

08/10/2014

em4 ↔ MTPX/XX MODBUS COMMUNICATION

em4 TUTORIAL

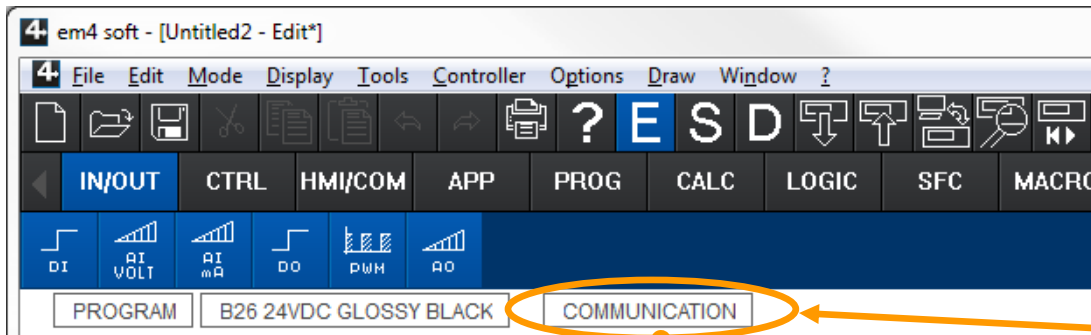


SUMMARY

- Configuring Modbus Communication
- em4 Modbus Addresses
- MTPX/XX Modbus RTU Wiring
- MTPX/XX Modbus RTU Network Configuration
- Modbus RTU Addressing
- Modbus RTU Addressing using BIN/DEC converter

CONFIGURING MODBUS COMMUNICATION

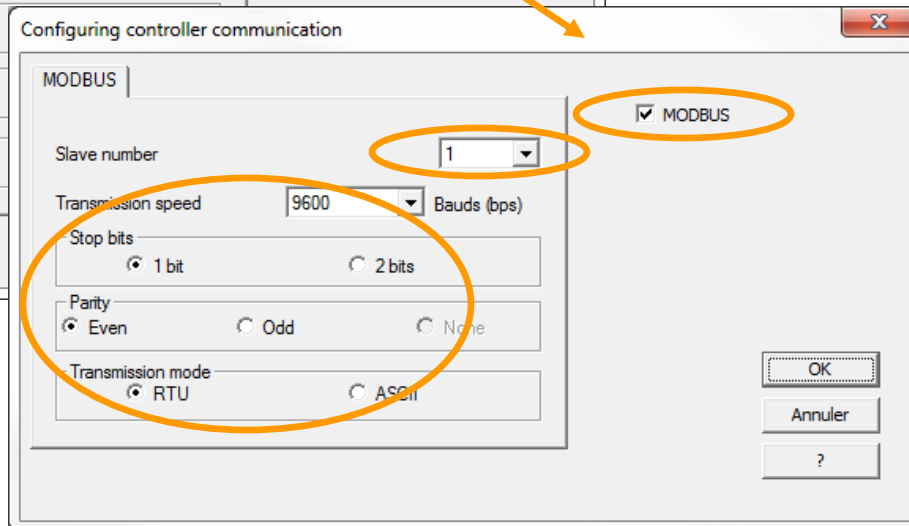
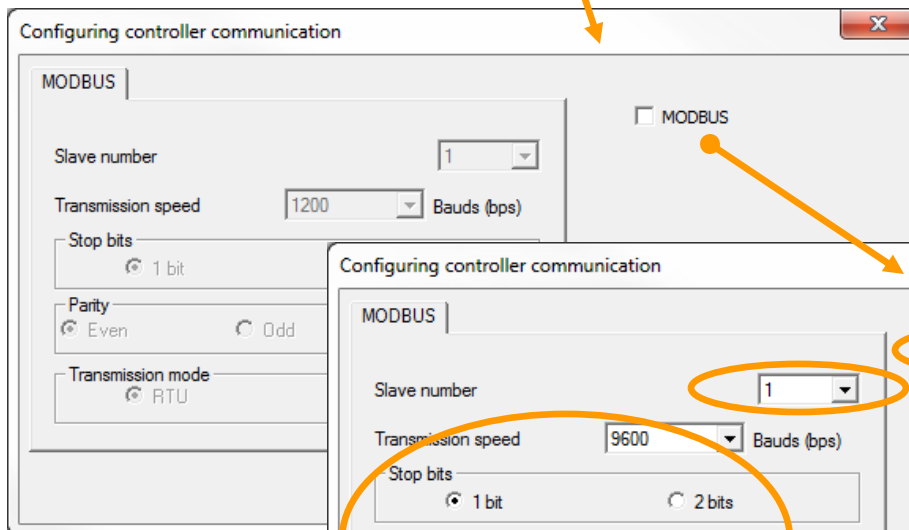
Configuring Modbus Communication



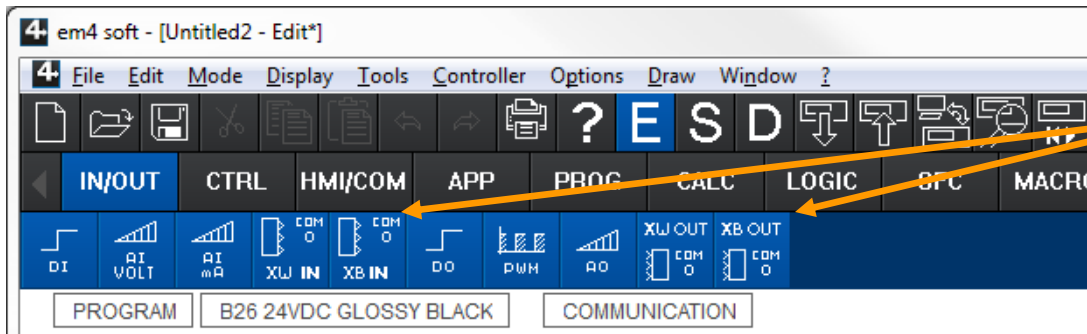
Open the software and select your controller.

In order to activate the Modbus communication and access the Modbus addresses click on *Communication*.

