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 EtherCAT®

Doc: HMSI-168-95
Rev: 3.20

HMS Connecting Devices™

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.

.

EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

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

Table of Contents

Chapter 15 Control and Status Registers

Chapter 16 Advanced Fieldbus Configuration

Appendix A Connector Pin Assignments

Appendix B Technical Specification

Appendix C Troubleshooting

Appendix D 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 - EtherCAT 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 (3.11... 3.20)

Revision	Change	Page(s)
3.20	Updated LED description	11

Revision List

Revision	Date	Author	Chapter	Description
2.00	2007-05-07	PeP	All	1st official release
2.01	2007-08-13	PeP	-	Minor update
2.03	2009-04-24	KeL	All	Misc. minor corrections and updates
3.00	2011-02-09	KaD	All	Misc. minor updates, new template and DF1 functionality
3.01	2011-09-30	KaD	All	Misc corrections and updates, new Anybus Configuration Manager name
3.02	2012-11-20	KeL	I	Added information on EtherCAT trademark
3.10	2015-02-11	ThN	All	Misc. corrections and updates
3.11	2015-03-20	ThN	B	Added compliance/conformance info
3.20	2016-09-07	ThN	1	Updated LED description

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.4 Glossary

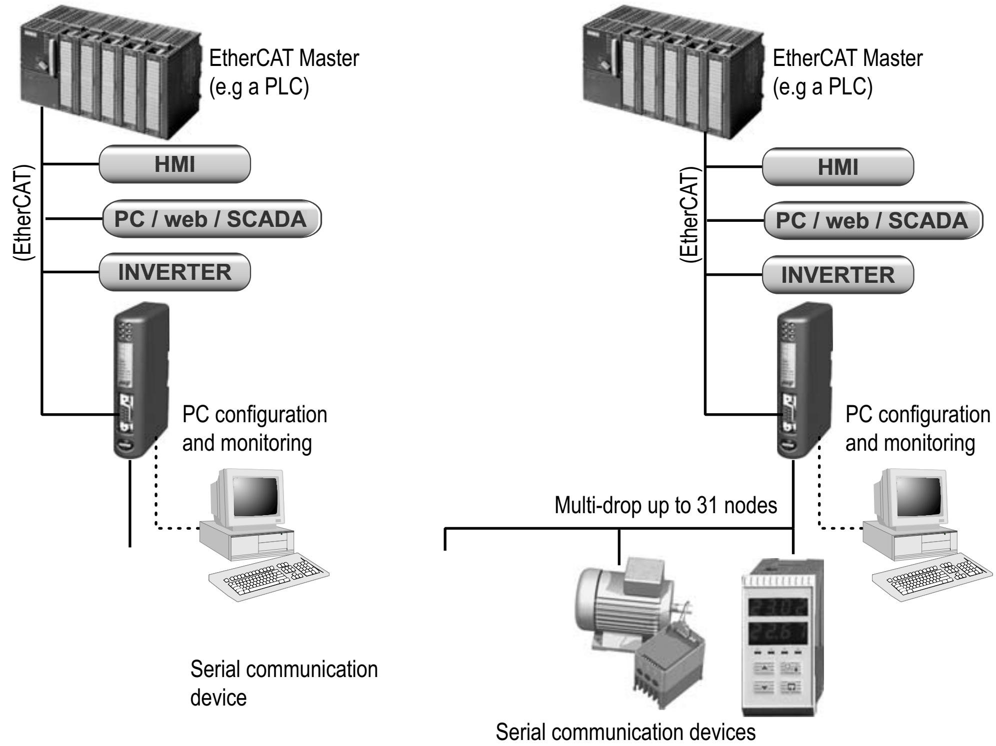
Term	Meaning
ABC	Anybus Communicator
ACM	Anybus Configuration Manager
Broadcaster	A protocol-specific node in the configuration that handles transactions destined to all nodes.
ECT	EtherCAT
Command	A predefined transaction.
Configuration	List of configured nodes with transactions on the subnetwork.
Fieldbus	The higher level network to which the communicator is connected.
Control System	Device which controls the communication on the higher level network, i.e. a fieldbus master
Frame Object	Low level entities which are used to describe the different parts of a transaction.
Monitor	A tool for debugging the gateway 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 this product 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, EtherCAT
Network	
Fieldbus	

