Beispiel.h“ könnte die init-Funktion wie folgt aussehen:

```
char t_teststructinit(struct t_teststruct *pteststruct, char
bRetain)
{
    pteststruct->a = 0;
    pteststruct->b = 0;
    pteststruct->c = 0;
    pteststruct->d = 0;
    pteststruct->e = 0;
    pteststruct->f = NULL;
    return 1;
}
```

Dabei muss sich der Name der Funktion aus dem Namen der Struktur und dem String „init“ zusammensetzen (z. B. `t_teststructinit`). Die Funktion wird beim Starten von CODESYS einmalig aufgerufen. Sie hat als Übergabeparameter einen Pointer auf die Struktur selbst und einen BOOL-Wert, der in diesem Fall nicht relevant ist.

13.2.3 Parameterübergabe per Referenz oder per Value

Es ist möglich, die Parameter mittels Referenz oder per Value zu übergeben. Dabei ist es wichtig, dass die richtige Reihenfolge und die Datentypen der Parameter zur Übergabe in die Funktion verwendet werden. Dazu sind die Datentypen aus der Tabelle im Kapitel „Datentypen“ zu beachten. Wird innerhalb des C-Programms auf falsch deklarierte Variablen zugegriffen, können Speicherzugriffsfehler auftreten. Diese führen zum Löschen (suspendieren) der Task durch das CODESYS-Laufzeitsystem.

13.3 Weitere Anwendungen

Ein Starten von beliebigen Linux-Programmen oder -Skripten ist über die Kapselung in eine C-Funktion möglich.

C-Funktionen können auch z. B. eine init-Funktion enthalten, welche eigene Linux-Threads erzeugt und somit eigenständige Programme enthält. Diese init-Funktionen lassen sich über CODESYS-System-Events, wie z. B. PLC-Start oder PLC-Stopp, aufrufen. Auf diesem Weg können auch komplette Applikationen in einem eigenen Thread gekapselt werden.

In init-Funktionen lassen sich auch Pointer auf gemeinsame Datenstrukturen übergeben, mit denen eine komfortable Datenschnittstelle zwischen CODESYS und C-Applikation ermöglicht wird.

14 Betriebssystem

14.1 Verwendeter Linux-Kernel

Für den I/O-IPC wird ein RT-Preempt Realtime-Kernel verwendet. Hierbei handelt es sich um einen Kernel, der mit dem entsprechenden Echtzeit-Patch versehen wurde. Dieser steht wie auch der Kernel unter GPL im Internet zur Verfügung: <http://www.kernel.org/pub/linux/kernel/projects/rt/>.

Diese Echtzeiterweiterung bringt die folgenden Vorteile:

- Voll prioritätsgesteuerte Prozesse im Echtzeitbereich.
- Möglichkeit zur Verwendung von Prozessen im User-Bereich mit dem CVS-Scheduler („Completely Fair Scheduling“).
- Priorisierung der Interrupt-Verarbeitung.
- Der System-Timer basiert auf dem Dynamic-Tick.
Dadurch sind Reaktionszeiten im I/O-IPC nicht mehr an ein festes Zeitraster gebunden. Infolgedessen kann der I/O-IPC zyklische Prozesse in μs -Bereich verarbeiten.

14.2 Grand Unified Bootloader (GRUB)

Als Bootloader für den I/O-IPC wird der GRUB verwendet. Zum Verändern der Starteinstellungen des Linux-Kernels drücken Sie innerhalb der von Ihnen eingestellten Wartezeit während der Startphase des GRUB eine der folgenden Tasten:

- Eine Taste auf der am I/O-IPC angeschlossenen Tastatur
- Bei einem geöffneten Terminalprogramm eine Taste auf der PC-Tastatur

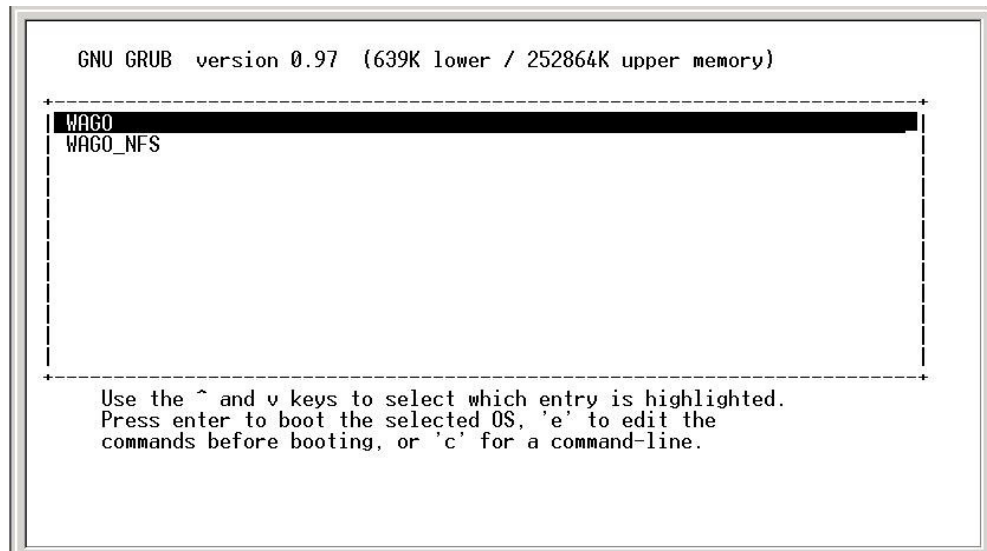


Abbildung 117: Serielle Konsole „Hyperterminal“

Im GRUB haben Sie die Wahl aus zwei Starteinstellungen für das Linux-Dateisystem:

- **WAGO**
Starten des Linux-Dateisystems vom internen Flash-Speicher.
- **WAGO_NFS**
Starten des Linux-Dateisystems von einer zu definierenden, entfernten NFS-Partition.

Zum Ändern der Bootstrings (Übergabeparameter an den Kernel) drücken Sie auf der Tastatur die **E**-Taste. Dadurch können Sie eine feste IP-Adresse unter **IP=** festlegen bzw. unter **VGA=** die Auflösung des am I/O-IPC angeschlossenen Monitors einstellen.

Die Änderung des Startverhaltens wird nicht remanent gespeichert. Für die dauerhafte Speicherung ist im Linux-Dateisystem in der Datei */boot/grub/menu.lst* der Parameter „default“ zu verändern.

14.3 Startablauf von Linux

Nach dem Einschalten des I/O-IPC startet zuerst das BIOS. Falls Sie dort die vom PC bekannten Einstellungen durchführen möchten, drücken Sie die Taste **[Entf]** bzw. **[Del]** auf der am I/O-IPC angeschlossenen Tastatur. Für den I/O-IPC sind an dieser Stelle keine Änderungen notwendig.

Im Anschluss an das BIOS startet der Bootloader GRUB, der den Kernel startet. Während der Hochlaufphase des Kernels wird die gesamte Hardware konfiguriert.

Nach der Hardwarekonfiguration startet der Kernel den ersten Userspace-Prozess (init). Wie bei großen Linux-Distributionen starten durch init die Startskripte in */etc/rc.d/...* in alphabetischer und numerischer Reihenfolge.

Falls Sie eigene Anwenderprogramme starten möchten, können Sie weitere Einträge als Verweis auf Startskripte in diesem Verzeichnis anlegen. Diese Skripte werden automatisch beim Starten des I/O-IPCs ausgeführt.

Mit dem letzten rc.d-Skript wird CODESYS gestartet. Auf dem angeschlossenen Monitor wird die Target-Visualisierung von CODESYS angezeigt. Über **[Alt] + [F2]** der angeschlossenen Tastatur wechseln Sie auf die Linux-Konsole und mittels **[Alt] + [F1]** greifen Sie wieder auf die Target-Visualisierung zu.

Note



Verhalten beim Bootvorgang

Beim Bootvorgang wird eine Überprüfung des Dateisystems (Filesystem check) durchgeführt.

Diese Überprüfung kann die Anlaufzeit des Systems um mehrere Sekunden verzögern.

14.4 Linux-Konsole

Die Linux-Konsolen sind über die Tastatur wie folgt erreichbar:

1. Linux-Konsole	2. Linux-Konsole	3. Linux-Konsole
Target-Visualisierung	Linux	IPC-Configuration-Tool
[Alt] + [F1]	[Alt] + [F2]	[Alt] + [F3]

Wenn Sie einzelne Konsolen deaktivieren wollen, müssen Sie die Datei */etc/inittab* auf dem Gerät modifizieren. Das können Sie mit Hilfe des integrierten Editors *vi* oder über einen FTP-Upload / -Download tun. Die Datei */etc/inittab* enthält die folgenden Zeilen:

```
tty1:2345:respawn:/sbin/getty 38400 tty1
```

```
tty2:23:respawn:/sbin/getty 38400 tty2
```

```
tty3:23:respawn:/etc/init.d/ipcconfig start
```

Die drei Kommandozeilen starten die in der Tabelle beschriebenen drei Konsolen. Kommentieren Sie einzelne Zeilen mit dem Zeichen „#“, wird die entsprechend kommentierte Linux-Konsole während des Systemstarts nicht mehr automatisch ausgeführt.

14.4.1 Zugriff auf die Linux-Konsole

GEFAHR



Passwörter ändern

Die Standard-Passwörter sind in dieser Betriebsanleitung dokumentiert und bieten so keinen hinreichenden Schutz. Ändern Sie die Passwörter entsprechend Ihren Erfordernissen.

Sie können auf die Linux-Konsole über verschiedene Wege zugreifen. Zum einen über Telnet, zum anderen über die RS-232-Schnittstelle. Auch über einen Monitor an der DVI-I-Schnittstelle in Verbindung mit einer USB-Tastatur ist der Zugriff auf die Linux-Konsole möglich.

Im Auslieferungszustand des I/O-IPC sind die folgenden Benutzer eingerichtet:

Tabelle 72: Benutzer für die Linux-Konsole

Name	Passwort
root	wago
admin	wago
user	user
guest	guest

Passwort ändern:

```
passwd [Benutzer]
```

Sie können auch eigene Benutzer anlegen und löschen:

```
sudo adduser [Benutzer]
```

```
sudo deluser [Benutzer] sudo
```

Hinweis



Vorsicht beim Löschen von Benutzern (user)!

Mit `deluser` können Sie auch Benutzer vom Typ **superuser** löschen. Das kann dazu führen, dass Sie anschließend keinen Zugriff mehr auf das Gerät haben. Wollen Sie den Zugriff wieder herstellen, muss das Gerät über einen Firmwaredownload zurückgesetzt werden.

14.4.1.1 Zugriff über Telnet

Um über Telnet auf den I/O-IPC zuzugreifen, verwenden Sie ein Terminalprogramm wie z. B. minicom (unter Linux) oder auch Hyperterminal (unter Windows).

Bei Verwendung des Hyperterminals sind in der Anmeldeoberfläche folgende Einstellungen anzupassen:

Hostadresse: IP-Adresse der verwendeten ETHERNET-Schnittstelle des I/O-IPC
Verbinden über: TCP/IP

Ferner können Sie auch über die Konsole von Linux bzw. von MS-DOS mittels Telnet auf den I/O-IPC zugreifen. Gehen Sie dazu folgendermaßen vor:

1. Verbinden Sie die ETHERNET-Schnittstelle X9 des I/O-IPC über ein ETHERNET-Patchkabel mit Ihrem PC.
2. Öffnen Sie eine Konsole Ihres PC.
3. Geben Sie den Befehl `telnet <IP-ADRESSE des I/O-IPC>` ein.



Abbildung 118: Beispiel mit DOS-Konsole 1

4. Geben Sie Ihren Benutzernamen ein (siehe Kapitel „Zugriff auf die Linux-Konsole“).



Abbildung 119: Beispiel mit DOS-Konsole 2

5. Geben Sie das für Ihren Benutzer zugehörige Passwort ein. Die Linux-Konsole des I/O-IPC öffnet sich im HOME-Verzeichnis (~) des gewählten Benutzers.

14.4.1.2 Zugriff über RS-232-Schnittstelle und Terminalprogramm

Um über die RS-232-Schnittstelle mit Hilfe eines Terminalprogramms auf die Linux-Konsole zuzugreifen, gehen Sie folgendermaßen vor:

1. Wählen Sie im WBM oder über das IPC-Configuration-Tool für die Schnittstelle die Linux-Konsole aus. Siehe dazu Kapitel „Administration“.
2. Verbinden Sie die serielle Schnittstelle des PC über ein Nullmodemkabel mit der seriellen Schnittstelle X6 (12) des I/O-IPC.

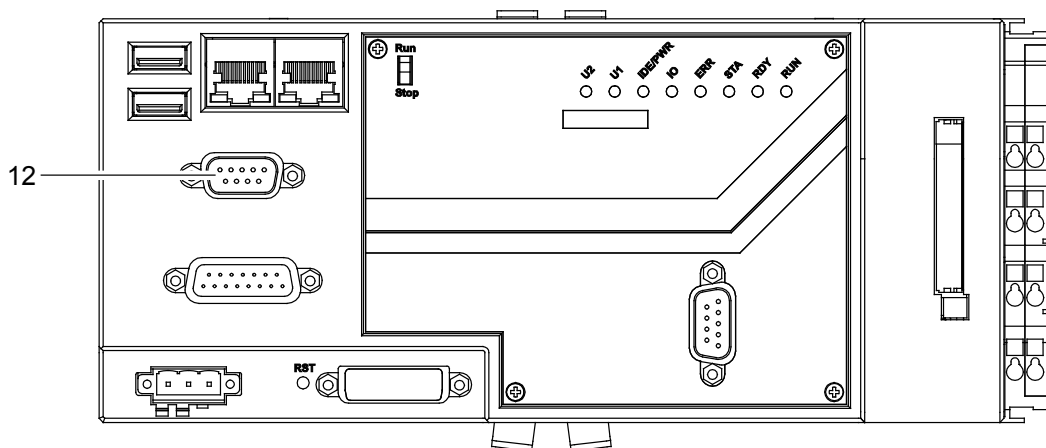


Abbildung 120: RS-232-Schnittstelle X6

3. Öffnen Sie auf Ihrem PC ein Terminalprogramm (Linux: z. B. minicom, Windows: z. B. Hyperterminal).
4. Geben Sie im Terminalprogramm die voreingestellten Kommunikationsparameter der RS-232-Schnittstelle des I/O-IPC ein:
 - Geschwindigkeit: 115200 bit/sek
 - Datenbreite: 8 Bit
 - Parität: keine
 - Stoppbits: 1 Bit
 - Flusssteuerung: keine
5. Es erscheint das Startbild der Linux-Konsole.
6. Geben Sie Ihren Benutzernamen ein (siehe Kapitel „Zugriff auf die Linux-Konsole“).
7. Geben Sie das für Ihren Benutzer zugehörige Passwort ein. Die Linux-Konsole öffnet sich im HOME-Verzeichnis (ˆ) des gewählten Benutzers.

14.4.1.3 Zugriff über Tastatur und Monitor (DVI- und USB-Schnittstelle)

Um über einen an der DVI-I-Schnittstelle angeschlossenen Monitor/Touchscreen und eine mittels USB angeschlossene Tastatur auf die Linux-Konsole zuzugreifen, gehen Sie folgendermaßen vor:

1. Schließen Sie einen Monitor an die DVI-I-Schnittstelle X7 (9) des I/O-IPC an.
2. Schließen Sie eine USB-Tastatur an einem der beiden USB-Schnittstellen X10 (1) oder X11 (2) an.

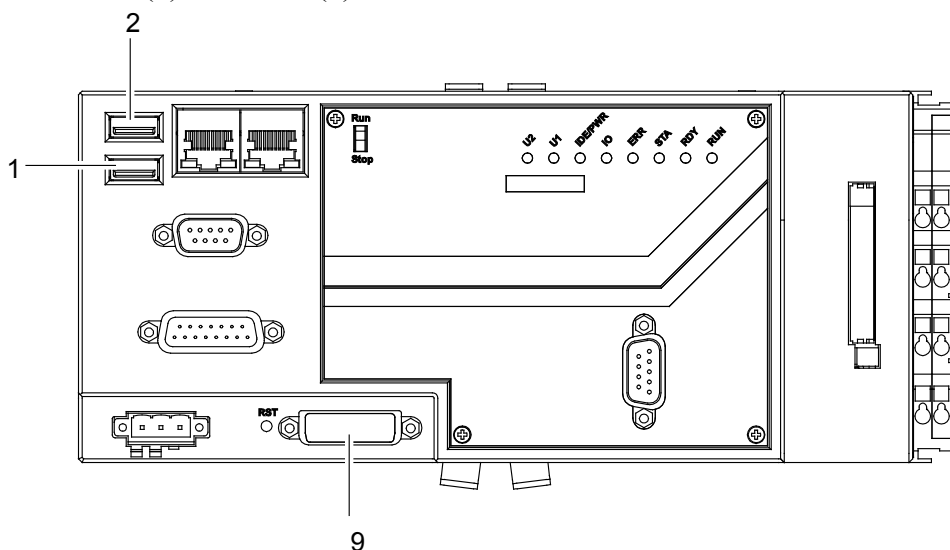


Abbildung 121: DVI-I-Schnittstelle X7 und USB-Schnittstellen X10/11

3. Nach dem Start von Linux erscheint – wenn erstellt – auf dem Monitor das Startbild der Target-Visualisierung.

Hinweis



Auflösung des Monitors/Touchscreen

Wenn nur ein dunkles Bild angezeigt wird, dann stimmt evtl. die Auflösung des Monitors/Touchscreen nicht. Ändern Sie die Auflösung mittels WBM (siehe Kapitel „Seite ‚HMI Settings‘“).

4. Über die Tastenkombination **[Alt] + [F2]** wechseln Sie in die Linux-Konsole.
5. Geben Sie Ihren Benutzernamen ein (siehe Kapitel „Zugriff auf die Linux-Konsole“).
6. Geben Sie das für Ihren Benutzer zugehörige Passwort ein. Die Linux-Konsole des I/O-IPC öffnet sich im HOME-Verzeichnis (ˆ) des gewählten Benutzers.

Über `cat /proc/kmsg` (alternativ `dmesg`) können Sie sich die Startmeldungen von Linux erneut anzeigen lassen.

14.4.2 Installierte Anwendungen

Ausgeliefert wird der I/O-IPC mit einem Basis-Image, welches bereits die wichtigsten Anwendungen im Dateisystem enthält. Folgende Anwendungen sind unter anderem enthalten:

- Bootloader: GRUB
- Dateisystem-Unterstützung für Ext2, Fat
- Konsoleninitialisierung: getty
- FTP-Client/-Server
- Telnet-Client/-Server
- SSH-Client/-Server
- Webserver (lighttpd)
- PHP5
- BootP-/DHCP-Clients
- NFS-Client
- Event-Manager (udev) zum automatischen Einbinden von USB-Speichern
- NTP-Client

14.4.3 Aufbau des Dateisystems

Das Dateisystem des internen Flash-Speichers ist im Auslieferungszustand wie folgt partitioniert:

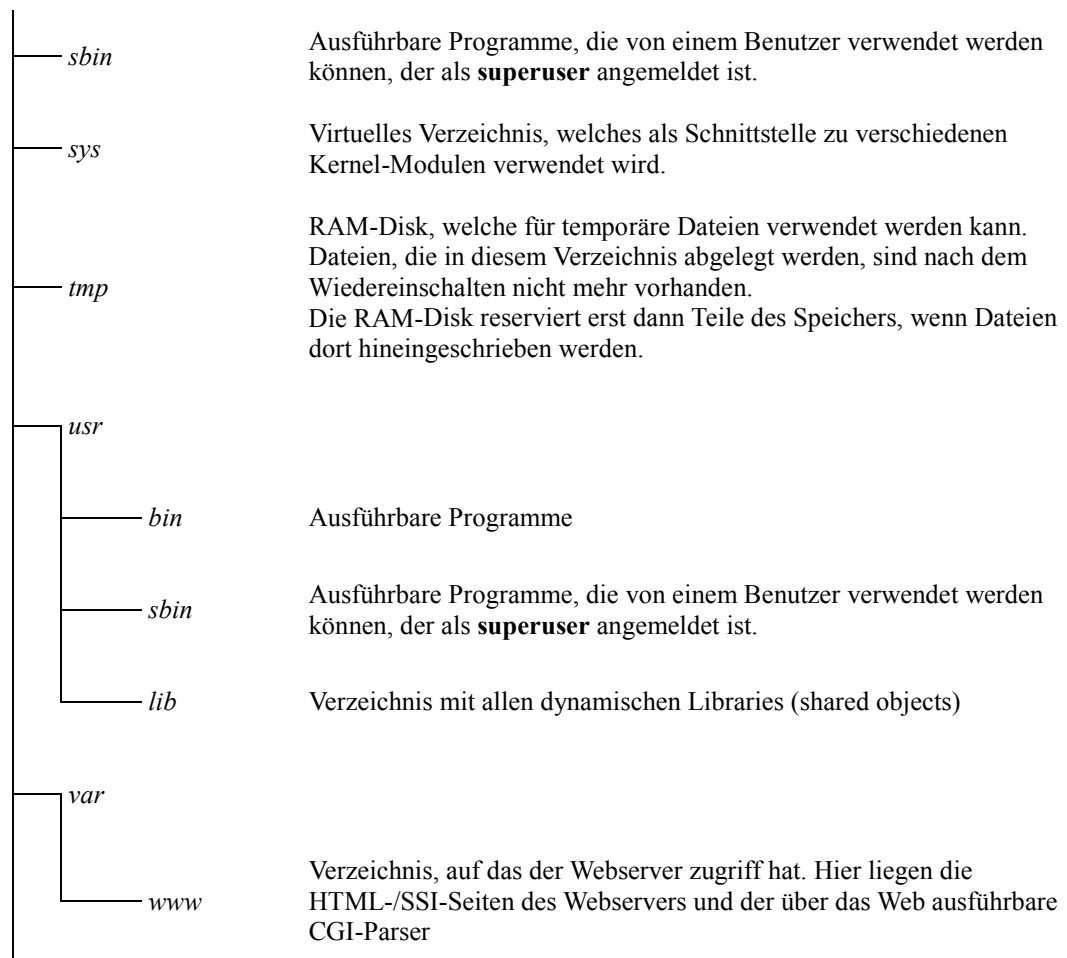
Tabelle 73: Aufbau des Dateisystems

Bezeichnung	Größe
Master-Boot-Record und Bootloader (grub)	ca. 1,5 MB
1. Linux-System-Partition	ca. 40 MB
2. Linux-System-Partition	ca. 40 MB
Home-Partition	ca. 40 MB
Restlicher Flash-Speicher, nicht partitioniert	ca. 400 MB

Dabei enthält das Dateisystem, wie bei modernen Linux-Distributionen üblich, die folgenden Verzeichnisse mit den gängigen Programmen/Dateien:

<i>bin</i>	Ausführbare Programme
<i>boot</i>	Kernel-Image und Konfiguration des Bootloaders.
<i>dev</i>	Gerätetreiberdateien für die gesamte Peripherie des I/O-IPC
<i>etc</i>	Globale Konfigurationsdateien des I/O-IPC
<i>config-tools</i>	WAGO-spezifische Konfigurationstools
<i>network</i>	Konfigurationsdateien für die ETHERNET-Schnittstellen.
<i>init.d</i>	Startskripte
<i>rc.d</i>	Links auf Startskripte. Alle mit Sxx_ benannten Skripte werden beim Booten des I/O-IPC ausgeführt.
<i>home</i>	
<i>guest</i>	Home-Verzeichnis eines Benutzers, der als guest angemeldet ist.
<i>user</i>	Home-Verzeichnis eines Benutzers, der als user angemeldet ist.
<i>codesys</i>	CODESYS-Verzeichnis: Dies beinhaltet ggf. die Boot-Applikation und die benötigten Dateien der Target- bzw. der Webvisualisierung.
<i>lib</i>	Verzeichnis mit allen dynamischen Libraries (shared objects).

<i>media</i>	Verzeichnis, welches als mountpoint für die über automount (udev) eingebundenen Geräte (z. B. USB-Speicher) verwendet wird. Geräte die mit dem Dateisystem FAT formatiert sind, werden in /media in einem Unterverzeichnis mit Ihrem Partitionsnamen eingebunden (weitere Details siehe Kapitel „SysLibFile, SysLibDir, SysLibFileAsync“)
<i>mnt</i>	mountpoint , der vom Benutzer verwendet werden kann. Im Auslieferungszustand hat dieses Verzeichnis keine Funktion.
<i>proc</i>	Virtuelles Verzeichnis, welches Informationen aus dem Kernel bereitstellt.
<i>root</i>	Home-Verzeichnis der Benutzer root und admin



14.4.4 Installierte Shell (BASH)

Für den I/O-IPC ist eine BASH (Bourne-Again-Shell) installiert, welche die Builtin-Kommandos wie z. B.: `cd` beinhaltet. Außerdem stellt die BASH die Umgebungsvariablen zur Verfügung und ermöglicht das Navigieren im Dateisystem ebenso wie das Starten von Programmen.

14.4.5 Busybox und andere Hilfsprogramme

Busybox vereint viele Programme aus den Standard-Linux-Distributionen in einem Programm, um den Speicherbedarf im Dateisystem zu minimieren. Das Programm wird nur über symbolische Links aufgerufen. Busybox wertet den aufrufenden Namen aus und kann so verschiedene Funktionen in einem Programm zusammenfassend realisieren.

Unter anderem stellt das Programm Busybox folgende Funktionen zur Verfügung:

- `mount`
Einbinden von Laufwerken
Durch die Namensvergabe (PC: Format > Volumenbezeichnung) eingebundener Speichermedien (z. B. CF-Karte, USB-Speicher), werden diese im Dateisystem (`/media`) angezeigt.
- `reboot`
Neustart des I/O-IPC
- `ifconfig`
Zeigt die aktuellen Einstellungen der ETHERNET-Schnittstelle an. Zum Ändern dieser Einstellungen verwenden Sie das WBM oder IPC-Configuration-Tool.
- `rmdir`
Verzeichnis löschen

Hinweis



Kompilieren der Busybox

Beim Kompilieren von Busybox können Sie die zu installierenden Programme auswählen. Dadurch passen Sie die Programmgröße entsprechend der benötigten Funktion an. Beim I/O-IPC sind für das System alle notwendigen Funktionen implementiert.

Zudem sind weitere Programme auf dem I/O-IPC installiert wie z. B. `cp` oder `ftp`. Die im Kapitel „Installierte Anwendungen“ aufgeführten Programme gehören ebenso dazu wie folgende Hilfsprogramme (Beispiele):

- `htop`, `top`
Programm zur Anzeige der Prioritäten und der verbrauchten Ressourcen einzelner Prozesse.
- `sed`
Hilfsprogramm zum einfachen Parsen von Text-/Konfigurationsdateien.
- `gdbserver`
Remote-Debugger
- `cyclictest`
Messprogramm zum Erfassen der Echtzeitfähigkeit des Systems.
- `zip`, `unzip`
Zum Packen bzw. Entpacken von Zip-Archiven.

14.5 Treiber für spezielle Hardwareteile

Durch den echtzeitfähigen Kernel sind Userspace-I/O-Treiber (UIO) realisierbar. Dabei wird über eine Speicher-Mapping-Funktionalität direkt vom Userspace auf die Hardware zugegriffen. Dadurch greifen Sie auf die Prozessabbilder der angeschlossenen Busklemmen zu sowie auf andere Hardware-Bereiche.

14.6 Einbinden eines USB-Druckers

An den USB-Schnittstellen können Sie einen Drucker für ASCII-Texte anschließen.

Um beispielsweise die Wörter Test0, Test1 und Test2 auszudrucken, geben Sie folgende Befehle ein:

```
echo -e "\n\nTest0\n" >/dev/lp0  
echo -e "\n\nTest1\n" >/dev/lp0  
echo -e "\n\nTest2\n" >/dev/lp0
```

Gedruckt erscheinen die Wörter Test auf einer Seite wie folgt:

```
Test0  
      Test1  
                Test2
```

14.7 Installierte Dienste der ETHERNET-Schnittstelle

Für die ETHERNET-Schnittstelle sind im Auslieferungszustand des I/O-IPC verschiedenste Client-/Server-Dienste aktiviert. Nachfolgend ist eine Auswahl an installierten Diensten aufgeführt:

- **Telnet-Server**
Der Telnet-Server ermöglicht die Verbindung mehrerer Teilnehmer eines Netzwerks mit der Linux-Konsole des I/O-IPC.
- **Telnet-Client**
Dieser dient zum Zugriff über ein Netzwerk auf die Konsole eines entfernten Telnet-Servers.
- **FTP-Server**
Der FTP-Server ermöglicht den Zugriff mehrerer Teilnehmer eines Netzwerks auf das Dateisystem des I/O-IPC.
- **FTP-Client**
Dieser ermöglicht den Austausch von Dateien mit fernen FTP-Servern.
- **Webserver**
Teilnehmer im Netzwerk können mit einem Internet-Browser Informationen über die Einstellungen des I/O-IPC abrufen und diesen konfigurieren.
- **NTP-Client**
Der NTP-Client erlaubt die Abfrage der genauen Uhrzeit von einem NTP-Server.
- **NFS-Client**
Dieser dient zum Einbinden von freigegebenen Netzwerklaufwerken von NFS-Servern.
- **SNMP-Server**
Durch den SNMP-Server ist es möglich, den I/O-IPC von einem PC zu überwachen und zu steuern. Ebenfalls besteht die Möglichkeit mittels SNMP Daten mit dem SPS-Programm auszutauschen (Siehe Anhang, Kapitel „WagoLibNetSnmp.lib“).

14.7.1 Telnet-Server (telnetd)

Der Telnet-Server ist im Auslieferungszustand des I/O-IPC aktiviert. Der telnetd-Daemon wird bei einer entsprechenden Anfrage über das ETHERNET aktiviert. Dadurch wird eine neue Linux-Konsole mit dem Zugang über Telnet erzeugt. Der Telnet-Daemon wird mit dem Skript `/etc/rc.d/S07_telnetd` gestartet bzw. gestoppt.

Zur Anmeldung am Telnet-Server geben Sie Ihren Benutzernamen und das dazugehörige Passwort ein. Die Linux-Konsole des I/O-IPC öffnet sich und das HOME-Verzeichnis des gewählten Benutzers wird angezeigt.

14.7.2 FTP-Server (pure-ftpd)

Das „File Transfer Protocol“ dient dazu, Dateien zwischen PC und dem I/O-IPC auszutauschen. Auf dem PC muss dabei nicht Linux installiert sein, da auch Windows FTP-Client-Funktionen bereitstellt.

Im Auslieferungszustand des I/O-IPC ist der FTP-Server aktiviert. Der FTP-Daemon wird mit dem Skript `/etc/rc.d/S09_pureftd` gestartet bzw. gestoppt und wartet danach auf entsprechende Anfragen.

Austausch der FTP-Dateien zwischen PC und I/O-IPC über ETHERNET

Um über ETHERNET mit FTP Dateien zwischen einem PC und dem I/O-IPC auszutauschen, ist ein FTP-Client nötig. Dazu können Sie sowohl einen Internet-Browser (Internet Explorer) als auch ein FTP-Programm (z. B. Filezilla, DOS-Konsole, Linux-Konsole) verwenden. Verwenden Sie für die FTP-Verbindung Port 21. Bei manchen FTP-Programmen, wie beispielsweise „Filezilla“, ist dieser nachträglich einzutragen.

Um beispielsweise den Windows-Internet-Explorer als FTP-Client zu nutzen, geben Sie folgende Adresse in die Adresszeile des Windows-Explorers ein:

```
ftp://username:password@hostname.  
Beispiel: ftp://user:user@192.168.1.17.
```

Informationen zu Benutzern und Passwörtern der Linux-Konsole erhalten Sie im Kapitel „Zugriff auf die Linux-Konsole“.

Um beispielsweise die DOS-Konsole als FTP-Client zu nutzen, geben Sie in der Konsole folgenden Befehl ein:

```
ftp <hostname/IP>
```



```
C:\WINNT\system32\cmd.exe - ftp 192.168.1.17  
Microsoft Windows 2000 [Version 5.00.2195]  
<C> Copyright 1985-2000 Microsoft Corp.  
  
U:\>ftp 192.168.1.17  
Verbindung mit 192.168.1.17 wurde hergestellt.  
220 FTP server ready.  
Benutzer (192.168.1.17:(none)): root  
331 User root OK. Password required  
Kennwort:  
230 OK. Current directory is /home  
ftp> _
```

Abbildung 122: DOS-Konsole

Nachdem Sie sich angemeldet haben, können Sie mit `help` die Hilfe aufrufen, welche die verfügbaren Befehle der FTP-Konsole des Slaves beschreibt. Zum Beispiel:

```
put Datei.html // Schreibt die Datei in den I/O-IPC  
get Bild.gif // Liest die Datei vom I/O-IPC  
cd/ // Wechselt in das Root-Verzeichnis  
help // Zeigt alle implementierten Kommandos an
```

14.7.3 NFS-Server

Das NFS („Network File System“) ist ein Dienst, der den netzwerkübergreifenden Zugriff auf Dateien erlaubt. Wollen Sie z. B. das lokale Verzeichnis */home* im Netzwerk freigeben, dann fügen Sie in der Konfigurationsdatei */etc/exports* folgende Zeile hinzu:

```
/home *(rw,sync,all_squash,anonuid=<uid>,anongid=<gid>)
```

Für *<uid>* und *<gid>* geben Sie die Linux-Benutzernummer und Gruppennummer ein, über die Sie angemeldet sind. Diese Nummern lassen sich wie folgt ermitteln:

```
> id  
uid=0(root) gid=0(root)
```

Für dieses Beispiel lautet die Zeile der Datei */etc/exports* folgendermaßen:

```
/home *(rw,sync,all_squash,anonuid=0,anongid=0)
```

14.7.4 FTP-Client

Der FTP-Client ermöglicht Dateien von einem FTP-Server zu laden bzw. zu schreiben. Der FTP-Client ist im Verzeichnis */bin* installiert und steht so jedem User zur Verfügung. Verwenden Sie für die FTP-Verbindung Port 21.

Bedienung des FTP-Clients

Um den FTP-Client zu nutzen, muss auf einem fernen PC ein FTP-Server mit einem bekannten Benutzer für den FTP-Zugang vorhanden sein. Zum Starten des FTP-Clients geben Sie folgenden Befehl ein:

```
ftp <IP/hostname>
```

Beispiel: `ftp 192.168.1.11`

Der FTP-Server fragt Benutzer und Passwort ab. Nach erfolgreicher Anmeldung können Sie Befehle auf dem Server ausführen. Mit `help` fragen Sie die verfügbaren Befehle des Servers ab. Der Server liefert dann eine Liste aller verfügbaren FTP-Befehle. Eine Beschreibung eines Befehls erhalten Sie mit `help <Befehl>`, wie z. B. `help cd`.

14.7.5 Webserver (lighttp)

Lighttp ist ein Programm unter GPL und zeichnet sich besonders durch seine Schnelligkeit aus. Die Syntax der Konfigurationsdatei ist am Apache-Webserver angelehnt wodurch er einfach einzurichten ist. Der Webserver verfügt auch über eine PHP5-Unterstützung, die bereits für die Webseiten des WBM verwendet wird.

Der Webserver ist im Auslieferungszustand des I/O-IPC aktiviert. Er stellt über das Web-based Management eine grafische Oberfläche zur Verfügung, über die Sie den I/O-IPC konfigurieren können. Siehe dazu Kapitel „Konfiguration mittels Web-based Management (WBM)“.

Die bereits hinterlegten Webseiten liegen im Verzeichnis `/var/www`. Im Verzeichnis `/var/www/cgi-bin/` befindet sich ein CGI-Parser, der es ermöglicht, dynamische Webseiten zu erzeugen. Beispiele, die den CGI-Parser verwenden, liegen im Verzeichnis `/var/www/wbm` und realisieren das WBM zur Konfiguration des I/O-IPC.

14.7.6 NTP-Client

Der I/O-IPC stellt eine NTP-Client-Funktionalität durch das Programm `ntpclient` bereit. Mit NTP kann von einem entfernten NTP-Server die Uhrzeit abgefragt werden. Verwenden Sie für die NTP-Verbindung Port 123.

Für im Internet anzusprechende NTP-Server sind Routing und Firewall entsprechend einzustellen. Bei NTP wird die Zeit in einem 64-Bit-Wert übertragen und hat so eine Auflösung von ca. 0,25 ns. Die Genauigkeit der Zeitübertragung wird im Internet mit +/-10 ms und in lokalen Netzwerken mit bis zu +/-200 µs angegeben.

Die Konfiguration und Aktivierung/Deaktivierung des NTP-Clients führen Sie über das WBM durch.

14.7.7 NFS-Client

Ein NFS-Client ist im Kernel integriert, wodurch es ermöglicht wird, entfernte Laufwerke dem eigenen Dateisystem hinzuzufügen. Um ein Verzeichnis eines fernen Systems einzubinden, wird es wie eine Partition einer Festplatte mit dem Befehl `mount` der Linux-Verzeichnisstruktur zugeordnet. Um den NFS-Dienst zu nutzen, muss auf dem fernen PC ein NFS-Server mit einem entsprechend freigegebenen Verzeichnis vorhanden sein. Das Einbinden des entfernten Verzeichnisses in das Dateisystem des I/O-IPC wird mit dem folgenden Befehl durchgeführt:

```
mount -t nfs -o nolock <IP/hostname>:/<Verzeichnis>  
/<lokales Verzeichnis>
```

Beispiel: `> mount -t nfs -o nolock 192.168.1.12:/targetfs /mnt`

Im Auslieferungszustand ist das Laufwerk `/mnt` vorhanden. Es dient zur Einbindung fremder Laufwerke. Der Zugriff auf das über NFS eingebundene Laufwerk wird wie bei einem Zugriff auf ein lokales Verzeichnis vorgenommen. Soll der I/O-IPC z. B. beim Systemstart automatisch Laufwerke einbinden, können Sie diese im Verzeichnis `/etc/rc.d` über ein Skript aufrufen.

14.7.8 SNMP-Agent

Das „Simple Network Management Protocol“ dient zum Überwachen und Steuern von Netzwerkkomponenten. Bei der Kommunikation via SNMP kommen SNMP-Manager (Clients) und SNMP-Agenten (Server) zum Einsatz.

Der auf einem PC installierte Manager steuert über ein TCP/IP-Netzwerk den auf dem I/O-IPC installierten Agenten. Er kann sowohl Anfragen an den I/O-IPC senden als auch Antworten von ihm erhalten. Der Agent dient zum Erfassen und Übertragen von Gerätedaten (Name, Status, OIDs usw.) dienen.