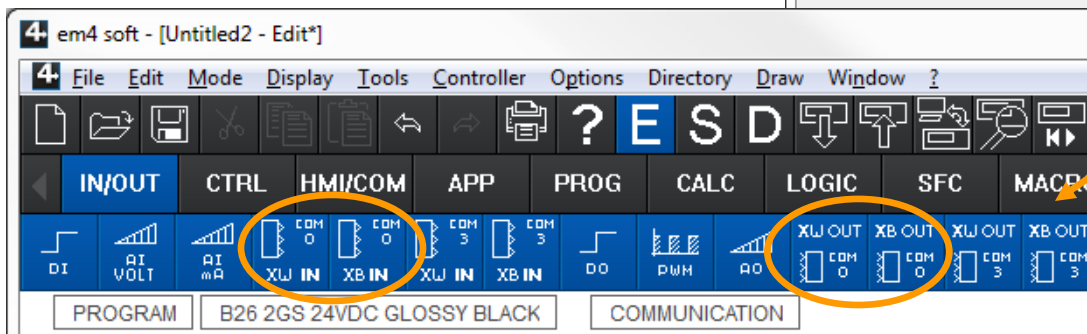
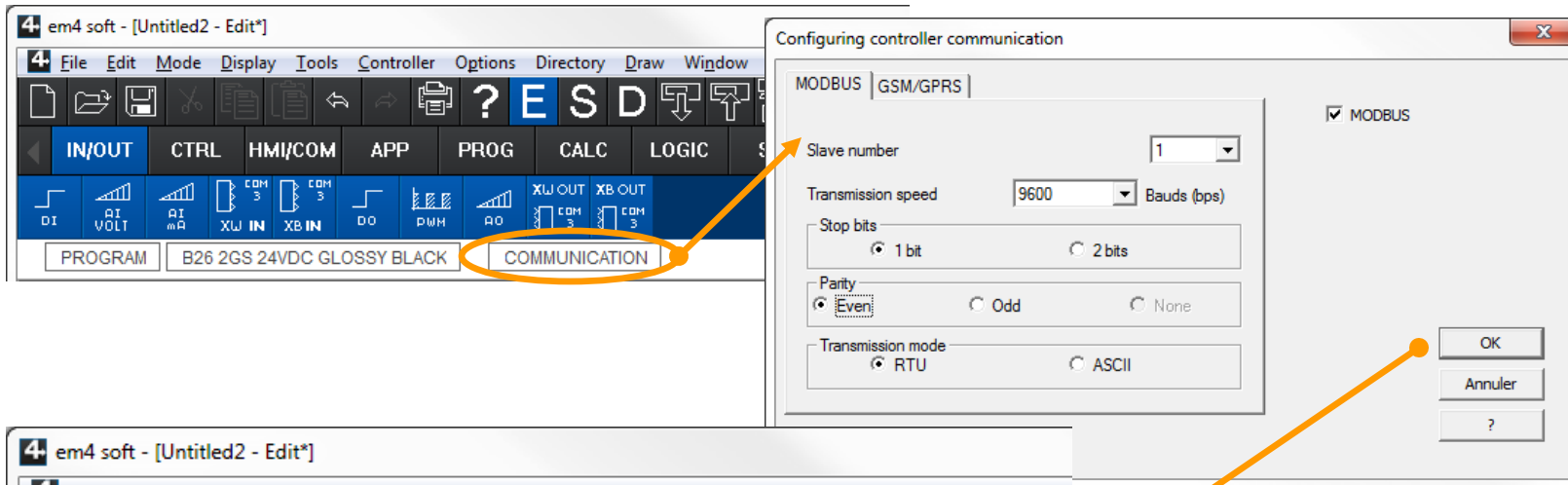
Check *Modbus*, define the *Slave number*, and and adjust the communication parameters to the values defined in the MTPX/XX (Modbus master).
Then click *OK*.



Configuring Modbus Communication



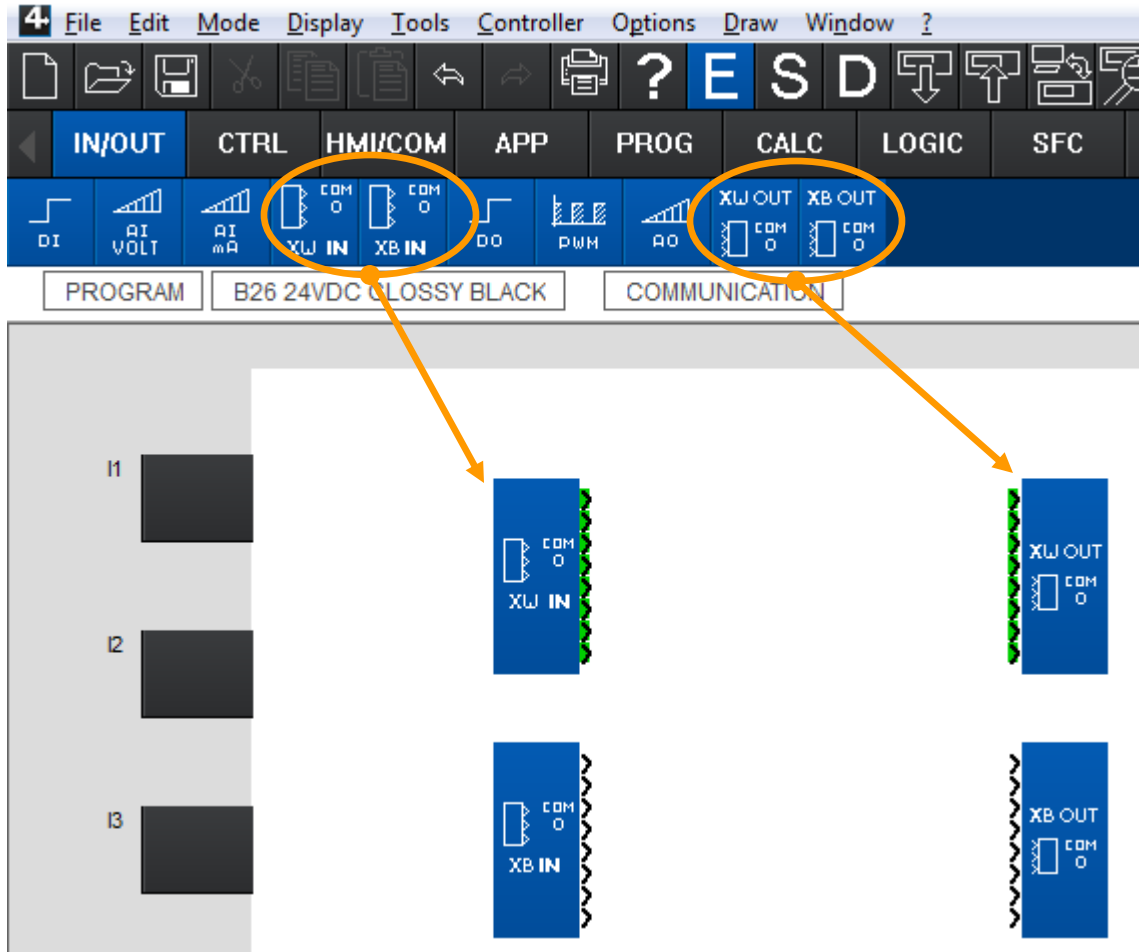
Immediately afterwards the *COM 0* functions become available.
Example em4 local



In an em4 alert or em4 remote the *COM 0* functions become accessible next to the *COM 3* functions.

em4 MODBUS ADDRESSES

em4 Modbus Addresses



Drag and drop the *COM 0* functions into your worksheet.

XW IN \Rightarrow Byte input from network, 8 inputs each, can be used 3 times, allows to enter 24 words into an em4 program.

XB IN \Rightarrow Bit input from network, 8 inputs each, can be used 2 times, allows to enter 16 bit into an em4 program.

XW OUT \Rightarrow Byte output to network, 8 outputs each, can be used 3 times, allows to make 24 words accessible to a network.

XB OUT \Rightarrow Bit output to network, 8 outputs each, can be used 2 times, allows to make 16 bit accessible to a network.

em4 Modbus Addresses

XW IN (Input byte Network)

Comments Parameters

Input address range

?

?
1-8
9-16
17-24

XB IN (Input bit Network)

Comments Parameters

Input address range

?

?
25-1
25-2

XW OUT (Output byte Network)

Comments Parameters

Output address range

?

?
26-33
34-41
42-49

XB OUT (Output bit Network)

Comments Parameters

Output address range

?

?
50-1
50-2

Double click on one of the *COM 0* functions opens the parameters window.

