



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China





User Manual Anybus[®] Communicator[™] for DeviceNet[™]

Doc. Id. HMSI-27-305
Rev. 3.12

Important User Information

This document contains a general introduction as well as a description of the technical features provided by the Anybus Communicator, including the PC-based configuration software.

The reader of this document is expected to be familiar with PLC and software design, as well as communication systems in general. The reader is also expected to be familiar with the Microsoft® Windows® operating system.

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. Microsoft® and Windows® are registered trademarks of Microsoft, Inc. DeviceNet™ and ODVA™ are trademarks of ODVA, Inc. 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.

Anybus Communicator DeviceNet User Manual
Copyright© HMS Industrial Networks AB
Doc: HMSI-27-305

Table of Contents

Preface	About This Document	
	Related Documents	7
	Document History	7
	Conventions & Terminology	8
	<i>Glossary</i>	8
	Support.....	8
Chapter 1	About the Anybus Communicator for DeviceNet	
	External View.....	10
	Status LEDs	11
	Configuration Switches.....	11
	Hardware Installation.....	12
	Software Installation	13
	<i>Anybus Configuration Manager</i>	13
	<i>Electronic Datasheet (EDS-file)</i>	13
Chapter 2	Basic Operation	
	General.....	14
	Data Exchange Model	15
	<i>Memory Map</i>	15
	<i>Data Exchange Example</i>	16
	Subnetwork Protocol.....	17
	<i>Protocol Modes</i>	17
	<i>Protocol Building Blocks</i>	17
	<i>Master Mode</i>	18
	<i>Generic Data Mode</i>	19
	<i>DF1 Master Mode</i>	19
	Data Representation on DeviceNet.....	20
	<i>General</i>	20
	<i>Data Types</i>	20
	<i>Memory Layout</i>	21
Chapter 3	Navigating ACM	
	Main Window.....	22
	<i>Drop-down Menus</i>	23
	<i>Toolbar Icons</i>	26

Chapter 4	Basic Settings	
	Fieldbus Settings.....	27
	Communicator Parameters	28
	Sub-network Parameters	29
Chapter 5	Nodes	
	General.....	30
	Adding & Managing Nodes.....	30
	Node Parameters	30
	<i>Master Mode and Generic Data Mode</i>	30
Chapter 6	Transactions	
	General.....	31
	Adding & Managing Transactions.....	32
	Transaction Parameters (Master Mode).....	33
	<i>Parameters (Query & Broadcast)</i>	33
	<i>Parameters (Response)</i>	34
	Transaction Parameters (Generic Data Mode).....	35
	<i>Produce Transactions</i>	35
	<i>Consume Transactions</i>	36
	Transaction Editor	37
Chapter 7	Frame Objects	
	General.....	38
	Adding and Editing Frame Objects	38
	Constant Objects (Byte, Word, Dword).....	39
	Limit Objects (Byte, Word, Dword)	40
	Data Object.....	41
	Variable Data Object	41
	Checksum Object.....	43
Chapter 8	Commands	
	General.....	44
	Adding & Managing Commands	44
	<i>Drop-down Menu</i>	45
	<i>Toolbar Icons</i>	45
	The Command Editor	46
	<i>General</i>	46
	<i>Basic Navigation</i>	46
	<i>Drop-down Menu</i>	47
	<i>Editing a Command</i>	47
	<i>Example: Specifying a Modbus-RTU Command in Master Mode</i>	48

Chapter 9	DF1 Protocol Mode	
	General.....	49
	Communicator Parameters	50
	Sub-network Parameters	51
	Node Parameters	52
	Services.....	52
	<i>Available Services</i>	53
	Integrity Check	54
	Read Diagnostics	54
	Read Data	55
	Write Data	55
Chapter 10	Sub-network Monitor	
	General.....	56
	Operation.....	56
Chapter 11	Node Monitor	
	General.....	57
	Navigating the Node Monitor.....	58
	<i>Drop-down Menu</i>	59
	<i>Toolbar Icons</i>	60
Chapter 12	Data Logger	
	General.....	61
	Operation.....	61
	Configuration	62
Chapter 13	Configuration Wizards	
	General.....	63
	Selecting a Wizard Profile	63
	Wizard - Modbus RTU Master	64
Chapter 14	Control and Status Registers	
	General.....	65
	<i>Handshaking Procedure</i>	65
	<i>Data Consistency</i>	66
	Status Register Contents (Gateway to Control System).....	67
	<i>General Information</i>	67
	<i>Status Codes in Master Mode and DF1 Master Mode</i>	67
	<i>Status Code in Generic Data Mode</i>	68
	Control Register Contents (Control System to Gateway).....	69
	<i>General Information</i>	69
	<i>Control Codes in Master Mode and DF1 Master Mode</i>	69
	<i>Control Codes in Generic Data Mode</i>	69

