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 CANopen

Doc. Id. HMSI-27-312
Rev. 3.11

Connecting Devices™

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

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

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

Important User Information

This document 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. 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 CANopen User Manual

Doc: HMSI-27-312

Table of Contents

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 - CANopen Installation Sheet	HMS
Anybus-S CANopen Fieldbus Appendix	HMS
CiA Draft Standard 301 v4.02	CAN in Automation
DF1 Protocol and Command Set - Reference Manual, 1770-6.5.16, October 1996	Allen-Bradley

P.2 Document History

Summary of Recent Changes (3.01... 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	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... 3.11)

Revision	Change	Page(s)
3.11	Added compliance/conformance info	77

Revision List

Revision	Date	Author	Chapter	Description
2.00	2003-06-18	PeP	All	Second major release
2.10	2005-08-25	PeP	All	Major update
2.50	2006-03-28	PeP	All	Major rewrite
2.51	2006-12-22	PeP	-	Minor corrections & updates
2.52	2008-02-08	PeP	1	Minor update
2.53	2009-04-23	KeL	All	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	March 2015	ThN	All	Misc corrections and updates, new Doc ID.
3.11	March 2015	ThN	B	Minor corrections

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 'gateway' 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

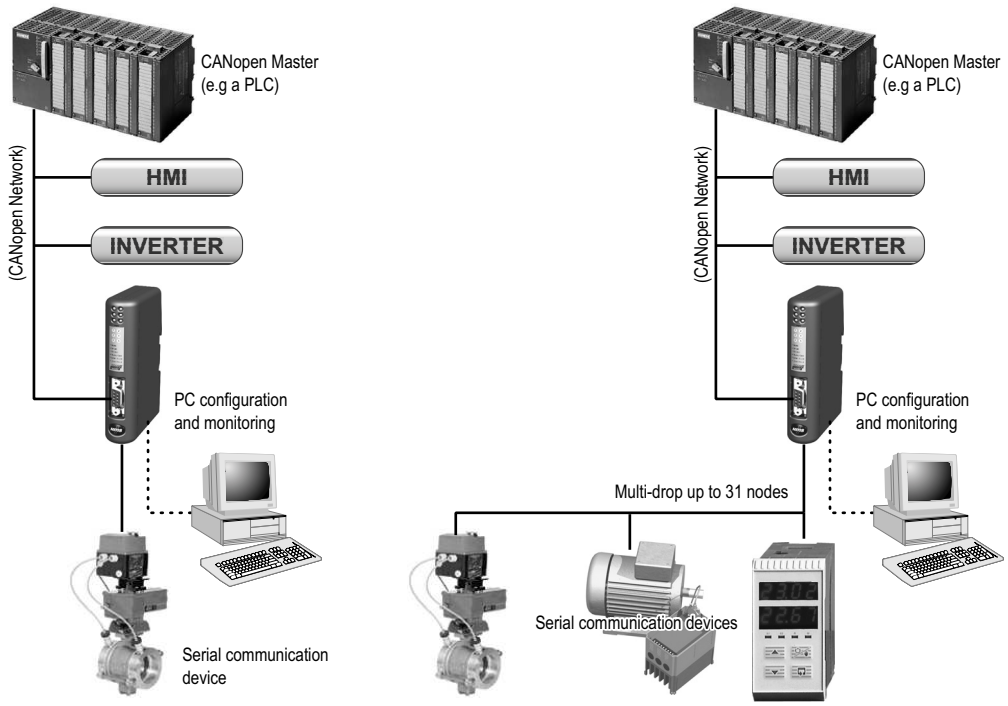
Term	Meaning
ABC	Anybus Communicator™
Gateway	
ACM	Anybus Configuration Manager
Broadcaster	A protocol-specific node in the configuration that handles transactions destined to all nodes.
COP	CANopen
Command	A predefined transaction.
Configuration	List of configured nodes with transactions on the subnetwork.
Fieldbus	The higher level network to which the Anybus 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
Subnetwork	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, CANopen
Network	
Fieldbus	

P.4 Support

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

1. About the Anybus Communicator for CANopen

The Anybus Communicator for CANopen acts as a gateway between virtually any serial application protocol and a CANopen-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.



