



ibaPDA-Request-S7-DP/PN/ibaNet-E

Anwendungen der Request-Datenschnittstelle zu
SIMATIC S7-300/S7-400

Handbuch Teil 2

Ausgabe 4.0

Messsysteme für Industrie und Energie

www.iba-ag.com

Hersteller

iba AG
Königswarterstraße 44
90762 Fürth
Deutschland

Kontakte

Zentrale	+49 911 97282-0
Support	+49 911 97282-14
Technik	+49 911 97282-13
E-Mail	iba@iba-ag.com
Web	www.iba-ag.com

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts sind nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz.

© iba AG 2024, alle Rechte vorbehalten.

Der Inhalt dieser Druckschrift wurde auf Übereinstimmung mit der beschriebenen Hard- und Software überprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass für die vollständige Übereinstimmung keine Garantie übernommen werden kann. Die Angaben in dieser Druckschrift werden jedoch regelmäßig aktualisiert. Notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten oder können über das Internet heruntergeladen werden.

Die aktuelle Version liegt auf unserer Website www.iba-ag.com zum Download bereit.

Version	Datum	Revision	Autor	Version SW
4.0	04-2024	GUI neu, ibaNNet-E hinzu, S7-Routing via TIA Portal	st, mm	8.7.0

Windows® ist eine Marke und eingetragenes Warenzeichen der Microsoft Corporation. Andere in diesem Handbuch erwähnte Produkt- und Firmennamen können Marken oder Handelsnamen der jeweiligen Eigentümer sein.

Inhalt

1	Zu dieser Dokumentation.....	7
1.1	Zielgruppe und Vorkenntnisse.....	7
1.2	Schreibweisen.....	8
1.3	Verwendete Symbole.....	9
1.4	Aufbau der Dokumentation.....	10
2	Request-S7-Varianten.....	11
2.1	Request-S7 für ibaBM-PN	11
2.1.1	Allgemeine Informationen.....	11
2.1.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	12
2.1.2.1	Projektierung Hardware	13
2.1.2.2	Projektierung in STEP 7.....	14
2.1.3	Konfiguration und Projektierung ibaPDA.....	17
2.1.3.1	Allgemeine Einstellungen der Schnittstelle	17
2.1.3.2	Allgemeine Moduleinstellungen.....	19
2.1.3.3	Modul S7 Request.....	19
2.1.3.4	Modul S7 Request Decoder	19
2.1.4	Diagnose	20
2.2	Request-S7 für ibaBM-PN im Redundanzmodus	21
2.2.1	Allgemeine Informationen.....	21
2.2.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	22
2.2.2.1	Projektierung Hardware	22
2.2.2.2	Projektierung in STEP 7.....	23
2.2.3	Konfiguration und Projektierung ibaPDA.....	23
2.3	Request-S7 für ibaBM-DP	25
2.3.1	Allgemeine Informationen.....	25
2.3.1.1	ibaCom-L2B Kompatibilitätsbetrieb.....	27
2.3.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	27
2.3.2.1	Projektierung Hardware	28
2.3.2.2	Projektierung in STEP 7 (KOP, FUP, AWL)	29
2.3.2.3	Projektierung in STEP 7 (CFC)	30

2.3.3	Konfiguration und Projektierung ibaPDA.....	32
2.3.3.1	Allgemeine Einstellungen der Schnittstelle	32
2.3.3.2	Allgemeine Moduleinstellungen.....	34
2.3.3.3	Modul S7 Request.....	35
2.3.3.4	Modul S7 Request Decoder	35
2.3.3.5	Modul S7 Request (ibaCom-L2B kompatibel)	37
2.3.3.6	Modul S7 Request Dig512 (ibaCom-L2B kompatibel)	37
2.3.4	Diagnose	38
2.4	Request-S7 für ibaBM-DP im Redundanzmodus.....	39
2.4.1	Allgemeine Informationen.....	39
2.4.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	40
2.4.2.1	Projektierung Hardware	40
2.4.2.2	Projektierung in STEP 7 (KOP, FUP, AWL)	42
2.4.3	Konfiguration und Projektierung ibaPDA.....	44
2.5	Request-S7 für ibaBM-DPM-S.....	46
2.5.1	Allgemeine Informationen.....	46
2.5.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	47
2.5.3	Konfiguration und Projektierung ibaPDA.....	48
2.5.3.1	Allgemeine Einstellungen der Schnittstelle	48
2.5.3.2	Allgemeine Moduleinstellungen.....	50
2.5.3.3	Verbindungseinstellungen.....	50
2.5.3.4	Modul S7 Request.....	50
2.5.3.5	Modul S7 Request Decoder	50
2.5.3.6	Modul Dig512 S7 Request	51
2.5.4	Diagnose	51
2.6	Request-S7 für ibaBM-DPM-S im Redundanzmodus	52
2.6.1	Allgemeine Informationen.....	52
2.6.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	53
2.6.3	Konfiguration und Projektierung ibaPDA.....	53
2.7	Request-S7 für ibaCom-L2B	54
2.7.1	Allgemeine Informationen.....	54
2.7.2	Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC	54
2.7.2.1	Projektierung Hardware	54
2.7.2.2	Projektierung in STEP 7 (KOP, FUP, AWL)	55
2.7.2.3	Projektierung in STEP 7 (CFC)	60

2.7.3	Konfiguration und Projektierung ibaPDA.....	63
2.7.3.1	Allgemeine Einstellungen der Schnittstelle	63
2.7.3.2	Allgemeine Moduleinstellungen.....	64
2.7.3.3	Modul L2B S7 Request.....	64
2.7.3.4	Modul L2B S7 Request Dig512.....	65
2.7.4	Diagnose	67
2.7.4.1	ibaCom-L2B-Karte.....	67
2.7.4.2	Moduldiagnose.....	68
3	Beschreibung der Request-Blöcke.....	69
3.1	iba-Baustein-Familie ibaREQ.....	69
3.1.1	ibaREQ_M (FB140).....	70
3.1.2	ibaREQ_PN (FB141)	71
3.1.3	ibaREQ_DP (FB142)	72
3.1.4	ibaREQ_PNdev (FB150).....	73
3.2	Beschreibung der Request-FCs	73
3.2.1	Request-FC ibaDP_Req (FC122)	74
3.2.2	Request-FC ibaDP_Req_H (FC123).....	76
3.2.3	Initialisierungs-FC ibaL2B_Init (FC111)	80
3.2.4	Kommunikations-FC ibaL2B_Req_CP (FC113) für CP342-5.....	83
3.2.5	Kommunikations-FC ibaL2B_Req (FC112).....	85
4	Diagnose	87
4.1	Lizenz	87
4.2	Sichtbarkeit der Schnittstelle	87
4.3	Protokolldateien	88
4.4	Verbindungsdiagnose mittels PING	89
4.5	Verbindungsdiagnose mittels PG/PC-Schnittstelle	90
4.6	Moduldiagnose.....	91
5	Anhang	92
5.1	iba S7-Bibliothek.....	92
5.1.1	iba S7-Bibliothek für SIMATIC Manager.....	92
5.1.1.1	Bibliothek in SIMATIC Manager einbinden	93
5.1.1.2	Bausteine in SIMATIC Manager übernehmen.....	94
5.2	Anwendungsbeispiele.....	95

5.3	S7-Zykluszeitmessungen	97
5.3.1	ibaCom-L2B.....	97
5.3.2	ibaBM-DP	98
5.3.3	ibaBM-PN.....	99
5.4	S7-Routing	100
5.4.1	Routing von Ethernet auf Ethernet.....	100
5.4.1.1	Konfiguration von STEP 7/NetPro	101
5.4.1.2	Konfiguration von ibaPDA.....	103
5.4.2	Routing von Ethernet auf PROFIBUS.....	104
5.4.2.1	Konfiguration von STEP7/NetPro.....	105
5.4.2.2	Konfiguration von ibaPDA.....	107
5.5	Ablösung Request-S7 auf ibaCom-L2B durch ibaBM-DP	108
5.6	Fehlercodes Request-Blöcke	111
5.7	Nutzung von MPI/DP-TCP-Adapttern	114
6	Support und Kontakt	115

1 Zu dieser Dokumentation

Dieses Handbuch beschreibt die Anwendung der Request-Datenschnittstelle zu SIMATIC S7.

Das Produkt *ibaPDA-Request-S7-DP/PN/ibaNet-E* ist eine Erweiterung von *ibaPDA* für den wahlfreien Zugriff auf S7-Symbole und S7-Operanden bei der Aufzeichnung von Daten aus SIMATIC S7-CPU's. In dem vorliegenden Handbuch werden nur die Erweiterungen und Abweichungen dargestellt. Für alle anderen Funktionen und Bedienungsmöglichkeiten wird auf das Handbuch von *ibaPDA* verwiesen.

Andere Dokumentation



Dieses Dokument ist eine Ergänzung zu dem allgemeinen Handbuch von *ibaPDA*.

1.1 Zielgruppe und Vorkenntnisse

Diese Dokumentation wendet sich an ausgebildete Fachkräfte, die mit dem Umgang mit elektrischen und elektronischen Baugruppen sowie der Kommunikations- und Messtechnik vertraut sind. Als Fachkraft gilt, wer auf Grund der fachlichen Ausbildung, Kenntnisse und Erfahrungen sowie Kenntnis der einschlägigen Bestimmungen die übertragenen Arbeiten beurteilen und mögliche Gefahren erkennen kann.

Im Besonderen wendet sich diese Dokumentation an Personen, die mit Projektierung, Test, Inbetriebnahme oder Instandhaltung von Speicherprogrammierbaren Steuerungen der unterstützten Fabrikate befasst sind. Für den Umgang mit *ibaPDA-Request-S7-DP/PN/ibaNet-E* sind folgende Vorkenntnisse erforderlich bzw. hilfreich:

- Betriebssystem Windows
- Grundkenntnisse *ibaPDA*
- Grundkenntnisse Netzwerktechnik
- Kenntnis von Projektierung und Betrieb von SIMATIC S7 Steuerungen

1.2 Schreibweisen

In dieser Dokumentation werden folgende Schreibweisen verwendet:

Aktion	Schreibweise
Menübefehle	Menü <i>Funktionsplan</i>
Aufruf von Menübefehlen	<i>Schritt 1 – Schritt 2 – Schritt 3 – Schritt x</i> Beispiel: Wählen Sie Menü <i>Funktionsplan – Hinzufügen – Neuer Funktionsblock</i>
Tastaturtasten	<Tastename> Beispiel: <Alt>; <F1>
Tastaturtasten gleichzeitig drücken	<Tastename> + <Tastename> Beispiel: <Alt> + <Strg>
Grafische Tasten (Buttons)	<Tastename> Beispiel: <OK>; <Abbrechen>
Dateinamen, Pfade	<i>Dateiname, Pfad</i> Beispiel: <i>Test.docx</i>

1.3 Verwendete Symbole

Wenn in dieser Dokumentation Sicherheitshinweise oder andere Hinweise verwendet werden, dann bedeuten diese:

Gefahr!



Wenn Sie diesen Sicherheitshinweis nicht beachten, dann droht die unmittelbare Gefahr des Todes oder der schweren Körperverletzung!

- Beachten Sie die angegebenen Maßnahmen.
-

Warnung!



Wenn Sie diesen Sicherheitshinweis nicht beachten, dann droht die mögliche Gefahr des Todes oder schwerer Körperverletzung!

- Beachten Sie die angegebenen Maßnahmen.
-

Vorsicht!



Wenn Sie diesen Sicherheitshinweis nicht beachten, dann droht die mögliche Gefahr der Körperverletzung oder des Sachschadens!

- Beachten Sie die angegebenen Maßnahmen.
-

Hinweis



Hinweis, wenn es etwas Besonderes zu beachten gibt, wie z. B. Ausnahmen von der Regel usw.

Tipp



Tipp oder Beispiel als hilfreicher Hinweis oder Griff in die Trickkiste, um sich die Arbeit ein wenig zu erleichtern.

Andere Dokumentation



Verweis auf ergänzende Dokumentation oder weiterführende Literatur.

1.4 Aufbau der Dokumentation

Die Dokumentation der Schnittstelle *ibaPDA-Request-S7-DP/PN/ibaNet-E* (PDF-Ausgabe) ist in zwei separate Teile gegliedert. Jeder Teil hat seine eigene bei 1 beginnende Kapitel- und Seitennummerierung und wird unabhängig aktualisiert.

Teil	Titel	Inhalt
Teil 1	Anwendungen der Request-Daten-schnittstelle zu SIMATIC S7 TIA Portal/S7-1500	Funktionen, Einstellungen und Request-Bau-steine der Request-Schnittstelle in Verbindung mit SIMATIC TIA Portal und S7-Steuerung S7-1500
Teil 2	Anwendungen der Request-Daten-schnittstelle zu SIMATIC S7-300/S7-400	Funktionen, Einstellungen und Request-Bau-steine der Request-Schnittstelle in Verbindung mit SIMATIC STEP 7 sowie S7-Steuerungen S7-300 und S7-400

2 Request-S7-Varianten

Im folgenden Kapitel sind die Request-S7-Varianten für die Anwendung mit S7-300/400 Steuerungen beschrieben. Für Anwendungen mit S7-1500 Steuerungen siehe Handbuch Teil 1.

S7-Familie	Firmware	Adressierungsart	iba-Baustein-Familie	iba-Datenpfade
S7-300	unbeschränkt	Operanden	ibaREQ	DP, PN, UDP
S7-400	unbeschränkt	Operanden	ibaREQ	DP, PN, UDP

2.1 Request-S7 für ibaBM-PN

Im Folgenden wird die Request-S7-Variante für das PROFINET-Busmodul *ibaBM-PN* beschrieben.

2.1.1 Allgemeine Informationen

Request-S7 für ibaBM-PN können Sie in folgenden Systemkonfigurationen projektieren.

SIMATIC S7-CPU	SIMATIC STEP 7 V5.x (SIMATIC Manager)	SIMATIC STEP 7 V1x Professional (TIA Portal)
S7-300 integrierte PN-Schnittstelle	X	X
S7-400 integrierte PN-Schnittstelle und CP443-1	X	X

Beim PN-Request erfolgt die Anforderung der Messwerte (Request-Handshake) über eine separate TCP/IP-Verbindung.

Je nach vorhandener Hardware und Software stehen verschiedene Zugangspunkte zur Auswahl, über welche die Anforderung erfolgen kann:

- **TCP/IP:** Die Verbindung zur SIMATIC S7 wird über eine integrierte PN-Schnittstelle der S7-CPU oder entsprechende CP-Baugruppen in der SPS und der Standard-Netzwerkschnittstelle des Rechners hergestellt. Es ist keine weitere Siemens-Software für die Verbindung erforderlich.
- **PC/CP:** Hinter dieser Bezeichnung verbergen sich verschiedene Zugangspunkte, die SIMATIC-spezifisch sind. Im Gegensatz zur TCP/IP-Verbindung ist allen Verbindungsarten der Gruppe PC/CP ist gemeinsam, dass auf dem Rechner die SIMATIC-Kommunikationssoftware mit den entsprechenden Freischaltungen installiert sein muss.
 - **MPI, PROFIBUS:** Die Verbindung zur SIMATIC S7 wird über die MPI- bzw. PROFIBUS-Schnittstelle des Rechners hergestellt, z. B. mit der PCI-Karte CP5611 oder dem MPI-Adapter für USB-Schnittstellen oder serieller PC-Schnittstelle.
 - **TCP/IP, ISO:** Hier wird entweder die Standard-Netzwerkschnittstelle des Rechners oder eine geeignete Schnittstellenkarte für die Verbindung zur S7 verwendet.

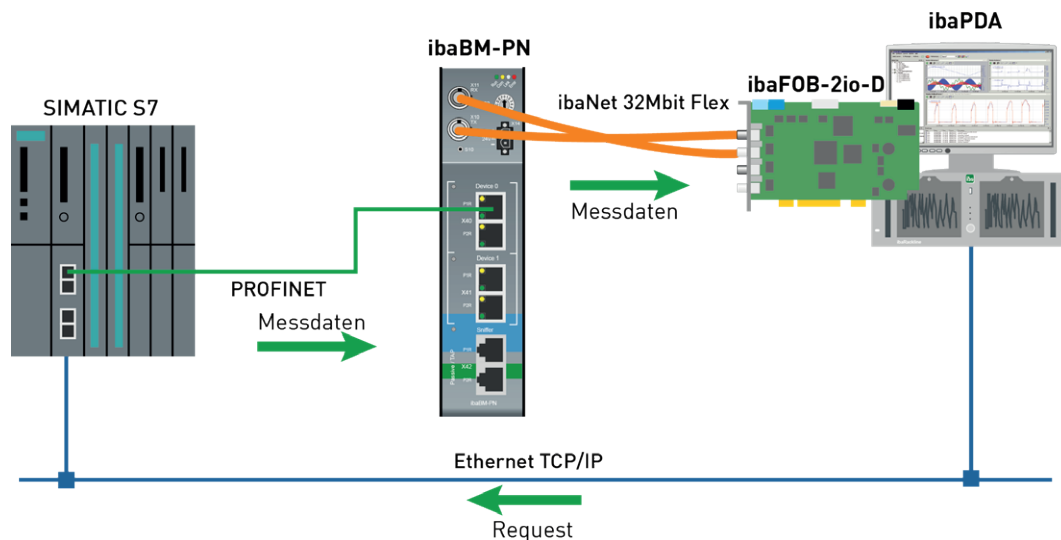
Systemintegration mit ibaBM-PN

Die Übertragung der Messdaten erfolgt über PROFINET an das *ibaBM-PN*-Gerät.

Sie benötigen insgesamt folgende Verbindungen:

- Onlineverbindung zwischen *ibaPDA* und S7-CPU (TCP/IP, MPI oder DP)
- Lichtwellenleiterverbindung zwischen *ibaPDA/ibaFOB-io-D* und *ibaBM-PN*
- PROFINET-Verbindung zwischen *ibaBM-PN* und S7-CPU

Der Busmonitor bietet zwei separate PROFINET-Devices. Pro Device ist eine Übertragung von max. 1440 Byte Daten möglich.



Andere Dokumentation



Detaillierte Informationen zu *ibaBM-PN* finden Sie im Gerätehandbuch.

Informationen zu Anwendungsbeispielen finden Sie in Kapitel [↗ Anwendungsbeispiele](#), Seite 95.

2.1.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

Im Folgenden wird die Konfiguration und Projektierung auf SIMATIC S7-Seite mit dem SIMATIC Manager (STEP 7 Version ≤ V5) beschrieben.

Nehmen Sie auf SIMATIC-Seite folgende Konfigurations- und Projektierungsschritte vor:

- Projektierung Hardware:
Einbinden des PROFINET-Devices in die Hardwarekonfiguration
- Projektierung Software (STEP 7 V5):
Einbinden der Request-Blöcke in das S7-Programm

2.1.2.1 Projektierung Hardware

Pro genutztem PROFINET-Device müssen Sie ein PROFINET-Device projektieren.

Verwenden Sie die GSDML-Datei [GSDML-Vx.yy-ibaBM-PN-yyyymmdd.xml](#).

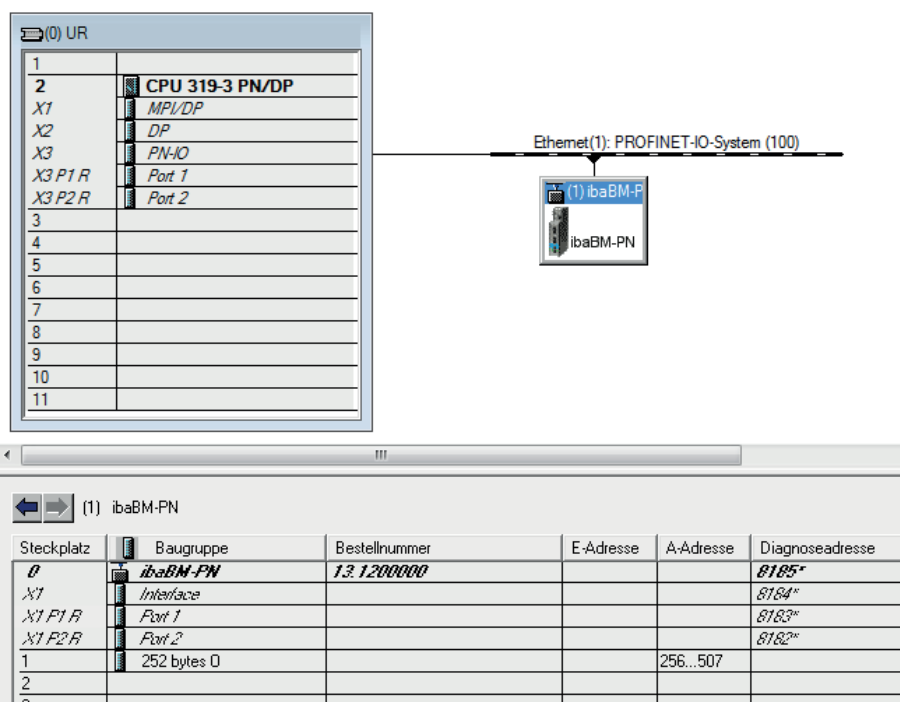
Hinweis



Die aktuelle GSDML-Datei finden Sie auf dem Datenträger "iba Software & Manuals" in folgendem Verzeichnis:

[\02_iba_Hardware\ibaBM-PN\01_GSD_Files\](#)

S7-CPU's ermöglichen konsistente Slots mit max. 252 Bytes. Sie benötigen einen Slot je Request-Block. Sie können auch Slots mit geringerer Größe verwenden.



2.1.2.2 Projektierung in STEP 7

Im Folgenden wird die Projektierung der Request-Blöcke in STEP 7 V5 beschreiben.

Request-S7 ist für die Verwendung mit CPUs S7-300/400 mit integrierter PN-Schnittstelle oder CPUs S7-400 mit externer Schnittstelle CP 443-1 (PROFINET-Controller) vorgesehen.

Hinweis



Die Verwendung der externen PN-Schnittstelle CP343-1 einer CPU S7-300 wird nicht unterstützt!

Hinweis



Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!
Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

- Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe ↗ *iba S7-Bibliothek*, Seite 92. Je Request-Modul (Verbindung) in *ibaPDA* muss ein Satz Request-Blöcke aufgerufen werden.
 - ibaREQ_M (FB140), siehe ↗ *ibaREQ_M (FB140)*, Seite 70
 - ibaREQ_PN (FB141), siehe ↗ *ibaREQ_PN (FB141)*, Seite 71
 - ibaREQ_PNdev (FB150), siehe ↗ *ibaREQ_PNdev (FB150)*, Seite 73
 - ibaREQ_DB (DB15)

Hinweis



Falls die Bausteinnummern in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

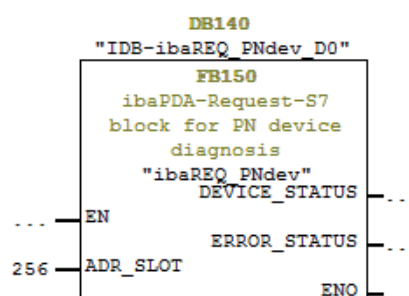
Hinweis



Die Request-Blöcke unterstützen keinen Aufruf als Multiinstanz.

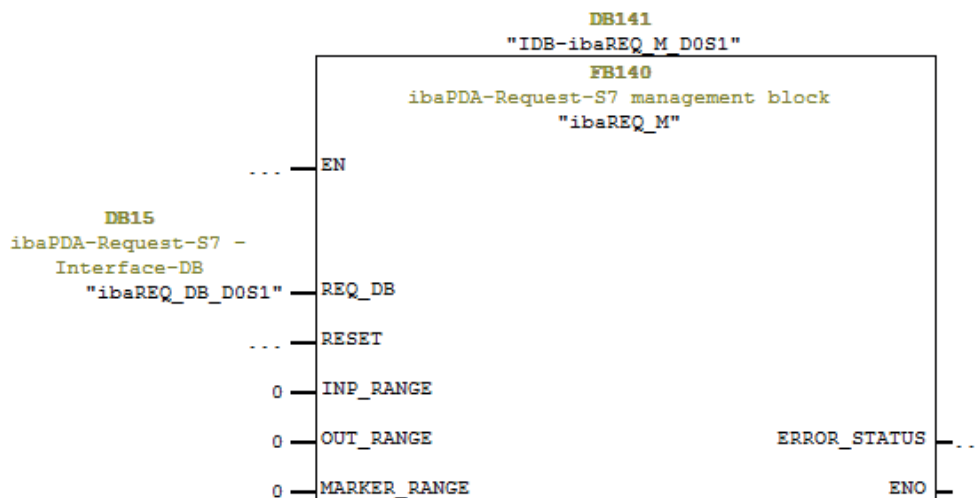
Für jedes verwendete PROFINET Device des ibaBM-PN

- ibaREQ_PNdev (FB150) innerhalb des OB1 oder eines Weckalarm-OB (OB3x) aufrufen.



Für jedes Request-Modul

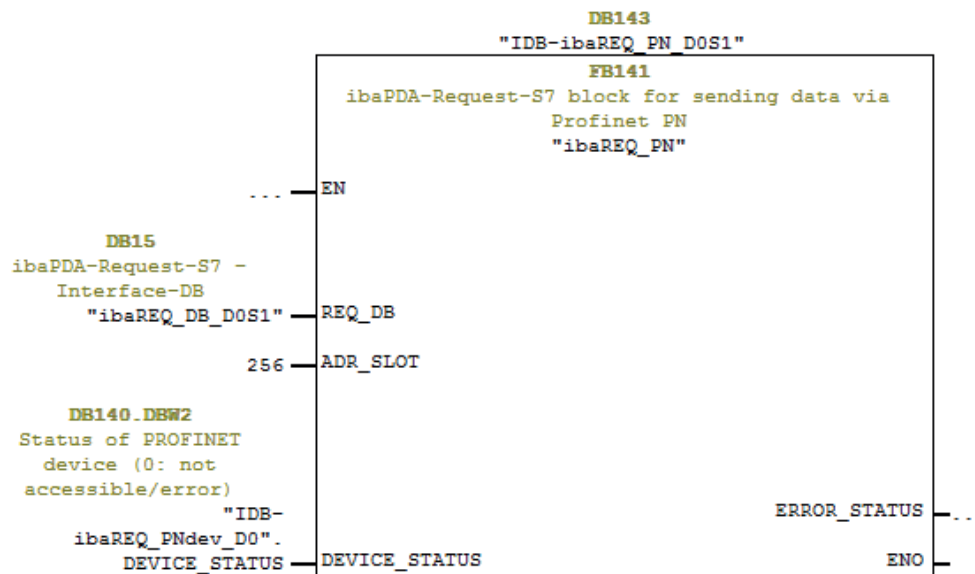
1. ibaREQ_M (FB140) vorzugsweise innerhalb des OB1 aufrufen.



2. Aktivieren Sie die Option *Remanenz* für den gesamten soeben angelegten Instanz-Datenbaustein.

Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a...	Sichtbar i...
1	Input					
2	RESET	Bool	0.0	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Output					
4	ERROR_STATUS	Word	2.0	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	InOut					
6	REQ_DB	*ibaREQ_DB-Interfa...	4.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Static					
8	EXPERT	Struct	10.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	sIDBinitialized	Bool	12.0	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	sOperandsInvalid	Bool	12.1	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	sIMDataValid	Bool	12.2	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	sOk	Bool	12.3	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	sGET_IM_DATA	Get_IM_Data			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	sIM_DATA	Array[0..53] of Byte	14.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	sAdrOPList	DInt	68.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	cBAddr_PDA2S7	Int	72.0	64	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	cBAddr_S72PDA	Int	74.0	1440	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	cBAddr_Xchange	Int	76.0	2464	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	cBAddr_Ops	Int	78.0	2528	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

3. ibaREQ_PN (FB141) innerhalb des OB1 oder eines Weckalarm-OB (OB3x) aufrufen.



Für jedes weitere Request-Modul

- Im Bausteinordner muss für jedes Request-Modul ein Datenbaustein ibaREQ_DB (DB15) vorhanden sein. Kopieren Sie den Datenbaustein und vergeben Sie eine neue eindeutige DB-Nummer.
- Innerhalb des OB1 muss für jedes Request-Modul ein weiterer Aufruf des ibaREQ_M (FB140) mit der neuen DB-Nummer (Eingang REQ_DB) erfolgen.
- Innerhalb des OB1 oder eines Weckalarm-OB (OB3x) muss für jedes Request-Modul ein weiterer Aufruf des ibaREQ_PN (FB141) mit der neuen DB-Nummer (Eingang REQ_DB) erfolgen.
- Beachten Sie, dass alle Instanz-Datenbausteine eindeutig sind und dass die Werte für den Parameter ADR_SLOT eindeutig vergeben sind.
- Der ibaREQ_PNdev muss nicht für jedes Request-Modul separat aufgerufen werden, sondern nur einmal pro genutztem PROFINET-Device.

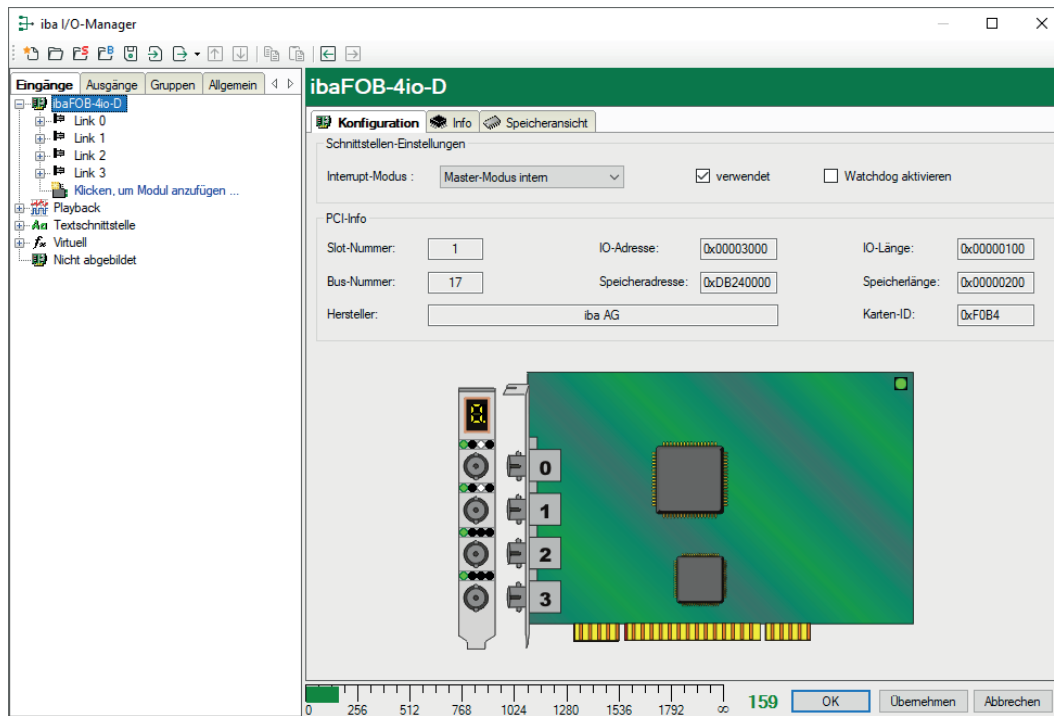
Abschluss

- Laden Sie alle Bausteine in die S7-CPU und starten Sie die S7-CPU neu.

2.1.3 Konfiguration und Projektierung ibaPDA

2.1.3.1 Allgemeine Einstellungen der Schnittstelle

Wenn eine *ibaFOB-D*-Karte im *ibaPDA*-Rechner installiert ist, bietet *ibaPDA* im Schnittstellenbaum des I/O-Managers die Schnittstelle für diese *ibaFOB-D*-Karte an.



Andere Dokumentation

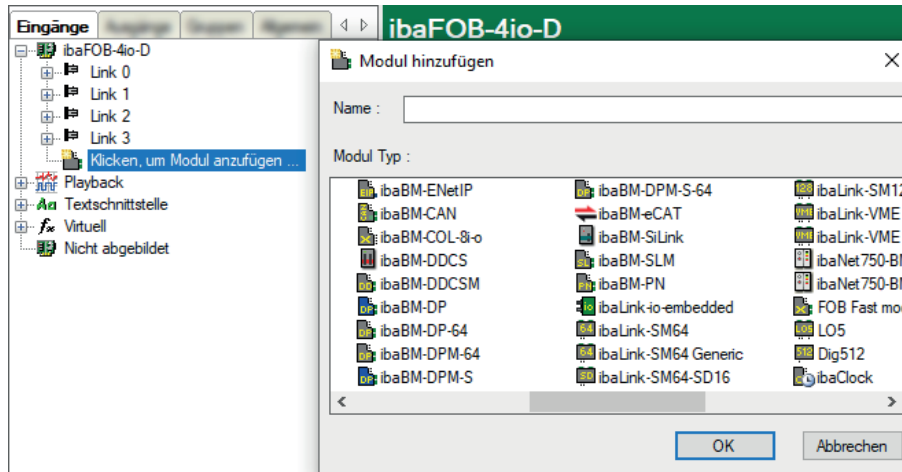


Detaillierte Informationen zur *ibaFOB-D*-Karte finden Sie im zugehörigen Gerätehandbuch.

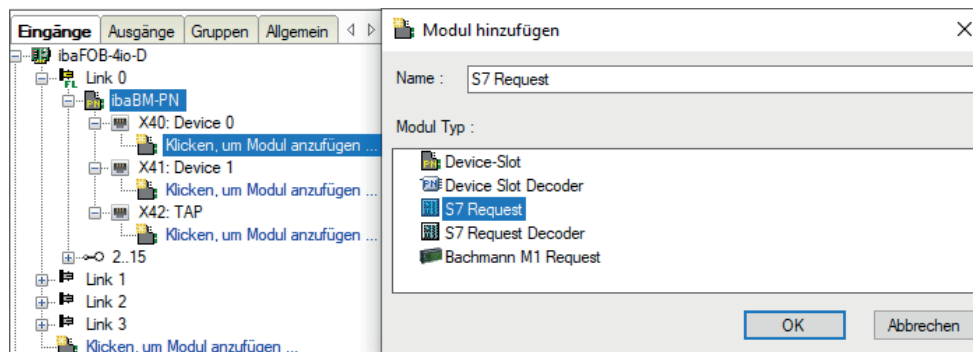
ibaBM-PN konfigurieren

1. Starten Sie den *ibaPDA* Client und öffnen Sie den I/O-Manager.
2. Markieren Sie im Schnittstellenbaum (links) den Link der *ibaFOB-D*-Karte, an dem *ibaBM-PN* angeschlossen ist.

Klicken Sie auf den blauen Befehl *Klicken, um Modul anzufügen*. Wählen Sie im Dialogfenster ein *ibaBM-PN*-Modul aus und vergeben Sie bei Bedarf einen Namen über das Eingabefeld. Bestätigen Sie Ihre Auswahl mit <OK>.



3. Fügen Sie zum *ibaBM-PN*-Modul am entsprechenden PROFINET-Anschluss Device 0 oder Device 1 ein Request-Modul hinzu (oder mehrere, falls Sie weitere Verbindungen zu einer bzw. zu verschiedenen S7-CPU's benötigen). Zur Auswahl stehen:
 - S7 Request (zum Erfassen von analogen und digitalen Signalen)
 - S7 Request Decoder (zum Erfassen von bis zu 1024 digitalen Signalen)



4. Nehmen Sie die erforderlichen Moduleinstellungen und Konfiguration der Signale vor, wie in den folgenden Kapiteln beschrieben.

Bei allen Request-Modulen sind die Register *Allgemein* und *Verbindung* identisch. Die Request-Module unterscheiden sich nur bei den Registern *Analog* bzw. *Digital*.

5. Wenn Sie die Konfiguration beendet haben, klicken Sie auf <Übernehmen> oder <OK>, um die neue Konfiguration ins Gerät zu übertragen und die Datenerfassung mit *ibaPDA* zu starten.

2.1.3.2 Allgemeine Moduleinstellungen

Die Beschreibung der für alle Request-S7-Module identischen allgemeinen Einstellungen finden Sie im allgemeinen Kapitel *Allgemeine Moduleinstellungen* in Teil 1 des Handbuchs.

Informationen zu den Verbindungseinstellungen im Register *Verbindung* finden Sie im allgemeinen Kapitel *Verbindungseinstellungen* in Teil 1 des Handbuchs.

Die *ibaBM-PN*-Module haben folgende spezifische Einstellmöglichkeiten im Register *Allgemein*:

PROFINET

Device

Nummer des PROFINET-Device, dem das Modul zugeordnet ist

Slot

Nummer des Slots, dem das Modul zugeordnet ist

S2 Redundanz

FALSE: Redundanzmodus deaktiviert. Weitere Informationen zum Redundanzmodus, siehe [↗ Request-S7 für ibaBM-PN im Redundanzmodus](#), Seite 21.

2.1.3.3 Modul S7 Request

Mit dem Modul *S7 Request* können Sie entsprechend der Größe des verwendeten PROFINET Slots Daten zu erfassen. Maximal sind bis zu 254 Bytes möglich.

Projektieren Sie für jedes Modul einen separaten PROFINET Slot und Request-Block-Aufruf.

Für weitere Informationen zur Moduleinstellung siehe Kapitel [↗ Allgemeine Moduleinstellungen](#), Seite 19, und das allgemeine Kapitel *Allgemeine Moduleinstellungen* in Handbuch Teil 1 beschrieben.

2.1.3.4 Modul S7 Request Decoder

Mit dem Modul *S7 Request Decoder* können Sie bis zu 16384 digitale Signale erfassen, die in Form von max. 128 Wörtern (16 Bit) gesendet werden. Dieser Modultyp eignet sich daher besonders für Anwendungen, bei denen sehr viele digitale Signale erfasst werden müssen und die max. 1024 direkt adressierbaren Digitalwerte des *ibaBM-PN* nicht ausreichen.

Projektieren Sie für jedes Modul einen separaten PROFINET Slot und Request-Block-Aufruf.

Register Allgemein

Für weitere Informationen zur Moduleinstellung siehe Kapitel [↗ Allgemeine Moduleinstellungen](#), Seite 19, und das allgemeine Kapitel *Allgemeine Moduleinstellungen* in Teil 1 des Handbuchs beschrieben.

Modulspezifische Einstellungen

Modul Struktur – Anzahl Decoder

Stellen Sie die Anzahl der konfigurierbaren Decoder in der digitalen Signaltabelle ein. Der Standardwert ist 32. Der Maximalwert beträgt 126. Die Signaltabelle wird entsprechend angepasst.

