

Anybus[®] CompactCom PROFIBUS DP-V0

Network Interface Appendix

Doc.Id. SCM-1200-059
Rev. 1.02



HALMSTAD • CHICAGO • KARLSRUHE • TOKYO • BEIJING • MILANO • MULHOUSE • COVENTRY • PUNE • COPENHAGEN

HMS Industrial Networks
Mailing address: Box 4126, 300 04 Halmstad, Sweden
Visiting address: Stationsgatan 37, Halmstad, Sweden

E-mail: info@hms-networks.com
Web: www.anybus.com

Important User Information

This document is intended to provide a good understanding of the functionality offered by PROFIBUS. The document only describes the features that are specific to the Anybus CompactCom PROFIBUS DP-V0. For general information regarding the Anybus CompactCom, consult the Anybus CompactCom design guides.

The reader of this document is expected to be familiar with high level software design, and communication systems in general. The use of advanced PROFIBUS-specific functionality may require in-depth knowledge in PROFIBUS networking internals and/or information from the official PROFIBUS specifications. In such cases, the people responsible for the implementation of this product should either obtain the PROFIBUS specification to gain sufficient knowledge or limit their implementation in such a way that this is not necessary.

Liability

Every care has been taken in the preparation of this manual. Please inform HMS Industrial Networks AB of any inaccuracies or omissions. 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 applications meet all performance and safety requirements including any applicable laws, regulations, codes, and standards.

HMS Industrial Networks AB will under no circumstances assume liability or responsibility for any problems that may arise as a result from the use of undocumented features, timing, or functional side effects found outside the documented scope of this product. The effects caused by any direct or indirect use of such aspects of the product are undefined, and may include e.g. compatibility issues and stability issues.

The examples and illustrations in this document are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular implementation, HMS Industrial Networks AB cannot assume responsibility for actual use based on these examples and illustrations.

Intellectual Property Rights

HMS Industrial Networks AB has intellectual property rights relating to technology embodied in the product described in this document. These intellectual property rights may include patents and pending patent applications in the US and other countries.

Trademark Acknowledgements

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

Warning:	This is a class A product. in a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.
ESD Note:	This product contains ESD (Electrostatic Discharge) sensitive parts that may be damaged if ESD control procedures are not followed. Static control precautions are required when handling the product. Failure to observe this may cause damage to the product.

Table of Contents

Preface	About This Document
	Related Documents..... 1
	Document History 1
	Conventions & Terminology..... 2
	Support 3
Chapter 1	About the Anybus-CC PROFIBUS DP-V0
	General..... 4
	Features..... 4
	Front View 5
Chapter 2	Tutorial
	Introduction 6
	Fieldbus Conformance Notes 6
	Certification..... 6
Chapter 3	Basic Operation
	General Information..... 7
	<i>Software Requirements</i> 7
	<i>Electronic Data Sheet (GSD)</i> 7
	Buffer Modes 8
	Communication Settings 9
	Device Identity 9
	Data Exchange..... 10
	<i>Process Data</i> 10
	Diagnostics 11
	<i>Standard Diagnostics</i> 11
	<i>Extended Diagnostics</i> 11
	Parametrization Data Handling..... 12
	<i>General Information</i> 12
	<i>Validation</i> 13
	Configuration Data Handling..... 14
	<i>General Information</i> 14
	<i>Validation</i> 14
	Set Slave Address 15
Chapter 4	Anybus Module Objects
	General Information..... 16
	Anybus Object (01h)..... 17
	Diagnostic Object (02h)..... 18

	Network Object (03h)	20
	Network Configuration Object (04h)	22
	PROFIBUS DP-V0 Diagnostic Object (10h).....	24
	Network PROFIBUS DP-V1 Object (0Bh)	26
Chapter 5	Host Application Objects	
	General Information	30
	PROFIBUS DP-V1 Object (FDh)	31
Appendix A	Categorization of Functionality	
	Basic	33
	Extended.....	33
	Advanced.....	33
Appendix B	Implementation Details	
	SUP-Bit Definition.....	34
	Anybus State Machine	34
	Watchdog Behaviour (Application Stopped)	34
Appendix C	GSD-file Customization	
	General.....	35
	Device Identification	36
	Supported Hardware Features.....	37
	Supported DP Features	37
	<i>GSD-file Entries</i>	37
	Supported Baudrates.....	38
	Maximum Responder Time for Supported Baudrates	39
	Maximum Polling Frequency	39
	I/O-related Keywords.....	40
	Definition of Modules	41
	Parametrization-related Keywords	42
	Diagnostic-related Keywords	43
	Status Diagnostic Messages	44
	DP-V1 Keywords.....	45

Appendix D Technical Specification

Protective Earth (PE) Requirements.....	48
Power Supply	48
Environmental Specification	48
EMC Compliance.....	48

Appendix E Timing & Performance

General Information.....	49
Process Data	50
<i>Overview</i>	50
<i>Anybus Read Process Data Delay (Anybus Delay)</i>	50
<i>Anybus Write Process Data Delay (Anybus Delay)</i>	50
<i>Network System Read Process Data Delay (Network System Delay)</i>	51
<i>Network System Write Process Data Delay (Network System Delay)</i>	51

P. About This Document

For more information, documentation etc., please visit the HMS website, 'www.anybus.com'.

P.1 Related Documents

Document	Author
Anybus-CompactCom Software Design Guide	HMS
Anybus-CompactCom Hardware Design Guide	HMS
Anybus-CompactCom Software Driver User Guide	HMS
Guideline Information & Maintenance Functions	PROFIBUS Nutzerorganisation e.V. (PNO)
PROFIBUS Network Specification (IEC 61158)	IEC

P.2 Document History

Summary of Recent Changes (1.01 ... 1.02)

Change	Page(s)
Updated support information	3
Added step about node address selection in the tutorial	6
Added information and rewrote chapter 3.8 about Configuration Data Handling	14
Added information about the PROFIBUS DPV1 Object	31
Clarified information in table on buffer sizes	8

Revision List

Revision	Date	Author(s)	Chapter(s)	Description
1.00	2010-03-22	KeL	-	First official version
1.01	2011-02-10	KeL	P, 3, 4, 5	Minor updates and corrections
1.02	2012-01-26	KeL	P, 2, 3, 5	Minor update

P.3 Conventions & Terminology

The following conventions are used throughout this manual:

- Numbered lists provide sequential steps
- Bulleted lists provide information, not procedural steps
- The terms 'Anybus' or 'module' refers to the Anybus-CompactCom module.
- The term 'module' may also, depending on context, refer to a PROFIBUS entity which represents blocks of data.
- The terms 'host' or 'host application' refers to the device that hosts the Anybus module.
- Hexadecimal values are written in the format NNNNh, where NNNN is the hexadecimal value.

P.4 Sales and Support

Sales		Support	
HMS Sweden (Head Office)			
E-mail:	sales@hms-networks.com	E-mail:	support@hms-networks.com
Phone:	+46 (0) 35 - 17 29 56	Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 - 17 29 09	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com
HMS North America			
E-mail:	us-sales@hms-networks.com	E-mail:	us-support@hms-networks.com
Phone:	+1-312 - 829 - 0601	Phone:	+1-312-829-0601
Toll Free:	+1-888-8-Anybus	Toll Free:	+1-888-8-Anybus
Fax:	+1-312-629-2869	Fax:	+1-312-629-2869
Online:	www.anybus.com	Online:	www.anybus.com
HMS Germany			
E-mail:	ge-sales@hms-networks.com	E-mail:	ge-support@hms-networks.com
Phone:	+49 (0) 721-989777-000	Phone:	+49 (0) 721-989777-000
Fax:	+49 (0) 721-989777-010	Fax:	+49 (0) 721-989777-010
Online:	www.anybus.de	Online:	www.anybus.de
HMS Japan			
E-mail:	jp-sales@hms-networks.com	E-mail:	jp-support@hms-networks.com
Phone:	+81 (0) 45-478-5340	Phone:	+81 (0) 45-478-5340
Fax:	+81 (0) 45-476-0315	Fax:	+81 (0) 45-476-0315
Online:	www.anybus.jp	Online:	www.anybus.jp
HMS China			
E-mail:	cn-sales@hms-networks.com	E-mail:	cn-support@hms-networks.com
Phone:	+86 (0) 10-8532-3183	Phone:	+86 (0) 10-8532-3023
Fax:	+86 (0) 10-8532-3209	Fax:	+86 (0) 10-8532-3209
Online:	www.anybus.cn	Online:	www.anybus.cn
HMS Italy			
E-mail:	it-sales@hms-networks.com	E-mail:	it-support@hms-networks.com
Phone:	+39 039 59662 27	Phone:	+39 039 59662 27
Fax:	+39 039 59662 31	Fax:	+39 039 59662 31
Online:	www.anybus.it	Online:	www.anybus.it
HMS France			
E-mail:	fr-sales@hms-networks.com	E-mail:	fr-support@hms-networks.com
Phone:	+33 (0) 3 68 368 034	Phone:	+33 (0) 3 68 368 033
Fax:	+33 (0) 3 68 368 031	Fax:	+33 (0) 3 68 368 031
Online:	www.anybus.fr	Online:	www.anybus.fr
HMS UK & Eire			
E-mail:	uk-sales@hms-networks.com	E-mail:	support@hms-networks.com
Phone:	+44 (0) 1926 405599	Phone:	+46 (0) 35 - 17 29 20
Fax:	+44 (0) 1926 405522	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.co.uk	Online:	www.anybus.com
HMS Denmark			
E-mail:	dk-sales@hms-networks.com	E-mail:	support@hms-networks.com
Phone:	+45 (0) 35 38 29 00	Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 17 29 09	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com
HMS India			
E-mail:	in-sales@hms-networks.com	E-mail:	in-support@hms-networks.com
Phone:	+91 (0) 20 40111201	Phone:	+91 (0) 20 40111201
Fax:	+91 (0) 20 40111105	Fax:	+91 (0) 20 40111105
Online:	www.anybus.com	Online:	www.anybus.com