Die Daten eines Gerätes, auf die der Agent zugreift oder modifizieren kann, heißen SNMP-Objekte. Die SNMP-Objekten werden über die MIB-Datei (MIB bedeutet „Management Information Base“) dem Manager bekannt gegeben. Für die eindeutige Adressierung der einzelnen Infos innerhalb einer MIB sind OIDs zuständig (Object Identifier).

Die Konfiguration von dem SNMP-Agenten des I/O-IPC führen Sie über das Web-based Management (WBM) durch.

Das SNMP wird in der Version 1, 2c und 3 unterstützt. Im Auslieferungszustand des I/O-IPC ist der SNMP-Agent aktiviert. Bei SNMP in der Version 1 und 2c handelt es sich um einen gerätegebundenen Nachrichtenaustausch. Dazu muss die IP-Adresse des Managers angegeben werden. Mit dieser eingestellten IP-Adresse kann ein Manager mit dem Netzwerk-Teilnehmer kommunizieren. In der Version 3 vom SNMP ist der Nachrichtenaustausch an Benutzer gebunden. Jedes Gerät, welches die über das WBM eingestellten Passwörter kennt, kann Werte aus dem I/O-IPC lesen bzw. schreiben. Bei SNMPv3 können die Nutzdaten der SNMP-Nachrichten zudem auch verschlüsselt übertragen werden. So können die

angefragten und zu schreibenden Werte nicht im ETHERNET mitgehört werden. Deshalb wird SNMPv3 häufig in sicherheitsrelevanten Netzwerken verwendet.

Zum Anlegen kundenspezifischer Variablen (OID) steht Ihnen die CODESYS-Bibliothek WagoLibNetSnmplib zur Verfügung. Ausführliche Informationen zu den Datenpaketen, die über SNMP die Kommunikation ermöglicht erhalten Sie im Anhang, Kapitel „WagoLibNetSnmplib“.

15 Diagnose

15.1 Betriebs- und Statusmeldungen

In der unten stehenden Tabelle sind alle Betriebs- und Statusmeldungen des I/O-IPC beschrieben, die durch die LEDs angezeigt werden:

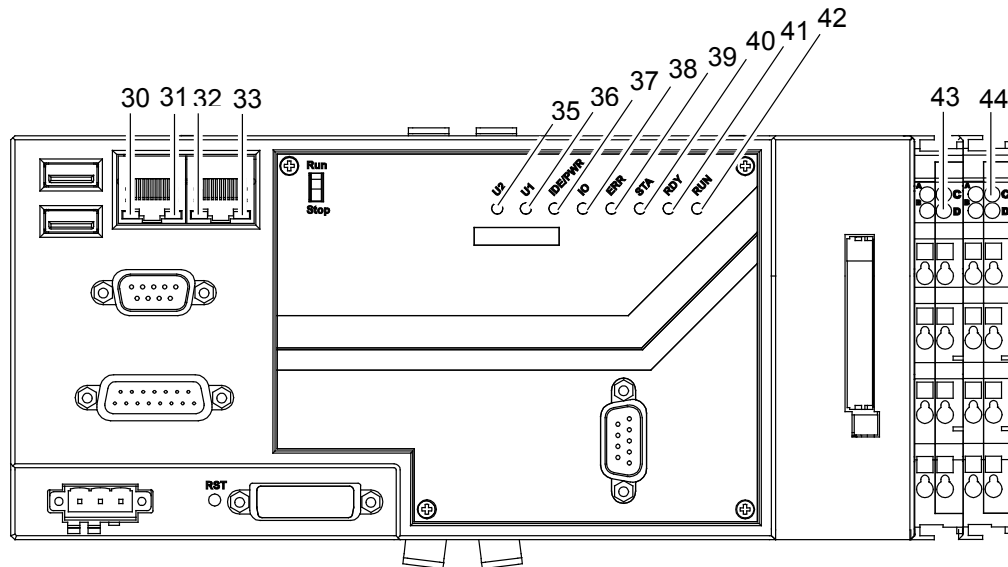


Abbildung 123: Kennzeichnung der LEDs

Tabelle 74: Betriebs- und Statusmeldungen des I/O-IPC

Position	LED	Farbe/Status	Ursache	Erläuterung/Abhilfe
30/32	ACT/LNK	Aus	I/O-IPC hat keine Verbindung zum Ethernet-Netzwerk.	Überprüfen Sie die Verkabelung der ETHERNET-Schnittstellen X8/X9.
		Gelb	Verbindung zum ETHERNET-Netzwerk ist vorhanden.	-
		Gelb blinkend	Datenaustausch über das ETHERNET-Netzwerk findet statt.	-
31/33	Speed	Grün	Übertragungsgeschwindigkeit 100 Mbit/s	Übertragungsgeschwindigkeit wird von den Teilnehmern des ETHERNET-Netzwerks definiert.
		Aus	Übertragungsgeschwindigkeit 10 Mbit/s	
35	U2	Grün, rot, gelb, aus	-	Vom Anwender frei programmierbare LED (mittels der MiscLib.lib).
36	U1	Grün, rot, gelb, aus	-	Vom Anwender frei programmierbare LED (mittels der MiscLib.lib).

Tabelle 74: Betriebs- und Statusmeldungen des I/O-IPC

Position	LED	Farbe/Status	Ursache	Erläuterung/Abhilfe
37	IDE/PWR	Grün	24-V-Versorgungsspannung ist am I/O-IPC vorhanden.	-
		Rot	24-V-Versorgungsspannung ist am I/O-IPC vorhanden und es wird auf den internen Flash-Speicher oder auf die CF-Karte zugegriffen.	-
38	IO	- Grün-rot blinkend - aus	Der I/O-IPC befindet sich in der Startphase.	-
		Grün blinkend	Keine Störung am I/O-IPC. Geschwindigkeitsanzeige zur Aktualisierung des Klemmenbusses. Ohne CODESYS-Anwendung ist nur ein kurzes Blinken sichtbar.	-
		Rot blinkend	Anzeige von Störmeldungen durch den Blinkcode.	Werten Sie den Fehlercode aus. Detaillierte Informationen dazu entnehmen Sie bitte Kapitel „Diagnosemeldungen (I/O-LED)“.
41	RDY	Gelb	-	I/O-IPC ist betriebsbereit.
		Grün blinkend (unregelmäßig)	Es liegt ein Systemfehler im CANopen-Controller vor.	Führen Sie einen Neustart des I/O-IPC durch. Tritt diese Störung weiterhin auf, kontaktieren Sie den WAGO-Support.
42	RUN	Grün	Der I/O-IPC hat min. eine Verbindung zu einem konfigurierten Slave aufgebaut.	-
		Grün blinkend (5 Hz)	Der Slave befindet sich im Pre-Operational-Zustand.	-
		Grün blinkend (unregelmäßig)	Es liegt eine fehlende oder fehlerhafte Konfiguration vor oder ein Zeitfehler des Host-Watchdog ist aufgetreten.	Überprüfen Sie die CODESYS-Konfiguration.

Tabelle 75: Betriebs- und Statusmeldungen des I/O-IPC

Position	LED	Farbe/Status	Ursache	Erläuterung/Abhilfe
43	Klemmenbus-schnittstelle	Aus	-	Leuchtet die LED, dann haben Sie die Versorgungsspannung nicht korrekt angeschlossen. Verwenden Sie zur Einspeisung die Potentialeinspeise- oder Filterklemme. Siehe dazu Kapitel „Versorgungsspannung anschließen“.
44	Potentialeinspeiseklemme 750-602, LED C	Grün	24-V-Versorgungsspannung an den Leistungskontakten vorhanden.	-
		Aus	24-V-Versorgungsspannung an den Leistungskontakten nicht vorhanden.	Schließen Sie die Versorgungsspannung an.
	Optionale Filterklemme 750-626, LED A	Grün	24-V-Versorgungsspannung vorhanden.	-
		Aus	24-V-Versorgungsspannung nicht vorhanden.	Schließen Sie die Versorgungsspannung an.
	LED C	Grün	24-V-Versorgungsspannung an den Leistungskontakten vorhanden.	-
		Aus	24-V-Versorgungsspannung an den Leistungskontakten nicht vorhanden.	Schließen Sie die Versorgungsspannung an.

Tabelle 76: Betriebs- und Statusmeldungen der „ERR“- und „STA“-LEDs

Position	LED	Farbe/Status	Ursache	Erläuterung/Abhilfe
39	ERR	Aus	CANopen-Master ist betriebsbereit Telegramme zu senden oder zu empfangen.	-
		Aus	CANopen-Master sendet ein Telegramm.	-
		Rot	Ein Kommunikationsproblem zwischen dem CANopen-Master und mindestens einem CANopen-Slave ist aufgetreten.	-
		Aus	Der CANopen-Master ist noch nicht konfiguriert.	
		Rot	Der CANopen-Master ist noch nicht betriebsbereit oder kein Slave ist konfiguriert.	
40	STA	Aus	CANopen-Master ist betriebsbereit Telegramme zu senden oder zu empfangen.	
		Gelb blinkend (unregelmäßig)	CANopen-Master sendet ein Telegramm.	
		Gelb blinkend (unregelmäßig)	Ein Kommunikationsproblem zwischen dem CANopen-Master und mindestens einem CANopen-Slave ist aufgetreten.	
		Gelb	Der CANopen-Master ist noch nicht konfiguriert.	
		Gelb	Der CANopen-Master ist noch nicht betriebsbereit oder kein Slave ist konfiguriert.	

15.2 Diagnosemeldungen (I/O-LED)

Dieses Kapitel beschreibt detailliert die I/O-LED (38).

Diese zeigt Diagnosemeldungen des I/O-IPC in Form eines Blinkcodes an. Die Anzahl der Blinkimpulse weist auf Fehlercodes und Fehlerargumente hin, die der genauen Fehlerfindung dienen. Siehe dazu die folgenden Kapitel.

Bei mehreren aufgetretenen Diagnosemeldungen blinkt immer die als erste aufgetretene Diagnosemeldung, bis diese behoben ist. Erst im Anschluss daran wird eine evtl. weitere Diagnosemeldung angezeigt.

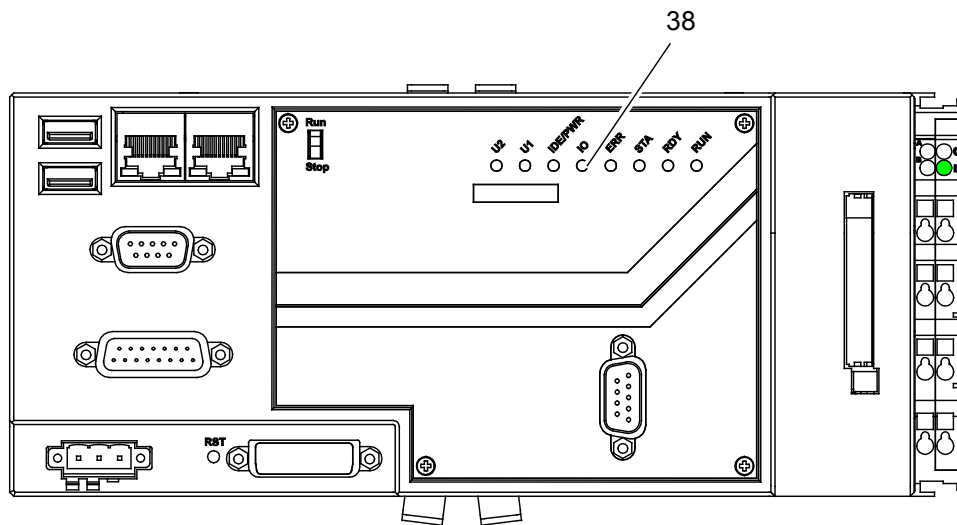


Abbildung 124: Anzeige der Blinkcodes durch die I/O-LED

15.2.1 Ablauf der Blinksequenz

Eine Diagnose (Störung) wird immer zyklisch mit drei Blinksequenzen dargestellt:

1. Die erste Blinksequenz (flackern) leitet die Störmeldung ein.
2. Nach einer Pause von ca. 1 Sekunde erscheint die zweite Blinksequenz. Die Anzahl der Blinkimpulse gibt den **Fehlercode** an, der die Art des Fehlers beschreibt.
3. Nach einer weiteren Pause erscheint die dritte Blinksequenz. Die Anzahl der Blinkimpulse gibt das **Fehlerargument** an, welches ergänzende Fehlerbeschreibungen liefert, z. B. an welchen der am I/O-IPC angeschlossenen Busklemmen ein Fehler vorliegt.

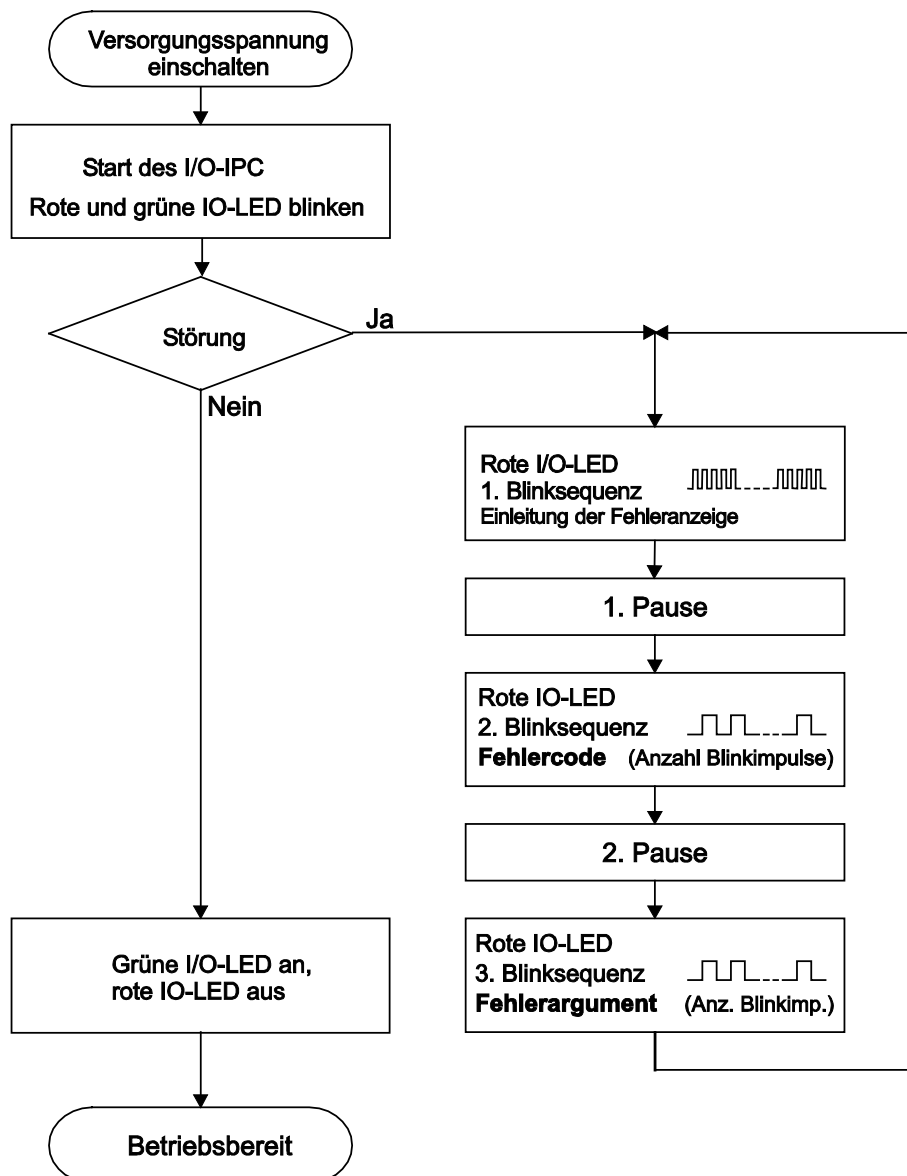


Abbildung 125: Ablaufdiagramm der Blinksequenz

15.2.2 Beispiel einer Diagnosemeldung mittels Blinkcode

Folgendes Beispiel verdeutlicht die Darstellung einer Diagnosemeldung mittels Blinkcode. Es wird ein Datenfehler am Klemmenbus angezeigt, der durch das Entfernen einer Busklemme verursacht wird, die sich an der 6. Position des I/O-IPC befindet.

Einleitung der Startphase

1. Die I/O-LED beginnt mit der Einleitung der Startphase: Ein Zyklus von ca. 10 Hz (10 Blinkzeichen/Sekunde).
2. Es folgt eine Pause von ca. einer Sekunde.

Fehlercode 4: Datenfehler am Klemmenbus

3. Die I/O-LED blinkt 4 Zyklen von ca. 1 Hz.
4. Es folgt eine Pause von ca. 1 Sekunde.

Fehlerargument 5: Busklemme auf dem 6. Steckplatz

5. Die I/O-LED blinkt 5 Zyklen von 1 Hz.
Dies bedeutet, dass am Klemmenbus nach der 5ten Busklemme eine Unterbrechung aufgetreten ist.
6. Der Blinkcode startet mit dem Flackern die erneute Einleitung der Startphase. Bei nur einer Störung wiederholt sich dieser Ablauf.

15.2.3 Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung

Dieses Kapitel beschreibt die durch die I/O-LED als Blinkcode ausgegebenen Diagnosen.

Lassen sich die nachfolgenden Diagnosen nicht mit den angegebenen Maßnahmen beseitigen, kontaktieren Sie bitte den WAGO-Support. Teilen Sie diesem den Blinkcode mit, der ausgegeben wird.

Tel.: +49 571 887 555
Fax: +49 571 887 8555
E-Mail: support@wago.com

Tabelle 77: Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung

Fehlerargument	Ursache	Beseitigung
Fehlercode 1: Hardware- und Konfigurationsfehler		
-	Ungültige Parameter-Prüfsumme des Klemmenbuscontrollers (Klemmenbusschnittstelle)	<ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung des I/O-IPC ab und wechseln Sie ihn aus. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
1	Während der Inlinecode-Generierung hat der interner Pufferspeicher die max. Datenmenge überschritten	<ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung des I/O-IPC ab. - Reduzieren Sie die Anzahl der Busklemmen. - Schalten Sie die Versorgungsspannung wieder ein.
2	Busklemme(n) mit nicht-unterstütztem Datentyp	<p>Aktualisieren Sie die Firmware des I/O-IPC. Bleibt der Fehler bestehen, liegt ein Fehler an einer Busklemme vor. Ermitteln Sie diese wie folgt:</p> <ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung aus. - Platzieren Sie die Endklemme in der Mitte der angeschlossenen Busklemmen. - Schalten Sie die Versorgungsspannung wieder ein. - Falls die I/O-LED noch rot blinkt, schalten Sie die Versorgungsspannung erneut aus und platzieren Sie die Endklemme in der Mitte der ersten Hälfte der Busklemmen (zum I/O-IPC hin). - Wenn die LED nicht mehr blinkt, schalten Sie die Versorgungsspannung ab und platzieren Sie die Endklemme in der Mitte der zweiten Hälfte der Busklemmen (weg vom I/O-IPC). - Schalten Sie die Versorgungsspannung wieder ein. <p>Wiederholen Sie diese Prozedur so oft, bis Sie die defekte Busklemme ermittelt haben. Tauschen Sie diese anschließend aus.</p>

Tabelle 77: Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung

Fehlerargument	Ursache	Beseitigung
3	Unbekannter Modultyp des Flash-Programmspeichers	- Schalten Sie die Versorgungsspannung des I/O-IPC ab und wechseln Sie ihn aus. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
4	Fehler beim Beschreiben des Flash-Speichers aufgetreten	- Schalten Sie die Versorgungsspannung des I/O-IPC ab und wechseln Sie ihn aus. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
5	Fehler beim Löschen eines Flash-Sektors aufgetreten	
6	Die Busklemmenkonfiguration nach einem Klemmenbus-Reset stimmt nicht mit der nach dem letzten Start des I/O-IPC überein.	Starten Sie den I/O-IPC neu, indem Sie - die Versorgungsspannung abschalten und anschließend wieder einschalten oder - die Reset-Taste auf dem I/O-IPC drücken.
7	Fehler beim Beschreiben des seriellen EEPROM aufgetreten	- Schalten Sie die Versorgungsspannung des I/O-IPC ab und wechseln Sie ihn aus. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
8	Unzulässige Hardware-/Firmware-Kombination	
9	Ungültige Prüfsumme im seriellen EEPROM	
10	Initialisierung des seriellen EEPROM fehlgeschlagen	
11	Fehler beim Lesezugriff auf dem seriellen EEPROM aufgetreten	- Schalten Sie die Versorgungsspannung des I/O-IPC ab und reduzieren Sie die Anzahl der Busklemmen. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
12	Zeit für Zugriff auf dem seriellen EEPROM überschritten	- Schalten Sie die Versorgungsspannung des I/O-IPC ab und wechseln Sie ihn aus. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
14	Maximale Anzahl an Gateway- oder Mailboxklemmen überschritten	- Schalten Sie die Versorgungsspannung des I/O-IPC ab. - Reduzieren Sie die Anzahl der Gateway- oder Mailboxklemmen. - Schalten Sie anschließend die Versorgungsspannung wieder ein.
Fehlercode 2: Nicht verwendet		
-	-	-

Tabelle 77: Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung

Fehlerargument	Ursache	Beseitigung
Fehlercode 3: Klemmenbus-Protokollfehler		
-	Störung der Klemmenbuskommunikation; defekte Busklemme kann nicht ermittelt werden	<p>Ist am I/O-IPC eine Potentialeinspeiseklemme (z. B. 750-602) angeschlossen, stellen Sie sicher, dass diese funktioniert (siehe dazu Kap. „LED-Signalisierung“). Ist die Einspeiseklemme fehlerfrei, dann liegt eine Störung an einer Busklemme vor. Ermitteln Sie diese Busklemme wie folgt:</p> <ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung aus. - Platzieren Sie die Endklemme in der Mitte der angeschlossenen Busklemmen. - Schalten Sie die Versorgungsspannung wieder ein. - Falls die I/O-LED noch rot blinkt, schalten Sie die Versorgungsspannung erneut aus und platzieren Sie die Endklemme in der Mitte der ersten Hälfte der Busklemmen (zum I/O-IPC hin). <p>Wenn nur noch eine Busklemme übrig ist, aber die LED noch blinkt, dann ist diese oder die Klemmenbusschnittstelle des I/O-IPC defekt. Tauschen Sie die Busklemme oder den I/O-IPC aus.</p> <ul style="list-style-type: none"> - Wenn die LED nicht mehr blinkt, schalten Sie die Versorgungsspannung ab und platzieren Sie die Endklemme in der Mitte der zweiten Hälfte der Busklemmen (weg vom I/O-IPC). - Schalten Sie die Versorgungsspannung wieder ein. <p>Wiederholen Sie diese Prozedur so oft, bis Sie die defekte Busklemme ermittelt haben. Tauschen Sie diese anschließend aus.</p>

Tabelle 77: Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung

Fehlerargument	Ursache	Beseitigung
Fehlercode 4: Physischer Fehler am Klemmenbus		
-	Fehler in der Klemmenbus-Datenkommunikation oder Unterbrechung des Klemmenbusses am I/O-IPC	<ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung des I/O-IPC ab. - Stecken Sie eine Busklemme für Prozessdaten an den I/O-IPC. - Stecken Sie als letztes die Endklemme an den I/O-IPC. <p>Wird kein Fehlerargument von der I/O-LED ausgegeben, liegt ein Fehler an der Klemmenbusschnittstelle vor und der I/O-IPC ist auszutauschen.</p>
n*	Klemmenbusunterbrechung nach der n-ten Prozessdatenklemme.	<ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung des I/O-IPC ab. - Tauschen Sie die (n+1)-te Prozessdatenklemme aus. - Schalten Sie die Versorgungsspannung wieder ein. <p>Busklemmen, die keine Daten liefern, werden nicht beachtet (z. B. Einspeiseklemme ohne Diagnose).</p>
Fehlercode 5: Klemmenbus-Initialisierungsfehler		
n*	Fehler in der Registerkommunikation während Klemmenbusinitialisierung	<ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung des I/O-IPC ab. - Tauschen Sie die (n+1)-te Prozessdatenklemme aus. - Schalten Sie die Versorgungsspannung wieder ein. <p>Busklemmen, die keine Daten liefern, werden nicht beachtet (z. B. Einspeiseklemme ohne Diagnose).</p>
Fehlercode 6: Designfehler in der Knotenkonfiguration		
5	Maximalgröße des Prozessabbilds überschritten	<ul style="list-style-type: none"> - Schalten Sie die Versorgungsspannung des I/O-IPC ab und reduzieren Sie die Anzahl der Busklemmen. - Schalten Sie die Versorgungsspannung wieder ein.
Fehlercode 7: Nicht verwendet		
-	-	-
Fehlercode 8: Nicht verwendet		
-	-	-

Tabelle 77: Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung

Fehlerargument	Ursache	Beseitigung
Fehlercode 9: CPU-Ausnahmefehler		
1	Ungültige Programmanweisung	Störung der Programmabfolge. Kontaktieren Sie den WAGO-Support.
2	Überlauf Stapelspeicher	Störung der Programmabfolge. Kontaktieren Sie den WAGO-Support.
3	Unterlauf Stapelspeicher	Störung der Programmabfolge. Kontaktieren Sie den WAGO-Support.
4	Ungültiges Ereignis (NMI)	Störung der Programmabfolge. Kontaktieren Sie den WAGO-Support.

16 Service

In diesem Kapitel erhalten Sie Informationen zu Wartungs- und Servicetätigkeiten.

VORSICHT **Heiße Oberfläche!**



Während des Betriebs können hohe Oberflächentemperaturen am Gehäuse des I/O-IPC auftreten. War der I/O-IPC in Betrieb, lassen Sie ihn abkühlen, bevor Sie diesen berühren.

16.1 Austausch der Batterie

Wenn Sie bei fehlender Spannungsversorgung die Batterie wechseln, stellen Sie sicher, dass Sie eine neue Batterie vom Typ CR2025 (Li/MnO₂, 165 mAh) folgender Hersteller einsetzen: Varta, Sony, Renata, Mitsubishi, Sanyo oder Toshiba.

Ein Kondensator sorgt kurze Zeit für die fehlende Spannungsversorgung der Echtzeituhr. Die Daten des SRAM bleiben dadurch beim Wechsel der Batterie erhalten.

VORSICHT **Explosionsgefahr!**



Eine falsch eingesetzte Batterie kann explodieren. Achten Sie darauf, dass Sie die Batterie korrekt einlegen (Pluspol obenliegend). Bei Nichtbeachtung besteht die Gefahr von Personen- und Sachschäden.

ESD



Elektrostatische Entladung!

Ohne die Frontplatte sind Teile der Leiterplatte zugänglich. Halten Sie notwendige ESD-Maßnahmen ein, um mögliche Schäden verursacht durch elektrostatische Entladung zu vermeiden.

Hinweis



Aufladen der Batterie

Sie dürfen diese Batterie nicht wieder aufladen. Öffnen Sie niemals die Batterie und werfen Sie diese auch nie ins Feuer.

Hinweis



Lebensdauer der Batterie

Die Lebensdauer der Batterie ist von der Umgebungstemperatur abhängig. Daher empfiehlt sich ein jährlicher Austausch.

Zum Auswechseln der Batterie gehen Sie folgendermaßen vor:

1. Entfernen Sie das Feldbuskabel vom Anschluss X3 (bei den Varianten ohne Feldbusanschluss entfällt dieser Schritt).
2. Demontieren Sie die Abdeckung (51) durch Entfernen der vier Schrauben (53) mit einem passenden Schraubendreher für Torx[®]-Schrauben.
Bei Wandmontage halten Sie dabei die Abdeckung vor dem Lösen der letzten Schraube fest, damit die Abdeckung nicht herunterfällt.
3. Entnehmen Sie die alte Batterie (52), indem Sie diese mit dem Finger nach unten gegen den Kontakt (54) drücken und dann zu sich herausziehen.

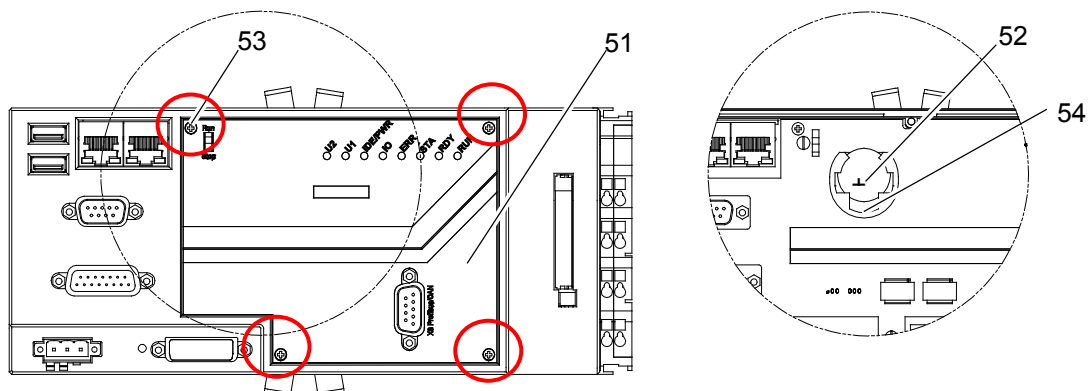


Abbildung 126: Batteriewechsel der Notstromversorgung 1

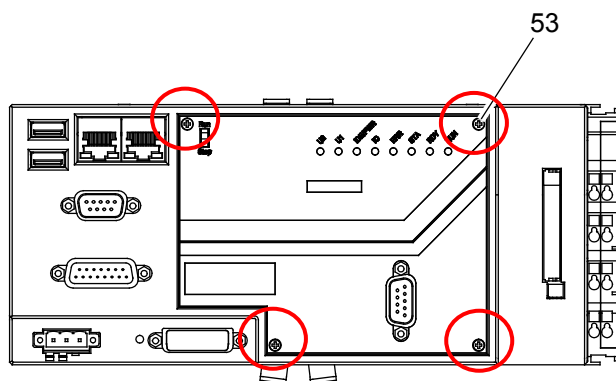


Abbildung 127: Batteriewechsel der Notstromversorgung 2

4. Legen Sie die neue Batterie, Typ CR2025, mit dem Pluspol obenliegend in die Halterung ein, sodass die Batteriekante am Kontakt (54) anliegt, und drücken Sie die Batterie hinein, bis diese fühlbar einrastet.
5. Befestigen Sie die Frontplatte durch Festziehen der vier Schrauben (53). Achten Sie darauf, dass Sie die Gewinde nicht überdrehen. Bei einer senkrechten Montage des I/O-IPC halten Sie bis zur Befestigung der ersten Schraube die Frontplatte fest, damit diese nicht herunterfällt.
6. Befestigen Sie das Feldbuskabel an Anschluss X3 (bei den Varianten ohne Feldbusanschluss entfällt dieser Schritt).

16.2 Entsorgung

Entsorgen Sie die 750-Komponenten entsprechend der für sie geltenden Gesetze. Sie können sich auch an einen zertifizierten Entsorgungsbetrieb wenden.

17 Einsatz in explosionsgefährdeten Bereichen

Das **WAGO-I/O-SYSTEM 750** (elektrische Betriebsmittel) ist für den Einsatz in explosionsgefährdeten Bereichen der Zone 2 ausgelegt.

Die nachfolgenden Kapitel beinhalten die allgemeine Kennzeichnung der Komponenten sowie die zu berücksichtigenden Errichtungsbestimmungen. Die einzelnen Abschnitte im Kapitel „Errichtungsbestimmungen“ müssen berücksichtigt werden, falls die Busklemme die entsprechende Zulassung besitzt oder dem Anwendungsbereich der ATEX-Richtlinie unterliegt.

17.1 Beispielhafter Aufbau der Kennzeichnung

17.1.1 Kennzeichnung für Europa gemäß ATEX und IEC-Ex

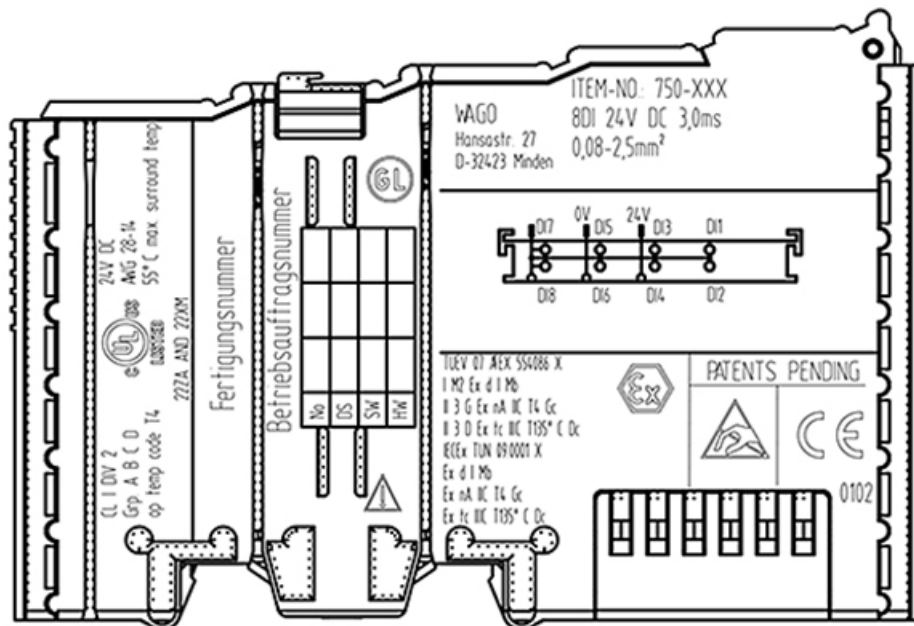


Abbildung 128: Beispiel für die seitliche Bedruckung der ATEX- und IECEx-zugelassenen Busklemmen.

TUEV 07 AEX 554086 X
I M2 Ex d I Mb
II 3 G Ex nA IIC T4 Gc
II 3 D Ex tc III C T135° C Dc
IECEx TUN 09.0001 X
Ex d I Mb
Ex nA IIC T4 Gc
Ex tc III C T135° C Dc



Abbildung 129: Textdetail - Beispielbedruckung der ATEX- und IECEx-zugelassenen Busklemmen.

Tabelle 78: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Busklemmen.

Bedruckungstext	Beschreibung
TÜV 07 ATEX 554086 X IECEx TUN 09.0001 X	Zulassungsbehörde bzw. Bescheinigungsnummern
Stäube	
II	Gerätegruppe: alle außer Bergbau
3D	Geräteklasse 3 (Zone 22)
Ex	Explosionsschutzkennzeichen
tc Dc	Zündschutzart und Geräteschutzniveau (EPL): Schutz durch Gehäuse
IIIC	Staubgruppe: explosionsfähige Staubatmosphäre
T 135°C	Max. Oberflächentemperatur des Gehäuses (ohne Staubablage)
Bergbau	
I	Gerätegruppe: Bergbau
M2	Geräteklasse: hohes Maß an Sicherheit
Ex	Explosionsschutzkennzeichen
d Mb	Zündschutzart und Geräteschutzniveau (EPL): druckfeste Kapselung
I	Elektrische Geräte im schlagwettergefährdeten Grubenbau
Gase	
II	Gerätegruppe: Alle außer Bergbau
3G	Geräteklasse 3 (Zone 2)
Ex	Explosionsschutzkennzeichen
nA Gc	Zündschutzart und Geräteschutzniveau (EPL): nicht funkendes Betriebsmittel
nC Gc	Zündschutzart und Geräteschutzniveau (EPL): funkendes Betriebsmittel, in denen die Kontakte in geeigneter Weise geschützt sind
IIC	Gasgruppe: explosionsfähige Gasatmosphäre
T4	Temperaturklasse: Max. Oberflächentemperatur 135°C

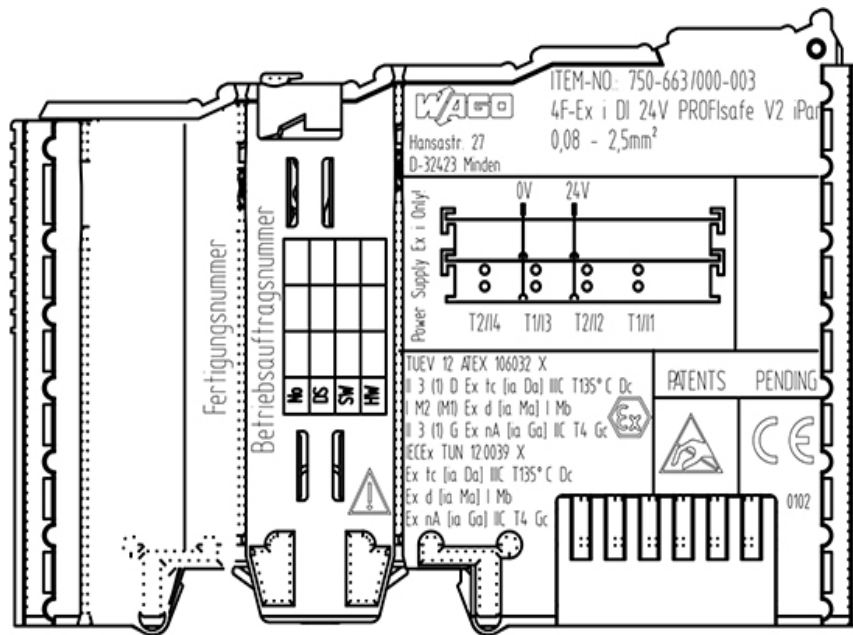


Abbildung 130: Beispiel für die seitliche Bedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.


