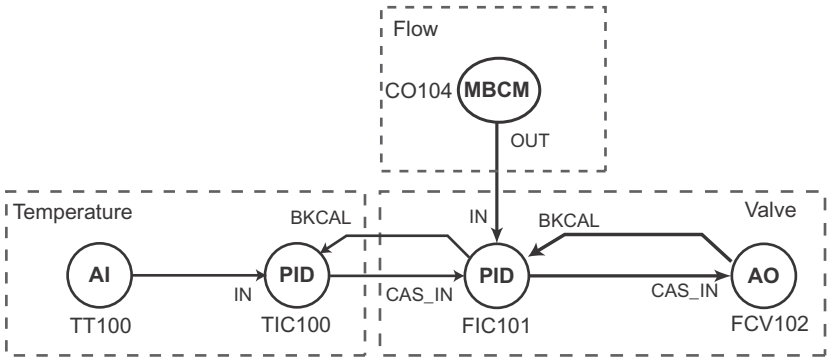
Level Pressure Flow Temperature Liquid Analysis Registration Systems Components Services Solutions

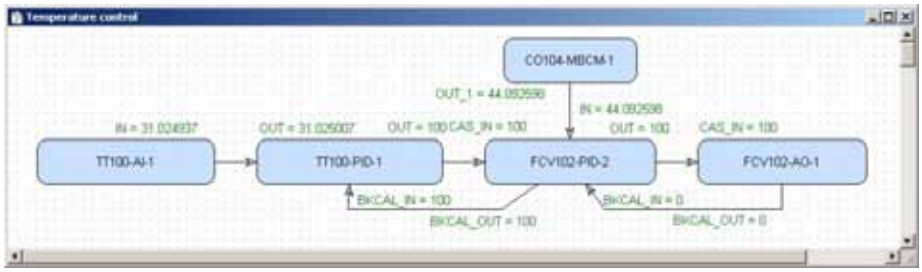
Operating Instructions

ControlCare Application Designer

MODBUS Tutorial

MODBUS





BA 037S/04/en/06.10
Product Version 2.05.xx
70101153

Endress+Hauser

People for Process Automation

Table of Contents

Revision History

Product version	Manual	Changes	Remarks
2.01.xx	BA037S/04/en/08.05	Original manual	
2.02.xx	BA037S/04/en/07.06	Product	• FB schedule configured by drag&drop (Chap.3.8 and 4.5.2) • Incremental download (Chap. 3.11.6 and 4.7.6)
		Editorial	• Update version and documentation tables
2.03.xx	BA037S/04/en/06.07	Program	• New preferences dialog (packing) • Assign All Tags added
		Going on-line	• New HSE Network Tools program • New Field Controller Web Server program
		Trouble-Shooting	• New FC Tools program and firmware download • New Exchange procedure
2.04.xx	BA037S/04/en/12.08	Modbus	• New Modbus implementation, see Chapter 2
		Editorial	• Contents Chapter 6 to moved to Chapter 2 • Tutorial revised to new Modbus implementaation
2.05.xx	BA037S/04/en/06.10	Modbus	• New: simultaneous operation as master and slave • Corresponfing tutorial in Chapter 5
		Editorial	• Version, documentation table, Windows support • Webserver screenshot updated
		Trouble-Shooting	• FRC LED description updated for battery power

Product Version

Details of product version and the individual components of Application Designer Suite can be seen in the About ControlCare dialog:

Start=>Programs=>Endress+Hauser=>ControlCare=>Tools=>About ControlCare

Registered Trademarks

PROFIBUS
Registered trademark of the PROFIBUS User Organisation, Karlsruhe Germany.

FOUNDATION™ Fieldbus
Trademark of the Fieldbus Foundation, Austin, TX 78759, USA

Microsoft®, Windows®, Windows 2000® ,Windows XP®, Windows 2003 Server®, Windows 2008 Server®,Windows 7®, Windows Vista® and the Microsoft logo are registered trademarks of the Microsoft Corporation.

Acrobat Reader® is a registered trade mark of the Adobe Systems Incorporated.

All other brand and product names are trademarks or registered trademarks of the companies and organisations in question

1 Safety

1.1 Designated use

ControlCare is a field-based control system comprising hardware and software modules. It can be used to visualize, monitor and control production processes. The approved usage of the individual units used in the system can be taken from the corresponding parts of the operating instructions.

The software described in this particular manual allows Modbus devices (master or slave) connected to a SFC162 FOUNDATION Fieldbus or SFC173 PROFIBUS Field Controller to be engineered, configured and commissioned. In addition, appropriate control strategies can be built using the function blocks contained in the controller and connected devices.

1.2 Installation, commissioning and operation

ControlCare Field Controller modules have been designed to operate safely in accordance with current technical safety and EU directives. Essential to their use is the ControlCare Application Designer software, which allows control strategies to be created for both FOUNDATION Fieldbus and PROFIBUS applications. Field devices, links, junction boxes, cables and other hardware comprising the Fieldbus sytem must also be designed to operate safely in accordance with current technical safety and EU directives.

If devices are installed incorrectly or used for applications for which they are not intended, or if the controller is not configured correctly, it is possible that dangers may arise. For this reason, the system must be installed, connected, configured, operated and maintained according to the instructions in this and the associated manuals: personnel must be authorised and suitably qualified.

1.3 Operational safety

Location

Field Controllers must be mounted in a permanent and weather-protected location in a safe area. The environment shall be a metal cabinet or an installation frame with a well grounded mounting plane. The environment shall be protected.

Hazardous areas

The controller must be connected to networks operating in explosion hazardous areas via barriers or other safety components. When installing components in explosion hazardous areas:

- Ensure that all installion and maintenance personnel are suitably qualified
- Check that all equipment has the appropriate safety certificates
- Observe the specifications in the device certificates as well as national and local regulations.

This topic is discussed in BA013S (FOUNDATION Fieldbus Guidelines) and BA034S (PROFIBUS Guidelines).

EMC

All modules are suitable for industrial use and conform with the following standard, see Appendix:

- EN 61326: 1997/A1: 1998
  Interference emmision: Class A apparatus
  Interference immunity: as per Annex A, industrial environment

Depending upon the environment in which the bus is operating, particular attention should be paid to the grounding of the bus cables. This topic is discussed in BA013S (FOUNDATION Fieldbus Guidelines) and BA034S (PROFIBUS Guidelines).

IP Address

Field Controller is normally configured from a workstation connected into the control system backbone. You will require a unique IP address to set it up.

Warning

The use of IP addresses is strictly controlled. Usually your system administrator will be authorised to allocate unique addresses. Assigning an unauthorised address to a Field Controller may result in conflicts within your system and the failure of the associated devices!

It is recommended that ControlCare Field Controllers and OPC servers are not installed in an office network, as the large data packets exchanged between office equipment may lead to timeouts and intermittent communication errors. Ideally, the ControlCare system network should operate within its own IP domain; if this is not possible it should be separated from other parts of the network by a managed switch.

Since the system can be accessed and manipulated through the various Field Contoller tools, it is advisable to control access both to the workstation and the folders in which the configuration is stored. Always make a back-up of the project.

Technical improvement

Endress+Hauser reserves the right to make technical improvements to its software and equipment at any time and without prior notification. Where such improvements have no effect on the operation of the equipment, they are not documentated. If the improvements effect operation, a new version of the operating instructions is normally issued.

1.4 Conventions and icons

In order to highlight safety relevant or alternative operating procedures in the manual, the following conventions have been used, each indicated by a corresponding icon in the margin.

Safety conventions

Icon	Meaning
	A note highlights actions or procedures which, if not performed correctly, may indirectly affect operation or may lead to an instrument response which is not planned
	Caution! Caution highlights actions or procedures which, if not performed correctly, may lead to personal injury or incorrect functioning of the instrument
	Warning! A warning highlights actions or procedures which, if not performed correctly, will lead to personal injury, a safety hazard or destruction of the instrument

1.5 ControlCare documents

Table 1.1 indicates the documents, planned and realized, containing safety relevant information, installation, commissioning and operating instructions for the equipment and software associated with Field Controller.

All documentation available at the time of release is included on the ControlCare CD-ROM and is installed in Start=>Programs=>Endress+Hauser=ControlCare=Manuals during set-up.

Component	Description	Document type	Designation	Order No.
System	ControlCare System Overview	Operating manual	BA016S/04/en	56004883
	ControlCare System Design	Operating manual	BA039S/04/en	Planned
	ControlCare System Specifications	Operating manual	BA040S/04/en	56004888
Software	Application Designer Overview	Operating manual	BA017S/04/en	70104301
	Application Designer: Local I/O Tutorial	Operating manual	BA032S/04/en	71095009
	Application Designer: FF Tutorial	Operating manual	BA019S/04/en	70101151
	Application Designer: PROFIBUS Tutorial	Operating manual	BA036S/04/en	70101152
	Application Designer: MODBUS Tutorial	Operating manual	BA037S/04/en	70101153
	Application Designer: IEC 61131-3 Ladder Logic Tutorial	Operating manual	BA038S/04/en	70101386
	Application Designer: IEC 61131-3 Structured Text Tutorial	Operating manual	BA056S/04/en	71060063
	Field Control (OPC) Servers	Operating manual	BA018S/04/en	71031428
	SFC162 Visitor	Operation manual	BA069S/04/en	71113457
Field Controller	Hardware Installation Guide	Operating manual	BA021S/04/en	56004885
	Commissioning and Configuration	Operating manual	BA035S/04/en	56004887
Function Blocks	Function Block Manual	Operating manual	BA022S/04/en	56004886
Set-Up	Getting Started	Operating manual	BA020S/04/en	56004884
General	FOUNDATION Fieldbus Guidelines	Operating manual	BA013S/04/en	70100707
	PROFIBUS Guidelines	Operating manual	BA034S/04/en	56004242

Tab. 1-1: ControlCare Documentation

2 Modbus

This tutorial describes all steps necessary for integrating Modbus values into a SFC162 or SFC173 Field Controller. It does not aim to give an exhaustive account of the associated Application Designer functions, but rather shows you one of a number of methods to reach your goal. It is assumed that the user is familiar with the programming of Modbus masters and slaves.

The tags and names used in the tutorial are imaginary and will be different in a proper application. A full description of Application Designer functions is to be found in Application Designer Overview BA017S/04/en. Function block descriptions are to be found in BA022S/04/en, Function Block manual.

2.1 Description

MODBUS is a quasi-industrial standard developed some years ago by Gould-Modicon and provides a messaging service that may run on a variety of physical layers. For the SFC162 and SFC173 Field Controllers, there are two possibilities for integrating Modbus:

- MODBUS RTU can be connected point-to-point to the RS-232C input on the front panel. If appropriate a RS-232C/RS-485 (or RS-232C/RS-422) interface is required for connection to the device.
- MODBUS TCP (also known as MODBUS TCP/IP) can be connected to the Ethernet connector on the front panel. This allows the exchange of data between the Field Controller and a PLC, Remote I/O and/or operator panel.

The MODBUS protocol exchanges data in a master-slave relationship. Each slave has a unique address, and the data are identified by their location in the slave address register. Certain characteristics of the MODBUS protocol are fixed, such as the frame format, frame sequences, handling of communications errors, exception conditions and the functions performed. Other characteristics are user selectable; these include transmission medium, baudrate, character parity, number of stop bits, and transmission modes. The contents of the data carried by the protocol are also freely selectable, i.e. nothing is said about strings, integers, floating-point numbers etc.

The MODBUS protocol controls the query and response cycle between master and slave devices. Only the master can initiate a transaction. A query and response may involve only a single slave, or it may be in the form of a broadcast, in which case the slaves do not answer. The query is contained in a frame that includes the address of the intended receiver, what this slave is to do, data needed to perform the action, and a means of checking for errors. The slave checks if errors have occurred and performs the desired action. After the action is performed the slave builds the response and returns it to the master. The master can send another message to any slave as soon as it receives a valid response or after a user-selected time interval. This "timeout" period has to be selected on the master device and depends on the slave response time.

Data can be exchanged in two transmission modes: ASCII (American Standard Code for Information Interchange) and RTU (Remote Terminal Unit). The major differences between them are the type of error check performed on the message and the number of characters used. MODBUS offers several read, write and test functions, each identified by a code number. They are designed as control commands for sensors and actuators, e.g. coils, inputs, input registers, holding or output registers, diagnosis and test reports, programs, polling control and reset. For MODBUS TCP the serial frame is simply inserted into the Ethernet data frame. In addition, not all codes are implemented.

2.2 Modbus in ControlCare

2.2.1 Implementation

ControlCare Field Controllers SFC162 and SFC173 are equipped with both a Modbus serial and Ethernet interface. As a result, they can be operated in one of the following roles:

- Modbus Serial or TCP Master
- Modbus Serial or TCP Slave
- Modbus TCP Master and Serial Slave, Modbus TCP Slave and Serial Master ot Modbus TCP Master and TCP Slave

The Modbus role and corresponding parameters are configured in the Modbus Configuration Block, MBCF. As serial master, a Field Controller supports up to 128 Modbus slaves; as TCP master eight Modbus slaves only. In serial (RTU) operation baudrates from 9600 to 115200 are possible. If configured as a serial or TCP slave, a field controller can be accessed by one Modbus master only.

Establishing communication

Modbus communication is initially established by pressing the ON_APPLY button in the MBCF block. If changes are made to the Modbus configuration, the button must always be pressed to re-establish communication with the new parameters.

On recovery after a power failure, the Field Controller resumes communication automatically within 30 seconds. If a Modbus slave device loses power or is disconnected, the Field Controller will continue to poll it until it recovers, is re-connected or is removed from the strategy.

When the Field Controller is configured as a TCP slave, access to the Modbus registers is controlled by entering the IP address of the Modbus master. The Field Controller will normally communicate with the master via Port 502. If this port is unavailable, it is possible to specify a secondary TCP/IP port number.

Modbus commands

Field Controller supports the following Modbus functions

Function	Function Code	Hex	Applies To
Read discrete inputs	2	0x02	Master, Slave
Read coils	1	0x01	Master, Slave
Write single coil	5	0x05	Master, Slave
Write multiple coils	15	0x0F	Master, Slave
Read input register	4	0x04	Master, Slave
Read holding register	3	0x03	Master, Slave
Write single register	6	0x06	Master, Slave
Write multiple registers	16	0x10	Master, Slave
Read/Write multiple register	23	0x17	Slave only

Registers

Modbus specifies four different types of register:

- Discrete input registers contain the discrete input values and possibly status
- Input registers contain analog input values and status
- Coil registers contain discrete output values and possibly status
- Holding registers contain analog output values and status

In addition, it is possible to "pack" discrete inputs and outputs into words, which are then stored as appropriate in the input or holding registers. Fig. 2-1 gives an overview of the register and reference address ranges used for each register type in Field Controller. The table shows the relationship between the function block input and output parameters and the registers for master and slave roles.

.

FB Parameter	Field Controller as Modbus Master	Field Controller as Modbus Slave
IN_D1 to IN_D4	Value written to slave coil	Value read by master from discrete input
OUT_D1 to OUT_D4	Value read from slave discrete input/coil	Value written by master to coil
IN_1 to IN_4	Value written to slave holding register	Value read from master from input register
OUT_1 to OUT_4	Value read from slave input/holding register	Value written by master to holding register

Access

Depending upon the Modbus implementation, the registers are accessed by specifying:

- the function code and the register address or
- the reference address

When Field Controller is used as a master, the addresses are entered in the LOCATOR and SCALE_LOC parameters. When it is acting as a slave, it offers the values by means of an address table which is dependent upon the unique MBCS block identifier LOCAL_MOD_MAP.

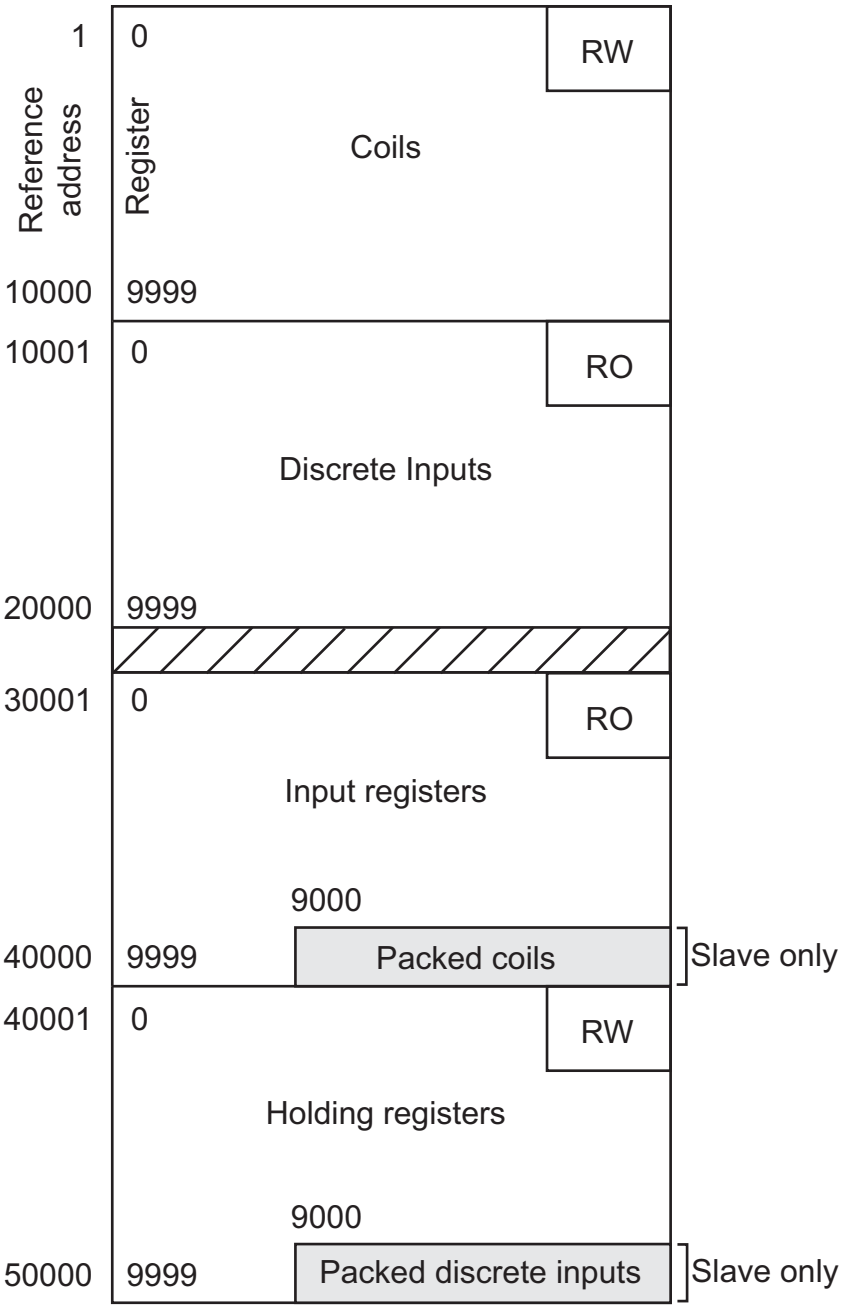


Fig. 2-1: Mapping of Modbus registers in ControlCare Field Controller

Refresh time

When the Field Controller is acting as a slave, the Modbus discrete inputs and input registers will be refreshed once every macrocycle. The length of the macrocycle depends on the number of blocks in use, the execution time of the blocks and the number of Modbus values.

In general, shorter refresh times can be attained by using the hybrid block with embedded I/Os, as this allows Modbus data to be input/accessed directly, rather than through additional input and output blocks, see BA032S/04/en, Local I/O Tutorial, for more details.

2.2.2 Modbus Configuration Block

The Modbus Configuration Block MBCF is used to configure the Field Controller for a role as master or slave. The table below lists the parameters and gives a short explanation of their function. Details of use can be found in the appropriate section of the tutorial.

Parameter	Valid range	Default value	Description / Action
ST_VER		0	Indicates the revision level of the block's static parameters and may be used in configuration management
TAG_DESC		blanks	Allows the entry of a block description (up to 32 characters) which may be used in a human interface or in block documentation to clarify the block application
STRATEGY	0 to 255	0	Allows the entry of a user assigned value that may be used in configuration or diagnostics as a key in sorting block information
ALERT_KEY	1 to 255	1	Allows the entry of a user assigned value that may be used in sorting alarms or events generated by a block
MODE_BLK	TARGET	O/S	Block mode, set to Auto
BLOCK_ERR	0 to 15		Block errors
MEDIA	0, 1	Serial	Defines Modbus medium, "( )" slave in master + slave below 0: Serial (+TCP/IP), 1: TCP/IP (+Serial), 2: TCP/IP (+TCP/IP)
MASTER_SLAVE	0,1	Slave	Defines if Field Controller is master or slave • 0: Master, 1: Master + Slave, 2: Slave
TIMEOUT	0 - 65535	1000	Time allowed for a response from a slave (for Field Controller master) or for the OUTs be updated (for Field Controller slave). Value 0 is used to disable. – In the case of a slave, the TIMEOUT must be set to a value greater than the write cycle of the Modbus master, otherwise the status of the mapped values is BAD.
SERIAL_CONFIG	-	-	Configures serial interface
.BAUD_RATE	6 - 10	19200	Defines the baud rate • 6: 9600, 7: 19200, 8: 38400, 9: 57600, 10: 115200
.STOP_BITS	0, 1	1	Defines the number of stop bits (only for media serial). • 0: 1, 1: 2
.PARITY	1 - 3	Even	Defines the parity (only for media serial) • 0 :None, 1: Even, 2: Odd
TCP_IP_CONFIG	-	-	Configures TCP/IP interface
.SECOND_MOD_PORT			Second communication port (Port 502 is always open)
MASTER_CONFIG			To be configured when Field Controller is Modbus master
.NUMBER_RETRIES	0 - 255	1	Number of times Field Controller tries to retransmit the data if it does not receive a response from a slave
.MOD_SCAN_TIME			When online, displays Modbus scan time in ms when Field Controller is Modbus master (average of the last 10 cycles)
.MAX_DATA_LENGTH			Max. number of registers that can be read from a slave with a single telegram – Use only if slave does not support the standard length of 125 registers (250 byte)
SLAVE_CONFIG			To be configured when Field Controller is Modbus slave
.DEVICE_ADDRESS	1 - 247	1	Modbus slave address of Field Controller
.MAX_MOD_REACTION_TIME			When online, displays max. time to respond to a master
TCP_SLAVE_ADDRESSES			To be configured when Field Controller is Modbus TCP master
.IP_SLAVE_X			IP address of Modbus TCP slave
.DEVICE_ID_X			Unit address of Modbus TCP slave
.COMM_STATUS_X			When online, displays status of output data
TCP_ACCESS_LIST			To be configured when Field Controller is Modbus TCP slave
.IP_x			IP address of TCP master allowed to access to registers
ON_APPLY	0, 1	None	Applies the changes made in the Modbus blocks • 0: None, 1: Apply
UPDATE_EVT			This alert is generated by any change to the static data
MB_COMM_STATUS			Displays status of Modbus communication (OK when working)
USED_MOD_REGISTERS	RO		When online, displays number of Modbus points in use
FREE_MOD_REGISTERS	RO		When online, displays available Modbus points in percentage

2.3 Operation as Modbus Control Master

The Modbus Control Master block is required when the Field Controller is to act as a Modbus master. In this role Field Controller reads from or writes to registers in a Modbus slave. The Modbus values are mapped through a FOUNDATION Fieldbus function block that has four sets of channels (AI, DI, AO, DO), allowing connection to other function blocks in a control strategy, see Fig. 2-2.

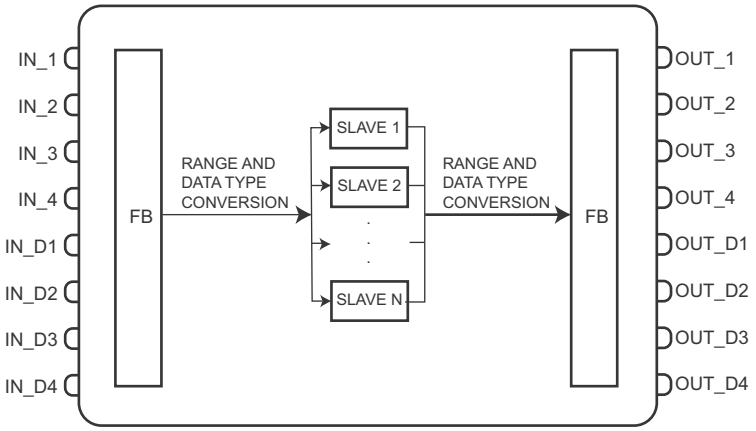


Fig. 2-2: Schematic diagram of the Modbus Control Master block

2.3.1 Block description

Up to 16 MBCM blocks can be created in a project. The blocks are managed via the parameter LOCAL_MOD_MAP which must have a unique value (0 to 15) for each block.

The mapping of the Modbus values is controlled by SCALE_LOC_XXX parameters for analog values and LOCATOR_XXX parameters for discrete values. These point the Field Controller to the register addresses to which data are to be written or from which data are to be read. The table gives an overview of the relationships of the input and output parameters to the mapping parameters:
.

Parameter	Type	Function in slave	Mapping parameter	Reference address
IN_1 to IN_4	Analog input	Write to holding register	SCALE_LOC_INx	40001 + register address
IN_D1 to IN_D4	Discrete input	Write to coil	LOCATOR_IN_Dx	1 + coil address
OUT_1 to OUT_4	Analog output	Read from input register	SCALE_LOC_OUTx	30001 + register address
		Read from holding register		40001 + register address
OUT_D1 to OUT_D4	Discrete output	Read from discrete input	LOCATOR_OUT_Dx	10001 + register address
		Read from coil		1 + coil address

Register addresses

Both SCALE_LOC_XXX and LOCATOR_XXX use three parameters to locate the register containing the Modbus value:

- SLAVE_ADDRESS points to the Modbus address of the connected slave
- MODBUS_ADDRESS_OF_VALUE points to the register address of the required value
- MODBUS_ADDRESS_OF_STATUS points to the register address of the associated status

In the case of analog data types that require two registers, only the first one is entered in MODBUS_ADDRESS_OF_VALUE. The second, adjacent address is automatically reserved according to the data type.

If the slave variable does not support a status or the status does not conform to the FF format, then zero must be entered in MODBUS_ADDRESS_OF_STATUS, see also Status handling. If required, an additional characterization can be made in the output status parameter OUT_x.STATUS or OUT_Dx.STATUS, e.g. "GoodCascade".

Scaling

SCALE_LOC_XXX contains additional parameters concerning data type and scaling of analog values:

- DATA_TYPE determines the data format of the value, see Chapter 2.5
- FROM_EU_XX determines the scaling of the "input" parameter
- TO_EU_XX determines the scaling of the "output" parameter

The scaling is performed as shown in Fig. 2-2, whereby the value to be scaled may lie outside the given limits.

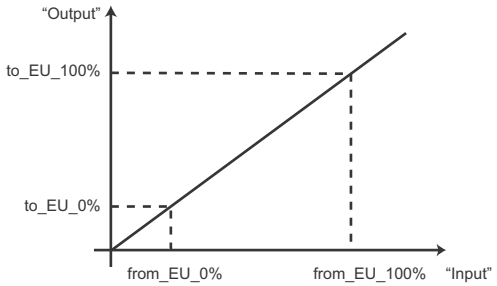


Fig. 2-3: Scaling of "input" to "output" units

The meaning of the scaling parameters in SCALE_LOC_INx and SCALE_LOC_OUTx is as follows:

Parameter	SCALE_LOC_INx	SCALE_LOC_OUTx
.FROM_EU_0	Lower range limit IN_x value	Lower range limit Modbus slave value
.FROM_EU_100	Upper range limit IN_x value	Upper range limit Modbus slave value
.TO_EU_0	Lower range limit Modbus slave value	Lower range limit OUT_x value
.TO_EU_100	Upper range limit Modbus slave value	Upper range limit OUT_x value

Status handling

The status of a slave output value is mapped in the corresponding OUT_XX.STATUS parameter. The information it carries is dependent on the entry in MODBUS_ADDRESS_OF_STATUS and whether the period entered in the TIMEOUT parameter in the MBCF block has elapsed without a response being received from the slave.

MODBUS_ADDRESS_OF_STATUS	TIMEOUT	Status
Slave address register	Slave responding (communicated within timeout period)	As Slave
	Slave not responding (timeout elapsed)	Bad
0 (no status or status not FF conform)	Slave responding (communicated within timeout period)	Good
	Slave not responding (timeout elapsed)	Bad

The communication status of each Modbus variable can be read from the COMM_STATUS parameter. Each bit corresponds to one variable and corresponds to a logical OR between its status and value, whereby:

- If only the value is used, the status is considered zero
- If only the status is used, the value is considered zero.

If the bit is set (=1), there was an error during writing/reading of the respective parameter. The table shows the relationship between bit number and parameter.

BIT	PARAMETER	BIT	PARAMETER	BIT	PARAMETER	BIT	PARAMETER
0	IN_1	4	IN_D1	8	OUT_1	12	OUT_D1
1	IN_2	5	IN_D2	9	OUT_2	13	OUT_D2
2	IN_3	6	IN_D3	10	OUT_3	14	OUT_D3
3	IN_4	7	IN_D4	11	OUT_4	15	OUT_D4

2.3.2 Block parameters

The table below lists the parameters of the MBCM block and gives a short explanation of their function. Details of use can be found in the appropriate section of the tutorial.