XW IN ⇒ Word address range for a FB: 1-8 or 9-16, or 17-24.

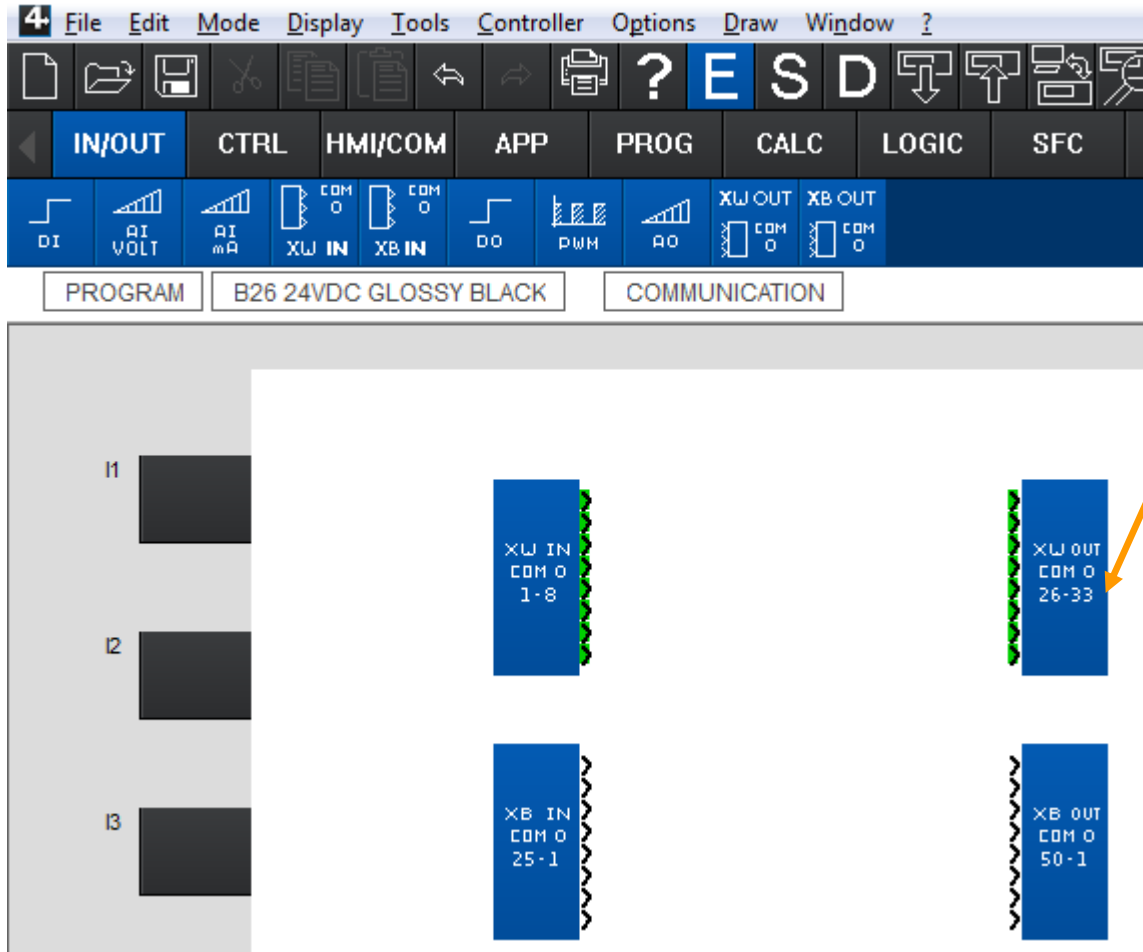
XB IN ⇒ Uses word address 25. Bit address range for a FB: 8 bit on 25-1, or 8 bit on 25-2.

XW OUT ⇒ Word address range for a FB: 26-33, or 34-41, or 42-49.

XB OUT ⇒ Uses word address 50. Bit address range for a FB: 8 bit on 50-1, or 8 bit on 50-2.

If no address range has been assigned to a *COM 0* FB placed in the worksheet, a error message will appear when the program is saved.

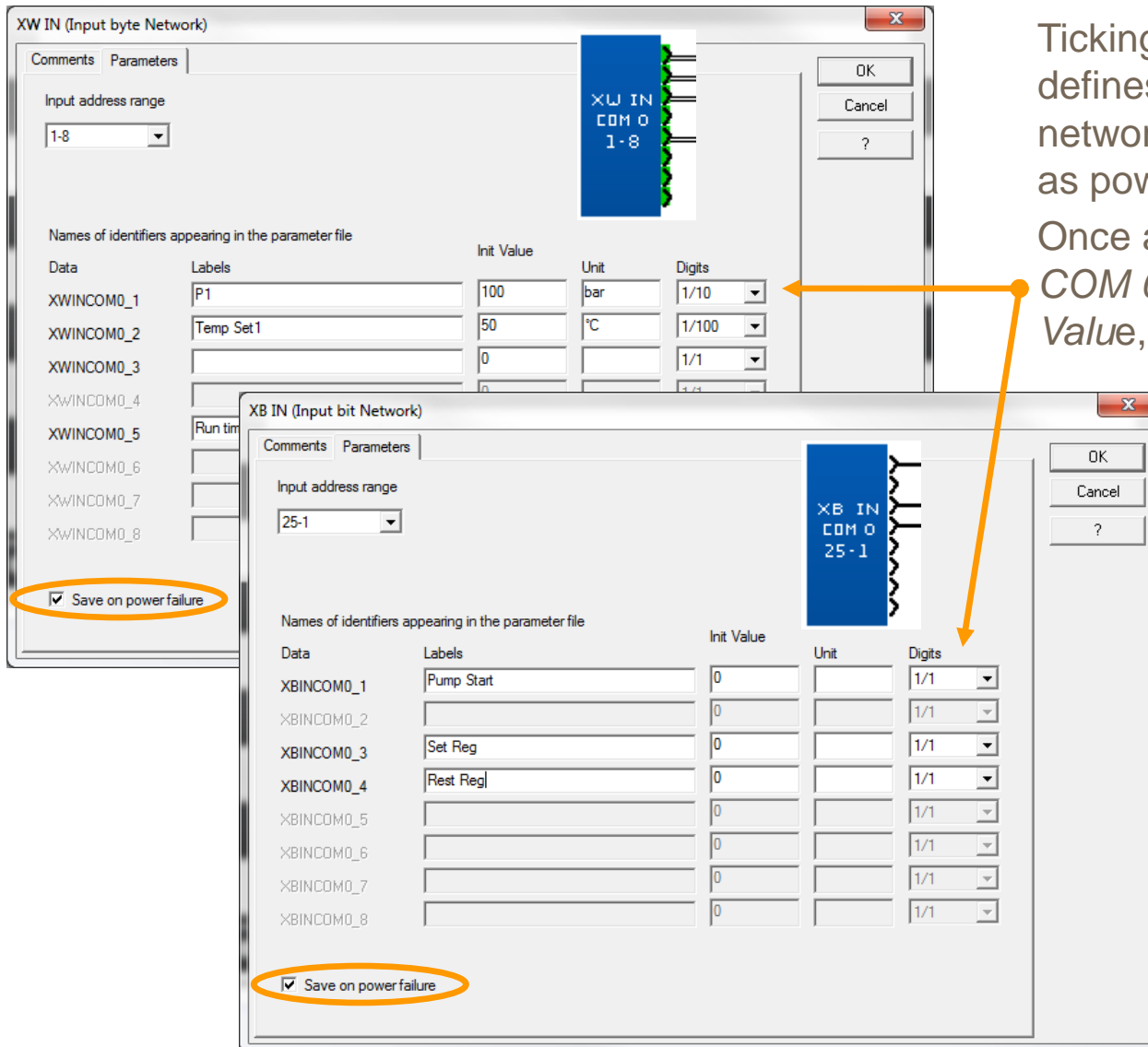
em4 Modbus Addresses



Once the address range of a FB has been defined, it is displayed on the block.