Verbindungskonfiguration

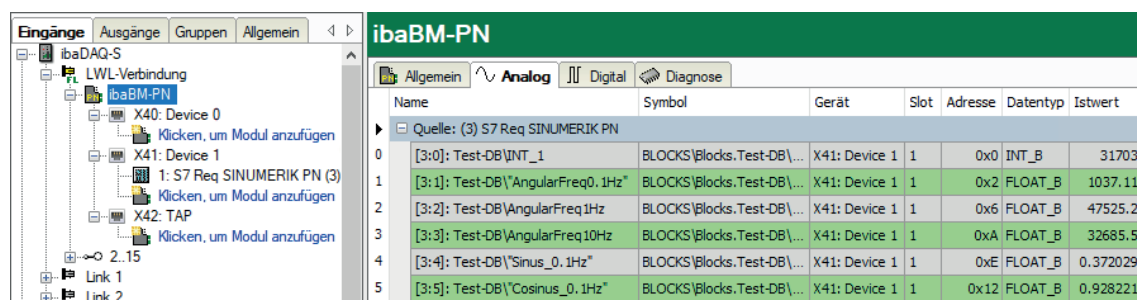
Die Verbindung des Moduls *S7 Request Decoder* konfigurieren Sie auf die gleiche Weise wie die Verbindung für ein S7-Request-Modul, siehe allgemeines Kapitel *Verbindungseinstellungen* in Teil 1 des Handbuchs.

Register Digital

Die Signale konfigurieren Sie auf die gleiche Weise wie für das Gerät *ibaBM-DP*, siehe Kapitel *Modul S7 Request Decoder* für *ibaBM-DP* in Teil 1 des Handbuchs.

2.1.4 Diagnose

Sie erhalten eine Auflistung aller im Busmodul erfassten Operanden mit Datentyp und Istwert, wenn Sie im Schnittstellenbaum den Busmodulknoten auswählen und das Register *Analog* bzw. *Digital* öffnen.



Name	Symbol	Gerät	Slot	Adresse	Datentyp	Istwert
Quelle: (3) S7 Req SINUMERIK PN						
[3:0]: Test-DB\INT_1	BLOCKS\Blocks.Test-DB\...	X41: Device 1	1	0x0	INT_B	31703
[3:1]: Test-DB\AngularFreq0.1Hz	BLOCKS\Blocks.Test-DB\...	X41: Device 1	1	0x2	FLOAT_B	1037.11
[3:2]: Test-DB\AngularFreq1Hz	BLOCKS\Blocks.Test-DB\...	X41: Device 1	1	0x6	FLOAT_B	47525.2
[3:3]: Test-DB\AngularFreq10Hz	BLOCKS\Blocks.Test-DB\...	X41: Device 1	1	0xA	FLOAT_B	32685.5
[3:4]: Test-DB\Sinus_0.1Hz	BLOCKS\Blocks.Test-DB\...	X41: Device 1	1	0xE	FLOAT_B	0.372029
[3:5]: Test-DB\Cosinus_0.1Hz	BLOCKS\Blocks.Test-DB\...	X41: Device 1	1	0x12	FLOAT_B	0.928221

Andere Dokumentation



Eine detaillierte Beschreibung der gerätespezifischen Diagnosemöglichkeiten des *ibaBM-PN* finden Sie im zugehörigen Gerätehandbuch.

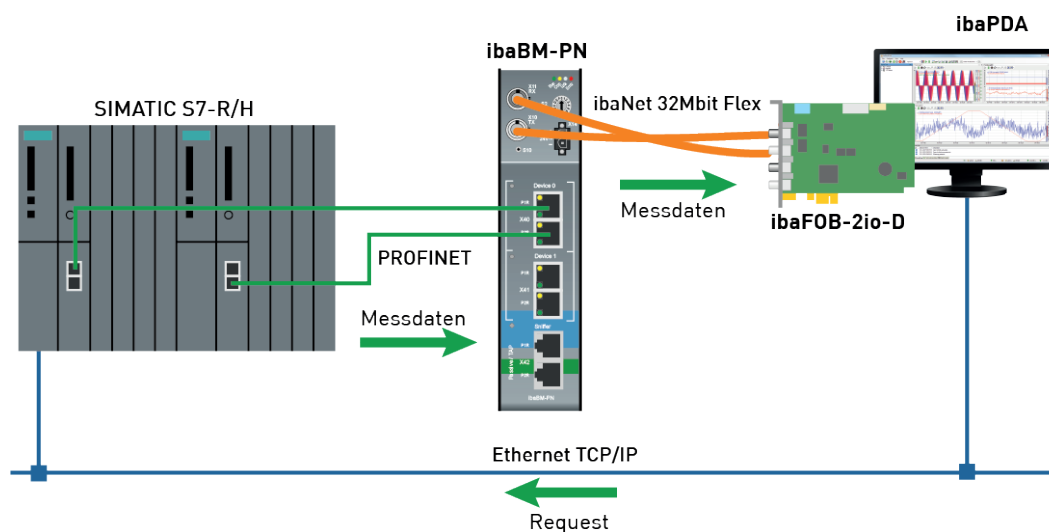
2.2 Request-S7 für ibaBM-PN im Redundanzmodus

Im Folgenden wird die Request-S7-Variante für das PROFINET-Busmodul *ibaBM-PN* im S2-Redundanzmodus beschrieben.

2.2.1 Allgemeine Informationen

Der S2-Redundanzmodus des *ibaBM-PN* ermöglicht den Betrieb an redundanten PROFINET-Systemen (S2-Systemredundanz) in Verbindung mit SIMATIC S7-R/H Steuerungen, deren Messdaten erfasst werden sollen. Um den S2-Redundanzmodus des *ibaBM-PN* verwenden zu können, benötigen Sie eine zusätzliche Lizenz. Wenden Sie sich hierfür an den iba-Support.

Die folgende Darstellung zeigt eine beispielhafte Einbindung eines *ibaBM-PN* im S2-Redundanzmodus:



Andere Dokumentation



Detaillierte Informationen zum S2-Redundanzmodus des *ibaBM-PN* finden Sie im Gerätehandbuch.

Die Funktionalität des Request-S7 für *ibaBM-PN* im Redundanzmodus entspricht weitestgehend der im Standardmodus (siehe Kapitel [Request-S7 für ibaBM-PN](#), Seite 11). Die Abweichungen und Erweiterungen sind nachfolgend beschrieben.

2.2.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

2.2.2.1 Projektierung Hardware

Pro genutztem PROFINET-Device müssen Sie ein PROFINET-Device projektieren.

Verwenden Sie die gleiche GSDML-Datei zu nutzen wie für den Normalbetrieb (ab Version V2.35-20200101).

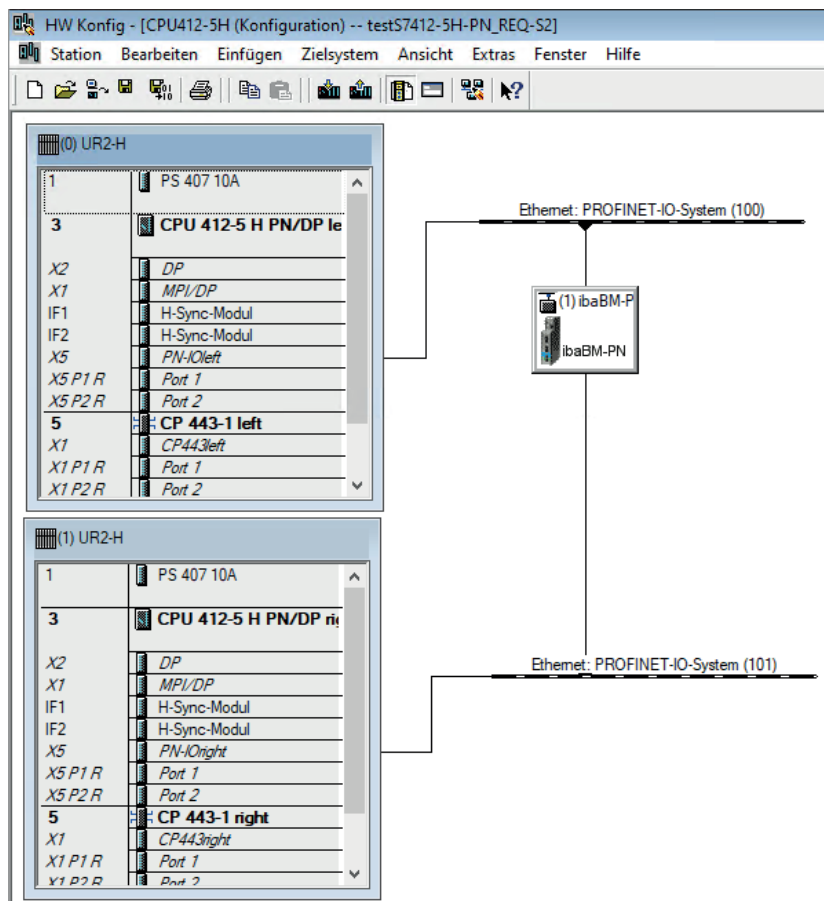
Hinweis



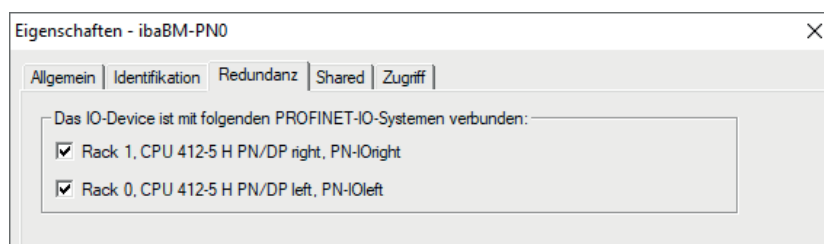
Die aktuelle GSDML-Datei finden Sie auf dem Datenträger "iba Software & Manuals" in folgendem Verzeichnis:

`\02_iba_Hardware\ibaBM-PN\01_GSD_Files\`

Wenn Sie ein *ibaBM-PN* in eine redundante Hardwarekonfiguration einfügen, wird dieser automatisch mit beiden PROFINET-IO-Systemen verbunden.



Diese Verbindung können Sie im Register *Redundanz* der Eigenschaften anpassen.



Wenden Sie die Hardwarekonfiguration an, um *ibaBM-PN* im S2-Redundanzmodus zu betreiben.

2.2.2.2 Projektierung in STEP 7

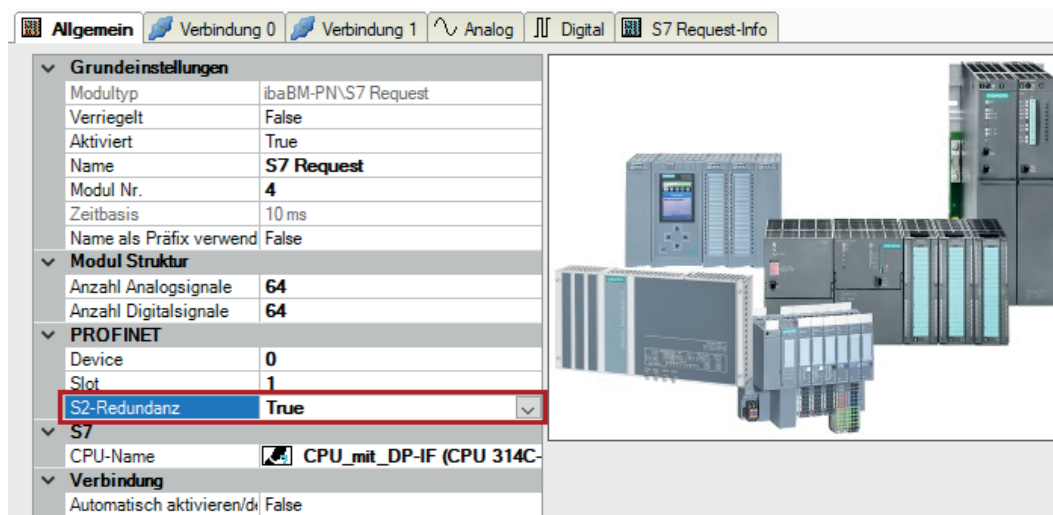
Die Projektierung der Request-Blöcke entspricht der Projektierung im Normalbetrieb (siehe Kapitel [↗ Projektierung in STEP 7](#), Seite 14).

2.2.3 Konfiguration und Projektierung ibaPDA

Schließen Sie zunächst *ibaBM-PN* an einen freien Link einer *ibaFOB-D*-Karte an. Im I/O-Manager fügen Sie an dem entsprechenden Link ein Gerätemodul *ibaBM-PN* ein.

Fügen Sie anschließend ein Modul *S7 Request* ein.

Im Register *Allgemein* setzen Sie den Parameter *S2 Redundanz* auf "True". Ein orange gefärbte Symbol des Request-Moduls zeigt den Redundanzmodus an.

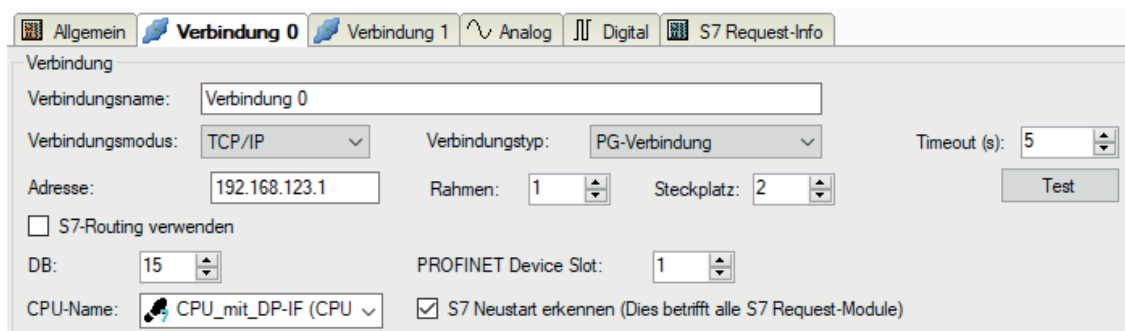


Die Konfiguration der Module entspricht ansonsten der im Standardmodus, siehe allgemeines Kapitel *Allgemeine Moduleinstellungen* in Teil 1 des Handbuchs.

Verbindungseinstellungen

Richten Sie für die Übertragung der Operandendaten zu den beiden CPUs der SIMATIC S7-R/H zwei getrennte Verbindungen ein. *ibaPDA* schaltet die Verbindung je nach Verfügbarkeit und Bedarf um.

Konfigurieren Sie beide Verbindungen in den Registern *Verbindung 0* und *Verbindung 1*. Den Name der Register können Sie über die Felder *Verbindungsname* ändern.



Die weiteren Einstellungen sind identisch zum Betrieb ohne Redundanzmodus, siehe allgemeines Kapitel *Verbindungseinstellungen* in Teil 1 des Handbuchs.

Berücksichtigen Sie folgende Besonderheiten:

- Die Rahmennummern bei einem H-System lauten 0 bzw. 1 für die beiden redundanten CPUs.
- Stellen Sie die DB-Nummer, die PROFINET-Device-Slots sowie den CPU-Name für die Zuweisung eines Adressbuchs nur einmal im Register *Verbindung 0* ein.

2.3 Request-S7 für ibaBM-DP

Im Folgenden wird die Request-S7-Variante für das PROFIBUS-Busmodul *ibaBM-DP* beschrieben.

2.3.1 Allgemeine Informationen

Request-S7 für *ibaBM-DP* kann in folgenden Systemkonfigurationen projektiert werden:

SIMATIC S7-CPU	SIMATIC STEP 7 V5.x (SIMATIC Manager)	SIMATIC STEP 7 V1x Professional (TIA Portal)
S7-300 integrierte DP-Schnittstelle	X	X
S7-400 integrierte DP-Schnittstelle und CP443-5	X	X
S7-400H integrierte DP-Schnittstelle und CP443-5	X	-
S7-1500 integrierte DP-Schnittstelle und CM1542-5 oder CP1542-5	-	X

Beim DP-Request erfolgt die Anforderung der Messwerte (Request-Handshake) nicht über den PROFIBUS, sondern über eine separate Verbindung.

Je nach vorhandener Hardware und Software stehen verschiedene Zugangspunkte zur Auswahl, über welche die Anforderung erfolgen kann:

- TCP/IP: Die Verbindung zur SIMATIC S7 wird über eine integrierte PN-Schnittstelle der S7-CPU oder entsprechende CP-Baugruppen in der SPS und der Standard-Netzwerkschnittstelle des Rechners hergestellt. Es ist keine weitere Siemens-Software für die Verbindung erforderlich.
- PC/CP: Hinter dieser Bezeichnung verbergen sich verschiedene Zugangspunkte, die SIMATIC-spezifisch sind. Im Gegensatz zur TCP/IP-Verbindung ist allen Verbindungsarten der Gruppe PC/CP ist gemeinsam, dass auf dem Rechner die SIMATIC-Kommunikationssoftware mit den entsprechenden Freischaltungen installiert sein muss.
 - MPI, PROFIBUS: Die Verbindung zur SIMATIC S7 wird über die MPI- bzw. PROFIBUS-Schnittstelle des Rechners hergestellt, z. B. mit der PCI-Karte CP5611 oder dem MPI-Adapter für USB-Schnittstellen oder serieller PC-Schnittstelle.
 - TCP/IP, ISO: Hier wird entweder die Standard-Netzwerkschnittstelle des Rechners oder eine geeignete Schnittstellenkarte für die Verbindung zur S7 verwendet.
- TCP/IP S7-1x00: Die Verbindung zu einer SIMATIC S7-1500 wird über eine integrierte PN-Schnittstelle der S7-CPU oder entsprechende CP-Baugruppen in der SPS und der Standard-Netzwerkschnittstelle des Rechners hergestellt. Es ist keine weitere Siemens-Software für die Verbindung erforderlich.

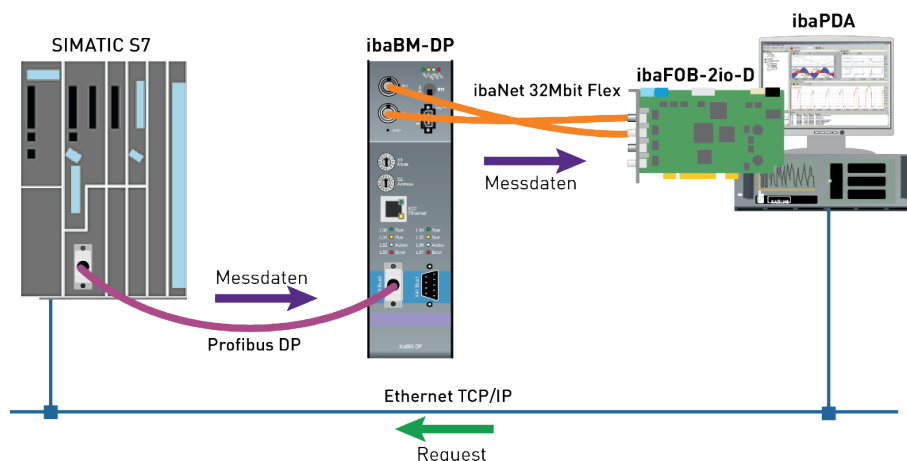
Systemintegration mit ibaBM-DP

Die Übertragung der Messdaten erfolgt über PROFIBUS DP an das *ibaBM-DP*-Gerät.

Sie benötigen insgesamt folgende Verbindungen:

- Onlineverbindung zwischen *ibaPDA* und S7-CPU (TCP/IP, MPI oder DP)
- Lichtwellenleiterverbindung zwischen *ibaPDA/ibaFOB-io-D* und *ibaBM-DP*
- PROFIBUS-Verbindung zwischen *ibaBM-DP* und S7 PROFIBUS-Master
- Eine Verbindung von *ibaBM-DP* an das Netzwerk (TCP/IP über Ethernet) wird nur benötigt, wenn das Gerät im Kompatibilitätsmodus, also nicht mit 32Mbit Flex (kurz: Flex), betrieben wird. In diesem Fall ist die Systemintegration wie mit *ibaBM-DPM-S*, siehe ➤ *Request-S7 für ibaBM-DPM-S*, Seite 46.

In der Standardausführung können Sie maximal 8 Verbindungen pro Gerät konfigurieren, d. h. 8 PROFIBUS-Slaves. Pro Slave ist die Übertragung von max. 244 Byte Daten möglich.



Hinweis



Die folgende Beschreibung bezieht sich auf die Request-Blöcke ab Version V4.0. Wenn Sie Erläuterungen zu älteren Versionen benötigen, wenden Sie sich an den iba-Support.

Andere Dokumentation

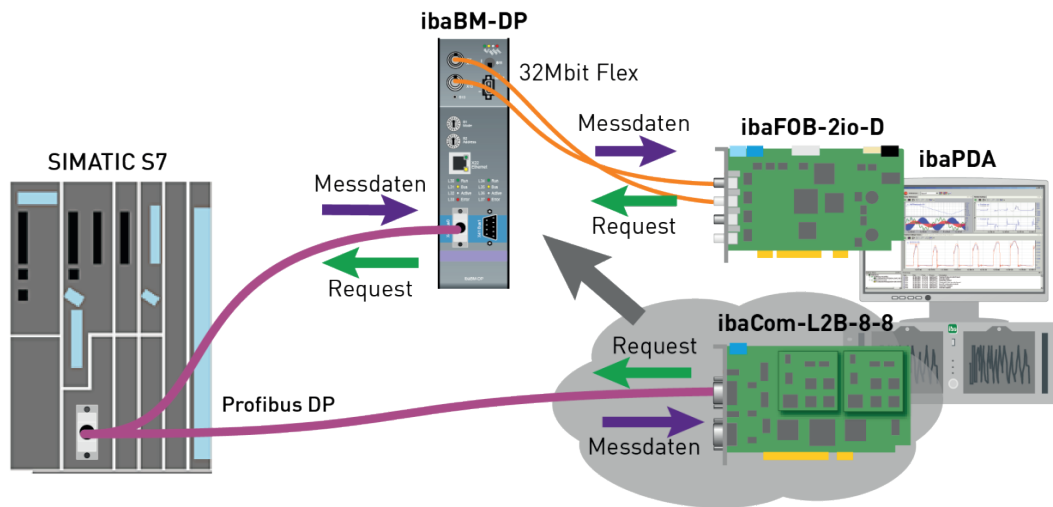


Detaillierte Informationen zu *ibaBM-DP* finden Sie im Gerätehandbuch.

Informationen zu Anwendungsbeispielen finden Sie in Kapitel ➤ *Anwendungsbeispiele*, Seite 95.

2.3.1.1 ibaCom-L2B Kompatibilitätsbetrieb

ibaBM-DP können Sie als voll funktionskompatiblen Nachfolger (ersatzteilkompatibel) für die Request-S7-Funktionalität der *ibaCom-L2B*-PROFIBUS-Karte einsetzen.



Verwenden Sie dafür folgende Module:

- S7 Request (*ibaCom-L2B kompatibel*)
- S7 Request Dig512 (*ibaCom-L2B kompatibel*)

Änderungen in der S7-Projektierung (Hardware oder S7-Programm) gegenüber der ursprünglichen Projektierung auf Basis der *ibaCom-L2B*-Karte sind nicht erforderlich.

Weitere Informationen dazu finden Sie unter [↗ Request-S7 für ibaCom-L2B](#), Seite 54 und [↗ Ablösung Request-S7 auf ibaCom-L2B durch ibaBM-DP](#), Seite 108.

Hinweis



Der *ibaCom-L2B* Kompatibilitätsbetrieb ist nur im Flex-Modus des *ibaBM-DP* nutzbar.

2.3.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

Im Folgenden wird die Konfiguration und Projektierung auf SIMATIC S7-Seite mit dem SIMATIC Manager (STEP 7 Version ≤ V5) bei Verwendung der Module *S7 Request* und *S7 Request Decoder* beschrieben.

Die Konfiguration und Projektierung bei Verwendung der Kompatibilitätsmodule *S7 Request (ibaCom-L2B kompatibel)* und *S7 Request Dig512 (ibaCom-L2B kompatibel)* entspricht der bei Verwendung einer *ibaCom-L2B*-Karte, siehe [↗ Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC](#), Seite 54.

2.3.2.1 Projektierung Hardware

Pro Modul müssen Sie einen iba-PROFIBUS-Slave definieren.

Verwenden Sie die GSD-Datei [ibaDPMSi.gsd](#) Version ab V2.2.

Hinweis



Die GSD-Datei [ibaDPMSi.gsd](#) finden Sie auf dem Datenträger "iba Software & Manuals" in folgendem Verzeichnis:

[\02_iba_Hardware\ibaBM-DP\02_GSD_Files\01_General\](#)

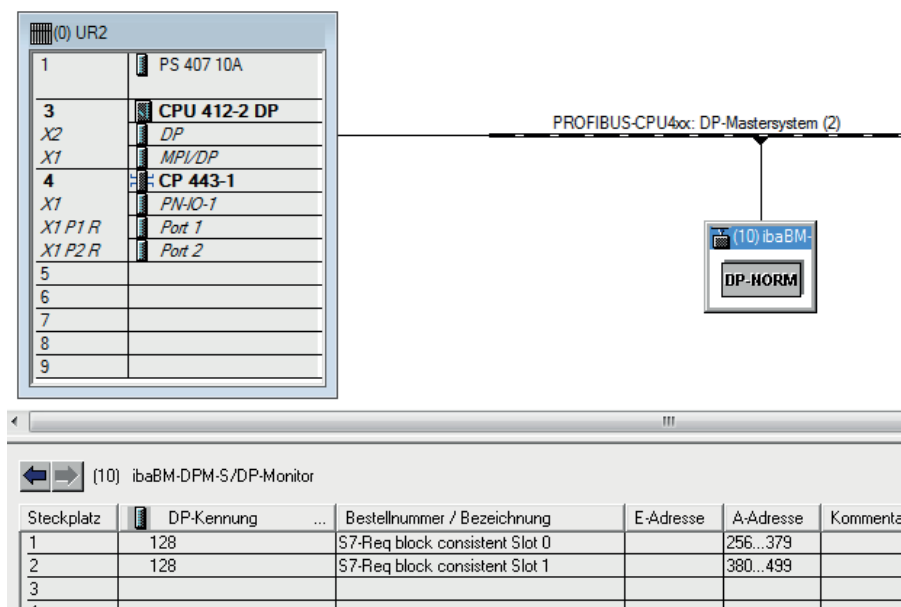
Hinweis



Die Slots müssen ohne Lücken und mit fortlaufendem Adressbereich angelegt werden.

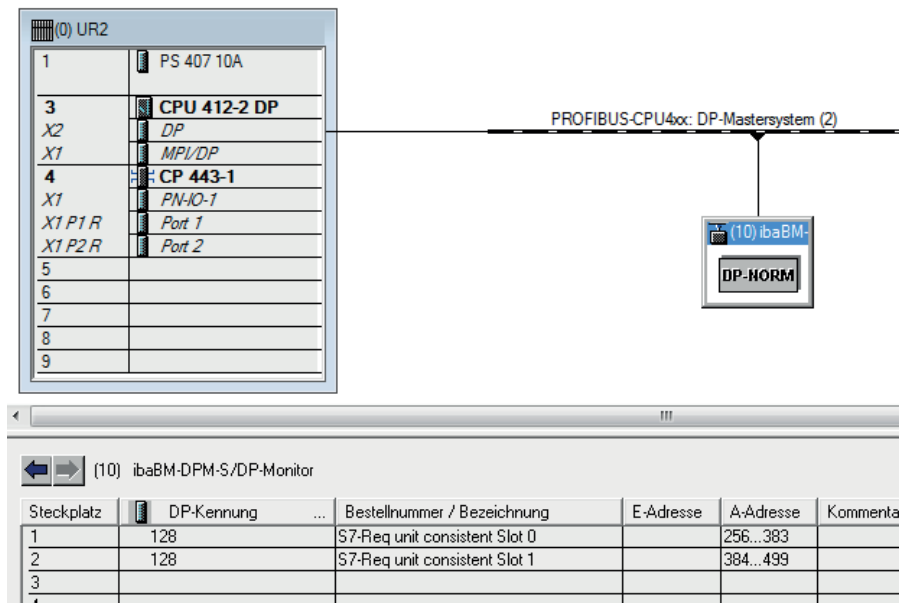
S7-CPU's neuerer Bauart¹⁾ ermöglichen konsistente Slots mit max. 128 Bytes.

Verwenden Sie die Elemente "S7-Req block consistent Slot 0 / Slot 1".



¹⁾ S7 neuerer Bauart sind im Allgemeinen: S7-400 ab Firmware-Version 3.0 und S7-300 ab Firmware-Version 2.0.

Bei **S7-CPUs und CP443-5 älterer Bauart** können Sie keine langen konsistenten Slots verwenden. Verwenden Sie die Elemente "S7-Req unit consistent Slot 0 / Slot 1"



2.3.2.2 Projektierung in STEP 7 (KOP, FUP, AWL)

Request-S7 ist für die Verwendung mit CPUs S7-300/400 mit integrierter DP-Schnittstelle oder CPUs S7-400 mit externer Schnittstelle CP 443-5 (PROFIBUS-Master) vorgesehen.

Hinweis



Die Verwendung der externen DP-Schnittstelle CP342-5 einer CPU S7-300 wird nicht unterstützt!

Für Request-S7 mit einem PROFIBUS-Slave

- Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe [iba S7-Bibliothek](#), Seite 92. Die Funktion ibaDP_Req muss einmal pro Slave im zyklischen Programm aufgerufen werden.
 - FC122 (ibaDP_Req), siehe [Request-FC ibaDP_Req \(FC122\)](#), Seite 74
 - DB10 (ibaDP_DB_PDA)
 - DB25 (ibaDP_DB_work)

Hinweis



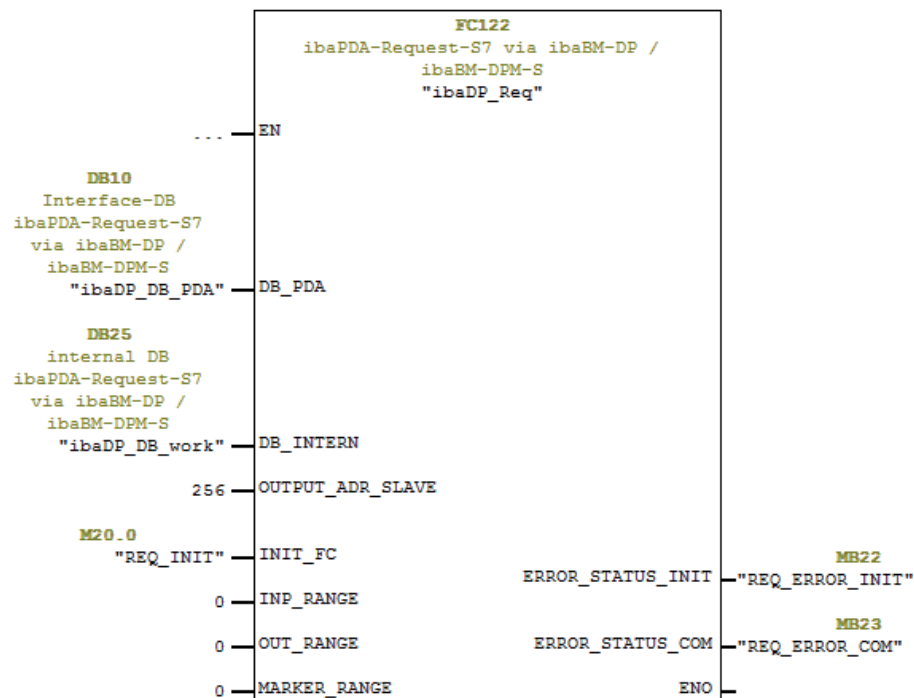
Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek! Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Hinweis



Falls die Bausteinnummern FC122, DB10 und DB25 in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

- Legen Sie die Fehler-OBs (OB82, OB85, OB86, OB87, OB122) an, um einen CPU-STOP im Fehlerfall zu verhindern.
- Im zyklischen Programm den ibaDP_Req (FC122) aufrufen und parametrieren.



Für Request-S7 mit weiteren PROFIBUS-Slaves

- Im Bausteinordner muss für jeden Slave ein Datenbaustein ibaDP_DB_PDA (DB10) vorhanden sein. Kopieren Sie den DB10 in einen DB mit einer neuen DB-Nummer.
- Im Bausteinordner muss für jeden Slave ein Datenbaustein ibaDP_DB_work (DB25) vorhanden sein. Kopieren Sie den DB25 in einen DB mit einer neuen DB-Nummer.
- Im zyklischen OB muss für jeden Slave ein weiterer Aufruf des ibaDP_Req (FC122) mit den neuen DB-Nummern und den Peripherieadressen des neuen PROFIBUS-Slaves erfolgen.
- Die Belegung des Eingangs INIT_FC und der Ausgänge ERROR_STATUS_INIT bzw. ERROR_STATUS_COM mit Merkern (oder DB-Elementen) sollte für jeden Slave eindeutig sein.

Abschluss

- Laden Sie alle Bausteine in die S7-CPU und starten Sie die S7-CPU neu.

2.3.2.3 Projektierung in STEP 7 (CFC)

Für Request-S7 mit einem PROFIBUS-Slave

- Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe [iba S7-Bibliothek](#), Seite 92. Die Funktion ibaDP_Req muss einmal pro Slave im zyklischen Programm aufgerufen werden.
 - FC122 (ibaDP_Req), siehe [Request-FC ibaDP_Req \(FC122\)](#), Seite 74
 - DB10 (ibaDP_DB_PDA)
 - DB25 (ibaDP_DB_work)

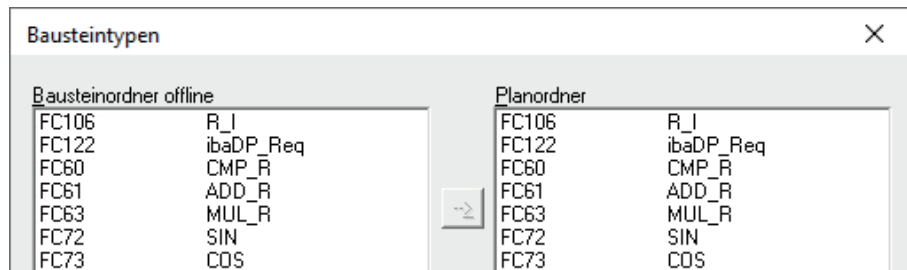
Hinweis

Falls die Bausteinnummern FC122, DB10 und DB25 in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

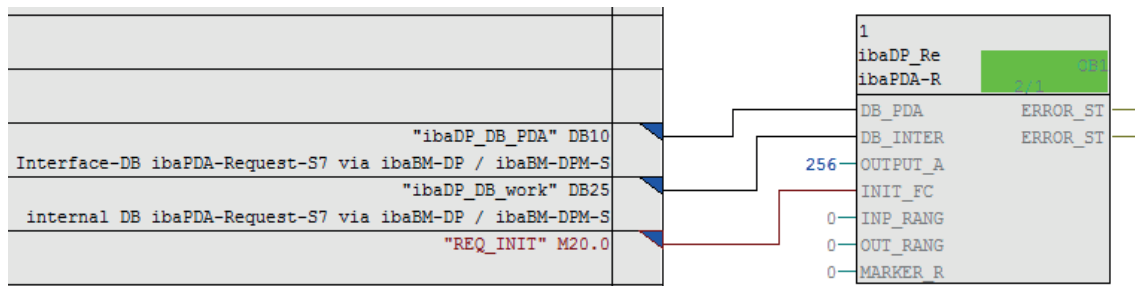
Die gewählten Bausteinnummern müssen zu den eingestellten, reservierten Bereichen für DBs und FCs unter den CFC-Einstellungen für *Übersetzen/Laden* passen.

- Starten Sie den CFC-Editor und importieren Sie den ibaDP_Req (FC122) (unter *Extras – Bausteintypen*).

Der FC wird in der Gruppe "ibaPDA" im Baustein-Katalog abgelegt.



- Ziehen Sie den ibaDP_Req (FC122) in Ihren CFC-Plan. Achten Sie auf die Ablaufreihenfolge. Der Baustein muss in einem zyklischen OB aufgerufen werden (Weckalarm-OB oder OB1).



Achten Sie darauf, dass die Fehler-OBs (OB82, OB85, OB86, OB87 und OB122) angelegt werden, sonst geht die CPU bei einem Zugriffsfehler auf STOP.

Dazu müssen Sie in der Ablaufreihenfolge in diesen Tasks, falls sie leer sind, jeweils eine (leere) Ablaufgruppe definieren. Alternativ können Sie auch einen beliebigen Dummy-Block im Fehler-OB platzieren, der ebenfalls nicht gelöscht werden darf.

Hinweis

Wählen Sie beim Kompilieren **nicht** die Option *Leere Ablaufgruppen löschen*, sonst werden die Fehler-OBs wieder entfernt! Bei einigen älteren CFC-Versionen ist diese Option beim Kompilieren auswählbar.

Für Request-S7 mit weiteren PROFIBUS-Slaves

- Im Bausteinordner muss für jeden Slave ein Datenbaustein ibaDP_DB_PDA (DB10) vorhanden sein. Kopieren Sie den DB10 in einen DB mit einer neuen DB-Nummer.
- Im Bausteinordner muss für jeden Slave ein Datenbaustein ibaDP_DB_work (DB25) vorhanden sein. Kopieren Sie den DB25 in einen DB mit einer neuen DB-Nummer.
- Im CFC-Plan müssen Sie für jeden Slave eine weitere Instanz des ibaDP_Req (FC122) anlegen. Parametrieren Sie die Konnektoren mit den neuen DB-Nummern und den Peripherieadressen des neuen PROFIBUS-Slaves.

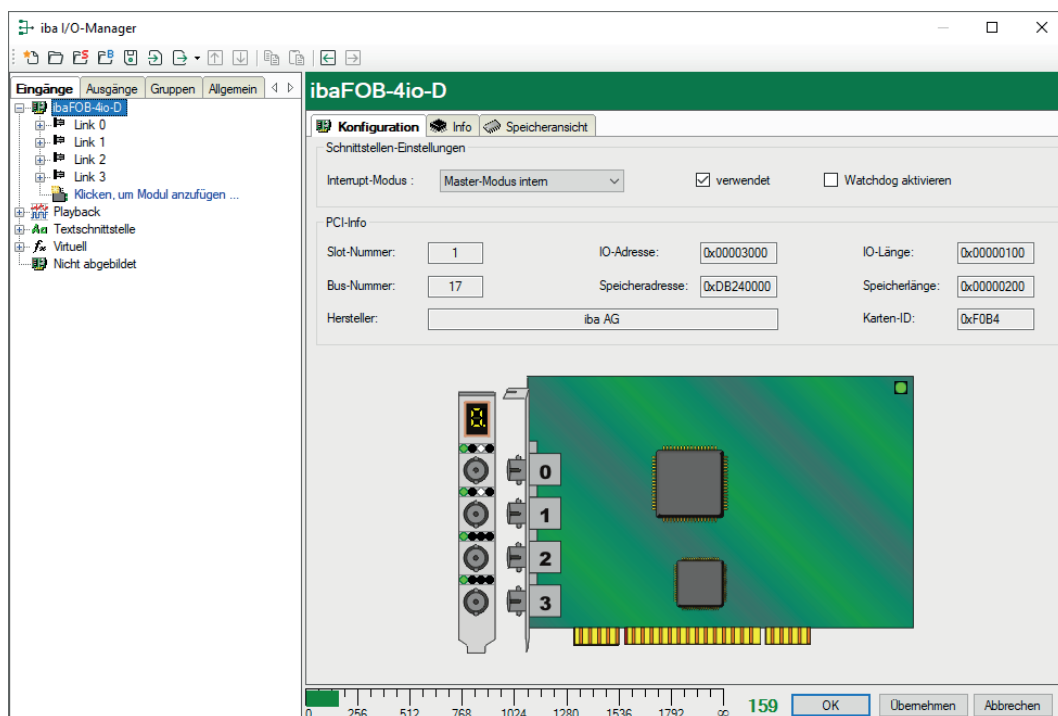
Nach jeder Programmänderung

1. Kompilieren Sie das komplette Programm:
 - a.) Aktivieren Sie die Option *Gesamtes Programm*.
 - b.) Aktivieren Sie die Option *SCL-Quelle erzeugen* (ab SIMATIC CFC Version 6.1).
 - c.) Deaktivieren Sie die Option *Leere Ablaufgruppen löschen* falls verfügbar.
2. Laden Sie das Programm und starten Sie die S7-CPU neu.

