


Fieldbus Appendix

Anybus-S PROFIBUS DP-V1

Rev 1.02

HMS Industrial Networks AB


Germany +49 - 721 - 96472 - 0
Japan +81 - 45 - 478 - 5340
Sweden +46 - 35 - 17 29 20
U.S.A +1 - 773 - 404 - 3486
UK +44 (0) 1908 - 359301


ge-sales@hms-networks.com
jp-sales@hms-networks.com
sales@hms-networks.com
us-sales@hms-networks.com
uk-sales@hms-networks.com



Table of Contents

Preface	About This Manual
	How To Use This Manual..... P-1
	Important user information..... P-1
	Related Documentation..... P-1
	Revision list P-1
	Conventions used in this manual P-2
	Support..... P-2
 Chapter 1	 About the Anybus-S PROFIBUS DP-V1
	Features1-1
	Compatible Products1-1
	GSD-File.....1-1
	Certification issues.....1-1
	Connectors, Indicators & Switches1-3
	<i>Connectors</i>1-3
	<i>Switches</i>1-3
	<i>Status Indicators</i>1-4
 Chapter 2	 Network Configuration
 Chapter 3	 Software Overview
	General.....3-1
	DP-V1 Functionality.....3-2
	<i>MSAC1/MSAC2_Read, MSAC1/MSAC2_Write and MSAC2_DataTransport</i>3-3
	<i>MSAC2_Initiate</i>3-4
	<i>MSAC2_Abort</i>3-5
	Set_Parameter telegram3-6
	<i>Structure</i>3-6
	<i>Verification of User Parameter Data</i>3-6
	Check_Configuration telegram3-10
	<i>General</i>3-10
	<i>Structure of the Identifier byte</i>3-10
	<i>Verification of Configuration data</i>3-10

Chapter 4 Mailbox Interface

Mailbox Commands initiated by the application.....	4-1
Mailbox Messages initiated by the module.....	4-1
Initialisation / Configuration.....	4-2
<i>FB_INIT</i>	4-2
<i>FB_APPL_SET_NODE_ADDRESS</i>	4-7
<i>FB_APPL_GET_NODE_ADDRESS</i>	4-8
Parameter Data.....	4-9
<i>FB_ABS_VERIFY_USER_PRM</i>	4-9
<i>FB_APPL_GET_USER_PRM</i>	4-10
<i>FB_APPL_SET_USER_PRM</i>	4-12
Configuration Data.....	4-14
<i>FB_ABS_VERIFY_CONFIG</i>	4-14
<i>FB_APPL_GET_CONFIG</i>	4-15
<i>FB_APPL_SET_CONFIG</i>	4-17
Diagnostics.....	4-19
<i>FB_APPL_WRITE_DIAGNOSTICS</i>	4-19
Acyclic Data Functions.....	4-21
<i>General</i>	4-21
<i>Error Codes</i>	4-22
<i>FB_ABS_ACYCLIC_WRITE</i>	4-23
<i>FB_ABS_ACYCLIC_READ</i>	4-25
<i>FB_ABS_ACYCLIC_DATATRANSPORT</i>	4-27
<i>FB_ABS_ACYCLIC_INITLATE</i>	4-29
<i>FB_ABS_ACYCLIC_ABORT_IND</i>	4-33
<i>FB_APPL_ACYCLIC_ABORT</i>	4-34
<i>FB_APPL_MAP_PARAMETER_OBJECT</i>	4-36

Chapter 5 Fieldbus Specific Area

Memory Map.....	5-1
-----------------	-----

Appendix A Environmental Specification

Temperature.....	A-1
Relative Humidity.....	A-1
EMC compliance.....	A-1

Appendix B Connectors

Appendix C Electrical Characteristics

Supply Voltage.....	C-1
Maximum Current Consumption.....	C-1
PE Grounding.....	C-1

Appendix D Mechanical Specification

Measurements, PCB D-1

Measurements, Connectors & Switches..... D-2

Straight Switches & Connectors D-2

Angled Switches & Connectors D-3

About This Manual

How To Use This Manual

This manual provides an overview of the ABS-DPV1. It describes how to configure and operate the module, and provides examples showing how to use its features over a PROFIBUS network.

This document is intended as a complement to the Anybus-S Parallel Design Guide. For general information about the Anybus-S, concerning mechanical and electrical specifications etc., consult the general Anybus-S Slave & Master Parallel Interface Design Guide.

The reader of this document is expected to be familiar with hardware and software design as well as to have basic knowledge in the PROFIBUS DP fieldbus system, and communication systems in general.

Important user information

The data and illustrations found in this document are not binding. We, HMS Industrial Networks AB, reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be considered as a commitment by HMS Industrial Networks AB. HMS Industrial Networks AB assumes no responsibility for any errors that may appear in this document.

There are many applications of this product. Those responsible for the use of this device must ensure that all the necessary steps have been taken to verify that the application meets all performance and safety requirements including any applicable laws, regulations, codes, and standards.

Anybus® is a registered trademark of HMS Industrial Networks AB. All other trademarks are the property of their respective holders.

Related Documentation

Document name	Author
Anybus-S Parallel Design Guide	HMS
PROFIBUS Specification IEC 61158	IEC
GSD-Specification (GSD Revision 3)	PNO

Revision list

Revision	Date	Author	Chapter	Description
1.00	2002-07-23	PeP	All	First release
1.01	2003-10-10	ToT	FB_APPL_SET_NODE_ADDRESS, FB_APPL_GET_NODE_ADDRESS, FB_ABS_VERIFY_CONFIG, FB_APPL_WRITE_DIAGNOSTICS, FB_ABS_ACYCLIC_ABORT_IND, FB_APPL_MAP_PARAMETER_OBJECT	Corrected erroneous mailbox messages
1.02	2005-12-01	ToT	All except Fieldbus Specific Area, Environmental Specification, Electrical Characteristics and Mechanical Specification.	Various corrections and additions

Conventions used in this manual

The following conventions are used throughout this manual:

- Numbered lists provide sequential steps
- Bulleted lists provide information, not procedural steps
- Mailbox commands that must to be sent “during module initialization” must be sent between the “START_INIT” and “END_INIT” commands.
- The term ‘module’ is used when referring to the ABS-DPV1
- The term ‘application’ is used when referring to the hardware that is connected to the Anybus Application Connector.
- Hexadecimal values are written in the format NNNNh or 0xNNNN, where NNNN is the hexadecimal value.
- All pictures in this manual shows the standard version of this product. However, other connectors, leds and switches may be present depending on configuration.

Support

HMS Sweden (Head Office)

E-mail:	support@hms-networks.com
Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com

HMS America

E-mail:	us-support@hms-networks.com
Phone:	+1-773-404-2271
Toll Free:	888-8-Anybus
Fax:	+1-773-404-1797
Online:	www.anybus.com

HMS Germany

E-mail:	ge-support@hms-networks.com
Phone:	+49-721-96472-0
Fax:	+49-721-964-7210
Online:	www.anybus.com

HMS Japan

E-mail:	jp-support@hms-networks.com
Phone:	+81-45-478-5340
Fax:	+81-45-476-0315
Online:	www.anybus.com

About the Anybus-S PROFIBUS DP-V1

The Anybus-S PROFIBUS DP-V1 is a communication module that is designed to communicate over a PROFIBUS DP-V1 network. It contains class 1 and class 2 services as well as the standard DP functionality.

The module is available only with the 2k Byte parallel Dual Port Ram interface.

Features

- PROFIBUS DP-V1 Slave functionality
- Supports Class 1 & Class 2 DP-V1 services
- Automatic baudrate detection (9600 bit/s - 12 Mbit/s)
- Supports PA baud rate 45.45 kbit/s
- Optically isolated PROFIBUS interface with on-board DC/DC converter.
- Save/Load configuration in Flash
- Onboard node address configuration and network termination switches
- Address range: 0-99 using onboard switches or 0-126 via the application interface
- Input/Output/User Parameter data/Diagnostics format defined via application interface
- Cyclic I/O data size up to 244 bytes in/244 bytes out, max 328 bytes total in standard setup
- User Parameter data/Diagnostics length - up to 237 bytes¹
- LED-indications: ON-line, OFF-line, Fieldbus related diagnostics

Compatible Products

This product is a member of the Anybus concept of interchangeable fieldbus modules. This makes it fully interchangeable with any Anybus-S supported fieldbus system.

GSD-File

Each device on a PROFIBUS DP network is associated with a GSD file, containing all necessary information about the device. This file is used by the network configuration program during configuration of the network.

The latest version of the GSD file for this product is available at our web site.

Certification issues

If the module is initialised as a standard HMS module (not using a fieldbus specific initialisation) no GSD-file changes needs to be made, nor is allowed, in order to pass a certification test.

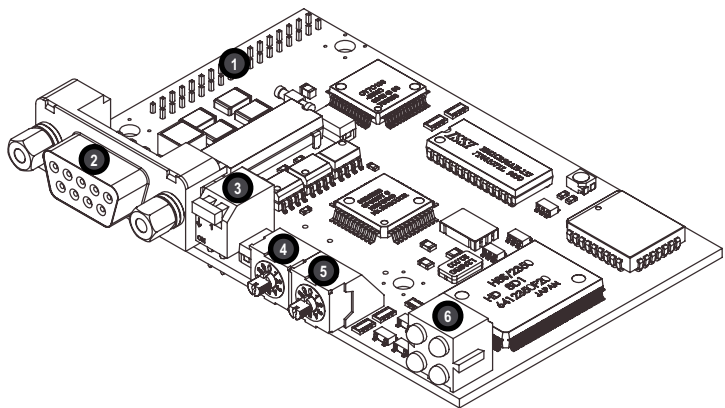
If a fieldbus specific initialisation is used in order to enable any of the PROFIBUS features normally disabled (such as user parameter data or diagnostic data) the GSD-file needs to be changed in order to

1. If the extended functions User Parameter data and Diagnostics are used the max.total I/O length is less, usage of these functions requires the FB_INIT mailbox message.

match the new properties of the module. If it is changed it is also necessary to acquire an own ID number that should be used both in the file and during the module initialisation.

If the configuration data usage in either the module or in the GSD-file needs to be changed it is also important to remember that all possible configuration data combinations that other nodes can read out of the module using the 'Get Config' command must also be possible to reconstruct with the configuration data available in the GSD-file that the module will use. If this is not possible, the certification will fail.