TUEV 12 ATEX 106032 X
 II 3 (1) D Ex tc [ia Da] IIC T135° C Dc
 I M2 (M1) Ex d [ia Ma] I Mb
 II 3 (1) G Ex nA [ia Ga] IIC T4 Gc 
 IECEx TUN 12.0039 X
 Ex tc [ia Da] IIC T135° C Dc
 Ex d [ia Ma] I Mb
 Ex nA [ia Ga] IIC T4 Gc

Abbildung 131: Textdetail - Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

Tabelle 79: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

Bedruckungstext	Beschreibung
TÜV 07 ATEX 554086 X IECEx TUN 09.0001X	Zulassungsbehörde bzw. Bescheinigungsnummern
TÜV 12 ATEX 106032 X IECEx TUN 12.0039 X	
Stäube	
II	Gerätegruppe: alle außer Bergbau
3(1)D	Geräteklasse 3 (Zone 22) die Sicherheitsvorrichtungen für Geräte der Kategorie 1 (Zone 20) enthalten
3(2)D	Geräteklasse 3 (Zone 22) die Sicherheitsvorrichtungen für Geräte der Kategorie 2 (Zone 21) enthalten
Ex	Explosionsschutzkennzeichen
tc Dc	Zündschutzart und Geräteschutzniveau (EPL): Schutz durch Gehäuse
[ia Da]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 20
[ib Db]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 21
IIIC	Staubgruppe: explosionsfähige Staubatmosphäre
T 135°C	Max. Oberflächentemperatur des Gehäuses (ohne Staubablage)
Bergbau	
I	Gerätegruppe: Bergbau
M2 (M1)	Geräteklasse: hohes Maß an Sicherheit, mit Stromkreisen, die ein sehr hohes Maß an Sicherheit darbieten
Ex d Mb	Explosionsschutzkennzeichen mit Zündschutzart und Geräteschutzniveau (EPL): druckfeste Kapselung
[ia Ma]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen
I	Elektrische Geräte im schlagwettergefährdeten Grubenbau

Tabelle 79: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

Gase	
II	Gerätegruppe: Alle außer Bergbau
3(1)G	Geräteklasse 3 (Zone 2) die Sicherheitsvorrichtungen für Geräte der Kategorie 1 (Zone 0) enthalten
3(2)G	Geräteklasse 3 (Zone 2) die Sicherheitsvorrichtungen für Geräte der Kategorie 2 (Zone 1) enthalten
Ex	Explosionsschutzkennzeichen
nA Gc	Zündschutzart und Geräteschutzniveau (EPL): nicht funkendes Betriebsmittel
[ia Ga]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 0
[ib Gb]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 1
IIC	Gasgruppe: explosionsfähige Gasatmosphäre
T4	Temperaturklasse: Max. Oberflächentemperatur 135°C

17.1.2 Kennzeichnung für Amerika gemäß NEC 500

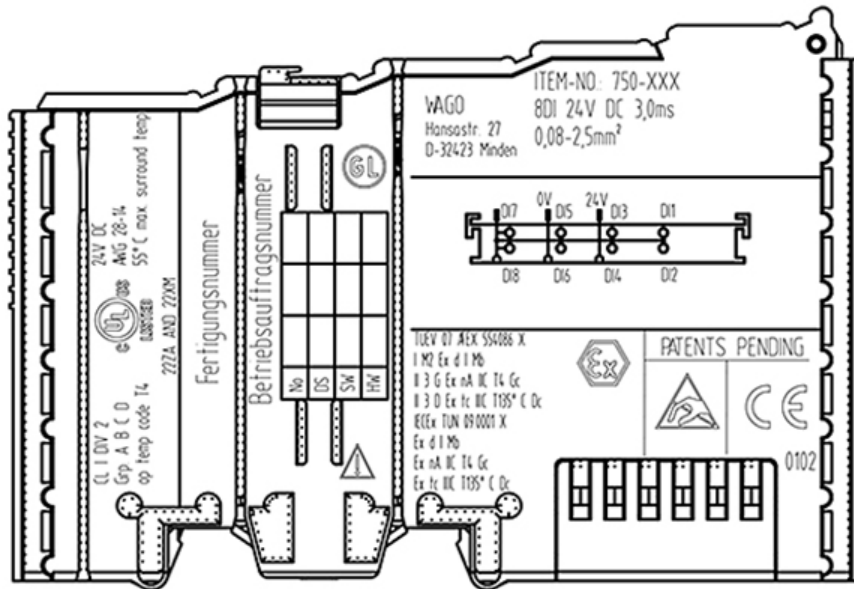


Abbildung 132: Beispiel für seitliche Bedruckung der Busklemmen gemäß NEC 500

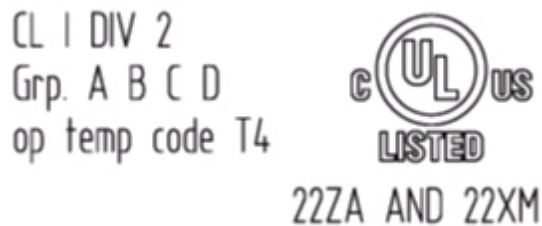


Abbildung 133: Textdetail - Beispielbedruckung der Busklemmen gemäß NEC 500

Tabelle 80: Beschreibung der Beispielbedruckung der Busklemmen gemäß NEC 500

Bedruckungstext	Beschreibung
CL I	Explosionsschutzgruppe (Gefahrenkategorie)
DIV 2	Einsatzbereich
Grp. ABCD	Explosionsgruppe (Gasgruppe)
Op temp code T4	Temperaturklasse

17.2 Errichtungsbestimmungen

Für die Errichtung und den Betrieb elektrischer Anlagen in explosionsfähigen Bereichen sind die am Einsatzort geltenden nationalen und internationalen Bestimmungen und Verordnungen zu beachten.

17.2.1 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 07 ATEX 554086 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) EN 60079-0, EN 60079-11, EN 60079-15 und EN 60079-31 erfüllt.
Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß EN 60079-0 und EN 60079-1 gewährleistet mit der Schutzart IP64.
Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExNB bescheinigt sein.
2. Außerhalb des Gerätes sind geeignete Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn eine explosionsfähige Atmosphäre ausgeschlossen werden kann.
4. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
Das ist ebenfalls und im Besonderen gültig für die Schnittstellen „Memory - Card“, „USB“, „Fieldbus connection“, „Configuration and programming interface“, „antenna socket“, „D-Sub“, „DVI-port“ und das „Ethernet interface“. Diese Schnittstellen sind keine energiebegrenzten oder eigensichere Kreise. Die Verwendung dieser Schnittstellen erfolgt in Verantwortung des Betreibers.
5. Für die Typen 750-606, 750-625/000-001, 750-487/003-000, 750-484 und 750-633 muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der EN 60664-1 definiert.
6. Bei austauschbaren Sicherungen ist Folgendes zu berücksichtigen: Die Sicherung darf nicht entfernt oder getauscht werden, wenn das Gerät in Betrieb ist.
7. In der Nähe des Gerätes sind die folgenden Warnhinweise anzubringen:
WARNHINWEIS – SICHERUNG NICHT UNTER SPANNUNG
HERAUSNEHMEN ODER WECHSELN
WARNHINWEIS – NICHT UNTER SPANNUNG TRENNEN
WARNHINWEIS – NUR IN EINEM NICHT
EXPLOSIONSGEFÄHRDETEN BEREICH TRENNEN

17.2.2 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 12 ATEX 106032 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** Ex i in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) EN 60079-0, EN 60079-11, EN 60079-15 und EN 60079-31 erfüllt.
Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß EN 60079-0 und EN 60079-1 gewährleistet mit der Schutzart IP64.
Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExNB bescheinigt sein.
2. Außerhalb des Gerätes sind geeignete Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
4. Für das Gerät muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der EN 60664-1 definiert.

17.2.3 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEx TUN 09.0001 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) IEC 60079-0, IEC 60079-11, IEC 60079-15, IEC 60079-31 erfüllt.
Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß IEC 60079-0 und IEC 60079-1 gewährleistet mit der Schutzart IP64.
Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExCB bescheinigt sein.
2. Außerhalb des Gerätes sind Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn eine explosionsfähige Atmosphäre ausgeschlossen werden kann.
4. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
Das ist ebenfalls und im Besonderen gültig für die Schnittstellen „Memory-Card“, „USB“, „Fieldbus connection“, „Configuration and programming interface“, „antenna socket“, „D-Sub“, „DVI-port“ und das „Ethernet interface“. Diese Schnittstellen sind keine energiebegrenzten oder eigensicheren Kreise. Die Verwendung dieser Schnittstellen erfolgt in Verantwortung des Betreibers.
5. Für die Typen 750-606, 750-625/000-001, 750-487/003-000, 750-484 und 750-633 muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der IEC 60664-1 definiert.
6. Bei austauschbaren Sicherungen ist Folgendes zu berücksichtigen: Die Sicherung darf nicht entfernt oder getauscht werden, wenn das Gerät in Betrieb ist.
7. In der Nähe des Gerätes sind die folgenden Warnhinweise anzubringen:
WARNHINWEIS – SICHERUNG NICHT UNTER SPANNUNG
HERAUSNEHMEN ODER WECHSELN
WARNHINWEIS – NICHT UNTER SPANNUNG TRENNEN
WARNHINWEIS – NUR IN EINEM NICHT
EXPLOSIONSGEFÄHRDETEN BEREICH TRENNEN

17.2.4 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEx TUN 12.0039 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** Ex i in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) IEC 60079-0, IEC 60079-11, IEC 60079-15 und IEC 60079-31 erfüllt.
Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß IEC 60079-0 und IEC 60079-1 gewährleistet mit der Schutzart IP64.
Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExCB bescheinigt sein.
2. Außerhalb des Gerätes sind Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
4. Für das Gerät muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der IEC 60664-1 definiert.

17.2.5 ANSI/ISA 12.12.01

- A. „Dieses Gerät ist ausschließlich für den Einsatz in Class I, Division 2, Gruppen A, B, C, D oder nicht explosionsgefährdeten Bereichen geeignet.“
- B. „Dieses Gerät muss in werkzeuggesicherte Gehäuse eingebaut werden.“
- C. „WARNUNG - Explosionsgefahr - Der Austausch von Komponenten kann die Eignung für Class I, Division 2 beeinträchtigen.“
- D. „WARNUNG - Klemmen Sie das Gerät nur dann ab, wenn die Versorgung ausgeschaltet ist oder wenn der Bereich als nicht-explosionsgefährdet gilt“ ist in der Nähe von Steckverbindern und Sicherungshaltern anzubringen, die für Bediener zugänglich sind.
- E. Falls eine Sicherung vorhanden ist, muss folgende Information vorhanden sein: „Es muss ein Schalter vorgesehen sein, der für den Einsatzort geeignet ist, in dem das Gerät installiert wird, um die Sicherung von der Versorgung zu trennen.“
- F. Für Baugruppen mit EtherCAT/Ethernet-Steckverbindern gilt: „Nur für den Einsatz in LAN, nicht für den Anschluss an Fernmeldeleitungen.“
- G. „WARNUNG - Die Klemme 750-642 ist nur mit dem Antennenmodul 758-910 zu benutzen.“
- H. Für Feldbuskoppler/-controller und Economy-Busmodule gilt:
„Die Service-Schnittstelle ist nur für einen vorübergehenden Anschluss bestimmt. Verbinden oder trennen sie diese nur, wenn der Bereich als nicht-explosionsgefährdet gilt. Das Verbinden oder Trennen in einer explosionsgefährdeten Atmosphäre könnte zu einer Explosion führen.“
- I. Für Geräte mit Sicherung gilt: „WARNUNG - Geräte mit Sicherungen dürfen nicht in Stromkreise integriert werden, die einer Überlast ausgesetzt sind, z.B. Motorkreise.“
- J. Für Geräte mit SD-Karte gilt: „WARNUNG - Stecken oder ziehen Sie die SD-Karte bei anliegender Spannung nur dann, wenn es sicher ist, dass der Bereich frei von zündfähigen Gasen oder Dämpfen ist.“

Information



Weitere Information

Einen Zertifizierungsnachweis erhalten Sie auf Anfrage. Beachten Sie auch die Hinweise auf dem Beipackzettel der Busklemme. Das Handbuch mit den oben aufgeführten Bedingungen für sicheren Gebrauch muss für den Anwender jederzeit zur Verfügung stehen.

18 Busklemmen

18.1 Übersicht

Für den Aufbau von Applikationen mit dem WAGO-I/O-SYSTEM 750/753 sind verschiedene Arten von Busklemmen verfügbar:

- Digitaleingangsklemmen
- Digitalausgangsklemmen
- Analogeingangsklemmen
- Analogausgangsklemmen
- Sonderklemmen
- Systemklemmen

Eine detaillierte Beschreibung zu jeder Busklemme und deren Varianten entnehmen Sie den Handbüchern zu den Busklemmen.

Sie finden diese Beschreibungen auf der Internetseite <http://www.wago.com>.

Information



Weitere Information zum WAGO-I/O-SYSTEM

Aktuelle Informationen zum modularen WAGO-I/O-SYSTEM finden Sie auf der Internetseite <http://www.wago.com>.

18.2 Aufbau der Prozessdaten für MODBUS/TCP

Der Aufbau der Prozessdaten ist bei einigen Busklemmen bzw. deren Varianten feldbuspezifisch.

Bei MODBUS/TCP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Im Folgenden wird für alle Busklemmen des WAGO-I/O-SYSTEMs 750 und 753 die feldbuspezifische Darstellung im Prozessabbild für MODBUS/TCP beschrieben und der Aufbau der Prozesswerte gezeigt.

ACHTUNG **Geräteschäden durch falsche Adressierung!**



Zur Vermeidung von Geräteschäden im Feldbereich, müssen Sie bei der Adressierung einer an beliebiger Position im Feldbusknoten befindlichen Busklemme, die Prozessdaten aller vorherigen byte- bzw. bitweise-orientierten Busklemmen berücksichtigen.

Für das PFC-Prozessabbild des Feldbuscontrollers ist der Aufbau der Prozesswerte identisch.

18.2.1 Digitaleingangsklemmen

Die Digitaleingangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Signalzustand des jeweiligen Kanals angibt. Diese Bits werden in das Eingangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie z. B. Drahtbruch und/oder Kurzschluss.

Sofern in dem Knoten auch Analogeingangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Eingangsdaten in dem Eingangsprozessabbild angehängt.

18.2.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose

750-435

Tabelle 81: 1-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 1	Datenbit DI 1

18.2.1.2 2-Kanal-Digitaleingangsklemmen

750-400, -401, -405, -406, -410, -411, -412, -427, -438, (und alle Varianten),
753-400, -401, -405, -406, -410, -411, -412, -427

Tabelle 82: 2-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

18.2.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose

750-419, -421, -424, -425
753-421, -424, -425

Tabelle 83: 2-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

18.2.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

750-418
753-418

Die Digitaleingangsklemme liefert über die Prozesswerte im Eingangsprozessabbild hinaus 4 Bit Daten, die im Ausgangsprozessabbild dargestellt werden.

Tabelle 84: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Quittierungsbit Q 2 Kanal 2	Quittierungsbit Q 1 Kanal 1	0	0

18.2.1.5 4-Kanal-Digitaleingangsklemmen

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433, -1420, -1421, -1422, -1423
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

Tabelle 85: 4-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

18.2.1.6 8-Kanal-Digitaleingangsklemmen

750-430, -431, -436, -437, -1415, -1416, -1417, -1418
753-430, -431, -434

Tabelle 86: 8-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

18.2.1.7 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten

750-1425

Die Digitaleingangsklemme PTC liefert über einen logischen Kanal 2 Byte für das Ein- und Ausgangsprozessabbild.

Der Signalzustand der PTC-Eingänge DI1 ... DI8 wird über das Eingangsdatenbyte D0 an den Feldbuskoppler/-controller übertragen.
Die Fehlerzustände werden über das Eingangsdatenbyte D1 übertragen.

Über das Ausgangsdatenbyte D1 werden die Kanäle 1 ... 8 ein- oder ausgeschaltet. Das Ausgangsdatenbyte D0 ist reserviert und hat immer den Wert „0“.

Tabelle 87: 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten

Eingangsprozessabbild															
Eingangsbyte D0								Eingangsbyte D1							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Signal- zu- stand DI 8	Signal- zu- stand DI 7	Signal- zu- stand DI 6	Signal- zu- stand DI 5	Signal- zu- stand DI 4	Signal- zu- stand DI 3	Signal- zu- stand DI 2	Signal- zu- stand DI 1	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS	Draht- bruch/ Kurz- schluss DB/KS
Kanal 8	Kanal 7	Kanal 6	Kanal 5	Kanal 4	Kanal 3	Kanal 2	Kanal 1	8 Kanal	7 Kanal	6 Kanal	5 Kanal	4 Kanal	3 Kanal	2 Kanal	1 Kanal

Ausgangsprozessabbild															
Ausgangsbyte D0								Ausgangsbyte D1							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
								DI Off 8 Kanal 8 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 7 Kanal 7 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 6 Kanal 6 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 5 Kanal 5 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 4 Kanal 4 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 3 Kanal 3 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 2 Kanal 2 0: Kanal einge- chaltet 1: Kanal ausge- schaltet	DI Off 1 Kanal 1 0: Kanal einge- chaltet 1: Kanal ausge- schaltet
0	0	0	0	0	0	0	0								

18.2.2 16-Kanal-Digitaleingangsklemmen

750-1400, -1402, -1405, -1406, -1407

Tabelle 88: 16-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Daten bit DI 16	Daten bit DI 15	Daten bit DI 14	Daten bit DI 13	Daten bit DI 12	Daten bit DI 11	Daten bit DI 10	Daten bit DI 9	Daten bit DI 8	Daten bit DI 7	Daten bit DI 6	Daten bit DI 5	Daten bit DI 4	Daten bit DI 3	Daten bit DI 2	Daten bit DI 1
Kanal 16	Kanal 15	Kanal 14	Kanal 13	Kanal 12	Kanal 11	Kanal 10	Kanal 9	Kanal 8	Kanal 7	Kanal 6	Kanal 5	Kanal 4	Kanal 3	Kanal 2	Kanal 1

18.2.2.1 Digitalausgangsklemmen

Die Digitalausgangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Status des jeweiligen Kanals angibt. Diese Bits werden in das Ausgangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie Drahtbruch und/oder Kurzschluss. Bei einigen Busklemmen müssen, bei gesetztem Diagnosebit, zusätzlich die Datenbits ausgewertet werden.

Sofern in dem Knoten auch Analogausgangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Ausgangsdaten in dem Ausgangsprozessabbild angehängt.

18.2.2.2 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

750-523

Die Digitalausgangsklemmen liefern über das eine Prozesswert-Bit im Ausgangsprozessabbild hinaus 1 Bit, das im Eingangsprozessabbild dargestellt wird. Dieses Statusbit zeigt den „Handbetrieb“ an.

Tabelle 89: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	Statusbit „Handbetrieb“

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	steuert DO 1 Kanal 1

18.2.2.3 2-Kanal-Digitalausgangsklemmen

750-501, -502, -509, -512, -513, -514, -517, -535, (und alle Varianten),
753-501, -502, -509, -512, -513, -514, -517

Tabelle 90: 2-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.4 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-507 (-508), -522,
753-507

Die Digitalausgangsklemmen liefern über die 2-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 2 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 91: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

750-506,
753-506

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die durch einen 2-Bit-Fehlercode eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 92: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 3 Kanal 2	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Diagnosebit S 0 Kanal 1

Diagnosebits S1/S0, S3/S2: = '00' normaler Betrieb

Diagnosebits S1/S0, S3/S2: = '01' keine Last angeschlossen/Kurzschluss gegen +24 V

Diagnosebits S1/S0, S3/S2: = '10' Kurzschluss gegen GND/Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				nicht genutzt	nicht genutzt	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.5 4-Kanal-Digitalausgangsklemmen

750-504, -516, -519, -531,
753-504, -516, -531, -540

Tabelle 93: 4-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.6 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-532

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 94: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Diagnosebit S = '0' kein Fehler
Diagnosebit S = '1' Drahtbruch, Kurzschluss oder Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.7 8-Kanal-Digitalausgangsklemmen

750-530, -536, -1515, -1516
753-530, -534

Tabelle 95: 8-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.8 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-537

Die Digitalausgangsklemmen liefern über die 8-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 8 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Diese sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 96: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Diagnosebit S 8 Kanal 8	Diagnosebit S 7 Kanal 7	Diagnosebit S 6 Kanal 6	Diagnosebit S 5 Kanal 5	Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Diagnosebit S = '0' kein Fehler

Diagnosebit S = '1' Drahtbruch, Kurzschluss oder Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.9 16-Kanal-Digitalausgangsklemmen

750-1500, -1501, -1504, -1505

Tabelle 97: 16-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 16 Kanal 16	steuert DO 15 Kanal 15	steuert DO 14 Kanal 14	steuert DO 13 Kanal 13	steuert DO 12 Kanal 12	steuert DO 11 Kanal 11	steuert DO 10 Kanal 10	steuert DO 9 Kanal 9	steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

18.2.2.10 8-Kanal-Digitaleingangsklemmen/-Digitalausgangsklemmen

750-1502, -1506

Tabelle 98: 8-Kanal-Digitalein-/ -ausgangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8	Datenbit DI 7	Datenbit DI 6	Datenbit DI 5	Datenbit DI 4	Datenbit DI 3	Datenbit DI 2	Datenbit DI 1
Kanal 8	Kanal 7	Kanal 6	Kanal 5	Kanal 4	Kanal 3	Kanal 2	Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8	steuert DO 7	steuert DO 6	steuert DO 5	steuert DO 4	steuert DO 3	steuert DO 2	steuert DO 1
Kanal 8	Kanal 7	Kanal 6	Kanal 5	Kanal 4	Kanal 3	Kanal 2	Kanal 1

18.2.3 Analogeingangsklemmen

Die Analogeingangsklemmen liefern je Kanal 16-Bit-Messwerte und 8 Steuer-/ Statusbits.

MODBUS/TCP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Eingangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/ -controller mit MODBUS/TCP deshalb nur die 16-Bit-Messwerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitaleingangsklemmen gesteckt sind, werden die analogen Eingangsdaten immer vor die digitalen Daten in das Eingangsprozessabbild abgebildet.

Information Informationen zum Steuer-/Statusbyteaufbau



Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Homepage unter: <http://www.wago.com>.

18.2.3.1 1-Kanal-Analogeingangsklemmen

750-491, (und alle Varianten)

Tabelle 99: 1-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Messwert U_D
1	D3	D2	Messwert U_{ref}

18.2.3.2 2-Kanal-Analogeingangsklemmen

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -492, (und alle Varianten),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, 476, -477, 478, -479, -483, -492, (und alle Varianten)

Tabelle 100: 2-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2

18.2.3.3 4-Kanal-Analogeingangsklemmen

750-453, -455, -457, -459, -460, -468, (und alle Varianten),
753-453, -455, -457, -459

Tabelle 101: 4-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2
2	D5	D4	Messwert Kanal 3
3	D7	D6	Messwert Kanal 4

18.2.3.4 3-Phasen-Leistungsmessklemme

750-493

Tabelle 102: 3-Phasen-Leistungsmessklemme

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S0	Statusbyte 0
1	D1	D0	Eingangsdatenwort 1
2	-	S1	Statusbyte 1
3	D3	D2	Eingangsdatenwort 2
4	-	S2	Statusbyte 2
5	D5	D4	Eingangsdatenwort 3

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C0	Steuerbyte 0
1	D1	D0	Ausgangsdatenwort 1
2	-	C1	Steuerbyte 1
3	D3	D2	Ausgangsdatenwort 2
4	-	C2	Steuerbyte 2
5	D5	D4	Ausgangsdatenwort 3

18.2.3.5 8-Kanal-Analogeingangsklemmen

Tabelle 103: 8-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2
2	D5	D4	Messwert Kanal 3
3	D7	D6	Messwert Kanal 4
4	D9	D8	Messwert Kanal 5
5	D11	D10	Messwert Kanal 6
6	D13	D12	Messwert Kanal 7
7	D15	D14	Messwert Kanal 8

18.2.4 Analogausgangsklemmen

Die Analogausgangsklemmen liefern je Kanal 16-Bit-Ausgabewerte und 8 Steuer-/Statusbits.

MODBUS/TCP verwendet die 8 Steuer-/Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Ausgangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/-controller mit MODBUS/TCP deshalb nur die 16-Bit-Ausgabewerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitalausgangsklemmen gesteckt sind, werden die analogen Ausgangsdaten immer vor die digitalen Daten in das Ausgangsprozessabbild abgebildet.

Information Informationen zum Steuer-/Statusbyteaufbau



Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Homepage unter: <http://www.wago.com>.

18.2.4.1 2-Kanal-Analogausgangsklemmen

750-550, -552, -554, -556, -560, -562, 563, -585, (und alle Varianten),
753-550, -552, -554, -556

Tabelle 104: 2-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2

18.2.4.2 4-Kanal-Analogausgangsklemmen

750-553, -555, -557, -559,
753-553, -555, -557, -559

Tabelle 105: 4-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2
2	D5	D4	Ausgabewert Kanal 3
3	D7	D6	Ausgabewert Kanal 4

18.2.4.3 8-Kanal-Analogausgangsklemmen

Tabelle 106: 8-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2
2	D5	D4	Ausgabewert Kanal 3
3	D7	D6	Ausgabewert Kanal 4
4	D9	D8	Ausgabewert Kanal 5
5	D11	D10	Ausgabewert Kanal 6
6	D13	D12	Ausgabewert Kanal 7
7	D15	D14	Ausgabewert Kanal 8

18.2.5 Sonderklemmen

Bei einzelnen Klemmen wird neben den Datenbytes auch das Control-/Statusbyte einblendet. Dieses dient dem bidirektionalen Datenaustausch der Busklemme mit der übergeordneten Steuerung.

Das Controlbyte wird von der Steuerung an die Klemme und das Statusbyte von der Klemme an die Steuerung übertragen. Somit ist beispielsweise das Setzen eines Zählers mit dem Steuerbyte oder die Anzeige von Bereichsunter- oder -überschreitung durch das Statusbyte möglich.

Das Control-/Statusbyte liegt im Prozessabbild stets im Low-Byte.

Information Informationen zum Steuer-/Statusbyteaufbau



Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der Internetseite <http://www.wago.com>.

18.2.5.1 Zählerklemmen

750-404, (und alle Varianten außer /000-005),
753-404, (und Variante /000-003)

Die Zählerklemmen belegen insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Die Busklemmen liefern dann 32-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 107: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003)

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S	Statusbyte
1	D1	D0	Zählerwert
2	D3	D2	

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C	Steuerbyte
1	D1	D0	Zählerwert
2	D3	D2	

750-404/000-005

Die Zählerklemmen belegen insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Diese Busklemmen liefern pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 108: Zählerklemmen 750-404/000-005

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S	Statusbyte
1	D1	D0	Zählerwert Zähler 1
2	D3	D2	Zählerwert Zähler 2

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C	Steuerbyte
1	D1	D0	Zählersetzwert Zähler 1
2	D3	D2	Zählersetzwert Zähler 2

750-638,
753-638

Diese Zählerklemmen belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Die Busklemmen liefern dann pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 109: Zählerklemmen 750-638, 753-638

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S0	Statusbyte von Zähler 1
1	D1	D0	Zählerwert von Zähler 1
2	-	S1	Statusbyte von Zähler 2
3	D3	D2	Zählerwert von Zähler 2

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C0	Steuerbyte von Zähler 1
1	D1	D0	Zählersetzwert von Zähler 1
2	-	C1	Steuerbyte von Zähler 2
3	D3	D2	Zählersetzwert von Zähler 2

18.2.5.2 Pulsweitenklemmen

750-511, (und alle Varianten /xxx-xxx)

Diese Pulsweitenklemmen belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 110: Pulsweitenklemmen 750-511, /xxx-xxx

Ein- und Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C0/S0	Steuer-/Statusbyte von Kanal 1
1	D1	D0	Datenwert von Kanal 1
2	-	C1/S1	Steuer-/Statusbyte von Kanal 2
3	D3	D2	Datenwert von Kanal 2

18.2.5.3 Serielle Schnittstellen mit alternativem Datenformat

750-650, (und die Varianten /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (und die Varianten /000-001, -002, -003),
750-653, (und die Varianten /000-002, -007),

753-650, -653

Hinweis



Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrisierten Betriebsart!

Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Der Aufbau des Prozessabbilds dieser Busklemme hängt dann davon ab, welche Betriebsart eingestellt ist.

Die seriellen Schnittstellenklemmen, die auf das alternative Datenformat eingestellt sind, belegen insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 111: Serielle Schnittstellen mit alternativem Datenformat

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	C/S	Datenbyte	Steuer-/Statusbyte
1	D2	D1	Datenbytes	

18.2.5.4 Serielle Schnittstellen mit Standard-Datenformat

750-650/000-001, -014, -015, -016
750-653/000-001, -006

Die seriellen Schnittstellenklemmen, die auf das Standard-Datenformat eingestellt sind, belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 112: Serielle Schnittstellen mit Standard-Datenformat

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	C/S	Datenbyte	Steuer-/ Statusbyte
1	D2	D1	Datenbytes	
2	D4	D3		

18.2.5.5 Datenaustauschklemmen

750-654, (und die Variante /000-001)

Die Datenaustauschklemmen belegen jeweils insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 113: Datenaustauschklemmen

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D1	D0	Datenbytes	
1	D3	D2		

18.2.5.6 SSI-Geber-Interface-Busklemmen

750-630, (und alle Varianten)

Hinweis



Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrisierten Betriebsart!

Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Der Aufbau des Prozessabbilds dieser Busklemme hängt dann davon ab, welche Betriebsart eingestellt ist.

Die SSI-Geber Interface Busklemmen mit Status belegen insgesamt 4 Datenbytes im Eingangsbereich des Prozessabbilds. Dabei werden mit word-alignment insgesamt 2 Worte im Prozessabbild belegt.

Tabelle 114: SSI-Geber Interface Busklemmen mit alternativem Datenformat

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Datenbytes
1	D3	D2	

18.2.5.7 Weg- und Winkelmessung

750-631/000-004, -010, -011

Die Busklemme 750-631 belegt 5 Bytes im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 115: Weg- und Winkelmessung 750-631/000-004, --010, -011

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S	nicht genutzt Statusbyte
1	D1	D0	Zählerwort
2	-	-	nicht genutzt
3	D4	D3	Latchwort

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C	Steuerbyte von Zähler 1
1	D1	D0	Zählerwert von Zähler 1
2	-	-	nicht genutzt
3	-	-	nicht genutzt

750-634

Die Busklemme 750-634 belegt 5 Bytes (in der Betriebsart Periodendauermessung mit 6 Bytes) im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 116: Incremental-Encoder-Interface 750-634

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S	nicht genutzt Statusbyte
1	D1	D0	Zählerwort
2	-	(D2) *)	nicht genutzt (Periodendauer)
3	D4	D3	Latchwort

*) Ist durch das Steuerbyte die Betriebsart Periodendauermessung eingestellt, wird in D2 zusammen mit D3/D4 die Periodendauer als 24-Bit-Wert ausgegeben.

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C	nicht genutzt Steuerbyte
1	D1	D0	Zählersetzwort
2	-	-	nicht genutzt
3	-	-	

750-637

Die Incremental-Encoder-Interface Busklemme belegt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbaus, 4 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 117: Incremental-Encoder-Interface 750-637

Ein- und Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C0/S0	Steuer-/Statusbyte von Kanal 1
1	D1	D0	Datenwerte von Kanal 1
2	-	C1/S1	Steuer-/Statusbyte von Kanal 2
3	D3	D2	Datenwerte von Kanal 2

750-635,
753-635

Die Digitale Impuls Schnittstelle belegt insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 118: Digitale Impuls Schnittstelle 750-635

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	C0/S0	Datenbyte	Steuer-/Statusbyte
1	D2	D1	Datenbytes	

18.2.5.8 DC-Drive Controller

750-636

Der DC-Drive-Controller 750-636 stellt dem Koppler über 1 logischen Kanal 6 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Positionsdaten werden in 4 Ausgangsbytes (D0 ... D3) und 4 Eingangsbytes (D0 ... D3) abgelegt. 2 Steuerbytes (C0, C1) und 2 Statusbytes (S0, S1) dienen zur Steuerung der Busklemme und des Antriebs. Alternativ zu den Positionsdaten im Eingangsprozessabbild (D0 ... D3) können erweiterte Statusinformationen (S2 ... S5) eingeblendet werden. Die 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen den Prozessdaten und den erweiterten Statusbytes im Eingangsprozessabbild erfolgt über Bit 3 (ExtendedInfo_ON) im Controlbyte C1 (C1.3). Mit Bit 3 des Statusbytes S1 (S1.3) wird die Umschaltung quittiert.

Tabelle 119: Antriebssteuerung 750-636

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	S1	S0	Status S1	Statusbyte S0
1	D1*) / S3**)	D0*) / S2**)	Istposition*) / Erweitertes Statusbyte S3**)	Istposition (LSB)*) / Erweitertes Statusbyte S2**)
2	D3*) / S5**)	D2*) / S4**)	Istposition (MSB)*) / Erweitertes Statusbyte S3**)	Istposition*) / Erweitertes Statusbyte S4**)

*) ExtendedInfo_ON = '0'.

***) ExtendedInfo_ON = '1'.

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	C1	C0	Steuerbyte C1	Steuerbyte C0
1	D1	D0	Sollposition	Sollposition (LSB)
2	D3	D2	Sollposition (MSB)	Sollposition

18.2.5.9 Steppercontroller

750-670

Der Steppercontroller RS 422 / 24 V / 20 mA 750-670 stellt dem Feldbuskoppler über 1 logischen Kanal 12 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Daten werden in Abhängigkeit von der Betriebsart in bis zu 7 Ausgangsbytes (D0 ... D6) und 7 Eingangsbytes (D0 ... D6) abgelegt. Das Ausgangsbyte D0 und das Eingangsbyte D0 sind reserviert und ohne Funktion. Ein Klemmenbus-Steuer- und Statusbyte (C0, S0) sowie 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen beiden Prozessabbildern erfolgt über das Bit 5 im Controlbyte C0 (C0.5). Mit dem Bit 5 des Statusbytes S0 (S0.5) wird das Einschalten der Mailbox quittiert.

Tabelle 120: Steppercontroller RS 422 / 24 V / 20 mA 750-670

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	Reserviert	S0	Reserviert	Statusbyte S0
1	D1	D0	Prozessdaten*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	S3	D6	Statusbyte S3	Prozessdaten*) / Reserviert**)
5	S1	S2	Statusbyte S1	Statusbyte S2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).

***) Mailboxprozessabbild (Mailbox eingeschaltet)

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	Reserviert	C0	Reserviert	Controlbyte C0
1	D1	D0	Prozessdaten*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	C3	D6	Controlbyte C3	Prozessdaten*) / Reserviert**)
5	C1	C2	Controlbyte C1	Controlbyte C2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).

***) Mailboxprozessabbild (Mailbox eingeschaltet)

18.2.5.10 RTC-Modul

750-640

Das RTC-Modul belegt insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes, ein zusätzliches Steuer-/Statusbyte und jeweils ein Befehlsbyte (ID). Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 121: RTC-Modul 750-640

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	ID	C/S	Befehlsbyte	Steuer-/Statusbyte
1	D1	D0	Datenbytes	
2	D3	D2		

18.2.5.11 DALI/DSI-Masterklemme

750-641

Die DALI/DSI-Masterklemme belegt insgesamt 6 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 122: DALI/DSI-Masterklemme 750-641

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	S	DALI-Antwort	Statusbyte
1	D2	D1	Message 3	DALI-Adresse
2	D4	D3	Message 1	Message 2

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	C	DALI-Befehl, DSI-Dimmwert	Steuerbyte
1	D2	D1	Parameter 2	DALI-Adresse
2	D4	D3	Command-Extension	Parameter 1

18.2.5.12 DALI-Multi-Master-Klemme

753-647

Die DALI-Multi-Master-Klemme belegt insgesamt 24 Byte im Ein- und Ausgangsbereich des Prozessabbildes.

Die DALI-Multi-Master-Klemme kann im „Easy-Modus“ (Standardeinstellung) und im „Full-Modus“ betrieben werden. Der „Easy-Modus“ wird zur Übermittlung einfacher binärer Signale für die Beleuchtungssteuerung verwendet. Eine Konfiguration oder Programmierung mittels DALI-Masterbaustein ist im „Easy-Modus“ nicht notwendig.

Veränderungen von einzelnen Bits des Prozessabbildes werden direkt in DALI-Kommandos für ein vorkonfiguriertes DALI-Netzwerk umgewandelt. Von dem 24-Byte-Prozessabbild können im „Easy-Modus“ 22 Bytes direkt zum Schalten von EVGs, Gruppen oder Szenen genutzt werden. Schaltbefehle werden über DALI- und Gruppenadressen übertragen, dabei wird jede DALI- und jede Gruppenadresse durch ein 2-Bit-Paar repräsentiert.

Der Aufbau der Prozessdaten ist im Einzelnen in den anschließenden Tabellen dargestellt.

Tabelle 123: Übersicht über das Eingangsprozessabbild im „Easy-Modus“

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S	res. Status Broadcast schalten: Bit 0: 1-/2-Tasten-Modus Bit 2: Broadcast-Status EIN/AUS Bit 1,3-7: -
1	DA4...DA7	DA0...DA3	Bitpaar für DALI-Adresse DA0: Bit 1: Bit gesetzt = EIN Bit nicht gesetzt = AUS Bit 2: Bit gesetzt = Fehler Bit nicht gesetzt = kein Fehler Bitpaare DA1 bis DA63 analog zu DA0.
2	DA12...DA15	DA8...DA11	
3	DA20...DA23	DA16...DA19	
4	DA28...DA31	DA24...DA27	
5	DA36...DA39	DA32...DA35	
6	DA44...DA47	DA40...DA43	
7	DA52...DA55	DA48...DA51	
8	DA60...DA63	DA56...DA59	
9	GA4...GA7	GA0...GA3	Bitpaar für DALI-Gruppenadresse GA0: Bit 1: Bit gesetzt = EIN Bit nicht gesetzt = AUS Bit 2: Bit gesetzt = Fehler Bit nicht gesetzt = kein Fehler Bitpaare GA1 bis GA15 analog zu GA0.
10	GA12...GA15	GA8...GA11	
11	-	-	

DA = DALI-Adresse
GA = Gruppenadresse

Tabelle 124: Übersicht über das Ausgangsprozessabbild im „Easy-Modus“

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	S	res. Broadcast EIN/AUS und schalten: Bit 0: Broadcast EIN Bit 1: Broadcast AUS Bit 2: Broadcast EIN/AUS/dimmen Bit 3: Broadcast kurz EIN/AUS Bit 4...7: reserviert
1	DA4...DA7	DA0...DA3	Bitpaar für DALI-Adresse DA0: Bit 1: kurz: DA schalten EIN lang: dimmen, heller Bit 2: kurz: DA schalten AUS lang: dimmen, dunkler Bitpaare DA1 bis DA63 analog zu DA0.
2	DA12...DA15	DA8...DA11	
3	DA20...DA23	DA16...DA19	
4	DA28...DA31	DA24...DA27	
5	DA36...DA39	DA32...DA35	
6	DA44...DA47	DA40...DA43	
7	DA52...DA55	DA48...DA51	
8	DA60...DA63	DA56...DA59	
9	GA4...GA7	GA0...GA3	Bitpaar für DALI-Gruppenadresse GA0: Bit 1: kurz: GA schalten EIN lang: dimmen heller Bit 2: kurz: GA schalten AUS lang: dimmen dunkler Bitpaare GA1 bis GA15 analog zu GA0.
10	GA12...GA15	GA8...GA11	
11	Bit 8...15	Bit 0...7	

DA = DALI-Adresse
GA = Gruppenadresse

18.2.5.13 LON[®]-FTT-Klemme

753-648

Das Prozessabbild der LON[®]-FTT-Klemme besteht aus einem Steuer-/Statusbyte und 23 Byte bidirektionaler Kommunikationsdaten, die von dem WAGO-I/O-PRO-Funktionsbaustein „LON_01.lib“ verarbeitet werden. Dieser Baustein ist für die Funktion der LON[®]-FTT-Klemme unbedingt erforderlich und stellt steuerungsseitig eine Anwenderschnittstelle zur Verfügung.