1. About the Anybus-CC PROFIBUS DP-V0

1.1 General

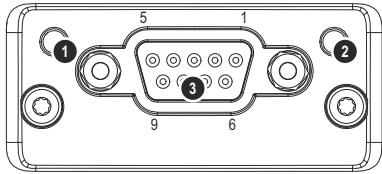
The Anybus-CompactCom PROFIBUS DP-V0 communication module provides instant PROFIBUS connectivity via the patented Anybus-CompactCom host interface. Any device that supports this standard can take advantage of the features provided by the module, allowing seamless network integration regardless of network type.

This product conforms to all aspects of the host interface for Active modules defined in the Anybus-CompactCom Hardware- and Software Design Guides, making it fully interchangeable with any other device following that specification. Generally, no additional network related software support is needed, however in order to take advantage of advanced network specific functionality, a certain degree of dedicated software support may be necessary.

1.2 Features

- Galvanically isolated bus electronics
- Automatic baud rate detection
- Generic and PROFIBUS-specific diagnostic support
- User Parametrization Data support
- Set Slave Address support
- Up to 380 bytes of I/O data
- Device identity customization
- Generic GSD-file provided

1.3 Front View

#	Item	
1	Operation Mode	
2	Status	
3	PROFIBUS Connector	

Operation Mode

State	Indication	Comments
Off	Not online / No power	-
Green	On-line, data exchange	-
Flashing Green	On-line, clear	-
Flashing Red (1 flash)	Parametrization error	See 3-12 "Parametrization Data Handling"
Flashing Red (2 flashes)	PROFIBUS Configuration error	See 3-14 "Configuration Data Handling"

Status

State	Indication	Comments
Off	No power or not initialised	Anybus state = 'SETUP' or 'NW_INIT'
Green	Initialised	Anybus module has left the 'NW_INIT' state
Flashing Green	Initialised, diagnostic event(s) present	Extended diagnostic bit is set
Red	Exception error	Anybus state = 'EXCEPTION'

PROFIBUS Connector (DB9F)

Pin	Signal	Description
1	-	-
2	-	-
3	B Line	Positive RxD/TxD, RS485 level
4	RTS	Request to send
5	GND Bus	ground (isolated)
6	+5V Bus Output ^a	+5V termination power (isolated, short-circuit protected)
7	-	-
8	A Line	Negative RxD/TxD, RS485 level
9	-	-
Housing	Cable Shield	Internally connected to the Anybus protective earth via cable shield filters according to the PROFIBUS standard.

a. The current drawn from this pin will affect the total power consumption. To simplify development, the output supplies up to 60mA when operated in room temperature (20 - 22 degrees Celsius), which is sufficient to power e.g. master simulators etc. During normal operating conditions (or higher temperatures), i.e. in an industrial environment, the specified max. current for this output is 10mA. See also D-48 "Power Consumption".

2. Tutorial

2.1 Introduction

This chapter is a complement to the Anybus CompactCom Implementation Tutorial. The ABCC tutorial describes and explains a simple example of an implementation with Anybus CompactCom. This chapter includes network specific settings that are needed for a host application to be up and running and possible to certify for use on PROFIBUS networks.

2.2 Fieldbus Conformance Notes

- When using the default settings of all parameters, the module is pre-certified for network compliance. However, any parameter changes which require deviations from the standard GSD-file supplied by HMS will require re-certification.
For further information, please contact HMS.

2.3 Certification

The following steps are necessary to perform to obtain a certification:

1. Change PNO Ident Number:

The PNO Ident Number can be requested from PNO (PROFIBUS Nutzerorganisation e.V.). Replace the default PNO Ident Number with this. This is done by implementing the Profibus DP-V1 object (FDh), instance 1, attribute 1 and returning the PNO Ident Number when receiving a Get_Attribute request.

2. Add Node Address Information:

If the host application does not set a valid node address by messaging the Network Configuration object (04h), instance 1 ("Node address"), the PROFIBUS Set Slave Address (SSA) service is enabled.

If SSA functionality is enabled, it is mandatory to provide a mechanism for resetting the node address to its default value (126). This is because it is possible to lock the value from the network side.

See "Set Slave Address" on page 15 for more information.

3. Choose Buffer Mode:

Choose a Buffer Mode that suits the application. Only 200 - 203 should be used. Implement the Profibus DP-V1 object, instance 1, attribute 6, and return the correct buffer mode when receiving a Get_Attribute request.

4. Modify the GSD-file:

Modify the Profibus DP-V0 GSD file so that it corresponds to the changes made above.

3. Basic Operation

3.1 General Information

3.1.1 Software Requirements

No additional network support code needs to be written in order to support the Anybus-CompactCom PROFIBUS DP-V0, however certain restrictions must be taken into account:

- Due to the nature of the PROFIBUS networking system, at least one ADI must be mapped to Process Data.
- By default, the module supports up to 80 bytes of Process Data. More data can be supported by changing the Buffer Mode, see 3-8 “Buffer Modes”.
- The order in which ADIs are mapped to Process Data is significant and must be replicated in the PROFIBUS master when setting up the network communication (i.e. the I/O modules must be set up in the same order, and with the same size and direction, as the mapped ADIs). If not taken into account, the network connection establishment will fail and no communication will take place.
- The use of advanced PROFIBUS-specific functionality may require in-depth knowledge in PROFIBUS networking internals and/or information from the official PROFIBUS specification (IEC 61158). In such cases, the ones responsible for the implementation of this product should either obtain the PROFIBUS specification to gain sufficient knowledge or limit their implementation in such a way that this is not necessary.

3.1.2 Electronic Data Sheet (GSD)

On PROFIBUS, the characteristics of a device is stored in an ASCII data file with the suffix GSD. This file is used by the PROFIBUS configuration tool when setting up the network.

HMS provides a generic GSD-file, which corresponds to the default settings in the module. However, due to the flexible nature of the Anybus-CompactCom concept, it is possible to alter the behaviour of the product in a way that invalidates the generic GSD-file.

See also...

- “Fieldbus Conformance Notes” on page 6
- “GSD-file Customization” on page 35

3.2 Buffer Modes

The module features several internal memory buffers which are associated with certain network entities such as Process Data, Parametrization Data, Configuration Data etc. There are several different buffer configurations, a.k.a. Buffer Modes, which determines how the available memory shall be distributed across the internal buffers. This is an essential part of the module since it affects most aspects of its communication capabilities.

Buffer Mode	Value ^a	Process Data Buffer Size (bytes) (Input + Output)	Parametrization Data Buffer Size (bytes)	Configuration Data Buffer Size (bytes)	Diagnostic Data Buffer Size (bytes)	Max. no. of Diagnostic Events
Mode 1(default)	200	80	240	80	240	10+1
Mode 2	201	200	80	80	244	10+1
Mode 3	202	300 ^b	80	80	80	7+1
Mode 4	203	380 ^b	40	40	40	3+1

a. This value should be entered in attribute 6 of the PROFIBUS-DPV1 object (FDh), see page 32.

b. On PROFIBUS, a maximum of 244 bytes can be mapped as either input or output data.

Note: The buffer size specifies the total memory area (both input and output data must be contained within its limit).

'Mode 1' (default) comply to the generic GSD-file supplied by HMS. By implementing the 'Buffer Mode' attribute (#6) of the PROFIBUS DP-V1 Object (FDh) into the host application, it is possible to redistribute the available memory to better suit particular application requirements. Note however that in such case, a custom GSD-file must be created, and fieldbus re-certification is necessary.

