



# M1710 Modbus Communication

TM1710DM22R Master & Slave configuration

Aidin Aliyazadeh

Technical Trainer

Marktheidenfeld/Germany, June 2016

# Agenda

**Final Destination/Function Description**

**Hardware**

**Slave Configuration**

**Master Configuration**

**1<sup>st</sup> Method: Generic Modbus RTU**

**2<sup>nd</sup> Method: Target Blocks**

**Modbus Monitoring & Debugging**

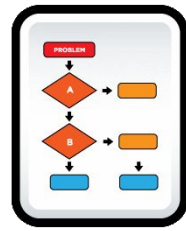
**Appendix 1, Function Codes**

# Final Destination

Function Description

# Exercise Goal

## Function Description



1. Read Slave's NTC probe value by Master controller via Modbus Serial Line.
2. Write to Slave's EEPROM Parameters by Master controller via Modbus Serial Line.
3. Enabling Modbus Master by related Function Block.
4. Enable/Disable Enumerators creation.
5. Menu creation to access port activation/deactivation from front face of product.
6. Modbus protocol properties configuration on both parties.
7. Continuous Modbus Communication watching/monitoring by dedicated vector.

Note: It is **not** needed to configure Modbus Read/Write function codes, protocol... in Connection Tool.

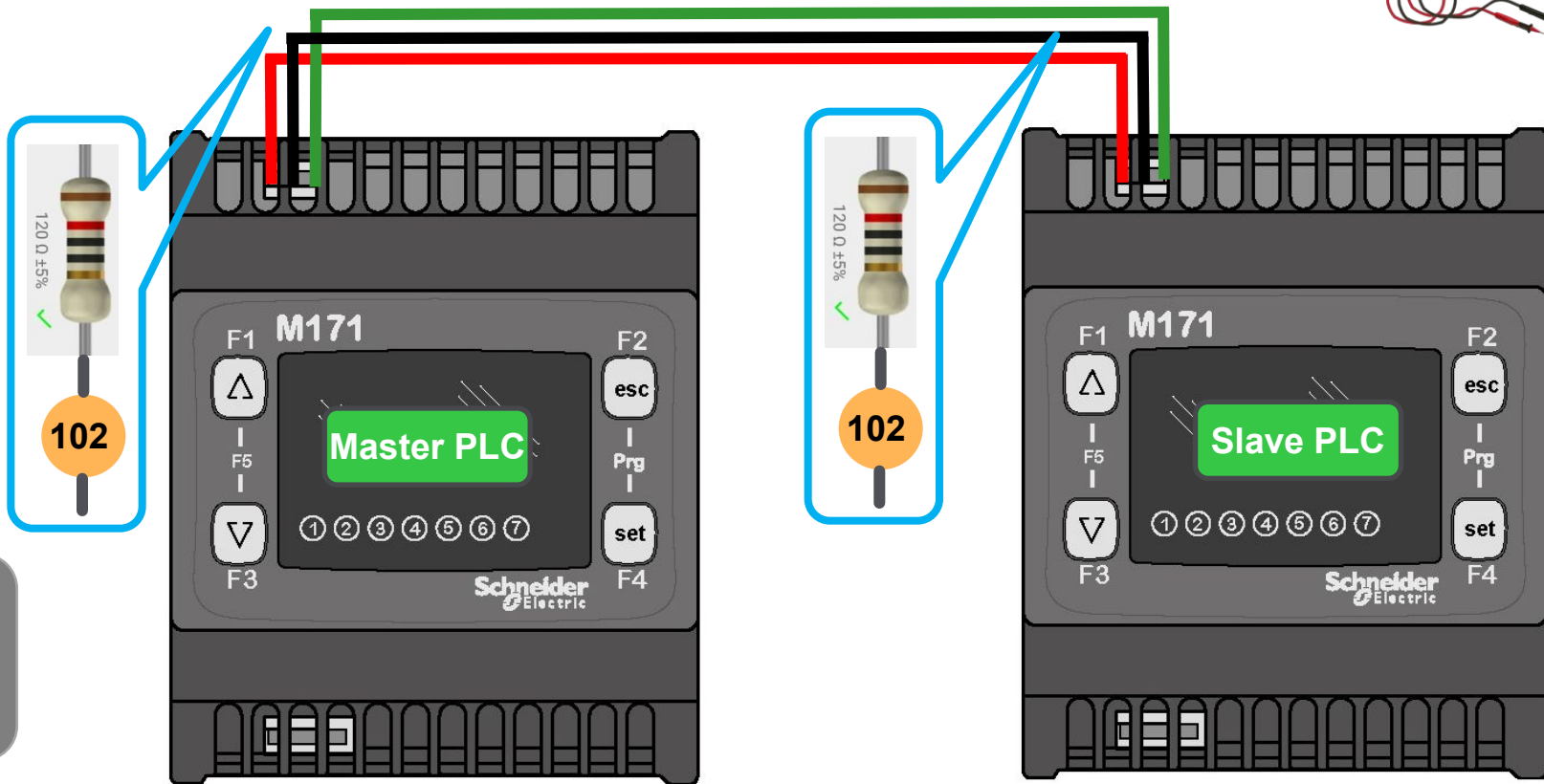
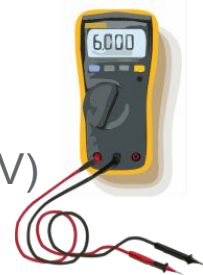
# M171O Controller Hardware

TM171ODM22R Modbus wiring and termination

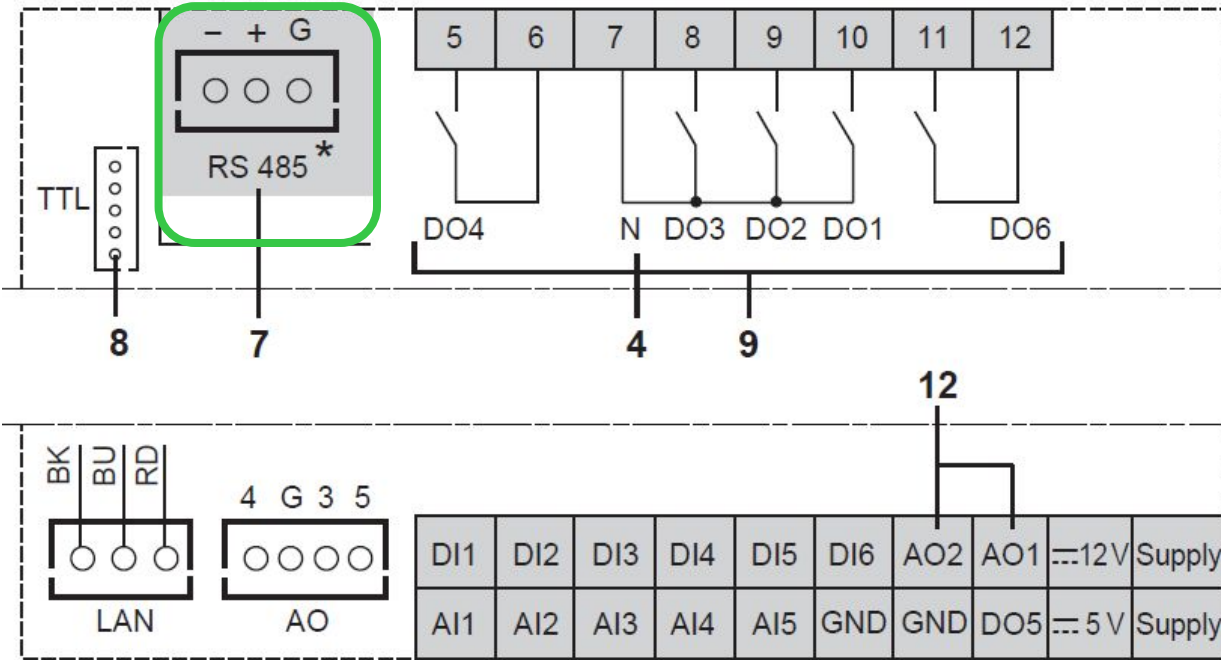


# M171 Optimized controller Hardware

TM171ODM22R Modbus Wiring & Termination (120 Ohms, 0.25 W series with 1nF, 10 V)



# TM171ODM22R Pin-Out & Modbus wire

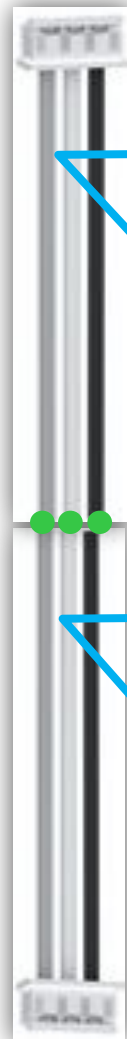


**7 - RS485 Serial for TM171O\*\*M\*\*\***

## Modbus Wire/Connection

Modbus SL connector  
**TM171ACB4ORS485**  
 Cordset equipped with a 3-pin connector on one end, 1 m length

Modbus SL connector  
**TM171ACB4ORS485**  
 Cordset equipped with a 3-pin connector on one end, 1 m length



# M171O Controller

Modbus Serial line Configuration, **Slave Side**





# Creating New project

## Slave Side



Welcome to Application

New project ...

Name: M171O\_Slave

Directory: C:\Electrical\Solution Architect\HVAC\Exercise\

Target selection		
	M171P Display	423
	M171P Blind	477
	M171P Flush Mounting	489
	M171O Modbus Slave Only	412
	M171O	542
	M172P	596

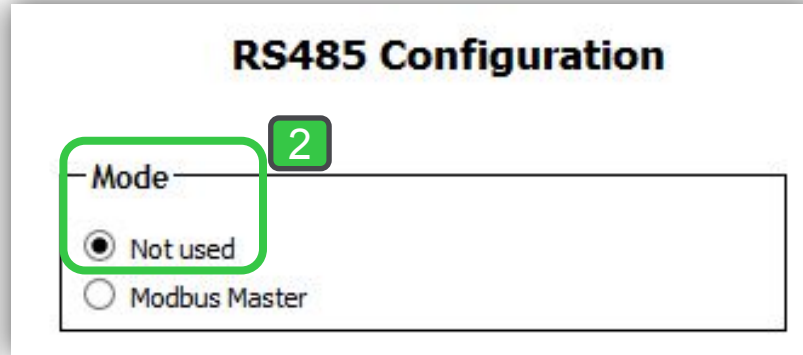
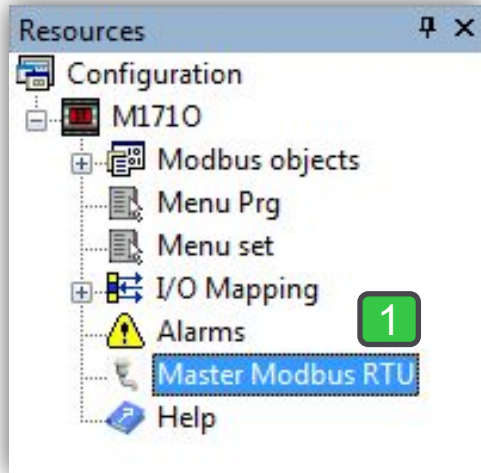
Case sensitive

Note: As a slave both 412 or 542 mask versions controllers could be use.