Connectors, Indicators & Switches



#	Part	Description
1	Application Connector	Anybus Application connector
2	Fieldbus Connector	Fieldbus Connector
3	Termination Switch	Termination Switch
4	Node Address (x10)	Node address switches
5	Node Address (x1)	
6	Status Indicators	Status Indicators

Connectors

Application Connector

Standard Anybus-S 2kbyte parallel Dual Port RAM interface. For more information about the Anybus-S application connector, please consult the general Anybus-S Parallel Design Guide.

Fieldbus Connector

The module can be fitted with several different types of connectors. The standard connector is a 9 pin Dsub connector. This connector is strongly recommended if the module is to be used at transfer rates above 1500kbit/s. This connector is also recommended by the PROFIBUS DP standard IEC 61158.

Depending on the protection class and type of application, other connectors are also supported.

For more information about the connectors and their pin assignments, see B-1 “Connectors”.

Switches

Termination Switch

Disable / enable bus termination. See 2-1 “Network Configuration”.

Node Address Switches

These switches are used when configuring the node address in the range 00-99. See 2-1 “Network Configuration”

Status Indicators

The module features four status leds and an additional Watchdog led.

The function of the Watchdog led is equal for all Anybus-S modules, consult the Anybus-S Design Guide for more information.



LED 1 - Acyclic Traffic

This led indicates that a DP-V1 service is currently being executed.

Colour	State	Indicates:
Green	On	A DP-V1 request is currently being executed
-	Off	No power on the module No DP-V1 request is currently being executed

LED 2 - Fieldbus On-line

This led indicates if the module is online on the fieldbus or not.

Colour	State	Indicates:
Green	On	Bus is online and data exchange is possible
	Flashing, 1 Hz	Clear mode
Red	On	Application stopped
-	Off	Bus is not online or no power on the module

LED 3 - Fieldbus Off-line

This led indicates if the module is off-line and is thus the opposite compared to led 2.

Colour	State	Indicates:
Red	On	Bus is offline
-	Off	Bus is not offline or no power on the module

LED 4 - Fieldbus Diagnostics

This led indicates certain faults on the fieldbus side.

Colour	State	Indicates:
-	Off	No diagnostics present. No power on the module.
	Flashing, 1 Hz	Error in configuration data
Red	Flashing, 2 Hz	Error in parameter data
	Flashing, 4 Hz	Error in initialisation of the PROFIBUS communication ASIC

Network Configuration

Before the module can be used on a PROFIBUS DP network some basic settings must be configured.

Baudrate

The baudrate on a PROFIBUS DP network is set during configuration from the master. The module features auto baudrate detection and the user does not have to configure the baudrate.

Supported baudrates:

1. 9.6 kbit/s
2. 19.2 kbit/s
3. 45.45 kbit/s
4. 93.75 kbit/s
5. 187.5 kbit/s
6. 500 kbit/s
7. 1.5 Mbit/s
8. 3 Mbit/s
9. 6 Mbit/s
10. 12 Mbit/s

Node Address

The node address can be configured either using the on board switches, or via mailbox commands. The switches can be used when selecting a node address in the range 00-99.

Address 100-126 is available via the mailbox command FB_APPL_SET_NODE_ADDRESS, see 4-7 “FB_APPL_SET_NODE_ADDRESS”.

Note: Node address 126 is reserved for commissioning purposes and shall not be used for data exchange.

Termination

Each bus segment in a PROFIBUS network must be terminated properly to ensure error-free operation.

If the module is used as the first or last node in a network segment, the termination switch has to be in ON position. Otherwise the switch has to be in OFF position.

Note: If an external termination connector is used, the switch must be in OFF position.

Software Overview

General

I/O Sizes

Max I/O length when ANYBUS_INIT is used: 328 bytes

Max I/O length when ANYBUS_INIT and FB_INIT is used: 392 bytes

Global Control Commands

The module supports the following global commands:

- **FREEZE/UNFREEZE of inputs**
Works according to the PROFIBUS DP specification
- **SYNC/UNSYNC of outputs**
Works according to the PROFIBUS DP specification
- **CLEAR**
How the module will handle the OUT-area when receiving a CLEAR command is configured in the FB_INIT command, message data word 5, bit 14-15. (See 4-2 “FB_INIT”)

Data Exchange

The master will read/write data every bus cycle. The amount of data copied each bus cycle is determined in the configuration phase of the fieldbus initialisation. The data will be exchanged via the I/O areas.

Application stopped

The application may enable the watchdog function in the module during initialisation. (Refer to the ANYBUS_INIT telegram in the Anybus-S Parallel Design Guide)

If this function is enabled and the application stops updating the watchdog register (application stopped) the bus will go off-line after the defined watchdog time and the on-line LED on the module will turn red.

This situation is considered fatal; there is no way to recover from it besides power cycling the module.

DP-V1 Functionality

The DP-V1 extension enables acyclic read/write operations between a Class 1/2 master and a DP-V1 slave in parallel with the cyclic user data communication.

Supported Class 1 Services

The following Class 1 services are available:

- **MSAC1_Read**
The master reads a data block from the module.
- **MSAC1_Write**
The master writes a data block to the module
- **Status diagnostics**
Transmission of a status diagnostic message from the module to the master. The master does not acknowledge the receipt of the status message; therefore these messages can be overwritten.

Supported Class 2 Services

The following Class 2 services are available:

- **MSAC2_Initiate**
Establishment of a connection for acyclic data communication
- **MSAC_Abort**
Termination of a connection for acyclic data communication
- **MSAC2_Read**
The master reads a data block from the module
- **MSAC2_Write**
The master writes a data block from the module.
- **MSAC2_Data_Transport**
With this service the master both reads and writes data from/to the module during the same service cycle.

Addressing

The acyclic read/write services defined in DP-V1 are slot- and index related. This means that Slot number and Index are used for addressing a desired data block.

It is possible for the application to configure if the data block should be accessed from the parameter area (DPRAM or Internal RAM) of the module *or* from the application.

If the module is configured to access data blocks from the application, the request from the master will be forwarded to the application via a so called spontaneous mailbox telegram. The application then has to create a mailbox response and send it back to the module.

The maximum response time for a C2 read/write request is specified in the GSD-file. If this time has elapsed and the master still has not received an answer from the module (application), a time-out will occur in the master and the connection will be terminated.

The module can handle up to 2 simultaneous active C2 connections.

MSAC1/MSAC2_Read, MSAC1/MSAC2_Write and MSAC2_DataTransport

The application can access data blocks (objects) in three different ways depending on how the objects are mapped.

1. Default object mapping
2. Application specific object mapping
3. Transparent object mapping (Only possible if the module is initialised with FB_INIT)

In case 1 and 2, objects are located in the internal parameter area (DPRAM or/and RAM) of the module. In case 3, objects are located at the application side.

Default Object Mapping

By default the parameter area in the module is divided into data blocks of 32 bytes, where each data block is associated with a predefined Slot number. Index is used to address a byte position within the data block.

Byte	Parameter Area	Slot	Index
0	Object 0 (32 bytes)	0	0
1		0	1
2		0	2
..	
31		0	31
32	Object 1 (32 bytes)	1	0
33		1	1
34		1	2
..	
63		1	31
X	Object Y (32 bytes)	Y	0
X+1		Y	1
X+2		Y	2
...	
X+31		Y	31

Example:

Byte 0-31 in the parameter area is associated with Slot number 0; Byte 32-63 is associated with Slot number 1, and so on.

If the DP-V1-master would like to access byte 34 in the parameter area it sends a read or write request where Slot number =1 and Index =2.

Application Specific Object Mapping

In this case the application “connects” a Slot Number and an Index to an object that is located at a certain position in the parameter area (DPRAM or/and RAM). When the master sends a Read, Write or a DataTransport request with this Slot Number and Index, it will gain access to the object.

Mapping of objects is performed with the mailbox command

FB_APPL_MAP_PARAMETER_OBJECT. (See 4-36 “FB_APPL_MAP_PARAMETER_OBJECT”)

Maximum number of objects that can be mapped is 512.

Example:

A read object (Object 1) that is 4 bytes long is supposed to be mapped to byte offset 3 in the parameter area. The master should get access to this object when sending a Read request with Slot number = 5 and Index = 10.

The parameters that go with the mailbox telegram FB_APPL_MAP_PARAMETER_OBJECT would be:

```
Slot number = 5
Index = 10
Length = 4
Offset = 3
Object type = Read
```

Byte	Parameter Area
0	
1	
2	
3	Object 1 (Length = 4 bytes)
4	
5	
6	
7	
...	
x	

Transparent Object Mapping

In this mode, a PROFIBUS acyclic Read-/Write-/DataTransport telegram is handed over to the application via the mailbox messages FB_ABS_ACYCLIC_WRITE, FB_ABS_ACYCLIC_READ or FB_ABS_ACYCLIC_DATA_TRANSPORT. The application then has to handle the request and respond to the module when it has finished processing the data.

This functionality is only available if bit 5 in the Mode selection word (Message data word 5 in FB_INIT) is set to 1 (i.e. Transparent Object handling enabled)

MSAC2_Initiate

Before the DP-V1 master can perform a MSAC2_Read, MSAC2_Write or MSAC2_Data_Transport it must open up a connection against the module. The module can handle the MSAC2_Initiate request in one of two ways:

- The module handles the request, and sends the response to the master (Default)
- The module sends the request to the application via a spontaneous mailbox message. The application evaluates the request and creates a response, which the module forwards to the master. (This is mode is only possible if the module is initialised using FB_INIT)

Anybus handles Initiate request

The module will run in this mode if it does not receive a FB_INIT command during initialisation or receives a FB_INIT command where bit 4 in the Mode selection word is cleared (i.e Anybus handles C2_Initiate)

Note: The module will reject the request if D-Type = 1 in the address data parameter (i.e. Anybus is not the endpoint of the connection)

Application handles Initiate request

The module will run in this mode if it has been initialised with FB_INIT where bit 4 in the Mode selection word is set (i.e Application handles C2_Initiate).

In this mode, the C2-Initiate telegram will be handed over to the application via the mailbox message FB_ABS_ACYCLIC_INITIATE. The application then has to handle the request and send a response back to the module when it has finished the evaluation.

If the request is accepted, the application should clear Extended word 8.

MSAC2_Abort

The DP-V1 master or DP-V1 slave are using the service MSAC2_Abort to abort a MSAC2-connection. This is handled in two different ways, depending on how the module has been initialised.

