Fieldbus Appendix

Anybus-IC CANopen

Doc.Id. SCM-1200-004
Rev 1.04

HMS Industrial Networks AB

	☎		✉
Germany	+49	- 721 - 96472 - 0	ge-sales@hms-networks.com
Japan	+81	- 45 - 478 -5340	jp-sales@hms-networks.com
Sweden	+46	- 35 - 17 29 20	sales@hms-networks.com
U.S.A.	+1	- 312 - 829 - 0601	us-sales@hms-networks.com
France	+33	- 3 89 32 76 76	fr-sales@hms-networks.com
Italy	+39	- 347 - 00894 - 70	it-sales@hms-networks.com
China	+86	- 10 - 8532 - 3183	cn-sales@hms-networks.com

Important User Information

This document is intended to provide a good understanding of the functionality offered by Anybus-IC CANopen. The document only describes the features that are specific to the Anybus-IC CANopen. For general information regarding the Anybus-IC, consult the Anybus-IC design guides.

The reader of this document is expected to be familiar with high level software design, and communication systems in general. The use of advanced CANopen-specific functionality may require in-depth knowledge in CANopen networking internals and/or information from the official CANopen specifications. In such cases, the people responsible for the implementation of this product should either obtain the CANopen specification to gain sufficient knowledge or limit their implementation in such a way that this is not necessary.

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 application of this product. Those responsible for the use of this device must ensure that all the necessary steps have been taken to verify that the application meets all performance and safety requirements including any applicable laws, regulations, codes, and standards

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. 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-IC CANopen Fieldbus Appendix
Rev 1.04

Aug 2009 Doc Id SCM-1200-004

Table of Contents

Chapter 3 Fieldbus Specific Parameters

Chapter 4 CANopen Object Dictionary

Chapter 5 Fieldbus Interface

Appendix A Technical Specification

Appendix B Electronic Data Sheet (EDS) example

About This Document

How To Use This Document

This document is intended to provide a good understanding of the functionality offered by the Anybus-IC CANopen. The document only describes the features that are specific to the Anybus-IC CANopen, i.e. for general information regarding the Anybus-IC platform, consult the Anybus-IC Design Guide.

The reader of this document is expected to be familiar with high level software design, and communication systems in general. The use of advanced CANopen-specific functionality may require in-depth knowledge in CANopen networking internals and/or information from the official CANopen specification. In such cases, those responsible for the implementation of this product should either obtain the CANopen specification to gain sufficient knowledge or limit their implementation is such a way that this is not necessary.

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

Related Documents

Document	Author
Anybus-IC Design Guide	HMS
CiA Draft Standard 301 v4.02	CAN in Automation
-	-

Document History

Summary of Recent Changes (1.03 ... 1.04)

Change	Page(s)
Updated description of Status Indicators	2-5

Revision List

Revision	Date	Author	Chapter	Description
1.00	2008-03-30	PeP	All	Initial revision
1.01	2008-06-26	PeP	-	Minor update
1.02	2008-07-02	PeP	-	Minor update
1.03	2008-11-07	HeS	P, A	Minor update
1.04	2009-08-12	KeL	2	Minor update

Conventions & Terminology

The following conventions are used throughout this document:

- Numbered lists provide sequential steps
- Bulleted lists provide information, not procedural steps
- The terms 'Anybus' or 'module' refers to the Anybus-IC module.
- The term 'application' refers to the device that hosts the Anybus-IC module.
- Hexadecimal values are written as NNNNh or 0xNNNN, where NNNN is the actual hexadecimal value.

Support

HMS Sweden (Head Office)

E-mail:support@hms-networks.com
Phone: +46 (0) 35 - 17 29 20
Fax: +46 (0) 35 - 17 29 09
Online:www.anybus.com

HMS North America

E-mail:us-support@hms-networks.com
Phone: +1-312-829-0601
Toll Free:+1-888-8-Anybus
Fax: +1-312-738-5873
Online:www.anybus.com

HMS Germany

E-mail:ge-support@hms-networks.com
Phone: +49-721-96472-0
Fax: +49-721-964-7210
Online:www.anybus.com

HMS Japan

E-mail:jp-support@hms-networks.com
Phone: +81-45-478-5340
Fax: +81-45-476-0315
Online:www.anybus.com

HMS China

E-mail:cn-support@hms-networks.com
Phone: +86 10 8532 3023
Online:www.anybus.com

HMS Italy

E-mail:it-support@hms-networks.com
Phone: +39 039 59662 27
Fax: +39 039 59662 31
Online:www.anybus.com

HMS France

E-mail:mta@hms-networks.com
Phone: +33 (0) 3 89 32 76 41
Fax: +33 (0) 3 89 32 76 31
Online:www.anybus.com

About the Anybus-IC CANopen

General

The Anybus-IC CANopen communication module provides instant CANopen connectivity via the generic Anybus-IC application interface. Any device that supports this standard can take advantage of the features provided by the module, allowing seamless network integration regardless of network type.

This product conforms to all aspects of the application interface defined in the Anybus-IC Design Guide, which means that no dedicated software support is needed to be able to support the Anybus-IC CANopen. However, to be able to take advantage of optional network-specific functionality, a certain degree of dedicated software support may be necessary.

