


Fieldbus Appendix

AnyBus-S DeviceNet

Rev. 2.08

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- 1908 - 359301



sales-ge@hms-networks.com
sales-jp@hms-networks.com
sales@hms-networks.com
sales-us@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 DeviceNet	
	Features.....	1-1
	Compatible Products	1-1
	EDS File	1-1
	Conformance Note	1-1
	Overview	1-2
	<i>Connectors</i>	1-2
	<i>Indicators</i>	1-3
Chapter 2	Configuration	
	General.....	2-1
	Network Settings.....	2-1
	<i>Mac ID</i>	2-1
	<i>Baudrate</i>	2-1

Chapter 3	Memory Structure	
	Overview	3-1
	I/O Input and Output Data Areas.....	3-2
	Parameter Input and Output Data Areas	3-2
Chapter 4	Implemented Objects	
	Identity Object, Class 01h.....	4-2
	Message router, Class 02h.....	4-2
	DeviceNet Object, Class 03h	4-3
	Assembly Object, Class 04h	4-4
	DeviceNet Connection Object, Class 05h	4-5
	Acknowledge Handler Object, Class 2Bh	4-9
	I/O data Input Mapping Object, Class A0h.....	4-10
	I/O data Output Mapping Object, Class A1h.....	4-10
	Diagnostic Object, Class AAh.....	4-11
	Parameter data Input Mapping Object, Class B0h.....	4-12
	Parameter data Output Mapping Object, Class B1h	4-12
Chapter 5	Mailbox Interface	
	Summary.....	5-1
	Configuration Switch.....	5-2
	<i>Set DipSwitch (SET_DIPSWITCH)</i>	5-2
	<i>Set DipSwitch 2 (SET_DIPSWITCH_2)</i>	5-3
	<i>Get DipSwitch (GET_DIPSWITCH)</i>	5-4
	Mapping.....	5-5
	<i>Parameter Data Input Area Mapping (PARAMETER_INPUT_MAP)</i>	5-5
	<i>Parameter Data Output Area Mapping (PARAMETER_OUTPUT_MAP)</i>	5-7
	<i>I/O Data Input Area Mapping (IO_INPUT_MAP)</i>	5-9
	<i>I/O Data Output Area Mapping (IO_OUTPUT_MAP)</i>	5-11
	Other.....	5-13
	<i>Set Product Info (PRODUCT_INFO)</i>	5-13
	<i>Set Product Info All (PRODUCT_INFO_ALL)</i>	5-15
Chapter 6	Fieldbus Specific Area	
	<i>Identity Status Attribute (Read Only)</i>	6-1
	<i>Connection Status Registers (Read Only)</i>	6-2
	<i>Master Status (Read Only)</i>	6-2

Appendix A Electrical Characteristics

Supply Voltage.....	A-1
Power Consumption.....	A-1
PE Grounding.....	A-1
Galvanic Isolation.....	A-1
Network Termination.....	A-1

Appendix B Connectors

Application Interface.....	B-1
Fieldbus Interface.....	B-1

Appendix C Environmental Specification

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

Appendix D Mechanical Specification

Measurements, Connectors & Switches.....	D-2
<i>Angled Switches & Connectors</i>	D-2
<i>Straight Switches & Connectors</i>	D-3

About This Manual

How To Use This Manual

This document is intended to be used in conjunction with the AnyBus-S Serial / Parallel Design Guide. For general information about the AnyBus-S platform, consult the general AnyBus-S Serial / Parallel 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 DeviceNet fieldbus, 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
AnyBus-S Serial Design Guide	HMS
DeviceNet Specification	ODVA

Revision List

Revision	Date	Author	Chapter	Description
2.00	2002-09-10	PeP	All	Created
2.01	2003-03-19	TTh	4	Corrected number of instances in assembly object
2.02	2003-09-05	PeP	All Appendix D	Various cosmetic updates Updated measurements
2.03	2003-09-30	TTh	Appendix A	Correction of the 'Galvanic Isolation' section
2.04	2003-09-30	TTh	6	Connection status value of 02h is not used, should be 03h
2.05	2003-11-10	PeP	4	Added missing attributes in DeviceNet and Assembly Connection objects.
2.06	2004-02-12	TTh	5 4	Removed obsolete message 'set product code'. Corrected hex/dec errors, removed obsolete attributes.
2.07	2004-08-10	PeP	Appendix D	Corrected measurements + miscellaneous minor updates
2.08	2005-11-08	ToT	Indicators Supply Voltage	Corrected LED block figure. Added bus supply voltage limits.

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
- The term ‘module’ is used when referring to the AnyBus-S DeviceNet module
- 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, where NNNN is the hexadecimal value.
- All measurements expressed in this document have a tolerance of $\pm 0.25\text{mm}$ unless otherwise stated.

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 DeviceNet

The AnyBus-S DeviceNet module is implemented according to the ODVA specification for a communication adapter (profile no 12) and acts as a “group two only server” on the DeviceNet network.

Features

- Both parallel and serial versions available
- MacID and Baudrate can be set via onboard switches or application interface
- Explicit Messaging
- Polled I/O
- Bit-strobed I/O
- Change-of-state / Cyclic I/O
- Galvanically isolated bus electronics

Compatible Products

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

However, due to the nature of the DeviceNet fieldbus system, the AnyBus-S DeviceNet requires that no mailbox commands are sent for at least 2 seconds after receiving the END_INIT response.

EDS File

Each device in a DeviceNet network is associated with an EDS file, containing all necessary information about the device. This file is used by the network configuration tool during network configuration.

The latest version of EDS file can either be downloaded from HMS website (www.hms-networks.com) or received by contacting HMS.

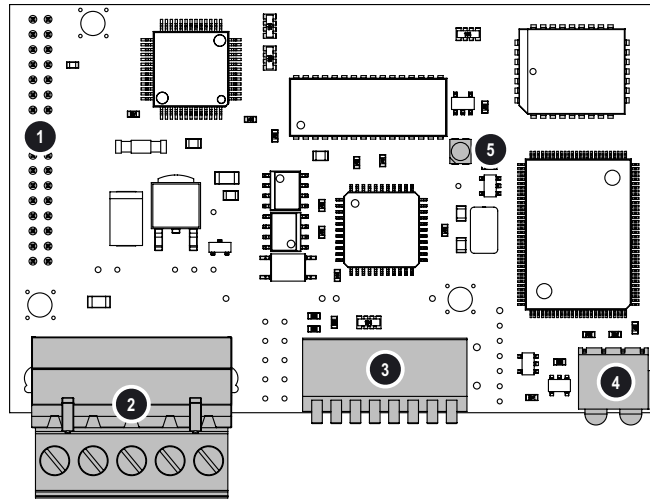
Conformance Note

The AnyBus-S DeviceNet has been tested standalone by ODVA’s authorized Independent Test Lab and found to comply with ODVA Conformance Test Software Version A12.

However, according to ODVA’s conformance test policy, the final product must still be re-certified as a complete product even if the AnyBus module is certified.

For more information regarding conformance testing, please contact HMS or ODVA.

Overview



#	Description
1	Application Connector
2	DeviceNet Connector
3	Configuration Switches
4	Status Indicators
5	AnyBus-S Watchdog

Connectors

Application Connector

The module exists in both parallel and serial versions.

- **Serial version**
Standard AnyBus-S SCI serial interface.
For further information, please consult the general AnyBus-S Serial Design Guide.
- **Parallel version**
Standard AnyBus-S DPM 2kbyte DPRAM interface
For further information, please consult the general AnyBus-S Parallel Design Guide.