2.3.3 Konfiguration und Projektierung ibaPDA

2.3.3.1 Allgemeine Einstellungen der Schnittstelle

Wenn eine *ibaFOB-D*-Karte im *ibaPDA*-Rechner installiert ist, bietet *ibaPDA* im Schnittstellenbaum des I/O-Managers die Schnittstelle für diese *ibaFOB-D*-Karte an.



Andere Dokumentation

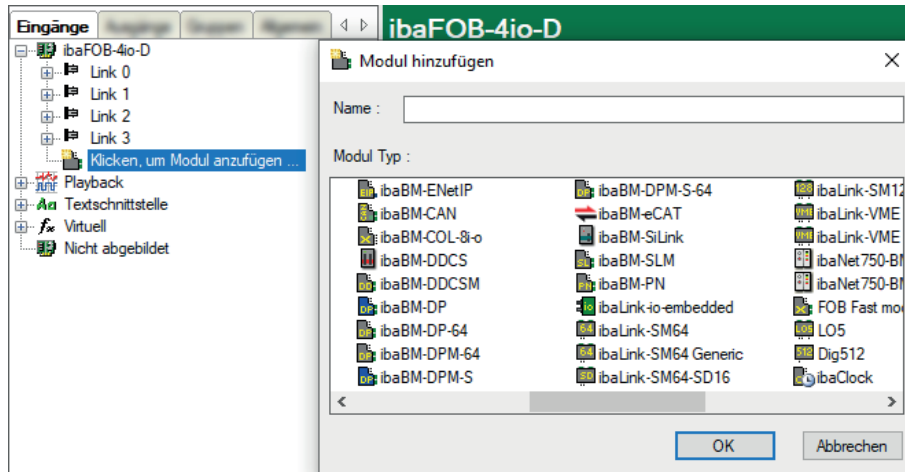


Detaillierte Informationen zur *ibaFOB-D*-Karte finden Sie im zugehörigen Gerätehandbuch.

ibaBM-DP konfigurieren

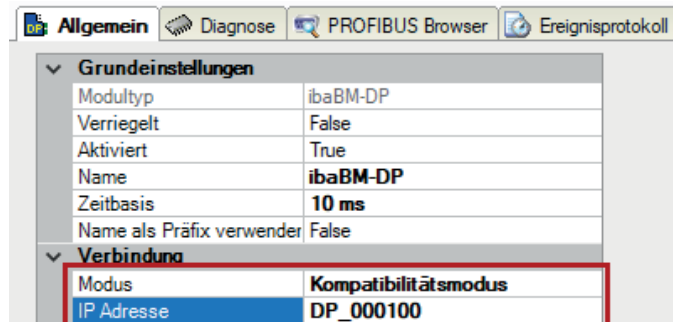
1. Starten Sie den *ibaPDA* Client und öffnen Sie den I/O-Manager.
2. Markieren Sie im Schnittstellenbaum (links) den Link der *ibaFOB-D*-Karte, an dem *ibaBM-DP* angeschlossen ist.

Klicken Sie auf den blauen Befehl *Klicken, um Modul anzufügen*. Wählen Sie im Dialogfenster ein *ibaBM-DP*-Modul aus und vergeben Sie bei Bedarf einen Namen über das Eingabefeld. Bestätigen Sie Ihre Auswahl mit <OK>.



3. Wenn Sie das Gerät im Flex-Modus betreiben, zeigt *ibaPDA* im Register *Allgemein* unter *Verbindung* die IP-Adresse des Geräts automatisch an.

Wenn Sie das Gerät im Kompatibilitätsmodus betreiben, tragen Sie im Register *Allgemein* unter *Verbindung* die IP-Adresse des Geräts ein: entweder als Namen z. B. "DP_000100" oder als IP-Adresse z. B. "192.168.81.123".



Hinweis

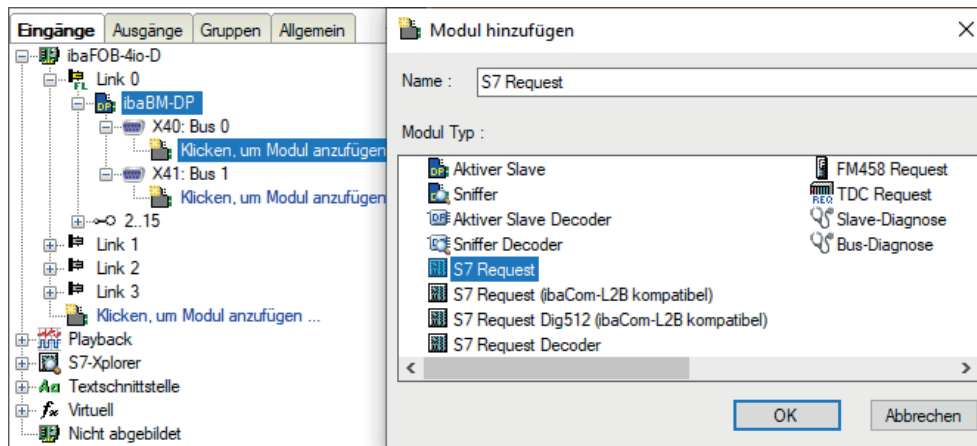


Im Kompatibilitätsmodus ist eine Onlineverbindung zu dem Gerät über Ethernet oder USB unbedingt notwendig. Wie Sie eine solche Verbindung herstellen und überprüfen können, finden Sie im Handbuch zum *ibaBM-DP*-Gerät.

Das Geräte-Handbuch beschreibt auch die Unterschiede zwischen Flex-Modus und Kompatibilitätsmodus näher.

4. Fügen Sie zum *ibaBM-DP*-Modul, am entsprechenden PROFIBUS-Anschluss Bus 0 oder Bus 1, ein Request-Modul hinzu (oder mehrere, falls Sie weitere Verbindungen zu einer bzw. zu verschiedenen S7-CPU's benötigen). Zur Auswahl stehen:

- S7 Request (zum Erfassen von analogen und digitalen Signalen)
- S7 Request Decoder (zum Erfassen von bis zu 1024 digitalen Signalen)



5. Nehmen Sie die erforderlichen Moduleinstellungen und Konfiguration der Signale vor, wie in den folgenden Kapiteln beschrieben.

Bei allen Request-Modulen sind die Register *Allgemein* und *Verbindung* identisch. Die Request-Module unterscheiden sich nur bei den Registern *Analog* bzw. *Digital*.

6. Wenn Sie die Konfiguration beendet haben, klicken Sie auf <Übernehmen> oder <OK>, um die neue Konfiguration ins Gerät zu übertragen und die Datenerfassung mit *ibaPDA* zu starten.

2.3.3.2 Allgemeine Moduleinstellungen

Die Beschreibung der für alle Request-S7-Module identischen allgemeinen Einstellungen finden Sie im allgemeinen Kapitel *Allgemeine Moduleinstellungen* in Teil 1 des Handbuchs.

Informationen zu den Verbindungseinstellungen im Register *Verbindung* finden Sie im allgemeinen Kapitel *Verbindungseinstellungen* in Teil 1 des Handbuchs.

Die *ibaBM-DP*-Module haben folgende spezifische Einstellmöglichkeiten im Register *Allgemein*:

PROFIBUS

Busnummer

0 = Stecker X40 links, 1 = Stecker X41 rechts

Slave Nummer

Dem Modul zugeordnete PROFIBUS-Slave-Adresse

Verbindung

Automatisch aktivieren/deaktivieren

Bei TRUE wird die Erfassung gestartet, auch wenn keine Verbindung zu der S7-CPU aufgebaut werden kann. Das Modul wird deaktiviert. Während der Messung versucht *ibaPDA* sich mit der S7-CPU zu verbinden. Bei Erfolg wird die Erfassung neu gestartet.

Bei FALSE wird die Erfassung nicht gestartet, falls keine Verbindung zur projektierten S7-CPU möglich ist.

2.3.3.3 Modul S7 Request

Mit dem Modul *S7 Request* können Sie bis zu 64 analoge und 64 digitale Signale erfassen.

Projektieren Sie für jedes Modul einen separaten PROFIBUS-Slave und Request-Block-Aufruf.

Die Moduleinstellungen sind im allgemeinen Kapitel *Allgemeine Moduleinstellungen* sowie in *Allgemeine Moduleinstellungen* für *ibaBM-DP* in Teil 1 des Handbuchs beschrieben.

2.3.3.4 Modul S7 Request Decoder

Mit dem Modul *S7 Request Decoder* können Sie bis zu 1024 digitale Signale erfassen, die in Form von max. 64 Wörtern (16 Bit) gesendet werden. Dieser Modultyp eignet sich daher besonders für Anwendungen, bei denen sehr viele digitale Signale erfasst werden müssen und die max. 1024 direkt adressierbaren Digitalwerte des *ibaBM-DP* nicht ausreichen.

Projektieren Sie für jedes Modul einen separaten PROFIBUS-Slave und Request-Block-Aufruf.

Register Allgemein

Für weitere Informationen zur Moduleinstellung siehe ➔ *Allgemeine Moduleinstellungen*, Seite 34, und das allgemeine Kapitel *Allgemeine Moduleinstellungen* in Teil 1 des Handbuchs beschrieben.

Verbindungskonfiguration

Die Verbindung des Moduls *S7 Request Decoder* konfigurieren Sie auf die gleiche Weise wie die Verbindung für ein S7-Request-Modul, siehe allgemeines Kapitel *Verbindungseinstellungen* in Teil 1 des Handbuchs.

Register Digital

Die Deklaration der Digitalsignale erfolgt auf zwei Ebenen.

- Definieren Sie zunächst die Wörter (Quellsignale), welche für die Digitalsignale (Bits) aufgeschlüsselt werden.

Die Wörter können Sie direkt als Basis-Signale für die Dekodierung über absolute S7-Operanden eintragen. Es sind nur Wort-Operanden (z. B. PEW, MW, DBW) erlaubt.

Ebenso können Sie S7-Symbole durch das Erzeugen von Adressbüchern verwenden. Für weitere Informationen siehe Kapitel *Auswahl über die symbolischen Operandenadressen* in Teil 1 des Handbuchs. Die im S7 CFC- und Symbol-Browser ausgewählten Signale werden übernommen und die Spalten *Name*, *S7 Symbol*, *S7 Operand* und *Datentyp* automatisch ausgefüllt.

- Jedes Wort (Quellsignal) können Sie über den Button <+> öffnen, um die Liste der zugehörigen Digitalsignalen anzuzeigen.

Definieren Sie danach die einzelnen Digitalsignale (Bits) des Wortes.

Decoder	S7 Operand	Datentyp	Aktiv
0 + PEW 1	PIW 1	WORD	<input checked="" type="checkbox"/>
1 + DB 3. DBW 2	DB 3.DBW 2	WORD	<input checked="" type="checkbox"/>
2 - DB 3. DBW 4	DB 3.DBW 4	WORD	<input checked="" type="checkbox"/>

Name	Aktiv
Digital Signal 0	<input checked="" type="checkbox"/>
Digital Signal 1	<input checked="" type="checkbox"/>
Digital Signal 2	<input checked="" type="checkbox"/>
Digital Signal 3	<input checked="" type="checkbox"/>
Digital Signal 4	<input checked="" type="checkbox"/>
Digital Signal 5	<input checked="" type="checkbox"/>
Digital Signal 6	<input checked="" type="checkbox"/>
Digital Signal 7	<input checked="" type="checkbox"/>
Digital Signal 8	<input checked="" type="checkbox"/>
Digital Signal 9	<input checked="" type="checkbox"/>
Digital Signal 10	<input checked="" type="checkbox"/>
Digital Signal 11	<input checked="" type="checkbox"/>
Digital Signal 12	<input checked="" type="checkbox"/>
Digital Signal 13	<input checked="" type="checkbox"/>
Digital Signal 14	<input checked="" type="checkbox"/>
Digital Signal 15	<input checked="" type="checkbox"/>

3 + DB 3. DBW 6	DB 3.DBW 6	WORD	<input checked="" type="checkbox"/>
4 +		WORD	<input type="checkbox"/>

Die einzelnen Spalten der Signaltabelle haben folgende Bedeutungen.

Quellsignal

Decoder

Tragen Sie einen Namen für das Quellsignal ein.

S7 Operand/S7 Symbol

Tragen Sie den S7 Operand und ggf. das S7 Symbol ein, dem das Signal zugeordnet ist.

Datentyp

Geben Sie den Datentyp des Signals an. Der Datentyp bestimmt auch die Anzahl der Digitalsignale. *ibaPDA* leitet den möglichen Datentyp automatisch vom S7-Operand bzw. S7-Symbol ab.

Aktiv

Wenn Sie das Quellsignal aktivieren, wird es mit allen Digitalsignalen erfasst. Sie können einzelne Digitalsignale abwählen.

Einzelne Digitalsignale (Bits)

Name

Tragen Sie einen Namen für die einzelnen Digitalsignale ein.

Aktiv

Wenn Sie das Digitalsignal aktivieren, wird das Signal erfasst und auch in der Prüfung der Anzahl der lizenzierten Signale berücksichtigt.

Hinweis

ibaPDA berücksichtigt jeweils nur die aktivierten Digitalsignale bei der Anzahl der lizenzierten Signale, also kein zusätzliches Signal für das Quellsignal.

2.3.3.5 Modul S7 Request (*ibaCom-L2B kompatibel*)

Das Modul *S7 Request (ibaCom-L2B kompatibel)* ist voll funktionskompatibel zu dem Modul *L2B S7 Request* bei Verwendung einer *ibaCom-L2B*-Karte, siehe [↗ Konfiguration und Projektierung ibaPDA](#), Seite 63.

Nehmen Sie die Konfiguration und Projektierung der SIMATIC S7-Seite entsprechend Kapitel [↗ Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC](#), Seite 54 vor.

Hinweis

Dieser Modultyp wird in der Regel dazu verwendet, eine bestehende auf einer *ibaCom-L2B*-Karte basierende Request-S7-Lösung durch ein *ibaBM-DP*-Gerät abzulösen. Weitere Hinweise zu diesem Anwendungsfall finden Sie im Kapitel [↗ Ablösung Request-S7 auf ibaCom-L2B durch ibaBM-DP](#), Seite 108.

Hinweis

Das Modul *S7 Request (ibaCom-L2B kompatibel)* ist nicht für Anbindung von Steuerungen des Typs S7-1500 freigegeben.

2.3.3.6 Modul S7 Request Dig512 (*ibaCom-L2B kompatibel*)

Das Modul *S7 Request Dig512 (ibaCom-L2B kompatibel)* ist voll funktionskompatibel zu dem Modul *L2B S7 Request Dig512* bei Verwendung einer *ibaCom-L2B*-Karte, siehe [↗ Konfiguration und Projektierung ibaPDA](#), Seite 63.

Die Konfiguration und Projektierung der SIMATIC S7-Seite ist entsprechend Kapitel [↗ Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC](#), Seite 54 vorzunehmen.

Hinweis

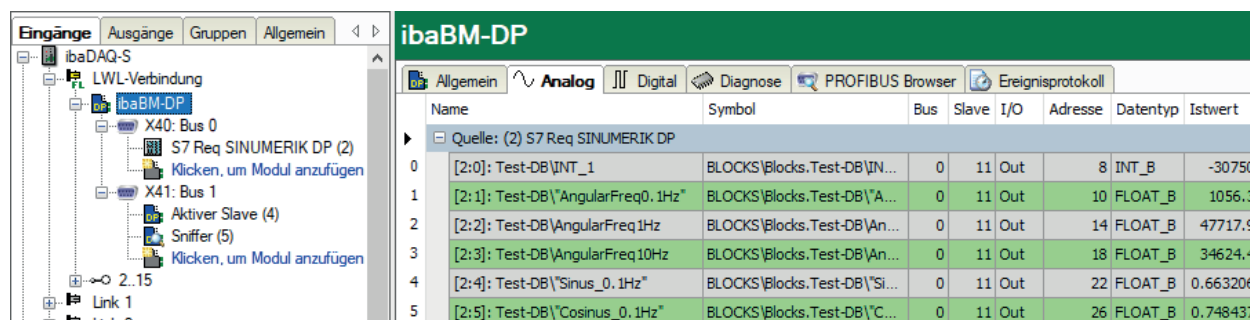
Dieser Modultyp wird in der Regel dazu verwendet, eine bestehende auf einer *ibaCom-L2B*-Karte basierende Request-S7-Lösung durch ein *ibaBM-DP*-Gerät abzulösen. Weitere Hinweise zu diesem Anwendungsfall finden Sie im Kapitel [↗ Ablösung Request-S7 auf ibaCom-L2B durch ibaBM-DP](#), Seite 108.

Hinweis

Das Modul *S7 Request Dig512 (ibaCom-L2B kompatibel)* ist nicht für Anbindung von Steuerungen des Typs S7-1500 freigegeben.

2.3.4 Diagnose

Sie erhalten eine Auflistung aller im Busmodul erfassten Operanden mit Datentyp und Istwert, wenn Sie im Schnittstellenbaum den Busmodulknoten auswählen und das Register *Analog* bzw. *Digital* öffnen.



Name	Symbol	Bus	Slave	I/O	Adresse	Datentyp	Istwert
Quelle: (2) S7 Req SINUMERIK DP							
0 [2:0]: Test-DB\INT_1	BLOCKS\Blocks.Test-DB\IN...	0	11	Out	8	INT_B	-30750
1 [2:1]: Test-DB\AngularFreq0.1Hz	BLOCKS\Blocks.Test-DB\A...	0	11	Out	10	FLOAT_B	1056.3
2 [2:2]: Test-DB\AngularFreq1Hz	BLOCKS\Blocks.Test-DB\An...	0	11	Out	14	FLOAT_B	47717.9
3 [2:3]: Test-DB\AngularFreq10Hz	BLOCKS\Blocks.Test-DB\An...	0	11	Out	18	FLOAT_B	34624.4
4 [2:4]: Test-DB\Sinus_0.1Hz	BLOCKS\Blocks.Test-DB\Si...	0	11	Out	22	FLOAT_B	0.663206
5 [2:5]: Test-DB\Cosinus_0.1Hz	BLOCKS\Blocks.Test-DB\C...	0	11	Out	26	FLOAT_B	0.748437

Andere Dokumentation



Eine detaillierte Beschreibung der gerätespezifischen Diagnosemöglichkeiten des *ibaBM-DP* finden Sie im zugehörigen Gerätehandbuch.

2.4 Request-S7 für ibaBM-DP im Redundanzmodus

Im Folgenden wird die Request-S7-Variante für das PROFIBUS-Busmodul *ibaBM-DP* im Redundanzmodus beschrieben.

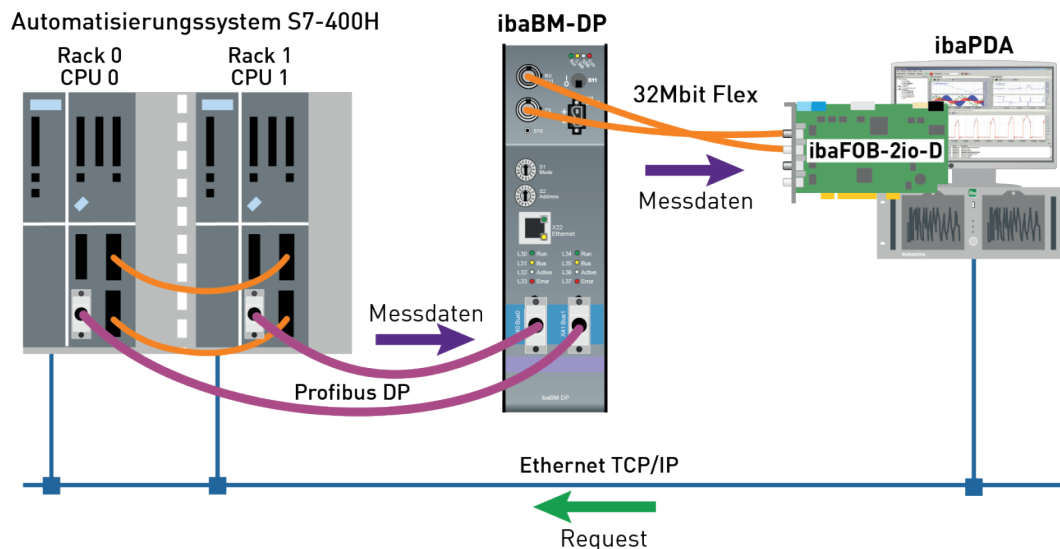
2.4.1 Allgemeine Informationen

Der Redundanzmodus des *ibaBM-DP* ermöglicht den Betrieb an redundanten PROFIBUS-Systemen in Verbindung mit SIMATIC S7-400H Steuerungen, deren Messdaten erfasst werden sollen.

Um den Redundanzmodus des *ibaBM-DP* verwenden zu können, benötigen Sie eine zusätzliche Lizenz. Wenden Sie sich hierfür an den Support der iba AG. Die Lizenz wird über die Administratorfunktionen im Web-Dialog des *ibaBM-DP* freigeschaltet.

Request-S7 redundant ist geeignet, um *ibaBM-DP* mit Request-Funktionalität als einkanalig geschaltete Peripherie an einer hochverfügbaren SIMATIC S7-400H Steuerung zu betreiben.

Die folgende Darstellung zeigt eine beispielhafte Einbindung eines *ibaBM-DP* im Redundanzmodus:



Hinweis



Die folgende Beschreibung bezieht sich auf die Request-Blöcke ab Version V4.0.

Andere Dokumentation



Detaillierte Informationen zum Redundanzmodus des *ibaBM-DP* finden Sie im Gerätehandbuch.

Die Funktionalität des Request-S7 für *ibaBM-DP* im Redundanzmodus entspricht weitestgehend der im Standardmodus (siehe Kapitel [Request-S7 für ibaBM-DP](#), Seite 25). Die Abweichungen und Erweiterungen sind nachfolgend beschrieben.

Informationen zu Anwendungsbeispielen finden Sie in Kapitel [Anwendungsbeispiele](#), Seite 95.

2.4.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

2.4.2.1 Projektierung Hardware

Pro PROFIBUS-Master-System müssen Sie einen separaten ibaBM-PROFIBUS-Slave konfigurieren.

Verwenden Sie die GSD-Datei [ibaDPMSi.gsd](#) Version ab V2.2.

Hinweis



Die GSD-Datei [ibaDPMSi.gsd](#) finden Sie auf dem Datenträger "iba Software & Manuals" in folgendem Verzeichnis:

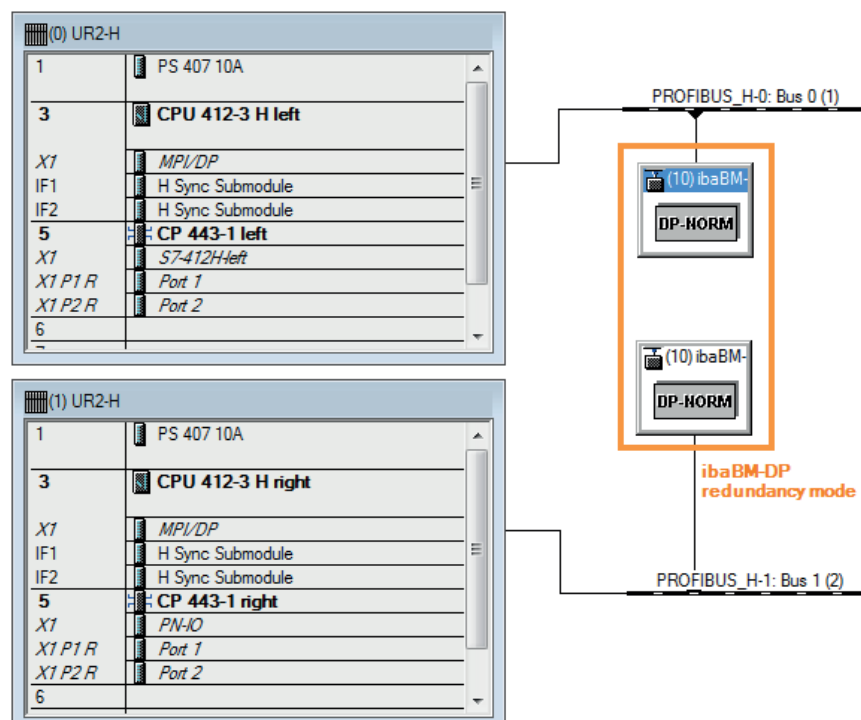
`\02_iba_Hardware\ibaBM-DP\02_GSD_Files\01_General\`

Hinweis



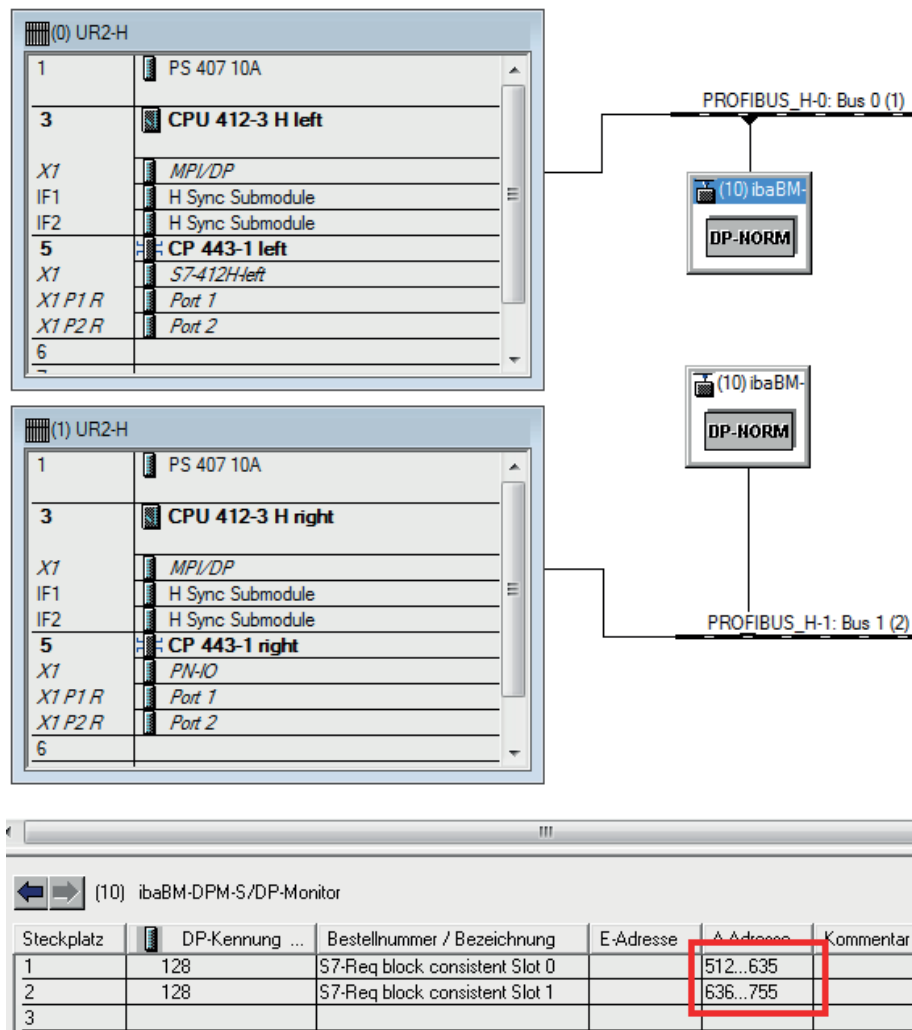
Die Module müssen pro Slave ohne Lücken und mit fortlaufenden Anfangsadressen angelegt werden.

Die nachfolgende Abbildung zeigt eine beispielhafte Hardware-Konfiguration für ein redundantes System.

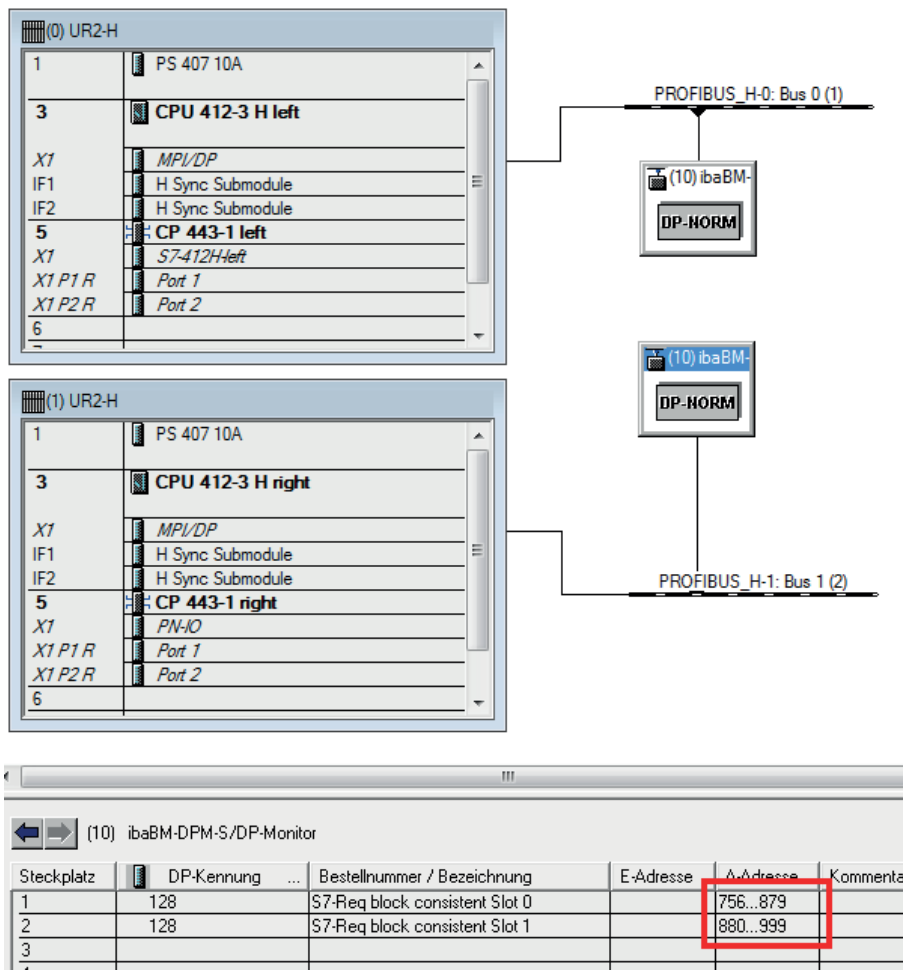


Projektieren Sie beide projektierten Slaves mit identischen Busadressen. Die I/O-Adressbereiche sind jedoch unterschiedlich. Verwenden Sie die Module "S7-Req block consistent Slot 0 / Slot 1".

Die nachfolgende Abbildung zeigt die Slave-I/O-Adressen für Bus 0.



Die nachfolgende Abbildung zeigt die Slave-I/O-Adressen für Bus 1.



2.4.2.2 Projektierung in STEP 7 (KOP, FUP, AWL)

Request-S7 redundant ist für die Verwendung mit CPUs S7-400H sowohl mit integrierter DP-Schnittstelle als auch mit externer Schnittstelle CP 443-5 (PROFIBUS-Master) vorgesehen.

Für Request-S7 mit einem PROFIBUS-Slave

- Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe [iba S7-Bibliothek](#), Seite 92. Die Funktion ibaDP_Req_H muss einmal pro Slave-Paar im zyklischen Programm aufgerufen werden.
 - FC123 (ibaDP_Req_H), siehe [Request-FC ibaDP_Req_H \(FC123\)](#), Seite 76
 - DB10 (ibaDP_DB_PDA)
 - DB25 (ibaDP_DB_work)

Hinweis

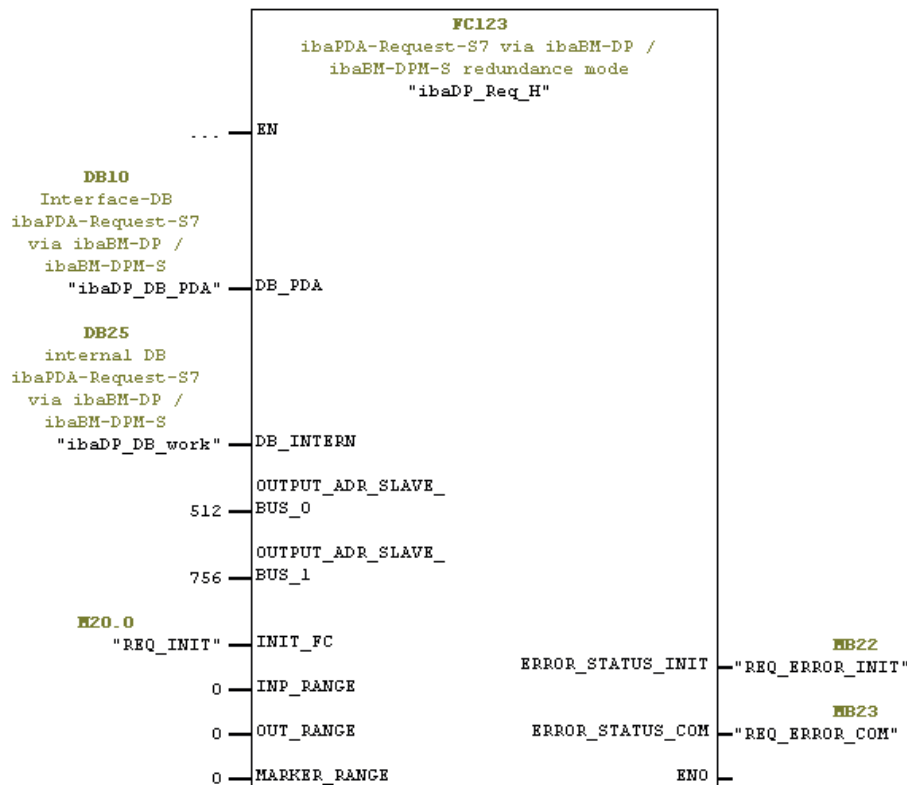


Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek! Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Hinweis

Falls die Bausteinnummern FC123, DB10 und DB25 in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

2. Legen Sie die Fehler-OBs (OB82, OB85, OB86, OB87, OB122) an, um einen CPU-STOP im Fehlerfall zu verhindern.
3. Im zyklischen Programm den ibaDP_Req_H (FC123) aufrufen und parametrieren.

**Für Request-S7 mit weiteren PROFIBUS-Slaves**

- Im Bausteinordner muss für jedes Slave-Paar ein Datenbaustein ibaDP_DB_PDA (DB10) vorhanden sein. Kopieren Sie den DB10 in einen DB mit einer neuen DB-Nummer.
- Im Bausteinordner muss für jedes Slave-Paar ein Datenbaustein ibaDP_DB_work (DB25) vorhanden sein. Kopieren Sie den DB25 in einen DB mit einer neuen DB-Nummer.
- Im zyklischen Programm muss für jedes Slave-Paar ein weiterer Aufruf des ibaDP_Req_H (FC123) mit den neuen DB-Nummern und den jeweiligen Peripherieadressen des neuen PROFIBUS-Slave-Paares erfolgen.
- Die Belegung des Eingangs INIT_FC und der Ausgänge ERROR_STATUS_INIT bzw. ERROR_STATUS_COM mit Merkern (oder DB-Elementen) sollte für jedes Slave-Paar eindeutig sein.

Abschluss

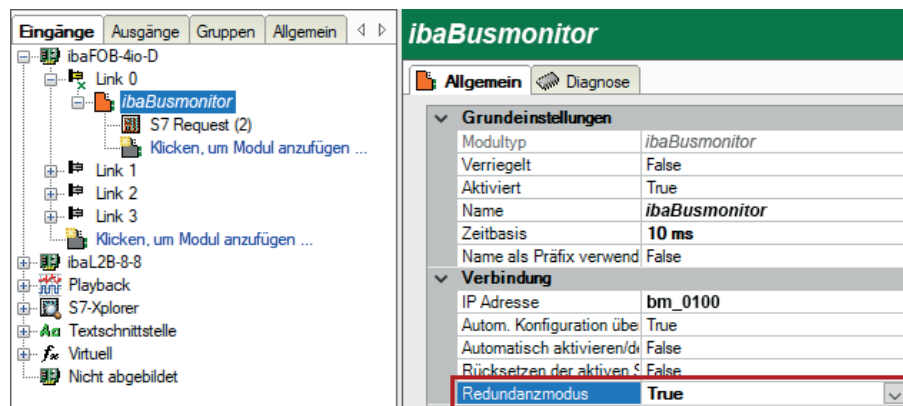
- Laden Sie alle Bausteine in die S7-CPU und starten Sie die S7-CPU neu.

2.4.3 Konfiguration und Projektierung ibaPDA

Schließen Sie zunächst *ibaBM-DP* an einen freien Link einer *ibaFOB-D*-Karte an. Im I/O-Manager fügen Sie an dem entsprechenden Link ein Gerätemodul *ibaBM-DP* ein.

Im Register *Allgemein* setzen Sie den Parameter *Redundanzmodus* auf "True". Ein orange gefärbte Symbol des Gerätemoduls zeigt den Redundanzmodus an.

Sie können das Gerät kann im Redundanzmodus sowohl mit 32Mbit Flex als auch im Kompatibilitätsmodus mit 32Mbit betreiben.



Da beim redundanten PROFIBUS beide Bussysteme 0 und 1 parallel betrieben werden, gibt es im Register *Allgemein* nur noch Einstellungen für *Redundanter PROFIBUS*.

Im Redundanzmodus stehen mit Request-S7 zusätzlich folgende Module zur Verfügung:

- S7 Request
- S7 Request Decoder

Hinweis



Nicht im Redundanzmodus einsetzbar sind folgende Module:

- Modul S7 Request (ibaCom-L2B kompatibel)
- Modul S7 Request Dig512 (ibaCom-L2B kompatibel)

Diese Module werden automatisch deaktiviert und können auch nicht manuell aktiviert werden.

Die Konfiguration der Module entspricht der im Standardmodus, siehe ➔ *Allgemeine Moduleinstellungen*, Seite 34.

