



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 EtherNet/IP[™] / Modbus-TCP

Doc. Id. HMSI-27-314
Rev. 3.10

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. EtherNet/IP™ 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 EtherNet/IP / Modbus-TCP User Manual
Copyright© HMS Industrial Networks AB
Doc: HMSI-27-314

Table of Contents

Preface	About This Document	
	Related Documents	9
	Document History	9
	Conventions & Terminology	10
	<i>Glossary</i>	10
	Sales and Support	10
Chapter 1	About the Anybus Communicator for EtherNet/IP	
	External View.....	12
	Status LEDs	13
	Configuration Switches	14
	Hardware Installation.....	15
	Software Installation	16
	<i>Anybus Configuration Manager</i>	16
Chapter 2	Basic Operation	
	General.....	17
	Data Exchange Model	18
	<i>Memory Map</i>	18
	<i>Data Exchange Example</i>	19
	Sub-network Protocol.....	20
	<i>Protocol Modes</i>	20
	<i>Protocol Building Blocks</i>	20
	<i>Master Mode</i>	21
	<i>Generic Data Mode</i>	22
	<i>DF1 Master Mode</i>	22
	EtherNet/IP.....	23
	<i>General</i>	23
	<i>Data Types</i>	23
	<i>Memory Layout</i>	24
	Modbus-TCP.....	25
	<i>General</i>	25
	<i>Addressing Modes</i>	25
	<i>Supported Exception Codes</i>	25
	<i>Modbus Addressing Mode</i>	26
	<i>Anybus Addressing Mode</i>	27

Chapter 3	File System	
	General.....	28
	File System Overview	29
	System Files.....	29
Chapter 4	FTP Server	
	General.....	30
	FTP Connection Example (Windows Explorer).....	31
Chapter 5	Telnet Server	
	General.....	32
	General Commands	33
	Diagnostic Commands	34
	File System Operations.....	34
Chapter 6	Basic Network Configuration	
	General Information.....	37
	Ethernet Configuration File ('ethcfg.cfg')	38
	<i>General</i>	38
	IP Access Control	39
	On/Offline Configuration.....	40
	Anybus IPconfig (HICP)	41
Chapter 7	Web Server	
	General.....	42
	Authorization	43
	Content Types.....	44
Chapter 8	Server Side Include (SSI)	
	Functions	46
	Changing SSI output.....	55
	<i>SSI Output String File</i>	55
	<i>Temporary SSI Output change</i>	56
Chapter 9	E-mail Client	
	General.....	57
	E-mail Definitions.....	58
Chapter 10	Navigating ACM	
	Main Window.....	59
	<i>Drop-down Menus</i>	60
	<i>Toolbar Icons</i>	63

Chapter 11	Basic Settings	
	Fieldbus Settings.....	64
	Communicator Parameters	65
	Sub-network Parameters	66
Chapter 12	Nodes	
	General.....	67
	Adding & Managing Nodes.....	67
	Node Parameters	67
	<i>Master Mode and Generic Data Mode</i>	67
Chapter 13	Transactions	
	General.....	68
	Adding & Managing Transactions.....	69
	Transaction Parameters (Master Mode).....	70
	<i>Parameters (Query & Broadcast)</i>	70
	<i>Parameters (Response)</i>	71
	Transaction Parameters (Generic Data Mode).....	72
	<i>Produce Transactions</i>	72
	<i>Consume Transactions</i>	73
	Transaction Editor	74
Chapter 14	Frame Objects	
	General.....	75
	Adding and Editing Frame Objects	75
	Constant Objects (Byte, Word, Dword).....	76
	Limit Objects (Byte, Word, Dword)	77
	Data Object.....	78
	Variable Data Object	78
	Checksum Object.....	80
Chapter 15	Commands	
	General.....	81
	Adding & Managing Commands	81
	<i>Drop-down Menu</i>	82
	<i>Toolbar Icons</i>	82
	The Command Editor	83
	<i>General</i>	83
	<i>Basic Navigation</i>	83
	<i>Drop-down Menu</i>	84
	<i>Editing a Command</i>	84
	<i>Example: Specifying a Modbus-RTU Command in Master Mode</i>	85

