



Basis Profil 10

V2.0

2011-12-07

07

Profil : Basis
Profil-Nummer : 10
Version: V2.0
Versionszähler: 07
Datum : 7. Dezember 2011
Herausgeber : INTERBUS Stabsstelle
Telefon : (0 52 35) 34 21 00
eMail: info@interbusclub.com
www.interbus.com

Copyright by INTERBUS Stabsstelle

Alle Abbildungen und Beschreibungen wurden nach bestem Wissen erstellt und geprüft, befreien den Anwender jedoch nicht von eigenen Prüfungen und Versuchen. Änderungen jeder Art, insbesondere soweit Sie sich aus technischem Fortschritt ergeben bleiben vorbehalten. Für Fehlhandlungen und Schäden, die durch Mißachtung der in diesem Profil enthaltenen Informationen entstehen, übernimmt die Stabsstelle des ehemaligen INTERBUS Club Deutschland e.V. keine Haftung. Dieses Profil, einschließlich aller darin enthaltenen Abbildungen, ist urheberrechtlich geschützt. Jede Drittverwendung dieses Profils, die von den urheberrechtlichen Bestimmungen abweicht, ist verboten.
Änderungen vorbehalten

Revisionsverzeichnis

Index	Datum	Änderungen, Ergänzungen oder Erläuterungen
1.00	14.10.2002	Erste Version
1.01	12.08.2004	<ul style="list-style-type: none"> • Ergänzung Kanalnummer im Diagnoseobjekt 0018 und folgende • Ergänzung der Prioritäten 81, 82, 83 „Störung ist beseitigt“ Diagnoseobjekt 0018 und folgende • Überarbeitung / Ergänzung der Störungscode • Erweiterung der Handhabungsbeschreibung von Diagnosemeldungen • Objekt 000F DeviceProfile Mandatory • Vorspann und Anhang eingefügt • Redaktionelle Überarbeitung • Einführung der Objekte <ul style="list-style-type: none"> - 002E – CheckSum - 002F - PDOUT_Subst - 0030 - PF_Code - 0031 - PDIN_Subst - 0032 - IBS_ID • Definition von Werten für die Objekte <ul style="list-style-type: none"> - 0024 – IBSResetCode - 0020 – PDTIMEOUTCODE
1.02	28.01.2005	<ul style="list-style-type: none"> • Fehlerbereinigung (Re/Gr/Format/etc.) • Objekt 0019 „ResetDiag“ Name und Bedeutung angepasst • Objekt 000F „DeviceProfile“ Typ geändert • Störungscode erweitert: 2344; 2345; 341X; 342X; 5230
1.10	20.06.06	<ul style="list-style-type: none"> • Objekt 0018 „DiagState.Kanal“ in „DiagState.Channel/Group“ geändert, Bedeutung angepaßt • Objekt 0018 „DiagState. MoreFollows“ Bedeutung angepasst • Objekt 001A „GetErrorRepMethod“, Name und Beschreibung erweitert • Objekt 0x001D „Password“ angepaßt • Objekt 0x0025 „PDIN“ zusätzlicher Erläuterung • Objekt 0x0026 „PDOUT“ zusätzlicher Erläuterung • Objekt 002E „CheckSum“, Typ geändert UINT16 => UNIT32 • Objekt 0033 „DiagStateChannelNumber“ neu eingefügt • Objekt 0034 „DiagStateAddValue“ neu eingefügt • Objekt 0035 „NoOfModules“ neu eingefügt • Objekt 0036 „DeviceStructure“ neu eingefügt • Objekt 0037 „DeviceType“ neu eingefügt • Objekt 0038 „ObjDescrReq“ neu eingefügt • Objekt 0039 „ObjDescr“ neu eingefügt • Objekt 003A „VersionCount“ neu eingefügt • Objekt E805 „ObjDescrLong“ neu eingefügt • Objekt E800 „DiagStateLong“ teilweise als Verweis beschrieben und doppelte Beschreibungen zu vermeiden • Modulare Geräte sind beschrieben. Als Moduladressierung wird die Invoke-ID, jetzt „Module Number“, genutzt • Liste der zulässigen Kommunikations- Fehlercodes überarbeitet und ergänzt • Datentyp „Bitstring“ eingeführt. • Kapitel „Kommunikationsobjekte“ überarbeitet, neu strukturiert und ergänzt • Kapitel Objektbeschreibung eingefügt • Einführung Error Class 8 Error Code 1 profilspezifisch

1.11	06.10.2009	<ul style="list-style-type: none"> • Additional Error Code überarbeitet • Handhabung Objekt 0x0038 „ObjDescrReq“ Objekt 0x0039 „ObjDescr“ erweitert • Error- Class “8” Other, Error- Code “1” Profile specific durchgängig korrigiert • Datentyp Visible-String (immer “0x00” terminiert!) • Objekt 0037 „DeviceType“ leicht überarbeitet • Objekt 0008 „ProductID“ um Missverständnisse zu vermeiden in „SerialNo“ umbenannt • Störungscode 6800 und 6810 eingefügt A000 ff erweitert und ausführlicher erläutert • Erläuterung des Objektraums für modulare Geräte • Objekt 003B „PDIN_Descr“ neu eingefügt • Objekt 003C „PDOUT_Descr“ neu eingefügt • Objekt 0019 „ResetDiag“ erweitert und konkretisiert
1.12	19.11.2010	<ul style="list-style-type: none"> • Device Family um digital / analog IN/OUT ergänzt • Liste der Datentypen vervollständigt • LanguageCode korrigiert • Länge bei Visible-String korrigiert • Objekt 003B „PDIN_Descr“ überarbeitet, neue Typen eingefügt • Objekt 003C „PDOUT_Descr“ überarbeitet, neue Typen eingefügt • Objekt 000C „FirmwareVersion“ und 000B „PChVersion“ konkretisiert um Einträge wenn keine FW/PChVersion vorhanden ist • Handhabung der Prioritäten im DiagnoseObjekt 0x0018 DiagState erläutert • Objekt 0019 „ResetDiag“ erweitert und konkretisiert • Objekt 0018.5 „ResetDiag.MoreFollows“ konkretisiert • Additional ErrorCodes ergänzt • Objekt 0037 „DeviceType“ Fehler korrigiert R/W => R, kann niemals R/W gewesen sein • SubIndex „Access-Rights“, „DisplayFormat“, „Resolution“ und „Offset“ in Objekt 0x0039 „ObjDescr“ eingefügt, Beschreibung ergänzt. • Größenindex für „Kraft“ und „el. Widerstand“ ergänzt • Objekt 0020, 0024, 0030 UINT16 => Array of UINT16 geändert und kommentiert
2.0	07.12.2011	<ul style="list-style-type: none"> • Basisprofil „neutralisiert“ D.h. den Bezug zu INTERBUS und PCP aufgelöst. • Kommentare zu den Versionscountern ergänzt • Objekt 003D „WakeUpTime“ eingeführt • ResetCode ausführlicher erläutert • Objekt 002D ResetParam Code “02” eingeführt • !!! Objekt 0x0039 „ObjDescr“ inkompatibel überarbeitet (.Min und . Max verschoben) • Übersetzungstabelle für die Objektnamen im Anhang hinzugefügt • Beschreibung der Download Services (Download-Write, Download-Read) ergänzt und konkretisiert. • Objekt 0011 Fehler in der Stringlänge (10+1) korrigiert • Objekt 0x0005 Capabilities eingeführt • Objekt 0038 “ObjDescrReq” Länge korrigiert
		<ul style="list-style-type: none"> •

Inhalt

1	Vorwort	7
2	Anwendung und Geräteeigenschaften	8
3	Allgemeines.....	9
4	Übersicht.....	9
4.1	Indexbereiche der Kommunikationsobjekte.....	9
4.2	Liste der Standardobjekte	10
4.3	Liste der zulässigen Kommunikations- Fehlercodes	12
4.3.1	Error-Class und Error-Code.....	12
4.3.2	Additional-Code:	13
5	Kommunikationsobjekte	15
5.1	Datentypen	15
5.2	Datenobjekte	15
5.2.1	Objekt Domain-Variable	15
5.2.1.1	Domain-Variable - formale Beschreibung.....	15
5.2.2	Objekt Simple- Variable	16
5.2.2.1	Simple-Variable - formale Beschreibung	16
5.2.3	Objekt Array	16
5.2.3.1	Array-Variable - formale Beschreibung.....	16
5.2.4	Objekt Record.....	17
5.2.4.1	Record- Variable - formale Beschreibung	17
5.2.5	Objekt Variable- List	17
5.2.5.1	Variable- List - formale Beschreibung.....	17
5.2.5.2	Statische Variable-List.....	18
5.2.5.3	Dynamische Variable-List	18
5.2.5.4	Übertragungsformat Variablenlisten	18
5.2.6	Objekt String-Variable	19
5.2.6.1	String-Variable - formale Beschreibung.....	19
6	Services.....	20
6.1	Download-Funktionalität mit Write-Service (Download-Write).....	20
6.2	Upload-Funktionalität mit Read-Service (Upload-Read)	23
6.3	Beispiel Download Variablen- Liste.....	26
7	Standardobjekte.....	27
7.1	Identifikation	28
7.1.1	DeviceFamily.....	32
7.1.2	CommunicationProfile	33
7.1.3	DeviceProfile	33
7.2	Geräte- Diagnose.....	34
7.2.1	Objekte	34
7.2.1.1	Benachrichtigung über Auslesen (Read) des DiagState Objektes... 39	
7.2.1.2	Benachrichtigung über Melden (Information Report) des DiagState Objektes 40	
7.2.1.3	Klassifizierung von Meldungen.....	40
7.2.1.4	Störungs_codes	42
7.2.2	Trace- Daten	48
7.3	User Daten Management	49
7.3.1	Prozeßdaten Management	49
7.3.2	Parameterkanal Management	55
7.4	Geräte Management.....	56
7.4.1	Blockparametrierung	57
7.4.1.1	Schreibsteuerung.....	57
7.4.1.2	Konflikt-Verzeichnis	58
7.4.2	Parametersatzkennung	58
7.4.3	Passwortschutz.....	59

7.5	Mehrsprachigkeit	61
7.6	Modulare Geräte	63
7.7	Objektbeschreibung	64
8	Anhang A	68
8.1	Begriffsdefinitionen	68
8.2	Symbole und Abkürzungen	70
9	Anhang B	71
9.1	Übersetzungstabelle für die Objektnamen	71

1 Vorwort

Im Rahmen der Fabrikautomatisierung werden in der industriellen Sensorik und Aktorik immer mehr leistungsfähigere und flexiblere Systeme benötigt. Intelligente Feldgeräte können diese Anforderungen erfüllen. Ihre volle Integration in komplexe Fertigungsabläufe setzt aber offene und standardisierte Kommunikationsfähigkeit voraus.

Der Grundgedanke von offenen Systemen ist, den Informationsaustausch zwischen Anwendungsfunktionen zu ermöglichen, die auf Geräten unterschiedlicher Hersteller implementiert sind. Hierzu gehören festgelegte Anwendungsfunktionen, eine einheitliche Anwenderschnittstelle zur Kommunikation und ein einheitliches Übertragungsmedium.

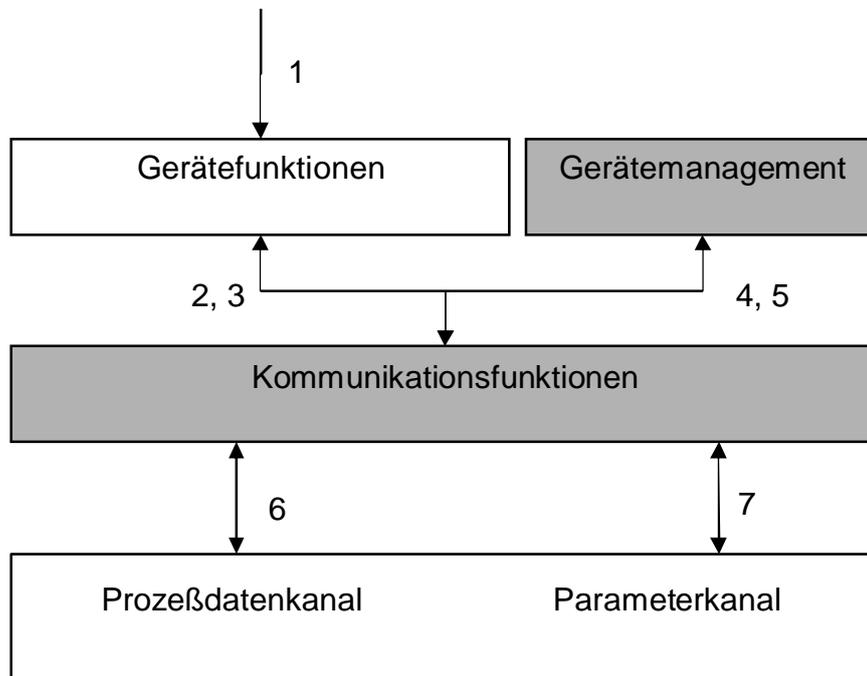
Der INTERBUS Club Deutschland e.V. hat sich zur Aufgabe gemacht, die wichtigsten Feldgerätefunktionen zu standardisieren und in diesem Profil zusammenzufassen. Um die Feldgerätefunktionen unabhängig vom Kommunikationsmedium definieren zu können, wurde eine international anerkannte und standardisierte Anwenderschnittstelle IEC 61158 zur Kommunikation verwendet. Damit wurde eine Durchgängigkeit zu MMS geschaffen. Als Übertragungsmedium wurde kein spezielles Feldbus- System ausgewählt. Es muß lediglich die Anforderungen der Feldkommunikation bezüglich Echtzeitverhalten und standardisierter Anwenderschnittstelle erfüllen.

Das Basisprofil richtet sich an alle Nutzer und Gerätehersteller von Feldgeräten die an einem Feldbus betrieben werden sollen. Diese Profil-Definition ist für den Anwender eine sinnvolle Ergänzung zur standardisierten Kommunikation und bringt eine allgemeingültige Absprache über Dateninhalte und Geräteverhalten. Diese Funktionsfestlegungen vereinheitlichen wesentliche Feldgeräteparameter. Hierdurch zeigen Geräte verschiedener Hersteller, bei Verwendung dieser Standardparameter, ein einheitliches Verhalten am Kommunikationsmedium.

Zur Konformitätsprüfung und Zertifizierung von Produkten mit dem Basisprofil wird eine unabhängige Sachverständigenstelle eingerichtet. Aufgrund von weiterführenden Standardisierungsarbeiten ist mit weiteren Ergänzungen zu rechnen.

2 Anwendung und Geräteeigenschaften

In diesem Kapitel wird die gesamte Anwendung aus der Kommunikationssicht beschrieben. Die Anwendung ist in folgende Funktionsblöcke aufgeteilt:



Kommunikationsfunktion

Die Kommunikationsfunktion führt alle kommunikationsspezifischen Funktionen aus.

Gerätefunktion

Die Gerätefunktion führt alle gerätespezifischen Funktionen aus.

Geräteinformation

Die Geräteinformation verwaltet Informationen über das Gerät in einen nichtflüchtigen Speicher.

Interaktionen zwischen den Funktionsblöcken

1 Prozeßgrößen

2 Prozeßdaten vom überlagertem Steuerungssystem zu den Gerätefunktionen

3 Prozeßdaten von den Gerätefunktionen zum überlagertem Steuerungssystem

4 Speicherung von Geräteinformationen zu Management, Diagnose, Identifikation

5 Auslesen der Geräteinformationen zu Management, Diagnose, Identifikation

6 Abbildung auf den Prozeßdatenkanal

7 Abbildung auf den Parameter-Kanal

3 Allgemeines

Zum Management und der Diagnose moderner Feldgeräte gehören neben Informationen über den augenblicklichen Zustand des Gerätes auch Informationen zum Gerät selbst, historische Zustandsinformationen, Fehlerabbilder usw.

Bisher stehen nur wenige der möglichen Informationen standardmäßig zur Verfügung. Ziel dieses Dokuments ist es, ein Basis- Profil zu definieren, welches ein einheitliches Management von Geräten die mit einem Parameterkanal ausgestattet sind, ermöglicht. Dieses Basis- Profil verschafft dem Gerätehersteller die Möglichkeiten dem Endanwender (und Tools), deutlich mehr Informationen über sein Gerät und dessen Eigenschaften mitzuteilen.

Dieses „Mehr“ an Informationen erleichtert aufgrund der Vereinheitlichung insbesondere die Inbetriebnahme der Geräte. Auch im Servicefall ist jetzt die Möglichkeit vorhanden, einheitlich und damit schneller auf bestimmte Diagnoseinformationen zuzugreifen.

Profilverständnis:

„Inhaltlich gleichartige Informationen und Abläufe auch in einer einheitlichen Form darzustellen und dabei Hersteller und Anwender nicht einzuschränken, sondern Leitfaden zu sein.“

Um diesem Anspruch gerecht zu werden, werden die relevanten Informationen einheitlich in Standardobjekten hinterlegt. Es werden Mechanismen definiert, die den Zugriff auf diese Objekte regeln und für den Fehlerfall einheitliche Fehlercodes festgelegt.

4 Übersicht

4.1 Indexbereiche der Kommunikationsobjekte

Die Kommunikationsobjekte im Basis- Profil sind in folgende Indexbereiche eingeteilt:

Index	Name	Objekttyp
0	Objektverzeichnis	OV
0001 – 003D	Allgemeingültige Standardobjekte Diese Objekte werden im weiteren in diesem Dokument beschrieben.	Variable
003E – 007F	Reserviert	
0080 – 5FFF	herstellerspezifische Applikationsobjekte In diesem Bereich legt der Gerätehersteller seine gerätespezifischen Variablen- Objekte fest.	Variable
6000 – CFFF	Profilspezifische Applikationsobjekte Weitere festgelegte Applikationsobjekte sind den entsprechenden Profilen zu entnehmen (z.B. Drivecom, Sensor/Aktor, usw.)	Variable
D000 – DFFF	Reserviert	
E000 – E3FF	statische VariablenListen	VariableList
E400 – E7FE	dynamische VariablenListe	VariableList
E7FF	Anlegen und Auslesen von Variablenlistendefinitionen	VariableListStruktur
E800 – E805	Allgemeingültige DomainVariablen Diese Objekte werden im weiteren in diesem Dokument beschrieben.	DomainVariable
E805 – E820	Reservierte DomainVariable	DomainVariable
E820 – E8FF	gerätespezifische für DomainVariable In diesem Bereich legt der Gerätehersteller seine gerätespezifischen Variablen- Objekte fest.	DomainVariable
F000 – FFFF	Reserviert	

4.2 Liste der Standardobjekte

Index	Objektname	R/W	Länge	Datentyp	M/O/D
0001	VendorName	R	max. 58 Byte	Visible-String	M
0002	VendorID	R	6+1 Byte	Visible-String	O
0003	VendorText	R	max. 58 Byte	Visible-String	O
0004	DeviceFamily	R	max. 58 Byte	Visible-String	M
0005	Capabilities	R	N x 8 Byte	Array of Visible-String	O
0006	ProductFamily	R	max. 58 Byte	Visible-String	O
0007	ProductName	R	max. 58 Byte	Visible-String	M
0008	SerialNo	R	max. 58 Byte	Visible-String	O
0009	ProductText	R	max. 58 Byte	Visible-String	O
000A	OrderNumber	R	max. 58 Byte	Visible-String	M
000B	HardwareVersion	R	2 Einträge	Record	M
000C	FirmwareVersion	R	2 Einträge	Record	M
000D	PChVersion	R	2 Einträge	Record	M
000E	CommProfile	R	3+1 Byte	Visible-String	M
000F	DeviceProfile	R	max. 4+1 Byte	Visible-String	M
0010	Reserved				
0011	ProfileVersion	R	2 Einträge	Record	M
0012	VendorURL	R	max. 58 Byte	Visible-String	O
0013	DeviceDescFile	R	max. 58 Byte	Visible-String	O
0014	Location	R/W	max. 58 Byte	Visible-String	O
0015	EquipmentIdent	R/W	max. 58 Byte	Visible-String	O
0016	AppDeviceAddr	R/W	2 Byte	UINT16	O
0017	Language	R/W	2 Einträge	Record	M
0018	DiagState	R	6 Einträge	Record	M
0019	ResetDiag	W	1 Byte	UINT8	O
001A	GetErrorRepMethod	R/W	1 Byte	UINT8	O
001B	TestMode	R/W	2 Byte	UINT16	O
001C	ControlTrace	R/W	1 Byte	UINT8	O
001D	Password	W	max. 40 Byte	Octet-String	O
001E	SetPassword	W	3 Einträge	Record	O
001F	PDTimeout	R/W	2 Byte	UINT16	O
0020	PDTimeoutCode	R/W	N x 2 Byte	Array of UINT16	O
0021	PChTimeout	R/W	2 Byte	UINT16	O
0022	PChTimeoutCode	R/W	2 Byte	UINT16	O
0023	AbortCode	R/W	2 Byte	UINT16	O
0024	ResetCode	R/W	N x 2 Byte	Array of UINT16	O
0025	PDIN	R	PD Länge	Octet String	M
0026	PDOUT	R/W	PD Länge	Octet String	M
0027	GetExRight	R/W	1 Byte	UINT8	O
0028	ChangePDSet	R/W	2 Byte	UINT16	O
0029	ParamSetWriteControl	R/W	1 Byte	UINT8	O
002A	ConflictDictionary	R	2*N Einträge	Record	O
002B	ParamSet	R/W	2 Byte	UINT16	O
002C	ParameterMoment	R/W	2 Einträge	Record	O
002D	ResetParam	W	1 Byte	UINT8	O
002E	Checksum	R	4 Byte	UINT32	O
002F	PDOUT_Subst	R/W	PD Länge	Octet String	D
0030	PF_Code	R/W	N x 2 Byte	Array of UINT16	O
0031	PDIN_Subst	R/W	PD Länge	Octet String	D
0032	FieldBus_ID	R	2 Einträge	Record	M
0033	DiagStateChannelNo	R	2 Einträge	Record	D
0034	DiagStateAddValue	R	2 Einträge	Record	D
0035	NoOfModules	R	1 Byte	UINT8	D