- **Master or Anybus is initiating MSAC2_Abort**
- **Application is initiating a MSAC2_Abort**

Master or Anybus is initiating MSAC2_Abort

If bit 6 in the Mode selection word (See FB_INIT 4-2 “FB_INIT”) is set (1), the MSAC2_Abort telegram is handed over to the application via the mailbox message FB_ABS_ACYCLIC_ABORT_IND so that the application can determine the reason for the abortion.

The module will take care of closing the actual connection. (i.e. cancel all pending request and clean up allocated memory).

Note 1: The application must *not* respond to FB_ABS_ACYCLIC_ABORT_IND.

Note 2: In firmware revisions below 1.20 the module will initiate a MSAC2_Abort if the application does not respond to a read, write or init request within the timeout time specified in FB_INIT. Firmware revisions 1.20 and above does not have any timeout against the application, only the master or the application itself can initiate a MSAC2_Abort.

Application is initiating a MSAC2_Abort

If bit 6 in the Mode selection word is clear (0), the application can cancel existing C2-connections using the mailbox command FB_APPL_ACYCLIC_ABORT

Set_Parameter telegram

The master uses this telegram to identify itself with the module and to specify how the module should operate. The message is comprised of a series of octets (max 244). For the module to operate in DP-V1 mode it has to be ensured that after the first seven bytes (specified in the DP-standard), three DP-V1 status bytes are provided and that bit 7 in DP-V1 Status_byte_1 is set.

Structure

The Parameter Data consists of the following information:

Octet	Value	Description
0	-	Status byte
1	0-255	Watchdog Factor 1
2	0-255	Watchdog Factor 2
3	0-255	Minimum station delay (Min T_{SDR})
4	4	PROFIBUS Identification number (High byte)
5	239	PROFIBUS Identification number (Low byte)
6	-	Group identity
7	-	DP-V1 Status 1
8	-	DP-V1 Status 2
9	-	DP-V1 Status 3 ^a
10	-	(Reserved for future use)
11 - 243	-	Optional User Parameter Data (The meaning of these bytes is defined by the application)

a. This byte will be ignored since the module does not support the Alarm functionality

Verification of User Parameter Data

The User Parameter Data can be evaluated in one of three ways:

1. The module always accepts the User Parameter Data
2. The module checks the User Parameter Data
3. The application checks the User Parameter Data

Notes: If the module receives a Set_Prm message and its internal parameter buffer (determined with FB_INIT) is not big enough to hold the message, the module will reject the message and not proceed with the User Parameter data check.

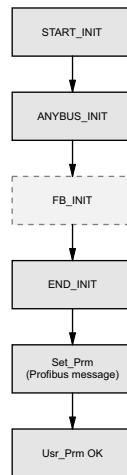
Evaluation of DP-V1 status bytes is done by the module exclusively.

Case 1: Anybus module always accepts the User Parameter Data

In this mode the module will accept the Set_Prm telegram as long as the number of parameter bytes is not more than the parameter length that were set using FB_INIT. Otherwise, the module will set the PrmFault bit in the Slave_Diag response.

The module will run in this mode if it is:

- Initialised with just ANYBUS_INIT (FB_INIT is not sent)
or
- Initialised with FB_INIT where bits 0-1 in the Mode selection word (Message data word 5 in FB_INIT) are set to 00 - “No verification of User Parameter data”.



Case 2: Anybus verifies the User Parameter Data

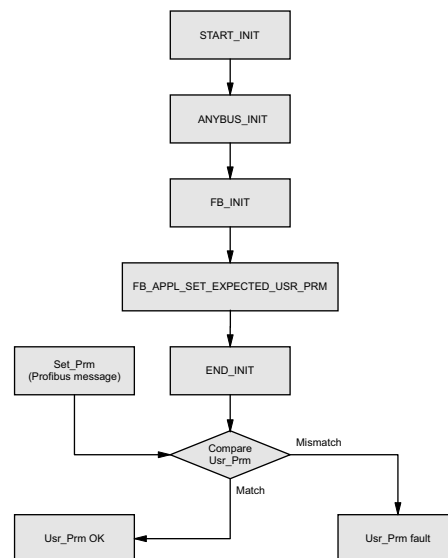
In this mode the application writes its expected user parameter data to the module with the mailbox command FB_APPL_SET_USER_PRM. When the Set_Prm is received from the master the module will check that the expected user parameter data matches the parameter data received from the master. If the check is OK, the module will send an acknowledgement telling the master that the configuration is OK. If the check is not OK, the module will set the PrmFault bit in the Slave_Diag response.

The first four bytes in the User Parameter Data received from the master are not included in the check since these bytes are reserved for the module exclusively.

Note: If the module does not receive FB_APPL_SET_USER_PRM before END_INIT it will assume that the expected User Parameter data is all zeros.

The module will run in this mode if:

- Bit 0-1 in the Mode selection word (Message data word 5 in FB_INIT) is set to 01 - “Anybus verifies User Parameter Data”

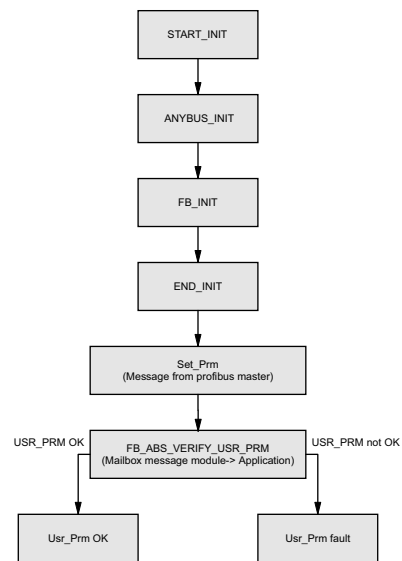


Case 3: Application verifies the User Parameter Data

When the Set_Prm telegram is received from the master the module will pass the User Parameter data over to the application via the FBS_ABS_VERIFY_USER_PRM mailbox command. The application then has to check the User Parameter data and reply back to the module if it was OK or not. If the application accepts the User Parameter data the module will send an acknowledgement telling the master that the User Parameter data is OK. If the application does not accept the User Parameter data, the module will set the PrmFault bit in the Slave_Diag response.

The module will run in this mode if:

- Bit 0-1 in the Mode selection word (message data word 5 in FB_INIT) is set to 10 - "Application verifies User Parameter Data".



Check_Configuration telegram

General

After the Set_Parameter telegram, the master sends a Check_Configuration telegram to the slave. This event causes the slave to check its default configuration, determined at power-up against the actual configuration, specified in the master configuration. The module can be configured to do this check in one of three different ways. (3-10 “Verification of Configuration data”)

The module is a modular slave, which means that the size of the input/output block size is flexible and not tied to a specific block length. The input/output block is described by so-called Identifier bytes. Such a byte defines the input- and/or output data length as well as the data consistency.

The Identifier bytes are defined in the GSD-file and are automatically extracted by the configuration tool when you add the modules to your configuration.

Structure of the Identifier byte

The format of the Identifier byte is briefly described below. Detailed information about the format can be found in the PROFIBUS DP standard.

bit(s)	Contents
0 - 3	Length of data. 0h = 1 byte / word Fh = 16 bytes / words
4 - 5	Input / Output 00 = special format 01 = input 10 = output 11 = input and output
6	0 = byte 1 = word
7	0 = Consistency of byte / word 1 = Consistency of entire length

Verification of Configuration data

The module can evaluate the Chk_Cfg message in one of three ways:

1. The module will always accept the configuration.
2. The module checks the configuration
3. The application checks the configuration

Note: If the module receives a Chk_Cfg message and it's internal configuration buffer (Determined with FB_INIT) is not big enough to hold the associated identifier bytes, the module will reject the message and not proceed with the configuration check. This will also be the case if the I/O lengths extracted from the identifier bytes do not fit into the internal I/O buffer (determined with ANYBUS_INIT).

Case 1: The module will always accept the configuration data

In this mode the module will accept the Chk_Cfg telegram as long as the I/O length (extracted from the identifier bytes in the telegram) is not bigger than the I/O lengths that were set with ANYBUS_INIT.

If the I/O lengths are bigger, the module will set the ConfigFault bit in the Slave_Diag response.

The module will run in this mode if:

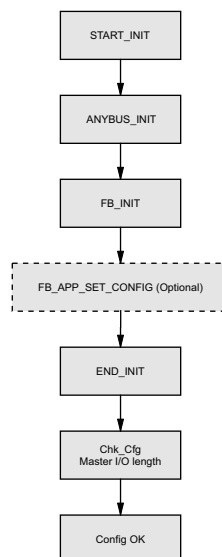
- Initialised with just ANYBUS_INIT (FB_INIT is not sent)
- Initialised with FB_INIT and bits 2-3 in the Mode selection word (Message data word 5 in FB_INIT) are set to 00 - “No verification of Configuration data”.

Default configuration:

- If FB_APPL_SET_CONFIG has been sent; the identifier bytes in the command will be used as default configuration for the PROFIBUS telegram Get_Cfg.

Note: If the standard GSD-file (HMS_1013.GSD) is used, the user must ensure that the default configuration can be found in the standard GSD-file.

- If FB_APPL_SET_CONFIG has not been sent; the module will extract the default configuration based on the I/O lengths specified with ANYBUS_INIT, the default configuration in this case will always be word based with consistency over word.



Case 2: The module verifies configuration data

In this mode the application writes the expected configuration to the module using the mailbox command FB_APPL_SET_CONFIG.

When the ChkCfg telegram is received from the master the module will first check that the number of identifier bytes fits into the allocated configuration buffer (set with FB_INIT). If this is the case, the module will extract the I/O lengths from the identifier bytes and verify that the lengths fit into the allocated I/O buffer (set with ANYBUS_INIT). If these two verifications are OK the module will continue with the configuration check described below. If not, the module will set the CfgFault bit in the Slave_Diag response and the configuration check will not be performed.

