Anybus® CompactCom™ B40 Modbus Serial

USER MANUAL

SCM-1202-130 2.2 en-US ENGLISH

Important User Information

Disclaimer

The information in this document is for informational purposes only. Please inform HMS Industrial Networks of any inaccuracies or omissions found in this document. HMS Industrial Networks disclaims any responsibility or liability for any errors that may appear in this document.

HMS Industrial Networks reserves the right to modify its products in line with its policy of continuous product development. The information in this document shall therefore not be construed as a commitment on the part of HMS Industrial Networks and is subject to change without notice. HMS Industrial Networks makes no commitment to update or keep current the information in this document.

The data, examples and illustrations found in this document are included for illustrative purposes and are only intended to help improve understanding of the functionality and handling of the product. In view of the wide range of possible applications of the product, and because of the many variables and requirements associated with any particular implementation, HMS Industrial Networks cannot assume responsibility or liability for actual use based on the data, examples or illustrations included in this document nor for any damages incurred during installation of the product. Those responsible for the use of the product must acquire sufficient knowledge in order to ensure that the product is used correctly in their specific application and that the application meets all performance and safety requirements including any applicable laws, regulations, codes and standards. Further, HMS Industrial Networks will under no circumstances assume liability or responsibility for any problems that may arise as a result from the use of undocumented features or functional side effects found outside the documented scope of the 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.

Table of Contents

Page

1 Preface

1.1 About this Document

This document is intended to provide a good understanding of the functionality offered by the Anybus CompactCom B40 Modbus Serial.

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

For additional information, please visit the support website at www.anybus.com/support.

1.2 Document History

Version	Date	Description
1.0	2019-04-30	First release
1.1	2019-05-16	Updated for PROFINET release
2.0	2019-11-28	Completely reworked
2.1	2019-12-18	Minor updates
2.2	2020-05-08	Minor updates

1.3 Document Conventions

Numbered lists indicate tasks that should be carried out in sequence:

1. First do this
2. Then do this

Bulleted lists are used for:

- Tasks that can be carried out in any order
- Itemized information

► An action

→ and a result

User interaction elements (buttons etc.) are indicated with bold text.

```
Program code and script examples
```

Cross-reference within this document: *Document Conventions, p. 3*

External link (URL): www.hms-networks.com

WARNING
Instruction that must be followed to avoid a risk of death or serious injury.

Caution
Instruction that must be followed to avoid a risk of personal injury.

Instruction that must be followed to avoid a risk of reduced functionality and/or damage to the equipment, or to avoid a network security risk.

Additional information which may facilitate installation and/or operation.

1.4 Document Specific Conventions

- The terms “Anybus” or “module” refers to the Anybus CompactCom module.
- The terms “host” or “host application” refer to the device that hosts the Anybus.
- Hexadecimal values are written in the format NNNNh or 0xNNNN, where NNNN is the hexadecimal value.
- A byte always consists of 8 bits.
- All dimensions in this document have a tolerance of ±0.10 mm unless otherwise stated.
- Outputs are TTL compliant unless otherwise stated.
- Signals which are “pulled to GND” are connected to GND via a resistor.
- Signals which are “pulled to 3V3” are connected to 3V3 via a resistor.
- Signals which are “tied to GND” are directly connected to GND,
- Signals which are “tied to 3V3” are directly connected to 3V3.

1.4.1 Pin Types

The pin types of the connectors are defined in the table below. The pin type may be different depending on which mode is used.

Pin type	Definition
I	Input
O	Output
I/O	Input/Output (bidirectional)
OD	Open Drain
Power	Pin connected directly to module power supply, GND or 3V3

1.5 Trademark Information

Anybus® is a registered trademark of HMS Industrial Networks.

EtherNet/IP is a trademark of ODVA, Inc.

All other trademarks are the property of their respective holders.

2 About the Anybus CompactCom B40 Modbus Serial

2.1 General Information

The Anybus CompactCom B40 Modbus Serial is a communication solution for simple industrial field devices. The host application communicates with the product using the Modbus RTU protocol. The Anybus CompactCom B40 Modbus Serial then communicates the data to the network. Typical applications are basic level I/O blocks, temperature controllers, measuring devices, and sensors.

The Anybus CompactCom B40 Modbus Serial software interface is designed to be network protocol independent, making it possible to support several networking systems using the same application software code/driver.

All Anybus CompactCom B40 Modbus Serial devices share footprint and electrical interface with the other members of the product family, independent of fieldbus or network. The host application connector provides an interface between the host application (Modbus RTU) and the Anybus CompactCom, while the network connector provides access to the chosen network. The Anybus CompactCom acts as a Modbus RTU slave on the host application side.

The Anybus CompactCom 40 family offers a wide range of functionality. For advanced products and applications, we recommend the standard Anybus CompactCom 40.

For general information about other products using the Anybus CompactCom 40 platform, consult www.anybus.com/support.

This 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.

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.

2.2 Features

- Simple implementation and great connectivity flexibility
- Serial Communications Interface (UART), that can be connected to almost any standard microcontroller
- Standard Modbus RTU serial protocol
- Galvanic isolation between the host application and the industrial network available if used with the CompactCom B40 connector board
- Low power consumption
- Control pins for status indications according to each network standard
- Separate network connector board available
- Precompliance tested for network conformance (where applicable)
- Precompliance tested for CE & UL. Contact HMS Industrial Networks for further information

> **!** All Anybus CompactCom 40 Modbus Serial devices will be precertified for network conformance. This is done to ensure that the final product can be certified, but it does not necessarily mean that the final product does not require recertification. Contact HMS Industrial Networks for further information.

2.3 Overview

The picture below shows the data flow in the Anybus CompactCom B40 Modbus Serial. The application sets up the Modbus RTU communication, and the Anybus CompactCom maps the process data to the industrial network/fieldbus.

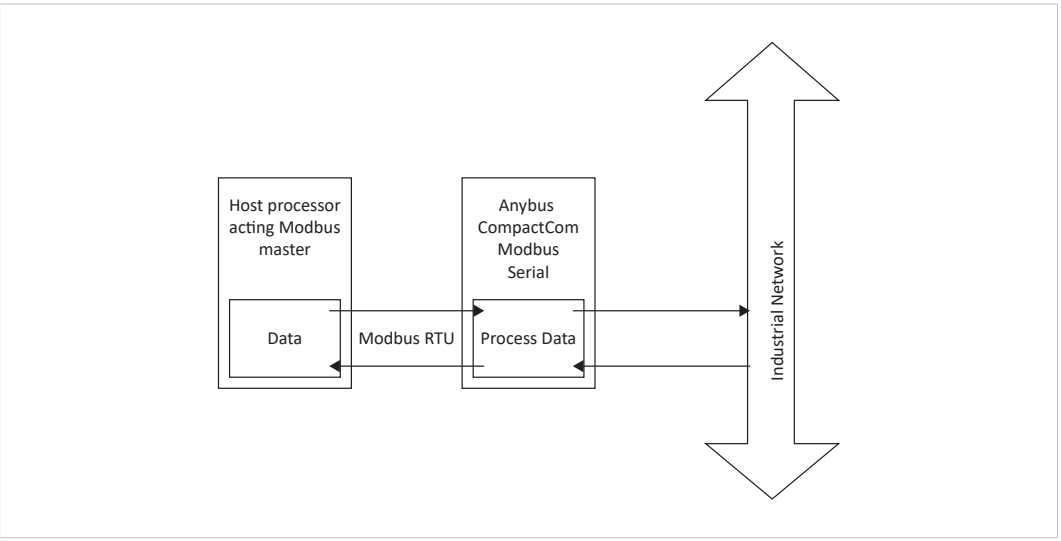

Fig. 1

2.4 About this Manual

This manual includes the following:

1. Instructions on how to indesign the Anybus CompactCom in your product (hardware design).
2. The implemented Modbus Serial application interface and how to use it (host application design).

Network specific hardware implementation and firmware configuration, performed from the network, are described in the Anybus CompactCom 40 Modbus Serial Network Guides, available at www.anybus.com/support.

2.4.1 Hardware Design

- When creating your hardware design, refer to *Hardware Design, p. 8* for descriptions of connectors etc. Please note that some inputs should be strapped to ensure correct behavior.
- Dimensions and other mechanical information is presented in *Mechanical Specification, p. 37*.
- Technical specifications, like temperature limits and electrical specifications, are given in *Technical Specification, p. 35*.

2.4.2 Modbus Serial Application Interface

The Modbus application interface is set up and configured in the same way, independent of which network the data will be sent to/read from. This process, including an example, is described in *Modbus Serial Application Interface, p. 21*.

3 Hardware Design

This section presents general information needed to design the hardware for a product embedding the Anybus CompactCom B40 Modbus Serial. This includes mechanical and electrical specification, connector pinnings etc.

Network specific information is presented in the respective network guides. This includes descriptions of the network interfaces, the optional connector boards and a typical example of how to design the network interface.

- For the host application interface design, see *Host Application Interface, p. 8*
- For further implementation guidelines see *Implementation Guidelines, p. 48*
- For brick dimensions, footprint dimensions and height restrictions, see *Mechanical Specification, p. 37*
- For electrical characteristics and other technical specifications, see *Technical Specification, p. 35*
- How to mount the Anybus CompactCom on the host application PCB is described in *Assembly, p. 41*

3.1 Host Application Interface

The host application connector provides an interface between the host application and the Anybus CompactCom B40 Modbus Serial.

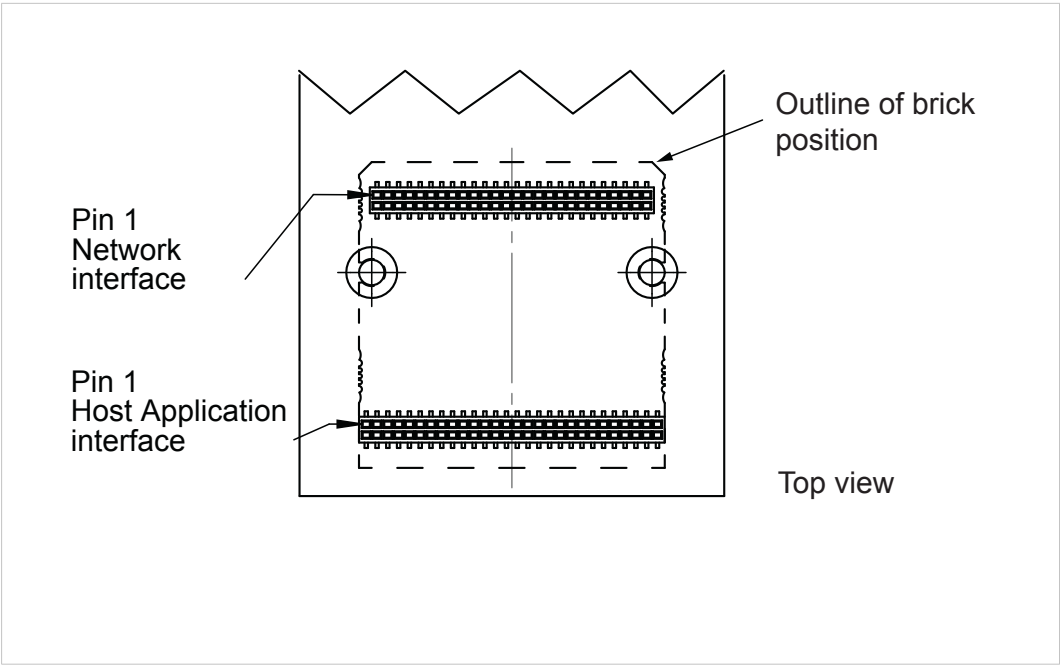

Fig. 2

The connector on the Anybus CompactCom is implemented by a standard 1.27 mm 56 pin header surface mounted to the bottom side of the PCB.

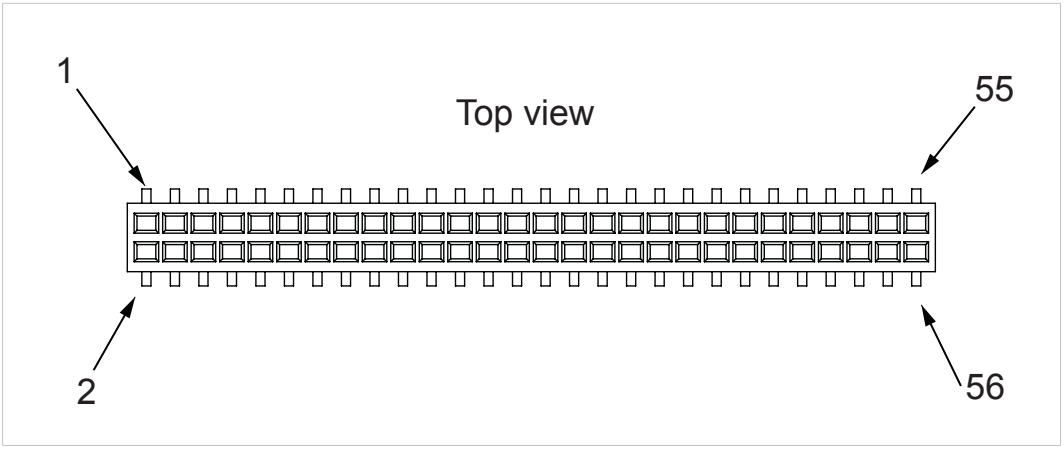

Fig. 3

The picture shows the pinning of the mating connector on the host application as seen from the top.