0036	DeviceStructure	R/W	4 Einträge	Record	D
0037	DeviceType	R	8 Byte	Octet String	O
0038	ObjDescrReq	R/W	2 Einträge	Record	O
0039	ObjDescr	R	16 Einträge	Record	O
003A	VersionCount	R	4 Einträge	Array	M
003B	PDIN_Descr	R	4 Einträge	Array	O
003C	PDOOUT_Descr	R	4 Einträge	Array	O
003D	WakeUpTime	R	2Byte	UJINT16	O
E800	DiagStateLong	R		Domain Variable Record	O
E801	DiagHistory	R		Domain Variable Record	O
E802	DiagHistoryLong	R		Domain Variable Record	O
E803	TraceBuffer	R		Domain Variable	O
E804	LanguageAvailable	R		Domain Variable Record	O
E805	ObjDescrLong	R		Domain Variable Record	O

R = Read only, dieses Objekt kann nur gelesen werden

W = Write only, dieses Objekt kann nur geschrieben werden

R/W = Read/Write, dieses Objekt kann gelesen und geschrieben werden

M = Mandatory (Pflicht, muss in jedem Fall implementiert werden)

O = Optional (Kann, muss aber nicht implementiert werden)

D = Dependent (abhängig von anderen Objekten, muss, wenn die Abhängigkeit gegeben ist, implementiert werden))

4.3 Liste der zulässigen Kommunikations- Fehlercodes

Bei Kommunikation über Parameterkanal auftretenden Fehlermeldungen:

Der Service-Parameter Error-Type setzt sich aus folgenden Parametern zusammen:

- Error-Class;
- Error-Code;
- Additional-Code.

4.3.1 Error-Class und Error-Code

Error-Class	Error-Code	Bedeutung
2 Application-Reference		Diese Fehlerklasse bezieht sich auf die Kommunikationsbeziehung, über die der Service abgearbeitet wird.
	1 Application unreachable	Der Zugriff auf das Objekt ist aufgrund der Tatsache, dass die Applikation nicht implementiert / nicht vorhanden ist, nicht möglich. Typisch ist diese Fehlermeldung wenn über die Modulnummer auf ein nicht existierendes Modul eines modularen Gerätes zugegriffen wird. Modul nicht vorhanden oder Modulnummer falsch.
	0 other	Dieser Fehlercode wird gemeldet, wenn sich der Fehler keinem der vorher angegebenen Fehlercodes zuordnen lässt.
5 Service		Diese Fehlerklasse wird bei einem fehlerhaften Service gemeldet.
	1 Object-State-Conflict	Dieser Fehlercode wird gemeldet, wenn der augenblickliche Zustand des Objekts eine Ausführung des Services verhindert.
	2 Service-PDU-Size	Dieser Fehlercode wird bei einem PDU-Größen Problem gemeldet.
	3 Object-Constrain-Conflict	Dieser Fehlercode wird gemeldet, wenn die Ausführung des Services augenblicklich nicht möglich ist.
	4 Parameter - inconsistent	Dieser Fehlercode wird gemeldet, wenn der Service inkonsistente Parameter enthält.
	5 Illegal-Parameter	Dieser Fehlercode wird gemeldet, wenn ein Parameter einen unzulässigen Wert angenommen hat.
	0 Other	Dieser Fehlercode wird gemeldet, wenn sich der Fehler keinem der vorher angegebenen Fehlercodes zuordnen lässt.
6 Access		Diese Fehlerklasse wird bei fehlerhaftem Zugriff gemeldet.
	1 Object-invalidated	Dieser Fehlercode wird gemeldet, wenn ein Zugriff sich auf ein definiertes Objekt bezieht, das ein undefiniertes Referenz-Attribut hat. Das stellt einen permanenten Fehler für den Zugriff auf dieses Objekt dar.
	2 Hardware-Fault	Der Zugriff auf das Objekt ist aufgrund eines Hardwarefehlers fehlgeschlagen. Genauere Angabe des Grundes im Additional-Code (glob.= 8 u. 9)
	3 Object-Access-Denied	Die Zugriffsrechte des Client sind nicht ausreichend
	4 Invalid-Address	Der Zugriff erfolgt auf eine ungültige interne Adresse.
	5 Object-Attribute-Inconsistent	Ein Service-Parameter hat einen unzulässigen Wert angenommen. Genauere Angabe des Grundes im Additional-Code (glob.=1)
	6 Object-Access-Unsupported	Das Objekt ist kein Variable- Access Objekt

	7 Object-Non-Existent	Unter diesem Index existiert kein Objekt
	8 Type-Conflict	Die Daten entsprechen nicht dem Datentyp des Objektes
	0 Other	Dieser Fehlercode wird gemeldet, wenn sich der Fehler keinem der vorher angegebenen Fehlercodes zuordnen lässt.
8 Other (Application)	0 Other	Der Service wurde nicht ausgeführt. Der Grund ist herstellerspezifisch. Genauere Angabe des Grundes im Additional- Code.
	1 Profile specific	Der Service wurde nicht ausgeführt. Der Grund ist profilspezifisch. Genauere Angabe des Grundes im Additional- Code Alle in diesem Profil aufgeführten Additional Codes sind profilspezifisch.

4.3.2 Additional-Code:

Der Additional-Code setzt sich aus einem globalen, spezifischen und gerätespezifischen Anteil zusammen. Die Angabe eines globalen oder spezifischen Codes ist optional. Das bedeutet: ist keine genaue Angabe des Fehlergrundes durch die Anwendung möglich, so muss der Additional-Code = 0000 hex ausgegeben werden. Der spezifische Code enthält eine genauere Beschreibung des im globalen Codes definierten Fehlergrundes. Entspricht der Fehlergrund nicht sinngemäß einem spezifischen Code, dann muss der spezifische Code = 0 angegeben werden. Der gerätespezifische Code (Bit 8 bis 15) wird durch den Gerätehersteller festgelegt.

2 Octets mit folgendem Aufbau:

Bit 15	8	7	4	3	0
	gerätespezifisch		globaler Code		spezifischer Code

Die Werte für die Bits 8 bis 15 sind reserviert und zur Zeit mit dem Wert 0 besetzt.

Alle in diesem Profil aufgeführten Additional Codes sind profilspezifisch.

Globaler Code [hex]	spezif. Code [hex]	Bedeutung
0	0	keine genauere Angabe des Fehlergrundes
1	0	Service-Parameter mit unzulässigem Wert
1	1	Subindex nicht vorhanden
1	2	Objekt Access ist kein Request
1	3	Reserved Service Code
1	4	SubSlot nicht unterstützt
1	5	Objekt Access Type auf dieses Objekt nicht unterstützt
1	6	Objekt Access Request Index für diesen AccessType muß 0x0000 sein
1	7	Objekt Access Request Länge für diesen AccessType muß 0 sein
1	8	Objekt Länge für dieses Objekt passt nicht
1	9	Objekt ist ReadOnly und kann nicht überschrieben werden.
2	0	Dienst kann momentan nicht ausgeführt werden
2	1	Dienst kann momentan nicht ausgeführt werden, wegen Lokal-Steuerung
2	2	Dienst kann im momentanen Gerätezustand (Gerätesteuerung) nicht ausgeführt werden.
2	3	Dienst kann momentan nicht ausgeführt werden, weil kein Objektverzeichnis vorhanden ist
3	0	Wertebereich eines Parameters ist verlassen (der Server will den Wert nicht liefern)
3	1	Wert des Parameters zu groß
3	2	Wert des Parameters zu klein
4	0	Kollision mit anderen Werten
4	1	Kommunikationsobjekt kann nicht auf Prozessdaten abgebildet werden
4	2	Prozessdaten-Länge überschritten
8	0	Hardware Fehler
8	1	Anwendung ausgefallen
A	0	ungültige Segment Nummer z.B. Upload ohne Einleitung mit Subindex == 0xFF
A	1	Ressource nicht verfügbar es sind keine Ressourcen (Speicher) für den Download mehr vorhanden
A	2	falscher CRC
A	3	Fehler beim Öffnen der Datei (wenn File- System vorhanden)
A	4	Fehler beim Schreiben der Datei (wenn File- System vorhanden)
A	5	Fehler beim Schließen der Datei (wenn File- System vorhanden)
A	6	Segment fehlt Es wurden weniger Datenblöcke empfangen als im letzten Segment angegeben.
A	7	Segment zuviel Es wurden mehr Datenblöcke empfangen als im letzten Segment angegeben.
A	8	Fehler beim Lesen der Datei (wenn File- System vorhanden)
A	9	Fehler beim Lesen der Datei (wenn File- System vorhanden)
B	1	Das Passwort kann nicht ersetzt (gelöscht) werden.
B	2	Das Passwort kann nicht hinzugefügt werden (zu viele Passworte).
B	3	Das Passwort kann für die gewünschte Art des Zugriffes nicht vergeben werden.

Nicht aufgeführte Codes sind reserviert.

Sollte sich ein Hersteller mit einer Fehlermeldung nicht in dieses Schema einordnen können, wird er gebeten sich mit dem INTERBUS Club in Verbindung zu setzen.

5 Kommunikationsobjekte

Die Kommunikationsobjekte werden nicht wie bei einer Standard Parameterkanal Implementierung in einem Objektverzeichnis beschrieben, sondern müssen dem Anwender implizit, also z.B. durch eine Beschreibung der Objekte im Handbuch zum entsprechenden Gerät, bekannt sein.

5.1 Datentypen

Index of Type (dez)	Beschreibung Symbol	Anzahl Bytes
1	Boolean	1
2	INT8	1
3	INT16	2
4	INT32	4
5	UINT8	1
6	UINT16	2
7	UINT32	4
8	Floating-Point	4
9	Visible-String (immer "0x00" terminiert!)	1,2,3 ...
10	Octet- String	1,2,3 ...
11	Date	7
12	Time-Of-Day	6
13	Time-Diff	6
14	Bit-String	1,2,3 ...

5.2 Datenobjekte

5.2.1 Objekt Domain-Variable

Das Objekt Domain-Variable dient zur Übertragung von anwenderspezifisch strukturierten Daten von meist größerem bis sehr großem Umfang (größer (PDU-Size - 6) Byte). Die Objektbeschreibung muss dem Applikationsprogramm implizit bekannt sein.



Das Objekt Domain-Variable kann nur vollständig und nur über die Makrodienste Download- Write und Upload- Read angesprochen werden.

Die Gesamtgröße der Domain-Variable darf die maximale Nutzdatenmenge von $[PDU-Size - 8) * FFF0_{hex}]$ nicht überschreiten (ca. 3,5 MB).

5.2.1.1 Domain-Variable - formale Beschreibung

Object: Domain-Variable
 Objectcode: 0x02
 Key Attribute (m): Index
 Attribute (m): Length (max. PDU-Size – 8) * FFF0_{hex} Byte)
 Attribute (o): Password

Services:

- (o) Download-Write
- (o) Upload-Read

5.2.2 Objekt Simple- Variable

Das Objekt Simple- Variable stellt eine einzelne, einfache Variable dar. Die Objektbeschreibung muss dem Applikationsprogramm implizit bekannt sein. Die Größe einer Simple- Variable darf die maximale Nutzdatenmenge von (PDU-Size - 6) Bytes nicht überschreiten.

5.2.2.1 Simple-Variable - formale Beschreibung

Object: Simple-Variable
Objectcode: 0x07
Key Attribute (m): Index
Attribute (m): Length (max. (PDU-Size - 6) Byte)
Attribute (o): Password

Services:

- (o) Read
- (o) Write
- (o) Information- Report

5.2.3 Objekt Array

Das Objekt Array besteht aus einer Reihung von Simple- Variables des gleichen Datentyps. Die Objektbeschreibung muss dem Applikationsprogramm implizit bekannt sein. Soll ein Element des Objekts angesprochen werden, so muss im Service neben dem Index auch ein Subindex angegeben werden. Der Subindex 1 greift auf das 1.Element des Objekts zu. Der Subindex 0 adressiert das gesamte Objekt. Es ist keine Pflicht jedes Element eines Arrays einzeln ansprechen zu können. Die Gesamtgröße des Arrays darf die maximale Nutzdatenmenge von (PDU-Size - 6) Bytes nicht überschreiten.

5.2.3.1 Array-Variable - formale Beschreibung

Object: Array-Variable
Objectcode: 0x08
Key Attribute (m): Index
Attribute (m): Length (max. (PDU-Size - 6) Byte)
Attribute (m): Number-of-Elements
Attribute (o): Password

Services:

- (o) Read
- (o) Write
- (o) Information- Report

5.2.4 Objekt Record

Das Objekt Record besteht aus einer Reihung von Simple- Variables und ggf. einer String Variables am Ende, unterschiedlichen Datentyps. Die Objektbeschreibung muss dem Applikationsprogramm implizit bekannt sein. Das Objekt Record kann vollständig oder elementweise angesprochen werden. Soll das Objekt vollständig angesprochen werden, so wird im Service der zugehörige Index angegeben.

Soll ein Element des Objekts angesprochen werden, so muss im Service neben dem Index auch ein Subindex angegeben werden. Der Subindex 1 greift auf das 1. Element des Objekts zu.

Es ist keine Pflicht jedes Element eines Records einzeln ansprechen zu können.

Die Gesamtgröße des Records darf die maximale Nutzdatenmenge von (PDU-Size - 6) Byte nicht überschreiten.

5.2.4.1 Record- Variable - formale Beschreibung

Object:	Record-Variable
Objectcode:	0x09
Key Attribute (m):	Index
Attribute (m):	Length (max. (PDU-Size - 6) Byte)
Attribute (m):	List-of-Elements
Attribute (o):	Password

Services:

- (o) Read
- (o) Write
- (o) Information-Report

5.2.5 Objekt Variable- List

Eine Variablen- Liste enthält eine Zusammenstellung einzelner Variablen. Die Struktur muß dem Applikationsprogramm implizit bekannt sein.



Da die zur Variablen- Liste gehörenden Werte schnell die maximale Größe eines Telegramms überschreiten können, muss der Zugriff auf Objekte des Datentyps Variablen- Liste über die Makrodienste Download- Write und Upload- Read erfolgen.

Das Zusammenfassen mehrerer Einzelvariablen zu eine Variablen- Liste ist nur für Variablen kleiner Datentypen (z.B. UINT8, Integer32, etc.) sinnvoll. Theoretisch kann die Länge einer einzelnen Variable einer Variablen- Liste bis zu maximal (PDU-Size – 8) Byte betragen. Auf eine solche Variable kann aber effektiver mit einem normalen Read- Dienst zugegriffen werden.

5.2.5.1 Variable- List - formale Beschreibung

Object:	Variable-List
Objectcode:	0x0A
Key Attribute (m):	Index
Attribute (m):	Number-Of-Elements
Attribute (m):	List of Element-Index
→ Attribute (m):	→ Index
Attribute (o):	Password

Services:

- (o) Download-Write
- (o) Upload-Read

5.2.5.2 Statische Variable-List

Die Variablenlisten können statisch durch den Gerätehersteller angelegt sein. Diese Variablenlisten liegen in dem Indexbereich "statische Variablenlisten" siehe Indexbereiche der Kommunikationsobjekte.

Die Struktur der statischen Variablenliste ist über die Variable "VariableListRecord" (Index E7FF) auslesbar. Hierzu wird zunächst der Index, der zulesenden Variablenliste auf das Objekt E7FF geschrieben. Hierbei ist der Download-Write zu verwenden, da Objekt E7FF ein Domain-Objekt ist. Anschließend kann durch einen Upload-Read-Zugriff auf Objekt E7FF die „VariableListRecord“ ausgelesen werden.

Die Übertragung der „VariableListRecord“ erfolgt nach folgendem Aufbau:

Index (2 Bytes)	Bedeutung
1. Wert	Index der 1. Variable in der Liste
2. Wert	Index der 2. Variable in der Liste
...	
...	
...	
n. Wert	Index der n. Variable in der Liste

5.2.5.3 Dynamische Variable-List

Dynamische Variablenlisten werden über die Variable "VariableListRecord" (Index E7FF) angelegt. Diese Variablenlisten liegen in dem Indexbereich "dynamische VariablenListen" (siehe Indexbereiche der Kommunikationsobjekte). Durch einen Download-Write-Zugriff auf Objekt E7FF wird die „VariableListRecord“ angelegt.

Die Übertragung der „VariableListRecord“ erfolgt nach folgendem Aufbau:

Index (2 Bytes)	Bedeutung
1. Wert	Index unter der die VariablenListRecord angelegt werden soll
2. Wert	Index der 1. Variable in der Liste
3. Wert	Index der 2. Variable in der Liste
...	
...	
...	
n. Wert	Index der n. Variable in der Liste

5.2.5.4 Übertragungsformat Variablenlisten

Zum Schreiben/Lesen einer Variablenliste werden alle Werte der Einzelvariablen nacheinander im Parameter Daten des Download-Write/Upload-Read eingetragen.

	Bedeutung
1. Wert	Inhalt der 1. Variable in der Liste
2. Wert	Inhalt der 2. Variable in der Liste
...	
...	
n. Wert	Inhalt der n. Variable in der Liste

5.2.6 Objekt String-Variable

Das Objekt String-Variable stellt eine einzelne, einfache Variable dar, die durch einen bestimmten Datentyp (Octet-String, Visible-String oder Bit-String) charakterisiert wird.

Die Objektbeschreibung des Variable Access-Objekts String-Variable ist statisch im Objektverzeichnis (S-OV) hinterlegt. Durch die Objektbeschreibung des Objekts String-Variable wird die Abbildung einer String-Variablen auf eine real existierende String-Variable im Anwendersystem festgelegt.

In der Beschreibung des Objekts String-Variable ist nur ein Datentyp aus Octet-String, Visible-String oder Bit-String zugelassen.

Eine String-Variable hat den gleichen Aufbau wie eine Simple-Variable.

Der Unterschied besteht in der variablen Länge der Datentypen. Im Objektverzeichnis wird die Maximallänge projektiert.

5.2.6.1 String-Variable - formale Beschreibung

Object: Array-Variable
Objectcode: 0x0B
Key Attribute (m): Index
Attribute (m): Max Length (max. (PDU-Size - 6) Byte)
Attribute (o): Password

Services:

- (o) Read
- (o) Write
- (o) Information- Report

6 Services

Der Zugriff auf alle einfachen Variablen, Arrays und Records erfolgt mit den Standard-Diensten, als da wären z.B.

- Read
- Write
- Information Report.

Da diese Dienste zur Funktionalität der Parameterkanal Implementierung gehören, werden Sie nicht an dieser Stelle beschrieben, sondern es wird auf die entsprechende Parameterkanal- Dokumentation)* verwiesen.

Auf die Objekte Domain-Variable und Variable-List wird hingegen mit den Makrodiensten Download-Write und Upload-Read zugegriffen. Deren Handhabung wird im folgenden beschrieben.