Note: The maximum number of Diagnostic Events are written as 'n+1', where 'n' signifies the maximum number of diagnostic instances that can be created. An extra instance ('+1') is reserved to allow a Major Unrecoverable Event to be reported at any time.

See also...

- "PROFIBUS DP-V1 Object (FDh)" on page 31 (Attribute #6, 'Buffer mode')
- "GSD-file Customization" on page 35

3.3 Communication Settings

As with other Anybus-CompactCom products, network related communication settings are grouped in the Network Configuration Object (04h).

In this case, this includes...

- **Node Address**

See also...

- “Network Configuration Object (04h)” on page 22

- **Baudrate**

The baudrate is detected automatically by the module. The following baud rates are supported:

- 9.6 kbps
- 19.2 kbps
- 45.45 kbps
- 93.75 kbps
- 187.5 kbps
- 500 kbps
- 1.5 Mbps
- 3 Mbps
- 6 Mbps
- 12 Mbps

3.4 Device Identity

By default, the Anybus module appears as a generic HMS device with the following network identity:

Vendor Name	“HMS Industrial Networks”
Model Name	“Anybus CompactCom DPV0”
Ident Number	1814h

It is possible to customize the network identity information so that the Anybus module appears as a vendor specific implementation rather than a generic HMS product. The Ident Number can be customized by the host application software or in the GSD-file. Vendor Name, Model Name, Revision, Software Revision and Hardware Revision can only be customized by changing the values in the GSD-file.

Note: Any changes to the GSD-file makes it mandatory to change the PROFIBUS ID.

See also...

- “PROFIBUS DP-V1 Object (FDh)” on page 31 , (Attribute #1, ‘PNO Ident Number’)
- “Device Identification” on page 36, (GSD-file Entries)

3.5 Data Exchange

3.5.1 Process Data

Mapping an ADI to Write Process Data results in PROFIBUS input data, and mapping an ADI to Read Process Data results in PROFIBUS output data. The maximum number of bytes that can be mapped depends on the size of the Process Data Buffer. If the host application tries to map more data than the Buffer Mode permits, the module will go into the EXCEPTION-state (exception code 06h) after 'Setup Complete'.

To guarantee consistency over an entire ADI, the ADI must not contain more than one element of a specific data type, since each element results by default in one identifier (a.k.a. 'module'). If consistency over an entire ADI which contains an array of elements is required, there are two possibilities:

- Implement the 'Configuration Data'-attribute (#3) in the PROFIBUS DP-V1 Object (FDh) and specify the configuration data manually.
- Use the network-specific ADI mapping commands in the Network PROFIBUS DP-V1 Object (0Bh)

In either case, a custom GSD-file must be created to support the size of the array.

See also...

- "Buffer Modes" on page 8
- "Network PROFIBUS DP-V1 Object (0Bh)" on page 26
- "PROFIBUS DP-V1 Object (FDh)" on page 31

3.6 Diagnostics

3.6.1 Standard Diagnostics

The Standard Diagnostics is handled automatically, with the exception of the following flags:

- **Ext Diag Overflow**

This flag can be controlled by the host application via the 'Ext diag overflow'-attribute in the PROFIBUS DP-V0 Diagnostic Object (10h), and indicates that there are pending diagnostic events which couldn't be reported.

- **Static Diag Flag**

This flag can be controlled by the host application via the 'Static Diag Flag'-attribute in the PROFIBUS DP-V0 Diagnostic Object (10h), and indicates that data from the slave is invalid.

3.6.2 Extended Diagnostics

Optionally, extended diagnostics can be supported via the Diagnostic Object (02h) or the PROFIBUS DP-V0 Diagnostic Object (10h). The attribute "Diagnostic data" in the PROFIBUS DP-V0 Diagnostic Object contains diagnostic data from the network, that is transparent to the application. The amount of data available depends on the initialized buffer mode, which gives an upper limit to the space available.

Note 1: It is not possible to use the instances of the generic Diagnostic Object (02h) and "Diagnostic data" (instance 1, attribute 1 in the PROFIBUS DP-V0 Diagnostic Object) simultaneously.

Note 2: When the PROFIBUS-master reads diagnostic data from the module, all pending events are reported; not only the ones that were recently added/removed.

See also...

- "Diagnostic Object (02h)" on page 18
- "PROFIBUS DP-V0 Diagnostic Object (10h)" on page 24

3.7 Parametrization Data Handling

3.7.1 General Information

The master identifies itself with the slaves by sending Parametrization Data, specifying how the slave shall operate (i.e. Master address, PNO-ID, Sync/Freeze capabilities etc.).

The Parametrization Data consists of two parts:

	DP Standard Parameters	User Parametrization Data
Size	7 bytes	Dynamic
Defined by	IEC 61158-6	Host application specific (optional) ^a
Evaluated by	Anybus module	Host application
Supported in the Generic HMS GSD-file	Yes	No

a. IMPORTANT: Parameter data byte 8 is reserved for the SPC3 ASIC and cannot be used by the application

As seen in the table above, User Parametrization Data is not supported by default. Optionally, User Parametrization Data can be supported by implementing the 'Parametrization Data'-attribute in the PROFIBUS Object (FDh). In such case, the generic GSD-file supplied by HMS cannot be used.

The maximum amount of User Parametrization Data that can be handled by the module is determined by the size of the Parametrization Data Buffer, see 3-8 "Buffer Modes".

See also...

- "Buffer Modes" on page 8
- "PROFIBUS DP-V1 Object (FDh)" on page 31 (Attribute #2, 'Parametrization Data')
- "Parametrization-related Keywords" on page 42

3.7.2 Validation

The DP Standard Parameters and the DP-V0 Status Bytes are always evaluated by the Anybus module, while the User Parametrization Data must be evaluated by the host application. This is handled through the 'Parametrization Data'-attribute in the PROFIBUS DP-V1 Object (FDh).

- **'Parametrization Data'-attribute not implemented**

In order for the Parametrization Data to be accepted by the module, it must not contain any User Parametrization Data.

- **'Parametrization Data'-attribute implemented**

The host application must evaluate the contents of the 'Parametrization Data'-attribute and provide a suitable response.

- To accept the Parametrization Data, respond with no error code.
- To reject the Parametrization Data, respond with one of the following error codes:
 - NOT_ENOUGH_DATA
 - TOO_MUCH_DATA
 - OUT_OF_RANGE
 - INVALID_STATE
 - NO_RESOURCES

3.8 Configuration Data Handling

3.8.1 General Information

The module determines its Expected Configuration Data based on the ADI mapping process. Alternatively, it can be specified by the host application by implementing the 'Configuration Data'-attribute in the PROFIBUS DP-V1 Object (FDh).

The maximum amount of configuration data that can be handled by the module is determined by the size of the Configuration Data Buffer, see "Buffer Modes" on page 8.

See also...

- "Buffer Modes" on page 8
- "PROFIBUS DP-V1 Object (FDh)" on page 31 (Attribute #3, 'Configuration Data')
- "I/O-related Keywords" on page 40
- "Definition of Modules" on page 41

3.8.2 Validation

Using the Chk_Cfg service, the PROFIBUS master will send the Actual Configuration Data needed for the application to the module. The module will compare the Actual Configuration Data with the Expected Configuration Data. In case of a mismatch, the module will send the Actual Configuration Data to the host application for further evaluation, using the Set service of the 'Configuration Data'-attribute in the PROFIBUS DP-V1 Object (FDh).

Implementing the 'Configuration Data'-attribute in the PROFIBUS DP-V1 Object (FDh) in the host application is optional.

- **'Configuration Data'-attribute not implemented**

In case of a mismatch, the Actual Configuration Data must be rejected.

- **'Configuration Data'-attribute implemented**

The host application must evaluate the contents of the 'Configuration Data'-attribute.

- To accept the Configuration Data, respond with a no error code.

Important: If the new configuration affects the Process Data mapping, it is important that the host application updates the Process Data before responding. Failure to observe this may cause erroneous data to be sent to the bus on the next state shift. Preferably, choose to reject the Actual Configuration Data and adapt to it by restarting the Anybus module and then revise the Process Data map and/or the Expected Configuration Data. Also note that the new configuration must exist in the GSD-file of the product.

- To reject the Configuration Data, respond with one of the following error codes:

NOT_ENOUGH_DATA
TOO_MUCH_DATA
OUT_OF_RANGE
INVALID_STATE
NO_RESOURCES

3.9 Set Slave Address

The module supports the ‘Set Slave Address’-service, which enables a master or configuration tool to set the node address from the network.

This service features a flag which specifies whether or not it is allowed to change the device address from the network again at a later stage. If the service is accepted, the module saves the value of this flag in non-volatile memory; the only way to restore it again is by performing a Factory Default-reset on the Network Configuration Object (consult the general Anybus-CompactCom Software Design Guide for more information).