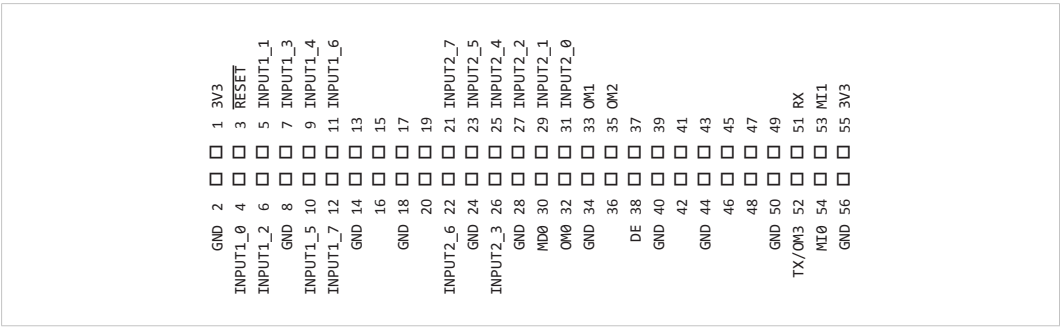

Fig. 4

See *Pin Overview, p. 10* for information on how each pin is used.

3.1.1 Pin Overview

Presented below is an overview of all pins except GND and 3V3.

The pin types of the connector are defined in *Pin Types, p. 4*.

> **!** The pin numbers of the Anybus CompactCom B40 (brick) host application connector are different from those of the Anybus CompactCom M40 (module) host application connector.

Pin	Signal Name	Type	Notes
4	INPUT1_0	I	These inputs are used to set the Modbus address of the Anybus CompactCom Modbus Serial interface See *Communication Settings, p. 14*
5	INPUT1_1	I	
6	INPUT1_2	I	
7	INPUT1_3	I	
9	INPUT1_4	I	
10	INPUT1_5	I	
11	INPUT1_6	I	
12	INPUT1_7	I	
13	(not used)	I	Connect directly to GND
15		I	
16		O, I	
17	(not used)	I	Connect directly to 3V3
19		I	
20	(not used)	O	Leave unconnected
31	INPUT2_0	I	These inputs are used to set baud rate, stop bits, and parity of the Anybus CompactCom Modbus Serial interface See *Communication Settings, p. 14*
29	INPUT2_1	I	
27	INPUT2_2	I	
26	INPUT2_3	I	
25	INPUT2_4	I	
23	INPUT2_5	I	
22	INPUT2_6	I	
21	INPUT2_7	I	
49	(not used)	O	Leave unconnected
48		O	
47		O	
46		O	
45		O	
43		O	
42		O	
41		O	
37	(not used)	I	Connect directly to 3V3
39		I	
36		I	
38	DE	O	Active high Data Enable for RS485 transceiver See *DE (Data Enable), p. 12*

Pin	Signal Name	Type	Notes
51	RX	I	Receive Input for Modbus communication • Direction: Host application -> Anybus CompactCom • Idle state = High
52	TX / OM3	I/O	Transmit Output for Modbus communication • Direction: Anybus CompactCom -> Host application • Idle state = High This pin doubles as OM3 strapping input on Anybus CompactCom. Connect a pull-down resistor on the application for this pin. The pull-down resistor shall have a value between 1 kΩ and 2.2 kΩ.
32	OM0	I	Set [OM2, OM1, OM0] to [0,1,1] For more information see *Settings , p. 12*.
33	OM1	I	
35	OM2	I	
54	MI0	O	Low at power-up and before reset release. See *Settings , p. 12* Please note that this pin can be used as SYNC pin for PROFINET. Must be accessible for PROFINET IRT conformance test.
53	MI1	O	Tied to 3V on the Anybus CompactCom. See *Settings , p. 12*
30	MD0	O	Tied to GND on the Anybus CompactCom. See *Settings , p. 12*
3	$\overline{\text{RESET}}$	I	See *RESET (Reset Input), p. 13*

3.1.2 Power Supply Pins

Pin	Signal Name	Type	Description
2, 8, 14, 18, 24, 28, 34, 40, 44, 50, 56	GND	Power	Ground Power and signal ground reference.
1, 55	3V3	Power	3.3 V power supply.

3.1.3 Settings

Pin	Signal Name	Type	Description
52 35 33 32	OM3 OM2 OM1 OM0	I I I I	Set [OM3, OM2, OM1, OM0] to [0,0,1,1] for Modbus Serial operation. OM3 is used to set operation mode at startup, and requires a pull-down resistor. It is used as TX output after startup. The pull-down resistor shall have a value between 1 kΩ and 2.2 kΩ.
54 53	MI0 MI1	O O	[MI1, MI0] = [1,0] indicates that the Anybus CompactCom connected belongs to the Anybus CompactCom 40-series.
30	MD0	O	**Module Detection** This signal can be used by the host application to determine that an Anybus CompactCom is inserted into the slot, see *Module Detection, p. 12*. The signal is connected directly to GND on the Anybus CompactCom.
51 52	RX TX	I O	Serial Communications Signals, used for Modbus Communication. TX is used as OM3 during setup

Module Detection

This signal is internally connected to GND, and can be used by the host application to detect whether a module is present or not. When connecting an external pull-up resistor, a low signal indicates that a module is present.

If not used, leave this signal unconnected.

3.1.4 DE (Data Enable)

The Anybus CompactCom B40 Modbus Serial supports RS485. This signal is then used as an active high data enable signal for RS485 transceivers.

Pin	Signal Name	Pin Type	Description
38	DE	O	**Data Enable** Active high signal used for RS485 transceiver.

This signal must be pulled to GND on the host application side to prevent spurious transmissions during startup.

3.1.5 RESET (Reset Input)

Pin	Signal Name	Pin Type	Description
3	$\overline{\text{RESET}}$	I	**Reset** Used to reset the module.

The Anybus CompactCom does not feature any internal reset regulation. To establish a reliable interface, the host application is solely responsible for resetting the Anybus CompactCom when the supply voltage is out of the specified range.

It is recommended to connect a pull-down resistor to this pin, to ensure that the input is low if the host processor is not driving the signal.

Power Up

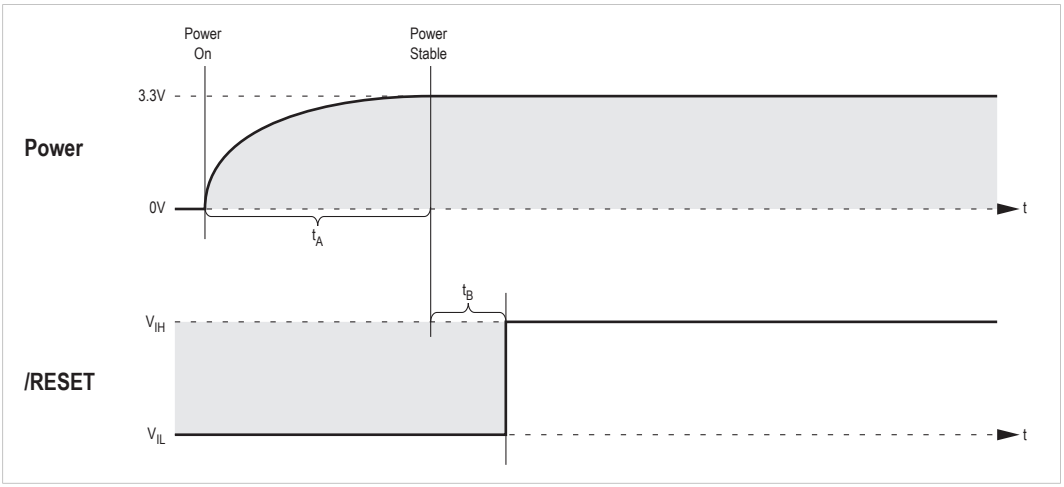

Fig. 5

Powerup time limits are given in the table below:

Symbol	Min.	Max.	Definition
t_A	-	-	Time until the power supply is stable after power-on; the duration depends on the power supply design of the host application and is thus beyond the scope of this document.
t_B	1 ms	-	Safety margin.

Restart

The reset pulse duration must be at least 10 µs in order for the Anybus CompactCom to properly recognize a reset.

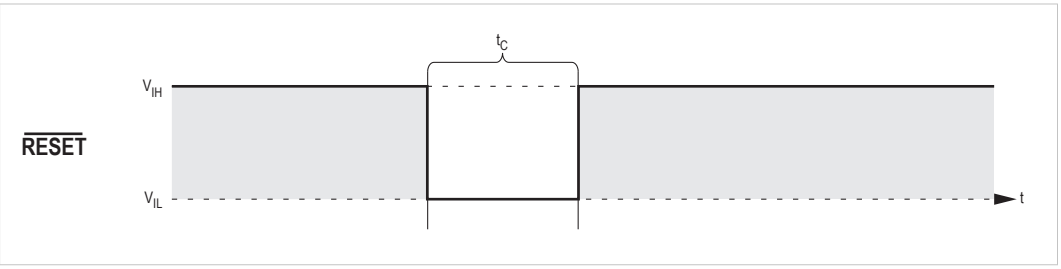

Fig. 6

Symbol	Min.	Max.	Definition
t_C	10 µs	-	Reset pulse width.

3.1.6 Communication Settings

The communication settings are configured by the INPUT1 and INPUT2 input pins in the application interface. Invalid settings result in the Anybus CompactCom entering the state EXCEPTION.

The Modbus address is configured using INPUT1. Valid range is 1 - 247.

Baud rate, stop bits, and parity are configured using INPUT2, as defined in the tables below.

INPUT2 bits 1..0 value	Stop bits, Parity
0	1, Even
1	1, Odd
2	2, None
3	1, None

INPUT2 bits 7..2 value	Baud rate
0 - 2	(reserved)
3	9600
4	19200
5	38400
6	57600
7	(reserved)
8	115200
9	625000
10 - 64	(reserved)

Example

The figure shows INPUT1 and INPUT2 signals directly for Modbus communication. INPUT1 signals are strapped to Modbus address 0x01. INPUT2 signals are set to configure 1 stop bit, even parity and baud rate 19200.

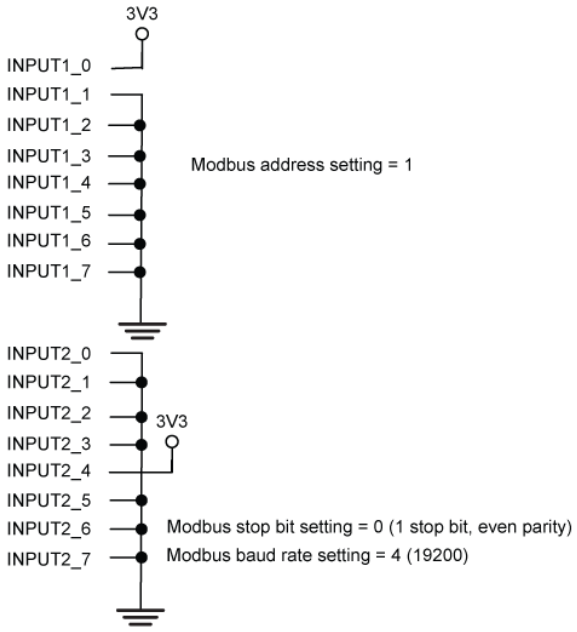

Fig. 7

3.2 Network Interface

The network connector provides network access to the Anybus CompactCom B40 Modbus Serial.

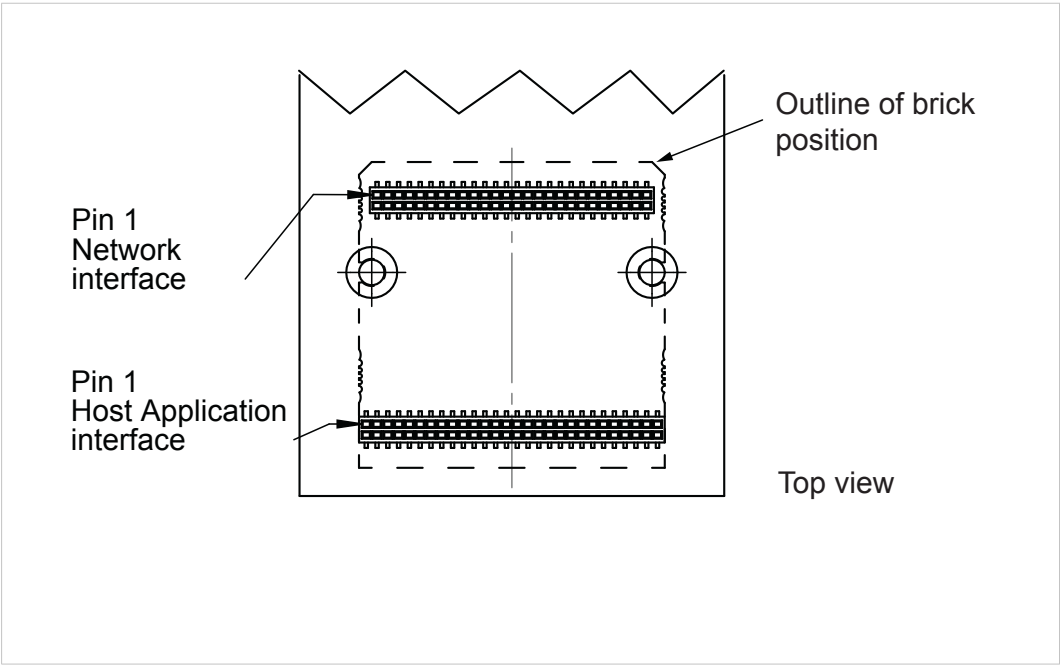

Fig. 8

The signals from the network connector can be directly routed to the (optional) connector board, which carries a network connector(s) identical or similar to the ones on the corresponding Anybus CompactCom M40 module.