Each addresses corresponds to %MW and can directly be entered into the communication parameters used by a Modbus master.

em4 Modbus Addresses



XW IN (Input byte Network)

Input address range: 1-8

Names of identifiers appearing in the parameter file

Data	Labels	Init Value	Unit	Digits
XWINCOM0_1	P1	100	bar	1/10
XWINCOM0_2	Temp Set1	50	°C	1/100
XWINCOM0_3		0		1/1
XWINCOM0_4				
XWINCOM0_5				
XWINCOM0_6				
XWINCOM0_7				
XWINCOM0_8				

☒ Save on power failure

XB IN (Input bit Network)

Input address range: 25-1

Names of identifiers appearing in the parameter file

Data	Labels	Init Value	Unit	Digits
XBINCOM0_1	Pump Start	0		1/1
XBINCOM0_2		0		1/1
XBINCOM0_3	Set Reg	0		1/1
XBINCOM0_4	Rest Reg	0		1/1
XBINCOM0_5		0		1/1
XBINCOM0_6		0		1/1
XBINCOM0_7		0		1/1
XBINCOM0_8		0		1/1

☒ Save on power failure

Ticking *Save on power failure* defines all values arriving from the network at this XW IN or XB IN FB as power fail safe.

Once a link has been connected to a COM 0 FB output, a *Label*, an *Initial Value*, a *Unit*, and a decimal point setting (*Digits*) can be assigned to this address in the *Parameters* window.

Init Value ⇒ base value for first power up (cold init) and fallback value in case of network communication failure.

Labels, *Unit* and *Digits* are commentaries that are saved in an XML file in addition to the program.

em4 Modbus Addresses

XW OUT (Output byte Network)

Comments Parameters

Output address range
26-33

Names of identifiers appearing in the parameter file

Data	Labels	Init Value	Unit	Digits
XWOUTCOM0_1	Actual Pressure	0	bar	1/10
XWOUTCOM0_2	Actual Temperature	0	°C	1/100
XWOUTCOM0_3		0		1/1
XWOUTCOM0_4		0		1/1
XWOUTCOM0_5		0		1/1
XWOUTCOM0_6		0		1/1
XWOUTCOM0_7		0		1/1
XWOUTCOM0_8		0		1/1

OK
Cancel
?

XB OUT (Output bit Network)

Comments Parameters

Output address range
50-1

Names of identifiers appearing in the parameter file

Data	Labels	Init Value	Unit	Digits
XBOUTCOM0_1	Pump Status	0		1/1
XBOUTCOM0_2		0		1/1
XBOUTCOM0_3	Reg Stauts	0		1/1
XBOUTCOM0_4	Trigger Alarm Page	0		1/1
XBOUTCOM0_5		0		1/1
XBOUTCOM0_6		0		1/1
XBOUTCOM0_7		0		1/1
XBOUTCOM0_8		0		1/1

OK
Cancel
?

Once a link has been connected to a COM 0 (XW OUT, XB OUT) FB input, a *Label*, *Unit* and a decimal point setting (*Digits*) can be assigned to this address in the *Parameters* window.

Labels, *Unit* and *Digits* are commentaries that are saved in an XML file in addition to the program.

em4 Modbus Addresses

Further Modbus addresses 1:

Name	R/W	Address		
MSB STATE ETAT MSB ESTADO MSB	R	0051	0x0000	
LSB STATE ETAT LSB ESTADO LSB	R	0052	0x0000 : Stop - Stop - Stop	0x0001 : Run - Run - Ejecutar 0x0002 : Debugging - Debugging - Debugging 0x0040 : Setting front - Paramétrage en face avant - Ajuste frontal 0x0800 : Défaut alimentation
MSB STATUS STATUS MSB ESTADO MSB	R	0053	See code warning / error - Voir code warning / erreur - Ver código de advertencia / error	
LSB STATUS STATUS LSB ESTADO LSB	R	0054	See code warning / error - Voir code warning / erreur - Ver código de advertencia / error	

Further Modbus addresses 2:

Name	R/W	Address	
CLOCK HORLOGE RELOJ	R/W	0055	Second - Seconde - Segundo
	R/W	0056	Minute - Minute - Minuto
	R/W	0057	Hour - Heure - Hora
	R/W	0058	Day - Jour - Día
	R/W	0059	Date - Date - Fecha
	R/W	0060	Month - Mois - Mes
	R/W	0061	Year - Année - Año
SUMMER/WINTER ETE/HIVER VERANO/INVIERNO	R/W	0063	Change - Changement - Cambio 0x0000 : invalid - non validé - no validado 0x0001 : Europe - Europe - Europa 0x0002 : USA - USA - EE. UU. 0x0003 : manual - manuel - manual
	R/W	0064	SUMMER month - Mois ETE - Mes de VERANO 0x0001 : January - Janvier - Enero 0x000C : December - Décembre - Diciembre
	R/W	0065	Date SUMMER - Date ETE - Fecha de VERANO 0x0001 : 1 st Sunday - 1 ^{er} dimanche - 1er domingo 0x0005 : 5 th Sunday - 5 ^{ème} dimanche - 5to domingo
	R/W	0066	WINTER Month - Mois HIVER - Mes de INVIERNO 0x0001 : January - Janvier - Enero 0x000C : December - Décembre - Diciembre
	R/W	0067	Date HIVER 0x0001 : 1 st Sunday - 1 ^{er} dimanche - 1er domingo 0x0005 : 5 th Sunday - 5 ^{ème} dimanche - 5to domingo
DRIFT DERIVE DERIVA	R/W	0068	Drift - Dérive - Deriva 0xFFC5 : -59 0x003B : +59
RUN/STOP	W	0069	0x0000 : Stop - Stop - Parar 0x0001 : Run - Run - Ejecutar