P.5 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 EtherCAT

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

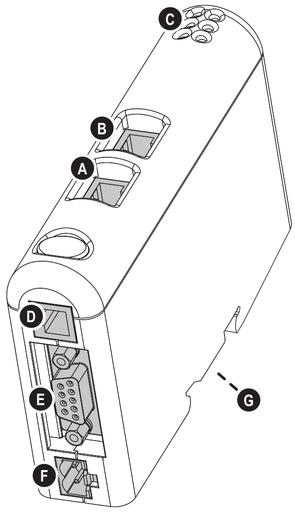
EtherCAT Interface

EtherCAT 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.

- CANopen over EtherCAT
- DS301 v4.02 compliant
- Galvanically isolated network electronics
- Supports segmented SDO access
- Exchanges up to 512 bytes of data in each direction

1.1 External View

For wiring and pin assignments, see "Connector Pin Assignments" on page 74.



A: EtherCAT Port 1 (In)

See also...

- "EtherCAT Connectors" on page 74

B: EtherCAT Port 2 (Out)

See also...

- "EtherCAT Connectors" on page 74

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 75

E: Subnetwork Connector

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

See also...

- "Subnetwork Interface" on page 76

F: Power Connector

This connector is used to apply power to the gateway.

See also...

- "Power Connector" on page 74
- "Troubleshooting" on page 131

G: DIN-rail Connector

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

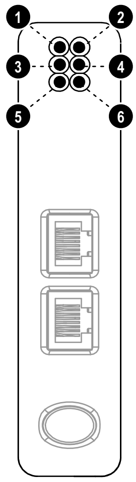
See also...

- "Hardware Installation" on page 12

1.2 Status LEDs

EtherCAT Interface Indications

#	State	Status
1 - RUN	Off	Device in INIT state
	Green	Device in OPERATIONAL state
	Green, blinking	Device in PREOPERATIONAL state
	Green, single flash	Device in SAFE-OPERATIONAL state
2 - ERR	Off	Normal operation (no errors)
	Red	Application watchdog timeout
	Red, blinking	General configuration error
	Red, single flash	Slave device application has changed the EtherCAT status autonomously: Parameter “Change” in the AL status register is set to 01 (change/error)
	Red, double flash	Sync manager watchdog timeout
3 - Link/Activity 1	Off	No link sensed on EtherCAT port 1
	Green	Link sensed on EtherCAT port 1
	Green, flickering	Activity on EtherCAT port 1
4 - Link/Activity 2	Off	No link sensed on EtherCAT port 2
	Green	Link sensed on EtherCAT port 2
	Green, flickering	Activity on EtherCAT port 2



General Gateway/Subnet Indications

#	State	Status
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

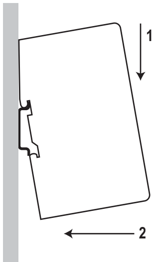
a. This LED turns green when all transactions have been active at least once. This includes any transactions using “change of state” or “change of state on trigger”. If a timeout occurs on a transaction, this led will turn red.

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.

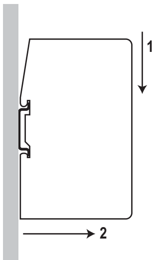
1.3 Hardware Installation

Perform the following steps when physically installing the gateway:

1. Snap the gateway on to the DIN-rail (See "DIN-rail Connector" 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 EtherCAT network
3. Connect the gateway to the serial subnetwork
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 on the PC
   (The Anybus Configuration Manager software attempts to detect the serial port automatically. If not successful, select the correct port manually in the "Port"-menu)
7. Configure the gateway using the Anybus Configuration Manager and download the configuration

1.4 Software Installation

1.4.1 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.