Parameter	Valid range / Options	Default value	Description / Action
ST_VER		0	See Chapter 2.2.2
TAG_DESC		blanks	
STRATEGY	0 to 255	0	
ALERT_KEY	1 to 255	1	
MODE_BLK	TARGET	O/S	Block mode, set to Auto
BLOCK_ERR	0 to 15		Block errors, see Chapter 2.2.2
LOCAL_MOD_MAP	0 to 15	0	Unique identifier for MBCM block
COMM_STATUS		0	Indicates if communication from slave is good or not (each bit corresponds to a Modbus variable
IN_1			Value and status of analog input 1
SCALE_LOC_IN1			Scaling, data format and register addresses for input signal 1
IN_2			Value and status of analog input 2
SCALE_LOC_IN2			Scaling, data format and register addresses for input signal 2
IN_3			Value and status of analog input 3
SCALE_LOC_IN3			Scaling, data format and register addresses for input signal 3
IN_4			Value and status of analog input 4
SCALE_LOC_IN4			Scaling, data format and register addresses for input signal 4
IN_D1			Value and status of discrete input 1
LOCATOR_IN_D1			Register addresses for discrete input signal 1
IN_D2			Value and status of discrete input 2
LOCATOR_IN_D2			Register addresses for discrete input signal 2
IN_D3			Value and status of discrete input 3
LOCATOR_IN_D3			Register addresses for discrete input signal 3
IN_D4			Value and status of discrete input 4
LOCATOR_IN_D4			Register addresses for discrete input signal 4
OUT_1			Value and status of analog output 1
SCALE_ LOC_OUT1			Scaling, data format and register addresses for output signal 1
OUT_2			Value and status of analog output 2
SCALE_ LOC_OUT2			Scaling, data format and register addresses for output signal 2
OUT_3			Value and status of analog output 3
SCALE_ LOC_OUT3			Scaling, data format and register addresses for output signal 3
OUT_4			Value and status of analog output 4
SCALE_ LOC_OUT4			Scaling, data format and register addresses for output signal 4
OUT_D1			Value and status of discrete output 1
LOCATOR_OUT_D1			Register addresses for discrete output signal 1
OUT_D2			Value and status of discrete output 2
LOCATOR_OUT_D2			Register addresses for discrete output signal 2
OUT_D3			Value and status of discrete output 3
LOCATOR_OUT_D3			Register addresses for discrete output signal 3
OUT_D4			Value and status of discrete output 4
LOCATOR_OUT_D4			Register addresses for discrete output signal 4
UPDATE_EVT			This alert is generated by any change to the static data
BLOCK_ALM			Block alarms

2.4 Operation as Modbus Control Slave

The Modbus Control Slave block is required when the Field Controller is to act as a Modbus slave. In this role the Modbus Master reads from or writes to registers in the Field Controller. The Modbus values are mapped through a FOUNDATION Fieldbus function block that has four sets of channels (AI, DI, AO, DO), allowing connection to other function blocks in a control strategy, see Fig. 2-3.

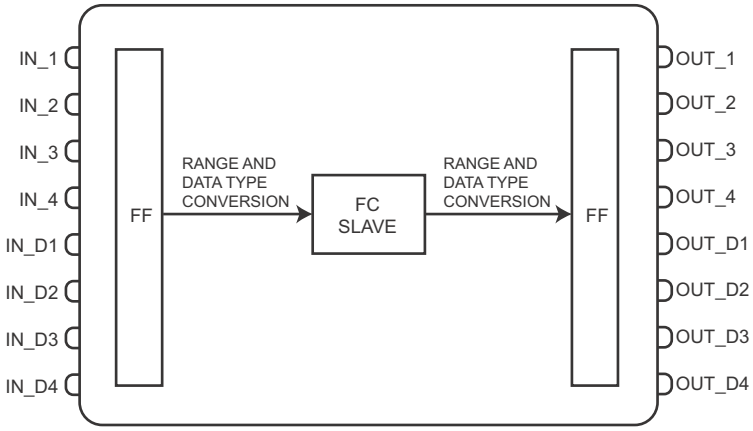


Fig. 2-4 Schematic diagram of the Modbus Control Slave block

2.4.1 Block description

Up to 16 MBCS blocks can be created in a project. The blocks are managed via the parameter LOCAL_MOD_MAP which must have a unique value (0 to 15) for each block.

Register addresses

The value of LOCAL_MOD_MAP is used to define a unique set of Modbus register address ranges for the particular MBCS block, whereby:

- Register address = Constant + 8 x Value of LOCAL_MOD_MAP for analog values
- Register address = Constant + 4 x Value of LOCAL_MOD_MAP for discrete values/status.

The Modbus master accesses the Field Controller slave registers by specifying the function code and the register address. The registers are assigned to the analog/discrete input/outputs as follows:

IN_x

Parameter	Type	Register Address	Access by Function Code
IN_1.Value	Input register	0 + (8 x LOCAL_MOD_MAP)	0x04 (Read input register)
IN_2.Value		2 + (8 x LOCAL_MOD_MAP)	
IN_3.Value		4 + (8 x LOCAL_MOD_MAP)	
IN_4.Value		6 + (8 x LOCAL_MOD_MAP)	
IN_1.Status		128 + (4 x LOCAL_MOD_MAP)	
IN_2.Status		129 + (4 x LOCAL_MOD_MAP)	
IN_3.Status		130 + (4 x LOCAL_MOD_MAP)	
IN_4.Status		131 + (4 x LOCAL_MOD_MAP)	

IN_Dx

Parameter	Type	Register Address	Access by Function Code
IN_D1.Value	Discrete Input	0 + (4 x LOCAL_MOD_MAP)	0x02 (Read discrete input)
IN_D2.Value		1 + (4 x LOCAL_MOD_MAP)	
IN_D3.Value		2 + (4 x LOCAL_MOD_MAP)	
IN_D4.Value		3 + (4 x LOCAL_MOD_MAP)	
IN_D1.Status	Input register	192 + (4 x LOCAL_MOD_MAP)	0x04 (Read input register)
IN_D2.Status		193 + (4 x LOCAL_MOD_MAP)	
IN_D3.Status		194 + (4 x LOCAL_MOD_MAP)	
IN_D4.Status		195 + (4 x LOCAL_MOD_MAP)	

OUT_x

Parameter	Type	Register Address	Access by Function Code
OUT_1.Value	Holding register	0 + (8 x LOCAL_MOD_MAP)	0x03 (Read holding registers) 0x06 (Write single register) 0x10 (Write multiple registers) 0x17 (Read/Write multiple registers)
OUT_2.Value		2 + (8 x LOCAL_MOD_MAP)	
OUT_3.Value		4 + (8 x LOCAL_MOD_MAP)	
OUT_4.Value		6 + (8 x LOCAL_MOD_MAP)	
OUT_1.Status		128 + (4 x LOCAL_MOD_MAP)	
OUT_2.Status		129 + (4 x LOCAL_MOD_MAP)	
OUT_3.Status		130 + (4 x LOCAL_MOD_MAP)	
OUT_4.Status		131 + (4 x LOCAL_MOD_MAP)	

OUT_Dx

Parameter	Type	Register Address	Access by Function Code
OUT_D1.Value	Coil	0 + (4 x LOCAL_MOD_MAP)	0x01 (Read coils) 0x05 (Write single coil) 0x0F (Write multiple coils)
OUT_D2.Value		1 + (4 x LOCAL_MOD_MAP)	
OUT_D3.Value		2 + (4 x LOCAL_MOD_MAP)	
OUT_D4.Value		3 + (4 x LOCAL_MOD_MAP)	
OUT_D1.Status	Holding register	192 + (4 x LOCAL_MOD_MAP)	0x03 (Read holding registers) 0x06 (Write single register) 0x10 (Write multiple registers) 0x17 (Read/Write multiple registers)
OUT_D2.Status		193 + (4 x LOCAL_MOD_MAP)	
OUT_D3.Status		194 + (4 x LOCAL_MOD_MAP)	
OUT_D4.Status		195 + (4 x LOCAL_MOD_MAP)	

The Field Controller also offers Discrete Inputs and Coils as packed words. The associated status is not available as packed words and must be acquired individually from the input or holding register.

Packed words

Type	Parameters	Register Address	Access by Function Code
Input register (Discrete inputs)	DI0 (Bit0) - DI15 (Bit15)	9000	0x04 (Read input register)
	DI16 - DI31	9001	
	...	9002 - 9624	
	DI9984 - DI9999	9625	
Holding register (Coils)	DO0 (Bit0) - DO15 (Bit15)	9000	0x03 (Read holding registers) 0x06 (Write single register) 0x10 (Write multiple registers) 0x17 (Read/Write multiple registers)
	DO16 - DO31	9001	
	...	9002 - 9624	
	DO9984 - DO9999	9625	

Configuration

The SCALE_CONV_XXX and STATUS_OUT_D parameters allow the configuration of the analog I/O and discrete output channels respectively.

- The Modbus register is assigned automatically according to the tables above.
- STATUS_OUT_Dx and STATUS_OUTPUT determine how the status of the OUT_Dx and OUT_x parameters is generated

The status can be generated in one of two ways:

- If the option "Set by master" in the picklist is used, the output status written by the Modbus master in the associated input (IN_Dx, IN_x) or holding register (OUT_Dx, OUT_x) is used.
- If any other option is set, the output status will be set automatically to the status selected, e.g Good_NonCascade: NonSpecific: NotLimited.

In both cases, a communication failure will force the status to Bad: NoCommunication, see Status handling

Scaling

SCALE_CONV_XXX contains additional parameters concerning data type and scaling of analog values:

- DATA_TYPE determines the data format of the value, see Chapter 2.5
- FROM_EU_XX determines the scaling of the "input" parameter
- TO_EU_XX determines the scaling of the "output" parameter

The scaling is performed as shown in Fig. 2-5, whereby the value to be scaled may lie outside the given limits.

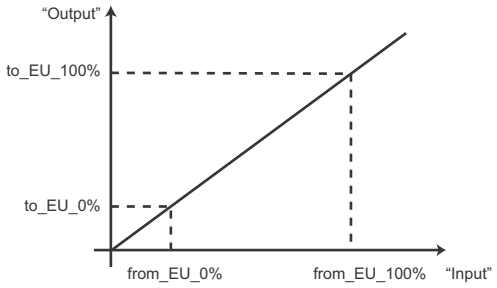


Fig. 2-5: Scaling of raw value to OUT_x value engineering units

The meaning of the scaling parameters in SCALE_CONV_INx and SCALE_CONV_OUTx is as follows:

Parameter	SCALE_CONV_INx	SCALE_CONV_OUTx
.FROM_EU_0	Lower range limit IN_x value	Lower range limit Modbus master value
.FROM_EU_100	Upper range limit IN_x value	Upper range limit Modbus master value
.TO_EU_0	Lower range limit Modbus master value	Lower range limit OUT_x value
.TO_EU_100	Upper range limit Modbus master value	Upper range limit OUT_x value

Status handling

The status of a slave output value is mapped in the corresponding OUT_XX.STATUS parameter. The information it carries is dependent on the entry in STATUS_OUT_Dx or STATUS_OUTPUT and whether the period entered in the TIMEOUT parameter in the MBCF block has elapsed.

STATUS_OUT_Dx or STATUS_OUTPUT	TIMEOUT	Status
Set by master	Slave responding (communicated within timeout period)	As master
	Slave not responding (timeout elapsed)	Bad
Status from pick list	Slave responding (communicated within timeout period)	As set
	Slave not responding (timeout elapsed)	Bad

2.4.2 Block parameters

The table below lists the parameters of the MBCM block and gives a short explanation of their function. Details of use can be found in the appropriate section of the tutorial.

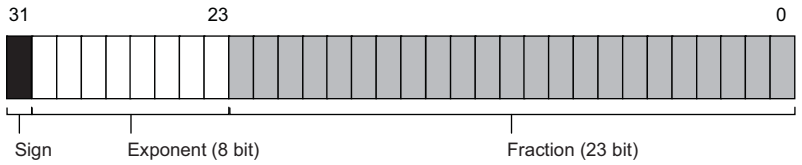
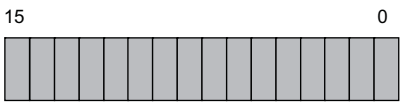
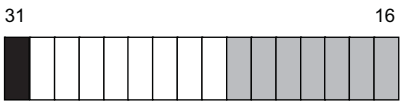
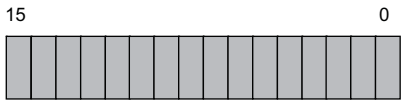
Parameter	Valid range / Options	Default value	Description / Action
ST_VER		0	See Chapter 2.2.2.
TAG_DESC		blanks	
STRATEGY	0 to 255	0	
ALERT_KEY	1 to 255	1	
MODE_BLK	TARGET	O/S	Block mode, set to Auto
BLOCK_ERR	0 to 15		Block errors, see Chapter 2.2.2
LOCAL_MOD_MAP	0 to 15	0	Unique identifier for MBCS block
IN_1			Value and status of analog input 1
SCALE_CONV_IN1			Scaling and data format for analog input signal 1
IN_2			Value and status of analog input 2
SCALE_CONV_IN2			Scaling and data format for analog input signal 2
IN_3			Value and status of analog input 3
SCALE_CONV_IN3			Scaling and data format for analog input signal 3
IN_4			Value and status of analog input 4
SCALE_CONV_IN4			Scaling and data format for analog input signal 4
IN_D1			Value and status of discrete input 1
IN_D2			Value and status of discrete input 2
IN_D3			Value and status of discrete input 3
IN_D4			Value and status of discrete input 4
OUT_1			Value and status of analog output 1
SCALE_CONV_OUT1			Scaling, data format and status for analog output signal 1
OUT_2			Value and status of analog output 2
SCALE_CONV_OUT2			Scaling, data format and status for analog output signal 2
OUT_3			Value and status of analog output 3
SCALE_CONV_OUT3			Scaling, data format and status for analog output signal 3
OUT_4			Value and status of analog output 4
SCALE_CONV_OUT4			Scaling, data format and status for analog output signal 4
OUT_D1			Value and status of discrete output 1
STATUS_OUT_D1			Status for discrete output signal 1
OUT_D2			Value and status of discrete output 2
STATUS_OUT_D2			Status for discrete output signal 2
OUT_D3			Value and status of discrete output 3
STATUS_OUT_D3			Status for discrete output signal 3
OUT_D4			Value and status of discrete output 4
STATUS_OUT_D4			Status for discrete output signal 4
UPDATE_EVT			This alert is generated by any change to the static data
BLOCK_ALM			Block alarms

2.5 Data types

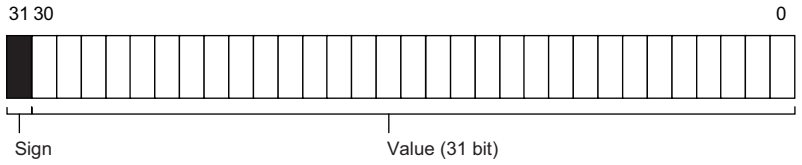
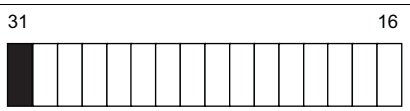
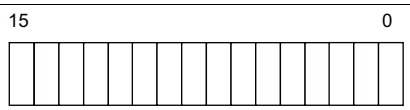
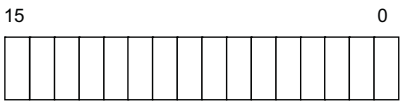
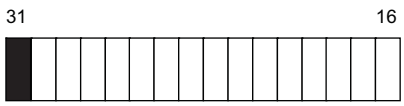
ControlCare Field Controllers support seven different data types, which are interpreted and stored as described below:

- Floating point
- Integer32, Unsigned Integer32, Integer 16, Unsigned Integer16, Integer8, Unsigned Integer8

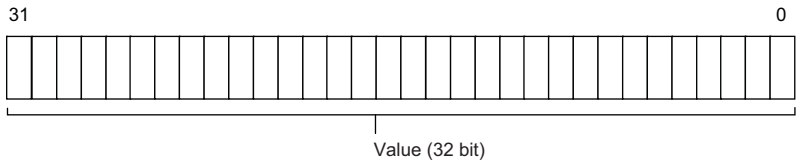
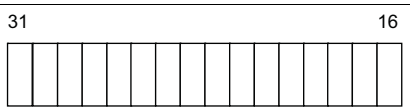
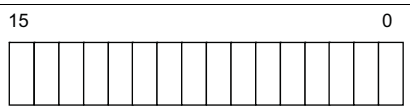
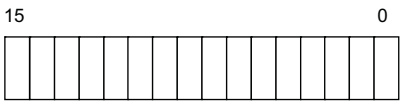
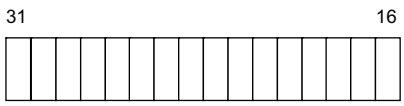
Floating point

Value range	(±) 1.175 * 10^{-38} to 3.403 * 10^{38}	
Byte Structure		
Storage	Float	Swapped Float
Modbus Register 1		
Modbus Register 2		

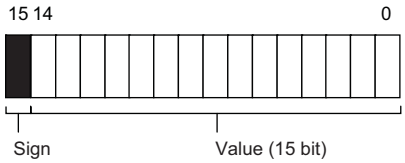
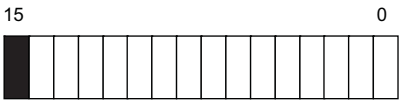
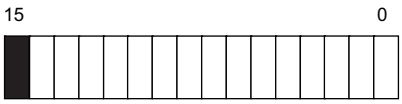
Integer32

Value range	-2147483648 to +2147483647	
Byte Structure		
Storage	Integer32	Swapped Integer32
Modbus Register 1		
Modbus Register 2		

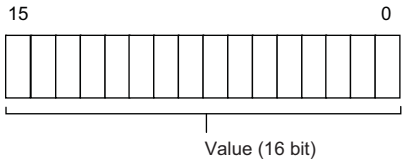
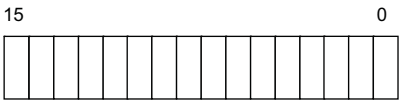
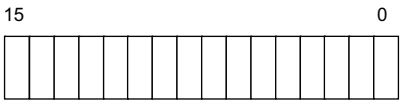
Unsigned32

Value range	0 to 4.294.967.295	
Byte Structure		
Storage	Unsigned Integer32	Swapped Unsigned Integer32
Modbus Register 1		
Modbus Register 2		

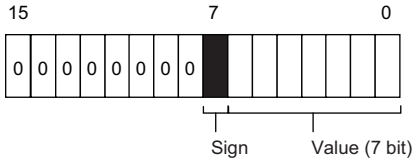
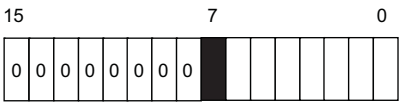
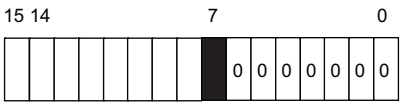
Integer16

Value range	-32.768 to 32.767	
Byte Structure		
Storage	Integer16	Swapped Integer16
Modbus Register 1		

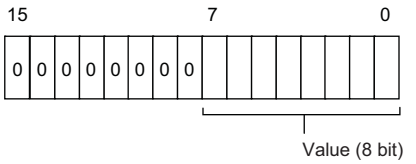
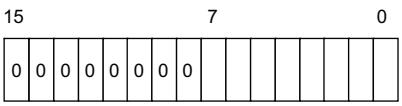
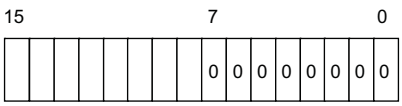
Unsigned16

Value range	0 to 65.535	
Byte Structure		
Storage	Unsigned Integer16	Swapped Unsigned Integer16
Modbus Register 1		

Integer8

Value range	-128 to +127	
Byte Structure		
Storage	Integer8	Swapped Integer8
Modbus Register 1		

Unsigned 8

Value range	0 to 255	
Byte Structure		
Storage	Unsigned Integer8	Swapped Unsigned Integer8
Modbus Register 1		

2.6 System architecture

2.6.1 Use as Modbus master

Serial Modbus

When used in Modbus master mode, the Field Controller can read data from and write data to the Modbus slaves. Fig. 2-6 shows a typical architecture for use with a serial (RTU) Modbus network. Up to 128 serial slaves can be accessed. This application is described in Chapter 3.

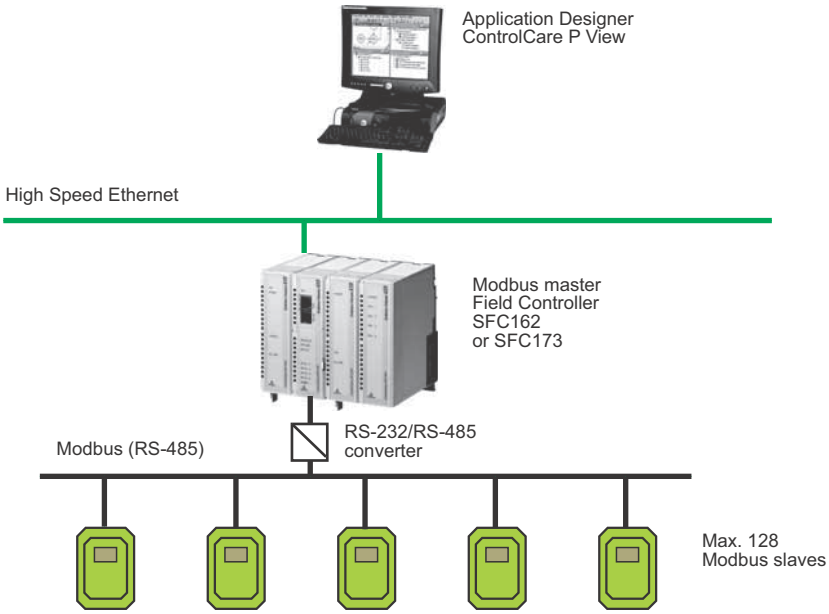
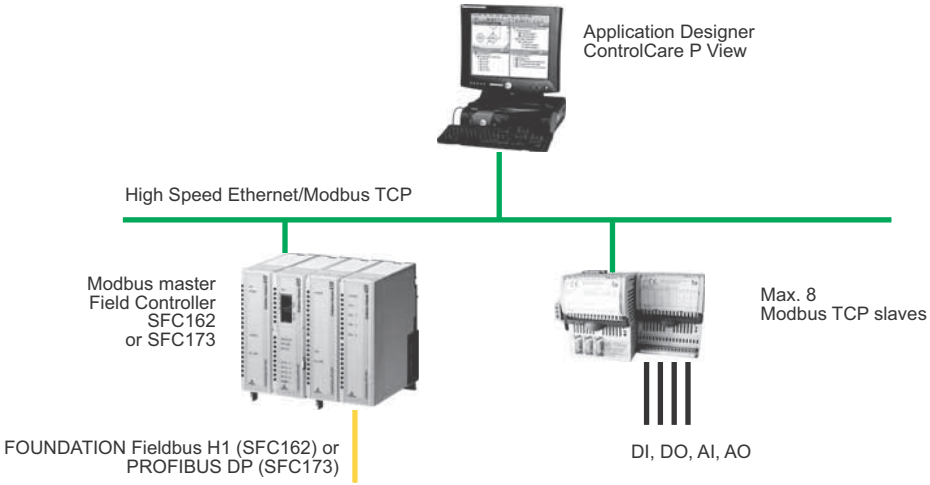


Fig. 2-6 Use of the Field Controller with Modbus RTU

Modbus TCP

Fig. 2-7 shows a typical architecture in which the Field Controller acts as a Modbus Master in a Modbus TCP network. In this case, input and output values are to be accessed via a Remote I/O acting as Modbus slave. Up to 8 TCP slaves can be accessed.



2.6.2 Use as a Modbus slave

Most legacy systems such as DCS or PLC have serial interface modules that support Modbus. When the Field Controller is used as a Modbus slave, it allows a Modbus master, e.g. a PLC or DCS, to access values provided by Fieldbus devices. Communication may be serial or via Modbus TCP.

Figs 2-7 and 2-8 show typical architectures for the connection of FOUNDATION Fieldbus and PROFIBUS networks. This application is described in Chapter 4.

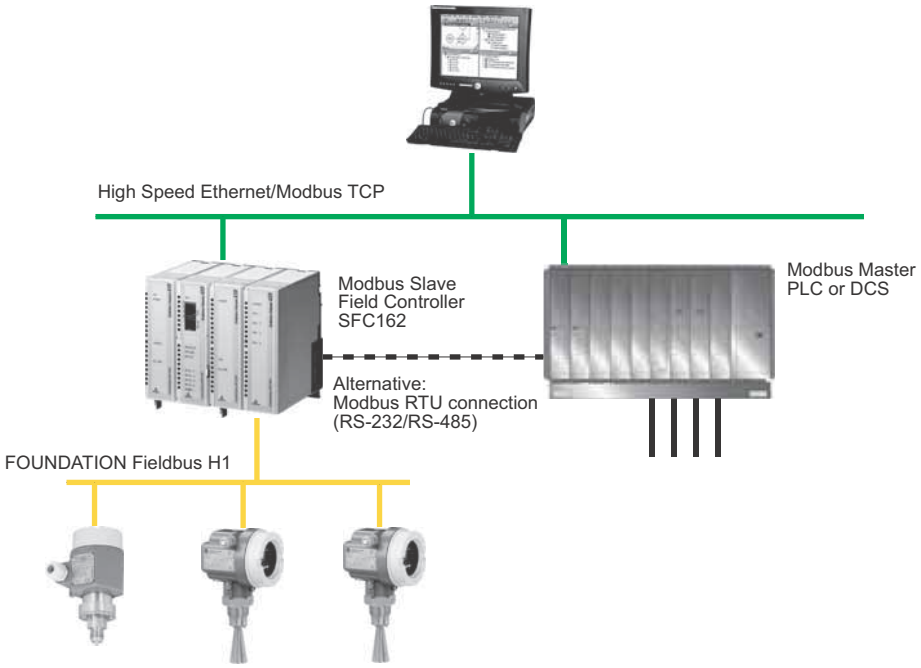


Fig. 2-7 Use of the SFC162 Field Controller to allow legacy systems access to selected FF parameters

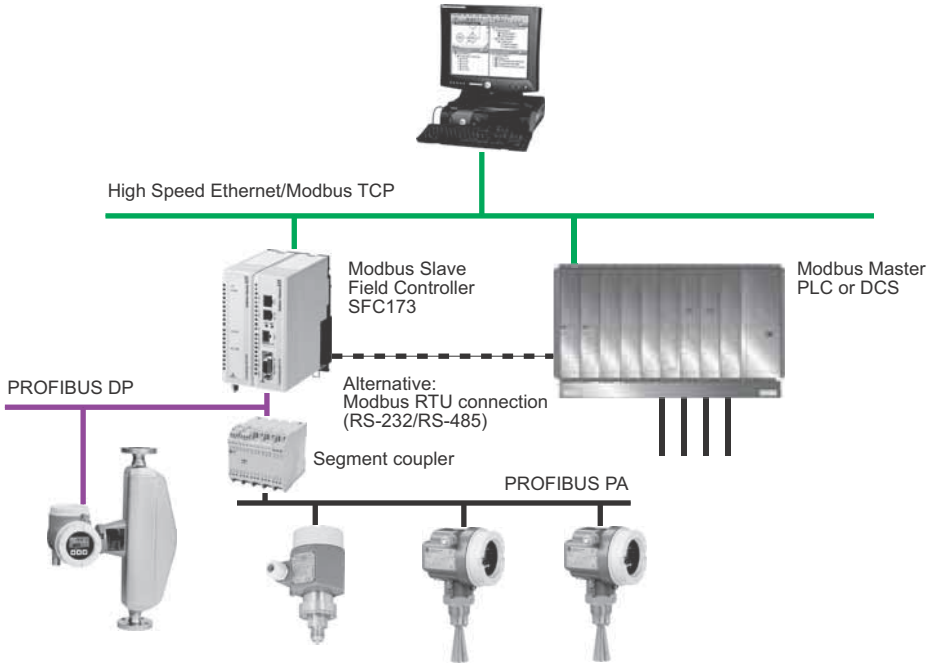


Fig. 2-8 Use of the SFC173 Field Controller to allow legacy systems access to selected PROFIBUS parameters

2.6.3 Use as Modbus master and slave

From version 2.05.xx upwards it is possible to use the FieldController as both master and slave. Such a configuration is required when. for example, the Field Controller uses a Modbus Remote I/O as slave to acquire 4–20mA, temperature and binary signals, but serves itself as slave to a Modbus controller.