# Modbus RTU configuration/Slave Mode

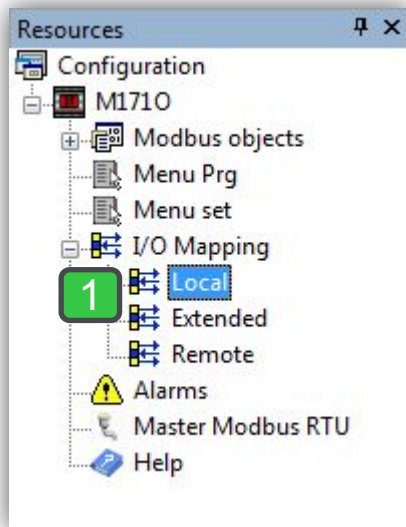
Slave side



Note: Not used taken into account as Slave

# Physical I/O Mapping

Slave side



## Local I/O Mapping **2**

#	Name	Variable	Type	Description
1	AIL1	NTC_Probe	INT	AIL1 analogue input
2	AIL2		INT	AIL2 analogue input
3	AIL3		INT	AIL3 analogue input
4	AIL4		INT	AIL4 analogue input
5	AIL5		INT	AIL5 analogue input
6	DIL1		BOOL	DIL1 digital input
7	DIL2		BOOL	DIL2 digital input
8	DIL3		BOOL	DIL3 digital input
9	DIL4		BOOL	DIL4 digital input
10	DIL5		BOOL	DIL5 digital input
11	DIL6		BOOL	DIL6 digital input
12	DOL1	Output_Cooling	BOOL	DOL1 digital output
13	DOL2	Alarm	BOOL	DOL2 digital output
14	DOL3		BOOL	DOL3 digital output
15	DOL4		BOOL	DOL4 digital output
16	DOL5		BOOL	DOL5 digital output
17	DOL6		BOOL	DOL6 digital output
18	AOL1		INT	AOL1 analogue output
19	AOL2		INT	AOL2 analogue output
20	AOL3		INT	AOL3 analogue output
21	AOL4		INT	AOL4 analogue output
22	AOL5		INT	AOL5 analogue output
23	TCL1		INT	TCL1 analogue output



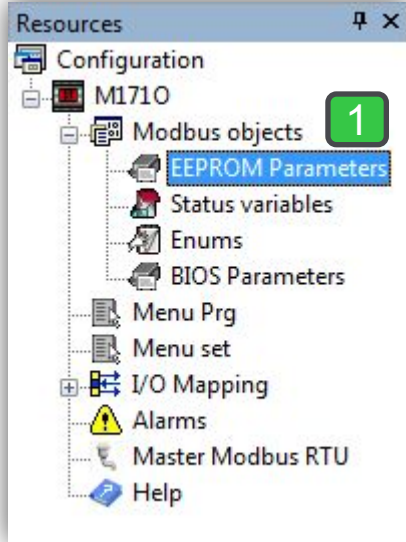
Note:

Schneider  
Electric

# EEPROM parameters definition/list



Slave side



Note:

## EEPROM Parameters

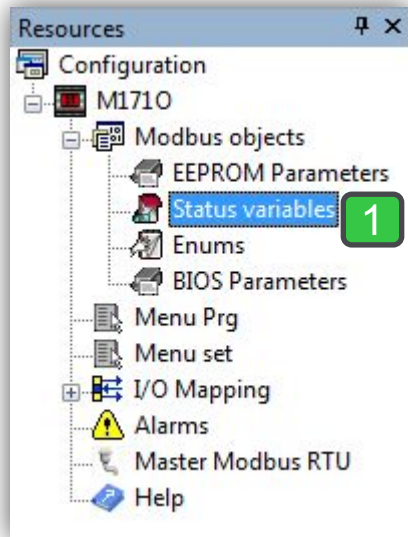
Add   Remove   Recalc

#	Address	Name	Display label	Device type	Application type	Default value	Min	Max	Unit	Format	AccessLevel
1	16384	SetPoint	SetP	Signed 16-bit	INT	180	120	300	°C	XXX.Y	Always visible
2	16385	Delta	Dlta	Signed 16-bit	INT	25	10	50	°C	XXX.Y	Always visible

2

# Status Variable definition/list

Slave side



### Status Variables

Add   Remove   Recalc

#	Address	Name	Display label	Device type	Application type	Read only
1	8960	Ambiant_Temp	ATMP	Signed 16-bit	INT	True

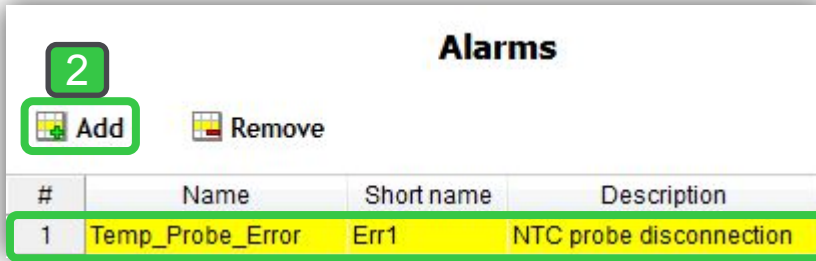
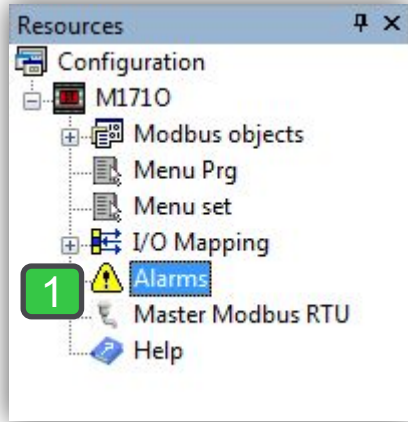
**2**

Note:



# Alarm definition/list

Slave side



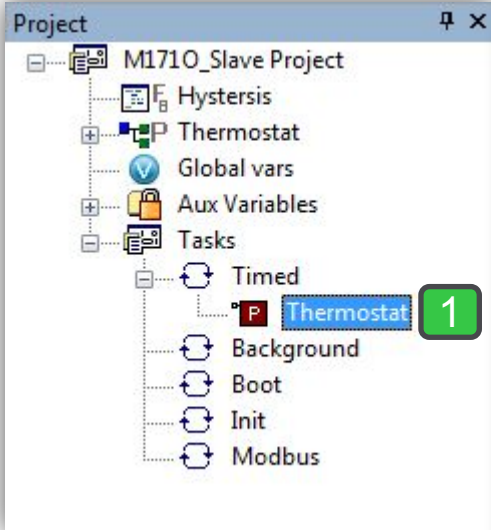
## Note:

Alarm list definition advantages are as below:

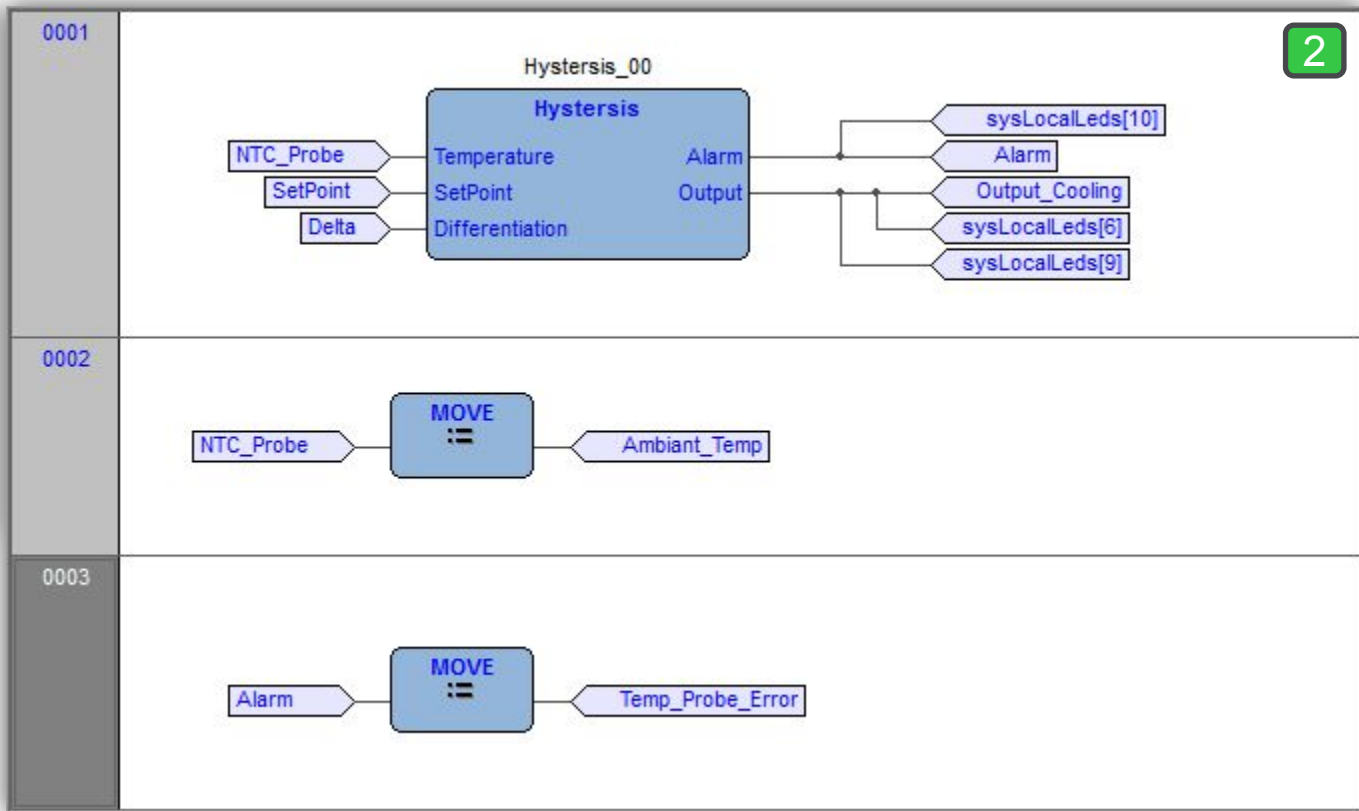
1. Automatic foldering when an alarm raised and shrink the list pack when it disappeared
2. Automatic launching of hazard sign/icon in front face of product (system LED)

# Thermostat Program

## Slave side



1



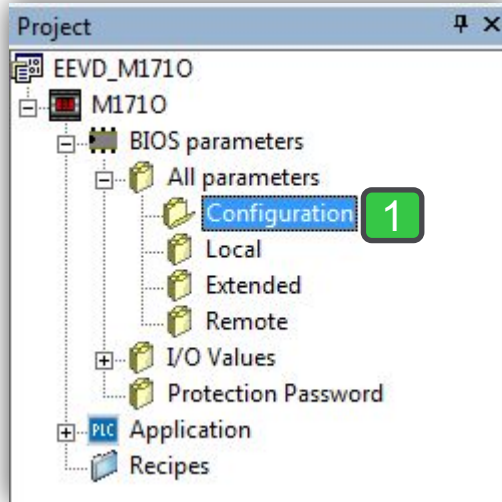
2

Note:

# Modbus Protocol Properties



Slave side



## 2 Configuration

Address	Name	Value	Um	Default	Min	Max	Description
53276	CF32	1=Even	num	1=Even	1	3	Modbus parity protocol
53274	CF30	1	num	1	1	255	Modbus protocol controller address
15636	Par_POLI	0	num	0	0	65535	Polycarbonate code
15715	Ui26	350	4ms	350	0	999	Key hold time to enable function
53456	CF50	1=Present	num	1=Present	0	1	RTC present
15640	CF61	0	num	0	0	999	Customer code 2
15639	CF60	0	num	0	0	999	Customer code 1
15744	Ui27	1	num	1	0	255	Installation engineer password
53275	CF31	3=9600	num	3=9600	0	7	Modbus baud rate protocol
15745	Ui28	2	num	2	0	255	Manufacturer password
53273	CF21	0	num	0	0	14	Protocol controller family
53272	CF20	0	num	0	0	14	Protocol controller address
53265	CF01	1	num	1	0	1	Select COM1 protocol

Note:



# Target and SoMachineHVAC



Parameters needed for correct connection between the **M1710 targets** and SoMachineHVAC

5. Set them as default value

parameter	description	values	default	visibility	notes
1 CF30	Modbus protocol controller address	1...255	1	3	Check that the set values correspond to those defined by the panel <b>Communication &gt; Settings &gt; Properties</b>
CF31**	Modbus protocol baud rate	0,1, 2 = not used 3 = 9600 baud 4 = 19200 baud 5 = 38400 baud 6 = 57600 baud 7 = 115200 baud	3	3	
3 CF32	Modbus protocol controller parity	1 = EVEN 2 = NONE 3 = ODD	1	3	

\*COM1 = TTL / RS485 (/S models only): cannot be used simultaneously

\*\*CF31

5=38400 baud (RS485: not supported)  
6=57600 baud (RS485: non supported)  
7=115200 baud (RS485: non supported)

4. Maximum supported baud rate is 19200

# Customize M171O Baud Rate

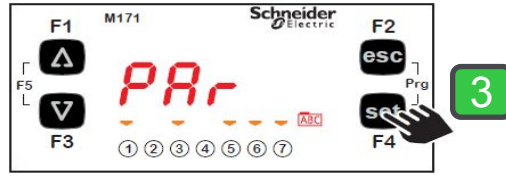


M171O parameters in the CF folder manages the connection between the target and Studio

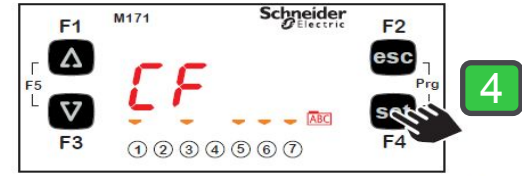
If the target is “empty”, i.e. there is no IEC application on the device, M171O will display the message FrEE, otherwise fundamental state is displayed (**Press F5 to switch to FrEE menu**)



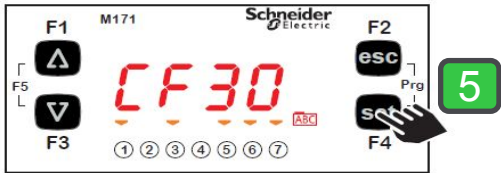
To view the parameter menu, press the Esc and Set keys at the same time. This will open the PAR menu.



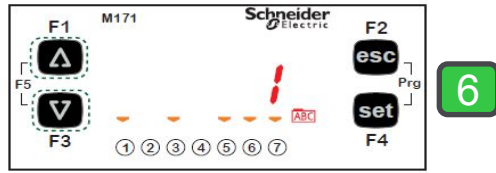
The parameters menu PAR contains all controller folders. Press the set key to view folders.



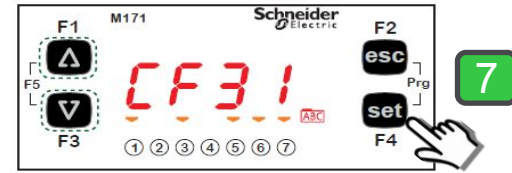
The first folder shown is the CF configuration folder. Press the set key to view the folder parameters.