Examples on how to design the network access circuitry, when not using the connector board, are presented in *Network Interface Examples, p. 51*.

The brick has a standard 1.27 mm 52 pin header surface mounted to the bottom side of the board as network interface.

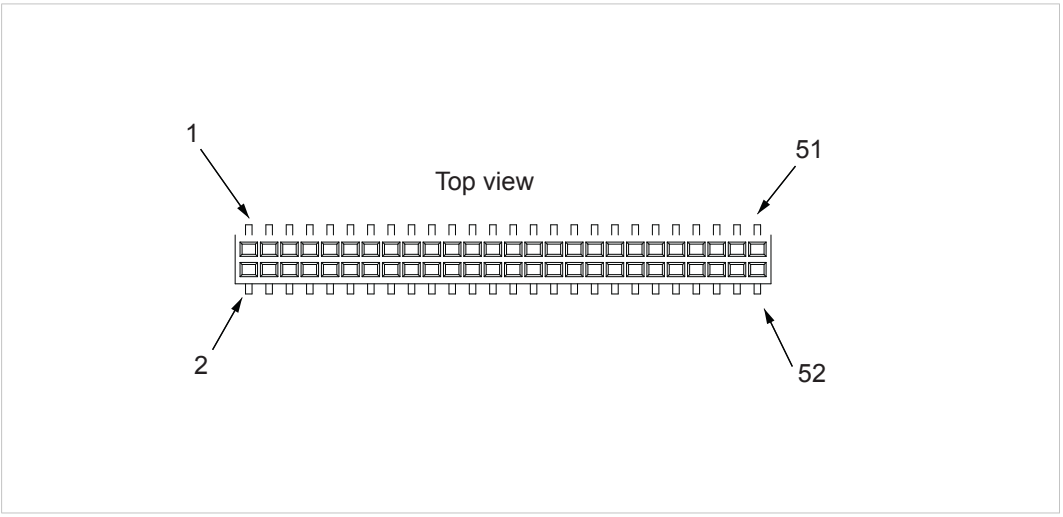

Fig. 9

The pictures show the pinning of the corresponding network connector on the host application board, seen from the top.

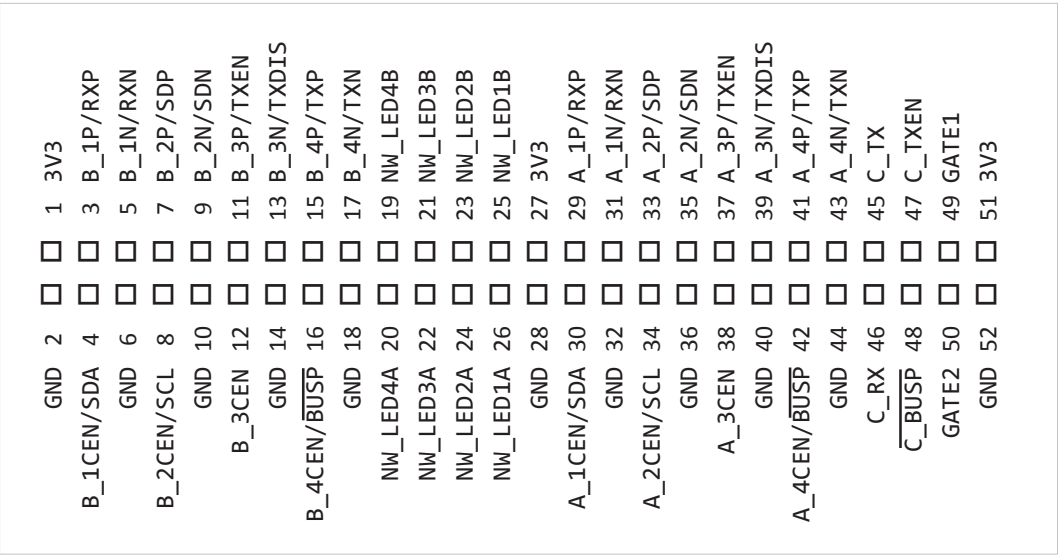

Fig. 10

3.2.1 Overview

Depending on network, the pins have different names and different functionalities. Presented below is an overview of all pins except GND and 3V3. More detailed descriptions of the signals are presented for each available network/fieldbus version in the respective network guides. Please note that at the moment only EtherNet/IP and PROFINET IRT are available.

Pin	Signal Name				
	Ethernet based networks, Copper	**Ethernet based networks, fiber optic**	**DeviceNet**	**PROFIBUS**	**CC-Link**
3	B_1P	B_RXP			
4	B_1CEN	B_SDA			
5	B_1N	B_RXN			
7	B_2P	B_SDP			
8	B_2CEN	B_SCL			
9	B_2N	B_SDN			
11	B_3P	B_TXEN			
12	B_3CEN				
13	B_3N	B_XDIS			
15	B_4P	B_TXP			
16	B_4CEN				
17	B_4N	B_TXN			
19	NW_LED4B	NW_LED4B	NW_LED4B	NW_LED4B	NW_LED4B
20	NW_LED4A	NW_LED4A	NW_LED4A	NW_LED4A	NW_LED4A
21	NW_LED3B	NW_LED3B	NW_LED3B	NW_LED3B	NW_LED3B
22	NW_LED3A	NW_LED3A	NW_LED3A	NW_LED3A	NW_LED3A
23	NW_LED2B	NW_LED2B	NW_LED2B	NW_LED2B	NW_LED2B
24	NW_LED2A	NW_LED2A	NW_LED2A	NW_LED2A	NW_LED2A
25	NW_LED1B	NW_LED1B	NW_LED1B	NW_LED1B	NW_LED1B
26	NW_LED1A	NW_LED1A	NW_LED1A	NW_LED1A	NW_LED1A
29	A_1P	A_RXP			
30	A_1CEN	A_SDA			
31	A_1N	A_RXN			
33	A_2P	A_SDP			

Pin	Signal Name				
	Ethernet based networks, Copper	**Ethernet based networks, fiber optic**	**DeviceNet**	**PROFIBUS**	**CC-Link**
34	A_2CEN	A_SCL			
35	A_2N	A_SDN			
37	A_3P	A_TXEN			
38	A_3CEN				
39	A_3N	A_TXDIS			
41	A_4P	A_TXP			
42	A_4CEN				
43	A_4N	A_TXN			
45			C_TX	C_TX	C_TX
46			C_RX	C_RX	C_RX
47				C_TXEN	C_TXEN
48			C_BUSP_N		
49			GATE1	GATE1	GATE1
50			GATE2	GATE2	GATE2

The LED signals are active high and should be connected to the respective LEDs via a resistor.

The pin types of the connector are defined in *Pin Types, p. 4*.

3.3 Network Configuration Switches

3.3.1 Application Switch Registers

Modbus registers 0x5200 (Application switch 1) and 0x5201 (Application switch 2) can be used for network configuration. Application switch 1 is generally used to configure the node address and application switch 2 to configure the other communication settings.

It is recommended to provide means for the end user to set the values of these registers, using physical switches like DIP switches, rotaries etc. The application is responsible for reading the switch values and then writing them to Modbus registers 0x5200 (Application switch 1) and 0x5201 (Application switch 2).

If the application processor does not have sufficient input pins to connect physical switches, there is a mechanism for I/O expansion that can be used. The switches are then connected to the Anybus CompactCom B40 Modbus Serial instead. For more information see *IO Expansion Mechanism, p. 19*.

Please refer to the applicable network appendix for a description of the contents of the registers.

3.3.2 IO Expansion Mechanism

If the application is to implement physical switches for network configuration, but does not have sufficient I/O pins, the switches can be connected to the Anybus CompactCom instead using the IO Expansion Mechanism. The INPUT1 and INPUT2 pins can then be used both to strap the Modbus communication settings and to connect the physical switches.

The Anybus CompactCom will sample the INPUT1 and INPUT2 pins to determine the Modbus communication settings before Modbus communication is performed. When the first Modbus telegram has been exchanged, the DE signal can be used to alter the INPUT1 and INPUT2 signals to sample the physical switches. The application processor can then read the INPUT1 and INPUT2 pins in Modbus register 0x0FFD (Anybus switch status) and write back the values to Modbus registers 0x5200 (Application switch 1) and 0x5201 (Application switch 2).

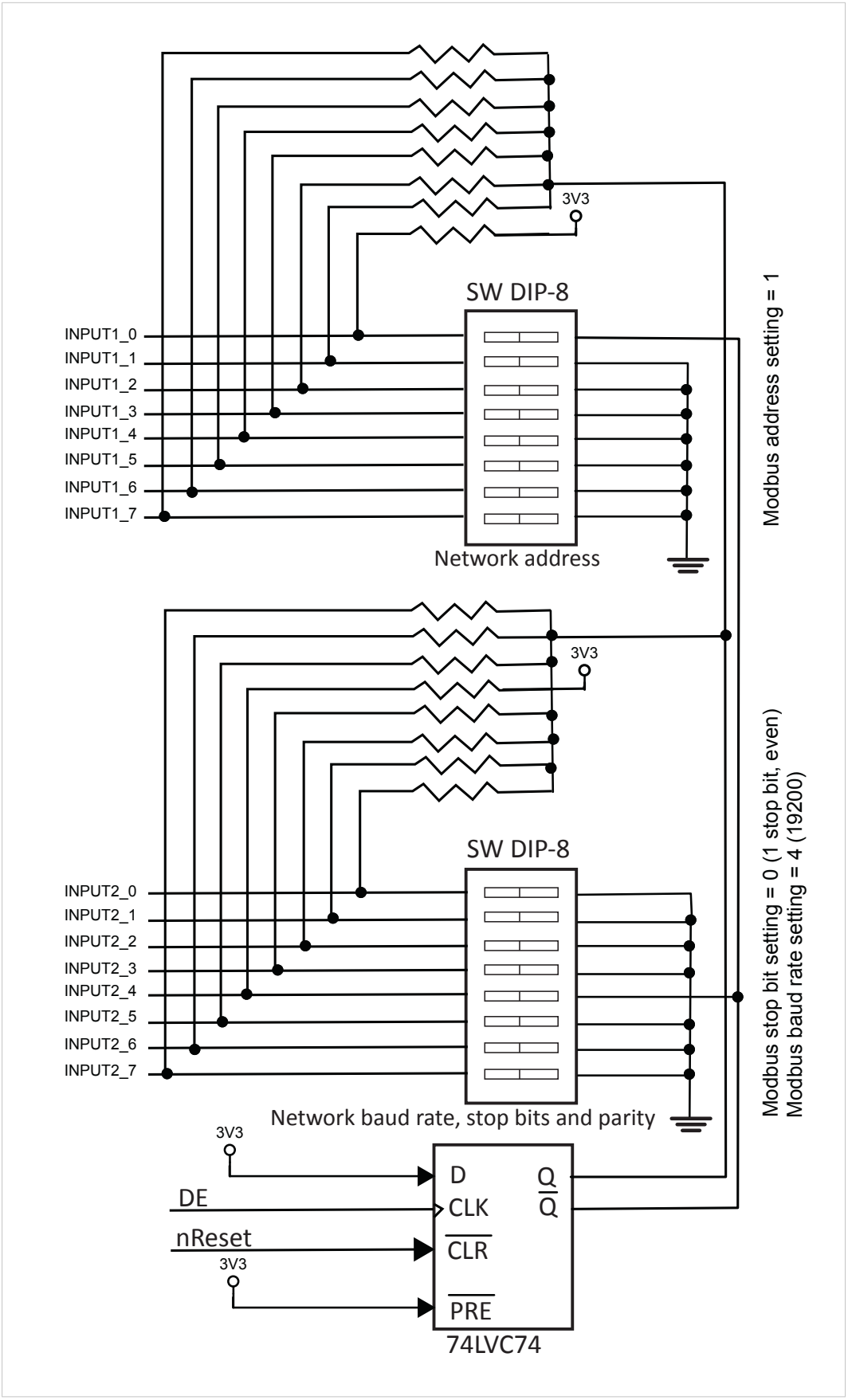
3V3
SW DIP-8
INPUT1_0
INPUT1_1
INPUT1_2
INPUT1_3
INPUT1_4
INPUT1_5
INPUT1_6
INPUT1_7
Network address
Modbus address setting = 1
3V3
SW DIP-8
INPUT2_0
INPUT2_1
INPUT2_2
INPUT2_3
INPUT2_4
INPUT2_5
INPUT2_6
INPUT2_7
Network baud rate, stop bits and parity
Modbus stop bit setting = 0 (1 stop bit, even)
Modbus baud rate setting = 4 (19200)
3V3
DE
nReset
3V3
D
CLK
CLR
PRE
Q
Q
74LVC74

Fig. 11

4 Modbus Serial Application Interface

The host application interface of the product consists of a Modbus Serial interface, which implements a reduced set of Modbus registers and commands. The Anybus CompactCom acts as a Modbus RTU slave.