)* z.B. IBS PCP RE HB (5052b.pdf) und IBS PCP RR HB (5054b.pdf)

6.1 Download-Funktionalität mit Write-Service (Download-Write)

Die Übertragung von großen Datenmengen (z.B. Applikationsprogramm, Firmware-Update, Variablen-Listen usw.) wird im Gegensatz zur Standard Parameterkanal Implementierung (enthält Download-Services) mit Hilfe des Write-Services abgewickelt. Hierzu wird in den Nutzdaten des Write-Services ein Download-Protokoll definiert, welches eine segmentierte Übertragung größerer Datenmengen erlaubt. Der Download kann nur auf Domain-Variable und Variablen-Listen durchgeführt werden.

Parameter	Req/Ind	Rsp/Cnf
Kommunikationsreferenz	M	M
Module Number	M	M
Index	M	
Subindex	O	
Datenfeld	M	
Segment-Nummer	M	
Daten	M	
Result(+)		S
Result(-)		S
Error-Type		M

Kommunikationsreferenz

Die Kommunikationsreferenz adressiert den gewünschten Kommunikationspartner. („Compact“ implementierte Teilnehmer können nur mit dem Master kommunizieren (kein Peer-to-Peer), so dass die Kommunikationsreferenz immer die 2 ist.)

Module Number (Invoke-ID)

Zur Adressierung von Modulen eines modularen Gerätes wird die bis dato nicht genutzte und nur aus Kompatibilitätsgründen vorhandene Invoke-ID benutzt. Aus diesem Grund wird auch der Name dieses Parameters in „Module Number“ geändert.

Die Handhabung des Parameters ändert sich hingegen nicht. In der Response wird weiterhin die Modulnummer der Indication eingetragen.

Näheres im Kapitel „Modulare Geräte“.

Index

Der Parameter Index adressiert das zuschreibende Objekt.

Subindex

Der Subindex hat für die Download- Funktionalität den Wert „0“

Datenfeld

Der Parameter Datenfeld enthält das Downloadprotokoll.

Datenfeld	
Segment-Nummer 2 Byte Download Steuerung	Nutzdaten 1 .. (PDU-Size – 8) Byte Download Datenblöcke z.B. 1 .. 56 Byte für die „Compact“ Implementierung

Segment-Nummer

Dieser Parameter enthält die Nummer des zu schreibenden Datenblocks, wobei folgende Definition gilt:

Segment-Nummer = 1	Einleiten einer Downloadsequenz mit dem ersten Datenblock.
Segment-Nummer = 2..0xFFFF _{hex} Segment-Nummer = 0xFFFFD _{hex}	Datenblock-Nummer n letzter Datenblock enthält ausschließlich <ul style="list-style-type: none"> die Anzahl der übertragenen Datenblöcke (ohne diesen Schlussdatenblock) (2 Byte) und das CRC16 Rest Polynom¹⁾ (2 Byte)
Segment-Nummer = 0xFFFFE _{hex}	letzter Datenblock enthält ausschließlich <ul style="list-style-type: none"> die Anzahl der übertragenen Datenblöcke (ohne diesen Schlussdatenblock) (2 Byte) und das CRC32 Rest Polynom¹⁾ (4 Bytes)
Segment-Nummer = 0xFFFFF _{hex}	letzter Datenblock enthält ausschließlich <ul style="list-style-type: none"> die Anzahl der übertragenen Datenblöcke (ohne diesen Schlussdatenblock) (2 Byte) und kein CRC Rest

¹⁾ Der CRC wird immer über die gesamten reinen Nutzdaten gebildet. Die Segmentnummer gehört nicht zu den Nutzdaten.

Der Download-Server muss die lückenlos aufsteigende Reihenfolge der Segmentnummer kontrollieren, und im Fehlerfall dem Client (typ. Feldbus Master) eine negative Response senden. Im Falle einer neg. Response kann der Client das korrekte Segment senden oder die Übertragung mit einem Schlusssegment beenden.

Nur ein Schlussdatenblock beendet den Download Service.

Daten

Dieser Parameter enthält die Datenblöcke. Die Länge dieser Datenblöcke ist abhängig von der maximalen PDU- Größe. Für die PDU- Größe von 64 Bytes beträgt die maximale Datenblocklänge 56 Bytes. Die maximale Datenblocklänge muß nicht ausgenutzt werden.

Result(+)

Der Parameter Result(+) kennzeichnet ein positives Ergebnis. Nur bei einem positiven Ergebnis darf die Segmentnummer weiter gezählt werden.

Result(-)

Der Parameter Result(-) kennzeichnet ein negatives Ergebnis.

Error-Type

Der Parameter Error-Type enthält die Fehlerursache.

Der Inhalt des Parameter Error-Type entspricht dem des Write-Service, mit folgender Definition für die spezifischen Download-Sequenzfehler:

Error-Class „8“ - Other

Error-Code „1“ – Profile specific

Additional-Code (hex)	Bedeutung
0x00A0 Invalid Segment Number - Segment missing	ungültige Segment Nummer (Segment fehlt)
0x00A1 Resource Unavailable	es sind keine Ressourcen (Speicher) für den Download mehr vorhanden
0x00A2 Invalid CRC	falscher CRC
0x00A3 File Open Error	Fehler beim Öffnen der Datei (wenn File-System vorhanden)
0x00A4 File Write Error	Fehler beim Schreiben der Datei (wenn File-System vorhanden)
0x00A5 File Close Error	Fehler beim Schließen der Datei (wenn File- System vorhanden)
0x00A6 Segment missing	Es wurden weniger Datenblöcke empfangen als im letzten Segment angegeben.
0x00A7 Segment overrun	Es wurden mehr Datenblöcke empfangen als im letzten Segment angegeben.
0x00A9 Invalid Segment Number – double Segment	ungültige Segment Nummer (Segment doppelt, Segment wurde ignoriert)

6.2 Upload-Funktionalität mit Read-Service (Upload-Read)

Das Auslesen von großen Datenmengen (z.B. Applikationsprogramm, Backups, Variablen-Listen usw.) wird im Gegensatz zur Standard- Parameterkanal Implementierung (enthält Upload-Services) mit Hilfe des Read- Services abgewickelt. Hierzu wird in dem Subindex des Read- Service ein Upload-Protokoll definiert, welches eine segmentierte Übertragung größerer Datenmengen erlaubt. Der Upload kann nur auf Domain-Variable und Variablen-Listen durchgeführt werden.

Parameter	Req/Ind	Rsp/Cnf
Kommunikationsreferenz	M	M
Module Number	M	M
Index	M	
Subindex	M	
Result(+)		S
Datenfeld		M
Segment-Nummer		M
Daten		M
Result(-)		S
Error-Type		M

Kommunikationsreferenz

wie Download-Funktionalität mit Write-Service (Download-Write)

Module Number

wie Download-Funktionalität mit Write-Service (Download-Write)

Index

wie Download-Funktionalität mit Write-Service (Download-Write)

Subindex

Über einen speziellen Subindex wird eine Upload-Sequenz eingeleitet, wobei folgende Definition gilt:

Subindex = $0xFF_{hex}$ Einleiten einer Upload-Sequenz mit Anfordern des ersten Datenblocks

Subindex = $0x00_{hex}$ Anforderung des nächsten Datenblocks

Subindex = $0x01_{hex}$ Anforderung des jeweils vorigen Datenblocks (z.B. Nach einem Fehler)

Result(+)

Der Parameter Result(+) kennzeichnet ein positives Ergebnis. Nur bei einem positivem Ergebnis darf die Segmentnummer weiter gezählt werden.

Datenfeld

Der Parameter Datenfeld enthält das Uploadprotokoll.

Datenfeld	
Segment-Nummer 2 Byte Upload Steuerung	Nutzdaten 1 .. (PDU-Size – 8) Byte Download Datenblöcke 1 .. 56 Byte für die „Compact“ Implementierung

Segment-Nummer

Dieser Parameter enthält die Nummer des zu lesenden Datenblocks, wobei folgende Definition gilt:

Segment-Nummer = 1	Einleiten einer Uploadsequenz mit dem ersten Datenblock.
Segment-Nummer = 2..0xFFFF _{hex}	Datenblock-Nummer n
Segment-Nummer = 0xFFFD _{hex}	letzter Datenblock enthält ausschließlich <ul style="list-style-type: none"> die Anzahl der übertragenen Datenblöcke (ohne diesen Schlussdatenblock) (2 Byte) und das CRC16 Rest Polynom¹⁾ (2 Byte)
Segment-Nummer = 0xFFFE _{hex}	letzter Datenblock enthält ausschließlich <ul style="list-style-type: none"> die Anzahl der übertragenen Datenblöcke (ohne diesen Schlussdatenblock) (2 Byte) und das CRC32 Rest Polynom¹⁾ (4 Bytes)
Segment-Nummer = 0xFFFF _{hex}	letzter Datenblock enthält ausschließlich <ul style="list-style-type: none"> die Anzahl der übertragenen Datenblöcke (ohne diesen Schlussdatenblock) (2 Byte) und kein CRC Rest

¹⁾ Der CRC wird immer über die gesamten reinen Nutzdaten gebildet. Die Segmentnummer gehört nicht zu den Nutzdaten.

Der Upload-Client muss die lückenlos aufsteigende Reihenfolge der Segmentnummer kontrollieren. Im Fehlerfall kann der Client dann das letzte Segment mit dem Subindex 0xFE erneut anfordern oder den Upload erneut mit Subindex 0xFF einleiten.

Daten

Dieser Parameter enthält die Datenblöcke. Die Länge dieser Datenblöcke ist abhängig von der maximalen PDU- Größe. Für die PDU- Größe von 64 Bytes beträgt die maximale Datenblocklänge 56 Bytes. Die maximale Datenblocklänge muß nicht ausgenutzt werden.

Result(-)

Der Parameter Result(-) kennzeichnet ein negatives Ergebnis.

Error-Type

Der Parameter Error-Type enthält die Fehlerursache.

Der Inhalt des Parameter Error-Type entspricht dem des Read- Service, mit folgender Definition für die spezifischen Upload-Sequenzfehler:

Error- Class "8" - Other

Error- Code „1“ - Profile specific

Additional-Code (hex)		Bedeutung
0x00A0	Invalid Segment Number - Segment missing	Upload ohne Einleitung mit subindex == 0xFF
0x00A3	File Open Error	Fehler beim Öffnen der Datei (wenn File- System vorhanden)
0x00A5	File Close Error	Fehler beim Schließen der Datei (wenn File- System vorhanden)
0x00A8	File Read Error	Fehler beim Lesen der Datei (wenn File- System vorhanden)

6.3 Beispiel Download Variablen- Liste

Ein „Compact“ Gerät verfügt in diesem Beispiel unter anderen über folgende Objekte, die zu einer Variablen-Liste zusammen gestellt werden.

Index	Subindex	Data-Type	Length	Example-Value (hex)
0080	0	UINT8	1	FF
0081	7	UINT16	2	12 34
0090	3	UINT8	1	00
0091	0	Octet-String	16	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0100	0	Octet-String	16	10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
0101	0	Octet-String	16	20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F
0105	0	Octet-String	16	30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F

Download-Write

1. Download-Write-Request/Indication

Parameter	Inhalt (hex)
Kom.Ref.	02
Module Number	00
Index	E0 00
Subindex	00
Segmentnr.	00 01
Daten	FF 12 34 00 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F

2. Download-Write-Request/Indication

Parameter	Inhalt (hex)
Kom.Ref.	02
Module Number	00
Index	E0 00
Subindex	00
Segmentnr.	00 02
Daten	30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F

3. Download-Write-Request/Indication

Parameter	Inhalt (hex)
Kom.Ref.	02
Module Number	00
Index	E0 00
Subindex	00
Segmentnr.	FF FF
Daten	00 02

7 Standardobjekte

Die folgenden Objekte werden in der Form beschrieben:

Index (hex)	Eindeutiges Kennzeichen des Objektes Der Index wird für die Dienste zum Zugriff auf das Objekt genutzt
Objektname	Sinnfälliger Name des Objekts
R/W	Zugriffbeschränkung R = nur lesend W = nur schreibend R/W = keine Zugriffsbeschränkung
Länge	Länge des Objekts in Byte
Datentyp	Datentyp des Objektes, s.o.
Bedeutung	Erklärung zum Objekthalt
M/O	Implementierungsanweisung M = Mandatory Dieses Objekt muss implementiert werden. O = Optional Dieses Objekt kann implementiert werden. O = Dependent Dieses Objekt muss implementiert werden, wenn eine bestimmte optionale Funktionalität implementiert ist

Um den einheitlichen Zugriff durch Tools zu gewährleisten, müssen die Objekte, wenn sie existieren, generell formal genau so implementiert werden, wie sie im folgenden definiert sind. Der Gerätehersteller kann den Inhalt jedoch nach seinen Anforderungen anpassen. Z.B. Könnte ein aus technischen Gründen nicht durch die Gerätefirmware bestimmbarer Hardwarestand, so dargestellt werden:

000B	HardwareVersion	R	2 Einträge	Record	Hardware Version (Gerät oder Kommunikationsmodul)	M
.1	• BuildDate	R	10+1 Byte	Visible-String	0000-00-00	M
.2	• Version	R	max. 40 Byte	Visible-String	Not available	M

7.1 Identifikation

Die in den folgenden Objekten festgelegten Informationen beschreiben das Gerät selbst, den Hersteller und den Einsatzfall des Gerätes. Bei Auslieferung müssen die eingetragenen Daten mit dem Aufdruck auf dem Gerät übereinstimmen.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
0001	VendorName	R	max. 58 Byte	Visible-String	Herstellername z.B. Phoenix Contact GmbH & Co KG	M
0002	VendorID	R	6+1 Byte	Visible-String	Eindeutige Herstellerkennung Organizationally Unique Identifiers (OUI) gemäß: http://standards.ieee.org/regauth/oui/index.shtml z.B. "00A045"	O
0003	VendorText	R	max. 58 Byte	Visible-String	Bemerkung zum Hersteller z.B. Adresse, Branche etc.	O
0012	VendorURL	R	max. 58 Byte	Visible-String	URL des Herstellers z.B. http://www.phoenixcontact.com	O
0007	ProductName	R	max. 58 Byte	Visible-String	herstellerspezifische eindeutige Produktbezeichnung z.B. IB IL 24 DO 8	M
0008	SerialNo (ProductId)	R	max. 58 Byte	Visible-String	Seriennummer herstellerspezifische eineindeutige Geräteerkennung z.B. 123456789	O
0009	ProductText	R	max. 58 Byte	Visible-String	herstellerspezifischer Produkttext z.B. digitales Ausgabemodul	O
000A	OrderNumber	R	max. 58 Byte	Visible-String	herstellerspezifische eindeutige Produkttypkennung z.B. Bestellnummer z.B. 2726269	M
0037	DeviceType	R	8 Byte	Octet String	Herstellerspezifische Geräte-/ Modulidentifikation ungleich „0“ Die herstellerepezifische Geräte-/ Modulidentifikation dient dazu gleichartige Geräte innerhalb einer Konfiguration austauschen und betreiben zu können. Z.B. kann ein 16 kanaliges Ausgangsmodul mit Schraubanschlusstechnik sehr wohl durch eins mit Zugfederanschlusstechnik ersetzt werden, obwohl es nicht die gleiche Artikelnummer besitzt. Eine andere Funktionalität erfordert auch einen anderen DeviceType. Wird dieses Objekt in einem modularen Gerät benutzt, kann es auch einen Platzhalter kennzeichnen. „0“ wenn Modul (noch) nicht gesteckt ist. In diesem Fall kann trotzdem eine Abbildung auf die Prozessdaten erfolgen.	O
0032	FieldBus_ID	R	2 Einträge	Record	Feldbus Identifikationinformation	M
.1	• ID-Code	R	1 Byte	UINT8	Feldbus spezifischer ID- Code (8 Bit, typ. dargestellt als dezimale Zahl)	M
.2	• PDLenght	R	2 Byte	UINT16	Anzahl der Prozessdatenbits (Einheit Bit)	M
000B	HardwareVersion	R	2 Einträge	Record	Hardware Version (Gerät oder Kommunikationsmodul) <i>Bei Auslieferung müssen die eingetragenen Daten mit dem Aufdruck auf dem Gerät übereinstimmen.</i>	M
.1	• BuildDate	R	10+1 Byte	Visible-String	Herstelldatum Datum der Version Format YYYY-MM-DD Gemäß ISO 8601 z.B. 2002-11-29	M

.2	• Version	R	max. 40 Byte	Visible-String	Versionskennung z.B. 4.01Beta Kunde	M
000C	FirmwareVersion	R	2 Einträge	Record	Firmware Version (Gerät oder Kommunikationsmodul) <i>Bei Auslieferung müssen die eingetragenen Daten mit dem Aufdruck auf dem Gerät übereinstimmen.</i>	M
.1	• BuildDate	R	10+1 Byte	Visible-String	BuildDate Datum der Version Format YYYY-MM-DD Gemäß ISO 8601 z.B. 2002-05-03 Ist keine FW vorhanden wird „0000-00-00“ eingetragen	M
.2	• Version	R	max. 40 Byte	Visible-String	Versionskennung z.B. 1.03-Kunde Ist keine FW vorhanden wird „-“ eingetragen	M
000D	PChVersion	R	2 Einträge	Record	Parameter Channel Version Implementierungsversion des Parameterkanals	M
.1	• BuildDate	R	10+1 Byte	Visible-String	BuildDate Datum der Version Format YYYY-MM-DD Gemäß ISO 8601 z.B. 2002-05-03 Ist kein Parameterkanal vorhanden wird „0000-00-00“ eingetragen	M
.2	• Version	R	max. 40 Byte	Visible-String	Versionskennung z.B. "PCP- Compact V1.00" Ist kein Parameterkanal vorhanden wird „-“ eingetragen	M
0005	Capabilities	R	Array of 8 Bytes	Visible-String	Eigenschaften / Funktionalitäten über die das Gerät, neben den Basisfunktionen, verfügt. Alle derzeit existierenden Funktionalitäten sind im Folgenden aufgeführt: <ul style="list-style-type: none"> • „Nothing“ - Dieser Eintrag definiert keine Zusatzfunktionen. Der Eintrag kann als Platzhalter auch mehrfach vorhanden sein. • „Syncl_0“ - Der Slave unterstützt die Synchronisation der Eingänge. • „SyncO_0“ - Der Slave unterstützt die Synchronisation der Ausgänge. Der Type ist genau 8 Zeichen lang. Nicht benutzte Zeichen sind mit 0x00 aufzufüllen. Wenn es keine Eigenschaften / Funktionalitäten im Gerät, über die Basisfunktionen hinaus gibt, muß dieses Objekt nicht implementiert werden. Wenn es Eigenschaften / Funktionalitäten im Gerät, über die Basisfunktionen hinaus gibt, muß dieses Objekt implementiert werden. Unterstützt der Slave das Objekt Capabilitis, aber keine weitere Zusatzfunktion, ist mindestens einmal der Eintrag „Nothing“ vorhanden.	D
0006	ProductFamily	R	max. 58 Byte	Visible-String	Herstellerspezifische Produktfamilie z.B. Inline	O

003D	WakeUpTime	R	2Byte	UINT16	Wake Up Time Zeit vom Einschalten der Versorgungsspannung (ggf. auch bei mehreren der letzten relevanten) zur Herstellung der Betriebsbereitschaft. Er dient zur Unterstützung der diversen Energieeinsparprofile. Der Parameter muß nur angelegt werden, wenn die WakeUpTime > 500ms ist. Ist dieser Parameter nicht vorhanden, wird von einem Defaultwert von < 500ms ausgegangen.	D
0004	DeviceFamily	R	max. 58 Byte	Visible-String	Gerätefamilie (z.B. I/O, Antrieb) lt. Spezifikation s.u., z.B. motion control / frequency inverters	M
000E	CommProfile	R	Max. 4+1 Byte	Visible-String	Kommunikationsprofil Systemspezifische Profilkennung s.u. z.B. 634	M
000F	DeviceProfile	R	Max. 4+1 Byte	Visible-String	Geräteprofil Kennzeichnung des Applikationsprofils s.u. z.B. „10“ für das Basisprofil oder „22“ für das Drivecom Profil	M
0011	ProfileVersion	R	2 Einträge	Record	Versionsbezeichnung dieses Profils	M
.1	• BuildDate	R	10+1 Byte	Visible-String	„2011-12-07“	M
.2	• Version	R	max. 40 Byte	Visible-String	„Basis - Profil V2.0“	M
0013	DeviceDescFile	R	max. 58 Byte	Visible-String	Dateiname der FDCML Gerätebeschreibungsdatei, z.B. XYZ.xml	O
003A	VersionCount	R	4 Einträge	Array	Versionszähler Eindeutige, fortlaufende Nummerierung der Version der entsprechenden Komponenten mit den Indices: • 0x0011 • 0x000D • 0x000B • 0x000C Wenn die entsprechende Komponente geändert wird, muß der dazugehörige VersionCount um 1 erhöht werden.	M
.1	• ProfileVersion	R	2 Byte	UINT16	07 Für dieses Profil.	M
.2	• PChVersion	R	2 Byte	UINT16	z.B. 01 für die Version "Compact" Implementierung	M
.3	• HardwareVersion	R	2 Byte	UINT16	z.B. 02 für die Target Hardware	M
.4	• FirmwareVersion	R	2 Byte	UINT16	z.B. 05 für die Target Firmware	M
0014	Location	R/W	max. 58 Byte	Visible-String	Einbauort Text, der vom Gerätebenutzer in diesem Parameter hinterlegt wurde. Dieser Text kennzeichnet der Einbauort des Gerätes und wird in einem nichtflüchtigen Speicher abgelegt. z.B. Maschine 1, hinten links	O
0015	EquipmentIdent	R/W	max. 58 Byte	Visible-String	Betriebsmittelkennzeichen Text, der vom Gerätebenutzer in diesem Parameter hinterlegt wurde. Die Gerätebeschreibung wird in einem nichtflüchtigen Speicher abgelegt. Hier kann z.B. der Gerätebenutzer eine Beschreibung über die Gerätebenutzung in der Anlage hinterlegen. z.B. M1H747h.l.	O