Chapter 16	DF1 Protocol Mode	
	General.....	86
	Communicator Parameters	87
	Sub-network Parameters	88
	Node Parameters	89
	Services.....	89
	<i>Available Services</i>	90
	Integrity Check	91
	Read Diagnostics	91
	Read Data	92
	Write Data	92
Chapter 17	Sub-network Monitor	
	General.....	93
	Operation.....	93
Chapter 18	Node Monitor	
	General.....	94
	Navigating the Node Monitor.....	95
	<i>Drop-down Menu</i>	96
	<i>Toolbar Icons</i>	97
Chapter 19	Data Logger	
	General.....	98
	Operation.....	98
	Configuration	99
Chapter 20	Configuration Wizards	
	General.....	100
	Selecting a Wizard Profile	100
	Wizard - Modbus RTU Master	101
Chapter 21	Control and Status Registers	
	General.....	102
	<i>Handshaking Procedure</i>	102
	<i>Data Consistency</i>	103
	Status Register Contents (Gateway to Control System).....	104
	<i>General Information</i>	104
	<i>Status Codes in Master Mode and DF1 Master Mode</i>	104
	<i>Status Code in Generic Data Mode</i>	105
	Control Register Contents (Control System to Gateway).....	106
	<i>General Information</i>	106
	<i>Control Codes in Master Mode and DF1 Master Mode</i>	106
	<i>Control Codes in Generic Data Mode</i>	106

Chapter 22	CIP Object Implementation	
	General.....	107
	Identity Object, Class 01h.....	108
	<i>General Information</i>	108
	<i>Class Attributes</i>	108
	<i>Instance Attributes</i>	108
	Message Router, Class 02h.....	109
	<i>General Information</i>	109
	<i>Class Attributes</i>	109
	<i>Instance Attributes</i>	109
	Assembly Object, Class 04h	110
	<i>General Information</i>	110
	<i>Class Attributes</i>	110
	<i>Instance 64b (100) Attributes</i>	110
	<i>Instance 96b (150) Attributes</i>	110
	<i>Instance C6b (198) Attributes (Heartbeat Input-Only)</i>	111
	<i>Instance C7b (199) Attributes (Heartbeat, Listen-Only)</i>	111
	Diagnostic Object, Class AAh.....	111
	<i>General Information</i>	111
	<i>Class Attributes</i>	111
	<i>Instance Attributes, Instance 01b</i>	111
	Parameter Data Input Mapping Object, Class B0h	112
	<i>General Information</i>	112
	<i>Class Attributes</i>	112
	<i>Instance Attributes, Instance 01b</i>	112
	Parameter Data Output Mapping Object, Class B1h	113
	<i>General Information</i>	113
	<i>Class Attributes</i>	113
	<i>Instance Attributes, Instance 01b</i>	113
	Port Object, Class F4h	114
	<i>General Information</i>	114
	<i>Class Attributes</i>	114
	<i>Instance Attributes, Instance 02h</i>	114
	TCP/IP Interface Object, Class F5h	115
	<i>General Information</i>	115
	<i>Class Attributes</i>	115
	<i>Instance Attributes</i>	115
	Ethernet Link Object, Class F6h	116
	<i>General Information</i>	116
	<i>Class Attributes</i>	116
	<i>Instance Attributes</i>	117
Chapter 23	Advanced Fieldbus Configuration	
	General.....	118
	Mailbox Editor.....	118

Appendix A	Parameter Data Initialization (Explicit Data)	
	General.....	119
	Add a Mailbox Message.....	119
	Mapping Input Parameter Data to EtherNet/IP.....	120
	Mapping Output Parameter Data to EtherNet/IP.....	122
Appendix B	Connector Pin Assignments	
	Ethernet Connector.....	124
	Power Connector.....	124
	PC Connector.....	125
	Sub-network Interface.....	126
	<i>General Information</i>	126
	<i>Bias Resistors (RS485 Only)</i>	126
	<i>Termination (RS485 & RS422 Only)</i>	126
	<i>Connector Pinout (DB9F)</i>	126
	<i>Typical Connection (RS485)</i>	127
	<i>Typical Connection (RS422 & 4-Wire RS485)</i>	127
	<i>Typical Connection (RS232)</i>	127
Appendix C	Technical Specification	
	Mechanical Properties.....	128
	Electrical Characteristics.....	128
	Environmental Characteristics.....	128
	Regulatory Compliance.....	129
Appendix D	Troubleshooting	
Appendix E	ASCII Table	
Appendix F	Copyright Notices	

P. About This Document

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

P.1 Related Documents

Document name	Author
ABC-EIP Installation Leaflet	HMS
DF1 Protocol and Command Set - Reference Manual, 1770-6.5.16, October 1996	Allen-Bradley
Open Modbus-TCP Specification, Release 1.0	Schneider Electric
RFC 821	Network Working Group
RFC 1918	Network Working Group
ENIP Specifications	ODVA

P.2 Document History

Summary of Recent Changes (3.03... 3.10)