DeviceNet Connector

The module supports both 5.08mm and 3.84mm pluggable screw connectors, as well as a 10 pin 2mm board to board connector. For more information about connectors and pin assignments, see Appendix B-1 “Connectors”

Configuration Switches

These switches can be used to set the DeviceNet Mac ID and baudrate settings, see 2-1 “Network Settings”. Note that these settings can also be configured via the application interface using mailbox commands, see 5-2 “Configuration Switch”.

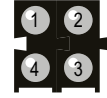
Indicators

AnyBus-S Watchdog

(Consult the general AnyBus-S Design Guides for further information)

Status Indicators

These leds indicate run time status and errors to the user.



Led	State	Description
1 - Reserved	-	Reserved for future use
2 - Network Status	Off	Not powered / Not online
	Green, steady	Link OK, On line, Connected
	Green, flashing	On line, Not connected
	Red, steady	Critical link failure
	Red, flashing	Connection timeout
3 - Module Status	Off	No power to device
	Green, steady	Device operational
	Green, flashing	Data size bigger than configured
	Red, steady	Unrecoverable fault
	Red, flashing	Minor fault
4 - Reserved	-	Reserved for future use

Configuration

General

A connection to the module is set up towards the Assembly object with a bus configuration tool or directly from a master.

Network Settings

On a DeviceNet network, each node must be assigned its own unique Mac ID. The Mac ID is a value between 0 and 63 used to identify each node. The Mac ID and Baud rate settings can be set either using a physical DIP-switch or via the application interface using mailbox commands (See 5-2 “Configuration Switch”).

The DIP switches are numbered 1 through 8. Switch 1 and 2 are used to configure the Baud rate, and switches 3 through 8 are used to configure the Mac ID.

Depending on how the module is mounted, it may be required to change the bit definition of the switch in order to comply with the DeviceNet specification. (e.g. if the module is mounted vertically in a product). The module provides a set of mailbox commands for reading and setting the DIP switch value, allowing the application to read out the physical value of the switch, alter it as required, and write it back to the module.

Mac ID

Switches 3 through 8 are used to set the Mac ID

Mac ID	sw. 3 (MSB)	sw. 4	sw. 5	sw. 6	sw. 7	sw. 8 (LSB)
0	OFF	OFF	OFF	OFF	OFF	OFF
1	OFF	OFF	OFF	OFF	OFF	ON
2	OFF	OFF	OFF	OFF	ON	OFF
3	OFF	OFF	OFF	OFF	ON	ON
...
62	ON	ON	ON	ON	ON	OFF
63	ON	ON	ON	ON	ON	ON

Baudrate

There are three different baudrates for DeviceNet; 125k, 250k, 500kbit/s, see table on the right.

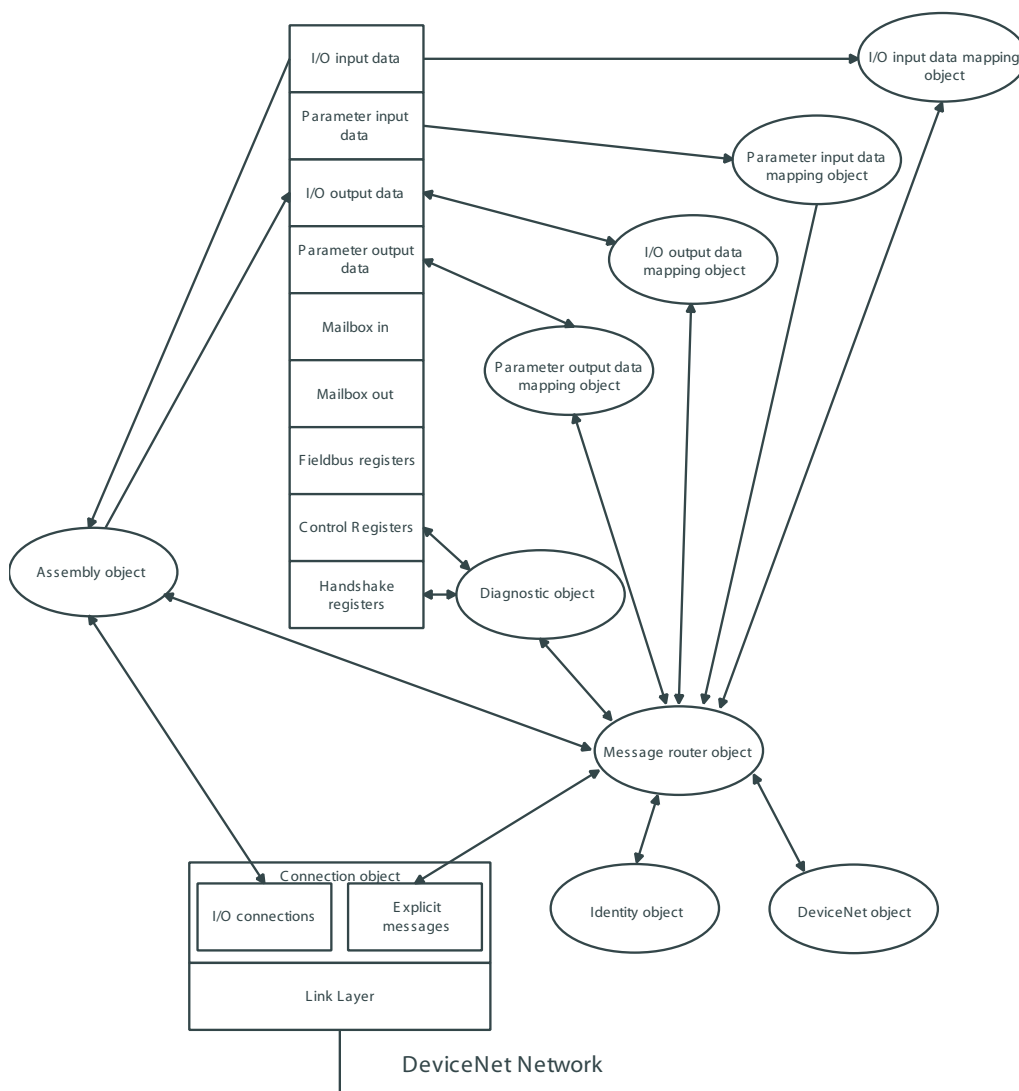
Baudrate	sw. 1	sw. 2
125k	OFF	OFF
250K	OFF	ON
500K	ON	OFF
(reserved)	ON	ON

Memory Structure

Overview

The interface from the fieldbus towards the module is based on the standard DeviceNet objects and five vendor specific objects. The vendor specific objects can be used to access different memory areas and for diagnostics during development. However, most applications requires only the standard DeviceNet objects.

For a complete description of the available objects, please see 4-1 “Implemented Objects”.



I/O Input and Output Data Areas

The I/O in/out data is read/written through I/O connections or Explicit messages from the assembly object.

Note: If other than the default Assembly instances should be used, those have to be instantiated with mailbox commands during the initialization of the module.

Parameter Input and Output Data Areas

The Parameter in/out data can be accessed via Explicit messages through application objects mapped by mailbox telegrams during the initialization.

Note: The Parameter data application objects have to be instantiated with mailbox commands during start-up of the module.