Figs 2-9, 2-10 and 2-11 show the three possible architectures for TCP and serial connection

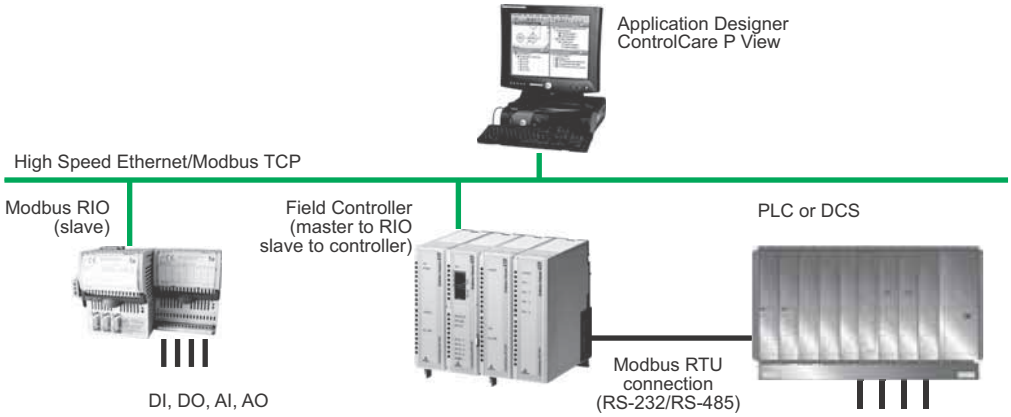


Fig. 2-9 Use of Field Controller as TCP master and RTU slave

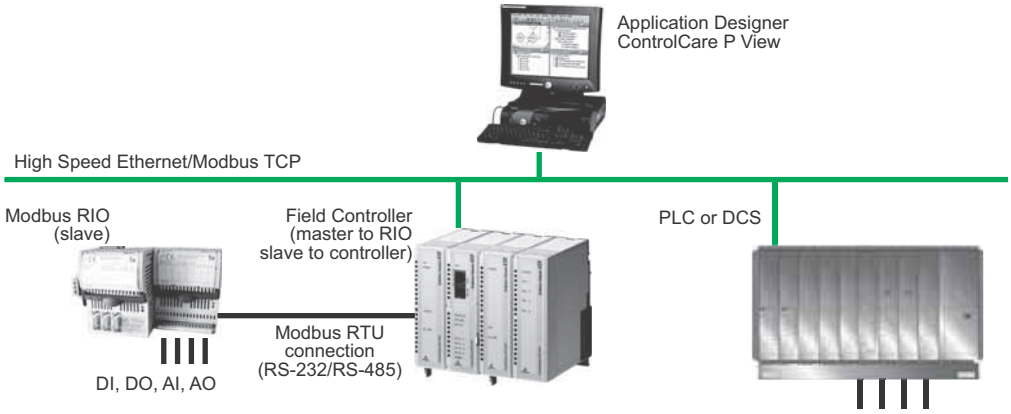


Fig. 2-10 Use of Field Controller as TCP master and RTU slave

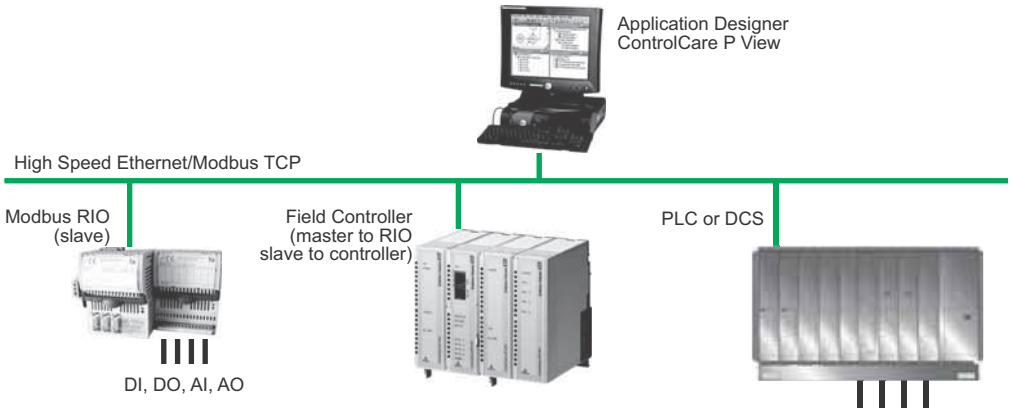


Fig. 2-11 Use of Field Controller as TCP master and TCP slave

3 Field Controller as Modbus Master

3.1 Task Description

This part of the tutorial describes all steps necessary for setting up the Field Controller as a Modbus Control Master. It does not aim to give an exhaustive account of Application Designer functions, but rather shows you one of a number of methods to reach your goal. The tags and names used in the tutorial are imaginary and will be different in a proper application. A full description of Application Designer functions is to be found in Application Designer Overview BA017S/04/en. Function block descriptions are to be found in BA022/04/en, Function Block manual.

3.1.1 Application

For this tutorial, the case of cascade control for a heat exchanger will be used, see Fig. 3-1.

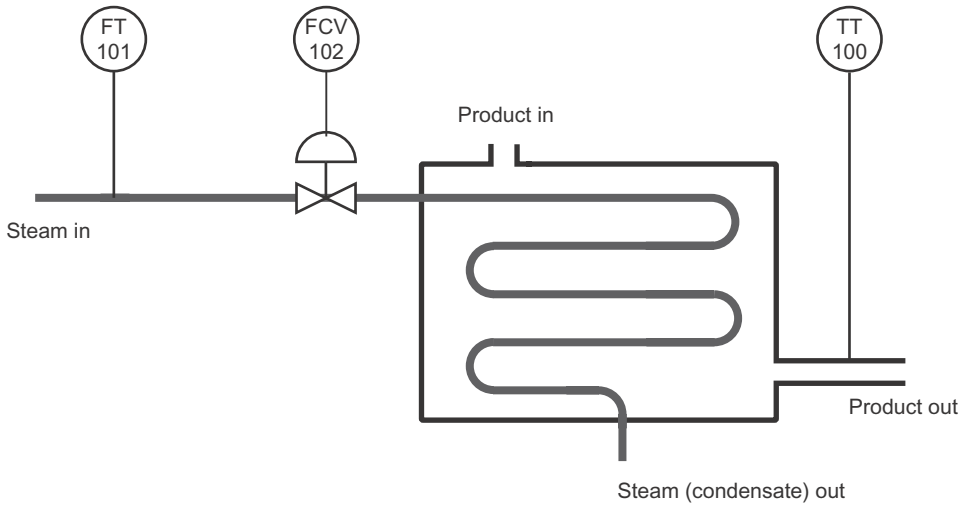


Fig. 3-1: Schematic diagram of heat exchanger application

A liquid flows through the heat exchanger and is heated by condensing steam. The controlled variable is the exit temperature of the liquid flowing through the exchanger. The manipulated variable is the steam flow to the exchanger. The temperature of the product defines the set point of the steam flow, which is controlled by a valve in order to avoid excessive waste of energy (=steam).

The flow values are delivered by a Modbus slave (e.g. flowmeter) via the MBCM block in the Field Controller, which acts as Modbus master. The corresponding control strategy is shown in Fig. 3-2.

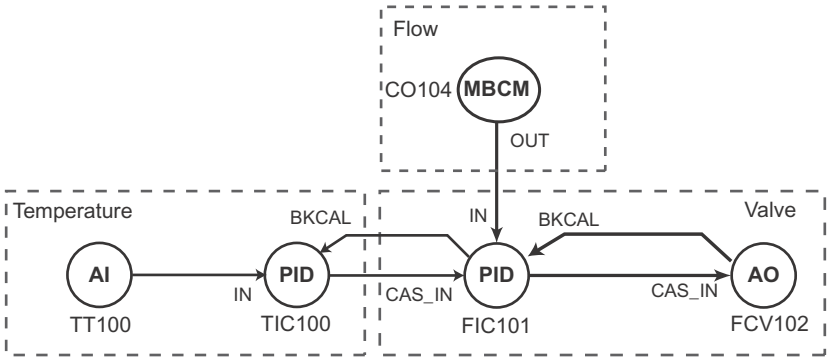


Fig. 3-2 Cascade control strategy for heat exchanger application

3.1.2 Network

The network is assumed to be constructed as shown in Fig. 3-3.

- The flowmeter is a Promass 83F with Modbus RS-485 interface, acting as Modbus slave, suitable for steam measurement
- The valve positioner is a Metso ND9103FN (FOUNDATION fieldbus)
- The temperature transmitter TMT162 (FOUNDATION fieldbus)

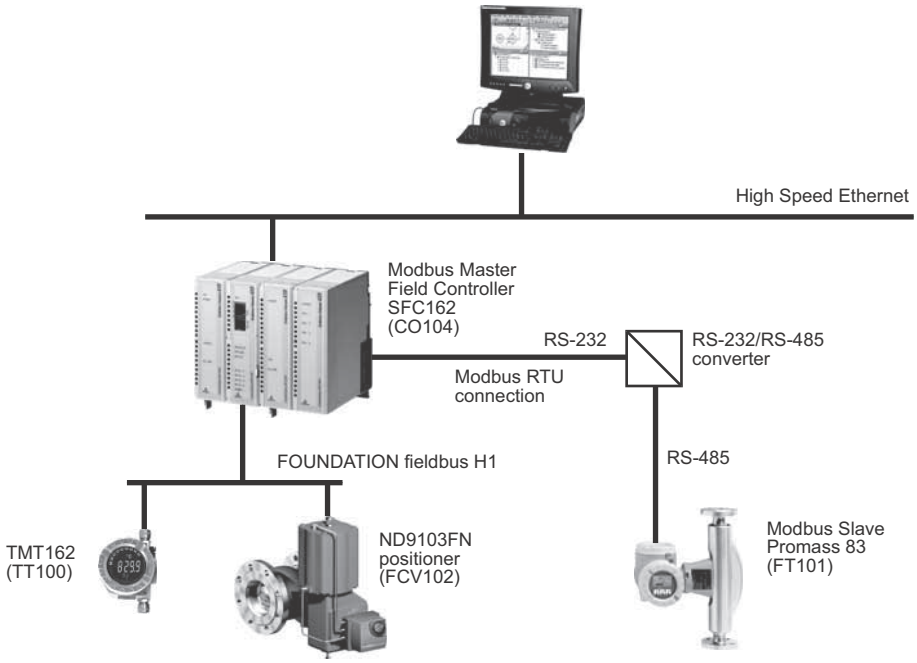


Fig. 3-3 Network for application example

As shown in Figure 3-2, the control will be done in the TMT162 temperature transmitter and the ND9103FN valve positioner. As far as traffic on the bus is concerned, this is the most efficient method. The user has, however, the alternative of performing all control in the controller, should this be preferred.

The Promass 83 delivers the flow measurement for the cascade control. To do this it must be connected to the RS-232 port of controller via a RS-232/RS-485 interface. Both the Field Controller and the flowmeter can be configured as master or slave. In this example, the Field Controller is used as master.

3.1.3 Installation and commissioning

Before you can start this part of the Modbus tutorial, Application Designer must be installed on your computer, the SFC162 FOUNDATION Fieldbus Controller installed and commissioned and a connection made to your computer. Instructions on how to do this are to be found in:

- Operating Instructions BA020S/04/en, Getting Started
- Operating Instructions BA021S/04/en, Field Controller: Hardware Installion
- Operating Instructions BA035S/04/en, Field Controller: Commissioning and Configuration

3.1.4 Device ID and tag

For a FOUNDATION Fieldbus system, each device that communicates has a unique bus address and tag. Addresses are assigned automatically during the start-up of the system on the basis of the device ID. The device ID is a unique identifier that is based on a Manufacturer ID and the serial number of the device. When the project goes online, the actual device IDs must be assigned to virtual devices that have been planned in Application Designer by using the Assign Tags procedure.

To aid the offline engineering of the network, it is necessary to keep a record of the measuring point tags (device tags), often as an Excel sheet. Measuring point tags are used in P&I diagrams to indicate the type of measurement or action performed at a particular location in a process. Table 3-1 below provides an example of how this might look for the application at hand.

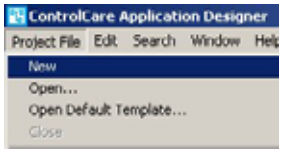
Area	Process Cell	Device	Vendor	Tag	Unit	Task
Pasteurization	Heat Exchanger	TMT162	E+H	TT100 TIC100	°C	Product temperature Temperature PID
Pasteurization	Heat Exchanger	Promass 83F	E+H	FT101	kg/h	Steam flow
Pasteurization	Heat Exchanger	ND9103FN	Metso	FCV102 FIC101	%	Steam valve positioner Flow PID
Pasteurization	Heat Exchanger	SFC162	E+H	CO104		Field Controller acquiring and passing on flow measurement

Tab. 3-1: : Measuring point tag list for tutorial application

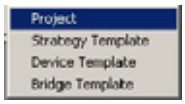
3.2 Create a project

1 Start ControlCare Application Designer by clicking on the icon on your destop or via Start => Programs => Endress+Hauser => ControlCare => ControlCare Application Designer

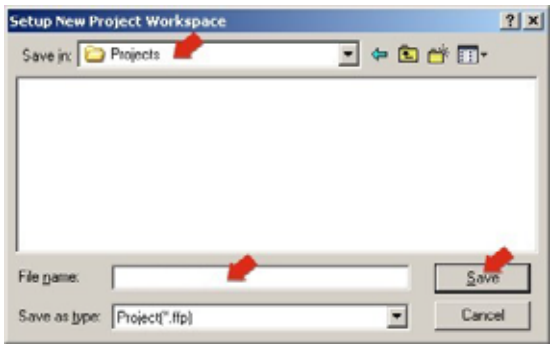
2 The project starts from a blank application screen
– With the right mouse key select Project File=>New



1 The Document Type box appears: Click the option Project



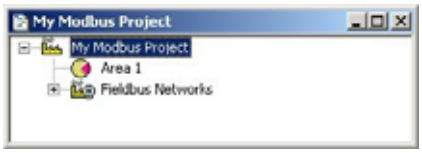
2 The New Project dialog box opens:



1. Choose the folder where the project will be saved.
2. Type the name of the project in the File Name box, e.g. My Modbus project.
3. Click Save.
   If the new project is not to be created, click Cancel.

3 ControlCare Application Designer automatically creates a folder with the entered file name within the selected folder.

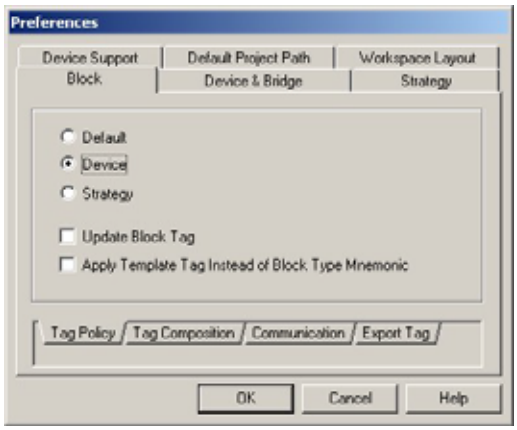
4 The project opens with the first branches of the plant and network view already created:



3.3 Determine the naming preferences

Before you start, you can set preferences for the way your project is created. Of particular interest at this stage is the labelling of the function blocks.

1 Press Project File => Preferences
  – The Preferences Dialog appears



Tag Policy

Tag Policy determines how the blocks are labelled by default if no tag names are entered

1 Select the folder Block and the subfolder Tag Policy, then activate the following (check box)
  – Device
  – Update Block Tag

2 Press OK to confirm your selection
  – Application Designer will now automatically rename any blocks created in the control strategy window as they are assigned to the devices by adding the device tag as prefix.

Tag Composition

Tag Composition determines how the block identifiers are added to the block tag if no block name is entered.

1 Select the subfolder Tag Composition:
  – Enter a mnemonic separator: for this manual the setting was "-"
    Default setting is "_" and mandatory for if flexible function blocks are to be used
  – Check Prefix

2 Press OK to confirm your selection
  – Application Designer will now automatically compose the blocks according to your selection, e.g. TagName-Block-n or TagName_Block_n.

Export Tag

Export Tag causes tags to be automatically exported every time the project goes online

Select the subfolder Export Tag
– Check the Automatic button
– Press OK to confirm your selection

Strategy

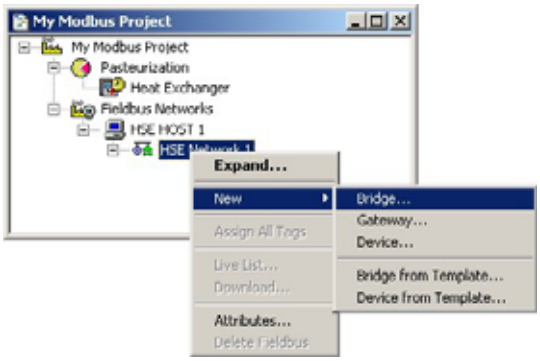
Strategy determines the default shape of the function block icons in the stratagy window and also whether the aliasing function is enabled

1 Select the subfolder Strategy
  – Select the default shape for function block objects
  – Select "Aliasing Input Dialog Box" if you want to use your own input and output labels in the strategy
  – Press OK to confirm your selection

3.4 Create a fieldbus network

3.4.1 Add the controller

1 Expand the Fieldbus Networks branch in the project window and right-click on HSE Network 1



– Select New => Bridge to add the Field Controller SFC162
– If the Field Controller SFC173 was to be used, it would be added by New => Gateway

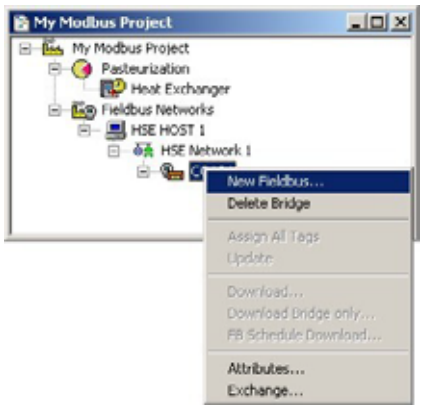
2 The New Bridge dialog opens

– Enter the Device Tag, e.g. CO104
– Press OK to create the bridge

3 The Field Controller SFC162 is added to the network

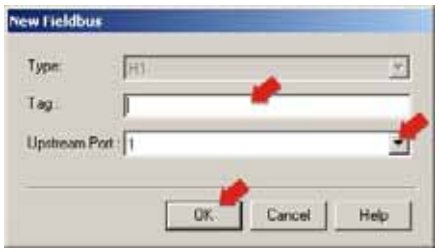
3.4.2 Add a fieldbus segment

1 Right-click on the Field Controller (CO104)



– Select New Fieldbus

2 The New Fieldbus dialog opens

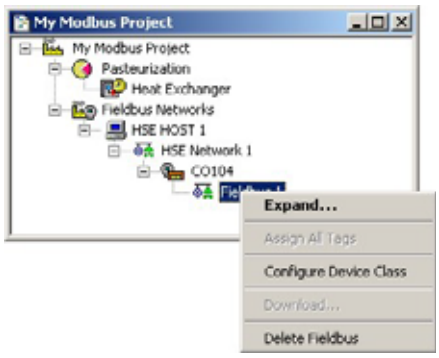


– Press OK to create the fieldbus with default settings

3 The fieldbus is created with the default name Fieldbus 1

3.4.3 Add the Modbus function blocks

1 Right-click on the **Fieldbus 1** leaf and select **Expand**



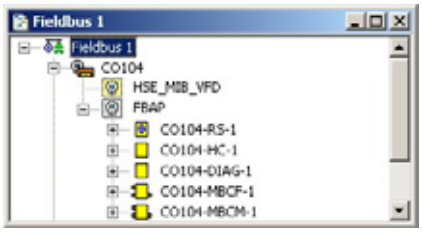
2 A new window opens with the name **Fieldbus 1**
– Expand the tree until all leaves until you see **FBAP** under the Field Controller

– Right-click on **FBAP** and select **New Block**

3 The **New Block** dialog for the SFC162 Field Controller opens

– In **Block Type**, select the function block **Modbus Configuration**
– Press **OK** to create the block with default values

4 Repeat Step 2 and create a **Modbus Control Master** block
– Depending on the number of inputs or outputs required, up to 16 MBCM blocks can be created)

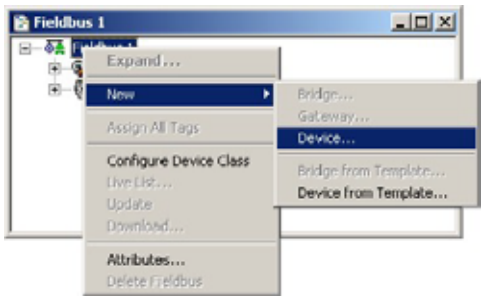


3.4.4 Add the FF field devices

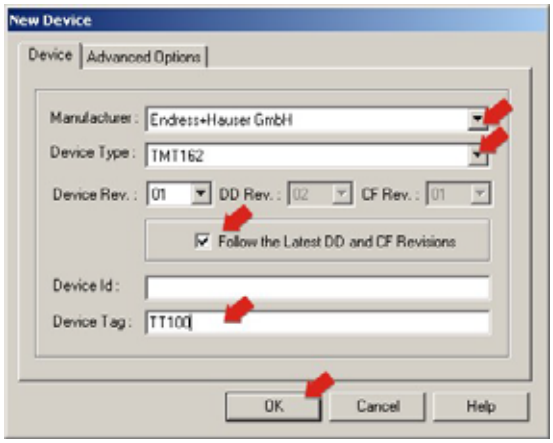
Note!

- Only FOUNDATION Fieldbus devices are added to the Fieldbus network, Modbus devices are not shown in this tool

1 In the Fieldbus 1 window, right-click on the Fieldbus 1 leaf and select New => Device



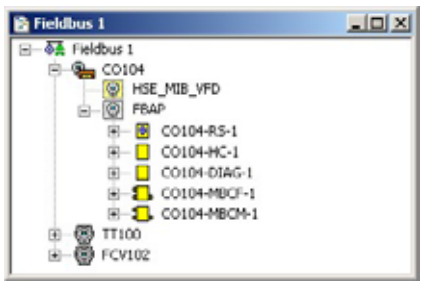
2 The New Device dialog appears



- Select Manufacturer: Endress+Hauser
- Select Device Type: TMT162
- Enter Device Tag: TT100
- Press OK to create the device

3 Repeat Step 2 to create the Metso positioner
- Select Manufacturer: Metso Automation
- Select Device Type: Metso FBLK Interface
- Enter Device Tag: FCV102
- Press OK to create the device

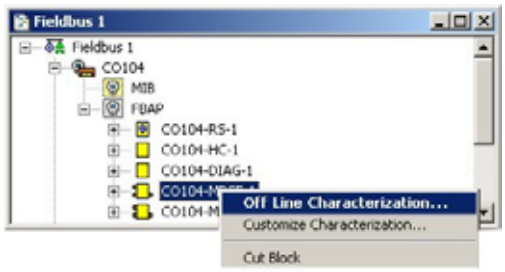
4 Fieldbus 1 now looks like this



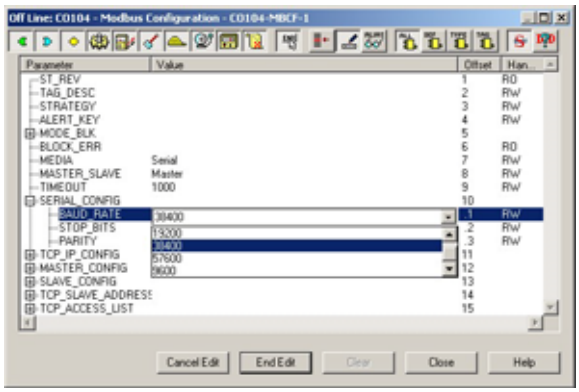
3.5 Configure the devices

3.5.1 MBCF Modbus Configuration block

1 In the Fieldbus 1 window, right-click on the Field Controller function block CO104-MBCF-1 and select Off Line Characterization



2 The Off Line Characterization dialog opens: Press All to show all parameters



3 Set the following parameters by double-clicking in the middle of the value space in the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCF
MODE BLOCK.TARGET	Normal operating mode of block	Auto
MEDIA	Channel for Modbus communication	Serial (+TCP/IP)
MASTER_SLAVE	Role of Field Controller in Modbus network	Master
TIMEOUT	Time allowed for slave response If there is no response, the slave status is set to BAD	1000
SERIAL_CONFIG BAUDRATE STOP_BITS PARITY	Configures serial interface (default for Promass 83) Baudrate used for communcation Number of stop bits used in telegram Parity used in telegram	 38400 1 Even
MASTER_CONFIG NUMBER_OF_RETRIES	Configures Controller when acting as master Number of retransmits if no response from a slave	 3

4 Click Close to close the dialog: the parameters are added to the MBCF function block

3.5.2 MBCM Modbus Control Master block

The MBCM function block is described in detail in Chapter 2.3. Each MBCM block allows 16 Modbus registers to be accessed via its input and output parameters as follows:

- Four OUT channels read analog values from Modbus slave holding or input registers
- Four OUT_D channels read discrete values from Modbus slave coils or discrete input registers
- Four IN channels for write analog values to Modbus slave holding registers
- Four IN_D channels for write discrete values to Modbus slave coils

SCALE_LOC_XXX and LOCATOR_XXX allow each channel to be individually configured for slave address, register and in the case of analog values, data type and scaling.

Up to 16 MBCM blocks can be created, each having a unique identifier (0 – 15) determined by the LOCAL_MOD_MAP parameter.

The operating instructions of the Modbus device indicates which values are available in which registers. In the case of the Promass 83, several measured values are offered which can be read from different MODBUS registers, see below. For our example, mass flow (register 2007) will be selected.

Measured value	Register 1	Register 2	Data type	Access
Mass flow	2007	247	Float	Read
Volume flow	2009	253	Float	Read
Corrected volume flow	2011	–	Float	Read
Density	2013	249	Float	Read
Reference density	2015	–	Float	Read
Temperature	2017	251	Float	Read
Totalizer 1	2610	259/261	Float	Read
Totalizer 2	2810	–	Float	Read
Totalizer 3	3010	–	Float	Read

Tab. 3-2: Modbus registers for Promass measured values

Field Controller uses reference addresses to communicate with the Modbus slave, see Fig. 2-1, Chapter 2.2.1. The input registers containing the analog value have a reference address range of 30001 to 39999, whereby the reference address is:

- 30001 + input register address, when the input register addresses are zero based or
- 30000 + input register address, when the input register addresses are based on 1

The registers in Endress+Hauser flow devices are based on one, thus the reference address for the mass flow value is 32007.

By default, the Promass 83 sends its float number with the bytes order 1-0-3-2 which corresponds to the DATA_TYPE "float" in the Field Controller, see Chapter 2.5.

The values mapped to the Field Controller or sent to a Modbus slave device register can be scaled in SCALE_LOC_XXX with the parameters:

- FROM_EU_0: lowest value of mass flow that the Promass 83 is adjusted to measure
- FROM_EU_100: highest value of mass flow that the Promass 83 is adjusted to measure
- TO_EU_0: lower range limit of the scaled value for the Field Controller
- TO_EU_100: lower range limit of the scaled value for the Field Controller

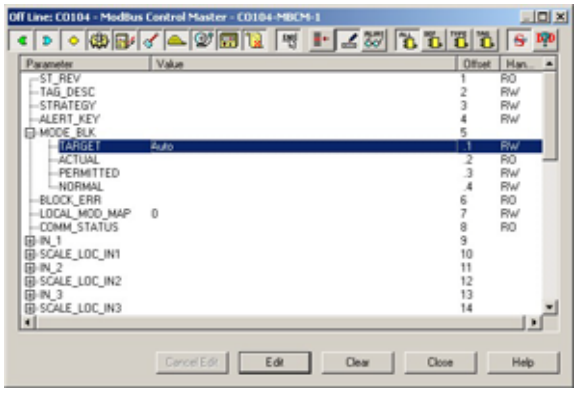
In our example, the mass flow values 0 kg/h to 8000 kg/h offered by the Promass 83 will be scaled from 0% to 100% by entering the range limits of the transmitter in the "FROM" parameters.

Note!

- Full details of how to parametrize the Promass 83 flow transmitter with Modbus slave interface are to be found Operating Instructions BA107D/06/en and BA108D/06/en respectively.

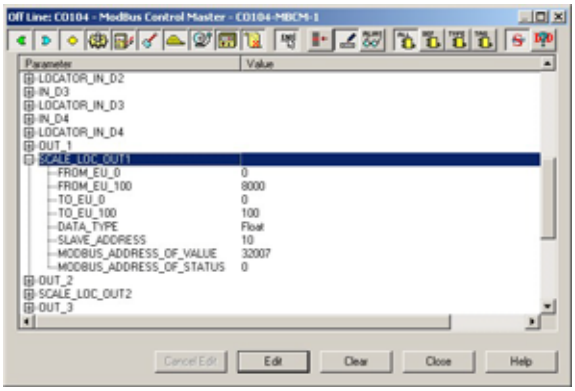
Procedure

1 Right-click on the CO104-MBCM-1 block and open the Off Line Characterization dialog

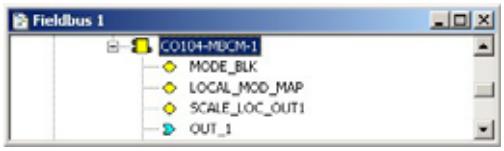


2 Click All to reveal all parameters and enter the following parameters by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCM
MODE BLOCK.TARGET	Normal operating mode of block	Auto
LOCAL_MOD_MAP	Identifier of Modbus block (first MBCM block)	0
SCALE_LOC_OUT1	Scaling and conversion of OUT_1 (flow)-	
FROM_EU_0	Lower range limit of Promass signal	0 (Kg/h)
FROM_EU_100	Upper range limit of Promass signal	8000 (kg/h)
TO_EU_0	Lower range limit of OUT_1	0%
TO_EU_100	Upper range limit of OUT_1	100%
DATA_TYPE	Type of data transmitted	float
SLAVE_ADDRESS	Modbus address of variable source (10 = Promass default)	10
MODBUS_ADDRESS_OF_VALUE	Reference address of input register	32007



3 Press Close to close the Off Line Characterization dialog. You should now see the parameters attached to the MBCM block:



4 Open Project File, then press Save, to save the project.

3.5.3 TMT162 transducer block

Full details of how to parametrize the TMT162 temperature transmitter are to be found in Operating Instructions BA224R/09/en.