The module will accept new settings received via this service under the following conditions:

- The ‘Device Address’-attribute is set to a value higher than 125
- The ‘SSA Enabled’-attribute (PROFIBUS DP-V1 Object (FDh)) is set to TRUE (or not implemented)
- The module is not in Data Exchange
- The module is addressed with the correct Ident Number
- No previous ‘Set Slave Address’-request prevents the module from accepting the new settings

See also...

- “PROFIBUS DP-V1 Object (FDh)” on page 31 (Attribute #4, ‘SSA Enabled’)
- “Supported DP Features” on page 37

Note: It is possible to disable support for this service by implementing the ‘SSA Enabled’-attribute in the PROFIBUS DP-V1 Object (FDh). In such case, a new GSD-file must be created, and fieldbus re-certification is necessary.

4. Anybus Module Objects

4.1 General Information

This chapter specifies the Anybus Module Object implementation and how they correspond to the functionality in the Anybus-CompactCom PROFIBUS DP-V1.

Standard Objects:

- “Anybus Object (01h)” on page 17
- “Diagnostic Object (02h)” on page 18
- “Network Object (03h)” on page 20
- “Network Configuration Object (04h)” on page 22

Network Specific Objects:

- “PROFIBUS DP-V0 Diagnostic Object (10h)” on page 24
- “Network PROFIBUS DP-V1 Object (0Bh)” on page 26

4.2 Anybus Object (01h)

Category

Basic

Object Description

This object assembles all common Anybus data, and is described thoroughly in the general Anybus-CompactCom Software Design Guide.

Supported Commands

Object: Get_Attribute
 Instance: Get_Attribute
 Set_Attribute
 Get_Enum_String

Object Attributes (Instance #0)

(Consult the general Anybus-CompactCom Software Design Guide for further information).

Instance Attributes (Instance #1)

Basic

#	Name	Access	Type	Value
1	Module type	Get	UINT16	0401h (Standard Anybus-CompactCom)
2... 11	-	-	-	Consult the general Anybus-CompactCom Software Design Guide for further information.
12	LED colours	Get	struct of: UINT8 (LED1A) UINT8 (LED1B) UINT8 (LED2A) UINT8 (LED2B)	<u>Value:</u> <u>Colour:</u> 01h Green 02h Red 01h Green 02h Red
13... 15	-	-	-	Consult the general Anybus-CompactCom Software Design Guide for further information.

4.3 Diagnostic Object (02h)

Category

Extended, advanced

Object Description

This object provides a standardised way of handling host application events & diagnostics, and is thoroughly described in the general Anybus-CompactCom Software Design Guide. In the case of PROFIBUS, each instance created in this object adds one Status PDU to the Extended Diagnostics.

Note: It is not possible to use the instances of the generic Diagnostic Object (02h) and “Diagnostic data” (instance 1, attribute 1 in the PROFIBUS DP-V0 Diagnostic Object) simultaneously, see also PROFIBUS DP-V0 Diagnostic Object (page 24)

Supported Commands

Object:	Get_Attribute
	Create
	Delete
Instance:	Get_Attribute

Object Attributes (Instance #0)

#	Name	Access	Data Type	Value
1... 4	-	-	-	Consult the general Anybus-CompactCom Software Design Guide for further information.
11	Max no. of instances	Get	UINT16	The maximum number of pending events/instances depends on the current Buffer Mode, see "Buffer Modes" on page 8.

Instance Attributes (Instance #1... #11)

Extended

#	Name	Access	Type	Value
1	Severity	Get	UINT8	Consult the general Anybus-CompactCom Software Design Guide for further information.
2	Event Code	Get	UINT8	

Advanced

#	Name	Access	Type	Value
3	NW specific extension ^a	Get	Array of UINT8: <u>Element:</u> 0 1 2...7	<u>Contents:</u> Slot Number (reserved) Application Specific Field

a. The use of this attribute is optional; if not implemented, the module will use slot no. 0 (zero), and the 'Severity' and 'Event Code'-attributes will be reported as application specific data. If implemented, a custom GSD-file may be required.

See also...

- "Diagnostics" on page 11
- GSD related keywords: "Status Diagnostic Messages" on page 44
- GSD related keywords: "Diagnostic-related Keywords" on page 43

4.4 Network Object (03h)

Category

Basic

Object Description

This object contains network specific data for the module. It also controls the mapping of ADIs to the process data part of the telegrams. For more information, consult the general Anybus-CompactCom Software Design Guide.

Note: The order in which ADIs are mapped to Process Data is significant and must be replicated in the PROFIBUS master when setting up the network communication.

Supported Commands

Object:	Get_Attribute
Instance:	Get_Attribute
	Set_Attribute
	Get_Enum_String
	Map_ADI_Write_Area
	Map_ADI_Read_Area

Object Attributes (Instance #0)

(Consult the general Anybus-compactCom Software Design Guide for further information).

Instance Attributes (Instance #1)

Basic

#	Name	Access	Type	Value
1	Network type	Get	UINT16	0001h
2	Network type string	Get	Array of CHAR	'PROFIBUS DP-V0'
3	Data format	Get	ENUM	0001h (MSB first)
4	Parameter Data support ^a	Get	BOOL	False
5	Write process data size	Get	UINT16	Current write process data size (in bytes) Updated on every successful Map_ADI_Write_Area ^b (and Map_ADI_Specified_Write_Area)
6	Read process data size	Get	UINT16	Current read process data size (in bytes) Updated on every successful Map_ADI_Read_Area ^b (and Map_ADI_Specified_Read_Area)
7	Exception Information	Get	UINT8	Additional PROFIBUS DP-V0-specific exception information is presented here in case the Anybus module has shifted to the EXCEPTION-state.
8... 10	-	-	-	Consult the general Anybus-CompactCom Software Design Guide for further information.

a. This attribute indicates if the network supports acyclic data services and must not be confused with PROFIBUS Parametrization Data.

b. Consult the general Anybus-CompactCom Software Design Guide for further information.

Exception Information

This attribute holds additional information when the Anybus module shifts to the EXCEPTION-state.

#	Value
00h	(no information available)
01h	Too much Configuration Data; the default Process Data map resulted in more Configuration Data than permitted by the current Buffer Mode.
02h	Too much Configuration Data; the 'Configuration Data'-attribute (PROFIBUS DP-V1 Object (FDh)) holds more data than permitted by the current Buffer Mode.
03h	Configuration error; The 'Configuration Data'-attribute does not match the actual Process Data map.
04h	Too much Process Data; the amount of mapped Process Data exceeds the capabilities of the current Buffer Mode.
05h	Implementation error; the host application has called Map_ADI_Specified_Read/Map_ADI_Specified_Write and specified Expected Configuration Data ('Configuration Data'-attribute, PROFIBUS DP-V1 Object (FDh))
06h	Implementation error; support for the Set Slave Address (SAS) telegram has been disabled ('SSA Enable'-attribute, PROFIBUS DP-V1 Object (FDh)), but no valid device address has been supplied by the application.
07h	Implementation error; invalid Buffer Mode specified ('Buffer Mode'-attribute, PROFIBUS DP-V1 Object (FDh))

4.5 Network Configuration Object (04h)

Category

Basic, advanced

Object Description

This object contains network specific configuration parameters that may be configured by the end user.

Note 1: A ‘Reset’-command towards this object will cause the module to revert all instance values to their factory default values.

Supported Commands

Object:	Get_Attribute Reset
Instance:	Get_Attribute Set_Attribute Get_Enum_String

Object Attributes (Instance #0)

(Consult the general Anybus-CompactCom Software Design Guide for further information).

Instance Attributes (Instance #1, ‘Node Address’)

Basic

The module must be assigned a unique node address (a.k.a. device address) in order to be able to communicate on the PROFIBUS network. Valid settings range from 0... 125.

Address 126 is reserved for SSA functionality, see “Set Slave Address” on page 15. This feature allows the device address to be set from the PROFIBUS master.

Note: In order to ensure fieldbus conformance, the recommendations stated in the Anybus-CompactCom Software Design Guide regarding this parameter must be followed.

#	Name	Access	Type	Description
1	Name ^a	Get	Array of CHAR	‘Node address’
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	01h (one element)
4	Descriptor	Get	UINT8	07h (get/set/shared access)
5	Value	Get/Set	UINT8	Value:Meaning: 0... 125 Node address (other) Get node address using SSA (default) (Note that support for SSA can be globally disabled by implementing the PROFIBUS object, see “PROFIBUS DP-V1 Object (FDh)” on page 31))

a. Multilingual, see “Multilingual Strings” on page 23.

Multilingual Strings

The instance names and enumeration strings in this object are multi-lingual, and are translated based on the current language settings as follows:

Instance	English	German	Spanish	Italian	French
1	Node address	Geräteadresse	Direcc nodo	Indirizzo	Adresse

4.6 PROFIBUS DP-V0 Diagnostic Object (10h)

Category

Extended

Object Description

This object provides completely transparent extended diagnostic data.