The first parameter shown is CF30. To view the value of the parameter press the set key.



Use the UP and DOWN keys to change the value if necessary. To confirm the value press the set key. To exit press Esc



Use the UP and DOWN keys to scroll the other parameters and repeat the procedure to view the values and - if necessary - edit them.

# M171O Controller

Modbus Serial line Configuration, **Master Side**



# M171O Modbus Limitations

## Master Side

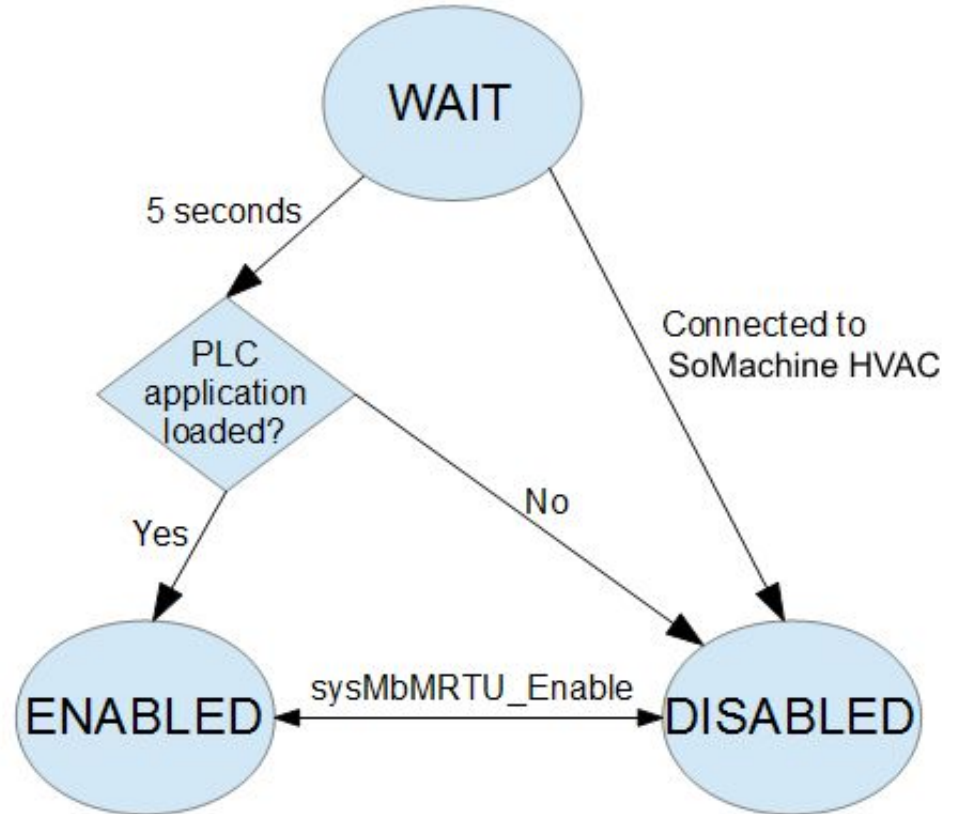
- Maximum number of nodes 127
- The maximum number of Modbus message is related of the available RAM in the device.
- Each message is a Function block that use around 64 byte of the RAM Memory.
- Each Modbus message can link up to 8 StatusVariable/Parameter/



# Modbus RTU master enable status

## Master Side

Since the communication port used by the Modbus RTU master is the same used by SoMachineHVAC to connect to a Modicon M171 programmable controller, you cannot connect to the device if the Modbus RTU master is running.



# Modbus Master Disabling methods,

## Master Side



### 1. Temporarily:

Delayed start: after the controller is powered on, it waits for approximately 5 seconds for a connection with the development environment to be established, before loading the PLC application and enabling the Modbus RTU master; if the connection is established, the PLC application is loaded but the Modbus RTU master is not enabled;

### 2. Permanently:

Run-time disable: Modbus RTU master can be dynamically disabled and enabled again by the PLC application (through a call to the target FUNCTION sysMbMRTU\_Enable), to allow SoMachine HVAC - Application's debuggers to attach to the target device even if the PLC application has already been started.

# Creating New project

Master side, Controller selection



Welcome to Application

New project ...

1

Name: M1710\_Master

Directory: C:\Electrical\Solution Architect\HVAC\Exercise\

Target selection		
	M171P Display	423
	M171P Blind	477
	M171P Flush Mounting	489
	M171O Modbus Slave Only	412
	M171O	542
	M172P	596

Case sensitive

2

Note: Only mask version 542 can support Modbus Master Functionality



# Modbus Master

sysMbMRTU\_Enable

Enable/Disable Modbus master: that allows to dynamically switch to programming/debug mode

Type : FUNCTION

Code type : EMBEDDED

Input vars num: **1**

enable : BOOL (\* If TRUE, enables Modbus master; otherwise, disables it \*)

Result type : BOOL

Note: By default Modbus Master Functionality is deactivated that's why it is needed to activate it by sysMbMRTU\_Enable function.



# Modbus Master Enable/Disable

Master side



**View object properties**

**Name:** sysMbMRTU\_Enable

**Type:** Function

**Return Value:** BOOL

**Language Type:**

**Description:**  
Enable/Disable Modbus master: that allows to dynamically switch to programming/debug mode

**Input:**

Name	Type	Description
enable	BOOL	If TRUE, enables Modbus master; otherwise, disables it

**Close**

Library

sysClockWrite	sysMbMRTU_BroadcastFC16	sysl
sysExecutionPassword	sysMbMRTU_Enable	sysl
sysMbMRTU_BroadcastFC05	sysMbMRTU_FC01	sysl
sysMbMRTU_BroadcastFC06	sysMbMRTU_FC02	sysl
sysMbMRTU_BroadcastFC15	sysMbMRTU_FC03	sysl

Operator and standard blocks | Target variables | Target blocks

1

sysMbMRTU\_Enable

enable

2

Note:

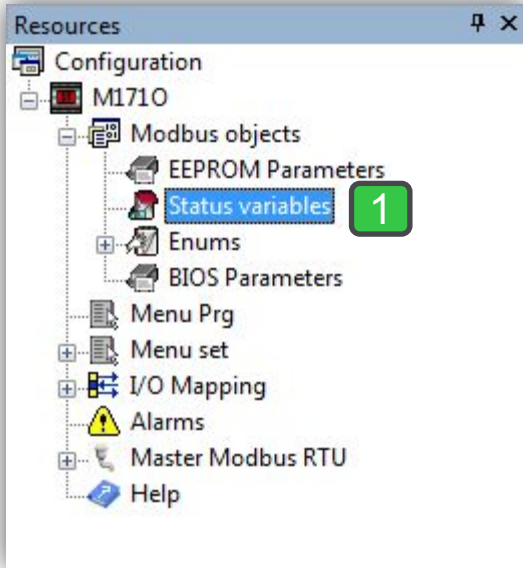
Life Is On





# Status Variable Defenition

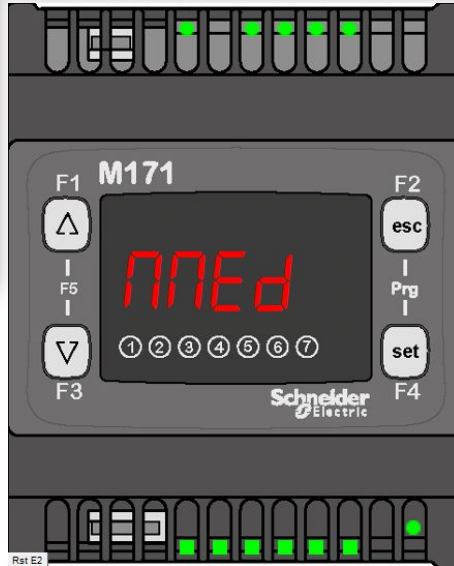
Master side, to Activate/Deactivate Modbus Master Functionality



### Status Variables

Add Remove Recalc 2

#	Address	Name	Display label	Device type	Application type	Default value	Read only
1	8960	Enable_Disable	MMED	Boolean	BOOL	False	False

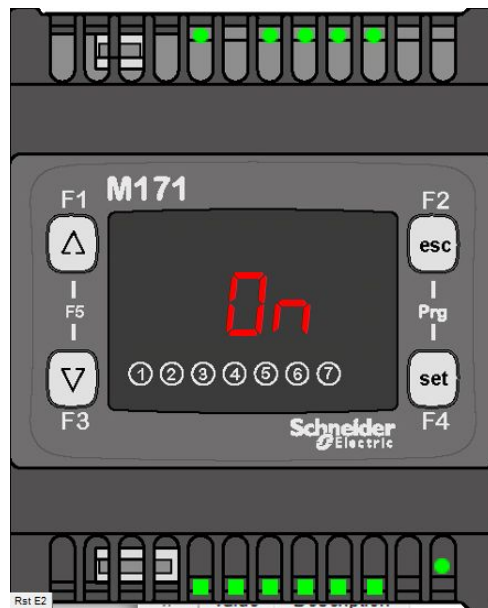
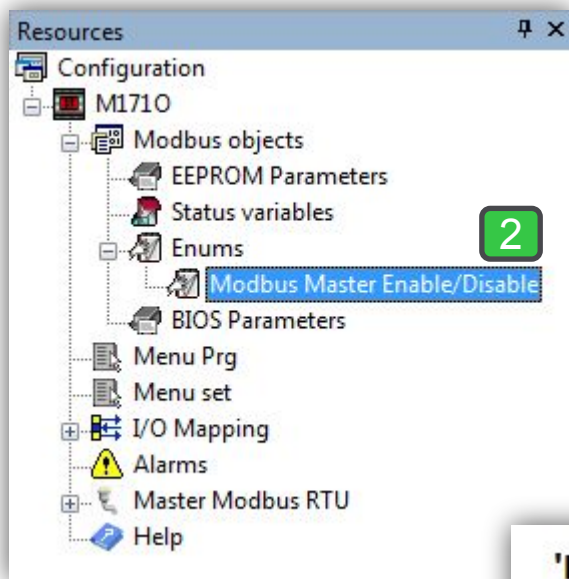
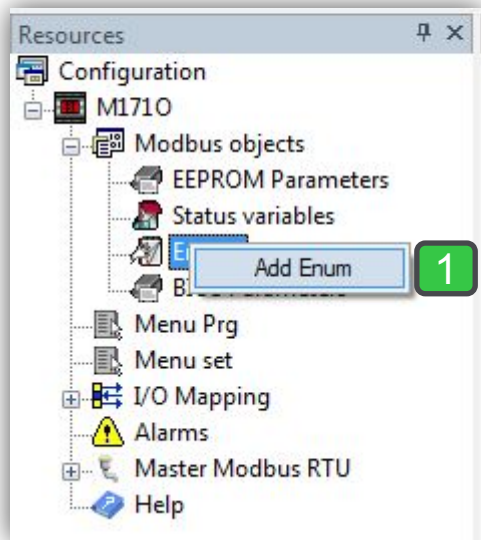


Note:

# Enable/Disable Modbus Master by Enums



Master side



## 'Modbus Master Enable/Disable' Enumerator



Add



Remove

#	Value	Description
1	0	Disable
2	1	Enable

3

Note:



# Set the Enumerator as Device type

Master side

**Status Variables**

Add Remove Recalc

#	Address	Name	Display label	1	Device type	Application type	Default value	Read only
1	8960	Enable_Disable	MMED	Boolean	BOOL	False	False	

Signed 16-bit  
Signed 32-bit  
Unsigned 16-bit  
Unsigned 32-bit  
Real  
Boolean  
Signed 8-bit  
Unsigned 8-bit  
Modbus Master Enable/Disable

**Status Variables**

Add Remove Recalc

#	Address	Name	Display label	2	Device type	Application type	Default value	Read only
1	8960	Enable_Disable	MMED	Modbus Master Enable/Disable	BOOL	Disable	False	

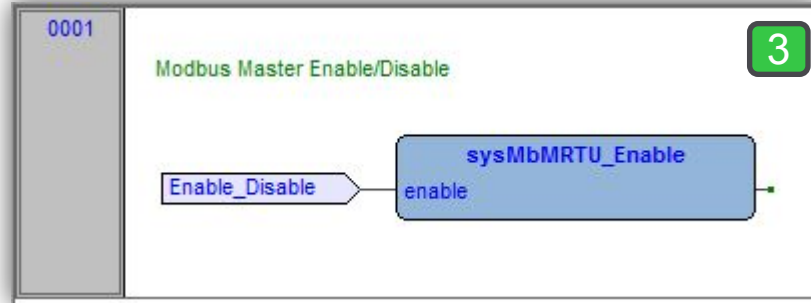
Note:



# Enable/Disable Modbus Master Program

Master side

The Project Explorer window displays the project structure for 'M1710\_Master Project'. The 'Tasks' folder is highlighted with a green box and labeled '2'. Inside 'Tasks', the 'Modbus\_Master\_Enable\_Disable' task is selected and highlighted with a red box. The 'Variables' folder under 'Global shared' is also highlighted with a red box and labeled '1', containing the 'Enable\_Disable' variable.