Transducer Block

Table 3-3 shows the parameters that must be set in the TMT162 TEMP_1 transducer block

Parameter	Function	Temperature TT100
MODE BLOCK/TARGET	Normal operating mode of block	Auto
PRIMARY_VALUE_TYPE	Calculation method for primary process value • Process temperature SV1 or SV2 • Average 0.5 (SV1 + SV2) with/without redundancay • Differential (SV1 - SV2) • Conditional (SV1 or SV2), (SV2 if SV1 >T)	Sensor Value 1
SENSOR_TYPE	Type of sensor connected to the transmitter • All types of standardized temperature sensors	Pt 100 IEC 751
SENSOR_CONNECTION	Way in which the sensor is connected • 4-wire (if two sensors are connected only one can be 4-wire) • 3-wire • 2-wire	2-wire

Tab. 3-3: Basic parameters for TMT162 transducer block

Order of parameters

Some block parameters have a write check based on the value of others parameters. It is therefore important to set the parameters in the order shown in Table 3-3 (the same order in which they are displayed in the Off Line Characterization dialog. After parametrization of the block, the parameters will appear in the FOUNDATION Fieldbus tree. If you find a parameter in the wrong position, it can be move by dragging and dropping to the correct one.

Parametrize

1 In the Fieldbus network workspace, expand the TT100 tree until the function blocks are visible
   – Right click on the transducer function block TT100-BLK-1 and select Off Line Characterization...

2 The Off Line Characterization dialog opens: Press All to reveal all parameters.

3 Now set the following parameters to the values in Table 3-3 by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:
   – MODE_BLK TARGET = Auto
   – PRIMARY_VALUE_TYPE = PV = Sensor value 1 (SV1)
   – SENSOR_TYPE = Pt 100 IEC (a=3.85 E-03)
   – SENSOR_CONNECTION = 2-wire

4 Press Close to quit the Off Line Characterization dialog

3.5.4 Metso ND9103FN positioner

Full details of how to parametrize the Metso ND9103FN valve positioner are to be found in the ND900F User Guide which can be downloaded from www.metso.com.

Transducer Block

In the tutorial, you have probably only the valve positioner, but not the valve itself. In real life, the positioner must be told what it is driving and certain assembly information must always be entered into the transducer block. Rotary valve parameters are included in Table 3-4.

Parameter	Function	Positioner FCV102
MODE BLOCK/TARGET	Normal operating mode of block	Cas
VALVE_TYPE	Type of valve the positioner is actuating • Select from drop-down menu	Rotary
FINAL_VALUE_RATE_DN	Maximum travel rate in closing direction • 0 = parameter not in use	0
FINAL_VALUE_RATE_UP	Maximum travel rate in opening direction • 0 = parameter no in use	0
POSITIONER_FAIL_ACTION	Action of position on loss of electrical power or reception of an output signal with a bad status • Select from drop-down menu	Close
POS_SENSOR_ROT	Relationship between valve action and position sensor rotation • Select from drop-down menu	Standard: Clockwise to close
DEAD_ANGLE_COMP	Dead angle for segment and rotary valves	0
ACT_TYPE	Type of positioner action • Select from drop-down menu	Double-acting actuator
PERFORMANCE_LEVEL	Target performance level of valve position control Select from drop-down menu	Optimum
CHAR_TYPE	Type of linearization • Select from drop-down menu	No characterization

Tab. 3-4: : Basic parameters for Promass 83F transducer block

Order of parameters

Some block parameters have a write check based on the value of others parameters. It is therefore important to set the parameters in the order shown in Table 3-4 (in the same order in which they are displayed in the Off Line Characterization dialog). After parametrization of the block, the parameters will appear in the FOUNDATION Fieldbus tree. If you find a parameter in the wrong position, it can be move by dragging and dropping to the correct one.

Parametrize

1 In the Fieldbus network workspace, expand the FCV102 tree to reveal the function blocks.
   – Right click on the transducer function block FCV102-BLK-2 block and select Off Line Characterization...

2 The Off Line Characterization dialog opens: Press All to reveal all parameters
   – Expand the Mode Block parameter tree

3 Now set the parameters to the values in Table 3-4:
   – Double-click on the "Value space" next to the parameter
   – Enter a value or select a parameter from the drop-down menu
   – Press End Edit to register your change

4 Press Close to quit the Characterization dialog

5 Open Project File, then press Save, to save the project

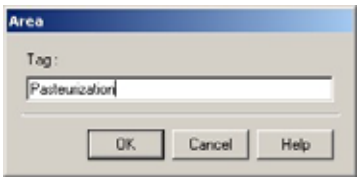
3.6 Create the Control Strategy

Having created a physical view of the process, the next step is to create control strategy. This is done in the logical view of the plant. This represents the plant as Areas/Process Cells in accordance with ISA S88/IEC 61518.

3.6.1 Add a Process Cell

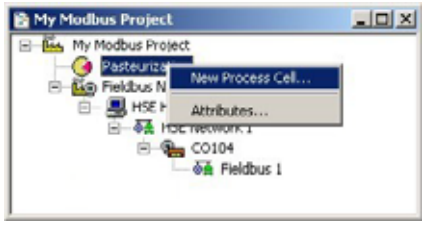
1 Click on the "Area 1" leaf in the project and select Attributes...

2 The Attributes dialog box appears

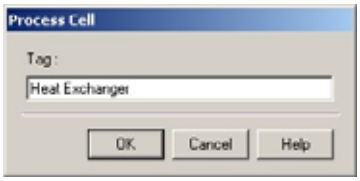


– Enter a name for the area, e.g. Pasteurization (see Table 3-1, Chapter 3.1)
– Click OK to store your changes

3 Click on the Area leaf again and select New Process Cell...



4 The Process Cell dialog box appears

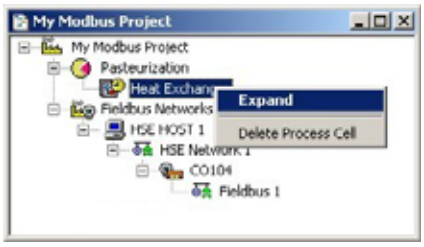


– Enter a name for the process cell, e.g. Heat Exchanger (see Table 3-1, Chapter 3.1.4)
– Click OK to store your changes

5 Open Project File, then press Save, to save the project.

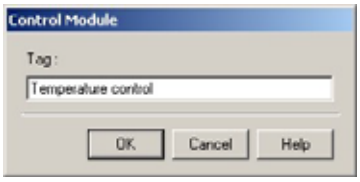
3.6.2 Add a Control Module

1 Right-click on the Process Cell leaf you just created and select Expand



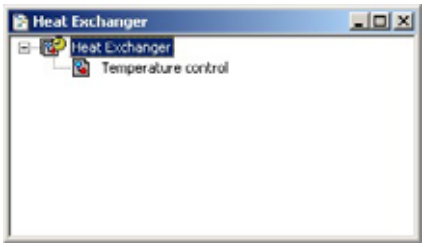
1 A new window with the name of the leaf opens
– Right-click on the top leaf and select New Control Module

2 The Control Module dialog box appears



– Enter a name for the control module, e.g. Temperature Control (see Table 3-1)
– Click OK to store your changes

3 The project now looks something like this:



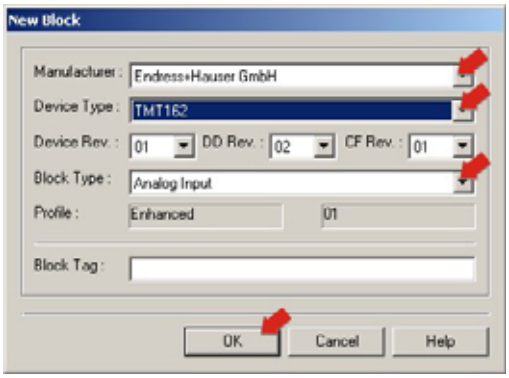
4 For a real project, Step 2 and 3 would be repeated until all the required control modules for a particular process cell have been added.

5 Open Project File, then press Save, to save the project.

3.6.3 Create the function blocks

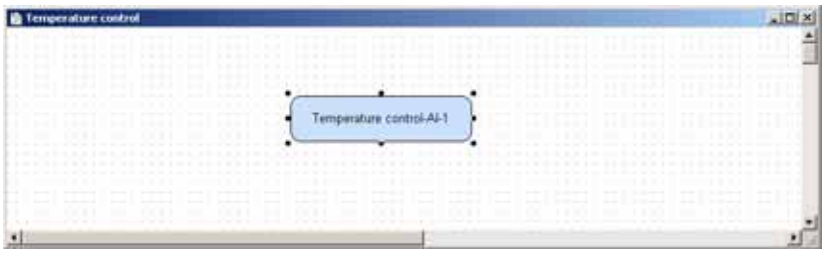
1 Double-click on the control module leaf or right-click and select Expand to open the Control Strategy workspace - this has the same name as the leaf

2 Press the Function Block button in the toolbar and click in the workspace
– The New Block dialog appears



– Select the Manufacturer = Endress+Hauser
– Select the Device Type = TMT162
– Select the Block Type = Analog Input
– Press OK to create the function block

3 The block now appears in the strategy window with the default name



4 Repeat Steps 2 and 3 for the Temperature PID, Flow PID and Valve AO blocks

– Temperature PID:
Manufacturer = Endress+Hauser
Device Type = TMT162
Block Type = PID Control

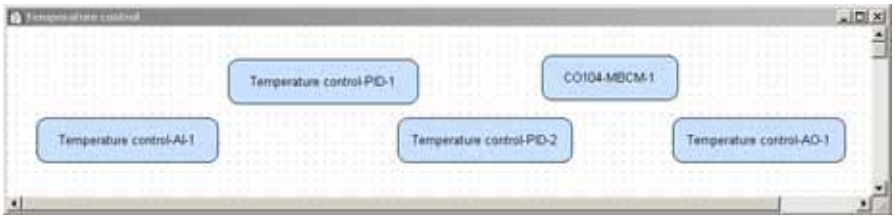
– Flow PID
Manufacturer = Metso Automation
Device Type = FBLK Interface
Block Type = PID Control

– Positioner AO
Manufacturer = Metso Automation
Device Type = FBLK Interface
Block Type = Analog Output

5 The Modbus master/slave data exchange is managed by the MBCM block which you have already created

– In the Fieldbus 1 window expand the CO104 leaf and click on CO104-MBCM-1
– Drag and drop the block into the control strategy window

6 The control strategy now looks like this



7 Open Project File, then press Save, to save the project

Note!

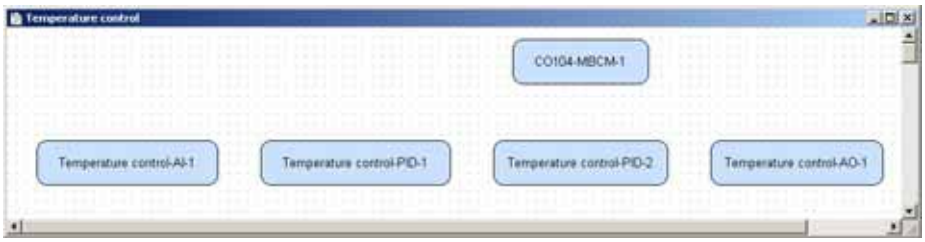
• In the tutorial we created the MBCM block together with the MBCF block. Is is also possible to create the MBCM block in the strategy window as we have done with the PID, AI and AO blocks:

Modbus Control Master MBCM
Manufacturer = Endress+Hauser
Device Type = SFC162 (or SFC173 with PROFIBUS Field Controller)
Block Type = Modbus Control Master

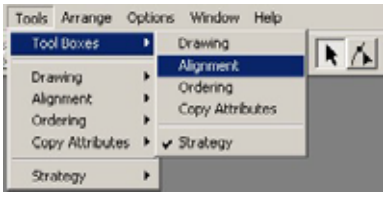
Then drag and drop it to the Field Controller in the Fieldbus Network view, see Chapter 3.6.7.

3.6.4 Add the Function Block Links

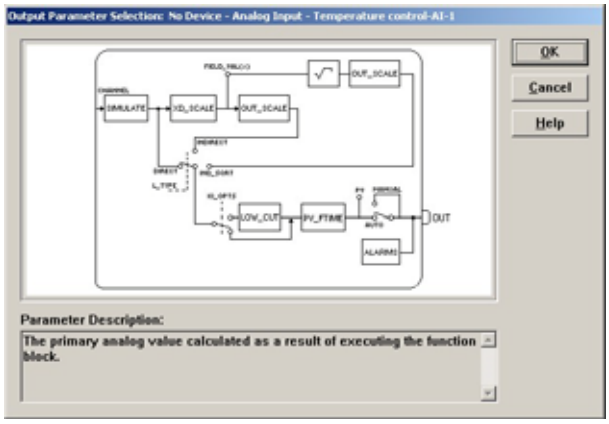
1 In the Control Strategy workspace position the blocks according to your strategy



- The blocks can be dragged and dropped by selecting and holding down the right mouse key
- The blocks can be aligned by selecting, then via Tools => Alignment => e.g. Middle followed by a click on the block to which the alignment is to be made
- The Tools menu also contains other standard drawing functions such as toolbars, standard shapes, line thickness, colours etc.



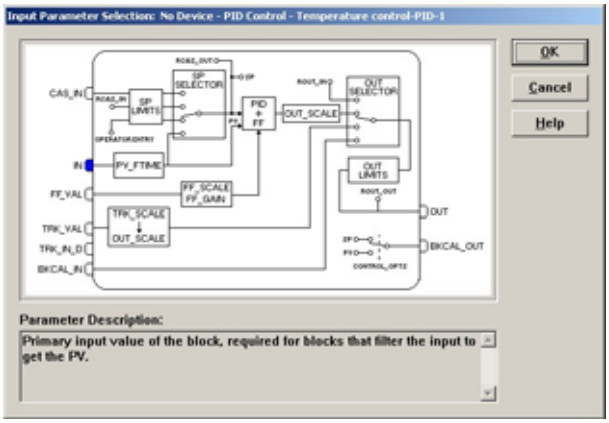
2 Click on Function Block Link button in the tool bar, the cursor changes to a cross
- Select the Temperature AI block with the cross: the Output Parameter Selection dialog appears



3 Click the box next to OUT – it changes color – then click on OK

- The Output Parameter Selection dialog closes
- The cursor is now connected to a blue dotted line
- Place the Cursor in the Controller PID Block 1 and click to make the link

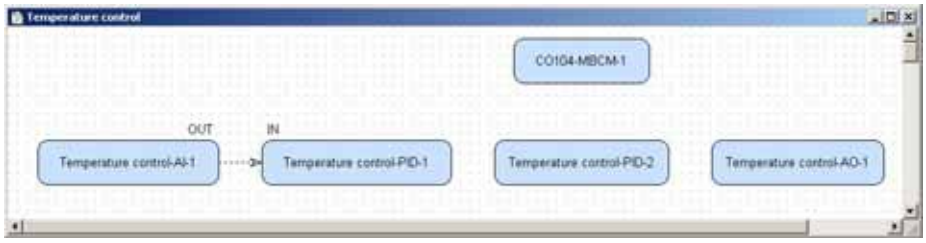
4 When the link is made, the Input Parameter Selection dialog for the PID block appears



– Click the box next to IN – it changes color – then click on OK

5 If the Aliasing Input dialog box is enabled, see Chapter 3.3, the Rename dialog now appears
– Enter the desired link name and press OK
– If nothing is entered, the link retains the standard name

6 When the Input Parameter Selection dialog changes, the link is made and appears as below:

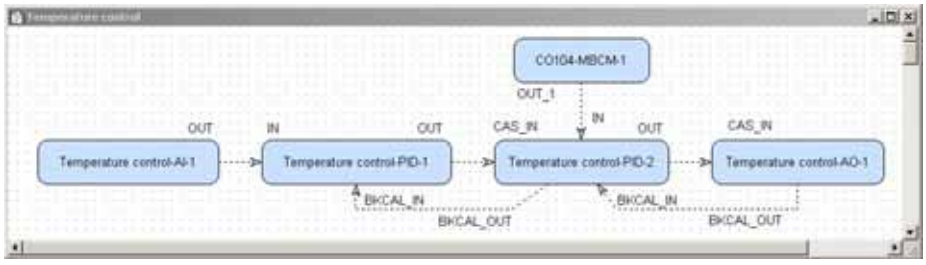


– You may have to move the parameter legends "IN" and "OUT" by selecting and positioning with the left mouse key depressed

7 Repeat steps 2 to 5 and make the following links between the function blocks

– PID1 and PID2 = OUT to CAS_IN
– PID2 and PID1 = BKCAL_OUT to BKCAL_IN
– CO104-MBMC-1 to PID2 = OUT1 to IN
– PID2 to Valve AO = OUT to CAS_IN
– Valve AO to PID2 = BKCAL_OUT to BKCAL_IN

8 Your Control Strategy now looks something like this



9 Open Project File, then press Save, to save the project.

3.7 Configure the strategy

FOUNDATION Fieldbus offers the possibility of storing complete control strategies as fully configured generic templates. This is especially useful when particular control strategies occur several times within a project. The strategies are stored independent of device assignment, which is performed as a separate step, see Chapter 3.8.

In this tutorial, the strategy will be configured to receive the incoming process value from the input block as % of full range, and to output a % value to the positioner.

3.7.1 Analog Input parameters

The function block Temperature control- A1-1 for the TMT162 has to be configured. The basic parameters required are shown in Table 3-5. A full description of the parameters are to be found in the Operating Instructions BA224REN.

Parameter	Function	Temperature TC100
MODE BLOCK/TARGET	Normal operating mode of block	Auto
XD_SCALE/EU_100* XD_SCALE/EU_0 XD_SCALE/UNITS_INDEX	Upper range value for process variable Lower range value for process variable Unit of process variable	150 (max.850) –50 (min. –200) °C
OUT_SCALE/EU_100 OUT_SCALE/EU_0 OUT_SCALE/UNITS_INDEX	Upper range limit for output variable Lower range limit for output variable Unit of output variable	100 0 %
CHANNEL	Output channels of Transducer Block aasigned to Ananlog Input Block. • Primary, RJ or Sensor vlaue 1/2 depending on whether one or two sensors are connected	Sensor Value 1
L_TYPE	Selects the type of linearisation for the input value. • Direct: PV value = OUT value, Identical XD_SCALE and OUT_SCALE • Indirect: PV value scaled to OUT value • Indirect Square Root: as Indirect but scaling with root function	Indirect
PV_FTIME	Output damping constant (in seconds).	1
*The range limits for the TMT162 temperature transmitter are determined by the transducer block parameters SENSOR_TYPE and PRIMARY_OUTPUT_TYPE. For SENSOR_TYPE = Pt100 and PRIMARY_OUTPUT_TYPE = SV_1 the transducer block outputs a temperature signal in the range -200°C to +850°C. The XD_SCALE and OUT_SCALE parameters generate the OUT value of the Anlaog Input block from any part of this range, in our case -50°C to 150°C.		

Tab. 3-5: Basic parameters for Analog Input blocks

Order of parameters

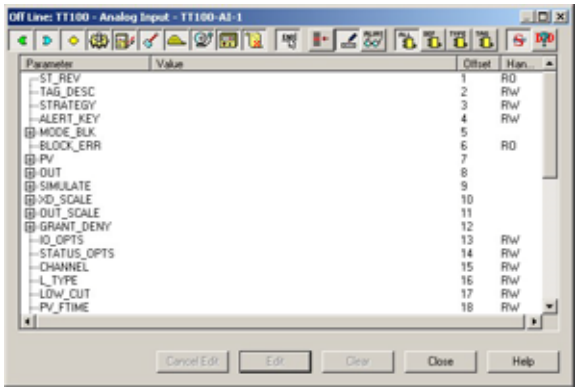
Some block parameters have a write check based on the value of others parameters. It is therefore important to set the parameters in the order shown in Table 3-5 (in the same order in which they are displayed in the Off Line Characterization dialog). After parametrization of the block, the parameters will appear in the FOUNDATION Fieldbus tree. If you find a parameter in the wrong position, it can be move by dragging and dropping to the correct one.

Note!

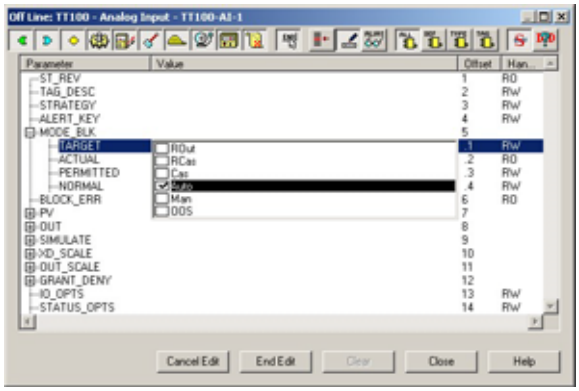
- In the tutorial we configured the MBCM block together with the MBCF block, see Chapter 3.5.2. If the MBCM block had been created in the strategy window as we have done with the PID, AI and AO blocks, it could also be configured in the same manner has described here.

3.7.2 Configuring the Analog Input blocks

1 In the Control strategy workspace, double-click on the **Temperature control AI-1** block
   – The **Off Line Characterization** dialog opens
   – Press **All** to display all the parameters



2 The Mode Block **Target** must be set to **Auto**.



   – Expand the **Mode Block** parameter tree
   – Double-click on the "Value space" next to the **Target** parameter
   – Select **"Auto"** from the drop-down menu
   – Press **End Edit** to register your change

3 Repeat this procedure for the remainder of the Temperature parameters in Table 3-5
   – Remember to press **End Edit** after every change
   – When all parameters have been entered, press **Close** to quit the Characterization dialog

4 Open **Project File**, then press **Save**, to save the project

3.7.3 Basic PID parameters

The controller PID blocks must now be parametrized. In practice, the values for the GAIN, RESET and RATE as well as the setpoint value SP for the temperature loop will be known. The other values determine the the way the control is handled when the block moves from auto/cas or a value becomes bad. See also ControlCare Function Block manual BA022S/04/en.

Parameter	Function	Temperature TC100	Flow FC101
MODE BLOCK/TARGET	Normal operating mode of block	Auto	CAS
SP/VALUE	Setpoint for product temperature	40%	–
PV_SCALE/EU_100 PV_SCALE/EU_0 PV_SCALE/UNITS_INDEX	Upper range limit for process variable Lower range limit for process variable Unit of process variable	100 0 %	100 0 %
OUTSCALE/EU_100 OUTSCALE/EU_0 OUTSCALE/UNITS_INDEX	Upper range limit for output variable Lower range limit for output variable Unit of output variable	100 0 %	15 3 psi
CONTROL_OPTS	Sets control options for bad input	Bypass Enable	Bypass Enable
BYPASS	When ON, SP value is transferred to the OUT without the calculation of PID terms.	OFF	OFF
SP_RATE_DN SP_RATE_UP	Rate of change from old to new, higher SP Rate of change from old to new, lower SP	0 0	0 0
GAIN RESET RATE	Tuning constants for the P, I and D terms, of the PID block respectively.	1.5 0.1 0.5	2 0.2 0.6
SHED_OPT	Behaviour when shedding from remote mode	Normal shed, normal return	Normal shed, normal return

Tab. 3-6: Basic parameters for temperature and flow PID blocks

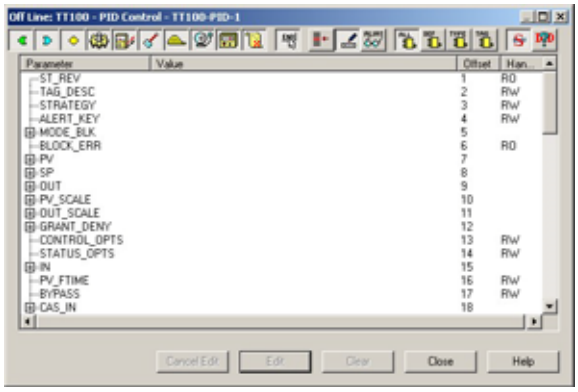
Order of parameters

Some block parameters have a write check based on the value of others parameters. It is therefore important to set the parameters in the order shown in Table 3-6 (in the same order in which they are displayed in the Off Line Characterization dialog).

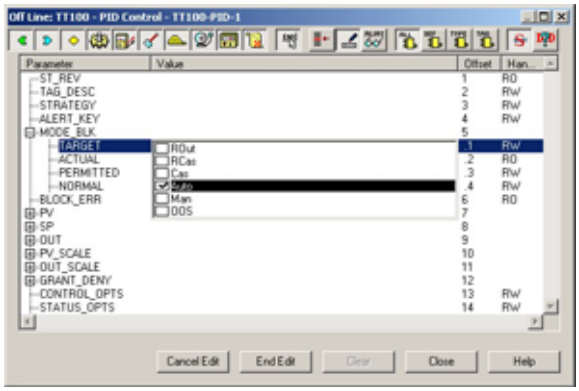
After parametrization of the block, the parameters will appear in the FOUNDATION Fieldbus tree. If you find a parameter in the wrong position, it can be move by dragging and dropping to the correct one.

3.7.4 Configure the PID blocks

1 In the Control strategy workspace, double-click on the **Temperature control PID-1** block
   – The **Off Line Characterization** dialog opens
   – Press **All** to display all the parameters



2 The Mode Block **Target** must be set to **Auto**.



   – Expand the **Mode Block** parameter tree
   – Double-click on the "Value space" next to the **Target** parameter
   – Select **"Auto"** from the drop-down menu
   – Press **End Edit** to register your change

3 Repeat this procedure for the remainder of the Temperature PID parameters in Table 3-6
   – Remember to press **End Edit** after each entry
   – When all parameters have been entered, press Close to quit the Characterization dialog

4 Repeat Steps 1 to 3 for the Flow PID parameters in the **Temperature control-PID-2** block
   – Remember to set the Block Mode **Target** to CAS

5 Open **Project File**, then press **Save**, to save the project

3.7.5 Analog Output parameters

The function block Temperature control- AO-1 for the ND9103FN now has to be configured. The basic parameters required are shown in Table 3-7. A full description of the parameters are to be found in the Operating Instructions

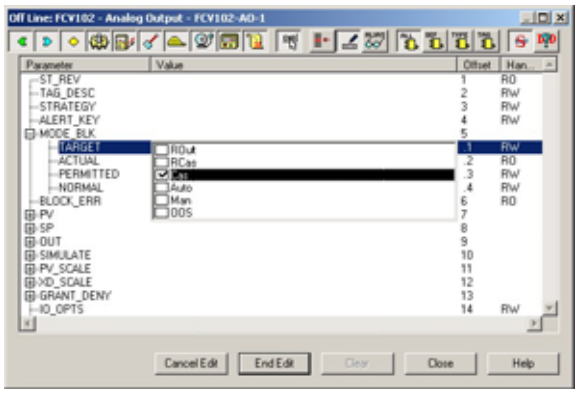
- ND9000F Users Guide, downloadable from www.metso.com

Parameter	Function	Positioner FCV102
MODE BLOCK/TARGET	Normal operating mode of block	Cas
PV_SCALE/EU_100 PV_SCALE/EU_0 PV_SCALE/UNITS_INDEX	Upper range limit for process variable Lower range limit for process variable Unit of process variable	100 0 %
XD_SCALE/EU_100 XD_SCALE/EU_0 XD_SCALE/UNITS_INDEX	Upper range limit for output variable Lower range limit for output variable Unit of output variable	15 3 psi
CHANNEL	Defines the signal configuration between the AO block and transducer block.. • 1 = AO, valve control • 2 = AO, no transducer connected	1 = valve control
SHED_OPT	Behaviour when shedding from remote mode	Normal shed, normal return

Tab. 3-7: Basic parameters for Analog Output block

3.7.6 Configuring the Analog Output block

1 In the Control strategy workspace, double-click on the Temperature control AO-1 block
  – The Off Line Characterization dialog opens
  – Press All to display all the parameters



2 The Mode Block Target must be set to Cas.

  – Expand the Mode Block parameter tree
  – Double-click on the "Value space" next to the Target parameter
  – Select "CAS" from the drop-down menu
  – Press End Edit to register your change

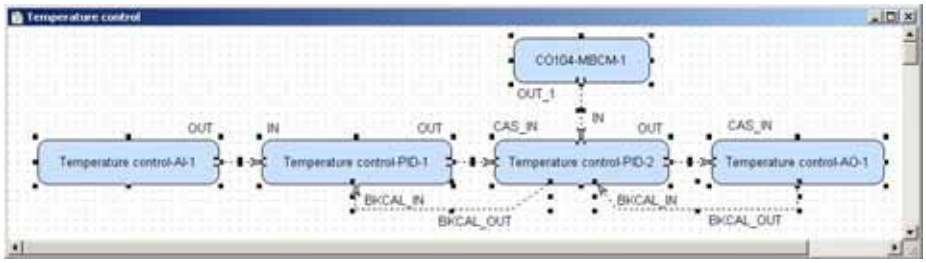
3 Repeat this procedure for the remainder of the Temperature AO parameters in Table 4-3
  – Remember to press End Edit after each entry

4 Open Project File, then press Save, to save the project

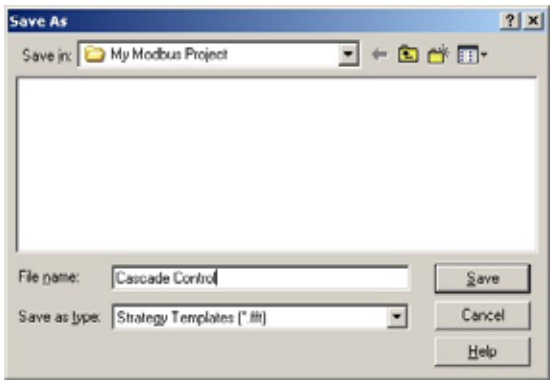
3.7.7 Store the strategy as a template

After configuration, the strategy can be stored as a template for similar applications

1 In the Strategy workspace, drag the mouse across the complete strategy with the lefthand button depressed, until all function blocks are selected



2 Now press the **Strategy Export** button in the toolbar
– The **Save As** dialog appears



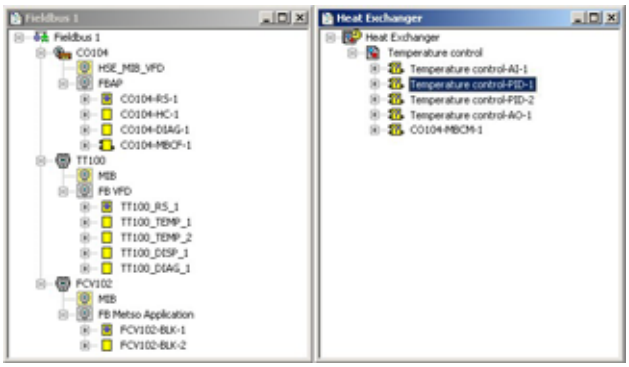
– If appropriate, browse to another folder
– Enter a **File Name**
– Press **Save Entire Configuration** to save the strategy

3 The strategy can be imported again pressing the **Strategy Import** button in the toolbar and double-clicking on the appropriate file name.