18.2.5.14 Funkreceiver EnOcean

750-642

Die EnOcean Funkreceiverklemme belegt insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Die 3 Bytes Ausgangsdaten werden jedoch nicht genutzt. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 125: Funkreceiver EnOcean 750-642

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	S	Datenbyte	Statusbyte
1	D2	D1	Datenbytes	

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	C	nicht genutzt	Steuerbyte
1	-	-	nicht genutzt	

18.2.5.15 MP-Bus-Masterklemme

750-643

Die MP-Bus-Masterklemme belegt insgesamt 8 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbildes, 6 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 126: MP-Bus-Masterklemme 750-643

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	C1/S1	C0/S0	erweitertes Steuer-/Statusbyte	Steuer-/Statusbyte
1	D1	D0	Datenbytes	
2	D3	D2		
3	D5	D4		

18.2.5.16 Bluetooth® RF-Transceiver

750-644

Die Größe des Prozessabbildes der *Bluetooth*®-Busklemme ist in den festgelegten Größen 12, 24 oder 48 Byte einstellbar.

Es besteht aus einem Steuerbyte (Eingang) bzw. Statusbyte (Ausgang), einem Leerbyte, einer 6, 12 oder 18 Byte großen, überlagerbaren Mailbox (Modus 2) und den *Bluetooth*®-Prozessdaten in einem Umfang von 4 bis 46 Byte.

Die *Bluetooth*®-Busklemme belegt also jeweils 12 bis maximal 48 Bytes im Prozessabbild, wobei die Größen des Eingangs- und Ausgangsprozessabbildes stets übereinstimmen.

Das erste Byte enthält das Steuer-/Statusbyte, das zweite ein Leerbyte. Daran schließen sich bei ausgeblendeter Mailbox unmittelbar Prozessdaten an. Bei eingblendeter Mailbox werden je nach deren Größe die ersten 6, 12 oder 18 Byte Prozessdaten von Mailbox-Daten überlagert. Die Bytes im Bereich hinter der optional einblendbaren Mailbox enthalten grundsätzlich Prozessdaten. Den internen Aufbau der *Bluetooth*®-Prozessdaten entnehmen Sie der Dokumentation des *Bluetooth*® RF-Transceivers 750-644.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-CHECK.

Tabelle 127: Bluetooth® RF-Transceiver 750-644

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte
1	D1	D0	Mailbox (0, 3, 6 oder 9 Worte) sowie Prozessdaten (2-23 Worte)	
2	D3	D2		
3	D5	D4		
...		
max. 23	D45	D44		

18.2.5.17 Schwingstärke/Wälzlagerüberwachung VIB I/O

750-645

Die Schwingstärke/Wälzlagerüberwachung VIB I/O belegt insgesamt 12 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 8 Datenbytes und vier zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 8 Worte im Prozessabbild belegt.

Tabelle 128: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte (log. Kanal 1, Sensoreingang 1)
1	D1	D0	Datenbytes (log. Kanal 1, Sensoreingang 1)	
2	-	C1/S1	nicht genutzt	Steuer-/Statusbyte (log. Kanal 2, Sensoreingang 2)
3	D3	D2	Datenbytes (log. Kanal 2, Sensoreingang 2)	
4	-	C2/S2	nicht genutzt	Steuer-/Statusbyte (log. Kanal 3, Sensoreingang 3)
5	D5	D4	Datenbytes (log. Kanal 3, Sensoreingang 3)	
6	-	C3/S3	nicht genutzt	Steuer-/Statusbyte (log. Kanal 4, Sensoreingang 4)
7	D7	D6	Datenbytes (log. Kanal 4, Sensoreingang 4)	

18.2.5.18 KNX/EIB/TP1-Klemme

753-646

Die KNX/TP1-Klemme erscheint im Router- sowie im Gerätemodus mit insgesamt 24 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbildes, 20 Datenbytes und 1 Steuer-/Statusbyte. Die zusätzlichen Bytes S1 bzw. C1 werden als Datenbytes transferiert, aber als erweiterte Status- und Steuerbytes verwendet. Der Opcode dient als Schreib- und Lesekommando für Daten oder als Auslöser bestimmter Funktionen der KNX/EIB/TP1-Klemme. Mit word-alignment werden jeweils 12 Worte im Prozessabbild belegt. Im Routermodus ist kein Zugriff auf das Prozessabbild möglich. Telegramme werden nur getunnelt übertragen.

Im Gerätemodus erfolgt der Zugriff auf KNX-Daten über spezielle Funktionsbausteine der IEC-Applikation. Eine Konfiguration mittels der allgemeinen Engineering-Tool-Software (ETS) für KNX ist notwendig.

Tabelle 129: KNX/EIB/TP1-Klemme 753-646

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	S0	nicht genutzt	Statusbyte
1	S1	OP	Erweitertes Statusbyte	Opcode
2	D1	D0	Datenbyte 1	Datenbyte 0
3	D3	D2	Datenbyte 3	Datenbyte 2
4	D5	D4	Datenbyte 5	Datenbyte 4
5	D7	D6	Datenbyte 7	Datenbyte 6
6	D9	D8	Datenbyte 9	Datenbyte 8
7	D11	D10	Datenbyte 11	Datenbyte 10
8	D13	D12	Datenbyte 13	Datenbyte 12
9	D15	D14	Datenbyte 15	Datenbyte 14
10	D17	D16	Datenbyte 17	Datenbyte 16
11	D19	D18	Datenbyte 19	Datenbyte 18

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	C0	nicht genutzt	Steuerbyte
1	C1	OP	Erweitertes Steuerbyte	Opcode
2	D1	D0	Datenbyte 1	Datenbyte 0
3	D3	D2	Datenbyte 3	Datenbyte 2
4	D5	D4	Datenbyte 5	Datenbyte 4
5	D7	D6	Datenbyte 7	Datenbyte 6
6	D9	D8	Datenbyte 9	Datenbyte 8
7	D11	D10	Datenbyte 11	Datenbyte 10
8	D13	D12	Datenbyte 13	Datenbyte 12
9	D15	D14	Datenbyte 15	Datenbyte 14
10	D17	D16	Datenbyte 17	Datenbyte 16
11	D19	D18	Datenbyte 19	Datenbyte 18

18.2.5.19 AS-Interface-Masterklemme

750-655

Das Prozessabbild der AS-Interface-Masterklemme ist in seiner Länge einstellbar in den festgelegten Größen von 12, 20, 24, 32, 40 oder 48 Byte.

Es besteht aus einem Control- bzw. Statusbyte, einer 0, 6, 10, 12 oder 18 Byte großen Mailbox und den AS-interface Prozessdaten in einem Umfang von 0 bis 32 Byte.

Mit word-alignment belegt die AS-Interface-Masterklemme also jeweils 6 bis maximal 24 Worte im Prozessabbild.

Das erste Ein- bzw. Ausgangswort enthält das Status- bzw. Controlbyte sowie ein Leerbyte.

Daran schließen sich für die fest eingblendete Mailbox (Modus 1) die Worte mit Mailboxdaten an.

Wenn die Mailbox überlagerbar eingestellt ist (Modus 2), enthalten diese Worte Mailbox- oder Prozessdaten.

Die weiteren Worte enthalten die restlichen Prozessdaten.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool *WAGO-I/O-CHECK*.

Tabelle 130: AS-Interface-Masterklemme 750-655

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte
1	D1	D0	Mailbox (0, 3, 5, 6 oder 9 Worte) sowie Prozessdaten (0-16 Worte)	
2	D3	D2		
3	D5	D4		
...		
max. 23	D45	D44		

18.2.6 Systemklemmen

18.2.6.1 Systemklemmen mit Diagnose

750-610, -611

Die Potentialeinspeiseklemmen 750-610 und -611 mit Diagnose liefern zur Überwachung der Versorgung 2 Bits Diagnosedaten.

Tabelle 131: Systemklemmen mit Diagnose 750-610, -611

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 2 Sicherung	Diagnosebit S 1 Spannung

Binäre Platzhalterklemmen

750-622

Die binären Platzhalterklemmen 750-622 verhalten sich wahlweise wie 2-Kanal-Digitaleingangs- oder -ausgangsklemmen und belegen je nach angewählter Einstellung pro Kanal 1, 2, 3 oder 4 Bits.

Dabei werden dann entsprechend 2, 4, 6 oder 8 Bits entweder im Prozesseingangs- oder -ausgangsabbild belegt.

Tabelle 132: Binäre Platzhalterklemmen 750-622 (mit dem Verhalten einer 2 DI)

Ein- oder Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(Datenbit DI 8)	(Datenbit DI 7)	(Datenbit DI 6)	(Datenbit DI 5)	(Datenbit DI 4)	(Datenbit DI 3)	Datenbit DI 2	Datenbit DI 1

18.3 Mailboxklemmen

Der I/O-IPC unterstützt Busklemmen, die nach dem Mailboxprinzip arbeiten, wie z.B. die 750-655 AS-Interface-Masterklemmen (Prozessdatengröße max. 500 Byte) oder die 750-670 Steppercontroller-Klemmen.

19 Anhang

19.1 WagoConfigToolLIB.lib

Mit dem Funktionsbaustein der CODESYS-Bibliothek „WagoConfigToolLIB.lib“ konfigurieren und parametrieren Sie den I/O-IPC wie mit dem WBM und „IPC Configuration Tool“. Dazu benötigen Sie die Aufrufe aus dem Kapitel „Aufrufe zur Bibliothek ‚WagoConfigToolLIB.lib‘“. Diese haben keine Auswirkung auf die Laufzeit der Tasks, da die Funktionen asynchron aufgerufen werden.

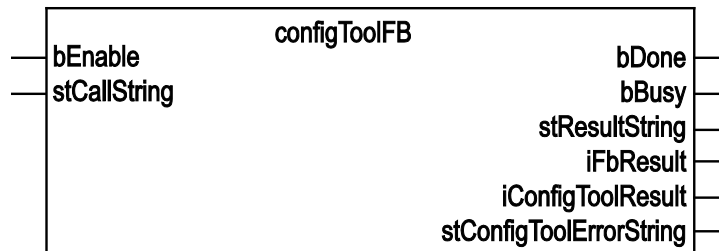


Abbildung 134: Grafische Darstellung des Funktionsbausteins „ConfigToolFB“

Tabelle 133: Funktionsbaustein „ConfigTool“

Parameter	Name	Datentyp	Beschreibung
Input	bEnable	BOOL	Der Funktionsbaustein startet mit der Verarbeitung, wenn an dem Eingang eine steigende Flanke registriert wird.
	stCallString	STRING (250)	Geben Sie dort einen Aufruf so ein wie in der Linux-Konsole. Eine Übersicht der Aufrufe erhalten Sie im Kapitel „Aufrufe zur Bibliothek ‚WagoConfigToolLIB.lib‘“.
Output	bDone	BOOL	Anzeige, ob der Funktionsbaustein vollständig ausgeführt wird oder nach Meldung eines Fehlercodes abgebrochen wurde.
	bBusy	BOOL	Der Funktionsbaustein bearbeitet einen Aufruf.
	stResultString	STRING (80)	Rückgabewert, der auf der Linux-Konsole angezeigt wird.
Output	iFbResult	INT	Die Rückgabewerte haben folgende Bedeutung: 0: kein Fehler 1: unzulässiger Eingabeparameter, z.B. ein Leerstring bei stCallString 2: Fehler bei der Ausführung des Funktionsbausteins 3: Unbekanntes Configtool 4: Der Ergebnisstring des Configtools ist zu groß für den FB-Rückgabeparameter stResultString
	iConfigToolResult	INT	Der Wert entspricht direkt dem Rückgabeparameter des Configtools.
	stConfigToolErrorString	STRING (150)	Dieser Parameter zeigt die Fehlerbeschreibung an, wenn der Aufruf „iConfigToolResult“ einen Wert $\neq 0$ liefert.

Tabelle 134: Funktion STRING_TO_IP

Parameter	Name	Datentyp	Beschreibung
Input	stIpAddress	STRING (15)	String mit der IP-Adresse in der Form xxx.xxx.xxx.xxx
In-/Output	ipAddress	ARRAY [0..3] OF BYTE	Array mit den Werten der einzelnen Bytes der IP-Adresse.
Return Value	-	BYTE	Status-Rückmeldung (Fehlercode). 0 = kein Fehler 255 = unzulässiger Parameter

Tabelle 135: Funktion IP_TO_STRING

Parameter	Name	Datentyp	Beschreibung
Input	ipAddress	ARRAY [0..3] OF BYTE	Array mit den Werten der einzelnen Bytes der IP-Adresse.
In-/Output	stIpAddress	STRING (15)	String mit der IP-Adresse in der Form xxx.xxx.xxx.xxx
Return Value	-	BYTE	Status-Rückmeldung (Fehlercode). Immer 0 = kein Fehler

19.1.1 Aufrufe zur Bibliothek „WagoConfigToolLIB“

Die nachfolgende Tabelle erläutert die Aufrufe, die es Ihnen ermöglichen, über den Funktionsbaustein „ConfigToolFB“ (siehe Parameter „stCallString“) den I/O-IPC aus dem SPS-Programm oder aus Linux heraus zu konfigurieren und zu parametrieren. Dies ist neben WBM und „IPC Configuration Tool“ eine weitere Variante, den I/O-IPC für betriebliche Anforderungen zu konfigurieren.

Das Konfigurationsverzeichnis unter Linux lautet: `/etc/config-tools/`

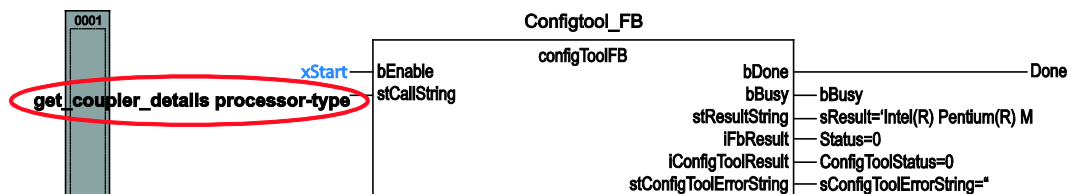


Abbildung 135: Grafische Darstellung des Funktionsbausteins „ConfigToolFB“

Tabelle 136: Beschreibung der Konfigurierungsskripte zu „Information“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Information				
Coupler Details: Ermittelt diverse Informationen des I/O-IPC				
Order Number	read	get_coupler_details order-number	Bestellnummer des I/O-IPC	sofort
Processor Type	read	get_coupler_details processor-type	Prozessortyp des I/O-IPCs	
Fieldbus Type	read	get_coupler_details fieldbus-type	Feldbustyp des I/O-IPCs	
Firmware Revision	read	get_coupler_details firmware-revision	Firmware-Version des I/O-IPCs	
Licence Information	read	get_coupler_details license-information	CODESYS-Lizens- Information	
Kbus FW Revision	read	get_coupler_details kbus-fw-revision	Firmware-Version des Klemmenbus-Controllers	
CODESYS Webserver Version	read	get_coupler_details codesys-webserver- version	CODESYS Webserver Version	
Network Details Eth0: Ermittelt die aktuell benutzten Parameter der ETHERNET-Schnittstellen				
State	read	get_actual_eth_config eth0 state	Status der Schnittstelle: enabled disabled	sofort
Mac Address	read	get_actual_eth_config eth0 mac-address	Anzeige der MAC-Adresse	
IP Adress	read	get_actual_eth_config eth0 ip-address	Anzeige der aktuellen IP-Adresse	
Subnet Mask	read	get_actual_eth_config eth0 subnet-mask	Anzeige der aktuellen Subnet-Maske	
Network Details Eth1				
Siehe „Network Details Eth0“. Bei den Aufrufen jeweils eth0 durch eth1 ersetzen.				

Tabelle 137: Beschreibung der Konfigurierungsskripte zu „CODESYS“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
CODESYS				
Project Details				
Date	read	get_rts_info project date	Anzeige der in CODESYS angegebenen Projektinformationen (Menü > Projekt > Projektinformationen)	sofort
Title	read	get_rts_info project title		
Version	read	get_rts_info project version		
Author	read	get_rts_info project author		
Description	read	get_rts_info project description		
CODESYS State				
State	read	get_rts_info state	Anzeige des CODESYS- Status (RUN oder STOP)	sofort

Tabelle 138: Beschreibung der Konfigurierungsskripte zu „TCP/IP“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
TCP/IP				
Common Configuration Data				
Hostname	read	get_coupler_details hostname	Anzeige des Hostnamens	sofort
	write	change_hostname hostname=<String>	Änderung des Hostnamens. Geben Sie für <String> einen Hostnamen an.	nach Neustart
Default Gateway				
„Default Gateway“- Schnittstelle	read	get_coupler_details default-gateway	Anzeige des eingestellten Standard-Gateways	sofort
	write	config_default_gateway interface=<Wert>	Hier wählen Sie die Schnittstelle aus, die Sie als Standard-Gateway nutzen möchten. Eingaben für <Wert> sind: eth0 eth1 none (kein Standard- Gateway ausgewählt)	
Default Gateway Value	read	get_eth_config eth0 default-gateway get_eth_config eth1 default-gateway	Anzeige der Adresse des Standard-Gateways. Führt beides zum gleichen Ergebnis, da der Wert immer gleichzeitig für beide Schnittstellen geschrieben wird.	sofort
	write	config_default_gateway default-gateway- value=<Wert>	Hier stellen Sie die Adresse des Standard-Gateways ein. Der <Wert> ist eine IP- Adresse im Format Zahl. Zahl. Zahl. Zahl.	nach Neustart
DNS-Server				
Domain Name	read	get_coupler_details domain-name	Anzeige des Domainnamens	sofort
	write	edit_dns_server domain-name=<String>	Änderung des Domainnamens. Geben Sie für <String> den Domainnamen an.	

Tabelle 138: Beschreibung der Konfigurierungsskripte zu „TCP/IP“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
DNS-Server 1	read	get_dns_server 1	DNS-Server-Adresse mit der laufenden Nummer 1.	sofort
	write/ change	edit_dns_server dns-server-nr=1 change=change dns-server-name=<Wert>	Hier stellen Sie die Adresse des DNS-Servers mit der laufenden Nummer 1 ein. Der <Wert> ist eine IP-Adresse im Format Zahl.Zahl.Zahl.Zahl .	
	write/ delete	edit_dns_server dns-server-nr=1 delete=delete	Hier löschen sie den DNS-Server mit der laufenden Nummer 1.	
DNS-Server 2-n	Siehe „DNS-Server“ 1. Bei den Aufrufen jeweils die Servernummer anpassen (hochzählen).			sofort
Add DNS-Server	write	edit_dns_server add=add dns-server-name=<Wert>	Hier fügen Sie weitere DNS-Adressen hinzu. Der <Wert> ist eine IP-Adresse im Format Zahl.Zahl.Zahl.Zahl .	sofort
TCP/IP Configuration Eth0				
State				
Device ID Eth0	read	get_eth_config eth0 device-id	Device-ID der ETHERNET-Schnittstelle: X8 X9	sofort
State Eth0	read	get_eth_config eth0 state	Status der ETHERNET-Schnittstelle Eth0: enabled disabled	
	write	config_interfaces interface=eth0 config-type=static state=enabled	Schnittstelle ausschalten: disabled Beim Einschalten ist immer der config-type mit anzugeben.	

Tabelle 138: Beschreibung der Konfigurierungsskripte zu „TCP/IP“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Type of IP address configuration Eth0	read	get_eth_config eth0 config-type	Weg, über den die Schnittstelle ihrer IP-Adresse erhält: static (statisch eingestellt), dhcp (per DHC) oder bootp (per BootP)	sofort
	write	config_interfaces interface=eth0 config-type=<Wert> state=enabled	Verfahren einschalten, über den die Schnittstelle ihrer IP-Adresse erhält. Eingaben für <Wert> sind: static (statisch eingestellt), dhcp (per DHC) oder bootp (per BootP)	
IP address Eth0	read	get_eth_config eth0 ip-address	Für die Verwendung einer statischen IP-Adresse (Static IP) eingestellte Adresse.	
	write	config_interfaces interface=eth0 ip-address=<Wert>	IP-Adresse für Static IP ändern. Der <Wert> muss eine IP-Adresse im Format Zahl.Zahl.Zahl.Zahl enthalten.	
Subnet Mask Eth0	read	get_eth_config eth0 subnet-mask	Für die Verwendung einer statischen IP-Adresse (Static IP) eingestellte subnet mask.	
	write	config_interfaces interface=eth0 subnet-mask=<Wert>	Subnet-Mask für Static IP ändern. Der <Wert> muss eine IP-Adresse im Format Zahl.Zahl.Zahl.Zahl enthalten.	
TCP/IP Configuration Eth1				
Siehe „TCP/IP Configuration Eth0“. Bei den Aufrufen jeweils eth0 durch eth1 ersetzen.				

Tabelle 139: Beschreibung der Konfigurierungsskripte zu „ETHERNET“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
ETHERNET				
Transmission Mode Eth0				
Autonegotiation	read	get_eth_config eth0 autoneg	Status der Autonegotiation-Funktion abfragen: on off	sofort
	write	config_interfaces eth0 autoneg=on	Autonegotiation-Funktion einschalten: on	
		config_interfaces interface=eth0 autoneg=off speed-duplex=<Wert>	Autonegotiation-Funktion ausschalten: off Hinweis: Beim Ausschalten der Autonegotiation-Funktion ist der Speed- und Duplex-Wert mit anzugeben. Eingaben für <Wert> sind: 10-half 10-full 100-half 100-full	
Speed and Duplex Settings	read	get_eth_config eth0 speed	Anzeige der ETHERNET-Geschwindigkeit	
	read	get_eth_config eth0 duplex	Anzeige des Duplex-Modus	
	write	config_interfaces interface=eth0 autoneg=off speed-duplex=<Wert>	Ändern der ETHERNET-Geschwindigkeit und des Duplex-Modus. Eingaben für <Wert> sind: 10-half 10-full 100-half 100-full	
Transmission Mode Eth1				
Siehe „Transmission Mode Eth0“. Bei den Aufrufen jeweils eth0 durch eth1 ersetzen.				

Tabelle 140: Beschreibung der Konfigurierungsskripte zu „NTP“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
NTP				
Configuration Data				
State	read	get_ntp_config state	Zustand des NTP-Servers abfragen: enabled disabled	sofort
	write	config_sntp state=<Wert>	Eingaben für <Wert> sind: enabled disabled	
Port	read	get_ntp_config port	Portnummer des NTP-Servers	
	write	config_sntp port=<Wert>	Geben Sie für <Wert> die Portnummer an.	
Time Server	read	get_ntp_config time-server	IP-Adresse des Time-Servers abfragen.	
	write	config_sntp time-server=<Wert>	IP-Adresse des Time-Servers eingeben. Der <Wert> kann eine IP-Adresse im Format Zahl.Zahl.Zahl.Zahl oder einen Domain-Namen als String enthalten.	
Update Time (seconds)	read	get_ntp_config update-time	Abfrage des Abfragezyklus des Time-Servers.	
	write	config_sntp update-time=<Wert>	Geben Sie für <Wert> den Abfragezyklus (in s) des Time-Servers an.	

Tabelle 141: Beschreibung der Konfigurierungsskripte zu „Clock“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit	
Clock					
Time and Date					
Date on device, local	read	get_clock_data date-local	Lokale Zeit und Datum	sofort	
	write	config_clock type=local date=<Datum>	Datum ändern. Das Format für <Datum> lautet: DD.MM.YYYY		
Time on device, UTC	read	get_clock_data time-utc	Uhrzeit/UTC		
	write	config_clock type=utc time=<Time>	Uhrzeit ändern, bezogen auf UTC-Zeit. Das Format für <Time> lautet: hh:mm:ss xx		
Time on device, local	read	get_clock_data time-local	Uhrzeit/Lokalzeit		
	write	config_clock type=local time=<Time>	Uhrzeit ändern, bezogen auf Lokalzeit. Das Format für <Time> lautet: hh:mm:ss xx		
12-Hour-Format	read	get_clock_data display-mode	Darstellungs-Format der Uhrzeit im 12 oder 24 Stunden-Format: 12-hour-format oder 24-hour-format		
	write	config_clock_ display_mode display-mode=<Wert>	Darstellungs-Format der Uhrzeit einstellen. Eingaben für <Wert> sind: 12-hour-format 24-hour-format		
Timezone					
TZ-String	read	get_clock_data tz-string	Aktuell eingestellte Zeitzone – originaler TZ-String wie er im Betriebssystem abgelegt ist.		nach Neustart
	write	config_timezone tz-string=<String>	TZ-String direkt ändern. Beispiel für <String>: CET-1CEST, M3.5.0/2,M10.5.0/3		

Tabelle 142: Beschreibung der Konfigurierungsskripte zu „HMI Settings“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
HMI Settings				
Screensaver				
Display-Status	read	get_touchscreen_config display-state	on – Display wird angezeigt (Screensaver ist aus). off – Display ist ausgeschaltet (Screensaver ist an).	sofort
	write	change_screen_state display-state=<Wert>	<Wert>= on – schaltet das Display sofort an. <Wert>= off – schaltet das Display sofort aus. Das Ein-/Ausschalten erfolgt unabhängig davon, ob der Screensaver aktiviert ist. Ob das Display nach einer Wartezeit wieder ausgeschaltet, bzw. nach einer Berührung am Touchscreen wieder eingeschaltet wird, hängt davon ab, ob die Screensaver-Funktionalität aktiviert ist (siehe nächster Punkt).	

Tabelle 142: Beschreibung der Konfigurationsskripte zu „HMI Settings“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Screensaver-Status	read	get_rts3scfg_value SCREENSAVER Enabled	<p>enabled – die Screensaver-Funktionalität ist aktiv geschaltet, d.h. nach der konfigurierten Wartezeit wird das Display automatisch ausgeschaltet und bei Benutzereingabe an Touchscreen oder Tastatur wieder eingeschaltet.</p> <p>disabled – die Screensaver-Funktionalität ist nicht frei geschaltet. Das Display wird somit nicht automatisch nach der Wartezeit ausgeschaltet, (oder nach Tastendruck eingeschaltet), sondern es verbleibt in dem Zustand, den der User explizit einstellt.</p>	sofort
	write	change_rts_config area=SCREENSAVER state=<Wert>	<p><Wert>=enabled – die Screensaver-Funktionalität wird frei geschaltet, d.h. nach der konfigurierten Wartezeit wird das Display automatisch ausgeschaltet.</p> <p><Wert>=disabled – die Screensaver-Funktionalität wird ausgeschaltet, d.h. der Screensaver beeinflusst das Display nicht. Es kann allerdings vom Benutzer generell ein- oder ausgeschaltet werden.</p>	
Screensaver-Wait time	read	get_rts3scfg_value SCREENSAVER WaitTime	Liefert den Zeitwert in Sekunden, nach dem bei aktiviertem Screensaver der Bildschirm ausgeschaltet wird.	
	write	change_rts_config area=SCREENSAVER WaitTime=<Wert>	<p>Ändert den Zeitwert, nach dem bei aktiviertem Bildschirmschoner der Bildschirm ausgeschaltet wird.</p> <p><Wert> = Integer, Zeit in Sekunden</p>	

Tabelle 142: Beschreibung der Konfigurierungsskripte zu „HMI Settings“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Cleanmode-Status	read	get_touchscreen_config cleanmode-state	on – Cleanmode ist aktuell eingeschaltet, d.h. Berührungen auf dem Touchscreen werden für die Dauer der eingestellten Timeout-Zeit ignoriert. Auf dem Screen wird ein Hinweisbild angezeigt. off – Cleanmode ist aktuell nicht eingeschaltet, Berührungen auf dem Touchscreen werden verarbeitet.	sofort
	write	change_screen_state cleanmode-state=<Wert>	<Wert>= on : - der Cleanmode wird aktiviert für die im Timeout angegebene Zeitdauer. <Wert>= off : - wenn der Cleanmode gerade aktiv ist, wird dieser wieder ausgeschaltet, ohne auf den Ablauf der Timeout-Zeit zu warten.	
Cleanmode-Timeout	read	get_rts3scfg_value CLEANMODE Timeout	Liefert den eingestellten Timeout-Wert des Cleanmodes in Sekunden, d.h. wenn der Cleanmode aktiviert wird, werden Benutzereingaben am Touchscreen für diese Zeitdauer ignoriert.	
	write	change_rts_config area=CLEANMODE Timeout=<Wert>	<Wert>=Integer, Zeitwert in Sekunden. Ändert den Timeout-Wert des Cleanmodes.	
VGA-Configuration				
Video-Mode	read	show_video_mode string	Anzeige des konfigurierten Video-Mode (Auflösung und Farbtiefe).	sofort
	write	change_video_mode video-string=<Wert>	Video-Mode ändern: Mögliche Eingaben für <Wert> sind z.B.: 640x480+256+color 800x600+16+bit 1024x768+32+bit ... (je nach unterstützter Auflösung und Farbtiefe)	nach Neustart
Show Mouse-Pointer	read	get_touchscreen_config mouse-pointer	Einstellung Mouse-Pointer sichtbar: yes no	sofort
	write	config_mousepointer show-mouse- pointer=<Wert>	Einstellung des Mouse-Pointer ändern. Mögliche Eingaben für <Wert>: yes no	nach Neustart

Tabelle 142: Beschreibung der Konfigurationsskripte zu „HMI Settings“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Touchscreen Configuration				
Device-Name	read	get_touchscreen_ config device-name	Gerätenamen auslesen	sofort
Driver-Name	read	get_touchscreen_ config driver-name	Treibernamen auslesen	
Execute calibration of touch-screen at next start	read	get_touchscreen_config calibrate-touchscreen-flag	Liefert Text checked , falls Kalibrierung eingestellt ist.	sofort
	write	config_touchscreen calibrate- touchscreen=<Wert>	Kalibrieren des Touchscreen beim nächsten Starten des I/O-IPC. Eingaben für <Wert> sind: yes no	nach Neustart
Keyboard Layout				
Keyboard Layout	read	get_coupler_details keyboard-layout	Keyboard layout: German English	sofort
	write	change_keyboard_layout keyboard-layout=<Wert>	Eingaben für <Wert> sind: German English	

Tabelle 143: Beschreibung der Konfigurationsskripte zu „Administration“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Administration				
Configuration of Serial Interface				
Configuration of serial interface	read	get_coupler_details RS232-owner	Benutzer der seriellen Schnittstelle (RS-232). Mögliche Werte sind: CODESYS IO-Check MODBUS Linux None	sofort
	write	config_RS232 owner=<Wert>	Benutzer der seriellen Schnittstelle (RS-232). Eingaben für <Wert> sind: CODESYS IO-Check MODBUS Linux None	

Tabelle 143: Beschreibung der Konfigurierungsskripte zu „Administration“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
File System Check				
Filesystem Check	write	filesystem_check device=<Wert>	Dateisystem der angegebenen Gerätenamen oder für alle Geräte prüfen. Eingaben für <Wert> sind: hda1 hda2 hda3 hda4 hdb1 hdb2 hdb3 hdb4 all	sofort
Start Backup System				
Start Backup System	write	switch_bootloader	Umschalten des Bootloaders, damit nach dem nächsten Reboot die andere ältere Version der System-Firmware gestartet wird.	nach Neustart
Reboot IPC				
-	write	start_reboot	Neustart des I/O-IPC durchführen.	sofort

Tabelle 144: Beschreibung der Konfigurierungsskripte zu „Package Server“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Firmware Update				
Medium der aktiven Partition	read	get_filesystem_data active-partition-medium	Gibt das Medium der aktiven Partition aus (cf-card, internal-flash, usb1, usb2, ...).	
Firmware-Backup erstellen	write	firmware_backup package-settings=<Wert1> package-codesys=<Wert2> package-system=<Wert3> device-medium=<Wert4> auto-update=<Wert5>	Erstellt ein Backup des angewählten Paketes auf dem angegebenen Medium. Parameter: <Wert1> = 1 , wenn Paket Settings ausgewählt sein soll. <Wert2> = 1 , wenn Paket CODESYS Project ausgewählt sein soll. <Wert3> = 1 , wenn Paket System ausgewählt sein soll. <Wert4> = Zielmedium zum Speichern des Backups. (cf-card, internal-flash, usb1, usb2) <Wert5> = 1 , wenn das Auto-Update aktiviert werden soll. Parameter, die nicht gesetzt (1) sein sollen, können entweder gleich 0 gesetzt werden oder komplett entfallen.	sofort

Tabelle 145: Beschreibung der Konfigurierungsskripte zu „Mass Storage“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Mass Storage				
Bootflag	read	get_device_data bootflag <Wert>	Liefert den Wert des Bootflags eines Devices (0 = Device ist nicht bootable oder 1 = Device ist bootable). Geben Sie für <Wert> den Device-Namen an, z.B. hda, hdb,	sofort
	write	change_bootflag device=<Wert1> bootflag=<Wert2>	Bootflag eines Devices setzen oder rücksetzen. <Wert1> = Device-Name, z.B. hda, hdb, ... <Wert2> = 0 (Bootflag rücksetzen), 1 (Bootflag setzen).	

Tabelle 146: Beschreibung der Konfigurierungsskripte zu „Port“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Port				
Telnet				
Telnet Port	read	get_port_state telnet	Status des Telnet-Servers auslesen: enabled disabled	sofort
	write	config_port port=telnet state=<Wert>	Eingaben für <Wert> sind: enabled disabled	
CODESYS Webserver				
CODESYS Webserver Port	read	get_port_state codesys-webserver	Status des CODESYS- Webserver auslesen. Eingaben für <Wert> sind: enabled disabled	sofort
	write	config_port port=codesys-webserver state=<Wert>	Aktivieren/deaktivieren des CODESYS-Webservers. Eingaben für <Wert> sind: enabled disabled	
FTP				
FTP Port	read	get_port_state ftp	Status des FTP-Servers auslesen. Eingaben für <Wert> sind: enabled disabled	sofort
	write	config_port port=ftp state=<Wert>	Eingaben für <Wert> sind: enabled disabled	
CODESYS				
CODESYS Port	read	get_rts3scfg_value PLC DisableTcpIp Programming	Status des Wert für „DisableTcpIpProgrammin g“ in der CODESYS- Konfiguration abfragen: YES: CODESYS-Port wird nicht benutzt. NO: CODESYS-Port wird benutzt.	sofort
	write	change_rts_config area=PLC disable- tcpip=<Wert>	Eingaben für <Wert> sind: YES: CODESYS-Port wird nicht benutzt. NO: CODESYS-Port wird benutzt.	
CODESYS Port Number	read	get_rts3scfg_value PLC TcpIpPort	In der CODESYS- Konfiguration eingestellter Wert des TCP/IP-Ports.	
	write	change_rts_config area=PLC TcpIpPort=<Wert>	CODESYS Portnummer ändern. Geben Sie für <Wert> die TCP/IP-Portnummer an.	

Tabelle 147: Beschreibung der Konfigurationsskripte zu „MODBUS“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
MODBUS				
MODBUS/UDP				
MODBUS/UDP Status	read	get_rts3scfg_value MODBUS_UDP state	Status von MODBUS/UDP: enabled disabled	sofort
	write	change_rts_config area=MODBUS_UDP state=<Wert>	Aktivieren/deaktivieren des MODBUS/UDP-Servers. Eingaben für <Wert> sind: enabled disabled	
MODBUS/TCP				
MODBUS/TCP Status	read	get_rts3scfg_value MODBUS_TCP state	Status von MODBUS/TCP: enabled disabled	sofort
	write	change_rts_config area=MODBUS_TCP state=<Wert>	Aktivieren/deaktivieren des MODBUS/TCP-Servers. Eingaben für <Wert> sind: enabled disabled	
Timeout (msec)	read	get_rts3scfg_value MODBUS_TCP TCPTimeout	Timeout-Wert für MODBUS/TCP	sofort
	write	change_rts_config area=MODBUS_TCP timeout=<Wert>	Hier stellen Sie für <Wert> die Zeitspanne (Timeout) der MODBUS/TCP- Verbindung ein (ms), nach der die Verbindung bei einer Unterbrechung der Kommunikation automatisch beendet wird.	
MODBUS/RTU				
State	read	get_rts3scfg_value MODBUS_RTU state	Status von MODBUS/RTU. Eingaben für <Wert> sind: enabled disabled	sofort
	write	change_rts_config area=MODBUS_RTU timeout=<Wert>	Aktivieren/deaktivieren des MODBUS/RTU-Servers. Eingaben für <Wert> sind: enabled disabled	
Node-ID	read	get_rts3scfg_value MODBUS_RTU Node_ID	Node-ID für MODBUS/RTU	sofort
	write	change_rts_config area=MODBUS_RTU node-id=<Wert>	Node-ID (Zahl) für MODBUS/RTU. Geben Sie für <Wert> die Node-ID an.	
Timeout (msec)	read	get_rts3scfg_value MODBUS_RTU Timeout	Timeout-Wert für MODBUS/RTU	sofort
	write	change_rts_config area=MODBUS_RTU Timeout=<Wert>	Timeout-Wert für MODBUS/RTU ändern. Geben Sie für <Wert> den Timeout-Wert in ms an.	