Verbindungseinstellungen

Richten Sie für die Übertragung der Operandendaten zu den beiden CPUs der SIMATIC S7-400H zwei getrennte Verbindungen ein. *ibaPDA* schaltet die Verbindung je nach Verfügbarkeit und Bedarf um.

Konfigurieren Sie beide Verbindungen in den Registern *Verbindung 0* und *Verbindung 1*. Den Name der Register können Sie über die Felder *Verbindungsname* ändern.

Die Konfiguration der Module entspricht ansonsten der im Standardmodus, siehe allgemeines Kapitel *Verbindungseinstellungen* in Teil 1 des Handbuchs.

Berücksichtigen Sie folgende Besonderheiten:

- Die Rahmennummern bei einem H-System lauten 0 bzw. 1 für die beiden redundanten CPUs.
- Stellen Sie die DB-Nummer, die PROFIBUS-Slave-Nummer sowie den CPU-Name für die Zuweisung eines Adressbuchs nur einmal im Register *Verbindung 0* ein.

2.5 Request-S7 für ibaBM-DPM-S

Im Folgenden wird die Request-S7-Variante für das PROFIBUS-Busmodul *ibaBM-DPM-S* beschrieben.

Die Lösung Request-S7 für *ibaBM-DPM-S* wird durch Request-S7 für *ibaBM-DP* funktionskompatibel abgelöst.

2.5.1 Allgemeine Informationen

Beim DP-Request erfolgt die Anforderung der Messwerte (Request-Handshake) nicht über den PROFIBUS, sondern über eine separate Verbindung.

Je nach vorhandener Hardware und Software stehen verschiedene Zugangspunkte zur Auswahl, über welche die Anforderung erfolgen kann:

- TCP/IP: Die Verbindung zur SIMATIC S7 wird über eine integrierte PN-Schnittstelle der S7-CPU oder entsprechende CP-Baugruppen in der SPS und der Standard-Netzwerkschnittstelle des Rechners hergestellt. Es ist keine weitere Siemens-Software für die Verbindung erforderlich.
- PC/CP: Hinter dieser Bezeichnung verbergen sich verschiedene Zugangspunkte, die SIMATIC-spezifisch sind. Im Gegensatz zur TCP/IP-Verbindung ist allen Verbindungsarten der Gruppe PC/CP ist gemeinsam, dass auf dem Rechner die SIMATIC-Kommunikationssoftware mit den entsprechenden Freischaltungen installiert sein muss.
 - MPI, PROFIBUS: Die Verbindung zur SIMATIC S7 wird über die MPI- bzw. PROFIBUS-Schnittstelle des Rechners hergestellt, z. B. mit der PCI-Karte CP5611 oder dem MPI-Adapter für USB-Schnittstellen oder serieller PC-Schnittstelle.
 - TCP/IP, ISO: Hier wird entweder die Standard-Netzwerkschnittstelle des Rechners oder eine geeignete Schnittstellenkarte für die Verbindung zur S7 verwendet.

Systemintegration mit ibaBM-DPM-S

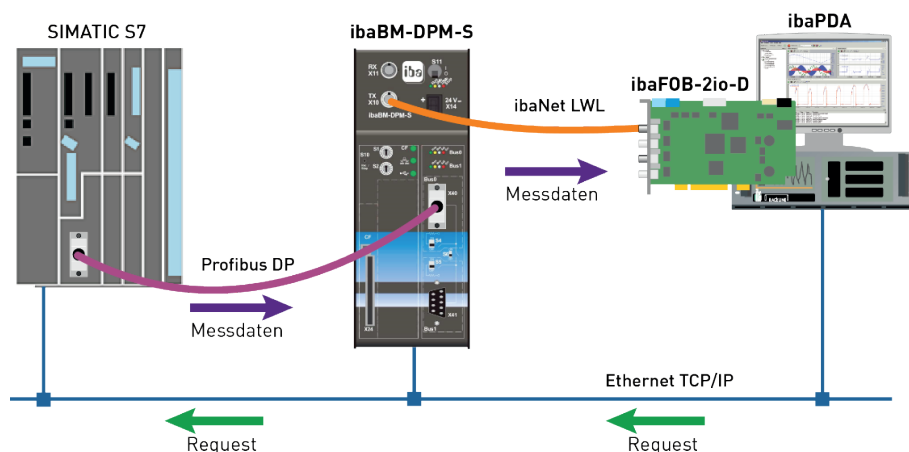
Die Übertragung der Messdaten erfolgt über PROFIBUS DP an das *ibaBM-DPM-S*-Gerät.

Sie benötigen insgesamt folgende Verbindungen:

- Onlineverbindung zwischen *ibaPDA* und S7-CPU (TCP/IP, MPI oder DP)
- Onlineverbindung zwischen *ibaPDA* und *ibaBM-DPM-S* (TCP/IP über Ethernet oder USB)
- Lichtwellenleiterverbindung zwischen *ibaPDA/ibaFOB-io-D* und *ibaBM-DPM-S*
- PROFIBUS-Verbindung zwischen *ibaBM-DPM-S* und S7 PROFIBUS-Master

In der Standardausführung können Sie maximal 8 Verbindungen pro Gerät konfigurieren, d. h. 8 PROFIBUS-Slaves. Pro Slave ist die Übertragung von max. 244 Byte Daten möglich.

Die folgende Darstellung zeigt die Variante mit einer TCP/IP-Onlineverbindung zwischen *ibaPDA* und S7-CPU.



Hinweis



Die folgende Beschreibung bezieht sich auf die Request-Blöcke ab Version V4.0. Wenn Sie Erläuterungen zu älteren Versionen benötigen, wenden Sie sich an den iba-Support.

Andere Dokumentation



Detaillierte Informationen zu *ibaBM-DPM-S* finden Sie im Gerätehandbuch.

Informationen zu Anwendungsbeispielen finden Sie in Kapitel [↗ Anwendungsbeispiele](#), Seite 95.

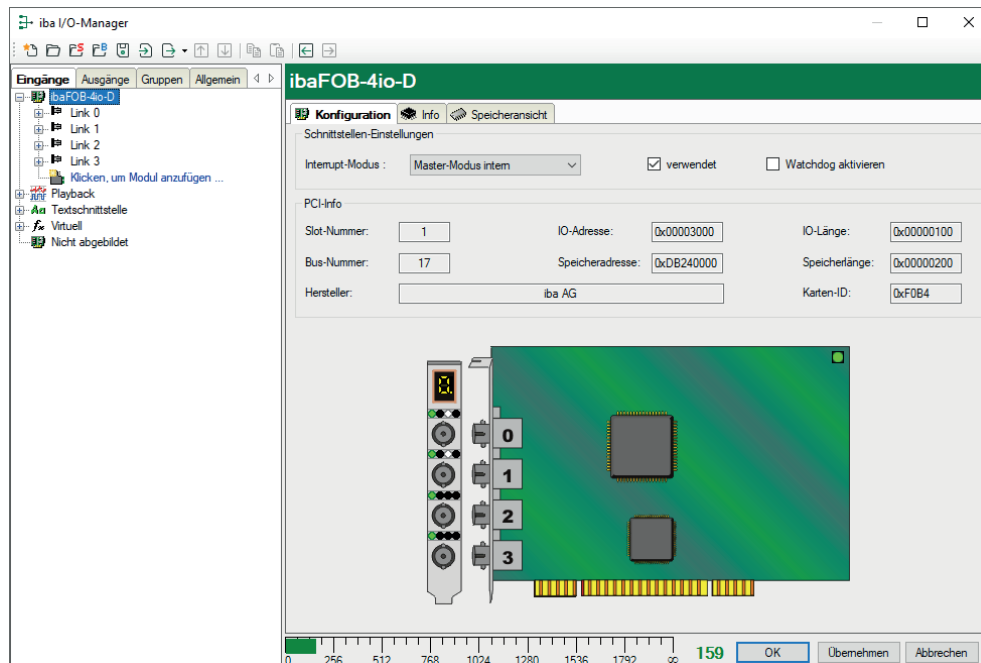
2.5.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

Die Konfiguration und Projektierung auf SIMATIC S7-Seite entspricht der des *ibaBM-DP* (siehe Kapitel [↗ Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC](#), Seite 27).

2.5.3 Konfiguration und Projektierung ibaPDA

2.5.3.1 Allgemeine Einstellungen der Schnittstelle

Wenn eine *ibaFOB-D*-Karte im *ibaPDA*-Rechner installiert ist, bietet *ibaPDA* im Schnittstellenbaum des I/O-Managers die Schnittstelle für diese *ibaFOB-D*-Karte an.



Andere Dokumentation

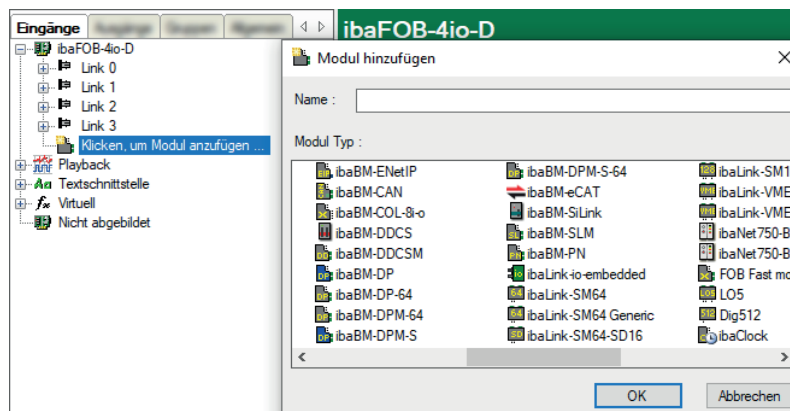


Detaillierte Informationen zur *ibaFOB-D*-Karte finden Sie im zugehörigen Handbuch.

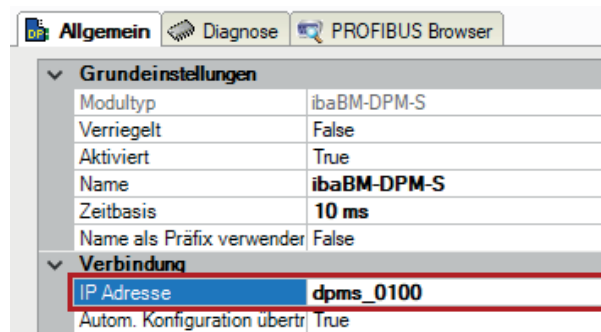
ibaBM-DPM-S konfigurieren

1. Starten Sie den *ibaPDA* Client und öffnen Sie den I/O-Manager.
2. Markieren Sie im Schnittstellenbaum (links) den Link der *ibaFOB-D*-Karte, an dem *ibaBM-DPM-S* angeschlossen ist.

Klicken Sie auf den blauen Befehl *Klicken, um Modul anzufügen*. Wählen Sie im Dialogfenster ein *ibaBM-DPM-S*-Modul aus und vergeben Sie bei Bedarf einen Namen über das Eingabefeld. Bestätigen Sie Ihre Auswahl mit <OK>.



3. Tragen Sie im Register *Allgemein* unter *Verbindung* die IP-Adresse des Geräts ein: entweder als Namen z. B. "dpms_000100" oder als IP-Adresse z. B. "192.168.81.123".

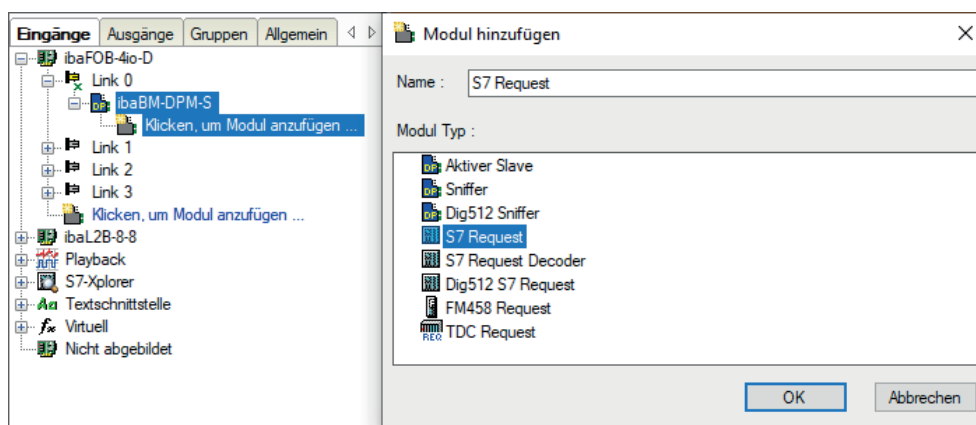


Hinweis



Eine Onlineverbindung zu dem Gerät über Ethernet oder USB ist unbedingt notwendig. Wie Sie eine solche Verbindung herstellen und überprüfen können, finden Sie im Handbuch zum *ibaBM-DPM-S*-Gerät.

4. Fügen Sie zum *ibaBM-DPM-S*-Modul ein Request-Modul hinzu (oder mehrere, falls Sie weitere Verbindungen zu einer bzw. zu verschiedenen S7-CPU's benötigen). Zur Auswahl stehen:
- S7 Request (zum Erfassen von analogen und digitalen Signalen)
 - S7 Request Decoder (zum Erfassen von bis zu 1024 digitalen Signalen)
 - S7 Request Dig512 (zum Erfassen von bis zu 512 digitalen Signalen)



5. Nehmen Sie die erforderlichen Moduleinstellungen und Konfiguration der Signale vor, wie in den folgenden Kapiteln beschrieben.

Bei allen Request-Modulen sind die Register *Allgemein* und *Verbindung* identisch. Die Request-Module unterscheiden sich nur bei den Registern *Analog* bzw. *Digital*.

6. Wenn Sie die Konfiguration beendet haben, klicken Sie auf <Übernehmen> oder <OK>, um die neue Konfiguration ins Gerät zu übertragen und die Datenerfassung mit *ibaPDA* zu starten.

2.5.3.2 Allgemeine Moduleinstellungen

Die Beschreibung der für alle Request-S7-Module identischen allgemeinen Einstellungen finden Sie im Handbuch Teil 1.

Die *ibaBM-DPM-S*-Module haben folgende spezifische Einstellmöglichkeiten:

PROFIBUS

Busnummer

0 = Stecker X40 oben, 1 = Stecker X41 unten

Slave Nummer

Dem Modul zugeordnete PROFIBUS-Slave-Adresse.

Verbindung

Automatisch aktivieren/deaktivieren

Bei TRUE wird die Erfassung gestartet, auch wenn keine Verbindung zu der S7-CPU aufgebaut werden kann. Das Modul wird deaktiviert. Während der Messung versucht *ibaPDA* sich mit der S7-CPU zu verbinden. Gelingt dies, wird die Erfassung neu gestartet.

Bei FALSE wird die Erfassung nicht gestartet, falls keine Verbindung zur projektierten S7-CPU möglich ist.

2.5.3.3 Verbindungseinstellungen

Die Verbindungseinstellungen sind identisch zum allgemeinen Kapitel *Verbindungseinstellungen* in Teil 1.

PROFIBUS-Slave-Nummer

Dem Modul zugeordnete PROFIBUS-Slave-Adresse. Wenn Sie den Eintrag im Register *Verbindung* ändern, ändert sich der Eintrag im Register *Allgemein* entsprechend und umgekehrt.

2.5.3.4 Modul S7 Request

Mit diesem Modultyp ist es möglich, 64 analoge und 64 digitale Signale aufzuzeichnen.

Projektieren Sie für jedes Modul einen separaten PROFIBUS-Slave und Request-Block-Aufruf.

Die Moduleinstellungen sind in Kapitel [↗ Allgemeine Moduleinstellungen](#), Seite 50, und im allgemein Kapitel *Allgemeine Moduleinstellungen* in Handbuch Teil 1 beschrieben.

2.5.3.5 Modul S7 Request Decoder

Mit dem Modul *S7 Request Decoder* können Sie bis zu 1024 digitale Signale erfassen, die in Form von max. 64 Wörtern (16 Bit) gesendet werden. Dieser Modultyp eignet sich daher besonders für Anwendungen, bei denen sehr viele digitale Signale erfasst werden müssen und die max. 512 direkt adressierbaren Digitalwerte des *ibaBM-DPM-S* nicht ausreichen.

Eine detaillierte Beschreibung des Moduls *S7 Request Decoder* finden Sie in der Modulbeschreibung zum Gerät *ibaBM-DP* in Teil 1 dieses Handbuchs.

2.5.3.6 Modul Dig512 S7 Request

Für das Modul *Dig512 S7 Request* gilt Gleiches wie für das Modul *S7 Request Decoder*. Jedoch können Sie nur 32 Wörter (512 digitale Werte) erfassen.

Das Modul *Dig512 S7 Request* ist der Vorgänger zum Modul *S7 Request Decoder* und wird in *ibaPDA* aus Gründen der Abwärtskompatibilität weiterhin unterstützt.

Verwenden Sie für neue Projektierungen das Modul *S7 Request Decoder*, siehe [➔ Modul S7 Request Decoder](#), Seite 50.

2.5.4 Diagnose

Sie erhalten eine Auflistung aller im Busmodul erfassten Operanden mit Datentyp und Istwert, wenn Sie im Schnittstellenbaum den Busmodulknoten auswählen und das Register *Analog* bzw. *Digital* öffnen.

Name	Symbol	Bus	Slave	I/O	Adresse	Datentyp	Istwert
Quelle: (2) S7 Request DP							
[2:0]: Test-DB\INT_1	BLOCKS\Blocks.Test-DB\INT_1	0	13	Out	8	INT_B	5122
[2:1]: Test-DB\AngularFreq0.1Hz	BLOCKS\Blocks.Test-DB\Angula...	0	13	Out	10	FLOAT_B	48323,3
[2:2]: Test-DB\AngularFreq1Hz	BLOCKS\Blocks.Test-DB\Angular...	0	13	Out	14	FLOAT_B	39668,3
[2:3]: Test-DB\AngularFreq10Hz	BLOCKS\Blocks.Test-DB\Angular...	0	13	Out	18	FLOAT_B	37132,5
[2:4]: Test-DB\Sinus_0.1Hz	BLOCKS\Blocks.Test-DB\Sinus_...	0	13	Out	22	FLOAT_B	-0,646176
[2:5]: Test-DB\Cosinus_0.1Hz	BLOCKS\Blocks.Test-DB\Cosinu...	0	13	Out	26	FLOAT_B	0,763188

Andere Dokumentation



Eine detaillierte Beschreibung der gerätespezifischen Diagnosemöglichkeiten des *ibaBM-DPM-S* finden Sie im zugehörigen Gerätehandbuch.

2.6 Request-S7 für ibaBM-DPM-S im Redundanzmodus

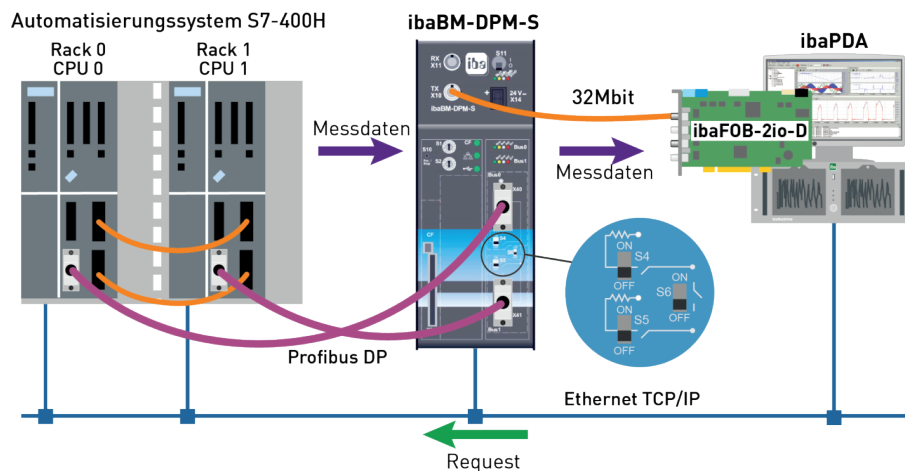
Im Folgenden wird die Request-S7-Variante für das PROFIBUS-Busmodul *ibaBM-DPM-S* im Redundanzmodus beschrieben. Die Lösung "Request-S7 für *ibaBM-DPM-S* im Redundanzmodus" wird durch "Request-S7 für *ibaBM-DP* im Redundanzmodus" funktionskompatibel abgelöst.

2.6.1 Allgemeine Informationen

Der Redundanzmodus des *ibaBM-DPM-S* ermöglicht den Betrieb an redundanten PROFIBUS-Systemen in Verbindung mit SIMATIC S7-400H Steuerungen, deren Messdaten erfasst werden sollen. Um den Redundanzmodus des *ibaBM-DPM-S* verwenden zu können, benötigen Sie eine zusätzliche Lizenz. Wenden Sie sich hierfür an den iba-Support. Die Lizenz wird über die Administratorfunktionen im Web-Dialog des *ibaBM-DPM-S* freigeschaltet.

Request-S7 redundant ist geeignet, um *ibaBM-DPM-S* mit Request-Funktionalität als einkanalig geschaltete Peripherie an einer hochverfügbaren SIMATIC S7-400H Steuerung zu betreiben.

Die folgende Darstellung zeigt eine beispielhafte Einbindung eines *ibaBM-DPM-S* im Redundanzmodus:



Hinweis



Die folgende Beschreibung bezieht sich auf die Request-Blöcke ab Version V4.0.

Andere Dokumentation



Detaillierte Informationen zum Redundanzmodus des *ibaBM-DPM-S* finden Sie im Gerätehandbuch.

Die Funktionalität des Request-S7 für das *ibaBM-DPM-S* im Redundanzmodus entspricht weitestgehend der im Standardmodus (siehe Kapitel [Request-S7 für ibaBM-DPM-S](#), Seite 46). Die Abweichungen und Erweiterungen sind nachfolgend beschrieben.

Informationen zu Anwendungsbeispielen finden Sie in Kapitel [Anwendungsbeispiele](#), Seite 95.

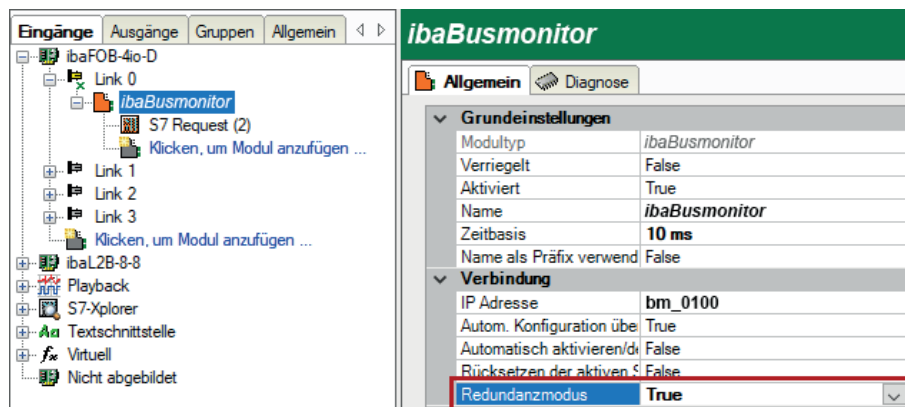
2.6.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

Die Konfiguration und Projektierung auf SIMATIC S7-Seite entspricht der des *ibaBM-DP* (siehe Kapitel [↗ Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC](#), Seite 40).

2.6.3 Konfiguration und Projektierung ibaPDA

Schließen Sie zunächst *ibaBM-DPM-S* an einen freien Link einer *ibaFOB-D*-Karte an. Im I/O-Manager fügen Sie an dem entsprechenden Link ein Gerätemodul *ibaBM-DPM-S* ein.

Im Register *Allgemein* setzen Sie den Parameter *Redundanzmodus* auf "True". Ein orange gefärbte Symbol des Gerätemoduls zeigt den Redundanzmodus an.



Da beim redundanten PROFIBUS die beiden Bussysteme 0 und 1 parallel betrieben werden, gibt es im Register *Allgemein* nur noch Einstellungen für *Redundanter PROFIBUS*.

Im Redundanzmodus stehen mit Request-S7 zusätzlich folgende Module zur Verfügung:

- S7 Request
- S7 Request Decoder
- Dig512 S7 Request

Die Konfiguration der Module entspricht der im Standardmodus, siehe [↗ Allgemeine Moduleinstellungen](#), Seite 50.

Verbindungseinstellungen

Richten Sie für die Übertragung der Operandendaten zu den beiden CPUs der SIMATIC S7-400H zwei getrennte Verbindungen ein. *ibaPDA* schaltet die Verbindung je nach Verfügbarkeit und Bedarf um.

Die Verbindungseinstellungen nehmen Sie identisch zum Request-S7 bei *ibaBM-DP* im Redundanzmodus vor (siehe hierzu Kapitel [↗ Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC](#), Seite 40).

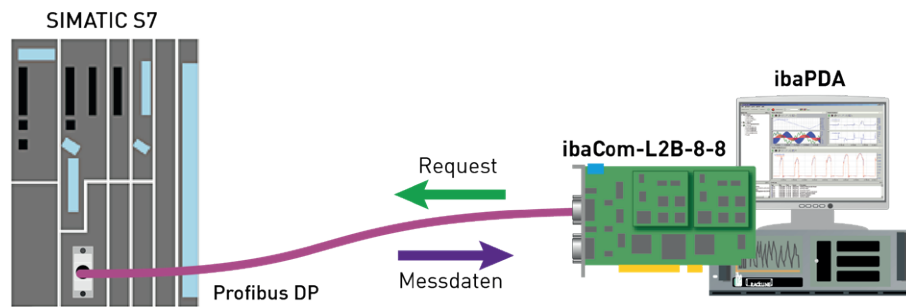
2.7 Request-S7 für ibaCom-L2B

Im Folgenden wird die Request-S7-Variante für die *ibaCom-L2B* PROFIBUS-Karte beschrieben.

Die Lösung Request S7 für *ibaCom-L2B* wird durch Request-S7 für *ibaBM-DP* funktionskompatibel abgelöst.

Informationen hierzu finden Sie in Kapitel [↗ Ablösung Request-S7 auf ibaCom-L2B durch ibaBM-DP](#), Seite 108.

2.7.1 Allgemeine Informationen



Beim L2B-Request erfolgt die Anforderung der Messwerte (Request-Handshake) über den PROFIBUS. Hierzu dient ein Teil des I/O-Bereichs des jeweiligen PROFIBUS-Slaves. Sie benötigen keine weitere TCP/IP-Verbindung vom *ibaPDA*-Rechner zur S7-Steuerung für die Konfiguration und Projektierung.

Hinweis



Die folgende Beschreibung bezieht sich auf die Request-Blöcke ab Version V3.4. Wenn Sie Erläuterungen zu älteren Versionen benötigen, wenden Sie sich an den iba-Support.

Ältere Versionen sind erforderlich, wenn *ibaPDA-Request-S7-DP/PN/ibaNet-E* in Verbindung mit älteren S7-CPUs verwendet werden soll, z. B. CPU 315 vor Ausgabestand 2AF03 oder CPU 314 vor Ausgabestand 2AF04.

Informationen zu Anwendungsbeispielen finden Sie in Kapitel [↗ Anwendungsbeispiele](#), Seite 95.

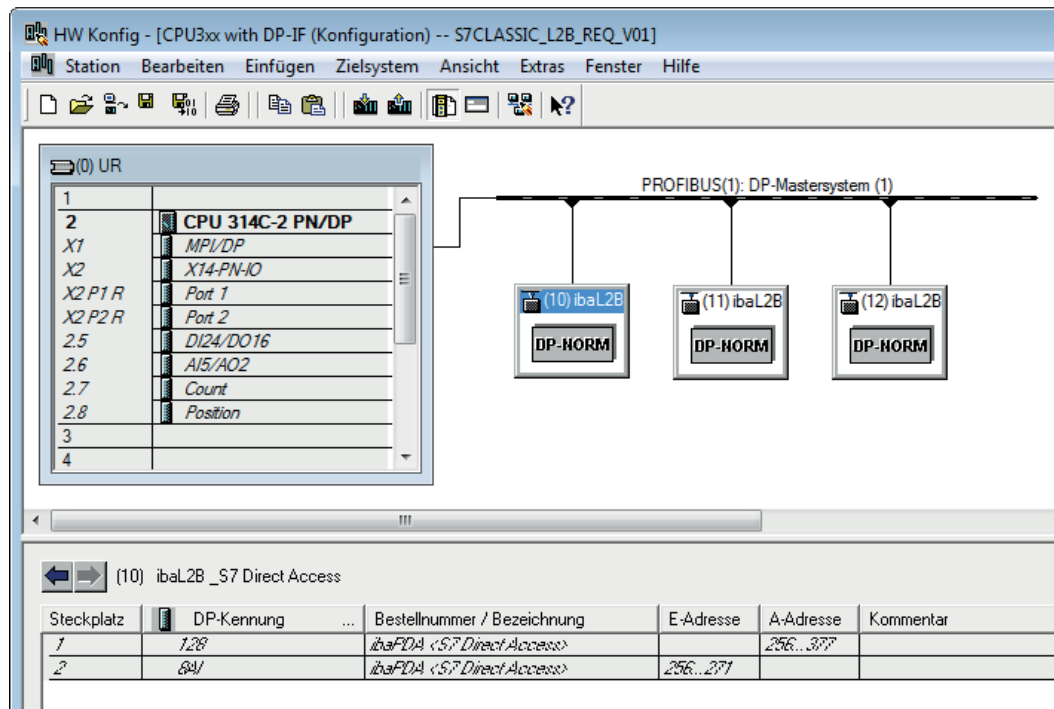
2.7.2 Konfiguration und Projektierung SIMATIC S7-300, S7-400 und WinAC

2.7.2.1 Projektierung Hardware

Führen Sie folgende Schritte aus:

1. Vernetzen Sie die DP-Schnittstelle der CPU oder eines CP als DP-Master mit einem PROFIBUS.
2. Installieren Sie die GSD-Datei [IBA_0F05.GSD](#) über den Menüpunkt *Extras – GSD-Dateien*.
3. Nach Installation finden Sie die iba-GSD-Datei im HW-Katalog unter:
PROFIBUS-DP – weitere Feldgeräte – Allgemein – ibaL2B <S7 Direct Access>

4. Hängen Sie die gewünschte Anzahl von DP-Slaves "ibaL2B <S7 Direct Access>" an PROFIBUS an und stellen Sie die DP-Adressen ein. Dabei werden die Peripherie-Adressen den Slaves automatisch nach aufsteigenden Adressen zugewiesen.
5. Laden Sie die HW Konfig./Systemdaten in die S7-CPU.



2.7.2.2 Projektierung in STEP 7 (KOP, FUP, AWL)

2.7.2.2.1 CPU-interne DP-Schnittstelle oder CP 443-5 (bei S7-400)

Wenn Sie eine CPU S7-300/-400 mit integrierter DP-Schnittstelle verwenden oder bei einer S7-400 die externe Schnittstelle CP 443-5 verwenden, führen Sie die nachfolgenden Schritte durch.

Wenn Sie die externe DP-Schnittstelle CP342-5 einer CPU S7-300 verwenden, gehen Sie vor wie in Kapitel [↗ Externe DP-Schnittstelle CP342-5](#), Seite 57 beschrieben.

Für Request-S7 mit einem PROFIBUS-Slave

1. Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe [↗ iba S7-Bibliothek](#), Seite 92. Die Funktionen müssen jeweils einmal pro aktiviertem PROFIBUS-Slave im Programm aufgerufen werden.
 - FC111 (ibaL2B_Init) und FC112 (ibaL2B_Req), siehe [↗ Initialisierungs-FC ibaL2B_Init \(FC111\)](#), Seite 80 und [↗ Kommunikations-FC ibaL2B_Req \(FC112\)](#), Seite 85
 - DB22 (ibaL2B_DB_work) und UDT22 (ibaL2B_DB_Struct)

Hinweis



Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!
Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Hinweis

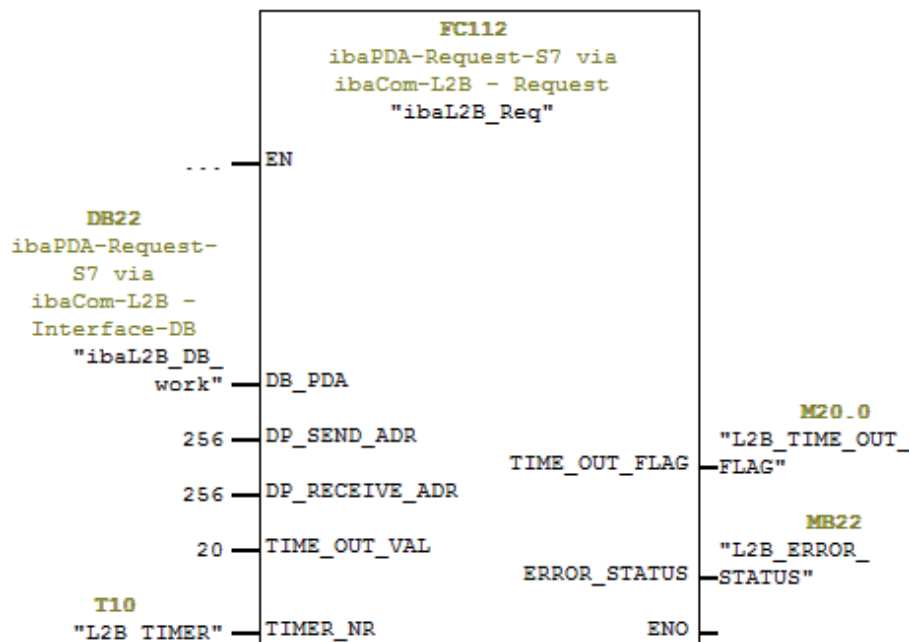
Falls die Bausteinnummern FC111, FC112, DB22 und UDT22 in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

- Legen Sie die Fehler-OBs (OB82, OB85, OB86, OB87, OB122) an, um einen CPU-STOP im Fehlerfall zu verhindern.

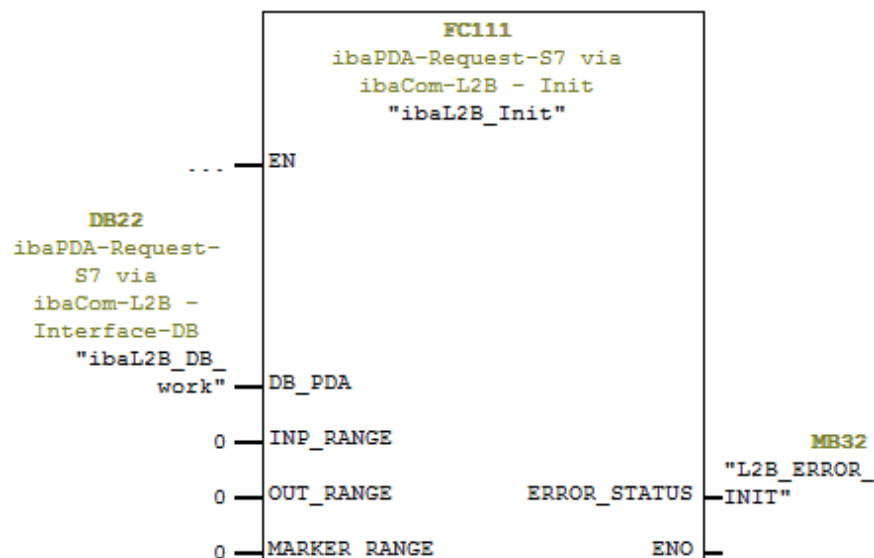
Hinweis

DB22 und der UDT22 enthalten eine Datenstruktur, die für die Bearbeitung nicht notwendig ist. Im Prinzip genügt ein DB mit Länge 1500 Bytes. Die Datenstruktur ist nur für Diagnosezwecke interessant.

- Im zyklischen Programm den ibaL2B_Req (FC112) aufrufen und parametrieren.



- In den Anlauf-OBs (OB100, OB101, OB102) den ibaL2B_Init (FC111) aufrufen und parametrieren.



Für Request-S7 mit weiteren PROFIBUS-Slaves

- Im Bausteinordner muss für jeden Request-Slave ein Datenbaustein vorhanden sein, auf den sich beide FCs beziehen. Kopieren Sie den ibaL2B_DB_work (DB22) in einen DB mit einer neuen DB-Nummer.
- In den Anlauf-OBs müssen Sie den ibaL2B_Init (FC111) ein weiteres Mal mit der neuen DB-Nummer aufrufen.
- Im zyklischen OB müssen Sie den ibaL2B_Req (FC112) ein weiteres Mal mit der neuen DB-Nummer und den Peripherieadressen des neuen PROFIBUS-Slaves aufrufen.

Abschluss

- Laden Sie alle Bausteine in die S7-CPU und starten Sie die S7-CPU neu.

2.7.2.2.2 Externe DP-Schnittstelle CP342-5

Bei Verwendung einer externen DP-Schnittstelle CP342-5 führen Sie die nachfolgenden Schritte durch.