3.8 Attach the Function Blocks to the Devices

Now attach the function blocks in the control strategy to the devices where they are to run. The order of assignation determines the order of executions (can be changed by drag&drop).

1 Expand the **Fieldbus** workspace (Fieldbus 1) and the **Process Cell** workspace (Heat exchanger)



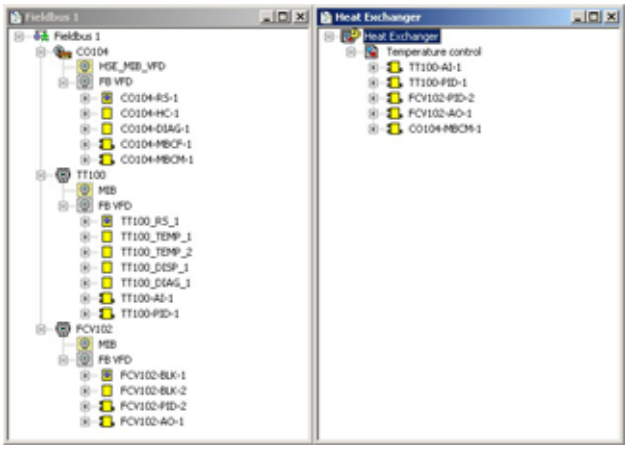
– Note that the Heat Exchanger tree now contains all the function blocks that you created in the Control Strategy workspace

2 Now drag and drop the **Temperature control-AI-1** block to the greyed Function Block Application leaf of the TT100 tree
– When you drop the block, it is attached to tree
– Its name changes to TT100-AI-1 in both views
– The question mark disappears from the block icon in the Process Cell tree
– You have now assigned the Temperature AI block to the temperature transmitter

3 Repeat Step 2 for the other function blocks

– **Temperature control-PID-1** => TT100
– **Temperature control-PID-2** => FCV102
– **Temperature control-AO-1** => FCV102

4 Your project now looks like this



5 Open **Project File**, then press **Save Entire Configuration**, to save the project.

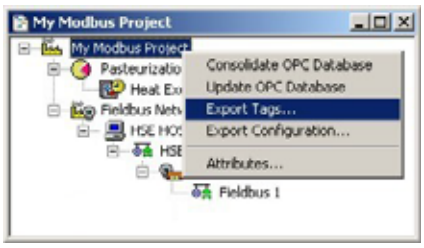
3.9 Export tags

Note!

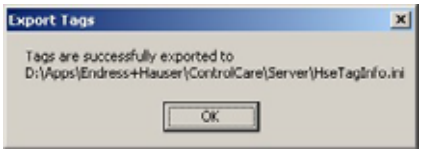
- You should use the Export Tags function everytime you change the configuration of the project, so the the OPC server information is always up-to-date.
- Application Designer will do this automatically everytime you go online, if the corresponding preferences are set, see Chapter 3.3.

1 Active the project view by clicking in its workspace.

2 Right click on the project name, a context menu appears



3 Select the option Export Tags...
  – The Export Tags dialog confirms the successful export



  – Press OK to close the dialog

4 Open Project File, then press Save Entire Configuration, to save the project

3.10 Connect to the Field Controller

In order to download the project, the host computer and Field Controllers must be allocated IP addresses in the same address range. It is possible to do this on the workbench before installation or after the Field Controller and other components have been physically installed in the Fieldbus network (subnet).

Warning

- The use of IP addresses is strictly controlled. Usually your system administrator will be authorised to allocate unique addresses. Assigning an unauthorised address to a Field Controller may result in conflicts within your system and the failure of the associated devices!

Note!

- The tools that setup the network use Ethernet services that may be blocked by Windows Firewall. Normally the firewall will be unblocked for the tools during installation, but it might be necessary to stop the firewall should they not function properly. If you are not sure how to stop the firewall, consult your system administrator.

Before starting, check the following:

- Internet Protocol TCP / IP is installed on your computer
- You have administration rights for your computer
- You have an set of IP addresses that have been authorized by your IT department
- Any proxy server for your Internet Browser is disabled

The procedures described in this chapter are for Windows XP. For other Windows systems consult your system administrator.

Note!

- When the Field Controllers are physically connected together with the Host computer via Ethernet, HSE Network Setup will see the them irrespective of the IP address domain to which they belong

3.10.1 Set the IP address of the host computer

SFC162 Field Controllers are delivered with the default IP address:

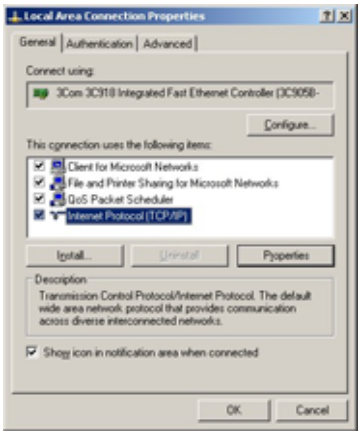
- 192.168.164.100

In order that the host computer can communicate with the Field Controller Web Server, it must be allocated an IP address in the same address domain, e.g. 192.168.164.200. If you are not sure how to do this, consult your network administrator.

Procedure

1 Right-click Start =>Settings =>Control Panel =>Network Connections

2 Right-click Local Area Connection => Properties



3 Using the left mouse button, double-click Internet Protocol (TCP/IP) or click once, then click Properties.

4 Note the original values of IP address and Subnet Mask of the computer to restore them if necessary at end of the operation.

5 Change the IP address and the Subnet Mask of the host computer to those required by the application. In the example, an address in the same subnet as the Field Controller.

- IP Address 192.168.164.XXX and network mask (Subnet Mask) 255.255.255.0.
- Do not use the address 192.168.164.100,as these are reserved as default addresses for Field Controller SFC162

6 Click on the OK button to complete the procedure, close the other dialogs with OK and Close.

3.10.2 Set the Field Controller IP address

Note!

- It is recommended that Field Controllers of the same type are introduced one by one to the network.

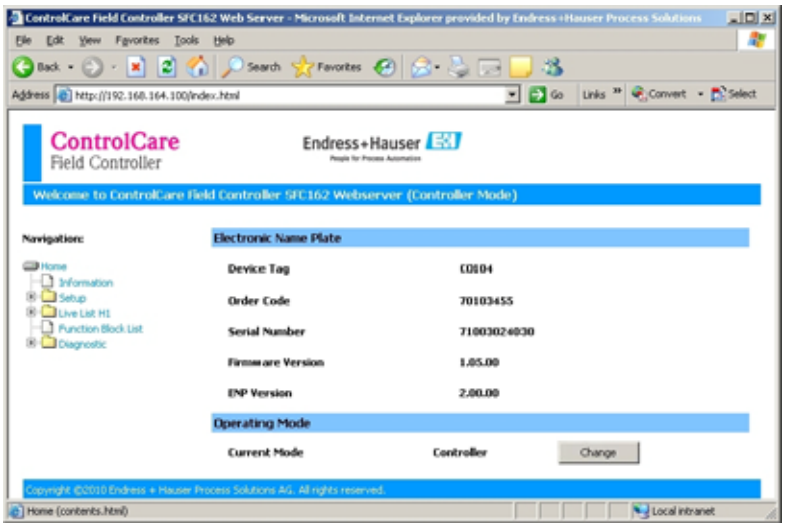
1 Call HSE Network Setup:
Programs =>Endress+Hauser=>ControlCare=>Tools=>HSE Network Setup

2 HSE Network Setup is launched and searches for Field Controllers in the Ethernet network.

– All Field Controllers in the network appear, irrespective of their IP domain.
If this is not the case:
– Check that the proxy server of your Internet Browser is switched off
– Check that the windows firewall is not blocking the program (switch off)
– Check all cables and switches
– If you find two or more Field Controllers with the same IP address, disconnect all but one from the network

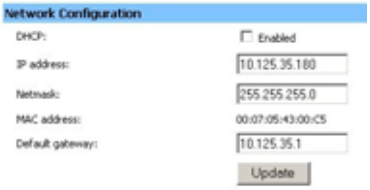
3 If your computer has more than on NIC card, select the one you want to use for communication with the Field Controllers by ticking "Active NIC" and Press .

4 Right-click on the Field Controller, the address of which is to be changed:
the Field Controller Web Server opens



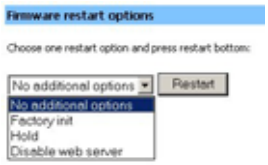
– The Web Server will only open if the host computer and the Field Controller have IP addresses in the same IP domain.

5 Expand the Setup node and click Network
– Enter User Name "pcps" and Password "pcps" to open the Network Configuration dialog



– Enter the required IP address, in our example 10.125.35.180
– Enter a netmask, normally 255.255.255.0
– If required, enter a default gateway, usually address xxx.xxx.xxx.1 in the selected domain

6 Press Update to change the IP address
– You are now asked to restart the Field Controller
– Select the Restart node



– Select "No additional options" from the drop-down menu and press Restart
– Close the Web Browser
– The Field Controller disappears from HSE Network Setup and reappears with the new IP address

7 Now set the address of he host computer to the same domain as the Field Controllers, see Chapter 6.1.1 - in our example 10.125.35.200
– Restart HSE Network Setup

– Tick the Field Controller, so that it appears in the HSE Live List associated with the computer's active NIC card.
– Press to save the configuration.
– You are now ready to download the project

Note!

• If you have more than one Field Controller on the network, Repeat Steps 4 to 6 for all other Field Controllers, introducing them one by one to the network.

3.11 Go online

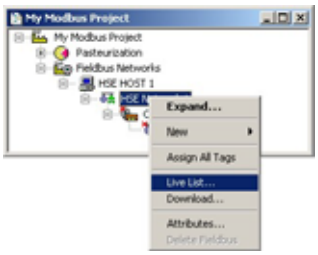
3.11.1 Create the HSE live list

Once the Computer and Field Controller are able to communicate with each other, the connection to the network can be checked by creating a live list.

1 Press the On-Line button in the menu toolbar
   – The project goes on on-line

   – Red crosses temporarily appear against the Field Controller and Fieldbus network in the Project workspace

2 In the Project workspace, right click on HSE Network and select Live List



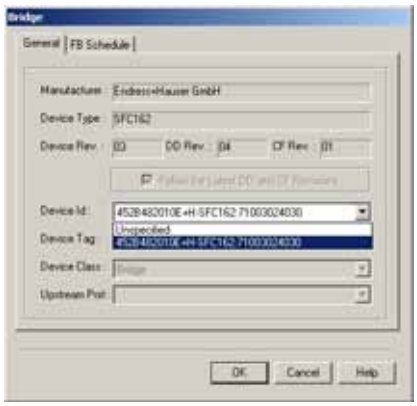
   – A live list is generated of the devices on the HSE network

3.11.2 Assign the HSE Device IDs

1 In the project workspace, right click on the **Field Controller** (CO104) and select **Attributes**...

– The **Attributes** dialog opens

2 Open the drop-down menu of the **Device ID** and select the Field Controller that is associated with the displayed TAG (in our case CO104) - the serial number is on the front panel
– Do this even though the correct ID is already displayed - the program expects it!
– Confirm your choice with **OK**



3 In the HSE Live list, the Field Controller icon goes grey, then reappears with the correct tag

4 Click on the **Project View** workspace and **Export Tags**..., see Chapter 3.9
– Open **Project File**, then press **Save Entire Configuration**, to save the project

3.11.3 Create the FOUNDATION Fieldbus live list

1 In the Fieldbus network workspace, right-click on the Field Controller (CO104) and select the option Live List

2 The Fieldbus live list is created

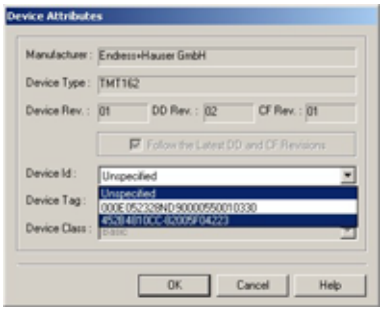
– Check that all the devices in the project appear in the live list
– If this is not the case, check connections, power etc.

3.11.4 Assign the Fieldbus Device IDs

1 In the fieldbus workspace, right click on TT100 and select Attributes...

– The Attributes dialog opens

2 Open the drop-down menu of the Device ID and select the Field Controller that is associated with the displayed TAG (in our case TT100) - the serial number is on the nameplatel

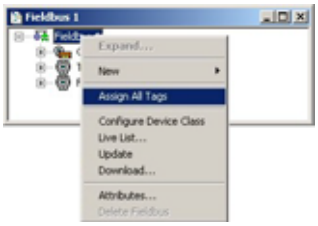


– Confirm your choice with OK
– After a short period of time, the red cross disappears from the device in the Fieldbus network

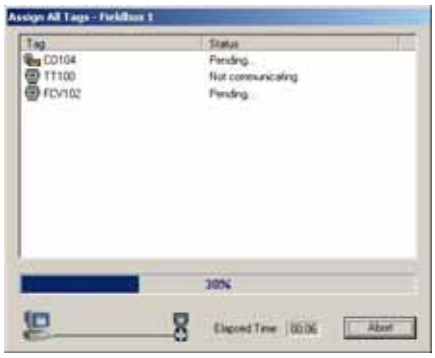
3 Repeat the process for the rest of the devices in the Fieldbus network (FCV102)
– After a period of time, the red cross disappears from the device in the Fieldbus network

3.11.5 Assign All Tags

1 In the Fieldbus workspace, right-click on the Fieldbus node at the top of the tree and select Assign All Tags



2 The Assign All Tags dialog appears with the list of device and a progress bar



- On completion, the message "Profile reading done" stands next to the Field Controller and "Tag has been confirmed" next to the devices
- If there are any failures in tag assignment these are logged with reasons at the bottom of the screen.

3 As the assignment proceeds, the Field Controller transmits the change to the device
- The device goes grey in the live list
- The Field Controller initiates a new network scan
- After about 1 - 2 minutes, the device goes black and appears with the new tag

4 At the end of the process the live list looks like this

5 The "bright" dot next to the SFC162 Field Controller indicates that it is the ACTIVE LAS of this segment
- Any devices configured as backup LAS have an ordinary dot next to them, see Chapter 3.12.2

6 Click on the Project View workspace and Export Tags..., see Chapter 3.9
- Open Project File, then press Save Entire Configuration, to save the project

3.12 Download the project

Note!

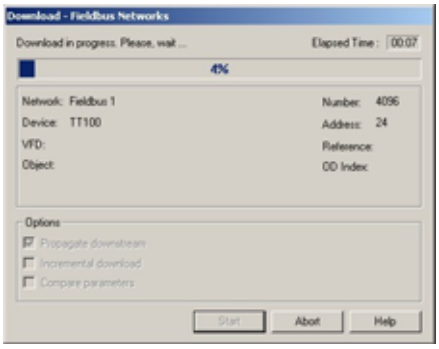
- The procedure below describes the initial download for the entire HSE network.
- Partial downloads can be made later from lower leaves, when changes are confined to this level
- Incremental downloads can be made to a running project by checking the boxes Incremental Download and Compare Parameters: Unaffected Local I/Os will hold their last values.

3.12.1 Download

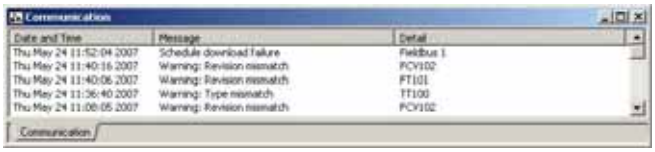
When the devices in the Fieldbus live list correspond to those configured in the project, the download can begin.

1 In the Project workspace right-click on HSE Network and select Download

2 The Download dialog appears



– Press Start to start the download
– A log at the bottom of the screen allows you to identify download problems



3 The download will be interrupted if the project encounters a critial problem, e.g.

– The Project tags are not up-to-date => Chapter 3.10
– The Device IDs have not been assigned correctly => Chapter 6.2.2, Chapter 6.2.4
– The Device Tags have not been assigned => Chapter 6.2.5

4 When the download is successfully completed, the dialog is closed or disappears and you are ready to test the control strategy

3.12.2 Configure device class

In our project the Field Controller acts as the Link Active Scheduler by default. The FOUNDATION fieldbus protocol allows other devices to take over the this role should the Field Controller fail. This ensures that any control loop not involving function blocks in the controller will continue to function in this event. A device can be setup to be a backup LAS by using the Configure Device Class function. This function is only effective when the device is online
.

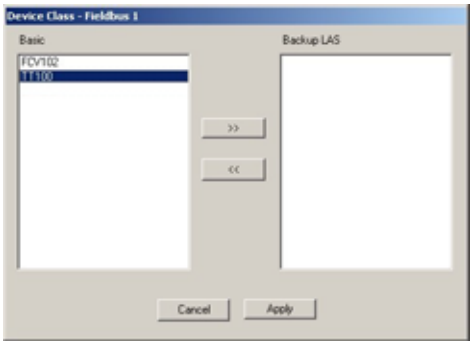
Note!
- If you are offline, the process finishes at Step 2. The change is registered in the project but is not downloaded with it. The procedure must then be repeated with the device online, otherwise a mismatch when you try to download.

Procedure

1 In the Fieldbus window, right click on the Fieldbus node to open the context menu

– Select the menu Configure Device Class

2 The Device Class dialog opens



– Select the device(s) you want as backup LAS
(usually one with few FB links, in our case TT100)
– Press >> to transfer the device to the righthand pane
(if the device does not support LAS functionality, an error message appears)
– Press Apply to confirm the choice and close the dialog.

3 When online, the Change Device Class dialog appears and shows the download progress

4 On completion the TT100 device appears in the live list with a dot indicating backup LAS functionality

3.13 Make the Modbus Connection

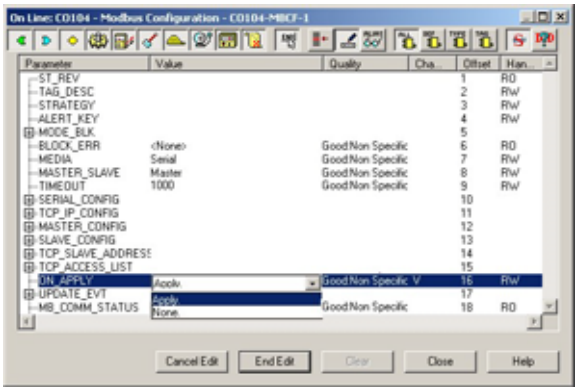
3.13.1 Start the Modbus

The Modbus must be started after the download:

1 In the Fieldbus window, expand the Field Controller tree

– Right-click on the CO104 -MBCF-1 block and select On Line Characterization

2 In the On Line Characterization dialog
– Select All to display all parameters
– Select the parameter ON_APPLY



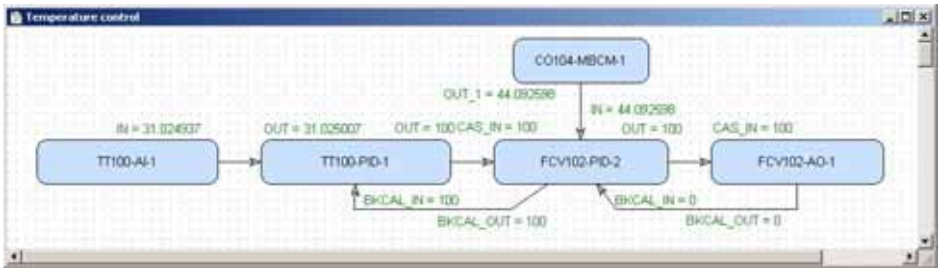
3 Click in the value space at the center of the line and open the drop-down menu
– Select Apply
– Click End Edit to confirm your choice
– The parameter remains for two or three seconds, then reverts to None
– The Modbus starts
– Press Close to close the dialog

.

Note!
• This procedure must be repeated every time the Modbus configuration is changed

3.13.2 Check the control stategy

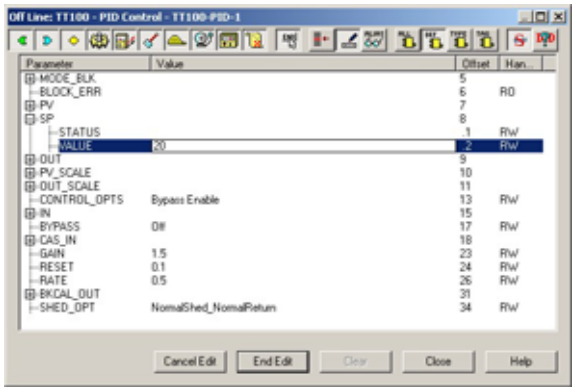
1 Click in the Control Strategy workspace (Temperature control) and press the button  in the menu toolbar – the control strategy goes "on-line"



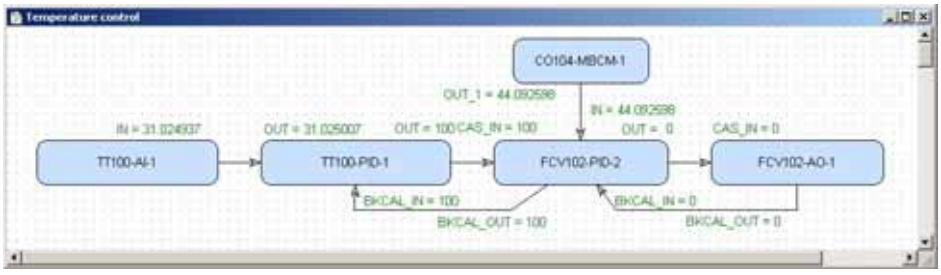
– Values appear in green when the status is good
– Values appear in red if the status is bad - at this stage this is an indication of a Fieldbus configuration, a strategy configuration or a device parametrization error

2 Check that the loop is working by changing the Setpoint parameter in the temperature PID TT100

– Double-click on the Temperature PID block TT100-PID-1, the On-line Characterization dialog appears
– Press the All icon to reveal all parameters
– Open the SP leaf and double-click in the space next to Value
– Enter a new SP value – for T > SP, set SP < T or vice versa
– Click End Edit to set the parameter
– Press Close to store the value (if you are prompted - answer with Yes)



3 Now check that the control loop has responded properly



3.14 Modify the project

3.14.1 On-line characterization

Once the project is on-line you may want to change parameters to e.g. tune the control-loop or eliminate configuration errors. With the exception of the SP (PID block) and Apply (MBCF block) parameters, the function block must be put out of service before the parameter is changed:

1 In the Control strategy workspace double-click on the function block you want to modify, or in the Fieldbus network or Control module workspace, right-click on the function block and select On-line Characterization

2 The function block On-line Characterization dialog appears:

- Press the All icon to reveal all parameters
- Open the Mode leaf and double-click in the space next to Target
- Set the Target to OOS (Out of Service)
- Click End Edit to set the parameter

3 Change the parameters you wish to modify

- If appropriate, open the parameter leaf and double-click in the space next to the parameter you require
- Enter the new parameter or select it from the drop-down menu
- Click End Edit to set the parameter
- Repeat the procedure for all the parameters you wish to modify

4 Put the function block back into standard operating mode

- Open the Mode leaf and double-click in the space next to Target
- Set the Target back to the original value (Auto (Automatic) or Cas (Cascade))
- Click End Edit to set the parameter
- If you have modified the Modbus configuration (MBCF or MBCM blocks), start the Modbus again with the ON_APPLY parameter, see Chapter 3.12.1
- Check that the Mode really changes to the Target Mode (failure to do so indicates a configuration error)
- Press Close to store the values (if you are prompted - answer with Yes)

5 Click on the Project View workspace and Export Tags..., see Chapter 3.9
- Open Project File, then press Save Entire Configuration, to save the project

6 Put the Control stategy back "on-line" to check the results of your modification, see Chapter 3.12.2.

3.14.2 Off Line characterization

You may prefer to change parameters off-line, e.g. when modifying the control strategy or adding new functions to the project.

1 If you are on-line, press the Off-line button in the menu toolbar
alternatively, in the FOUNDATION Fieldbus network or Control module workspace, right-click on the function block and select Off Line Characterization
– The function block Off Line Characterization dialog appears

2 Change the parameters you wish to modify

– If appropriate, open the parameter leaf and double-click in the space next to the parameter you require
– Enter the new parameter or select it from the drop-down menu
– Click End Edit to set the parameter
– Repeat the procedure for all the parameters you wish to modify
– Press Close to store the values

3 Click on the Project View workspace and Export Tags..., see Chapter 3.9
– Open Project File, then press Save Entire Configuration, to save the project

4 Press the On-line button in the menu toolbar to go on-line again

5 Download the modified project
– In the Project workspace right-click on HSE Network 1 and select Download
– Follow the procedure in Chapter 3.11.6.

6 Start the Modbus with the ON_APPLY parameter, see Chapter 3.12.1

7 Put the Control stategy back "on-line" to check the results of your modification, see Chapter 3.12.2.

Note!

- The ON_APPLY parameter must be activated after every download because the Modbus parameters are overwritten during the download process. ON_APPLY confirms the changes and starts the execution of the blocks.

3.15 Export the configuration

The project configuration can be exported to an existing ODBC file data source, e.g. Oracle, a machine database, e.g. Excel to provide a record of the current status of the project or to an XML sheet for viewing with a browser.

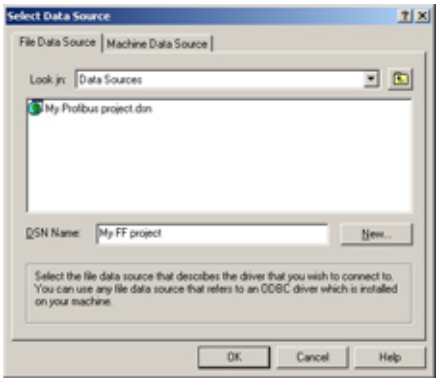
3.15.1 File data source folder

The file data source must have been created before the export.

1 In the Project window, right-click on the Project icon and select Export Configuration:

2 The Select Data Source dialog box appears

3 In the File Data Source folder, select the source that describes the driver that you wish to connect to. You can use any file data source that refers to an ODBC driver which is installed on your machine.
- Use the New... button and Look In dropdown menu to browse or
- Click the data source icon to select the driver:



- Press OK to make the connection

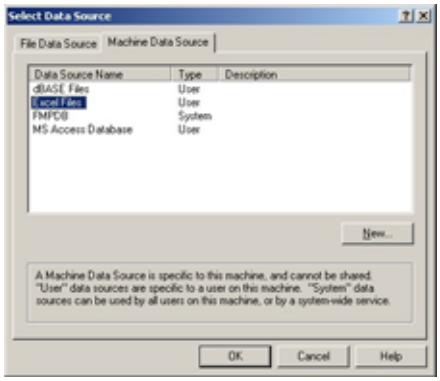
3.15.2 Machine data source folder

The Machine Data Source is specific to the machine, and cannot be shared. "User" data sources are specific to a user on the machine; "System" data sources can be used by all users on the machine, or by a system-wide service. The Machine Data Source must have been created before export.

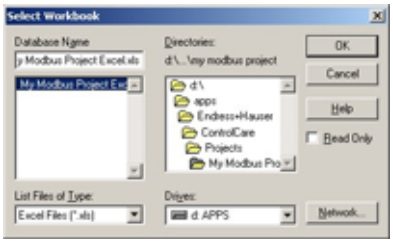
Procedure

1 In the Project window, right-click on the Project icon and select **Export Configuration**:

2 The Select Data Source dialog box appears
- Click on the **Machine Data Source** tab to open the folder
- Double-click the data source name to select the machine, e.g. Excel:

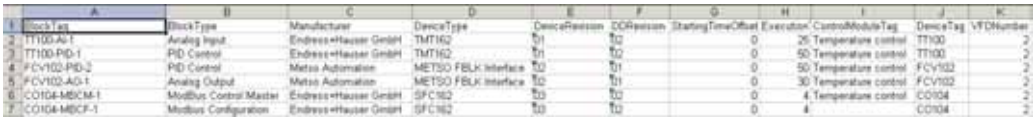


3 The Select Workbook dialog box will appear:
- Select the folder where the data file is and double-click the workbook icon.