Note: Instance #1 can not be used in conjunction with the standard Diagnostic Object (page 18).

Supported Commands

Object: Get_Attribute
 Set_Attribute

Instance: Get_Attribute
 Set_Attribute

Object Attributes (Instance #0)

#	Name	Access	Data Type	Value
1	Name	Get	Array of CHAR	'PROFIBUS DP-V0 Diagnostic'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	1
4	Highest instance no.	Get	UINT16	1
12	Ext Diag Overflow	Get/Set	BOOL	<div>Value: Meaning:</div> <div>False Ext Diag Overflow is cleared (default)</div> <div>True Ext Diag Overflow is set</div> <div>See also "Standard Diagnostics" on page 11</div>
13	Static Diag	Get/Set	BOOL	<div>Value: Meaning:</div> <div>False Static Diag is cleared (default)</div> <div>True Static Diag is set</div> <div>See also "Standard Diagnostics" on page 11</div>

Instance Attributes (Instance #1)

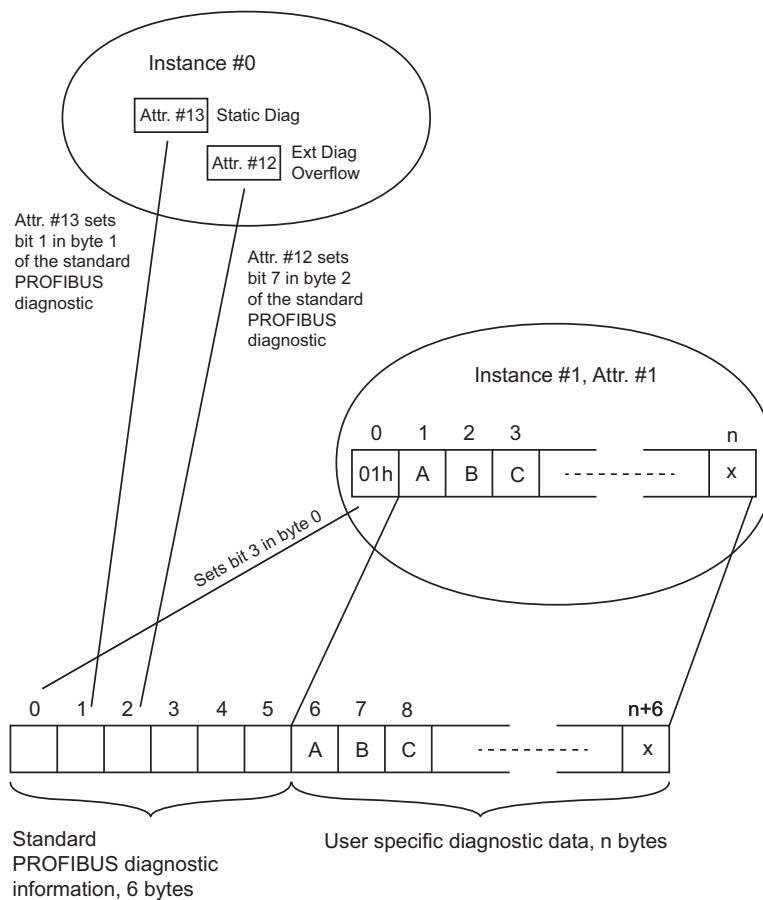
Extended

#	Name	Access	Data Type	Description
1	Diagnostic data	Get/Set	Array of UINT8	Array of bytes containing the diagnostic data. See "Diagnostic Data" on page 25.

Diagnostic Data

Byte #	Contents	Description
0	Extended Diagnostic Data Flag	<div>Value: Meaning:</div> <div>0 No extended diagnostic data</div> <div>1 Diagnostic data is reported as extended diagnostic data</div>
1 - n	Diagnostic Data	The possible amount of diagnostic data depends on the initialized buffer mode, see "Buffer Modes" on page 8. The selected buffer mode sets the total length for the diagnostic data buffer (n+6). The user specific part (n bytes) can be set with this attribute.

The picture below shows how the attributes in this object are mapped to the PROFIBUS diagnostic data telegram.¹



- Byte 0 in attribute #1 in instance #1 is a flag that indicates when extended diagnostics is available for reading. This byte is mapped to bit 3 of byte 0 in the standard PROFIBUS diagnostic data telegram.
- If the extended diagnostic data is larger than n bytes, the application must set attribute #12 in instance #0, thus setting bit 7 in byte 2 of the telegram.
- Attribute #13 in instance #0 has to be set by the application to indicate whether the extended diagnostics is static or not. If this attribute is set, bit 1 in byte 1 of the telegram is set.

1. See PROFIBUS Network Specification (IEC 61158) for information on the standard PROFIBUS diagnostic telegram.

4.7 Network PROFIBUS DP-V1 Object (0Bh)

Category

Extended

Object Description

-

Supported Commands

Object: Get_Attribute

Instance: Get_Attribute
 Map_ADI_Specified_Write_Area
 (see “Command Details: Map_ADI_Specified_Write_Area” on page 27)
 Map_ADI_Specified_Read_Area
 (see “Command Details: Map_ADI_Specified_Read_Area” on page 28)

Object Attributes (Instance #0)

#	Name	Access	Data Type	Value
1	Name	Get	Array of CHAR	“Network PROFIBUS DP-V1”
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	0001h
4	Highest instance no.	Get	UINT16	0001h

Instance Attributes

-

Command Details: Map_ADI_Specified_Write_Area

Category

Extended

Details

Command Code: 10h

Valid for: Object

Description

This command is functionally equivalent to Map_ADI_Write_Area, with the exception of certain additional PROFIBUS-specific parameters specifying custom module identifiers, which is particularly useful when mapping ADIs with multiple elements.

Note 1: Mixed calls to Map_ADI_Specified_Write_Area and Map_ADI_Write_Area is not permitted.

Note 2: The module definitions must be represented in the GSD-file.

Note 3: The order in which ADIs are mapped to Process Data is significant and must be replicated in the PROFIBUS master when setting up the network communication.

- **Command**

Field	Contents
CmdExt[0... 1]	(See specification for Map_ADI_Write_Area)
Msg_Data[0... 3]	
Msg_Data[4... n]	Module definitions; PROFIBUS identifier(s) which defines this particular ADI in the expected configuration data. The maximum number of bytes, along with module definitions for already mapped ADIs, must not exceed the size of the Configuration Data Buffer. Module definitions must agree with the size of the mapped ADI and the direction of the process data.

- **Response (Success)**

Field	Contents
Msg_Data[0]	Offset of the mapped ADI from the start of the Write Process Data.

See also...

- “Network Object (03h)” on page 20
- “Object Specific Error Codes” on page 29
- Map_ADI_Write_Area (consult the Anybus-CompactCom Software Design Guide)

Command Details: Map_ADI_Specified_Read_Area

Category

Extended

Details

Command Code: 11h

Valid for: Object

Description

This command is functionally equivalent to Map_ADI_Read_Area, with the exception of certain additional PROFIBUS-specific parameters specifying custom module identifiers, which is particularly useful when mapping ADIs with multiple elements.

Note 1: Mixed calls to Map_ADI_Specified_Read_Area and Map_ADI_Read_Area is not permitted.

Note 2: The module definitions must be represented in the GSD-file.

Note 3: The order in which ADIs are mapped to Process Data is significant and must be replicated in the PROFIBUS master when setting up the network communication.

- **Command**

Field	Contents
CmdExt[0... 1]	(See specification for Map_ADI_Read_Area)
Msg_Data[0... 3]	
Msg_Data[4... n]	Module definitions; PROFIBUS identifier(s) which defines this particular ADI in the expected configuration data. The maximum number of bytes, along with module definitions for already mapped ADIs, must not exceed the size of the Configuration Data Buffer. Module definitions must agree with the size of the mapped ADI and the direction of the process data.

- **Response (Success)**

Field	Contents
Msg_Data[0]	Offset of the mapped ADI from the start of the Read Process Data.

See also...

- “Network Object (03h)” on page 20
- “Object Specific Error Codes” on page 29
- Map_ADI_Read_Area (consult the Anybus-CompactCom Software Design Guide)

Object Specific Error Codes

Code	Meaning
01h	Invalid data type
02h	Invalid number of elements
03h	Invalid total size
04h	Invalid order number
05h	Invalid command sequence
06h	Invalid module definition
07h	Total size of expected configuration data exceeds the size of the configuration data buffer.

5. Host Application Objects

5.1 General Information

This chapter specifies the host application object implementation in the module. The objects listed here may optionally be implemented within the host application firmware to expand the PROFIBUS implementation.