Change	Page(s)
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	71, 72
Added description for Consume/Response to "Object Delimiter" parameter	79
Changed "Maximum Data Length" limit	79
Removed obsolete "Start Bits" parameter	88
Removed obsolete "ABCC ExtLink Wizard" entry	100
Replaced "Sales and Support" info with link to website	8
Added parameters to checksum object description	80
Minor text edits, typo corrections	Multiple
Updated screenshots in examples	120, 122

Revision List

Revision	Date	Author	Chapter	Description
2.00	2006-03-27	PeP	All	1st release
2.01	2006-12-22	PeP	All	Misc. minor corrections
2.02	2008-02-08	PeP	2, 8, A	Minor update
2.03	2008-11-03	HeS	1	Minor update
2.04	2009-04-24	KeL	All	Misc. minor corrections and updates
3.00	2011-02-01	KaD	All	Misc. corrections, new template and DF1 functionality
3.01	2011-09-30	KaD	All	Misc corrections and updates, new Anybus Configuration Manager name
3.02	2011-11-15	KaD	P, 2, 3, 6, 8	Minor corrections and updates
3.03	2012-06-08	KaD	P, 8, 22	Minor updates
3.10	March 2015	ThN	All	Misc. corrections and updates, new Doc. ID.

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

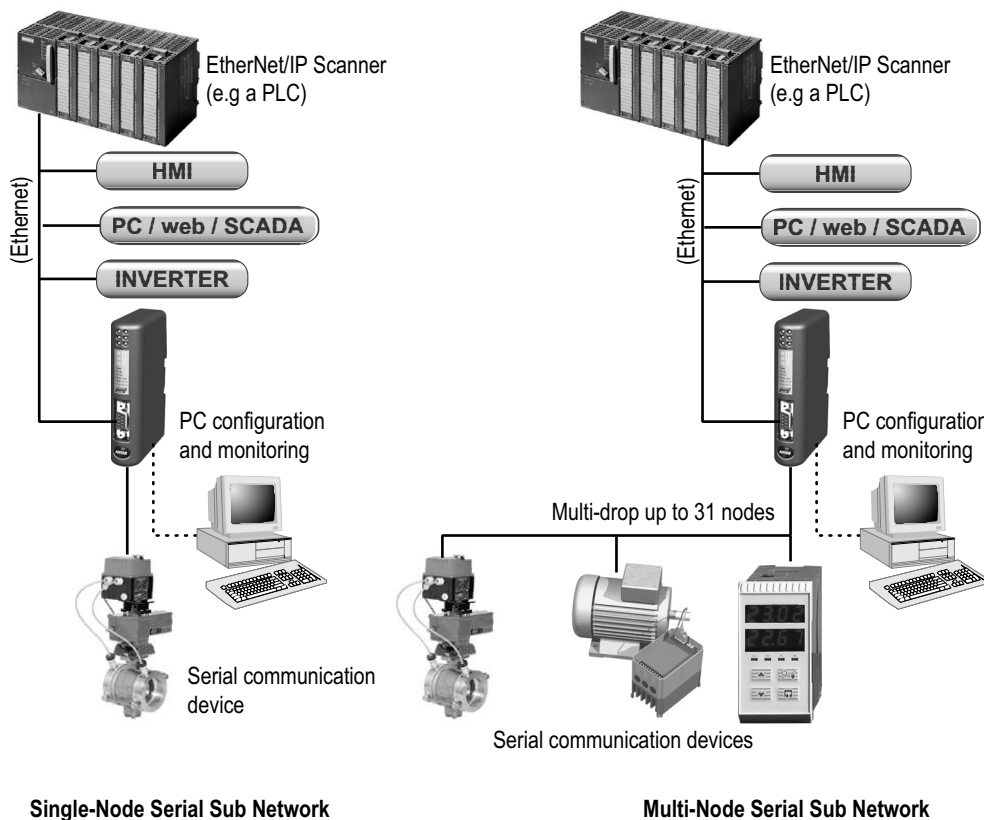
Term	Meaning
ABC	Anybus Communicator
ACM	Anybus Configuration Manager
EIP	EtherNet/IP
Broadcaster	A protocol specific node in the sub-network scan that hold transactions destined to all nodes
Command	A protocol specific transaction.
Configuration	List of configured nodes with transactions on the sub-network
Fieldbus	The network to which the communicator is connected.
Frame	Higher level series of bytes forming a complete telegram on the sub-network
Monitor	A tool for debugging the Anybus Communicator and the network connections
Node	A device in the scan-list that defines the communication with a slave on the sub-network
Scan list	List of configured slaves with transactions on the sub-network
sub-network	The network that logically is 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 sub-network scan-list and defines the data that is sent out the sub-network
Fieldbus Control System	Fieldbus master
Higher Level Network	In this case, Ethernet (including EtherNet/IP and Modbus-TCP)
Network	
Fieldbus	

P.4 Sales and 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 EtherNet/IP

The Anybus Communicator for EtherNet/IP acts as a gateway between virtually any serial application protocol and an EtherNet/IP-based network. Integration of industrial devices is enabled with no loss of functionality, control and reliability, both when retro-fitting to existing equipment as well as when setting up new installations.