Features

- Complete CANopen slave functionality
- Up to 144 bytes of fieldbus I/O in each direction
- Fieldbus baudrates from 20kbps to 1Mbps
- Supports the Layer Setting Service (LSS)
- Automatic baudrate detection
- Supports PDO message types: COS, Cyclic Synchronous and Acyclic Synchronous
- Device identity customization
- Generic EDS-file provided

Conformance Notes

Fieldbus certification is pending; this product will be pre-certified for network compliance. While this is done to ensure that the final product ***can*** be certified, it does not necessarily mean that the final product will not require recertification.

- The EDS-file associated with this product must be altered to match the final implementation.
  See also 2-1 “Electronic Data Sheet (EDS)”.
- To ensure interoperability, the device identity must be customized. CiA (CAN in Automation) members should apply for a unique Vendor ID; non-members may contact HMS to obtain a custom Product ID. In either case, a custom Vendor ID is required when certifying the final product.

Contact HMS for further information.

Basic Operation

General Information

Software Requirements

Generally, no network-specific support code needs to be written in order to support the Anybus-IC CANopen. However, due to natural reasons, advanced fieldbus-specific features may require the use of CANopen-specific parameters.

For general information about the Anybus-IC software interface, consult the Anybus-IC Design Guide.

See also...

- 1-1 “Conformance Notes”

Electronic Data Sheet (EDS)

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

HMS supplies a generic. EDS-file which can serve as a basis for new implementations; however this file must be altered to match the end product (i.e. I/O configuration, identity settings etc.).

See also...

- 1-1 “Conformance Notes”
- B-1 “Electronic Data Sheet (EDS) example”

Device Identity

In its’ default state, the module appears as a generic HMS device with the following identity information:

Object Entry	Value
Vendor ID	0000 001Bh (HMS Industrial Networks)
Product Code	0000 000Bh (Anybus-IC)
Manufacturer Device Name	‘Anybus-IC’
Manufacturer Hardware Revision	-
Manufacturer Software Revision	(Anybus software revision)

See also...

- 1-1 “Conformance Notes”
- 4-1 “CANopen Object Dictionary”
- 3-5 “FB Vendor ID (Parameter #108)”
- 3-6 “FB Product Code (Parameter #109)”
- 3-6 “FB Product Name (Parameter #110)”
- 3-7 “FB Revision (Parameter #111)”

Data Exchange

General Information

The Fieldbus Input- and Output Data can be accessed as object entries in the manufacturer specific range (2000h...5FFFh). Separate object ranges are used for byte, word, and double-word access.

Words and double-words use Motorola (high byte first) format.

See also...

- 4-3 “Fieldbus I/O”

Fast vs. Slow Data

CANopen makes a distinction between fast cyclical I/O, and slower acyclic data. The former is suitable for time critical data, while the latter is more suitable for non-critical operations such as parameter settings etc.

The amount of data that shall be exchanged as fast cyclical I/O is specified by the CANopen master configuration. If not specified, the module will default to exchanging the first 32 Fieldbus I/O bytes as fast cyclical data. The remainder (if applicable) will still be available for acyclic communication through the CANopen object dictionary.

See also...

- 2-7 “CANopen Implementation”
- 4-3 “Fieldbus I/O”

Communication Settings

General Information

The module supports the Layer Setting Service (LSS). This service can be used to set the Baud Rate and Device Address via the network, and may address the module by it's Vendor-ID, Product Code, Revision number and serial number.

It is possible to enforce LSS during startup by setting the Device Address and/or Baud Rate to 0 (zero), or by setting the FBNA-bit parameter #8 to 1 (one).

Baud rate

On CANopen, the module supports 20 kbps, 50 kbps, 125 kbps, 250 kbps, 500 kbps, 800 kbps and 1 Mbps operation. The baud rate can be specified either via the fieldbus specific input register (in case of binary switches), via the SCI-interface, or by the network by means of the Layer Setting Service (LSS).

Automatic baud rate detection is also supported.

See also...

- Anybus-IC Design Guide (parameter #8 'Configuration Bits')
- 2-4 "Switches (Fieldbus Specific Input)"
- 2-8 "Network Services"
- 3-5 "FB Baud Rate Config (Parameter #106)"
- 3-5 "FB Baudrate Actual (Parameter #107)"

Node Address

The module supports BCD-coded switches (node address range 1... 99) or binary switches (node address range 1... 63). The node address can also be set via the SCI-interface or from the CANopen network by means of the Layer Setting Service (LSS).

See also...

- Anybus-IC Design Guide (parameter #8 'Configuration Bits')
- 2-4 "Switches (Fieldbus Specific Input)"
- 2-8 "Network Services"
- 3-3 "FB Node Address Config (Parameter #103)"
- 3-4 "FB Node Address SSC (Parameter #104)"
- 3-4 "FB Node Address Actual (Parameter #105)"

Fieldbus Specific Input/Output Registers

Switches (Fieldbus Specific Input)

The Fieldbus Specific Input register is used for fieldbus-specific configuration settings and supports two types of switches/coding.

- **BCD-coded input (BCD Switches)**

  This type of switches allows the CANopen node address to be set in the range 1... 99.

Switch Value	Node Address
00	(Use LSS setting)
01	1
02	2
03	3
...	...
97	97
98	98
99	99

  Note: The baud rate can not be specified using this type of switch. The module will instead establish the baud rate by means of auto-detection.