Standard Objects:

- Application Object (see Anybus-CompactCom Software Design Guide)
- Application Data Object (see Anybus-CompactCom Software Design Guide)

Network Specific Objects:

- w

5.2 PROFIBUS DP-V1 Object (FDh)

Category

Basic, extended, advanced

Object Description

This object implements PROFIBUS-specific settings in the host application.

The implementation of this object is optional; the host application can support none, some, or all of the attributes specified below. The module will attempt to retrieve the values of these attributes during start-up; if an attribute is not implemented in the host application, simply respond with an error message (06h, "Invalid CmdExt[0]"). In such case, the module will use its default value.

If the module attempts to retrieve a value of an attribute not listed below, respond with an error message (06h, "Invalid CmdExt[0]").

Note 1: During operation, the host application must always be able to respond to requests from the module. Respond either with the requested data or an adequate error message. Never leave a request from the module unattended.

Note 2: Altering the default settings within this object may require a new GSD-file, which in turn requires fieldbus re-certification.

See also...

- "Front View" on page 5
- "Electronic Data Sheet (GSD)" on page 7
- "GSD-file Customization" on page 35
- Anybus CompactCom Software Design Guide, "Error Codes"

Supported Commands

Object: Get_Attribute

Instance: Get_Attribute
Set_Attribute

Object Attributes (Instance #0)

#	Name	Access	Data Type	Value
1	Name	Get	Array of CHAR	'PROFIBUS DP-V1'
2	Revision	Get	UINT8	02h
3	Number of instances	Get	UINT16	0001h
4	Highest instance no.	Get	UINT16	0001h

Instance Attributes (Instance #1)

Basic

#	Name	Access	Type	Comment										
1	PNO Ident Number	Get	UINT16	Ident Number (default =1814h) See also... <ul style="list-style-type: none">- "Device Identity" on page 9- "Device Identification" on page 36										
6	Buffer mode	Get	UINT8	This attribute specifies the Buffer Mode: <table><tr><td><u>Value:</u></td><td><u>Meaning:</u></td></tr><tr><td>200</td><td>Mode 1 (default)</td></tr><tr><td>201</td><td>Mode 2</td></tr><tr><td>202</td><td>Mode 3</td></tr><tr><td>203</td><td>Mode 4</td></tr></table> <p>Note: Other values will cause the module to enter the EXCEPTION-state (exception code 07h).</p> See also... <ul style="list-style-type: none">- "Buffer Modes" on page 8	<u>Value:</u>	<u>Meaning:</u>	200	Mode 1 (default)	201	Mode 2	202	Mode 3	203	Mode 4
<u>Value:</u>	<u>Meaning:</u>													
200	Mode 1 (default)													
201	Mode 2													
202	Mode 3													
203	Mode 4													

Extended

#	Name	Access	Type	Comment						
3	Configuration Data	Get/Set	Array of UINT8	<p>This attribute is used both to specify the Expected Configuration Data, and for evaluation of the Actual Configuration Data. See also...</p> <ul style="list-style-type: none">- “Configuration Data Handling” on page 14- “Definition of Modules” on page 41 <p>Note: If this attribute doesn’t match the Process Data map, or if it doesn’t fit into the Configuration Data Buffer, the module will enter the EXCEPTION-state (exception code 06h)</p>						
4	SSA Enabled	Get	BOOL	<p>This attribute enables/disables ‘Set Slave Address’-support.</p> <table><tr><td><u>Value:</u></td><td><u>Meaning:</u></td></tr><tr><td>True</td><td>Enabled (default)</td></tr><tr><td>False</td><td>Disabled</td></tr></table> <p>See also...</p> <ul style="list-style-type: none">- “Set Slave Address” on page 15- “Supported DP Features” on page 37	<u>Value:</u>	<u>Meaning:</u>	True	Enabled (default)	False	Disabled
<u>Value:</u>	<u>Meaning:</u>									
True	Enabled (default)									
False	Disabled									

Advanced

#	Name	Access	Type	Comment
2	Parametrization Data	Set	Array of UINT8	The module attempts to forward the Parametrization Data to this attribute during network startup. See also... - "Parametrization Data Handling" on page 12 - "Diagnostic-related Keywords" on page 43

A. Categorization of Functionality

The objects, including attributes and services, of the Anybus CompactCom and the application are divided into three categories: basic, advanced and extended.

A.1 Basic

This category includes objects, attributes and services that are mandatory to implement or to use. They will be enough for starting up the Anybus CompactCom and sending/receiving data with the chosen network protocol. The basic functions of the industrial network are used.

Additional objects etc, that will make it possible to certify the product also belong to this category.

A.2 Extended

Use of the objects in this category extends the functionality of the application. Access is given to the more specific characteristics of the industrial network, not only the basic moving of data to and from the network. Extra value is given to the application.

A.3 Advanced

The objects, attributes and services that belong to this group offer specialized and/or seldom used functionality. Most of the available network functionality is enabled and accessible. Access to the specification of the industrial network is normally required.

B. Implementation Details

B.1 SUP-Bit Definition

The supervised bit (SUP) indicates that the network participation is supervised by another network device. For PROFIBUS, this bit is set when any of the following conditions are fulfilled.

- Parametrization and Configuration Data has been accepted (i.e. MS0 connection established)
- An MS2 connection is open

B.2 Anybus State Machine

The table below describes how the Anybus state machine relates to the PROFIBUS network.

State	Description
WAIT_PROCESS	No MS0 connection DP state = Power-On/WaitPm/WaitCfg
ERROR	Not used
PROCESS_ACTIVE	Master Mode = Operate DP State = DataExchange MS0 connection established
IDLE	Master Mode = Clear DP State = DataExchange MS0 connection established
EXCEPTION	MS0, MS1 and MS2 connections will be closed. The module will enter this state in the following cases: <ul style="list-style-type: none">- Invalid Device Address and 'SSA Enabled'=FALSE- Invalid Buffer Mode- Size of 'Configuration Data'-attribute is larger than the size of the Configuration Data Buffer- Major Unrecoverable event created in Diagnostic Object- Configuration Data does not match the mapped Process Data

B.3 Watchdog Behaviour (Application Stopped)

If the application watchdog expires, the module will enter the 'EXCEPTION' state, terminate all open PROFIBUS connections and leave the network.

C. GSD-file Customization

C.1 General

The GSD-file specifies the characteristics of the device, and is used by the PROFIBUS configuration tool when setting up the network.

HMS provides a generic GSD-file, which corresponds to the default settings in the module. However, due to the flexible nature of the Anybus-CompactCom concept, it is possible to alter the behavior of the product in a way that invalidates the generic GSD-file. In such case, a custom GSD-file must be created, and fieldbus re-certification is necessary.

This chapter is intended to provide a brief overview of the GSD entries that may need alteration, and how they correspond to settings within the Anybus module. Some of the entries should not be changed, and the others are divided in the same way as the objects and object attributes, into the groups Basic, Extended and Advanced.

For further information, consult the Specification for PROFIBUS Device Description and Device Integration Volume 1: GSD (order. no. 2.122).

Note: The user is expected to have sufficient knowledge in the PROFIBUS networking system to understand the concepts involved when performing the changes specified in this chapter. In case of uncertainties, send the customized GSD-file to HMS for verification.

C.2 Device Identification

General

By default, the module will appear as a generic Anybus implementation ('Anybus CompactCom DPV0') from HMS Industrial Networks (PROFIBUS ident no. 1814h).

However, the identity of the module can be customized to appear as a vendor specific implementation by creating a custom GSD-file and implementing the 'PNO Ident Number'-attribute in the PROFIBUS DP-V1 Object (FDh).

Contact PNO to obtain a unique Ident Number.

GSD-file Entries

```
; Device identification
Vendor_Name      = "<vendor>"
Model_Name       = "<product>"
Revision         = "<prod_rev>"
Ident_Number     = "<ident_no>"
Protocol_Ident   = 0                ; DP protocol
Station_Type     = 0                ; Slave device
FMS_supp         = 0                ; FMS not supported
Slave_Family     = 0                ; General device
Hardware_Release = "Version <hw_rev>"
Software_Release = "Version <sw_rev>"
```

Setting	Description
<vendor>	Vendor name as text (e.g. "HMS Industrial Networks")
<product>	Product name as text (e.g. "Anybus CompactCom DPV0")
<prod_rev>	Product revision (major.minor) (e.g. "1.01")
<ident_no>	PNO Ident Number in HEX. Written as 0xNNNN, where NNNN is the hexadecimal value.
<hw_rev>	Hardware revision (major.minor) (e.g. "Version 1.00")
<sw_rev>	Software revision (major.minor) (e.g. "Version 1.00")

Related Information

Information	Page(s)
Device Identity	9
PROFIBUS DP-V1 Object (FDh) (Attribute #1)	31
-	-

C.3 Supported Hardware Features

General

Do not change the standard settings.