This section covers the software implementation for the Modbus Serial Application Interface. Further on in this section there are flowcharts that describe the complete flow from when the device is powered on until it is running. The flowcharts also cover reset and what happens if the communication halts. The first flowchart describes the main flow, including setup, and the second describes what happens when the application is running (step 7 in the first flowchart).

4.1 Hardware Prerequisites

All items in the following list must be finished before continuing with the software part:

- Pins OM[3..0] are strapped to [0,0,1,1]
- INPUT1 is strapped to the correct Modbus address
- INPUT2 is strapped to correct baud rate and parity settings
- Rx and Tx are connected to a UART on the host application processor running Modbus RTU
- The RESET signal must also be under software control

4.2 Host Application Main Software Flowchart

The flowchart in this section describes how to set up the main communication of the host application.

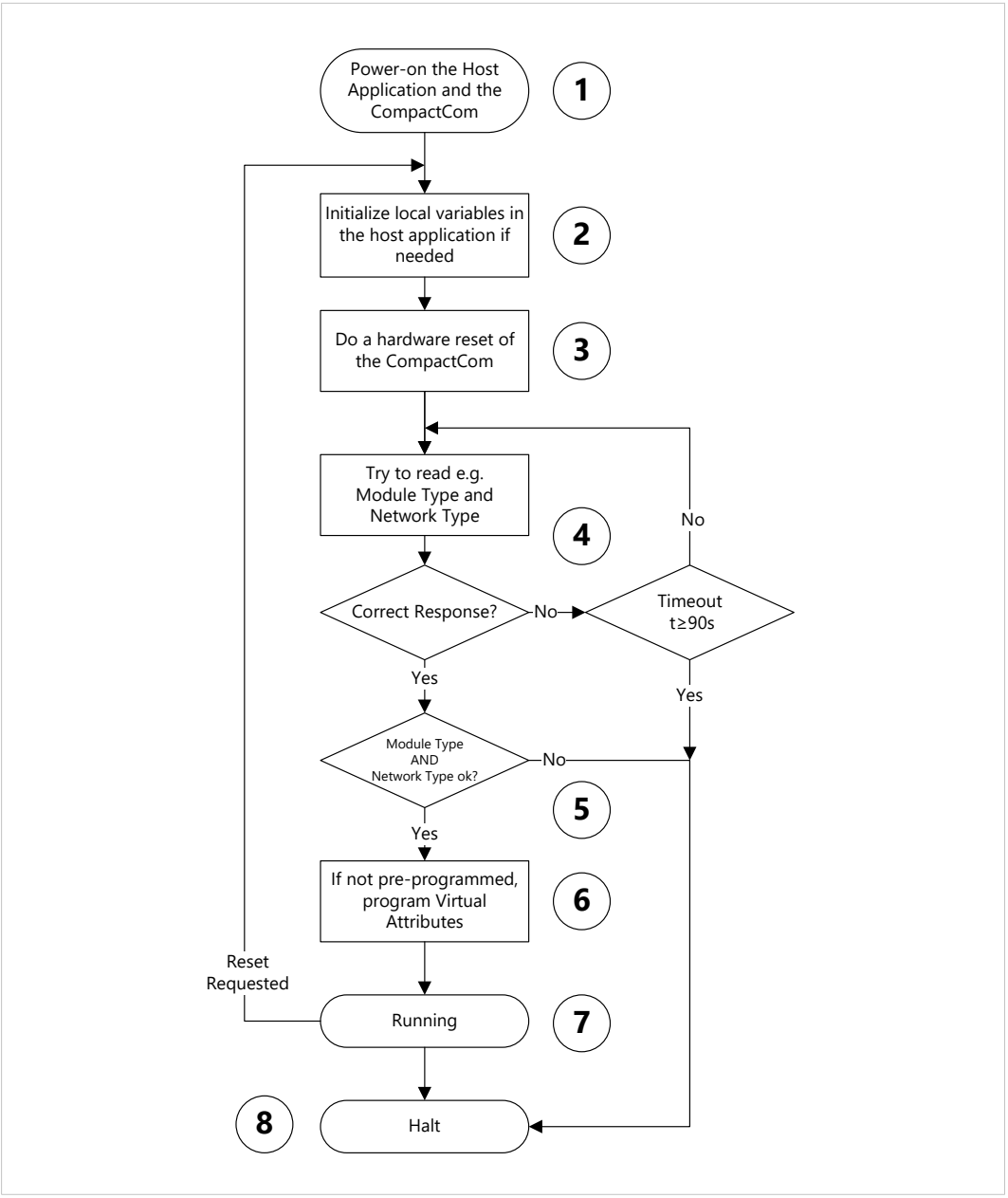

Fig. 12

The numbers in the list refer to the numbers in the figure.

1. Power-on the Host Application and the Anybus CompactCom. Keep the Anybus CompactCom in reset (the reset line in low state) until otherwise stated.
2. Initialize internal variables in the host application, if needed.
3. Release hardware reset of the Anybus CompactCom according to *RESET (Reset Input), p. 13*. For a proper reset, it is important that all requirements are fulfilled.

4. Try to read from the Anybus CompactCom to determine if it is ready for operation. When a correct response is received the Anybus CompactCom is ready for communication. E.g. send a Read Input Registers command to Modbus registers 0x5003 and 0x5004. The normal startup time is max 1.5 s. If the Anybus CompactCom does not answer within 1.5 s after reset, it is recommended to give a warning to the user. A startup time longer than 1.5 s indicates that a firmware update is in progress. If the Anybus CompactCom does not answer within 90 s, an error should be indicated.

   Example, using the Modbus function Read Input Registers:

   Request

	Value	**Note**
Modbus Address	0xXX	
Function Code	0x04	Read Input Registers
Starting address	0x50	Reading the Module Type register and the Network Type register
	0x03	
Quantity of Input registers	0x00	
	0x02	
CRC	0xXX	CRC-16
	0xXX	

   Expected response

	Value	**Note**
Modbus Address	0xXX	
Function Code	0x04	
Byte Count	0x04	
Input Registers	0x04	Value of the Module Type register (0x0403 expected)
	0x03	
	0xXX	Value of the Network Type register
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

5. Check that the Module Type is 0x0403 (Anybus CompactCom 40) and that the Network Type is the expected type. If both are correct, save them for later.
6. If you need to change the identity information of the Anybus CompactCom, you will have to use “Virtual Attributes”. The virtual attributes can be preprogrammed in the Anybus CompactCom by using the Anybus CompactCom starter Kit, or by using either the Anybus CompactCom specific Modbus function code 70 or the Virtual Attributes Manager software. For detailed information, see the respective network guides.
7. The application will be in this state when everything is working as expected. For detailed information about this state, continue to the next section.

   If a reset is requested from the network, restart the application from step 2.
8. E.g. a timeout when communicating with the Anybus CompactCom or an erroneous Module Type. Halt the application and indicate an error to the user.

   If communication with the Anybus CompactCom is lost during normal operation, it is also recommended to issue a hardware reset to the Anybus CompactCom, to stop it from further network participation.

4.3 Running State Flowchart

The configuration of and the data exchange with the Anybus CompactCom takes place in the RUNNING state. The host application will remain in this state until either a reset is requested from the network or if something goes wrong.

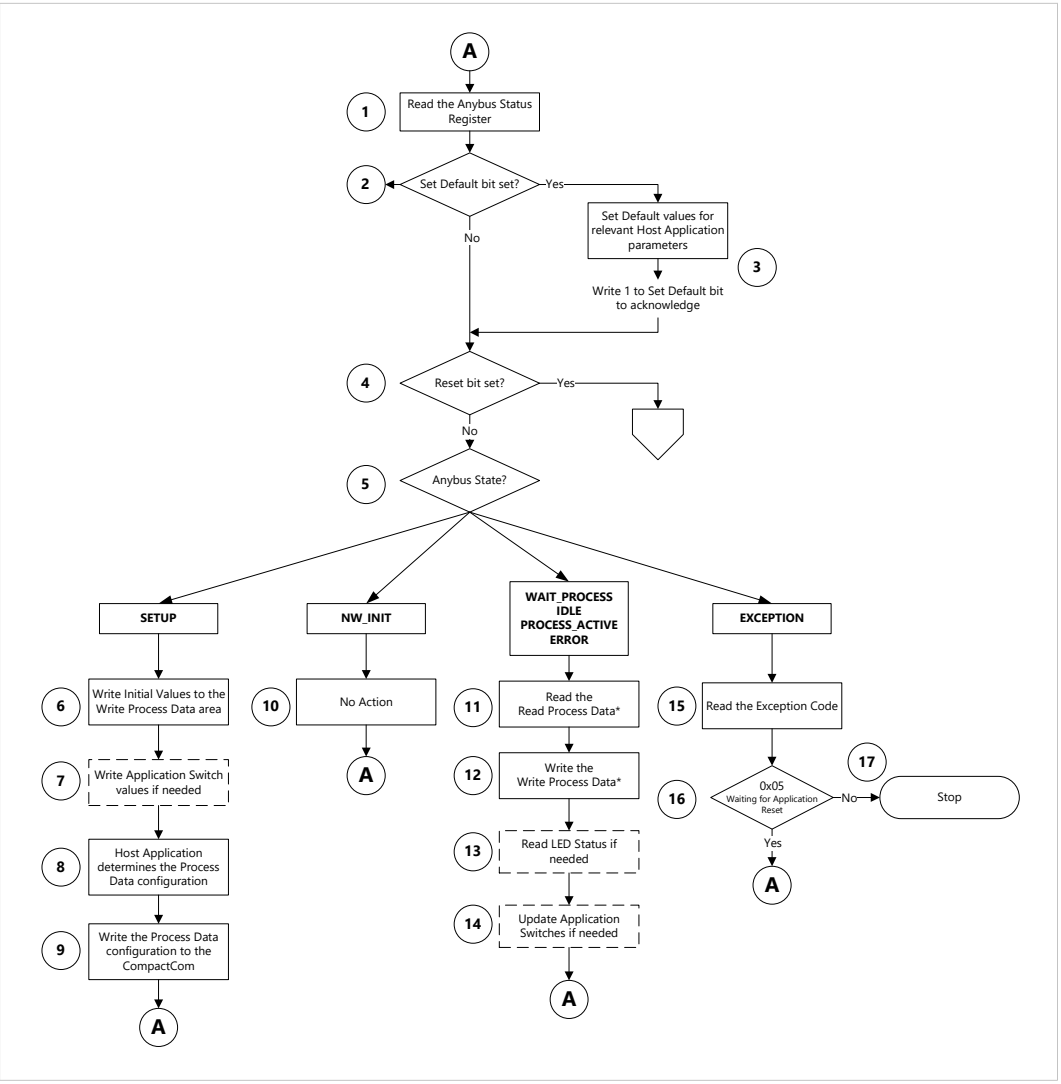

Fig. 13

The numbers in the list refer to the numbers in the figure.

1. Read the Anybus Status Register (Modbus register 0x0FFF). This register holds information about the Anybus State and if a Reset or Set Default has been requested from the network.

   Example, using the Modbus function Read Input Registers:

   Request

	Value	Note
Modbus Address	0xXX	
Function Code	0x04	Read Input Registers
Starting address	0x0F	Reading the Anybus Status Register
	0xFF	
Quantity of Input registers	0x00	
	0x01	
CRC	0xXX	CRC-16
	0xXX	

   Expected response

	Value	Note
Modbus Address	0xXX	
Function Code	0x04	
Byte Count	0x02	
Input Registers	0xXX	Value of the Anybus Status Register
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

2. If bit 14 in the Anybus Status Register is set, it is requested from the network to set default values of selected parameters.
3. Parameters handled by the Anybus CompactCom are automatically set to default values. Parameters in the host application, that need default values, should also be set. Acknowledge the Set Default request by writing 1 to bit 14.

It is not necessary to reset and re-initialize the Anybus CompactCom because of a Set Default request.

4. If bit 15 in the Anybus Status Register is set, a Reset is requested from the network. Reset and re-initialize the Anybus CompactCom. Restart from step 2 in the main software flowchart, see *Host Application Main Software Flowchart, p. 21*
5. Determine which state the Anybus CompactCom is in by checking bit 0-2 in the Anybus Status Register. The Anybus state machine shall be regarded as a Moore machine, i.e. the host application is not required to keep track of all state transitions, however it is expected to perform certain tasks in each state. For more information, see *The Anybus State Machine, p. 46*.

6. Write Initial values to the Write Process Data area (Modbus registers starting with address 0x0000).

Example, using the Modbus Function Write Multiple Registers

Request

	Value	Note
Modbus Address	0xXX	
Function Code	0x10	Write Multiple Registers
Starting Address	0x00	Start of Write Process Data
	0x00	
Quantity of Registers	0xXX	Number of Write Process Data registers
	0xXX	
Byte Count	0xXX	Number of Write Process Data registers x2
Registers Value	0xXX	Write Process Data Values
	...	
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

Expected response

	Value	Note
Modbus Address	0xXX	
Function Code	0x10	Write Multiple Registers
Starting Address	0x00	Start of Write Process Data
	0x00	
Quantity of Registers	0xXX	Number of Write Process Data registers
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

7. If needed, write Application Switch values to Modbus register 0x5200-0x5201. How to use the Application Switches is network specific. Consult the applicable network guide for information.