MTPX/XX MODBUS RTU WIRING

MTPX/XX Modbus RTU Wiring

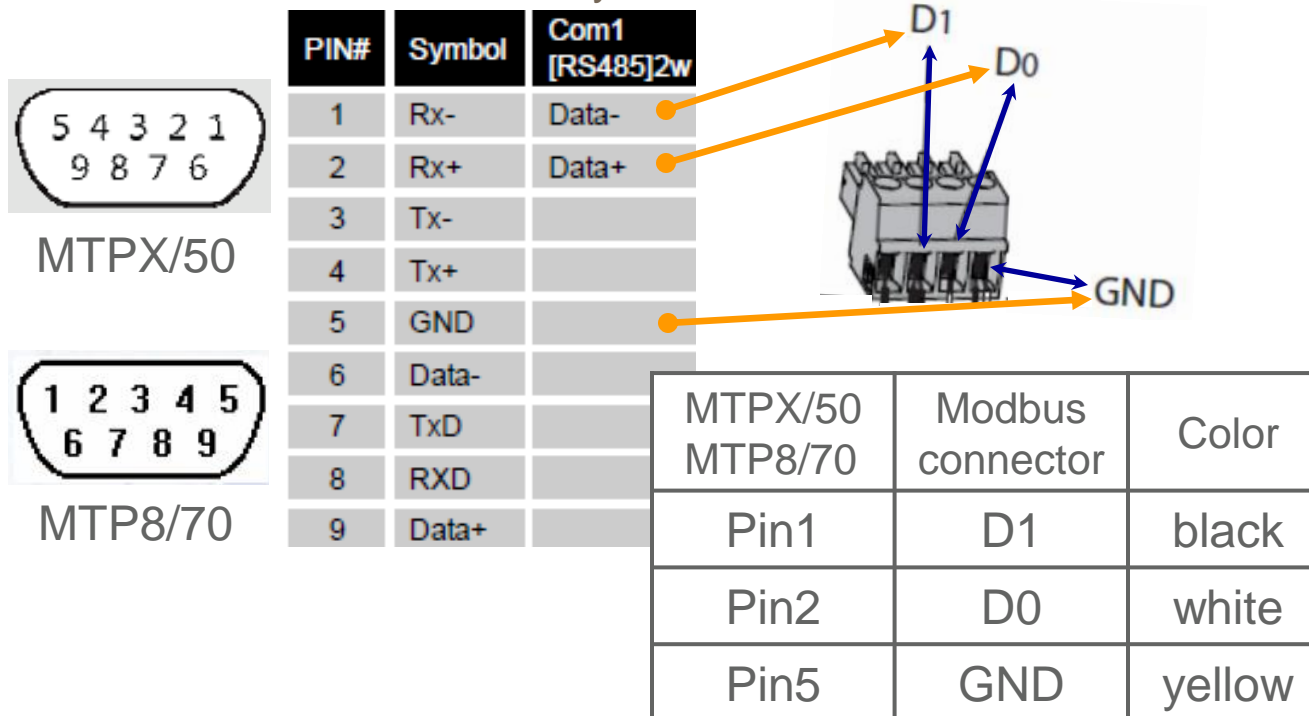
Two different Modbus communication cables exist today:

- 88 970 503 (SubD9 Female to RJ45) for the *MTPX/50*
- 88 970 504 (SubD9 Male to RJ45) for the *MTP8/70*

**Preliminary
solution**

To be used with the em4 Modbus extension, the RJ45 connectors have to be cut off, the wires need to be stripped and connected according to the table below.

MTPX/50 & 8/70 connector layout:



If you are unsure about the pin ⇔ color layout of your cable, or if the Modbus communication doesn't work, please 'beep' (check) the connections!

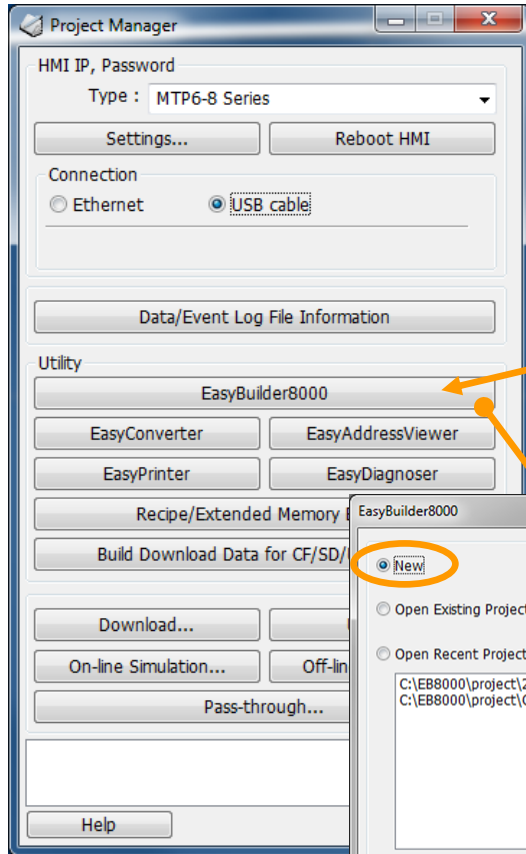
MTPX/XX Modbus RTU Wiring



- Insert the *88 970 503 Modbus communication cable* with the wired connector to the serial port of the *MTPX/50*
or
- The *88 970 504 Modbus communication cable* with the wired connector to the serial port of the *MTP8/70*
- Plug the connector into the *em4 Modbus S RS485* interface inserted in the the *em4*

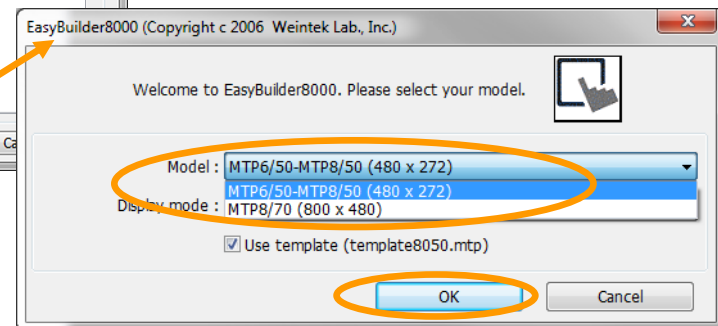
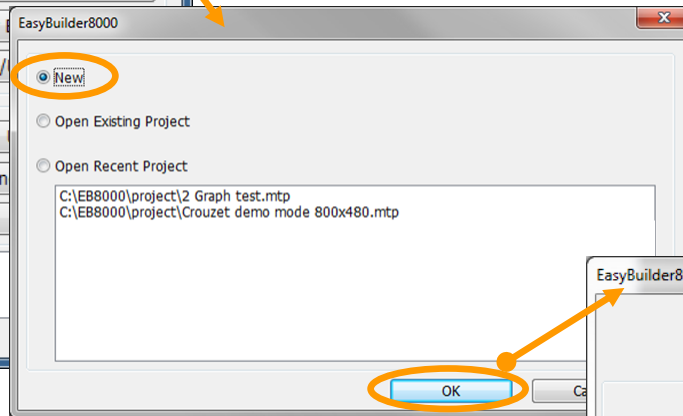
MTPX/XX MODBUS RTU NETWORK CONFIGURATION