The Library window displays a list of system blocks. The 'sysMbMRTU\_Enable' block is highlighted with a blue box. The 'Target blocks' tab is selected and highlighted with a green box and labeled '4'.

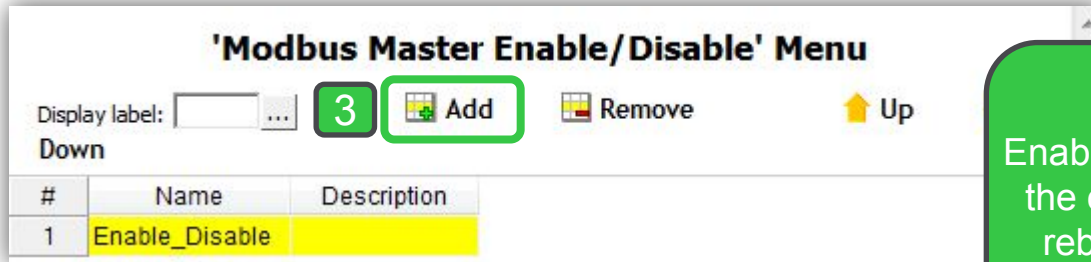
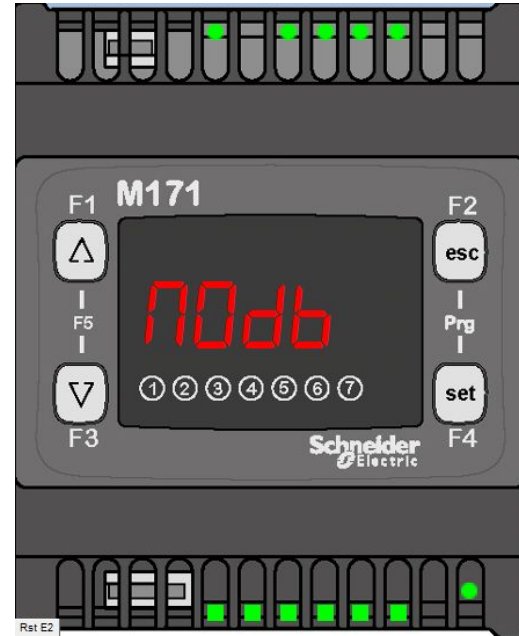
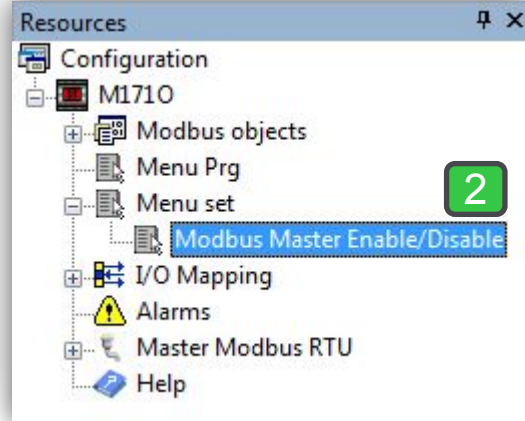
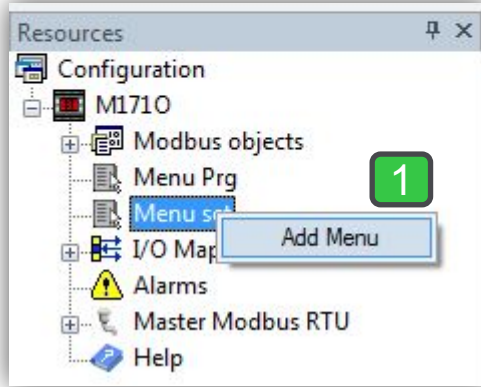
Block Name	Block Name	Block Name
sysClockWrite	sysMbMRTU_BroadcastFC16	sysf
sysExecutionPassword	sysMbMRTU_Enable	sysf
sysMbMRTU_BroadcastFC05	sysMbMRTU_FC01	sysf
sysMbMRTU_BroadcastFC06	sysMbMRTU_FC02	sysf
sysMbMRTU_BroadcastFC15	sysMbMRTU_FC03	sysf

Note:



# Set Menu Creation

Master side, to access Enable/Disable from dashboard



Note: If you do create a menu to access Enable/Disable from front face of product you have the option of Modbus Master deactivation during rebooting the controller (in 5 sec) to be able to connect to controller via SoMachineHVAC.

# Status Variable Definition

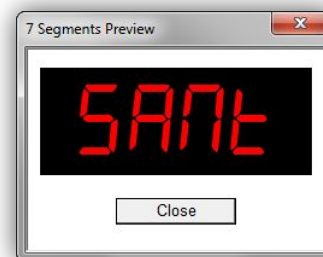
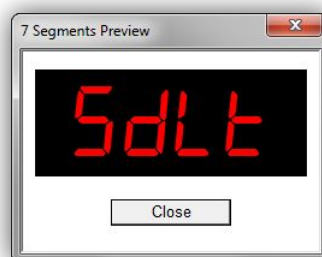


## Master side

Resources

- Configuration
  - M1710
    - Modbus objects
      - EEPROM Parameters
      - Status variables **1**
      - Enums
      - Modbus Master Enable/Disable
      - BIOS Parameters
    - Menu Prg
    - Menu set
    - Modbus Master Enable/Disable
    - I/O Mapping
    - Alarms
    - Master Modbus RTU
    - Help

Note: As far as Set point & Delta are defined as EEPROM parameters in slave side no need to keep their values again in master side and just reading/writing them via status variable is enough.



**3**

### Status Variables

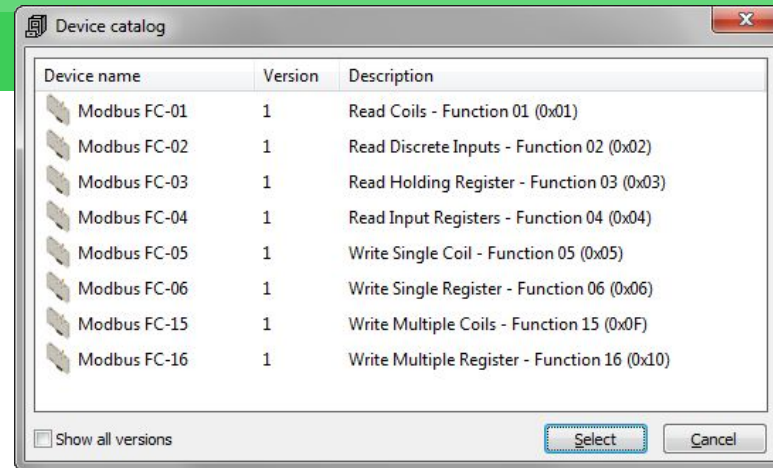
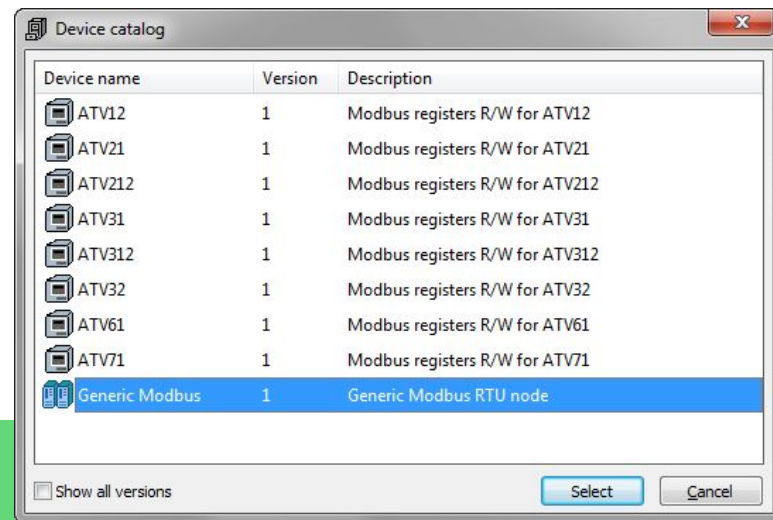
Recalc

**2**

#	Address	Name	Display label	Device type	Application type	Default value	Read only
1	8960	Enable_Disable	MMED	Modbus Master Enable/Disable	BOOL	Disable	False
2	8961	Slave_Setpoint	SSEP	Signed 16-bit	INT		False
3	8962	Slave_Delta	SDLT	Signed 16-bit	INT		False
4	8963	Slave_Ambiant_Temp	SAMT	Signed 16-bit	INT		False

# 1<sup>st</sup> Method: Generic Modbus RTU

By variable assignment







# Modbus Master, RS485 Configuration

Master side

Resources

- Configuration
  - M1710
    - Modbus objects
    - Menu Prg
    - Menu set
    - I/O Mapping **1**
    - Alarms
    - Master Modbus RTU
    - Help

## RS485 Configuration **2**

**Mode**

Not used

Modbus Master

**Baud rate**

1200 b/s

2400 b/s

4800 b/s

9600 b/s

19200 b/s

38400 b/s

57600 b/s

115200 b/s

**Serial Mode**

E,8,1 (Even parity, 8 data bits, 1 stop bit) ▾

Catalog

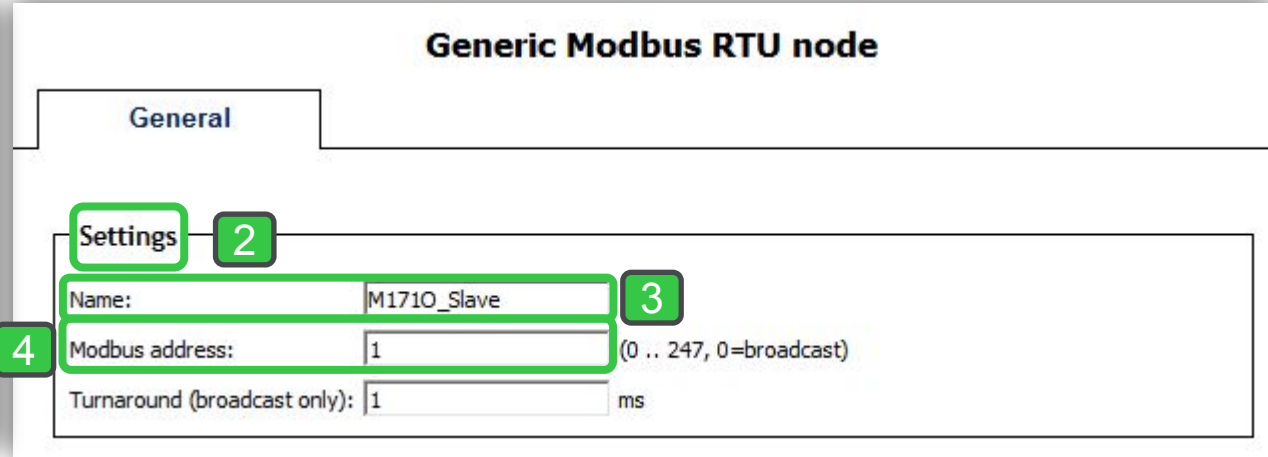
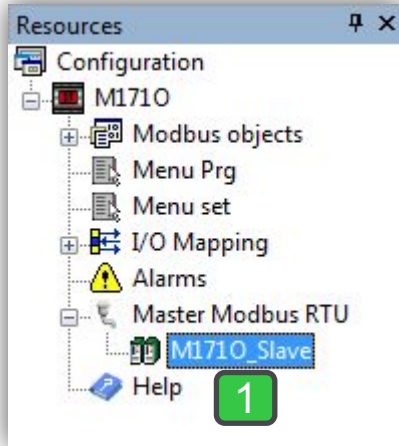
Device name	Version	Description
ATV12	1	Modbus registers R/W for ATV12
ATV21	1	Modbus registers R/W for ATV21
ATV212	1	Modbus registers R/W for ATV212
ATV31	1	Modbus registers R/W for ATV31
ATV312	1	Modbus registers R/W for ATV312
ATV32	1	Modbus registers R/W for ATV32
ATV61	1	Modbus registers R/W for ATV61
ATV71	1	Modbus registers R/W for ATV71
Generic Modbus	1 <b>3</b>	Generic Modbus RTU node

Note:



# Generic Modbus RTU

## Master side

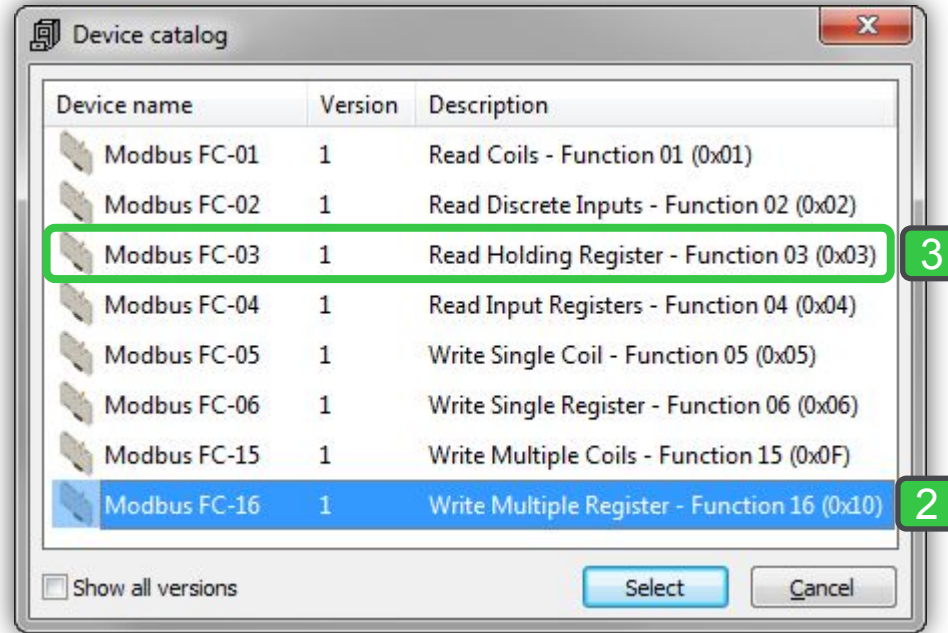
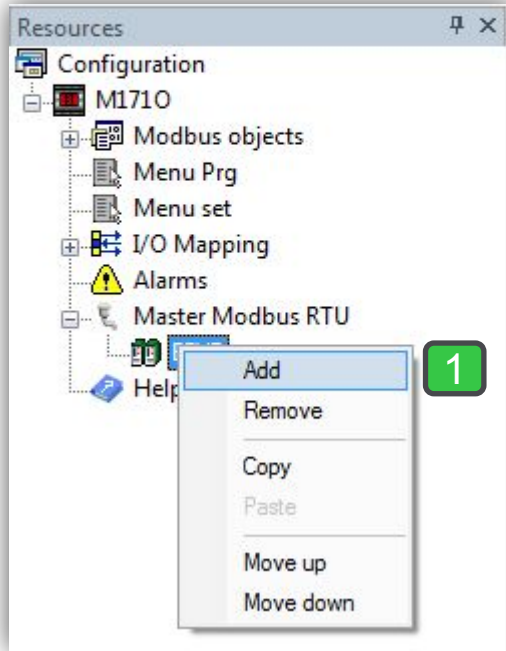


- 3. Rename it to a meaning full name
- 4. Se the slave's Modbus address



# Read and Write Function Codes

Master side, Read & Write



2 & 3. Drag and drop the desired function code/s into the Modbus Master RTU link



# Writing Multiple Register General Settings

Master side, Address matching

Resources

- Configuration
  - M1710
    - Modbus objects
      - Menu Prg
      - Menu set
      - I/O Mapping
      - Alarms
      - Master Modbus RTU
        - M1710\_Slave
          - Modbus write FC-16
    - Help

1

## Modbus FC 03(0x03) - Read Holding Register

General Holding Reg. Master Side

Settings

3

Start address: 16384 (1 .. 65536)

Polling time: 0 ms (0 = Continuous Read)

Time out: 1000 ms

Wait before send: 0 ms

## EEPROM Parameters Slave Side

Add Remove Recalc

3

	Address	Name	Display label	Device type	Application type	Default value	Min	Max	Unit	Format	AccessLevel
1	16384	SetPoint	SetP	Signed 16-bit	INT	180	120	300	°C	XXX.Y	Always visible
2	16385	Delta	Delta	Signed 16-bit	INT	25	10	50	°C	XXX.Y	Always visible



# Variable Assignment/Witre Function Code

Master side, Assignment

Resources

- Configuration
  - M1710
    - Modbus objects
    - Menu Prg
    - Menu set
    - I/O Mapping
    - Alarms
    - Master Modbus RTU
      - M1710\_Slave
        - Modbus write FC-16**
    - Help

1

### Modbus FC 03(0x03) - Read Holding Register

General

Holding Reg. 2

Add Remove Assign UnAssign

#	Name	ObjType	Label	Type	Address	Description
1	Register	WORD	Slave_Setpoint	INT	16384	4
2	Register	WORD	Slave_Delta	INT	16385	

### Choose PLC variable

Filter:

- Enable Disable (BOOL)
- Slave\_Setpoint (INT)** 3
- Slave\_Delta (INT)
- Slave\_Ambiant\_Temp (INT)

Note:



# Read Holding Register General Settings

Master side, Address matching

Resources

- Configuration
  - M1710
    - Modbus objects
    - Menu Prg
    - Menu set
    - I/O Mapping
    - Alarms
    - Master Modbus RTU
      - M1710\_Slave
        - Modbus write FC-16
        - Modbus Read FC-03** (1)
  - Help

## Modbus FC 03(0x03) - Read Holding Register

General | **Holding Reg.** | Master Side

Settings

Start address:  (1 .. 65536) (2)

Polling time:  ms (0 = Continuous Read)

Time out:  ms

Wait before send:  ms

## Status Variables

Slave Side

Add Remove Recalc

#	Address	Name	Display label	Device type	Application type	Read only
1	8960	Ambiant_Temp	ATMP	Signed 16-bit	INT	True

Note:

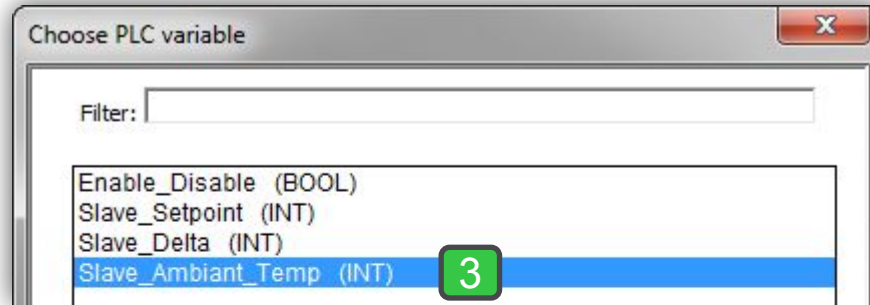
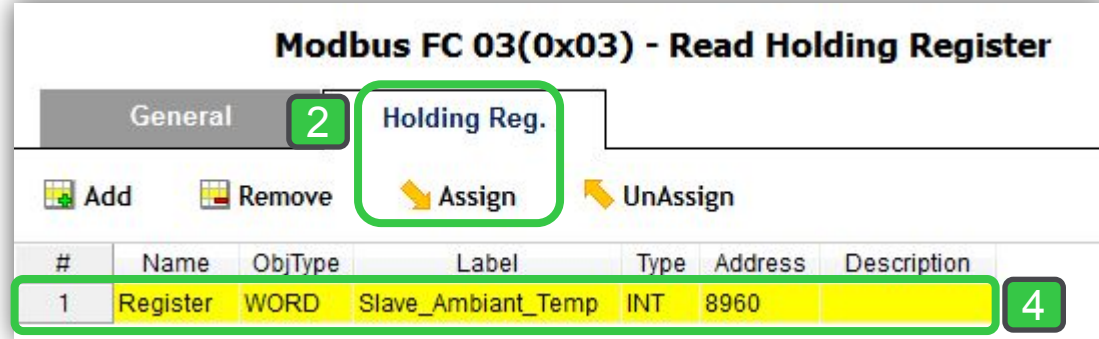
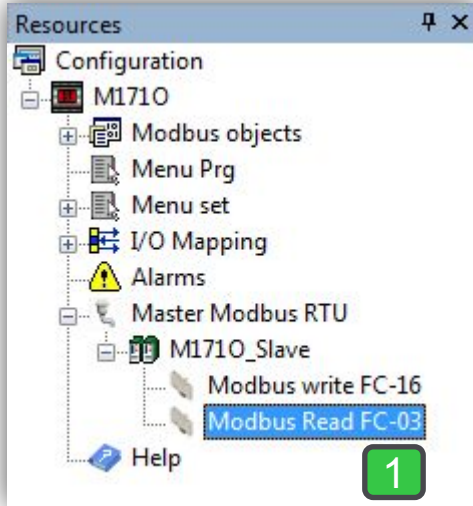
Life Is On





# Variable Assignment/Read Function Code

Master side, Assignment



Note:

Life Is On





# Read Holding Register General Settings

Master side, address matching & assignment

Resources

- Configuration
- M1710
  - Modbus objects
  - Menu Prg
  - Menu set
  - I/O Mapping
  - Alarms
  - Master Modbus RTU
  - M1710\_Slave
    - Modbus write FC-16
    - Modbus Read FC-03
    - Modbus Read FC-03 EEPROM Par.**
- Help

1

2

## Modbus FC 03(0x03) - Read Holding Register

General Holding Reg.

Settings

Start address:  (1 .. 65536)

Polling time:  ms (0 = Continuous Read)

Time out:  ms

Wait before send:  ms

## Modbus FC 03(0x03) - Read Holding Register

General Holding Reg.

Add Remove Assign UnAssign

#	Name	ObjType	Label	Ty...	Address	Description
1	Register	WORD	Slave_Setpoint	INT	16384	
2	Register	WORD	Slave_Delta	INT	16385	

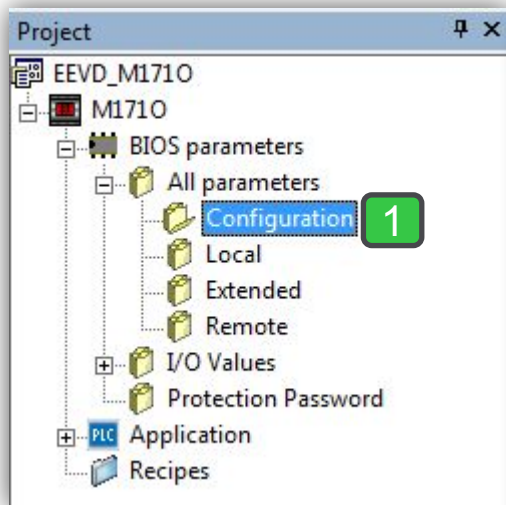
4

Note:



# Modbus Protocol Properties

Master side, via Device tool



## 2 Configuration

Address	Name	Value	Um	Default	Min	Max	Description
53276	CF32	1=Even	num	1=Even	1	3	Modbus parity protocol
53274	CF30	1	num	1	1	255	Modbus protocol controller address
15636	Par_POLI	0	num	0	0	65535	Polycarbonate code
15715	Ui26	350	4ms	350	0	999	Key hold time to enable function
53456	CF50	1=Present	num	1=Present	0	1	RTC present
15640	CF61	0	num	0	0	999	Customer code 2
15639	CF60	0	num	0	0	999	Customer code 1
15744	Ui27	1	num	1	0	255	Installation engineer password
53275	CF31	3=9600	num	3=9600	0	7	Modbus baud rate protocol
15745	Ui28	2	num	2	0	255	Manufacturer password
53273	CF21	0	num	0	0	14	Protocol controller family
53272	CF20	0	num	0	0	14	Protocol controller address
53265	CF01	1	num	1	0	1	Select COM1 protocol

Note:

Life Is On

Schneider  
Electric

# Target and SoMachineHVAC



Parameters needed for correct connection between the **M1710 targets** and SoMachineHVAC

5. Set them as default value

parameter	description	values	default	visibility	notes
1 <b>CF30</b>	Modbus protocol controller address	1...255	1	3	Check that the set values correspond to those defined by the panel <b>Communication &gt; Settings &gt; Properties</b>
<b>CF31**</b>	Modbus protocol baud rate	0,1, 2 = not used 3 = 9600 baud 4 = 19200 baud 5 = 38400 baud 6 = 57600 baud 7 = 115200 baud	3	3	
3 <b>CF32</b>	Modbus protocol controller parity	1 = EVEN 2 = NONE 3 = ODD	1	3	

\*COM1 = TTL / RS485 (/S models only): cannot be used simultaneously

\*\*CF31  
 5=38400 baud (RS485: not supported)  
 6=57600 baud (RS485: non supported)  
 7=115200 baud (RS485: non supported)

4. Maximum supported baud rate is 19200

# Customize M171O Baud Rate

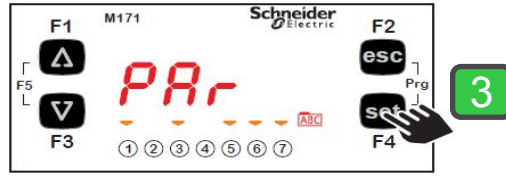


M171O parameters in the CF folder manages the connection between the target and Studio

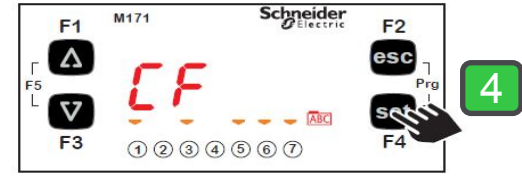
If the target is “empty”, i.e. there is no IEC application on the device, M171O will display the message FrEE, otherwise fundamental state is displayed (**Press F5 to switch to FrEE menu**)



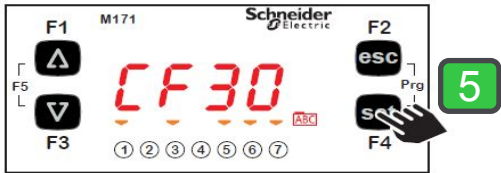
To view the parameter menu, press the Esc and Set keys at the same time. This will open the PAR menu.