   Example, using the Modbus Function Write Multiple Registers

   Request

	Value	**Note**
Modbus Address	0xXX	
Function Code	0x10	Write Multiple Registers
Starting Address	0x52	
	0x00	
Quantity of Registers	0x00	Writing 2 registers
	0x02	
Byte Count	0x04	Writing 4 bytes
Registers Value	0xXX	Application Switch 1 Value
	0xXX	
	0xXX	Application Switch 2 Value
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

   Expected response

	Value	Note
Modbus Address	0xXX	
Function Code	0x10	Write Multiple Registers
Starting Address	0x52	
	0x00	
Quantity of Registers	0x00	Writing 2 registers
	0x02	
CRC	0xXX	CRC-16
	0xXX	

8. Determine the needed settings for Data Type, Offline Action, Write Process Data Size, and Read Process Data Size.

9. Write Data Type, Offline Action, Number of Write Parameters, and Number of Read Parameters to Modbus registers 0x5100-0x5103. The initialization is now finished.

   Example, using the Modbus Function Write Multiple Registers

   Request

	Value	Note
Modbus Address	0xXX	
Function Code	0x10	Write Multiple Registers
Starting Address	0x51	
	0x00	
Quantity of Registers	0x00	Writing 4 registers
	0x04	
Byte Count	0x08	Writing 8 bytes
Registers Value	0xXX	Data Type
	0xXX	
	0xXX	Offline Action
	0xXX	
	0xXX	Number of Write Parameters
	0xXX	
	0xXX	Number of Read Parameters
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

   Expected response

	Value	Note
Modbus Address	0xXX	
Function Code	0x10	Write Multiple Registers
Starting Address	0x51	
	0x00	
Quantity of Registers	0x00	Writing 4 registers
	0x04	
CRC	0xXX	CRC-16
	0xXX	

10. The host application is not involved in any way when in the NW_INIT state. The only action is to keep the Write Process Data updated.
11. Read the Read Process Data, Modbus registers starting with address 0x1000. The Read Process Data is only valid in the PROCESS_ACTIVE state.
12. Write the Write Process Data, Modbus registers starting with address 0x0000.

13. Read the LED status to use for external LED indicators, Modbus register 0x0FFE.

Example, using the Modbus Function Read/Write Multiple Registers

In this example steps 11, 12, and 13 are performed in one command. Also the status of the Anybus Status Register and the Anybus Switches are read in the same command.

Request

	Value	Note
Modbus Address	0xXX	
Function Code	0x17	Read/Write Multiple Registers
Read Starting Address	0x0F	
	0xFD	
Quantity to Read	0xXX	3+Number of Read Process Data registers
	0xXX	
Write Starting Address	0x00	
	0x00	
Quantity to Write	0xXX	Number of Write Process Data registers
	0xXX	
Write Byte Count	0xXX	Number of Write Process Data register x2
Write Registers Value	0xXX	Write Process Data Values
	…	
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

Expected response

	Value	Note
Modbus Address	0xXX	
Function Code	0x17	Read/Write Multiple Registers
Byte Count	0xXX	
Read Registers Value	0xXX	Anybus Switch Status
	0xXX	
	0xXX	LED Status
	0xXX	
	0xXX	Anybus Status
	0xXX	
	0xXX	Read Process Data Values
	…	
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

14. If the application switches are changed, update the Application Switch registers, Modbus registers 0x5200-0x5201.

15. The Anybus CompactCom has entered the EXCEPTION state. Read the Exception Code in Modbus register 0x5005.

    Example, using the Modbus Function Read Input Registers

    Request

	Value	Note
Modbus Address	0xXX	
Function Code	0x04	Read Input Registers
Starting Address	0x50	
	0x05	
Quantity of Input Registers	0x00	1 register
	0x01	
CRC	0xXX	CRC-16
	0xXX	

    Expected write response

	Value	Note
Modbus Address	0xXX	
Function Code	0x04	Read Input Registers
Byte Count	0x02	
Input Registers	0xXX	Exception Code
	0xXX	
CRC	0xXX	CRC-16
	0xXX	

16. If the exception code is “Waiting for Application Reset”, a reset has been requested from the network. Continue to A and act on the Reset bit.

17. For all other exception codes, indicate error to the user. A reset and a reinitialization of the Anybus CompactCom is needed to leave the EXCEPTION state.

4.4 Modbus Register Map

The Modbus register space is designed to allow easy and efficient communication using a single cyclic read/write Modbus command.

Register#	Name	Access	Description
0x0000 -	Write process data	Read/ Write	For data consistency reasons the Write process data content is not forwarded to the network until the last byte of it has been written
0x0FFD	Anybus Switch status	Read only	Value of INPUT1 and INPUT2 pins. LSB: INPUT1 MSB: INPUT2
0x0FFE	LED status	Read only	Bit field holding the current state of the network status LEDs as follows: Bit: Contents: b0: LED1A status (0=OFF, 1=ON) b1: LED1B status (0=OFF, 1=ON) b2: LED2A status (0=OFF, 1=ON) b3: LED2B status (0=OFF, 1=ON) b4: LED3A status (0=OFF, 1=ON) b5: LED3B status (0=OFF, 1=ON) b6: LED4A status (0=OFF, 1=ON) b7: LED4B status (0=OFF, 1=ON) b8 - b15 (reserved)
0x0FFF	Anybus status	Read/ Write	Anybus status register Bit(s) – Description 0 - 2: Anybus state (See *The Anybus State Machine, p. 46*) 0. Setup 1: NW_INIT 2: WAIT_PROCESS 3: IDLE 4: PROCESS_ACTIVE 5: ERROR 6: (reserved) 7: EXCEPTION 3: SUP 0: Module is not supervised 1: Module is supervised by another network device The SUP bit indicates the overall status of the network communication. For example, on CIP, this bit indicates that the master has a connection towards the module. Exactly how this functionality should be handled, the offered level of security, and how to correctly activate it is network specific and often depends on external circumstances, e.g. configuration of the network as well as other network devices. Whether use of the SUP-bit is appropriate must therefore be decided for each specific application and network. 4 -13: (reserved) 14: Set Default 0: No action 1: Application is requested to set “factory default” to all parameters/settings. Application shall acknowledge this by writing 1 to this bit to clear it. 15: Reset 0: No action 1: Application is requested to reset the Anybus CompactCom
0x1000 -	Read process data	Read only	For data consistency reasons the content is only updated when register 0x1000 is read
0x5003	Module type	Read only	Anybus CompactCom 40 (0x0403)
0x5004	Network type	Read only	0x0087: EtherCAT 0x0089: PROFINET IRT 0x009B: EtherNet/IP
0x5005	Anybus Exception Code	Read only	0x00: No exception

Register#	Name	Access	Description
			0x01: Application timeout 0x02: Invalid device address 0x03: Invalid communication settings 0x04. Major unrecoverable application event 0x05: Waiting for application reset 0x06: Invalid process data configuration 0x07: Invalid application response 0x08: NVS memory checksum error 0x09: Functional Safety Module error 0x0A: Insufficient application implementation 0x0B: Missing serial number 0x0C: File system is corrupt
0x5100	Data type	Read/ Write	Data type of the Write and Read Parameters . UINT8: 0x0004 (Default) UINT16: 0x0005 Read/Write in SETUP state. Read only in other states.
0x5101	Offline action	Read/ Write	Action taken on the Read process data, when the Anybus CompactCom is not in PROCESS_ACTIVE state. 0x0000: Clear “Read Process data area” 0x0001: No action (Default) Read/Write in SETUP state. Read only in other states.
0x5102	Number of Write Parameters	Read/ Write	Number of Write Parameters. If the value written is bigger than module can handle, the value will automatically be adjusted to max possible value (according to network) after the SETUP state. Read/Write in SETUP state. Read only in other states.
0x5103	Number of Read Parameters	Read/ Write	Number of Read Parameters. If the value written is bigger than module can handle, the value will automatically be adjusted to max possible value (according to network) after the SETUP state. Read/Write in SETUP state. Read only in other states.
0x5200	Application switch 1	Read/ Write	This register can be used for the node address, see the respective Network Guides. Value written to this register may be defined by DIP switches on the application. See *Network Configuration Switches, p. 19*.
0x5201	Application switch 2	Read/ Write	This register can be used for communication settings, see the respective Network Guides.. Value written to this register may be defined by DIP switches on the application. See *Network Configuration Switches, p. 19*.

4.5 Supported Modbus Commands

#	Name	Description
3	Read multiple registers	All mapped to the same register space.
4	Read input registers	
6	Write single register	
16	Write multiple registers	
23	Read/Write multiple registers	
70	Anybus command	Custom Modbus command, see below

4.5.1 Error Handling

- Modbus reads, addressing undefined registers, return zeroes.
- Modbus writes, addressing undefined registers, do not return any error.
- Modbus writes containing invalid data to registers do not return any error.
- Modbus writes, addressing read-only registers, do not return any error.

4.5.2 Handling of 8-bit and 16-bit Data Types

The input process data and output process data are handled in different ways depending on the setting of parameter 'Data Type' (Modbus address 0x5100). This parameter determines the data type of each unit of the process data, UINT8 or UINT16. It will also set the data type correspondingly for networks that have support for data type for the process data.

The representation of the data is different depending on if the network is a Big-Endian network or a Little-Endian network.

The examples below show how the data is represented on a Big-Endian network compared to a Little-Endian network in UINT8 mode and UINT6 mode.

Example Data Type UINT8

When data type UINT8 is configured, the Anybus CompactCom will map the first element to the low half of the first Modbus register, and the second element to the high half of the first Modbus register and so on.

Modbus Register	Value	PROFINET Big-Endian		EtherNet/IP Little-Endian	
		Byte	Value	Byte	Value
0x0000	0xBBAA	1	0xAA	1	0xAA
		2	0xBB	2	0xBB
0x0001	0xDDCC	3	0xCC	3	0xCC
		4	0xDD	4	0xDD
0x0002	0xFFEE	5	0xEE	5	0xEE
		6	0xFF	6	0xFF

Example Data Type UINT16

When data type UINT16 is configured, the Anybus CompactCom will perform an automatic endian conversion of the Big-Endian Modbus registers to the network specific byte order.

Modbus Register	Value	PROFINET Big-Endian		EtherNet/IP Little-Endian	
		Byte	Value	Byte	Value
0x0000	0xBBAA	1	0xBB	1	0xAA
		2	0xAA	2	0xBB
0x0001	0xDDCC	3	0xDD	3	0xCC
		4	0xCC	4	0xDD
0x0002	0xFFEE	5	0xFF	5	0xEE
		6	0xEE	6	0xFF

This page intentionally left blank

A Technical Specification

A.1 Environmental

A.1.1 Operating

-40 to 85° C (-40 to 185° F)

A.1.2 Storage

-40 to 85°C (-40 to 185° F)

A.1.3 Humidity

5 to 95% non-condensing

A.2 Shock and Vibration

A.2.1 Shock

The Anybus CompactCom B40 Modbus Serial is tested according to IEC 68–2–27

- half-sine 30 g, 11 ms, 3 positive and 3 negative shocks in each of three mutually perpendicular directions
- half-sine 50 g, 11 ms, 3 positive and 3 negative shocks in each of three mutually perpendicular directions

Connector boards/interface cards are tested for 30 g.

A.2.2 Sinusoidal Vibration

The Anybus CompactCom B40 Modbus Serial is tested according to IEC 68–2–6

Frequency range:	10–500 Hz
Amplitude 10–49 Hz:	0.35 mm
Acceleration 50–500 Hz	5 g
Sweep rate:	1 oct/min
	10 double sweep in each of the three mutually perpendicular directions

A.3 Electrical Characteristics

> **!** Failure to follow the requirements may lead to permanent hardware damage

It is recommended for Anybus CompactCom B40 Modbus Serial users to make sure that each signal controlling the Anybus CompactCom B40 Modbus Serial has a drive strength enough to fulfill level and timing constraints even if the signal is loaded with 20 pF in parallel with 2.2 kΩ to GND or 3V3.

A.3.1 Operating Conditions

Symbol	Parameter	Pin Types	Conditions	Min.	Typ.	Max.	Unit
3V3	Supply Voltage (DC)			3.15	3.30	3.45	V
	Ripple (AC)			-	-	± 100	mV
GND	Ground reference		-	0.00	0.00	0.00	V
I_{IN}	Current consumption (also including network interfaces and network status LEDs)		Class A	-	-	250	mA
			Class B	-	-	500	mA
		PWR	Class C	-	-	1000	mA
V_{IH}	Input High Voltage	I, BI	-	2.0	-	3.45	V
V_{IL}	Input Low Voltage			-0.3	-	0.8	V
I_{OH}	Current, Output High	O, BI	-	-8.0	-	8.0	mA
I_{OL}	Current, Output Low						
V_{OH}	Output High Voltage		I_{OH} = -4mA	2.4	-	-	V
V_{OL}	Output Low Voltage		I_{OL} = 4mA	-	-	0.4	V
I_{OH} (NW_ LEDx)	Output Current , network LEDs	O				20	mA

I= Input, CMOS (3.3V)

O= Output, CMOS (3.3V)

BI= Bidirectional, Tristate

PWR= Power supply inputs

A.4 Regulatory Compliance

EMC Compliance (CE)

Since the Anybus CompactCom is considered a component for embedded applications it cannot be CE-marked as an end product.