Single-Node Serial Sub Network **Multi-Node Serial Sub Network**

Subnetwork

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

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

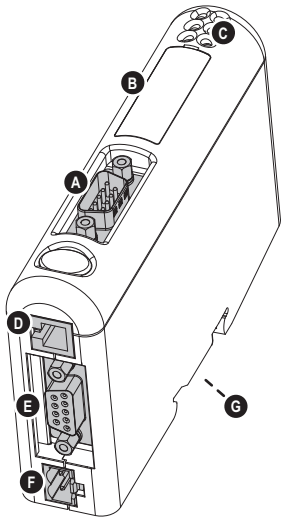
CANopen Interface

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

- Galvanically isolated bus electronics
- Supports all standard baudrates
- Customizable PDO mapping
- Up to 512+512 bytes of slow I/O
- Up to 80 RPDO's and 80 TPDO's
- Transmission types: PDO Sync, Change-of-state, Event based and RTR
- Network store/restore functionality
- Heartbeat & Node Guarding support

1.1 External View

For wiring and pin assignments, see "Connector Pin Assignments" on page 78.



A: CANopen Connector

This connector is used to connect the gateway to the field-bus.

See also...

- "CANopen Connector" on page 78

B: Configuration Switches

See also...

- "Configuration Switches" on page 12

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...

- "Hardware Installation" on page 13
- "PC Connector" on page 79

E: Subnetwork Connector

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

See also...

- "Hardware Installation" on page 13
- "Subnetwork Interface" on page 80

F: Power Connector

This connector is used to apply power to the gateway.

See also...

- "Hardware Installation" on page 13
- "Power Connector" on page 78
- "Technical Specification" on page 128

G: DIN-rail Connector

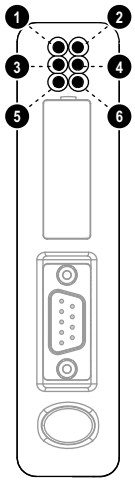
The DIN-rail mechanism connects the gateway to PE (Protective Earth).

See also...

- "Hardware Installation" on page 13

1.2 Status LEDs

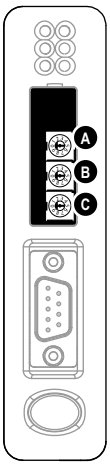
#	State	Status
1 - Run	Off	-
	Single Flash	Device in STOPPED state
	Blinking	Device in PRE-OPERATIONAL state
	On	Device in OPERATIONAL state
2 - Error	Off	No error
	Single Flash	Warning limit reached
	Double Flash	Error Control Event
	Triple Flash	Sync Error
	On	Bus off
3 - Status	Off	Normal operation
	Red	Unrecoverable fault detected
4 - Power	Off	Device not powered
	On	Device powered
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 61.

1.3 Configuration Switches

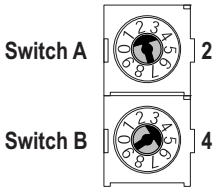


The on-board switches are used to set the CANopen node address and operating baudrate. Normally, these switches are covered by a plastic hatch. When removing the hatch, avoid touching the circuit boards and components. If tools are used when opening the hatch, be cautious. Note that these settings cannot be changed during runtime, i.e. the gateway must be restarted in order for any changes to have effect.

The node address is configured using two rotary switches as follows:

Node Address = (Switch B x 10) + (Switch A x 1)

Example: To set node address 42, set switch A to '2' and switch B to '4'.



The baudrate is set via switch C, and is interpreted as follows:

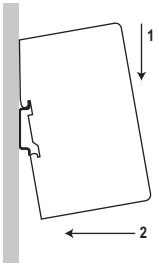
#	Baudrate
0	(reserved)
1	10 kbit/s
2	20 kbit/s
3	50 kbit/s
4	125 kbit/s

#	Baudrate
5	250 kbit/s
6	500 kbit/s
7	800 kbit/s
8	1 Mbit/s
9	(reserved)

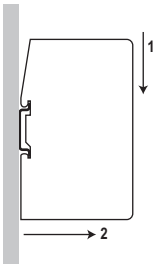
1.4 Hardware Installation

Perform the following steps when physically installing the gateway:

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 CANopen network
3. Set the CANopen node ID and operating baudrate (see "Configuration Switches" on page 12)
4. Connect the gateway to the serial subnetwork
5. Connect the gateway to the PC via the Configuration Cable
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 gateway using the Anybus Configuration Manager and download the configuration
9. Set up the CANopen communication in accordance with 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.