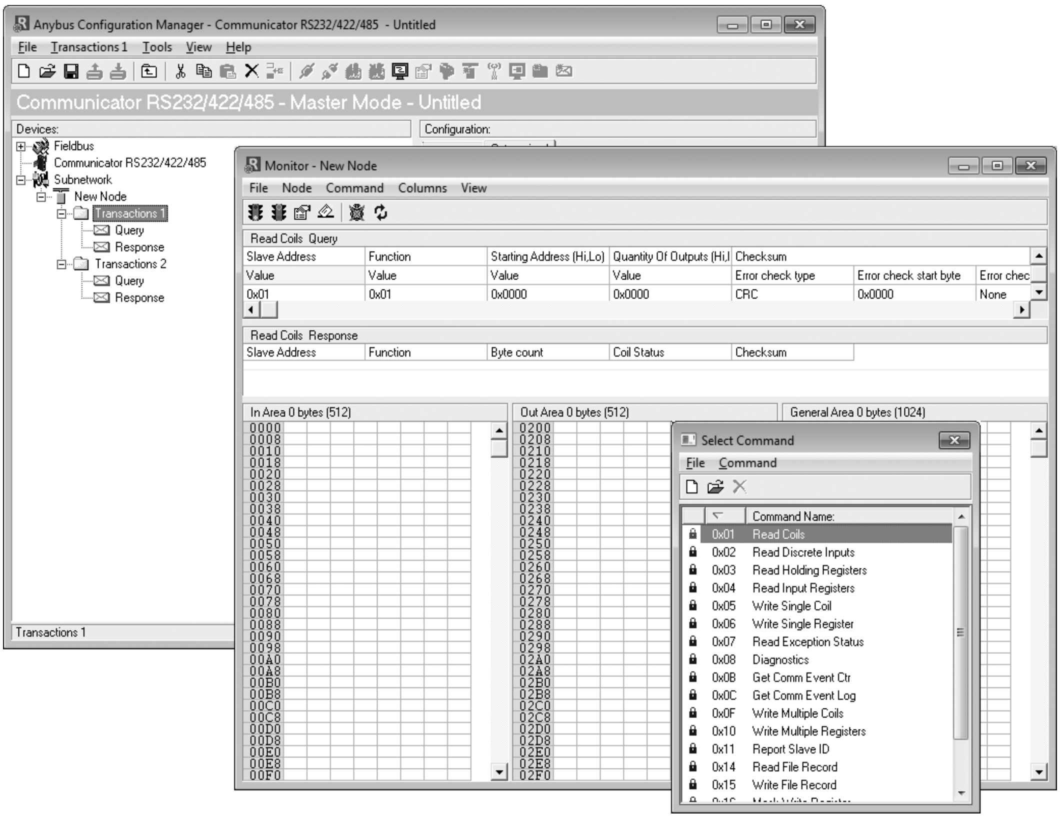
2. Basic Operation

2.1 General

The Anybus Communicator 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.



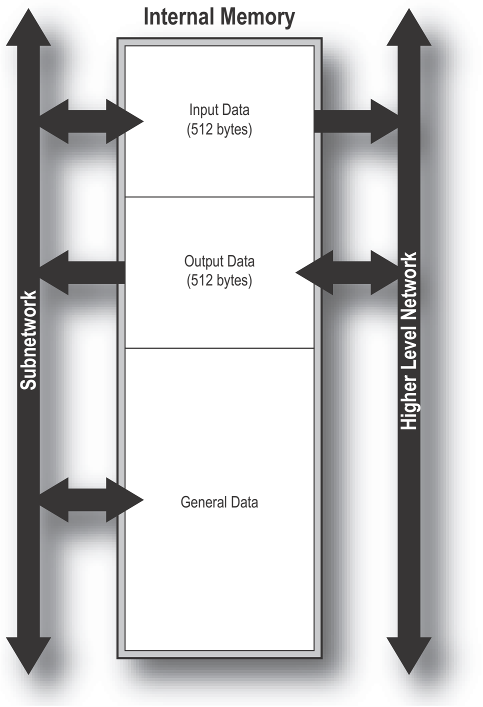
2.2 Data Exchange Model

Internally, the data exchanged on the subnetwork, and the data exchanged on the higher level network (i.e. EtherCAT), 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.