Tabelle 147: Beschreibung der Konfigurierungsskripte zu „MODBUS“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Baudrate	read	get_rts3scfg_value MODBUS_RTU Baud	Baudrate für MODBUS/RTU	sofort
	write	change_rts_config area=MODBUS_RTU Baud=<Wert>	Ändern der Baudrate für MODBUS/RTU. Eingaben für <Wert> sind: 2400 4800 9600 19200 38400 57600 115200	
Databit	read	get_rts3scfg_value MODBUS_RTU Data_Bits	Data-Bit-Anzahl für MODBUS/RTU	
Parity	read	get_rts3scfg_value MODBUS_RTU Parity	Parity-Wert für MODBUS/RTU	
	write	change_rts_config area=MODBUS_RTU Parity=<Wert>	Ändern der Parity- Einstellung für MODBUS/RTU. Eingaben für <Wert> sind: None Odd Even	
Stop-Bits	read	get_rts3scfg_value MODBUS_RTU Stop_Bits	Stop-Bits für MODBUS/RTU	
	write	change_rts_config area=MODBUS_RTU Stop_Bits=<Wert>	Einstellen der Anzahl der Stop-Bits für MODBUS/RTU. Eingaben für <Wert> sind: 1 2	
Flow-Control	read	get_rts3scfg_value MODBUS_RTU Flow_control	Flow-Control-Wert für MODBUS/RTU	

Tabelle 148: Beschreibung der Konfigurationsskripte zu „Allgemeine SNMP-
Informationsparameter“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Allgemeine SNMP-Informationsparameter				
Name of device	read	get_snmp_data device-name	Gibt den SNMP-Parameter „sysName“ aus.	sofort
	write	config_snmp device-name=<Wert>	Ändern des SNMP-Parameters „sysName“ (<Wert> = String). *	nach Neustart
Description	read	get_snmp_data description	Gibt den SNMP-Parameter „sysDescr“ aus.	sofort
	write	config_snmp description=<Wert>	Ändern des SNMP-Parameters „sysDescr“ (<Wert> = String). *	nach Neustart
Physical location	read	get_snmp_data physical-location	Gibt den SNMP-Parameters „sysLocation“ aus.	sofort
	write	config_snmp physical-location=<Wert>	Ändern des SNMP-Parameters „sysLocation“ (<Wert> = String). *	nach Neustart
Contact	read	get_snmp_data contact	Gibt den SNMP-Parameters „sysContact“ aus.	sofort
	write	config_snmp contact=<Wert>	Ändern des SNMP-Parameters „sysContact“ (<Wert> = String).	nach Neustart
* Bei der Eingabe der Werte müssen die Leerzeichen entweder mit „+“ oder „%20“ aufgefüllt werden. Andernfalls wird die Eingabe nicht als zusammenhängender String erkannt.				
SNMP-Manager -Konfiguration für v1 und v2c				
Protokoll Status	read	get_snmp_data v1-v2c-state	Liefert den Status des SNMP-Protokolls für v1/v2c als String: enabled disabled	sofort
Local Community Name	read	get_snmp_data v1-v2c-community-name	Gibt den für v1/v2c eingestellten Community-Namen aus.	
Protokoll Status/ Community Name	write	config_snmp v1-v2c-state=<Wert1> v1-v2c-community-name=<Wert2>	Aktiviert/deaktiviert das v1/v2c-Protokoll (<Wert1> = enabled oder disabled) und vergibt einen Community-Namen. (<Wert2> = String ohne Leerzeichen, min. 1, max. 32 Zeichen). Hinweis: Beim Ausschalten ist kein Community-Name erforderlich. Das Einschalten ist nur mit der Angabe eines Community-Namens möglich. Das Speichern des Community-Namens ist nur bei aktiviertem Protokoll möglich.	nach Neustart
SNMP-Trap-Receiver-Konfiguration für v1 und v2c				
Es können beliebig viele Trap-Receiver konfiguriert werden. Ein angelegter Trap-Receiver ist immer aktiv; zum Deaktivieren muss der Datensatz komplett gelöscht werden.				

Tabelle 148: Beschreibung der Konfigurierungsskripte zu „Allgemeine SNMP-
Informationsparameter“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
IP-Adresse eines Trap-Receiver	read	get_snmp_data v1-v2c-trap-receiver- address <Nummer>	Gibt die IP-Adresse des Trap-Receiver aus, zu dem der I/O-IPC die v1- oder v2- Traps senden soll. Der Parameter <Nummer> (Zahl) dient dazu, die zusammengehörigen Daten der einzelnen konfigurierten Trap-Receiver kurzfristig (ohne zwischenzeitliche Änderungen der Daten) nacheinander auslesen zu können. Es ist eine laufende Nummer, die nicht mit den Daten selbst in Verbindung steht. Wird die Nummer weggelassen, werden die Daten des ersten Receivers ausgelesen.	sofort
Community Name	read	get_snmp_data v1-v2c-trap-receiver- community-name <Nummer>	Gibt den Community-Namen aus, den der SNMP-Agent des IPCs im Trap-Header sendet. Parameter <Nummer> (Zahl) siehe Punkt „IP-Adresse eines Trap-Receiver“.	
Trap-Version	read	get_snmp_data v1-v2c-trap-receiver- version <Nummer>	Gibt die SNMP-Version aus („v1“ oder „v2c“), über die der SNMP-Agent die Traps an die zugehörige Trap-Receiver-Adresse sendet. Parameter <Nummer> (Zahl) siehe Punkt „IP-Adresse eines Trap-Receiver“.	

Tabelle 148: Beschreibung der Konfigurationsskripte zu „Allgemeine SNMP-
Informationsparameter“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Anlegen/ Löschen eines Trap-Receiver	write	<pre>config_snmp v1-v2c-trap-receiver- edit=<Wert1> v1-v2c-trap-receiver- address=<Wert2> v1-v2c-trap-receiver- community- name=<Wert3> v1-v2c-trap-receiver- version=<Wert4></pre>	<p>Einen neuen Trap-Receiver hinzufügen (Wert1=add) oder Löschen eines bereits konfigurierten Trap-Receiver (Wert1=delete).</p> <p>Weitere Parameter: <Wert2> = IP-Adresse (Zahl.Zahl.Zahl.Zahl), an die der IPC die Traps senden soll. <Wert3>: Community-String (String), den der IPC in den Header des Traps einträgt. <Wert4>: SNMP-Version, über die die Traps gesendet werden (v1 oder v2c).</p> <p>Hinweis: Auch beim Löschen eines Trap-Empfängers müssen alle Parameter mitgegeben werden, da nur darüber der Datensatz eindeutig zu identifizieren ist.</p>	nach Neustart
<p>Konfiguration von SNMP v3 Es können beliebig viele SNMP-v3-User angelegt werden. Ein angelegter User ist immer aktiv; zum Deaktivieren muss der komplette Datensatz gelöscht werden.</p>				
Authentication- Name	read	<pre>get_snmp_data v3-auth-name <Nummer></pre>	<p>Gibt den User-Namen des v3-Users aus. Der Parameter <Nummer> dient dazu, die zusammengehörigen Daten der einzelnen konfigurierten Trap-Receiver kurzfristig (ohne zwischenzeitliche Änderungen der Daten) nacheinander auslesen zu können. Es ist eine laufende Nummer, die nicht mit den Daten selbst in Verbindung steht. Wird die Nummer weggelassen, werden die Daten des ersten Users ausgelesen.</p>	sofort
Authentication- Verschlüs- selungs-Typ	read	<pre>get_snmp_data v3-auth-type <Nummer></pre>	<p>Gibt den Verschlüsselungstyp aus, den der v3-User benutzt (none, MD5 oder SHA). Parameter <Nummer> siehe Punkt „Authentication-Name“.</p>	

Tabelle 148: Beschreibung der Konfigurationsskripte zu „Allgemeine SNMP-
Informationsparameter“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Authentication-Schlüssel	read	get_snmp_data v3-auth-key <Nummer>	Gibt den Schlüssel-String für die Authentication aus. Parameter <Nummer> siehe Punkt „Authentication-Name“.	sofort
Privacy-Verschlüsselungs-Typ	read	get_snmp_data v3-privacy <Nummer>	Gibt den Privacy-Verschlüsselungstyp des v3-Users aus (none , DES oder AES). Parameter <Nummer> siehe Punkt „Authentication-Name“.	
Privacy-Schlüssel	read	get_snmp_data v3-privacy-key <Nummer>	Gibt den Schlüssel-String für Privacy aus. Ist hier nichts angegeben, wird der SNMP-Agent hierfür den „Authentication Key“ benutzen. Parameter <Nummer> siehe Punkt „Authentication-Name“.	
Trap-Receiver-Adresse	read	get_snmp_data v3-notification-receiver <Nummer>	IP-Adresse eines SNMP-Managers, an den der Agent Traps für diesen v3-User sendet. Ist hier nichts angegeben, werden für diesen User keine Traps gesendet. Parameter <Nummer> siehe Punkt „Authentication-Name“.	

Tabelle 148: Beschreibung der Konfigurierungsskripte zu „Allgemeine SNMP-
Informationsparameter“

Parameter	Status	Aufruf	Ausgabe/Eingabe	Gültigkeit
Add new v3-User	write	<pre>config_snmp v3-edit=add v3-auth-name=<Wert1> v3-auth-type=<Wert2> v3-auth-key=<Wert3> v3-privacy=<Wert4> v3-privacy-key=<Wert5> v3-notification-receiver=<Wert6></pre>	<p>Anlegen eines neuen v3-Users. v3-auth-name: User-Name, String ohne Leerzeichen, maximal 32 Zeichen. Der User-Name darf noch nicht vergeben worden sein.</p> <p>Parameter: User-Name (<Wert1> = String) Verschlüsselungstyp. (<Wert2> = none, MD5 oder SHA). Schlüssel-String für die Authentifizierung, (<Wert3> = String mit mindestens 8 und maximal 32 Zeichen) Privacy-Verschlüsselungstyp (<Wert4> = none, DES oder AES). Privacy-Schlüssel-String (<Wert5> = String, mindestens 8 und maximal 32 Zeichen), kann leer sein; in diesem Fall wird der Authentication-Key verwendet. Als Notification Receiver (<Wert6> = zahl.zahl.zahl.zahl) wird die IP-Adresse eines Trap-Empfängers übertragen. Sollen keine v3-Traps gesendet werden, entfällt diese Angabe.</p>	nach Neustart
Delete v3-User	write	<pre>config_snmp v3-edit=delete v3-auth-name=<Wert></pre>	<p>Löschen eines vorhandenen v3-Users. Da beim Anlegen eines Users die doppelte Vergabe desselben User-Namens vom Skript unterbunden wird, reicht beim Löschen der Name, um einen Datensatz eindeutig zu identifizieren (<Wert> = String).</p>	

19.2 WagoLibNetSnmp.lib

Die Bibliothek WagoLibNetSnmp.lib ist eine externe CODESYS-Bibliothek. Sie dient dem Anlegen kundenspezifischer OIDs und Setzen/Abfragen deren Werte vom SPS Programm aus.

Dafür stehen folgende Funktionen zur Verfügung:

- Anlegen/Registrieren: snmpRegisterCustomOID_XXXXX
- Abfragen: snmpGetValueCustomOID_XXXXX
- Setzen: snmpSetValueCustomOID_XXXXX

Hinweis



OID-Variablen

Bereits angelegte OID-Variablen bleiben solange bestehen bis das System neugestartet wurde bzw. das Programm neu auf die Steuerung geladen wurde. Bei einem "Online-Change" bleiben die OIDs erhalten.

Hinweis



Zur Verfügung stehender Variablenspeicher

Es stehen 8 kB an Variablenspeicher zur Verfügung. Somit können Sie max. 32 OIDs vom Typ „Octet String“ anlegen bzw. 2048 OIDs vom Typ „Integer“ oder „Gauge32“.

Variablen

Folgende Datentypen werden unterstützt:

Tabelle 149: Datentypen

OID-Datentyp	CODESYS-Datentyp	Länge (in Bytes)
Integer	DINT	4
UInteger, Gauge32	UDINT, DWORD	4
Octet String	STRING(255)	255

Funktionen

Funktionen zum Registrieren kundenspezifischer OIDs:

Eine OID kann nur einmal registriert werden. Ein mehrfacher Aufruf dieser Funktion für gleiche OIDs liefert den Fehler:

2 (RET_ERR_OID_ALREADY_REGISTERED)

19.2.1 snmpRegisterCustomOID_INT32()

Registrieren einer OID vom Datentyp: Integer (4 Byte)

Tabelle 150: Parameter snmpRegisterCustomOID_INT32()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: 1.3.6.1.4.1.8072.2.4.1.1.1.0
iDefValue : DINT	default value i.e.: 2
bReadOnly : BOOL	true, if read-only variable

Tabelle 151: Return snmpRegisterCustomOID_INT32()

Return	Beschreibung
WORD	error number: 0 = ok

```

FUNCTION snmpRegisterCustomOID_INT32 : WORD
VAR_INPUT
    sOID : STRING(128) := '';
    iDefValue : DINT := 0;
    bReadOnly : BOOL := FALSE;
END_VAR

```

19.2.2 snmpRegisterCustomOID_STRING()

Registrieren einer OID vom Datentyp: Octet String (255 Byte)

Tabelle 152: Parameter snmpRegisterCustomOID_STRING()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
sDefValue : STRING(255)	default value i.e.: 'hallo world'
bReadOnly : BOOL	true, if read-only variable

Tabelle 153: Return snmpRegisterCustomOID_STRING()

Return	Beschreibung
WORD	error number: 0 = ok

```

FUNCTION snmpRegisterCustomOID_STRING : WORD
VAR_INPUT
    sOID : STRING(128) := '';
    sDefValue : STRING(255) := '';
    bReadOnly : BOOL := FALSE;
END_VAR

```

19.2.3 snmpRegisterCustomOID_UINT32()

Registrieren einer OID vom Datentyp: UInteger, Gauge32 (4 Byte)

Tabelle 154: Parameter snmpRegisterCustomOID_UINT32()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
uiDefValue : UDINT;	default value i.e.: 2
bReadOnly : BOOL	true, if read-only variable

Tabelle 155: Return snmpRegisterCustomOID_UINT32()

Return	Beschreibung
WORD	error number: 0 = ok

```
FUNCTION snmpRegisterCustomOID_UINT32 : WORD
VAR_INPUT
    sOID : STRING(128) := '';
    uiDefValue : UDINT := 0;
    bReadOnly : BOOL := FALSE;
END_VAR
```

19.2.4 snmpGetValueCustomOID_INT32()

Abfragen des Wertes einer OID vom Datentyp: Integer (4 Byte)

Tabelle 156: Parameter snmpGetValueCustomOID_INT32()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
iValue : DINT	OID value

Tabelle 157: Return snmpGetValueCustomOID_INT32()

Return	Beschreibung
WORD	error number: 0 = ok
iValue : DINT	OID value

```

FUNCTION snmpGetValueCustomOID_INT32 : WORD
VAR_INPUT
    sOID : STRING(128) := '';
END_VAR
VAR_IN_OUT
    iValue : DINT;
END_VAR

```

19.2.5 snmpGetValueCustomOID_STRING()

Abfragen des Wertes einer OID vom Datentyp: Octet String (255 Byte)

Tabelle 158: Parameter snmpGetValueCustomOID_STRING()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
sValue : STRING(255);	OID value

Tabelle 159: Return snmpGetValueCustomOID_STRING()

Return	Beschreibung
WORD	error number: 0 = ok
sValue : STRING(255);	OID value

```
FUNCTION snmpGetValueCustomOID_STRING : WORD
VAR_INPUT
    sOID : STRING(128) := '';
END_VAR
VAR_IN_OUT
    sValue : STRING(255);
END_VAR
```

19.2.6 snmpGetValueCustomOID_UINT32()

Abfragen des Wertes einer OID vom Datentyp: UInteger, Gauge32 (4 Byte)

Tabelle 160: Parameter snmpGetValueCustomOID_INT32()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
uiValue : UDINT;	OID value

Tabelle 161: Return snmpGetValueCustomOID_INT32()

Return	Beschreibung
WORD	error number: 0 = ok
uiValue : UDINT;	OID value

```

FUNCTION snmpGetValueCustomOID_INT32 : WORD
VAR_INPUT
    sOID : STRING(128) := '';
END_VAR
VAR_IN_OUT
    uiValue : UDINT := 0;
END_VAR

```

19.2.7 snmpSetValueCustomOID_INT32()

Setzen des Wertes einer OID vom Datentyp: Integer (4 Byte)

Tabelle 162: Parameter snmpSetValueCustomOID_INT32()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
iValue : DINT	new default value i.e.: 2

Tabelle 163: Return snmpSetValueCustomOID_INT32()

Return	Beschreibung
WORD	error number: 0 = ok

```
FUNCTION snmpRegisterCustomOID_INT32 : WORD
VAR_INPUT
    sOID : STRING(128) := '';
    iDefValue : DINT := 0;
END_VAR
```

19.2.8 snmpSetValueCustomOID_STRING()

Setzen des Wertes einer OID vom Datentyp: Octet String (255 Byte)

Tabelle 164: Parameter snmpSetValueCustomOID_STRING()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
sDefValue : STRING(255)	new value i.e.: 'hallo world'

Tabelle 165: Return snmpSetValueCustomOID_STRING()

Return	Beschreibung
WORD	error number: 0 = ok

```

FUNCTION snmpRegisterCustomOID_STRING : WORD
VAR_INPUT
    sOID : STRING(128) := '';
    sDefValue : STRING(255) := '';
END_VAR

```

19.2.9 snmpSetValueCustomOID_UINT32()

Setzen des Wertes einer OID vom Datentyp: UInteger, Gauge32 (4 Byte)

Tabelle 166: Parameter snmpSetValueCustomOID_UINT32()

Parameter	Beschreibung
sOID : STRING(128)	numerical OID i.e.: .1.3.6.1.4.1.8072.2.4.1.1.1.0
uiDefValue : UDINT;	new value i.e.: 2

Tabelle 167: Return snmpSetValueCustomOID_UINT32()

Return	Beschreibung
WORD	error number: 0 = ok

```
FUNCTION snmpRegisterCustomOID_UINT32 : WORD
VAR_INPUT
    sOID : STRING(128) := '';
    uiDefValue : UDINT := 0;
END_VAR
```

19.2.10 Rückmeldungen

Folgende Fehlermeldungen können von den Funktionen zurückgeliefert werden:

Tabelle 168: Fehlermeldungen

Wert	Definition	Beschreibung
0	RET_SUCCESS	Alles ok, keine Fehler
1	RET_ERR_WRONG_OID	Falsche OID, nur numerische Variablen werden unterstützt z. B: .1.3.6.1.4.1.8072.2.4.1.1.1.0 Dabei sind maximal 32 Punkte zulässig. Der höchste numerische Wert darf nur $2^{31}-1 = 2147483647$ betragen.
2	RET_ERR_OID_ALREADY_REGISTERED	OID ist bereits registriert
3	RET_ERR_OID_NOT_FOUND	OID ist nicht registriert -> OID über die snmpRegisterOID_xxx – Funktion registrieren
4	RET_ERR_IPC_COMM_NOT_INITIALIZED	Kommunikation zwischen der PLC-Runtime-Umgebung und dem Net-SNMP Agenten ist gestört -> System neu starten
5	RET_ERR_IPC_COMM_FAILED	Nicht genügend Variablenspeicher vorhanden -> nur 8 kB großer Speicherbereich steht zur Verfügung

19.2.11 Beispielprogramm „Test.pro“

Das Beispiel-Programm „Test.pro“ veranschaulicht das Registrieren, Abfragen und Setzen kundenspezifischer OID's:

Programmvariablen

```
PROGRAM PLC_PRG
VAR

    (* Flags *)
    bRegisterOID_INT32:BOOL := FALSE;
    bRegisterOID_STRING:BOOL := FALSE;
    bRegisterOID_UINT32:BOOL := FALSE;
    bSetValueOID_INT32:BOOL := FALSE;
    bSetValueOID_STRING:BOOL := FALSE;
    bSetValueOID_UINT32:BOOL := FALSE;
    bGetValueOID_INT32:BOOL := FALSE;
    bGetValueOID_STRING:BOOL := FALSE;
    bGetValueOID_UINT32:BOOL := FALSE;

    (*CustomOIDs *)
    sCustomOID1:STRING(128) := '1.3.6.1.4.1.8072.2.4.1.1.1.0'; (* Integer32 *)
    sCustomOID2:STRING(128) := '1.3.6.1.4.1.8072.2.4.1.1.2.0'; (* OctetString *)
    sCustomOID3:STRING(128) := '1.3.6.1.4.1.8072.2.4.1.1.3.0'; (* UInteger32 *)

    (* Values *)
    iValue:DINT := 11;
    sValue:STRING(255) := 'test';
    uiValue:UDINT := 33;

    (* Error *)
    wError:WORD := 0;

END_VAR
```

Programmblock

```
(* Register new OID with Integer value *)
IF bRegisterOID_INT32 = TRUE THEN
    wError := snmpRegisterCustomOID_INT32(sOID1, iValue, FALSE);
    bRegisterOID_INT32 := FALSE;
END_IF;

(* Register new OID with OctetString value*)
IF bRegisterOID_STRING = TRUE THEN
    wError := snmpRegisterCustomOID_STRING(sOID2, sValue, FALSE);
    bRegisterOID_STRING := FALSE;
END_IF;

(* Register new OID with UInteger value *)
IF bRegisterOID_UINT32 = TRUE THEN
    wError := snmpRegisterCustomOID_UINT32(sOID3, uiValue, FALSE);
    bRegisterOID_UINT32 := FALSE;
END_IF;

(* Set Integer value *)
IF bSetValueOID_INT32 = TRUE THEN
    wError := snmpSetValueCustomOID_INT32(sOID1, iValue+1);
    bSetValueOID_INT32 := FALSE;
END_IF;

(* Set OctetString value *)
IF bSetValueOID_STRING = TRUE THEN
    sValue := 'hello wolrd';
    wError := snmpSetValueCustomOID_STRING(sOID2, sValue);
    bSetValueOID_STRING := FALSE;
END_IF;

(* Set UInteger value *)
IF bSetValueOID_UINT32 = TRUE THEN
    wError := snmpSetValueCustomOID_UINT32(sOID3, uiValue+1);
    bSetValueOID_UINT32 := FALSE;
END_IF;

(* Get Integer value *)
IF bGetValueOID_INT32 = TRUE THEN
    wError := snmpGetValueCustomOID_INT32(sCustomOID1, iValue);
    bGetValueOID_INT32 := FALSE;
END_IF;

(* Get OctetString value *)
IF bGetValueOID_STRING = TRUE THEN
    wError := snmpGetValueCustomOID_STRING(sOID2, sValue);
    bGetValueOID_STRING := FALSE;
END_IF;

(* Get UInteger value *)
```

```
IF bGetValueOID_UINT32 = TRUE THEN
    wError := snmpGetValueCustomOID_UINT32(sOID3, uiValue);
    bGetValueOID_UINT32 := FALSE;
END_IF;
```

Über das Visualisierungsformular „TEST“ können die CustomOID's registriert, abgefragt und gesetzt werden:

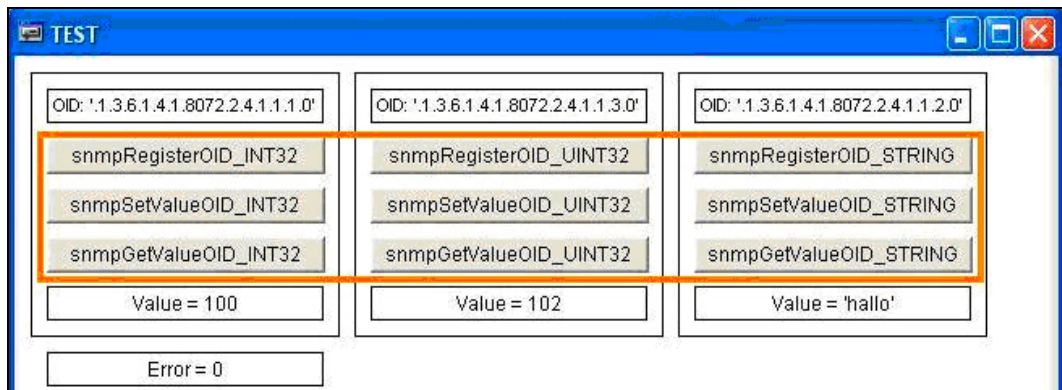


Abbildung 1: Visualisierungsformular "TEST"

19.3 WAGO_CANopen_02.lib

Diese Bibliothek enthält Funktionsbausteine, die nach dem CiA Draft Standard 405 spezifiziert sind.

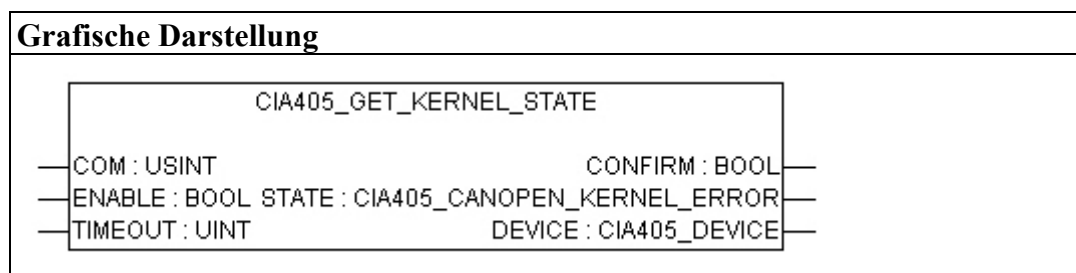
19.3.1 CIA405_GET_KERNEL_STATE

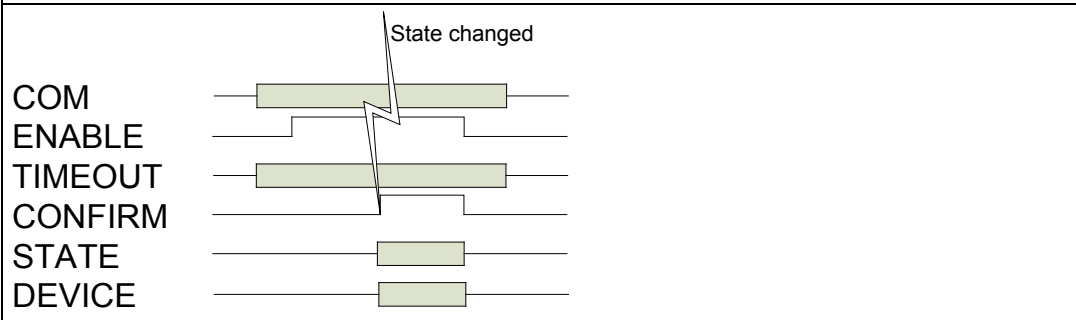
Der Funktionsbaustein CIA405_GET_CANOPEN_STATE gibt den Zustand des CANopen-Kerns aus.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_KERNEL_STATE
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Aus Kompatibilitätsgründen implementiert. Eingang wird nicht erfasst.

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
STATE	CIA405_CANOPEN_KERNEL_ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ERROR
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE



Zeitliches Verhalten**Beschreibung**

Der Status „Anderer Fehler“ der *CIA405_CANOPEN_KERNEL_ERROR* Struktur wird an diesem Baustein nicht ausgegeben, da hier der Baustein mit dem Ausgang „*CIA405_SDO_ERROR*“, der weitergehende Informationen liefert, nicht belegt ist.

Dieser Baustein kann die folgenden Errorstatus der *CANOPEN_KERNEL_ERROR* Struktur ausgeben:

- 0010h CAN-Controller im Zustand „Bus off“
- 0011h CAN-Controller hat den “Error warning limit” überschritten
- 0021h Keine Antwort vom Knoten erhalten
- 0026h Interner Fehler.
- 0027h Node-Guarding-Fehler
- 0028h Device nicht im Operational Mode

Wechselt der *ENABLE*-Eingang auf *TRUE*, ist die Überwachungsfunktion aktiviert. Mit jedem SPS-Zyklus wird der *CIA405_GET_CANOPEN_STATE* überprüft. *CONFIRM* gibt *TRUE* aus, wenn ein Errorstatus lt. Auflistung detektiert wird.

Dieser Status wird am Ausgang *STATE* ausgegeben.

Sofern vorhanden wird zusätzlich das erste Device angegeben, bei dem dieser Status erkannt wurde.

Das Verlassen des Operational-Modus eines Nodes kann nur detektiert werden, wenn Node Guarding für das Device aktiviert ist.

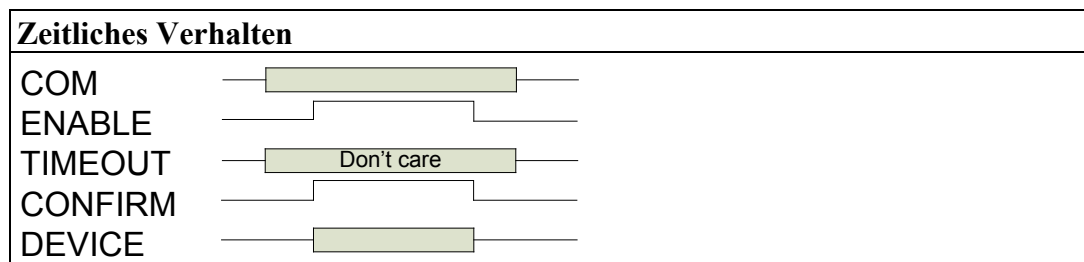
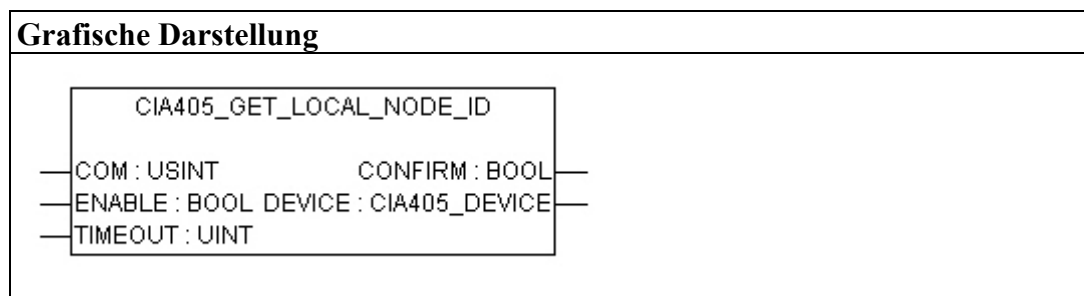
19.3.2 CIA405_GET_LOCAL_NODE_ID

Der Funktionsbaustein CIA405_GET_LOCAL_NODE_ID gibt die eigene Modul-ID zurück.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Aus Kompatibilitätsgründen implementiert. Eingang wird nicht erfasst.

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE



Beschreibung

Nach Setzen des *ENABLE*-Eingangs wechselt *CONFIRM* auf den Status *TRUE* und der Ausgang *DEVICE* stellt die Modul-Id zur Verfügung.

Es werden intern keine blockierenden Zugriffe durchgeführt. Die Verwendung des Timeout-Eingangs ist damit nicht erforderlich. Dieser Eingang ist, ebenso wie der *COM*-Eingang, aus Kompatibilitätsgründen implementiert.

Mit dem Rücksetzen des *ENABLE*-Eingangs wird der Baustein wieder in den Grundzustand versetzt.

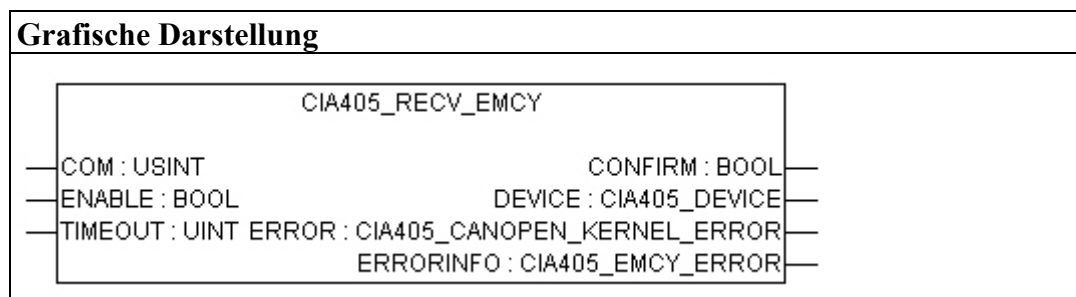
19.3.3 CIA405_RECV_EMCY

Der Funktionsbaustein CIA405_RECV_EMY prüft, ob eine Emergency-Nachricht von einem beliebigen konfigurierten Device empfangen wurde.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Aus Kompatibilitätsgründen implementiert. Eingang wird nicht erfasst.

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	TRUE: Emergency empfangen FALSE: keine Emergency empfangen
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
ERROR	CIA405_CANOPEN_KERNEL_ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ERROR
ERRORINFO	CIA405_EMCY_ERROR	STRUCT	Siehe Abschnitt Datatypes: CIA405_EMCY_ERROR



Zeitliches Verhalten**Beschreibung**

Wird der *ENABLE*-Eingang auf *TRUE* gesetzt, ist die Überwachungsfunktion aktiviert. Mit jedem SPS-Zyklus wird der Empfang einer Emergency-Nachricht und das Auftreten eines *CANOPEN_KERNEL_ERROR* überprüft.

Vor dem ersten Aufruf des Bausteins mit *ENABLE = TRUE* werden Emergencies aus dem Diagnosespeicher gelöscht und nicht angezeigt. Im weiteren Programmablauf werden Emergencies auch gespeichert, wenn der Baustein nicht aktiviert ist. Nach der Aktivierung des Bausteins mit *ENABLE = TRUE* wird die zuletzt empfangene Emergency ausgegeben.

Bei dem Empfang eines Emergencies wird *CONFIRM* auf *TRUE* gesetzt. *ERRORINFO* beinhaltet die Emergency Informationen, die nach DS301 spezifiziert sind und der Ausgang *DEVICE* die Modul ID des Emergency-Senders.

Wird ein *CANOPEN_KERNEL_ERROR*:

0010h CAN-Controller im Zustand „Bus off“

0011h CAN-Controller hat den “Error warning limit” überschritten

0021h Keine Antwort vom Knoten erhalten

0026h Interner Fehler.

0027h Node-Guarding-Fehler

0028h Device nicht im Operational-Mode

identifiziert bleibt *CONFIRM* auf *FALSE* und es wird der Fehler am *ERROR*-Ausgang ausgegeben. Wenn vorhanden wird die Node-ID am *DEVICE*-Ausgang ausgegeben.

Es werden intern keine blockierenden Zugriffe durchgeführt. Die Verwendung des Timeout-Eingangs ist damit nicht erforderlich. Dieser Eingang ist, ebenso wie der *COM*-Eingang, aus Kompatibilitätsgründen implementiert.

Mit dem Rücksetzen des *ENABLE*-Eingangs wird der Baustein wieder in den Grundzustand versetzt.

19.3.4 CIA405_RECV_EMCY_DEV

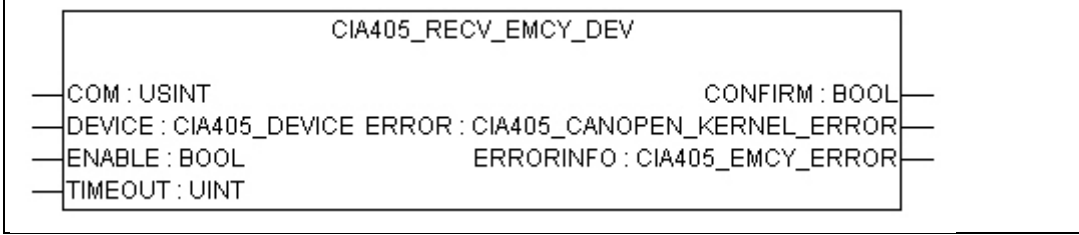
Der Funktionsbaustein CIA405_RECV_EMY_DEV prüft, ob eine Emergency-Nachricht von einem dedizierten und konfigurierten Device empfangen wurde.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

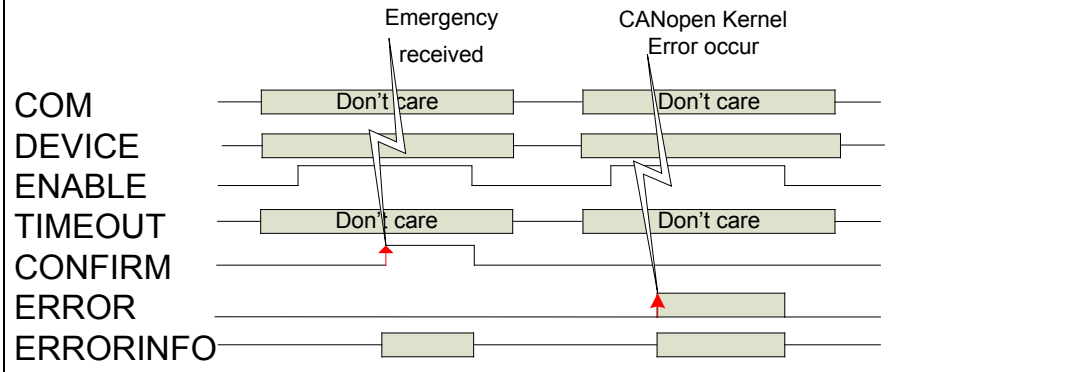
Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Maximale Ausführungszeit in ms. Disabled bei keiner oder Eingabe außerhalb des gültigen Bereiches [10..900].

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	TRUE: Emergency vom spezifizierten Modul empfangen FALSE: keine Emergency vom spezifizierten Modul empfangen
ERROR	CIA405_CANOPEN_KERNEL_ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ERROR
ERRORINFO	CIA405_EMCY_ERROR	STRUCT	Siehe Abschnitt Datatypes: CIA405_EMCY_ERROR

Grafische Darstellung



Zeitliches Verhalten



Beschreibung

Wird der *ENABLE*-Eingang auf *TRUE* gesetzt, ist die Überwachungsfunktion aktiviert. Mit jedem SPS-Zyklus wird der Empfang einer Emergency-Nachricht und das Auftreten eines *CANOPEN_KERNEL_ERROR* überprüft.

Vor dem ersten Aufruf des Bausteins mit *ENABLE = TRUE* werden Emergencies aus dem Diagnosespeicher gelöscht und nicht angezeigt. Im weiteren Programmablauf werden Emergencies auch gespeichert, wenn der Baustein nicht aktiviert ist. Nach der Aktivierung des Bausteins mit *ENABLE = TRUE* wird die zuletzt empfangene Emergency ausgegeben.

Die Anzeige des Emergencies erfolgt bei Übereinstimmung des Eingangs *DEVICE* mit der Node-ID der produzierten Emergency.

Diese Fehler werden nur gesetzt, wenn sie vom Device mit der Node-ID des Eingang *DEVICE* produziert wurden.

Bei dem Empfang eines Emergencies wird *CONFIRM* auf *TRUE* gesetzt. *ERRORINFO* beinhaltet die Emergency-Informationen, die nach DS301 spezifiziert sind.