GSD-file Entries

```
; Supported hardware features
Redundancy          = 0          ; not supported
Repeater_Ctrl_Sig = 2          ; TTL
24V_Pins            = 0          ; not connected
Implementation_Type= "NP30"
```

Setting	Unit	Description
-	-	-

Related Information

Information	Page(s)
-	-

C.4 Supported DP Features

General

-

C.4.1 GSD-file Entries

```
; Supported DP features
Freeze_Mode_supp = 1
Sync_Mode_supp   = 1
Auto_Baud_supp   = 1
Set_Slave_Add_supp= <SSA>
Fail_Safe        = 1
```

Setting	Description
<SSA>	This value must be set to match the 'SSA enable'-attribute in the PROFIBUS DP-V1 Object (FDh): 0: 'SSA enabled'-attribute set to FALSE 1: 'SSA enabled'-attribute set to TRUE (or not implemented)

Related Information

Information	Page(s)
Set Slave Address	15
PROFIBUS DP-V1 Object (FDh) (Attribute #4)	31
-	-

C.5 Supported Baudrates

General

Do not change the standard settings.

GSD-file Entries

```
; Supported baudrates
9.6_supp      = 1
19.2_supp     = 1
45.45_supp    = 1
93.75_supp    = 1
187.5_supp    = 1
500_supp      = 1
1.5M_supp     = 1
3M_supp       = 1
6M_supp       = 1
12M_supp      = 1
```

Setting	Unit	Description
-	-	-

Related Information

Information	Page(s)
-	-

C.6 Maximum Responder Time for Supported Baudrates

General

Do not change the standard settings.

GSD-file Entries

```
; Maximum responder time for supported baudrates
MaxTsdr_9.6      = 15
MaxTsdr_19.2     = 15
MaxTsdr_45.45    = 15
MaxTsdr_93.75    = 15
MaxTsdr_187.5    = 15
MaxTsdr_500      = 15
MaxTsdr_1.5M     = 25
MaxTsdr_3M       = 50
MaxTsdr_6M       = 100
MaxTsdr_12M      = 200
```

Setting	Unit	Description
-	-	-

Related Information

Information	Page(s)
-	-

C.7 Maximum Polling Frequency

General

Do not change the standard settings.

GSD-file Entries

```
; Maximum polling frequency
Min_Slave_Intervall= 1 ; 0.1 ms
```

Setting	Unit	Description
-	-	-

Related Information

Information	Page(s)
-	-

C.8 I/O-related Keywords

General

-

GSD-file Entries

```

; I/O related keywords
Modular_Station  = 1
Max_Module       = <module>
Max_Input_Len    = <input>
Max_Output_Len   = <output>
Max_Data_Len     = <total>
Modul_Offset     = 1

```

Basic

Setting	Unit	Description															
<module>	bytes	This value must be set within the range imposed by the current Buffer Mode as follows: <table> <tr> <th>Buffer Mode:</th><th>Min:</th><th>Max:</th></tr> <tr> <td>Mode 1</td><td>1</td><td>80</td></tr> <tr> <td>Mode 2</td><td>1</td><td>80</td></tr> <tr> <td>Mode 3</td><td>1</td><td>80</td></tr> <tr> <td>Mode 4</td><td>1</td><td>40</td></tr> </table>	Buffer Mode:	Min:	Max:	Mode 1	1	80	Mode 2	1	80	Mode 3	1	80	Mode 4	1	40
Buffer Mode:	Min:	Max:															
Mode 1	1	80															
Mode 2	1	80															
Mode 3	1	80															
Mode 4	1	40															
<input>	bytes	This value must be set within the range imposed by the current Buffer Mode as follows: <table> <tr> <th>Buffer Mode:</th><th>Min:</th><th>Max:</th></tr> <tr> <td>Mode 1</td><td>0</td><td>80</td></tr> <tr> <td>Mode 2</td><td>0</td><td>200</td></tr> <tr> <td>Mode 3</td><td>0</td><td>244</td></tr> <tr> <td>Mode 4</td><td>0</td><td>244</td></tr> </table>	Buffer Mode:	Min:	Max:	Mode 1	0	80	Mode 2	0	200	Mode 3	0	244	Mode 4	0	244
Buffer Mode:	Min:	Max:															
Mode 1	0	80															
Mode 2	0	200															
Mode 3	0	244															
Mode 4	0	244															
<output>	bytes	This value must be set within the range imposed by the current Buffer Mode as follows: <table> <tr> <th>Buffer Mode:</th><th>Min:</th><th>Max:</th></tr> <tr> <td>Mode 1</td><td>0</td><td>80</td></tr> <tr> <td>Mode 2</td><td>0</td><td>200</td></tr> <tr> <td>Mode 3</td><td>0</td><td>244</td></tr> <tr> <td>Mode 4</td><td>0</td><td>244</td></tr> </table>	Buffer Mode:	Min:	Max:	Mode 1	0	80	Mode 2	0	200	Mode 3	0	244	Mode 4	0	244
Buffer Mode:	Min:	Max:															
Mode 1	0	80															
Mode 2	0	200															
Mode 3	0	244															
Mode 4	0	244															
<total>	bytes	This value must be set within the range imposed by the current Buffer Mode as follows: <table> <tr> <th>Buffer Mode:</th><th>Min:</th><th>Max:</th></tr> <tr> <td>Mode 1</td><td>1</td><td>80</td></tr> <tr> <td>Mode 2</td><td>1</td><td>200</td></tr> <tr> <td>Mode 3</td><td>1</td><td>300</td></tr> <tr> <td>Mode 4</td><td>1</td><td>380</td></tr> </table>	Buffer Mode:	Min:	Max:	Mode 1	1	80	Mode 2	1	200	Mode 3	1	300	Mode 4	1	380
Buffer Mode:	Min:	Max:															
Mode 1	1	80															
Mode 2	1	200															
Mode 3	1	300															
Mode 4	1	380															

Related Information

Information	Page(s)
Buffer Modes	8
-	-

C.9 Definition of Modules

General

These parameters generally only need to be altered if customizing module names, if using network-specific ADI mapping commands (Map_ADI_Specified_Read_Area and Map_ADI_Specified_Write_Area), or if the Configuration Data attribute (“PROFIBUS DP-V1 Object (FDh)” on page 31) has been implemented.

GSD-file Entries

```
; Definition of modules
Module = "<name>" <identifier>
<module_id>
EndModule
```

Extended

Setting	Description
<name>	Name of module
<identifier>	Configuration Identifier; hexadecimal value (written as 0xNN where NN is the hexadecimal value) specifying the properties of the module (see below).
<module_id>	Decimal number, must be unique for each module.

Identifier Explanation

b7	b6	b5	b4	b3	b2	b1	b0	Contents	Usage
■	■	■	■	■	■	■	■	Number of configured data units	<div>Value: Meaning:</div> <div>0000 1 unit</div> <div>0001 2 units</div> <div>0010 3 units</div> <div>... ..</div> <div>1111 16 units</div>
									<div>Value: Meaning:</div> <div>00 Special Format, see note.</div> <div>01 Input</div> <div>10 Output</div> <div>11 Input and Output</div>
									<div>Value: Meaning:</div> <div>0 Byte</div> <div>1 Word</div>
									<div>Value: Meaning:</div> <div>0 Consistency over unit</div> <div>1 Consistency over module</div>

Note: Advanced users may want to specify modules using the special (extended) ID format. Exactly how this is done is beyond the scope of this document.

Related Information

Information	Page(s)
-	-

C.10 Parametrization-related Keywords

General

These parameters generally only need to be altered in advanced implementations which requires the use of User Parameterization Data. The details about such implementations are beyond the scope of this document and requires in-depth knowledge in the PROFIBUS networking system.