Sub-network

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

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

Ethernet Interface

Ethernet 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.

- EtherNet/IP group 2 and 3 server
- Modbus-TCP slave functionality
- Server Side Include (SSI) functionality
- Web server and E-mail client capabilities
- FTP & Telnet servers
- 10/100 Mbit/s, twisted pair

1.1 External View

For wiring and pin assignments, see “Connector Pin Assignments” on page 124.

A: Ethernet Connectors

These connectors are used to connect the Anybus Communicator to the network.

See also...

- “Ethernet Connector” on page 124

B: Configuration Switches

See also...

- “Configuration Switches” on page 14

C: Status LEDs

See also...

- “Status LEDs” on page 13

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 125

E: Sub-network Connector

This connector is used to connect the gateway to the serial sub-network.

See also...

- “Sub-network Interface” on page 126

F: Power Connector

This connector is used to apply power to the gateway.

See also...

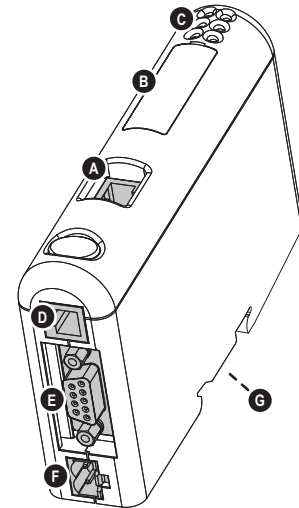
- “Power Connector” on page 124

G: DIN-rail Connector

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

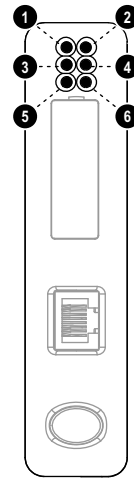
See also...

- “Configuration Switches” on page 14



1.2 Status LEDs

#	State	Status
1 - Module Status <i>(EtherNet/IP only)</i>	Off	No power
	Green	Controlled by a scanner in run state
	Green, flashing	Not configured, or scanner in idle state
	Red	Major fault (unrecoverable)
	Red, flashing	Minor fault (recoverable)
	Alternating Red/Green	Self-test
2 - Network Status <i>(EtherNet/IP only)</i>	Off	No IP address (or no power)
	Green	Online, EtherNet/IP connection(s) established
	Green, flashing	Online, no EtherNet/IP connections established
	Red	Duplicate IP address detected, fatal error
	Red, flashing	One or more connections timed out
	Alternating Red/Green	Self-test
3 - Link	Off	No link (or no power)
	Green	Connected to an ethernet network
4 - Activity	Off	No ethernet activity (or no power)
	Green	Receiving or transmitting ethernet packet
5 - Subnet Status ^a	Off	(no power)
	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	(no power)
	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 HMS support.



- a. This LED shows 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 show 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. Select **Tools/Options/Module**. Click **Factory Restore** to restore firmware. See “Tools” on page 61.

1.3 Configuration Switches

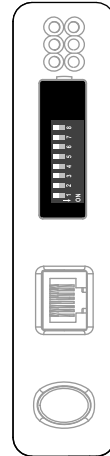
If set to a nonzero value, the configuration switches forces the Anybus Communicator to use an IP address in the range 192.168.0.1 - 192.168.0.255.

If set to zero, these settings are specified by the system file 'ethcfg.cfg', or by settings in Anybus Configuration Manager.

Note that the switches are read once during startup; any changes require a reset in order to have effect.

See also...

- “Basic Network Configuration” on page 37
- “Basic Settings” on page 64



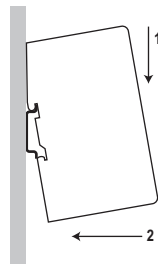
SW1	SW2	SW3	SW4	SW5	SW6	SW7	SW8	DHCP	Subnet	Gateway	IP
OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF		(settings determined by 'ethcfg.cfg')		
OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	255.255.255.0	192.168.0.255	192.168.0.1
OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	255.255.255.0	192.168.0.255	192.168.0.2
...
ON	ON	ON	ON	ON	ON	ON	OFF	OFF	255.255.255.0	192.168.0.255	192.168.0.254
ON	ON	ON	ON	ON	ON	ON	ON		(invalid setting)		

1.4 Hardware Installation

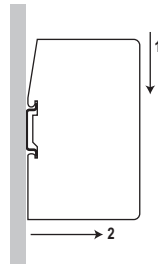
Perform the following steps to install the Anybus Communicator module:

1. Snap the gateway on to the DIN-rail.

The DIN-rail mechanism works as follows:



To snap the ABC *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 ABC *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 an Ethernet network.
3. Connect the gateway to the serial sub-network.
4. Connect the gateway to the PC via the configuration cable.
5. Connect the power cable and apply power.
6. Start the Anybus Configuration Manager program on the PC.
(The Anybus Configuration Manager software will automatically attempt to detect the serial port. If unsuccessful, select the correct port manually in the "Port"-menu)
7. Configure the gateway using the Anybus Configuration Manager and download the configuration.
8. Set up the EtherNet/IP communication according to the gateway 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.

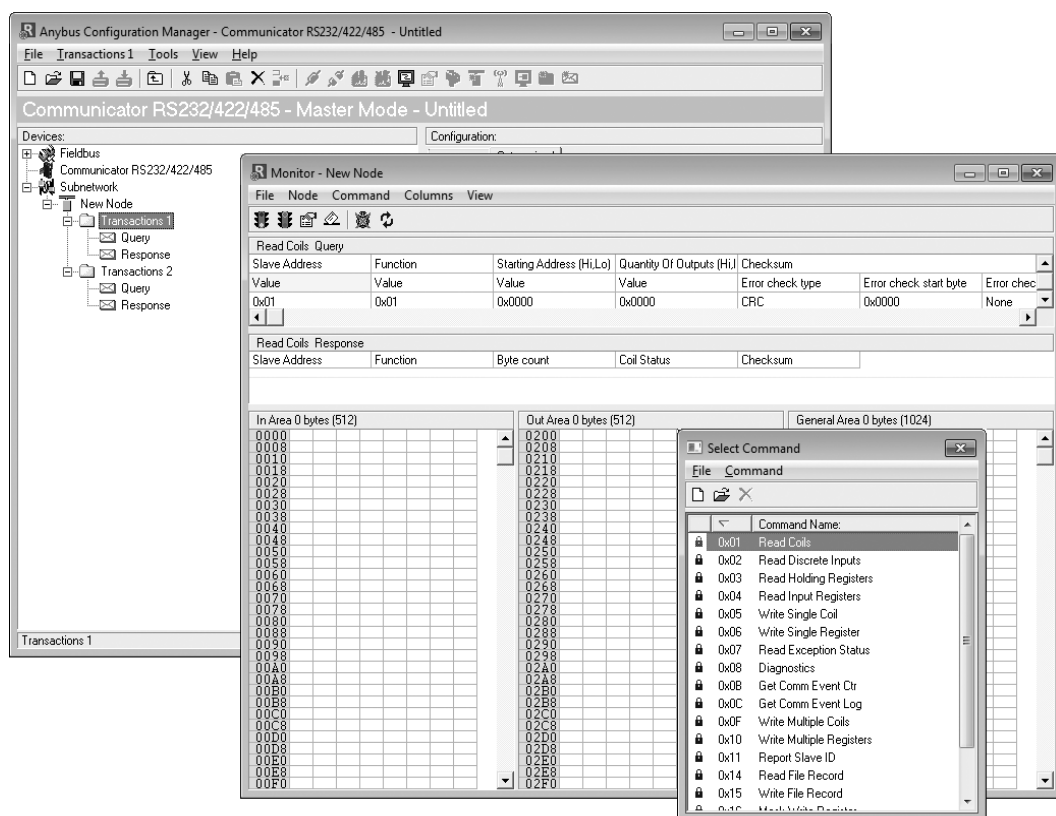
2. Basic Operation

2.1 General

The Anybus Communicator is designed to exchange data between a serial sub-network and a higher level network. Unlike most other similar devices, the Anybus Communicator has no fixed protocol for the sub-network, and consequently 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 in 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, data exchanged on the sub-network and on the higher level network all resides in the same memory.

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

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

- **Input Data (512 bytes)**

This area can be read by the higher level network, the web server and the E-mail client.

(Data representation on the higher level network is described later in this chapter).

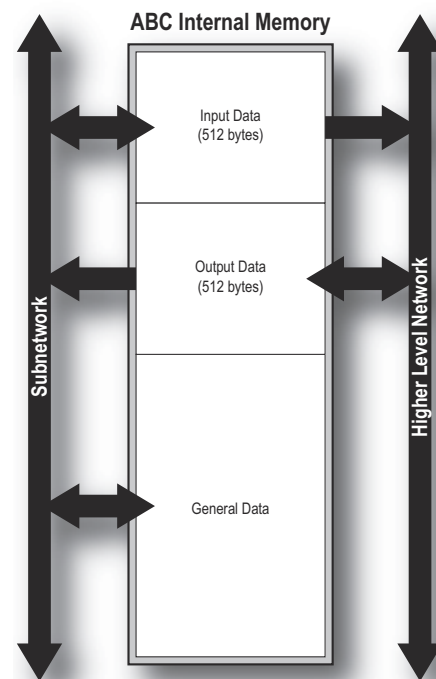
- **Output Data (512 bytes)**

This area can be read/written to by the higher level network, the web server and the E-mail client.