Wird ein *CANOPEN_KERNEL_ERROR*:

0010h CAN-Controller im Zustand „Bus off“

0011h CAN-Controller hat den “Error warning limit” überschritten

0021h Keine Antwort vom Knoten erhalten

0026h Interner Fehler.

0027h Node-Guarding-Fehler

0028h Device nicht im Operational-Mode

des dedizierten Devices identifiziert bleibt *CONFIRM* auf *FALSE* und es wird der Fehler am *ERROR*-Ausgang ausgegeben.

Es werden intern keine blockierenden Zugriffe durchgeführt. Die Verwendung des Timeout-Eingangs ist damit nicht erforderlich. Dieser Eingang ist, ebenso wie der *COM*-Eingang, aus Kompatibilitätsgründen implementiert.

Mit dem Rücksetzen des *ENABLE*-Eingangs wird der Baustein wieder in den Grundzustand versetzt.

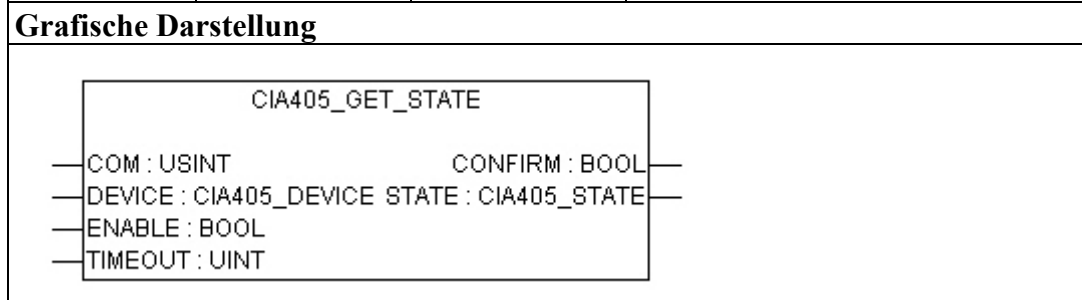
19.3.5 CIA405_GET_STATE

Der Funktionsbaustein CIA405_GET_STATE liefert den CANopen-Netzwerkzustand des ausgewählten CANopen Node zurück.

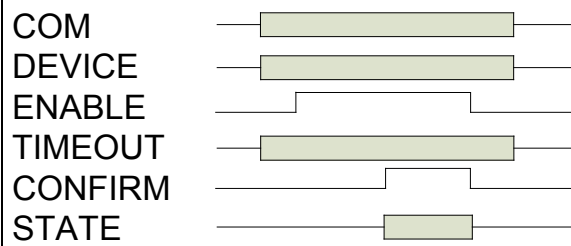
Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Maximale Ausführungszeit in ms. Disabled bei keiner oder Eingabe außerhalb des gültigen Bereiches [10..900].

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
STATE	CIA405_STATE	0..7	Siehe Abschnitt Datatypes „CIA405_STATE“.



Zeitliches Verhalten



Beschreibung

Der Status *UNKNOWN* wird zurückgegeben, wenn das ausgewählte Device nicht mit Node Guarding überwacht wird. Ist das Device nicht erreichbar, wird der Zustand *NOT_AVAIL* zurückgegeben.

Der Netzwerkzustand des Masters wird durch Eingabe der eigenen Device-ID abgefragt. Vom Master können die Zustände *STOPPED*, *OPERATIONAL* und *UNKNOWN* ausgegeben werden.

Um das Blockieren der SPS zu verhindern, kann dieser Baustein mit einer Timeout-Zeit beschrieben werden. Bei nicht vorhandenen Slaves wird diese Funktion damit nach der eingestellten Zeit verlassen. Initial ist keine Zeit vorgegeben und damit nicht aktiv.

19.3.6 CIA405_NMT

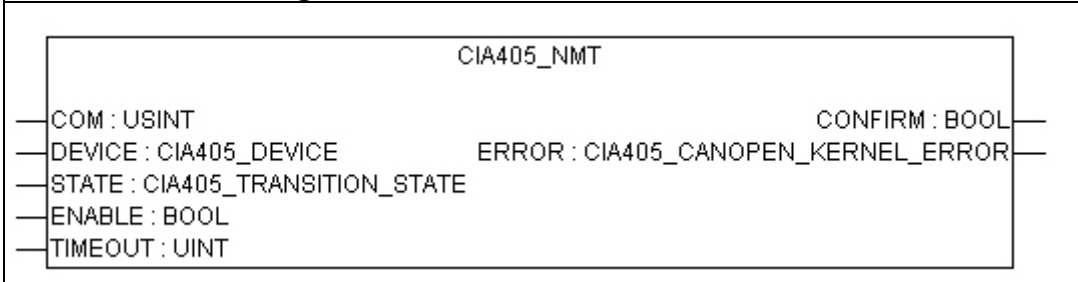
Mit dem Funktionsbaustein CIA405_NMT kann eine Netzwerk-Zustandsänderung eines vorher konfigurierten Moduls ausgelöst werden.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert.
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
STATE	CIA405_TRANSITION_STATE	0..4	Siehe Abschnitt Datatypes: CIA405_TRANSITION_STATE
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Aus Kompatibilitätsgründen implementiert. Eingang wird nicht erfasst.

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
ERROR	CIA405_CANOPEN_KERNEL_ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ERROR

Grafische Darstellung



Zeitliches Verhalten

--

Beschreibung

Wird am Eingang *DEVICE* eine ‚0‘ übergeben, werden alle am Bus kommunizierenden CANopen-Module in den neuen Zustand überführt.

Durch setzen von *ENABLE* auf *TRUE* wird der Zustandswechsel ausgeführt. Nach der Ausführung wird *CONFIRM* auf *TRUE* gesetzt. Am Ausgangsparameter *ERROR* wird der Status des Zustandswechsels angezeigt. Durch Setzen des *ENABLE*-Parameters auf *FALSE* (durch den Aufrufer) wird automatisch *CONFIRM* auf *FALSE* gesetzt und der Funktionsbaustein befindet sich wieder im Grundzustand.

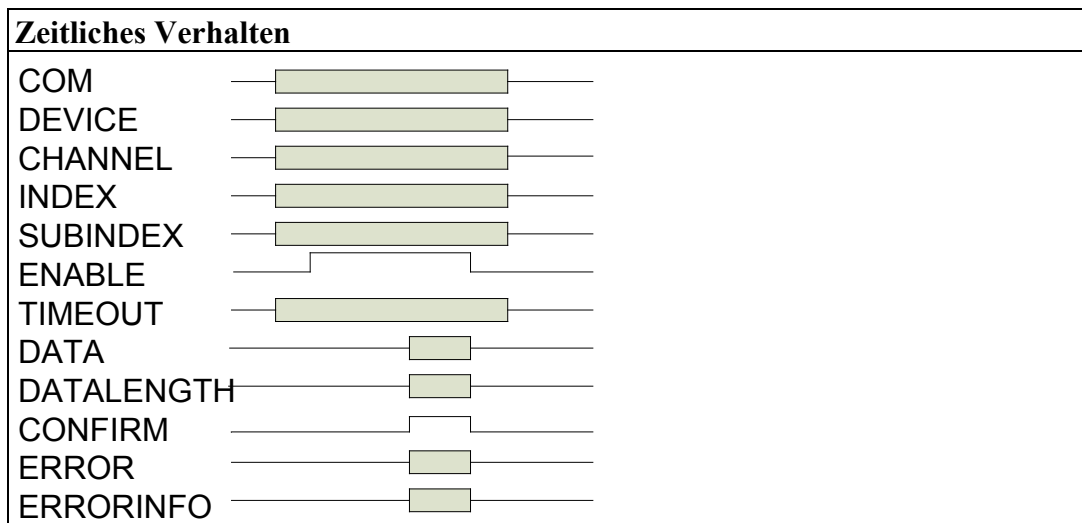
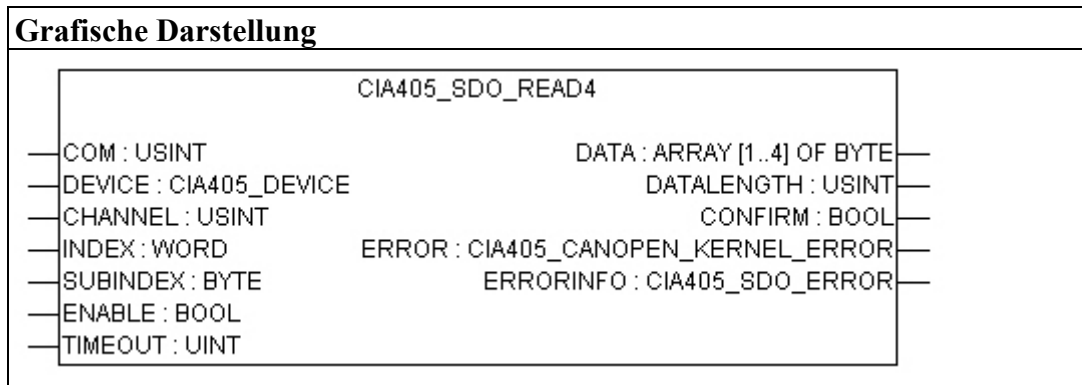
19.3.7 CIA405_SDO_READ4

Der Funktionsbaustein CIA405_SDO_READ4 gibt den Wert eines Objektverzeichniseintrages mit der maximalen Datenlänge von vier Byte zurück.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
CHANNEL	USINT	1..128d	SDO Kanalnummer. Dieser Eingang ist aus Kompatibilitätsgründen vorhanden und wird vom IPC ignoriert
INDEX	WORD	0..65535d	Index des Objektverzeichnisses
SUBINDEX	BYTE	0..255d	Subindex des Objektverzeichnisses
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Maximale Ausführungszeit in ms. Disabled bei keiner oder Eingabe außerhalb des gültigen Bereiches [10..900].

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
DATA	ARRAY [1..4] of BYTE	for each: 0..255d	Empfangene Daten
DATALENGTH	USINT	1..4	Anzahl der gültigen Daten.
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
ERROR	CIA405_ CANOPEN_ KERNEL_ ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ ERROR
ERRORINFO	CIA405_ SDO_ ERROR	0..FFFFFFFFh	Siehe Abschnitt Datatypes: CIA405_SDO_ERROR



Beschreibung

Der CANopen-Node wird durch den Parameter „DEVICE“ ausgewählt. Index und Subindex beschreiben den zu lesenden Objektverzeichniseintrag. Das Ergebnis des Funktionsbausteinanrufs kann erst einige Zyklen später zur Verfügung stehen.

Um das Blockieren der SPS zu verhindern, kann diese Funktion mit einer Timeout-Zeit beschrieben werden. Bei nicht vorhandenen Slaves kann diese Funktion damit vor der SDO-Abort Zeit verlassen werden. Initial ist keine Zeit vorgegeben und damit nicht aktiv.

Nachdem alle Eingangsparameter mit gültigen Werten besetzt sind, wird durch den Aufrufer *ENABLE* auf *TRUE* gesetzt. Damit wird das SDO-Frame gesendet. Wenn das SDO-Antworttelegramm fehlerfrei empfangen wurde, wird dies durch Setzen von *CONFIRM* auf *TRUE* angezeigt. Die Ausgangsparameter *DATA* und *DATALLENGTH* besitzen nun gültige Werte und *ERROR* ist 0. Durch Setzen des *ENABLE*-Parameters auf *FALSE* (durch den Aufrufer) wird automatisch *CONFIRM* auf *FALSE* gesetzt und der Funktionsbaustein befindet sich wieder im Grundzustand.

Tritt ein Fehler während der SDO-Übertragung auf (z.B. Timeout, Index nicht vorhanden, etc.) wird ebenfalls *CONFIRM* auf *TRUE* gesetzt, jedoch besitzt *ERROR* einen Wert ungleich 0. Der Parameter *ERRORINFO* enthält, falls vorhanden, weitergehende Informationen. Diese Abort-Informationen sind nach DS301 spezifiziert. *ENABLE* muss nach dem Auswerten des Fehlers auf *FALSE* gesetzt werden, um den Funktionsbaustein wieder in den Grundzustand zu bringen.

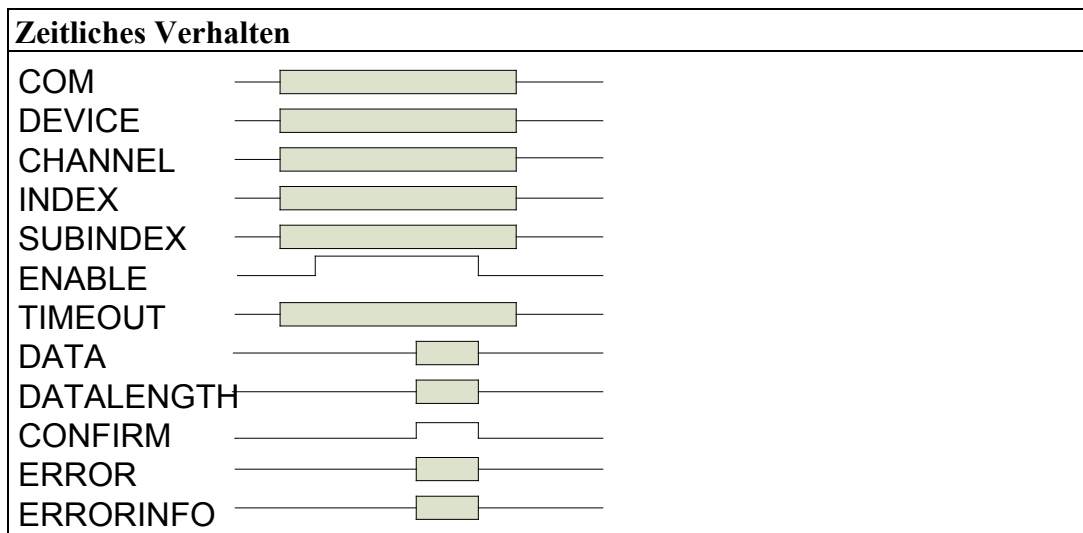
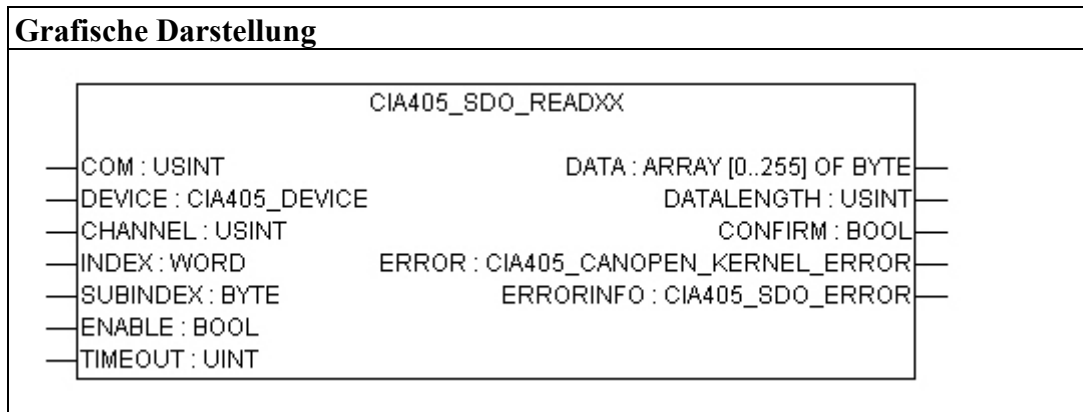
19.3.8 CIA405_SDO_READxx

Der Funktionsbaustein CIA405_SDO_READxx gibt den Wert eines Objektverzeichniseintrages mit der maximalen Datenlänge von 256 Byte zurück.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
CHANNEL	USINT	1..128d	SDO Kanalnummer. Dieser Eingang ist aus Kompatibilitätsgründen vorhanden und wird vom IPC ignoriert
INDEX	WORD	0..65535d	Index des Objektverzeichnisses
SUBINDEX	BYTE	0..255d	Subindex des Objektverzeichnisses
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
TIMEOUT	UINT	0..65535d	Maximale Ausführungszeit in ms. Disabled bei keiner oder Eingabe außerhalb des gültigen Bereiches [10..900].

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
DATA	ARRAY [1..248] of BYTE	for each: 0..255d	Empfangene Daten
DATALENGTH	USINT	1..248	Anzahl der gültigen Daten.
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
ERROR	CIA405_ CANOPEN_ KERNEL_ ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ ERROR
ERRORINFO	CIA405_ SDO_ ERROR	0..FFFFFFFFh	Siehe Abschnitt Datatypes: CIA405_SDO_ERROR



Beschreibung

Der CANopen-Node wird durch den Parameter „DEVICE“ ausgewählt. Index und Subindex beschreiben den zu lesenden Objektverzeichniseintrag.

Das Ergebnis des Funktionsbausteinanrufs kann erst einige Zyklen später zur Verfügung stehen.

Um das Blockieren der SPS zu verhindern, kann diese Funktion mit einer Timeout-Zeit beschrieben werden. Bei nicht vorhandenen Slaves kann diese Funktion damit vor der SDO-Abort Zeit verlassen werden. Initial ist keine Zeit vorgegeben und damit nicht aktiv.

Nachdem alle Eingangsparameter mit gültigen Werten besetzt sind, wird durch den Aufrufer *ENABLE* auf *TRUE* gesetzt. Damit wird das SDO-Frame gesendet. Wenn das SDO-Antworttelegramm fehlerfrei empfangen wurde, wird dies durch Setzen von *CONFIRM* auf *TRUE* angezeigt. Die Ausgangsparameter *DATA* und *DATALLENGTH* besitzen nun gültige Werte und *ERROR* ist 0. Durch Setzen des *ENABLE*-Parameters auf *FALSE* (durch den Aufrufer) wird automatisch *CONFIRM* auf *FALSE* gesetzt und der Funktionsbaustein befindet sich wieder im Grundzustand.

Tritt ein Fehler während der SDO-Übertragung auf (z.B. Timeout, Index nicht vorhanden, etc.), wird ebenfalls *CONFIRM* auf *TRUE* gesetzt, jedoch besitzt *ERROR* einen Wert ungleich 0. Der Parameter *ERRORINFO* enthält, falls vorhanden, weitergehende Informationen. Diese Abort-Informationen sind nach DS301 spezifiziert. *ENABLE* muss nach dem Auswerten des Fehlers auf *FALSE* gesetzt werden, um den Funktionsbaustein wieder in den Grundzustand zu bringen.

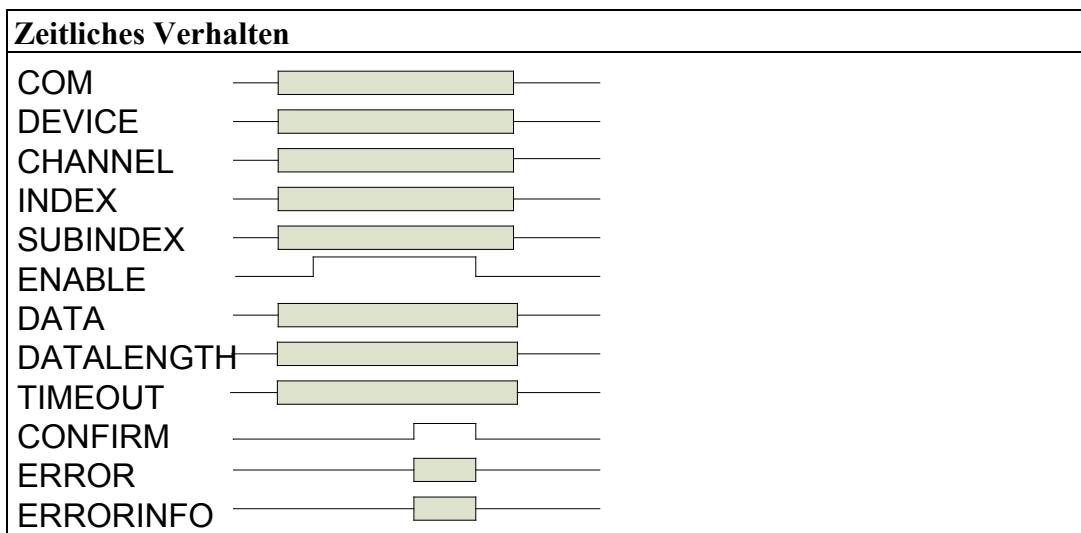
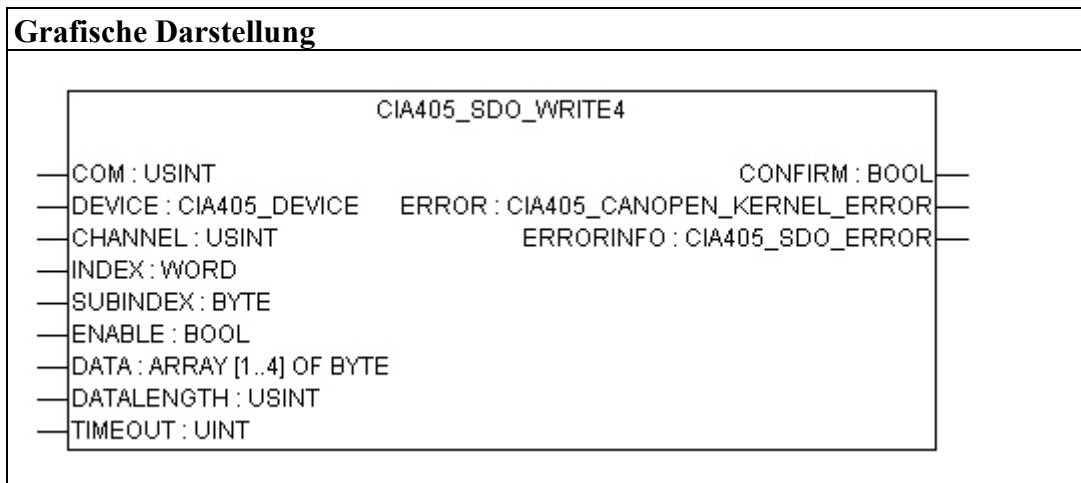
19.3.9 CIA405_SDO_WRITE4

Der Funktionsbaustein CIA405_SDO_WRITE4 schreibt die angegebenen Daten mit der maximalen Datenlänge von 4 Byte in das Objektverzeichnis eines Moduls.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
CHANNEL	USINT	1..128d	SDO Kanalnummer. Dieser Eingang ist aus Kompatibilitätsgründen vorhanden und wird vom IPC ignoriert
INDEX	WORD	0..65535d	Index des Objektverzeichnisses
SUBINDEX	BYTE	0..255d	Subindex des Objektverzeichnisses
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
DATA	ARRAY [1..4] of BYTE	for each: 0..255d	Zu schreibende Daten
DATALENGTH	USINT	1..4	Anzahl der gültigen Daten
TIMEOUT	UINT	0..65535d	Maximale Ausführungszeit in ms. Disabled bei keiner oder Eingabe außerhalb des gültigen Bereiches [10..900].

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
ERROR	CIA405_ CANOPEN_ KERNEL_ ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ ERROR
ERRORINFO	CIA405_ SDO_ ERROR	0..FFFFFFFFh	Siehe Abschnitt Datatypes: CIA405_SDO_ERROR



Beschreibung

Das Modul wird durch den Parameter „DEVICE“ ausgewählt. Index und Subindex spezifizieren den zu beschreibenden Objektverzeichniseintrag.

Das Ergebnis des Funktionsbausteinaufrufes kann erst einige Zyklen später zur Verfügung stehen.

Um das Blockieren der SPS zu verhindern kann diese Funktion mit einer Timeout-Zeit beschrieben werden. Bei nicht vorhandenen Slaves kann diese Funktion damit vor der SDO-Abort Zeit verlassen werden. Initial ist keine Zeit vorgegeben und damit nicht aktiv.

Nachdem alle Eingangsparameter mit gültigen Werten besetzt sind, wird durch den Aufrufer *ENABLE* auf *TRUE* gesetzt. Damit wird das SDO-Frames gesendet. Wenn die CAN-Software die zu schreibenden Daten fehlerfrei gesendet hat, meldet sie dies durch Setzen von *CONFIRM* auf *TRUE* und *ERROR* = 0. Durch setzen des *ENABLE*-Parameters auf *FALSE* (durch den Aufrufer) wird automatisch *CONFIRM* auf *FALSE* gesetzt und der Funktionsbaustein befindet sich wieder im Grundzustand.

Tritt ein Fehler während der SDO-Übertragung auf (z.B. Timeout, Index nicht vorhanden, etc.) wird ebenfalls *CONFIRM* auf *TRUE* gesetzt, jedoch besitzt *ERROR* einen Wert ungleich 0. Der Parameter *ERRORINFO* enthält, falls vorhanden, weitergehende Informationen. Diese Abort-Informationen sind nach DS301 spezifiziert. *ENABLE* muss nach dem Auswerten des Fehlers auf *FALSE* gesetzt werden.

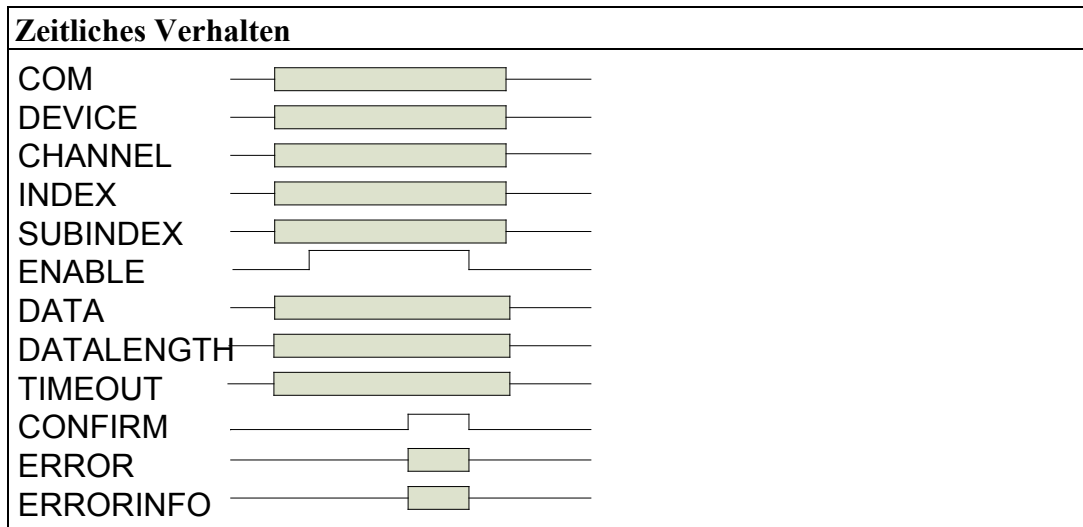
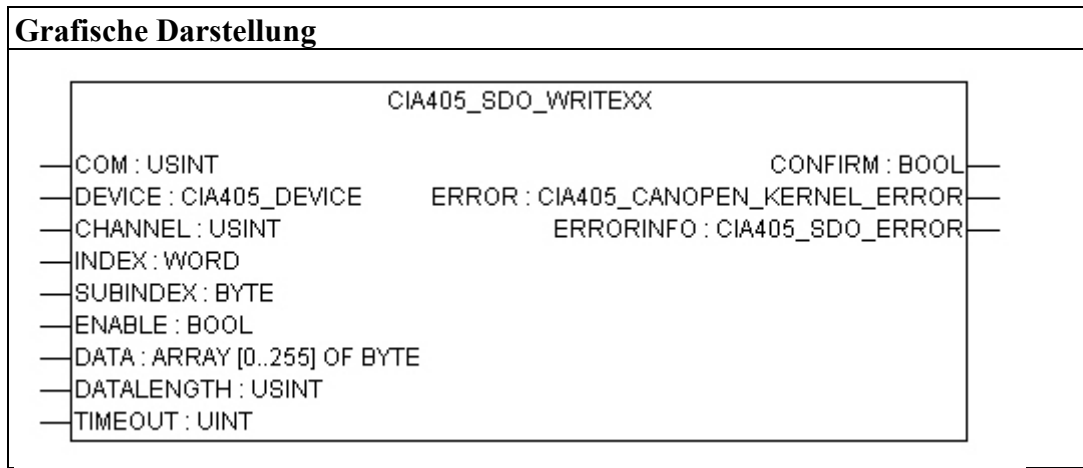
19.3.10 CIA405_SDO_WRITExx

Der Funktionsbaustein CIA405_SDO_WRITExx schreibt die angegebenen Daten mit der maximalen Datenlänge von 256 Byte in das Objektverzeichnis eines Moduls.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
CHANNEL	USINT	1..128d	SDO Kanalnummer. Dieser Eingang ist aus Kompatibilitätsgründen vorhanden und wird vom IPC ignoriert
INDEX	WORD	0..65535d	Index des Objektverzeichnisses
SUBINDEX	BYTE	0..255d	Subindex des Objektverzeichnisses
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion
DATA	ARRAY [0..248] of BYTE	for each: 0..255d	Zu schreibende Daten
DATALENG TH	USINT	0..248	Anzahl der gültigen Daten
TIMEOUT	UINT	0..65535d	Maximale Ausführungszeit in ms. Disabled bei keiner oder Eingabe außerhalb des gültigen Bereiches [10..900].

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	Ausführungsbestätigung
ERROR	CIA405_ CANOPEN_ KERNEL_ ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ ERROR
ERRORINFO	CIA405_ SDO_ ERROR	0..FFFFFFFFh	Siehe Abschnitt Datatypes: CIA405_SDO_ERROR



Beschreibung

Das Modul wird durch den Parameter „DEVICE“ ausgewählt. Index und Subindex spezifizieren den zu beschreibenden Objektverzeichniseintrag.

Das Ergebnis des Funktionsbausteinaufrufes kann erst einige Zyklen später zur Verfügung stehen.

Um das Blockieren der SPS zu verhindern kann diese Funktion mit einer Timeout-Zeit beschrieben werden. Bei nicht vorhandenen Slaves kann diese Funktion damit vor der SDO-Abort Zeit verlassen werden. Initial ist keine Zeit vorgegeben und damit nicht aktiv.

Nachdem alle Eingangsparameter mit gültigen Werten besetzt sind, wird durch den Aufrufer *ENABLE* auf *TRUE* gesetzt. Damit wird das SDO-Frames gesendet. Wenn die CAN-Software die zu schreibenden Daten fehlerfrei gesendet hat, meldet sie dies durch setzen von *CONFIRM* auf *TRUE* und *ERROR = 0*. Durch setzen des *ENABLE*-Parameters auf *FALSE* (durch den Aufrufer) wird automatisch *CONFIRM* auf *FALSE* gesetzt und der Funktionsbaustein befindet sich wieder im Grundzustand.

Tritt ein Fehler während der SDO-Übertragung auf (z.B. Timeout, Index nicht vorhanden, etc.), wird ebenfalls *CONFIRM* auf *TRUE* gesetzt, jedoch besitzt *ERROR* einen Wert ungleich 0. Der Parameter *ERRORINFO* enthält, falls vorhanden, weitergehende Informationen. Diese Abort-Informationen sind nach DS301 spezifiziert. *ENABLE* muss nach dem Auswerten des Fehlers auf *FALSE* gesetzt werden.

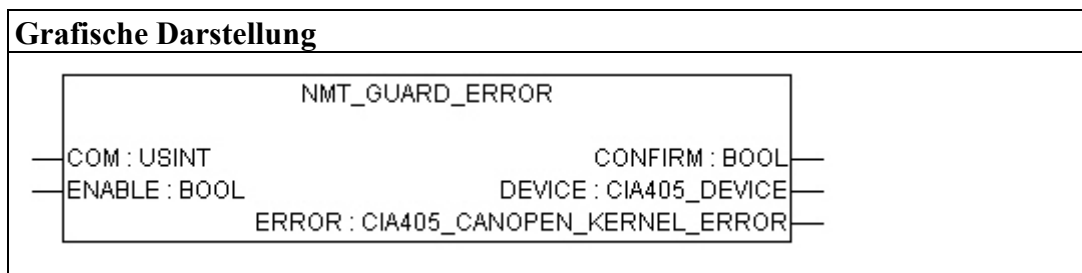
19.3.11 NMT_GUARD_ERROR

Der Funktionsbaustein NMT_GUARD_ERROR prüft ob ein Node-Guarding-Protokollverstoß von einem konfigurierten Knoten erkannt wurde.

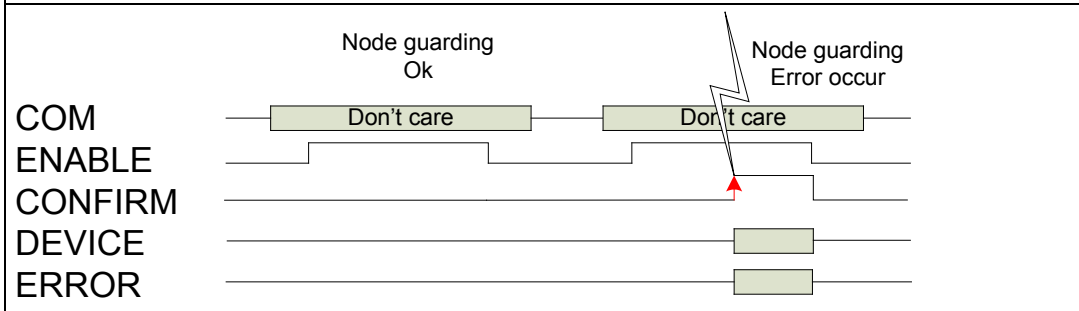
Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	TRUE: Guarding Error aufgetreten FALSE: Node Guarding OK
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
ERROR	CIA405_CANOPEN_KERNEL_ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ERROR



Zeitliches Verhalten



Beschreibung

Wird der *ENABLE*-Eingang auf *TRUE* gesetzt, ist die Überwachungsfunktion aktiviert. Mit jedem SPS-Zyklus wird diese Überprüfung durchgeführt.

Bei einem Protokollverstoß wird *CONFIRM* auf *TRUE* gesetzt. Der Ausgang *ERROR* gibt den Wert 0x0027h (Node-Guarding-Fehler) aus.

Sind mehrere Nodes betroffen, wird mit jedem Aufruf des Bausteins der Reihe nach jede Knotennummer ausgegeben, beginnend mit der kleinsten.

Mit dem Rücksetzen des *ENABLE*-Eingangs wird der Baustein wieder in den Grundzustand versetzt.

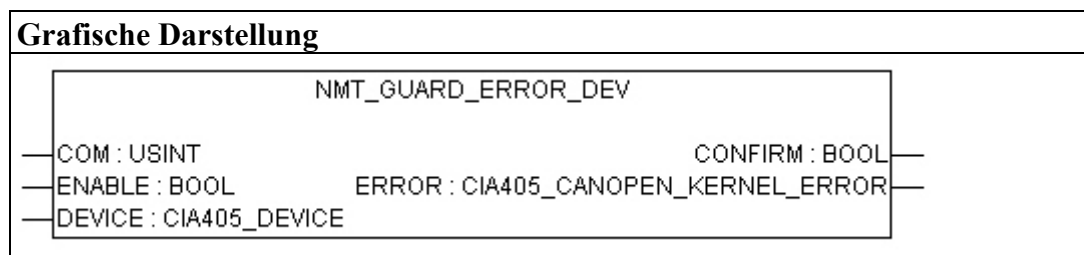
19.3.12 NMT_GUARD_ERROR_DEV

Der Funktionsbaustein NMT_GUARD_ERROR_DEV prüft ob ein Node-Guarding-Protokollverstoß von einem dedizierten und konfigurierten Knoten erkannt wurde.

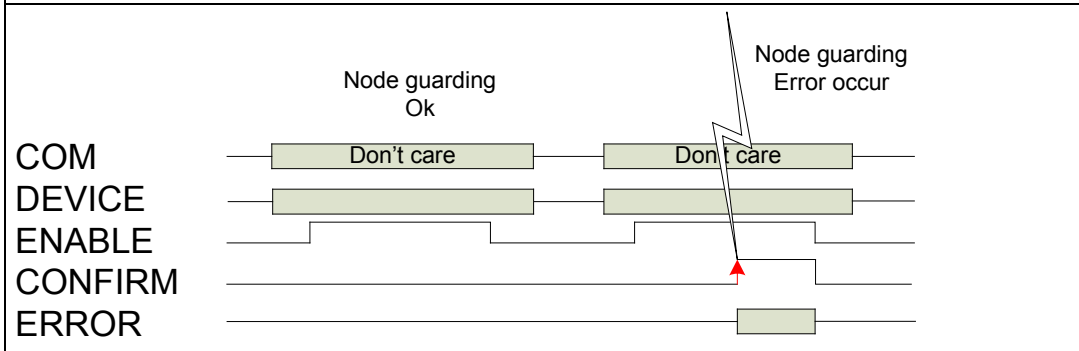
Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
COM	USINT	0..255d	Auswahl der CAN-Schnittstelle. Dieser Eingang wird bei Geräten mit einer Schnittstelle ignoriert
DEVICE	CIA405_DEVICE	0..127d	Siehe Abschnitt Datatypes: CIA405_DEVICE
ENABLE	BOOL	TRUE, FALSE	Aktivierung der Funktion

Ausgangsparameter			
Name	Datentyp	Wertebereich	Beschreibung
CONFIRM	BOOL	TRUE, FALSE	TRUE: Guarding Error aufgetreten FALSE: Node Guarding OK
ERROR	CIA405_CANOPEN_KERNEL_ERROR	0..65535d	Siehe Abschnitt Datatypes: CIA405_CANOPEN_KERNEL_ERROR



Zeitliches Verhalten



Beschreibung

Wird der *ENABLE*-Eingang auf *TRUE* gesetzt, ist die Überwachungsfunktion aktiviert. Mit jedem SPS-Zyklus wird diese Überprüfung durchgeführt.

Bei einem Protokollverstoß wird *CONFIRM* auf *TRUE* gesetzt. Der Ausgang *ERROR* gibt den Wert 0x0027h (Node-Guarding-Fehler) aus.

Soll ein Knoten überwacht werden, der nicht konfiguriert wurde, wird *CONFIRM* auf *TRUE* gesetzt und *ERROR* gibt den Wert 0024h (Knoten ist nicht konfiguriert) aus.

Bei einer ungültigen Node-ID wird der Fehler 0023h (Keine gültige Knotenadresse) ausgegeben.

Mit dem Rücksetzen des *ENABLE*-Eingangs wird der Baustein wieder in den Grundzustand versetzt.