Modbus RTU Network Configuration

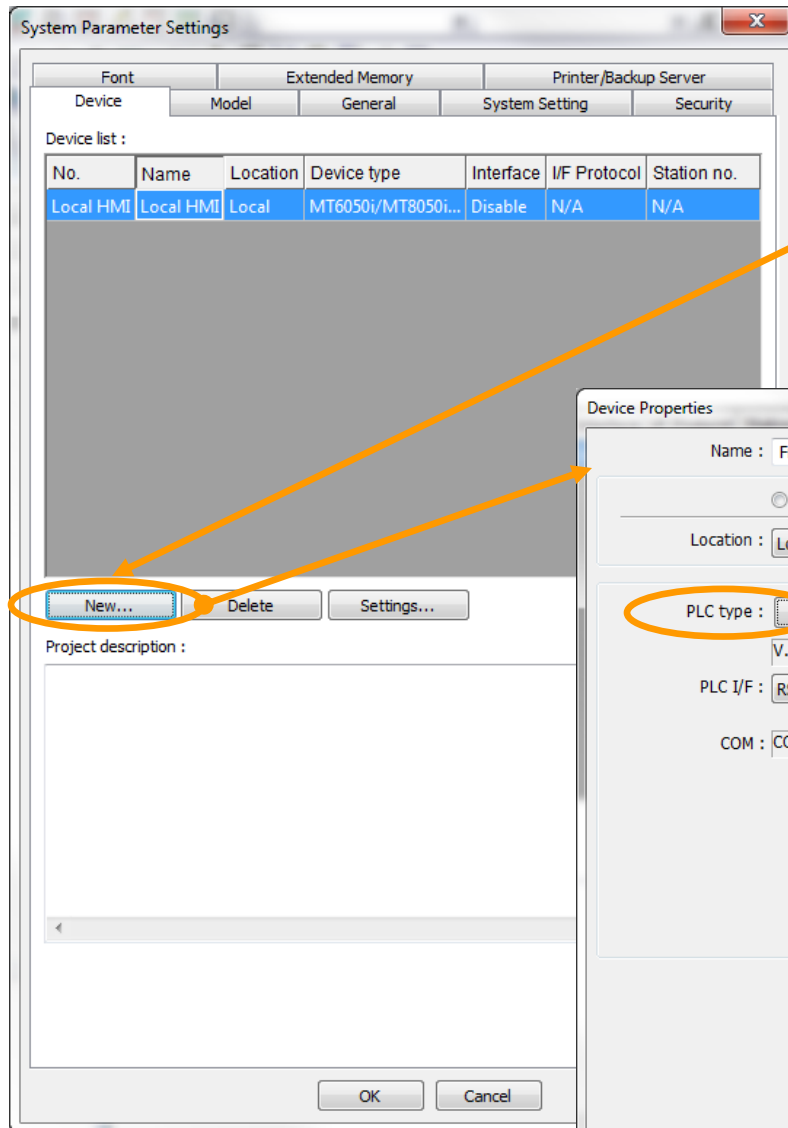


In order to define the Modbus network in the EB software:

- Create a new project
- Select the MTP screen version that is to be used and click on *OK*

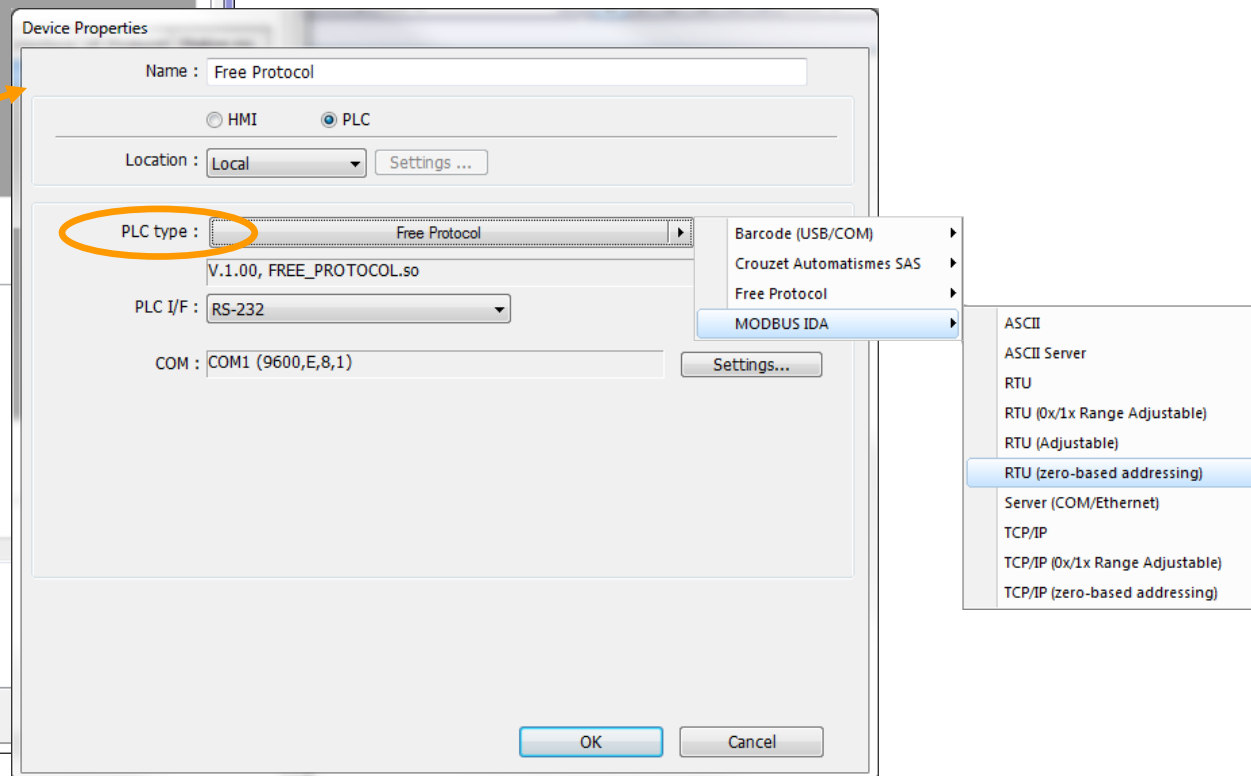


Modbus RTU Network Configuration



In the window that opens click *New* to define the *Device* (the network)

- In *PLC type* select *Modbus IDA*, then *RTU* (zero-based addressing)



Modbus RTU Network Configuration

Device Properties

Name : MODBUS RTU (zero-based addressing)

☐ HMI ☒ PLC

Location : Local Settings...

PLC type : MODBUS RTU (zero-based addressing)
V.1.40, MODBUS_RTU_ZERO_BASED.si

PLC I/F : RS-485 2W

COM : COM1 (9600,E,8,1) Settings...

PLC default station no. : 1

☐ Default station no. use station no. variable

☐ Use broadcast command

[How to designate the station no. in object's address ?...](#)

Interval of block pack (words) : 5

Max. read-command size (words) : 120

Max. write-command size (words) : 120

Sequence of events

☐ Enable

OK

- Enter the em4 *PLC default station no.* (slave number)
- Click *Settings* to define the communication parameters
- The communication parameters have to be set *identical* in the MTPX/XX and in each em4 slave!
- Repeat the steps *New, Device, Slave n°* and *COM Settings* for every em4 in the Modbus network

COM Port Settings

COM : COM 1

Baud rate : 9600

Data bits : 8 Bits

Parity : Even

Stop bits : 1 Bit

Timeout (sec) : 1.0

Turn around delay (ms) : 0

Send ACK delay (ms) : 0

Parameter 1 : 0

Parameter 2 : 0

Parameter 3 : 0

The number of resending commands : 0

OK Cancel

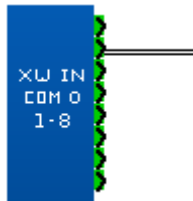
MODBUS RTU ADDRESSING

Modbus RTU Addressing Word

Example of how to address a word

Writing a set point from MTP to
em4 (slave n° 5)

⇒ em4: COM 0, XW IN 2



⇒ EB: *Device type 4x*
Address 5#2

Slave n°5, write address XW IN 2

Address

PLC name : MODBUS RTU (zero-based addressing)

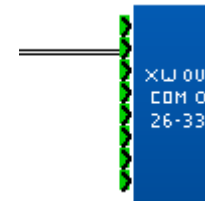
Device type : 4x

Address : 5#2

Address format : DDDDD [range : 0 ~ 65535]