- **Binary-coded input (Hex Switches, Binary Switches)**

  This type if switch allows the CANopen node address and baud rate to be specified as follows:

b7	b6	b5	b4	b3	b2	b1	b0	Baud Rate	Node Address
0	0							(Autobaud)	-
0	1							250kbps	
1	0							500kbps	
1	1							1Mbps	
		0	0	0	0	0	0	-	(Use LSS setting)
		0	0	0	0	0	1		1
		0	0	0	0	1	0		2
		0	0	0	0	1	1		3
		...	...	...	...	...	...		...
		1	1	1	1	0	1		61
		1	1	1	1	1	0		62
		1	1	1	1	1	1		63

Note: The switch type is specified by parameter #9 ('Switch Coding'). On CANopen, the default value of this parameter is 01h (Binary Switches). Note however that there is no guarantee that the same default value is used on other networks.

See also...

- Anybus-IC Design Guide (parameter #9 'Switch Coding')
- 2-3 "Communication Settings"

Status Indicators (Fieldbus Specific Output)

General

The Anybus-IC CANopen uses status indications as follows:

Bit	Color	LED	Comments
0	-	-	(not used on CANopen)
1	Red	Status	-
2	Green	Run	-
3	-	-	(not used on CANopen)
4	Green	Init	-
5	-	-	(not used on CANopen)
6	-	-	(not used on CANopen)
7	Red	Error	-

The standard indications are as follows:

LED	State	CANopen Indication	Comments
Status	Off	Normal Operation	-
	Red	Unrecoverable Fault	-
Run	Off	-	(not Initialized)
	Green	OPERATIONAL	Device in 'OPERATIONAL'-state
	Single flash	STOPPED	Device in 'STOPPED'-state
	Toggling	PRE-OPERATIONAL	Device in 'PRE-OPERATIONAL'-state
	Flickering	Autobaud/LSS	Baud rate auto-detection or LSS services in progress (alternating flickering with Error LED)
Init	Off	Not initialized	This LED indicates when the Anybus module has passed its internal initialisation procedures
	Green	Initialized	
Error	Off	-	(no error)
	Red	Bus Off	CAN controller is in Bus Off
	Single flash	Warning Limit Reached	At least one of the error counters in the CAN controller has reached or exceeded its warning level (e.g. too many error frames)
	Double flash	Error Control Event	A guard event (NMT-slave or NMT-master) or a heart-beat event (Heartbeat consume) has occurred.
	Flickering	Autobaud/LSS	Baud rate auto-detection or LSS services in progress (alternating flickering with Run LED).

Recommended indications for labels etc.:

LED	Recommended label on product
Status	'CAN-STAT' or 'STAT'
Run	'CAN-RUN' or 'RUN'
Init	'INIT'
Error	'CAN-ERR' or 'ERR'

See also...

- Anybus-IC Design Guide (parameter #7 'LED State')

Network Reset Handling

General Information

The module may receive reset commands from the network or spontaneously generate reset due to an error. The application can be notified of network reset events through the interrupt mechanisms outlined in the Anybus-IC Design Guide.

The following parameters are involved when dealing with network reset requests:

- Parameter #12 ('Interrupt Config')
- Parameter #13 ('Interrupt Cause')

Note: Reset type 'Set Default' (i.e. the 'DEF'-bit in parameters #12 and #13) is not available on the Anybus-IC CANopen.

For more information about network reset procedures, consult the general Anybus-IC Design Guide.

Reset Node

Upon receiving a 'Reset Node'-request, the module will shift the CANopen bus interface into a physically passive state, reset all internal parameters, and re-initialise the CANopen bus system.

The behaviour that follows depends on the value of the 'RES'-bit in parameter #13:

- **'RES'-bit set**

  Upon receiving a 'Reset Node'-request, the module will shift the bus interface into a physically passive state and reset all internal parameters. It will then indicate an interrupt to the application through the 'RES'-bit of parameter #12, and await a hardware reset.

- **'RES'-bit cleared**

  Upon receiving a 'Reset Node'-request, the module will reset it's internal parameters.

  No reset command will be issued to the host application.

Reset Communication

Upon receiving a 'Reset Communication'-request, the module will reset all communication object entries to their default values, and shift to the 'Reset Communication'-state.

No reset command will be issued to the host application.

CANopen Implementation

General Information

The Anybus module implements slave functionality according to revision 4 of the CANopen protocol.

See also...

- 2-8 “Network Services”
- 2-9 “Default COB-IDs”
- 2-10 “PDO Triggering Modes”
- 3-10 “FB Initialisation (Parameter #116)”

Network Services

The module implements support for the following CANopen services:

- **Service Data Object (SDO Server)**

  SDO uses asynchronous data transmission, and provides unscheduled access to all objects in the module. SDO transmission is usually associated with less time critical data, and enables transfers larger than 8 bytes, which is the upper limit of PDO transfers (below).

  The module supports both Expedited- and Segmented SDO transfers.

  See also...

  - 2-2 “Data Exchange”
  - 4-1 “CANopen Object Dictionary”(2-2 “Data Exchange”)

- **Process Data Objects (PDO)**

  PDO provides a fast communication channel for time critical I/O. The module supports up to 24 PDOs in each direction, each carrying up to 8 bytes of data.

  Multiplexed PDOs are not supported by the Anybus-IC implementation.

  See also...

  - 2-9 “Default COB-IDs”
  - 2-10 “PDO Triggering Modes”

- **Synchronisation Object (SYNC)**

  This object synchronises the PDO communication. The module acts as a sync-consumer.

- **Emergency Object (EMCY)**

  The Emergency Object provides mechanisms of reporting serious conditions to the network control system. The module uses this service spontaneously when appropriate, but it may also be used directly by the application through parameter #118.