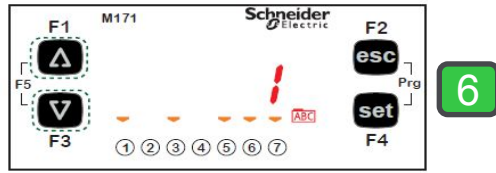
The parameters menu PAR contains all controller folders. Press the set key to view folders.



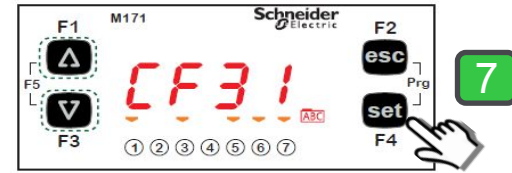
The first folder shown is the CF configuration folder. Press the set key to view the folder parameters.



The first parameter shown is CF30. To view the value of the parameter press the set key.



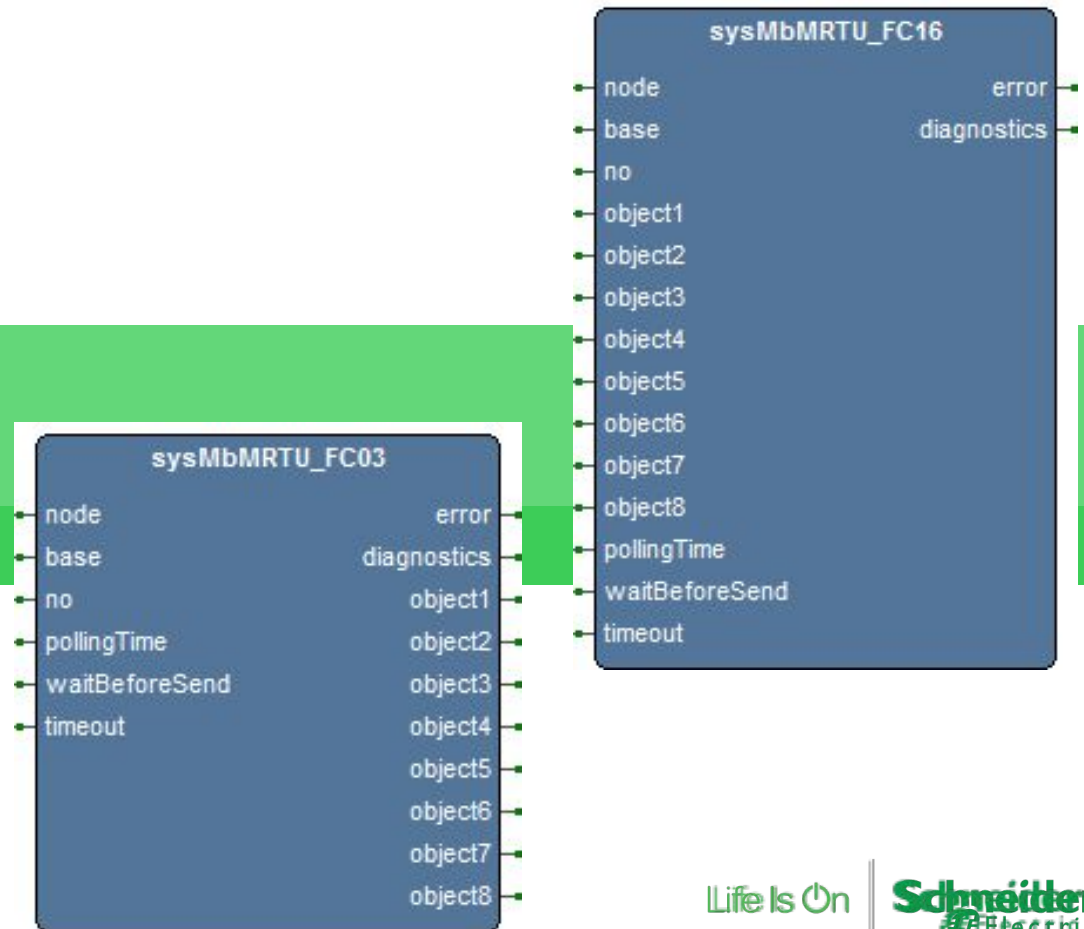
Use the UP and DOWN keys to change the value if necessary. To confirm the value press the set key. To exit press Esc



Use the UP and DOWN keys to scroll the other parameters and repeat the procedure to view the values and - if necessary - edit them.

## 2<sup>nd</sup> Method: Target Blocks

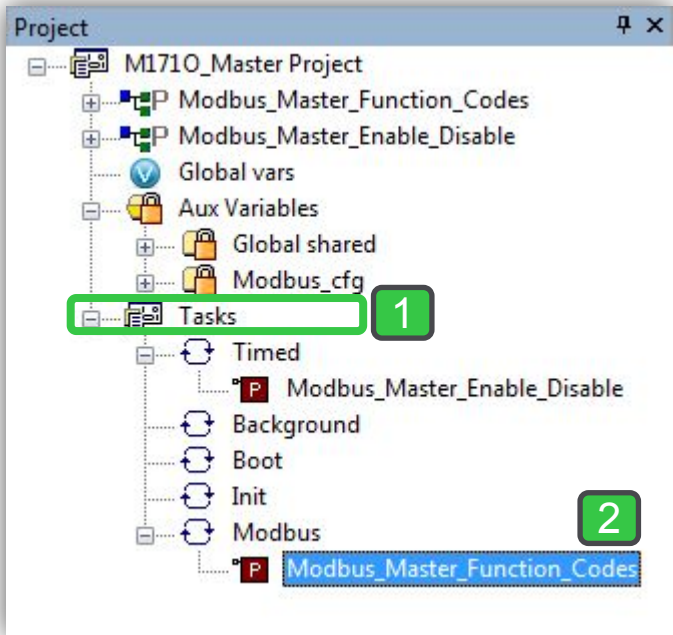
By Function Block Configuration





# Modbus Task

## Master side



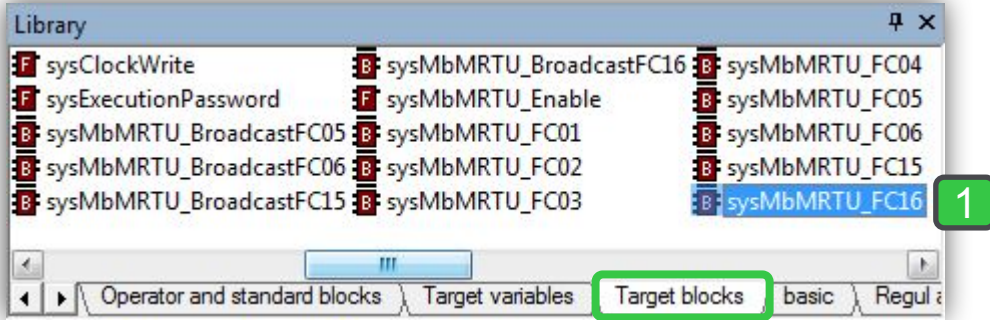
### Note:

Dedicated PLC task (Modbus task): operations on the bus do not affect the PLC Timed task and have limited impact on lower-priority tasks, since they are carried on asynchronously.



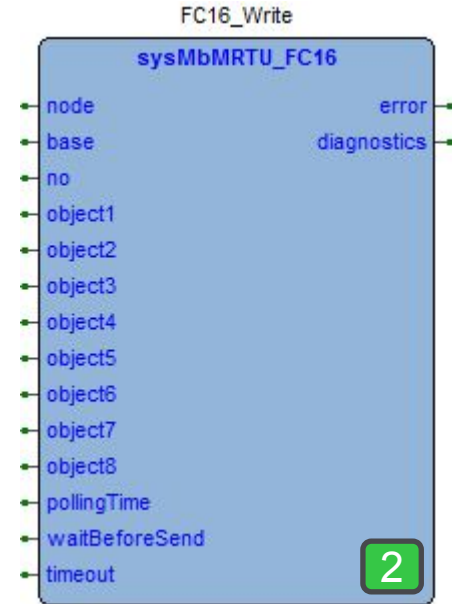
# sysMbMRTU write function code, FC16

Master side, library & Output Pin-Out



## Output:

Name	Type	Description
error	BOOL	TRUE=error occurred,FALSE=no error
diagnostics	MbMRTUDiagnostics	Message diagnostics



Note: This function block can be used in the Modbus task only.

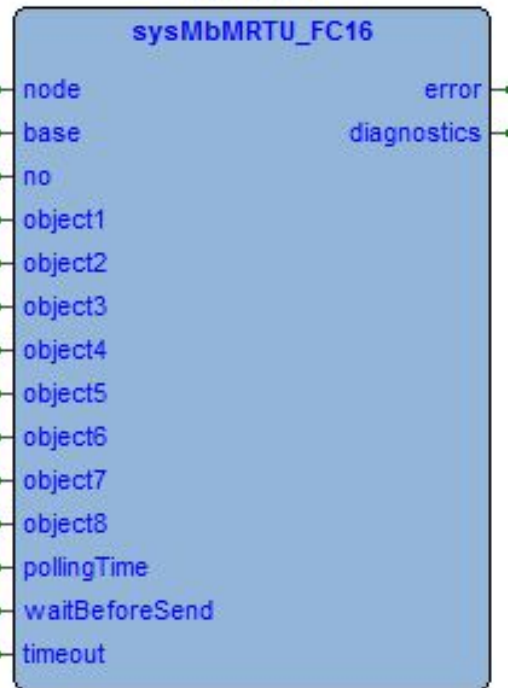
# sysMbMRTU write function code, FC16



Master side, Input Pin-Out

Input:

Name	Type	Description
node	@MbMRTUNode	Slave node
base	UINT	Address of the first Holding Register to write
no	UINT	Number of contiguous Holding Registers to write
object1	INT	1st Holding Register value
object2	INT	2nd Holding Register value
object3	INT	3rd Holding Register value
object4	INT	4th Holding Register value
object5	INT	5th Holding Register value
object6	INT	6th Holding Register value
object7	INT	7th Holding Register value
object8	INT	8th Holding Register value
pollingTime	UINT	Polling time [ms]. 0=write on variation
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]



Note:



# Modbus Communication Error Detection

## Master Side, Status Variable declaration

Resources

- Configuration
  - M1710
    - Modbus objects
      - EEPROM Parameters
      - Status variables **1**
      - Enums
      - Modbus Master Enable/Disable
      - BIOS Parameters
    - Menu Prg
    - Menu set
    - I/O Mapping
    - Alarms
    - Master Modbus RTU
    - Help

### Status Variables

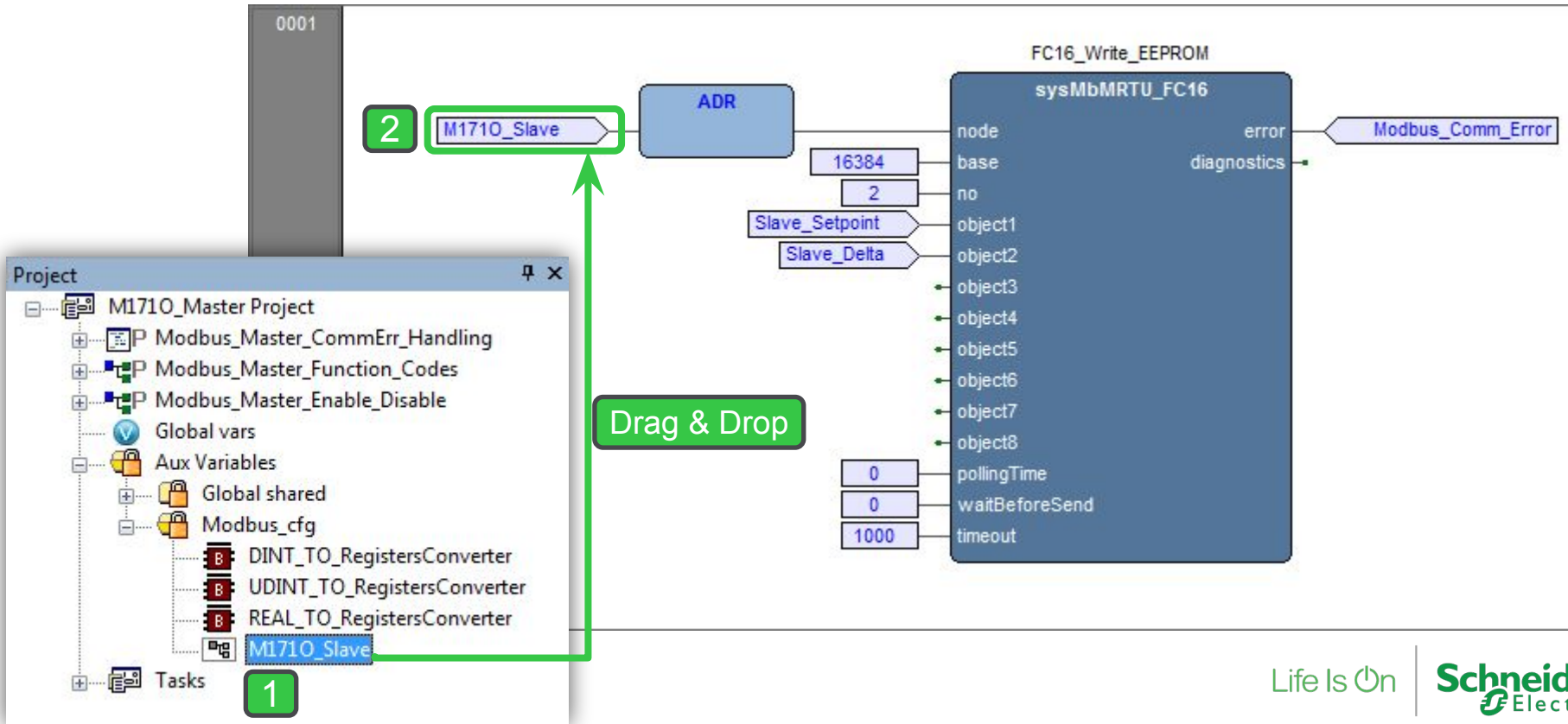
Remove Recalc

#	Address	Name	Display label	Device type	Application type	Format	Read only
1	8960	Enable_Disable	MMED	Modbus Master Enable/Disable	BOOL		False
2	8961	Slave_Setpoint	SSEP	Signed 16-bit	INT	XXX.Y	False
3	8962	Slave_Delta	SDLT	Signed 16-bit	INT	XXX.Y	False
4	8963	Slave_Ambiant_Temp	SAMT	Signed 16-bit	INT	XXX.Y	False
5	8964	Modbus_Comm_Error	MCER	Boolean	BOOL		False