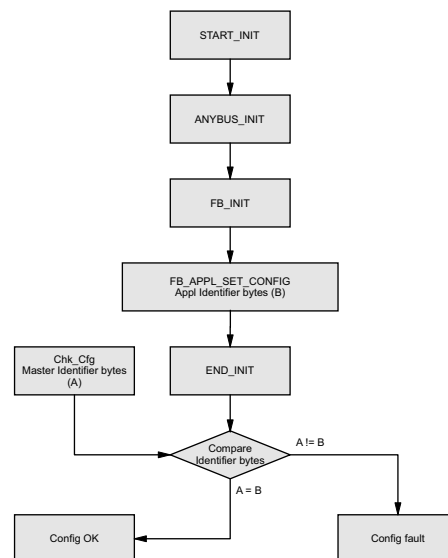
The module will check the expected configuration against the actual master configuration. If the expected configuration matches the actual one, the module will send an acknowledgement telling the master that the configuration is OK. If the configuration does not match, the module will set the CfgFault bit in the Slave_Diag response.

The module will run in this mode if:

- Bits 2-3 in the Mode selection word (Message data word 5 in FB_INIT) are set to 01 - “Anybus Verifies Configuration Data”.

Default configuration:

- The identifier bytes in the mailbox will be used as default configuration for the PROFIBUS telegram Get_Cfg.



Case 3: Application verifies the configuration data

When the ChkCfg telegram is received from the master the module will first check that the number of identifier bytes fits into the allocated configuration buffer (Set with FB_INIT). If this is the case, the module will extract the I/O lengths from the identifier bytes and verify that the lengths fit into the allocated I/O buffer (Set with ANYBUS_INIT). If these two verifications are OK the module will pass the identifier bytes over to the application via the FB_ABS_VERIFY_CONFIG mailbox telegram. If not, the module will set the CfgFault bit in the Slave_Diag response and it will not send the mailbox to the application for further verification.

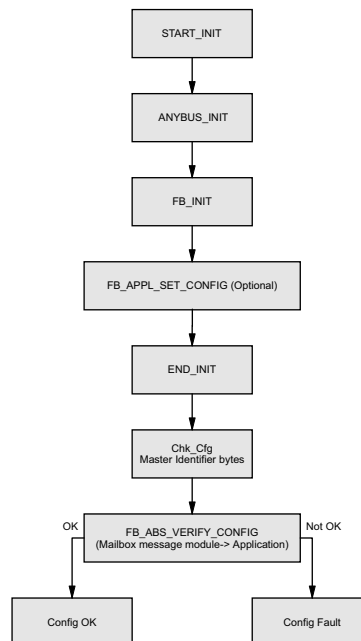
When the application receives this mailbox, it has to check the identifier bytes and respond to the module if the configuration is OK or not. If the application accepts the configuration the module will send an acknowledgement telling the master that the configuration is OK. If the application does not accept the configuration the module will set the CfgFault bit in the Slave_Diag response.

The module will run in this mode if:

- Bits 2-3 in the Mode selection word (Message data word 5 in FB_INIT) are set to 10 - “Application verifies Configuration data”.

Default Configuration:

- If FB_APPL_SET_CONFIG is received; the identifier bytes in the mailbox will be used as default configuration for the PROFIBUS telegram Get_Cfg.
- If FB_APPL_SET_CONFIG is not received; The module will extract the default configuration based on the I/O lengths that are specified in ANYBUS_INIT, the default configuration in this case will always be word based with consistency over word.



Mailbox Interface

This chapter describes the fieldbus specific mailbox commands supported by the module. Consult the Anybus-S Design Guide for more information regarding mailbox functionality.

Mailbox Commands initiated by the application

Name	Description
FB_INIT	This mailbox command shall be sent to the module during the initialisation sequence and contains information required to initialise the module for PROFIBUS specific functionalities.
FB_APPL_GET_USER_PRM	This command is used for reading the actual User Parameter data
FB_APPL_WRITE_DIAGNOSTICS	This command is used for sending diagnostic information to the PROFIBUS master.
FB_APPL_SET_NODE_ADDRESS	This command is used for setting the node address.
FB_APPL_SET_USER_PRM	This mailbox is used for sending expected User Parameter data to the module to evaluate
FB_APPL_GET_NODE_ADDRESS	This command is used for reading the currently used node address
FB_APPL_SET_CONFIG	This mailbox is used for sending expected configuration data for the module to evaluate.
FB_APPL_GET_CONFIG	This mailbox is used for reading out the configuration data.
FB_APPL_MAP_PARAMETER_OBJECT	This mailbox is used for mapping DP-V1 objects to the acyclic parameter area
FB_APPL_ACYCLIC_ABORT	This command terminates a DP-V1 class 2 connection.

Mailbox Messages initiated by the module

Name	Description
FB_ABS_VERIFY_USER_PRM	This message is sent to the application for evaluation of the User Parameter data.
FB_ABS_VERIFY_CONFIG	This message is sent to the application for evaluation of the configuration data
FB_ABS_ACYCLIC_WRITE	If the application has initialized the module to handle acyclic read/write requests in a transparent way, this mailbox will be send to the application whenever the module receives a DP-V1 write request from the master.
FB_ABS_ACYCLIC_READ	If the application has initialized the module to handle acyclic read/write requests in a transparent way, this mailbox will be sent to the application whenever the module receives a DP-V1 read request from the master.
FB_ABS_ACYCLIC_INITIATE	The DP-V1 C2-Initiate telegram will be sent to the application for verification if the module is configured to do so.
FB_ABS_ACYCLIC_ABORT_IND	The DP-V1 C2-Abort telegram will be send to the application if the module is configured to do so.
FB_ABS_ACYCLIC_DATA_TRANSPORT	If the application has initialized the module to handle acyclic read/write requests in a transparent way, this mailbox will be send to the application whenever the module receives a DP-V1 data transport request from the master.

Initialisation / Configuration

FB_INIT

Description

This command contains information required to initialise the module for PROFIBUS specific functionalities.

Note: This command can only be sent after ANYBUS_INIT, during module initialisation.

Parameter	Description
Command initiator	Application
Command Name	FB_INIT
Message type	02h
Command number	0001h
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: Contains information required for PROFIBUS specific functionalities. Response: The command message data is returned
Response message	The response data is a copy of the command data.

Command and response layout:

	Command	Expected response
Message ID	(ID)	(ID)
Message information	4002h	0002h
Command	0001h	0001h
Data size	0010h	0010h
Frame count	0001h	0001h
Frame number	0001h	0001h
Offset high	0000h	0000h
Offset low	0000h	0000h
Extended word 1	-	-
Extended word 2	-	-
Extended word 3	-	-
Extended word 4	-	-
Extended word 5	-	-
Extended word 6	-	-
Extended word 7	-	-
Extended word 8	-	Fault information
Message data word 1	Ident number	Ident number
Message data word 2	User parameter length	User parameter length
Message data word 3	Diagnostic length	Diagnostic length
Message data word 4	Configuration length	Configuration length
Message data word 5	Mode selection	Mode selection
Message data word 6	C1-length	C1-length
Message data word 7	C2-length	C2-length
Message data word 8	(see notes)	(see notes)

FB_INIT

Ident number

The Ident Number is an application specific number assigned by the PNO (PROFIBUS User Organisation)

User Parameter length (in bytes)

Maximum amount of User Parameter data for the application. If this parameter is set to 0 (zero) the user parameter data functionality will be disabled. Valid settings range from 0 to 233. The module will also round the given total Parameter length up to the nearest multiple of 8 bytes.

User Parameter length = 0

The module will allocate 11 bytes of parameter data (7 bytes according to the DP-standard + 3 DP-V1 status bytes + 1 Anybus status byte)

User Parameter length > 0

The module will allocate 11 bytes described above + the number of bytes specified in the User Parameter length.

Diagnostic length (in bytes)

Maximum size of the Diagnostic Block structure for the application. If this parameter is set to 0 (zero) the diagnostic functionality will be disabled. Valid settings range from 0 to 238. The module will also round the given Diagnostic length up to the nearest multiple of 8 bytes.

Diagnostic length = 0

The module will allocate 6 bytes of diagnostic data according to the DP-standard

Diagnostic length > 0

The module will allocate 6 bytes of diagnostic data according to the DP-standard + the number of bytes specified in the Diagnostic length.

Configuration length (in bytes)

Maximum amount of configuration data bytes (identifier bytes) for the application. Valid settings range from 0 to 244. If Configuration length = 0 the default value of 24 bytes will be used. The module will also round the given Configuration length up to the nearest multiple of 8 bytes.

Mode selection

The settings in this word determines how the module shall operate.

User Parameter data handling:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
		(reserved)													
b1	b0	User Parameter data handling:													
0	0	No verification of User Parameter data													
0	1	Anybus verifies User Parameter data													
1	0	Application verifies User Parameter data													
1	1	(reserved)													

Configuration data handling:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
			(reserved)												
b3	b2	Configuration data handling													
0	0	No verification of Configuration data													
0	1	Anybus verifies Configuration data													
1	0	Application verifies Configuration data													
1	1	(reserved)													

C2 - Initiate handling:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	
		(reserved)														
C2 - Initiate handling																
0: Module handles C2-Initiate																
1: Application handles C2-Initiate																

Transparent Object handling:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
		(reserved)													
Transparent Object handling															
0: Disabled															
1: Enabled															

C2 - Abort handling:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
		(reserved)													

C2 - Abort handling															
0: Application is not informed of C2_Abort reason															
1: Application is informed of C2_Abort reason															

Output action when Global command CLEAR is received from the Master:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
(reserved)															
b15	b14	Action													
0	0	Out-area cleared (0)													
0	1	Out-area freeze													
1	0	Out-area set (1)													
1	1	(reserved)													

C1-length (in bytes)

Specifies the maximum number of data bytes allowed for a C1 read/write request. Valid settings range from 0 to 240. The module will set up its internal C1-buffer based on this value + 4 bytes of header data. I.e C1-buffer length = C1-length + 4 (C1 buffer length range = 4 to 244)

C2-length (in bytes)

Specifies the maximum number of data bytes allowed for a C2 read, write and data transport request. Valid settings range from 0 to 240. The module will set up its internal C2-buffer based on this value + 4 bytes of header data. I.e C2-buffer length = C2-length + 4 (C2 buffer length range = 4 to 244)

If the C2-buffer length is less than 20 bytes, it will be adjusted to 20 bytes.