  See also...

  - 3-11 “FB Emcy Code (Parameter #118)”

- **Layer Setting Services (LSS)**

  These services enables baudrate and node address configuration by means of software only.

  The module always acts as an LSS slave in this context.

  See also...

  - 2-3 “Communication Settings”
  - 2-4 “Fieldbus Specific Input/Output Registers”

- **Node Guarding & Heartbeat Message (Online-Offline Functionality)**

  When the module detects a bus error that prevents communication on the bus (i.e. BUSOFF), the module will indicate the network as being offline. In addition to this, the module can optionally also report the network as being offline based on Node Guarding and Heartbeat mechanisms.

  See also...

  - 3-8 “FB Configuration Bits (Parameter #113)”

Default COB-IDs

By default, the module uses the following COB-IDs:

- **RPDO**

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, bytes 0... 7	Enabled
2	300h + Node ID	300h + Node ID	Output Data, bytes 8...15	
3	400h + Node ID	400h + Node ID	Output Data, bytes 16... 23	
4	500h + Node ID	500h + Node ID	Output Data, bytes 24... 31	
5	240h + Node ID	500h	Output Data, bytes 32... 39	Disabled
6	340h + Node ID	500h	Output Data, bytes 40... 47	
7	440h + Node ID	500h	Output Data, bytes 48... 55	
8	540h + Node ID	500h	Output Data, bytes 56... 63	
9... 24	500h	500h	-	

- **TPDO**

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, bytes 0... 7	Enabled
2	280h + Node ID	280h + Node ID	Input Data, bytes 8... 15	
3	380h + Node ID	380h + Node ID	Input Data, bytes 16... 23	
4	480h + Node ID	480h + Node ID	Input Data, bytes 24... 31	
5	1C0h + Node ID	500h	Input Data, bytes 32... 39	Disabled
6	2C0h + Node ID	500h	Input Data, bytes 40... 47	
7	3C0h + Node ID	500h	Input Data, bytes 48... 55	
8	4C0h + Node ID	500h	Input Data, bytes 56... 63	
9... 24	500h	500h	-	

See also...

- 2-10 “PDO Triggering Modes”

PDO Triggering Modes

Two triggering modes are supported:

- **Event Driven**

  Message transmission is triggered by:

Transmission Type	Description	Notes
254/255	COS	When Process data have been changed (performance depends on the number of PDO's using COS)
1... 240	Cyclic Synchronous	For synchronous this is the expiration of the specified transmission period, synchronized by the reception of the SYNC object. The data will be synced only to the Anybus module (current process data in buffer) and not all the way down to the application.
0	Acyclic Synchronous	Sent on SYNC and on the COS event.

- **Timer Driven**

  Message transmission is either triggered by the occurrence of a device-specific event (COS) or if a specified has elapsed without the occurrence of the event.

Transmission Type	Description	Notes
254/255	COS/Timer	Message transmission is either triggered by the occurrence of a device-specific event (COS) or if a specified time has elapsed without occurrence of the event (Event Timer; specified separately for each TPDO in object entries 1800h-1817h, sub-index 05h)

See also...

- 2-9 "Default COB-IDs"

Fieldbus Specific Parameters

The following fieldbus-specific parameters are available to the application:

Note: Byte sized parameter values are placed in the least significant byte of the word.

FB Bus Status (Parameter #100)

This parameter holds information about the current status of the fieldbus communication.

Parameter Name	'FB Bus Status'
Parameter Number	100
Modbus Address	7000h
Default Value	-
Range	01h... 04h
Size	1 byte
Stored in NV RAM	No
Access	R

- **Value**
  - 1: Bus running
  - 2: Bus off error
  - 3: Error passive
  - 4: Other error

FB Module Status (Parameter #101)

This parameter holds information about the current status of the module with regards to the CANopen communication.

Parameter Name	'FB Module Status'
Parameter Number	101
Modbus Address	7001h
Default Value	00h
Range	00h... 04h
Size	1 byte
Stored in NV RAM	No
Access	R

- **Value**
  - 0: initialisation
  - 1: Init error
  - 2: Stopped
  - 3: Pre-operational
  - 4: Operational

FB Password (Parameter #102)

This parameter grants write access to the following parameter provided that a valid password is supplied:

- 3-5 “FB Vendor ID (Parameter #108)”
- 3-6 “FB Product Code (Parameter #109)”
- 3-6 “FB Product Name (Parameter #110)”
- 3-7 “FB Revision (Parameter #111)”
- 3-11 “Serial Number (Parameter #117)”

(The password can be obtained by contacting HMS)

Parameter Name	‘FB Password’
Parameter Number	102
Modbus Address	7002h
Default Value	-
Range	0000h... FFFFh
Size	2 bytes
Stored in NV RAM	No
Access	W

FB Node Address Config (Parameter #103)

This parameter holds the manually configured CANopen node address. Note that in order for this value to be valid, bit 4 of parameter #8 (“Configuration Bits”) must be set.

Note: This parameter is read once during startup, i.e. any changes requires a reset in order to have effect.

Parameter Name	‘FB Address Cfg’
Parameter Number	103
Modbus Address	7003h
Default Value	01h
Range	00h... 7Fh
Size	1 byte
Stored in NV RAM	Yes
Access	R/W

- **Value**

  Valid node address settings range from 1 to 128.