Reading an em4 value (slave n° 5)
by the MTP

⇒ em4: COM 0, XW OUT 27



⇒ EB: *Device type 4x*
Address 5#27

Slave n°5, read address XW OUT 27

Address

PLC name : MODBUS RTU (zero-based addressing)

Device type : 4x

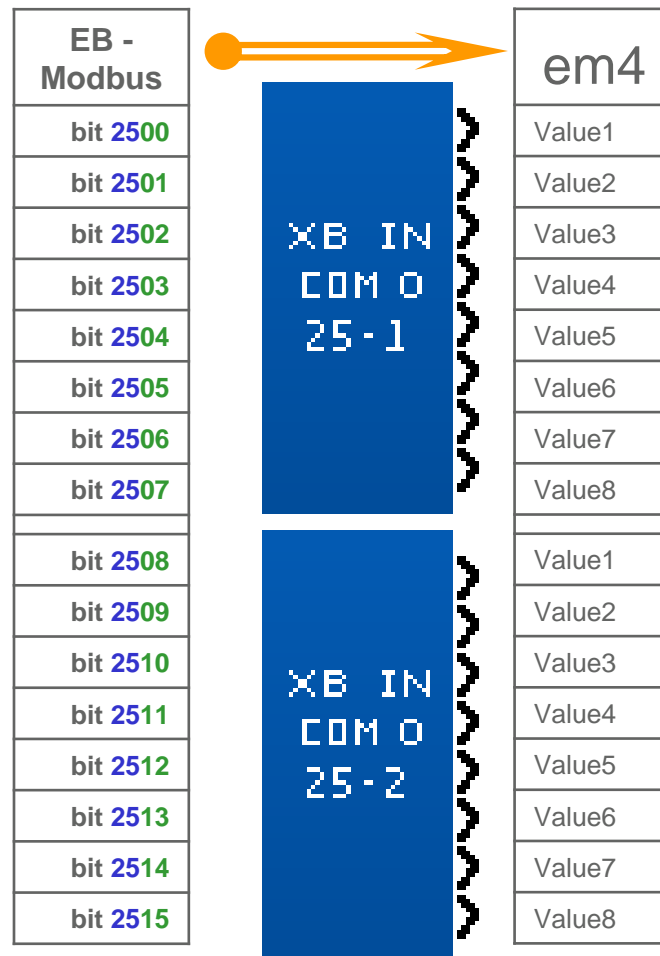
Address : 5#27

Address format : DDDDD [range : 0 ~ 65535]

Modbus RTU Addressing Write Bit

EB software: write/read a bit to em4 via Modbus RTU

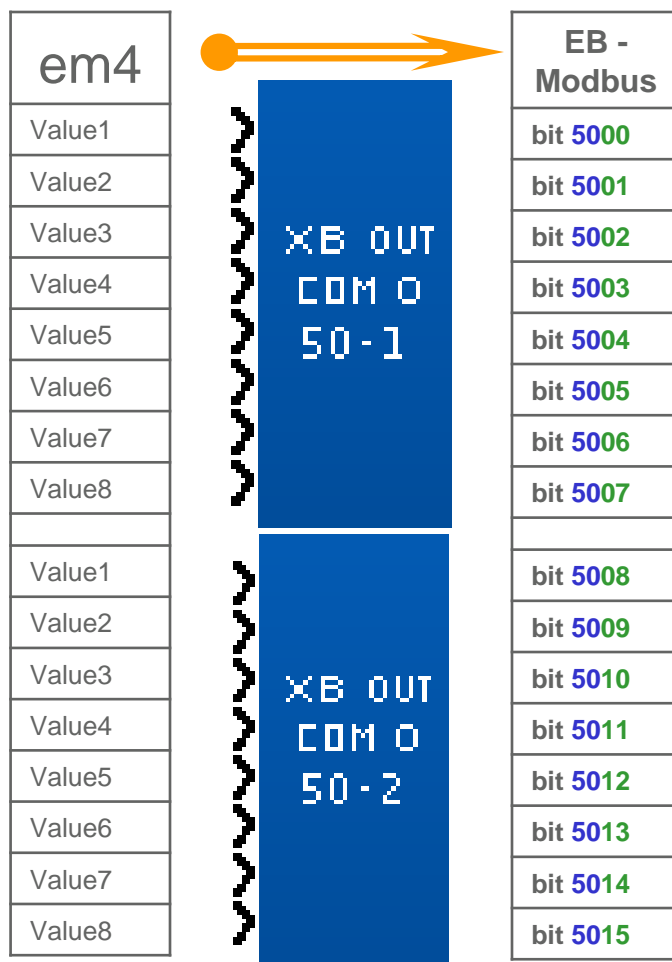
Modbus to em4



Modbus RTU Addressing Read Bit

EB software: reading a bit from em4 via Modbus RTU

em4 to Modbus



Modbus RTU Addressing

Example of how to address a bit

Writing a bit from the MTP to em4
(slave n° 5)

⇒ em4: COM 0, XB IN 25-1 Value4



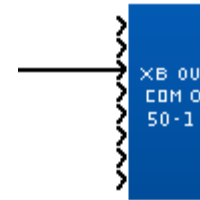
⇒ EB: *Device type:* 4x_Bit
Address: 5#2503

Slave n°5, write address XB IN 25-1

Address	
PLC name :	MODBUS RTU (zero-based addressing)
Device type :	4x_Bit
Address :	5#2503
Address format : DDDDDdd [range : 0 ~ 6553515, dd (bit no.) : 00 ~ 15]	