However the Anybus CompactCom 40 family is pre-compliance tested in a typical installation providing that all modules are conforming to the EMC directive in this installation.

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

Emission: EN61000-6-4	EN55016-2-3 Radiated emission
	EN55022 Conducted emission
Immunity: EN61000-6-2	EN61000-4-2 Electrostatic discharge
	EN61000-4-3 Radiated immunity
	EN61000-4-4 Fast transients/burst
	EN61000-4-5 Surge immunity
	EN61000-4-6 Conducted immunity

Since all Anybus CompactCom B40 Modbus Serial modules have been evaluated according to the EMC directive through above standards, this serves as a base for our customers when certifying Anybus CompactCom B40 Modbus Serial based products.

B Mechanical Specification

> ! 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.

> ! This equipment contains parts that can be damaged by electrostatic discharge (ESD). Use ESD prevention measures to avoid damage.

All dimensions are in millimeters, tolerance ±0.10 mm, unless otherwise stated.

B.1 Anybus CompactCom B40 Modbus Serial

The dimensions for the Anybus CompactCom B40 Modbus Serial are given in the picture below.

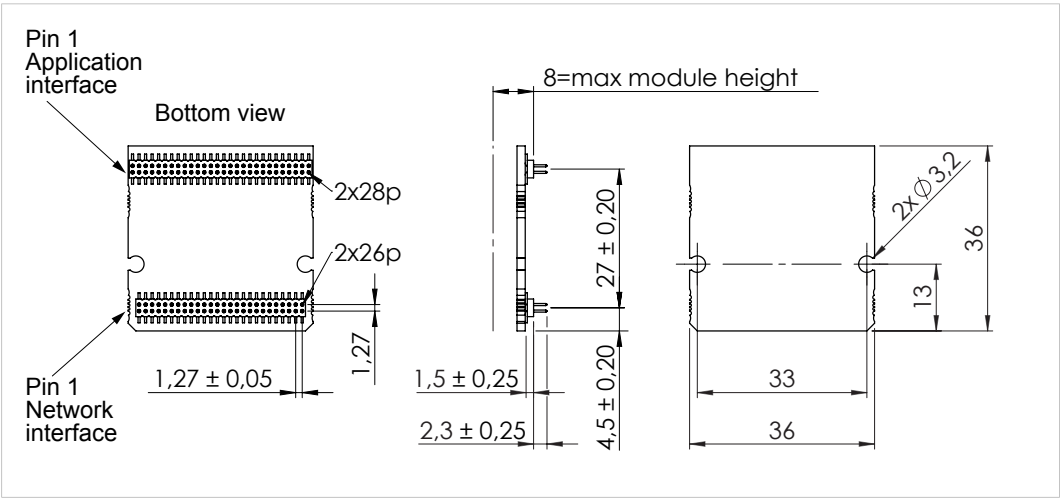

Fig. 14

B.2 Footprint

The Anybus CompactCom is connected to the host application board through the host application interface connector and a network interface connector. The footprint for the Anybus CompactCom is shown in the picture below.

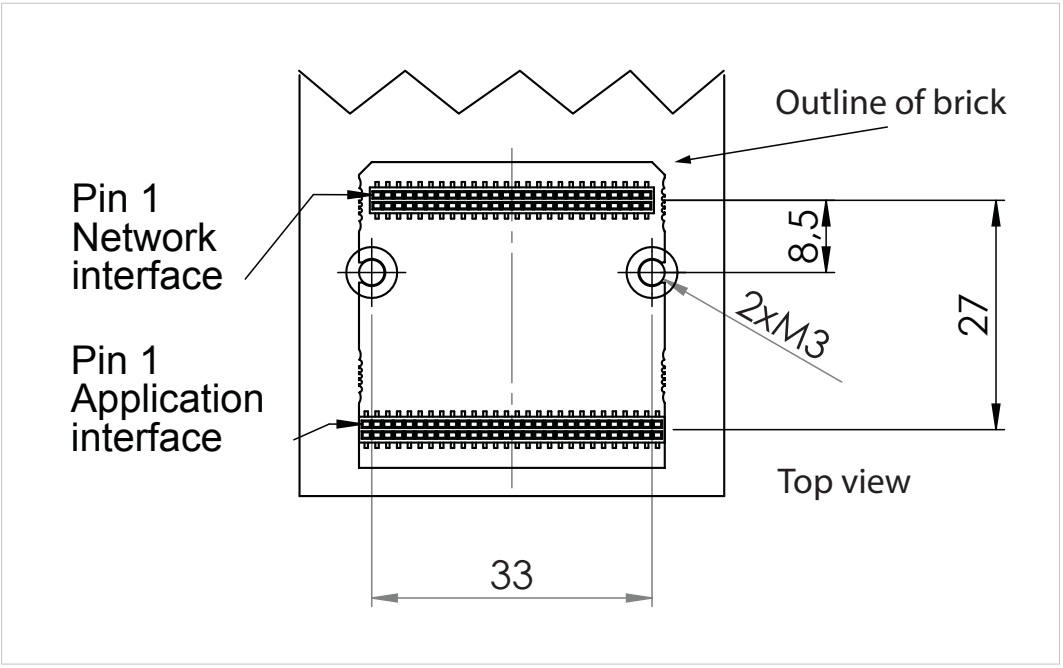

Fig. 15

B.3 Height Restrictions

All dimensions are in millimeters

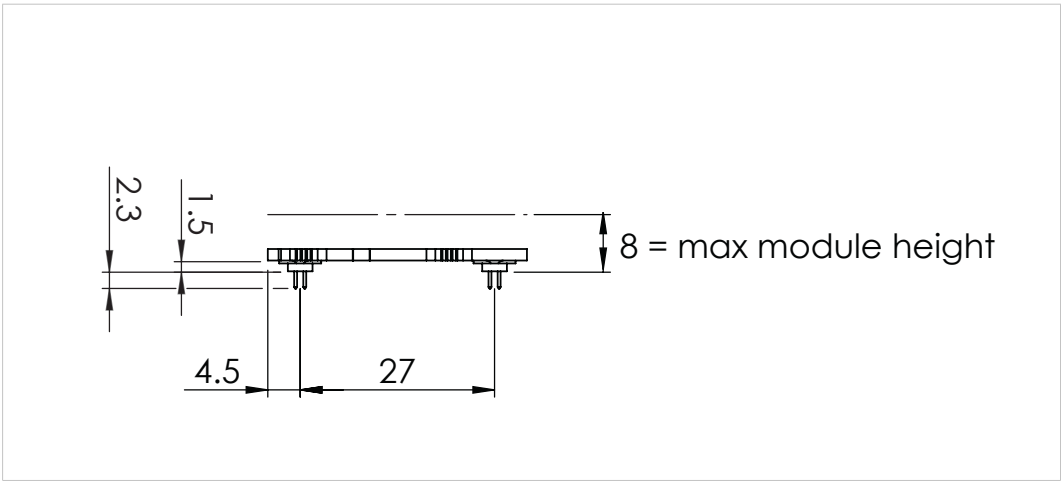

Fig. 16

The maximum height occupied by onboard components of the Anybus module is 8 mm. To ensure isolation, it is recommended to add an additional 2.5 mm on top of these dimensions.

B.4 Network Connector Boards (Optional)

B.4.1 Connector Board for Copper Based Ethernet

The connector board for the copper based Ethernet network interfaces carries two RJ45 connectors.

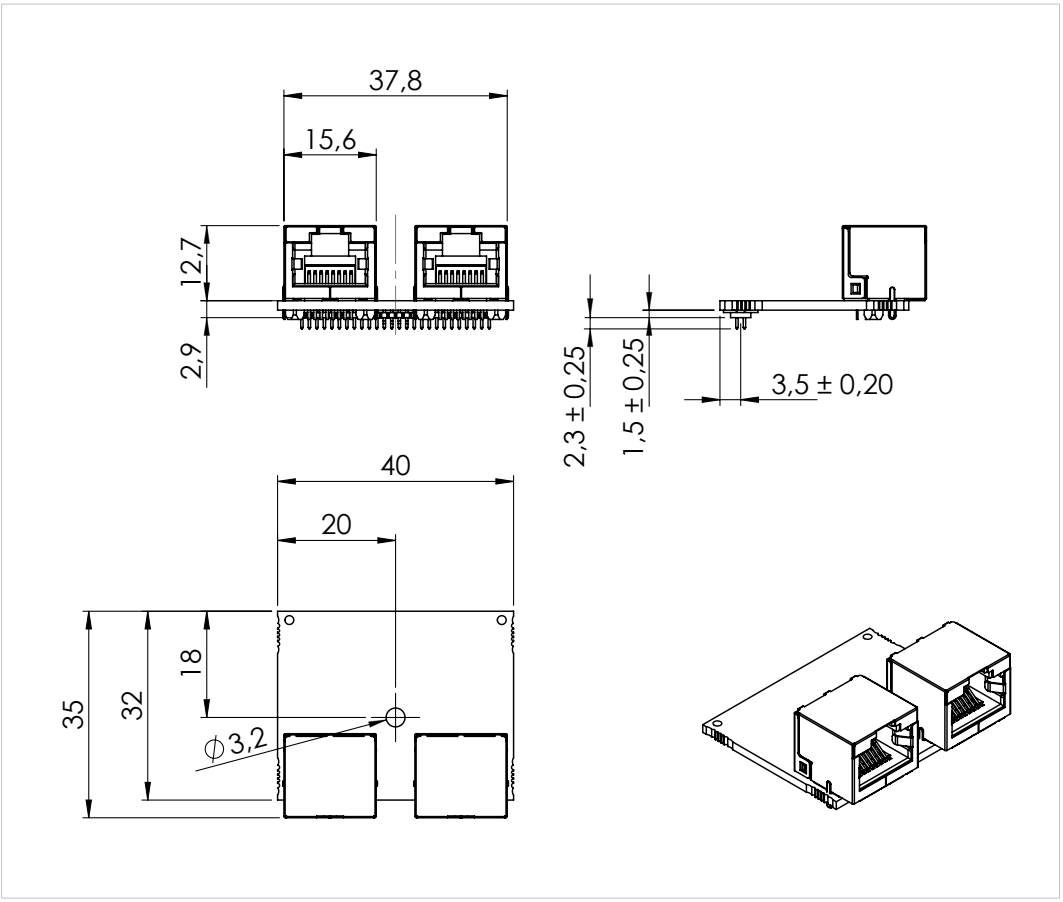

Fig. 17

B.5 Connector Board for Fiber Optic Ethernet

The connector board for the Fiber Optic Ethernet network interface carries two fibre optic transceivers.

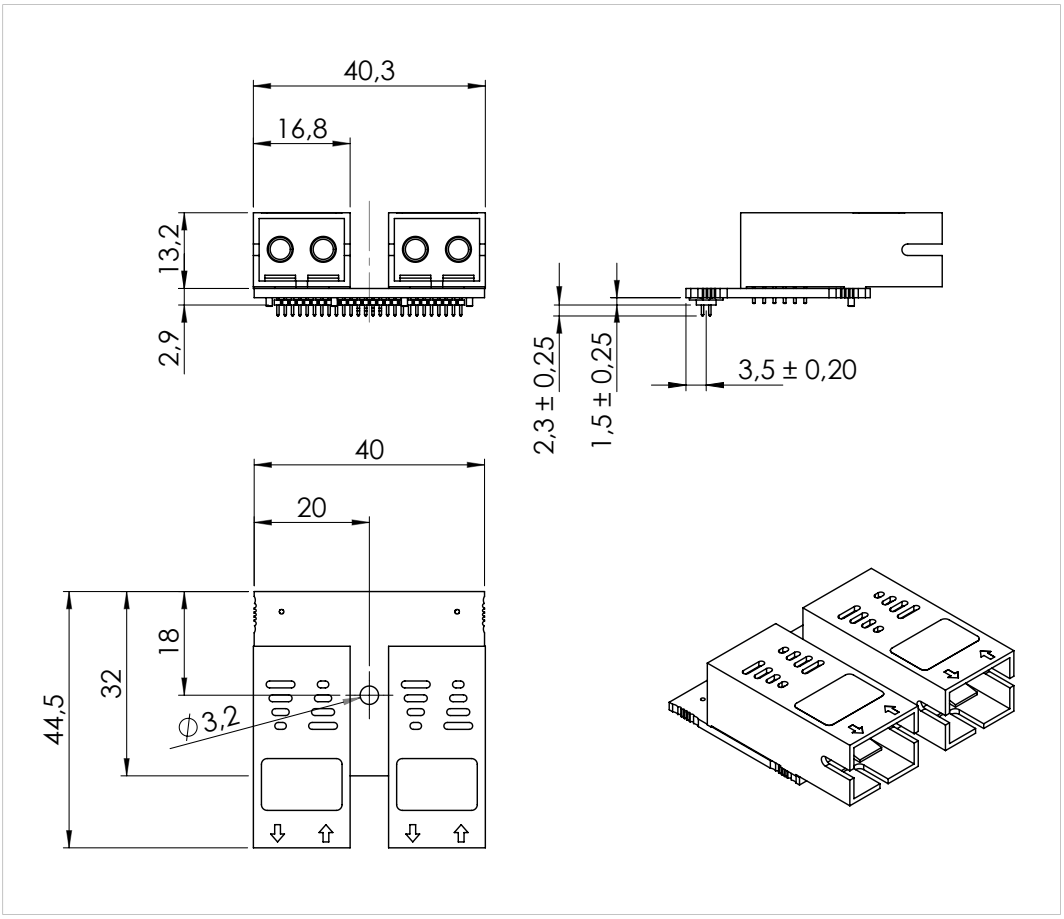

Fig. 18

B.6 Assembly

The Anybus CompactCom B40 Modbus Serial and the connector board are mounted separately on to the host application board. The connector board has to be secured using a screw, joining FE (functional earth) on the connector board to FE on the host application board. The screw holes of the Anybus CompactCom B40 Modbus Serial are not connected to FE, but to GND. If suggested components are used, the Anybus CompactCom B40 Modbus Serial can be mounted without screws in a low vibration environment, see *Shock and Vibration, p. 35* for more information.