0016	AppIDeviceAddr	RW	2 Byte	UINT16	Applikations- Geräte- Adresse (Anwenderdefinierte Gerätenummer) Beliebige Kennung dieses Gerätes in dieser speziellen Applikation. Diese Nummer muß nicht eindeutig sein. Die Verwaltung obliegt ausschließlich dem Endanwender. z.B. 123	O
------	----------------	----	-----------	--------	---	---

7.1.1 DeviceFamily

Eindeutige Kennzeichnung der Gerätefamilie.

en	de
Actuator	Aktor
Bus Coupler	Buskoppler
Closed Loop Controller	Regler
Dosing Device	Dosiergerät
Drive	Antrieb
Drive - Frequency Inverter	Antrieb - Frequenzumrichter
Drive – Motor Starter	Antrieb - Motorschalter
Drive – Servo Amplifier	Antrieb - Servoverstärker
Drive – Stepper Motor Controller	Antrieb - Schrittmotor-Steuerungcontroller
Encoder	Encoder
Gateway	Gateway
General	Allgemeines
HMI	HMI
HMI Display	HMI-Anzeige
HMI Operator Panel	HMI-Bediengerät
Hydraulic Device	Hydraulik-Gerät
I/O	E/A
I/O analog IN / OUT	E/A analog IN / OUT
I/O analog IN	E/A analog IN
I/O analog OUT	E/A analog OUT
I/O digital IN / OUT	E/A digital IN / OUT
I/O digital IN	E/A digital IN
I/O digital OUT	E/A digital OUT
I/O Function Module	E/A-Funktionsmodul
Identification System	Identifikationssystem
Media Converter active	Medienkonverter aktiv
Media Converter passive	Medienkonverter passiv
NC	NC
NC/RC	NC/RC
PC	PC
PC Board	PC-Karte
PLC	SPS
PLC board	SPS-Karte
Pneumatic Device	Pneumatik-Gerät
Positioning Controller	Positionier-Steuerung
Power Supply	Stromversorgung
Robot Control	Roboter
Sensor	Sensor
Switching Device	Schaltgerät
Technology Controller	Technologie-Steuerung
Valve	Ventil
Weighing or Batching System	Wiege- oder Dosiersystem
Welding Controller	Schweißsteuerung
Wrenching Controller	Schraubersteuerung

Sollte sich ein Hersteller in dieses Schema nicht einordnen können, wird er gebeten sich mit dem INTERBUS Club in Verbindung zu setzen.

7.1.2 CommunicationProfile

Dieser Parameter enthält eine systemspezifische Kommunikationsprofilkennung
Für die „Compact“ Implementierung kommen folgende Kommunikationsprofile in Betracht:

63	Implementierung	Kanaltyp	Geeignet für
632	Compact	Feldbus- Management, Compact- Parameterkanal	keine zyklischen Prozessdaten, kein Down/Up-Load von Variablenlisten, Programmen etc.
633	Compact Prozessdaten	Feldbus- Management zyklische Prozessdaten Compact- Parameterkanal	Parametrierbare Geräte, nur Parameter, kein Down/Up-Load von Variablenlisten und Programmen etc.
634	Compact Prozessdaten Up- / Download- Protokoll	Feldbus- Management, zyklische Prozessdaten Compact- Parameterkanal Download,Upload	Komplexe Geräte, Download/Up-Load von Variablenlisten
635	Compact Up- / Download- Protokoll	Feldbus- Management, Compact- Parameterkanal Download, Upload	Komplexe Geräte, Download/Up-Load von Variablenlisten, kein zyklischen Prozessdaten

7.1.3 DeviceProfile

Dieser Parameter enthält Informationen über das implementierte Deviceprofil des Gerätes.
Aufbau des Parameters:

B15b12	b11b8	b7b4	b3b0
Profilgruppe			Version

DeviceProfile	Bedeutung
0000	Kein Profil
0010	Basisprofil (Grundlage aller anderen Profile)
0012	Sensor/Aktor
0020	DRIVECOM nur Prozessdaten
0021	DRIVECOM Frequenzumrichter
0022	DRIVECOM Servo
0030	Reserve
0040	Anschaltbaugruppen
0050	reserve
0060	reserve
0070	Encoder
0080	Prozessregler
0090	Robotersteuerungen
00A0	Schraubersteuerungen
00B0	ISO-Ventile
00C0	Schweißsteuerungen
00D0	Bedien- / Anzeigegeräte
00E0	Hydraulische Geräte
FFFF	mehr als ein Applikationsprofile

Unterstützt ein Gerät mehr als ein Geräteprofil, dann wird FFFF_{hex} das in dem Parameter Geräteprofil eingetragen.

7.2 Geräte- Diagnose

Diese Objekte dienen dazu der Applikation Diagnoseinformationen über den Zustand des Gerätes und evtl. angeschlossener Peripherie mitzuteilen. Die aktuelle Diagnoseinformation des Gerätes wird im Objekt „DiagState“ abgelegt. Außerdem kann diese Information in der Langform in der Domainvariablen (Index E800) abgelegt werden.

Eine Historie der Diagnoseinformationen findet sich in der Domainvariablen „DiagHistory“ bzw. „DiagHistoryLong“ für die Langform.

7.2.1 Objekte

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
0018	DiagState	R	6 Einträge	Record	aktueller Diagnosestatus des Gerätes	M
.1	• Lfd.Nr.	R	2 Byte	UINT16	Eindeutige, fortgezählte Fehlernummer seit dem letzten Power-Up-Reset oder HistoryReset	M
.2	• Priority	R	1 Byte	UINT8	Priorität der Meldung „1“ ist die höchste Priorität	M
.3	• Channel/Group/Module	R	1 Byte	UINT8	Kanal, Gruppe oder Modul auf dem die Störung aufgetreten ist. Kanal/Gruppe/Modul „FF“ betrifft das ganze Gerät. Zusätzliche Informationen siehe „MoreFollows“	M
.4	• Code	R	2 Byte	Octet String	Störungscode s.u.	M

.5	<ul style="list-style-type: none"> MoreFollows 	R	1 Byte	Bit-String 8	<p>= „00_{hex}“ Es handelt sich beim SubIndex .3 „Channel/ Group/ Module“ um eine Kanalnummer.</p> <p>= „01_{hex}“ Es gibt weitere Informationen zu diesem Fehler die über das Objekt „DiagStateLong“ auslesbar sind.</p> <p>= „02_{hex}“ Zu der Diagnose gibt es einen Zusatzwert. Dieser Wert ist über das Objekt DiagStateAddValue auslesbar.</p> <p>= „04_{hex}“ Es handelt sich bei SubIndex .3 „Channel/ Group/ Module“ um eine Gruppennummer. Der Kanal, so eine Unterscheidung möglich ist, kann dann optional im Objekt DiagStateChannelNumber ausgelesen werden.</p> <p>= „08_{hex}“ Es handelt sich um ein modulares Gerät. SubIndex .3 „Channel/ Group/ Module“ gibt das betroffene Modul an. Über einen weiteren Zugriff auf dieses Objekt (0x0018) mit der entsprechenden Modul Nummer, können Gruppe und ggf. Kanal ausgelesen werden.</p>	M
.6	<ul style="list-style-type: none"> Text 	R	Max 51 Byte	Visible-String	Ersten max. 51 Zeichen der Meldung was aufgetreten ist. Default: „Status OK“	M
E800	DiagStateLong	R		Domain Variable Record	aktuelle Diagnoseinformationen des Gerätes in der Langform	O
Seg.1	<ul style="list-style-type: none"> Lfd.Nr. 				- 0018.1	D
-a	<ul style="list-style-type: none"> Priority 				- 0018.2	D
-b	<ul style="list-style-type: none"> Channel/Group 				- 0018.3	D
.c	<ul style="list-style-type: none"> Code 				- 0018.4	D
-d	<ul style="list-style-type: none"> MoreFollows 				- 0018.5	D
-e	<ul style="list-style-type: none"> Date 	R	10+1 Byte	Visible-String	Datum an dem die Störung aufgetreten ist. YYYY-MM-DD 0000-00-00 = kein Datum vorhanden	D
-f	<ul style="list-style-type: none"> Time 	R	8+1 Byte	Visible-String	Uhrzeit zu der die Störung aufgetreten ist. hh:mm:ss 00:00:00 = keine Uhrzeit	D
-g						

-h	<ul style="list-style-type: none"> TimeofOperation 	R	4 Byte	UINT32	Betriebsstundenzählerstand absolut zu der die Störung aufgetreten ist. 0 .. 4294967295Sekunden rund 136 Jahre = 0 kein Betriebsstundenzähler vorhanden	D
-i	<ul style="list-style-type: none"> ParamSet 	R	2 Byte	UINT16	Aktuell gültige Parametersatznummer bei der die Störung aufgetreten ist. 0000 = keine spezielle Parametersatznummer vorhanden	D
-j	<ul style="list-style-type: none"> TraceData 	R	2 Byte	UINT16	Startindex evtl. Zur Meldung zugehöriger Tracedaten. = 0000 keine Tracedaten vorhanden	D
Seg.2 -a	<ul style="list-style-type: none"> RecipientLength 	R	2 Byte	UINT16	Länge des nachfolgenden Adressaten des Meldungstextes in Zeichen (Byte) = „00hex“ keine Angabe	D
-b	<ul style="list-style-type: none"> Recipient 	R	Beliebig	Visible-String	Empfänger der Meldung	D
Seg.N -a	<ul style="list-style-type: none"> Senderlength 	R	2 Byte	UINT16	Länge des nachfolgenden Absenders des Meldungstextes in Zeichen (Byte) = „00hex“ keine Angabe	D
-b	<ul style="list-style-type: none"> Sender 	R	Beliebig	Visible-String	Absender der Meldung	D
Seg.M -a	<ul style="list-style-type: none"> Textlength 	R	2 Byte	UINT16	Länge des nachfolgenden Meldungstextes in Zeichen (Byte)	D
-b	<ul style="list-style-type: none"> Text – 0018.6 	R	Beliebig	Visible-String	Meldung was aufgetreten ist. Default: „Status OK“	D
E801	DiagHistory	R		Domain Variable Record	Diagnoseinformationen des Gerätes in der Kurzform mit Historie (die älteste Information / niedrigste Fehlernummer wird zuerst übertragen) Die Struktur ist die selbe wie die von DiagState	O
	<ul style="list-style-type: none"> DiagStateM 				Ältester Diagnosestatus in der Kurzform	D
	<ul style="list-style-type: none"> DiagStateM-1 					D
	D
	<ul style="list-style-type: none"> DiagStateN+1 					D
	<ul style="list-style-type: none"> DiagStateN 				Neuster Diagnosestatus in der Kurzform	D
E802	DiagHistoryLong	R		Domain Variable Record	Diagnoseinformationen des Gerätes in der Langform mit Historie (die älteste Information / niedrigste Fehlernummer wird zuerst übertragen) Die Struktur ist die selbe wie die von DiagStateLong	O
	<ul style="list-style-type: none"> DiagStateLongM 				Ältester Diagnosestatus in der Langform	D
	<ul style="list-style-type: none"> DiagStateLongM-1 					D
	...					D
	<ul style="list-style-type: none"> DiagStateLongN+1 					D

	• DiagStateLongN				Neuster Diagnosestatus in der Langform	D
0019	ResetDiag	W	1 Byte	UINT8	<p>Löscht den entsprechenden Diagnosespeicher des Gerätes und quittiert die Diagnosemeldung(en)</p> <p>= „00_{hex}“ keine Beschränkung bei den Diagnosemeldungen</p> <p>= „01_{hex}“ Löscht die Diagnosehistorie. Dieser Wert wird nur unterstützt, wenn das Objekt „DiagHistory(Long)“ existiert.</p> <p>= „02_{hex}“ Löscht (und quittiert) alle nicht ausgelesenen und die noch anstehenden DiagStates (Fehler).</p> <p>= „03_{hex}“ Löscht (und quittiert) die gesamte Diagnose</p> <p>= „04_{hex}“ Löscht (und quittiert) alle schon per Information Report gemeldeten DiagStates</p> <p>= „05_{hex}“ Löscht (und quittiert) den letzten gemeldeten DiagState</p> <p>= „06_{hex}“ Löscht (und quittiert) die gesamte Diagnose (wie „03_{hex}“) und lässt auch keine neuen Diagnosemeldungen zu.</p> <p>= +„80_{hex}“ Quittiert ohne zu löschen</p> <p>sonst: reserviert.</p>	O

001A	GetErrorRepMethod	R/W	1 Byte	Bit-String 8	<p>Legt die Methode fest mit der eine Störung an das überlagerte System gemeldet werden soll.</p> <p>= „01_{hex}“ schaltet das Erzeugen eines Information Reports bei aufgetretener Störung mit dem Inhalt des Objektes „DiagState“ ein (=“1“) /aus (=“0“).</p> <p>= „02_{hex}“ schaltet das Erzeugen einer Peripherie Fehler Statusmeldung bei aufgetretener Störung mit dem Inhalt des Objektes „DiagState“ ein (=“1“) /aus (=“0“).</p> <p>= „04_{hex}“ schaltet das Erzeugen einer Prozeßdaten Statusmeldung (vom Hersteller beliebig zu wählendes Prozeßdatenbit) bei aufgetretener Störung mit dem Inhalt des Objektes „DiagState“ ein (=“1“) /aus (=“0“).</p> <p>= „09_{hex}“ schaltet das Erzeugen eines Information Reports bei aufgetretener Störung mit dem Inhalt des Objektes „DiagState“ ein (=“1“) /aus (=“0“). Und löscht den entsprechenden DiagState ohne explizite Quittierung.</p> <p>Werden mehrere Meldemethoden gleichzeitig gewählt, muß die Diagnosemeldung (und die folgenden Diagnosemeldungen) im Objekt 0018 solange vorgehalten werden, bis sie ausgelesen wurde(n), selbst wenn ein oder mehrere andere Diagnosemeldungen bereits über den Reportmechanismus gesendet wurden.</p>	O
0033	DiagStateChannelNo	R	6 Einträge	Record	aktueller Diagnosestatus des Gerätes	D
.1	• Lfd.Nr.	R	2 Byte	UINT16	Eindeutige, fortgezählte Fehlernummer seit dem letzten Power-Up-Reset oder HistoryReset	D

.2	• ChannelNo	R	1 Byte	UINT8	Hat das Gerät zu Gruppen zusammengefasste Kanäle, kann hier der betroffene Kanal der entsprechenden Gruppe eingetragen werden. Wird „DiagState“ gelesen, so ist dieses Objekt wenigstens für den nächsten Lesezugriff konsistent zu „DiagState“ zu halten. Wird „DiagState“ als Information Report gemeldet und ist dieses Objekt dazu relevant, wird es ebenfalls über einen Information Report gemeldet. Zeitl. Nach dem DiagState.	D
0034	DiagStateAddValue	R	6 Einträge	Record	aktueller Diagnosestatus des Gerätes	D
.1	• Lfd.Nr.	R	2 Byte	UINT16	Eindeutige, fortgezählte Fehlernummer seit dem letzten Power-Up-Reset oder HistoryReset	D
.2	• AddValue	R	4 Byte	UINT32	Hier kann ein Begleitwertwert zu einer Kanaldiagnose wie z.B: „aktuelle Temperatur bei Grenzwertüberschreitung“ eingetragen werden. Details: siehe Profinet I&M Profil Wird „DiagState“ gepollt, so ist dieses Objekt wenigstens für den nächsten Lesezugriff konsistent zu „DiagState“ zu halten. Wird „DiagState“ als Information Report gemeldet und ist dieses Objekt dazu relevant, wird es ebenfalls über einen Information Report gemeldet. Zeitl. nach dem DiagState.	D
001B	TestMode	R/W	2 Byte	UINT16	Mit einem herstellerspezifischen Code wird in einen Testmode geschaltet. Hierdurch werden andere Testparameter aktiviert. Mit dem Wert 0 wird wieder in den normalen Betriebsmode zurückgeschaltet = „0000 _{hex} “ normaler Betriebsmode (Default)	O

Die Tiefe der Diagnose Historie, d.h. die Anzahl der Einträge in die Diagnose-Historie, ist vom Geräteentwickler entsprechend seinen Anforderungen und Ressourcen frei zu wählen.

7.2.1.1 Benachrichtigung über Auslesen (Read) des DiagState Objektes

Eine vorhandene Diagnoseinformation wird (z.B. per Statusbit [z.B. Bit7 im Byte 0 des Prozessdatenkanals] im Prozessdatenkanal bzw. über den Modulfehlereingang /StatErr der Feldbus Slave Protokollchips) je nach Priorität, gemeldet. Es können auch mehrere Diagnoseinformationen gleichzeitig vorhanden sein. Der Master oder ein Software- Tool kann sich die entsprechenden Informationen über den Parameterkanal abholen.

Die aktuelle Diagnoseinformation in DiagState(Long) bleibt so lange bestehen bis,

- diese Information wenigstens einmal ausgelesen wurde
und
- die Ursache für diese Diagnoseinformation nicht mehr vorhanden ist.

Erst danach wird die nächste Information, (z.B. „Störung beseitigt“ oder „Störung Nr.2“) zur Verfügung gestellt.

Das Statusbit(s) im Prozessdatenkanal oder der Modulfehlereingang bleibt so lange anstehen bis,

- jede Diagnoseinformation wenigstens einmal ausgelesen wurde
und
- keine Diagnoseinformation mehr vorhanden ist.

Erst danach wird die dann aktuelle Information, „Status OK“ zur Verfügung gestellt.

7.2.1.2 Benachrichtigung über Melden (Information Report) des DiagState Objektes

Alternativ kann das Melden von Störungen durch den Dienst „Information- Report“ (einmalig, nur das Objekt „DiagState“) dem Master gemeldet. Diese Funktionalität wird mit dem Objekt „GetErrorRepMethod“ ein- / ausgeschaltet. Bei aktivierter Funktion "GetErrorRepMethod" wird jede neue Diagnoseinformation (Inhalt: Objekt 0018) einmalig und selbsttätig (ohne Request des Masters) dem Master gemeldet.

In diesem Fall muss auf der Feldbus- Masterseite die Auswertung des Information- Reports z.B. als SPS-Funktionsbaustein oder als Hochsprachenprogramm vorhanden sein.

Die normale Diagnosebehandlung von Objekt 0018 "DiagState" (s.o) wird dadurch nicht beeinflusst (siehe Beschreibung Objekt DiagState). D.h. hier steht immer noch die erste (und folgende) Diagnoseinformation(en) an, obwohl diese durch den "Information-Report" gemeldet wurden.

Die Diagnoseinformation in DiagState(Long) ist hier stets, die erste noch anstehende Störung.

Das Statusbit(s) im Prozessdatenkanal oder der Modulfehlereingang bleibt so lange anstehen bis,

- keine Diagnoseinformation mehr vorhanden ist.

Erst danach wird die dann aktuelle Information, „Status OK“ gemeldet.