- Your project configuration will be exported to the workbook file.
- A message box appears on completion - press OK

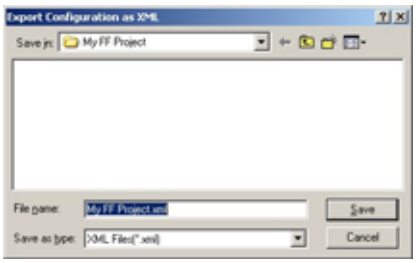
4 Open the Excel file to check the result:



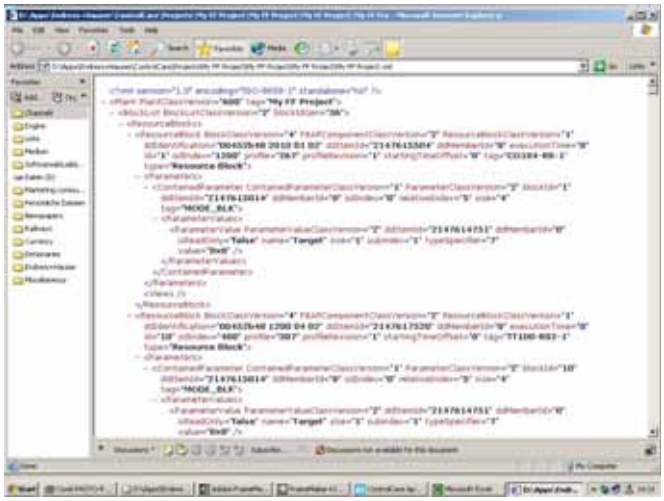
	A	B	C	D	E	F	G	H	I	J	K
1	BlockTag	BlockType	Manufacturer	DeviceType	DeviceRevision	DDRevision	StartingTimeOffset	Execution	ControlModuleTag	DeviceTag	VFDNumber
2	TT100-AI-1	Analog Input	Endress+Hauser GmbH	TMT162	01	02	0	25	Temperature control	TT100	2
3	TT100-PID-1	PID Control	Endress+Hauser GmbH	TMT162	01	02	0	50	Temperature control	TT100	2
4	FCV102-PID-2	PID Control	Metso Automation	METSO FBLK Interface	02	01	0	50	Temperature control	FCV102	2
5	FCV102-AO-1	Analog Output	Metso Automation	METSO FBLK Interface	02	01	0	30	Temperature control	FCV102	2
6	CO104-MBCM-1	ModBus Control Master	Endress+Hauser GmbH	SFC162	03	02	0	4	Temperature control	CO104	2
7	CO104-MBCF-1	Modbus Configuration	Endress+Hauser GmbH	SFC162	03	02	0	4		CO104	2

3.15.3 XML file

1 Click in the Project workspace and select Project File => Export => Configuration as XML
– The Export Configuration as XML dialog appears



2 Enter a File Name and Save In location, then press Save
– The project is saved as an XML file at the selected location



3.16 Close Application Designer

When you have completed your session, close Application Designer

1 If you are on-line, press the Off-line button  in the menu toolbar

2 If you have made any modifications while you were on line, you will be prompted to store them
   – If appropriate answer with Yes

3 Close the project by clicking on Project File => Close

4 Exit Application Designer by clicking on Project File => Exit

5 The Field Controller continues to operate with the project configured according to the last download/on-line correction
   – If you switch off the Field Controller, the project remains stored in its memory (back-up switch must be set as described in BA021S/04/en, Field Controller, Hardware Installation)
   – It is initialized and re-executed as soon as the Field Controller is switched on again

3.16.1 Reconnecting

Provided your computer is operating in the same IP address domain as the Field Controller, you can reconnect at any time.

1 Start up Application Designer and select the Project you require

2 Press the On-line button  in the menu toolbar

3 Expand the various workplaces as required

4 Click in the Control Strategy workspace and press the button  in the menu toolbar – the control strategy goes "on-line" with the last configuration that was downloaded.

4 Field Controller as Modbus Slave

4.1 Task Description

This part of tutorial describes all steps necessary for setting up the Field Controller as a Modbus Control Slave. It does not aim to give an exhaustive account of Application Designer functions, but rather shows you one of a number of methods to reach your goal. The tags and names used in the tutorial are imaginary and will be different in a proper application. A full description of Application Designer functions is to be found in Application Designer Overview BA017S/04/en. Function block descriptions are to be found in BA022/04/en, Function Block manual.

4.1.1 Application

For this part of the tutorial it is assumed that a controller, e.g. PLC, acting as a Modbus master acquires a temperature value from a FOUNDATION Fieldbus device through the Field Controller, which acts as a Modbus slave. The Modbus master returns a control signal to a FOUNDATION Fieldbus positioner, which in turn reports its new position to the Modbus controller. It is assumed that all control is done in the Modbus master, so that the FOUNDATION Fieldbus network supplies and receives values only. The corresponding control strategy is shown in Fig. 4-1.

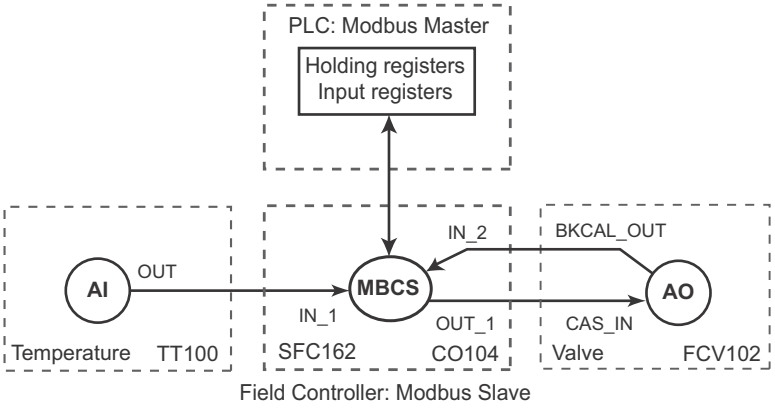


Fig. 4-1 Cascade control strategy for heat exchanger application

Since it is not the object of this tutorial to describe the programming of a Modbus controller, the control function has been simulated by the ModSim32 software. This provides a simple means of testing the application and checking what is being transmitted across the Modbus interface. The simulator operates via Ethernet TCP, but other simulators are available for RS232.

Note!

- The Modbus standard restricts the maximum slave telegram length to 256 bytes, which corresponds to 125 measuring points of two bytes each plus six bytes overhead. If more measuring points are required, the master must be programmed to send more than one read request per cycle.

4.1.2 Network

The network is assumed to be constructed as shown in Fig. 4-2.

- The Modbus master is simulated by the ModSim32 application
- The valve positioner is a Metso ND9103FN
- The temperature transmitter TMT162

In the example the SFC162 is used as Modbus Slave, but the SFC173 could also be used.

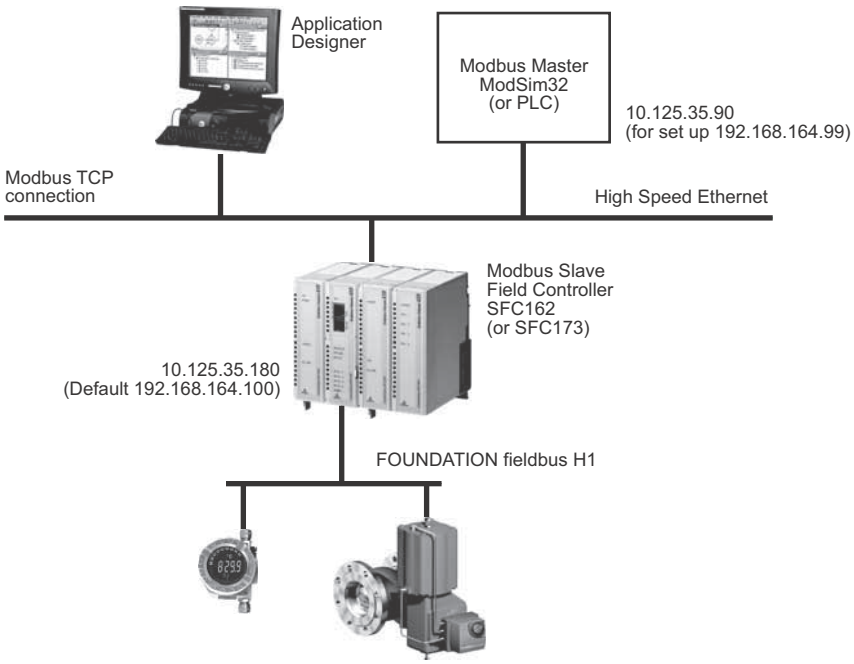


Fig. 4-2 Network for application example

4.1.3 Installation and commissioning

Before you can start this part of the Modbus tutorial, Application Designer must be installed on your computer, the SFC162 FOUNDATION Fieldbus Controller installed and commissioned and a connection made to your computer. Instructions on how to do this are to be found in:

- Operating Instructions BA020S/04/en, Getting Started
- Operating Instructions BA021S/04/en, Field Controller: Hardware Installion
- Operating Instructions BA035S/04/en, Field Controller: Commissioning and Configuration

4.1.4 Device ID and tag

For a FOUNDATION Fieldbus system, each device that communicates has a unique bus address and tag. Addresses are assigned automatically during the start-up of the system on the basis of the device ID. The device ID is a unique identifier that is based on a Manufacturer ID and the serial number of the device. When the project goes online, the actual device IDs must be assigned to virtual devices that have been planned in Application Designer by using the Assign Tags procedure.

To aid the offline engineering of the network, it is necessary to keep a record of the measuring point tags (device tags), often as an Excel sheet. Measuring point tags are used in P&I diagrams to indicate the type of measurement or action performed at a particular location in a process. Table 4-1 below provides an example of how this might look for the application at hand.

Area	Process Cell	Device	Vendor	Tag	Unit	Task
Pasteurization	Heat Exchanger	TMT162	E+H	TT100	°C	Product temperature
Pasteurization	Heat Exchanger	ND9103FN	Metso	FCV102	%	Steam valve positioner
Pasteurization	Heat Exchanger	SFC162	E+H	CO104		Field Controller acting as Modbus slave

Tab. 4-1: Measuring point tag list for tutorial application

4.2 Create a project

4.2.1 Create the project and add the fieldbus

1 Following the instructions in Chapter 3.2 create a new project, e.g. My Modbus Slave Project

2 Following the instructions in Chapter 3.3, set the preferences

3 Following the instructions in Chapter 3.4, add the bridge, fieldbus and devices
– In this case, do not add the MBCF and MBCM blocks to the controller yet.

4 Your project should now look something like this:

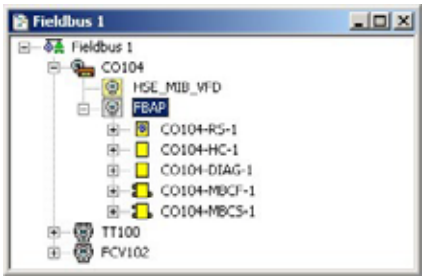
and the fieldbus like this

4.2.2 Add the Modbus blocks

1 Expand the Field Controller (CO104) leaf on the Fieldbus tree and add the Modbus blocks MBCF and MBCS

– Right-click on the FBAP leaf and select New Block
– Select the Modbus Configuration block
– Repeat and select the Modbus Control Slave block

2 The Fieldbus tree now looks like this

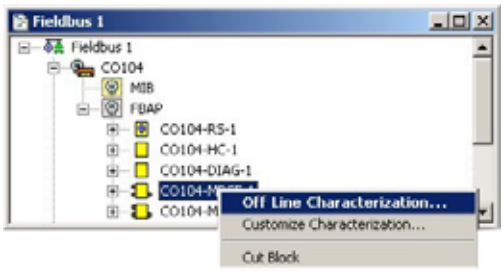


3 Open **Project File**, then press **Save**, to save the project

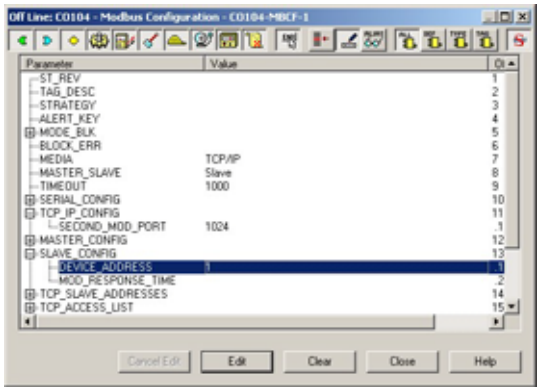
4.3 Configure the devices

4.3.1 MBCF Modbus Configuration block

1 In the Fieldbus 1 window, right-click on the Field Controller function block CO104-MBCF-1 and select Off Line Characterization



2 The Offline Characterization dialog opens: Press All to show all parameters



3 Set the following parameters by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCF
MODE BLOCK.TARGET	Normal operating mode of block	Auto
MEDIA	Channel for Modbus communication	TCP/IP (+TCP/IP)
MASTER_SLAVE	Role of Field Controller in Modbus network	Slave
TIMEOUT	Time allowed for OUT value update	1000 (ms)
TCP/IP_CONFIG SECOND_MOD_PORT	Configures TCP/IP interface Second communication port (Port 502 is always open)	 e.g. 1024
SLAVE_CONFIG DEVICE_ADDRESS	Configures Controller when acting as slave Modbus address of Field Controller	 1
TCP_ACCESS_LIST IP_1	List of masters allows to access field controller registers IP address of TCP master	 10.125.35.90

4 Click Close to close the dialog: the parameters are added to the MBCF function block

4.3.2 MBCS Modbus Control Slave block

The MBCS function block allows fieldbus and Modbus data to be exchanged through 16 Modbus registers see Chapter 2.4. If more than four registers of a particular type are required, up to 16 MBCM blocks can be created. These are managed by the LOCAL_MOD_MAP parameter. By default LOCAL_MOD_MAP = 0, valid range 0 – 15.

The value of LOCAL_MOD_MAP is used to define a unique set of Modbus register address ranges for the particular MBCS block, whereby:

- Register address = Constant + 8 x Value of LOCAL_MOD_MAP for analog values
- Register address = Constant + 4 x Value of LOCAL_MOD_MAP for discrete/status values. where by Constant = 0 for e.g. IN_1, 1 IN_2 etc.

The Modbus master accesses the Field Controller slave registers by specifying the function code and the register address. Alternatively, the reference address can be used, whereby the relationship to the input and output parameters is as follows
.

FC parameter	Type	Master action	Scaling parameter	Reference address
IN_1 to IN_4	Analog input	Reads from input register	SCALE_CONV_XXX	30001 + register address
IN_D1 to IN_D4	Discrete input	Reads from discrete input	–	10001 + coil address
OUT_1 to OUT_4	Analog output	Writes to holding register	SCALE_CONV_XXX	40001 + register address*
OUT_D1 to OUT_D4	Discrete output	Writes to coil	–	1 + coil address*
*The holding registers and coils are Read/Write				

For the tutorial we require the IN_1, IN_2 and OUT_1 channels. When LOCAL_MOD_MAP is set to zero, the data will be mapped to the registers in Table 4-2, see also Chapter 2.4.

Parameter	Channel	Register	Reference address	Data type	Master access
Temperature	IN_1	0	30001	Float	Read
Status		128	30128	Integer16	
Positioner	OUT_1	0	40001	Float	Write
Status		128	40128	Integer16	
Back calculation	IN_2	2	30003	Float	Read
Status		129	30129	Integer16	

Tab. 4-2: Modbus registers for exchange of data

The SCALE_CONV_XXX parameter allows each analog channel to be individually configured for scaling and data type. By default, the Field Controller sends its float number with the byte order 1-0-3-2 and not in the sequence 3-2-1-0. The 1-0-3-2 sequence corresponds to the DATA_TYPE "float".

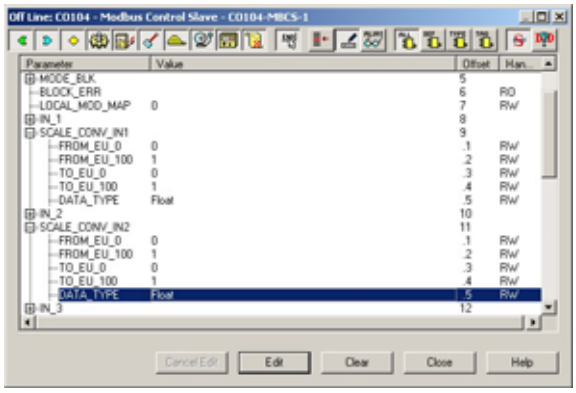
The values mapped to the Field Controller or sent to a Modbus device register can be scaled.

Parameter	SCALE_CONV_INx	SCALE_CONV_OUTx
.FROM_EU_0	Lower range limit IN_x value	Lower range limit Modbus master value
.FROM_EU_100	Upper range limit IN_x value	Upper range limit Modbus master value
.TO_EU_0	Lower range limit Modbus master value	Lower range limit OUT_x value
.TO_EU_100	Upper range limit Modbus master value	Upper range limit OUT_x value

The STATUS_OUTPUT element defines how the OUT.STATUS will be managed. As the status provided by the master is probably not in conformance with the FOUNDATION Fieldbus protocol, the option "Good: Cascade:NonSpecific:Not Limited" will be selected. This will always be transmitted by the Field Controller as the OUT status unless there is a timeout, in which case the OUT status will be set to Bad.

Procedure

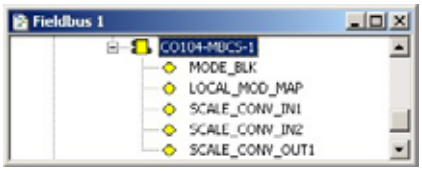
1 Right-click on the CO104-MBCS-1 block and open the Off Line Characterization dialog



2 Click All to reveal all parameters and enter the following parameters by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCS
MODE BLOCK.TARGET	Normal operating mode of block	Auto
LOCAL_MOD_MAP	Identifier of modbus block (0 - 15)	0
SCALE_CONV_IN1	Scaling and conversion of IN_1 (temperature)	
FROM_EU_0	Lower range limit of IN_1 (= OUT temperature AI block)	0 (%)
FROM_EU_100	Upper range limit of IN_1 (= OUT temperature AI block)	100 (%)
TO_EU_0	Lower range limit of Modbus master signal	0 (%)
TO_EU_100	Upper range limit of Modbus master signal	100 (%)
DATA_TYPE	Type of data transmitted	float
SCALE_CONV_IN2	Scaling and conversion of IN_2 (back calculation)	
FROM_EU_0	Lower range limit of IN_1 (= BKCAL_OUT valve AO block)	0 (%)
FROM_EU_100	Upper range limit of IN_1 (= BKCAL_OUT valve AO block)	100 (%)
TO_EU_0	Lower range limit of Modbus master signal	0 (%)
TO_EU_100	Upper range limit of Modbus master signal	100 (%)
DATA_TYPE	Type of data transmitted	float
SCALE_CONV_OUT1	Scaling and conversion of OUT_1 (valve positioner)	
FROM_EU_0	Lower range limit of Modbus master signal	0 (%)
FROM_EU_100	Upper range limit of Modbus master signal	100 (%)
TO_EU_0	Lower range limit of OUT_1	0 (%)
TO_EU_100	Upper range limit of OUT_1	100 (%)
DATA_TYPE	Type of data transmitted	float
STATUS_OUTPUT	Status to be used by good output	GoodC:NSpecNLim

3 Press Close to close the Off Line Characterization dialog. You should now see the parameters attached to the MBCM block:



4 Open Project File, then press Save, to save the project.

4.3.3 Configure the Fieldbus devices

1 Configure the devices according to Tables 4-3 and 4-4 as described in Chapter 3.5.3 and 3.5.4 respectively.

2 Open **Project File**, then press **Save**, to save the project.

TMT162

Table 4-3 shows the parameters that must be set in the TMT162 transducer block

Parameter	Function	Temperature TT100
MODE BLOCK/TARGET	Normal operating mode of block	Auto
PRIMARY_VALUE_TYPE	Calculation method for primary process value • Process temperature SV1 or SV2 • Average 0.5 (SV1 + SV2) with/without redundancay • Differential (SV1 - SV2) • Conditional (SV1 or SV2), (SV2 if SV1 >T)	Sensor Value 1
SENSOR_TYPE	Type of sensor connected to the transmitter • All types of standardized temperature sensors	Pt 100 IEC 751
SENSOR_CONNECTION	Way in which the sensor is connected • 4-wire (if two sensors are connected only one can be 4-wire) • 3-wire • 2-wire	2-wire

Tab. 4-3: Basic parameters for TMT162 transducer block

ND9103FN

Rotary valve parameters are included in Table 4-4.

Parameter	Function	Positioner FCV102
MODE BLOCK/TARGET	Normal operating mode of block	Auto
VALVE_TYPE	Type of valve the positioner is actuating • Select from drop-down menu	Rotary
FINAL_VALUE_RATE_DN	Maximum travel rate in closing direction • 0 = parameter not in use	0
FINAL_VALUE_RATE_UP	Maximum travel rate in opening direction • 0 = parameter not in use	0
POSITIONER_FAIL_ACTION	Action of position on loss of electrical power or reception of an output signal with a bad status • Select from drop-down menu	Close
POS_SENSOR_ROT	Relationship between valve action and position sensor rotation • Select from drop-down menu	Standard: Clockwise to close
DEAD_ANGLE_COMP	Dead angle for segment and rotary valves	0
ACT_TYPE	Type of positioner action • Select from drop-down menu	Double-acting actuator
PERFORMANCE_LEVEL	Target performance level of valve position control Select from drop-down menu	Optimum
CHAR_TYPE	Type of linearization • Select from drop-down menu	No characterization

Tab. 4-4: Basic parameters for Metso ND9103FN transducer block

4.4 Create the Control Strategy

Having created a physical view of the process, the next step is to create control strategy. This is done described in Section 3.6.

4.4.1 Add a Process Cell

1 Click on the "Area 1" leaf in the project and select Attributes...

- The Attributes dialog box appears
- Enter a name for the area, e.g. Pasteurization (see Table 3-1, Chapter 3.1.4)
- Click OK to store your changes

2 Click on the Area leaf again and select New Process Cell..

- The Process Cell dialog box appears
- Enter a name for the process cell, e.g. Heat Exchanger (see Table 3-1)
- Click OK to store your changes

3 Open Project File, then press Save, to save the project.

4.4.2 Add a Control Module

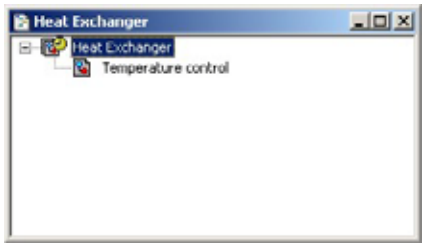
1 Right-click on the Process Cell leaf you just created and select Expand

- A new window with the name of the leaf opens

2 Right-click on the top leaf and select New Control Module

- The Control Module dialog box appears
- Enter a name for the control module, e.g. Temperature Control (see Table 3-1)
- Click OK to store your changes

3 The project looks something like this



4 Open Project File, then press Save, to save the project.

4.4.3 Create the function blocks

1 Double-click on the control module leaf or right-click and select **Expand** to open the **Control Strategy** workspace - this has the same name as the leaf

2 Press the Function Block button in the toolbar and click in the workspace

- The **New Block** dialog appears
- Select the **Manufacturer** = Endress+Hauser
- Select the **Device Type** = TMT162
- Select the **Block Type** = Analog Input
- Press **OK** to create the function block

3 The block now appears in the strategy window with the default name

4 Repeat Steps 2 and 3 for the Positioner AO block

- Positioner AO
  Manufacturer = Metso Automation
  Device Type = FBLK Interface
  Block Type = Analog Output

5 The flow value is supplied by the MBCS block which you have already created

- In the **Fieldbus 1** window expand the **CO104** leaf and click on **CO104-MBCS-1**
- Drag and drop the block into the control strategy window

6 The control strategy now looks like this

7 Open **Project File**, then press **Save Entire Configuration**, to save the project

Note

- In the tutorial we created the MBCS block together with the MBCF block. Is is also possible to create the MBCS block in the strategy window as we have done with the AI and AO blocks, and then drag and drop it to the Fieldbus, see Chapter 3.8.

4.4.4 Add the Function Block links

1 In the Control Strategy workspace position the blocks according to your strategy

2 Click on Function Block Link button in the tool bar, the cursor changes to a cross

3 Select the Temperature AI block with the cross: the Output Parameter Selection dialog appears

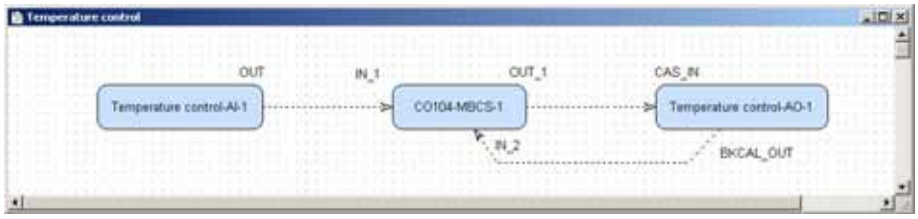
- Click the box next to OUT – it changes color – then click on OK
- The Output Parameter Selection dialog closes
- The cursor is now connected to a blue dotted line
- Place the Cursor in the CO104-MBCS-1 block and click to make the link
- When the link is made, the Input Parameter Selection dialog for the PID block appears
- Click the box next to IN_1 – it changes color – then click on OK

4 If the Aliasing Input dialog box is enabled, see Chapter 3.3, the Rename dialog now appears
- Enter the desired link name and press OK
- If nothing is entered, the link retains the standard name

5 Repeat steps 2 to 4 and make the following links between the function blocks

- CO104-MBMS-1 to Valve AO = OUT_1 to CAS_IN
- Valve AO to CO104-MBMS-1 = BKCAL_OUT to IN_2

6 Your Control Strategy now looks something like this



7 Open Project File, then press Save Entire Configuration, to save the project.

4.5 Configure the strategy

4.5.1 Configure the blocks

1 In the Control strategy workspace, double-click on the Temperature control AI-1 block
Set the parameters in Table 4-5 as described in Chapter 3.7.2

2 In the Control strategy workspace, double-click on the Temperature control AO-1 block
Set the parameters in Table 4-6 as described in Chapter 3.7.6

3 Open Project File, then press Save, to save the project

Analog Input parameters

Parameter	Function	Temperature TC100
MODE BLOCK/TARGET	Normal operating mode of block	Auto
XD_SCALE/EU_100* XD_SCALE/EU_0 XD_SCALE/UNITS_INDEX	Upper range value for process variable Lower range value for process variable Unit of process variable	150 (max.850) –50 (min. –200) °C
OUT_SCALE/EU_100 OUT_SCALE/EU_0 OUT_SCALE/UNITS_INDEX	Upper range limit for output variable Lower range limit for output variable Unit of output variable	100 0 %
CHANNEL	Output channels of Transducer Block aasigned to Ananlog Input Block. • Primary, RJ or Sensor vlaue 1/2 depending on whether one or two sensors are connected	Sensor Value 1
L_TYPE	Selects the type of linearisation for the input value. • Direct: PV value = OUT value, Identical XD_SCALE and OUT_SCALE • Indirect: PV value scaled to OUT value • Indirect Square Root: as Indirect but scaling with root function	Indirect
PV_FTIME	Output damping constant (in seconds).	1

Tab. 4-5: Basic parameters for Analog Input block

Analog Output parameters

Parameter	Function	Positioner FCV102
MODE BLOCK/TARGET	Normal operating mode of block	Cas
PV_SCALE/EU_100 PV_SCALE/EU_0 PV_SCALE/UNITS_INDEX	Upper range limit for process variable Lower range limit for process variable Unit of process variable	100 0 %
XD_SCALE/EU_100 XD_SCALE/EU_0 XD_SCALE/UNITS_INDEX	Upper range limit for output variable Lower range limit for output variable Unit of output variable	15 3 psi
CHANNEL	Defines the signal configuration between the AO block and transducer block.. • 1 = AO, valve control • 2 = AO, no transducer connected	1 = valve control
SHED_OPT	Behaviour when shedding from remote mode	Normal shed, normal return

Tab. 4-6: : Basic parameters for Analog Output block

4.5.2 Attach the Function Blocks to the Devices

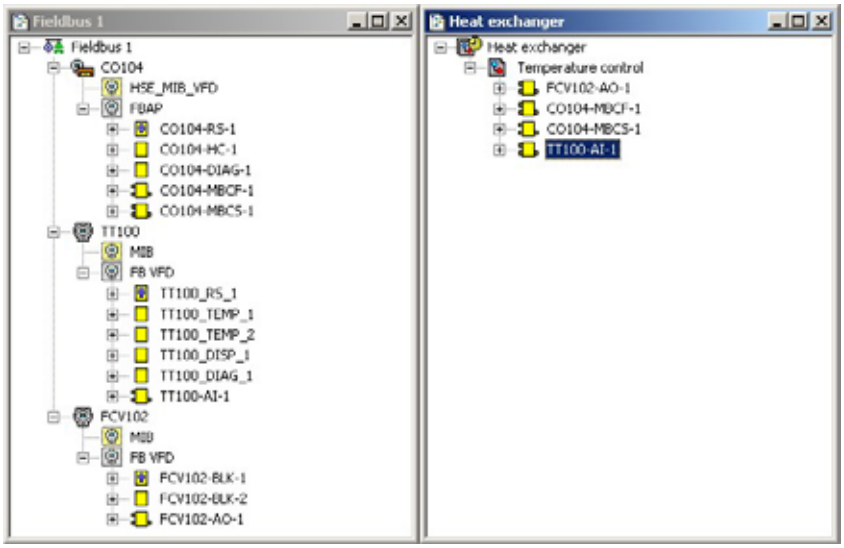
Now attach the function blocks in the control strategy to the devices where they are to run. The order of assignation determines the order of executions (can be changed by drag&drop).