Implemented Objects

The following objects are implemented:

DeviceNet Objects

Object	Class
Identity Object	01h
Message Router Object	02h
DeviceNet Object	03h
Assembly Object	04h
Connection Object	05h
Acknowledge Handler Object	2Bh

Vendor Specific Objects

Object	Class
I/O data Input Mapping Object	A0h
I/O data Output Mapping Object	A1h
Diagnostic Object	AAh
Parameter data Input Mapping Object	B0h
Parameter data Output Mapping Object	B1h

Identity Object, Class 01h

Class Attributes

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	Revision	Get_Attribute_Single	Revision of the Identity Object.	The Revision attribute, which consists of Major and Minor.	1,1,1	UINT

Instance Attributes

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	Vendor Id	Get_Attribute_Single	Identification of each vender by number	Vendor IDs are managed by ODVA. The value zero is not valid	90, 90, 90	UINT
02h	Device Type	Get_Attribute_Single	Indication of the general type of product	The list of device types is managed by ODVA. It is used to identify the device profile that a particular product is using. Device profiles define minimum requirements a device must implement as well as common options.	12, 12, 12	UINT
03h	Product Code	Get_Attribute_Single	This is a code assigned by the vendor to describe the device	The vendor assigned Product Code identifies a particular product within a device type. Each vendor assigns this code to each of its products. The Product Code typically maps to one or more catalog/model numbers. Products shall have different codes if their configuration and/or runtime options are different. Such devices present a different logical view to the network.	12, 12, 12	UINT
04h	Revision	Get_Attribute_Single	Revision of the item the Identity Object represents	The Revision attribute, which consists of Major and Minor Revisions, identifies the Revision of the item the Identity Object is representing. The value zero is not valid for either the Major and Minor Revision fields. The Major and Minor Revision are typically displayed as major.minor. Minor revisions shall be displayed as three digits with leading zeros as necessary. The Major Revision attribute is limited to 7 bits. The eighth bit is reserved by DeviceNet and must have a default value of zero.	{1,1}, {1,1}, {1,1}	Array of: USINT USINT
05h	Status	Get_Attribute_Single	Summary Status of the Device	This attribute represents the current status of the entire device. Its value changes as the state of the device changes.	0, 0, 255	WORD
06h	Serial Number	Get_Attribute_Single	Serial Number of the device	This attribute is a number used in conjunction with the Vendor ID to form a unique identifier for each device on DeviceNet. Each vendor is responsible for guaranteeing the uniqueness of the serial number across all of its devices.	N/A, N/A, N/A	UDINT
07h	Product Name	Get_Attribute_Single	Human readable identification	This text string should represent a short description of the product/product family represented by the product code in attribute 3.	"AnyBus-S DNET ", "AnyBus-S DNET ", "ANYBUS-S DNET"	"SHORT_STRING"

Message router, Class 02h

Class Attributes

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
1	Revision	Get_Attribute_Single	Revision of the Identity Object.	The Revision attribute, which consists of Major and Minor.	1, 1, 1	UINT

DeviceNet Object, Class 03h

Class Attributes

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Revision	Get_Attribute_Single	Revision of the DeviceNet Object Class Definition upon which the implementation is based	The current value assigned to this attribute is two (2). If updates which require an increase in this value are made, then the value will be increased. Support of this attribute is required.	2, 2, 2	UINT

Instance Attributes

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	MAC ID	Get_Attribute_Single	Node Address.	This attribute contains the MAC ID of this device. The range of values is 0 to 63 decimal. A device that uses switches to set the MAC ID must return an Error Response whose General Error Code is set to 0E (Attribute not settable) in response to a Set_Attribute_Single Request specifying the MAC ID attribute. The exception to this rule is that if the MAC ID switches are not user accessible when the product is installed, then the switch settings can be overridden via a Set_Attribute_Single. The MAC ID attribute is considered non-volatile in that once configured the attribute must be remembered after a power cycle or device reset. If a device does not know what its MAC ID is it should default to 63. The modification of the MAC ID requires a device to delete all Connection Objects and re-execute the Network Access State Machine.	Dip-switch, 0, 63	USINT
02h	Baud Rate	Get_Attribute_Single	The baud rate of the device	The Baud Rate attribute indicates the selected baud rate. Values are 00 - 125k 01 - 250k 02 - 500k	Dip-switch, 0, 2	USINT
05h	Allocation Information	Get_Attribute_Single	Allocation Choice Master's Mac ID	Struct of: BYTE: Allocation Choice USINT: Master's Mac ID	N/A, N/A, N/A	Struct of: BYTE USINT

Assembly Object, Class 04h

Class Attributes

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Revision	Get_Attribute_All	Revision of the Assembly Object.	The revision attribute, which contains the object revision.	1, 1, 1	UINT
02h	Max Instance	Get_Attribute_Single	Maximum instance number of this object	The largest instance created by this object.	N/A, N/A, N/A	UINT

DPRAM INPUT AREA, Instance 64h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
03h	Data	Get_Attribute_Single	The data is produced from I/O I/O data input object, attribute 1. Default is the data configured as I/O input data in the init command of the module. This can be overwritten by mailbox telegrams during initialisation.		N/A, N/A, N/A	Array of USINT

Note: If the I/O input data size is set to 0 this instance will NOT be initialised.

DPRAM INPUT AREA, Instance 65h - 69h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
03h	Data	Get_Attribute_Single	The data is produced from I/O I/O data input object, attribute 2-5. Those instances can only be configured by mailbox telegrams during initialisation.		N/A, N/A, N/A	Array of USINT

Note: If the I/O input data size is set to 0 the instance will NOT be initialised.

DPRAM OUTPUT AREA, Instance 96h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
03h	Data	Get_Attribute_Single Set_Attribute_Single	The data is produced from I/O I/O data output object, attribute 1. Default is the data configured as I/O output data in the init command of the module. This can be overwritten by mailbox telegrams during initialisation.		N/A, N/A, N/A	Array of USINT

Note: If the I/O output data size is set to 0 this instance will NOT be initialised.

DPRAM INPUT AREA, Instance 97h - 9Bh

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
03h	Data	Get_Attribute_Single	The data is produced from I/O I/O data output object, attribute 2-4. Those instances can only be configured by mailbox telegrams during initialisation.		N/A, N/A, N/A	Array of USINT

Note: If the I/O input data size is set to 0 the instance will NOT be initialised.

DeviceNet Connection Object, Class 05h

Class Attributes (0)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	Revision	Get_Attribute_Single	Revision of the DeviceNet Object		2, 2, 2	UINT

Explicit Connection Instance (1)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	State	Get_Attribute_Single	State of the object	0=Nonexistent 1=Configuring 3=Established 4=Timed out 5=Deferred delete	1, 0, 5	USINT
02h	Instance Type	Get_Attribute_Single	Indicates either IO or messaging connection		0, 0, 0	USINT
03h	Transport Class Trigger	Get_Attribute_Single, Set_Attribute_Single	Defines Behaviour of the connection		83h, 83h, 83h	BYTE
04h	Produced Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	Placed in CAN Identifier Field when the Connection Transmits		N/A, N/A, N/A	UINT
05h	Consumed Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	CAN Identifier Field value that denotes message to be received		N/A, N/A, N/A	UINT
06h	Initial Comm Characteristics	Get_Attribute_Single, Set_Attribute_Single	Defines the Message Group(s) across which productions and consumptions associated with this		N/A, N/A, N/A	BYTE
07h	Produced Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes transmitted across this Connection	512	512, 512, 512	UINT
08h	Consumed Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes received across this Connection	512	512, 512, 512	UINT
09h	Expected Packet Rate	Get_Attribute_Single, Set_Attribute_Single	Defines timing associated with this Connection	Resolution is 10 ms	N/A, N/A, N/A	UINT
0Ch	Watchdog Timeout Action	Get_Attribute_Single, Set_Attribute_Single	Defines how to handle Inactivity/Watchdog timeouts	0-Transition to time out 1-Auto Delete 2-Auto Reset 3-Deferred Delete	N/A, N/A, N/A	USINT
0Dh	Produced Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the produced_connection_path length attribute		0, 0, 0	UINT
0Eh	Produced Connection Path	Get_Attribute_Single, Set_Attribute_Single	Application Obj. producing data on this connection		0, 0, 0	ARRAY OF: USINT
0Fh	Consumed Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the consumed_connection_path length attribute		0, 0, 0	UINT
10h	Consumed Connection Path	Get_Attribute_Single, Set_Attribute_Single	Specifies the Application Object(s) that are to receive the data consumed by this Connection Object		N/A, N/A, N/A	ARRAY OF: 01 UINT
11h	Production Inhibit Time	Get_Attribute_Single	Defines the minimum time between new data production		0,0,0	UINT