7.2.1.3 Klassifizierung von Meldungen

Prio 3 (grün) Nachricht, Information, Mitteilung
Beispiel: Allgemeine Betriebsmeldung 10.000 Betriebsstunden gelaufen

Prio 2 (gelb) Warnung
Bedeutung: Gefahr einer Störung
Beispiel: Unter- oder Überschreitung eines Grenzwertes
Eine Warnung führt nicht zu einer Aktivität im Gerät

Prio 1 (rot) Störung (Alarm)
Bedeutung: Es liegt eine Störung vor, auf die reagiert werden muss.
Eine Störung führt zu einer Aktivität im Antrieb, aber nicht notwendigerweise zum sofortigen Stillsetzen.

Prio 81, 82, 83 Aufgehoben

Bedeutung: Die mit der selben Nr. gemeldete Störung ist beseitigt.

Nachrichten und Informationen sind benutzerdefiniert. Warnungen und Störungen können sowohl vordefiniert als auch benutzerdefiniert sein.

Eine gehende Meldung (DiagnoseStatus) überschreibt die zugehörigen kommenden Meldung (DiagnoseStatus), falls diese noch nicht vom Master abgeholt wurde. Damit wird das Kommunikationsaufkommen reduziert, ohne daß Informationen verloren gehen. Daraus ergibt sich folgende Reihenfolge der Prioritäten:

Codierung	Priorität	Bedeutung
0x81 – höchste Priorität	Prio 1 (rot) gehend	Störung (Alarm) aufgehoben
0x01	Prio 1 (rot) kommend	Störung (Alarm)
0x82	Prio 2 (gelb) gehend	Warnung aufgehoben
0x02	Prio 2 (gelb) kommend	Warnung
0x83	Prio 3 (grün) gehend	Information, Mitteilung aufgehoben
0x03 – niedrigste Priorität	Prio 3 (grün) kommend	Information, Mitteilung

7.2.1.4 Störungscode

Ziel der Vereinheitlichung der Störungscode ist es, dem Anwender auch ohne genaue Kenntnis der Gerätes, eine schnelle und einfache Orientierung für die Problembeseitigung zu geben.

Der Störungscode wird als ein Octet-String von der Länge 2 Byte dargestellt. Die Codierung erfolgt hierarchisch von einer groben zu einer immer feiner werdenden Unterscheidung.

Bit	Gruppierung
15 ... 12	Hauptgruppen
11 ... 8	Untergruppen
7 ... 0	Details

Ist das Gerät im Zustand Störung, so enthält der Parameter einen Wert ungleich 0.
Befindet sich das Gerät nicht im Zustand Störung, dann enthält dieser den Wert 0.

7.2.1.4.1 Haupt- und Untergruppen

Code (hex) Bedeutung

0000	keine Störung
1000	Störung allgemein
2000	Strom
2100	Strom geräteeingangseitig
2200	Strom geräteintern
2300	Strom geräteausgangseitig
3000	Spannung
3100	Netzspannung
3200	Spannung geräteintern
3300	Ausgangsspannung
3400	Versorgungsspannung für die Peripherie
4000	Temperatur
4100	Temperatur Umgebung
4200	Temperatur Gerät
4300	Temperatur extern (z.B. Antrieb)
4400	Temperatur Versorgung
5000	Geräte-Hardware (nur innerhalb des Gerätegehäuses)
5100	Versorgung im und durch das Gerät
5200	Steuerung (Geräteapplikation)
5300	Bedien u. Anzeigeeinheit
5400	Leistungsteil
5500	Kommunikation zur Zusatzbaugruppe
6000	Geräte-Software
6100	Interne Software (Firmware)
6200	Anwender-Software
6300	Datensatz nicht OK
7000	Zusatzbaugruppe(n) (fest mit dem Gerät verbunden) defekt
7100	Leistung
7200	Messschaltung
7300	Sensor (am Gerät)
7400	Rechenschaltung
7500	Kommunikation
7600	Datenspeicher
7700	Drahtbruch / Kabelfehler
8000	Überwachung
8100	Kommunikation
8200	Regelung
8300	Momentenregler
8400	Drehzahlregler
8500	Lageregler
8600	Positionierregler
8700	Gleichlaufregler
8800	Wickelregler
8900	Sensoren (eigenes Gerät) Meßfehler
8A00	Aktoren
8B00	vorbeugende Wartung erforderlich (Condition Monitoring)
9000	Externe Störung
A000	Modulare Geräte
F000	Zusatzfunktionen

7.2.1.4.2 Hauptgruppen mit Untergruppen und Details

Code_(hex) Bedeutung

0000	keine Störung
1000	Störung allgemein
1800	Störung allgemein (herstellerspezifisch)
...	Störung allgemein (herstellerspezifisch)
1FFF	Störung allgemein (herstellerspezifisch)
2000	Strom

2100	Strom geräteeingangseitig
2110	Kurz-/Erdschluß
2120	Erdschluß
2121	Erdschluß Phase L1
2122	Erdschluß Phase L2
2123	Erdschluß Phase L3
2130	Kurzschluß
2131	Kurzschluß Phasen L1-L2
2132	Kurzschluß Phasen L2-L3
2133	Kurzschluß Phasen L3-L1
2136	Kurzschluß nach VCC
2137	Kurzschluß nach GND
2200	Strom geräteintern
2211	Strom geräteintern Nr. 1
2212	Strom geräteintern Nr. 2
2213	Überstrom im Hochlauf
2214	Überstrom im Ablauf
2220	Dauerüberstrom
2221	Dauerüberstrom Nr. 1
2222	Dauerüberstrom Nr. 2
2230	Kurz-/Erdschluß
2240	Erdschluß
2250	Kurzschluß
2300	Strom geräteausgangseitig
2310	Dauerüberstrom
2311	Dauerüberstrom Nr. 1
2312	Dauerüberstrom Nr. 2
2320	Kurz-/Erdschluß
2330	Erdschluß
2331	Erdschluß Phase U
2332	Erdschluß Phase V
2333	Erdschluß Phase W
2340	Kurzschluß
2341	Kurzschluß Phasen U-V
2342	Kurzschluß Phasen V-W
2343	Kurzschluß Phasen W-U
2344	Überlast Ausgang
2345	Überlast Initiatorversorgung
2346	Kurzschluß nach VCC
2347	Kurzschluß nach GND
2350	offene Leitungen
2360	Leitungsunterbrechung
3000	Spannung
3100	Netzspannung
3110	Netzüberspannung
3111	Netzüberspannung Phase L1
3112	Netzüberspannung Phase L2
3113	Netzüberspannung Phase L3
3120	Netzunterspannung
3121	Netzunterspannung Phase L1
3122	Netzunterspannung Phase L2
3123	Netzunterspannung Phase L3
3130	Phasenausfall
3131	Phasenausfall L1
3132	Phasenausfall L2
3133	Phasenausfall L3
3134	Phasenfolge
3140	Netzfrequenz
3141	Netzfrequenz zu groß
3142	Netzfrequenz zu klein
3200	Spannung geräteintern
3210	Überspannung geräteintern
3211	Überspannung Nr. 1
3212	Überspannung Nr. 2
3220	Unterspannung geräteintern
3221	Unterspannung Nr. 1
3222	Unterspannung Nr. 2
3230	Aufladefehler
3300	Ausgangsspannung

3310	Ausgangsüberspannung
3311	Ausgangsüberspannung Phase U
3312	Ausgangsüberspannung Phase V
3313	Ausgangsüberspannung Phase W
3320	Ausgangsunterspannung
3320	Ankerkreis
3321	Ankerkreis unterbrochen
3330	Feldkreis
3331	Feldkreis unterbrochen
3400	Versorgungsspannung für die Peripherie
3410	Initiatorversorgung
3411	Initiatorversorgung - Unterspannung
3412	Initiatorversorgung fehlt
3413	Initiatorversorgung - Überspannung
3420	Aktorversorgung
3421	Aktorversorgung - Unterspannung
3422	Aktorversorgung fehlt
3423	Aktorversorgung - Überspannung
4000	Temperatur
4100	Temperatur Umgebung
4110	Übertemperatur Umgebung
4120	Untertemperatur Umgebung
4130	Temperatur Zuluft
4140	Temperatur Abluft
4200	Temperatur Gerät
4210	Übertemperatur Gerät
4220	Untertemperatur Gerät
4300	Temperatur extern (z.B. Antrieb)
4310	Übertemperatur Antrieb
4320	Untertemperatur Antrieb
4400	Temperatur Versorgung
4410	Übertemperatur Versorgung
4420	Untertemperatur Versorgung
5000	Geräte-Hardware (nur innerhalb des Gerätegehäuses)
5010	Bauteilefehler
5100	Versorgung im und durch das Gerät
5110	Versorgung Niederspannung allgemein
5111	Versorgung +/- 15 V
5112	Versorgung + 24 V
5113	Versorgung + 5 V
5114	Versorgung + 3,3 V
5115	Versorgung + 2,5 V
5116	Versorgung + 1,2 V
5118	U8 = herstellerspezifisch
.... für U8 bis U15 herstellerspezifisch
511F	U15 = herstellerspezifisch
5120	Versorgung Luft
5130	Versorgung Lack
5140	Versorgung Zwischenkreis
5150	Versorgung Initiator
5151	Kurzschluß
5160	Versorgung der Peripherie, die durch das Gerät versorgt wird.
5200	Steuerung (Geräteapplikation)
5210	Meßschaltungen
5220	Rechenschaltungen
5230	Kommunikation (geräteintern)
5300	Bedien u. Anzeigeeinheit
5400	Leistungsteil
5410	Endstufen
5420	Chopper
5430	Eingangsstufen
5440	Schuetze
5441	Kanal 1
...	...
5448	Kanal 8
5450	Sicherungen
5451	S1 = L1
5452	S2 = L2
5453	S3 = L3

5454	S4 = herstellerspezifisch
.... für S5, S6, S7, S8
5459	S9 = herstellerspezifisch
5500	Kommunikation zur Zusatzbaugruppe
5510	Schnittstelle Nr. 1
5520	Schnittstelle Nr. 2
6000	Geräte-Software
6010	Software-Reset (Watchdog)
6100	Interne Software (Firmware)
6200	Anwender-Software
6210	PD-Index nicht vorhanden
6211	Variablen-Nr. nicht vorhanden
6300	Datensatz nicht OK
6301	Datensatz Nr. 1
..	von 2 bis 14 entsprechend
630F	Datensatz Nr. 15
6310	Parameterverlust
6320	Parameterfehler
6330	Parameter noch nicht initialisiert
6800	Konfiguration
6810	Daten Konfiguration doppelt gemapped
7000	Zusatzbaugruppe(n) (fest mit dem Gerät verbunden) defekt
7100	Leistung
7110	Bremschopper
7111	Ausfall Bremschopper
7112	Überstrom Bremschopper
7113	Beschaltung Bremschopper
7120	Motor
7121	Motor blockiert
7122	Motor fehlt oder Kommutierung gestört
7123	Motor gekippt
7200	Messschaltung
7300	Sensor (am Gerät)
7301	Sensor 1 defekt
730F	Sensor 15 defekt
7301	Tacho defekt
7302	Tacho verpolt
7303	Resolver 1 defekt
7304	Resolver 2 defekt
7305	Inkrementalgeber 1 defekt
7306	Inkrementalgeber 2 defekt
7307	Inkrementalgeber 3 defekt
7310	Geschwindigkeit
7320	Lage
7400	Rechenschaltung
7500	Kommunikation
7510	Serielle Schnittstelle Nr. 1
7520	Serielle Schnittstelle Nr. 2
7600	Datenspeicher
7610	RAM
7620	EPROM
7630	EEPROM
7700	Drahtbruch / Kabelfehler
7701	Kabel 1 defekt
770F	Kabel 15 defekt
7710	Drahtbruch Kabel zum Sensor
8000	Überwachung
8100	Kommunikation
8110	Prozeßdaten- Überwachung
8120	Host-Überwachung
8121	iPD-Kanal Handshake Timeout
8200	Regelung
8210	Regelabweichung Soll > Ist, Die Abweichung steht länger als eine bestimmte Zeit an (herstellerspezifisch)
8211	max. Stellgröße erreicht
8220	Regelabweichung Soll < Ist, Die Abweichung steht länger als eine bestimmte Zeit an (herstellerspezifisch)
8221	min. Stellgröße erreicht
...	reserviert für profilspezifische Regelfehler

827f	reserviert für profilspezifische Regelfehler
8280	Herstellerspezifische Regelfehler
...	herstellerspezifische Regelfehler
82FF	herstellerspezifische Regelfehler
8300	Momentenregler
8311	Übermoment
8312	Schweranlauf
8313	Stillstandsmoment
8321	Untermoment
8331	Momentenbruch
8400	Drehzahlregler
8500	Lageregler
8600	Positionierregler
8611	Schleppfehler
8612	Referenzgrenze
8700	Gleichlaufregler
8800	Wickelregler
8900	Sensoren (eigenes Gerät) Meßfehler
8910	Messbereichsüberschreitung
8920	Messbereichsunterschreitung
8A00	Aktoren
8B00	vorbeugende Wartung erforderlich (Condition Monitoring)
9000	Externe Störung
A000	Modulare Geräte
A001	Kein Modul vorhanden
A002	Falsches Modul vorhanden
A003	Ersetzt (Das erwartete Modul wurde durch ein kompatibles Modul ausgetauscht.)
A004	Subbus mehr Geräte als erwartet
A010	Geräte Fehler
A012	Applikation auf dem Modul nicht bereit
A013	Subbus Geräte Reset
A020	Subbus Kommunikations- Fehler
A021	Subbus Fehler - Timeout
A022	Multiple Übertragungsfehler im Subbus
A023	Subbus I/O Daten Kommunikation Fehler
A024	Subbus Management Daten Kommunikation Fehler
A030	Subbus Konfigurations- Fehler
A031	E/A Daten Konfiguration doppelt gemapped
A040	Common Errors
A041	Hardware Fehler
A042	Firmware Fehler
A043	Subbus asynchron zum überlagerten System
B0XX	reserviert
F000	Zusatzfunktionen
F001	Verzögerung
F002	Untersynchronlauf
F003	Hubwerk
F004	Steuerung

Nicht aufgeführte Codes sind reserviert.

ANMERKUNG: Weitere Störungs_codes sind in den Profilen definiert.

Sollte sich ein Hersteller in dieses Schema der Störungs_codes nicht einordnen können, wird er gebeten sich mit dem INTERBUS Club in Verbindung zu setzen.

7.2.2 Trace- Daten

Traces sind fortlaufende Aufnahmen von bestimmten Daten eines Gerätes. Sie dienen u.a. dazu, Abläufe zu rekapitulieren um Fehler oder Probleme zu finden oder Daten archivieren zu können.

Grundsätzlich kann es sehr unterschiedliche Traces geben. Inhalt, Struktur, Umfang und Handhabung werden vom Gerätehersteller festgelegt.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
001C	ControlTrace	R/W	1 Byte	UINT8	Steuern eines Aufnahme- Vorganges. „00 _{hex} “ Reset der Aufnahme und der dazugehörigen Parameter „01 _{hex} “ Start einer Aufnahme „02 _{hex} “ Stop einer Aufnahme Weitere Details werden vom Hersteller festgelegt	O
E803	TraceBuffer	R		Domain Variable	Beinhaltet die aufgenommenen Trace- Daten. Die Struktur wird vom Hersteller festgelegt	O

Die Trace- Daten werden generell mit dem Upload Mechanismus übertragen. Beim Upload der Trace- Daten sollten die grundlegenden Informationen im Header enthalten sein.

Treten beim Schreiben / Lesen der Objekte Fehler auf, z.B. es werden unzulässige Werte übertragen, so sind die Zugriffe mit den entsprechenden Fehlercodes (siehe: Kapitel „Liste der zulässigen Fehlercodes“) abzulehnen.

7.3 User Daten Management

7.3.1 Prozeßdaten Management

Dieser Abschnitt definiert das Verhalten der Prozessdaten. Das Verhalten bei Timeouts und Feldbus Reset wird definiert wie auch das Lesen und Schreiben der Prozessdaten über den Parameterkanal.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
001F	PDTimeout	R/W	2 Byte	UINT16	Prozessdaten- Überwachungszeit Max. zulässige Zeit in ms (z.B. 100ms) bis zu der neue Daten über den Prozessdatenkanal übertragen worden sein müssen. Die dann ausgelöste Aktion wird im Objekt 0x0020 „PDTimeoutCode“ festgelegt. FFFFh = ausgeschaltet	O
0020	PDTimeoutCode	R/W	N x 2Byte	Array of UINT16	Bei Ablaufen der Prozessdaten- Überwachungszeit wird die dem „Prozessdaten- Überwachungs-Auswahlcode“ zugeordnete Funktion durch den Teilnehmer ausgeführt. Eine solche Funktion kann z.B. das definierte Herunterfahren eines Antriebs sein oder das Kontinuierliche Senden von Fehlertelegrammen über eine serielle Schnittstelle. 0000h Ausgabe von „0“ auf allen Ausgangsbits 0001h Ausgabe von „1“ auf allen Ausgangsbits 0002h Halten des letzten gültigen Wertes 0003h Übernahme des Ersatzwertes aus Objekt 002F „PDOOUT_Subst“ 0004h bis 7FFFh reserviert 8000h bis FFFFh herstellerspezifisch Da das Verhalten ggf. für jeden Kanal / jede Gruppe unterschiedlich gewünscht sein kann, ist dieser Parameter als Array angelegt. Die Anzahl der Elemente des Arrays ist durch die Anzahl der Elemente des jeweiligen PDOOUT (0x0026) (PDIN 0x0025) Objektes definiert. N= Anzahl der Elemente Wird nur ein Eintrag unterstützt (geschrieben) obwohl das Gerät mehrere Elemente (Kanäle / Gruppen) besitzt, so gilt der Parameter für das gesamte Gerät.	O

0024	ResetCode	RW	N x 2Byte	Array of UINT16	<p>Fieldbus Reset Code Bei Auftreten eines Bus Resets wird die dem „Bus Reset- Auswahlcode“ zugeordnete Funktion durch den Teilnehmer ausgeführt.</p> <p>Eine solche Funktion kann z.B. das definierte Herunterfahren eines Antriebs sein oder das Kontinuierliche Senden von Fehlertelegrammen über eine serielle Schnittstelle.</p> <p>0000h Ausgabe von „0“ auf allen Ausgangsbits 0001h Ausgabe von „1“ auf allen Ausgangsbits 0002h Halten des letzten gültigen Wertes 0003h Übernahme des Ersatzwertes aus Objekt 002F „PDOOUT_Subst“</p> <p>0004h bis 7FFFh reserviert 8000h bis FFFFh herstellerspezifisch</p> <p>Da das Verhalten ggf. für jeden Kanal / jede Gruppe unterschiedlich gewünscht sein kann, ist dieser Parameter als Array angelegt. Die Anzahl der Elemente des Arrays ist durch die Anzahl der Elemente des jeweiligen PDOOUT (0x0026) (PDIN 0x0025) Objektes definiert. N= Anzahl der Elemente Wird nur ein Eintrag unterstützt (geschrieben) obwohl das Gerät mehrere Elemente (Kanäle / Gruppen) besitzt, so gilt der Parameter für das gesamte Gerät.</p>	O
0030	PF_Code	RW	N x 2Byte	Array of UINT16	<p>Peripheral Fail Code Bei Auftreten eines Fehlers an der Peripherie des Feldbus Protokollchips, der die Bearbeitung der Prozesseingangsdaten nicht mehr zulässt, wird die dem „Peripherie Fehler- Auswahlcode“ zugeordnete Funktion durch den Teilnehmer ausgeführt.</p> <p>Eine solche Funktion kann z.B. über den Eingang /StatErr oder µP WD an den Protokollchips angezeigt werden.</p> <p>0000h Ausgabe von „0“ auf allen Eingangsbits 0001h Ausgabe von „1“ auf allen Eingangsbits 0002h Halten des letzten gültigen Wertes 0003h Übernahme des Ersatzwertes aus Objekt 0031 „PDIN_Subst“</p> <p>0004h bis 7FFFh reserviert 8000h bis FFFFh herstellerspezifisch</p> <p>Da das Verhalten ggf. für jeden Kanal / jede Gruppe unterschiedlich gewünscht sein kann, ist dieser Parameter als Array angelegt. Die Anzahl der Elemente des Arrays ist durch die Anzahl der Elemente des jeweiligen PDOOUT (0x0026) (PDIN 0x0025) Objektes definiert. N= Anzahl der Elemente Wird nur ein Eintrag unterstützt (geschrieben) obwohl das Gerät mehrere Elemente (Kanäle / Gruppen) besitzt, so gilt der Parameter für das gesamte Gerät.</p>	O