Reading an em4 bit (slave n° 5) by the
MTP

⇒ em4: COM 0, XB OUT 50-1 Value3



⇒ EB: *Device type:* 4x_Bit
Address: 5#5002

Slave n°5, read address XB OUT 50-1

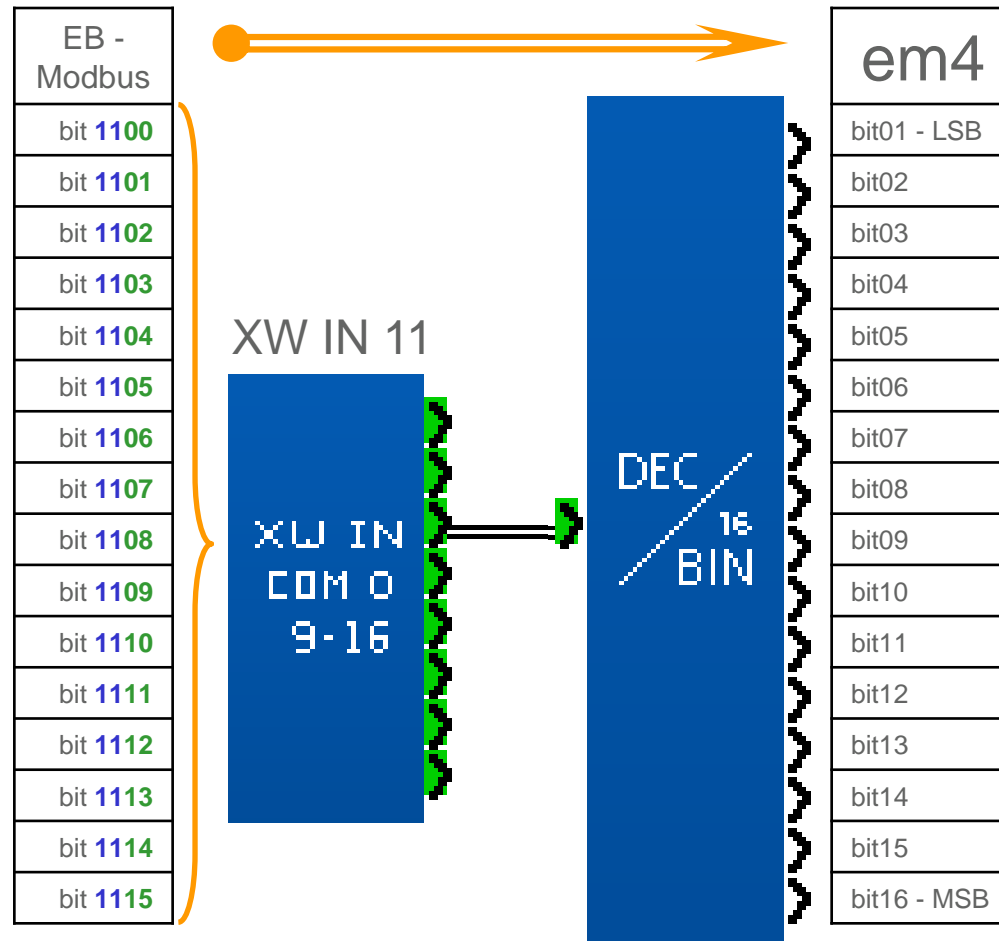
Address	
PLC name :	MODBUS RTU (zero-based addressing)
Device type :	4x_Bit
Address :	5#5002
Address format : DDDDDdd [range : 0 ~ 6553515, dd (bit no.) : 00 ~ 15]	

MODBUS RTU ADDRESSING USING BIN/DEC CONVERTER FB'S

Modbus RTU Addressing

EB software: writing/reading a bit to em4 via Modbus RTU
Using DEC/BIN converter option

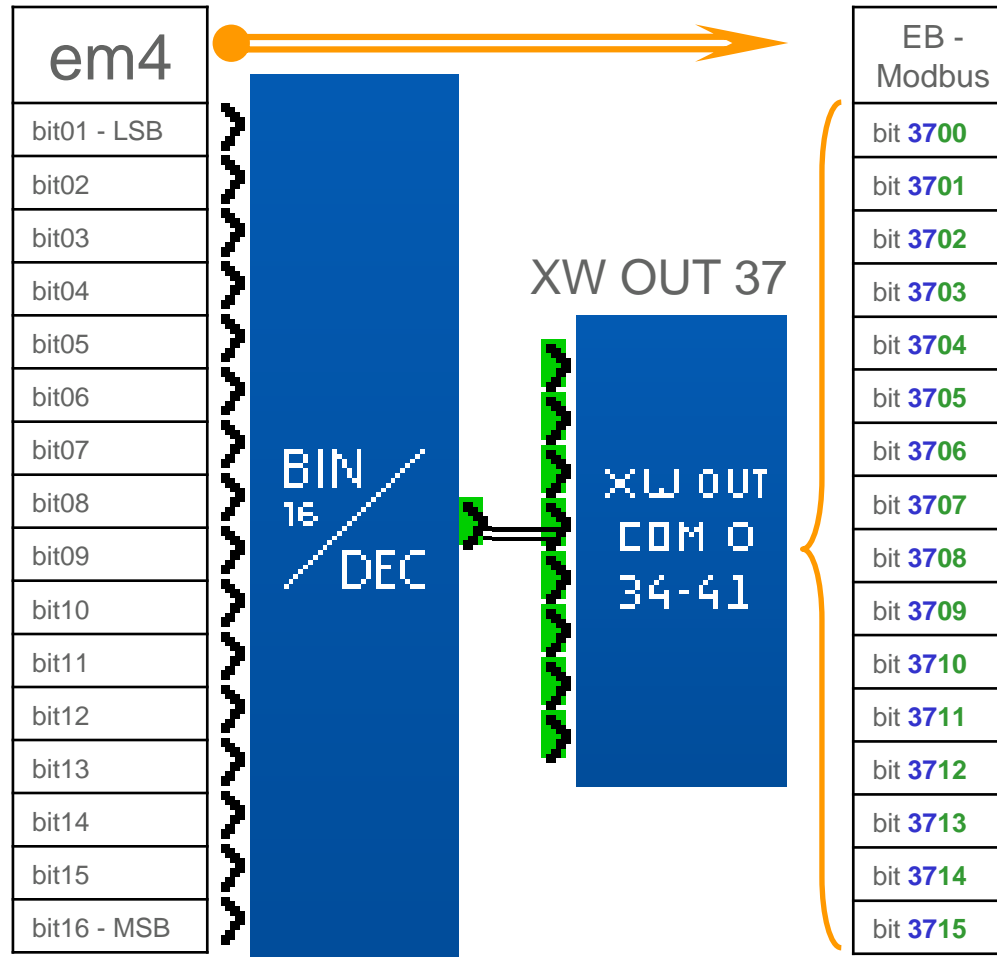
Modbus to em4



Modbus RTU Addressing

EB software: reading a bit from em4 via Modbus RTU
Using BIN/DEC converter option

em4 to Modbus



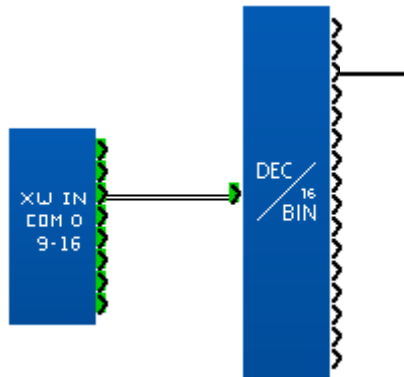
Modbus RTU Addressing

Example of how to address a bit using DEC/BIN and BIN/DEC converters

Writing a bit from the MTP to em4
(slave n° 1)

⇒ em4: COM 0, XW IN 11, bit 03

⇒ EB: *4x_Bit*, 1#1102



Address

PLC name : MODBUS RTU (zero-based addressing)

Device type : **4x_Bit**

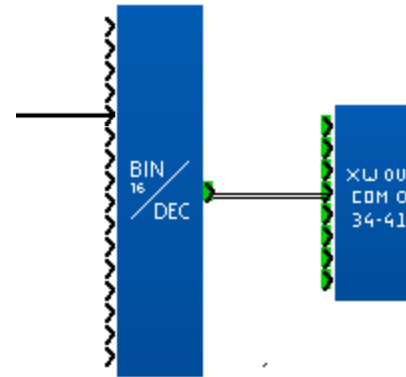
Address : 1#1102

Address format : DDDDDdd [range : 0 ~ 6553515, dd (bit no.) : 00 ~ 15]

Reading an em4 bit (slave n° 1) by the
MTP

⇒ em4: COM 0, XW OUT 37, bit 05

⇒ EB: *4x_Bit*, 1#3704



Address

PLC name : MODBUS RTU (zero-based addressing)

Device type : **4x_Bit**

Address : 1#3704

Address format : DDDDDdd [range : 0 ~ 6553515, dd (bit no.) : 00 ~ 15]

THANK YOU FOR YOUR ATTENTION

WWW.CROUZET.COM

www.em4-remote-plc.com