1.5.2 Electronic Data Sheet

Each device on CANopen is associated with an Electronic Data Sheet (EDS-file), which holds a description of the device and its functions. Most importantly, the file describes the object dictionary implementation in the device.

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

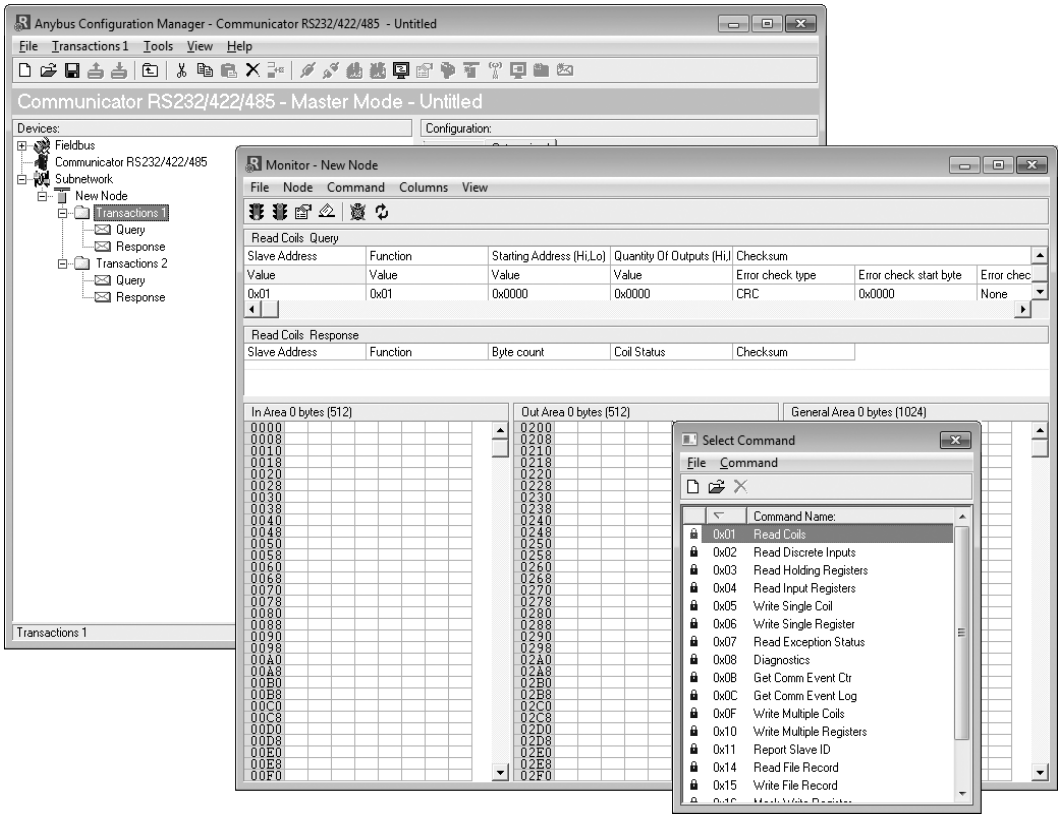
2. Basic Operation

2.1 General

The Anybus Communicator is designed to exchange data between a serial subnetwork and a higher level network. Unlike most other gateway devices of similar kind, it does not have a fixed protocol for the subnetwork, 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 subnetwork 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.

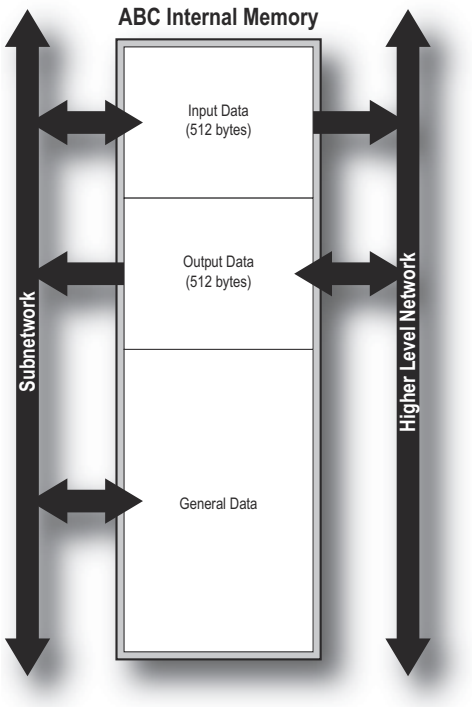


2.2 Data Exchange Model

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.