0025	PDIN	R	PD Länge	Record	<p>Process Input Data Abbildung der Prozesseingangsdaten (vom Teilnehmer zum Master) auf ein Objekt Die Zählung der PD-Bytes wie auch der Bytes im Octet String erfolgt mit 0 beginnend von links (oben) aufsteigend.</p> <p>Sind die Prozessdaten strukturiert (z.B. mehrere Kanäle), sollte dieses Objekt ebenfalls strukturiert sein und der Zugriff auf die einzelnen Strukturelemente erfolgt über den Subindex.</p>	M
.1	• Part 1	R		dependent	erster Teil des Prozeßdatums	O
.2	• Part 2	R		dependent	zweiter Teil des Prozeßdatums	O
...	•					
.N	• Part N	R		dependent	N-ter Teil des Prozeßdatums	O
0026	PDOOUT	R/W _D	PD Länge	Record	<p>Process Output Data Abbildung der Prozessausgangsdaten (vom Master zum Teilnehmer) auf ein Objekt Die Zählung der PD-Bytes wie auch der Bytes im Octet String erfolgt mit 0 beginnend von links (oben) aufsteigend.</p> <p>Sind die Prozessdaten strukturiert (z.B. mehrere Kanäle), sollte dieses Objekt ebenfalls strukturiert sein und der Zugriff auf die einzelnen Strukturelemente erfolgt über den Subindex.</p> <p>Ist das Objekt „GetExRight“ nicht implementiert, hat das Objekt PDOOUT den Status „Nur Lesen“. Ist das Objekt „GetExRight“ implementiert, hat das Objekt PDOOUT den Status „Lesen und Schreiben“. Hat das Objekt „GetExRight“ einen Wert = „0“, wird ein schreibender Zugriff auf das Objekt PDOOUT mit</p> <p style="margin-left: 20px;">Error-Class „8“ - Other Error-Code „1“ – Profile specific Additional-Code 0022_{hex}</p> <p>„Dienst kann im momentanen Gerätezustand nicht ausgeführt werden“ abgewiesen.</p>	M
.1	• Part 1	R		dependent	erster Teil des Prozeßdatums	O
.2	• Part 2	R		dependent	zweiter Teil des Prozeßdatums	O
...	•					
.N	• Part N	R		dependent	N-ter Teil des Prozeßdatums	O
002F	PDOOUT_Subst	R/W	PD Länge	Octet String	<p>Process Output Data Substitute Ersatzwert für die Prozessausgangsdaten (vom Master zum Teilnehmer) im Fehlerfall. Die Zählung der PD-Bytes wie auch der Bytes im Octet String erfolgt mit 0 beginnend von links (oben) aufsteigend. Muß implementiert werden, wenn für die Objekte 0020 „PDTIMEOUTCODE“ oder 0024 „RESETCODE“ der Wert 0003h zulässig ist. Siehe dort</p>	D

.1	• Type	R	8 Byte	Visible-String	<p>Type des E/A Datums. Definiert sind derzeit:</p> <ul style="list-style-type: none"> • „ACK“ - Antwort auf ein Kommando • „STATUS“ - Statusinformationen • „DIAG“ - Diagnoseinformationen • „DI“ - digital In • „AI“ - analog In • „CNT“ - Counter • „CmpP“ - Complex Protokoll • „SubMa“ - SubBus Master • „SubSl“ - SubBus Slave • „SI“ - Safety Input • „PLC“ - PLC (Daten eines intelligentes Moduls) • „MotSt“ - Motor Starter • „PWM“ - PWM Daten • „POS“ - Positionier Daten • „NO“ - Number • „NC“ - not connected, nicht benutzt • „DO_F“ - DO Feedback • „AO_F“ - AO Feedback • „SO_F“ - SO Feedback <p>Der Type ist genau 8 Zeichen lang. Nicht benutzte Zeichen sind mit 0x00 aufzufüllen.</p>	D
.2	• ChNo	R	2 Byte	UINT16	<p>Channel Number Anzahl der Kanäle eines Typs</p> <p><u>Anmerkung:</u> Ein Kanal ist immer eine Anzahl (auch 1) von Bits, die zusammen eine Bedeutung haben. Beispiel: Vier analoge Eingänge sind 4 Kanäle oder 16 digital Eingänge sind 16 Kanäle</p>	D
.3	• ChLength	R	2 Byte	UINT16	<p>Channel Length Länge eines Kanals</p>	D
003C	PDOOUT_Descr	R	N x 3 Einträge	Array of Records	<p>Process Data Description Beschreibung des Aufbaus der Prozeßdaten Die Beschreibung muß lückenlos aufsteigend, von links beginnend erfolgen.</p> <p>Die Anzahl der Elemente des Arrays ist durch die Anzahl der Elemente des PDOOUT (0x0026) Objektes definiert. N= Anzahl der Elemente PDOOUT (0x0026)</p>	O

.1	• Type	R	8 Byte	Visible-String	Type des E/A Datums. Definiert sind derzeit: <ul style="list-style-type: none"> • „CMD“ - Kommandos • „CTRL“ - Steuerung / Konfiguration • „DO“ - digital Out • „AO“ - analog Out • „CNT“ - Counter • „CmpP“ - Complex Protokoll • „SubMa“ - SubBus Master • „SubSl“ - SubBus Slave • „SO“ - Safety Output • „PLC“ - PLC (Daten eines intelligentes Moduls) • „MotSt“ - Motor Starter • „PWM“ - PWM Daten • „POS“ - Positionier Daten • „NO“ - Number • „NC“ - not connected, nicht benutzt <p>Der Type ist genau 8 Zeichen lang. Nicht benutzte Zeichen sind mit 0x00 aufzufüllen.</p>	D
.2	• ChNo	R	2 Byte	UINT16	Channel Number Anzahl der Kanäle eines Typs	D
.3	• ChLength	R	2 Byte	UINT16	Channel Length Länge eines Kanals	D

Treten beim Schreiben / Lesen der Objekte Fehler auf, z.B. es werden unzulässige Werte übertragen, so sind die Zugriffe mit den entsprechenden Fehlercodes (siehe: Kapitel „Liste der zulässigen Fehlercodes“) abzulehnen.

Hinweis Power On Verhalten der Ausgänge:

Ausgänge müssen nach Power-On einen im Sinne der Maschinenrichtlinie sicheren (definierten) Zustand einnehmen (inaktiv, typ. „0“). Dieser Zustand ist so lange beizubehalten bis überprüft wurde, dass die ggf. im Modul hinterlegte Parametrierung des Ersatzwertverhaltens zur Anlagenkonfiguration (Einbauort) passt oder neue gültige Werte übertragen wurden.

7.3.2 Parameterkanal Management

Dieser Abschnitt definiert das Verhalten des Parameterkanals bei Timeouts und Kommunikationsabbruch.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
0021	PChTimeout	R/W	2 Byte	UINT16	Kommunikations- Überwachungszeit Max. zulässige Zeit in ms (z.B. 1000ms) bis zu der neue Daten über den Parameterkanal übertragen worden sein müssen. Die dann ausgelöste Aktion wird im Objekt 0x0022 „PChTimeoutCode“ festgelegt. FFFFh = ausgeschaltet	O
0022	PChTimeoutCode	R/W	2 Byte	UINT16	Bei Ablauf der Kommunikations- Überwachungszeit wird die dem Kommunikations- Überwachungs- auswahlcode zugeordnete Funktion ausgeführt. Eine solche Funktion kann z.B. das definierte Herunterfahren eines Antriebs sein oder das Kontinuierliche Senden von Fehlertelegrammen über eine serielle Schnittstelle. 0 keine Aktion 0001h bis 7FFFh reserviert 8000h bis FFFFh herstellerspezifisch	O
0023	AbortCode	R/W	2 Byte	UINT16	Bei Auftreten eines Verbindungsabbruches wird die dem Verbindungsabbau-Auswahlcode zugeordnete Funktion ausgeführt. Eine solche Funktion kann z.B. das definierte Herunterfahren eines Antriebs sein oder das Kontinuierliche Senden von Fehlertelegrammen über eine serielle Schnittstelle. 0 keine Aktion 0001h bis 7FFFh reserviert 8000h bis FFFFh herstellerspezifisch	O

Treten beim Schreiben / Lesen der Objekte Fehler auf, z.B. es werden unzulässige Werte übertragen, so sind die Zugriffe mit den entsprechenden Fehlercodes (siehe: Kapitel „Liste der zulässigen Fehlercodes“) abzulehnen.

7.4 Geräte Management

Feldgeräte erlangen immer komplexere Funktionalitäten, die durch die Anwendung oder Software- Tools geeignet parametrierbar werden müssen. Im folgenden wird ein Grundgerüst festgelegt, das es erlaubt, die dabei üblichen grundlegenden Handlungsabläufe zu vereinheitlichen.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
0029	ParamSetWriteControl	R/W	1 Byte	UINT8	Steuerung der Blockparametrierung s.u.	O
002A	ConflictDictionary	R	2*N Einträge	Record	Ergebnis der Blockparametrierung s.u.	O
.1	• Index1	R	2 Byte	UINT16	Index des 1. abhängigen Parameters	D
.2	• Add. Code1	R	2 Byte	UINT16	Add. Info zum 1. abhängigen Parameter	D
.3	• Index2	R	2 Byte	UINT16	Index des 2. abhängigen Parameters	D
.4	• Add. Code2	R	2 Byte	UINT16	Add. Info zum 2. abhängigen Parameter	D
	• ...					
.2*N-1	• IndexN (max. 14)	R	2 Byte	UINT16	Index des N. abhängigen Parameters	D
.2*N	• Add. CodeN (max. 14)	R	2 Byte	UINT16	Add. Info zum N. abhängigen Parameter	D
002B	ParamSet	R/W	2 Byte	UINT16	Parametersatzkennung Wenn das Gerät über Parametersätze verfügt, wird hier der jeweils aktuelle eingestellt / angezeigt. 0000h Default 0001h – FFFEh Der Parametersatz des Gerätes wurde über den Bus initialisiert. Zur Kennung hat er die frei wählbare Nr. 0001- FFFE erhalten. FFFFh Das Gerät wurde auf „Local Mode“ oder Handbetrieb geschaltet und es ist nicht sicher gestellt, dass der zuvor geladenen Parametersatz unverändert ist.	O
002C	ParameterMoment	R/W	2 Einträge	Record	Zeitpunkt der letzten Änderung der Parametrierung	O
.1	• Date	R/W	10+1 Byte	Visible-String	Datum YYYY-MM-DD	D
.2	• Time	R/W	8+1 Byte	Visible-String	Uhrzeit hh:mm:ss	D
002D	ResetParam	W	1 Byte	UINT 8	„01 _{hex} “ = Parametrierung zurücksetzen Mit diesem Kommando werden alle Einstellungen rückgängig gemacht und durch die Default Werte ersetzt. Das gilt auch für die Passworte und sonstige, durch den Anwender vorgenommenen Einstellungen. „02 _{hex} “ = Applikationsparametrierung zurücksetzen Mit diesem Kommando werden nur die Applikationsparameter rückgängig gemacht und durch die Default Werte ersetzt.	O

002E	Checksum	R	4 Byte	UINT32	Gerätespezifisch erzeugte eindeutige Checksumme (z.B. CRC16), die die Integrität der Parameterdaten sicher stellt. Der dafür verwendete Sicherungsmechanismus bezieht alle Parameterdaten des Gerätes ein. Auch ein „Änderungszähler“ ist möglich. Die Checksumme wird bei jeder Änderung eines Parameters neu generiert und ermöglicht es den Anwender zu kontrollieren, ob sich noch alle Parametereinstellungen unverändert sind. Dazu muß die aktuell ausgelesene Checksumme mit der bei der Parametrierung ausgelesenen Checksummer verglichen werden.	O
------	----------	---	-----------	--------	--	---

Treten beim Schreiben / Lesen der Objekte Fehler auf, z.B. es werden unzulässige Werte übertragen, so sind die Zugriffe mit den entsprechenden Fehlercodes (siehe: Kapitel „Liste der zulässigen Fehlercodes“) abzulehnen.

7.4.1 Blockparametrierung

Die Blockparametrierung dient dazu, voneinander abhängige Parameter gemeinsam übergeben zu können. Über den Parameter „Schreibsteuerung = 01“ wird die Blockparametrierung eingeleitet, und mit „Schreibsteuerung = 00“ beendet. Aufgetretene Konflikte werden im Parameter „Konflikt-Verzeichnis“ abgelegt.

Wird versucht voneinander abhängige Parameter einzeln zu parametrieren, kann es zum Fehler:

„Abhängigkeit von anderem Parameter nicht berücksichtigt“
kommen. Der oder die abhängigen Parameter finden sich im Konflikt-Verzeichnis. Der Parameter der gerade versucht wurde zu beschreiben, befindet sich an der ersten Position.

Result(-) für das Schreiben auf Objekte die nur in Abhängigkeit geändert werden dürfen

Error- Class “8” Other, Error- Code “1” Profile specific

Additional-Code	Bedeutung
0x0040 DependencyIgnored	Kollision mit anderen Werten Abhängigkeit nicht beachtet

7.4.1.1 Schreibsteuerung

Die Übergänge zwischen Einzelparametrierung und Blockparametrierung werden durch diesen Parameter ausgelöst.

Schreibsteuerung = „00_{hex}“ : Einzelparametrierung

Schreibsteuerung = „01_{hex}“ : Blockparametrierung

Folgende Aktionen werden bei einer Änderung des Parameterinhalts durchgeführt:

Schreibsteuerung = „00_{hex}“ -> Schreibsteuerung = „01_{hex}“:

- Einleiten der Blockparametrierung
- Keine Einzelparametrierung
- Zurücksetzen des Parameters 'Konflikt-Verzeichnis'

Schreibsteuerung = „01_{hex}“ -> Schreibsteuerung = „00_{hex}“:

- Beenden der Blockparametrierung
- ab jetzt wieder Einzelparametrierung

- Überprüfung auf Verträglichkeit:
 - Wenn diese gegeben ist,
 - werden die Parameterinhalte übernommen
 - der Schreibzugriff auf den Parameter 'Schreibsteuerung' positiv quittiert.
 - Bei Unverträglichkeit
 - bleiben die alten Inhalte aller an der Blockparametrierung beteiligten Parameter wirksam
 - Setzen des Parameters 'Konflikt-Verzeichnis'
 - der Schreibzugriff auf den Parameter 'Schreibsteuerung' wie folgt negativ quittiert.

Error- Class "8" Other, Error- Code "1" Profile specific

Additional-Code	Bedeutung
0x0040 DependencyIgnored	Kollision mit anderen Werten, Abhängigkeit nicht beachtet In der Regel wurden abhängige Werte nicht berücksichtigt.

7.4.1.2 Konflikt-Verzeichnis

Der Parameter enthält die Indizes und die Fehlermeldungen (Add.-Code) der am Konflikt beteiligten Parameter. Das 'Konflikt-Verzeichnis' muß bei

- negativer Quittierung von 'Blockschreib-Steuerung', und bei
- der Einzelparametrierung von voneinander abhängigen Parametern wenn die Abhängigkeit von anderem Parameter nicht berücksichtigt wird
- kann aber auch nach jedem Schreibzugriff, innerhalb und/oder außerhalb des Zustands 'Blockparametrierung', aktualisiert werden.

7.4.2 Parametersatzkennung

Dieser Parameter dient zur Kennzeichnung des aktuell wirksamen Geräte-Parametersatzes. Dies kann sich z.B. auf ein Schweißprogramm oder eine Rezeptur beziehen.

Parametersätze können vom Hersteller voreingestellt sein oder vom Anwender je nach Gerät mit Hilfe von Einzel oder Blockparametrierung oder auch „Download- Write“ Diensten geladen werden.

Kann das Gerät die empfangenen Geräte-Parameterwerte nicht netzausfallsicher speichern, so wird nach Einschalten der Netzspannung diese Parametersatzkennung vom Gerät selbständig auf 0000 gesetzt. Diese Information kann der Gerätenutzer auswerten und entsprechend neu initialisieren.

0000h	Der Parametersatz des Gerätes wurde noch nicht über den Bus initialisiert. Es ist der Default Parametersatz gültig
0001h-FFFFh	Der Parametersatz des Gerätes wurde über den Bus initialisiert. Zur Kennung hat er die frei wählbare Kenn-Nr. 1-FFFFh erhalten.
FFFFh	Das Gerät wurde oder ist auf Local-Mode geschaltet und es ist nicht sichergestellt, dass der zuvor geladene Parametersatz unverändert ist.

Der Gerätenutzer kann alle Werte zwischen 0 und FFFFh eintragen.

7.4.3 Passwortschutz

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
001D	Password	W	Max. 40 Byte	Octet-String	Mit diesem Kommando wird ein Passwort (z.B. „Superuser“) übertragen, dass für die folgenden Aktionen gültig ist. Der Zugriff ist beim Zugriff mit falschem oder fehlenden Passwort mit Verweis auf die fehlenden Rechte abzulehnen. Das Passwort wird deaktiviert bei: <ul style="list-style-type: none"> • Kommunikationsabbruch • Busstop (Bus Reset). • Schreiben keines Passwortes (String mit der Länge „0“) • Schreiben eines falschen Passwortes Für dieses Objekt kann es keinen Zugriffschutz geben.	O
001E	SetPassword	W	3 Einträge	Record	Mit diesem Objekt wird ein oder mehrere Passwort(e) für einen bestimmten Index gesetzt. Das Passwort wird nur beim Kommando „ResetParam“ wieder gelöscht. Der Hersteller kann defaultmäßig für jedes beliebige Objekt ein Passwort definiert haben. Um Änderungen der Passworte zu verhindern, kann dieses Objekt selbst geschützt werden. Derjenige der das Objekt „SetPassword“ schützt, ist quasi der „Superuser“. Durch den Hersteller können Passworte voreingestellt worden sein. Diese müssen nicht löschar sein.	O
	• Index	W	2 Byte	UINT16	„FFFF“ als Index setzt das Passwort für alle Objekte.	D
	• Add/Replace	W	1 Byte	Bit-String 8	0Xh = alle vorhandenen Passworte ersetzen 1Xh = zu vorhandenen Passwort hinzufügen X1h = gilt für den lesenden Zugriff X2h = gilt für den schreibenden Zugriff X3h = gilt für den lesenden und schreibenden Zugriff	D
	• Password	W	Max. 40 Byte	Visible-String	Passwort für das Objekt. „Index“ „Ersetzen“ und kein Passwort hebt den Passwortschutz auf.	D

Treten beim Schreiben / Lesen der Objekte Fehler auf, z.B. es werden unzulässige Werte übertragen, so sind die Zugriffe mit den entsprechenden Fehlercodes (siehe: Kapitel „Liste der zulässigen Fehlercodes“) abzulehnen.

Additional Code (hex)	Bedeutung
00B1	Das Passwort kann nicht ersetzt (gelöscht) werden.
00B2	Das Passwort kann nicht hinzugefügt werden (zu viele Passworte).
00B3	Das Passwort kann für die gewünschte Art des Zugriffes nicht vergeben werden.

7.5 Mehrsprachigkeit

Über das Objekt „Language“ kann die aktuell ausgewählte Sprache ausgelesen werden und auch die Sprachauswahl vorgenommen werden.