Polled I/O Connection Instance (2)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	State	Get_Attribute_Single	State of the object	0=Nonexistent 1=Configuring 3=Established 4=Timed out	1, 0, 4	USINT
02h	Instance Type	Get_Attribute_Single	Indicates either IO or messaging connection		0, 0, 1	USINT
03h	Transport Class Trigger	Get_Attribute_Single, Set_Attribute_Single	Defines Behaviour of the connection		N/A, N/A, N/A	BYTE
04h	Produced Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	Placed in CAN Identifier Field when the Connection Transmits		N/A, N/A, N/A	UINT
05h	Consumed Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	CAN Identifier Field value that denotes message to be received		N/A, N/A, N/A	UINT
06h	Initial Comm Characteristics	Get_Attribute_Single, Set_Attribute_Single	Defines the Message Group(s) across which productions and consumptions associated with this		N/A, N/A, N/A	BYTE
07h	Produced Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes transmitted across this Connection		I/O in length, 0, I/O in length	UINT
08h	Consumed Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes received across this Connection		I/O out length, 0, I/O out length	UINT
09h	Expected Packet Rate	Get_Attribute_Single, Set_Attribute_Single	Defines timing associated with this Connection		N/A, N/A, N/A	UINT
0Ch	Watchdog Timeout Action	Get_Attribute_Single, Set_Attribute_Single	Defines how to handle Inactivity/Watchdog timeouts	0-Transition to time out 1-Auto Delete 2-Auto Reset 3-Deferred Delete	N/A, N/A, N/A	USINT
0Dh	Produced Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the produced_connection_path length attribute		6, 6, 6	UINT
0Eh	Produced Connection Path	Get_Attribute_Single, Set_Attribute_Single	Application Obj. producing data on this connection		20 04 24 64 30 03, N/A, N/A	ARRAY OF: USINT
0Fh	Consumed Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the consumed_connection_path length attribute		6, 6, 6	UINT
10h	Consumed Connection Path	Get_Attribute_Single, Set_Attribute_Single	Specifies the Application Object(s) that are to receive the data consumed by this Connection Object		20 04 24 96 30 03, N/A, N/A	ARRAY OF: USINT
11h	Production Inhibit Time	Get_Attribute_Single	Defines the minimum time between new data production		0,0,0	UINT

Bit-Strobe Connection Instance (3)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	State	Get_Attribute_Single	State of the object	0=Nonexistent 1=Configuring 3=Established 4=Timed out	1, N/A, N/A	USINT
02h	Instance Type	Get_Attribute_Single	Indicates either IO or messaging connection		1, 0, 1	USINT
03h	Transport Class Trigger	Get_Attribute_Single, Set_Attribute_Single	Defines Behaviour of the connection		N/A, N/A, N/A	BYTE
04h	Produced Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	Placed in CAN Identifier Field when the Connection Transmits		N/A, N/A, N/A	UINT
05h	Consumed Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	CAN Identifier Field value that denotes message to be received		N/A, N/A, N/A	UINT
06h	Initial Comm Characteristics	Get_Attribute_Single, Set_Attribute_Single	Defines the Message Group(s) across which productions and consumptions associated with this		N/A, N/A, N/A	BYTE
07h	Produced Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes transmitted across this Connection		N/A, N/A, N/A	UINT
08h	Consumed Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes received across this Connection		8, 8, 8	UINT
09h	Expected Packet Rate	Get_Attribute_Single, Set_Attribute_Single	Defines timing associated with this Connection		0, 0, FFFFh	UINT
0Ch	Watchdog Timeout Action	Get_Attribute_Single, Set_Attribute_Single	Defines how to handle Inactivity/Watchdog timeouts	0-Transition to time out 1-Auto Delete 2-Auto Reset 3-Deferred Delete	N/A, N/A, N/A	USINT
0Dh	Produced Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the produced_connection_path length attribute		6, 6, 6	UINT
1Eh	Produced Connection Path	Get_Attribute_Single, Set_Attribute_Single	Application Obj. producing data on this connection		20 04 24 65 30 03, N/A, N/A	ARRAY OF: USINT
1Fh	Consumed Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the consumed_connection_path length attribute		6, 6, 6	UINT
10h	Consumed Connection Path	Get_Attribute_Single, Set_Attribute_Single	Specifies the Application Object(s) that are to receive the data consumed by this Connection Object		20 04 24 97 30 03, N/A, N/A	ARRAY OF: UINT

Change of state / Cyclic (4) (Acknowledged)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	State	Get_Attribute_Single	State of the object	0=Nonexistent 1=Configuring 3=Established 4=Timed out	1, N/A, N/A	USINT
02h	Instance Type	Get_Attribute_Single	Indicates either IO or messaging connection		1, 0, 1	USINT
03h	Transport Class Trigger	Get_Attribute_Single, Set_Attribute_Single	Defines Behaviour of the connection		N/A, N/A, N/A	BYTE
04h	Produced Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	Placed in CAN Identifier Field when the Connection Transmits		N/A, N/A, N/A	UINT
05h	Consumed Cnxn Id	Get_Attribute_Single, Set_Attribute_Single	CAN Identifier Field value that denotes message to be received		N/A, N/A, N/A	UINT
06h	Initial Comm Characteristics	Get_Attribute_Single, Set_Attribute_Single	Defines the Message Group(s) across which productions and consumptions associated with this		N/A, N/A, N/A	BYTE
07h	Produced Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes transmitted across this Connection		0, 0, N/A	UINT
08h	Consumed Connection Size	Get_Attribute_Single, Set_Attribute_Single	Maximum number of bytes received across this Connection		0, 0, N/A	UINT
09h	Expected Packet Rate	Get_Attribute_Single, Set_Attribute_Single	Defines timing associated with this Connection		0, 0, FFFFh	UINT
0Ch	Watchdog Timeout Action	Get_Attribute_Single, Set_Attribute_Single	Defines how to handle Inactivity/Watchdog timeouts	0-Transition to time out 1-Auto Delete 2-Auto Reset 3-Deferred Delete	N/A, N/A, N/A	USINT
0Dh	Produced Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the produced_connection_path length attribute		0, 0, 6	UINT
0Eh	Produced Connection Path	Get_Attribute_Single, Set_Attribute_Single	Application Obj. producing data on this connection		20 04 24 66 30 03, 0, N/A	ARRAY OF: USINT
0Fh	Consumed Connection Path Length	Get_Attribute_Single, Set_Attribute_Single	Number of bytes in the consumed_connection_path length attribute		4, 4, 4	UINT
10h	Consumed Connection Path	Get_Attribute_Single, Set_Attribute_Single	Specifies the Application Object(s) that are to receive the data consumed by this Connection Object		20 2B 24 01, 20 2B 24 01, 20 2B 24 01	ARRAY OF: UINT