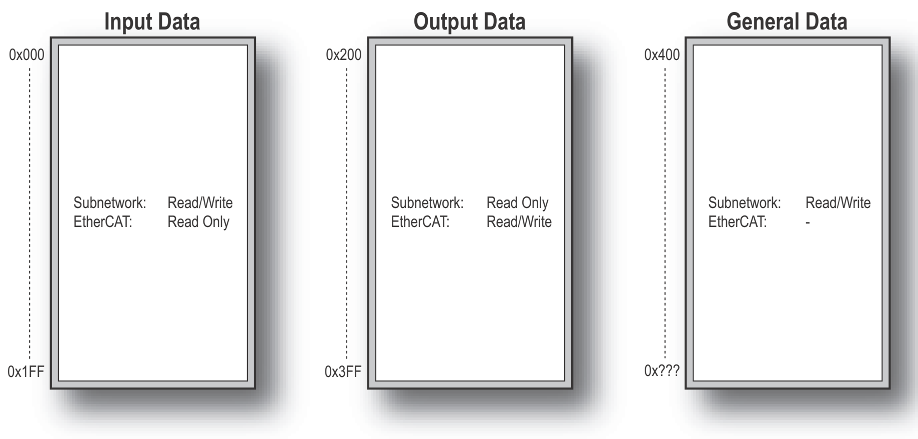
The internal memory buffer is divided into three areas based on their function:

- **Input Data (Up to 512 bytes)**
  This area can be read from the higher level network (i.e. by the EtherCAT master).
- **Output Data (Up to 512 bytes)**
  This area can be written to by the higher level network (i.e. by the EtherCAT master)
- **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 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.