Message dataword 8

In firmware revisions below 1.20 this specifies the C2-Send Timeout (Timebase = 10ms, e.g a value of 100 equals 1 second) as well as the timeout time allowed on the mailbox interface for transparent C2 read/write operations. Valid settings range from 1 to 32767, if a value of 0 (zero) is given a default value of 100 milliseconds will be used. The value should be as low as possible and shall not exceed the following values:

Baudrate in kbaud	<= 187.5	500	1500	3000	6000	12000
C2-Send Timeout in seconds	4	2	1	1	1	1

In firmware revision 1.20 and upwards this word is reserved and should be set to zero since the C2 send timeout time is determined by the module firmware. There is also no timeout against the application system.

Fault Information

If the error code in the Message Information registers is 'Undefined ERROR' (Fh) the fault information in Extended word 8 can be interpreted using the following table:

Register value	Description
0031h	Not enough memory in the PROFIBUS communication controller
0033h	Invalid Input/Output data length
0034h	Invalid Diagnostics data length
0035h	Invalid User Parameter data length
0036h	Invalid Configuration data length
0021h	Invalid C2-length
0024h	Invalid C1-length
000Ch	Invalid C2-Send timeout
000Dh	Invalid Mode selection setting
(all other)	Reserved

FB_APPL_SET_NODE_ADDRESS

Description

This command is used to change the network node address of the module.

Address range: 0 - 126

Note: This command can only be sent during module initialisation.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_SET_NODE_ADDRESS
Message type	02h
Command number	0004h
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: Contains the node address (0 - 126) Response: The command message data is returned
Response message	If the command was not accepted the reason for the rejection is indicated in the Message information word. If the message information indicates "Undefined Error" further fault information is available in Extended word 8. No Message data will then be returned.

Command and response layout:

	Command	Expected response
Message ID	(ID)	(ID)
Message information	4002h	0002h
Command	0004h	0004h
Data size	0002h	0002h
Frame count	0001h	0001h
Frame number	0001h	0001h
Offset high	0000h	0000h
Offset low	0000h	0000h
Extended word 1	-	-
Extended word 2	-	-
Extended word 3	-	-
Extended word 4	-	-
Extended word 5	-	-
Extended word 6	-	-
Extended word 7	-	-
Extended word 8	-	Fault information
Message data word 1	Node address	Node address

FB_APPL_SET_NODE_ADDRESS

Fault Information

If the error code in the Message Information registers is 'Undefined ERROR' (Fh) the fault information in Extended word 8 can be interpreted using the following table:

Register value	Description
0001h	The node address is not in the range 0 - 126.
(all other)	Reserved.

FB_APPL_GET_NODE_ADDRESS

Description

This command returns the currently used node address of the module.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_GET_NODE_ADDRESS
Message type	02h
Command number	0006h
Fragmented	No
Extended Header data	Command: None Response: None
Message data	Command: None Response: The response holds the currently used node address.
Response message	If the command is not accepted the reason for the rejection is indicated in the Message Information word.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0006h	0006h	FB_APPL_GET_NODE_ADDRESS
Data size	0000h	0002h	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	-	
		Node address	Message dataword 1

Parameter Data

FB_ABS_VERIFY_USER_PRM

Description

This mailbox message can be sent to the application whenever the module receives a Set_Prm message from the PROFIBUS master. The application must evaluate the User Parameter data and send a positive or negative response to the module. If the response is negative the Set_Prm telegram will be rejected and the PrmFault bit will be set in the Slave_Diag response.

This procedure will only be carried out if bit 0-1 in the Mode selection word (Message data word 5 in FB_INIT) is set to b10 - Application verifies User Parameter data.

Note: The first four bytes (DP-V1 status bytes + Anybus status byte) of User Parameter data will not be sent to the application as these bytes are reserved for the module exclusively.

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_VERIFY_USER_PRM
Message type	02h
Command number	0007h
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: User Parameter data (Excluding DP-V1- and Anybus status bytes) Response: None
Response message	If the application finds the User Parameter data to be correct, it should clear (0) bit 0 in the Fault Information word (Extended word 8). If the application finds the User Parameter data to be incorrect, it should set (1) bit 0 in the Fault information word (Extended word 8)

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0007h	0007h	FB_ABS_VERIFY_USER_PRM
Data size	(User Parameter data size)	0000h	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	Fault Information	Fault Information
Message data	User Parameter data		

FB_APPL_GET_USER_PRM

Description

With this mailbox command the application can read out the User parameter data buffer.

This telegram will only be accepted after a successful END_INIT.

Note 1: The module will reject this command if the application sends it before the module has received the Set_Prm telegram from the master. An error code in Extended word 8 of the response will indicate this.

Note 2: The first four bytes (DP-V1 status bytes + Anybus status byte) of User Parameter data will not be sent to the application as these bytes are reserved for the module exclusively.

Parameter	Description
Command initiator	Application
Command Name	FB_ABS_GET_USER_PRM
Message type	02h
Command number	0002h
Fragmented	No
Extended Header data	Command: None Response: None
Message data	Command: None Response: User Parameter Data
Response message	If the command is rejected the reason is indicated in the Message information word.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0002h	0002h	FB_APPL_GET_USER_PRM
Data size	0000h	(User parameter data size)	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	Fault Information	Fault Information
		User Parameter data	

Fault Information

If the error code in the Message Information registers is 'Undefined ERROR' (Fh) the fault information in Extended word 8 can be interpreted using the following table:

Register value	Description
0001h	No parameter data available, the module has not received the Set_Parameter message from the master.
(all other)	Reserved.

FB_APPL_SET_USER_PRM

Description

This mailbox command must be sent between the FB_INIT and END_INIT messages.

The application can send its expected User Parameter data to the module during the initialisation. This data will then be compared to the User Parameter data received from the master during the parameterisation sequence. If the expected User Parameter data from the application does not equal the User Parameter data received from the master, the Set_Prm telegram will be rejected and the PrmFault bit will be set in the Slave_Diag response.

This check will only be carried out if bit 0-1 in the Mode selection word (Message data word 5 in FB_INIT) is set to b01 - Anybus verifies User Parameter data.

If bit 0-1 in the Mode Selection word is set to any other value this command will be rejected by the module.

Note: The first four bytes (DP-V1 status bytes + Anybus status byte) of User Parameter data should not be included in the message data as these bytes are reserved for the Anybus exclusively. The first byte included in the check will be Octet 11.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_SET_USER_PRM
Message type	02h
Command number	0005h
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: Expected User Parameter data Response: The command message data is returned.
Response message	If the command is rejected the reason is indicated in the Message information word. If the message identification register indicates "Undefined ERROR", detailed fault information is found in Extended word 8. No message data will be returned.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0005h	0005h	<i>FB_ABS_SET_USER_PRM</i>
Data size	(User Parameter data size)	(User Parameter data size)	<i>Size = User parameter data size</i>
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	Fault Information	<i>Fault Information</i>
Message data	User Parameter data	User Parameter data	

Fault Information

If the error code in the Message Information registers is 'Undefined ERROR' (Fh) the fault information in Extended word 8 can be interpreted using the following table:

Register value	Description
0001h	Bit 0-1 in the Mode selection word in the FB_INIT message did not equal '01', that is, the module is not allowed to verify the User Parameter Data.
(all other)	Reserved.

Configuration Data

FB_ABS_VERIFY_CONFIG

Description

This mailbox telegram can be sent to the application when the module receives a Chk_Cfg telegram from the PROFIBUS master. The application must then evaluate the configuration data (identifier bytes) and send a positive or negative response to the module. If the response is negative the Chk_Cfg telegram will be rejected and the Cfg_Fault bit will be set in the Diagnostic message response.

This check will only be carried out if bit 2-3 in the Mode selection word (Message data word 5 in FB_INIT) is set to b10 - Application verifies Configuration data.

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_VERIFY_CONFIG
Message type	02h
Command number	0008h
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: Configuration data Response: None.
Response message	If the application finds the configuration to be correct, it should clear (0) bit 0 in the Fault information register (Extended word 8) If the application finds the configuration to be incorrect, it should set (1) bit 0 in the Fault information register (Extended word 8)

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0008h	0008h	FB_ABS_VERIFY_CONFIG
Data size	(Configuration data size)	0000h	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	Fault Information	Fault Information
Message data	Configuration data		

FB_APPL_GET_CONFIG

Description

With this mailbox the application can read out the contents of the configuration data buffer.

This command will only be accepted after a successful END_INIT.

Note: The module will reject this command if the application sends it before the module has received the Chk_Cfg telegram from the master. An error code in Extended word 8 in the response will indicate this situation. The returned configuration data in this case is the default configuration for the module.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_GET_CONFIG
Message type	02h
Command number	000Ah
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: None Response: Configuration data
Response message	If the telegram is not accepted the reason for the rejection is indicated in the Message information word.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	000Ah	000Ah	<i>FB_APPL_GET_CONFIG</i>
Data size	0000h	(Configuration data size)	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	Fault Information	<i>Fault Information</i>
		Configuration data	

Fault Information

If the error code in the Message Information registers is 'Undefined ERROR' (Fh) the fault information in Extended word 8 can be interpreted using the following table:

Register value	Description
0001h	No Configuration Data is available, the module has not received the Chk_Cfg message from the master. The returned data will in this case be the default configuration data for the present setup.
(all other)	Reserved.

FB_APPL_SET_CONFIG

Description

This telegram must be sent to the module before END_INIT.

This message serves two purposes:

1. With this message the application can send its expected configuration to the module during the initialisation. The expected configuration will then be compared with the actual configuration that is received from the master during the check configuration sequence. If the expected configuration from the application side does not equal the configuration from the master side, the Chk_Cfg telegram will be rejected and the Cfg_Fault bit will be set in the Diagnostic message response to the master.

This check will only be carried out if bit 2-3 in the Mode selection word (Message data word 5 in FB_INIT) is set to b01 - Anybus verifies Configuration data.

2. Set default configuration:

The configuration data received with this mailbox will be used as default configuration for the PROFIBUS telegram Get_Cfg.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_SET_CONFIG
Message type	02h
Command number	0009h
Fragmented	No
Extended Header data	Command: None Response: Fault information
Message data	Command: Configuration data Response: The command message data is returned
Response message	If the telegram is not accepted the reason for the rejection is indicated in the Message information word.

Command and response layout:

	Command	Expected response
Message ID	(ID)	(ID)
Message information	4002h	0002h
Command	0009h	0009h
Data size	(Configuration data size)	0000h
Frame count	0001h	0001h
Frame number	0001h	0001h
Offset high	0000h	0000h
Offset low	0000h	0000h
Extended word 1	-	-
Extended word 2	-	-
Extended word 3	-	-
Extended word 4	-	-
Extended word 5	-	-
Extended word 6	-	-
Extended word 7	-	-
Extended word 8	-	-
Message data	Configuration data	Configuration data

FB_APPL_SET_CONFIG

Diagnostics

FB_APPL_WRITE_DIAGNOSTICS

Description

This telegram can only be sent to the module after END_INIT.