Chapter 15**CIP Object Implementation**

Identity Object, Class 01h.....	71
<i>General Information</i>	71
<i>Class Attributes</i>	71
<i>Instance Attributes</i>	71
Message Router, Class 02h.....	72
<i>General Information</i>	72
<i>Class Attributes</i>	72
<i>Instance Attributes</i>	72
DeviceNet Object, Class 03h	73
<i>General Information</i>	73
<i>Class Attributes</i>	73
<i>Instance Attributes</i>	73
Assembly Object, Class 04h	74
<i>General Information</i>	74
<i>Class Attributes</i>	74
<i>Instance Attributes - Instance/Connection Point 64h</i>	74
<i>Instance Attributes - Instance/Connection Point 96h</i>	74
Connection Object, Class 05h.....	75
<i>General Information</i>	75
<i>Class Attributes</i>	75
<i>Instance 1 & 10...14 (Explicit Messaging Connection) Attributes</i>	75
<i>Instance 2 (Polled Connection) Attributes</i>	75
<i>Instance 3 (Bit-strobe connection) Attributes</i>	76
<i>Instance 4 (COS/Cyclic connection) Attributes</i>	76
Acknowledge Handler Object, Class 2Bh	77
<i>General Information</i>	77
<i>Class Attributes</i>	77
<i>Instance Attributes</i>	77
Diagnostic Object, Class AAh.....	78
<i>General Information</i>	78
<i>Class Attributes</i>	78
<i>Instance Attributes, Instance 01h</i>	78
Parameter Data Input Mapping Object, Class B0h	79
<i>General Information</i>	79
<i>Class Attributes</i>	79
<i>Instance Attributes, Instance 01h</i>	79
Parameter Data Output Mapping Object, Class B1h	80
<i>General Information</i>	80
<i>Class Attributes</i>	80
<i>Instance Attributes, Instance 01h</i>	80

Chapter 16**Advanced Fieldbus Configuration**

General.....	81
Mailbox Editor.....	81

Appendix A	Parameter Data Initialization (Explicit Data)	
	General.....	82
	Add a Mailbox Message.....	82
	Mapping Input Parameter Data to DeviceNet.....	83
	Mapping Output Parameter Data to DeviceNet.....	85
Appendix B	Connector Pin Assignments	
	DeviceNet Connector.....	87
	Power Connector.....	87
	PC Connector.....	88
	Subnetwork Interface.....	89
	<i>General Information</i>	89
	<i>Bias Resistors (RS485 Only)</i>	89
	<i>Termination (RS485 & RS422 Only)</i>	89
	<i>Connector Pinout (DB9F)</i>	89
	<i>Typical Connection (RS485)</i>	90
	<i>Typical Connection (RS422 & 4-Wire RS485)</i>	90
	<i>Typical Connection (RS232)</i>	90
Appendix C	Technical Specification	
	Mechanical Properties.....	91
	Electrical Characteristics.....	91
	Environmental Characteristics.....	91
	Regulatory Compliance.....	92
Appendix D	Troubleshooting	
Appendix E	ASCII Table	

P. About This Document

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

P.1 Related Documents

Document name	Author
Anybus Communicator - DeviceNet Installation Sheet	HMS
DF1 Protocol and Command Set - Reference Manual, 1770-6.5.16, October 1996	Allen-Bradley

P.2 Document History

Summary of Recent Changes (3.00... 3.10)

Change	Page(s)
Device type changed from "Group two only server" to "Group two server"	9
Screenshots and descriptions of ABC Tool updated for Anybus Configuration Manager	Multiple
Changed "ABC" to "Communicator RS232/422/485"	Multiple
Amended description of "Update time" parameter	34, 35
Added description for Consume/Response to "Object Delimiter" parameter	42
Changed "Maximum Data Length" limit	42
Removed obsolete "Start Bits" parameter	51
Removed obsolete "ABCC ExtLink Wizard" entry	64
Replaced "Sales and Support" info with link to website	8
Added parameters to checksum object description	43
Minor text edits, typo corrections	Multiple

Summary of Recent Changes (3.10... 3.12)

Revision	Change	Page(s)
3.11	Added trademark symbol	Front cover
3.12	Added compliance/conformance info	92

Revision List

Revision	Date	Author	Chapter	Description
2.00	2003-12-03	PeP	All	Second major release
2.01	2004-03-08	PeP	16	Corrected Control Codes
2.50	2006-04-05	PeP	All	Major update
2.51	2006-12-22	PeP	-	Misc. minor corrections & updates
2.52	2009-04-23	KeL	All	Misc. minor corrections and updates
3.00	2011-02-09	KaD	All	Misc. corrections, new template and DF1 functionality
3.01	2011-09-30	KaD	All	Misc corrections and updates, new Anybus Configuration Manager
3.10	2015-02-20	ThN	All	Misc. corrections and updates, new Doc ID
3.11	2015-03-13	ThN	Cover	Added trademark symbol
3.12	2015-03-20	ThN	C	Added compliance/conformance info

P.3 Conventions & Terminology

The following conventions are used throughout this document:

- Numbered lists provide sequential steps
- Bulleted lists provide information, not procedural steps
- The term “user” refers to the person or persons responsible for installing the Anybus Communicator in a network.
- The term “ABC” refers to the Anybus Communicator.
- Hexadecimal values are written in the format 0xNNNN, where NNNN is the hexadecimal value.
- Decimal values are represented as NNNN where NNNN is the decimal value
- As in all communication systems, the terms “input” and “output” can be ambiguous, because their meaning depend on which end of the link is being referenced. The convention in this document is that “input” and “output” are always being referenced to the master/scanner end of the link.