Acknowledge Handler Object, Class 2Bh

Class Attributes (0)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	Revision	Get_Attribute_Single	Revision of the DeviceNet Object Class Definition upon which the implementation is based	The current value assigned to this attribute is two (2). If updates, which require an increase in this value, are made, then the value will be increased. Support of this attribute is required.	1, 1, 1	UINT
02h	Max Instance	Get_Attribute_Single	Maximum instance number of an object currently created in this class level of the device.	The largest instance number of a created object at this class hierarchy level.		UINT

Instance Attributes (1)

#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	Acknowledge Timer	Get_Attribute_Single, Set_Attribute_Single	Time to wait for acknowledge before resending	Range 1-65535 ms (0 invalid) default=16	16, 1, 65535	UINT
02h	Retry Limit	Get_Attribute_Single, Set_Attribute_Single	Number of Ack Timeouts to wait before informing the producing application of a Retry-Limit_Reached event.	Range 0-255 default.	1, 0, 255	USINT
03h	COS Producing Connection Instance	Get_Attribute_Single, Set_Attribute_Single	Connection Instance which contains the path of the producing I/O application object a which will be notified of Ack Handler events.	Connection instance Id	N/A	UINT
04h	Ack List Size	Get_Attribute_Single	Maximum number of members in Ack List	0=Dynamic		
05h	Ack List	Get_Attribute_Single	List of active connection instances, which are receiving Acks.	Number of members followed by list of Connection Instance ID	N/A	BYTE Array of USINT
06h	Data with Ack Path List Size	Get_Attribute_Single	Maximum number of members in Data with Ack Path List.	0=Dynamic		
07h	Data with Ack Path List	Get_Attribute_Single	List of connection instance/consuming application object pairs. This attribute is used to forward data received with acknowledgment.	List of connection instance/consuming application object pairs. This attribute is used to forward data received with acknowledgments	N/A	BYTE Array of UINT USINT Array of USINT

I/O data Input Mapping Object, Class A0h

Class Attributes

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Revision	Get_Attribute_All	Revision of the Parameter data input object.	The revision attribute, which contains the object revision.	1, 1, 1	UINT

Instance Attribute, 01h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Data	Get_Attribute_Single	Data that is read	In bytes	N/A	Array of USINT
02h	Data	Get_Attribute_Single	Data that is read	In bytes	N/A	Array of USINT
...
06h	Data	Get_Attribute_Single	Data that is read	In bytes	N/A	Array of USINT

Note: Which attributes that exist are depending on which attributes that are initialised from the DPRAM during initialisation.

I/O data Output Mapping Object, Class A1h

Class Attributes

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Revision	Get_Attribute_All	Revision of the Parameter data output object.	The revision attribute, which contains the object revision.	1, 1, 1	UINT

Instance Attributes, Instance 01h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Data	Get_Attribute_Single, Set_Attribute_Single	Data that is read or written	In bytes	N/A	Array of USINT
02h	Data	Get_Attribute_Single, Set_Attribute_Single	Data that is read or written	In bytes	N/A	Array of USINT
...
06h	Data	Get_Attribute_Single, Set_Attribute_Single	Data that is read or written	In bytes	N/A	Array of USINT

Note: Which attributes that exist are depending on which attributes that are initialised from the DPRAM during initialisation.

Diagnostic Object, Class AAh

Class Attributes

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data Type
01h	Revision	Get_Attribute_All	Revision of the Diagnostic Object.	The revision attribute, which contains the object revision.	1, 1, 1	UINT

Diagnostic Instance 01h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	Data type	DPM	SCI
01h	Module serial number	Get_Attribute_Single	Serial number			UDINT	*	*
02h	Vendor ID	Get_Attribute_Single	Manufacture ID			UINT	*	*
03h	Fieldbus type	Get_Attribute_Single	Fieldbus type			UINT	*	*
04h	Module software version	Get_Attribute_Single	Software version			UINT	*	*
05h	Interrupt count	Get_Attribute_Single	Counter incremented each handshake interrupt			UINT		
06h	Watchdog counter in	Get_Attribute_Single	Not implemented			UINT		
07h	Watchdog counter out	Get_Attribute_Single	Counter incremented each 1ms			UINT		
09h	LED status	Get_Attribute_Single	LED indication status 1byte / LED			STRUCT of {USINT, USINT, USINT, USINT, USINT}	*	*
0Ah	Module type	Get_Attribute_Single	Module type			UINT	*	*
0Bh	Module status	Get_Attribute_Single	Bit information, Freeze, clear, etc.			WORD	*	*
0Ch	New data field	Get_Attribute_Single	Array of new data flags for 8 bytes area			LWORD		
0Dh	Interrupt cause	Get_Attribute_Single	Interrupt cause register			WORD		
0Eh	Interrupt notification	Get_Attribute_Single	Interrupt notification setting register			WORD		
0Fh	IN cyclic I/O length	Get_Attribute_Single	Size of I/O IN data (bytes)			UINT	*	*
10h	IN DPRAM length	Get_Attribute_Single	Number of valid IN bytes in DPRAM			UINT		
11h	IN total length	Get_Attribute_Single	Total number of IN bytes supported			UINT	*	*
12h	OUT cyclic I/O length	Get_Attribute_Single	Size of I/O OUT (bytes) data			UINT	*	*
13h	OUT DPRAM length	Get_Attribute_Single	Number of valid OUT bytes in DPRAM			UINT		
14h	OUT total length	Get_Attribute_Single	Total number of OUT bytes supported			UINT	*	*
15h	Reserved	Get_Attribute_All	Reserved for future 16-bit compatibility			UINT	*	*
16h	Application indication	Get_Attribute_Single	Application indication register			USINT	*	
17h	AnyBus indication	Get_Attribute_Single	AnyBus indication register			USINT	*	

Parameter data Input Mapping Object, Class B0h

Class Attributes

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Revision	Get_Attribute_All	Revision of the Parameter data input object.	The revision attribute, which contains the object revision.	1, 1, 1	UINT

Instance Attributes, Instance 01h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Data	Get_Attribute_Single	Data that is read	In bytes	N/A	Array of USINT
02h	Data	Get_Attribute_Single	Data that is read	In bytes	N/A	Array of USINT
...
32h	Data	Get_Attribute_Single	Data that is read	In bytes	N/A	Array of USINT

Note: Which attributes that exist are depending on which attributes that are initialised from the DPRAM during initialisation.

Parameter data Output Mapping Object, Class B1h

Class Attributes

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Revision	Get_Attribute_All	Revision of the Parameter data output object.	The revision attribute, which contains the object revision.	1, 1, 1	UINT

Instance Attributes, Instance 01h

ID#	Attribute Name	Services	Description	Semantics	Default, Min, Max	DataType
01h	Data	Get_Attribute_Single, Set_Attribute_Single	Data that is read or written	In bytes	N/A	Array of USINT
02h	Data	Get_Attribute_Single, Set_Attribute_Single	Data that is read or written	In bytes	N/A	Array of USINT
...
32h	Data	Get_Attribute_Single, Set_Attribute_Single	Data that is read or written	In bytes	N/A	Array of USINT

Note: Which attributes that exist are depending on which attributes that are initialised from the DPRAM during initialisation.