The maximum size of the Diagnostic data is defined in FB_INIT.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_WRITE_DIAGNOSTICS
Message type	02h
Command number	0003h
Fragmented	No
Extended Header data	Command: Service code (See description below) Response: Fault information
Message data	Command: Diagnostic data Response: The command message data is returned
Response message	If the telegram is not accepted the reason for the rejection is indicated in the Message information word.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0003h	0003h	FB_APPL_WRITE_DIAGNOSTICS
Data size	(Diagnostic data size)	(Diagnostic data size)	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	- Service code	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	-	-	
Extended word 6	-	-	
Extended word 7	-	-	
Extended word 8	-	Fault Information	Fault Information
Message data	Diagnostic data	Diagnostic data	

Service Code (Low byte of Extended Word 1)

In this byte the application can control the Ext_Diag, Ext_Diag_Overflow and Static_Diag bits in the Standard diagnostic bytes (Station_Status_1 - 3).

Extended Diagnostic: (Ext_Diag)

b7	b6	b5	b4	b3	b2	b1	b0
(reserved)					(reserved)		

The application should set this bit whenever the FB_APPL_WRITE_DIAGNOSTICS message contains extended diagnostic data that should be sent to the master. When the reason for the diagnostic information is corrected this bit should be cleared by the application.

b0	Description
0	Only status diagnostic data is available.
1	Extended diagnostic information is available in the message.

Static Diagnostic: (Stat_Diag)

b7	b6	b5	b4	b3	b2	b1	b0
(reserved)					(reserved)		

The application should set this bit whenever the fault reported is of a kind that makes it impossible to continue to supply the fieldbus with valid cyclic I/O data. The master will then stop the cyclic data exchange until the fault is corrected, which the application reports by clearing this bit.

b1	Description
0	Static diagnostic not active. (I/O data is valid.)
1	Static diagnostic is active. (I/O data is not valid.)

Extended Diagnostic Overflow (Ext_Diag_Overflow):

b7	b6	b5	b4	b3	b2	b1	b0
(reserved)					(reserved)		

The application should set this bit if the diagnostic data available in the application is larger than the current size of the diagnostic data buffer. For example, too many channel-related diagnostic messages can be pending at the same time.

b3	Description
0	No diagnostic overflow, the application can include all diagnostic information in the Diagnostic data buffer.
1	Diagnostic overflow, the application have more diagnostic information than what is possible to enter in the Diagnostic data buffer.

Fault Information

If the error code in the Message Information registers is 'Undefined ERROR' (Fh) the fault information in Extended word 8 can be interpreted using the following table:

Register value	Description
0001h	The module is processing a previous diagnostic request and cannot accept a new one.
(all other)	Reserved.

Acyclic Data Functions

General

The Extended header data in the response message contains the following parameters

- **Slot Number**
Meaning: Parameter for addressing
Range: 0-254 is available to the application, 255 is reserved.
- **Index**
Meaning: Parameter for addressing
Range: 0-254 is available to the application, 255 is reserved.
- **Length**
Meaning: Number of data bytes to read/write
Range: 0-240
- **SAP Number**
Meaning: SAP (Service Access Point) for the acyclic request. With this information the application can determine if a Class 1 or Class 2 master issued the Read/Write
For a Class 1 connection SAP 51 (50) is permanently used.
For a Class 2 connection SAP's 45-48 are used in decreasing sequence, starting with SAP 48. (The module supports a maximum of 2 simultaneous active C2 connections)
- **Error_Code_1 and Error_Code_2**
If the acyclic request was accepted the application should respond with Error_Code_1 and Error_Code_2 cleared.
If the acyclic request was not accepted the application must respond with a suitable error code according to the table below.

Error Codes

This table shows how the 'Error_Code_1', 'Error_Code_2', 'Error Class', 'Error Code' and 'User Specific' corresponds to the Extended word 8:

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Error_Code_1								Error_Code_2							
Error Class				Error Code				User Specific							

This table contains the applicable values for 'Error Class' and 'Error Code'

Error Class	Meaning	Error Code
0	No error	When 'No error' is reported both 'Error Code' and 'User Specific' must also be set to zero.
1 - 9	(reserved)	-
10	Application	0 = Read error 1 = Write error 2 = Module error 3-7 = (reserved) 8 = Version conflict 9 = Feature not supported 10-15 = User Specific
11	Access	0 = Invalid index 1 = Write length error 2 = Invalid slot 3 = Type conflict 4 = Invalid area 5 = State conflict 6 = Access denied 7 = Invalid range 8 = Invalid parameter 9 = Invalid type 10 - 15 = User Specific
12	Resource	0 = Read constrain conflict 1 = Write constrain conflict 2 = Resource busy 3 = Resource unavailable 4 - 7 = (reserved) 8 - 15 = User Specific
13 - 15	User Specific	- = User Specific

FB_ABS_ACYCLIC_WRITE

Description

This functionality is only available if bit 5 in the Mode selection word (Message data word 5 in FB_INIT) is set to 1 (i.e. Transparent Object handling enabled)

The application must respond to this message. If the request is accepted, the application should clear Error_code_1 and Error_code_2, and write the requested read data in the Message data word field.

If the application does not accept the request, it should specify the reason for the rejection in Error_Code_1 and Error_Code_2. (See 4-22 “Error Codes”)

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_ACYCLIC_WRITE
Message type	02h
Command number	000Ch
Fragmented	No
Extended Header data	Command: Slot number, Index, Length and SAP-number Response: Error_Code_1, Error_Code_2
Message data	Command: Write data Response: None
Response message	If the application does not accept the request the reason for the rejection should be indicated in Extended word 8.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	000Ch	000Ch	FB_ABS_ACYCLIC_WRITE
Data size	(data size)	0000h	Number of bytes to write
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	SAP-number	-	
Extended word 6	Slot number	-	
Extended word 7	Index	-	
Extended word 8	Length	Error_Code_1 1	Error_Code_2 2
Message data	Write data	DP-V1 specific fault information	

SAP-number, Slot number, Index & Length

See page 4-21 “General” for more information.

Write data

The data that should be written to the addressed object.

Error_Code_1 & Error_Code_2

See page 4-22 “Error Codes” for more information.

FB_ABS_ACYCLIC_READ

Description

This functionality is only available if bit 5 in the Mode selection word (Message data word 5 in FB_INIT) is set to 1 (i.e. Transparent Object handling enabled)

The application must respond to this message. If the request is accepted, the application should clear Error_code_1 and Error_code_2, and write the requested read data in the Message data word field.

If the application does not accept the request, it should specify the reason for the rejection in Error_Code_1 and Error_Code_2. (See 4-22 “Error Codes”)

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_ACYCLIC_READ
Message type	02h
Command number	000Dh
Fragmented	No
Extended Header data	Command: Slot number, Index, Length and SAP-number Response: Error Code1, Error Code2
Message data	Command: None Response: Read data
Response message	If the application does not accept the request the reason for the rejection should be indicated in Extended word 8.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	000Dh	000Dh	<i>FB_ABS_ACYCLIC_READ</i>
Data size	0000h	(data size)	<i>Number of bytes to read</i>
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	SAP-number	-	
Extended word 6	Slot number	-	
Extended word 7	Index	-	
Extended word 8	Length	Error_Code_1 Error_Code_2	<i>DP-V1 specific fault information</i>
		Read data	Response data

SAP-number, Slot number, Index & Length

See page 4-21 “General” for more information.

Error_Code_1 & Error_Code_2

See page 4-22 “Error Codes” for more information.

Read data

The data read from the addresses object.

FB_ABS_ACYCLIC_DATATRANSPORT

Description

This functionality is only available if bit 5 in the Mode selection word (Message data word 5 in FB_INIT) is set to 1 (i.e. Transparent Object handling enabled)

The application must respond to this message. If the request is accepted, the application should clear Error_code_1 and Error_code_2, and write the requested read data in the Message data word field.

If the application does not accept the request, it should specify the reason for the rejection in Error_Code_1 and Error_Code_2. (See 4-22 “Error Codes”)

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_ACYCLIC_DATATRANSPORT
Message type	02h
Command number	0012h
Fragmented	No
Extended Header data	Command: Slot number, Index, Length and SAP-number Response: Error Code1, Error Code2
Message data	Command: Write data Response: Read data
Response message	If the application does not accept the request the reason for the rejection should be indicated in Extended word 8.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0012h	0012h	
Data size	(write data size)	(read data size)	
Frame count	0001h	0001h	
Frame number	0001h	0001h	
Offset high	0000h	0000h	
Offset low	0000h	0000h	
Extended word 1	-	-	
Extended word 2	-	-	
Extended word 3	-	-	
Extended word 4	-	-	
Extended word 5	SAP-number	-	
Extended word 6	Slot number	-	
Extended word 7	Index	-	
Extended word 8	Length	Error_Code_1	Error_Code_2
Message data	Write data	Read data	

FB_ABS_ACYCLIC_TRANSPORT

DP-V1 specific fault information

Response data

SAP-number, Slot number, Index & Length

See page 4-21 “General” for more information.

Write data

The data that should be written to the addressed object.

Error_Code_1 & Error_Code_2

See page 4-22 “Error Codes” for more information.

Read data

The data read from the addresses object.

FB_ABS_ACYCLIC_INITIATE

Description

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_ACYCLIC_INITIATE
Message type	02h
Command number	000Fh
Fragmented	No
Extended Header data	Command: SAP-number used for this connection Response: Error Code1, Error Code2
Message data	Command: Master Initiate parameters Response: Slave Initiate parameters
Response message	If the application does not accept the request, the reason for the rejection should be indicated in Extended word 8.