Die Domain- Variable „LanguageAvailable“ zeigt an, welche Sprachen verfügbar sind. Der erste Eintrag enthält die Default Sprache.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O	Beispiel
0017	Language	R/W	2 Einträge	Record	Objekt für die Sprachauswahl des Gerätes Hier kann die aktuell gültige Sprache ausgelesen und geändert werden.	M	
.1	• LanguageCode	R/W	5+1 Byte	Visible-String	Sprachen Code nach ISO 639-1 und optional durch “-“ getrennt, den Ländercode nach ISO 3166-1-Alpha-2 code z.B. en[-us] Wird der Ländercode nicht benutzt, wird mit 0x00 aufgefüllt. Der Sprachen Code der auszuwählenden Sprache wird hier eingeschrieben. Nach erfolgter pos. Response sind alle Textausgaben in der gewählten Sprache durchzuführen.	M	en
.2	• NameLanguage	R	Max. 50 Byte	Visible-String	Hier kann der Textstring für die aktuell gültige Sprache ausgelesen werden. Der Name ändert sich, sobald eine neue Nummer eingetragen wurde.	M	English
E804	LanguageAvailable	R		Domain Variable Record	Objekt für die Sprachauswahl des Gerätes – Anzeige aller verfügbaren Sprachen Sprachenname, schon in der entsprechenden Sprache	O	
Seg.1 -a	• LanguageCode1 (Default)	R	5+1	Visible-String	Sprachen Code nach ISO 639-2 (Default)	D	En
-b	• NameLanguage1 (Default)	R	Max. 50	Visible-String	Textstring für die 1. verfügbare Sprache. (Default)	D	English
Seg.2 -a	• LanguageCode2	R	5+1	Visible-String	Sprachen Code nach ISO 639-2 der 2. verfügbaren Sprache	O	De
-b	• NameLanguage2	R	Max. 50	Visible-String	Textstring für die 2. verfügbare Sprache	O	Deutsch
Seg.3 -a	• LanguageCode3	R	5+1	Visible-String	Sprachen Code nach ISO 639-2 der 3. verfügbaren Sprache	O	Fr
-b	• NameLanguage3	R	Max. 50	Visible-String	Textstring für die 3. verfügbare Sprache	O	Français
Seg.4 -a	• LanguageCode4	R	5+1	Visible-String	Sprachen Code nach ISO 639-2 der 4. verfügbaren Sprache	O	Es
-b	• NameLanguage4	R	Max. 50	Visible-String	Textstring für die 4. verfügbare Sprache	O	Español

Seg.N -a	• LanguageCodeN	R	5+1	Visible-String	Sprachen Code nach ISO 639-2 der N. verfügbaren Sprache	O	It
-b	• NameLanguageN	R	Max. 50	Visible-String	Textstring für die N. verfügbare Sprache.	O	Italiano

Treten beim Schreiben / Lesen der Objekte Fehler auf, z.B. es werden unzulässige Werte übertragen, so sind die Zugriffe mit den entsprechenden Fehlercodes (siehe: Kapitel „Liste der zulässigen Fehlercodes“) abzulehnen.

Das SubObjekt „LanguageCode“ (Subindex 1) enthält die Sprachcodes nach Sprachen Code nach ISO 639-1 und optional durch „-“ getrennt, den Ländercode nach ISO 3166-1-Alpha-2 code. Wird der Ländercode nicht benutzt, wird mit 0x00 aufgefüllt.

Durch Schreiben dieses Subobjektes wird die Sprache ausgewählt. Diese Sprache taucht dann im SubObjekt „NameLanguage“ auf.

Beim schreibenden Zugriff auf das gesamte Objekt ist „Language“ wird der Eintrag für das SubObjekt „NameLanguage“ ignoriert, da das SubObjekt „NameLanguage“ den „Nur lesen“ Status hat.

Jedes Segment der Domainvariable „LanguageAvailable“ enthält genau die beiden SubObjekte „LanguageCode und „NameLanguage“ der verfügbaren Sprache. Die einzelnen Read- Dienste des „Upload-Read“ Makrodienstes haben also (fast) immer andere Längen, je ein Spracheintrag wird aber immer komplett in einem Segment übertragen. Dadurch wird eine einfache Handhabung erreicht.

7.6 Modulare Geräte

Neben kompakten Geräten gibt es vermehrt auch modulare Geräte. Deren Handhabung soll ebenfalls in diesem Profil berücksichtigt werden. Ein modulares Gerät definiert sich wie folgt:

- Es besitzt genau ein Kommunikationszugangsmodul (Kopf).
- Es besitzt N Module, wobei $0 \leq N \leq 253$.
- Ein Modul muß nicht notwendigerweise aktiv / vorhanden sein.
- Jedes Modul hat seinen eigenen unabhängigen Satz von Objekten.
Dieser Objektraum muß nicht der in diesem Basisprofil beschriebene Objektraum sein, sondern richtet sich nach dem Subsystem.

Die Modulstruktur eines Modularen Gerätes ist im Objekt 0x0036 „DeviceStructure“ beschrieben.

Zur Adressierung von Modulen eines modularen Gerätes wird die bis dato nicht genutzte und nur aus Kompatibilitätsgründen vorhandene Invoke-ID benutzt. Aus diesem Grund wird auch der Name dieses Parameters in „Module Number“ (Modulnummer) geändert. Die Handhabung des Parameters ändert sich hingegen nicht. In der Response wird weiterhin die Modulnummer der Indication eingetragen.

Das Verhalten für kompakte Geräte ändert sich ebenfalls nicht. Hier kann weiterhin eine beliebige Modulnummer genutzt werden. Es wird jedoch aus Gründen der Durchgängigkeit empfohlen, als Modulnummer „00“ zu wählen.

Ein Modulares Gerät als Ganzes kann immer über die Modulnummer 0xFF angesprochen werden. Die einzelnen Module über die entsprechende Modulnummer.

Die Modulnummer sollte mit 1 beginnen. Der Kopf eines modularen Gerätes muß nicht notwendigerweise die Modulnummer „1“ haben

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
0035	NoOfModules	R	1 Byte	UINT8	Nummer des letzten aktuell vorhandenen Moduls. (Anzahl der Module inklusive Platzhalter.) Der Parameter „Module Number“ hat für dieses Objekt keine Bedeutung. <ul style="list-style-type: none"> • Ist dieses Objekt nicht existent, handelt es sich um ein kompaktes und nicht um ein modulares Gerät. • Wird „0“ zurück gegeben, handelt es sich um ein modulares Gerät, in dem z.Zt. keine Module eingebaut (gesteckt) sind. 	D
0036	DeviceStructure	R/W	4 Einträge	Record	Hier wird die Struktur des modularen Gerätes und die Abbildung der Prozessdaten der Module auf die Prozessdaten des Gerätes festgelegt. Dabei kann Start und Länge der Abbildung der Prozessdaten für jedes Modul unter seiner Modulnummer aufgerufen werden. Diese Objekt eignet sich auch, auch um kompakte Geräte zu beschreiben. Pflichtobjekt für modulare Geräte.	D
.1	<ul style="list-style-type: none"> • PDINOffset 	R/W	2 Byte	UINT16	Bit-Offset im Eingangs-Prozessdatenkanal	D

.2	• PDINLength	RW	2 Byte	UINT16	Bit-Länge in Bit im Eingangs-Prozessdatenkanal	D
.3	• PDOOUTOffset	RW	2 Byte	UINT16	Bit-Offset im Ausgangs-Prozessdatenkanal	D
.4	• PDOOUTLength	RW	2 Byte	UINT16	Bit-Länge in Bit im Ausgangs-Prozessdatenkanal	D

7.7 Objektbeschreibung

Im Inbetriebnahme- und Servicefall ist es nicht nur wichtig die Soll- Parametrierung zu kennen, sondern es ist notwendig die Ist- Parametrierung des Gerätes zu kennen. Voraussetzung dafür ist die Kenntnis der vorhandenen Anwender- Objekte. Diese Objekte und deren Bedeutung kann mit den folgenden beiden Profilobjekten ausgelesen werden.

Index (hex)	Objektname	R/W	Länge	Datentyp	Bedeutung	M/O
0038	ObjDescrReq	RW	2 Einträge	Record	Objektes dessen Beschreibung angefragt wird.	O
	• Index	RW	2 Byte	UINT16		D
	• SubIndex	RW	1 Byte	UINT8		D
0039	ObjDescr	R	16 Einträge	Record	Beschreibung des Objektes dessen Index angefragt wurde. Selbstinkrementierend zum nächsten gültigen Index / Subindex beim nächsten Zugriff. Leerobjekte bzw. nicht vorhandene Indices werden übersprungen.	O
.1	• Index	R	2 Byte	UINT16	Objektnummer Z.B. 0x6051	D
.2	• SubIndex	R	1 Byte	UINT8	Substrukturnummer eines Objektes Z.B. 0x00	D
.3	• ObjectCode	R	1 Byte	UINT8	Wie unter „Datenobjekte“ definiert Z.B. 0x07 - Simple-Variable	D
.4	• IndexOfType	R	1 Byte	UINT8	Wie unter „Datentypen“ definiert Z.B. 0x0A – OctetString	D
.5	• Length	R	1 Byte	UINT8	Länge des (Sub-) Objektes in Byte Z.B. 0x04	D
.6	• UnitText	R	5+1 Byte	Visible-String	Einheit des Wertes – 0x00 terminiert Z.B. U/min	D
.7	• UnitCode	R	1 Byte	UINT8	Einheiten Größen Code des Wertes lt. Tabelle „Indices der Physikalischen Größen“ Z.B. 0x11	D
.8	• UnitCodeExp	R	1 Byte	UINT8	Einheiten Exponenten Code des Wertes lt. Tabelle „Indices der Physikalischen Größen“ Der Einheiten Exponenten Code gibt die Zehnerpotenz des Einheitenwertes an. Der Einheiten Exponenten Code hat den Wertebereich - 127 bis 128. Bsp. <ul style="list-style-type: none"> • Einheitenindex 0 für 10⁰ • Einheitenindex 3 für 10³ • Einheitenindex -3 für 10⁻³ usw. Die aufgelisteten Einheiten Exponenten Codes für SI-kompatible Einheiten	D

					(Einheiten Exponenten Code < 64) sollen nur als Beispiel dienen. Die Einheitenindizes für weitere SI-kompatible Vorsätze (Pico usw.) ergeben sich analog. Einheiten Exponenten Code die größer als +64 sind, haben eine Sonderbedeutung, die aus der Tabelle ermittelt werden müssen. Bei diesen Einheiten handelt es sich z.B. um die Einheiten Tag, Stunde, Minute oder um nicht SI-kompatible Einheiten, wie Fahrenheit. Z.B. 0x73	
.9	• Offset	R	2 Byte	INT16	Offset des entsprechenden Wertes. Dieser muß zu den Prozeßdaten hinzu addiert werden, um so, unter Berücksichtigung der Auflösung den Messwert zu errechnen. Bsp: der ProzessdatenWert ist 12345, der Offset ist -15000, daraus ergibt sich bei obiger Auflösung ein Messwert von -1,77V	D
.10	• RDR	R	2 Byte	UINT16	ResolutionDimensionRange $\text{Resolution} = \frac{RDR}{RNR}$ Dimensionsbereich der Auflösung, bezogen auf die Einheit in der die Größe dargestellt wird, in dem sich das Prozessdatum bewegt. Wird zur Berechnung der Auflösung herangezogen. Bsp: 10 (Volt) Auflösung = 10V / 3000 =3,333mV	D
.11	• RNR	R	2 Byte	INT16	ResolutionNumberRange $\text{Resolution} = \frac{RDR}{RNR}$ Zahlenbereich der Auflösung, bezogen auf den Dimensionsbereich, in dem sich das Prozessdatum bewegt. Wird zur Berechnung der Auflösung herangezogen. Bsp: 3000 Auflösung = 10V / 3000 =3,333mV	D
.12	• Access-Rights	R	1 Byte	Bit-String 8	Bit 0 = WriteProtected (1=aktiv) Bit 1 = ReadProtected (1=aktiv) Bit 2 = Write_OnlyWithPassword (1=aktiv) Bit 3 = Read_OnlyWithPassword (1=aktiv) Bit 4-7 Reserved	D
.13	• DisplayFormat	R	1 Byte	UINT8	0x00 – Undefined 0x01 - Binary 0x02 - Unsigned Decimal 0x03 - Signed Decimal 0x04 - hexadecimal 0x05 - Text 0x06 - Float 0x07 - Time 0x08 – 0xFF Reserved	D
.14	• Min	R	“Length” Byte	Nach Datentyp	Untere zulässige Grenze eines Wertes. Die Länge richtet sich nach „.5 Length“ des Objekts. Für alle Stringvariablen gibt es keinen Min-Wert. Die Länge ist in diesem Fall 0. Z.B. -500	D
.15	• Max	R	“Length” Byte	Nach Datentyp	Obere zulässige Grenze eines Wertes. Die Länge richtet sich nach „.5 Length“ des Objekts.	D

					Für alle Stringvariablen gibt es keinen Min-Wert. Die Länge ist in diesem Fall 0. Z.B. 500	
.16	• Symbol	R	M Byte	Visible-String	Name des Objektes. Die Länge kann bis max. 58 Byte aufgefüllt werden. maxM=58-2xLength-22 Z.B. Solldrehzahl – 0x00 terminiert	D
E805	ObjDescrLong	R	N Segmente	Domain Variable	Beschreibung aller vorhandener Objekte (Index / Subindex) Pro Segment wird eine Beschreibung übertragen. Strings werden mit 0x00 aufgefüllt.	O
Seg.1	ObjDescr1	R	58 Byte	Record	Siehe oben	D
...	
Seg.N	ObjDescr1	R	58 Byte	Record	Siehe oben	D

Besitzt ein Objekt Subindices, so reihen sich diese hinter dem Index ein.

Wird über das Objekt 0x0038 „ObjDescrReq“ ein nicht vorhandener Index/Subindex angefordert erfolgt eine Fehlermeldung Error-Class 8 (Application), Error-Code 1 (Profile specific), Additional Code 0x0030 (Wertebereich eines Parameters ist verlassen). Im Objekt 0x0039 „ObjDescr“ wird dann die Beschreibung des nächsten vorhandenen Index/Subindex zur Verfügung gestellt.

Nach dem letzten Index/Subindex, wird wieder von vorn begonnen.

Durch die Autoinkrementfunktion (nach einem Lesezugriff wird für den nächsten Lesezugriff die Beschreibung des nächsten vorhandenen Index/Subindex zur Verfügung gestellt.) des Objektes 0x0039 ist es einer Applikation leicht möglich, alle vorhandenen Indices/Subindices auszulesen und somit ein vollständiges Abbild aller im Gerät vorhandenen Objekte zu erhalten.

Für die Objekte des Basisprofils muß keine Beschreibung hinterlegt werden, da diese (inklusive der Subindices) ja durch das Basisprofil systemweit bekannt ist. Das Objekt 0x0039 enthält dann lediglich den Index. Die restlichen Einträge fehlen (Länge des Nutzdatums: 2Byte). Damit wird das Vorhandensein von optionalen Objekten angezeigt. Ein Basisprofil Objekt darf deshalb nicht übersprungen werden.

Eine Ausnahme bilden die Prozessdatenobjekte 0x0025 und 0x0026, da deren Vorhandensein und Funktion zwar profilspezifisch, deren Struktur aber herstellerspezifisch ist und nur so bestimmt werden kann. Sie müssen deshalb vollständig beschrieben werden.

Ist das Objekt 0x0039 nicht vorhanden, sind die Prozessdatenobjekte 0x0025 und 0x0026 defaultmäßig als Bitstring der entsprechenden Länge definiert.

Tabelle: Indices der Physikalischen Größen

Physikalische_Größe	Größenindex	Einheit	Einheitenindex (nicht vollständig)
	0	dimensionslos	0
Länge	1	Meter	0
Fläche	2	Quadratmeter	0
Volumen	3	Kubikmeter	0
Zeit	4	Sekunde	0
		Minute	70
		Stunde	74
		Tag	77
		Halbwellen	78
		Millisekunde	-3
		Mikrosekunde	-6
Kraft	6	Newton	0
Wirkleistung	9	Watt	0
Scheinleistung	10	Voltampere	0
Drehzahl	11	1/Sekunde	0
		1/Minute	73
		1/Stunde	74
Winkel	12	Radian	0
		Sekunde	75
		Minute	76
		(Alt-)Grad	77
		Neugrad	78
Geschwindigkeit	13	Meter/Sekunde	0
		Millimeter/Sek.	-3
		Millimeter/Min.	79
		Meter/Min.	80
		Kilometer/Min.	81
		Millimeter/Std.	82
		Meter/Std.	83
		Kilometer/Std.	84
Drehmoment	16	Newtonmeter	0
Temperatur	17	Kelvin	0
		Grad_Celsius	94
		Grad_Fahrenheit	95
Elektrische_Spannung	21	Volt	0
Elektrischer_Strom	22	Ampere	0
Elektrischer_Widerstand	23	Ohm	0
Verhältnis	24	Prozent	0
Frequenz	28	Hertz	0
Schritte	32	Schritte	0
Encoderauflösung	33	Schritte/Umdrehung	0
Durchfluß	34	Kubikmeter/Sekunde	0
		Liter/Sekunde	-3
		Milliliter/Sekunde	-6
		Kubikmeter/Minute	85
		Liter/Minute	86
		Milliliter/Minute	87
Beschleunigung	35	1/Sekunde ²	0

Der EinheitenIndex entspricht im Standardfall dem Exponenten (zur Basis 10):

z.B.

UnitCodeExp -6 für 10^3 =>kilo

UnitCodeExp -6 für 10^0 =>ohne

UnitCodeExp -6 für 10^{-3} =>m

UnitCodeExp -6 für 10^{-6} => μ

UnitCodeExp -9 für 10^{-9} =>n

8 Anhang A

8.1 Begriffsdefinitionen

Ersatzwerte

Wenn die optionalen Kommunikationsobjekte nicht implementiert sind, verhält sich das Gerät, adäquat zu dem, für diesen Parameter definierten Ersatzwert.

Fehlermeldung

Die Fehlermeldung wird zurückgegeben, wenn ein Service nicht ausgeführt werden konnte.

Funktionseinheiten - Definition der

Die Gerätefunktion wird durch eine Funktionseinheit beschrieben. Die Funktion wird durch die Eingänge gesteuert und parametrisiert. Außerdem können interne Signale auf die Funktion einwirken oder die Funktion wird über interne Parameter beeinflusst. Der Ausgang der Funktion kann auf die Eingänge von anderen Funktionen geschaltet oder über den Bus zugänglich gemacht werden.

Funktionseinheit
Eingang:
Interne Signale
Aktionen
Externe Signale
Parameter (Services, Prozeßdaten)
Kommunikationsobjekte
Funktionsausgang:
interne Variable
Kommunikationsobjekte

Geräteparameter

Für alle Geräteparameter sind Defaultwerte in diesem Profil festgelegt.

Geräteprofil

Das Geräteprofil legt - die über die Kommunikation - sichtbaren Anwendungsfunktionen fest. Die Anwendungsfunktionen werden durch folgende Festlegungen auf die Kommunikation abgebildet:

- Durch das Kommunikationsprofil,
- durch die Interaktionen zwischen den Anwendungsfunktionen, soweit sie über das Kommunikationssystem ausgeführt werden, sowie
- durch die genutzten Kommunikationsdienste und der damit manipulierbaren Kommunikationsobjekte.

Das Ergebnis der Abbildung ist das sichtbare Verhalten der Anwendung. Die Festlegungen eines Anwendungsprofils ermöglichen die Interoperabilität in einem Anwendungsfeld. Die Prämisse hierfür ist, daß die genutzten Geräteeigenschaften dieses zulassen.

Weiterhin werden Eigenschaften der Geräte festgelegt, die für den Nutzer von Bedeutung sind. Es wird unterschieden zwischen Pflichtfunktionen (mandatory), optionalen und herstellerspezifischen Gerätefunktionen sowie Parametern. Beschränkt sich der Anwender auf die Pflichtfunktionen oder Parameter, ist eine Austauschbarkeit der Geräte möglich - wenn die genutzten Geräteeigenschaften und -einstellungen dieses zulassen. In Bezug auf die Kommunikation sind die Geräte - unabhängig von der Funktion - bei gleichen Parametern immer austauschbar.

Index, Subindex

Der Index dient dazu, einen Parameter (Kommunikationsobjekt) zu adressieren. Der Subindex adressiert innerhalb eines Parameters, der als Struktur angelegt ist, einen Subparameter (Element eines Kommunikationsobjektes).

Kommunikationsinterface

Das Kommunikationsinterface setzt sich aus einem Prozeßdatenkanal und einem Parameter-Kanal zusammen.

Kommunikationsprofil

Im Kommunikationsprofil werden, die in der Spezifikation des Übertragungsmediums enthaltenen Freiheitsgrade, anwendungs- oder gerätegruppenspezifisch eingeschränkt bzw. klassifiziert. Im Kommunikationsprofil werden Kommunikations-Dienste und -Parameter festgelegt, die in der Spezifikation als optional gekennzeichnet sind. Alle im Kommunikationsprofil nicht genannten optionalen Funktionen und Parameter bleiben optional. Mandatory-Dienste und -Parameter sind auch ohne Nennung im Profil bindend. In dem Profil werden weiterhin Wertebereiche von Attributen und Parametern eingegrenzt bzw. festgelegt.

Kommunikationsreferenz

Jede Kommunikationsbeziehung zwischen zwei Geräten wird projiziert, unabhängig vom Zeitpunkt ihrer Nutzung. Die Projektierung wird in jedem Busteilnehmer in einer kommunikationsbeziehungsliste (KBL) gespeichert. Ein Anwendungsprozeß identifiziert die Kommunikationsbeziehung über eine lokale Kommunikationsreferenz. Diese dient also zum Adressieren des Kommunikationspartners.