Mailbox Interface

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

Summary

Name	Description	Page
SET_DIPSWITCH	Set DIP Switch value, cannot override physical setting	5-2
SET_DIPSWITCH_2	Set DIP Switch value, overrides physical setting	5-3
GET_DIPSWITCH	Read DIP switch value	5-4
PARAMETER_INPUT_MAP	Map attributes in the Parameter Data Input Mapping Object	5-5
PARAMETER_OUTPUT_MAP	Map attributes in the Parameter Data Output Mapping Object	5-7
IO_INPUT_MAP	Map instances in the Assembly Object	5-9
IO_OUTPUT_MAP	Map instances in the Assembly Object	5-11
PRODUCT_INFO	Customize Vendor ID, Product Code, and Product Name	5-13
PRODUCT_INFO_ALL	Customize Vendor ID, Device Type, Product Code, Major Revision, Minor Revision and Product Name	5-15

Configuration Switch

Set DipSwitch (SET_DIPSWITCH)

Description

The SET_DIPSWITCH telegram is used to set the MAC ID and baudrate from the application instead of using the dip switch mounted on the module. This is useful in embedded applications, where the switches may be inaccessible. Note that all DIP switches must be set to 'ON' in order for this command to work.

When using this method to set the MAC ID and the baudrate the dip switch should not be visible to the end user.

Parameter	Description
Command initiator	Application
Command Name	SET_DIPSWITCH
Message type	02h
Command number	0001h
Fragmented	No
Extended Header data	-
Command data	New DIP switch value. ON=1, OFF=0
Response data	If the value is out of range this will force the module to stay off-line

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0001h	0001h	SET_DIPSWITCH
Data size	0001h	0001h	
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 byte 1	Configuration byte	Configuration byte	00h=On line, 01h=Off line

Configuration byte:

Switch:	8	7	6	5	4	3	2	1
Bit:	7	6	5	4	3	2	1	0

Set DipSwitch 2 (SET_DIPSWITCH_2)

Description

This command is an updated version of the SET_DIPSWITCH command and is identical except that it overrides the physical DIP switch settings completely, i.e it is not required to set all switches to 'ON'. The old SET_DIPSWITCH command has been kept for backwards compatibility.

This command can only be sent during the module initialisation.

This command in conjunction with GET_DIPSWITCH can be used to alter the meaning of the DIP switches. This is useful if the module is mounted e.g upside down or vertical, as this changes the position of the switches in a way that does not comply with the DeviceNet specification.

Parameter	Description
Command initiator	Application
Command Name	SET_DIPSWITCH_2
Message type	02h
Command number	000Ah
Fragmented	No
Extended Header data	-
Command data	New DIP switch value. ON=1, OFF=0
Response data	If the value is out of range this will force the module to stay off-line

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	000Ah	000Ah	SET_DIPSWITCH_2
Data size	0001h	0001h	
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 byte 1	Value	Value	

Value:

Switch:	8	7	6	5	4	3	2	1
Bit:	7	6	5	4	3	2	1	0

Get DipSwitch (GET_DIPSWITCH)

Description

This command is used to read out the physical setting of the DIP switch. In conjunction with SET_DIPSWITCH_2, this command can be used to alter the meaning of the DIP switches. This is useful if the module is mounted e.g upside down or vertical, as this changes the position of the switches in a way that does not comply with the DeviceNet specification.

Parameter	Description
Command initiator	Application
Command Name	GET_DIPSWITCH
Message type	02h
Command number	0008h
Fragmented	No
Extended Header data	-
Command data	-
Response data	The response contains the physical DIP switch setting. ON=1, OFF=0

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0008h	0008h	GET_DIPSWITCH
Data size	0000h	0001h	
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 byte 1		Value of dipswitch	

DIP Switch Value:

Switch:	8	7	6	5	4	3	2	1
Bit:	7	6	5	4	3	2	1	0

Mapping

Parameter Data Input Area Mapping (PARAMETER_INPUT_MAP)

Description

This command makes it possible to map attributes in the “Parameter data input mapping object”, Class B0h. By mapping data a Get_Attribute_Single command from the fieldbus to the “Parameter data input mapping object” can return a specified block of data. Up to 50 attributes can be mapped.

Attributes are mapped with start of attribute 1. Offset is set from the start of the Parameter data input area and length specifies the amount of bytes to map. After attribute 1 in the telegram attribute 2 follows and then all attributes to attribute 50. If length is set to zero the attribute will not exist. It is for example possible to map only object 1 and 10 by letting the length of attributes 2-9 be zero. It is only necessary to include information in the telegram for attributes up to the last attribute number used.

If any offset or length is invalid, the length and offset will be set to zero in the mailbox answer and the attribute will not be mapped

Parameter	Description
Command initiator	Application
Command Name	PARAMETER_INPUT_MAP
Message type	02h
Command number	0004h
Fragmented	No
Extended Header data	-
Command data	Offset and length of the attributes to map.
Response data	The response indicates if the message was accepted.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0004h	0004h	PARAMETER_INPUT_MAP
Data size	0014h	0014h	
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 word 1	Offset for Attribute 1	Offset for Attribute 1	Offset from parameter data input area start
Message data word 2	Length for Attribute 1	Length for Attribute 1	
Message data word 3	Offset for Attribute 2	Offset for Attribute 2	
Message data word 4	Length for Attribute 2	Length for Attribute 2	
Message data word 5	Offset for Attribute 3	Offset for Attribute 3	
Message data word 6	Length for Attribute 3	Length for Attribute 3	
Message data word 7	Offset for Attribute 4	Offset for Attribute 4	
Message data word 8	Length for Attribute 4	Length for Attribute 4	
Message data word 9	Offset for Attribute 5	Offset for Attribute 5	
Message data word 10	Length for Attribute 5	Length for Attribute 5	

Note: Example when only setting attribute 1-5.

Parameter Data Output Area Mapping (PARAMETER_OUTPUT_MAP)

Description

This command makes it possible to map attributes in the “Parameter data output mapping object”, Class B1h. By mapping data a Get_Attribute_Single command or a Set_Attribute_Single command from the fieldbus to the “Parameter data output mapping object” can return a specified block of data or write a block of data. Up to 50 attributes can be mapped.

Attributes are mapped with start of attribute 1. Offset is set from the start of the Parameter data input area and length specifies the amount of bytes to map. After attribute 1 attribute in the telegram attribute 2 follows and then all attributes to attribute 50. If length is set to zero the attribute will not exist. It is for example possible to map only object 1 and 10 by letting the length of attributes 2-9 be zero. It is only necessary to include information in the telegram for attributes up to the last attribute number used.

If any offset or length is invalid the length and offset will be set to zero in the mailbox answer and the attribute will not be mapped.

Parameter	Description
Command initiator	Application
Command Name	PARAMETER_OUTPUT_MAP
Message type	02h
Command number	0005h
Fragmented	No
Extended Header data	-
Command data	Offset and length of the attributes to map.
Response data	The response indicates if the message was accepted.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0005h	0005h	PARAMETER_OUTPUT_MAP
Data size	0014h	0014h	
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 word 1	Offset for Attribute 1	Offset for Attribute 1	Offset from parameter data output area start
Message data word 2	Length for Attribute 1	Length for Attribute 1	Number of bytes to map
Message data word 3	Offset for Attribute 2	Offset for Attribute 2	
Message data word 4	Length for Attribute 2	Length for Attribute 2	
Message data word 5	Offset for Attribute 3	Offset for Attribute 3	
Message data word 6	Length for Attribute 3	Length for Attribute 3	
Message data word 7	Offset for Attribute 4	Offset for Attribute 4	
Message data word 8	Length for Attribute 4	Length for Attribute 4	
Message data word 9	Offset for Attribute 5	Offset for Attribute 5	
Message data word 10	Length for Attribute 5	Length for Attribute 5	

Note: Example when only setting attribute 1-5.