P.3.1 Glossary

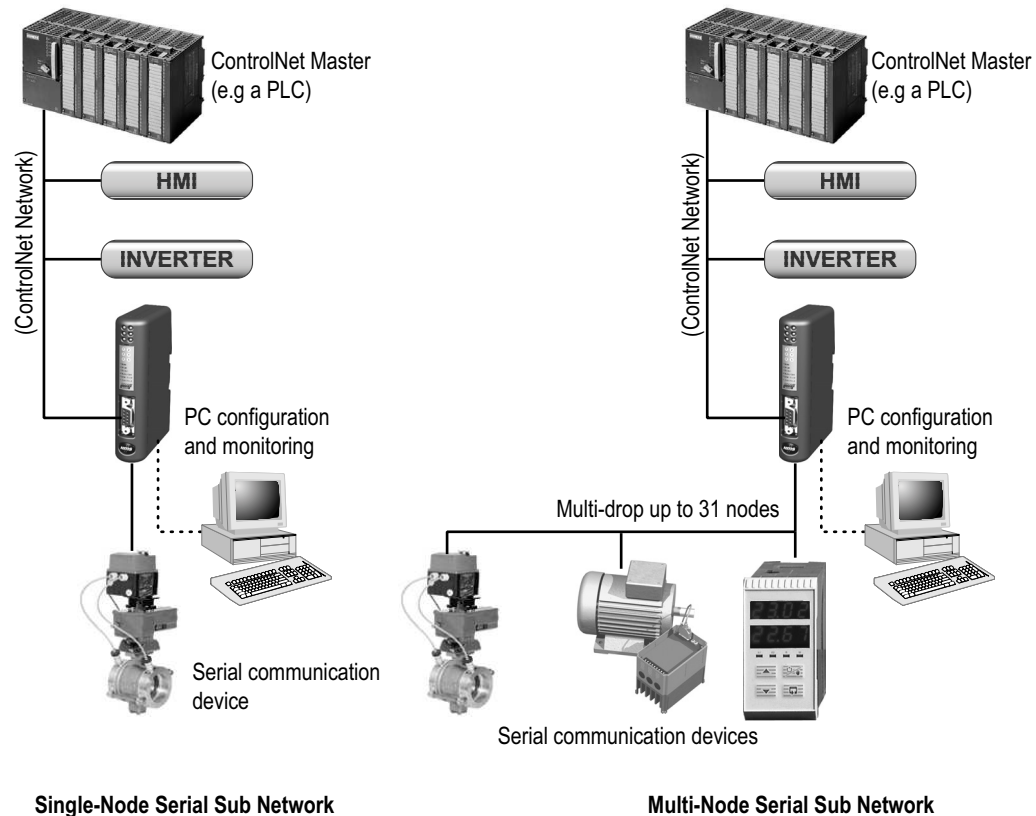
Term	Meaning
ABC	Anybus Communicator
ACM	Anybus Configuration Manager
Broadcaster	A protocol-specific node in the configuration that handles transactions destined to all nodes.
DEV	DeviceNet
Command	A predefined transaction
Configuration	List of configured nodes with transactions on the subnetwork
Fieldbus	The higher level network to which the communicator is connected
Fieldbus Control System	Fieldbus master
Frame Object	Low level entities which are used to describe the different parts of a transaction
Monitor	A tool for debugging the Anybus Communicator and the network connections
Node	A device in the configuration which defines the communication with a node on the subnetwork
Sub-network	The network that is logically located on a subsidiary level with respect to the fieldbus, and to which the Anybus Communicator acts as a gateway
Transaction	A generic building block that is used in the subnetwork configuration and defines the data that is sent and received on the subnetwork
User	Person or persons responsible for installing the Anybus Communicator
Higher Level Network	In this case, DeviceNet
Network	
Fieldbus	

P.4 Support

For general contact information and support, please refer to the contact and support pages at www.anybus.com

1. About the Anybus Communicator for DeviceNet

The Anybus Communicator for DeviceNet acts as a gateway between virtually any serial application protocol and a DeviceNet-based network. Integration of industrial devices is enabled without loss of functionality, control and reliability, both when retro-fitting to existing equipment as well as when setting up new installations.



Subnetwork

The Anybus Communicator can address up to 31 nodes, and supports the following physical standards:

- RS-232
- RS-422
- RS-485

DeviceNet Interface

DeviceNet connectivity is provided through the patented Anybus technology; a proven industrial communication solution used all over the world by leading manufacturers of industrial automation products.

- Communications Adapter, profile no. 12
- Group two server
- MacID and baudrate configuration via on-board switches
- Polled, Change-of-state and Bit strobed I/O

1.1 External View

For wiring and pin assignments, see Appendix “Connector Pin Assignments” on page 87.

A: DeviceNet Connector

This connector is used to connect the Anybus Communicator to the fieldbus network.

See also...

- “DeviceNet Connector” on page 87

B: Configuration Switches

See also...

- “Configuration Switches” on page 11

C: Status LEDs

See also...

- “Status LEDs” on page 11

D: PC-connector

This connector is used to connect the gateway to a PC for configuration and monitoring purposes.

See also...

- “PC Connector” on page 88

E: Subnetwork Connector

This connector is used to connect the gateway to the serial subnetwork.

See also...

- “Subnetwork Interface” on page 89

F: Power Connector

This connector is used to apply power to the gateway.

See also...

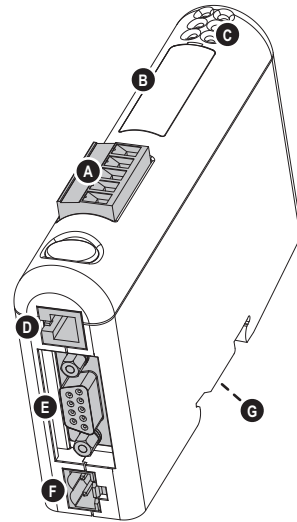
- “Power Connector” on page 87
- “Connector Pin Assignments” on page 87

G: DIN-rail Connector

The DIN-rail mechanism connects the gateway to PE (Protective Earth).

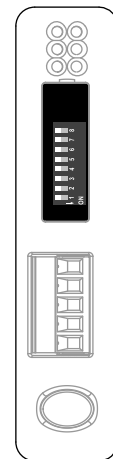
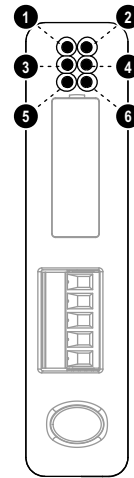
See also...