(Data representation on the higher level network is described later in this chapter).

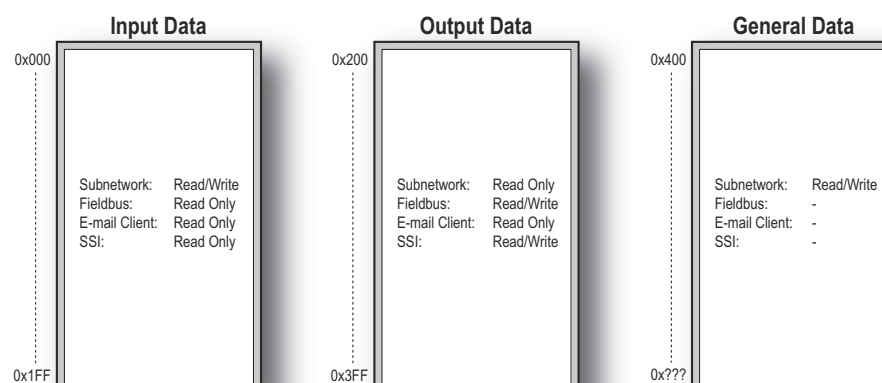
- **General Data (up to 1024 bytes)**

This area cannot be accessed from the higher level network, but can be used for transfers between individual nodes on the sub-network, 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 sub-network. The gateway can handle up to 1024 bytes of general data.