I/O Data Input Area Mapping (IO_INPUT_MAP)

Description

This command makes it possible to map instances in the Assembly Object, Class 04h. Attributes are mapped with start at Instance 100. Offset is set from the start of the I/O data input area and length specifies the amount of bytes to map. After attribute 100 in the telegram attribute 101 follows and then all instances up to 104. If length is set to zero the instance will not be instantiated. It is for example possible to map only object 100 and 104 by letting the length of instances 101-103 be zero. It is only necessary to include information in the telegram for instances up to the last instance number used.

If any offset or length is invalid the length and offset will be set to zero in the mailbox answer and the attribute will not be mapped

Parameter	Description
Command initiator	Application
Command Name	IO_INPUT_MAP
Message type	02h
Command number	0006h
Fragmented	No
Extended Header data	-
Command data	Offset and length of the instance to map.
Response data	The response indicates if the message was accepted.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0006h	0006h	<i>IO_INPUT_MAP</i>
Data size	0014h	0014h	
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 word 1	Offset for Attribute 1	Offset for Attribute 1	<i>Offset from I/O input area start</i>
Message data word 2	Length for Attribute 1	Length for Attribute 1	<i>Number of bytes to map</i>
Message data word 3	Offset for Attribute 2	Offset for Attribute 2	
Message data word 4	Length for Attribute 2	Length for Attribute 2	
Message data word 5	Offset for Attribute 3	Offset for Attribute 3	
Message data word 6	Length for Attribute 3	Length for Attribute 3	
Message data word 7	Offset for Attribute 4	Offset for Attribute 4	
Message data word 8	Length for Attribute 4	Length for Attribute 4	
Message data word 9	Offset for Attribute 5	Offset for Attribute 5	
Message data word 10	Length for Attribute 5	Length for Attribute 5	

Note: Example when only setting attribute 1-5.

I/O Data Output Area Mapping (IO_OUTPUT_MAP)

Description

This command makes it possible to map instances in the Assembly Object, Class 04h. Attributes are mapped with start at Instance 150. Offset is set from the start of the I/O data input area and length specifies the amount of bytes to map. After attribute 150 in the telegram attribute 151 follows and then all instances up to 150. If length is set to zero the instance will not be instantiated. It is for example possible to map only object 150 and 154 by letting the length of instances 151-153 be zero. It is only necessary to include information in the telegram for instances up to the last instance number used.

If any offset or length is invalid the length and offset will be set to zero in the mailbox answer and the attribute will not be mapped

Parameter	Description
Command initiator	Application
Command Name	IO_OUTPUT_MAP
Message type	02h
Command number	0007h
Fragmented	No
Extended Header data	-
Command data	Offset and length of the attributes to map.
Response data	The response indicates if the message was accepted.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0007h	0007h	<i>IO_OUTPUT_MAP</i>
Data size	0014h	0014h	
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 word 1	Offset for Attribute 1	Offset for Attribute 1	<i>Offset from I/O output area start</i>
Message data word 2	Length for Attribute 1	Length for Attribute 1	<i>Number of bytes to map</i>
Message data word 3	Offset for Attribute 2	Offset for Attribute 2	
Message data word 4	Length for Attribute 2	Length for Attribute 2	
Message data word 5	Offset for Attribute 3	Offset for Attribute 3	
Message data word 6	Length for Attribute 3	Length for Attribute 3	
Message data word 7	Offset for Attribute 4	Offset for Attribute 4	
Message data word 8	Length for Attribute 4	Length for Attribute 4	
Message data word 9	Offset for Attribute 5	Offset for Attribute 5	
Message data word 10	Length for Attribute 5	Length for Attribute 5	

Note: Example when only setting attribute 1-5.

Other

Set Product Info (PRODUCT_INFO)

Description

Set Product Info is used to change Vendor ID, Product Code and Product Name in the Identity object to customise your product.

This command can only be sent during module initialisation.

Note: To change the identity object information, a Vendor ID from ODVA international MUST have been received.

Parameter	Description
Command initiator	Application
Command Name	PRODUCT_INFO
Message type	02h
Command number	0001h
Fragmented	No
Extended Header data	-
Command data	Vendor ID, product code, Name length and Name.
Response data	The response indicates if the message was accepted.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0002h	0002h	<i>PRODUCT_INFO</i>
Data size	N	N	<i>Data size N</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	-	-	
Message data byte 1	Vendor ID (msb)	Vendor ID (msb)	
Message data byte 2	Vendor ID (lsb)	Vendor ID (lsb)	
Message data byte 3	Product Code (msb)	Product Code (msb)	
Message data byte 4	Product Code (lsb)	Product Code (lsb)	
Message data byte 5	Product Name Length	Product Name Length	<i>Maximum 32 characters</i>
Message data byte 6	Product Name (1st char.)	Product Name (1st char.)	
Message data byte 7	Product Name (2nd char.)	Product Name (2nd char.)	
...	
Message data byte N	Product Name (last char.)	Product Name (last char.)	

Set Product Info All (PRODUCT_INFO_ALL)

Description

Set Product Info All is used to change Vendor ID, Device Type, Product Code, Major Revision, Minor Revision and Product Name in the Identity object to customise the developed product. The EDS file also need to be modified accordingly in order to fit the product.

This command can only be sent during the module initialisation.

Note: To change the identity object information, a VendorID, a Device Type number and a Product code number will be needed from ODVA international.

Parameter	Description
Command initiator	Application
Command Name	PRODUCT_INFO_ALL
Message type	02h
Command number	0009h
Fragmented	No
Extended Header data	-
Command data	Vendor ID, Device Type, Product Code, Major revision, Minor revision, Name length and Name (ascii values)
Response data	The response indicates if the message was accepted.

Command and response layout:

	Command	Expected response	
Message ID	(ID)	(ID)	
Message information	4002h	0002h	
Command	0009h	0009h	<i>PRODUCT_INFO_ALL</i>
Data size	n	n	
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 byte 1	Vendor ID (msb)	Vendor ID (msb)	
Message data byte 2	Vendor ID (lsb)	Vendor ID (lsb)	
Message data byte 3	Device Type (msb)	Device Type (msb)	
Message data byte 4	Device Type (lsb)	Device Type (lsb)	
Message data byte 5	Product Code (msb)	Product Code (msb)	
Message data byte 6	Product Code (lsb)	Product Code (lsb)	
Message data byte 7	Major revision	Major revision	
Message data byte 8	Minor revision	Minor revision	
Message data byte 9	Product Name Length	Product Name Length	<i>Maximum 32 characters</i>
Message data byte 10	Product Name (1st char.)	Product Name (1st char.)	<i>Product name, 1st character</i>
Message data byte 11	Product Name (2nd char.)	Product Name (2nd char.)	
...	
Message data byte N	Product Name (last char.)	Product Name (last char.)	<i>Product name, last character</i>

Fieldbus Specific Area

This area holds runtime status information from the DeviceNet network. The information that can be read from this area is specified below:

DPRAM Address:	Modbus Address:	Area	Description
640h - 641h	2120h	Identity Status Attribute	(see below)
642h	2121h (high)	Explicit Connection Status	See Connection Status below.
643h	2121h (low)	Polled I/O Connection Status	See Connection Status below.
644h	2122h (high)	Strobed Connection Status	See Connection Status below.
645h	2122h (low)	Change of state / Cyclic Connection Status	See Connection Status below.
646h	2123h (high)	(reserved)	-
647h	2123h (low)	Master Status	DeviceNet master status
648h - 7BFh	2124h -21DFh	(reserved)	-

Identity Status Attribute (Read Only)