- “Connector Pin Assignments” on page 87



1.2 Status LEDs

#	State	Status
1 - Network Status	Off	Not powered / not online
	Green	Link OK, On line, Connected
	Green, flashing	On line, Not connected
	Red	Critical Link Failure
	Red, flashing	Connection timeout
2 - Module Status	Off	No power to device
	Green	Device Operational
	Green, flashing	Data size bigger than configured
	Red	Unrecoverable fault
	Red, flashing	Minor Fault
3 - (Not used)	-	-
4 - (Not used)	-	-
5 - Subnet Status ^a	Off	Power off
	Green, flashing	Running correctly, but one or more transaction error(s) have occurred
	Green	Running
	Red	Transaction error/timeout or subnet stopped
6 - Device Status	Off	Power off
	Alternating Red/Green	Invalid or missing configuration
	Green	Initializing
	Green, flashing	Running
	Red	Bootloader mode ^b
	Red, flashing	If the Device Status LED is flashing in a sequence starting with one or more red flashes, please note the sequence pattern and contact the HMS support department



- a. This led turns green when all transactions have been active at least once. This includes any transactions using “change of state” or “change of state on trigger”. If a timeout occurs on a transaction, this led will turn red.
- b. The gateway is in bootloader mode, and firmware must be restored in order for it to work properly. Start up the Anybus Configuration Manager and connect to the Anybus Communicator. Choose Tools/Options/Module. Click “Factory Restore” to restore firmware. See “Tools” on page 24.

1.3 Configuration Switches

The configuration switches are used to set the DeviceNet MacID and baudrate settings.

Note that the these settings cannot be changed during runtime, i.e. the Anybus Communicator requires a reset for any changes to have effect.

The switches are interpreted as follows:

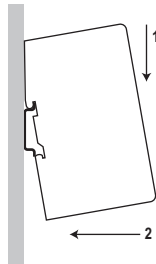
Baudrate	sw. 1	sw. 2	Mac ID	sw. 3	sw. 4	sw. 5	sw. 6	sw. 7	sw. 8
125k	OFF	OFF	0	OFF	OFF	OFF	OFF	OFF	OFF
250K	OFF	ON	1	OFF	OFF	OFF	OFF	OFF	ON
500K	ON	OFF
(reserved)	ON	ON	62	ON	ON	ON	ON	ON	OFF
			63	ON	ON	ON	ON	ON	ON

1.4 Hardware Installation

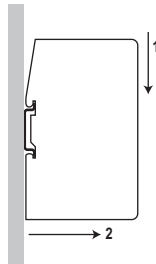
Perform the following steps when physically installing the Anybus Communicator:

1. Snap the gateway on to the DIN-rail (See “External View” on page 10)

The DIN-rail mechanism works as follows:



To snap the gateway *on*, first press it downwards (1) to compress the spring in the DIN-rail mechanism, then push it against the DIN-rail as to make it snap on (2)



To snap the gateway *off*, push it downwards (1) and pull it out from the DIN-rail (2), as to make it snap off from the DIN-rail

2. Connect the gateway to the DeviceNet network
3. Connect the gateway to the serial subnetwork
4. Connect the gateway to a free COM-port on the PC via the PC-cable
5. Set the DeviceNet baudrate and Mac-ID using the on-board switches
6. Connect the power cable and apply power
7. Start the Anybus Configuration Manager program on the PC
(The Anybus Configuration Manager attempts to detect the serial port automatically. If not successful, select the correct port manually in the “Port”-menu)
8. Configure the ABC using the Anybus Configuration Manager and download the configuration

1.5 Software Installation

1.5.1 Anybus Configuration Manager

System requirements

- Pentium 133 MHz or higher
- 650 MB of free space on the hard drive
- 32 MB RAM
- Screen resolution 800 x 600 (16 bit color) or higher
- Microsoft Windows® 2000 / XP / Vista / 7 (32- or 64-bit)
- Internet Explorer 4.01 SP1 or newer (or any equivalent browser)

Installation

- **Anybus Communicator resource CD**
 - Insert the CD and follow the on-screen instructions.
 - If the installation does not start automatically: right-click on the CD drive icon and select “Explore” to show the contents of the CD. Locate the installation executable and double-click on it to start the installation, then follow the on-screen instructions.
- **From HMS website**
 - Download the latest version of Anybus Configuration Manager from www.anybus.com.
 - Unzip the archive on your computer and double-click on the installation executable.

1.5.2 Electronic Datasheet (EDS-file)

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

The latest version of this file can be obtained from the HMS website www.anybus.com.