1 Expand the Fieldbus workspace (Fieldbus 1) and the Process Cell workspace (Heat exchanger)

2 Now drag and drop the Temperature control-AI-1 block to the greyed Function Block Application leaf of the TT100 tree
- When you drop the block, it is attached to tree
- Its name changes to TT100-AI-1 in both views
- You have now assigned the Temperature AI block to the temperature transmitter

3 Repeat Step 2 for the other function block
- Temperature control-AO-1 => FCV102

4 Your project now looks like this



5 Open Project File, then press Save, to save the project.

4.5.3 Export tags

1 Active the project view by clicking in its workspace.

2 Right click on the project name, a context menu appears
- Select the option Export Tags...
- The Export Tags dialog confirms the successful export
- Press OK to close the dialog

3 Open Project File, then press Save, to save the project

4.6 Go Online

Now go online as described in more detail in Chapter 3.11.

4.6.1 Connect to the Field Controller

After the Field Controller and other components have been physically installed in the network, connection must be established as described in Chapter 3.10.

4.6.2 Create the HSE live list

Once the Computer and Field Controller are able to communicate with each other, the connection to the network can be checked by creating a live list as decribed in Chapter 3.11.1

1 Press the On-Line button in the menu toolbar
   – The project goes on on-line
   – Red crosses appear against the Field Controller and Fieldbus network in the Project workspace

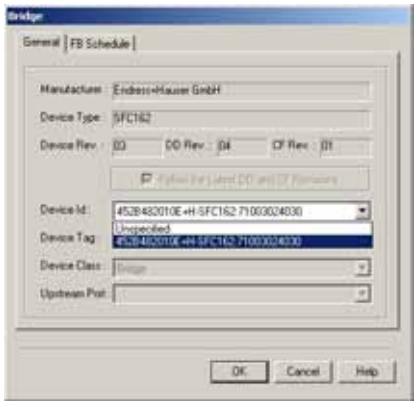
2 In the Project workspace, right click on HSE Network and select Live List
   – A live list is generated of the devices on the HSE network

4.6.3 Assign the HSE Device IDs

Assign the HSE Device IDs as decribed in Chapter 3.11.2

1 In the project workspace, right click on the Field Controller (CO104) and select Attributes...
   – The Attributes dialog opens

2 Open the drop-down menu of the Device ID and select the Field Controller that is associated with the displayed TAG (in our case CO104) - the serial number is on the front panel
   – Do this even though the correct ID is already displayed - the program expects it!
   – Confirm your choice with OK



3 In the HSE Live list, the Controller Icon goes grey, then reappears with the correct tag

4 Click on the Project View workspace and Export Tags..., see Chapter 3.9
   – Open Project File, then press Save Entire Configuration, to save the project

4.6.4 Create the FOUNDATION Fieldbus live list

Create the Foundation Fieldbus live list as described in Chapter 3.11.3

1 In the Fieldbus network workspace, right-click on the Field Controller (CO104) and select the option Live List: The Fieldbus live list is created

– Check that all the devices in the project appear in the live list
– If this is not the case, check connections, power etc.

4.6.5 Assign the Fieldbus Device IDs

Assign the Fieldbus Device IDs as decribed in Chapter 3.11.4.

1 In the fieldbus workspace, right click on TT100 and select Attributes...
– The Attributes dialog opens

2 Open the drop-down menu of the Device ID and select the Field Controller that is associated with the displayed TAG (in our case TT100) - the serial number is on the nameplatel
– Confirm your choice with OK
– After a short period of time, the red cross disappears from the device in the Fieldbus network

3 Repeat the process for the rest of the devices in the Fieldbus network (FCV102)
– After a period of time, the red cross disappears from the device in the Fieldbus network

4.6.6 Assign All Tags

1 In the Fieldbus workspace, right-click on the Fieldbus node at the top of the tree and select Assign All Tags

2 The Assign All Tags dialog appears with the list of device and a progress bar
– On completion, the message "Profile reading done" stands next to the Field Controller and "Tag has been confirmed" next to the devices

3 As the assignment proceeds, the Field Controller transmits the change to the device
– The device goes grey in the live list
– The Field Controller initiates a new network scan
– After about 1 - 2 minutes, the device goes black and appears with the new tag

4 At the end of the process the live list looks like this

5 The "bright" dot next to the SFC162 Field Controller indicates that it is the ACTIVE LAS of this segment

6 Click on the Project View workspace and Export Tags..., see Chapter 3.9
– Open Project File, then press Save Entire Configuration, to save the project

4.7 Download the project

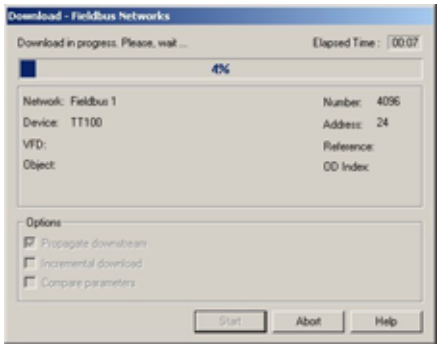
Note!

- The procedure below describes the initial download for the entire HSE network.
- Partial downloads can be made later from lower leaves, when changes are confined to this level
- Incremental downloads can be made to a running project by checking the boxes Incremental Download and Compare Parameters: Unaffected Local I/Os will hold their last values.

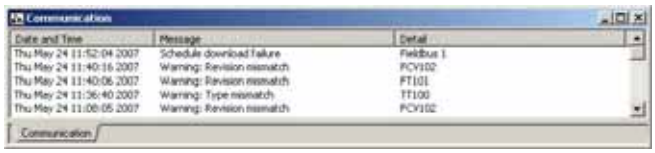
When the devices in the Fieldbus live list correspond to those configured in the project, the download can begin.

1 In the Project workspace right-click on HSE Network and select Download

2 The Download dialog appears



– Press Start to start the download
– A log at the bottom of the screen allows you to identify and non-critical download problems



3 The download will be interrupted if the project encounters a critial problem, e.g.

– The Project tags are not up-to-date => Chapter 3.10
– The Device IDs have not been assigned correctly => Chapter 6.2.2, Chapter 6.2.4
– The Device Tags have not been assigned => Chapter 6.2.5

4 When the download is successfully completed, the dialog is closed, and you are ready to test the control strategy

4.8 Make the Modbus Connection

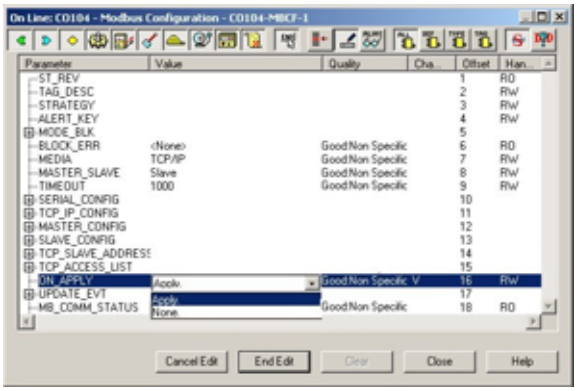
4.8.1 Start the Modbus

The Modbus must be started after the download (and restarted on every change in the Modbus configuration)

1 In the Fieldbus window, expand the Field Controller tree

– Right-click on the CO104 -MBCF-1 block and select On Line Characterization

2 In the On Line Characterization dialog
– Select All to display all parameters
– Select the parameter ON_APPLY



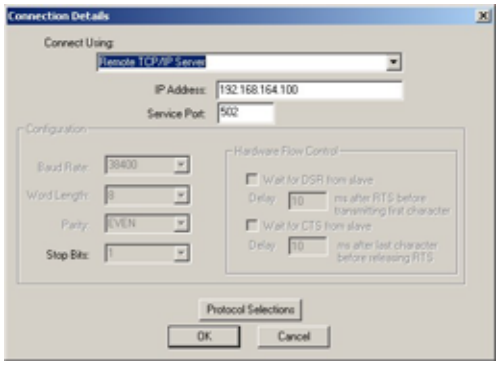
3 Click in the value space at the center of the line and open the drop-down menu
– Select Apply
– Click End Edit to confirm your choice
– The parameter remains for two or three seconds, then reverts to None
– The Modbus starts
– Press Close to close the dialog

Note!
• This procedure must be repeated every time the Modbus configuration is changed, including project download.

4.8.2 Start the Modbus simulator

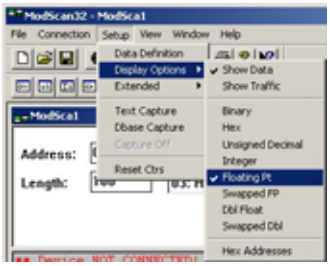
1 Start the ModSim32 or other Modbus simulator as master from the desktop or through the Explorer

2 Set up the communication by selecting Connections => Connect



- Set the Connect Using to Remote TCP / IP Server
- Enter the IP Address of the Field Controller (here default 192.168.164.100)
- Press OK to confirm the settings

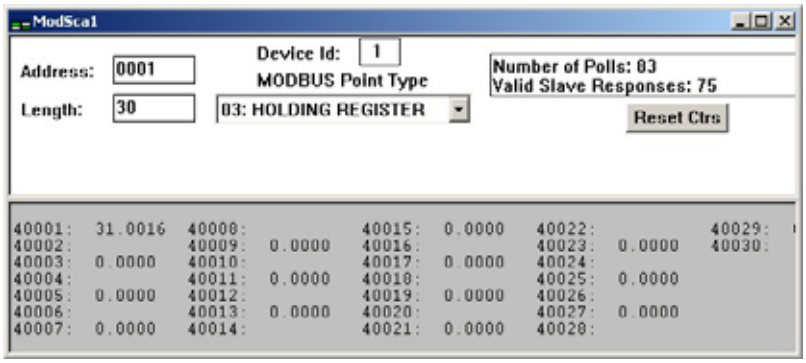
3 Set up the data display by selection Setup => Display Options



- Select Show Data
- Select Floating Pt

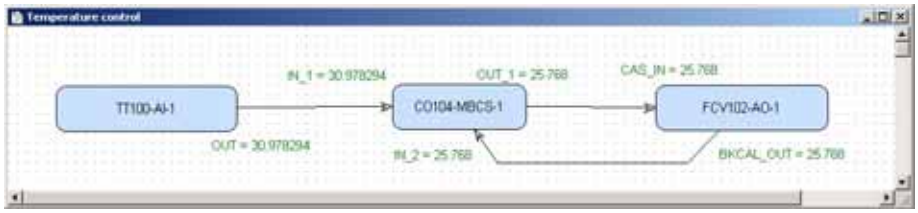
4 Select the view Input Registers from the drop-down menu in the user interface.
- Set the Address to 0001 and Length to 50
- You should now be able to see IN_1 and IN_2 in registers 30001 and 30003

5 Select the view Holding Registers from the drop-down menu in the user interface.
- Double-click on register 40001 and enter a value to simulate the OUT_1 value.



4.8.3 Check the control stategy

1 Click in the Control Strategy workspace (Temperature control) and press the button  in the menu toolbar – the control strategy goes "on-line"



– Values appear in green when the status is good
– Values appear in red if the status is bad - at this stage this is an indication of a Fieldbus configuration, a strategy configuration or a device parametrization error

4.8.4 Modify, export and close the project

See Chapters 3.13 to 3.15.

5 Field Controller as Modbus Master and Slave

5.1 Task Description

This part of tutorial describes all steps necessary for setting up the Field Controller as a Modbus Control Master for a Remote I/O and as Control Slave for a supervisory controller. It does not aim to give an exhaustive account of Application Designer functions, but rather shows you one of a number of methods to reach your goal. The tags and names used in the tutorial are imaginary and will be different in a proper application. A full description of Application Designer functions is to be found in Application Designer Overview BA017S/04/en. Function block descriptions are to be found in BA022/04/en, Function Block manual.

5.1.1 Application

For this part of the tutorial a remote I/O acts as a slave to the Field Controller and acquire two analog measurements and a limit value. After averaging in the Input Selector block, the voltage measurement is sent to the supervisory controller, e.g. PLC. This also receives the limit measurement. In this case the PLC acts as Modbus master and the Field Controller as Modbus slave. The corresponding control strategy is shown in Fig. 5-1.

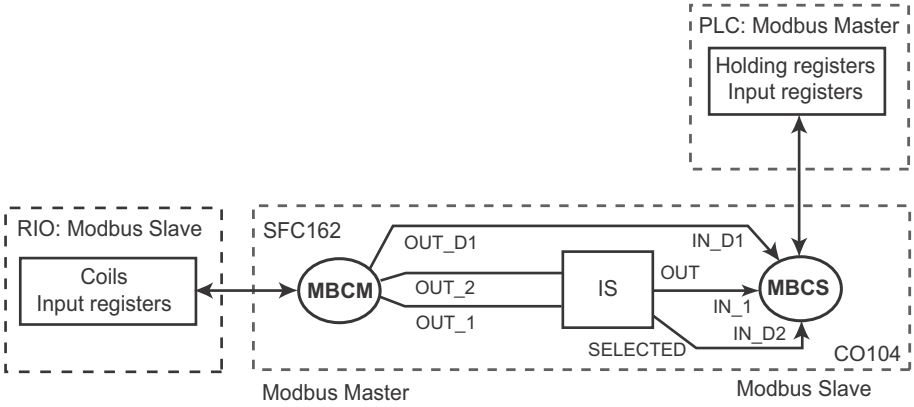


Fig. 5-1 Control strategy for Field Controller master and slave operation

Since it is not the object of this tutorial to describe the programming of a Modbus controller, the control function has been simulated by the ModSim32 software. This provides a simple means of testing the application and checking what is being transmitted across the Modbus interface. The simulator operates via Ethernet TCP, but other simulators are available for RS232.

Note!

- The Modbus standard restricts the maximum slave telegram length to 256 bytes, which corresponds to 125 measuring points of two bytes each plus six bytes overhead. If more measuring points are required, the master must be programmed to send more than one read request per cycle.

5.1.2 Network

The network is assumed to be constructed as shown in Fig. 5-2.

- The Modbus master is simulated by the ModSim32 application
- The Remote I/O is a WAGO 750-341 unit

In the example the SFC173 is used as Modbus Slave, but the SFC162could also be used.

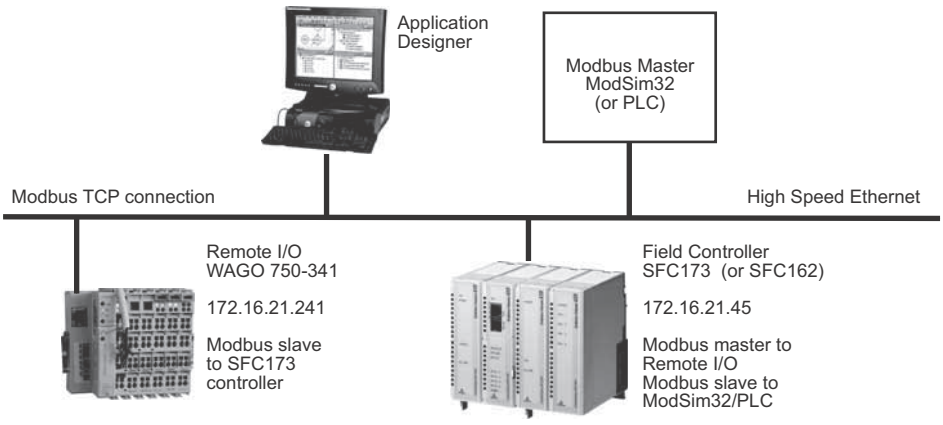


Fig. 5-2 Network for application example

5.1.3 Installation and commissioning

Before you can start this part of the Modbus tutorial, Application Designer must be installed on your computer, the SFC173 FOUNDATION Fieldbus Controller installed and commissioned and a connection made to your computer. Instructions on how to do this are to be found in:

- Operating Instructions BA020S/04/en, Getting Started
- Operating Instructions BA021S/04/en, Field Controller: Hardware Installion
- Operating Instructions BA035S/04/en, Field Controller: Commissioning and Configuration

5.1.4 Device ID and tag

In this tutorial, the default IDs and Tags values have been taken for all devices, including the field controller.

5.2 Create a project

5.2.1 Create the project and add the fieldbus

1 Following the instructions in Chapter 3.2 create a new project,
e.g. My Modbus Master and Slave

2 Following the instructions in Chapter 3.3, set the preferences

3 For the SFC 173 add a gateway and Profibus:
- The PROFIBUS Configurator opens
- As we have no devices, quit the program and PROFIBUS I/O Mapping dialog, click OK
- The PROFIBUS is added

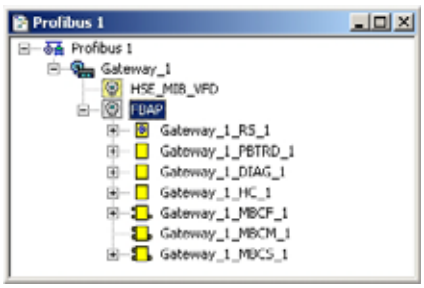
For the SFC162, follow the instructions in Chapter 3.4, add the bridge, fieldbus and devices
- In this case, do not add the MBCF and MBCM blocks to the controller yet.

4 Your project and field bus should now look something like this:

5.2.2 Add the Modbus blocks

1 Expand the Field Controller (CO104) leaf on the Fieldbus tree and add the Modbus blocks MBCF, MBCS and MBCM
- Right-click on the FBAP leaf and select New Block
- Select the Modbus Configuration block
- Repeat and select the Modbus Control Slave block
- Repeat and select the Modbus Control Master block

2 The Fieldbus tree now looks like this



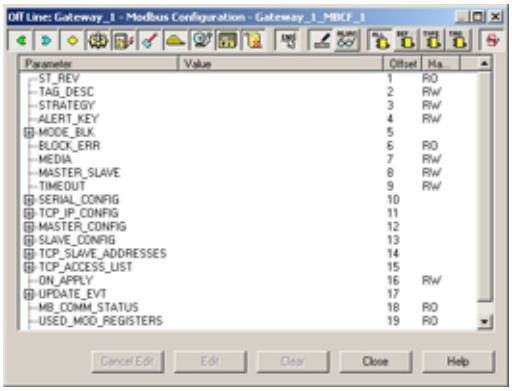
3 Open **Project File**, then press **Save**, to save the project

5.3 Configure the Modbus blocks

5.3.1 MBCF Modbus Configuration block

1 In the Fieldbus 1 window, right-click on the Field Controller function block Gateway-MBCF-1 and select Off Line Characterization

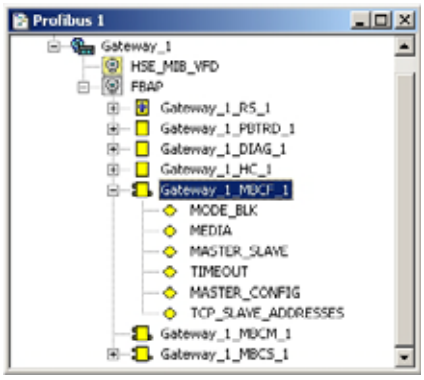
2 The Offline Characterization dialog opens: Press All to show all parameters



3 Set the following parameters by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCF
MODE BLOCK.TARGET	Normal operating mode of block	Auto
MEDIA	Channel for Modbus communication – Media 1 = Master channel, (Media 2) = Slave channel	TCP/IP (+TCP/IP)
MASTER_SLAVE	Role of Field Controller in Modbus network	Maaster + Slave
TIMEOUT	Time allowed for OUT value update	1000 (ms)
MASTER_CONFIG NUMBER RETRIES MAX_DATA_LENGTH	Configures Master interface • Number of retries is communication fails • Maximum length of dataa sent by Field Controller	 e.g. 3 50
SLAVE_CONFIG DEVICE_ADDRESS	Configures Controller when acting as slave • Modbus address of Field Controller	 ê.g. 12
TCP_SLAVE_ADDRESSES IP_SLAVE_1	List of masters allows to access field controller registers IP address of fisrt TCP slave	 172.16.21.241

4 Click Close to close the dialog: the parameters are added to the MBCF function block



5.3.2 MBCM Modbus Control Master block

The MBCM function block is described in detail in Chapter 2.3. Each MBCM block allows 16 Modbus registers to be accessed via its input and output parameters as follows:

- Four OUT channels read analog values from Modbus slave holding or input registers
- Four OUT_D channels read discrete values from Modbus slave coils or discrete input registers
- Four IN channels for write analog values to Modbus slave holding registers
- Four IN_D channels for write discrete values to Modbus slave coils

SCALE_LOC_XXX and LOCATOR_XXX allow each channel to be individually configured for slave address, register and in the case of analog values, data type and scaling.

Up to 16 MBCM blocks can be created, each having a unique identifier (0 – 15) determined by the LOCAL_MOD_MAP parameter.

The operating instructions of the Modbus device indicates which values are available in which registers. In the case of the WAGO 750-341, the Remote I/O is self-configuring. The values are arranged according to the order of the modules and inputs and can be read from the registers 40001 onwards. The discrete values can be read from registers 10001 onwards. The data is transferred in Integer16 format and the range extends from 0 to 32761.

Measured value	Register	Data type	Range	Access
Voltage 1	40001	Integer16	0 - 32761	Read
Voltage 2	40002	Integer16	0 - 32761	Read
Switch 1	10001	Integer16		Read

Tab. 5-1: Modbus registers for WAGI I/O

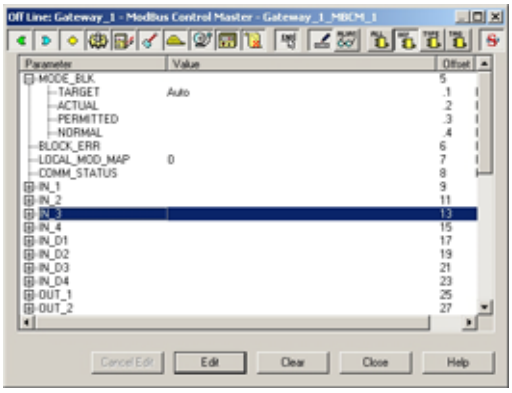
The values mapped to the Field Controller can be scaled in SCALE_LOC_XXX with the parameters:

- FROM_EU_0: lowest value that the Remote I/O can send
- FROM_EU_100: highest value hat the Remote I/O can send
- TO_EU_0: lower range limit of the scaled value for the Field Controller
- TO_EU_100: lower range limit of the scaled value for the Field Controller

In our example, the Remote I/O offers values from 0 to 32761 will be scaled from 0% to 100% by entering the range limits of the transmitter in the "FROM" parameters.

Procedure

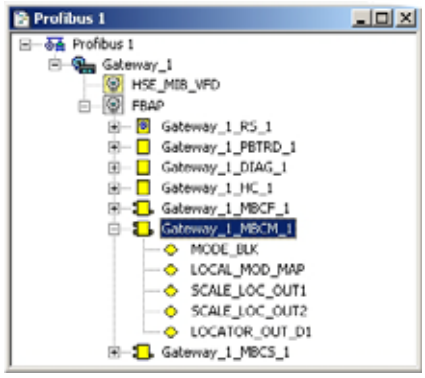
1 Right-click on the Gateway-MBCM-1 block and open the Off Line Characterization dialog



2 Click All to reveal all parameters and enter the following parameters by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCM
MODE BLOCK.TARGET	Normal operating mode of block	Auto
LOCAL_MOD_MAP	Identifier of Modbus block (first MBCM block)	0
SCALE_LOC_OUT1	Scaling and conversion of OUT_1 (flow)-	
FROM_EU_0	Lower range limit of WAGO Remote I/O signal	0
FROM_EU_100	Upper range limit of WAGO Remote I/O signal	32761
TO_EU_0	Lower range limit of OUT_1	0%
TO_EU_100	Upper range limit of OUT_1	100%
DATA_TYPE	Type of data transmitted	Integer16
SLAVE_ADDRESS	Modbus address of variable source (not required for TCP/IP)	0
MODBUS_ADDRESS_OF_VALUE	Reference address of input register	40001
SCALE_LOC_OUT2	Scaling and conversion of OUT_1 (flow)-	
FROM_EU_0	Lower range limit of WAGO Remote I/O signal	0
FROM_EU_100	Upper range limit of WAGO Remote I/O signal	32761
TO_EU_0	Lower range limit of OUT_1	0%
TO_EU_100	Upper range limit of OUT_1	100%
DATA_TYPE	Type of data transmitted	Integer16
SLAVE_ADDRESS	Modbus address of variable source (not required for TCP/IP)	0
MODBUS_ADDRESS_OF_VALUE	Reference address of input register	40002
LOCATOR_OUT_D1		
SLAVE_ADDRESS	Modbus address of variable source (not required for TCP/IP)	0
MODBUS_ADDRESS_OF_VALUE	Reference address of input register	10001

3 Press Close to close the Off Line Characterization dialog. You should now see the parameters attached to the MBCM block:



4 Open Project File, then press Save, to save the project.

5.3.3 MBCS Modbus Control Slave block

The MBCS function block allows fieldbus and Modbus data to be exchanged through 16 Modbus registers see Chapter 2.4. If more than four registers of a particular type are required, up to 16 MBCM blocks can be created. These are managed by the LOCAL_MOD_MAP parameter. By default LOCAL_MOD_MAP = 0, valid range 0 – 15.

The value of LOCAL_MOD_MAP is used to define a unique set of Modbus register address ranges for the particular MBCS block, whereby:

- Register address = Constant + 8 x Value of LOCAL_MOD_MAP for analog values
- Register address = Constant + 4 x Value of LOCAL_MOD_MAP for discrete/status values. where by Constant = 0 for e.g. IN_1, 1 IN_2 etc.

The Modbus master accesses the Field Controller slave registers by specifying the function code and the register address. Alternatively, the reference address can be used, whereby the relationship to the input and output parameters is as follows
.

FC parameter	Type	Master action	Scaling parameter	Reference address
IN_1 to IN_4	Analog input	Reads from input register	SCALE_CONV_XXX	30001 + register address
IN_D1 to IN_D4	Discrete input	Reads from discrete input	–	10001 + coil address
OUT_1 to OUT_4	Analog output	Writes to holding register	SCALE_CONV_XXX	40001 + register address*
OUT_D1 to OUT_D4	Discrete output	Writes to coil	–	1 + coil address*
*The holding registers and coils are Read/Write				

For the tutorial we require the IN_1, IN_D1 and IN_D2 channels. When LOCAL_MOD_MAP is set to zero, the data will be mapped to the registers in Table 5-2, see also Chapter 2.4.

Parameter	Channel	Register	Reference address	Data type	Master access
Average voltage	IN_1	0	30001	Float	Read
Status		128	30128	Integer16	
Switch value	IN_D1	0	10001	Float	Write
Status		128	10128	Integer16	
SELECTED	IN_D2	1	10002	Float	Read
Status		129	10129	Integer16	

Tab. 5-2: Modbus registers for exchange of data

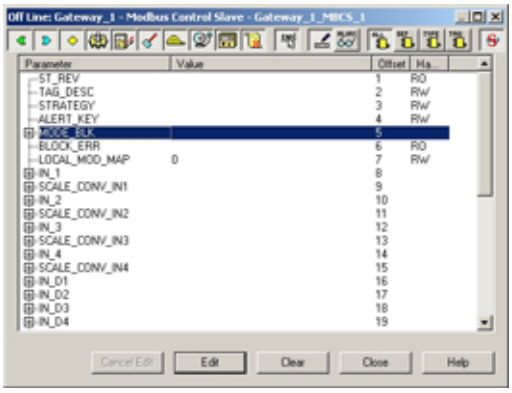
The SCALE_CONV_XXX parameter allows each analog channel to be individually configured for scaling and data type. The values mapped to the Field Controller or sent to a Modbus device register can also be scaled. In this tutorial, the paramters will not be scaled.

Parameter	SCALE_CONV_INx	SCALE_CONV_OUTx
.FROM_EU_0	Lower range limit IN_x value	Lower range limit Modbus master value
.FROM_EU_100	Upper range limit IN_x value	Upper range limit Modbus master value
.TO_EU_0	Lower range limit Modbus master value	Lower range limit OUT_x value
.TO_EU_100	Upper range limit Modbus master value	Upper range limit OUT_x value

The STATUS_OUTPUT element defines how the OUT.STATUS will be managed. As the status provided by the master is probably not in conformance with the FOUNDATION Fieldbus protocol, the option "Good: Cascade:NonSpecific:Not Limited" will be selected. This will always be transmitted by the Field Controller as the OUT status unless there is a timeout, in which case the OUT status will be set to Bad.

Procedure

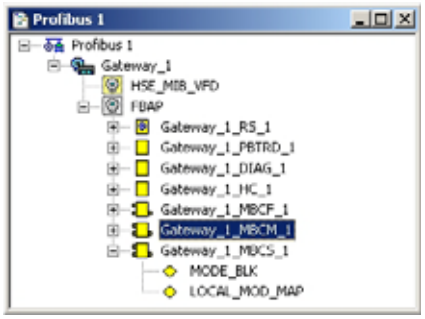
1 Right-click on the Gateway-MBCS-1 block and open the Off Line Characterization dialog



2 Click All to reveal all parameters and enter the following parameters by double-clicking in the middle of the parameter line, entering or selecting the parameter from the drop-down menu, and clicking End Edit to register the change:

Parameter	Function	MBCS
MODE BLOCK.TARGET	Normal operating mode of block	Auto
LOCAL_MOD_MAP	Identifier of modbus block (0 - 15)	0

3 Press Close to close the Off Line Characterization dialog. You should now see the parameters attached to the MBCM block:



4 Open Project File, then press Save, to save the project.

5.4 Create the Control Strategy

Having created a physical view of the process, the next step is to create control strategy. This is done described in Section 3.6.