FB Node Address SSC (Parameter #104)

This parameter holds the auto configured fieldbus node address from the SSC interface. Note that in order for this value to be valid, bit 4 of parameter #8 (“Configuration Bits”) must be cleared.

If the SSC node address is larger than allowed by the fieldbus, the default value of parameter #103 (“FB Node Address Config”) will be used as the actual node address.

Note: This parameter is read once during startup, i.e. any changes requires a reset in order to have effect.

Parameter Name	‘FB Address SSC’
Parameter Number	104
Modbus Address	7004h
Default Value	-
Range	00h... 3Fh
Size	1 byte
Stored in NV RAM	No
Access	R

FB Node Address Actual (Parameter #105)

After initialisation, this parameter holds the actual fieldbus node address. Prior to module initialisation, the value of this parameter is not defined.

Parameter Name	‘FB Address Act’
Parameter Number	105
Modbus Address	7005h
Default Value	-
Range	00h... 7Fh
Size	1 byte
Stored in NV RAM	No
Access	R

FB Baud Rate Config (Parameter #106)

This parameter holds the actual fieldbus baud rate after the Anybus-IC is on-line. Before the Anybus-IC is on-line, the value of this parameter is not defined.

Parameter Name	'FB Baud Rate Config'
Parameter Number	106
Modbus Address	7006h
Default Value	-
Range	00h... FFh
Size	1 byte
Stored in NV RAM	Yes
Access	R

- **Value**

  00h: Baud rate specified by means of the Layer Setting Service (LSS)
  01h: 20kbps
  02h: 50kbps
  03h: 125kbps
  04h: 250kbps
  05h: 500kbps
  06h: 800kbps
  07h: 1Mbps
  FFh: Automatic baud rate detection

FB Baudrate Actual (Parameter #107)

This parameter reflects the actual CANopen baudrate.

Parameter Name	'FB Baudrate Actual'
Parameter Number	107
Modbus Address	7007h
Default Value	1810h
Range	01h... 07h
Size	1 byte
Stored in NV RAM	No
Access	R

- **Value**

  01h: 20kbps
  02h: 50kbps
  03h: 125kbps
  04h: 250kbps
  05h: 500kbps
  06h: 800kbps
  07h: 1Mbps

FB Vendor ID (Parameter #108)

This parameter holds the CANopen Vendor ID.

Note: This parameter is password protected (see 3-3 "FB Password (Parameter #102)").

Parameter Name	'FB Vendor ID'

Parameter Number	108
Modbus Address	7008h
Default Value	-
Range	0000 0000h... FFFF FFFFh
Size	4 bytes
Stored in NV RAM	Yes
Access	R(W)

FB Product Code (Parameter #109)

This parameter holds the CANopen Product Code.

Note: This parameter is password protected (see 3-3 “FB Password (Parameter #102)”).

Parameter Name	‘FB Product Code’
Parameter Number	109
Modbus Address	700Ah
Default Value	-
Range	0000 0000h... FFFF FFFFh
Size	4 bytes
Stored in NV RAM	Yes
Access	R(W)

FB Product Name (Parameter #110)

This parameter holds the CANopen Product Name.

Note: This parameter is password protected (see 3-3 “FB Password (Parameter #102)”).

Parameter Name	‘FB Product Name’
Parameter Number	110
Modbus Address	700Ch... 701Bh
Default Value	‘Anybus-IC’
Range	(null terminated string, max 32 bytes)
Size	32 bytes
Stored in NV RAM	Yes
Access	R(W)

FB Revision (Parameter #111)

This parameter holds the CANopen product revision number.

Note: This parameter is password protected (see 3-3 "FB Password (Parameter #102)").

Parameter Name	'FB Revision'
Parameter Number	111
Modbus Address	701Ch
Default Value	(Anybus product revision)
Range	0000... FFFFh
Size	2 bytes
Stored in NV RAM	Yes
Access	R(W)

FB Status Bits (Parameter #112)

This parameter indicates LSS and parameter storage status.

Parameter Name	'FB Status Bits'
Parameter Number	112
Modbus Address	701Dh
Default Value	00h
Range	00h... FFh
Size	1 byte
Stored in NV RAM	No
Access	R

Bit-field Layout

b7	b6	b5	b4	b3	b2	b1	b0
-	-	-	-	-	LSS2	LSS1	MEM

- **MEM**
  - 1: Current parameter settings stored in non-volatile memory
  - 0: Current parameter settings not stored in non-volatile memory
- **LSS1**
  - 1: Node address set by LSS
  - 0: Node address not set by LSS
- **LSS2**
  - 1: Baud rate set by LSS
  - 0: Baud rate not set by LSS

FB Configuration Bits (Parameter #113)

This parameter holds various fieldbus-related configuration parameters.

Parameter Name	'FB Configuration Bits'
Parameter Number	113
Modbus Address	701Eh
Default Value	00h
Range	00h... FFh
Size	2 bytes
Stored in NV RAM	Yes
Access	R/W

Bit-field Layout

b7	b6	b5	b4	b3	b2	b1	b0
-	-	-	-	-	-	-	OL

- **OL**

  1: Offline is also set at node guarding event and heartbeat event
  0: Offline is set only at bus off

Restore Parameters (Parameter #114)

Parameter #112 indicates whether or not parameters have been saved in non-volatile memory.

If bit #0 of parameter #112 is set, parameter values have been save in non-volatile memory. To restore these parameters to their default values, 'load' is written to this parameter.