The Anybus CompactCom B40 Modbus Serial can either be connected to the application board using headers, or soldered directly to the host application PCB.

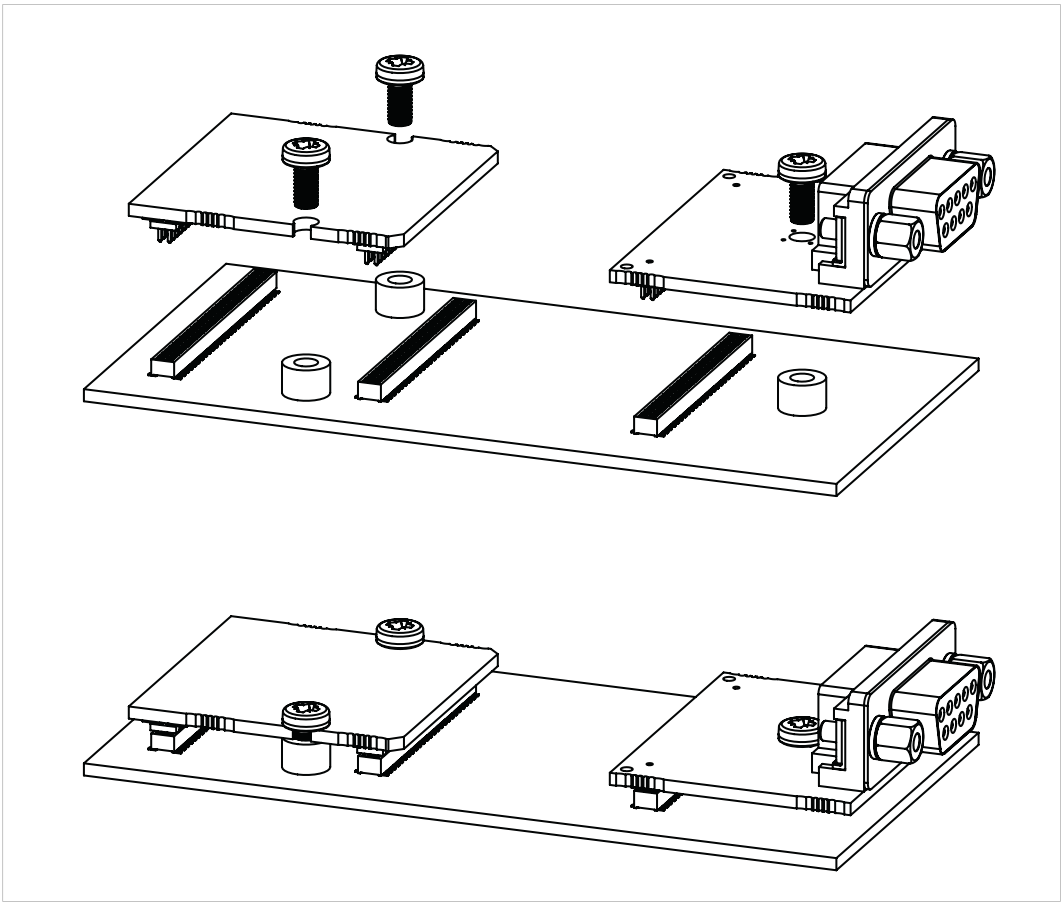

Fig. 19

Suggested components		
Header	Application interface	Samtec CLP-128-02-L-D (56 pin)
	Network interface	Samtec CLP-126-02-L-D (52 pin)
Stand-off (M3)	Pemnet SMTSO-M3-4-ET	

The screw standoffs are typically 4 mm tall. If the Anybus CompactCom B40 Modbus Serial and connector board are to be soldered directly to the host application board, standoffs should be 2 mm tall. Outer diameter may be 6 mm max. The standoffs should not extend outside the screw mount pads.

Recommended torque is 0.2 Nm. Locking paint can be used to secure the screws against loosening.

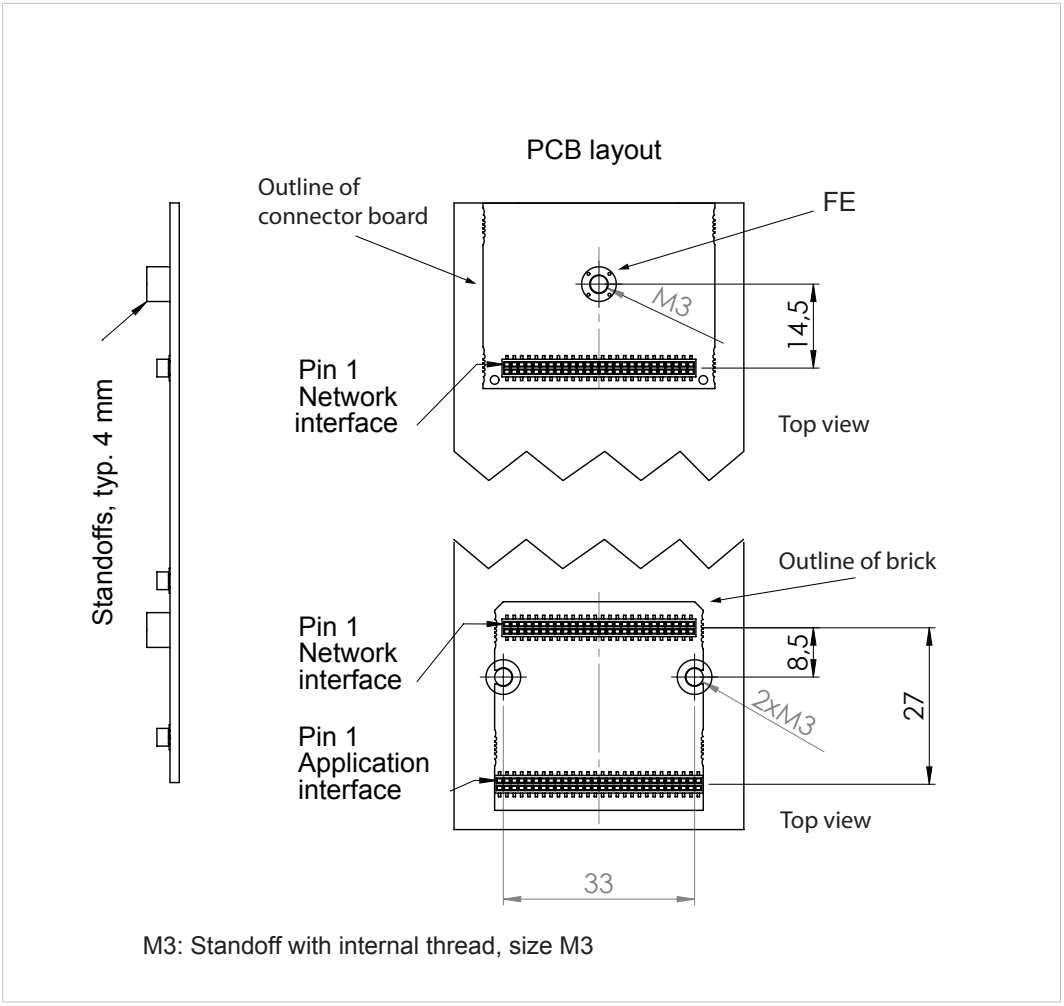
PCB layout
Outline of
connector board
FE
M3
14,5
Pin 1
Network
interface
Top view
Standoffs, typ. 4 mm
Outline of brick
Pin 1
Network
interface
8,5
2xM3
27
Pin 1
Application
interface
Top view
33
M3: Standoff with internal thread, size M3

Fig. 20

C Firmware Upgrade

Download and upgrade of network communication firmware for a specific fieldbus or industrial network can be performed in different ways, depending on which Anybus CompactCom B40 Modbus Serial that is to be upgraded.

Ethernet versions with FTP enabled	Use the Firmware Manager II tool or download the firmware to the designated folder in the file system.
Ethernet versions with FTP disabled **Fieldbus versions**	Unplug the Anybus CompactCom from the host application pcb and use the Anybus CompactCom 40 Starter Kit to perform the firmware upgrade. See the starter kit documentation for further information.

C.1 Using Firmware Manager II

This tool is available without cost from www.anybus.com and can be downloaded from the product pages. It can be used to download new firmware for any Ethernet Anybus CompactCom B40 Modbus Serial. Please note that FTP has to be enabled.

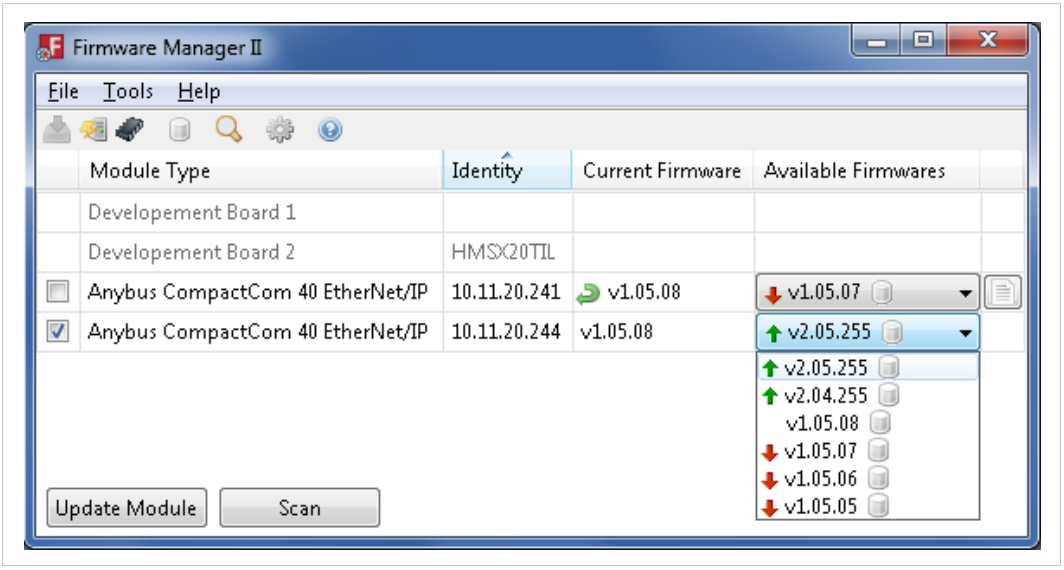

Fig. 21

Using the tool, perform the following steps to download new firmware to the module.

1. Connect a computer with the Firmware Manager II software installed to the network containing the module.
2. Start the Firmware Manager II tool.
3. Scan the network and find the module.
4. Click the Firmware Repository icon in the menu, to open the Firmware Repository window. Drag the firmware folder into the window to add the new firmware to the repository. Close the Firmware Repository window.
5. In the scan window, under the “Available Networks” tab, select the appropriate firmware for the module. Click the **Change Network** button. A confirmation window will appear. Click **Yes** to start the download of the new firmware. Please make sure that download is completely finished before continuing.
6. After download, a restart of the module is needed to install the new firmware. If the application allows it, it is possible to restart the module via the **Restart Module** button in the Firmware Manager II tool. If the application does not allow restart from the network, a manual restart of the module is needed.

For more information, see the help file in the Firmware Manager II software.

C.2 Using FTP

If the module supports FTP, this can be used to access the file system and upload the new firmware directly to the /firmware directory. The next time the module is started the firmware will be upgraded. After the firmware is installed, the firmware file is deleted from the /firmware directory.

D The Anybus State Machine

D.1 General Information

A fundamental part of the Anybus CompactCom B40 Modbus Serial is the Anybus State Machine.