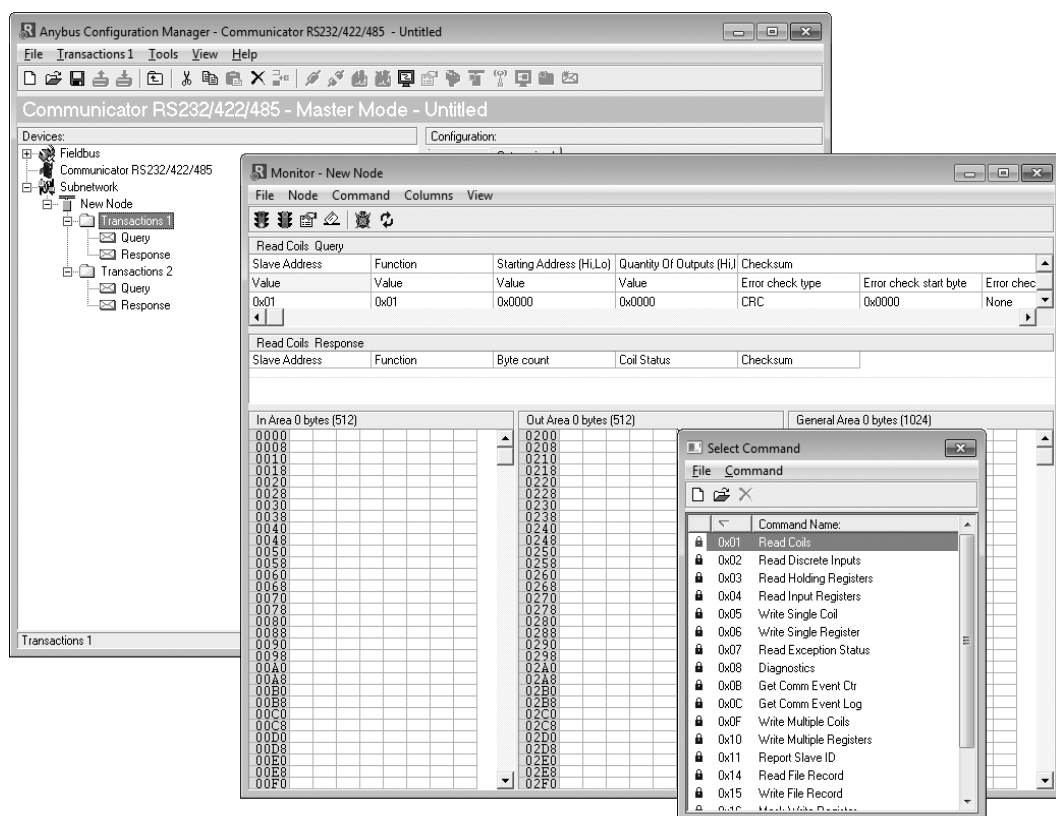
2. Basic Operation

2.1 General

The Anybus Communicator gateway is designed to exchange data between a serial sub-network and a higher level network. Unlike most other gateway devices of similar kind, it does not have a fixed protocol for the sub-network, and can be configured to handle almost any form of serial communication.

The gateway can issue serial telegrams cyclically, on change of state, or based on trigger events issued by the control system of the higher level network (i.e. the fieldbus master or PLC). It can also monitor certain aspects of the sub-network communication and notify the higher level network when data has changed.

An essential part of the Anybus Communicator package is Anybus Configuration Manager (ACM), a Windows®-based application used to supply the gateway with a description of the sub-network protocol. No programming skills are required; instead, a visual protocol description-system is used to specify the different parts of the serial communication.



2.2 Data Exchange Model

Internally, the data exchanged on the subnetwork, and the data exchanged on the higher level network, resides in the same memory.

This means that in order to exchange data with the subnetwork, the higher level network simply reads and writes data to memory locations specified using the Anybus Configuration Manager. The very same memory locations can then be exchanged on the subnetwork.

The internal memory buffer is divided into three areas based on their function:

- **Input Data (512 bytes)**

This area can be read by the higher level network.

(how this data is represented on the higher level network will be described later in this chapter).

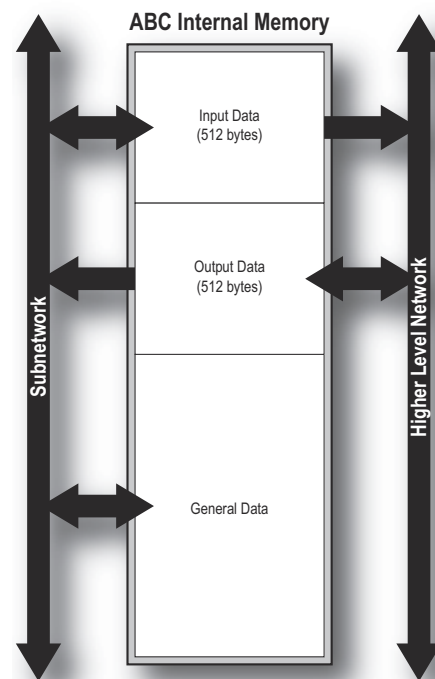
- **Output Data (512 bytes)**

This area can be written to by the higher level network.

(how this data is represented on the higher level network will be described later in this chapter).

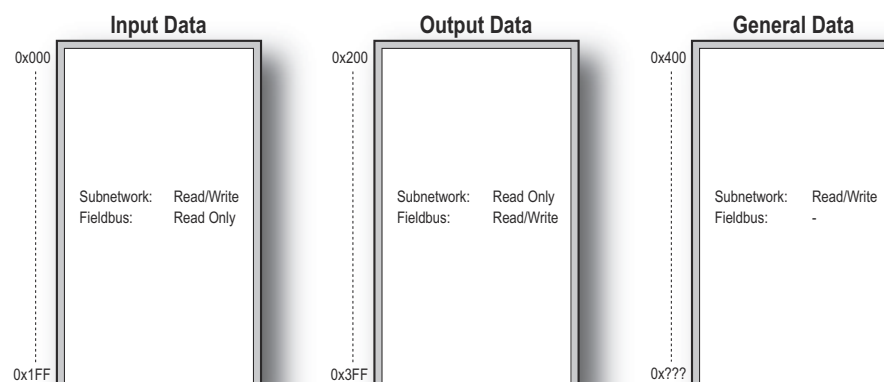
- **General Data**

This area is not exchanged on the higher level network, and can be used for transfers between individual nodes on the subnetwork, or as a general “scratch pad” for data. The actual size of this area depends on the amount of data that is exchanged on the subnetwork. The gateway can handle up to 1024 bytes of General Data.