5.4.1 Add a Process Cell

1 Click on the "Area 1" leaf in the project and select Attributes...

- The Attributes dialog box appears
- If desired, enter a name for the area
- Click OK to store your changes

2 Click on the Area leaf again and select New Process Cell..

- The Process Cell dialog box appears
- If desired, enter a name for the process cell
- Click OK to store your changes

3 Open Project File, then press Save, to save the project.

5.4.2 Add a Control Module

1 Right-click on the Process Cell leaf you just created and select Expand

- A new window with the name of the leaf opens

2 Right-click on the top leaf and select New Control Module

- The Control Module dialog box appears
- If desired, enter a name for the control module
- Click OK to store your changes

3 The project looks something like this

4 Open Project File, then press Save, to save the project.

5.4.3 Create the function blocks

1 Double-click on the control module leaf or right-click and select Expand to open the Control Strategy workspace - this has the same name as the leaf

2 Press the Function Block button in the toolbar and click in the workspace

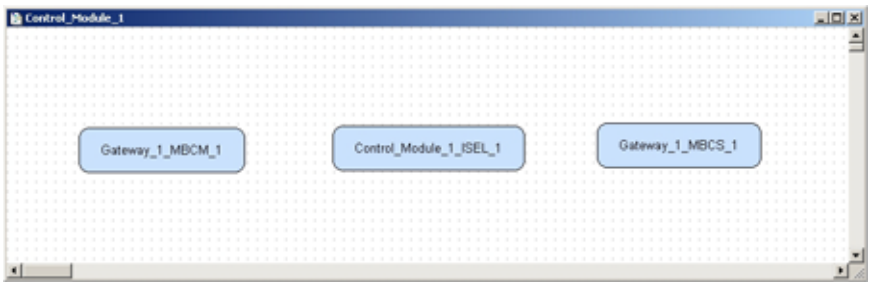
- The New Block dialog appears
- Select the Manufacturer = Endress+Hauser
- Select the Device Type = SPC173
- Select the Block Type = IS Block
- Press OK to create the function block

3 The block now appears in the strategy window with the default name

4 Drag and drop the Gateway_MBCS_1 and Gateway_MBCM_1 blocks from the Gateway FBAP node in the PRFIBUS window into the Control Strattegy workspace:

- The corresponding blocks are created

5 The control strategy now looks like this



6 Open Project File, then press Save Entire Configuration, to save the project

Note

- In the tutorial we created the MBCS and MBCM blocks block together with the MBCF block. Is is also possible to create the MBCS block in the strategy window as we have done with the IS block, and then drag and drop it to the Fieldbus, see Chapter 3.8.

5.4.4 Add the Function Block links

1 In the Control Strategy workspace position the blocks according to your strategy

2 Click on **Function Block Link** button in the tool bar, the cursor changes to a cross

3 Select the **Gateway_MBCM_1** blck with the cross: the **Output Parameter Selection** dialog appears

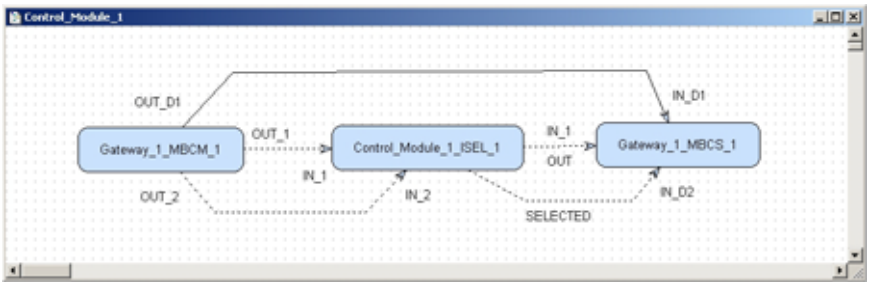
- Click the box next to **OUT1** – it changes color – then click on **OK**
- The **Output Parameter Selection** dialog closes
- The cursor is now connected to a blue dotted line
- Place the Cursor in the **Control_Module_1_ISEL_1** block and click to make the link
- When the link is made, the Input Parameter Selection dialog for the PID block appears
- Click the box next to **IN_1** – it changes color – then click on **OK**

4 If the Aliasing Input dialog box is enabled, see Chapter 3.3, the **Rename** dialog now appears
- Enter the desired link name and press **OK**
- If nothing is entered, the link retains the standard name

5 Repeat steps 2 to 4 and make the following links between the function blocks

- Gateway_MBCM_1 to Control_Module_1_ISEL_1 = **OUT2** to **IN_2**
- Gateway_MBCM_1 to Gateway_MBCS_1 = **OUT_D1** to **IN_D1**
- Control_Module_1_ISEL_1 to Gateway_MBCS_1 = **OUT** to **IN_1**
- Control_Module_1_ISEL_1 to Gateway_MBCS_1 = **SELECTED** to **IN_D2**
  (SELECTED is actually a discrete output with more than two values. Here we are misusing it to show how a binary value is shown in the master!)

6 Your Control Strategy now looks something like this



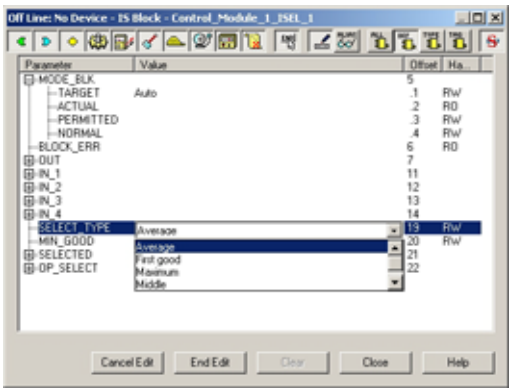
7 Open **Project File**, then press **Save Entire Configuration**, to save the project.

5.5 Configure the strategy

5.5.1 Configure the Input selector block

1 In the Control strategy workspace, double-click on the Control_Module_1_ISEL_1 block

2 The Offline Characterization dialog opens



3 Set the parameters as shown below

Parameter	Function	Temperature TC100
MODE BLOCK/TARGET	Normal operating mode of block	Auto
SELECT_TYPE	Action to be performed on the input values	Average

5.5.2 Attach the Function Blocks to the devices

Now attach the ISEL function block in the control strategy to the controller.

1 Expand the Fieldbus workspace (Profibus 1) and the Process Cell workspace

2 Now drag and drop the Control_Module_1_ISEL_1 block from the Process Cell tree to the greyed Function Block Application leaf of the Gateway_1 node in the Profibus_1 tree
- When you drop the block, it is attached to the Gateway
- The question mark in the yellow box in the Process Cell tree disappears
- Depending on your preference settings, the name may change in all trees and the strategy to that of the gateway

5.5.3 Export tags

1 Active the project view by clicking in its workspace.

2 Right click on the project name, a context menu appears
- Select the option Export Tags...
- The Export Tags dialog confirms the successful export
- Press OK to close the dialog

3 Open Project File, then press Save, to save the project

5.6 Go Online

Now go online as described in more detail in Chapter 3.11.

5.6.1 Connect to the Field Controller

After the Field Controller and other components have been physically installed in the network, connection must be established as described in Chapter 3.10.

5.6.2 Create the HSE live list

Once the Computer and Field Controller are able to communicate with each other, the connection to the network can be checked by creating a live list as decribed in Chapter 3.11.1

1 Press the On-Line button in the menu toolbar
   – The project goes on on-line
   – Red crosses appear against the Field Controller and Fieldbus network in the Project workspace

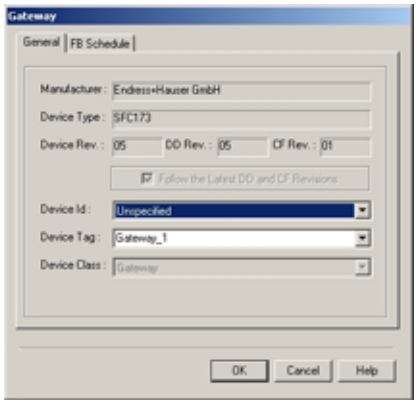
2 In the Project workspace, right click on HSE Network and select Live List
   – A live list is generated of the devices on the HSE network

5.6.3 Assign the HSE Device IDs

Assign the HSE Device IDs as decribed in Chapter 3.11.2

1 In the project workspace, right click on Gateway_1 and select Attributes...
   – The Attributes dialog opens

2 Open the drop-down menu of the Device ID and select the Field Controller that is associated with the displayed Tag - the serial number is on the front panel
   – Confirm your choice with OK



3 In the HSE Live list, the Controller Icon goes grey, then reappears with the correct tag

4 Click on the Project View workspace and Export Tags..., see Chapter 3.9
   – Open Project File, then press Save Entire Configuration, to save the project

5.7 Download the project

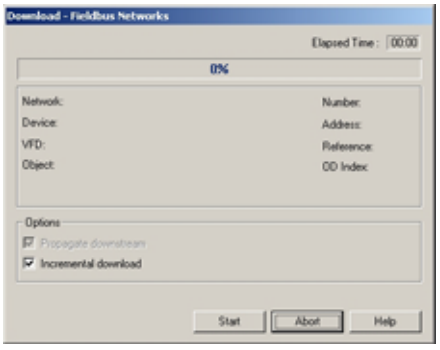
Note!

- The procedure below describes the initial download for the entire HSE network.
- Partial downloads can be made later from lower leaves, when changes are confined to this level
- Incremental downloads can be made to a running project by checking the boxes Incremental Download and Compare Parameters: Unaffected Local I/Os will hold their last values.

When the devices in the Profibus live list correspond to those configured in the project, the download can begin, see Chapter 3.12.1 for more details.

1 In the Project workspace right-click on HSE Network and select Download

2 The Download dialog appears
   – Press Start to start the download
   – A log at the bottom of the screen allows you to identify and non-critical download problem s



3 The download will be interrupted if the project encounters a critial problem, e.g.

   – The Project tags are not up-to-date => Chapter 3.10
   – The Device IDs have not been assigned correctly => Chapter 6.2.2, Chapter 6.2.4
   – The Device Tags have not been assigned => Chapter 6.2.5

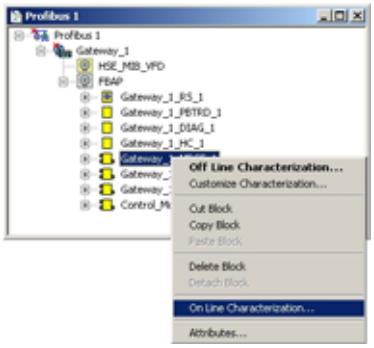
4 When the download is successfully completed, the dialog is closed, and you are ready to test the control strategy

5.8 Make the Modbus Connection

5.8.1 Start the Modbus

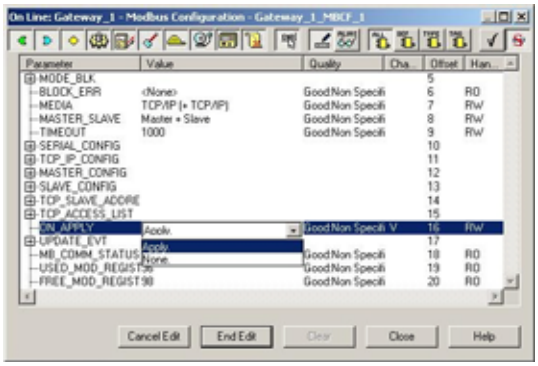
The Modbus must be started after the download (and restarted on every change in the Modbus configuration)

1 In the Profibus 1 window, expand the Gateway_1 tree



– Right-click on the Gateway_1_MBCF_1 block and select On Line Characterization

2 In the On Line Characterization dialog
– Select All to display all parameters
– Select the parameter ON_APPLY

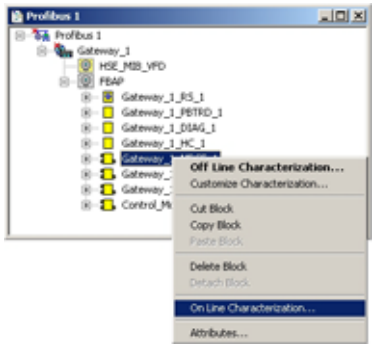


3 Click in the value space at the center of the line and open the drop-down menu
– Select Apply
– Click End Edit to confirm your choice
– The parameter remains for two or three seconds, then reverts to None
– The Modbus starts
– Press Close to close the dialog

Note!
• This procedure must be repeated every time the Modbus configuration is changed, including project download.

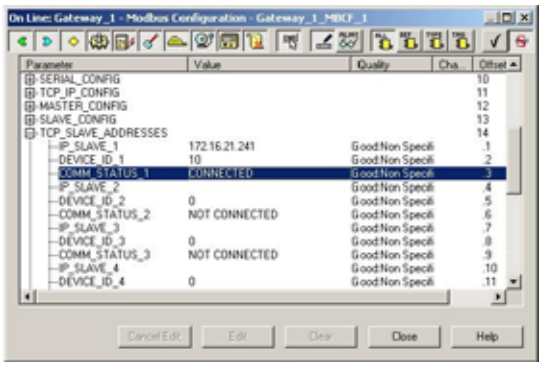
5.8.2 Check the connection to the Modbus slave

1 In the Profibus 1 window, expand the Gateway_1 tree

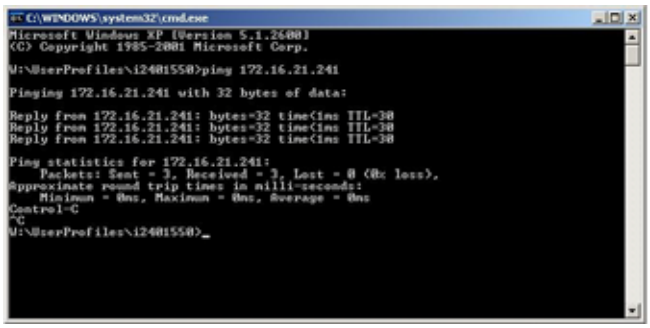


– Right-click on the Gateway_1_MBCF_1 block and select On Line Characterization

2 Open the parameter TCP_SLAVE_ADDRESSES
– The parameter COMM_STATUS_1 should show "CONNECTED"



3 If there is no connection, check the presence of the slave on the network by pinging the slave using the Command Prompt in Windows Accessories
– No response indicates an address error, network error or the slave is not powered up



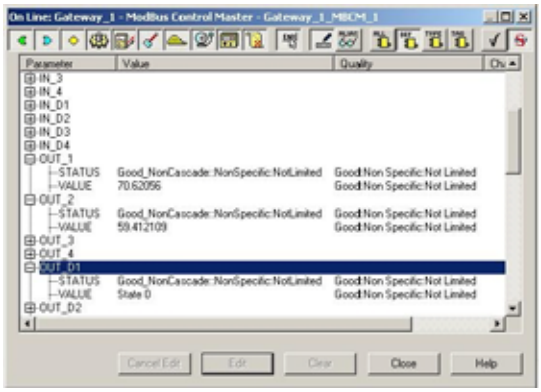
4 If the slave responds
– check its configuration
– check the MBCF block configuration

5.8.3 Check the slave input and control stategy

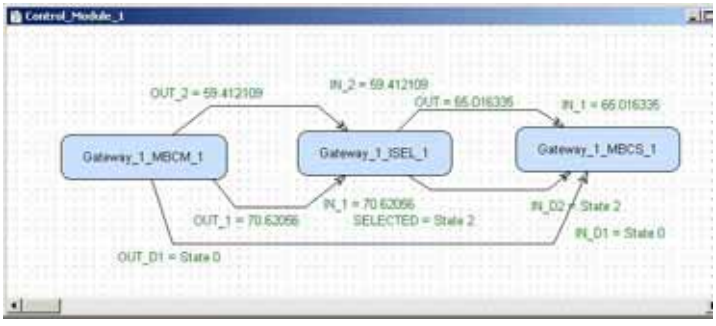
1 In the Profibus 1 window, expand the Gateway_1 tree

– Right-click on the Gateway_1_MBCM_1 block and select On Line Characterization

2 In the On Line Characterization dialog open the parameters OUT_1, OUT_2 and OUT_D1:
– The values of the contained in the Modbus registers are diaplayed:



3 Click in the Control Strategy workspace and press the button  in the menu toolbar – the control strategy goes "on-line"



– Values appear in green when the status is good
– Values appear in red if the status is bad - at this stage this is an indication of a Fieldbus configuration, a strategy configuration or a device parametrization error

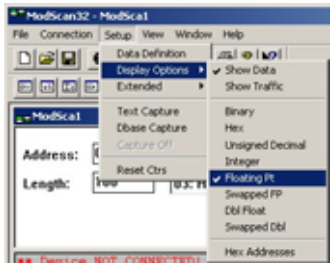
5.8.4 Check the master data

1 Start the ModSim32 or other Modbus simulator

2 Set up the communication by selecting Connections => Connect

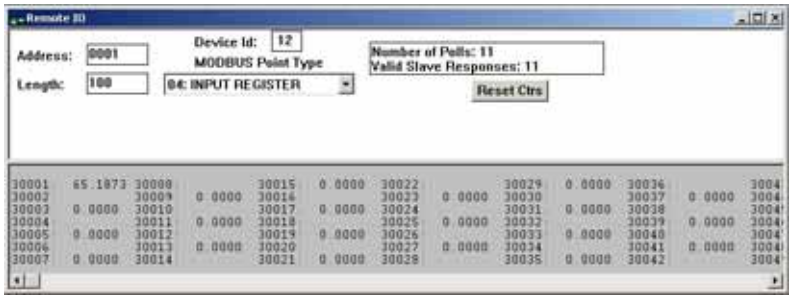
– Set the Connect Using to Remote TCP / IP Server
– Enter the IP Address of the Field Controller (here default 172.16.21.45)
– Press OK to confirm the settings

3 Set up the data display by selection Setup => Display Options



– Select Show Data and Floating Pt

4 Select the view Input Registers from the drop-down menu in the user interface.
– Set the Address to 0012 and Length to 50
– You should now be able to see IN_1 in register 30001



5 Select the view Input Status from the drop-down menu in the user interface.
– The values of IN_D1 and IN_D2 are in registers 10001 and 10002 respectively

5.8.5 Modify, export and close the project

See Chapters 3.13 to 3.15.

6 Trouble-Shooting

6.1 Factory initialisation and reset

Warning!

- Do not use the pushbuttons located in the Field Controller unless you are certain that you want to reset the system.

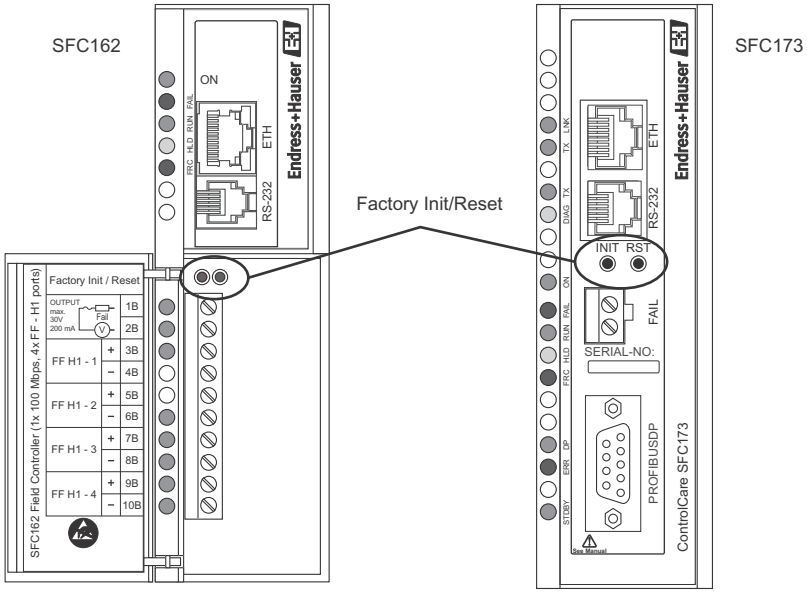


Fig. 6-1: Front panels of SFC162 and SFC173 Field Controllers

Two pushbuttons, located in the Field Controller module allow the system to be initialised and reset. The function and effect of the buttons is described in the table below.

- To "click " the pushbuttons use a pointed instrument (e.g. a ballpoint pen).
- Once started, any mode (Factory Init or HOLD Mode) can be exited by keeping the right pushbutton pressed and releasing the left pushbutton.
- If you loose the count of the times that the right pushbutton was pressed, check the rate at which the FORCE LED is flashing. It will return to a rate of once a second after the fourth touch (the function is rotative).

Function	Effect
Reset	Click the right pushbutton (see details in Fig. 5.1) and the system will execute the RESET, taking some seconds for correct system initialization. In accordance with the procedure via FC Tools, a new IP will be attributed automatically or the last configured IP will be accepted for the system. Verify that the RUN and ETH LNK LEDs remain lit.
Factory Init	Keep the left pushbutton pressed and then click the right pushbutton, checking that the FORCE LED flashes once a second. Release the left push button and the system will execute the RESET, deleting the previous configurations.
HOLD mode	Keep the left pushbutton pressed and then double click the right pushbutton, checking that the assuring that FORCE LED flashes twice a second.Release the left pushbutton and the system will execute the RESET and then enter the HOLD mode. Verify that the HOLD and ETH LNK LEDs remain lit. With the Field Controller in this mode, you can use the FC Tools Wizard to update the firmware or change the IP address. Use the Reset again, case you want to return to the execution mode (RUN).

6.2 Exchanging devices

It may be that during Device ID assignment, see Chapters 6.2.2 and 6.2.4, Application Designer detects a revision mismatch. If this occurs, the device revision must be changed in the project and the project downloaded as described below.

Note!

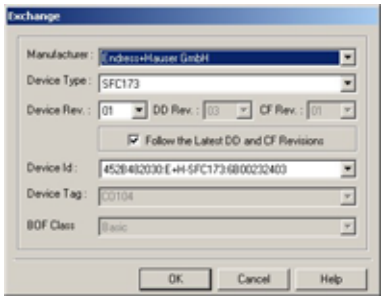
- For exchange of controllers, see Chapter 7.3, Updating the Firmware, in BA035S/04/en
- The process also applies to exchange of devices of the same type but different device revision
- In the case of devices from different manufacturers, manually check the incompatibilities (press NO at Step 3). Additional adjustments to the strategy may also be required.
- Device revisions not supported by Application Designer can be downloaded from www.fieldbus.org and integrated using the Import Device Support function

Procedure

1 Open the project in Application Designer and select the device to be updated, here SFC162

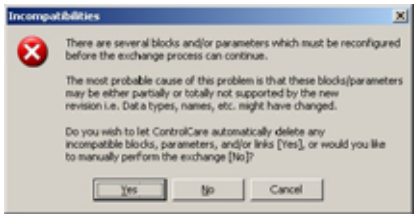
– Right-click on the node and select Exchange...

2 The Exchange dialog for the device appears



– Open the Device Rev menu and select the correct revision
– Press OK to confirm the change and close the dialog.

3 The Incompatibilities dialog may now appear



– Press OK to automatically update all links
– Close the successful exchange dialog by pressing OK.

4 Export the tags, press  to go on-line and download the project as described in Chapter 3.12.

6.3 Trouble-shooting tables

6.3.1 Field Controller

	Problem	Remedy
1	HSE Network Setup/FC Tools does not find any Field Controller	• Disable the Windows firewall (normally a message appears ask whether you should unbock the program) • Disable the proxy server for your Internet browser • Check that you are using the correct Ethernet cables, see ETH LINK below • Check that all etehrnet switches are powered up • Check that the network adapter is on and OK: Execute a PING command to its own IP, via DOS PROMPT. • Check if the Ethernet connection is OK: Execute a PING command to the Field Controller.
2	Field Controller appears intermittently in FC Tools	Host and Field Controller are in different subnets. • Normal behaviour, but for firmware download both host and Field Controller must be in the same subnet
3	HSE Network Setup/FC Tools does not show all the Field Controllers that are in the network	There is probably an IP address conflict in the network. • Disconnect all the Field Controllers except one from the from the sub-network • If necessary, change its IP address • Now reconnect the other Field Controllers one after the other, if necessary changing their IP addresses
4	Field Controller Web Server does not open	No Ethernet connection • Disable the Windows firewall • Disable the proxy server for your Internet browse • Wrong subnet IP address Host and Field Controller must be in same subnet • Wrong subnet mask Host and Field Controller must have same subnet mask
5	Firmware begins to execute but after a certain time it stops	It might be a configuration problem. • Use the Factory Init procedure and configure the Field Controller again. • If the problem persists, see the relevant chapter in Operating Instructions BA035S/04/en, Field Controller, Commissioning and Configuration
6	HOLD LED remains lit	If the HOLD LED remains lit after the Field Controller has been turned on, the firmware may be invalid. • Update the firmware, see the relevant chapter in Operating Instructions BA035S/04/en, Field Controller, Commissioning and Configuration
7	ETH LNK LED does not light	Check if the cable is connected correctly, or that the cable is not damaged. Check the specification of the cables: • SFC954 - Cable Standard. To be used in a network between the Field Controller and a Switch/Hub. (preferred configuration) • SFC955 - Crossed Cable (Cross). To be used point to point between a PC and the Field Controller (some PCs/laptops may have problems with crossed cable)
8	FRC LED is flashing (Force)	Field Controller is powered up for the first time • Battery is not switched on (see BA021S/04/en, p50) Field Controller is in reset mode • Complete the RESET procedure Field Controller is in normal operation • Battery is flat: – No problem if controller remains powered up – If power is switched off, the project will be lost and must be downloaded again from Application Designer on repowering

6.3.2 Application Designer

	Problem	Remedy
1	Field Controller does not appear in HSE live list	No connection to Field Controller • See Remedies for Items 1, 2 and 4, Chapter 7.3.1 • Field Controller is on HOLD, set it to RUN mode • IP address is not configured correctly, use PING to check
2	Field Controller appears but always stays grey in HSE Live List	No connection to Field Controller • Check that host and Field Controller are in same subnet
3	Red cross appears on the Field Controller	No communication with Field Controller • No Ethernet connection with Field Controller, check connection, IP address etc, see above • No Device ID set in the Field Controller (Attributes)
4	Red cross appears on Fieldbus/Profibus	No communication with fieldbus/Profibus • No communicaton with Field Controller, see above • Fieldbus/Profibus not connected to controller
5	Red cross appears on field device	No communication with fieldbus device • No communicaton with Field Controller, see above • No communicaton with fieldbus, see above • No Device ID set (Attributes) • Tag not assigned (Assign Tag)
6	A device does not appear in the live list	Communication error • The device is not powered up • The project has been updated but no download has been made yet
7	Configuration will not download	You have either a communication problem or the configuration is not complete • Check that you are on-line - press the On-line button • Check that your computer is in the same address subnet • Check that you have assigned the Field Controller tag • Check that you have exported all tags OPC server • Check that the parameters are in the recommended order • Check that the OPC server is running (look for icon in bottom line) • Try "Update" from the Field Controller node and download again
9	Parameter appears red in the on-line control strategy	The parameter has a bad status • Check that the Block Mode is Auto (or Cas) • Check that the block has been correctly configured • Check that the device is still live (live list) • Check that the device address is the same as that you have in your configuration (live list) • Check that the parameter has been correctly configured • Check that the tags were exported (Export Tags)
10	FB links do not work	Project not downloaded correctly, e.g. partial download when bridge has HSE links • Repeat full download from the HSE Network node

6.3.3 Modbus

Problem	Remedy
No communication via Modbus RS-232	• If you have changed the configuration of a Modbus block, check that you have restarted the bus by using the ON_APPLY parameter, Chapter 3.17 • Check that the MEDIA parameter is correct, TCP or serial, Chapter 3.5.1 • Check that the Field Controller and Modbus device are using the same communication settings: RTU/ASCII, Data bytes, Stop bits, Parity, Baudrate, Chapter 3.5.1 • Check that you are using the correct Modbus Block, Chapter 3.5.2, Chapter 4.3.2 • Check that you are using the correct Modbus addresses • Check that the Controller connector, all cables and any interfaces, e.g. RS-232/RS-485 are correctly wired
No communication via Modbus TCP	• Check that the IP addresses are in the same domain • Check that any slave IP addresses have been property entered, Chapter 6.2 • Check that the correct cables have been used, see Chapter 5.2, ETH LNK LED and any switches etc. are powered • If you have changed the configuration of a Modbus block, check that you have restarted the bus by using the ON_APPLY parameter, Chapter 3.17
A Modbus block does not switch to "Auto" but remains "OOS"	• Check that the Mode Block target is set to "Auto" • Check that the Local Mod Map of each Modbus Block has a unique identifier between 0 and 15 • Check that the Modbus has been started by using the ON_APPLY parameter, Chapter 3.17
Local Mod Map identifier is not accepted	• Check that the value is between 0 and 15 • Check that no other Modbus block of the same type is using the value you entered
A static value in the Modbus Block was changed, but the value does not update.	• Put the block out of service (OOS) before editing the parameter. • After the edit, put the block back to "Auto" and restart the bus with ON_APPLY
The Modbus parameters cannot be found or displayed	• Check that you are looking at the correct registers • Check that the data format has been set correctly Chapter 6.
Status of Modbus value always bad	• Check TIMEOUT parameter is greater than Modbus master write cycle, see Chapter 4.3.1

Index

www.endress.com/worldwide

Endress+Hauser

People for Process Automation