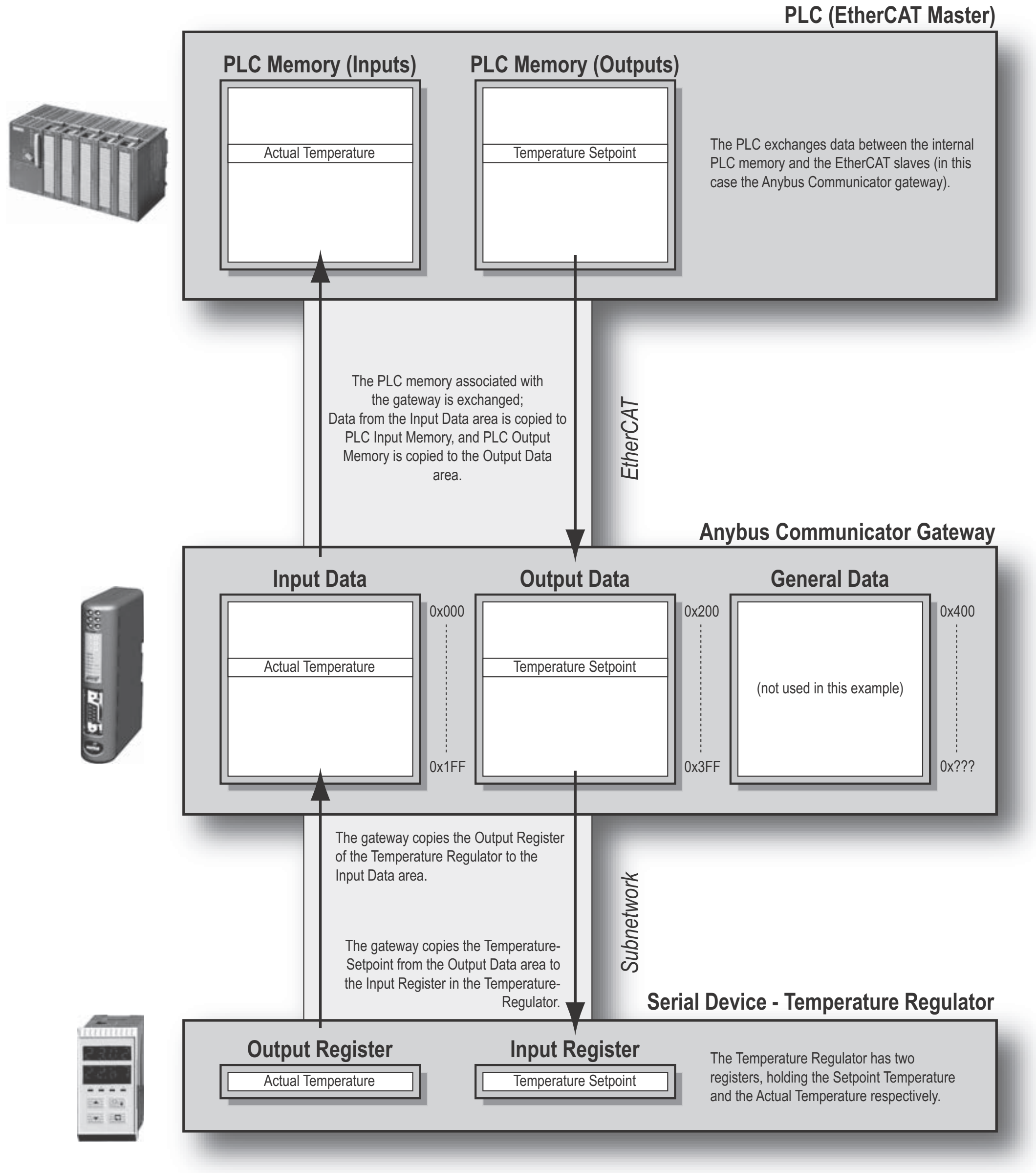
2.2.1 Memory Map

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.

  See also "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.

  See also "DF1 Protocol Mode" on page 52.

- **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.

  See also "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.

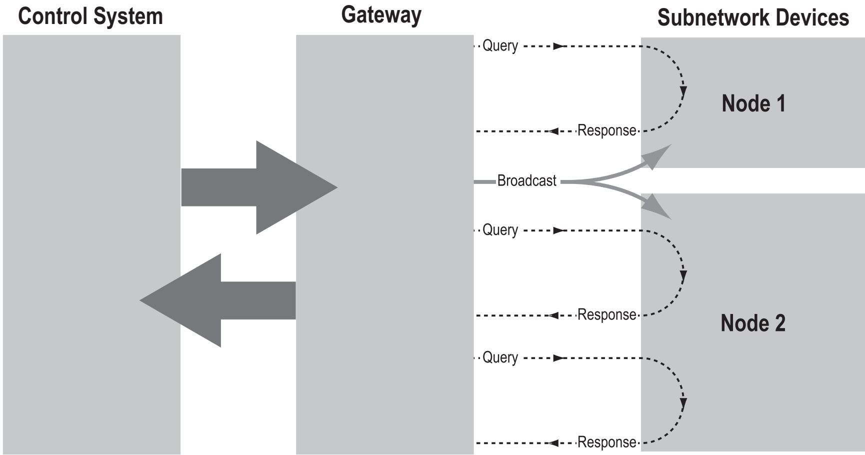
- **Frame Object**

  "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 gateway 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.