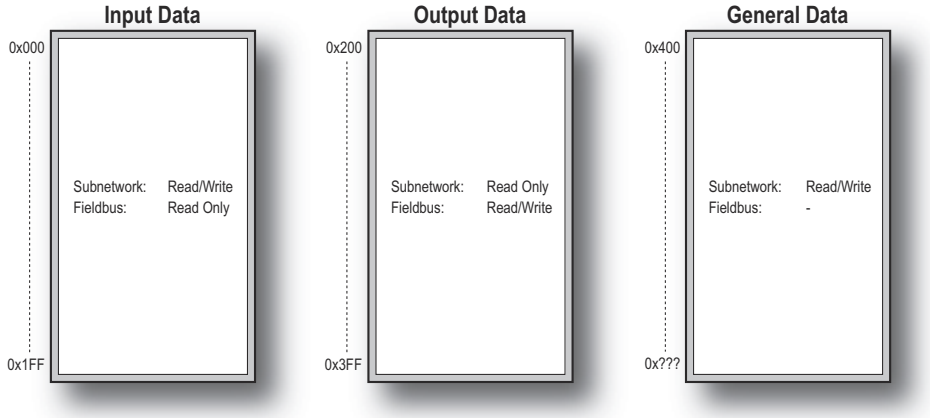
  (how this data is represented on the higher level network will be described later in this chapter).

- **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.