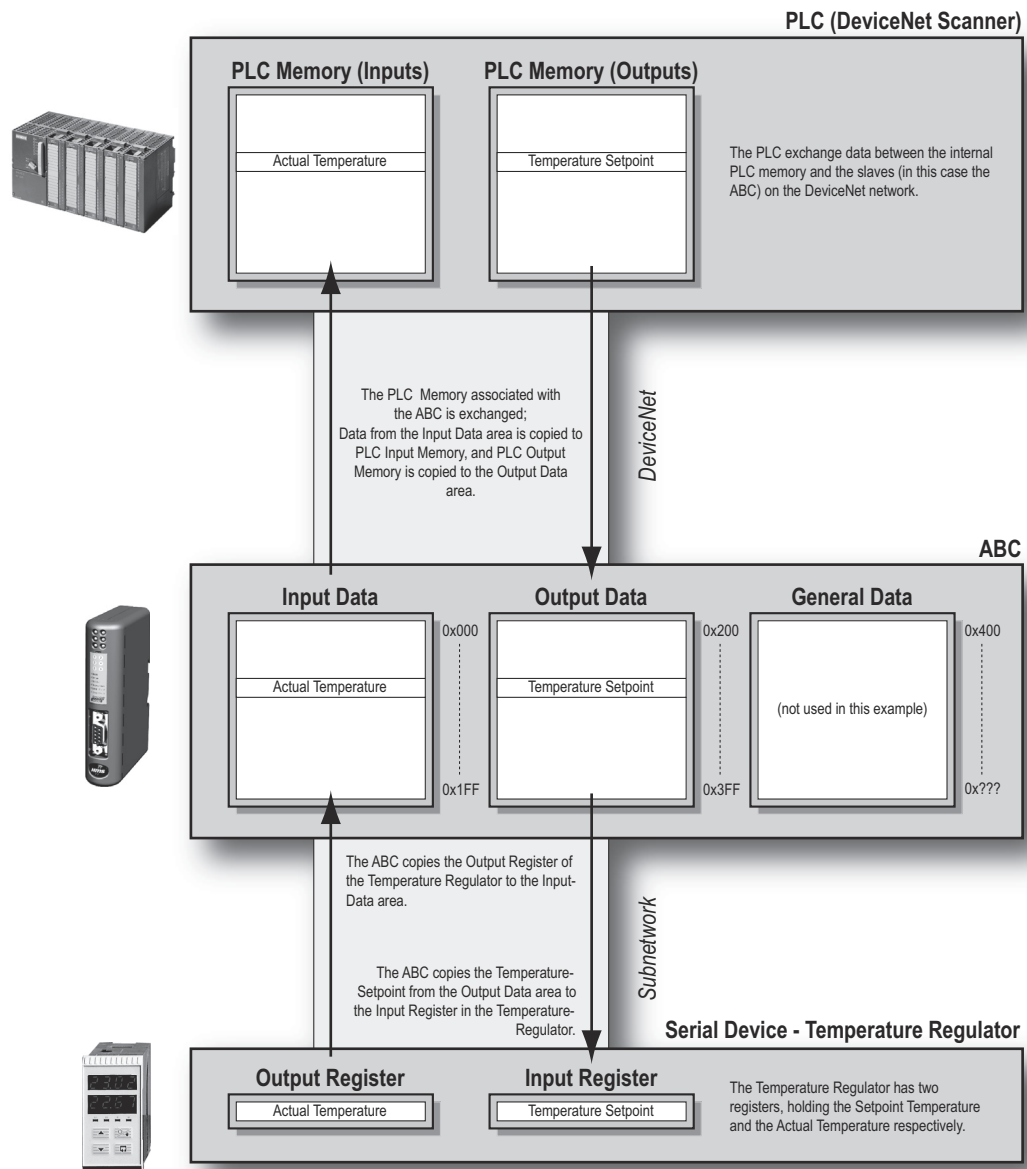
2.2.1 Memory Map

When building the subnetwork configuration using the Anybus Configuration Manager, the different areas described above are mapped to the memory locations (addresses) specified below.



2.2.2 Data Exchange Example

In the following example, a temperature regulator on the subnetwork exchanges information with a PLC on the higher level network, via the internal memory buffers in the Anybus Communicator.



2.3 Subnetwork Protocol

2.3.1 Protocol Modes

The Anybus Communicator features three distinct modes of operation regarding the subnetwork communication, called “Master Mode”, “DF1 Master Mode” and “Generic Data Mode”. Note that the protocol mode only specifies the basic communication model, not the actual subnetwork protocol.

- **Master Mode**

In this mode, the gateway acts as a master on the subnetwork, and the serial communication takes place in a query-response fashion. The nodes on the network are not permitted to issue messages unless they have been addressed by the gateway first.

For more information about this mode, see “Master Mode” on page 18.

- **DF1 Master Mode**

In this mode, the gateway acts as a master on the subnetwork, using the DF1 protocol. The serial communication takes place in a query-response fashion. For more information about this mode, see “DF1 Protocol Mode” on page 49.

- **Generic Data Mode**

In this mode, there is no master-slave relationship between the subnetwork nodes and the gateway; any node on the subnetwork, including the gateway, may spontaneously produce or consume messages.

For more information about this mode, see “Generic Data Mode” on page 19.

2.3.2 Protocol Building Blocks

The following building blocks are used in Anybus Configuration Manager to describe the subnetwork communication. How these blocks apply to the two protocol modes will be described later in this document.

- **Node**

A node represents a single device on the subnetwork. Each node can be associated with a number of transactions, see below.

- **Transaction**

A “transaction” represents a complete serial telegram, and consists of a number of frame objects (see below). Each transaction is associated with a set of parameters controlling how and when to use it on the subnetwork.

- **Commands**

A “command” is simply a predefined transaction stored in a list in the Anybus Configuration Manager. This simplifies common operations by allowing transactions to be stored and reused.