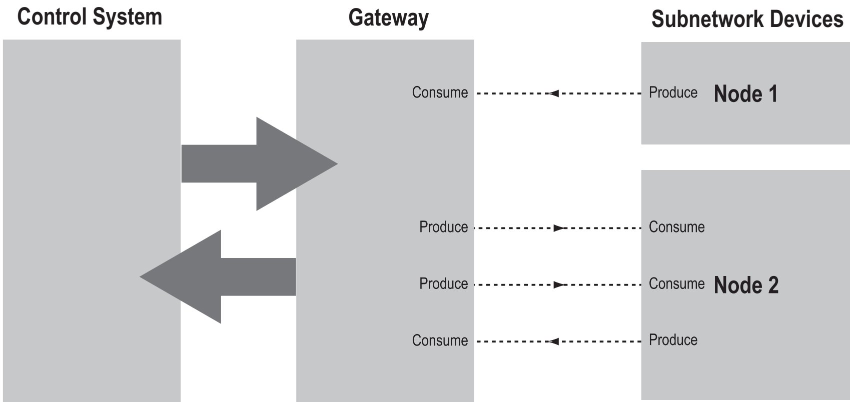
There is one exception to this rule; the broadcaster. Most protocols offer some way of broadcasting messages to all nodes on the network, without expecting them to respond to the broadcasted message. This is also reflected in the gateway, which features a dedicated broadcaster node.



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 itself, 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 52.

2.4 EtherCAT Communication Properties

2.4.1 General

The EtherCAT interface acts as a full EtherCAT slave device with the following properties:

General Properties

- Full EtherCAT slave device
- 4 FMMU channels (can be used freely by the EtherCAT master for any purpose.)
- 4 SM channels
- 4kByte RAM
- Bit-oriented FMMU operation
- Position addressing, node addressing and logical addressing
- PDI Watchdog
- Output I/O Sync Manager Watchdog

Supported Features

- CANopen over EtherCAT
- DS301 v4.02 compliant
- Supports segmented SDO access
- Up to 512 bytes of cyclic data in each direction

2.4.2 Sync Managers

The gateway features four Sync Managers:

- **Sync Manager 0**
  Used for mailbox write transfers (Master to Slave).
  The gateway supports mailbox sizes of 50... 256 bytes (default = 192).

- **Sync Manager 1**
  Used for mailbox read transfers (Slave to Master).
  The gateway supports mailbox sizes of 50... 256 bytes (default = 192).

- **Sync Manager 2**
  Contains the RxPDOs specified by the PDO assignment.
  In practice, Sync Manager 2 holds the unmodified Output I/O data.

- **Sync Manager 3**
  Contains the TxPDOs specified by the PDO assignment.
  In practice, Sync Manager 3 holds the unmodified Input I/O data.

2.4.3 Data Representation

As mentioned previously, the gateway implements CANopen over EtherCAT. The object implementation is based on the DS301 communication profile.

Input and output data is mapped to dedicated objects in the manufacturer-specific range. Data declared as I/O data is exchanged by means of Process Data Objects (PDOs), while the remainder is exchanged through Service Data Objects (SDOs).

The PDO mapping is static and looks as follows:

PDO	Corresponding Object	Internal Memory
TxPDO 1A00h	Index 2000h, sub-index 1...128	Input Data, bytes 0... 127
TxPDO 1A01h	Index 2001h, sub-index 1...128	Input Data, bytes 128... 255
TxPDO 1A02h	Index 2002h, sub-index 1...128	Input Data, bytes 256... 383
TxPDO 1A03h	Index 2003h, sub-index 1...128	Input Data, bytes 384... 511
RxPDO 1600h	Index 2100h, sub-index 1...128	Output Data, bytes 0... 127
RxPDO 1601h	Index 2101h, sub-index 1...128	Output Data, bytes 128... 255
RxPDO 1602h	Index 2102h, sub-index 1...128	Output Data, bytes 256... 383
RxPDO 1603h	Index 2103h, sub-index 1...128	Output Data, bytes 384... 511

Note: The gateway will only map as many PDOs as required to hold the specified amount of I/O data.