Parameter Name	'Restore Parameters'
Parameter Number	114
Modbus Address	701Fh
Default Value	7Eh
Range	0000 0000h... 6C6F 6164h
Size	4 bytes
Stored in NV RAM	No
Access	W

Error Messages (Parameter #115)

This parameter holds error messages and fatal events generated in the module.

Parameter Name	'Error Messages'
Parameter Number	115
Modbus Address	7022h... 7041h
Default Value	-
Range	-
Size	64 bytes
Stored in NV RAM	Yes
Access	R

Contents:

Bytes	Contents
0... 1	Error code #1
2... 11	File where the first error occurred
12... 13	Row where the first error occurred
14... 31	(reserved)
32... 33	Error code #2
34... 43	File where the second error occurred
44... 45	Row where the second error occurred
46... 63	(reserved)

FB Iinitialisation (Parameter #116)

This parameter is used to acknowledge information from the Anybus-IC module during the fieldbus-specific initialisation.

Note: This parameter shall only be used when parameter #1 is set to (0002h).

Parameter Name	'FB initialisinitialisationation'
Parameter Number	116
Modbus Address	7042h
Default Value	00h
Range	00h... FFh
Size	1 byte
Stored in NV RAM	No
Access	R

Bit-field Layout

b7	b6	b5	b4	b3	b2	b1	b0
-	-	-	-	-	-	-	VER3

- **VER3**
  1: Use CANopen protocol version 3
  0: Use CANopen protocol version 4 (default)

Serial Number (Parameter #117)

This parameter holds the serial number of the module.

Note 1: This parameter is password protected (see 3-3 “FB Password (Parameter #102)”).

Note 2: If using custom serial numbers, always make sure that the combination of Vendor ID and serial number is unique. Failure to comply with this requirement may have a negative impact on network functionality such as LSS.

Parameter Name	‘Serial Number’
Parameter Number	117
Modbus Address	7043h
Default Value	-
Range	0000 0000h... FFFF FFFFh
Size	4 bytes
Stored in NV RAM	Yes
Access	R(W)

FB Emcy Code (Parameter #118)

This parameter is used to forward an emergency code from the application to the CANopen network.

Parameter Name	‘FB Emcy Codes’
Parameter Number	118
Modbus Address	7045h
Default Value	-
Range	00000000h... FFFFFFFFh
Size	4 bytes
Stored in NV RAM	No
Access	R

Contents:

Byte	Contents
0 (high)	Value:Meaning: 00h Reset 01h: Set
1	(reserved, set to zero)
2	CANopen emergency code
3 (low)	

CANopen Object Dictionary

Standard Objects

General Information

The standard object dictionary is implemented according to the DS302 specification (v4.02) from CiA (CAN in Automation). Note that certain object entries correspond to parameter settings in the module.

Object Entries

Index	Object Name	Sub-Index	Description	Type	Access	Notes
1000h	Device Type	00h	Device Type	U32	RO	0000 0000h (No profile)
1001h	Error register	00h	Error register	U8	RO	-
1003h	Pre-defined error field	00h	Number of errors	U8	RW	See 3-11 “FB Emcy Code (Parameter #118)”
		01h... 05h	Error field	U32	RO	
1005h	COB-ID Sync	00h	COB-ID Sync	U32	RW	Default value is 0000 0080h
1008h	Manufacturer device name	00h	Manufacturer device name	Visible string	RO	See 3-6 “FB Product Name (Parameter #110)”
100Ah	Manufacturer software version	00h	Manufacturer software version	Visible string	RO	See 3-7 “FB Revision (Parameter #111)”
100Ch	Guard time	00h	Guard time	U16	RW	-
100Dh	Life time factor	00h	Life time factor	U8	RW	-
1010h	Store Parameters[a]	00h	Largest sub index supported	U8	RO	02h
		01h	Store all parameters	U32	RW	Baud rate and Node ID cannot be stored using this command.
		02h	Store Communication parameters	U32	RW	
1011h	Restore parameters	00h	Largest sub index supported	U8	RO	02h
		01h	Restore all default parameters	U32	RW	-
		02h	Restore communication default parameters	U32	RW	-
1014h	COB ID EMCY	00h	COB ID EMCY	U32	RO	-
1015h	Inhibit Time EMCY	00h	Inhibit Time EMCY	U16	RW	Default value is 0000h
1016h	Consumer Heartbeat Time	00h	Number of entries	U8	RO	01h
		01h	Consumer Heartbeat Time	U32	RW	Node ID + Heartbeat Time. Value must be a multiple of 1ms.
1017h	Producer Heartbeat Time	00h	Producer Heartbeat Time	U16	RW	-