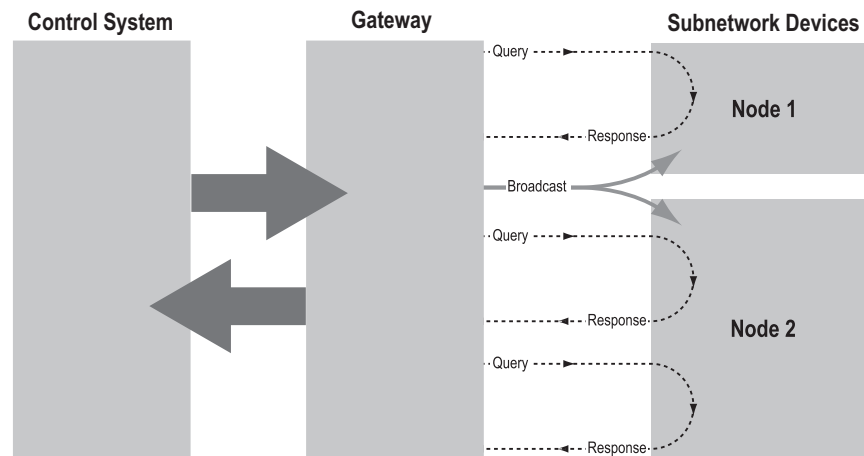
- **Frame Object**

“Frame objects” are low level entities used to compose a transaction (see above). A frame object can represent a fixed value (a constant), a range of values (limit objects), a block of data or a calculated checksum.

2.3.3 Master Mode

In this mode, the communication is based on a query-response scheme; when the Anybus Communicator issues a query on the subnetwork, the addressed node is expected to issue a response to that query. Nodes are not permitted issue responses spontaneously, i.e. without first receiving a query.

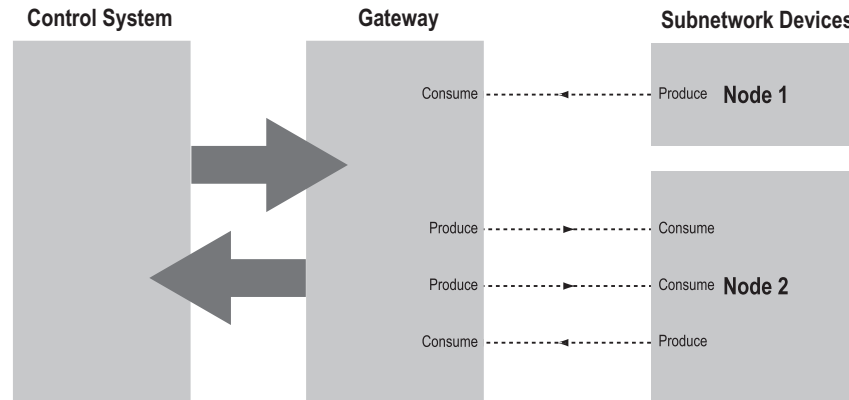
There is one exception to this rule; the broadcaster. Most protocols offer some way of broadcasting messages to all nodes on the network, without expecting them to respond to the broadcasted message. This is also reflected in the gateway, which features a dedicated broadcaster node.



In Master Mode, Anybus Configuration Manager comes preloaded with most commonly used Modbus RTU commands, which can conveniently be reached by right-clicking on a node in the Anybus Configuration Manager and selecting “Insert New Command”. Note however that this does not in any way prevent other protocols based on the same query-response message-scheme to be implemented.

2.3.4 Generic Data Mode

In this mode, there is no master-slave relationship between the nodes on the subnetwork and the gateway. Any node, including the gateway, may spontaneously produce or consume a message. Nodes do not have to respond to messages, nor do they have to wait for a query in order to send one.



In the figure above, the gateway “consumes” data that is “produced” by a node on the subnetwork. This “consumed” data can then be accessed from the higher level network. This also works the other way around; the data received from the higher level network is used to “produce” a message on the subnetwork to be “consumed” by a node.

2.3.5 DF1 Master Mode

Please refer to “DF1 Protocol Mode” on page 49.

2.4 Data Representation on DeviceNet

2.4.1 General

DeviceNet is based on the Control and Information protocol (CIP) which is also the application layer for ControlNet and EtherNet/IP.

The input and output data is accessed using I/O connections or explicit messages towards the Assembly Object and the Parameter Input/Output Mapping Objects.

See also...

- “CIP Object Implementation” on page 70

2.4.2 Data Types

The input and output data holds two types of data; I/O Data and parameter data. I/O Data is exchanged on change of value, and can be accessed using I/O connections towards the Assembly Object.

Parameter data can be accessed acyclically via the Parameter Input- and Output Mapping Objects. Note however that each instance attribute within these objects must be created manually using the Anybus Configuration Manager. For more information see “Parameter Data Initialization (Explicit Data)” on page 82.

See also...

- “Assembly Object, Class 04h” on page 74
- “Parameter Data Input Mapping Object, Class B0h” on page 79
- “Parameter Data Output Mapping Object, Class B1h” on page 80
- “Fieldbus Settings” on page 27

2.4.3 Memory Layout

The I/O sizes are specified using the Anybus Configuration Manager and correlates to gateway memory as follows:

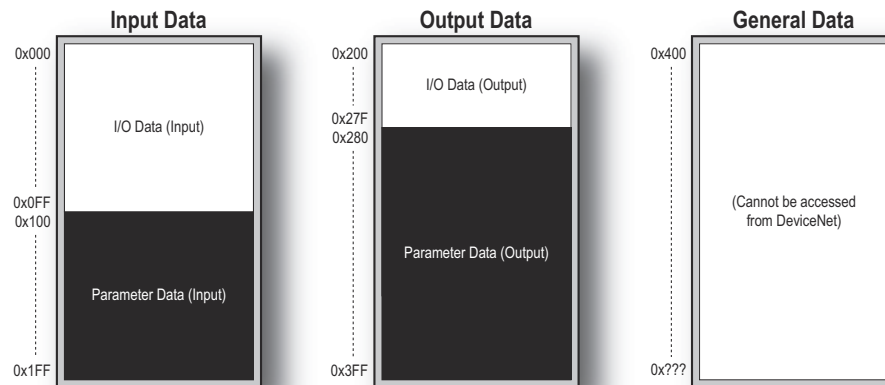
Example:

In this example, the I/O Sizes for the gateway have been set to the following values:

IO Size In= 256 bytes(0x0100)

IO Size Out= 128 bytes(0x0080)

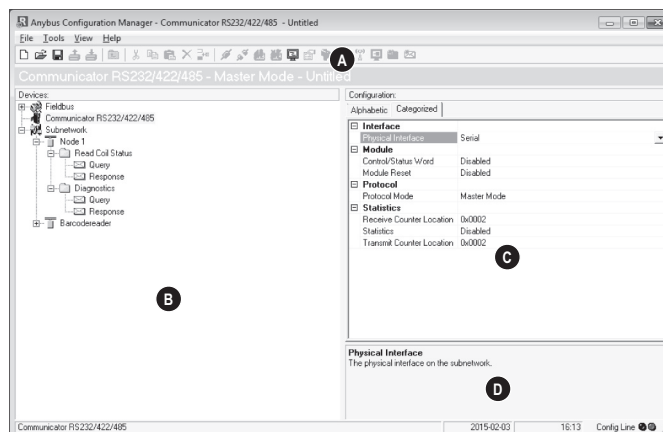
Resulting memory layout:



3. Navigating ACM

3.1 Main Window

The main window in ACM can be divided into 4 sections as follows:



- **A: Drop-down Menus & Tool Bar**

The second drop-down menu from the left will change depending on the current context. The Tool Bar provides quick access to the most frequently used functions.

- **B: Navigation Section**

This section is the main tool for selecting and altering different levels of the sub-network configuration.

Entries preceded by a “+” holds further configuration parameters or “sub menus”. To gain access to these parameters, the entry must be expanded by clicking “+”.

There are three main levels in the navigation window, namely Fieldbus, Communicator RS232/422/485, and Subnetwork.

Right-clicking on entries in this section brings out additional selections related to that particular entry.

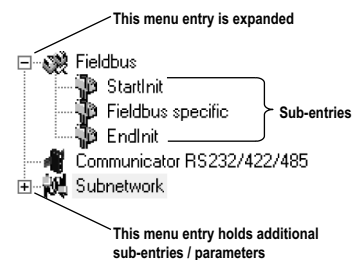
- **C: Parameter Section**

This section holds a list of parameters or options related to the currently selected entry in the Navigation Section.

The parameter value may be specified either using a selection box or manually, depending on the parameter itself. Values can be specified in decimal form (e.g. “42”), or in hexadecimal format (e.g. “0x2A”).

- **D: Information Section**

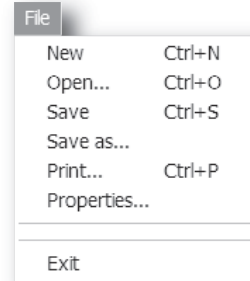
This section holds information related to the currently selected parameter.



3.1.1 Drop-down Menus

File

- **New**
Create a new configuration.
See also “Configuration Wizards” on page 64.
- **Open...**
Open a previously created configuration.
- **Save**
Save the current configuration.
- **Save As...**
Save the current configuration under a new name.
- **Print...**
Send details about the current configuration to a printer.
- **Properties...**
Set the name and (optional) passwords for the configuration.



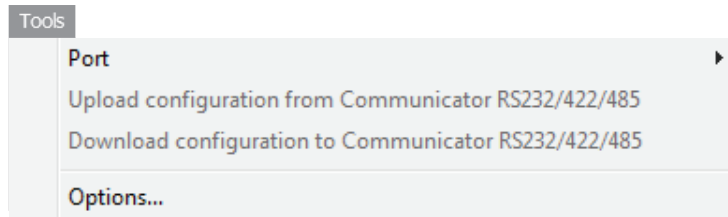
Item	Description
Select a Name for the Configuration	Enter a descriptive name for the new configuration
Enable Password	Enables password protection
Download Password(6)	Set passwords for downloading and uploading the configuration (max. 6 characters)
Upload Password(6)	

CAUTION: Always keep a copy of the password in a safe place. A lost password cannot be retrieved!

- **Exit**
Close ACM.

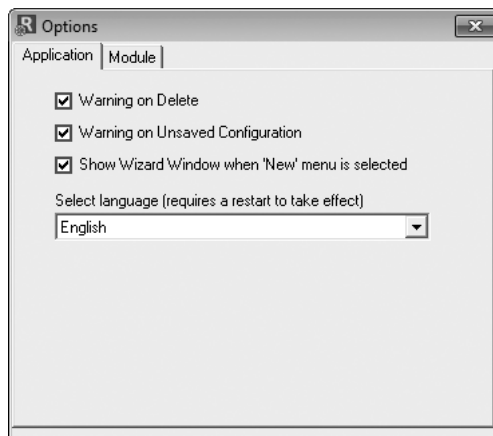


Tools



- **Port**
Select the COM-port used for the configuration of the gateway.
- **Upload configuration from Communicator RS232/422/485**
Upload the configuration from the gateway to ACM.
- **Download configuration to Communicator RS232/422/485**
Download the current configuration to the gateway.
- **Start Logging**
Start the Data Logger (see “Data Logger” on page 61).
Note that when the Data Logger is active, this menu entry is changed to “Stop Logging”.
- **Options**

This will open the following window:



Item	Description
Warning on Delete	A confirmation dialog is displayed each time something is deleted.
Warning on Unsaved Configuration	A confirmation dialog is displayed when closing ACM with unsaved data.
Show Wizard when “New” menu is selected	The Wizard is displayed each time a new configuration is created.
Select language	Selects which language to use. The new setting will be active the next time the program is launched.