Parameter-Kanal

Über den Parameter-Kanal kann auf alle Kommunikationsobjekte zugegriffen werden. Die Parameterkanal-Dienste erlauben einen quittierten Zugriff auf die Geräteparameter, d.h. der Zugriff auf einen Geräteparameter wird vom Gerät bestätigt.

Pflichtbereich

Der Pflichtbereich ist der Wertebereich, auf den ein Parameter, falls er implementiert ist, in jedem Fall parametrisiert werden kann.

Prozeßdatenbeschreibung

Falls das Gerät parallel mehrere Prozeßdaten überträgt oder empfängt, muß die Struktur und Bedeutung dieser Prozeßdaten durch den Hersteller beschrieben werden.

Prozeßdatenkanal

Der Prozeßdatenkanal dient zur schnellen Übertragung von Prozeßdaten. Über den Prozeßdatenkanal werden Daten unquittiert und äquidistant übertragen. Prozeßdaten können gelesen und geschrieben werden.

Die Richtungsangabe der Prozeßdaten wird vom Bus gesehen; d.h.,

- Prozeßausgangsdaten sind Daten, die vom Steuerungssystem zum Gerät übertragen werden. Dieses ließt diese Daten aus dem Prozeßdatenkanal und gibt sie je nach Funktion an den Prozeß aus.
- Prozeßeingangsdaten sind Daten, die vom Gerät zum Steuerungssystem übertragen werden. Das Gerät schreibt diese Daten in den Prozeßdatenkanal und überträgt sie damit zum Steuerungssystem.

VFD-Objekt

Das Virtual Field Device (VFD) ist ein abstraktes Modell zur Beschreibung der Daten und des Verhaltens eines Automatisierungsgerätes aus der Sicht seines Kommunikationspartners. Die Basis des VFD-Modells ist das VFD-Objekt. Das VFD-Objekt beinhaltet alle Objekte und Objektbeschreibungen, die von einem Kommunikationspartner durch einen Services genutzt werden können. Die Objektbeschreibungen sind in einem Objektverzeichnis hinterlegt. Je VFD gibt es genau ein Objektverzeichnis.

Zustandsmaschine

In diesem Profil sind einige Funktionen mit Hilfe einer Zustandsmaschine beschrieben. Ein Zustand repräsentiert ein bestimmtes internes und externes Verhalten. Er kann nur durch definierte Ereignisse

verlassen werden. Den Ereignissen sind entsprechende Zustandsübergänge zugeordnet. In einem Übergang können Aktionen ausgeführt werden. In diesem wird das Zustandsverhalten geändert. Mit Beendigung des Übergangs wird der aktuelle Zustand auf den Folgezustand geändert.

8.2 Symbole und Abkürzungen

Netzwerkspezifische Abkürzungen

ALI	Application Layer Interface
KBL	Kommunikationsbeziehungsliste
KBL-Header	Kopf der Kommunikationsbeziehungsliste
KR	Kommunikationsreferenz
m	mandatory
MAP	Manufacturing Automation Protocol
o	optional
OV	Objektverzeichnis
PCP	Peripherals Communication Protocol
PCh	Parameter Channel
PD	Prozeßdaten
PMS	Peripherals Message Specification
S-OV	Statisches-Objektverzeichnis
ST-OV	Statisches Typverzeichnis
VFD	Virtual Field Device
.con	Confirmation-Primitive
.ind	Indication-Primitive
.req	Request-Primitive
.res	Response-Primitive

9 Anhang B

9.1 Übersetzungstabelle für die Objektnamen

Index	Sub-Index	Objektname (Basisprofil)	DE	EN-US	ES	FR	IT	ZH-CN
0001		VendorName	Herstellername	Vendor Name	Nombre del fabricante	Nom du fabricant	Nome del produttore	厂商名称
0002		VendorID	Herstellerkennung	Vendor ID	Identificación del fabricante	Identificateur du fabricant	Identificativo produttore	厂商标识
0003		VendorText	Herstellertext	Vendor Text	Texto del fabricante	Texte du fabricant	Testo produttore	厂商文本
0004		DeviceFamily	Gerätefamilie	Device Family	Familia de dispositivos	Famille d'appareils	Famiglia di dispositivi	设备系列
0005		Capabilities	Eigenschaften	Capabilities				
0006		ProductFamily	Produktfamilie	Product Family	Familia de productos	Famille de produits	Famiglia di prodotti	产品系列
0007		ProductName	Produktname	Product Name	Nombre del producto	Nom du produit	Nome del prodotto	产品名称
0008		SerialNo	Seriennummer	Serial Number	Número de serie	Numéro de série	Numero di serie	序列号
0009		ProductText	Produkttext	Product Text	Texto de producto	Texte du produit	Testo prodotto	产品文本
000A		OrderNumber	Bestellnummer	Order Number	Número de pedido	Numéro d'ordre	Numero d'ordinazione	订货号
000B		HardwareVersion	Hardware-Version	Hardware Version	Versión del hardware	Version du matériel	Versione hardware	硬件版本
	.1	BuildDate	Hardware-Version - Herstelldatum	Hardware Version - Build Date	Versión del hardware - Fecha de fabricación	Version du matériel - Date de fabrication	Versione hardware - Data di produzione	硬件版本 - 创建日期
	.2	Version	Hardware-Version - Versionskennung	Hardware Version - Version	Versión del hardware - Identificación de versión	Version du matériel - Identificateur de version	Versione hardware / ID versione	硬件版本 - 版本标识
000C		FirmwareVersion	Firmware-Version	Firmware Version	Versión del firmware	Version du microprogramme	Versione firmware	固件版本
	.1	BuildDate	Firmware-Version - Datum	Firmware Version - Build Date	Versión del firmware - Fecha	Version du microprogramme - Date	Versione firmware - Data	固件版本 - 日期
	.2	Version	Firmware-Version - Versionskennung	Firmware Version - Version	Versión del firmware - Identificación de	Version du microprogramme - Identificateur de	Versione firmware / ID versione	固件版本 - 版本标识

Index	Sub-Index	Objektname (Basisprofil)	DE	EN-US	ES	FR	IT	ZH-CN
					versión	version		
000D		PChVersion	Parameterkanal-Version	Parameter Channel Version	Versión del canal de parámetros	Version du canal de paramètres	Versione PCh	参数通道版本
	.1	BuildDate	Parameterkanal-Version - Datum	PCP Version - Build Date	Versión de canal de parámetros - Fecha	Version du canal de paramètres - Date	Versione PCP - Data	参数通道版本 - 日期
	.2	Version	Parameterkanal-Version - Versionskennung	PCP Version - Version	Versión de canal de parámetros - Identificación de versión	Version du canal de paramètres - Identificateur de version	Versione PCP - ID versione	参数通道版本 - 版本标识
000E		CommProfile	Kommunikationsprofil	Communication Profile	Perfil de comunicación	Profil de communication	Profilo di comunicazione	通信概要
000F		DeviceProfile	Geräteprofil	Device Profile	Perfil del dispositivo	Profil	Profilo dispositivo	设备概要
0010		Reserved	Reserviert	Reserved	Reservado	Réservé	Riservato	保留
0011		ProfileVersion	Profil-Version	Profile Version	Versión del perfil	Version du profil	Versione profilo	概要版本
	.1	BuildDate	Profil-Version - Datum	Profile Version - Build Date	Versión del perfil - Fecha	Version du profil - Date	Versione profilo - Data	概要版本 - 日期
	.2	Version	Profil-Version - Versionskennung	Profile Version - Version	Versión del perfil - Identificación de versión	Version du profil - Identificateur de version	Versione profilo - ID versione	概要版本 - 版本标识
0012		VendorURL	Hersteller URL	Vendor URL	URL del fabricante	URL du fabricant	URL venditore	厂商URL
0013		DeviceDescFile	Gerätebeschreibungsdatei	Device Description File	Archivo de descripción del dispositivo	Fichier de description d'appareil	File descrizione dispositivo	设备描述文件
0014		Location	Einbauort	Location	Lugar de montaje	Emplacement de montage	Posizione di montaggio	安装地点
0015		EquipmentIdent	Betriebsmittel-Kennzeichen	Equipment Identifier	Identificación del equipo	Identificateur de matériel	Identificatore dispositivo (BMK)	设备标识符
0016		ApplDeviceAddr	applikationsspezifische Geräteadresse	Application Device Address	Dirección del dispositivo específica de la aplicación	Adresse d'appareil spécifique à l'application	indirizzo dispositivo per la specifica applicazione	应用特定的设备地址
0017		Language	Sprache	Language	Idioma	Langue	Lingua	语言
	.1	LanguageCode	Sprache - Ländercode	Language - Language Code	Idioma - Código del país	Langue - Code pays	Lingua - Codice paese	语言 - 国家代码
	.2	NameLanguage	Sprache - Name	Language - Name	Idioma - Nombre	Langue - Nom	Lingua - Nome	语言 - 名称
0018		DiagState	Diagnosestatus	Diagnostic State	Estado de diagnóstico	État du diagnostic	Stato diagnostico	诊断状态

Index	Sub-Index	Objektname (Basisprofil)	DE	EN-US	ES	FR	IT	ZH-CN
	.1	Lfd.Nr.	Diagnosestatus - Lfd. Nr.	Diagnostic State - Cons. No.	Estado de diagnóstico - N° de orden	État du diagnostic - N° de série	Stato diagnostico - Numero d'ordine	诊断状态 - 连续编号
	.2	Priority	Diagnosestatus - Priorität	Diagnostic State - Priority	Estado de diagnóstico - Prioridad	État du diagnostic - Priorité	Stato diagnostico - priorità	诊断状态 - 优先级
	.3	Channel/Group/Module	Diagnosestatus - Kanal/Gruppe/Modul	Diagnostic State - Channel/Group/Module	Estado de diagnóstico - Canal/grupo/módulo	État du diagnostic - Canal/groupe/module	Stato diagnostico - Canale/gruppo/modulo	诊断状态 - 通道/组/模块
	.4	Code	Diagnosestatus - Code	Diagnostic State - Code	Estado de diagnóstico - Código	État du diagnostic - Code	Stato diagnostico - Codice	诊断状态 - 代码
	.5	MoreFollows	Diagnosestatus - Zusatzinformationen	Diagnostic State - More Follows	Estado de diagnóstico - Información adicional	État du diagnostic - Informations supplémentaires	Stato diagnostico - Informazioni aggiuntive	诊断状态 - 附加信息
	.6	Text	Diagnosestatus - Text	Diagnostic State - Text	Estado de diagnóstico - Texto	État du diagnostic - Texte	Stato diagnostico - Testo	诊断状态 - 文本
0019		ResetDiag	Diagnosemeldungen quittieren	Reset Diagnostic Messages	Confirmar mensajes de diagnóstico	Acquitter les messages de diagnostic	Consenso messaggi diagnostici	应答诊断信息
001A		GetErrorRepMethod	Meldemethode für Störung	Get Error Report Method	Método de avisar un fallo	Méthode de signalisation des perturbations	Metodo avvertimento guasti	故障提示方法
001B		TestMode	Testbetrieb	Test Mode	Servicio de prueba	Mode d'essai	Modalità di prova	试运行
001C		ControlTrace	Steuerung der Protokollierung	Control Trace	Mando del registro	Commande de la journalisation	Controllo protocollo	记录控制
001D		Password	Passwort	Password	Contraseña	Mot de passe	Password	密码
001E		SetPassword	Passwort setzen	Set Password	Establecer contraseña	Définir le mot de passe	Impostare password	设置密码
001F		PDTimeout	Prozessdaten-Überwachungszeit	Process Data Timeout	Tiempo de monitorización de los datos de proceso	Temps de surveillance des données de process	Tempo di monitoraggio dei dati di processo	过程数据监视时间
0020		PDTimeoutCode	Prozessdaten-Überwachungs-Code	Process Data Timeout Code	Código de monitorización de los datos de proceso	Code de surveillance des données de process	Codice di monitoraggio dei dati di processo	过程数据监视代码
0021		PCPTimeout	Parameterkanal-Überwachungszeit	PCP Timeout	Tiempo de monitorización del	Temps de surveillance du canal	Tempo di monitoraggio dei dati	参数通道监视时间

Index	Sub-Index	Objektname (Basisprofil)	DE	EN-US	ES	FR	IT	ZH-CN
					canal de parámetros	de paramètres	di processo	
0022		PCPTimeoutCode	Parameterkanal-Überwachungs-Code	PCP Timeout Code	Código de monitorización del canal de parámetros	Code de surveillance du canal de paramètres	Codice di monitoraggio PCP	参数通道监视代码
0023		AbortCode	Verbindungsabbruch-Code	Connection Abort Code	Código de interrupción de comunicación	Code d'interruption de la connexion	Codice per interruzione connessione	连接中断代码
0024		IBSResetCode	Reset-Code	INTERBUS Reset Code	Código de restablecimiento	Code de réinitialisation	Codice di reset	复位代码
0025		PDIN	Prozesseingangsdaten	Process Input Data	Datos de proceso de entrada	Données de process entrantes	Dati ingresso processo	过程输入数据
0026		PDOUT	Prozessausgangsdaten	Process Output Data	Datos de proceso de salida	Données de process sortantes	Dati uscita processo	过程输出数据
0027		GetExRight	Exklusive Schreibrechte anfordern	Get Exclusive Process Data Write Rights	Solicitar derechos de escritura exclusivos	Demander les autorisations d'accès en écriture exclusives	Richiedere diritto di scrittura esclusivo	□□□□□□
0028		ChangePDSet	Prozessdatenzuordnungen einstellen	Change Process Data Settings	Ajustar las asignaciones de datos de proceso	Régler les affectations des données de process	Impostare assegnazione dati di processo	设置过程数据分配
0029		ParamSetWriteControl	Parametersatz-Schreibsteuerung	Parameter Set Write Control	Control de escritura de juego de parámetros	Commande d'écriture du jeu de paramètres	Controllo scrittura set parametri	参数组写控制
002A		ConflictDictionary	Konflikt-Verzeichnis	Conflict Dictionary	Directorio de conflicto	Répertoire des conflits	Elenco conflitti	冲突目录
002B		ParamSet	Parametersatzkennung	Parameter Set Identification	Identificación de juego de parámetros	Identificateur de jeu de paramètres	Identificatore set parametri	参数组标识
002C		ParameterMoment	Zeitpunkt der letzten Parametrierungsänderung	Moment of Last Parameterization Modification	Hora y fecha del cambio de parametrización más reciente	Moment de la dernière modification du paramétrage	Ultima modifica parametro	最后一次参数化更改的时间点
002D		ResetParam	Parametrierung zurücksetzen	Reset Parameterization	Restablecer parametrización	Réinitialiser le paramétrage	Reset parametrizzazione	复位参数化
002E		CheckSum	Prüfsumme	Checksum	Suma de comprobación	Somme de contrôle	Somma di controllo	检查和
002F		PDOUT_Subst	Ersatzwert Prozessausgangsdaten	Process Output Data Substitute	Valor de reserva para datos de proceso de salida	Valeur de remplacement des données de process	Valore sostitutivo per dati di processo in uscita	过程输出数据替换值

Index	Sub-Index	Objektname (Basisprofil)	DE	EN-US	ES	FR	IT	ZH-CN
						sortantes		
0030		PF_Code	Peripherie-Fehler-Auswahlcode	Peripheral Fail Code	Código de selección de fallo en periferia	Code de sélection de l'erreur périphérique	Codice selezione errore periferia	外设错误选择代码
0031		PDIN_Subst	Ersatzwert Prozesseingangsdaten	Process Input Data Substitute	Valor de reserva para datos de proceso de entrada	Valeur de remplacement des données de process entrantes	Valore sostitutivo per dati di processo in ingresso	过程输入数据替换值
0032		IBS_ID	INTERBUS Identifikation / reserviert	INTERBUS Identification / reserved	Identificación de INTERBUS / reservado	Identification INTERBUS / Réserve	Identificazione INTERBUS / riservato	INTERBUS识别 / 保留
	.1	ID-Code	INTERBUS Identifikation - ID-Code	INTERBUS Identification - ID Code	Identificación de INTERBUS - Código ID	Identification INTERBUS - Code d'identification	Identificazione INTERBUS - Codice ID	INTERBUS识别- ID代码
	.2	PDLength	INTERBUS Identifikation - Prozessdatenlänge	INTERBUS Identification - Process Data Length	Identificación de INTERBUS - Longitud de datos de proceso	Identification INTERBUS - Longueur des données de process	Identificazione INTERBUS - Lunghezza dati di processo	INTERBUS识别 - 过程数据长度
0033		DiagStateChannelNo	Diagnosestatus-Kanal-Nr.	Diagnostic State Channel No.	Nº de canal de diagnóstico	Nº de canal d'état du diagnostic	N. canale stato diagnostico	诊断状态通道号
0034		DiagStateAddValue	Diagnosestatus-Zusatzwert	Diagnostic State Additional Value	Valor adicional del estado de diagnóstico	Valeur supplémentaire d'état du diagnostic	Valore aggiuntivo stato diagnostico	诊断状态附加值
0035		NoOfModules	Anzahl der Submodule	Number of modules	Cantidad de submódulos	Nombre de sous-modules	Numero di moduli subordinati	子模块数
0036		DeviceStructure	Gerätestruktur	Device Structure	Estructura del dispositivo	Structure d'appareil	Struttura dispositivo	设备结构
0037		DeviceType	Gerätetyp	Device Type	Tipo de dispositivo	Type d'appareil	Tipo di dispositivo	设备类型
0038		ObjDescrReq	Anfrage Objektbeschreibung	Object Description Request	Solicitud de descripción del objeto	Requête de description d'objet	Richiesta descrizione oggetto	对象描述询问
0039		ObjDescr	Objektbeschreibung	Object Description	Descripción del objeto	Description d'objet	Descrizione oggetto	对象描述
003A		VersionCount	Versionszähler	Version Counter	Contador de versiones	Compteur de version	Conteggio versione	□□□□□
	.1	ProfileVersion	Versionszähler - Profil-Version	Version Counter - Profile Version	Contador de versiones - Versión del perfil	Compteur de version - Version du profil	Contatore versioni - Versione profilo	□□□□□ - □□□□
	.2	PCPVersion	Versionszähler - Parameterkanal-Version	Version Counter - PCP Version	Contador de versiones - Versión del canal de	Compteur de version - Version du canal de paramètres	Contatore versioni - Versione canale parametri	□□□□□ - □□□□□□

Index	Sub-Index	Objektname (Basisprofil)	DE	EN-US	ES	FR	IT	ZH-CN
					parámetros			
	.3	HardwareVersion	Versionszähler - Hardware-Version	Version Counter - Hardware Version	Contador de versiones - Versión del hardware	Compteur de version - Version du matériel	Contatore versioni - Versione hardware	□□□□□ - □□□□
	.4	FirmwareVersion	Versionszähler - Firmware-Version	Version Counter - Firmware Version	Contador de versiones - Versión del firmware	Compteur de version - Version du microprogramme	Contatore versioni - Versione firmware	□□□□□ - □□□□
003B		PDIN_Descr	Prozesseingangsdatenbeschreibung	Process Input Data Description	Descripción de datos de proceso de entrada	Description des données de process entrantes	Descrizione dati in ingresso processo	□□□□□□□□
003C		PDOOUT_Descr	Prozessausgangsdatenbeschreibung	Process Output Data Description	Descripción de datos de proceso de salida	Description des données de process sortantes	Descrizione dati in uscita processo	□□□□□□□□
E800		DiagStateLong	Diagnosestatus (Langform)	Diagnostic State (Long Form)	Estado de diagnóstico (de forma detallada)	État du diagnostic (forme longue)	Stato diagnostico (forma estesa)	□□□□(□□)
E801		DiagHistory	Diagnoseinformationen	Diagnostic History	Datos de diagnóstico	Informations de diagnostic	Informazioni diagnostiche	□□□□
E802		DiagHistoryLong	Diagnoseinformationen (Langform)	Diagnostic History (Long Form)	Datos de diagnóstico (de forma detallada)	Informations de diagnostic (forme longue)	Informazioni diagnostiche (forma estesa)	□□□□(□□)
E803		TraceBuffer	Puffer für Fehlerprotokollierung	Trace Buffer	Búfer para registro de errores	Tampon pour le journal des erreurs	Buffer per protocollo errori	□□□□□□□□
E804		LanguageAvailable	Landessprache	Language	Idioma nacional	Langue	Lingua nazionale	□□
E805		ObjDescrLong	Objektbeschreibung (Langform)	Object Description (Long Form)	Descripción del objeto (de forma detallada)	Description d'objet (forme longue)	Descrizione oggetto (forma estesa)	□□□□(□□)

▪
▪
▪
▪
- Ende -