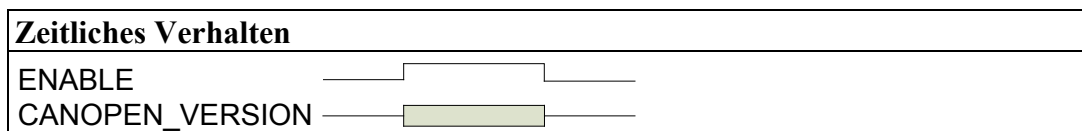
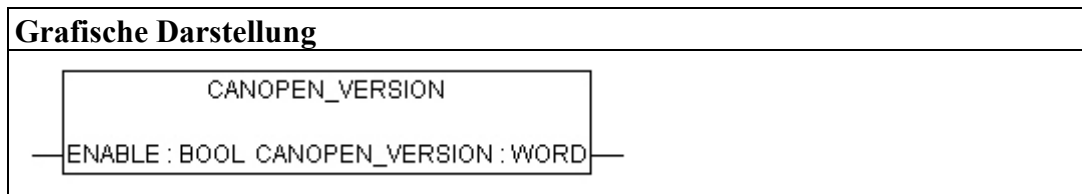
19.3.13 CANOPEN_VERSION

Gibt die Version der Library aus.

Kategorie	Funktionsbausteine für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANopen_02.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112

Eingangsparameter		
Name	Datentyp	Beschreibung
ENABLE	BOOL	Aktivierung der Funktion

Ausgangsparameter		
Name	Datentyp	Beschreibung
CIA405_VERSION	WORD	Version der Bibliothek



Beschreibung				
<p>Diese Funktion kann während der Programmentwicklung zur Information eingesetzt werden. Ferner können zur Laufzeit Versionskonflikte vermieden werden.</p>				
<table border="1"> <thead> <tr> <th>Version:</th> <th>Beschreibung:</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>IPC Version erstellt</td> </tr> </tbody> </table>	Version:	Beschreibung:	1	IPC Version erstellt
Version:	Beschreibung:			
1	IPC Version erstellt			

19.3.14 CIA405_DEVICE

Hier könnte ein einleitender Text zum Datentyp stehen ...

Kategorie	Datentyp für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Datentyp
Name der Bibliothek	WAGO_CANopen_02.lib
Anwendbar für	758-87x-112
Struktur	TYPE CIA405_DEVICE : BYTE; END_TYPE

Element	Wert	Beschreibung
CIA405_DEVICE	0..127	CANopen-Moduladresse

Beschreibung
Dieser Typ entspricht der CANopen-Moduladresse.

19.3.15 CIA405_SDO_ERROR

Kategorie	Datentyp für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Datentyp
Name der Bibliothek	WAGO_CANopen_02.lib
Anwendbar für	758-87x-112
Struktur	TYPE CIA405_SDO_ERROR : UDINT; END_TYPE

Element	Wert	Beschreibung
CIA405_SDO_ERROR	0..FFFFFFFFh	Fehlerinformation nach DS301

Beschreibung
Diese Strukturvariable beinhaltet die Fehlerinformation wie im Standard DS301 Kapitel „Protocol SDO abort transfer“ spezifiziert.

19.3.16 CIA405_EMCY_ERROR

Kategorie	Datentyp für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Datentyp
Name der Bibliothek	WAGO_CANopen_02.lib
Anwendbar für	758-87x-112
Struktur	TYPE CIA405_EMCY_ERROR : STRUCT EMCY_ERROR_CODE : WORD; ERROR_REGISTER : BYTE; ERROR_FIELD : ARRAY [1..5] OF BYTE; END_STRUCT END_TYPE

Element	Wert	Beschreibung
EMCY_ERROR_CODE	WORD	Fehlerinformation nach DS301
ERROR_REGISTER	BYTE	Fehlerinformation nach DS301
ERROR_FIELD	ARRAY [1..5]	Manufacturer specific array

Beschreibung
Diese Struktur beinhaltet die Emergency- Fehlerinformation wie im Standard DS301 spezifiziert. Die Manufacturer-Specific-Errorcodes müssen aus der Dokumentation der Gerätehersteller entnommen werden.

19.3.17 CIA405_STATE

Kategorie	Datentyp für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Datentyp
Name der Bibliothek	WAGO_CANopen_02.lib
Anwendbar für	758-87x-112
Struktur	<pre> TYPE CIA405_STATE : (INIT, RESET_COMM, RESET_APP, PRE_OPERATIONAL, STOPPED, OPERATIONAL, UNKNOWN, NOT_AVAIL); END_TYPE </pre>

Element	Wert	Beschreibung
INIT	0	Status nach Power on oder Hardware Reset
RESET_COMM	1	Status in dem die Parameter des Kommunikationsprofils initialisiert werden. Dieser Status wird nicht vom IPC angezeigt.
RESET_APP	2	Status in dem der Manufacturer-Specific-Bereich initialisiert wird. Dieser Status wird nicht vom IPC angezeigt.
PRE_OPERATIONAL	3	SDO-Kommunikationsstatus
STOPPED	4	Kommunikationsstop; Node Guarding und Heartbeat ist weiterhin aktiv.
OPERATIONAL	5	Alle Kommunikationsobjekte sind aktiv.
UNKNOWN	6	Status kann nicht ermittelt werden, da Node Guarding oder Heartbeat nicht aktiv ist.
NOT_AVAIL	7	Device ist nicht erreichbar (Timeout).

Beschreibung
Diese ENUM-Elemente geben den aktuellen CANopen-Netzwerkstatus, wie im Draft Standard 301 spezifiziert, wieder.

19.3.18 CIA405_TRANSITION_STATE

Kategorie	Datentyp für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Datentyp
Name der Bibliothek	WAGO_CANopen_02.lib
Anwendbar für	758-87x-112
Struktur	<pre> TYPE CIA405_TRANSITION_STATE : (START_REMOTE_NODE, STOP_REMOTE_NODE, ENTER_PRE_OPERATIONAL, RESET_NODE, RESET_COMMUNICATION); END_TYPE </pre>

Element	Wert	Beschreibung
START_REMOTE_NODE	0	Modul starten
STOP_REMOTE_NODE	1	Modul in Stopped überführen
ENTER_PRE_OPERATION AL	2	Modul in Pre-Operational überführen
RESET_NODE	3	Reset des Moduls
RESET_COMMUNICATIO N	4	Reset der Kommunikation

Beschreibung
Diese Enumeration gibt eine Übersicht der Transitionszustände in die ein Modul überführt werden kann wie im Draft Standard 301 spezifiziert wieder.

19.3.19 CANOPEN_KERNEL_ERROR

Kategorie	Datentyp für CANopen nach CiA 405
Name	CIA405_GET_LOCAL_NODE_ID
Typ	Datentyp
Name der Bibliothek	WAGO_CANopen_02.lib
Anwendbar für	758-87x-112
Struktur	<pre> TYPE CIA405_CANOPEN_KERNEL_ERROR : WORD; END_TYPE </pre>

Element	Wert	Beschreibung
CIA405_CANOPEN_KERNEL_ERROR	0000h	Kein Fehler vom CANopen-Kernel erkannt.
	0001h	Anderer Fehler. Weitere Fehlerinformationen können, soweit vorhanden, dem Ausgang CIA405_SDO_ERROR entsprechend des Standards DS301 entnommen werden.
	0002h	Ungültige Datenlänge.
	0003h	Zeitüberschreitung bei der Bausteinausführung.
	0010h	CAN-Controller im Zustand „Bus off“.
	0011h	CAN-Controller hat den “Error warning limit” überschritten.
	0021h	Keine Antwort vom Knoten erhalten.
	0022h	Der SDO Kanal wird z:Zt. vom IPC für die Boot Phase des Knotens verwendet und ist für diese Funktion gesperrt.
	0023h	Keine gültige Knotenadresse.
	0024h	Knoten ist nicht konfiguriert.
	0025h	NMT Kommando ungültig oder konnte nicht gesendet werden.
	0026h	Interner Fehler.
	0027h	Node-Guarding-Fehler.
	0028h	Device hat den Operational-Mode verlassen. Ausgabe nur bei aktiviertem Node Guarding möglich.

Beschreibung
Diese Tabelle gibt eine Übersicht der Werte, die <i>CANOPEN_KERNEL_ERROR</i> annehmen kann, wie im Draft Standard 405 spezifiziert, wieder.

19.4 WAGO_CANLayer2_01.lib

Diese Bibliothek stellt Funktionsblöcke zur Verfügung, die es ermöglichen, eine Kommunikation mit CAN Layer 2 über 11Bit- und 29Bit-Identifizier aufzubauen.

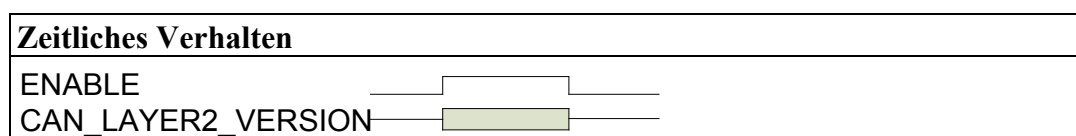
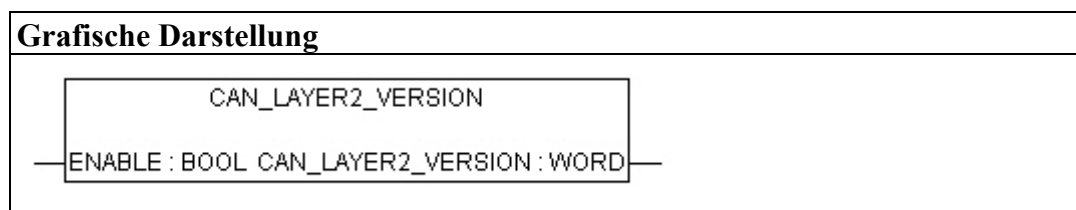
19.4.1 CAN_LAYER2_VERSION

Die Funktion CAN_LAYER2_VERSION gibt die aktuelle Versionsnummer der Bibliothek zurück.

Kategorie	CANLayer 2 Kommunikation
Name	CAN_LAYER2_VERSION
Typ	Funktion
Name der Bibliothek	WAGO_CANLayer2_01.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112, 762-3xx/000-001 bzw. 762-3xx/000-003

Eingangsparameter		
Name	Datentyp	Beschreibung
ENABLE	BOOL	Aktivierung der Funktion

Ausgangsparameter		
Name	Datentyp	Beschreibung
CAN_LAYER2_VERSION	WORD	Version der Bibliothek



Beschreibung	
Diese Funktion kann während der Programmentwicklung zur Information eingesetzt werden. Ferner können zur Laufzeit Versionskonflikte vermieden werden.	
Version:	History:
1	I/O-IPC Version erstellt
2	Library überarbeitet

19.4.2 CAN_ERROR_INFO

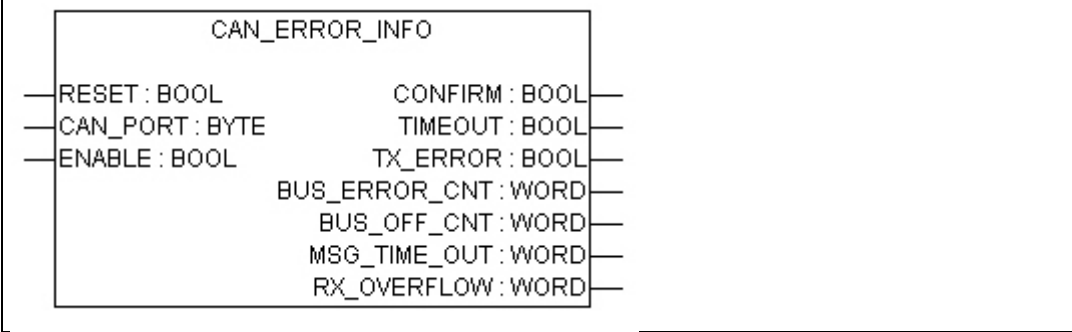
Der Funktionsbaustein CAN_ERROR_INFO gibt den physikalischen Zustand des CAN-Bus aus.

Kategorie:	CAN Layer 2 Funktionsbausteine
Name:	CAN_ERROR_INFO
Typ:	Funktion
Name der Bibliothek:	WAGO_CANLayer2_01.lib
Erforderliche Bibliotheken	
Anwendbar für:	758-87x-112, 762-3xxx/000-001 bzw. 762-3150/000-003

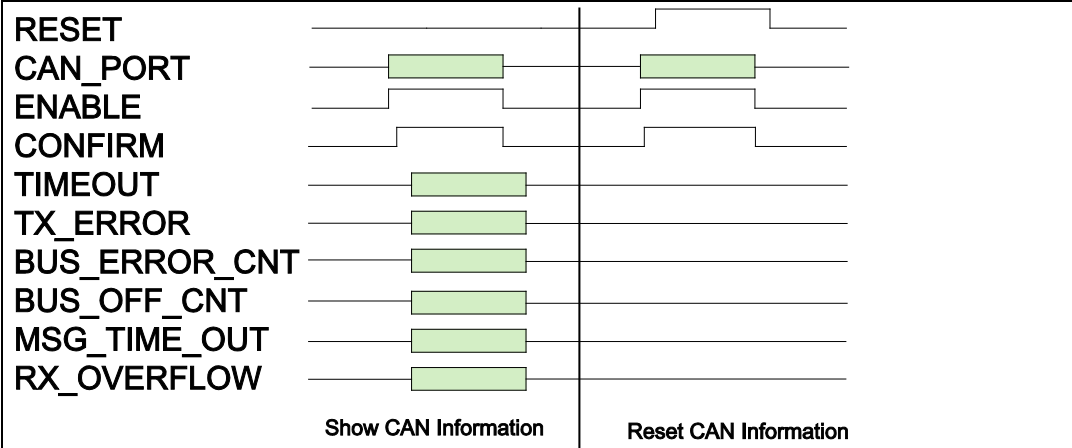
Eingangsparameter:		
Name	Datentyp	Beschreibung
RESET	BOOL	Rücksetzen der Errorcounter
CAN_PORT	BYTE	Auswahl des CAN-Interfaces 0: CAN 0 Interface aktiviert 1: CAN 1 Interface aktiviert Bei Geräten mit einer Schnittstelle ohne Funktion.
ENABLE	BOOL	Aktivierung der Funktion

Ausgangsparameter:		
Name	Datentyp	Beschreibung
CONFIRM	BOOL	Ausführungsbestätigung
TIMEOUT	BOOL	Eine CANopen-Nachricht wurde von keinem CAN-Knoten bestätigt (Acknowledged). Indiz, das kein weiterer Knoten am Bus kommuniziert.
TX_ERROR	BOOL	Der I/O-IPC hat mindestens einen Sendefehler erkannt.
BUS_ERROR_CNT	WORD	Anzahl von Warning-Level-Überschreitungen.
BUS_OFF_CNT	WORD	Anzahl der automatisch initiierten CAN-Controller-Reinitialisierungen.
MSG_TIME_OUT	WORD	Anzahl von verworfenen CAN-Nachrichten, aufgrund von fehlendem Acknowledge.
RX_OVERFLOW	WORD	Anzahl von CAN-Nachrichten, die nicht verarbeitet wurden, aufgrund eines Überlaufes des Empfangsbuffers im CAN-Controller.

Grafische Darstellung:



Zeitliches Verhalten:



Funktionsbeschreibung:

Nach dem Setzen des *ENABLE*-Einganges ist die Überwachung aktiviert. Dies wird durch den *CONFIRM*-Ausgang angezeigt.

Der *TIMEOUT*-Ausgang wird gesetzt, wenn innerhalb von 20 ms nach der Übertragung einer CAN-Nachricht kein Acknowledge von einem anderen Busteilnehmer empfangen wird. Die Anzahl von fehlenden Acknowledges kann der *MSG_TIME_OUT*-Variable entnommen werden. Diese Überwachung ist nur für CANopen-Nachrichten aktiviert.

Der *TX_ERROR*-Ausgang wird bei der Erkennung eines Übertragungsfehlers gesetzt. Die Anzahl der Events, die zum Setzen der Variable geführt hat, können an den Ausgängen *BUS_ERROR_CNT* und *BUS_OFF_CNT* ausgelesen werden. Nach der Auswertung einer oder aller Errorvariablen können diese über den *RESET*-Eingang zurückgesetzt werden. Hier werden nur die Errorvariablen zurückgesetzt, ein Reset des CAN-Controllers wird nicht durchgeführt. Ist eine Beeinflussung des CAN-Busses weiterhin vorhanden, erfolgt die erneute Anzeige und Inkrementierung der Fehleranzeigen.

Durch einen Flankenwechsel *ENABLE = FALSE* wird der Baustein in den Grundzustand zurückgesetzt.

19.4.3 CAN_RX_11BIT_FRAME

Mit dem Funktionsbaustein CAN_RX_11BIT_FRAME werden CAN-Nachrichten im 11Bit-Identifizier-Format empfangen.

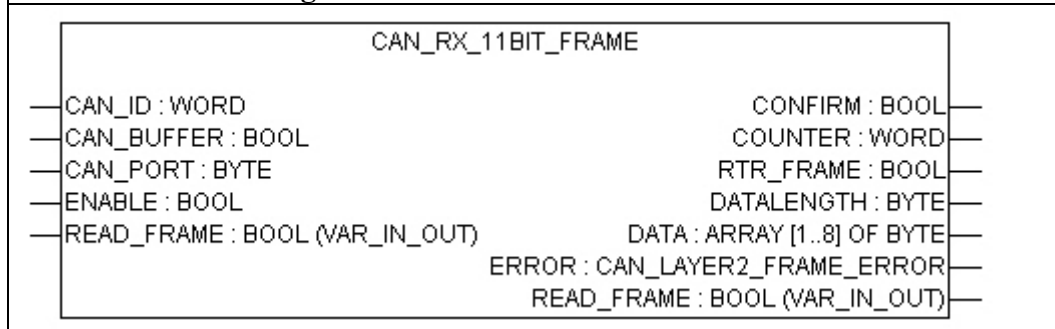
Kategorie	CANLayer 2 Kommunikation
Name	CAN_RX_11BIT_FRAME
Typ	Funktion
Name der Bibliothek	WAGO_CANLayer2_01.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112, 762-3xxx/000-001 bzw. 762-3150/000-003

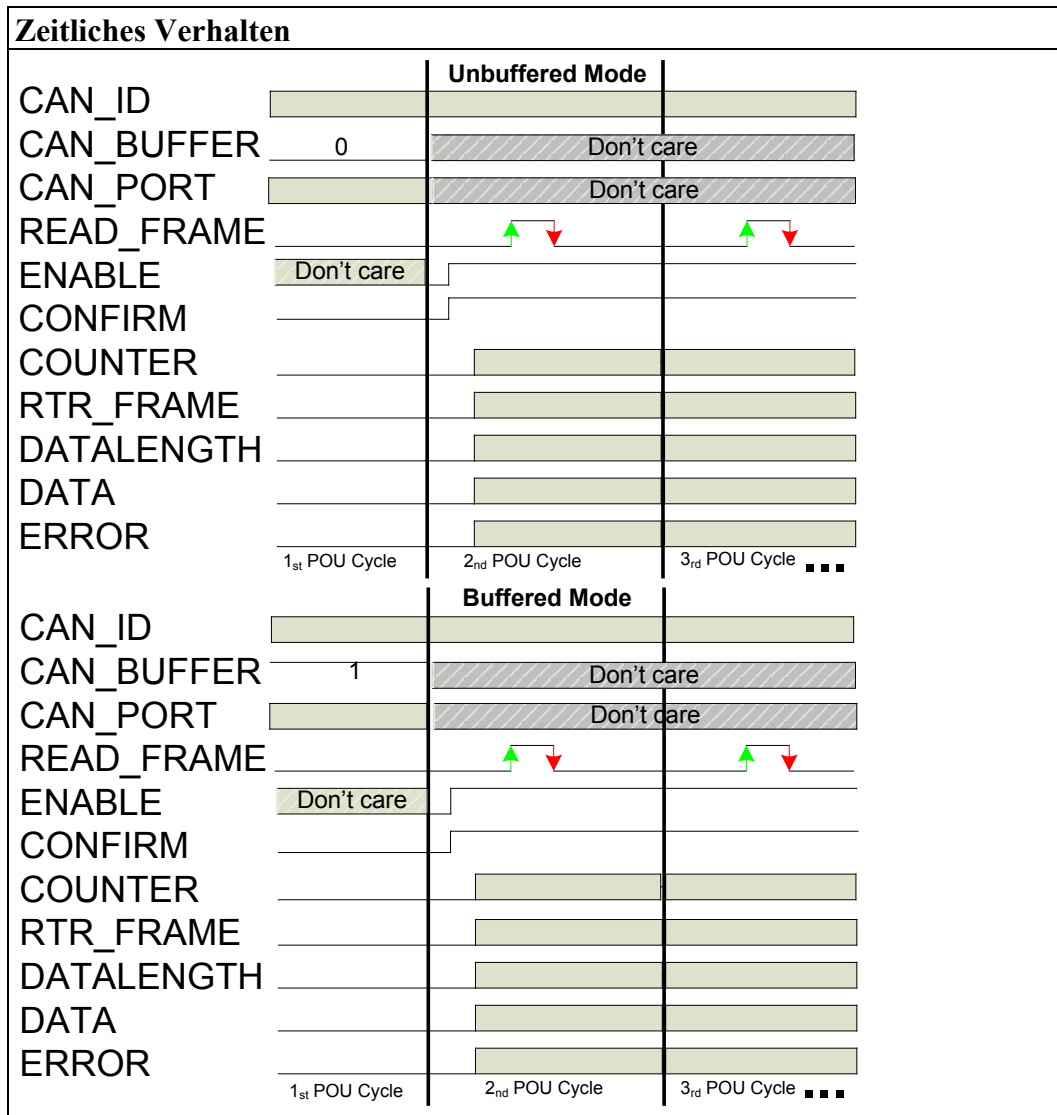
Eingangsparameter		
Name	Datentyp	Beschreibung
CAN_ID	WORD	Registrierung des 11Bit-Identifiers, dessen Nachricht empfangen werden soll.
CAN_BUFFER	BOOL	Schaltet den internen Buffer ein oder aus. FALSE: Es wird immer die zuletzt empfangene CAN-Nachricht ausgegeben. TRUE: Über einen internen Buffer können 8 Nachrichten zwischengespeichert werden und diese zu einem späteren Zeitpunkt folgerichtig (FIFO) ausgelesen werden.
CAN_PORT	BYTE	Auswahl des CAN-Interfaces 0: CAN 0 Interface aktiviert 1: CAN 1 Interface aktiviert Bei Geräten mit einer Schnittstelle ohne Funktion.
ENABLE	BOOL	Aktivierung der Funktion

Ausgangsparameter		
Name	Datentyp	Beschreibung
CONFIRM	BOOL	Ausführungsbestätigung
COUNTER	WORD	Diese Variable wird bei jedem Empfang einer CAN-Nachricht mit registriertem CAN-Identifizier inkrementiert.
RTR_FRAME	BOOL	Empfang eines Remote-Frames.
DATALENGTH	BYTE	Datenlänge der empfangenen Nachricht.
DATA	ARRAY[1..8] OF BYTE	Beinhaltet die übertragenen Daten der CAN Nachricht. Die Anzahl der Bytes entspricht dem Ausgang <i>DATALENGTH</i> .
ERROR	CAN_LAYER2_FRAME_ERROR	Siehe Abschnitt Data Types: CAN_LAYER2_FRAME_ERROR

Ein/Ausgangsparameter		
Name	Datentyp	Beschreibung
READ_FRAME	BOOL	<p>Übernahme der CAN-Daten mit steigender Flanke aus dem Funktionsbaustein in die Applikation.</p> <p>(ENABLE = TRUE) & (CAN_BUFFER = FALSE): Ausgabe der zuletzt empfangenen Nachricht. Automatische Rücknahme nach Kopiervorgang durch den Baustein</p> <p>(ENABLE = TRUE) & (CAN_BUFFER = TRUE): Ausgabe der ältesten Nachricht im FIFO-Puffer. Automatische Rücknahme nach Kopiervorgang durch den Baustein.</p>

Grafische Darstellung





Beschreibung

Mit *ENABLE = TRUE* wird die Registrierung des CAN-Identifiers und die Einstellung des Buffermodus vorgenommen. Diese Einstellungen können durch Rücksetzen von *ENABLE = FALSE* und anschließendem Setzen (*ENABLE = TRUE*) mit neuen Einstellungen überschrieben werden.

Der CAN-Identifier darf für das Zeitintervall *ENABLE = TRUE* nicht geändert werden. Eine Änderung von *CAN_PORT* oder *CAN_BUFFER* hat in diesem Zeitintervall keine weitere Auswirkung auf den Programmablauf.

Es können im Programm maximal 127 Standard-Identifier und 127 Extended-Identifier vergeben werden.

Steuerungskonfiguration:

In der Steuerungskonfiguration muss auch bei Nutzung eines reinen CAN Layer 2 Systems der CANopen Master konfiguriert werden. Hier wird die Einstellung der Baudrate vorgenommen. Weitere Slaves sind in diesem Betriebsmodus nicht zu konfigurieren.

Buffered Modus (*CAN_BUFFER = 1*):

In diesem Modus wird eine folgerichtige Abarbeitung von bis zu 8 CAN-Nachrichten ermöglicht.

Ist der Baustein über *ENABLE* aktiviert, wird der Empfang einer neuen Nachricht durch einen inkrementierten Counter-Wert angezeigt.

Durch Flankenwechsel *READ_FRAME = TRUE* wird dem Baustein angezeigt, dass die Applikation bereit zur Verarbeitung einer Nachricht ist. Die Speicherung einer neuen Nachricht vom Baustein in den Applikationsspeicher wird durch Flankenwechsel *READ_FRAME = FALSE* mitgeteilt. Dieser Statuswechsel wird durch den Baustein initiiert.

Bei einem Buffer overrun (*COUNTER > 8*) kann durch Auswerten der Variable *COUNTER* ermittelt werden, wie viele Nachrichten nicht mehr abgespeichert werden konnten. Dies ist ein Indikator für die CAN-Prozesslast zur SPS-Zykluszeit.

Die gespeicherten Telegramme im Buffer werden ungültig und können nicht mehr ausgelesen werden. Weitere Telegramme vom CAN-Bus werden nicht mehr in den Buffer gespeichert.

Durch einen Flankenwechsel *ENABLE = FALSE* wird der Baustein in den Grundzustand zurückgesetzt.

Unbuffered Modus (*CAN_BUFFER = 0*):

In diesem Modus wird nur die zeitrichtige letzte empfangene CAN-Nachricht ausgegeben.

Ist der Baustein über *ENABLE* aktiviert, wird der Empfang einer neuen Nachricht durch einen inkrementierten Counter-Wert angezeigt. Der Counter-Wert läuft bei 65535 über.

Durch Flankenwechsel *READ_FRAME = TRUE* wird dem Baustein angezeigt, dass die Applikation bereit zur Verarbeitung einer Nachricht ist. Die Speicherung einer neuen Nachricht vom Baustein in den Applikationsspeicher wird durch Flankenwechsel *READ_FRAME = FALSE* mitgeteilt. Dieser Statuswechsel wird durch den Baustein initiiert.

Durch einen Flankenwechsel *ENABLE = FALSE* wird der Baustein in den Grundzustand zurückgesetzt.

19.4.4 CAN_RX_29BIT_FRAME

Mit dem Funktionsbaustein CAN_RX_29BIT_FRAME werden CAN-Nachrichten im 29Bit-Identifizier-Format empfangen.

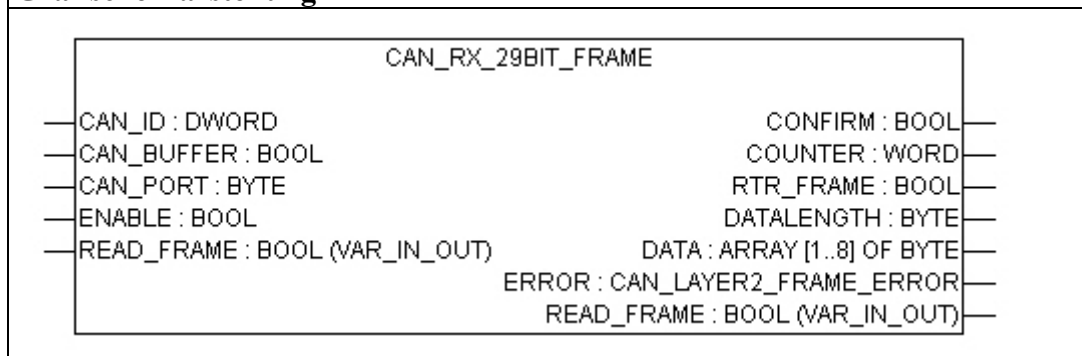
Kategorie	CANLayer 2 Kommunikation
Name	CAN_RX_29BIT_FRAME
Typ	Funktion
Name der Bibliothek	WAGO_CANLayer2_01.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112, 762-3xxx/000-001 bzw. 762-3150/000-003

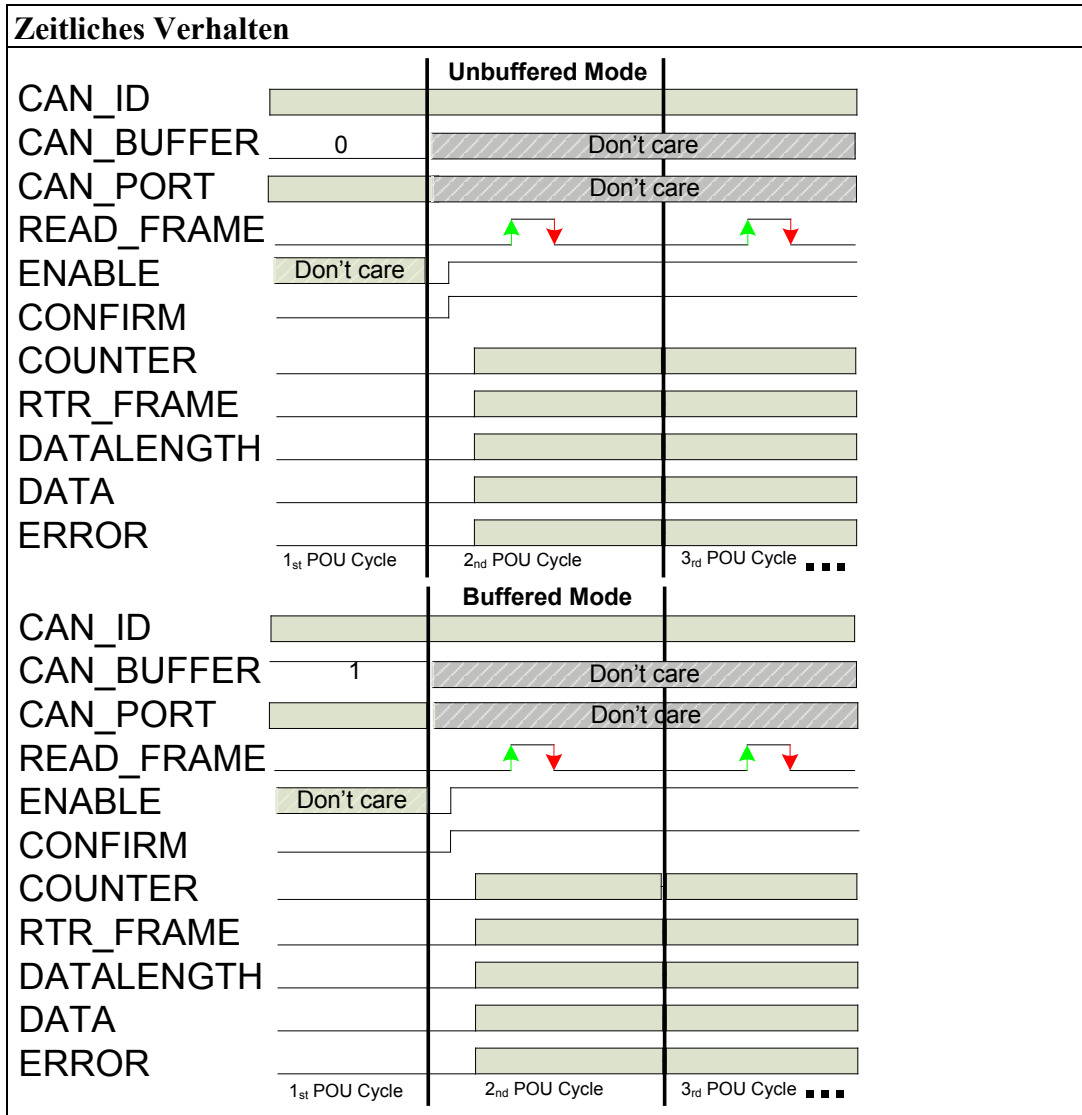
Eingangsparameter		
Name	Datentyp	Beschreibung
CAN_ID	DWORD	Registrierung des 29Bit-Identifiers, dessen Nachricht empfangen werden soll.
CAN_BUFFER	BOOL	Schaltet den internen Buffer ein oder aus. FALSE: Es wird immer die zuletzt empfangene CAN-Nachricht ausgegeben. TRUE: Über einen internen Buffer können 8 Nachrichten zwischengespeichert werden und diese zu einem späteren Zeitpunkt folgerichtig (FIFO) ausgelesen werden.
CAN_PORT	BYTE	Auswahl des CAN-Interfaces 0: CAN 0 Interface aktiviert 1: CAN 1 Interface aktiviert Bei Geräten mit einer Schnittstelle ohne Funktion.
ENABLE	BOOL	Aktivierung der Funktion

Ausgangsparameter		
Name	Datentyp	Beschreibung
CONFIRM	BOOL	Ausführungsbestätigung
COUNTER	WORD	Diese Variable wird bei jedem Empfang einer CAN-Nachricht mit registriertem CAN-Identifizier inkrementiert.
RTR_FRAME	BOOL	Diese Variable wird bei jedem Empfang einer CAN-Nachricht mit registriertem CAN-Identifizier inkrementiert.
DATALENGTH	BYTE	Empfang eines Remote-Frames.
DATA	ARRAY [1..8] OF BYTE	Datenlänge der empfangenen Nachricht.

ERROR	CAN_LAYER2_FRAME_ERROR	Beinhaltet die übertragenen Daten der CAN-Nachricht. Die Anzahl der Bytes entspricht dem Ausgang <i>DATALENGTH</i> .
Ein/Ausgangsparameter		
Name	Datentyp	Beschreibung
READ_FRAME	BOOL	Übernahme der CAN-Daten mit steigender Flanke aus dem Funktionsbaustein in die Applikation. (<i>ENABLE</i> = TRUE) & (<i>CAN_BUFFER</i> = FALSE): Ausgabe der zuletzt empfangenen Nachricht. Automatische Rücknahme nach Kopiervorgang durch den Baustein (<i>ENABLE</i> = TRUE) & (<i>CAN_BUFFER</i> = TRUE): Ausgabe der ältesten Nachricht im FIFO-Puffer. Automatische Rücknahme nach Kopiervorgang durch den Baustein.

Grafische Darstellung





Beschreibung

Die Registrierung des CAN-Identifiers und die Einstellung des Buffermodus erfolgt im ersten SPS-Zyklus nachdem der Baustein mit *ENABLE = TRUE* aktiviert wurde. Der CAN-Identifier darf im weiteren Programmablauf nicht mehr geändert werden. Eine Änderung von *CAN_PORT* oder *CAN_BUFFER* hat keine weitere Auswirkung auf den Programmablauf.

Es können im Programm maximal 127 Standard-Identifier und 127 Extended-Identifier vergeben werden.

Steuerungskonfiguration:

In der Steuerungskonfiguration muss auch bei Nutzung eines reinen CAN-Layer-2-Systems der CANopen-Master konfiguriert werden. Hier wird die Einstellung der Baudrate vorgenommen. Weitere Slaves sind in diesem Betriebsmodus nicht zu konfigurieren.

Buffered Modus (*CAN_BUFFER = 1*):

In diesem Modus wird eine folgerichtige Abarbeitung von bis zu 8 CAN-Nachrichten ermöglicht.

Ist der Baustein über *ENABLE* aktiviert, wird der Empfang einer neuen Nachricht durch einen inkrementierten Counter-Wert angezeigt.

Durch Flankenwechsel *READ_FRAME = TRUE*, wird dem Baustein angezeigt, dass die Applikation bereit zur Verarbeitung einer Nachricht ist. Die Speicherung einer neuen Nachricht vom Baustein in den Applikationsspeicher wird durch Flankenwechsel *READ_FRAME = FALSE* mitgeteilt. Dieser Statuswechsel wird durch den Baustein initiiert.

Bei einem Buffer overrun (*COUNTER > 8*) kann durch Auswerten der Variable *COUNTER* ermittelt werden, wie viele Nachrichten nicht mehr abgespeichert werden konnten. Dies ist ein Indikator für die CAN-Prozesslast zur SPS-Zykluszeit.

Die gespeicherten Telegramme im Buffer werden ungültig und können nicht mehr ausgelesen werden. Weitere Telegramme vom CAN-Bus werden nicht mehr in den Buffer gespeichert.

Durch einen Flankenwechsel *ENABLE = FALSE* wird der Baustein in den Grundzustand zurückgesetzt.

Unbuffered Modus (*CAN_BUFFER = 0*):

In diesem Modus wird nur die zeitrichtige letzte empfangene CAN-Nachricht ausgegeben.

Ist der Baustein über *ENABLE* aktiviert, wird der Empfang einer neuen Nachricht durch einen inkrementierten Counter-Wert angezeigt. Der Counter-Wert läuft bei 65535 über.

Durch Flankenwechsel *READ_FRAME = TRUE* wird dem Baustein angezeigt, dass die Applikation bereit zur Verarbeitung einer Nachricht ist. Die Speicherung einer neuen Nachricht vom Baustein in den Applikationsspeicher wird durch Flankenwechsel *READ_FRAME = FALSE* mitgeteilt. Dieser Statuswechsel wird durch den Baustein initiiert.

Durch einen Flankenwechsel *ENABLE = FALSE* wird der Baustein in den Grundzustand zurückgesetzt.

19.4.5 CAN_TX_11BIT_FRAME

Mit dem Funktionsbaustein CAN_TX_11BIT_FRAME werden CAN-Nachrichten im 11Bit-Identifizier-Format gesendet.