Für Request-S7 mit einem PROFIBUS-Slave

1. Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe [↗ iba S7-Bibliothek](#), Seite 92. Die Funktionen müssen jeweils einmal pro aktiviertem PROFIBUS-Slave im Programm aufgerufen werden.
 - ibaL2B_Req_CP (FC113) und ibaL2B_Init (FC111), siehe [↗ Kommunikations-FC ibaL2B_Req_CP \(FC113\) für CP342-5](#), Seite 83 und [↗ Initialisierungs-FC ibaL2B_Init \(FC111\)](#), Seite 80
 - ibaL2B_CP_SNDRCV (DB10), ibaL2B_DB_work (DB22) und ibaL2B_DB_Struct (UDT22)

Hinweis



Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!
Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Hinweis

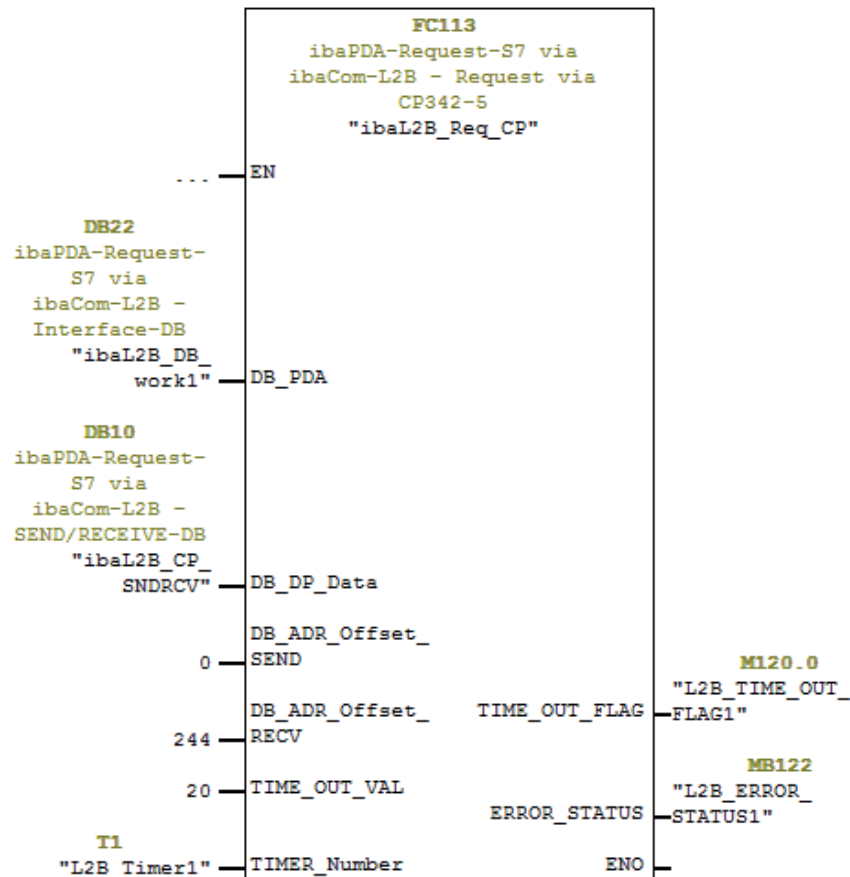


Falls die Bausteinnummern FC111, FC113, DB10, DB22 und UDT22 in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

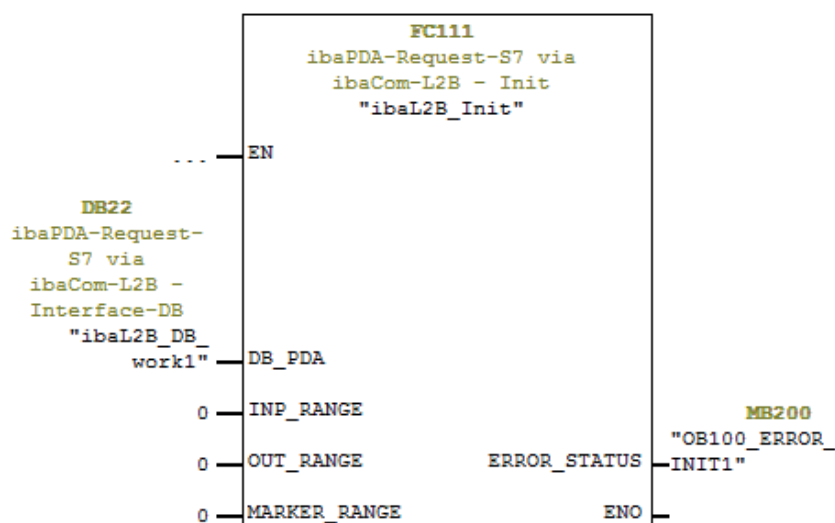
2. Legen Sie die Fehler-OBs (OB82, OB85, OB86, OB87, OB122) an, um einen CPU-STOP im Fehlerfall zu verhindern.

3. Im zyklischen Programm den ibaL2B_Req_CP (FC113) aufrufen und parametrieren.

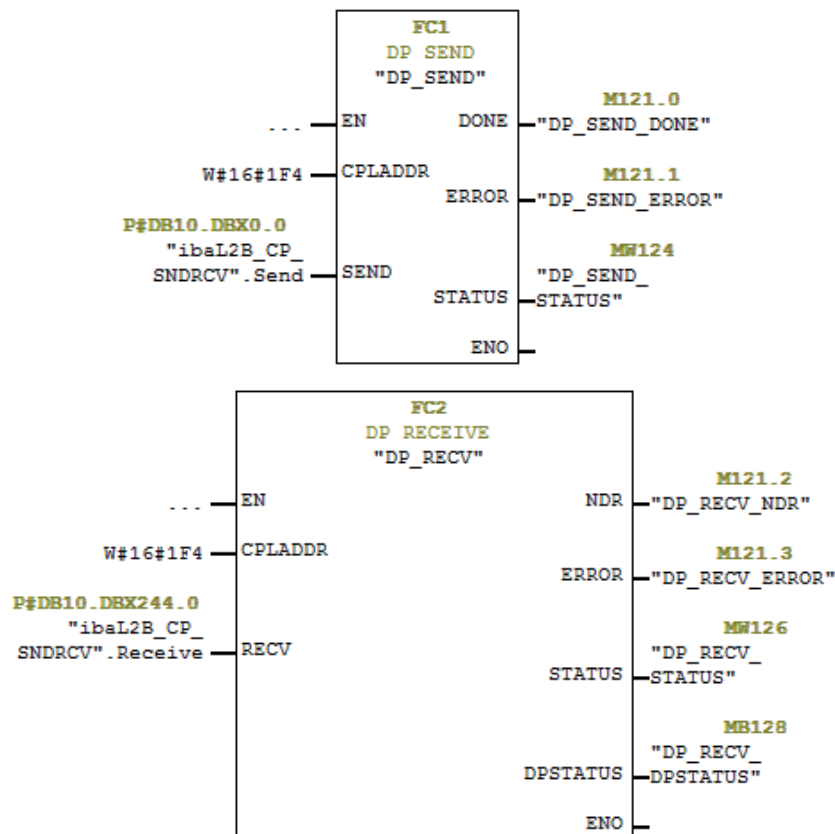
Als Offset-Adressen für Sende- und Empfangsdaten müssen hier für jeden Slave eigene Bereiche innerhalb des DBs "ibaPDA_SENDRECV" (DB10) festgelegt und als Parameter "DB_ADR_Offset_SEND" bzw. "DB_ADR_Offset_RECV" übergeben werden!



4. In den Anlauf-OBs (OB100, OB101, OB102) den ibaL2B_Init (FC111) aufrufen und parametrieren.



5. Aufrufen und parametrieren von FC1 (DP_SEND) und FC2 (DP_RECV) im zyklischen Programm.



Für Request-S7 mit weiteren PROFIBUS-Slaves

- Im Bausteinordner muss für jeden Request-Slave ein Datenbaustein vorhanden sein, auf den sich beide FCs beziehen. Kopieren Sie den ibaL2B_DB_work (DB22) in einen DB mit neuer DB-Nummer.
- Im ibaL2B_CP_SNDRCV (DB10) müssen pro Request-Slave 122 Bytes im Array "Send" und 16 Bytes im Array "Receive" reserviert werden. Ggf. müssen Sie die Arrays erweitern.
- In den Anlauf-OBs müssen Sie den ibaL2B_Init (FC111) ein weiteres Mal mit der neuen DB-Nummer aufrufen.
- Im zyklischen OB müssen Sie den ibaL2B_Req_CP (FC113) ein weiteres Mal mit der neuen DB-Nummer und den entsprechenden Adress-Offsets für den Sendebereich und Empfangsbereich im ibaL2B_CP_SNDRCV (DB10) des neuen PROFIBUS-Slaves aufrufen.

Abschluss

- Laden Sie alle Bausteine in die S7-CPU und starten Sie die S7-CPU neu.

Hinweis



Die Struktur der Sendedaten und Empfangsdaten im DB10 muss den Ausgangs-Adressen bzw. den Eingangs-Adressen der DP-Slaves entsprechen.

Bei mehr als 2 Slaves müssen Sie die Arrays im DB10 um jeweils 122 Bytes für die Senderichtung und jeweils 16 Bytes für die Empfängerichtung erweitern.

Das bedeutet z. B. bei vier Slaves folgende Zuordnung:

	DB10		PROFIBUS P-Adressen	ibaL2B_Req_CP. ...Offset_SEND	ibaL2B_Req_CP. ...Offset_RECV
1. Slave	Send	Offset 0	A-Adresse 0	0	
2. Slave		Offset 122	A-Adresse 122	122	
3. Slave		Offset 244	A-Adresse 244	244	
4. Slave		Offset 366	A-Adresse 366	366	
1. Slave	Receive	Offset 0	E-Adresse 0		488
2. Slave		Offset 16	E-Adresse 16		504
3. Slave		Offset 32	E-Adresse 32		520
4. Slave		Offset 48	E-Adresse 48		536

Hinweis



Alle Parameter, wie Zeiten und Merker, müssen unterschiedlich sein!

2.7.2.3 Projektierung in STEP 7 (CFC)

2.7.2.3.1 CPU-interne DP-Schnittstelle oder CP 443-5 (bei S7-400)

Wenn Sie eine CPU S7-300/400 mit integrierter DP-Schnittstelle verwenden oder bei einer S7-400 die externe Schnittstelle CP 443-5 verwenden, führen Sie die nachfolgenden Schritte durch.

Wenn Sie die externe DP-Schnittstelle CP342-5 einer CPU S7-300 verwenden, gehen Sie vor wie in Kapitel [↗ Externe DP-Schnittstelle CP342-5](#), Seite 57 beschrieben.

Für Request-S7 mit einem PROFIBUS-Slave

- Kopieren Sie aus der iba S7-Bibliothek folgende Bausteine in den Bausteinordner Ihres STEP 7-Projekts, siehe [↗ iba S7-Bibliothek](#), Seite 92. Die Funktionen müssen jeweils einmal pro aktiviertem PROFIBUS-Slave im Programm aufgerufen werden.
 - FC111 (ibaL2B_Init) und FC112 (ibaL2B_Req), siehe [↗ Initialisierungs-FC ibaL2B_Init \(FC111\)](#), Seite 80 und [↗ Kommunikations-FC ibaL2B_Req \(FC112\)](#), Seite 85
 - DB22 (ibaL2B_DB_work) und UDT22 (ibaL2B_DB_Struct)

Hinweis



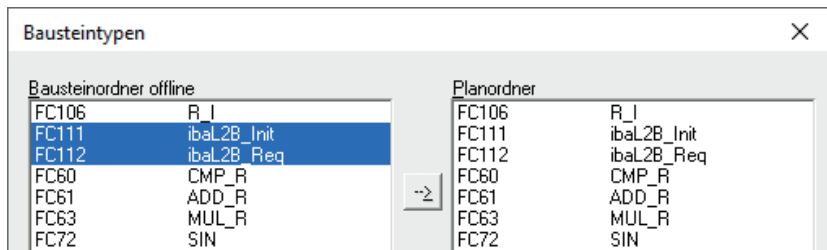
Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!
Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Hinweis

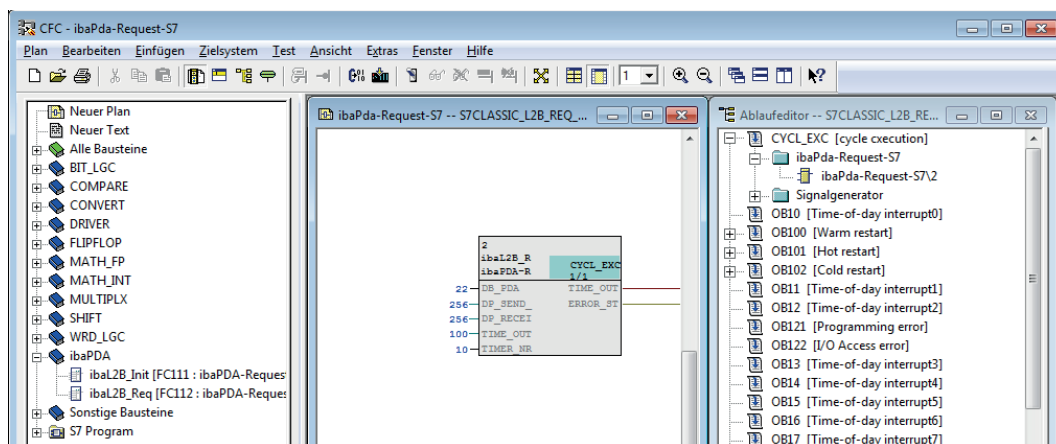
Falls die Bausteinnummern FC111, FC112, DB22 und UDT22 in Ihrem Projekt schon verwendet sind, weisen Sie den Bausteinen aus der iba S7-Bibliothek beim Kopieren andere freie Nummern zu.

- Starten Sie den CFC-Editor und importieren Sie diese Bausteine (unter *Extras – Baustein-typen*).

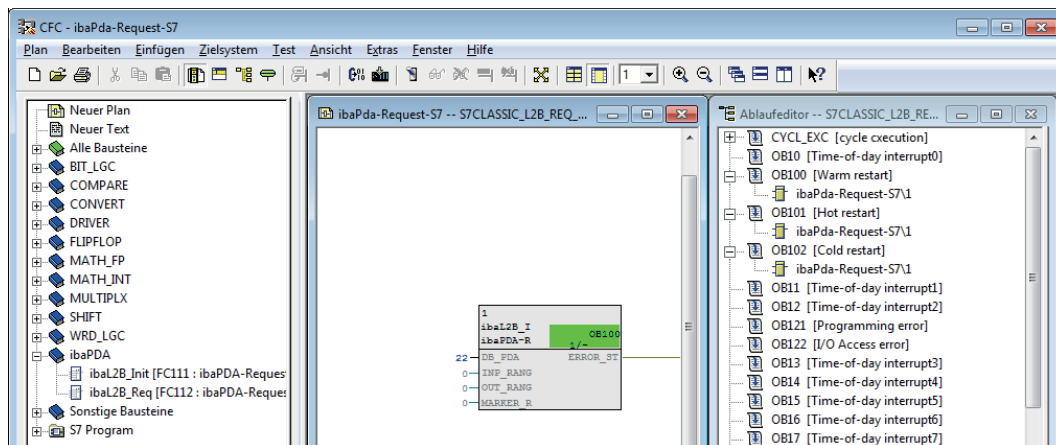
Die FCs werden in der Gruppe "ibaPDA" im Baustein-Katalog abgelegt.



- Ziehen Sie den ibaL2B_Req (FC112) in Ihren CFC-Plan. Achten Sie auf die Ablaufreihenfolge. Der ibaL2B_Req muss in einer zyklischen Task aufgerufen werden (z. B. Weckalarm OB35 oder freilaufend OB1).

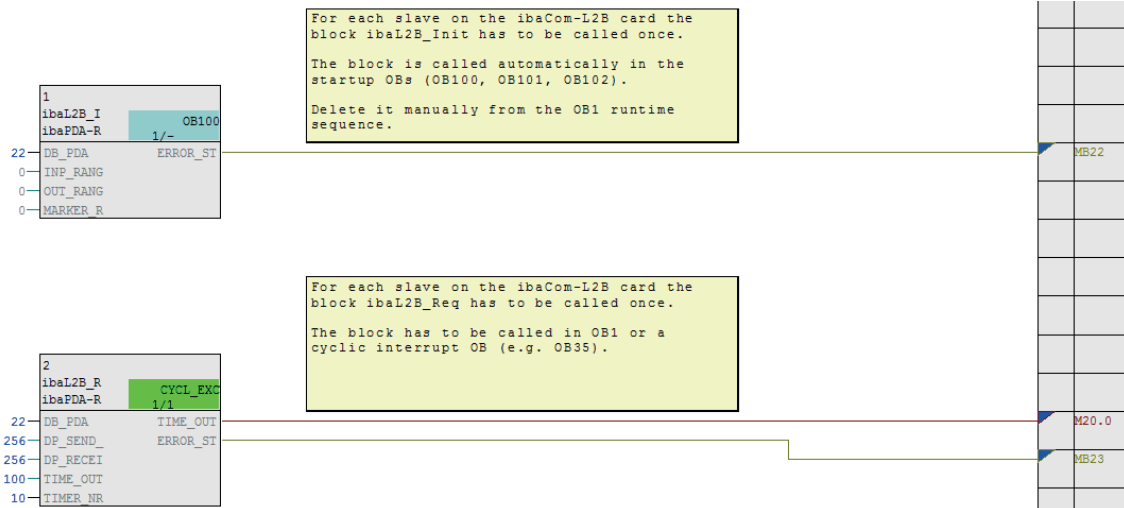


- Ziehen Sie den ibaL2B_Init (FC111) in Ihren CFC-Plan. Achten Sie auf die Ablaufreihenfolge. Der ibaL2B_Init wird automatisch in die Anlauf tasks (OB100, OB101 und OB102) eingetragen. Entfernen Sie ihn aus den zyklischen Tasks (z. B. aus OB35).



5. Verschalten Sie die Bausteine:

- Tragen Sie an beiden FCs am Eingang DB_NUMBER die DB-Nummer des Interface-DBs ein (DB22).
- Tragen Sie unter DP_SEND_ADR und DP_RECEIVE_ADR des Bausteins ibaL2B_Req die Peripherieadressen des PROFIBUS-Slaves aus der Hardwarekonfiguration ein.



Achten Sie darauf, dass die Fehler-OBs (OB82, OB85, OB86, OB87, OB122) angelegt werden, sonst geht die CPU bei einem Zugriffsfehler auf STOP.

Dazu müssen Sie in der Ablaufreihenfolge in diesen Tasks, falls sie leer sind, jeweils eine (leere) Ablaufgruppe definieren. Alternativ können Sie auch einen beliebigen Dummy-Block im Fehler-OB platzieren, der ebenfalls nicht gelöscht werden darf.

Hinweis



Wählen Sie beim Kompilieren **nicht** die Option *Leere Ablaufgruppen löschen*, sonst werden die Fehler-OBs wieder entfernt! Bei einigen älteren CFC-Versionen ist diese Option beim Kompilieren auswählbar.

Für Request-S7 mit weiteren PROFIBUS-Slaves

- Im Bausteinordner muss für jeden Request-Slave ein Datenbaustein vorhanden sein, auf den sich beide FCs beziehen. Kopieren Sie den ibaL2B_DB_work (DB22) in einen DB mit einer neuen DB-Nummer.
- Der ibaL2B_Init (FC111) muss ein weiteres Mal auf einem CFC-Plan mit der neuen DB-Nummer aufgerufen werden.
- Der ibaL2B_Req (FC112) muss ein weiteres Mal auf einem CFC-Plan mit der neuen DB-Nummer und den Peripherieadressen des neuen Slaves aufgerufen werden.

Nach jeder Programmänderung

1. Kompilieren Sie das komplette Programm:
 - a.) Aktivieren Sie die Option *Gesamtes Programm*.
 - b.) Aktivieren Sie die Option *SCL-Quelle erzeugen* (ab SIMATIC CFC Version 6.1).
 - c.) Deaktivieren Sie die Option *Leere Ablaufgruppen löschen* falls verfügbar.
2. Laden Sie das Programm und starten Sie die S7-CPU neu.

Hinweis



Je nach S7-CPU-Typ kommen nach Kompilieren und Laden folgende Warnmeldungen:

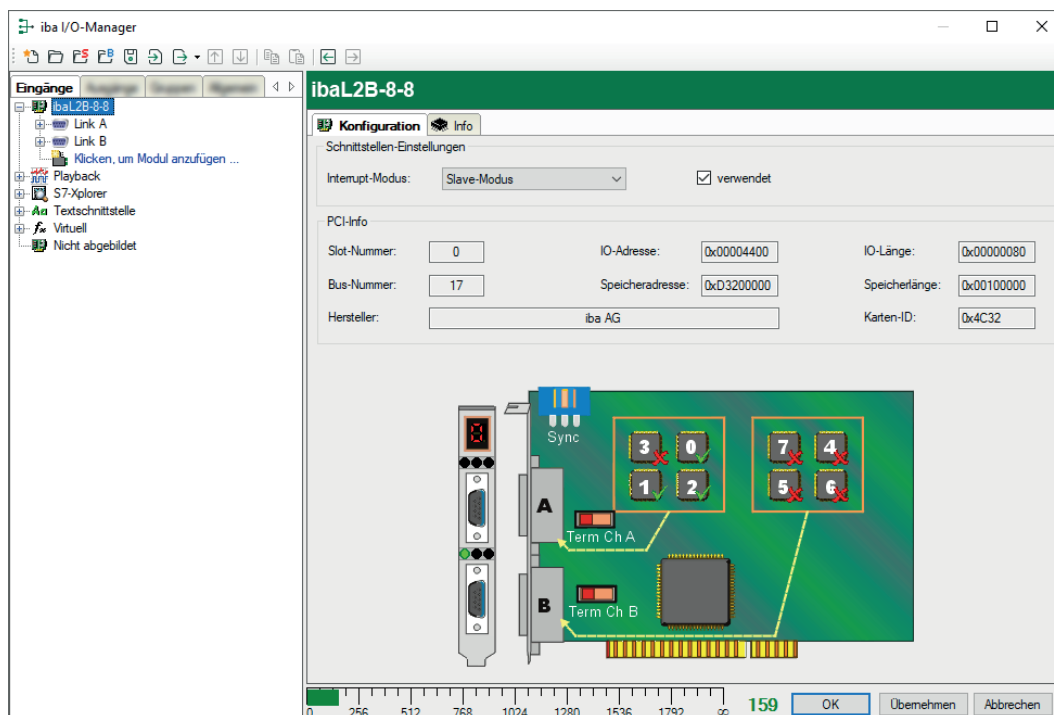
- "W: Der OB101/ OB102 wird in dieser CPU nicht unterstützt."
- "W: Der OB101/OB102 konnte nicht geladen werden, weil er von der angeschlossenen Online-CPU nicht unterstützt wird."

Diese Meldungen erscheinen, da der ibaL2B_Init automatisch in die Anlauftasks OB101 und OB102 eingetragen wird. Einige S7-CPU-Typen unterstützen aber OB101 und OB102 nicht. Sie können die Warnungen ignorieren.

2.7.3 Konfiguration und Projektierung ibaPDA

2.7.3.1 Allgemeine Einstellungen der Schnittstelle

Wenn eine *ibaCom-L2B*-Karte im *ibaPDA*-Rechner installiert ist, bietet *ibaPDA* im Schnittstellenbaum des I/O-Managers die Schnittstelle für diese *ibaCom-L2B*-Karte an.



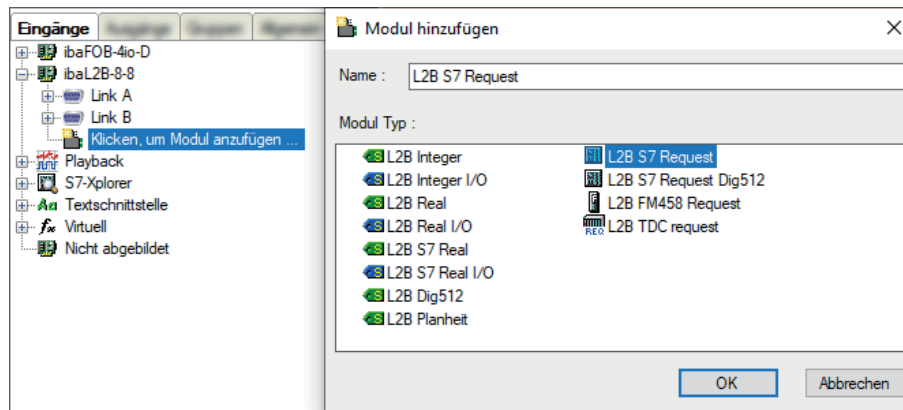
Andere Dokumentation



Detaillierte Informationen zur *ibaCom-L2B*-Karte finden Sie im zugehörigen Handbuch.

2.7.3.2 Allgemeine Moduleinstellungen

Es stehen die Modultypen *L2B S7 Request* und *L2B S7 Request Dig512* zur Verfügung.



Die Beschreibung der für alle Request-S7-Module identischen allgemeinen Einstellungen finden Sie im Handbuch Teil 1.

Die L2B-Module haben folgende gemeinsame spezifische Einstellmöglichkeiten:

PROFIBUS

Slave Nummer

Dem Modul zugeordnete PROFIBUS-Slave-Adresse

Timeout

Wartezeit in Sekunden bevor die Firmware die Verbindung als unterbrochen betrachtet.

Verbindungseinstellungen

Im Gegensatz zu den S7-Request-Lösungen mit den Geräten *ibaBM-DP* bzw. *ibaBM-DPM-S* sind hier keine zusätzlichen Verbindungseinstellungen erforderlich. Der Request-Handshake erfolgt über einen Teil des I/O-Peripheriebereichs.

2.7.3.3 Modul L2B S7 Request

Mit dem Modul *L2B S7 Request* ist es möglich, bis zu 32 analoge und 32 digitale Signale zu erfassen.

Projektieren Sie für jedes Modul einen separaten PROFIBUS-Slave und Request-Block-Aufruf.

Die Moduleinstellungen sind in Kapitel **7 Allgemeine Moduleinstellungen**, Seite 64 sowie in Teil 1 beschrieben.

2.7.3.4 Modul L2B S7 Request Dig512

Mit dem Modul *L2B S7 Request Dig512* können Sie bis zu 512 digitale Signale erfassen, die in Form von max. 32 Wörtern (16 Bit) gesendet werden. Dieser Modultyp eignet sich daher besonders für Anwendungen, bei denen sehr viele digitale Signale erfasst werden müssen und die direkt adressierbaren Digitalwerte der Module *L2B S7 Request* nicht ausreichen.

Projektieren Sie für jedes Modul ist ein separater PROFIBUS-Slave und Request-Block-Aufruf.

Register Allgemein

Für weitere Informationen zur Moduleinstellung siehe [Allgemeine Moduleinstellungen](#), Seite 64.

Register Digital

Die Deklaration der Digitalsignale erfolgt zweistufig.

- Definieren Sie zunächst die Wörter (Quellsignale), welche für die Digitalsignale (Bits) aufgeschlüsselt werden.

Die Wörter können Sie direkt als Basis-Signale für die Dekodierung über absolute S7-Operanden eintragen. Es sind nur Wort-Operanden (z. B. PEW, MW, DBW) erlaubt.

Ebenso ist die Verwendung von S7-Symbolen durch das Erzeugen von Adressbüchern möglich. Weitere Informationen siehe Handbuch Teil 1, Abschnitt *Auswahl über die symbolischen Operandenadressen*. Die im S7 CFC- und Symbol-Browser ausgewählten Signale werden übernommen und die Spalten *Name*, *S7 Symbol* und *S7 Operand* automatisch ausgefüllt.

- Jedes Wort (Quellsignal) können Sie über den Button <+> öffnen, um die Liste der zugehörigen Digitalsignalen anzuzeigen.

Definieren Sie danach die einzelnen Digitalsignale (Bits) des Wortes.

L2B S7 Request Dig512 (6)

Decoder

	S7 Operand	Aktiv
0	PEW 1	<input checked="" type="checkbox"/>
1	DB 3.DBW 2	<input type="checkbox"/>
2	DB 3.DBW 4	<input type="checkbox"/>

Name

Name	Aktiv
Digital Signal 0	<input type="checkbox"/>
Digital Signal 1	<input type="checkbox"/>
Digital Signal 2	<input type="checkbox"/>
Digital Signal 3	<input type="checkbox"/>
Digital Signal 4	<input type="checkbox"/>
Digital Signal 5	<input type="checkbox"/>
Digital Signal 6	<input type="checkbox"/>
Digital Signal 7	<input type="checkbox"/>
Digital Signal 8	<input type="checkbox"/>
Digital Signal 9	<input type="checkbox"/>
Digital Signal 10	<input type="checkbox"/>
Digital Signal 11	<input type="checkbox"/>
Digital Signal 12	<input type="checkbox"/>
Digital Signal 13	<input type="checkbox"/>
Digital Signal 14	<input type="checkbox"/>
Digital Signal 15	<input type="checkbox"/>

3

	S7 Operand	Aktiv
3	DB 3.DBW 6	<input checked="" type="checkbox"/>

Die einzelnen Spalten der Signaltabelle haben folgende Bedeutungen.

Quellsignal**Decoder**

Tragen Sie einen Namen für das Quellsignal ein.

S7 Operand/S7 Symbol

Tragen Sie den S7 Operand und ggf. das S7 Symbol ein, dem das Signal zugeordnet ist.

Aktiv

Wenn Sie das Quellsignal aktivieren, wird es mit allen Digitalsignalen erfasst. Sie können einzelne Digitalsignale abwählen.

Einzelne Digitalsignale (Bits)**Name**

Tragen Sie einen Namen für die einzelnen Digitalsignale ein.

Aktiv

Wenn Sie das Digitalsignal aktivieren, wird das Signal erfasst und auch in der Prüfung der Anzahl der lizenzierten Signale berücksichtigt.

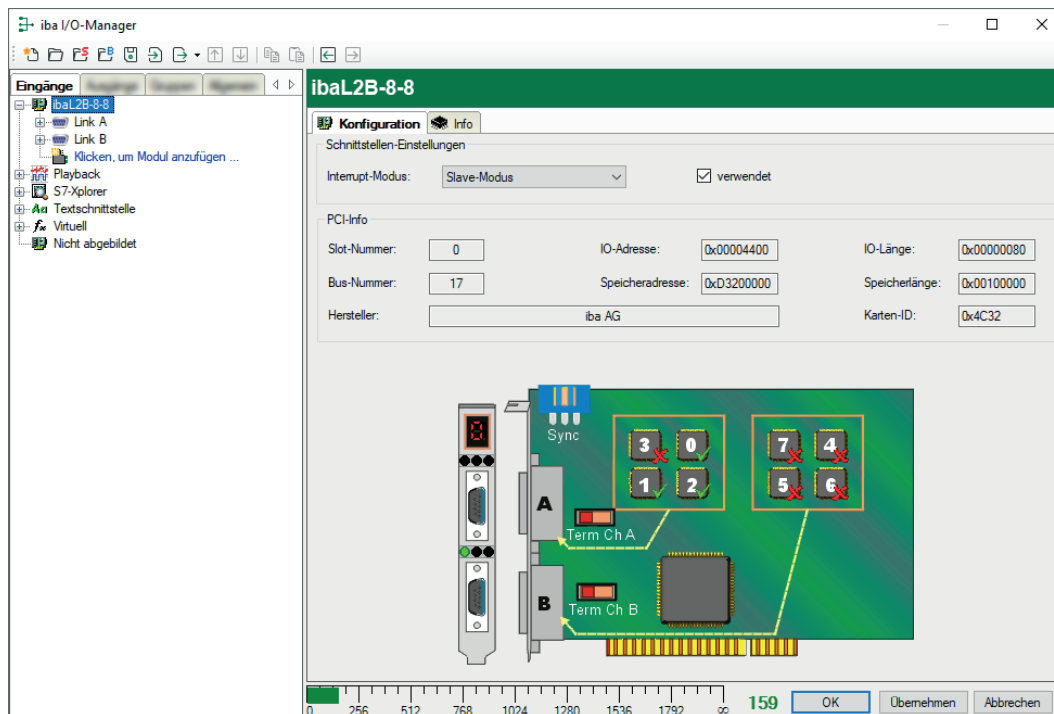
Hinweis

ibaPDA berücksichtigt jeweils nur die aktivierten Digitalsignale bei der Anzahl der lizenzierten Signale, also kein zusätzliches Signal für das Quellsignal.

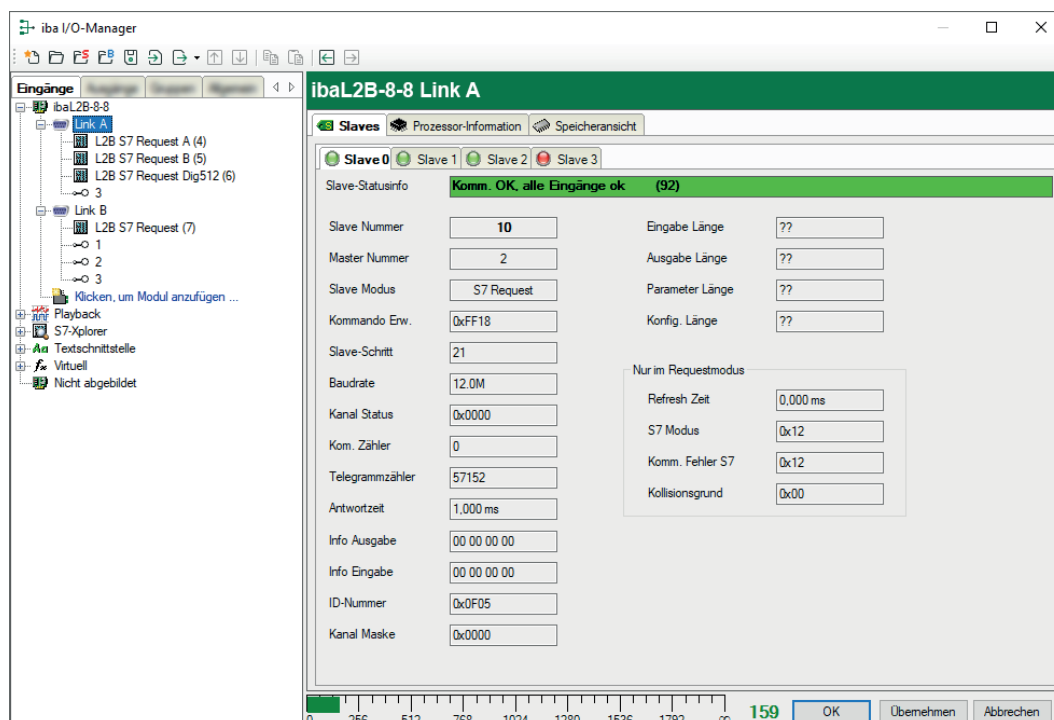
2.7.4 Diagnose

2.7.4.1 ibaCom-L2B-Karte

Wenn Sie im Schnittstellenbaum eine L2B-Karte auswählen, zeigt *ibaPDA* eine Status-Übersicht der Karte an. Der grüne Haken an der Slave-Nummer signalisiert eine aktive Verbindung. In der Abbildung sind Slaves 0 bis 2 aktiv.



Wenn Sie einen PROFIBUS-Link auswählen, sehen Sie Detail-Informationen zum Status der einzelnen Slaves. Diese Informationen sind vor allem für Support-Zwecke relevant.



2.7.4.2 Moduldiagnose

Wenn Sie im Schnittstellenbaum ein L2B-Request-Modul auswählen, können Sie im Register *Diagnose* Details zu den einzelnen gemessenen Werten und einige Systeminformationen der S7-CPU im Unterregister *S7 System Info* einsehen.

The screenshot displays the Iba I/O Manager interface. On the left, the 'Eingänge' (Inputs) tree shows the hierarchy: ibaFOB-4io-D > ibaL2B-8-8 > Link A > L2B S7 Request A (4), L2B S7 Request B (5), L2B S7 Request Dig512 (6), and Link B > L2B S7 Request (7). The 'L2B S7 Request (7)' module is selected.

The main window shows the 'L2B S7 Request (7)' configuration. The 'Diagnose' tab is active, displaying the 'Analogwerte' (Analog Values) register. The table below shows the measured values:

Name	S7-Operand	Datentyp	Wert	Anzeige-Modus
0 MW 10	MW 10	INT	-14904	DEC
1 MW 12	MW 12	INT	-14136	DEC
2 MW 14	MW 14	INT	16638	DEC
3 MW 16	MW 16	INT	0	DEC
4 MW 5000	MW 5000	INT	I/O-Adresse existiert nicht (6E)	DEC
5 MW 20	MW 20	INT	0	DEC

Below the 'Analogwerte' register, the 'S7 System Info' tab is active, displaying system information:

Version zykl. FC	3.1	Anz. Eingänge	8192
Version Anlauf FC	3.1	Anz. Ausgänge	8192
ID Nummer	1	Anz. Merker	4096
Karten-ID	192	Anz. Timer	512
OS-Version	6	Anz. Zähler	512
PG-Version	1	I/O Bereich	8192
Ges. Speicher	1048576	Lokale Daten	32768
DB Speicher	0		
DB benutzt	0		
Code Speicher	1048576		
Code benutzt	20526		
Auftragsnummer	6ES7 317-2EK14-0AB0		

At the bottom of the window, a progress bar shows the current value 159, with buttons for 'OK', 'Übernehmen', and 'Abbrechen'.

3 Beschreibung der Request-Blöcke

3.1 iba-Baustein-Familie ibaREQ

Diese Blöcke initialisieren und steuern die Kommunikation zwischen *ibaPDA* und der S7-Steuerung.

Die iba-Baustein-Familie ibaREQ erlaubt den Zugriff ausschließlich auf nicht-optimierte Datenbausteine. Die Adressierung erfolgt über die Operandenadresse.

Je Request-Modul (Verbindung) in *ibaPDA* muss ein Satz Request-Blöcke aufgerufen werden. Die verwendeten Bausteine sind Bestandteil der iba S7-Bibliothek, siehe [↗ iba S7-Bibliothek](#), Seite 92.

Hinweis



Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!

Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Verwenden Sie je nach vorhandener Systemkonfiguration unterschiedliche Request-Block-Kombinationen:

Request-Block	S7-300 mit integrierter DP-Schnittstelle	S7-400(H) mit integrierter DP-Schnittstelle und CP443-5	S7-300 mit integrierter PN-Schnittstelle	S7-400 mit integrierter PN-Schnittstelle und CP443-1 (PROFINET)	Empfohlene Aufrufebene
ibaREQ_M (FB140)	X	X	X	X	OB1
ibaREQ_PN (FB141)	X	X	X	X	OB1 oder OB3x
ibaREQ_DP (FB142)	X	X	X	X	OB1 oder OB3x
ibaREQ_DB (DB15)	X	X	X	X	-
ibaREQ_PNdev (FB150)	-	-	X	X	OB1 oder OB3x
ibaREQ_DB-Interface	X	X	X	X	-

■ ibaREQ_M (Management)

Der Baustein realisiert die Kommunikation mit *ibaPDA*. Der Aufruf des Bausteines erfolgt idealerweise im OB1.

■ ibaREQ_PN/ibaREQ_DP (Bereitstellung und Senden der aktuellen Signalwerte)

Der Baustein stellt im Sendezyklus die aktuellen Signalwerte zur Verfügung. Der Aufruf des Bausteines erfolgt idealerweise in einem Weckalarm-OB.

- **ibaREQ-PNdev** (Diagnose des PROFINET Device)
Der Baustein diagnostiziert zyklisch die Verfügbarkeit des genutzten PROFINET Device.
- **ibaREQ_DB** (Schnittstellen-DB)
Dieser DB dient als Schnittstelle zu *ibaPDA* sowie zwischen den verschiedenen Request-Blöcken.

3.1.1 ibaREQ_M (FB140)

Beschreibung der Formalparameter

Name	Art	Typ	Beschreibung
REQ_DB	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle ibaREQ_DB
RESET	IN	BOOL	FALSE: kein Reset (Standard) TRUE: Reset des Bausteins
INP_RANGE	IN	INT	Anzahl der Eingangsbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
OUT_RANGE	IN	INT	Anzahl der Ausgangsbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
MARKER_RANGE	IN	INT	Anzahl der Markerbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
ERROR_STATUS	OUT	DWORD	Interner Fehlercode

Folgende SFCs werden intern verwendet:

- SFC 20 (BLKMOV)
- SFC 21 (FILL)
- SFC 24 (TEST_DB)
- SFC 51 (RDSYSST)

Detaillierte Beschreibung

REQ_DB

Über diesen DB findet der Datenaustausch mit *ibaPDA* statt. Konfigurieren Sie bei allen zusammengehörigen Request-Blöcken den identischen DB.

Länge:

5280 Bytes : bis zu 128 Zeiger (Minimum)

9120 Bytes : bis zu 512 Zeiger

14240 Bytes : bis zu 1024 Zeiger (Maximum)

Es sind auch beliebige andere Längen innerhalb dieser Grenzen zulässig. Die Anzahl der nutzbaren Zeiger wird aus der Länge errechnet.