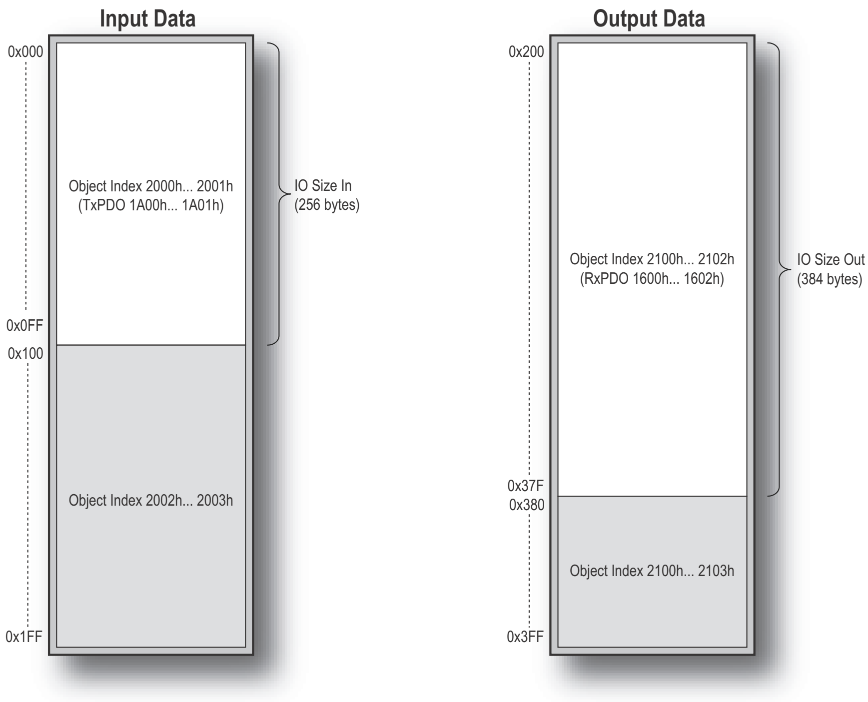
Example:

The following example illustrates how the internal memory buffers are represented on the EtherCAT network.

The I/O Sizes are set as follows:

IO Size In= 256 bytes (0x0100)
IO Size Out= 384 bytes (0x0180)

Resulting memory layout:



3. CANopen Object Dictionary Implementation

3.1 Standard Objects

3.1.1 General

The standard object dictionary is implemented in accordance with the DS301 specification (v4.02).

3.1.2 Object Entries

Index	Object Name	Sub-Index	Description	Type	Access	Notes
1000h	Device Type	00h	Device Type	U32	RO	0000 0000h (No profile)
1008h	Manufacturer device name	00h	Manufacturer device name	Visible string	RO	-
1018h	Identity object	00h	Number of entries	U8	RO	04h
		01h	Vendor ID	U32	RO	-
		02h	Product Code	U32	RO	
		03h	Revision Number	U32	RO	
		04h	Serial Number	U32	RO	
1600h ... 1603h	Receive PDO mapping	00h	Number of entries	U8	R	See 2-21 “Data Representation”
		01h... n	Mapped object #n	U32	R	
1A00h ... 1A03h	Transmit PDO mapping	00h	Number of entries	U8	R	See 2-21 “Data Representation”
		01h... Nn	Mapped object #n	U32	R	
1C00h	Sync Manager Communica-tion Type	00h	Number of entries	U8	R	04h
		01h	Mailbox wr	U8	R	01h
		02h	Mailbox rd	U8	R	02h
		03h	Process Data out	U8	R	03h
		04h	Process Data in	U8	R	04h
1C12h	Sync Manager Rx PDO Assign	00h	Number of entries [a]	U8	R	No. of assigned RxPDOs (0... 4)
		01h	Assigned RxPDO	U8	R	Assigned to RxPDO 1600h
		02h	Assigned RxPDO	U8	R	Assigned to RxPDO 1601h
		03h	Assigned RxPDO	U8	R	Assigned to RxPDO 1602h
		04h	Assigned RxPDO	U8	R	Assigned to RxPDO 1603h
1C13h	Sync Manager Tx PDO Assign	00h	Number of entries[a]	U8	R	No. of assigned TxPDOs (0... 4)
		01h	Assigned TxPDO	U8	R	Assigned to TxPDO 1A00h
		02h	Assigned TxPDO	U8	R	Assigned to TxPDO 1A01h
		03h	Assigned TxPDO	U8	R	Assigned to TxPDO 1A02h
		04h	Assigned TxPDO	U8	R	Assigned to TxPDO 1A03h

a. The number of entries equals the number of mapped PDOs, see “Data Representation” on page 21.