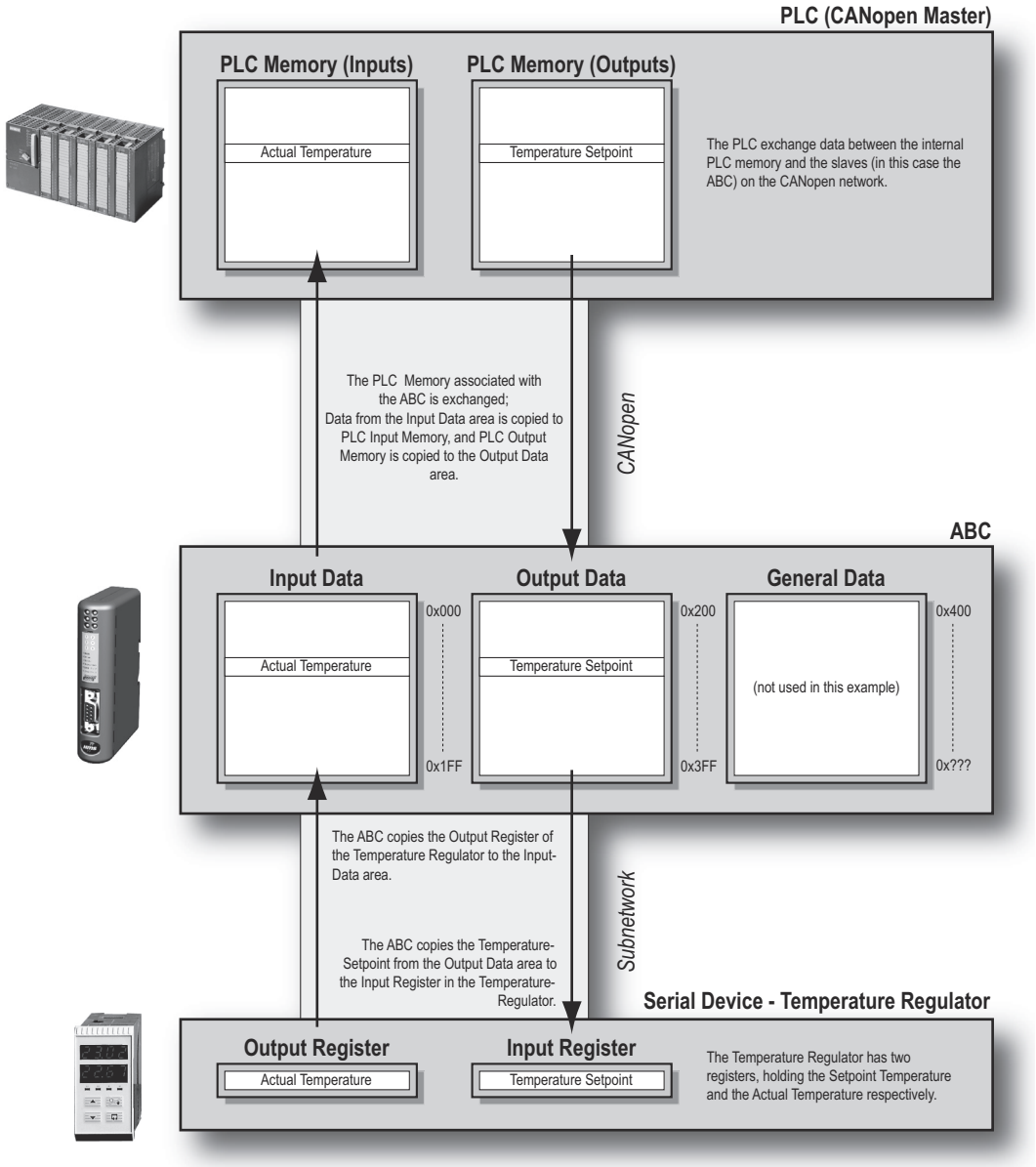
2.2.1 Memory Map

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 gateway.



2.3 Subnetwork Protocol

2.3.1 Protocol Modes

The gateway 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 19.

- **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 86.

- **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 20.

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 three 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.

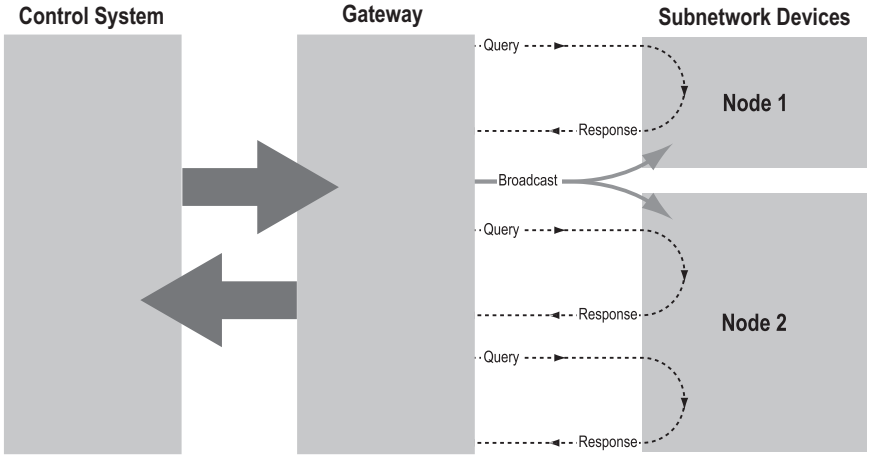
- **Frame Object**

  '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 gateway issues a query on the subnetwork, the addressed node is expected to issue a response to that query. Nodes are not permitted to issue responses spontaneously, i.e. without first receiving a query.

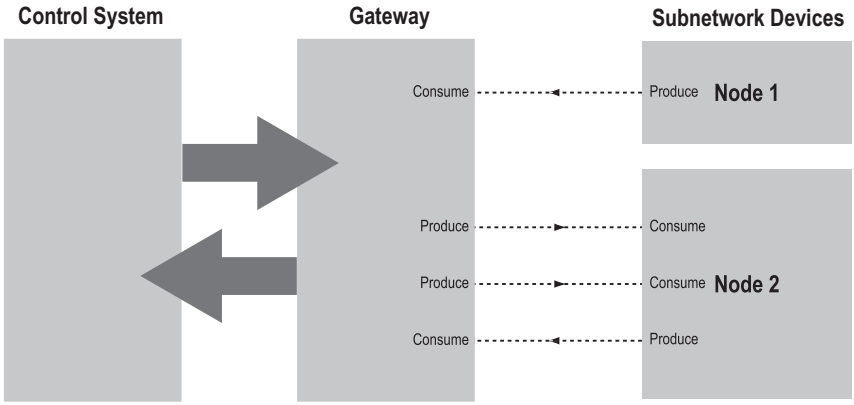
There is one exception to this rule; the broadcaster. Most protocols offer some way of broadcasting messages to all nodes on the network, without expecting them to respond to the broadcasted message. This is also reflected in the gateway, which features a dedicated broadcaster node.