RESET

Dient zum manuellen Rücksetzen der Request-Blöcke. Es werden alle Request-Blöcke einer Kombination automatisch gemeinsam zurückgesetzt. Der Parameter muss im Regelfall nicht beschaltet werden.

INP_RANGE

Beschränkt die Anzahl der messbaren Eingangsbytes.

Bei INP_RANGE = 0 wird die Größe des verfügbaren Prozessabbildes der Eingänge vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

OUT_RANGE

Beschränkt die Anzahl der messbaren Ausgangsbytes.

Bei OUT_RANGE = 0 wird die Größe des verfügbaren Prozessabbildes der Ausgänge vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

MARKER_RANGE

Beschränkt die Anzahl der messbaren Merkerbytes.

Bei MARKER_RANGE = 0 wird die Anzahl der verfügbaren Merker vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

ERROR_STATUS

Interner Fehlercode des Bausteins. Liegt kein Fehler vor, wird der Wert 0 ausgegeben.

Eine Auflistung aller möglichen Fehlercodes finden Sie unter [Fehlercodes Request-Blöcke](#), Seite 111.

3.1.2 ibaREQ_PN (FB141)**Beschreibung der Formalparameter**

Name	Art	Typ	Beschreibung
REQ_DB	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle ibaREQ_DB
ADR_SLOT	IN	INT	Anfangsadresse des Peripherie-Ausgangsbereiches
DEVICE_STATUS	OUT	INT	Status des PROFINET Device
ERROR_STATUS	OUT	INT	Interner Fehlercode

Folgende SFCs werden intern verwendet:

- SFC 15 (DPWR_DAT)
- SFC 20 (BLKMOV)
- SFC 21 (FILL)
- SFC 50 (RD_LGADR)
- SFC 71 (LOG_GEO)

Detaillierte Beschreibung

REQ_DB

Über diesen DB findet der Datenaustausch mit *ibaPDA* statt. Konfigurieren Sie bei allen zusammengehörigen Request-Blöcken den identischen DB.

ADDR_SLOT

Peripherieanfangsadresse des verwendeten Slots im *ibaBM-PN* im Peripherie-Ausgangsbereich für das Senden von Daten.

DEVICE_STATUS

Status des adressierten PROFINET Device im *ibaBM-PN*.
(0: Device nicht erreichbar/Fehler; 1: Gerät erreichbar)

ERROR_STATUS

Interner Fehlercode des Bausteins. Liegt kein Fehler vor, wird der Wert 0 ausgegeben.

Eine Auflistung aller möglichen Fehlercodes finden Sie unter [↗ Fehlercodes Request-Blöcke](#), Seite 111.

3.1.3 ibaREQ_DP (FB142)

Beschreibung der Formalparameter

Name	Art	Typ	Beschreibung
ADR_SLOT_0	IN	VARIANT	Anfangsadresse des Ausgangsbereiches von Slot 0
ADR_SLOT_1	IN	VARIANT	Anfangsadresse des Ausgangsbereiches von Slot 1
RESET_CON	IN	BOOL	FALSE: kein Reset (Standard) TRUE: Reset der Kommunikationsverbindung
REQ_DB	INOUT	UDT	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle <i>ibaREQ_DB</i>
ERROR_STATUS	OUT	WORD	Interner Fehlercode

Detaillierte Beschreibung

ADR_SLOT_0

Anfangsadresse des verwendeten Slots 0 im *ibaBM-DP* im Ausgangsbereich des Prozessabbilds.

ADR_SLOT_1

Anfangsadresse des verwendeten Slots 1 im *ibaBM-DP* im Ausgangsbereich des Prozessabbilds.

RESET_CON

Dient zum manuellen Rücksetzen der Kommunikationsverbindung.

REQ_DB

Über diesen DB findet der Datenaustausch mit *ibaPDA* statt. Konfigurieren Sie bei allen zusammengehörigen Request-Blöcken den identischen DB.

ERROR_STATUS

Interner Fehlercode des Bausteins. Liegt kein Fehler vor, wird der Wert 0 ausgegeben.

Eine Auflistung aller möglichen Fehlercodes finden Sie unter [↗ Fehlercodes Request-Blöcke](#), Seite 111.

3.1.4 ibaREQ_PNdev (FB150)**Beschreibung der Formalparameter**

Name	Art	Typ	Beschreibung
ADR_SLOT	IN	INT	Anfangsadresse des Peripherie-Ausgangsbereiches
DEVICE_STATUS	OUT	INT	Status des PROFINET Device
ERROR_STATUS	OUT	INT	Interner Fehlercode

Folgende SFCs werden intern verwendet:

- SFC 51 (RDSYSST)
- SFC 71 (LOG_GEO)

Detaillierte Beschreibung**ADDR_SLOT**

Peripherieanfangsadresse des verwendeten Slots im *ibaBM-PN* im Peripherie-Ausgangsbereich für das Senden von Daten.

DEVICE_STATUS

Status des adressierten PROFINET Device im *ibaBM-PN*.
(0: Device nicht erreichbar/Fehler; 1: Gerät erreichbar)

ERROR_STATUS

Interner Fehlercode des Bausteins. Liegt kein Fehler vor, wird der Wert 0 ausgegeben.

Eine Auflistung aller möglichen Fehlercodes finden Sie unter [↗ Fehlercodes Request-Blöcke](#), Seite 111.

3.2 Beschreibung der Request-FCs

Diese Funktionen initialisieren und steuern die Kommunikation zwischen S7-Steuerung (nur SIMATIC S7-300, S7-400, WinAC), *ibaPDA* und *ibaBM-DP*-Gerät bzw. *ibaCom-L2B*-Karte.

Die verwendeten Bausteine sind Bestandteil der iba S7-Bibliothek, siehe [↗ iba S7-Bibliothek](#), Seite 92.

Hinweis

Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!
Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

3.2.1 Request-FC ibaDP_Req (FC122)

Diese Funktion initialisiert und steuert die Kommunikation zwischen der S7-Steuerung, *ibaPDA* und *ibaBM-DP*-Gerät.

Die Funktion ibaDP_Req muss einmal pro Slave im zyklischen Programm aufgerufen werden.

Die verwendeten Bausteine sind Bestandteil der iba S7-Bibliothek, siehe [iba S7-Bibliothek](#), Seite 92.

Hinweis



Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!

Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Beschreibung der Formalparameter des ibaDP_Req (FC122)

Name	Art	Typ	Beschreibung
DB_PDA	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle ibaDP_DB_PDA (2064 Byte)
DB_INTERN	IN	BLOCK_DB	DB der internen Datenschnittstelle ibaDP_DB_work (1900 Byte)
OUTPUT_ADR_SLAVE	IN	INT	Anfangsadresse des Peripherie-Ausgangsbereiches, zusammenhängend und 244 Bytes lang
INIT_FC	IN	BOOL	TRUE: Initialisierung durchführen
INP_RANGE	IN	INT	Anzahl der Eingangsbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
OUT_RANGE	IN	INT	Anzahl der Ausgangsbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
MARKER_RANGE	IN	INT	Anzahl der Merkerbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
ERROR_STATUS_INIT	OUT	BYTE	Fehler bei Initialisierung
ERROR_STATUS_COM	OUT	BYTE	Fehler während der Kommunikation

Folgende SFCs werden intern verwendet:

- SFC 13 (DPNRM_DG)
- SFC 15 (DPRD_DAT)
- SFC 20 (BLKMOV)
- SFC 21 (FILL)
- SFC 24 (TEST_DB)
- SFC 49 (LGC_GADR)
- SFC 50 (RD_LGADR)
- SFC 51 (RDSYSST)

Detaillierte Beschreibung

DB_PDA

DB im Bereich von 1 bis n (siehe Leistungsmerkmale der CPU). Dieser DB wickelt den Datenaustausch mit *ibaPDA* ab. Länge mind. 2064 Bytes.

DB_INTERN

DB im Bereich von 1 bis n (siehe Leistungsmerkmale der CPU).
Länge mind. 1900 Bytes.

In diesem DB werden die folgenden Daten gespeichert:

- Leistungsmerkmale der CPU, die bei der Initialisierung ermittelt werden
- von *ibaPDA* angeforderte S7-Zeiger
- binäre und analoge Sendedaten

OUTPUT_ADR_SLAVE

DP-Anfangsadresse des *ibaBM-DP* im Peripherie-Ausgangsbereich für das Senden von Daten.
Länge 244 Bytes, zusammenhängend (ohne Lücken!).

INIT_FC

Dient zur Initialisierung des Request-Blocks. Die Initialisierung des Request-Blocks wird automatisch intern durchgeführt. Über den Eingang INIT_FC = TRUE kann die Initialisierung durch eine externe Logik manuell zusätzlich durchgeführt werden.

INP_RANGE

Beschränkt die Anzahl der messbaren Eingangsbytes.

Bei INP_RANGE = 0 wird die Größe des verfügbaren Prozessabbildes der Eingänge vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

OUT_RANGE

Beschränkt die Anzahl der messbaren Ausgangsbytes.

Bei OUT_RANGE = 0 wird die Größe des verfügbaren Prozessabbildes der Ausgänge vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

MARKER_RANGE

Beschränkt die Anzahl der messbaren Markerbytes.

Bei MARKER_RANGE = 0 wird die Anzahl der verfügbaren Marker vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

ERROR_STATUS_INIT

Die folgenden Fehlercodes können ausgegeben werden:

- 1: DB_PDA ist schreibgeschützt.
- 2: DB_PDA-Nr = 0 oder größer als die max. zulässige DB-Nummer dieser CPU
- 3: DB_PDA mit der angegebenen Nummer ist nicht vorhanden.
- 5: DB_PDA ist zu kurz.
- 11: DB_INTERN ist schreibgeschützt
- 12: DB_INTERN-Nr = 0 oder größer als die max. zulässige DB-Nummer dieser CPU

- 13: DB_INTERN mit der angegebenen Nummer ist nicht vorhanden.
- 15: DB_INTERN ist zu kurz.
- 16: Fehler beim Lesen der Identifikationsdaten der CPU
- 19: Initialisierung nicht abgeschlossen
- 21: nicht genügend Speicherplatz für den Datensatz
- 22: SZL_ID ist falsch oder unbekannt in dieser CPU
- 23: Der Index der SZL ist falsch oder unzulässig.
- 30: OUT_ADR_SLAVE ist kein Peripherie-Ausgangsbereich.
- 31: OUT_ADR_SLAVE ist keinem PROFIBUS DP zugeordnet.

ERROR_STATUS_COM

Die folgenden Fehlercodes können ausgegeben werden:

- 100: Bitnummer ungleich 0
- 101: Bitnummer ungleich 0-7
- 103: Der Operandenbereich ist nicht definiert.
- 104: Der Datentyp ist nicht definiert.
- 105: DB0 wurde als Datenquelle angefordert.
- 106: DB-Nummer ist höher als die max. zulässige Anzahl DBs dieser CPU.
- 107: DB mit der angegebenen Nummer ist nicht vorhanden.
- 109: DB ist zu kurz für die gewählte Datenadresse.
- 110: Angegebene Datenadresse existiert nicht.
- 111: Initialisierung mit Fehler abgebrochen (Fehlerursache ist im ERROR_STATUS_INIT zu sehen)
- 112: Initialisierung nicht durchgeführt (Fehlerursache ist im ERROR_STATUS_INIT zu sehen)
- 150: Auftragsfragmentierung wird nicht unterstützt.
- 151: Auftrag Request: Anzahl Werte gesamt ungleich Werte im Auftrag
- 152: Auftrag Request: Maximal 64 Binärwerte zulässig
- 153: Auftrag Request: Maximal 64 Analogwerte zulässig
- 200: DP Station ist nicht vorhanden.

3.2.2 Request-FC ibaDP_Req_H (FC123)

Diese Funktion initialisiert und steuert die Kommunikation zwischen der S7-Steuerung, *ibaPDA* und *ibaBM-DP*-Gerät.

Die Funktion ibaDP_Req_H muss einmal pro Slave-Paar im zyklischen Programm aufgerufen werden.

Die verwendeten Bausteine sind Bestandteil der iba S7-Bibliothek, siehe ➔ *iba S7-Bibliothek*, Seite 92.

Hinweis



Verwenden Sie nur Request-Blöcke aus der aktuellen iba S7-Bibliothek!
Request-Blöcke aus Anwendungsbeispielen können veraltet sein und daher zu Fehlern führen.

Beschreibung der Formalparameter des ibaDP_Req_H

Name	Art	Typ	Beschreibung
DB_PDA	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle ibaDP_DB_PDA (2064 Byte)
DB_INTERN	IN	BLOCK_DB	DB der internen Datenschnittstelle ibaDP_DB_work (1900 Byte)
OUTPUT_ADR_SLAVE	IN	INT	Anfangsadresse des Peripherie-Ausgangsbereiches, zusammenhängend und 244 Bytes lang
INIT_FC	IN	BOOL	TRUE: Initialisierung durchführen
INP_RANGE	IN	INT	Anzahl der Eingangsbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
OUT_RANGE	IN	INT	Anzahl der Ausgangsbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
MARKER_RANGE	IN	INT	Anzahl der Merkerbytes (Auswertung nur bei Initialisierung), 0: automatische Erkennung (empfohlen)
ERROR_STATUS_INIT	OUT	BYTE	Fehler bei Initialisierung
ERROR_STATUS_COM	OUT	BYTE	Fehler während der Kommunikation

Folgende SFCs werden intern verwendet:

- SFC 13 (DPNRM_DG)
- SFC 15 (DPRD_DAT)
- SFC 20 (BLKMOV)
- SFC 21 (FILL)
- SFC 24 (TEST_DB)
- SFC 49 (LGC_GADR)
- SFC 50 (RD_LGADR)
- SFC 51 (RDSYSST)

Detaillierte Beschreibung**DB_PDA**

DB im Bereich von 1 bis n (siehe Leistungsmerkmale der CPU). Über diesen DB wird der Datenaustausch mit *ibaPDA* abgewickelt. Länge mind. 2064 Bytes.

DB_INTERN

DB im Bereich von 1 bis n (siehe Leistungsmerkmale der CPU).
Länge mind. 1900 Bytes.

In diesem DB werden die folgenden Daten gespeichert:

- Leistungsmerkmale der CPU, die bei der Initialisierung ermittelt werden
- von *ibaPDA* angeforderte S7-Zeiger
- binäre und analoge Sendedaten

OUTPUT_ADR_SLAVE

DP-Anfangsadresse des *ibaBM-DP* im Peripherie-Ausgangsbereich für das Senden von Daten.
Länge 244 Bytes, zusammenhängend (ohne Lücken!).

INIT_FC

Dient zur Initialisierung des Request-Blocks. Die Initialisierung des Request-Blocks wird automatisch intern durchgeführt. Über den Eingang INIT_FC = TRUE kann die Initialisierung durch eine externe Logik manuell zusätzlich durchgeführt werden.

INP_RANGE

Beschränkt die Anzahl der messbaren Eingangsbytes.

Bei INP_RANGE = 0 wird die Größe des verfügbaren Prozessabbildes der Eingänge vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

OUT_RANGE

Beschränkt die Anzahl der messbaren Ausgangsbytes.

Bei OUT_RANGE = 0 wird die Größe des verfügbaren Prozessabbildes der Ausgänge vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

MARKER_RANGE

Beschränkt die Anzahl der messbaren Merkerbytes.

Bei MARKER_RANGE = 0 wird die Anzahl der verfügbaren Merker vom Request-FB selbst ermittelt (empfohlen). Die Auswertung erfolgt nur während der Initialisierungsphase des Bausteins.

ERROR_STATUS_INIT

Die folgenden Fehlercodes können ausgegeben werden:

- 1: DB_PDA ist schreibgeschützt.
- 2: DB_PDA-Nr = 0 oder größer als die max. zulässige DB-Nummer dieser CPU
- 3: DB_PDA mit der angegebenen Nummer ist nicht vorhanden.
- 5: DB_PDA ist zu kurz
- 11: DB_INTERN ist schreibgeschützt.
- 12: DB_INTERN-Nr = 0 oder größer als die max. zulässige DB-Nummer dieser CPU
- 13: DB_INTERN mit der angegebenen Nummer ist nicht vorhanden.
- 15: DB_INTERN ist zu kurz.
- 16: Fehler beim Lesen der Identifikationsdaten der CPU
- 19: Initialisierung nicht abgeschlossen
- 21: nicht genügend Speicherplatz für den Datensatz

- 22: SZL_ID ist falsch oder unbekannt in dieser CPU.
- 23: Der Index der SZL ist falsch oder unzulässig.
- 30: OUT_ADR_SLAVE ist kein Peripherie-Ausgangsbereich.
- 31: OUT_ADR_SLAVE ist keinem PROFIBUS DP zugeordnet.
- 32: RM²⁾: Die parametrierte "OUTPUT_ADR_SLAVE_BUS_0" ist fehlerhaft.
- 33: RM: Die parametrierte "OUTPUT_ADR_SLAVE_BUS_0" ist keinem PROFIBUS DP-Slave zugeordnet.
- 34: RM: Die parametrierte "OUTPUT_ADR_SLAVE_BUS_1" ist fehlerhaft.
- 35: RM: Die parametrierte "OUTPUT_ADR_SLAVE_BUS_1" ist keinem PROFIBUS DP Slave zugeordnet.
- 36: RM: SLAVE BUS0 und SLAVE BUS1 haben nicht die gleiche DP-Adresse.

ERROR_STATUS_COM

Die folgenden Fehlercodes können ausgegeben werden:

- 100: Bitnummer ungleich 0
- 101: Bitnummer ungleich 0–7
- 103: Der Operandenbereich ist nicht definiert.
- 104: Der Datentyp ist nicht definiert.
- 105: DB0 wurde als Datenquelle angefordert.
- 106: DB-Nummer ist höher als die max. zulässige Anzahl DBs dieser CPU.
- 107: DB mit der angegebenen Nummer ist nicht vorhanden.
- 109: DB ist zu kurz für die gewählte Datenadresse.
- 110: Angegebene Datenadresse existiert nicht.
- 111: Initialisierung mit Fehler abgebrochen (Fehlerursache ist im ERROR_STATUS_INIT zu sehen)
- 112: Initialisierung nicht durchgeführt (Fehlerursache ist im ERROR_STATUS_INIT zu sehen)
- 150: Auftragsfragmentierung wird nicht unterstützt.
- 151: Auftrag Request: Anzahl Werte gesamt ungleich Werte im Auftrag
- 152: Auftrag Request: Maximal 64 Binärwerte zulässig
- 153: Auftrag Request: Maximal 64 Analogwerte zulässig
- 200: DP Station ist nicht vorhanden.
- 201: RM³⁾: Slave Bus 0 ist ausgefallen.
- 202: RM: Slave Bus 1 ist ausgefallen.
- 203: RM: Slaves Bus 0 + 1 sind ausgefallen.
- 210: Ausgangsmodule des Slaves Bus 0 und 1 sind unterschiedlich konfiguriert.

²⁾ RM: Redundanzmodus

³⁾ RM: Redundanzmodus

3.2.3 Initialisierungs-FC **ibaL2B_Init (FC111)**

Diese Funktion fragt Parameter der S7-CPU ab und setzt die Voreinstellungen für die DBs. Die Funktion **ibaL2B_Init (FC111)** muss in allen verfügbaren Anlauf-OBs (OB100, OB101 und OB102) aufgerufen werden.

Beschreibung der Formalparameter des **ibaL2B_Init**

Name	Art	Typ	Beschreibung
DB_PDA	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle ibaL2B_DB_work Bereich: 1 bis n (s. Leistungsmerkmale CPU); jeder Slave benötigt einen eigenen DB.
INP_RANGE	IN	INT	Beschränkt die Anzahl der messbaren Eingangsbytes (CPU-abhängig); empfohlen wird grundsätzlich "0" (automatische Erkennung) vorzugeben. Nur wenn in der "S7 System Info" (<i>ibaPDA</i>) die falsche Anzahl angezeigt wird, können Sie die Anzahl der Eingangsbytes manuell vorgeben! "0": automatische Erkennung (empfohlen)
OUT_RANGE	IN	INT	Beschränkt die Anzahl der messbaren Ausgangsbytes (CPU-abhängig); empfohlen wird grundsätzlich "0" (automatische Erkennung) vorzugeben. Nur wenn in der "S7 System Info" (<i>ibaPDA</i>) die falsche Anzahl angezeigt wird, können Sie die Anzahl der Ausgangsbytes manuell vorgeben! "0": automatische Erkennung (empfohlen)
MARKER_RANGE	IN	INT	Beschränkt die Anzahl der messbaren Merkerbytes (CPU-abhängig); empfohlen wird grundsätzlich "0" (automatische Erkennung) vorzugeben. Nur wenn in der "S7 System Info" (<i>ibaPDA</i>) die falsche Anzahl angezeigt wird, können Sie die Anzahl der Merkerbytes manuell vorgeben! "0": automatische Erkennung (empfohlen)

Name	Art	Typ	Beschreibung
ERROR_STATUS	OUT	BYTE	<p>Parametrierfehler-Status</p> <p>Folgende Fehlercodes können ausgegeben werden:</p> <p>a) Bezüglich Parameter "DB_NUMBER"</p> <p>"11" – DB ist schreibgeschützt (→ DB-Attribut ändern)</p> <p>"12" – DB Nr=0 oder größer als die max. zulässige DB-Nummer dieser CPU (→ in zulässige DB-Nummer ändern)</p> <p>"13" – DB mit der angegebenen Nummer ist nicht vorhanden. (→ DB-22 kopieren/umbenennen)</p> <p>"14" – kein ablaufrelevanter DB (→ iba-Support kontaktieren)</p> <p>"15" – DB zu kurz (→ DB22 kopieren/umbenennen)</p> <p>b) während des Lesens der CPU-Parameter</p> <p>"21" – nicht genügend Speicherplatz für den Datensatz (→ iba-Support kontaktieren)</p> <p>"22" – SZL_ID ist falsch oder unbekannt in dieser CPU. (→ iba-Support kontaktieren)</p> <p>"23" – Index ist falsch oder unzulässig. (→ iba-Support kontaktieren)</p> <p>c) Sonderstatus</p> <p>"19" – Initialisierung nicht abgeschlossen (→ iba-Support kontaktieren)</p>

Folgende SFCs werden intern verwendet:

- SFC 6 (RD_SINFO)
- SFC 24 (TEST_DB)
- SFC 51 (RDSYSST)

Bereiche für Anzahl von Eingängen, Ausgängen und Merkern

Die Anzahl von Eingängen, Ausgängen und Merkern in einer CPU kann die Funktion `ibaL2B_Init` zuverlässig ermitteln. Dazu müssen die Parameter wie im Beispiel oben mit dem Wert "0" beschaltet werden.

In Ausnahmefällen können nach Rücksprache mit dem iba-Support diese Parameter angepasst bzw. geändert werden. Mit Hilfe dieser Parameter werden Fehlprojektierungen bei der Signalauswahl in *ibaPDA* abgefangen. Wenn Sie in *ibaPDA* z. B. einen Merker auswählen, den es in der S7 nicht gibt, so generiert *ibaPDA* eine Fehlermeldung. In der *ibaPDA*-Diagnose des entsprechenden Moduls ist ein solcher Konflikt leicht zu erkennen.

Im Register *Analogwerte* erscheint in der Spalte *Wert* die Meldung "I/O-Adresse existiert nicht (6e)", wenn der Wert nicht dem im `ibaL2B_Init` ermittelten Bereich entspricht.

	Name	S7-Operand	Datentyp	Wert	Anzeige-Modus
0	MW 10	MW 10	INT	-14904	DEC
1	MW 12	MW 12	INT	-14136	DEC
2	MW 14	MW 14	INT	16638	DEC
3	MW 16	MW 16	INT	0	DEC
4	MW 5000	MW 5000	INT	I/O-Adresse existiert nicht (6E)	DEC
5	MW 20	MW 20	INT	0	DEC

In einem solchen Fall überprüfen Sie folgendes:

- Sie haben eine in der CPU verfügbare Operanden-Adresse in *ibaPDA* ausgewählt. Wenn nicht, verwenden Sie einen vorhandenen Operanden.
- Die automatisch ermittelten Adressbereiche im *ibaPDA*-Menü *S7 System Info* mit den Leistungsdaten der S7-CPU übereinstimmen. Falls nicht, geben Sie beim Parametrieren des `ibaL2B_Init` die Anzahl der E-, A-, und M-Bytes manuell vor (aus den Leistungsdaten der S7-CPU).

3.2.4 Kommunikations-FC **ibaL2B_Req_CP (FC113)** für CP342-5

Nicht jeder S7-300 CPU-Typ hat eine integrierte bzw. freie (nicht verwendete) DP-Schnittstelle. In der S7-400-Familie kann wie bisher beschrieben auch ein externer CP mit den Funktionen **ibaL2B_Init** (FC111) und **ibaL2B_Req** (FC112) angesprochen werden. Der externe CP hängt hier am so genannten "Kommunikationsbus" und verhält sich wie eine interne DP-Schnittstelle!

Nicht ganz so einfach geht das bei CPUs der S7-300er Familie. Diese besitzen keinen "Kommunikationsbus". Die Anbindung der iba-Slaves erfolgt über die Standardbausteine FC1 (DP_SEND) und FC2 (DP_RCV) aus der Standard-Bibliothek von S7 (und nicht über den Peripherieadressraum). Die physikalische Ankopplung erfolgt über den externen CP342-5. Der **ibaL2B_Req** (FC112) wird für diesen Fall durch den **ibaL2B_Req_CP** (FC113) ersetzt. Des Weiteren wird für alle Slave-Verbindungen gemeinsam ein weiterer DB **ibaL2B_CP_SNDRCV** (im Beispiel DB10) benötigt, in dem die Peripheriedaten aller Slaves zwischengespeichert werden.

Die Funktion **ibaL2B_Req_CP** (FC113) muss anstelle des **ibaL2B_Req** (FC112) einmal pro Slave im zyklischen Programm aufgerufen werden.

Beschreibung der Formalparameter des **ibaL2B_Req_CP (FC113)**

Name	Art	Typ	Beschreibung
DB_PDA	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle ibaL2B_DB_work ; Bereich: 1 bis n (s. Leistungsmerkmale CPU); jeder Slave benötigt einen eigenen DB. Der DB muss identisch sein mit dem DB beim Aufruf des Blocks ibaL2B_Init .
DB_DP_Data	IN	BLOCK_DB	DB, in dem die Sende- und Empfangsdaten für FC_SEND und FC_RECV zwischengepuffert werden. Bereich: 1 bis n (s. Leistungsmerkmale CPU); es können die Daten aller Slaves in einem DB gesammelt werden.
DB_ADR_Offset_SEND	IN	INT	DB-OFFSET-Adresse des DB-Bereichs für die SEND-Daten (pro Slave werden 122 Byte belegt). Jeder Slave muss eine eigene OFFSET-Adresse für seine SEND-Daten haben.
DB_ADR_Offset_RECV	IN	INT	DB-OFFSET-Adresse des DB-Bereichs für die RECEIVE-Daten (pro Slave werden 16 Byte belegt). Jeder Slave muss eine eigene OFFSET-Adresse für seine RECEIVE-Daten haben.
TIME_OUT_VAL	IN	INT	Timeout (Zeitüberschreitung) der Kommunikation $\text{Timeout} = \text{TIME_OUT_VAL} \times 100 \text{ ms}$
TIMER_NR	IN	Timer	Timer für den internen (Warte-) Zeitgenerator. Bereich: T0 bis Tn (s. Leistungsmerkmale CPU); jeder Slave benötigt einen eigenen Timer.
TIME_OUT_FLAG	OUT	BOOL	Verbindungsabbruch zwischen S7 und <i>ibaCom-L2B</i> -Karte

Name	Art	Typ	Beschreibung
ERROR_STATUS	OUT	BYTE	<p>Parametrierfehler</p> <p>Folgende Fehlercodes können ausgegeben werden:</p> <p>"92" – Ungültiger Befehl von <i>ibaPDA</i> (→ iba-Support kontaktieren)</p> <p>"100" – zu wenig Platz im DB-Adressraum für die angeforderten Analogwerte (→ Datenmenge in <i>ibaPDA</i> auf max. 112 Byte reduzieren)</p> <p>"101" – Kanalnummer > 31 (→ iba-Support kontaktieren)</p> <p>"102" – Datentyp wird nicht unterstützt (nur Analogwerte) (→ iba-Support kontaktieren)</p> <p>"103" – Bereichskennung fehlt oder nicht interpretierbar (→ iba-Support kontaktieren)</p> <p>"104" – Dieser Variablentyp kann nicht interpretiert werden. (→ iba-Support kontaktieren)</p> <p>"105" – DB 0 wurde als Datenquelle angefordert. (→ anderen zulässigen DB verwenden)</p> <p>"106" – DB-Nummer ist höher als die max. zulässige Anzahl DBs dieser CPU. (→ in zulässige DB-Nummer ändern)</p> <p>"107" – DB mit der angegebenen Nummer ist nicht vorhanden. (→ vorhandene DB-Daten anfordern)</p> <p>"109" – DB zu kurz für die gewählte Datenadresse (→ DB22 kopieren/umbenennen)</p> <p>"110" – Operandenadresse (E,A,M,P) existiert nicht. (→ vorhandene Operanden anfordern)</p> <p>"111" – Initialisierung mit Fehler abgebrochen (→ Error-Status von <i>ibaL2B_Init</i> auswerten, ansonsten iba-Support kontaktieren)</p> <p>"112" – Initialisierung nicht bearbeitet (→ <i>ibaL2B_Init</i> in den Anlauf-OBs aufrufen)</p>

3.2.5 Kommunikations-FC **ibaL2B_Req (FC112)**

Diese Funktion steuert die Kommunikation zwischen der S7 und der *ibaCom-L2B*-Karte im *ibaPDA*-Rechner. Sie dient außerdem zur Überwachung der Kommunikationsverbindungen.

Die Funktion *ibaL2B_Req* (FC112) muss einmal pro Slave im zyklischen Programm aufgerufen werden.

Beschreibung der Formalparameter des **ibaL2B_Req (FC112)**

Name	Art	Typ	Beschreibung
DB_PDA	IN	BLOCK_DB	DB der <i>ibaPDA</i> -Kommunikationsschnittstelle <i>ibaL2B_DB_work</i> ; Bereich: 1 bis n (s. Leistungsmerkmale CPU); jeder Slave benötigt einen eigenen DB. Der DB muss identisch sein mit dem DB beim Aufruf des Blocks <i>ibaL2B_Init</i> .
DP_SEND_ADR	IN	INT	DP-Anfangsadresse der <i>ibaCom-L2B</i> -Karte im Periphere-Ausgabebereich für das Senden von Daten
DP_RECEIVE_ADR	IN	INT	DP-Anfangsadresse der <i>ibaCom-L2B</i> -Karte im Periphere-Eingangsbereich für den Empfang von Daten
TIME_OUT_VAL	IN	INT	Timeout (Zeitüberschreitung) der Kommunikation Timeout = TIME_OUT_VAL × 100 ms
TIMER_NR	IN	Timer	Timer für den internen (Warte-) Zeitgenerator. Bereich: T0 bis Tn (s. Leistungsmerkmale CPU); jeder Slave benötigt einen eigenen Timer
TIME_OUT_FLAG	OUT	BOOL	Verbindungsabbruch zwischen S7 und <i>ibaCom-L2B</i> -Karte

Name	Art	Typ	Beschreibung
ERROR_STATUS	OUT	BYTE	<p>Parametrierfehler</p> <p>Folgende Fehlercodes können ausgegeben werden:</p> <p>"92" – Ungültiger Befehl von <i>ibaPDA</i> (→ iba-Support kontaktieren)</p> <p>"100" – zu wenig Platz im DB-Adressraum für die angeforderten Analogwerte (→ Datenmenge in <i>ibaPDA</i> auf max. 112 Byte reduzieren)</p> <p>"101" – Kanalnummer > 31 (→ iba-Support kontaktieren)</p> <p>"102" – Datentyp wird nicht unterstützt (nur Analogwerte) (→ iba-Support kontaktieren)</p> <p>"103" – Bereichskennung fehlt oder nicht interpretierbar (→ iba-Support kontaktieren)</p> <p>"104" – Dieser Variablentyp kann nicht interpretiert werden. (→ iba-Support kontaktieren)</p> <p>"105" – DB 0 wurde als Datenquelle angefordert. (→ anderen zulässigen DB verwenden)</p> <p>"106" – DB-Nummer ist höher als die max. zulässige Anzahl DBs dieser CPU. (→ in zulässige DB-Nummer ändern)</p> <p>"107" – DB mit der angegebenen Nummer ist nicht vorhanden. (→ vorhandene DB-Daten anfordern)</p> <p>"109" – DB zu kurz für die gewählte Datenadresse (→ DB22 kopieren/ umbenennen)</p> <p>"110" – Operandenadresse (E,A,M,P) existiert nicht. (→ vorhandene Operanden anfordern)</p> <p>"111" – Initialisierung mit Fehler abgebrochen (→ Error-Status von <i>ibaL2B_Init</i> auswerten, ansonsten iba-Support kontaktieren)</p> <p>"112" – Initialisierung nicht bearbeitet (→ <i>ibaL2B_Init</i> in den Anlauf-OBs aufrufen)</p>

Folgende SFCs werden intern verwendet:

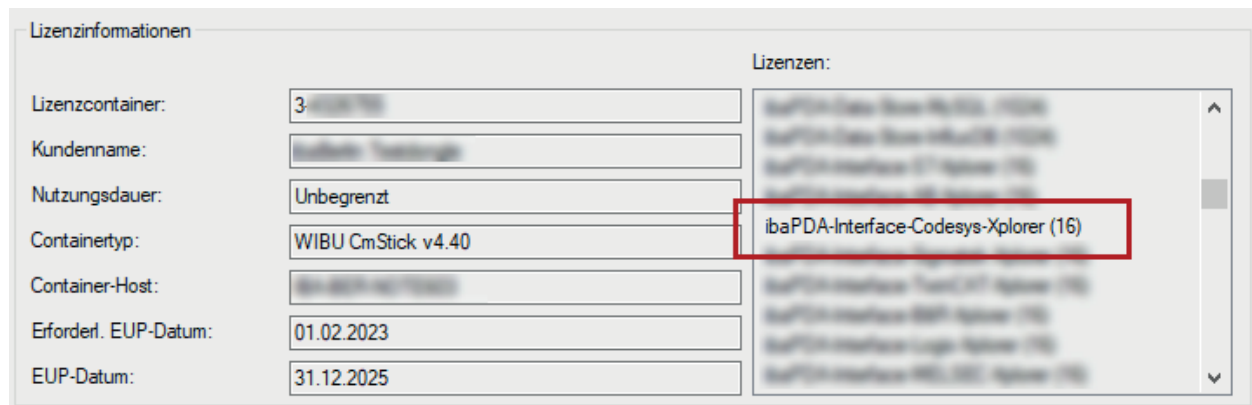
- SFC 20 (BLKMOV)
- SFC 24 (TEST_DB)
- SFC 36 (MSK_FLT)
- SFC 37 (DMSK_FLT)
- SFC 50 (RD_LGADR)

4 Diagnose

4.1 Lizenz

Falls die gewünschte Schnittstelle nicht im Signalbaum angezeigt wird, können Sie entweder in *ibaPDA* im I/O-Manager unter *Allgemein – Einstellungen* oder in der *ibaPDA* Dienststatus-Applikation überprüfen, ob Ihre Lizenz für diese Schnittstelle ordnungsgemäß erkannt wird. Die Anzahl der lizenzierten Verbindungen ist in Klammern angegeben.

Die folgende Abbildung zeigt beispielhaft die Lizenz für die Schnittstelle *Codesys-Xplorer*.



4.2 Sichtbarkeit der Schnittstelle

Ist die Schnittstelle trotz gültiger Lizenz nicht zu sehen, ist sie möglicherweise verborgen.

Überprüfen Sie die Einstellung im Register *Allgemein* im Knoten *Schnittstellen*.

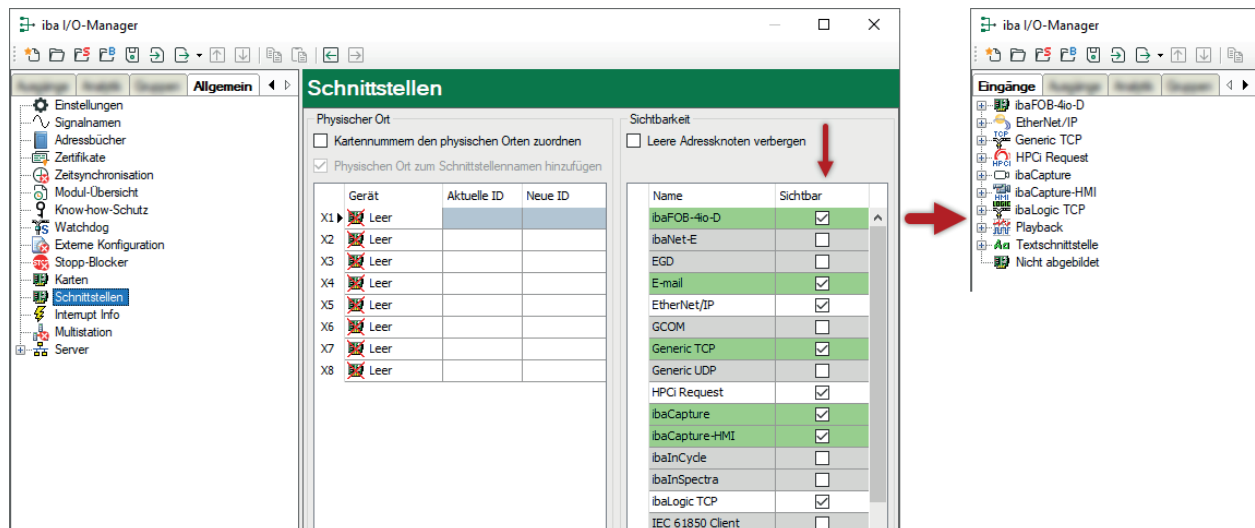
Sichtbarkeit

Die Tabelle *Sichtbarkeit* listet alle Schnittstellen auf, die entweder durch Lizenzen oder installierte Karten verfügbar sind. Diese Schnittstellen sind auch im Schnittstellenbaum zu sehen.

Mithilfe der Häkchen in der Spalte *Sichtbar* können Sie nicht benötigte Schnittstellen im Schnittstellenbaum verbergen oder anzeigen.

Schnittstellen mit konfigurierten Modulen sind grün hinterlegt und können nicht verborgen werden.

Ausgewählte Schnittstellen sind sichtbar, die anderen Schnittstellen sind verborgen:



4.3 Protokolldateien

Wenn Verbindungen zu Zielsystemen bzw. Clients hergestellt wurden, dann werden alle verbindungs-spezifischen Aktionen in einer Textdatei protokolliert. Diese (aktuelle) Datei können Sie z. B. nach Hinweisen auf mögliche Verbindungsprobleme durchsuchen.