3.2 Manufacturer Specific Objects

3.2.1 Input Buffer

Index	Object Name	Sub-Index	Description	Type	Access	Notes
2000h	Inputs	00h	No. of entries	U8	RO	-
		01h	Input byte 0000	U8	RO	-
		02h	Input byte 0001			
		...	...			
		80h	Input byte 0127			
2001h	Inputs	00h	No. of entries	U8	RO	-
		01h	Input byte 0128	U8	RO	-
		02h	Input byte 0129			
		...	...			
		80h	Input byte 0255			
2002h	Inputs	00h	No. of entries	U8	RO	-
		01h	Input byte 0256	U8	RO	-
		02h	Input byte 0257			
		...	...			
		80h	Input byte 0383			
2003h	Inputs	00h	No. of entries	U8	RO	-
		01h	Input byte 0384	U8	RO	-
		02h	Input byte 0385			
		...	...			
		80h	Input byte 0511			

Note: The gateway will only create the number of objects needed to hold the subnetwork configuration.

See also...

- “Data Representation” on page 21

3.2.2 Output Buffer

Index	Object Name	Sub-Index	Description	Type	Access	Notes
2100h	Outputs	00h	No. of entries	U8	RO	-
		01h	Output byte 0000	U8	R(W)	-
		02h	Output byte 0001			
		...	...			
		80h	Output byte 0127			
2101h	Outputs	00h	No. of entries	U8	RO	-
		01h	Output byte 0128	U8	R(W)	-
		02h	Output byte 0129			
		...	...			
		80h	Output byte 0255			
2102h	Outputs	00h	No. of entries	U8	RO	-
		01h	Output byte 0256	U8	R(W)	-
		02h	Output byte 0257			
		...	...			
		80h	Output byte 0383			
2103h	Outputs	00h	No. of entries	U8	RO	-
		01h	Output byte 0384	U8	R(W)	-
		02h	Output byte 0385			
		...	...			
		80h	Output byte 0511			

Note 1: For consistency reasons, data declared as I/O data will be read-only.

Note 2: The gateway will only create the number of objects needed to hold the subnetwork configuration.

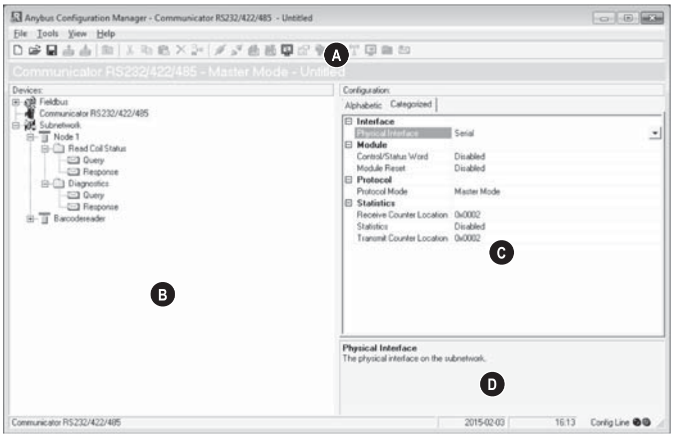
See also...

- “Data Representation” on page 21

10. Navigating ACM

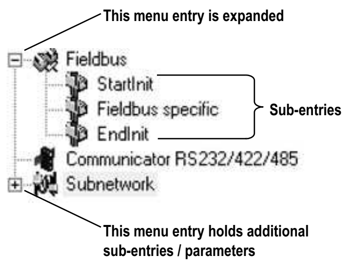
10.1 Main Window

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.