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 86.

2.4 Data Representation on CANopen

2.4.1 General

The Anybus Communicator acts as a slave on the CANopen network. As such, it does not initiate communication towards other nodes by itself, but can be read from/written to by a CANopen master.

2.4.2 Data Representation

On CANopen, the input and output data areas are represented as object entries in the manufacturer specific range (2000h...5FFFh). Separate object ranges are used for byte, word, and double-word access.

A portion of these objects may be exchanged as Process Data Objects (PDO). The gateway supports up to 80 RPDOs and 80 TPDOs, each capable of carrying up to 8 bytes of data.

Which objects that can be exchanged this way depend on two factors:

- **PDO Mapping**

  The objects that shall be exchanged as PDOs on the bus are specified (i.e. mapped) by the network configuration tool.

  If no mapping has been specified, the gateway defaults to the following mapping scheme:

RPDO no.	Default COB IDs		Mapped to...	Default State
	Node ID 1... 63	Node ID >= 64		
1	200h + Node ID	200h + Node ID	Output Data buffer, bytes 0... 7	Enabled
2	300h + Node ID	300h + Node ID	Output Data buffer, bytes 8...15	
3	400h + Node ID	400h + Node ID	Output Data buffer, bytes 16... 23	
4	500h + Node ID	500h + Node ID	Output Data buffer, bytes 24... 31	
5	240h + Node ID	500h	Output Data buffer, bytes 32... 39	Disabled
6	340h + Node ID	500h	Output Data buffer, bytes 40... 47	
7	440h + Node ID	500h	Output Data buffer, bytes 48... 55	
8	540h + Node ID	500h	Output Data buffer, bytes 56... 63	
9...80	500h	500h	-	Disabled

TPDO no.	Default COB IDs		Mapped to...	Default State
	Node ID 1... 63	Node ID >= 64		
1	180h + Node ID	180h + Node ID	Input Data buffer, bytes 0... 7	Eabled
2	280h + Node ID	280h + Node ID	Input Data buffer, bytes 8... 15	
3	380h + Node ID	380h + Node ID	Input Data buffer, bytes 16... 23	
4	480h + Node ID	480h + Node ID	Input Data buffer, bytes 24... 31	
5	1C0h + Node ID	500h	Input Data buffer, bytes 32... 39	Disabled
6	2C0h + Node ID	500h	Input Data buffer, bytes 40... 47	
7	3C0h + Node ID	500h	Input Data buffer, bytes 48... 55	
8	4C0h + Node ID	500h	Input Data buffer, bytes 56... 63	
9...80	500h	500h	-	Disabled

- **I/O Data Size**

  Object entries which do not correspond directly to I/O Data (see "Memory Layout (Internal Memory Buffer)" on page 22) may not be mapped as PDOs. Any attempt to map an object entry which does not fit this criteria will result in an error.

For more information about the object dictionary implementation, see "Object Dictionary Implementation" on page 71.

2.4.3 Memory Layout (Internal Memory Buffer)

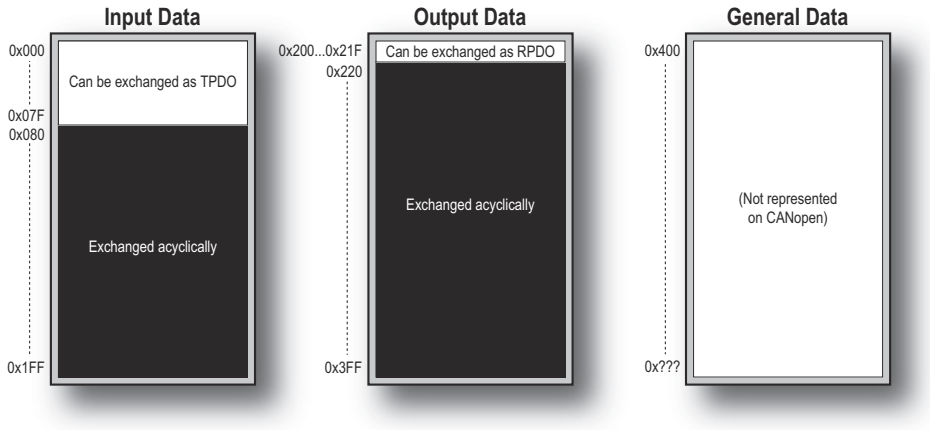
The I/O sizes specified in Anybus Configuration Manager correlate to gateway memory as follows:

Example:

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

IO Size In= 128 bytes(0x0080)
IO Size Out= 32 bytes(0x0020)

Resulting memory layout:



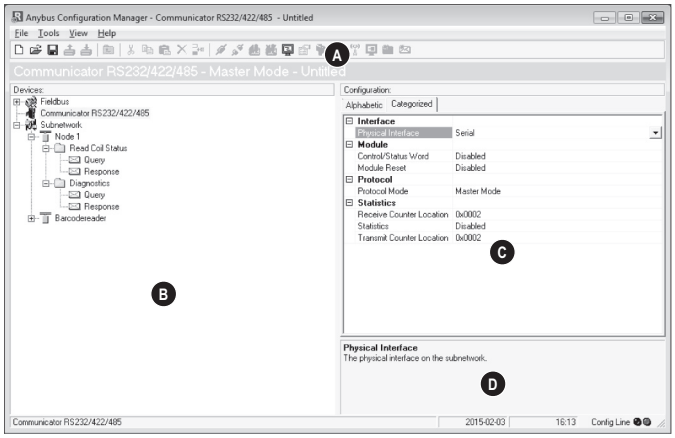
See also...

- "Data Representation" on page 21
- "Object Dictionary Implementation" on page 71
- "Fieldbus Settings" on page 28

3. Navigating ACM

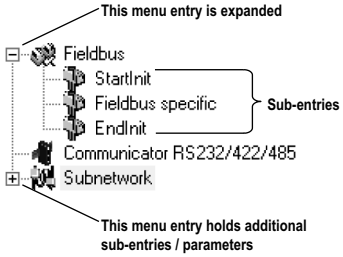
3.1 Main Window

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.

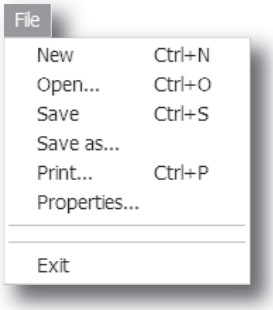
  The parameter value may be specified either using a selection box or manually, depending on the parameter itself. Values can be specified in decimal form (e.g. "42"), or in hexadecimal format (e.g. "0x2A").

- **D: Information Section**

  This section holds information related to the currently selected parameter.

3.1.1 Drop-down Menus

File



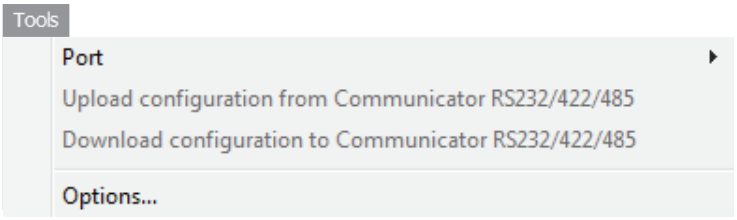
- **New**
  Create a new configuration.
  See also "Configuration Wizards" on page 64.
- **Open...**
  Open a previously created configuration.
- **Save**
  Save the current configuration.
- **Save As...**
  Save the current configuration under a new name.
- **Print...**
  Send details about the current 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.

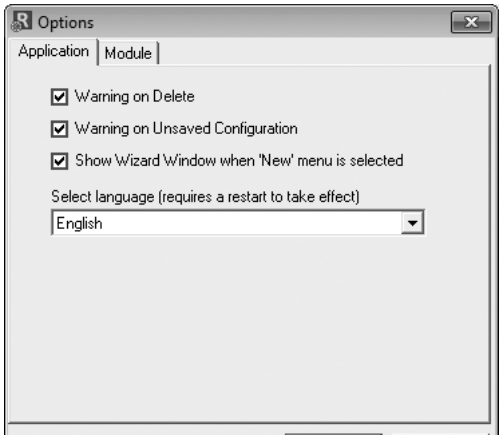
- **Start Logging**

  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.