Die Protokolldatei können Sie über den Button <Protokolldatei öffnen> öffnen. Der Button befindet sich im I/O-Manager:

- bei vielen Schnittstellen in der jeweiligen Schnittstellenübersicht
- bei integrierten Servern (z. B. OPC UA-Server) im Register Diagnose.

Im Dateisystem auf der Festplatte finden Sie die Protokolldateien von *ibaPDA*-Server (... \ProgramData\iba\ibaPDA\Log). Die Dateinamen der Protokolldateien werden aus der Bezeichnung bzw. Abkürzung der Schnittstellenart gebildet.

Dateien mit Namen `Schnittstelle.txt` sind stets die aktuellen Protokolldateien. Dateien mit Namen `Schnittstelle_yyyy_mm_dd_hh_mm_ss.txt` sind archivierte Protokolldateien.

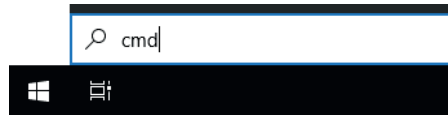
Beispiele:

- `ethernetipLog.txt` (Protokoll von EtherNet/IP-Verbindungen)
- `AbEthLog.txt` (Protokoll von Allen-Bradley-Ethernet-Verbindungen)
- `OpcUAServerLog.txt` (Protokoll von OPC UA-Server-Verbindungen)

4.4 Verbindungsdiagnose mittels PING

Ping ist ein System-Befehl, mit dem Sie überprüfen können, ob ein bestimmter Kommunikationspartner in einem IP-Netzwerk erreichbar ist.

1. Öffnen Sie eine Windows Eingabeaufforderung.



2. Geben Sie den Befehl "ping" gefolgt von der IP-Adresse des Kommunikationspartners ein und drücken Sie <ENTER>.

→ Bei bestehender Verbindung erhalten Sie mehrere Antworten.

```
Administrator: Eingabeaufforderung
Microsoft Windows [Version 10.0]
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>ping 192.168.1.10

Ping wird ausgeführt für 192.168.1.10 mit 32 Bytes Daten:
Antwort von 192.168.1.10: Bytes=32 Zeit=1ms TTL=30
Antwort von 192.168.1.10: Bytes=32 Zeit<1ms TTL=30
Antwort von 192.168.1.10: Bytes=32 Zeit<1ms TTL=30
Antwort von 192.168.1.10: Bytes=32 Zeit<1ms TTL=30

Ping-Statistik für 192.168.1.10:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0
    (0% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 1ms, Mittelwert = 0ms

C:\Windows\system32>
```

→ Bei nicht bestehender Verbindung erhalten Sie Fehlermeldungen.

```
Administrator: Eingabeaufforderung
Microsoft Windows [Version 10.0]
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>ping 192.168.1.10

Ping wird ausgeführt für 192.168.1.10 mit 32 Bytes Daten:
Antwort von 192.168.1.10: Zielhost nicht erreichbar.
Zeitüberschreitung der Anforderung.
Zeitüberschreitung der Anforderung.
Zeitüberschreitung der Anforderung.

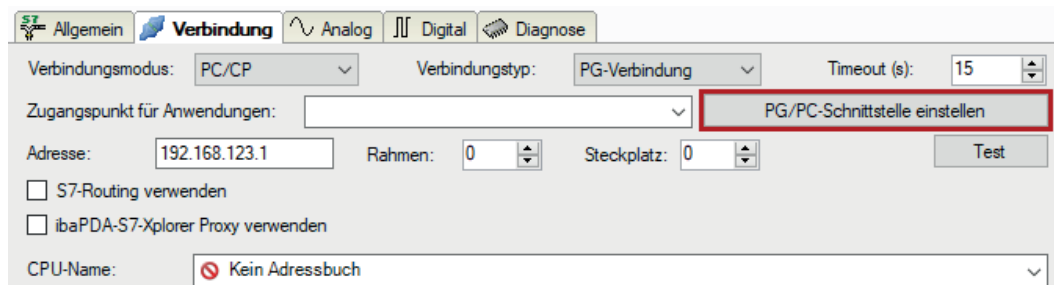
Ping-Statistik für 192.168.1.10:
    Pakete: Gesendet = 4, Empfangen = 1, Verloren = 3
    (75% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 1ms, Mittelwert = 0ms

C:\Windows\system32>
```

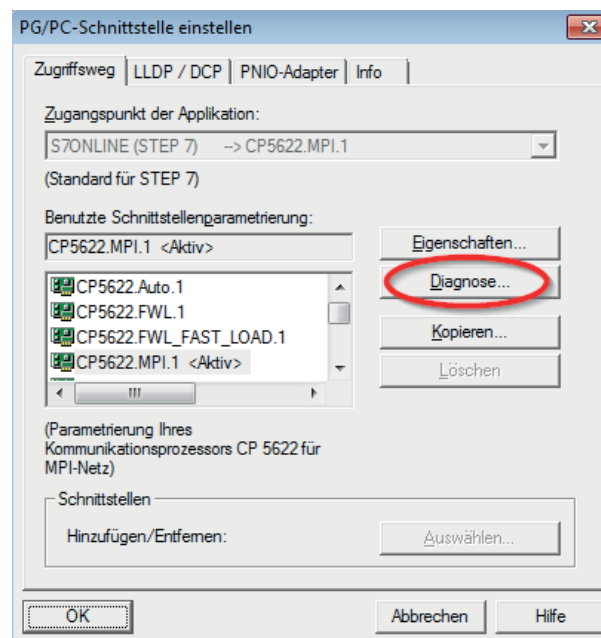
4.5 Verbindungsdiagnose mittels PG/PC-Schnittstelle

Mit der Diagnosefunktion der PG/PC-Schnittstelle können Sie die Funktionsfähigkeit und die Verbindungskonfiguration testen.

1. Öffnen Sie mit dem Button <PG/PC-Schnittstelle einstellen> den Dialog zur Einstellung der PG/PC-Schnittstelle.



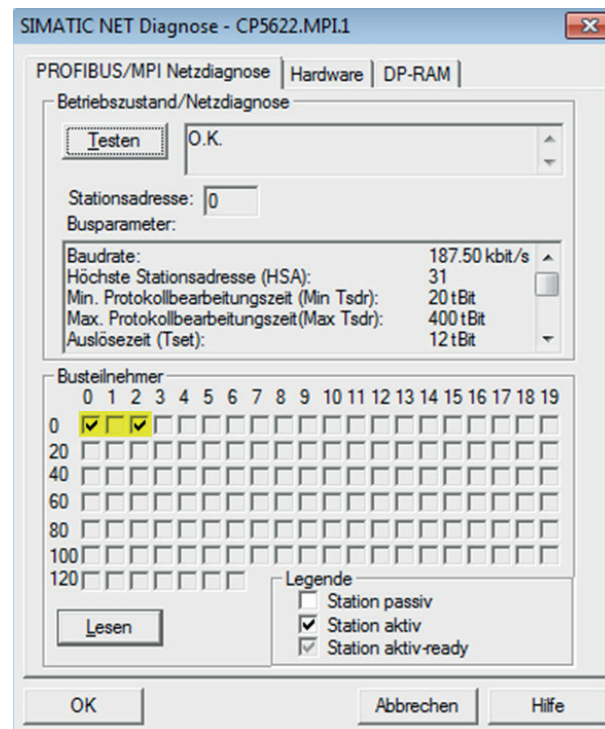
2. Öffnen Sie mit dem Button <Diagnose> den Diagnose-Dialog.



3. Starten Sie mit dem Button <Testen> eine Netzdiagnose.
Prüfen Sie mit dem Button <Lesen> die Erreichbarkeit der Busteilnehmer.

→ Die folgende Abbildung zeigt beispielhaft eine Diagnose einer SIMATIC Net CP5622 (PROFIBUS).

Auf den Adressen 0 und 2 ist jeweils eine aktive Station.



4.6 Moduldiagnose

Eine zusätzliche Diagnosehilfe mit einer tabellarischen Anzeige der analogen und digitalen Istwerte sowie der Datentypen finden Sie im Register *Diagnose* jedes S7 Request-Moduls.

Siehe Kapitel Request-S7 für ibaBM-PN, ➔ *Diagnose*, Seite 20

Siehe Kapitel Request-S7 für ibaBM-DP, ➔ *Diagnose*, Seite 38

Siehe Kapitel Request-S7 für ibaBM-DPM-S, ➔ *Diagnose*, Seite 51

Siehe Kapitel Request-S7 für ibaCom-L2B, ➔ *Diagnose*, Seite 67

5 Anhang

5.1 iba S7-Bibliothek

Die iba S7-Bibliothek liegt in zwei Varianten vor:

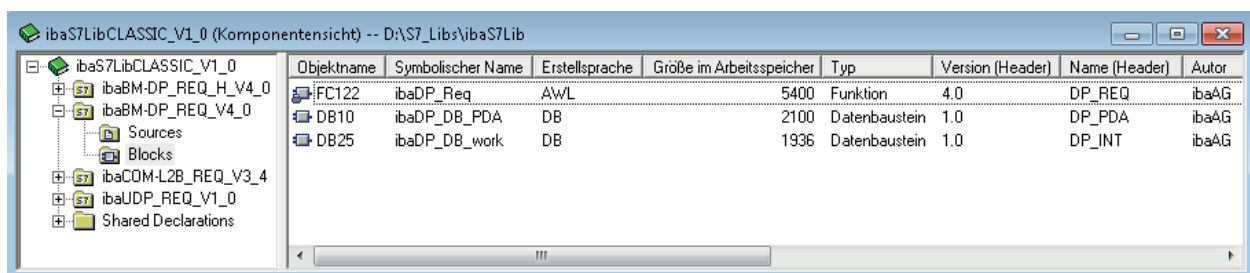
- SIMATIC Manager: STEP7 ≥ V5.5
- SIMATIC TIA-Portal: STEP 7 ≥ V16

5.1.1 iba S7-Bibliothek für SIMATIC Manager

Die iba S7-Bibliothek für SIMATIC Manager ("ibaS7LibCLASSIC_Vx_y") ist geeignet für die Verwendung mit SIMATIC Manager V5.5 oder höher. Sie enthält die im Handbuch beschriebenen Request-Blöcke, welche für die Nutzung von *ibaPDA-Request-S7-DP/PN/ibaNet-E* erforderlich sind.

Sie finden die iba S7-Bibliothek als Archivdatei auf dem Datenträger "iba Software & Manuals" in folgendem Verzeichnis:

`\04_Libraries_and_Examples\10_Libraries\01_SIMATIC_S7\`



Enthalten sind folgende Bausteine:

iba-Anschaltung	Bausteinname	Bausteinnr.	Bemerkung
ibaBM-DP ibaBM-DPM-S	ibaDP_Req	FC122	
	ibaDP_DB_PDA	DB10	
	ibaDP_DB_work	DB25	
ibaBM-DP ibaBM-DPM-S Redundanzmodus	ibaDP_Req_H	FC123	für S7-400H
	ibaDP_DB_PDA	DB10	
	ibaDP_DB_work	DB25	
ibaBM-PN	ibaREQ_M	FB140	
	ibaREQ_PN	FB141	
	ibaREQ_PNdev	FB150	
	ibaREQ_DB	DB15	
	ibaUDT_UDPact	UDT145	

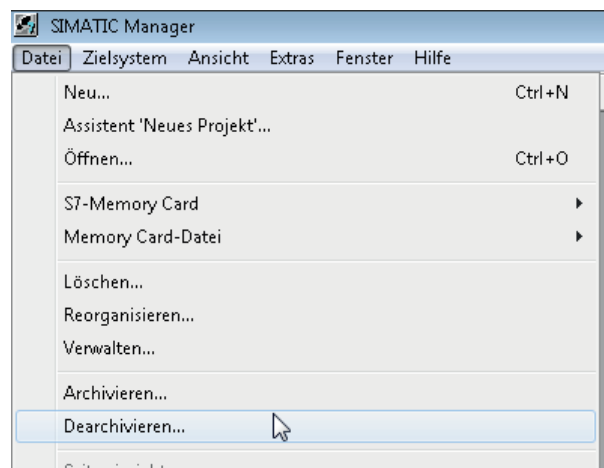
iba-Anschaltung	Bausteinname	Bausteinnr.	Bemerkung
ibaCom-L2B	ibaL2B_Init	FC111	ehemals FC23 und FC101
	ibaL2B_Req	FC112	ehemals FC22 und FC100
	ibaL2B_Req_CP	FC113	ehemals FC26 und FC102 nur nötig bei Verwendung eines CP342-5 anstelle des FC112
	ibaL2B_DB_work	DB22	
	ibaL2B_DB_Struct	UDT22	
	ibaL2B_CP_SNDRCV	DB10	nur nötig bei Verwendung eines CP342-5
ibaPDA-Interface-S7- TCP/UDP	ibaREQ_M	FB140	
	ibaREQ_UDPact	FB145	
	ibaREQ_UDPint	FB146	
	ibaREQ_UDPext3	FB147	
	ibaREQ_UDPext4	FB148	
	ibaREQ_DB	DB15	
	ibaUDT_UDPact	UDT145	

Tab. 1: ibaS7LibCLASSIC Bausteinübersicht

5.1.1.1 Bibliothek in SIMATIC Manager einbinden

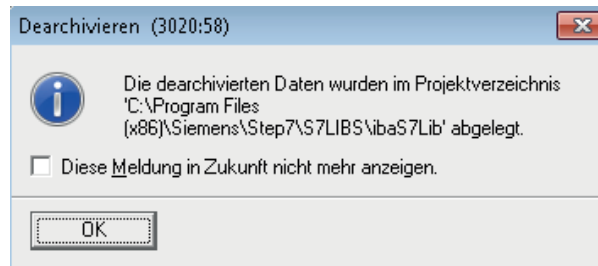
Um die Bibliothek einzubinden, müssen Sie die Bibliothek im SIMATIC Manager dearchivieren. Kopieren Sie die iba S7-Bibliothek in ein lokales Verzeichnis Ihres Rechners, auf dem SIMATIC Manager ausgeführt wird.

1. Wählen Sie den Menüpunkt *Datei – Dearchivieren*.

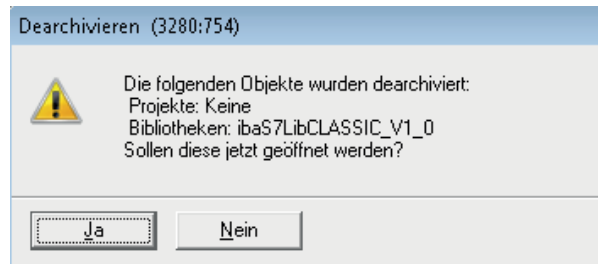


2. Wählen Sie die Archiv-Datei der iba S7-Bibliothek aus und wählen Sie im nächsten Schritt einen Ablageort für die extrahierte Bibliothek.

3. Bestätigen Sie die Meldung zur erfolgreichen Dearchivierung.



4. Öffnen Sie die Bibliothek indem Sie den folgenden Dialog mit <Ja> bestätigen.

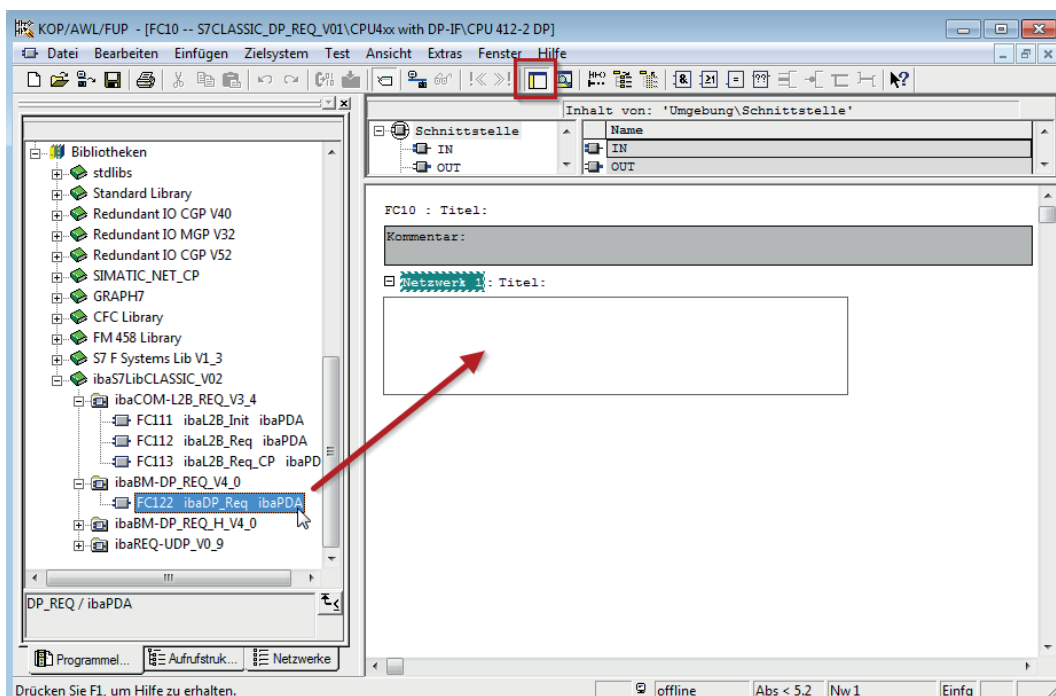


→ Nun ist die Bibliothek eingebunden und kann wieder geschlossen werden.

5.1.1.2 Bausteine in SIMATIC Manager übernehmen

Sie haben zwei Möglichkeiten, die Bausteine aus der Bibliothek zu übernehmen:

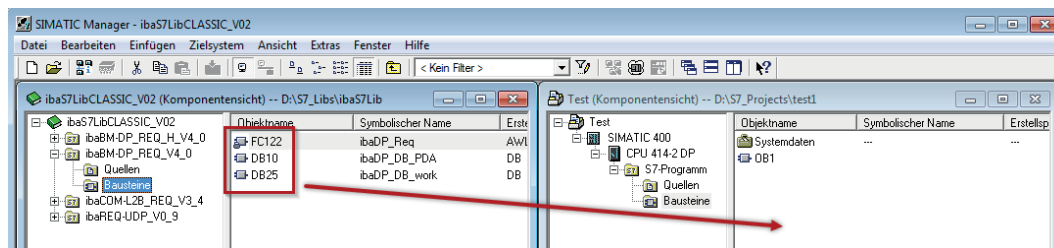
- Blenden Sie die Bausteinbibliothek ein und ziehen Sie die gewünschten Bausteine in den geöffneten Ziel-Baustein.



- Öffnen Sie die Bibliothek über *Datei – Öffnen – Bibliotheken*, sowie das gewünschte Ziel-Projekt parallel.

Mit dem Button <Anordnen horizontal> können Sie beide Projekte nebeneinander darstellen.

Die Bausteine können Sie nun entweder ziehen oder kopieren.



→ Sie können die Bausteine nun im Zielprojekt verwenden.

5.2 Anwendungsbeispiele

Anwendungsbeispiele für verschiedene Konfigurationen finden Sie auf dem Datenträger "iba Software & Manuals".

- Für das Gerät ibaBM-DPM-S im Verzeichnis
`\04_Libraries_and_Examples\30_ibaBM-DPM-S\01_SIMATIC_S7\Request-S7\`
- Für das Gerät ibaBM-DP im Verzeichnis
`\04_Libraries_and_Examples\31_ibaBM-DP\01_SIMATIC_S7\Request-S7\`
- Für das Gerät ibaBM-PN im Verzeichnis
`\04_Libraries_and_Examples\32_ibaBM-PN\01_SIMATIC_S7\Request-S7\`
- Für die Karte ibaCom-L2B im Verzeichnis
`\04_Libraries_and_Examples\40_ibaCOM-L2B\01_SIMATIC_S7\Request-S7\`
- Für die Schnittstelle ibaPDA-Interface-ibaNet-E
`\04_Libraries_and_Examples\53_ibaPDA-Interface-ibaNet-E\01_SIMATIC_S7\Request-S7\`

Für folgende Konfigurationen sind Beispiele vorhanden:

iba	S7-CPU	S7-Projekt	ibaPDA-Projekt
ibaBM-PN	S7-300 PN-IF	S7CLASSIC_ PN_REQ_Vxx.zip	ibaPDA_S7CLASSIC_ PN_REQ_Vxx.zip
	S7-400 PN-IF		
	S7-1500 PN-IF	S7TIA_ PN_REQ_Vxx.zip	ibaPDA_S7TIA_ PN_REQ_Vxx.zip
		ibaS7TIA_PN_REQsym_ Vxx.zap*	ibaPDA_S7TIA_PN_REQsym
ibaBM-DP	S7-300 DP-IF	S7CLASSIC_ DP_REQ_Vxx.zip	ibaPDA_S7CLASSIC_ DP_REQ_CPU3xx_Vxx.zip
	S7-400 DP-IF		ibaPDA_S7CLASSIC_ DP_REQ_CPU4xx_Vxx.zip
	S7-400 + CP443-5		ibaPDA_S7CLASSIC_ DP_REQ_CPU4xx_Vxx.zip
	WinAC		ibaPDA_S7CLASSIC_ DP_REQ_WinAC_Vxx.zip
	S7-400 mit CFC	S7CLASSIC_ DP_REQ_CFC_Vxx.zip	ibaPDA_S7CLASSIC_ DP_REQ_CFC_Vxx.zip
	S7-400H	S7CLASSIC_ DP_REQ_H_Vxx.zip	ibaPDA_S7CLASSIC_ DP_REQ_CPU4xxH_Vxx.zip
	S7-300 DP-IF	S7CLASSIC_ L2B_REQ_Vxx.zip	ibaPDA_S7CLASSIC_ DP_L2Bcomp_REQ_Vxx.zip
	S7-1500 DP-IF	S7TIA_ DP_REQ_Vxx.zip	ibaPDA_S7TIA_ DP_REQ_Vxx.zip
ibaBM-DPM-S	S7-300 DP-IF	S7CLASSIC_ DP_REQ_Vxx.zip	ibaPDA_S7CLASSIC_ DPMS_REQ_CPU3xx_Vxx.zip
	S7-400 DP-IF		ibaPDA_S7CLASSIC_ DPMS_REQ_CPU4xx_Vxx.zip
	WinAC		ibaPDA_S7CLASSIC_ DPMS_REQ_WinAC_Vxx.zip
	S7-400 mit CFC	S7CLASSIC_ DP_REQ_CFC_Vxx.zip	ibaPDA_S7CLASSIC_ DPMS_REQ_CFC_Vxx.zip
	S7-400H	S7CLASSIC_ DP_REQ_H_Vxx.zip	ibaPDA_S7CLASSIC_ DPMS_REQ_CPU4xxH_Vxx.zip
ibaCom-L2B	S7-300 DP-IF	S7CLASSIC_ L2B_REQ_Vxx.zip	ibaPDA_S7CLASSIC_ L2B_REQ_Vxx.zip
	S7-300 + CP342-5		
	S7-400 mit CFC	S7CLASSIC_ L2B_REQ_CFC_Vxx.zip	ibaPDA_S7CLASSIC_ L2B_REQ_CFC_Vxx.zip
ibaPDA-Interface-ibaNet-E	S7-1500	ibaS7TIA_NetE_REQ_Vxx.zap*	ibaPDA_S7TIA_NetE_REQ
		ibaS7TIA_NetE_REQsym_Vxx.zap*	ibaPDA_S7TIA_NetE_REQsym

Tab. 2: Konfigurationsbeispiele auf Datenträger

5.3 S7-Zykluszeitmessungen

5.3.1 ibaCom-L2B

Die nachfolgenden Tabellen geben Aufschluss darüber, mit welcher Zykluszeitverlängerung Sie bei Einsatz von *ibaPDA* in Verbindung mit L2B-Request S7-CPU-abhängig rechnen können.

Bei der Ermittlung der Zeitwerte wurden Testreihen mit bis zu 256 analogen und 256 digitalen Signalen (8 Slaves) gefahren. Die Messwerte wurden in einer Testumgebung ermittelt und geben nur Anhaltspunkte wieder. Die Werte können in anderen Systemumgebungen abweichen.

S7-CPUs mit integrierter DP-Schnittstelle

S7-CPU-Typ	Max. Zykluszeiterhöhung pro Slave (32 analoge+ 32 digitale Signale)
316- 2AG00	13 ms
314- 6CF00	8,63 ms
315- 2AG10	6,88 ms
317- 2EJ10	0,875 ms
317- 6FF00	1,125 ms
318- 2AG00	0,75 ms
416- 2XK00	1,25 ms
413- 2XG02	2,625 ms
414- 2XG03	0,875 ms
416- 2XK04	0,375 ms

S7-CPUs mit externer DP-Schnittstelle (CP)

S7-CPU-Typ/ CP	Max. Zykluszeiterhöhung pro Slave (32 analoge+ 32 digitale Signale)
CPU315 mit CP342- 5	6,0 ms
CPU318 mit CP342- 5	0,75 ms
CPU416 mit CP443- 5	0,625 ms

Hinweis



Bei Verwendung des externen PROFIBUS-CPs CP342-5 bei der S7-300 ist in der Regel keine zyklusgenaue Messung möglich, da die Datenübertragung über den Rückwandbus der S7-300 zu langsam ist. Ansonsten ist bei ausreichend schnellem PROFIBUS-Zyklus eine zyklusgenaue Messung möglich.

5.3.2 ibaBM-DP

Die nachfolgenden Tabellen geben Aufschluss darüber, welche Codelaufzeiten die Request-Blöcke beim Einsatz von *ibaPDA* in Verbindung mit Request-S7 für *ibaBM-DP* benötigen.

Die Messwerte wurden in einer Testumgebung ermittelt und geben nur Anhaltspunkte wieder. Die Werte können in anderen Systemumgebungen abweichen.

SIMATIC S7-CPU	Signalanzahl	Datenmenge	ibaDP_Req FC122
CPU412-2 PN 6ES7 412-2EK06-0AB0	1 INT + 0 BOOL (1 Zeiger)	2 Byte	564 µs
	59 REAL + 64 BOOL (2 Zeiger)	244 Byte	1614 µs
	59 REAL + 64 BOOL (123 Zeiger)	244 Byte	1632 µs

SIMATIC S7-CPU	Signalanzahl	Datenmenge	ibaREQ_M FB1400	ibaREQ_DP FB1402
CPU1516-3 PN/DP 6ES7 516-3AN00-0AB0	1 INT + 0 BOOL (1 Zeiger)	2 Byte	192 µs	286 µs
	59 REAL + 64 BOOL (2 Zeiger)	244 Byte	194 µs	296 µs
	59 REAL + 64 BOOL (123 Zeiger)	244 Byte	192 µs	834 µs
	122 INT + 0 BOOL (1 Zeiger)	244 Byte	195 µs	287 µs
	122 INT + 0 BOOL (122 Zeiger)	244 Byte	198 µs	706 µs

5.3.3 ibaBM-PN

Die nachfolgenden Tabellen geben Aufschluss darüber, welche Codelaufzeiten die Request-Blöcke beim Einsatz von *ibaPDA* in Verbindung mit Request-S7 für *ibaBM-PN* benötigen.

Die Messwerte wurden in einer Testumgebung ermittelt und geben nur Anhaltspunkte wieder. Die Werte können in anderen Systemumgebungen abweichen.

SIMATIC S7-CPU	Signalanzahl	Datenmenge	ibaREQ_M FB140	ibaREQ_PN FB141
CPU412-2 PN 6ES7 412-2EK06-0AB0	1 INT + 0 BOOL (1 Zeiger)	2 Byte	128 µs	302 µs
	59 REAL + 64 BOOL (2 Zeiger)	244 Byte	126 µs	376 µs
	59 REAL + 64 BOOL (123 Zeiger)	244 Byte	132 µs	937 µs
	122 INT + 0 BOOL (1 Zeiger)	244 Byte	126 µs	342 µs
	122 INT + 0 BOOL (122 Zeiger)	244 Byte	132 µs	954 µs

SIMATIC S7-CPU	Signalanzahl	Datenmenge	ibaREQ_M FB1400	ibaREQ_PN FB1401
CPU1516-3 PN/DP 6ES7 516-3AN00-0AB0	1 INT + 0 BOOL (1 Zeiger)	2 Byte	198 µs	276 µs
	59 REAL + 64 BOOL (2 Zeiger)	244 Byte	194 µs	282 µs
	59 REAL + 64 BOOL (123 Zeiger)	244 Byte	192 µs	806 µs
	122 INT + 0 BOOL (1 Zeiger)	244 Byte	195 µs	274 µs
	122 INT + 0 BOOL (122 Zeiger)	244 Byte	198 µs	672 µs

5.4 S7-Routing

Unter S7-Routing versteht man die Möglichkeit, S7-Steuerungen als Router einzusetzen, um auf unterlagerte Zielsysteme, z. B. Steuerungen oder Antriebe, zuzugreifen, die sich in unterschiedlichen Subnetzen befinden. Dies schließt auch den Wechsel des Bussystems (Ethernet, PROFIBUS, MPI) mit ein.

Referenz



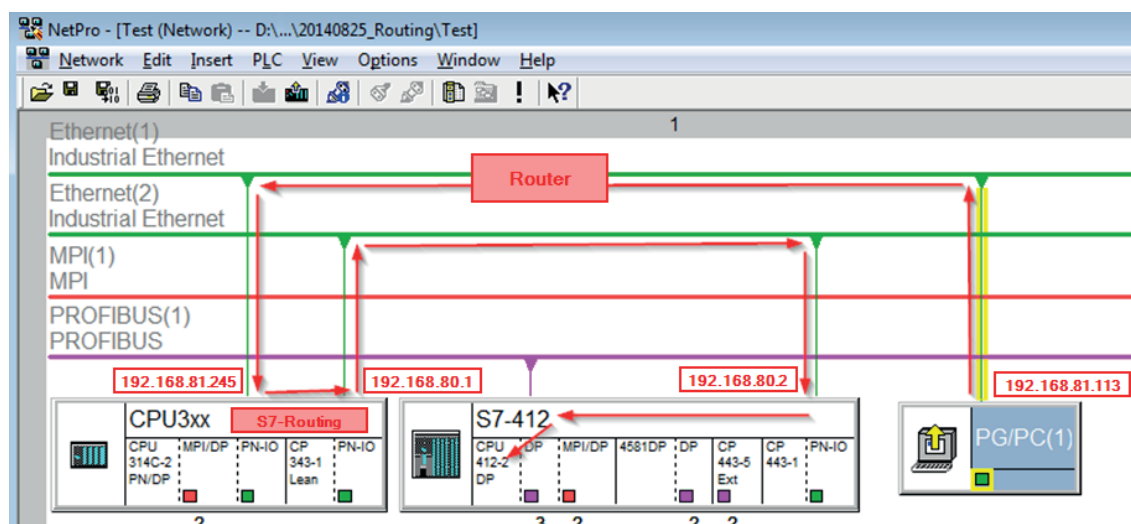
Weiterführende Informationen zum Thema S7-Routing finden Sie hier:

- Welche Baugruppen unterstützen die Funktion "S7-Routing" in S7-Subnetzen?
<https://support.industry.siemens.com/cs/ww/de/view/584459>
- Welche Voraussetzungen müssen erfüllt sein und was muss ich beachten, wenn ich Routing durchführen will?
<https://support.industry.siemens.com/cs/ww/de/view/2383206>
- Wie können Sie das S7-Routing im TIA Portal und in STEP 7 V5.x projektübergreifend aktivieren?
<https://support.industry.siemens.com/cs/ww/de/view/109474569>

5.4.1 Routing von Ethernet auf Ethernet

Die Funktion *S7-Routing* ist nicht mit IP-Routing zu verwechseln.

Das Beispiel zeigt die Realisierung des folgenden Zugriffswegs über S7-Routing in NetPro.



Der Engineering-Rechner (auch mit *ibaPDA*) soll auf die Steuerung CPU412 zugreifen. Der Rechner und die Steuerung sind nicht direkt miteinander über ein gemeinsames Netzwerk/Bus verbunden. Die Verbindung soll über die Steuerung CPU314C laufen.

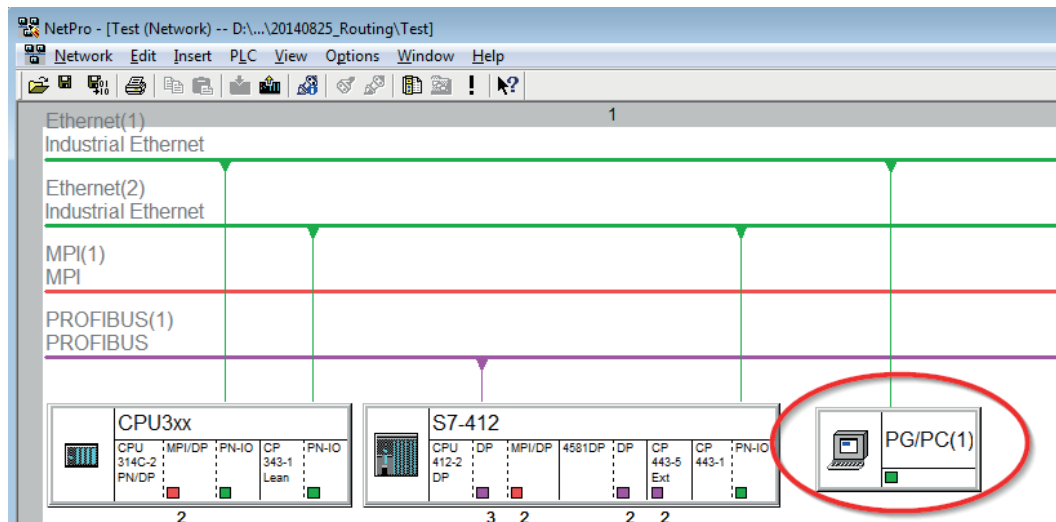
Das "Durchreichen" der Kommunikation in dieser Steuerung wird als *S7-Routing* bezeichnet.

In dem Beispiel befinden sich Engineering-Rechner und CPU314C ebenfalls in zwei unterschiedlichen (logischen) Subnetzen. Für eine Kommunikationsverbindung ist der Einsatz eines (IP-) Routers notwendig. Dies ist völlig unabhängig von der Funktion S7-Routing und nicht damit zu verwechseln.

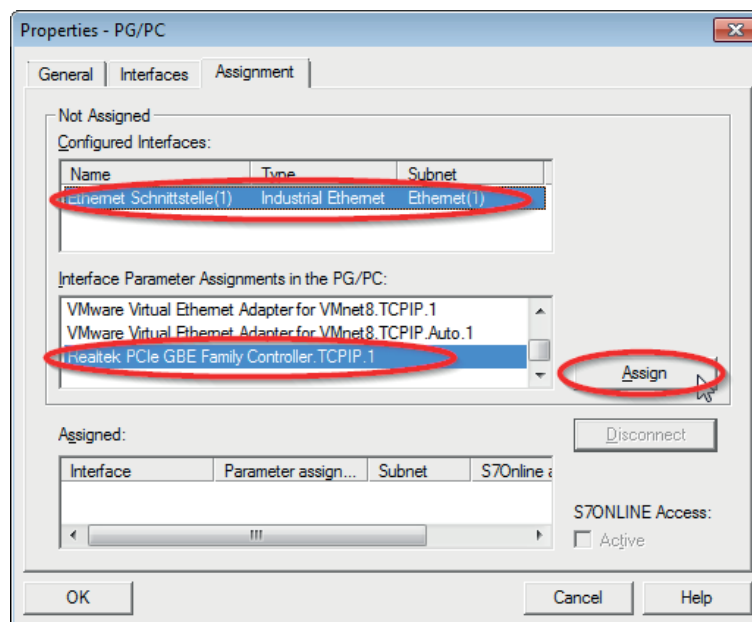
5.4.1.1 Konfiguration von STEP 7/NetPro

Folgende Konfigurationsschritte sind ausschließlich notwendig, um mit der Programmiersoftware SIMATIC STEP 7 auf die unterlagerte Steuerung CPU412 zugreifen zu können. Für SINUMERIK, SINAMICS oder SIMOTION können Sie ähnliche Schritte anwenden. Für die Verwendung von *ibaPDA* sind diese nicht notwendig.

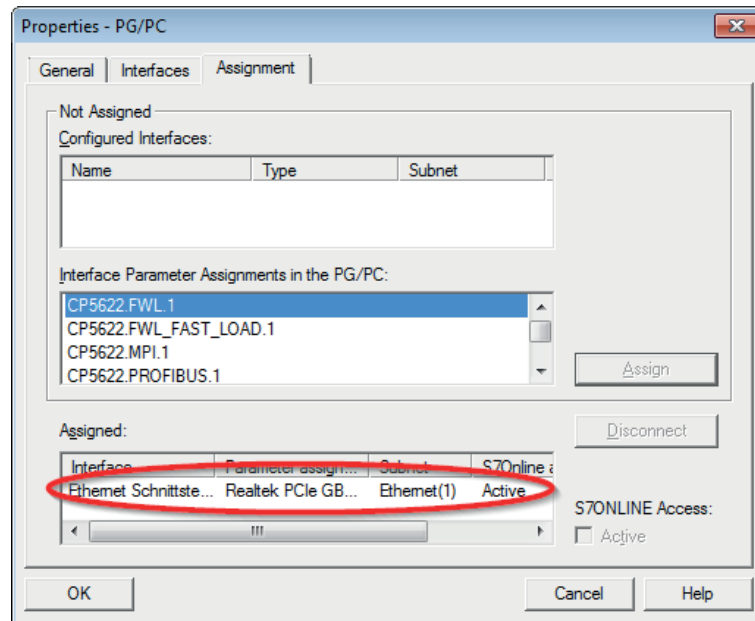
1. Fügen Sie eine PG/PC-Station in NetPro ein und konfigurieren Sie diese.



2. Weisen Sie der PG/PC-Station eine Schnittstelle (Netzwerkkarte) zu.

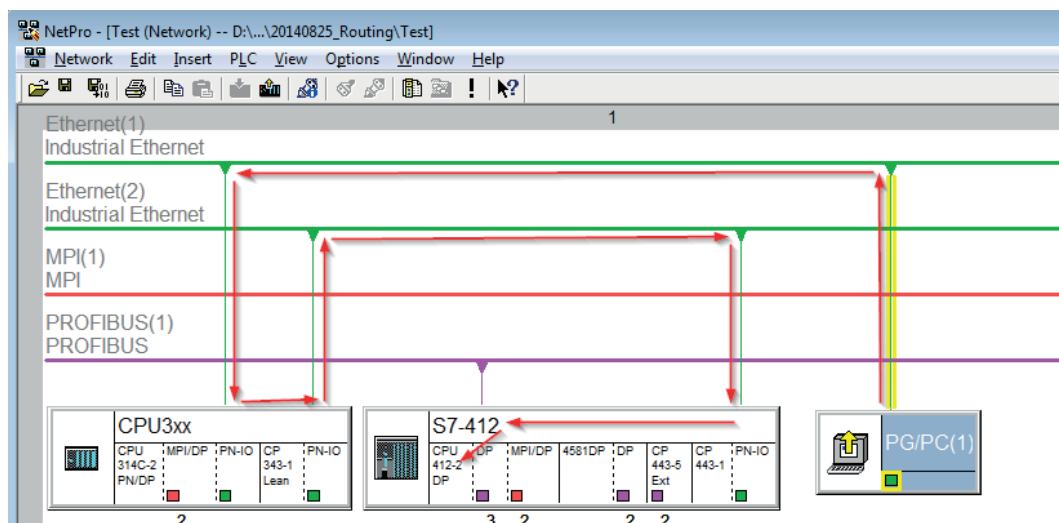


→ Ergebnis:



Die Verbindungslinie vom PG/PC zum Netzwerk muss nun gelb markiert sein.