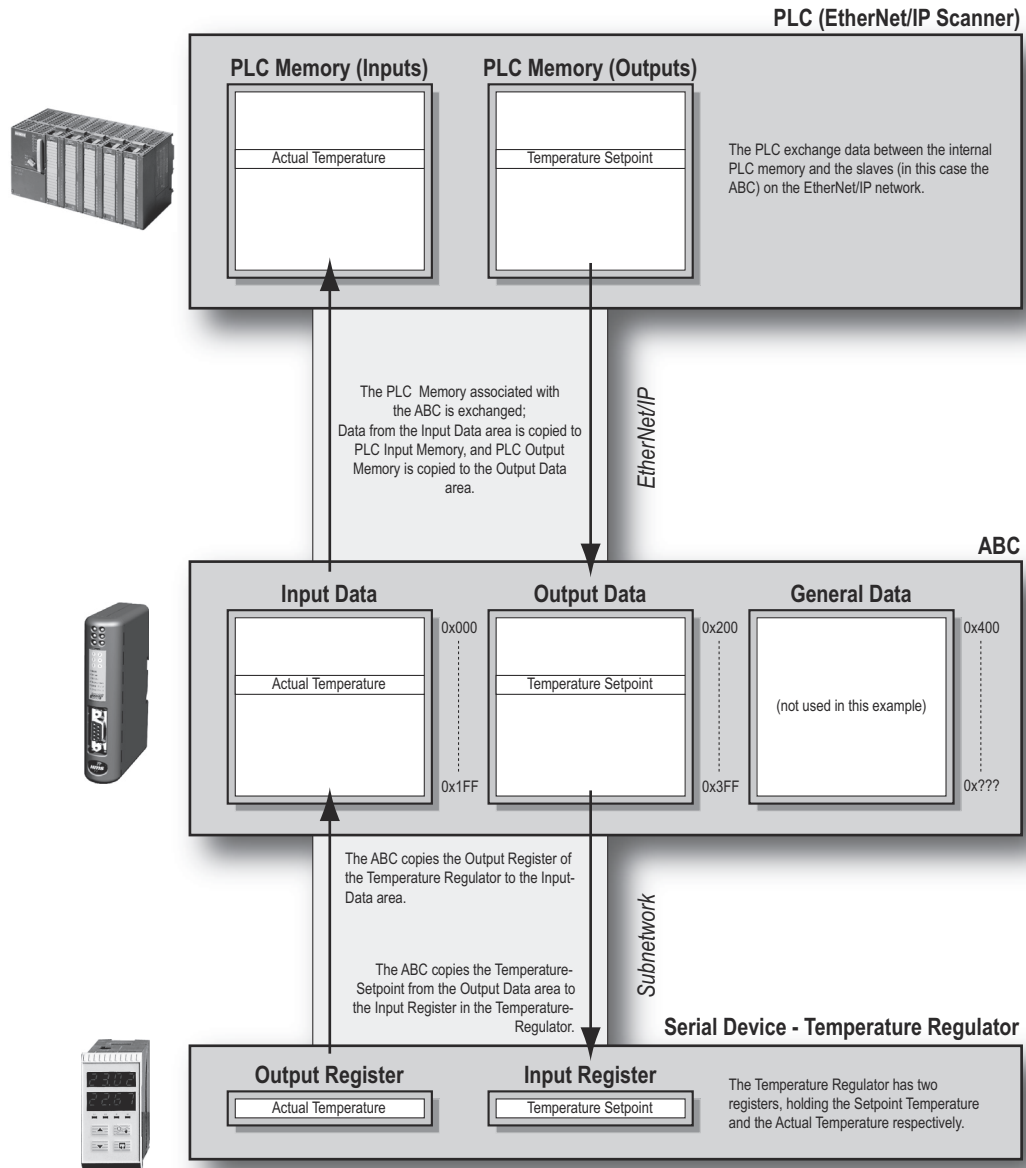
2.2.1 Memory Map

When building the sub-network 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 sub-network exchanges information with a PLC on the higher level network, via the internal memory buffers in the Anybus Communicator.



2.3 Sub-network Protocol

2.3.1 Protocol Modes

The Anybus Communicator features three distinct operating modes for sub-network communication: 'Master Mode', 'DF1 Master Mode' and 'Generic Data Mode'. Note that the protocol mode only specifies the basic communication model, not the actual sub-network protocol.

- **Master Mode**

In this mode, the gateway acts as a master on the sub-network, and the serial communication is query-response based. The nodes on the network are not permitted to issue messages unless first addressed by the gateway .

For more information about this mode, see "Master Mode" on page 21.

- **DF1 Master Mode**

In this mode, the gateway acts as a master on the sub-network, using the DF1 protocol. The serial communication is query-response based. For more information about this mode, see "DF1 Protocol Mode" on page 86.

- **Generic Data Mode**

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

For more information about this mode, see "Generic Data Mode" on page 22.

2.3.2 Protocol Building Blocks

The following building blocks are used in Anybus Configuration Manager to describe the sub-network communication. How these blocks apply to the three protocol modes is described later in this document.

- **Node**

A 'node' represents a single device on the sub-network. 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 sub-network.

- **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.

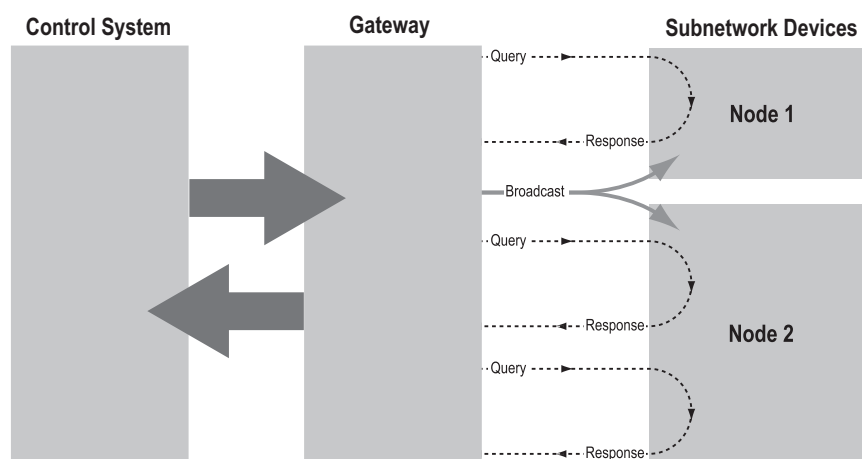
- **Frame Object**

'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 gateway issues a query on the sub-network, the addressed node is expected to issue a response. Nodes are not permitted to issue responses/messages spontaneously, i.e. without first receiving a query.

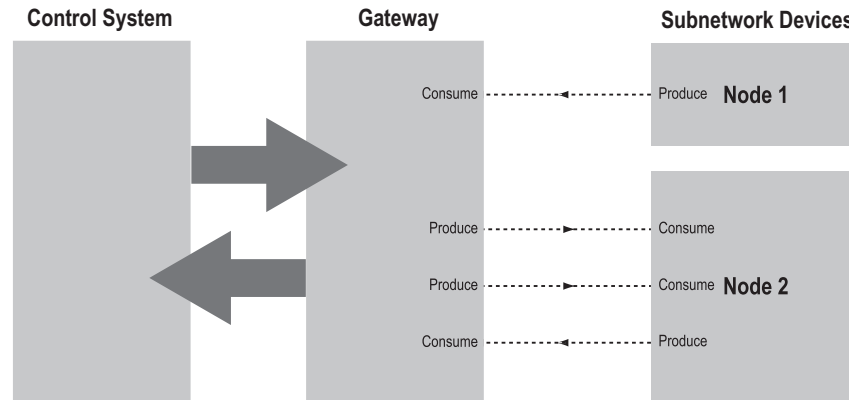
There is, however, one exception to this rule; the broadcaster. Most protocols offer some way of broadcasting messages to all nodes on the network, without expecting them to respond to the broadcasted message. This is also reflected in the gateway, which features a dedicated broadcaster node.