The state machine shall be regarded as a Moore machine; i.e. the host application is not *required* to keep track of all state transitions, however it is *expected* to perform certain tasks in each state

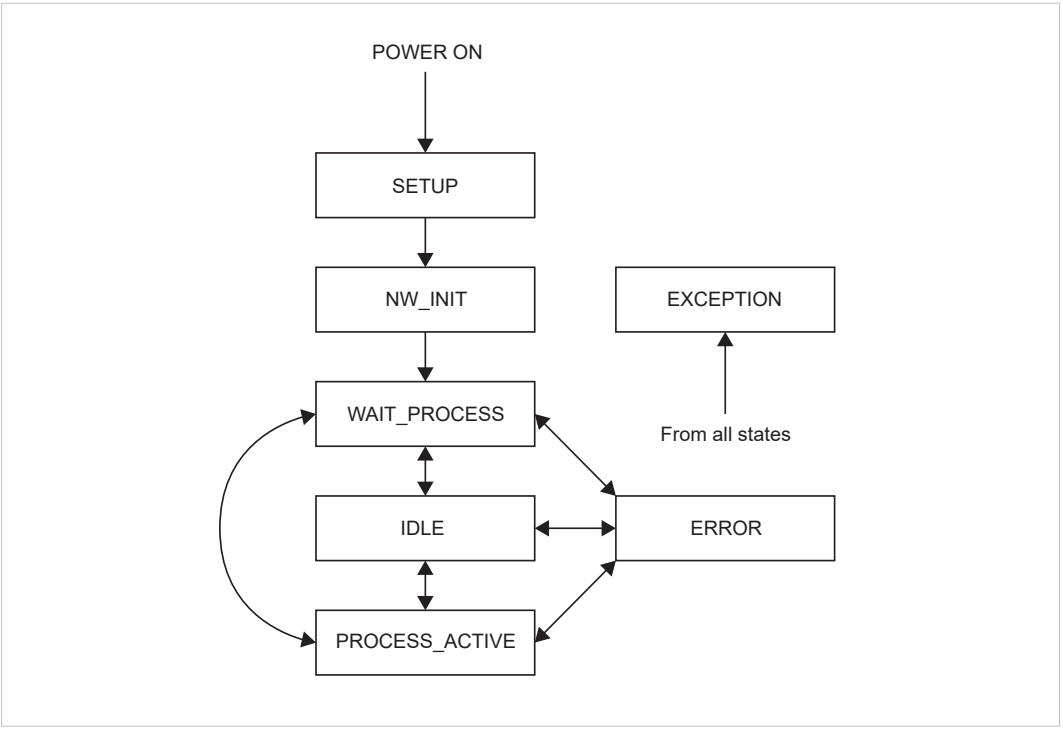

Fig. 22

D.2 State Dependent Actions

The expected actions for each state are listed below.

State	Description	Host Application Expected Action
SETUP	The host application configures the Anybus CompactCom.	The host application writes the process data settings to be used for the communication with the network. The Anybus CompactCom stay in this state until values have been written to Number of Write Parameters and Number of Read Parameters.
NW_INIT	The Anybus CompactCom starts its network interface parts.	If needed, update the Write Process Data, otherwise no action.
WAIT_PROCESS	Waiting for process data transporting connection to be allocated. E.g. a connection from the network master.	Update the Write Process Data. The Read Process Data is not valid.
IDLE	The process data transporting connection is idle. The exact interpretation of this state is network specific.	Update the Write Process Data. The host application may act upon the Read Process Data, or go to an idle state.
PROCESS_ACTIVE	Process data connection is allocated Normal data handling is possible	Update the Write Process Data and act upon the Read Process Data
ERROR	There is at least one fundamental network error, e.g. address conflict or configuration mismatch.	Update the Write Process Data. The Read Process Data is not valid.
EXCEPTION	The module has halted due to a host application related error. This state is unrecoverable, i.e. the triggering cause must be corrected, and the module must be restarted.	Read the Exception Code and if possible indicate error to the user.

> **!** The host application must keep the Write Process Data updated in NW_INIT (initial data), WAIT_PROCESS, IDLE, ERROR and PROCESS_ACTIVE since this data is buffered by the Anybus CompactCom, and may be sent to the network after a state shift.

E Implementation Guidelines

E.1 General

This appendix provides an example of a possible implementation for the Anybus CompactCom B40 Modbus Serial.

There are many different processors with different functionality available on the market today. The implementation in this appendix is to be regarded as an example that is designed for one single type of processor. Other hardware interfaces may require adjustments for timing, different functionality etc. It is important to fully understand the interface to take correct design decisions in order to obtain a stable and reliable design.

E.2 Example

The example in the figure below shows an implementation with Modbus Serial communication.

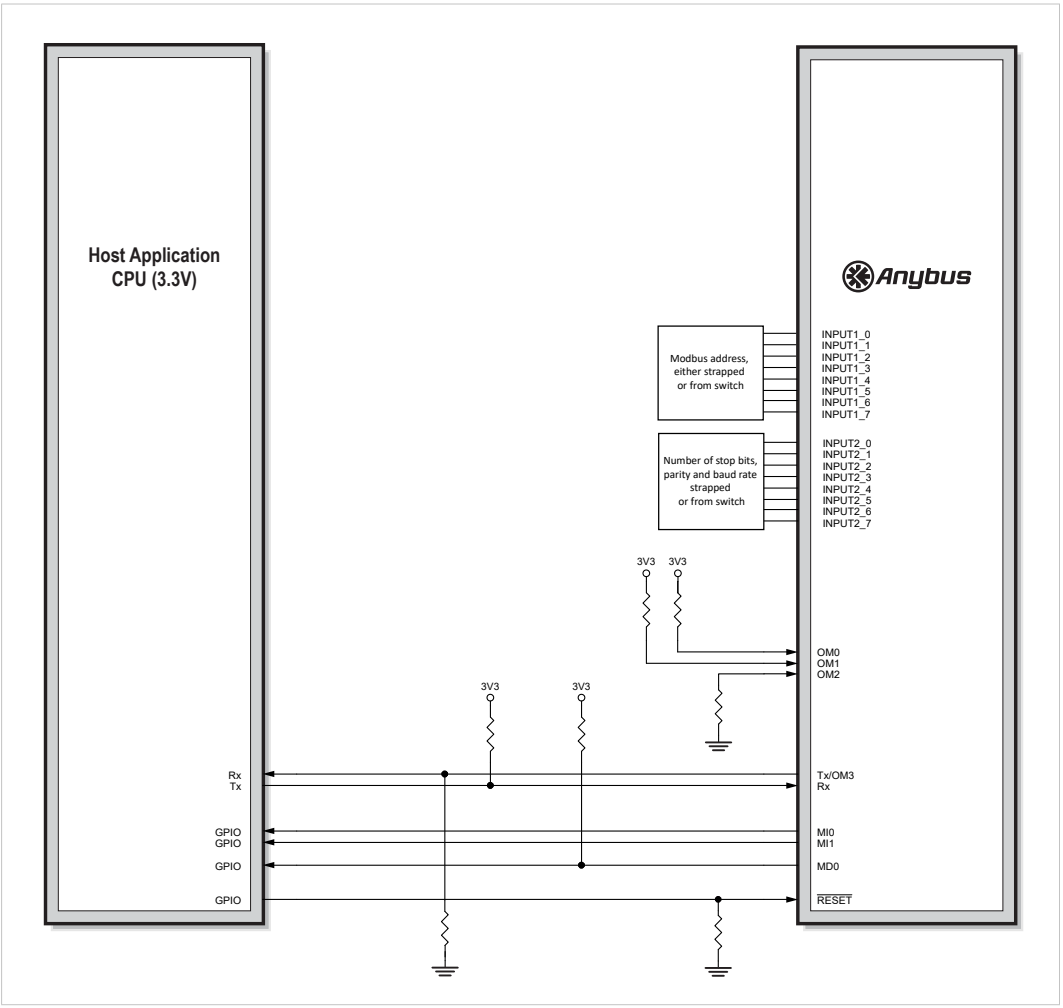

Fig. 23

E.3 Power Supply Considerations

E.3.1 General

The Anybus CompactCom 40 platform in itself is designed to be extremely power efficient. The exact power requirements for a particular networking system will however vary a lot depending on the components used in the actual bus circuitry.

While some systems usually require less than 250 mA of supply current at 3.3 V, some high performance networks, or networks which require the use of legacy ASIC technology, will consume up to 500 mA, or in rare cases even as much as 1000 mA.

As an aid when designing the power supply electronics, the networks have been divided into classes based on their power consumption as follows.

- Class A: less than 250 mA
- Class B: up to 500 mA
- Class C: up to 1000 mA

Please note that the power supply classifications take into account that the power budget is shared with a full fieldbus circuitry, e.g. the appropriate connector board and NW_LEDs with maximized consumption (20 mA each).

The following table lists the currently supported networking systems and their corresponding class.

Network	Class A	Class B	Class C
DeviceNet		X	
PROFIBUS	X		
CANopen	X		
EtherCAT		X	
PROFINET 2-Port		X	
PROFINET FO 2-Port			X
Ethernet/IP 2-Port		X	
EtherNet POWERLINK		X	
Common Ethernet		X	
CC-Link		X	
Modbus-TCP 2-Port		X	
CC-Link IE Field			X
BACnet/IP		X	

A power supply designed to fulfill Class A requirements (250 mA), will be able to support all networks belonging to class A, but none of the networks in Class B and C.

A power supply designed to fulfill Class C requirements, will be able to support all networks.

E.3.2 Bypass Capacitance

The power supply inputs must have adequate bypass capacitance for high-frequency noise suppression. It is therefore recommended to add extra bulk capacitors near preferably all the power supply inputs (or at least two):

Reference	Value (Ceramic)
C1	10 µF / 6.3 V

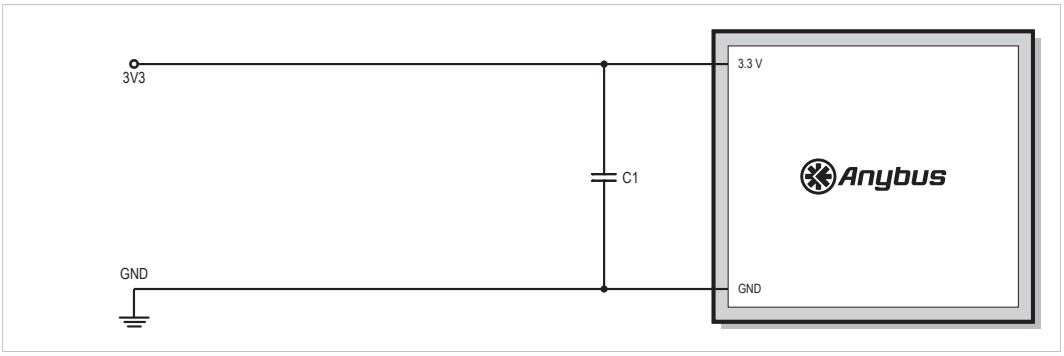

Fig. 24

E.3.3 3.3 V Regulation

The following example uses the LT1767 from Linear Technology to provide a stable 3.3 V power source for the module. Note that all capacitors in this example are of ceramic type.

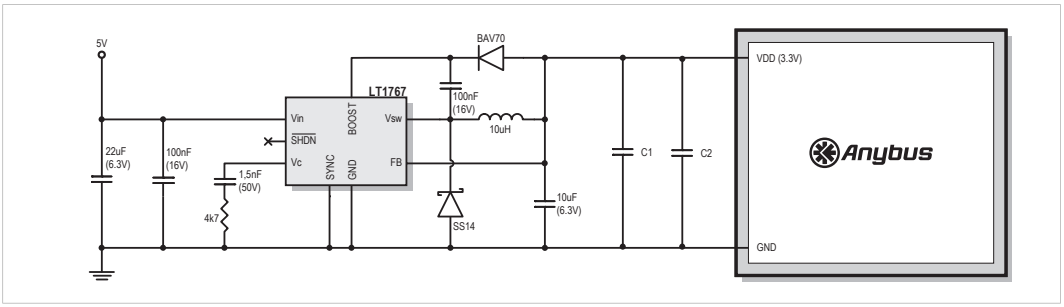

Fig. 25

For detailed information regarding this example, consult the data sheet for the LT1767 (Linear Technology).

F Network Interface Examples

This section contains typical examples of how to design the network interface, if the optional connector board is not to be used. Examples are given for the usual network connectors as well as for M12 connectors making a higher IP rating possible.

Example Schematics	Brick	Comments
10 and 100 Mbit Ethernet Network Interface(Copper)	EtherNet/IP EtherCAT Modbus TCP Common Ethernet POWERLINK PROFINET IRT BACnet/IP	All bricks for 100 Mb/s Ethernet based protocols, running on copper wire, use the same hardware. Ethernet trafo example: 7490100111A (Würth Elektronik Gmbh) RJ45 connector example: SS-60300-032 (Bel stewart)
100 Mbit Ethernet Network Interface (Fiber Optic)	PROFINET IRT	-
10 and 100 Mbit Ethernet Network Interface (M12)	EtherNet/IP EtherCAT Modbus TCP Common Ethernet POWERLINK PROFINET IRT BACnet/IP	All bricks for 100 Mb/s Ethernet based protocols, running on copper wire, use the same hardware. Ethernet trafo example: 7490100111A (Würth Elektronik Gmbh)
10, 100 and 1000 Mbit Ethernet Network Interface	CC-Link IE Field	RJ45 connector example: SS-60300-032 (Bel stewart)
PROFIBUS	PROFIBUS	-
PROFIBUS (M12)	PROFIBUS	-
DeviceNet	DeviceNet	-
DeviceNet (M12)	DeviceNet	-
CC-Link	CC-Link	-
CANopen	CANopen	-

Box 4126
300 04 Halmstad, Sweden

info@hms.se
SCM-1202-130 2.2 en-US / 2020-05-11 / 18401