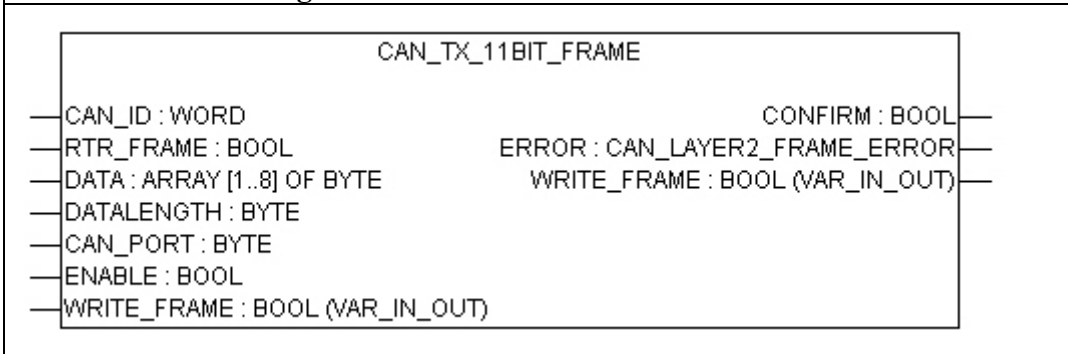
Kategorie	CANLayer 2 Kommunikation
Name	CAN_TX_11BIT_FRAME
Typ	Funktionsblock
Name der Bibliothek	WAGO_CANLayer2_01.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112, 762-3xxx/000-001 bzw. 762-3150/000-003

Eingangsparameter		
Name	Datentyp	Beschreibung
CAN_ID	WORD	11Bit-Identifizier der zu sendenden CAN-Nachricht.
RTR_FRAME	BOOL	Die zu sendende CAN-Nachricht ist ein Remote-Frame.
DATA	ARRAY[1..8] OF BYTE	Beinhaltet die Daten der CAN-Nachricht. Die Anzahl der Bytes entspricht dem Eingang <i>DATALENGTH</i> .
DATALENGTH	BYTE	Datenlänge der zu sendenden CAN-Nachricht.
CAN_PORT	BYTE	Auswahl des CAN-Interfaces 0: CAN 0 Interface aktiviert 1: CAN 1 Interface aktiviert Bei Geräten mit einer Schnittstelle ohne Funktion.
ENABLE	BOOL	Aktivierung der Funktion

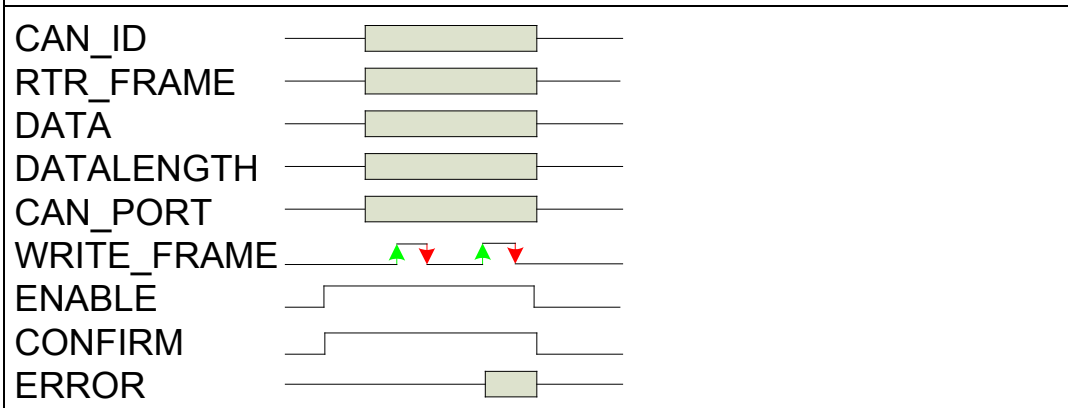
Ausgangsparameter		
Name	Datentyp	Beschreibung
CONFIRM	BOOL	Es wurde eine CAN-Nachricht gesendet.
ERROR	CAN_LAYER2 _FRAME_ ERROR	Siehe Abschnitt Data Types: CAN_LAYER2_FRAME_ERROR

Ein/Ausgangsparameter		
Name	Datentyp	Beschreibung
WRITE_FRAME	BOOL	Übertragung der CAN-Nachricht wird initiiert.

Grafische Darstellung



Zeitliches Verhalten



Beschreibung

Nach dem Setzen des *ENABLE*-Eingangs ist der Funktionsbaustein aktiviert. Danach wird eine CAN-Nachricht durch Flankenwechsel FALSE nach TRUE des *WRITE_FRAME*-Eingangs gesendet. Diese wird aus den Werten an den Eingängen: *CAN_ID*, *RTR_FRAME*, *DATA* und *DATALENGTH* gebildet. Nach der Übertragung der Nachricht wird vom Baustein automatisch die IN/OUT Variable *WRITE_FRAME* auf FALSE zurückgesetzt.

Am Ausgangsparameter *ERROR* wird der Status der CAN-Übertragung angezeigt. Der I/O-IPC setzt den Errorausgang *CAN_SEND_ERROR* nicht. Hier ist die fehlerfreie Buskommunikation durch Auswertung des Bausteins *CAN_ERROR_INFO* zu identifizieren.

Durch Flankenwechsel TRUE nach FALSE des *ENABLE*-Eingangs wird der Funktionsbaustein wieder in den Grundzustand gesetzt.

19.4.6 CAN_TX_29BIT_FRAME

Mit dem Funktionsbaustein CAN_TX_29BIT_FRAME werden CAN-Nachrichten im 29Bit-Identifizier-Format gesendet.

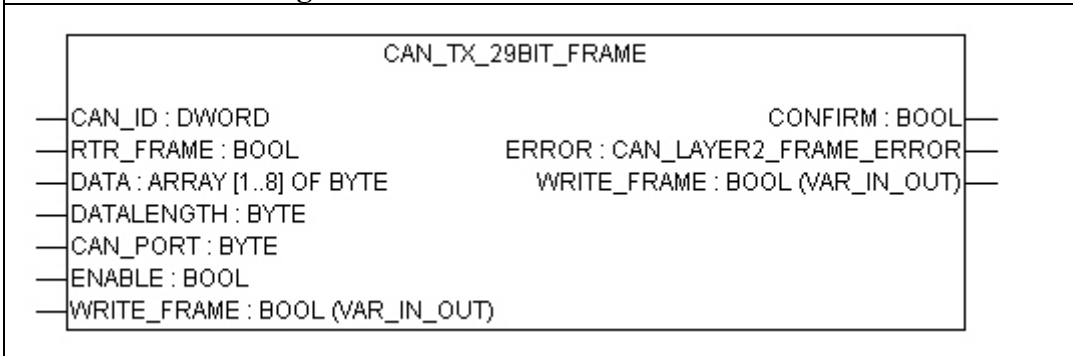
Kategorie	CANLayer 2 Kommunikation
Name	CAN_TX_29BIT_FRAME
Typ	Funktion
Name der Bibliothek	WAGO_CANLayer2_01.lib
Erforderliche Bibliotheken	
Anwendbar für	758-87x-112, 762-3xxx/000-001 bzw. 762-3150/000-003

Eingangsparameter		
Name	Datentyp	Beschreibung
CAN_ID	WORD	29Bit-Identifizier der zu sendenden CAN-Nachricht.
RTR_FRAME	BOOL	Die zu sendende CAN-Nachricht ist ein Remote-Frame.
DATA	ARRAY[1..8] OF BYTE	Beinhaltet die Daten der CAN-Nachricht. Die Anzahl der Bytes entspricht dem Eingang <i>DATALENGTH</i> .
DATALENGTH	BYTE	Datenlänge der zu sendenden CAN-Nachricht.
CAN_PORT	BYTE	Auswahl des CAN-Interfaces 0: CAN 0 Interface aktiviert 1: CAN 1 Interface aktiviert Bei Geräten mit einer Schnittstelle ohne Funktion.
ENABLE	BOOL	Aktivierung der Funktion

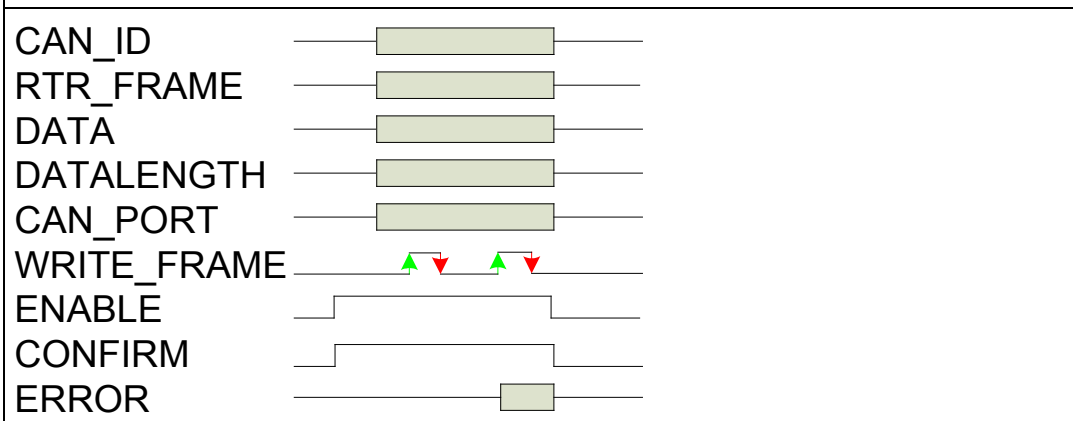
Ausgangsparameter		
Name	Datentyp	Beschreibung
CONFIRM	BOOL	Es wurde eine CAN-Nachricht gesendet.
ERROR	CAN_LAYER2_ FRAME_ ERROR	Siehe Abschnitt Data Types: CAN_LAYER2_FRAME_ERROR

Ein/Ausgangsparameter		
Name	Datentyp	Beschreibung
WRITE_FRAME	BOOL	Übertragung der CAN-Nachricht wird initiiert.

Grafische Darstellung



Zeitliches Verhalten



Beschreibung

Nach dem Setzen des *ENABLE*-Eingangs ist der Funktionsbaustein aktiviert. Danach wird eine CAN-Nachricht durch Flankenwechsel FALSE nach TRUE des *WRITE_FRAME*-Eingangs gesendet. Diese wird aus den Werten an den Eingängen: *CAN_ID*, *RTR_FRAME*, *DATA* und *DATALENGTH* gebildet. Nach der Übertragung der Nachricht wird vom Baustein automatisch die IN/OUT-Variable *WRITE_FRAME* auf FALSE zurückgesetzt.

Am Ausgangsparameter *ERROR* wird der Status der CAN-Übertragung angezeigt. Der IPC setzt den Errorausgang *CAN_SEND_ERROR* nicht. Hier ist die fehlerfreie Buskommunikation durch Auswertung des Bausteins *CAN_ERROR_INFO* zu identifizieren.

Durch Flankenwechsel TRUE nach FALSE des *ENABLE*-Eingangs wird der Funktionsbaustein wieder in den Grundzustand gesetzt.

19.4.7 CAN_LAYER2_FRAME_ERROR

Kategorie	CANLayer 2 Kommunikation
Name	CAN_TX_29BIT_FRAME
Typ	Datentyp
Name der Bibliothek	WAGO_CANLayer2_01.lib
Anwendbar für	758-87x-112, 762-3xxx/000-001 bzw. 762-3150/000-003
Struktur	<pre> TYPE CAN_LAYER2_FRAME_ERROR : (POU_NOT_ENABLED, CAN_BUS_IDLE, CAN_PORT_WRONG, CAN_PORT_BUSY, CAN_SEND_OK, CAN_RECEIVE_OK, CAN_SEND_ERROR, CAN_ID_ERROR, CAN_DATALENGTH_ERROR, CAN_RECEIVE_BUFFER_ERROR, CAN_REGISTER_ERROR); END_TYPE </pre>

Element	Wert	Beschreibung
<i>POU_NOT_ENABLED</i>	0	Der Baustein ist nicht aktiviert.
<i>CAN_BUS_IDLE</i>	1	Es wird z.Zt. kein CAN-Frame verarbeitet.
<i>CAN_PORT_WRONG</i>	2	Es wurde ein CAN-Port ausgewählt, der bei diesem Gerät nicht zur Verfügung steht. Bei Geräten mit einem CAN-Port ist der Eingang <i>CAN_PORT</i> „don't care“. Der Fehler wird nicht erzeugt.
<i>CAN_PORT_BUSY</i>	3	Das gewählte CAN-Interface steht durch CANopen-Slave Projektierung nicht für CAN-Layer 2 Kommunikation zur Verfügung. Gilt nur für PERSPECTO Geräte.
<i>CAN_SEND_OK</i>	4	Es wurde erfolgreich ein CAN-Frame gesendet.
<i>CAN_RECEIVE_OK</i>	5	Es wurde erfolgreich ein CAN-Frame empfangen.
<i>CAN_SEND_ERROR</i>	6	Es ist ein Fehler bei der Versendung eines CAN-Frames aufgetreten.
<i>CAN_ID_ERROR</i>	7	Der Identifier überschreitet den 11- bzw. 29Bit-Bereich oder es wurden mehr als 127 Identifier registriert.
<i>CAN_DATALENGTH_ERROR</i>	8	Bei der Versendung von CAN-Frames (11 und 29 Bit) wurde der zulässige Bereich von 8 Bytes überschritten.

<i>CAN_RECEIVE_BUFFER_ERROR</i>	9	Es ist zu einem Überlauf des internen CAN-Buffers gekommen. Dies deutet darauf hin, dass die Task-Zyklus-Zeit des SPS-Programms für die Bearbeitung der CAN-Frames zu groß eingestellt ist bzw. die CAN-Buslast zu hoch ist.
<i>CAN_REGISTER_ERROR</i>	10	Der Registrierungsbereich von 127 CAN-Frames wurde überschritten.

Beschreibung

Diese Enumeration gibt alle Fehlerzustände an, die bei dem Empfang oder der Übertragung von CAN-Nachrichten auftreten können.

19.5 mod_com.lib

Informationen zu der Bibliothek erhalten Sie unter www.wago.com im Bereich „Downloads“.

Hinweis



Folgende Funktionen sind nicht implementiert:

- SET_DIGITAL_INPUT_OFFSET
- SET_DIGITAL_OUTPUT_OFFSET
- WRITE_OUTPUT_BIT
- READ_OUTPUT_BIT
- READ_INPUT_BIT
- WRITE_OUTPUT_WORD
- READ_OUTPUT_WORD
- READ_INPUT_WORD

19.6 SerComm.lib

Informationen zu der Bibliothek erhalten Sie unter www.wago.com im Bereich „Downloads“.

19.7 WagoLibTerminalDiag.lib

Informationen zu der Bibliothek erhalten Sie unter www.wago.com im Bereich „Downloads“.

19.8 WagoLibKBUS.lib

Mit der WagoLibKBUS-Bibliothek können Sie über Funktionsbausteine den task-synchronen, konsistenten Zugriff auf die Prozessdaten erreichen.

Informationen zu der Bibliothek erhalten Sie unter www.wago.com im Bereich „Downloads“.

19.9 SysLibCom.lib

Mit der SysLibCom-Bibliothek können Sie mit dem WAGO-USB-to-Serial Adapter (761-9005) am I/O-IPC eine zusätzliche RS-232-Schnittstelle anschließen. Der über die USB-Schnittstelle angeschlossene Adapter ist über COM3(=3) zu erreichen.

Weitere Informationen zu der SysLibCom.lib erhalten Sie in der CODESYS-Online-Hilfe.

19.10 SysLibFile, SysLibDir, SysLibFileAsync

Mit den folgenden CODESYS-Bibliotheken SysLibFile, SysLibDir und SysLibFile_Async kann im CODESYS auf das Filesystem zugegriffen werden.

Dabei können die folgenden Verzeichnisse genutzt werden:

1. Speicher auf dem Bootmedium: home/codesys/
2. Speicher auf einer mit FAT formatierten CF-Karte / USB-Stick:
media/<Partitionsname>/
3. Flüchtiger Speicher (RAMdisk): tmp/

Beispiel:

```
h_file:=SysFileOpen(media/USBNAME1/data.log', 'a');
```

Der Partitionsname von CF-Karten/USB-Sticks kann über das WBM/Configtool bei der Formatierung angegeben werden. (siehe Kapitel „Konfiguration mittels Web-based Management (WBM)“).

Außerdem kann die CF-Karte/USB-Stick auch mit anderen Betriebssystemen formatiert werden. Der Partitionsname kann dabei auch angegeben werden.

Beispiel bei WinXP:



Abbildung 136: Formatieren von Partname

Hinweis



Weitere Informationen

Weitere Informationen zu der SysLibFile.lib, SysLibDir und SysLibFileAsync erhalten Sie in der CODESYS-Online-Hilfe.

Abbildungsverzeichnis

Abbildung 1: Übersicht der Schnittstellen	20
Abbildung 2: Kennzeichnung der LEDs	22
Abbildung 3: Bedienelemente	23
Abbildung 4: Batterie	25
Abbildung 5: Seitliche Beschriftung auf dem I/O-IPC	26
Abbildung 6: RJ-45 CP	32
Abbildung 7: Elektronikversorgung (X4)	33
Abbildung 8: Schnittstelle X3	34
Abbildung 9: Anschluss der integrierten Eingänge	36
Abbildung 10: Anschluss der integrierten Ausgänge	37
Abbildung 11: USB-Schnittstelle	38
Abbildung 12: RS-232-Schnittstelle	39
Abbildung 13: DVI-Schnittstelle	41
Abbildung 14: Einbaurichtungen des I/O-IPC; empfohlene Einbaurichtung (A 1)	43
Abbildung 15: Befestigung des I/O-IPC auf einer Tragschiene	44
Abbildung 16: Anstecken einer Busklemme an der Klemmenbusschnittstelle des I/O-IPC	46
Abbildung 17: Schnittstellen des I/O-IPC	47
Abbildung 18: I/O-IPC von der Tragschiene entfernen	49
Abbildung 19: 750-602	52
Abbildung 20: Elektronikversorgung (X4)	53
Abbildung 21: Einspeisung bei Verwendung 750-602 bis HW10 (mit Feldbus) ..	54
Abbildung 22: Einspeisung bei Verwendung 750-602 ab HW11 (mit Feldbus) ..	55
Abbildung 23: Einspeisung bei Verwendung 750-626 (mit Feldbus)	57
Abbildung 24: Konfigurationszeile in der Konfigurationsdatei	62
Abbildung 25: Dialogfenster des WAGO-BootP-Servers mit Nachrichten	64
Abbildung 26: Startbild des WAGO-IPC-Configuration-Tools	65
Abbildung 27: TCP/IP	65
Abbildung 28: TCP/IP-Configuration eth0 (X8)	66
Abbildung 29: IP-Address	66
Abbildung 30: Enter new address	67
Abbildung 31: Beispiel eines Funktionstests	68
Abbildung 32: Ausschalten/Neustart des I/O-IPC	69
Abbildung 33: Authentifizierung eingeben	71
Abbildung 34: Seite „Information“ (Beispiel)	73
Abbildung 35: Zugriff auf das IPC-Configuration-Tool mittels Telnet	89
Abbildung 36: Startbild des „IPC Configuration Tool“	90
Abbildung 37: Anpassen des remanenten Speicherbereichs	102
Abbildung 38: Zielsystem-Einstellungen (1)	105
Abbildung 39: Zielsystem-Einstellungen (2)	105
Abbildung 40: Anlegen eines neuen Bausteins	106
Abbildung 41: Programmieroberfläche mit dem Programmbaustein PLC_PRG	106
Abbildung 42: Registerkarte „Ressourcen“	107
Abbildung 43: Steuerungskonfiguration: Bearbeiten	108
Abbildung 44: Konfiguration	108
Abbildung 45: Schaltfläche „Busklemmen hinzufügen“	108
Abbildung 46: Fenster „Modulauswahl“	109

Abbildung 47: I/O-Konfigurator mit eingetragenen Busklemmen	109
Abbildung 48: Variablendeklaration	110
Abbildung 49: Steuerungskonfiguration: Busklemmen mit den dazugehörigen Adressen.....	110
Abbildung 50: Programmbaustein.....	111
Abbildung 51: Eingabehilfe zur Auswahl der Variablen	111
Abbildung 52: Beispiel einer Zuweisung.....	112
Abbildung 53: Anlegen einer Kommunikationsverbindung 1	113
Abbildung 54: Anlegen einer Kommunikationsverbindung 2	113
Abbildung 55: Anlegen einer Kommunikationsverbindung 3	114
Abbildung 56: Anlegen einer Kommunikationsverbindung (RS-232) 1	115
Abbildung 57: Anlegen einer Kommunikationsverbindung (RS-232) 2	116
Abbildung 58: Task-Konfiguration	118
Abbildung 59: Task-Namen ändern 1	119
Abbildung 60: Aufruf zum Anhängen des Programmbausteins	119
Abbildung 61: Task-Namen ändern 2	120
Abbildung 62: Freilaufende Tasks	121
Abbildung 63: Systemereignisse	122
Abbildung 64: Klemmenbussynchronisation 01	124
Abbildung 65: Klemmenbussynchronisation 02	125
Abbildung 66: Klemmenbussynchronisation 03	126
Abbildung 67: Klemmenbussynchronisation 04	127
Abbildung 68: Auswahl der Visualisierungsvariante in der Zielsystemeinstellung	128
Abbildung 69: Erzeugern der Startvisualisierung PLC_VISU.....	129
Abbildung 70: Anhängen des CANopen-Masters.....	136
Abbildung 71: DSP301 und DSP306 aktivieren	137
Abbildung 72: Anhängen der CANopen-Slaves	137
Abbildung 73: Busklemmen auswählen.....	138
Abbildung 74: Node-ID einstellen	139
Abbildung 75: Karteireiter „Basisparameter“	140
Abbildung 76: Karteireiter „CAN Parameter“	141
Abbildung 77: CANopen-Slaves Karteireiter „Basisparameter“	142
Abbildung 78: Karteireiter „CAN Parameter“ 1	143
Abbildung 79: Karteireiter „CAN Parameter“ 2	144
Abbildung 80: Karteireiter „PDO-Mapping Empfangen“	146
Abbildung 81: Karteireiter „PDO-Mapping Senden“	146
Abbildung 82: PDO-Eigenschaften-Fenster.....	147
Abbildung 83: Karteireiter „Service Data Objects“	148
Abbildung 84: Karteireiter „Modulparameter“	149
Abbildung 85: Steuerungskonfiguration: Busklemmen mit den dazugehörigen Adressen.....	150
Abbildung 86: PLC_PRG.....	151
Abbildung 87: Eingabehilfe zur Auswahl der Variablen	151
Abbildung 88: Beispiel einer Zuweisung der zuvor angelegten Variablen.....	152
Abbildung 89: Karteireiter „Ressourcen“	155
Abbildung 90: Dialog „Öffnen“	156
Abbildung 91: Baustein-Symbol in der Menüleiste; Programmiersprache FUP	156
Abbildung 92: Instanz des Funktionsbausteins DiagGetBusState() in FUP	156
Abbildung 93: Funktionsbaustein DiagGetState() in FUP	157

Abbildung 94: Offline-Ansicht des Variablenfensters in CODESYS.....	157
Abbildung 95: Online-Ansicht des Variablenfensters (oberes Fenster) in FUP .	158
Abbildung 96: Beispiel zur Diagnose	159
Abbildung 97: Diagnoseaufruf DiagGetState().....	160
Abbildung 98: Online-Ansicht des Arrays EXTENDEDINFO in der Binärdarstellung.....	161
Abbildung 99: Anhängen des CANopen-Masters.....	166
Abbildung 100: Baudrate einstellen.....	166
Abbildung 101: EDS-Datei „Generic CAN-Device“	167
Abbildung 102: Karteireiter „Modulparameter“ CAN.....	167
Abbildung 103: Karteireiter „CAN Parameter“ CAN.....	167
Abbildung 104: Datei „libmytest.c“	168
Abbildung 105: Datei „extlibs.ini“	169
Abbildung 106: Datei „extlibs.ini“	169
Abbildung 107: Fenster „Zielsystem Einstellungen“	171
Abbildung 108: Fenster „Neuer Baustein“	171
Abbildung 109: Fenster „MyTestFunction“	172
Abbildung 110: Fenster „Datei speichern unter“	172
Abbildung 111: Zielsystem-Einstellungen (1)	173
Abbildung 112: Zielsystem-Einstellungen (2)	173
Abbildung 113: Fenster „Neuer Baustein“	174
Abbildung 114: Karteireiter „Ressourcen“	174
Abbildung 115: Datei „Beispiel.lib“	177
Abbildung 116: Datei „Beispiel.h“	177
Abbildung 117: Serielle Konsole „Hyperterminal“	180
Abbildung 118: Beispiel mit DOS-Konsole 1.....	184
Abbildung 119: Beispiel mit DOS-Konsole 2.....	184
Abbildung 120: RS-232-Schnittstelle X6.....	185
Abbildung 121: DVI-I-Schnittstelle X7 und USB-Schnittstellen X10/11	186
Abbildung 122: DOS-Konsole	195
Abbildung 123: Kennzeichnung der LEDs	200
Abbildung 124: Anzeige der Blinkcodes durch die I/O-LED.....	204
Abbildung 125: Ablaufdiagramm der Blinksequenz.....	205
Abbildung 126: Batteriewechsel der Notstromversorgung 1	213
Abbildung 127: Batteriewechsel der Notstromversorgung 2	213
Abbildung 128: Beispiel für die seitliche Bedruckung der ATEX- und IECEx- zugelassenen Busklemmen.	216
Abbildung 129: Textdetail - Beispielbedruckung der ATEX- und IECEx- zugelassenen Busklemmen.	216
Abbildung 130: Beispiel für die seitliche Bedruckung der ATEX- und IECEx- zugelassenen Ex i Busklemmen.....	218
Abbildung 131: Textdetail - Beispielbedruckung der ATEX- und IECEx- zugelassenen Ex i Busklemmen.....	218
Abbildung 132: Beispiel für seitliche Bedruckung der Busklemmen gemäß NEC 500.....	221
Abbildung 133: Textdetail - Beispielbedruckung der Busklemmen gemäß NEC 500.....	221
Abbildung 134: Grafische Darstellung des Funktionsbausteins „ConfigToolFB“	261

Abbildung 135: Grafische Darstellung des Funktionsbausteins „ConfigToolFB“	263
Abbildung 136: Formatieren von Partname	353

Tabellenverzeichnis

Tabelle 1: Darstellungen der Zahlensysteme	12
Tabelle 2: Schriftkonventionen	12
Tabelle 3: Legende zur Abbildung „Übersicht der physikalischen Schnittstellen“	20
Tabelle 4: Legende zur Abbildung „Anzeigeelemente“	22
Tabelle 5: Legende zur Abbildung „Bedienelemente“	24
Tabelle 6: Technische Daten Gerät	27
Tabelle 7: Technische Daten – Systemdaten	28
Tabelle 8: Technische Daten – Versorgung	28
Tabelle 9: Technische Daten – Kommunikation	29
Tabelle 10: Technische Daten – Schutz und Sicherheit	29
Tabelle 11: Technische Daten – Laufzeitsystem	29
Tabelle 12: Technische Daten – Umgebungsbedingungen	29
Tabelle 13: Technische Daten – Anschlusstechnik	30
Tabelle 14: Technische Daten – Elektromagnetische Verträglichkeit	30
Tabelle 15: ACT/LNK- und Speed-LED	32
Tabelle 16: Ethernet-Schnittstellen: Anschlussbelegung	32
Tabelle 17: Schnittstelle für die Elektronikversorgung: Anschlussbelegung	33
Tabelle 18: CANOpen-Schnittstelle: Anschlussbelegung	34
Tabelle 19: Digitale Ein- und Ausgänge: Anschlussbelegung	35
Tabelle 20: USB-Schnittstellen: Anschlussbelegung	38
Tabelle 21: RS-232-Schnittstelle: Anschlussbelegung	39
Tabelle 22: DVI-I-Schnittstelle: Anschlussbelegung	41
Tabelle 23: Verwendung von 750-602/626 in Abhängigkeit des I/O-IPC- Anwendungsbereichs	50
Tabelle 24: Anschlüsse, Kontakte und LEDs der Einspeiseklemme	53
Tabelle 25: Anschluss für die Elektronikversorgung: Anschlussbelegung	53
Tabelle 26: Anschluss für die Elektronikversorgung: Anschlussbelegung	56
Tabelle 27: Voreingestellte IP-Adressierungen der Ethernet-Schnittstellen	61
Tabelle 28: Netzmaske 255.255.255.0	61
Tabelle 29: Erläuterungen der Konfigurationszeile	63
Tabelle 30: Benutzereinstellungen im Auslieferungszustand	72
Tabelle 31: Zugriffsrechte für die WBM-Seiten	72
Tabelle 32: Beschreibung der Parameter der Seite „Information“	73
Tabelle 33: Beschreibung der Parameter der Seite „CODESYS“	74
Tabelle 34: Beschreibung der Parameter der Seite „TCP/IP“	75
Tabelle 35: Beschreibung der Parameter der Seite „ETHERNET“	76
Tabelle 36: Beschreibung der Parameter der Seite „NTP“	76
Tabelle 37: Beschreibung der Parameter der Seite „Clock“	77
Tabelle 38: Beschreibung der Parameter der Seite „Users“	78
Tabelle 39: Beschreibung der Parameter Screensaver und Cleanmode der Seite „HMI Settings“	79
Tabelle 40: Beschreibung der Parameter der Seite „Administration“	81
Tabelle 41: Beschreibung der Parameter der Seite „Package Server“	82
Tabelle 42: Beschreibung der Parameter der Seite „Mass Storage“	84
Tabelle 43: Beschreibung der Parameter der Seite „MODBUS“	85
Tabelle 44: Beschreibung der Parameter der Seite „SNMP“	86
Tabelle 45: Beschreibung der Parameter der Seite „I/O Configuration“	88

Tabelle 46: Elemente eines MODBUS/TCP-Telegramms.....	91
Tabelle 47: MODBUS-Grunddatentypen.....	91
Tabelle 48: MODBUS-Funktionscodes	92
Tabelle 49: Lesen von Analogeingangsklemmen mittels FC3, FC4, FC23	93
Tabelle 50: Schreiben von Analogausgangsklemmen mittels FC6, FC16, FC23	93
Tabelle 51: Lesen von Digitaleingangsklemmen mittels FC1, FC2	94
Tabelle 52: Schreiben von Digitalausgangsklemmen mittels FC5, FC15.....	94
Tabelle 53: Konfigurationsregister.....	95
Tabelle 54: Anordnung der Busklemmen für das Adressierungsbeispiel	96
Tabelle 55: Schreibweise logischer Adressen	99
Tabelle 56: Zugriff auf die Prozessabbilder der Ein- und Ausgangsdaten über CODESYS	100
Tabelle 57: Anordnung der Busklemmen für das Adressierungsbeispiel	103
Tabelle 58: Events	123
Tabelle 59: Namenskonvention für Fonts (Beispiel)	130
Tabelle 60: Fehler und deren Abhilfe.....	133
Tabelle 61: Beschreibung der Basisparameter	140
Tabelle 62: Beschreibung der CAN-Parameter.....	141
Tabelle 63: Beschreibung der Basisparameter	142
Tabelle 64: Beschreibung der CAN-Parameter.....	143
Tabelle 65: Beschreibung der Parameter.....	145
Tabelle 66: Beschreibung zum PDO-Mapping empfangen und senden	146
Tabelle 67: Beschreibung des PDO-Eigenschaften-Fensters	147
Tabelle 68: Beschreibung der Modulparameter (Slave).....	149
Tabelle 69: Bits der Diagnoseinformation	159
Tabelle 70: Actual error	164
Tabelle 71: Datentypen	176
Tabelle 72: Benutzer für die Linux-Konsole.....	182
Tabelle 73: Aufbau des Dateisystems	188
Tabelle 74: Betriebs- und Statusmeldungen des I/O-IPC	200
Tabelle 75: Betriebs- und Statusmeldungen des I/O-IPC	202
Tabelle 76: Betriebs- und Statusmeldungen der „ERR“- und „STA“-LEDs	203
Tabelle 77: Bedeutung der Blinkcodes und Maßnahmen zur Fehlerbehebung... ..	207
Tabelle 78: Beschreibung der Beispielbedruckung der ATEX- und IECEx- zugelassenen Busklemmen.	217
Tabelle 79: Beschreibung der Beispielbedruckung der ATEX- und IECEx- zugelassenen Ex i Busklemmen.....	219
Tabelle 80: Beschreibung der Beispielbedruckung der Busklemmen gemäß NEC 500.....	221
Tabelle 81: 1-Kanal-Digitaleingangsklemmen mit Diagnose	230
Tabelle 82: 2-Kanal-Digitaleingangsklemmen	230
Tabelle 83: 2-Kanal-Digitaleingangsklemmen mit Diagnose.....	230
Tabelle 84: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten	231
Tabelle 85: 4-Kanal-Digitaleingangsklemmen	231
Tabelle 86: 8-Kanal-Digitaleingangsklemmen	231
Tabelle 87: 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten	232
Tabelle 88: 16-Kanal-Digitaleingangsklemmen	232
Tabelle 89: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten	233

Tabelle 90: 2-Kanal-Digitalausgangsklemmen	233
Tabelle 91: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten	234
Tabelle 92: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506	234
Tabelle 93: 4-Kanal-Digitalausgangsklemmen	235
Tabelle 94: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten	235
Tabelle 95: 8-Kanal-Digitalausgangsklemmen	235
Tabelle 96: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten	236
Tabelle 97: 16-Kanal-Digitalausgangsklemmen	236
Tabelle 98: 8-Kanal-Digitalein-/ -ausgangsklemmen	237
Tabelle 99: 1-Kanal-Analogeingangsklemmen	238
Tabelle 100: 2-Kanal-Analogeingangsklemmen	238
Tabelle 101: 4-Kanal-Analogeingangsklemmen	239
Tabelle 102: 3-Phasen-Leistungsmessklemme	239
Tabelle 103: 8-Kanal-Analogeingangsklemmen	240
Tabelle 104: 2-Kanal-Analogausgangsklemmen	241
Tabelle 105: 4-Kanal-Analogausgangsklemmen	241
Tabelle 106: 8-Kanal-Analogausgangsklemmen	242
Tabelle 107: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753- 404, (und Variante /000-003).....	243
Tabelle 108: Zählerklemmen 750-404/000-005	244
Tabelle 109: Zählerklemmen 750-638, 753-638	244
Tabelle 110: Pulsweitenklemmen 750-511, /xxx-xxx	245
Tabelle 111: Serielle Schnittstellen mit alternativem Datenformat	245
Tabelle 112: Serielle Schnittstellen mit Standard-Datenformat	246
Tabelle 113: Datenaustauschklemmen	246
Tabelle 114: SSI-Geber Interface Busklemmen mit alternativem Datenformat ..	247
Tabelle 115: Weg- und Winkelmessung 750-631/000-004, --010, -011	247
Tabelle 116: Incremental-Encoder-Interface 750-634	248
Tabelle 117: Incremental-Encoder-Interface 750-637	248
Tabelle 118: Digitale Impuls Schnittstelle 750-635	249
Tabelle 119: Antriebssteuerung 750-636	249
Tabelle 120: Steppercontroller RS 422 / 24 V / 20 mA 750-670	250
Tabelle 121: RTC-Modul 750-640	251
Tabelle 122: DALI/DSI-Masterklemme 750-641	251
Tabelle 123: Übersicht über das Eingangsprozessabbild im „Easy-Modus“	253
Tabelle 124: Übersicht über das Ausgangsprozessabbild im „Easy-Modus“	253
Tabelle 125: Funkreceiver EnOcean 750-642	254
Tabelle 126: MP-Bus-Masterklemme 750-643	255
Tabelle 127: Bluetooth® RF-Transceiver 750-644	255
Tabelle 128: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645	256
Tabelle 129: KNX/EIB/TP1-Klemme 753-646	257
Tabelle 130: AS-Interface-Masterklemme 750-655	258
Tabelle 131: Systemklemmen mit Diagnose 750-610, -611	258
Tabelle 132: Binäre Platzhalterklemmen 750-622 (mit dem Verhalten einer 2 DI)	259
Tabelle 133: Funktionsbaustein „ConfigTool“	261

Tabelle 134: Funktion STRING_TO_IP	262
Tabelle 135: Funktion IP_TO_STRING	262
Tabelle 136: Beschreibung der Konfigurierungsskripte zu „Information“	263
Tabelle 137: Beschreibung der Konfigurierungsskripte zu „CODESYS“	264
Tabelle 138: Beschreibung der Konfigurierungsskripte zu „TCP/IP“	264
Tabelle 139: Beschreibung der Konfigurierungsskripte zu „ETHERNET“	267
Tabelle 140: Beschreibung der Konfigurierungsskripte zu „NTP“	268
Tabelle 141: Beschreibung der Konfigurierungsskripte zu „Clock“	269
Tabelle 142: Beschreibung der Konfigurierungsskripte zu „HMI Settings“	270
Tabelle 143: Beschreibung der Konfigurierungsskripte zu „Administration“	273
Tabelle 144: Beschreibung der Konfigurierungsskripte zu „Package Server“	275
Tabelle 145: Beschreibung der Konfigurierungsskripte zu „Mass Storage“	275
Tabelle 146: Beschreibung der Konfigurierungsskripte zu „Port“	276
Tabelle 147: Beschreibung der Konfigurierungsskripte zu „MODBUS“	277
Tabelle 148: Beschreibung der Konfigurierungsskripte zu „Allgemeine SNMP- Informationsparameter“	279
Tabelle 149: Datentypen	285
Tabelle 150: Parameter snmpRegisterCustomOID_INT32()	285
Tabelle 151: Return snmpRegisterCustomOID_INT32()	285
Tabelle 152: Parameter snmpRegisterCustomOID_STRING()	286
Tabelle 153: Return snmpRegisterCustomOID_STRING()	286
Tabelle 154: Parameter snmpRegisterCustomOID_UINT32()	287
Tabelle 155: Return snmpRegisterCustomOID_UINT32()	287
Tabelle 156: Parameter snmpGetValueCustomOID_INT32()	288
Tabelle 157: Return snmpGetValueCustomOID_INT32()	288
Tabelle 158: Parameter snmpGetValueCustomOID_STRING()	289
Tabelle 159: Return snmpGetValueCustomOID_STRING()	289
Tabelle 160: Parameter snmpGetValueCustomOID_INT32()	290
Tabelle 161: Return snmpGetValueCustomOID_INT32()	290
Tabelle 162: Parameter snmpSetValueCustomOID_INT32()	291
Tabelle 163: Return snmpSetValueCustomOID_INT32()	291
Tabelle 164: Parameter snmpSetValueCustomOID_STRING()	292
Tabelle 165: Return snmpSetValueCustomOID_STRING()	292
Tabelle 166: Parameter snmpSetValueCustomOID_UINT32()	293
Tabelle 167: Return snmpSetValueCustomOID_UINT32()	293
Tabelle 168: Fehlermeldungen	294

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Telefon: 05 71/8 87 – 0
Telefax: 05 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: <http://www.wago.com>