Bit(s)	Name	Definition
0	Owned	TRUE indicates that the device (or and object within the device) has an owner. Within the Master / Slave paradigm the setting of this bit means that the Predefined Master/Slave Connection Set has been allocated to a master.
1	-	(reserved)
2	Configured	TRUE indicates that the application of the device has been configured to do something different than the "out-of-box" default. (This does not include communication settings)
3 ... 7	-	(reserved)
8	Minor Recoverable Fault	TRUE indicates that the device has detected a problem with itself, which is thought to be recoverable. The problem does not cause the device to go into one of the faulted states.
9	Minor Unrecoverable Fault	TRUE indicates that the device has detected a problem with itself, which is thought to be unrecoverable. The problem does not cause the device to go into one of the faulted states.
10	Major Recoverable Fault	TRUE indicates that the device has detected a major problem with itself, which has caused the device to go into the "Major Recoverable Fault" state.
11	Major Unrecoverable Fault	TRUE indicates that the device has detected a major problem with itself, which has caused the device to go into the "Major Unrecoverable Fault" state.
12 ... 15	-	(reserved)

Connection Status Registers (Read Only)

This register indicates the connection status. Possible values of these registers:

Value	Meaning
00h	Nonexistent
01h	Configuring
02h	(not used)
03h	Established
04h	Timed out
05h	Deferred delete

Master Status (Read Only)

This register indicates the status of the master on the network. Possible values are:

Value	Meaning
00h	Master not initialized to send data
01h	Master is Running
02h	Master is Idle

Note: If the master is not initialised to send data to the slave, it is not possible to tell if the master is in Run or Idle mode

Electrical Characteristics

Supply Voltage

Application Interface

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

Bus

The bus interface shall normally be supplied with 24VDC but will operate as intended as long as the supply voltage is within the DeviceNet-allowed 11 to 25V DC span.

Power Consumption

Symbol	Description	Min.	Typ.	Max	Unit
I _{INMOD}	Supply current, module electronics	-	170	200	mA
I _{INBUS}	Supply current, bus interface	-	25	30	mA

PE Grounding

A PE-connection is included on one of the mounting holes according to the AnyBus-S specification.

Galvanic Isolation

The bus interface power is taken from the DeviceNet connector. The send and receive signals are isolated via opto couplers.

Network Termination

DeviceNet uses termination resistors at each physical end of the bus. The termination resistor should be 121 ohm. This should be connected between CAN_H and CAN_L on the bus.

Connectors

Application Interface

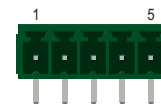
The module is available in both serial and parallel versions.

- **Serial Version**
Consult the general AnyBus-S Serial Design Guide
- **Parallel Version**
Consult the general AnyBus-S Parallel Design Guide

Fieldbus Interface

Pluggable Screw (5.08 and 3.84 mm)

Pin	Signal	Description
1	V-	Negative supply voltage ^a
2	CAN_L	CAN_L bus line
3	SHIELD	Cable shield
4	CAN_H	CAN_H bus line
5	V+	Positive supply voltage ^a



a. DeviceNet bus power. See Appendix A-1 “Electrical Characteristics”

2mm Board to Board

Pin	Signal	Description
1	SHIELD	Cable shield
2	-	-
3	CAN_L	CAN_L bus line
4	V-	Negative supply voltage ^a
5	CAN_H	CAN_H bus line
6	V-	Negative supply voltage ^a
7	-	-
8	V+	Positive supply voltage ^a
9	-	-
10	SHIELD	Cable shield



a. DeviceNet bus power. See Appendix A-1 “Electrical Characteristics”

Environmental Specification

Temperature

Operating

+0 to +70 degrees Celsius

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

Non Operating

-15 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

Mechanical Specification

The mechanical drawings in this chapter applies to the standard configuration of the module. For other connector configurations etc, contact the HMS marketing department.

All measurements expressed in this chapter have a tolerance of $\pm 0.25\text{mm}$ unless otherwise stated.

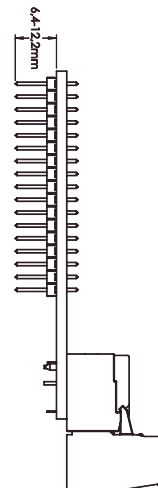
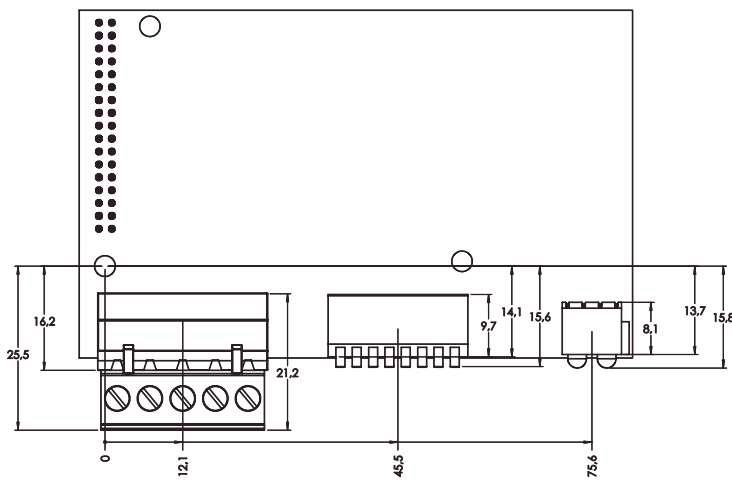
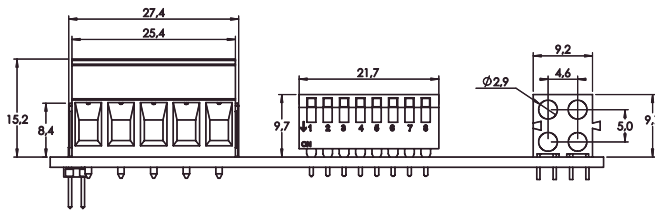
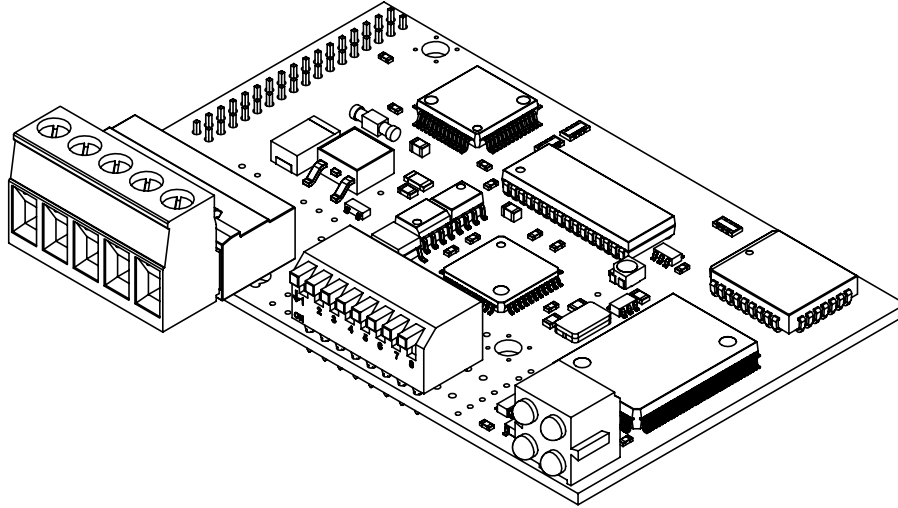
Standard Part Numbers

Parallel version: AB4004

Serial version: AB4026

Measurements, Connectors & Switches

Angled Switches & Connectors



Straight Switches & Connectors