# Write Multiple Registers, FC16

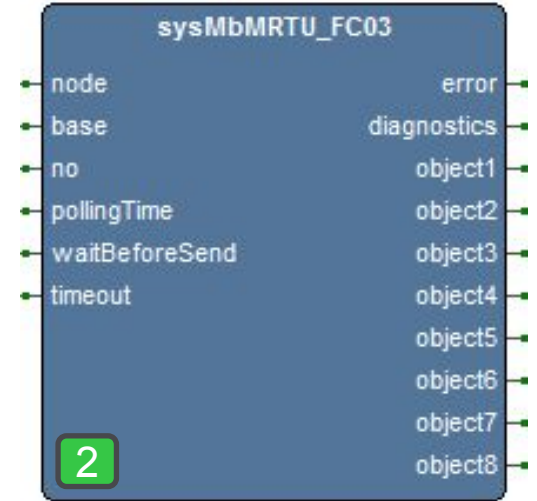
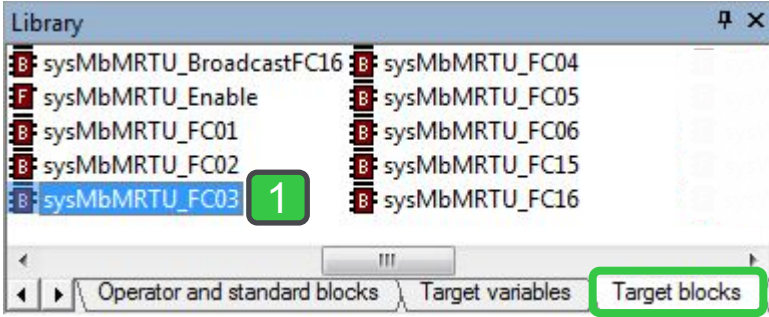
Master side, Assignment





# Read Holding Registers, FC03

Master side, Input Pin-Out



## Input:

Name	Type	Description
node	@MbMRTUNode	Slave node
base	UINT	Address of the first Holding Register to be read
no	UINT	Number of contiguous Holding Registers to be read
pollingTime	UINT	Polling time [ms]
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Note:

Life Is On

Schneider  
Electric

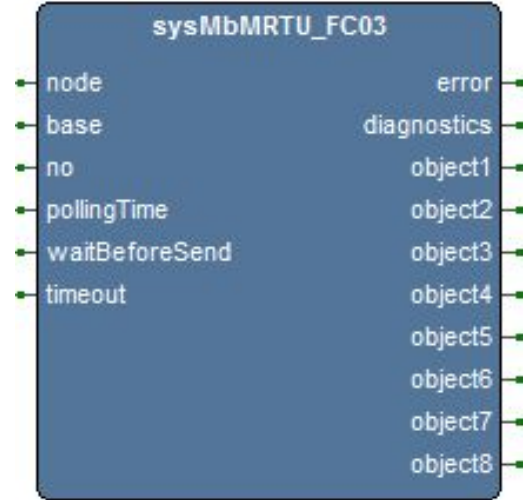


# Read Holding Registers, FC03

Master side, Output Pin-Out

## Output:

Name	Type	Description
error	BOOL	TRUE=error occurred,FALSE=no error
diagnostics	MbMRTUDiagnostics	Message diagnostics
object1	INT	1st Holding Register
object2	INT	2nd Holding Register
object3	INT	3rd Holding Register
object4	INT	4th Holding Register
object5	INT	5th Holding Register
object6	INT	6th Holding Register
object7	INT	7th Holding Register
object8	INT	8th Holding Register



Note:

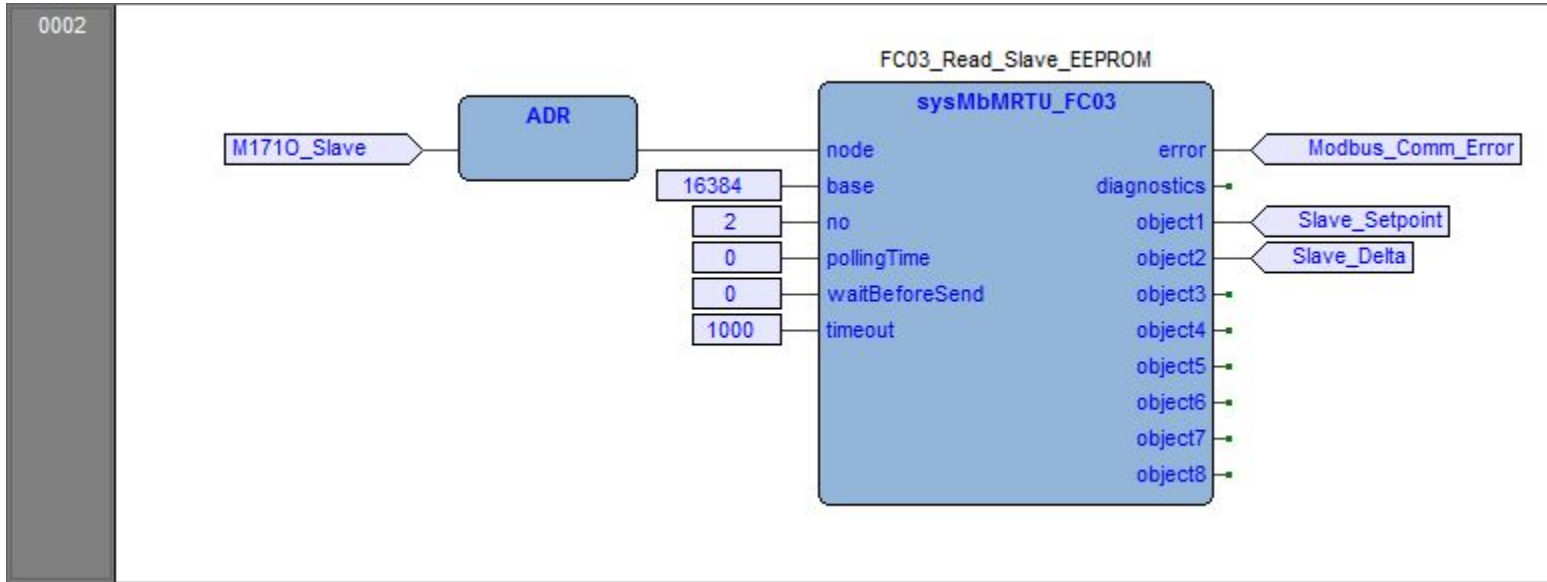
Life Is On





# Read Holding Registers, FC03

Master side, Assignment, Reading Slaves EEPROM Parameters



Note:

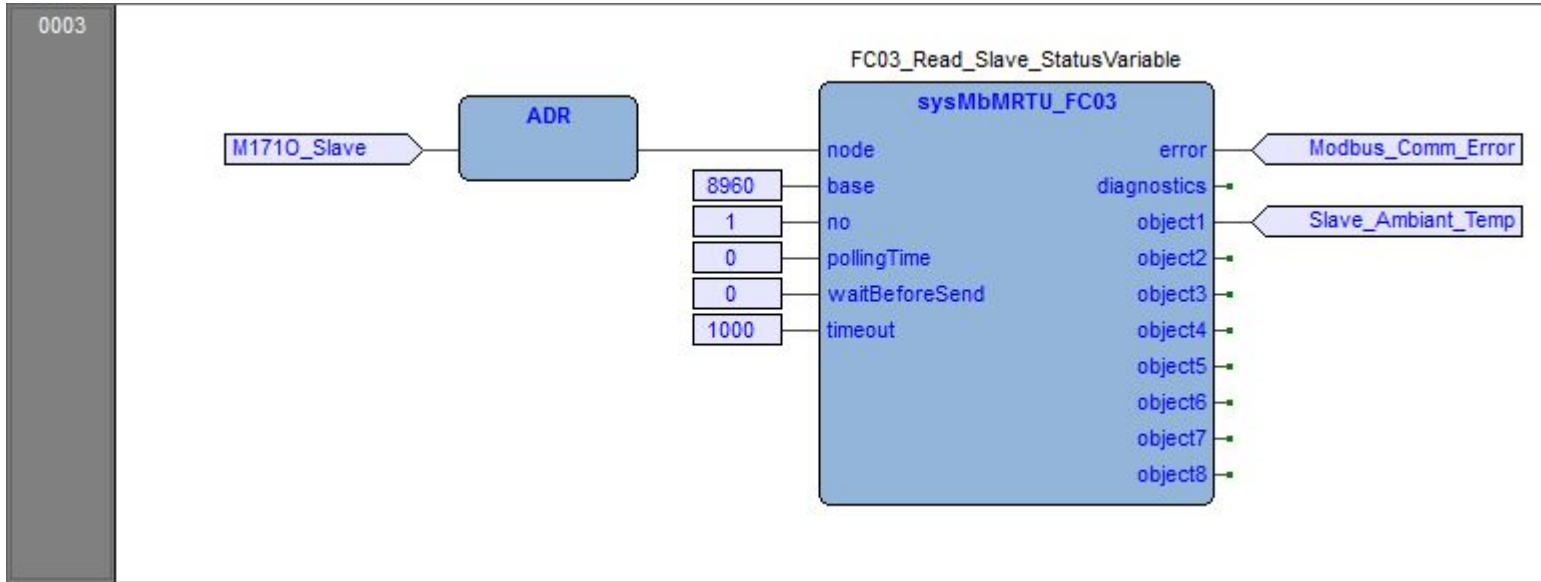
Life Is On





# Read Holding Registers, FC03

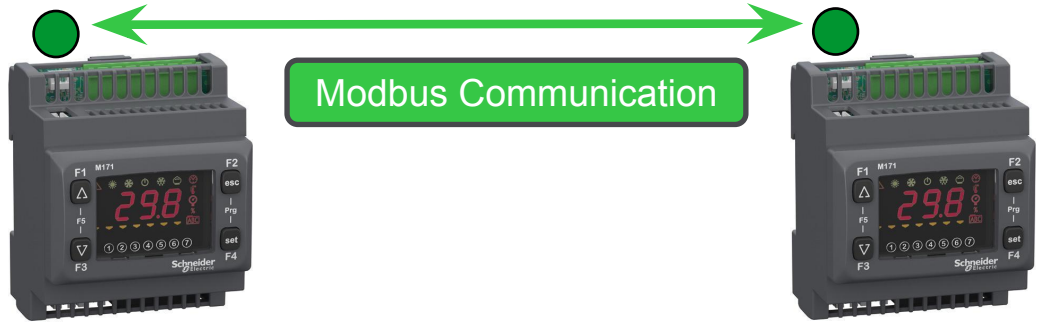
Master side, Assignment, reading slave's ambient Temperature



Note:

Life Is On





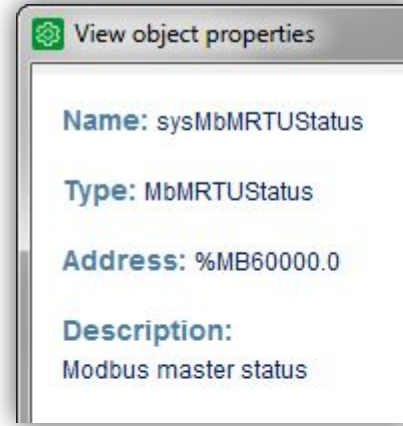
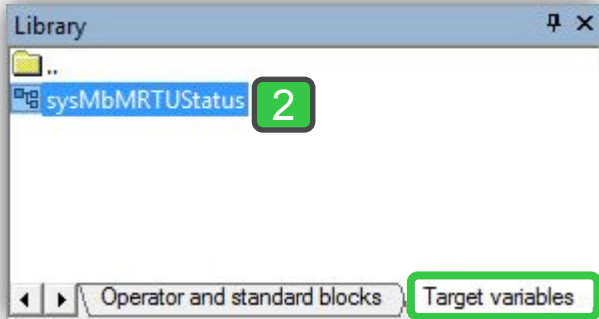
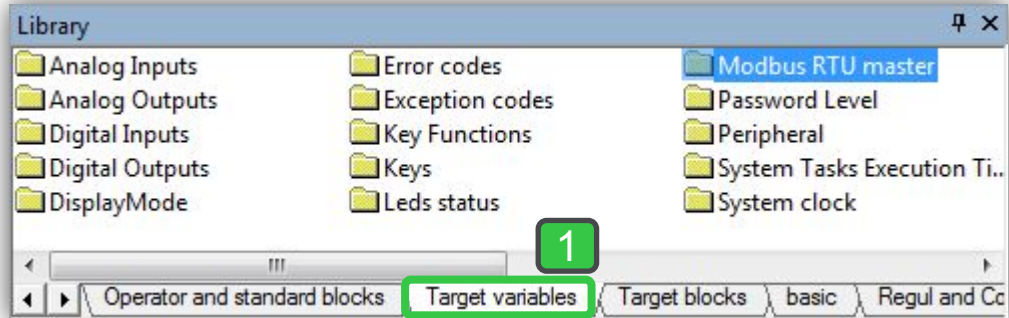
# Modbus Communication Monitoring & Debugging

By Target Variables



# Modbus Monitoring

Master Side, sysMbMRTUStatus

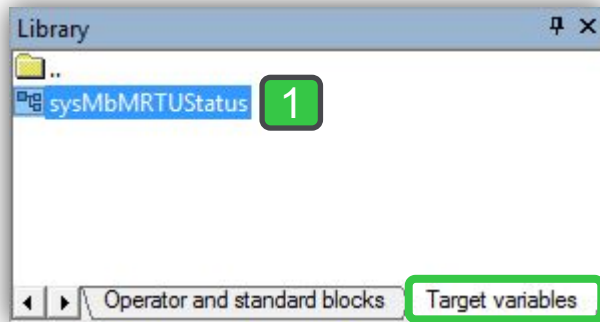


Note:

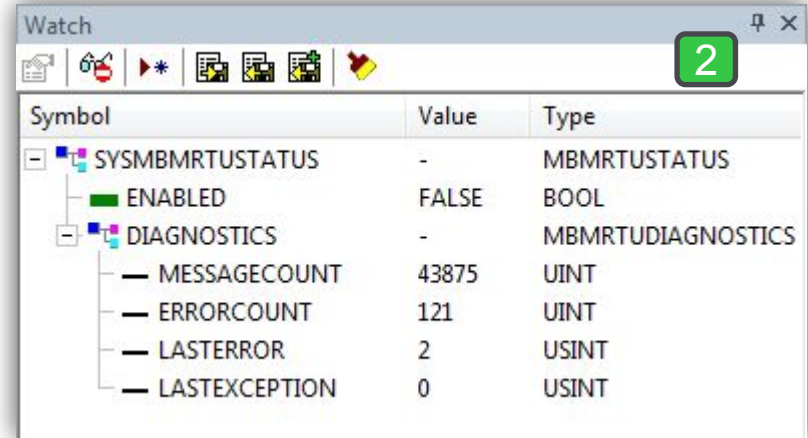


# Modbus Communication Monitoring

Master side, Watch Window



2. Drag & Drop



Note:

Life Is On





# Modbus Master Communication Error handling/detection



## Master Side, New Program

The image shows two screenshots from the Modbus Master software interface. The left screenshot shows the 'Project' window with the 'M171O\_Master Project' tree. A context menu is open over the project, and the 'New program' option is selected, highlighted with a green circle labeled '1'. The right screenshot shows the same project tree, but with the 'Modbus\_Master\_CommErr\_Handling' program added under the 'Timed' task, highlighted with a blue selection box. Below these screenshots is the 'New program' dialog box, which is also highlighted with a green circle labeled '2'. The dialog shows the 'Language' set to 'ST', the 'Name' field containing 'Modbus\_Master\_CommErr\_Handling', and the 'Task' set to 'Timed'.

Note:

Life Is On



# Monitoring Modbus Communication by third party



```
0001 (* Assigning structured target variable to the global
0002 status variable in order to monitor Modbus Master
0003 communication by third party *)
0004
0005 sysMbMRTUStatus.
0006
0007
0008
```

**2**

diagnostics MbMRTUDiagnostics Master global diagnostics  
enabled

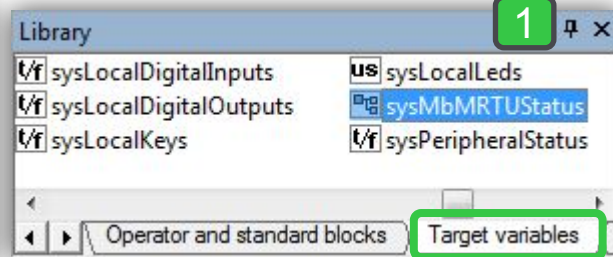
```
0001 (* Assigning structured target variable to the global
0002 status variable in order to monitor Modbus Master
0003 communication by third party *)
0004
0005 sysMbMRTUStatus.diagnostics.
0006
0007
0008
0009
0010
```

**3**

errorCount  
lastError  
lastException  
messageCount UINT Message count

```
0001 (* Assigning structured target variable to the global
0002 status variable in order to monitor Modbus Master
0003 communication by third party *)
0004
0005 ErrorCount:= sysMbMRTUStatus.diagnostics.errorCount;
0006 MessageCount:= sysMbMRTUStatus.diagnostics.messageCount;
0007 LastError:= sysMbMRTUStatus.diagnostics.lastError;
0008 LastException:= sysMbMRTUStatus.diagnostics.lastException;
```

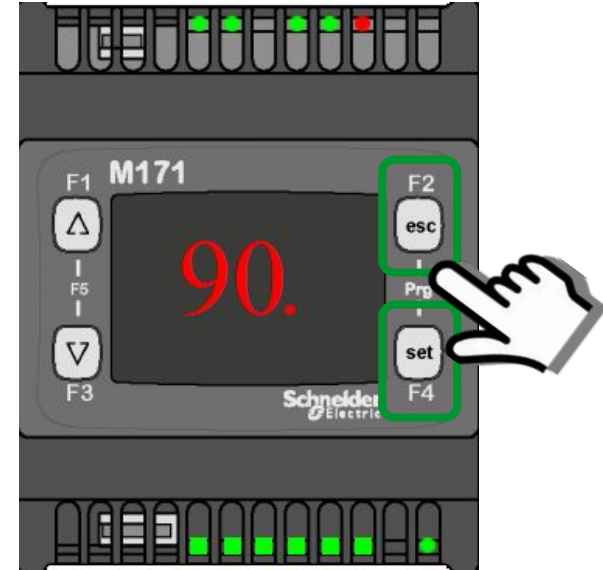
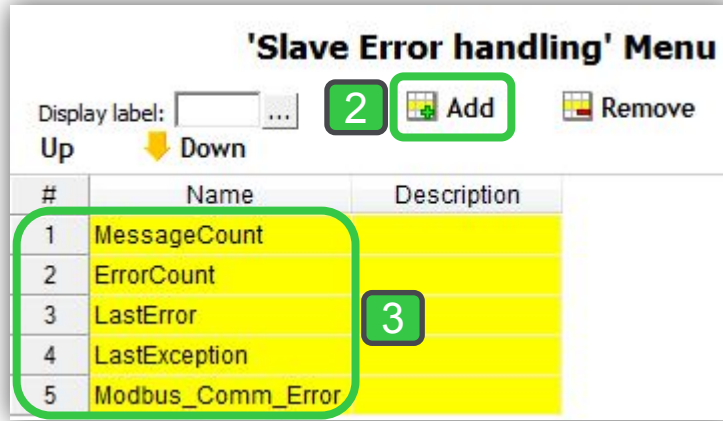
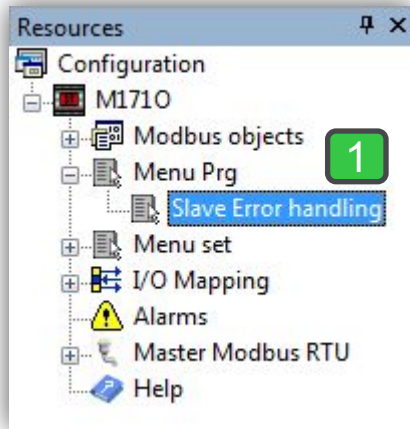
**4**



1. Drag & drop to from target variables list to the programming area
4. Assign desired global status variable to the diagnostic registers

# Menu Prg.

Master Side, to access them via dashboard



5	8964	Modbus_Comm_Error	MCER
6	8965	MessageCount	SMGC
7	8966	ErrorCount	SERC
8	8967	LastError	SLER
9	8968	LastException	SLEX

4. Short Form/label

4

Note:

Life Is On

Schneider  
Electric

# Appendix 1

Modbus Function codes Definition



# Modbus Master Function Code

## Read/write, **sysMbMRTU\_FC01**

Read Coils - Function 01 (0x01)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num: 5

node : @MbMRTUNode (\* Slave node \*)

base : UINT (\* Address of the first Coil to be read \*)

no : UINT (\* Number of contiguous Coils to be read \*)

pollingTime : UINT (\* Polling time [ms] \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 18

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

object1 : BOOL (\* 1st Coil \*)

...

object16 : BOOL (\* 16th Coil \*)

Extern vars num: 0



# Modbus Master Function Code

## Read/write, **sysMbMRTU\_FC02**

Read Discrete Inputs - Function 02 (0x02)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num: 5

node : @MbMRTUNode (\* Slave node \*)

base : UINT (\* Address of the first Discrete Input to be read \*)

no : UINT (\* Number of contiguous Discrete Inputs to be read \*)

pollingTime : UINT (\* Polling time [ms] \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 18

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

object1 : BOOL (\* 1st Discrete Input \*)

...

object16 : BOOL (\* 16th Discrete Input \*)

Extern vars num: 0

# Modbus Master Function Code

## Read/write, **sysMbMRTU\_FC03**



Read Holding Registers - Function 03 (0x03)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num: 5

node : @MbMRTUNode (\* Slave node \*)

base : UINT (\* Address of the first Holding Register to be read \*)

no : UINT (\* Number of contiguous Holding Registers to be read \*)

pollingTime : UINT (\* Polling time [ms] \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 10

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

object1 : INT (\* 1st Holding Register \*)

...

object8 : INT (\* 8th Holding Register \*)

Extern vars num: 0

# Modbus Master Function Code

## Read/write, **sysMbMRTU\_FC04**



Read Input Registers - Function 04 (0x04)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num : 5

node : @MbMRTUNode (\* Slave node \*)

base : UINT (\* Address of the first Input Register to be read \*)

no : UINT (\* Number of contiguous Input Registers to be read \*)

pollingTime : UINT (\* Polling time [ms] \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 10

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

object1 : INT (\* 1st Input Register \*)

...

object8 : INT (\* 8th Input Register \*)

Extern vars num: 0



# Modbus Master Function Code

## Read/write, **sysMbMRTU\_FC05**



Write Single Coil - Function 05 (0x05)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num : 5

node : @MbMRTUNode (\* Slave node \*)

address : UINT (\* Address of the Coil to write \*)

object : BOOL (\* Coil value \*)

pollingTime : UINT (\* Polling time [ms]. 0=write on variation \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 2

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

Extern vars num: 0

# Modbus Master Function Code

## Read/write, sysMbMRTU\_FC06



Write Single Register - Function 06 (0x06)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num: 5

node : @MbMRTUNode (\* Slave node \*)

address : UINT (\* Address of the Holding Register to write \*)

object : INT (\* Holding Register value \*)

pollingTime : UINT (\* Polling time [ms]. 0=write on variation \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 2

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

Extern vars num: 0

# Modbus Master Function Code

## Read/write, **sysMbMRTU\_FC15**



Write Multiple Coils - Function 15 (0x0F)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num : 21

node : @MbMRTUNode (\* Slave node \*)

base : UINT (\* Address of the first Coil to write \*)

no : UINT (\* Number of contiguous Coils to write \*)

object1 : BOOL (\* 1st Coil value \*)

...

object16 : BOOL (\* 16th Coil value \*)

pollingTime : UINT (\* Polling time [ms]. 0=write on variation \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 2

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

Extern vars num: 0

# Modbus Master Function Code

## Read/write, sysMbMRTU\_FC16



Write Multiple Registers - Function 16 (0x10)

Type : FUNCTION BLOCK

Code type : EMBEDDED

Input vars num: 13

node : @MbMRTUNode (\* Slave node \*)

base : UINT (\* Address of the first Holding Register to write \*)

no : UINT (\* Number of contiguous Holding Registers to write \*)

object1 : INT (\* 1st Holding Register value \*)

...

object8 : INT (\* 8th Holding Register value \*)

pollingTime : UINT (\* Polling time [ms]. 0=write on variation \*)

timeout : UINT (\* Timeout [ms] \*)

Inout vars num: 0

Output vars num: 2

error : BOOL (\* TRUE=error occurred,FALSE=no error \*)

diagnostics : MbMRTUDiagnostics (\* Message diagnostics \*)

Extern vars num: 0

# Questions?

Thank You