Index	Object Name	Sub-Index	Description	Type	Access	Notes
1018h	Identity object	00h	Number of entries	U8	RO	04h
		01h	Vendor ID	U32	RO	See 3-5 “FB Vendor ID (Parameter #108)”
		02h	Product Code	U32	RO	See 3-6 “FB Product Code (Parameter #109)”
		03h	Revision Number	U32	RO	See 3-7 “FB Revision (Parameter #111)”
		04h	Serial Number	U32	RO	See 3-11 “Serial Number (Parameter #117)”
1400h ... 1417h	Receive PDO parameter	00h	Largest sub-index supported	U8	RO	02h
		01h	COB ID used by PDO	U32	RW	-
		02h	Transmission type.	U8	RW	-
1600h ... 1617h	Receive PDO mapping	00h	No. of mapped application objects in PDO	U8	RW	-
		01h	Mapped object #1	U32	RW	-
		02h	Mapped object #2	U32	RW	-
		03h	Mapped object #3	U32	RW	-
		04h	Mapped object #4	U32	RW	-
		05h	Mapped object #5	U32	RW	-
		06h	Mapped object #6	U32	RW	-
		07h	Mapped object #7	U32	RW	-
		08h	Mapped object #8	U32	RW	-
1800h ... 1817h	Transmit PDO parameter	00h	Largest sub-index supported	U8	RO	05h
		01h	COB ID used by PDO	U32	RW	-
		02h	Transmission type	U8	RW	-
		03h	Inhibit time	U16	RW	-
		05h	Event Timer (ms)	U16	RW	-
1A00h ... 1A17h	Transmit PDO mapping	00h	No. of mapped application objects in PDO	U8	RW	-
		01h	Mapped object #1	U32	RW	-
		02h	Mapped object #2	U32	RW	-
		03h	Mapped object #3	U32	RW	-
		04h	Mapped object #4	U32	RW	-
		05h	Mapped object #5	U32	RW	-
		06h	Mapped object #6	U32	RW	-
		07h	Mapped object #7	U32	RW	-
		08h	Mapped object #8	U32	RW	-

a. Relevant only for communication parameters

Fieldbus I/O

Fieldbus Input Data

Index	Object Name	Sub-Index	Description	Type	Access	Notes
2000h	Input Buffer	00h	No. of entries	U8	RO	(byte access)
		01h	Input Buffer byte #0	U8	RO	
		02h	Input Buffer byte #1			
		...	...			
		80h	Input buffer byte #127			
2001h	Input Buffer	00h	No. of entries	U8	RO	
		01h	Input Buffer byte #128	U8	RO	
		02h	Input Buffer byte #129			
		...	...			
		10h	Input buffer byte #143			
-	-	-	-	-	-	-
2010h	Input Buffer	00h	No. of entries	U8	RO	(word access)
		01h	Input Buffer word #0	U16	RO	
		02h	Input Buffer word #1			
		...	...			
		40h	Input buffer word #63			
2011h	Input Buffer	00h	No. of entries	U8	RO	
		01h	Input Buffer word #64	U16	RO	
		02h	Input Buffer word #65			
		...	...			
		08h	Input buffer word #71			
-	-	-	-	-	-	-
2020h	Input Buffer	00h	No. of entries	U8	RO	(double word access)
		01h	Input Buffer dword #0	U32	RO	
		02h	Input Buffer dword #1			
		...	...			
		20h	Input buffer dword #31			
2021h	Input Buffer	00h	No. of entries	U8	RO	
		01h	Input Buffer dword #32	U32	RO	
		02h	Input Buffer dword #33			
		03h	Input Buffer dword #34			
		04h	Input buffer dword #35			

Fieldbus Output Data

Index	Object Name	Sub-Index	Description	Type	Access	Notes
2100h	Output Buffer	00h	No. of entries	U8	RO	(byte access)
		01h	Output Buffer byte #0	U8	R/W	
		02h	Output Buffer byte #1			
		...	...			
		80h	Output buffer byte #127			
2101h	Output Buffer	00h	No. of entries	U8	RO	
		01h	Output Buffer byte #128	U8	R/W	
		02h	Output Buffer byte #129			
		...	...			
		10h	Output buffer byte #143			
-	-	-	-	-	-	-
2110h	Output Buffer	00h	No. of entries	U8	RO	(word access)
		01h	Output Buffer word #0	U16	R/W	
		02h	Output Buffer word #1			
		...	...			
		40h	Output buffer word #63			
2111h	Output Buffer	00h	No. of entries	U8	RO	
		01h	Output Buffer word #64	U16	R/W	
		02h	Output Buffer word #65			
		...	...			
		08h	Output buffer word #71			
-	-	-	-	-	-	-
2120h	Output Buffer	00h	No. of entries	U8	RO	(double word access)
		01h	Output Buffer dword #0	U32	R/W	
		02h	Output Buffer dword #1			
		...	...			
		20	Output buffer dword #31			
2121h	Output Buffer	00h	No. of entries	U8	RO	
		01h	Output Buffer dword #32	U32	R/W	
		02h	Output Buffer dword #33			
		03h	Output Buffer dword #34			
		04h	Output buffer dword #35			

Miscellaneous

Index	Object Name	Sub-Index	Description	Type	Access	Notes
2205h	Module State Indicator	-	Reflects the state of the module on the network	U8	RO	1: Init error 2: Prepared 3: Pre-operational 4: Operational

Fieldbus Interface

General Considerations

CANopen specifies a maximum stub-length of 0.3m. This means that the internal stub-length (i.e. the distance between the Anybus-IC and the fieldbus connector) needs to be kept as short as possible, since this distance affects the maximum external stub length in the final installation.

Due to the individual requirements (i.e. differences in cable shield filters, max. stub-line length etc.) for each networking system, special care has to be taken if compatibility with several networking systems is required. It is therefore generally recommended to also study the design examples in the fieldbus appendices for the other members of the Anybus-IC family.