GSD-file Entries

```
; Parametrization related keywords
Max_User_Prm_Data_Len      = <up_len>
Ext_User_Prm_Data_Const(0) = <up_data>
```

Default setting of Max_User_Prm_Data_Len is 0. If the user wants to make use of User Parameterization Data this parameter must be changed to:

```
Max_User_Prm_Data_Len = (Size of Prm data attr - 7)
```

Setting	Unit	Description															
<up_len>	bytes	Size of User Parameterization Data, must be set within the range imposed by the current Buffer Mode as follows: <div> <table> <tr> <th>Buffer Mode:</th><th>Min:</th><th>Max:</th></tr> <tr> <td>Mode 1</td><td>0</td><td>233</td></tr> <tr> <td>Mode 2</td><td>0</td><td>73</td></tr> <tr> <td>Mode 3</td><td>0</td><td>73</td></tr> <tr> <td>Mode 4</td><td>0</td><td>33</td></tr> </table> </div>	Buffer Mode:	Min:	Max:	Mode 1	0	233	Mode 2	0	73	Mode 3	0	73	Mode 4	0	33
Buffer Mode:	Min:	Max:															
Mode 1	0	233															
Mode 2	0	73															
Mode 3	0	73															
Mode 4	0	33															
<up_data>	-	Actual User Parameterization Data as hexadecimal value															

Related Information

Information	Page(s)
Buffer Modes	8
Parametrization Data Handling	12
PROFIBUS DP-V1 Object (FDh) (Attribute #2)	31

C.11 Diagnostic-related Keywords

General

-

GSD-file Entries

```
; Diagnostic related keywords
Max_Diag_Data_Len = <diag_len>
```

Setting	Unit	Description	
<diag_len>	bytes	This value must be set to match the current Buffer Mode as follows:	
		<u>Buffer Mode:</u>	<u>Value:</u>
		Mode 1	240
		Mode 2	244
		Mode 3	80
		Mode 4	40

Related Information

Information	Page(s)
Buffer Modes	8
Diagnostics	11
Diagnostic Object (02h)	18
PROFIBUS DP-V0 Diagnostic Object (10h)	24
PROFIBUS DP-V1 Object (FDh) (Attribute #5)	31

C.12 Status Diagnostic Messages

General

These settings may need alteration when the 'NW specific extension' of the Diagnostic Object is used. It is generally recommended to remove Diagnostic Codes which are not used by the implementation (e.g. remove 'Value (48) = "Voltage"' if this code is not applicable for the end product).

These keywords are only applicable if the standard diagnostics object is used.

GSD-file Entries

```
;Status diagnostic messages
Unit_Diag_Area=16-17
Value(0)           = "Status not changed"
Value(1)           = "Status appears"
Value(2)           = "Status disappears"
Unit_Diag_Area_End

Unit_Diag_Area     = <start>-<end>
Value(<val>)        = "<text>"
Value(<val>)        = "<text>"
Value(<val>)        = "<text>"
...
Value(<val>)        = "<text>"
Unit_Diag_Area_End

Unit_Diag_Area     = <start>-<end>
Value(<val>)        = "<text>"
Value(<val>)        = "<text>"
Value(<val>)        = "<text>"
...
Value(<val>)        = "<text>"
Unit_Diag_Area_End

...

Unit_Diag_Area     = <start>-<end>
Value(<val>)        = "<text>"
Value(<val>)        = "<text>"
Value(<val>)        = "<text>"
...
Value(<val>)        = "<text>"
Unit_Diag_Area_End
```

Setting	Description
<start>	These settings specify the bit range to associated with <val> and <text>.
<end>	
<val>	Value
<text>	String associated with the value of <val>

Related Information

Information	Page(s)
Diagnostics	11
Diagnostic Object (02h)	18
-	-

C.13 DP-V1 Keywords

General

Do not change this entry

GSD-file Entries

```
; DPV1 related keywords
DPV1_Slave      = 0
```


Appendix D

D. Technical Specification

D.1 Protective Earth (PE) Requirements

In order to ensure proper EMC behaviour, the module must be properly connected to protective earth via the PE pad / PE mechanism described in the general Anybus-CompactCom Hardware Design Guide.

HMS Industrial Networks does not guarantee proper EMC behaviour unless these PE requirements are fulfilled.

D.2 Power Supply

Supply Voltage

The module requires a regulated 3.3V power source as specified in the general Anybus-CompactCom Hardware Design Guide.

Power Consumption

The Anybus-CompactCom PROFIBUS DP-V0 is designed to fulfil the requirements of a Class A module. For more information about the power consumption classification used on the Anybus-CompactCom platform, consult the general Anybus-CompactCom Hardware Design Guide.

The current hardware design consumes up to 230mA¹².

Note: It is strongly advised to design the power supply in the host application based on the power consumption classifications described in the general Anybus-CompactCom Hardware Design Guide, and not on the exact power requirements of a single product.

D.3 Environmental Specification

Consult the Anybus-CompactCom Hardware Design Guide for further information.

D.4 EMC Compliance

Consult the Anybus-CompactCom Hardware Design Guide for further information.

-
1. Note that in line with HMS policy of continuous product development, we reserve the right to change the exact power requirements of this product without prior notification. Note however that in any case, the Anybus-CompactCom PROFIBUS DP-V0 will remain as a Class A module.
 2. This value is valid under the condition that no current is being drawn from bus connector pin 6 (+5V termination power; see 1-5 "PROFIBUS Connector (DB9F)").

E. Timing & Performance

E.1 General Information

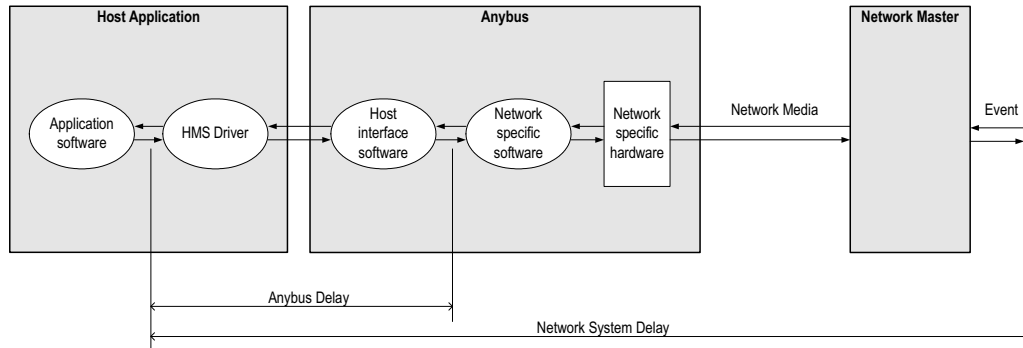
This chapter specifies timing and performance parameters that are verified and documented for the Anybus CompactCom PROFIBUS DP-V0.

The following timing aspects are measured:

Category	Parameters	Page
Startup Delay	T1, T2	Please consult the Anybus CompactCom Software Design Guide, App. B.
NW_INIT Delay	T3	
Telegram Delay	T4	
Command Delay	T5	
Anybus Read Process Data Delay (Anybus Delay)	T6, T7, T8	
Anybus Write Process Data Delay (Anybus Delay)	T12, T13, T14	
Network System Read Process Data Delay (Network System Delay)	T9, T10, T11	51
Network System Write Process Data Delay (Network System Delay)	T15, T16, T17	51

E.2 Process Data

E.2.1 Overview



E.2.2 Anybus Read Process Data Delay (Anybus Delay)

The Read Process Data Delay (labelled ‘Anybus delay’ in the figure above) is defined as the time measured from just before new data is buffered and available to the Anybus host interface software, to when the data is available to the host application (just after the new data has been read from the driver).

Please consult the Anybus CompactCom Software Design Guide, Appendix B, for more information.

E.2.3 Anybus Write Process Data Delay (Anybus Delay)

The Write Process Data Delay (labelled ‘Anybus delay’ in the figure) is defined as the time measured from the point the data is available from the host application (just before the data is written from the host application to the driver), to the point where the new data has been forwarded to the network buffer by the Anybus host interface software.

Please consult the Anybus CompactCom Software Design Guide, Appendix B, for more information.

E.2.4 Network System Read Process Data Delay (Network System Delay)

The Network System Read Process Data Delay (labelled 'Network System Delay' in the figure), is defined as the time measured from the point where an event is generated at the network master to when the corresponding data is available to the host application (just after the corresponding data has been read from the driver).

Parameter	Description	Avg.	Max.	Unit.
T9	Network System Read Process Data delay, 8 ADIs (single UINT8)	5	7	ms
T10	Network System Read Process Data delay, 16 ADIs (single UINT8)	6	7.2	ms
T11	Network System Read Process Data delay, 32 ADIs (single UINT8)	6	7.2	ms

Conditions:

Parameter	Conditions
Application CPU	-
Timer system call interval	1 ms
Driver call interval	0.2... 0.3 ms
No. of ADIs (single UINT8) mapped to Process Data in each direction.	8, 16 and 32
Communication	Parallel
Telegram types during measurement period	Process Data only
Profibus baud rate	12 Mbit/s
Bus load, no. of nodes etc.	Normal

E.2.5 Network System Write Process Data Delay (Network System Delay)

The Network System Write Process Data Delay (labelled 'Network System Delay' in the figure), is defined as the time measured from the time after the new data is available from the host application (just before the data is written to the driver) to when this data generates a corresponding event at the network master.

Parameter	Description	Avg.	Max.	Unit.
T15	Network System Write Process Data delay, 8 ADIs (single UINT8)	2.5	3.5	ms
T16	Network System Write Process Data delay, 16 ADIs (single UINT8)	2.5	3.5	ms
T17	Network System Write Process Data delay, 32 ADIs (single UINT8)	2.5	3.5	ms

Conditions: as in "Network System Read Process Data Delay (Network System Delay)" on page 51.