In Master Mode, Anybus Configuration Manager comes pre-loaded with the most commonly used Modbus RTU commands, which can be conveniently reached by right-clicking on a node in the Anybus Configuration Manager and selecting 'Insert New Command'. Note, however, that this in no way prevents other protocols based on the same query-response message-scheme from also being implemented.

2.3.4 Generic Data Mode

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



In the figure above, the Anybus Communicator ‘consumes’ data ‘produced’ by a node on the sub-network. 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 sub-network, for ‘consumption’ by a node.

2.3.5 DF1 Master Mode

Please refer to “DF1 Protocol Mode” on page 86.

2.4 EtherNet/IP

2.4.1 General

EtherNet/IP is based on the Control and Information Protocol (CIP), which is also the application layer for DeviceNet and ControlNet. The Anybus Communicator acts as a Group 2 or 3 server on the EtherNet/IP network.

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 107

2.4.2 Data Types

The input and output data hold 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 119.

See also...

- “Assembly Object, Class 04h” on page 110
- “Parameter Data Input Mapping Object, Class B0h” on page 112
- “Parameter Data Output Mapping Object, Class B1h” on page 113
- “Fieldbus Settings” on page 64

2.4.3 Memory Layout

I/O sizes are specified using the Anybus Configuration Manager and correlate to the Anybus Communicator memory as follows:

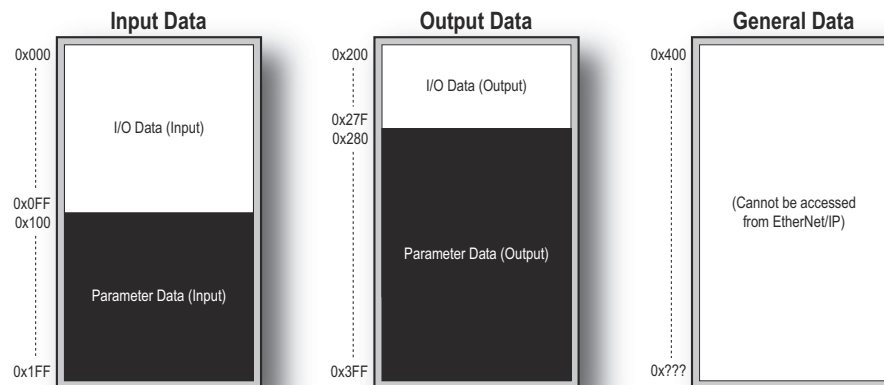
Example:

I/O Sizes for the gateway set to the following values:

IO Size In= 256 bytes (0x0100)

IO Size Out= 128 bytes (0x0080)

Resulting memory layout:



2.5 Modbus-TCP

2.5.1 General

The Modbus-TCP protocol is an implementation of the standard Modbus protocol running on top of TCP/IP. The built-in Modbus-TCP server provides access to the input and output data areas via a subset of the functions defined in the Modbus-TCP specification.

The server supports up to 8 simultaneous connections and communicates over TCP port 502. For detailed information regarding the Modbus-TCP protocol, consult the Open Modbus Specification.

2.5.2 Addressing Modes

The Anybus Communicator features two different modes of operation regarding the Modbus communication:

- **Modbus Addressing Mode (Default)**

In this mode, the input and output data areas are mapped to different function codes.

Note that coil addressing is not possible in this mode.

See also...

- “Modbus Addressing Mode” on page 26

- **Anybus Addressing Mode**

Compared to Modbus Addressing Mode, this mode allows data to be addressed in a more flexible way. Note however that several function codes can be used to access the same data in the gateway. While this may appear confusing at first, it allows data to be manipulated in ways not possible in Modbus Addressing Mode (e.g. it is possible to manipulate individual bits of a register by accessing coils associated with the same memory location).

See also...

- “Anybus Addressing Mode” on page 27

2.5.3 Supported Exception Codes

Code	Name	Description
0x01	Illegal function	The function code in the query is not supported
0x02	Illegal data address	The data address received in the query is outside the initialized memory area
0x03	Illegal data value	The data in the request is illegal