Der Kommunikationsweg ist in der folgenden Abbildung mit Pfeilen dargestellt (diese werden nicht in SIMATIC NetPro angezeigt).



3. Laden Sie abschließend alle Hardware-Konfigurationen und Verbindungsdaten von NetPro ausgehend.

5.4.1.2 Konfiguration von ibaPDA

Konfigurieren Sie die folgenden Einstellungen.

The screenshot shows the 'Verbindung' (Connection) tab of the ibaPDA configuration window. The settings are as follows:

- Verbindungsmodus: TCP/IP
- Verbindungstyp: PG-Verbindung
- Timeout (s): 15
- Adresse: 192.168.80.2
- Rahmen: 0
- Steckplatz: 0
- ☒ S7-Routing verwenden
- Adresse des Geräts mit Gateway-Funktion (z.B. IE/PB Link): 192.168.81.245
- S7-Subnetz-ID des Ziel-Netzes: 007E-000E
- CPU-Name: S7-412 (CPU 412-2 DP)

S7-Routing verwenden

Aktivieren Sie diese Option, um S7-Routing zu verwenden.

Adresse

Geben Sie die Adresse der Zielsteuerung an (hier CPU412).

Adresse des Geräts mit Gateway-Funktion

Geben Sie die Adresse des Gateways an (hier CPU314C).

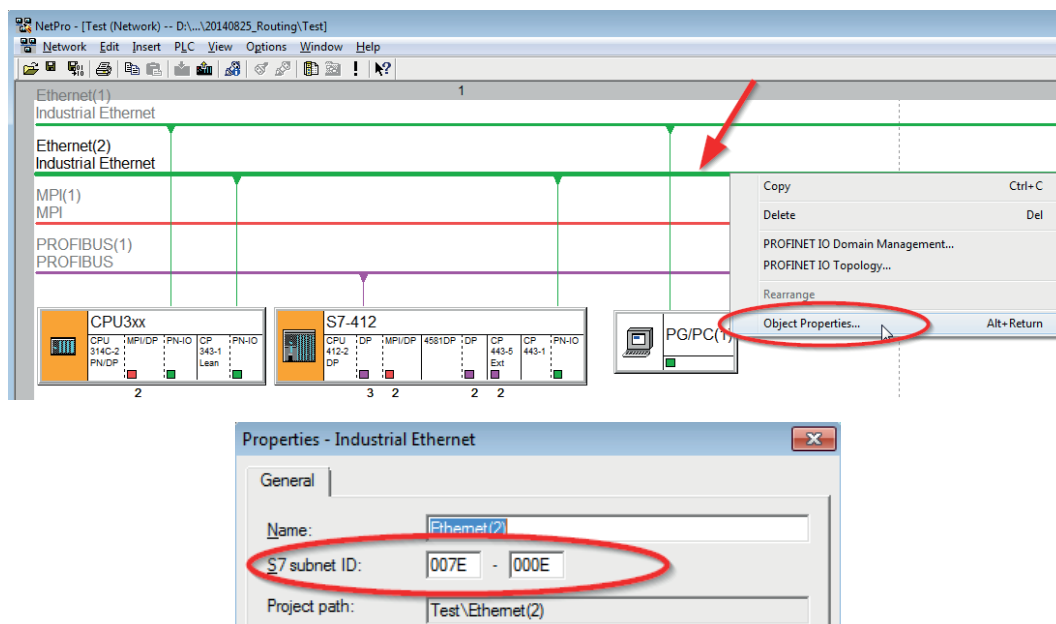
S7-Subnetz-ID des Zielnetzes

Geben Sie die Subnetz-ID aus STEP 7 NetPro oder TIA Portal an.

S7-Subnetz-ID in NetPro ermitteln

Die S7-Subnetz-ID können Sie in NetPro ermitteln.

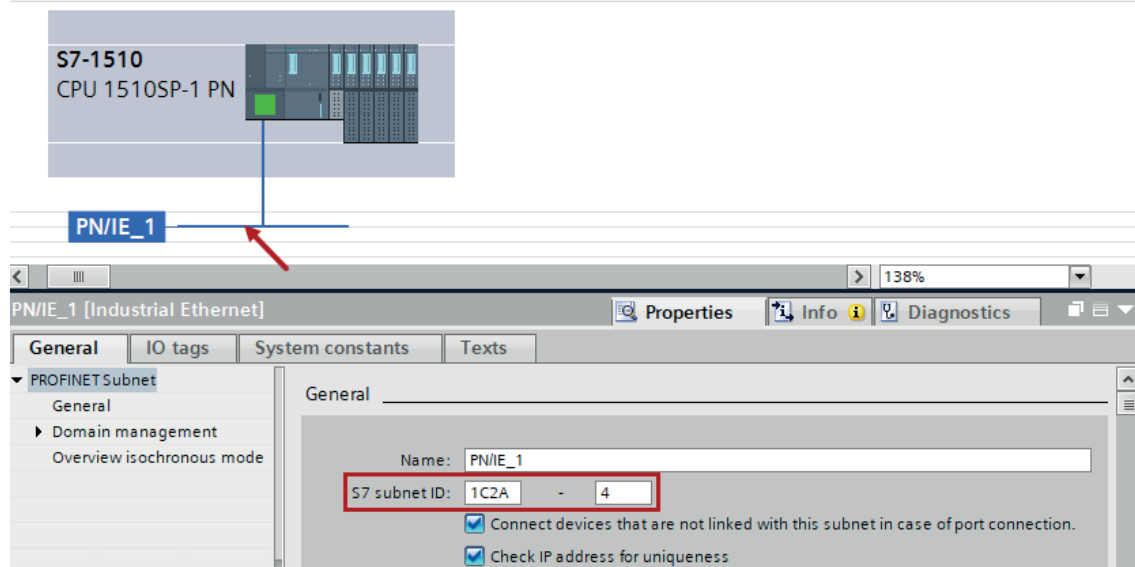
Klicken Sie mit der rechten Maustaste auf das unterlagerte Bussystem und öffnen Sie die *Objekteigenschaften*.



S7-Subnetz-ID im TIA Portal ermitteln

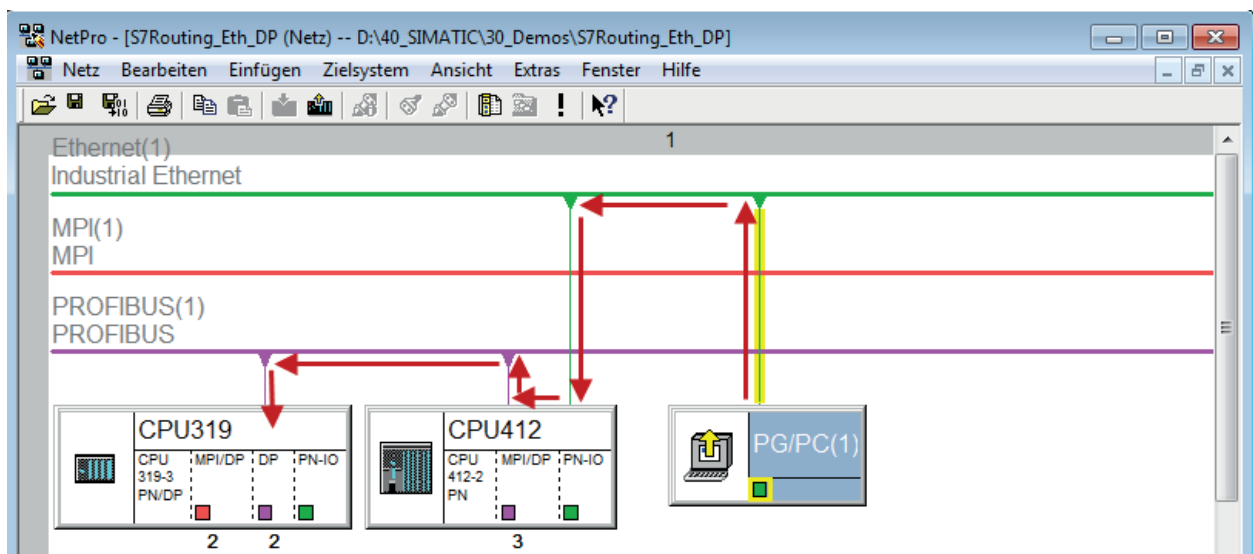
Die S7-Subnetz-ID können Sie im TIA Portal ermitteln.

Klicken Sie auf das Bussystem und gehen Sie zu *Properties – General – General*.



5.4.2 Routing von Ethernet auf PROFIBUS

Das Beispiel zeigt die Realisierung des folgenden Zugriffswegs über S7-Routing und eine beispielhafte Systemtopologie für Ethernet PROFIBUS in NetPro.



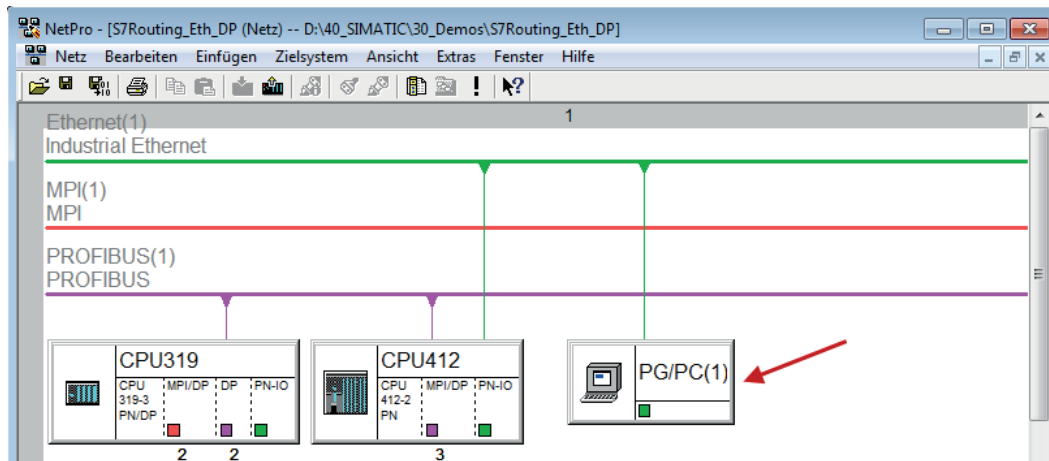
Der Engineering-Rechner (auch mit *ibaPDA*) soll auf die Steuerung CPU319 zugreifen. Der Rechner und die Steuerung sind nicht direkt miteinander über ein gemeinsames Netzwerk/Bus verbunden. Die Verbindung soll über die Steuerung CPU412 laufen.

Das "Durchreichen" der Kommunikation in dieser Steuerung wird als *S7-Routing* bezeichnet.

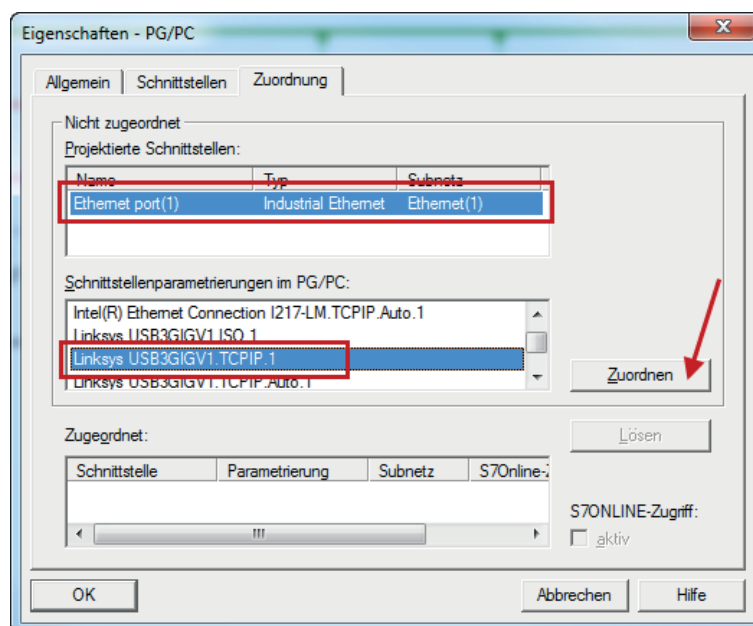
5.4.2.1 Konfiguration von STEP7/NetPro

Folgende Konfigurationsschritte sind ausschließlich notwendig, um mit der Programmiersoftware SIMATIC STEP 7 auf die unterlagerte Steuerung CPU319 zugreifen zu können. Für SINUMERIK, SINAMICS oder SIMOTION können Sie ähnliche Schritte anwenden. Für die Verwendung von *ibaPDA* sind diese nicht notwendig.

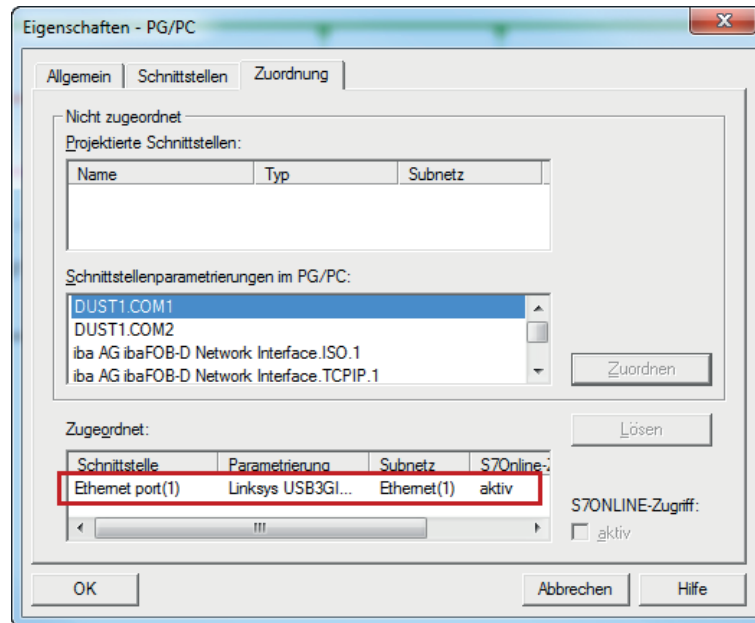
1. Fügen Sie eine PG/PC-Station in NetPro ein und konfigurieren Sie diese.



2. Weisen Sie der PG/PC-Station eine Schnittstelle (Netzwerkkarte) zu.

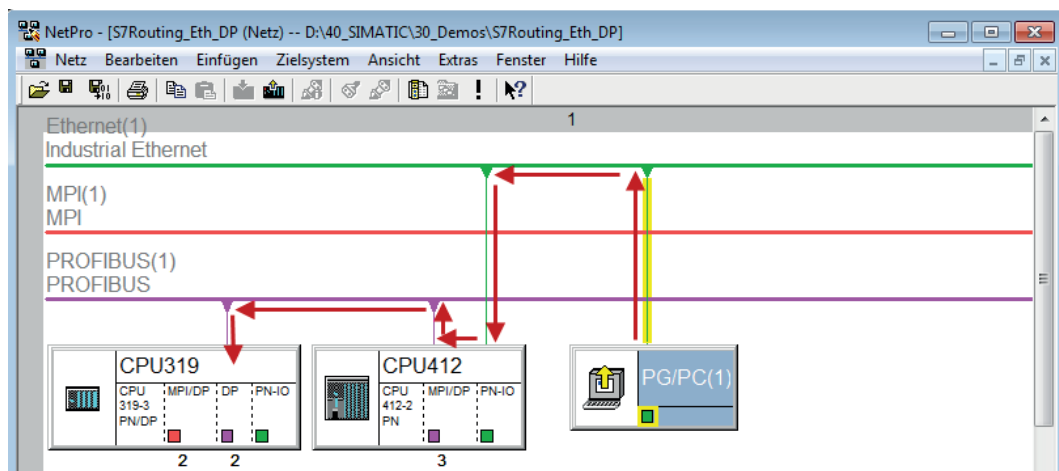


→ Ergebnis:



Die Verbindungslinie vom PG/PC zum Netzwerk muss nun gelb markiert sein.

Der Kommunikationsweg ist in der folgenden Abbildung mit Pfeilen dargestellt (diese werden nicht in SIMATIC NetPro angezeigt).



3. Laden Sie abschließend alle Hardware-Konfigurationen und Verbindungsdaten von NetPro ausgehend.

5.4.2.2 Konfiguration von ibaPDA

Konfigurieren Sie die folgenden Einstellungen.

The screenshot shows the 'Verbindung' (Connection) tab of the ibaPDA configuration window. The following settings are highlighted with red boxes:

- Verbindungsmodus:** TCP/IP
- Verbindungstyp:** PG-Verbindung
- Adresse:** 2
- S7-Routing verwenden:** ☒ (checked)
- Adresse des Gerätes mit Gateway-Funktion (z.B. IE/PB Link):** 192.168.50.95
- S7-Subnetz-ID des Ziel-Netzes:** 02D6-000B
- Timeout (s):** 15
- CPU-Name:** Kein Adressbuch

S7-Routing verwenden

Aktivieren Sie diese Option, um S7-Routing zu verwenden.

Adresse

Geben Sie die DP-Adresse der Zielsteuerung an (hier CPU319).

Adresse des Geräts mit Gateway-Funktion

Geben Sie die Adresse des Gateways an (hier CPU412).

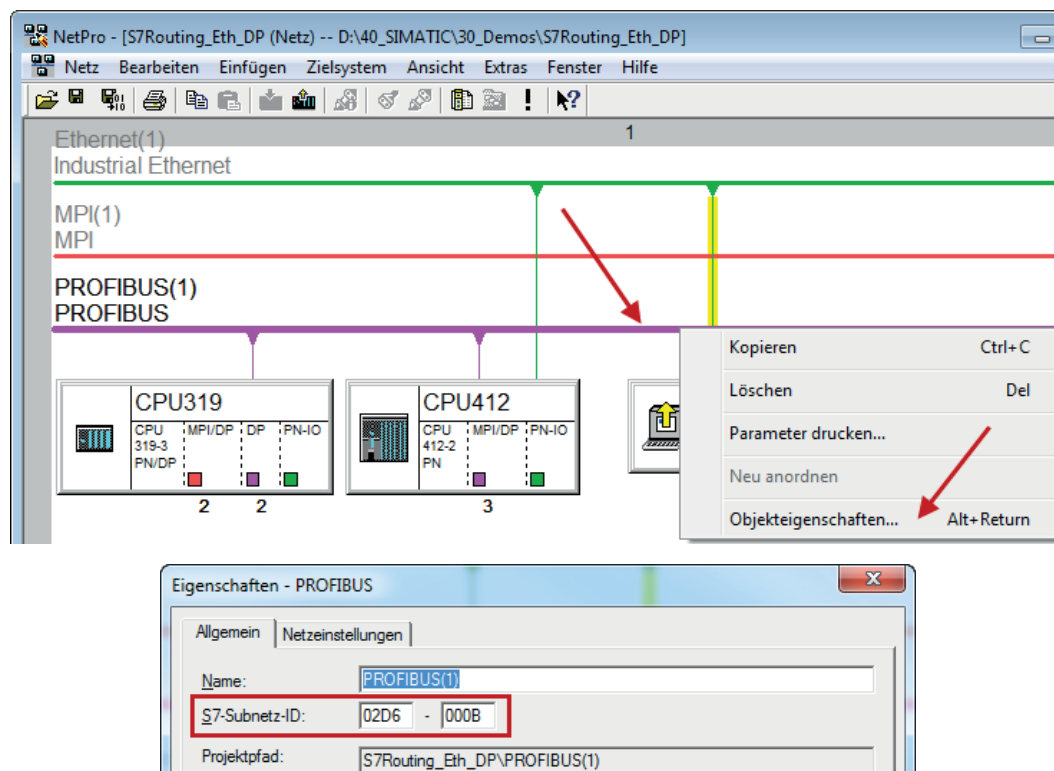
S7-Subnetz-ID des Zielnetzes

Geben Sie die Subnetz-ID aus STEP 7 NetPro oder TIA Portal an.

S7-Subnetz-ID in NetPro ermitteln

Die S7-Subnetz-ID können Sie in NetPro ermitteln.

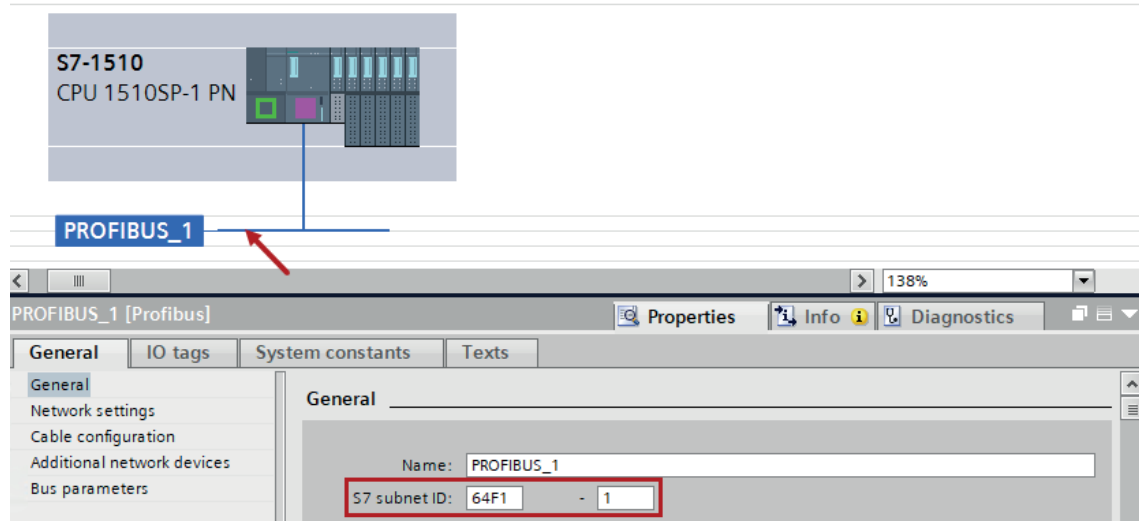
Klicken Sie mit der rechten Maustaste auf das unterlagerte Bussystem und öffnen Sie die *Objekteigenschaften*.



S7-Subnetz-ID in TIA Portal ermitteln

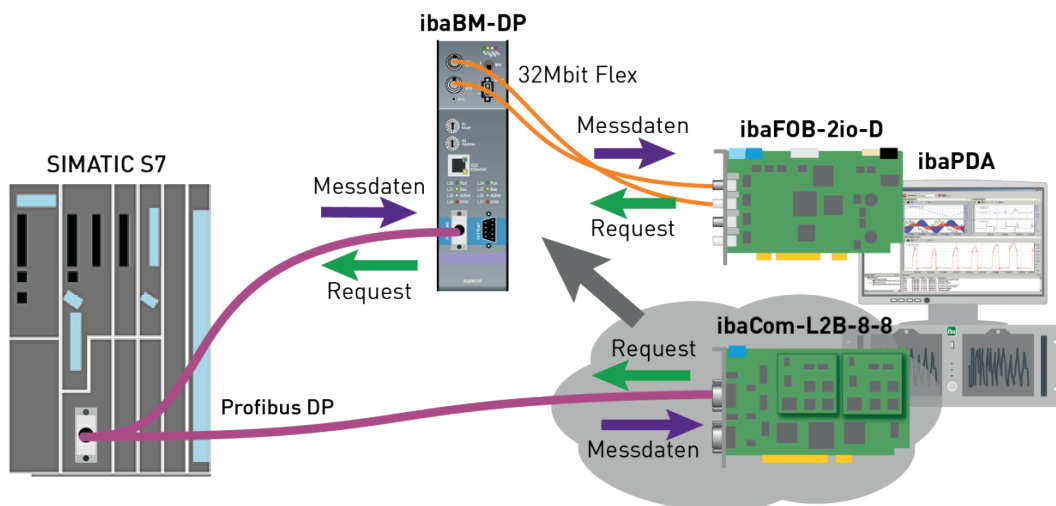
Die S7-Subnetz-ID können Sie im TIA Portal ermitteln.

Klicken Sie auf das Bussystem und gehen Sie zu *Properties – General – General*.



5.5 Ablösung Request-S7 auf ibaCom-L2B durch ibaBM-DP

Eine übliche Aufgabenstellung ist die Ablösung einer bestehenden auf der *ibaCom-L2B*-Karte basierenden Request-S7-Lösung durch ein *ibaBM-DP*. Erforderlich wird dies z. B. im Ersatzteillfall (die *ibaCom-L2B*-Karte ist abgekündigt) oder beim Einsatz eines neuen Rechners ohne PCI-Steckplätze.



Die Aufgabenstellung lässt sich einfach durch die *ibaCom-L2B* kompatiblen Module des *ibaBM-DP* lösen.

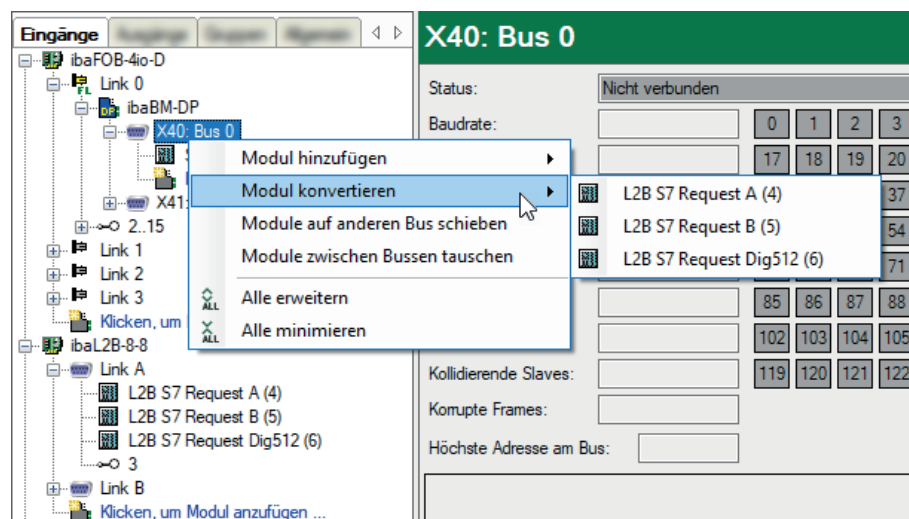
Dabei sind keine Änderungen im Programm und der Hardware-Konfiguration der S7-CPU erforderlich!

Wie bei der *ibaCom-L2B*-Karte ist auch bei der kompatiblen Lösung mit *ibaBM-DP* keine Netzwerk-Verbindung zwischen *ibaPDA*-Rechner und der S7-CPU erforderlich. Die Anforderung der Messwerte (Request-Handshake) wird per Lichtwellenleiter und über den PROFIBUS I/O-Bereich gesendet.

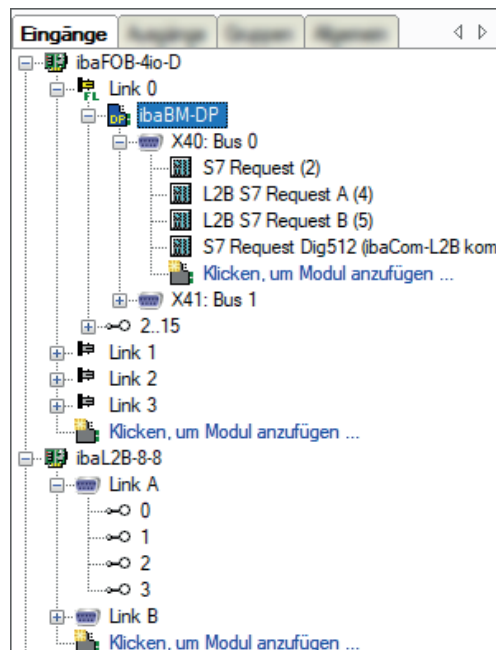
Folgende Schritte sind zur Ablöse erforderlich:

1. Falls erforderlich bauen Sie eine neue *ibaFOB-D*-Karte in den *ibaPDA*-Rechner ein. Eine Karte mit Eingang und Ausgang (*ibaFOB-io-D*, *ibaFOB-2io-D* oder *ibaFOB-4io-D*) ist erforderlich, da die kompatiblen Module nur im bidirektionalen 32Mbit Flex-Modus eingesetzt werden können.
 2. Schließen Sie das *ibaBM-DP*-Gerät über das bidirektionale Lichtwellenleiterkabel an die *ibaFOB-D* Karte an.
 3. Schließen Sie die PROFIBUS-Stecker der *ibaCom-L2B*-Karte an die Buchsen des *ibaBM-DP* an.
 4. Im I/O-Manager von *ibaPDA* erscheint die neu installierte *ibaFOB-D*-Karte. Projektieren Sie am entsprechenden Link ein *ibaBM-DP*-Gerät.
 5. Klicken Sie mit der rechten Maustaste auf den gewünschten PROFIBUS-Anschluss und wählen Sie im Kontextmenü *Modul konvertieren*.
- Alle vorhandenen L2B Request-S7-Module werden angezeigt.

Wählen Sie hier die Module aus, die Sie zu einem *ibaCom-L2B* kompatiblen Modul konvertieren wollen (im Normalfall alle Module).



- Die ausgewählten Module werden bei der *ibaCom-L2B*-Karte gelöscht und in kompatible Module auf dem *ibaBM-DP*-Gerät konvertiert.



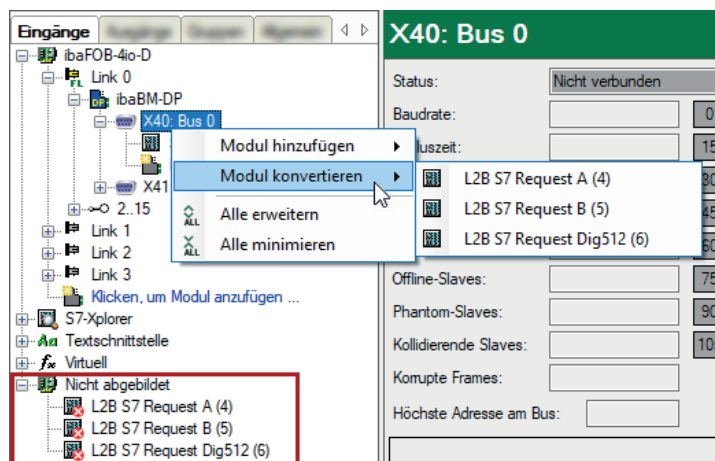
- Die Konvertierung ist abgeschlossen.

6. Übernehmen Sie die neue Konfiguration mit <OK>.

Hinweis



Sie können L2B Request-S7-Module auch konvertieren, wenn die Module bei den nicht abgebildeten Modulen abgelegt sind. Dies ist beispielsweise der Fall, wenn die bestehende I/O-Konfiguration inkl. L2B Request-S7-Modulen auf einem neuen *ibaPDA*-Rechner geladen wird, der keine *ibaCom-L2B*-Karte mehr hat, sondern nur die Kombination von *ibaFOB-D* und *ibaBM-DP*.



5.6 Fehlercodes Request-Blöcke

Die Request-Blöcke liefern folgende mögliche Fehlercodes:

FB140/141/...

Wert ERROR_STATUS	Beschreibung
1	datablock ibaREQ_DB is write protected
2	datablock ibaREQ_DB invalid (DB = 0 or > limit of CPU)
3	datablock ibaREQ_DB does not exist
4	datablock ibaREQ_DB undefined error
5	datablock ibaREQ_DB too short
6	datablock ibaREQ_DB too short for ibaREQ_UDP
9	internal error (RD_SINFO)
10	no access to datablock ibaREQ_DB (read)
11	no access to datablock ibaREQ_DB (write)
20	initialization not finished
21	insufficient memory for SZL
22	wrong SZL_ID
23	wrong or invalid index of SZL
24	error while reading I&M data from CPU
25	error while reading PLC data
31	initialization canceled with error
32	initialization not completed
41	too many pointers (ibaREQ_DB too small)
42	too many pointers in one command (> 128)
44	invalid command id
45	operand invalid (not defined)
46	operand invalid (datatype)
47	operand invalid (memory area)
200	no connection to PN device/DP-Slave
300	version of ibaREQ_UDPack does not match with ibaREQ_M (ID)
301	version of ibaREQ_UDPack does not match with ibaREQ_M (FB)
302	version of ibaREQ_UDPack does not match with ibaREQ_M (DB)
303	type of transmit agent does not match with configured request type in <i>ibaPDA</i>
305	PROFIBUS DP-Slave hardware configuration is invalid
306	configured peripheral address is invalid
310	no access to datablock ibaREQ_DB (read)
311	no access to datablock ibaREQ_DB (write)
315	error while masking of synchronous faults

Wert ERROR_STATUS	Beschreibung
316	error while demasking of synchronous faults
320	operand invalid (datatype)
321	operand invalid (pointer)
401	ADR_SLOT/ADR_SLOT_0 invalid hw-id
402	ADR_SLOT/ADR_SLOT_0 invalid hw-id, no IO-Device or DP-Slave
403	ADR_SLOT/ADR_SLOT_0 invalid hw-id, is no PROFIBUS or PROFINET
406	ADR_SLOT/ADR_SLOT_0 invalid configuration slot (0)
407	ADR_SLOT/ADR_SLOT_0 invalid configuration slot (0)
409	ADR_SLOT_1 invalid configuration slot 1
410	no connection to PN device/DP-Slave or error
411	ADR_SLOT_1 invalid hw-id
412	ADR_SLOT_1 invalid hw-id, no IO-Device or DP-Slave
413	ADR_SLOT_1 invalid hw-id, is no PROFIBUS
416	ADR_SLOT_1 invalid configuration slot 1
0x8yyy	errorcode of inner TUSEND/AG_SEND/AG_LSEND

Tab. 3: Fehlercodes Request-Blöcke FB140/141/...

FC122 (PROFIBUS)

Wert ERROR_STATUS	Beschreibung
1	DB_PDA is write protected
2	DB_PDA = 0 or > limit of CPU
3	DP_PDA does not exist
5	DB_PDA too short
11	DB_INTERN is write protected
12	DB_INTERN = 0 or > limit of CPU
13	DB_INTERN does not exist
15	DB_INTERN too short
16	error while reading identification data of CPU
19	initialization not completed
21	insufficient memory for system status list
22	wrong or unknown system status list
23	wrong or invalid index of system status list
30	invalid OUTPUT_ADR_SLAVE
31	OUTPUT_ADR_SLAVE no PROFIBUS DP-Slave
100	bit number not 0
101	bit number not 07
103	operand invalid (memory area)

Wert ERROR_STATUS	Beschreibung
104	operand invalid (datatype)
105	operand invalid (datablock 0)
106	datablock number > limit of CPU
107	datablock does not exist
109	datablock too short
110	address does not exist
111	initialization canceled with error
112	initialization not completed
150	request fragmentation not supported
151	wrong number of requested values
152	only < 64 digital signals are supported
153	only < 64 analog signals are supported
200	no connection to DP-Slave

Tab. 4: Fehlercodes Request-Block FC122

FC123

Wert ERROR_STATUS	Beschreibung
1	DB_PDA is write protected
2	DB_PDA = 0 or > limit of CPU
3	DP_PDA does not exist
5	DB_PDA too short
11	DB_INTERN is write protected
12	DB_INTERN = 0 or > limit of CPU
13	DB_INTERN does not exist
15	DB_INTERN too short
16	error while reading identification data of CPU
19	initialization not completed
21	insufficient memory for system status list
22	wrong or unknown system status list
23	wrong or invalid index of system status list
30	invalid OUTPUT_ADR_SLAVE
31	OUTPUT_ADR_SLAVE no PROFIBUS DP slave
32	RM: the parameterized "OUTPUT_ADR_SLAVE_BUS_0" is wrong.
33	RM: the parameterized "OUTPUT_ADR_SLAVE_BUS_0" is not assigned to a PROFIBUS DP-Slave
34	RM: the parameterized "OUTPUT_ADR_SLAVE_BUS_1" is wrong

Wert ERROR_STATUS	Beschreibung
35	RM: the parameterized "OUTPUT_ADR_SLAVE_BUS_1" is not assigned to a PROFIBUS DP-Slave
36	RM: SLAVE BUS0 and SLAVE BUS1 do not have the same DP address
100	bit number not 0
101	bit number not 07
103	operand invalid (memory area)
104	operand invalid (datatype)
105	operand invalid (datablock 0)
106	datablock number > limit of CPU
107	datablock does not exist
109	datablock too short
110	address does not exist
111	initialization canceled with error
112	initialization not completed
150	request fragmentation not supported
151	wrong number of requested values
152	only < 64 digital signals are supported
153	only < 64 analog signals are supported
200	no connection to DP-Slave
201	RM: slave bus 0 has failed
202	RM: slave bus 1 has failed
203	RM: slaves bus 0 + 1 have failed
210	output modules of the slaves bus 0 and 1 are configured differently

Tab. 5: Fehlercodes Request-Block FC123

5.7 Nutzung von MPI/DP-TCP-Adapttern

Siemens S7 CPUs, die über keine Ethernet-Schnittstelle verfügen, können Sie mittels MPI/DP-TCP-Adapttern an deren MPI-Schnittstelle anschließen.

Von verschiedenen Herstellern gibt es Adapter, die S7-Kommunikation von TCP/IP auf MPI/PROFIBUS DP umsetzen.

Diese Adapter können Sie grundsätzlich auch mit *ibaPDA* einsetzen. Hierzu projektieren Sie die Verbindung *ibaPDA*-seitig als TCP-Verbindung.

Der Adapter kann über eine IP-Adresse angesprochen werden, die Sie mittels einer vom Hersteller mitgelieferten Software einstellen.

Geben Sie in der Verbindungskonfiguration als Rahmen "0" und als Steckplatz die MPI- bzw. DP-Adresse der S7-CPU an.

Weitere Informationen entnehmen Sie der jeweiligen Gerätedokumentation.

6 Support und Kontakt

Support

Tel.: +49 911 97282-14

E-Mail: support@iba-ag.com

Hinweis



Wenn Sie Support benötigen, dann geben Sie bitte bei Softwareprodukten die Nummer des Lizenzcontainers an. Bei Hardwareprodukten halten Sie bitte ggf. die Seriennummer des Geräts bereit.

Kontakt

Hausanschrift

iba AG
Königswarterstraße 44
90762 Fürth
Deutschland

Tel.: +49 911 97282-0

E-Mail: iba@iba-ag.com

Postanschrift

iba AG
Postfach 1828
90708 Fürth

Warenanlieferung, Retouren

iba AG
Gebhardtstraße 10
90762 Fürth

Regional und weltweit

Weitere Kontaktadressen unserer regionalen Niederlassungen oder Vertretungen finden Sie auf unserer Webseite:

www.iba-ag.com