Command and response layout:

	Command		Expected response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	000Fh		000Fh		FB_ABS_ACYCLIC_INITIATE
Data size	12 + S_length+D_length		10 + S_length+D_length		Data size
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	-		-		
Extended word 2	-		-		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-		-		
Extended word 6	-		-		
Extended word 7	-		-		
Extended word 8	SAP-number		Subnet	Instance/ Reason	DP-V1 specific fault information
Command dataword 1	Send Timeout		Features supported 1	Features supported 2	Response dataword 1
Command dataword 2	Features supported 1	Features supported 2	Profile Features supported 1	Profile Features supported 2	Response dataword 2
Command dataword 3	Profile Features supported 1	Profile Features supported 2	Profile Ident number		Response dataword 3
Command dataword 4	Profile Ident number		S_Type	S_Length	Response dataword 4
Command dataword 5	S_Type	S_Length	D_Type	D_Length	Response dataword 5
Command dataword 6	D_Type	D_Length	S_Address (Size of S_address=S_length)		
	S_Address (Size of S_address=S_length)		D_Address (Size of D_address=D_length)		
	D_Address (Size of D_address=D_length)				

SAP-number

The associated 'Service Access Point'. See page 4-21 "SAP Number" for more information.

Send Timeout

Monitoring time for the connection.

Features supported 1 & 2

Identifies the supported C2-service functionality.

Features supported 1:

Bit 0: DP-V1_RW - This bit is set if the services C2_Read and C2_Write are supported.

Bit 1-7: (reserved)

Features supported 2:

Bit 0-7: (reserved)

Profile features supported 1 & 2

Identifies the supported service functionality regarding the used profile definition. The meaning of the defined bits is profile or vendor specific.

Profile features supported 1:

Bit 0-7: Defined by the profile.

Profile features supported 2:

Bit 0-7: Defined by the profile.

Profile Ident number

By means of this parameter a unique profile definition is identified. The Profile Ident Number will be taken from the pool of Ident Numbers handled by PNO.

S_Type & S_Length

S_Type

This parameter indicates the presence of the optional Network/MAC address of the source

Bit 0: This bit is set to one if a Network/MAC address is present in parameter S_Address (See below). If this bit is set to 0 (zero) it means that the source is the endpoint of the connection.

Bit 1-7: (reserved)

S_Length:

This parameter indicate the length of the S_Address parameter (See below)

D_type & D_length

D_Type

This parameter indicates the presence of the optional Network/MAC address of the destination.

Bit 0: This bit is set to one if a Network/MAC address is present in parameter D_Address (See below). If this bit is set to 0 (zero) it means that the source is the endpoint of the connection.

Bit 1-7: (reserved)

D_Length:

This parameter indicate the length of the D_Address parameter (See below)

S_Address

S_Address contains additional address information of the source. The structure of S_Address is dependant of the S_Type setting. (See below)

S_Type	Type	Subparameter	Description
0	Unsigned 8	API	Identifies the application process instance of the source
	Unsigned 8	SCL	Identifies the access level of the source. (0= No access level used)
1	Unsigned 8	API	Identifies the application process instance of the source.
	Unsigned 8	SCL	Identifies the access level of the source
	Octet-string [6]	Network Address	Identifies the access level of the source
	Octet-string [S_Length]	MAC Address	Identifies the MAC address of the source

D_Address

D_Address contains additional address information of the destination. The structure of D_Address is dependant of the D_Type settings. (See below)

D_Type	Type	Subparameter	Description
0	Unsigned 8	API	Identifies the application process instance of the destination
	Unsigned 8	SCL	Identifies the access level of the destination. (0= No access level used)
1	Unsigned 8	API	Identifies the application process instance of the destination.
	Unsigned 8	SCL	Identifies the access level of the destination
	Octet-string [6]	Network Address	Identifies the access level of the destination
	Octet-string [S_Length]	MAC Address	Identifies the MAC address of the destination

Subnet

This parameter is used to encode information about the location of the source of the abort initiator.

Possible values:

0 - Local source

1 - Local subnet

2 - Remote subnet

3 to 255 - (reserved)

Instance / Reason

Instance is used to encode information about the protocol instance that detected a problem that caused the abort request. Reason indicates the reason for the abort request. (See below)

Note: If the application sets Subnet to 0 (zero), the module will automatically set the Instance code to b10 (User).

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(reserved, set to b00)		Instance code: b00 - FDL b01 - C2 b10 - User b11 -(reserved)		Reason Code (See below)			

Instance Code	Reason Code	Name	Meaning
FDL	1	UE	Remote-DMPM/DL interface error
	2	RR	Resource not sufficient or not available
	3	RS	Service, remote address or SAP not activated
	9	NR	No response data
	10	DH	Acknowledge for sent data, reply data with high priority available
	11	LR	Local resource not available or not sufficient
	12	RDL	Response data low and no resource for sent data
	13	RDH	Response data high and no resource for sent data
	14	DS	Master is not in logical ring
	15	NA	No response from remote FDL
MSAC2	1	ABT_SE	Sequence error; service not allowed in this state
	2	ABT_FE	Invalid request PDU received
	3	ABT_TO	Timeout of the connection
	4	ABT_RE	Invalid response PDU received
	5	ABT_IV	Invalid service from User
	6	ABT_STO	Send_Timeout requested was too small
	7	ABT_IA	Invalid additional address information
	8	ABT_OC	Waiting for FDL_DATA_REPLY
	15	ABT_RES	Resource Error

FB_ABS_ACYCLIC_ABORT_IND

Description

This message is sent by the module if the master that initiated the C2-connection should decide to abort it.

Parameter	Description
Command initiator	Module
Command Name	FB_ABS_ACYCLIC_ABORT_IND
Message type	02h
Command number	0010h
Fragmented	No
Extended Header data	SAP-number of the aborted connection
Message data	Subnet, Instance, and Reason for the abortion
Response message	none

Command and response layout:

Command		
Message ID	(ID)	
Message information	4002h	
Command	0010h	<i>FB_ABS_ACYCLIC_ABORT_IND</i>
Data size	0002h	<i>2 bytes of data (1 word)</i>
Frame count	0001h	
Frame number	0001h	
Offset high	0000h	
Offset low	0000h	
Extended word 1	SAP-number	<i>SAP-number (low byte)</i>
Extended word 2	-	
Extended word 3	-	
Extended word 4	-	
Extended word 5	-	
Extended word 6	-	
Extended word 7	-	
Extended word 8	-	
Message data byte 1	Subnet	
Message data byte 2	Instance / Reason	

SAP-number

The associated ‘Service Access Point’. See page 4-21 “SAP Number” for more information.

Subnet & Instance / Reason

Information about the abort reason, see page 4-31 “Subnet” and page 4-32 “Instance / Reason”.

FB_APPL_ACYCLIC_ABORT

Description

The application can cancel an existing C2-connection by sending this mailbox.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_ACYCLIC_ABORT
Message type	02h
Command number	0011h
Fragmented	No
Extended Header data	Command: SAP-number of the aborted connection Response: Fault information
Message data	Command: Subnet, Instance, and Reason for the abortion Response: The command message data is returned
Response message	If the telegram is not accepted the reason for the rejection is indicated in the message information word

Command and response layout:

	Command		Expected response		
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	0011h		0011h		FB_APPL_ACYCLIC_ABORT
Data size	0002h		0002h		2 databytes (1 word)
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	-	SAP-number	-		
Extended word 2	-		-		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-		-		
Extended word 6	-		-		
Extended word 7	-		-		
Extended word 8	-		Fault information		DP-V1 specific fault information
Message dataword 1	Subnet	Instance / Reason	Subnet	Instance / Reason	Response data

SAP-number

The associated ‘Service Access Point’. See page 4-21 “SAP Number” for more information.

Subnet & Instance / Reason

Information about the abort reason, see page 4-31 “Subnet” and page 4-32 “Instance / Reason”.

Fault information

If the error code is 'Undefined ERROR' (Fh), extended error information is available in the Fault Information register (Extended word 8).

Error Code	Description
08h	Already disconnected.
0Fh	Invalid SAP.
11h	There is already an active disconnect request.
22h	DP-V1 Class 2 functionality not enabled.
(all other)	Reserved.

FB_APPL_MAP_PARAMETER_OBJECT

Description

It is possible to map several objects with one mailbox telegram. However, the maximum number of message data words per mailbox frame is 128. This means that it is not possible to map more than 42 objects with one mailbox. If more than 42 objects are required, several mailbox telegrams must be sent.

Note 1: The total maximum number of objects that can be mapped is 512.

Note 2: This command can only be sent during module initialisation.

Parameter	Description
Command initiator	Application
Command Name	FB_APPL_MAP_PARAMETER_OBJECT
Message type	02h
Command number	000Bh
Fragmented	No
Extended Header data	Command: Extended word 1 contains the number of objects to map (max 42) Response: Fault information
Message data	Slot Number, Index, Length, Offset and Access Type of the object.
Response message	The response data is a copy of the command data.

Command and response layout:

Command			Expected response		<div>FB_APPL_MAP_PARAMETER_OBJECT</div> <div>Datasize, 6 * no of objects to map</div>
Message ID	(ID)		(ID)		
Message information	4002h		0002h		
Command	000Bh		000Bh		
Data size	(size)		(size)		
Frame count	0001h		0001h		
Frame number	0001h		0001h		
Offset high	0000h		0000h		
Offset low	0000h		0000h		
Extended word 1	-		-		
Extended word 2	-		-		
Extended word 3	-		-		
Extended word 4	-		-		
Extended word 5	-		-		
Extended word 6	-		-		
Extended word 7	-		-		
Extended word 8	-		Error Code	Error Location	<div>(The response data is a copy of the command data)</div>
Message dataword 1	Slot 1	Index 1	Slot 1	Index 1	
Message dataword 2	Access type 1	Length 1	Access type 1	Length 1	
Message dataword 3	Offset 1		Offset 1		
Message dataword 4	Slot number 2	Index 2	Slot number 2	Index 2	
Message dataword 5	Access type 2	Length 2	Access type 2	Length 2	
Message dataword 6	Offset 2		Offset 2		
...		
Message dataword N	Slot number 'N'	Index 'N'	Slot number 'N'	Index 'N'	
..-	Access type 'N'	Length 'N'	Access type 'N'	Length 'N'	
..-	Offset 'N'		Offset 'N'		

Slot number

This parameter defines the slot number that the object should be mapped to, valid range is 0 - 254.

Index

This parameter defines the index that the object should be mapped to, valid range is 0 - 254.

Access Type

This parameter is used to set read/write permissions for the object.

Bit 0: This bit should be cleared if the object should be both readable and writeable. It should be set if the object should be read-only.