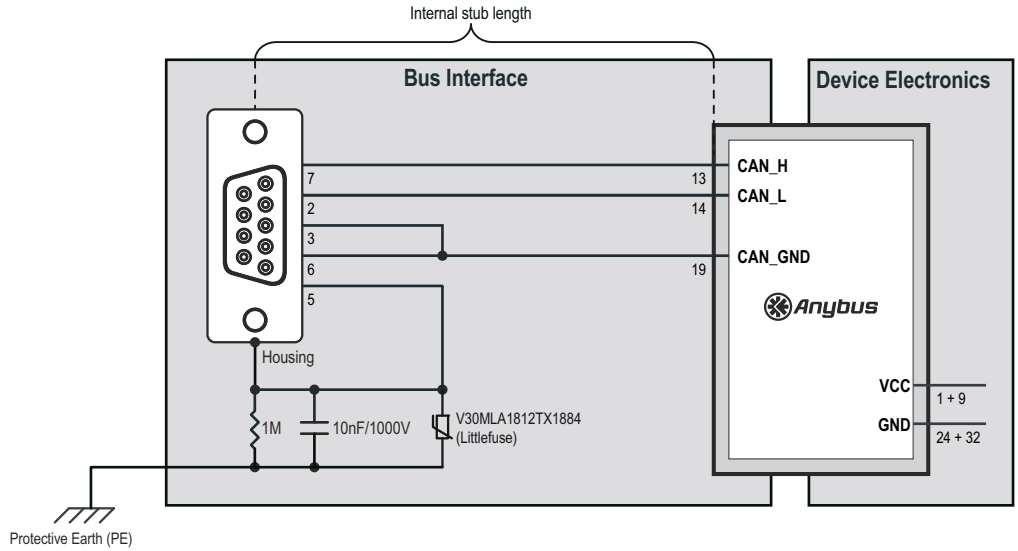
See also...

- 1-1 “Conformance Notes”
- 5-2 “DB9M Pinout”

IMPORTANT: *The recommendations regarding PE and cable shield filters etc. has changed slightly compared to that of older revisions of the Anybus-IC documentation. This has been done as to improve the EMC and EMI specification for the product and has no impact on backwards compatibility (i.e. the module remains compatible the older recommendations).*

Typical Implementation

CANopen requires a cable shield filter as shown below. If multiple networks are to be supported using the same application PCB, this has to be accounted for when routing the board, since other networking systems may need a different shielding approach.



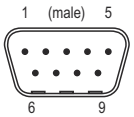
Note: To ensure proper EMC/EMI characteristics, the metal housing DB9M-connector must be connected to Protective Earth (PE) as illustrated above.

See also...

- 5-1 “General Considerations”

DB9M Pinout

CANopen Connector (DB9M)		Anybus	
Pin	**Signal**	**Pin**	**Signal**
1	-	-	-
2	CAN_L	14	FB2
3	CAN_GND	19	FB5
4	-	-	-
5	CAN_SHLD	-	-
6	CAN_GND	19	FB5
7	CAN_H	13	FB1
8	-	-	-
9	-	-	-
Housing	CAN_SHLD	-	-



Technical Specification

Electrical Specification

Protective Earth (PE) Requirements

See 5-1 "Fieldbus Interface".

Power Supply

Supply Voltage

The module requires a regulated 5V±5% DC power supply as specified in the Anybus-IC Design Guide.

Power Consumption

The maximum power consumption is 150mA.

Environmental Specification

- **Temperature**
  Test performed according to IEC-68-2-1 and IEC 68-2-2.
  Operating:-40 to +85 °C (-40 to 185°F)
  Storage: -40 to +85 °C (-40 to 185°F)

- **Humidity**
  The product is designed for a relative humidity of 5 to 95% non-condensing.
  Test performed according to IEC 68-2-30.

EMC Compliance (CE)

EMC pre-compliance testing has been conducted according to the following standards:

- **Emission:** EN 50 081-2:1993
  Tested perEN 55011:1998, class A, radiated

- **Immunity:** EN 61000-6-2: 1999
  Tested perEN 61000-4-2:1995
  EN 61000-4-3:1996
  EN 61000-4-4:1995
  EN 61000-4-5:1995
  EN 61000-4-6:1996

Electronic Data Sheet (EDS) example

Each device is associated with an EDS file, which describes the implementation of the product. This file is used by network configurations tool during network configuration.

The latest version of EDS file can either be downloaded from HMS website (www.hms-networks.com) or obtained by contacting HMS. The following is a screenshot of a CANopen module EDS-file where the following customizable information is shown:

- Vendor Number (ID)
- Product Name
- Product Code etc...

```
[FileInfo]
FileName=EDS_ABIC_COP_V_1_00_02.eds
FileVersion=1
FileRevision=00
Description=EDS file for the Anybus-IC CANopen module
CreationTime=08:00PM
CreationDate=06-23-2008
CreatedBy=Mikael Martensson, HMS Industrial Networks
ModificationTime=05:15PM
ModificationDate=06-25-2008
ModifiedBy=Mikael Martensson, HMS Industrial Networks
EDSVersion=4.0

[DeviceInfo]
VendorName=HMS Industrial Networks
ProductName=Anybus-IC
BaudRate_10=0
BaudRate_20=1
BaudRate_50=1
BaudRate_125=1
BaudRate_250=1
BaudRate_500=1
BaudRate_800=1
BaudRate_1000=1
Granularity=0x8
DynamicChannelsSupported=0
VendorNumber=0x1B
ProductNumber=11
SimpleBootUpMaster=0
SimpleBootUpSlave=1
RevisionNumber=0x00010000
OrderCode=0
GroupMessaging=0
NrOfRXPDO=24
NrOfTXPDO=24
LSS_Supported=1
```