Bit 1-7: (reserved)

Length

This parameter is used to set the length of the object in bytes. The valid range is from 0 - maximum C1- or C2-length. The default maximum C1- and C2-length is 32 bytes but may be altered in FB_INIT message.

Offset

This parameter defines the offset address in bytes to where the object should be mapped in the parameter data area. The valid range is 0 - maximum size of either the parameter in-area or the parameter out-area as specified in ANYBUS_INIT message, a value of zero points to the first byte in the parameter data area.

Error Code and Error Location

If the error code is 'Undefined ERROR' (Fh), extended error information is available in the Fault Information register (Extended word 8).

Error Code	Description
01h	Invalid slot number.
02h	Invalid Index.
03h	Invalid access type.
04h	Invalid length.
05h	Invalid offset.

The 'Error Location' specifies the object number of the faulty object. If the first object in the mailbox telegram (i.e. message data word 1-3) is incorrect the 'Error Location' will equal 1, if the second object is incorrect the 'Error Location' will equal 2, etc.

Fieldbus Specific Area

Memory Map

Address	Description
0 - 32	Not Used
33	Global Command
34	Actual Baudrate
35	User Parameter data available
36	Configuration data available

Global Command

bit	Description
0	Reserved
1	Set: CLEAR command received
2	Set: UnFreeze command received
3	Set: Freeze command received
4	Set: Unsync command received
5	Set: Sync command received
6-7	Reserved

Actual Baurate

Value	Description
00h	No Baudrate set
01h	12 Mbit
02h	6 Mbit
03h	3 Mbit
04h	1.5 MBit
05h	500 kBit
06h	187.5 kBit
07h	93.75 kBit
08h	45.45 kBit
09h	19.2 kBit
0Ah	9.6 kBit

User Parameter Data available

- 0:** No User Parameter data available for the application
- 1:** User Parameter data available for the application

Configuration Data available

- 0:** No Configuration data available for the application
- 1:** Configuration data available for the application

Environmental Specification

Temperature

Operating

+5 to +70 degrees Celsius

Test performed according to IEC-68-2-1 and IEC 68-2-2.

Non Operating

-25 to +85 degrees Celsius

Test performed according to IEC-68-2-1 and IEC 68-2-2.

Relative Humidity

The product is designed for a relative humidity of 5 to 95% non-condensing.

Test performed according to IEC 68-2-30.

EMC compliance

Emission

According to EN 50 081-2:1993

Tested per 55011:1990, class A, radiated

Immunity

According to EN 61000-6-2:1999

Tested per EN 61000-4-2:1995

 EN 61000-4-3:1996

 EN 61000-4-4:1995

 EN 61000-4-5:1995

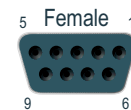
 EN 61000-4-6:1996

Connectors

The module supports the different types of connectors; 9-pin Dsub, screw terminal, and 2mm board-to-board. The 9-pin D-sub is the standard connector and is recommended for data transfer rates higher than 1500kbit/s..

9-pin D-sub

Pin	Signal
Housing	Shield
1	NC
2	NC
3	B-Line
4	RTS*
5	GND BUS**
6	+5V BUS**
7	NC
8	A-Line
9	NC



Screw Terminal

Pin	Signal
1	+5V BUS**
2	GND BUS**
3	A-Line
4	B-Line
5	Shield
6	RTS*

2mm Board to Board

Pin	Signal
1	Shield
2	GND BUS**
3	RTS*
4	NC
5	A-Line
6	B-Line
7	NC
8	+5V BUS**
9	NC
10	NC



*) RTS is used in some equipment to determine the direction of transmission. In normal applications only A-Line, B-Line and Shield are used.

**) +5V BUS and GND BUS are used for bus termination. Some devices, like optical transceivers (RS485 to fibre optics) might require external power supply from these pins. This device is not allowed to draw more than 80mA from these power pins.

Electrical Characteristics

Supply Voltage

Both the module electronics and the fieldbus interface requires a regulated 5V DC powersupply. For more information regarding power requirements, consult the Anybus-S Design Guide.

Maximum Current Consumption

The maximum current consumption is 300mA on the bus interface, 300 mA on the electronics interface. The total current consumption for both power ports is 400mA.

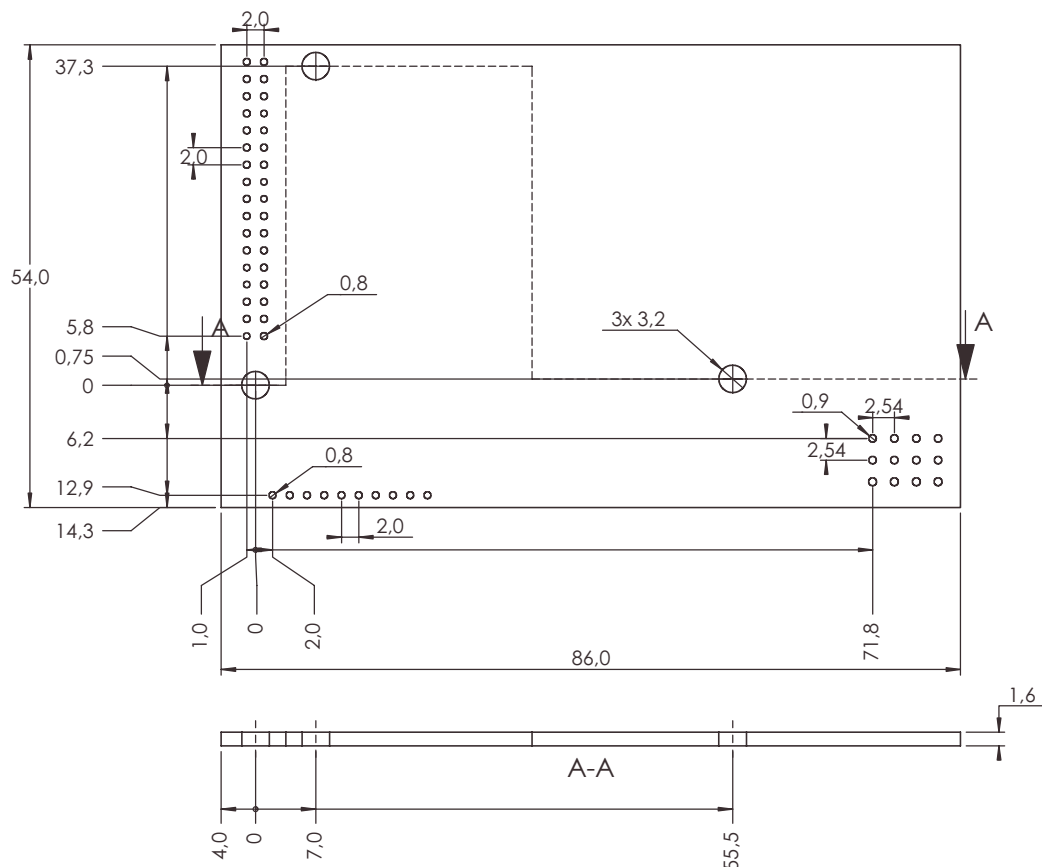
PE Grounding

A PE-connection is included on one of the front mounting holes according to the Anybus-S standard.

Mechanical Specification

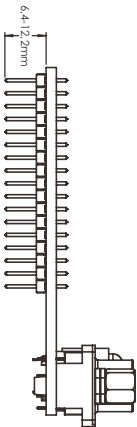
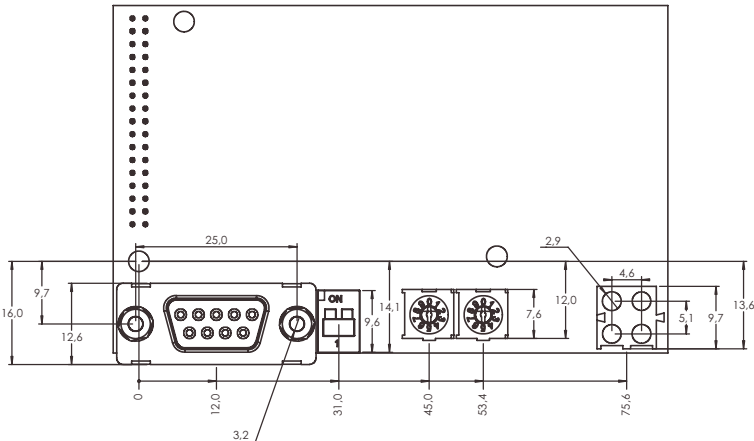
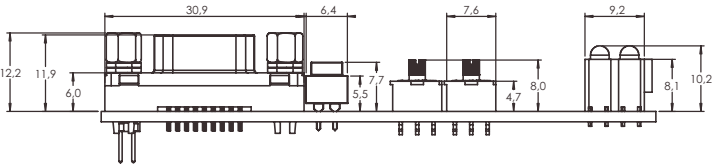
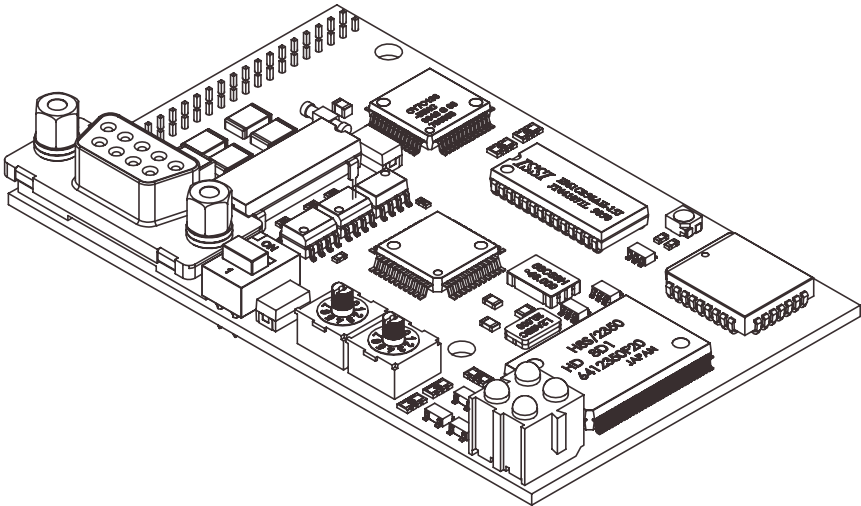
Measurements, PCB

The PCB is designed to fulfil the Anybus-S requirements.



Measurements, Connectors & Switches

Straight Switches & Connectors



Angled Switches & Connectors

