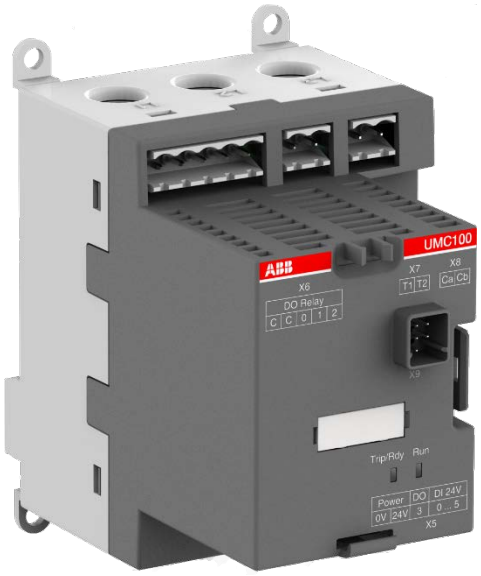
APPLICATION NOTE

Setting up an CP610 HMI as Modbus TCP master with connected MTQ22

Using the HMI as Modbus TCP master without additional PLC



For some applications only a visualization and the possibility for controlling and monitoring a few devices is needed. In these cases a PLC would be oversized and to expensive.

Instead of the PLC the ABB Eco Panel CP610 can be used. This HMI has an integrated Modbus master and can be used for controlling and monitoring some UMCs which are connected to the Modbus via the communication module MTQ22.0

In this application note an example will be shown, how to create the pages and the tags for writing and reading data. With this example the basic functions of a UMC100.3 can be controlled.

The configuration of the panel can be done completely by using this application note. The complete project is attached to this application note.

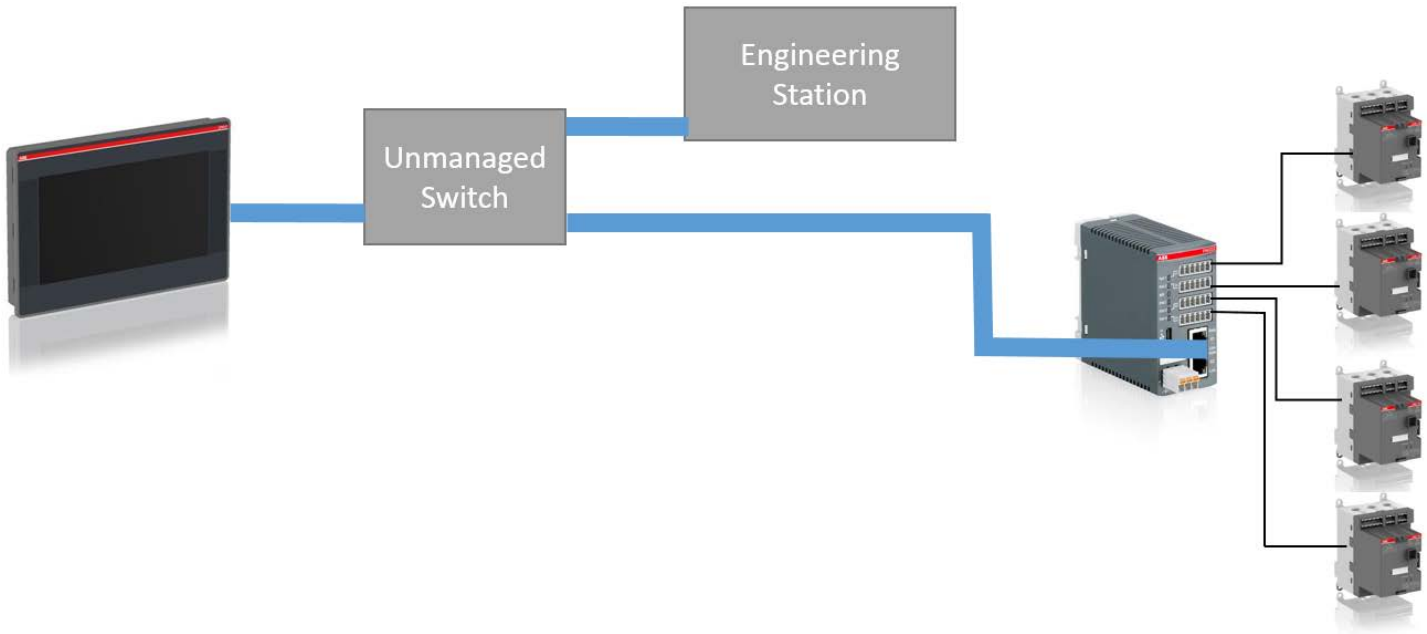
2CDC135050M0201

Hardware

In this application note the following hardware is used:

1. 1 pc. ABB Eco Panel CP610 A0 (1SAP510100R0001)
2. 1 pc. MTQ22.0 A6 (1SAJ260000R0100)
3. 4 pcs. UMC100.3 A1 (1SAJ530000R0100)
4. 1 pc. Hirschmann RS2 unmanaged Switch
5. Connection cables

The hardware will be connected like this:



Software

For the configuration the following software is needed:

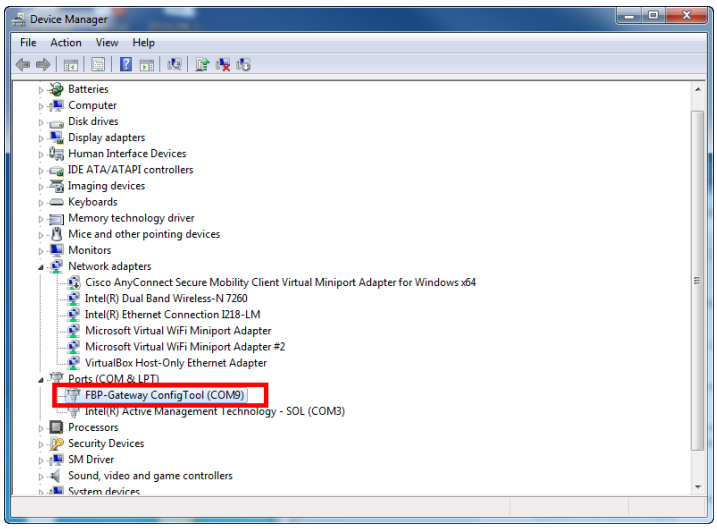
1. ABB Config Tool (for the configuration of the MTQ22.0)
2. ABB Panel Builder 600 Basic (for the configuration of the CP610)

Configuration of the MTQ22.0

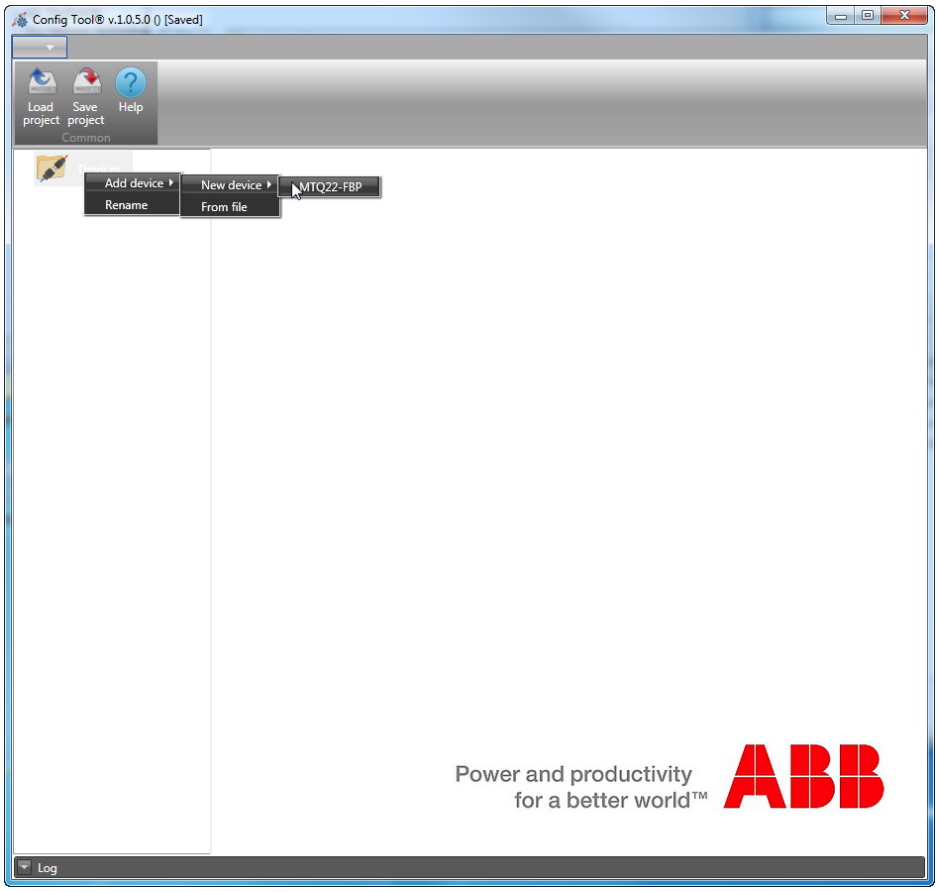
To configure the MTQ22.0 a MicroUSB cable is needed.

1. Open ABB ConfigTool
2. Connect MTQ22 to your PC via MicroUSB cable

3. Open Device Manager (Start → Control Panel → Hardware and Sound → Device Manager) and check the COM port, where MTQ22 is connected



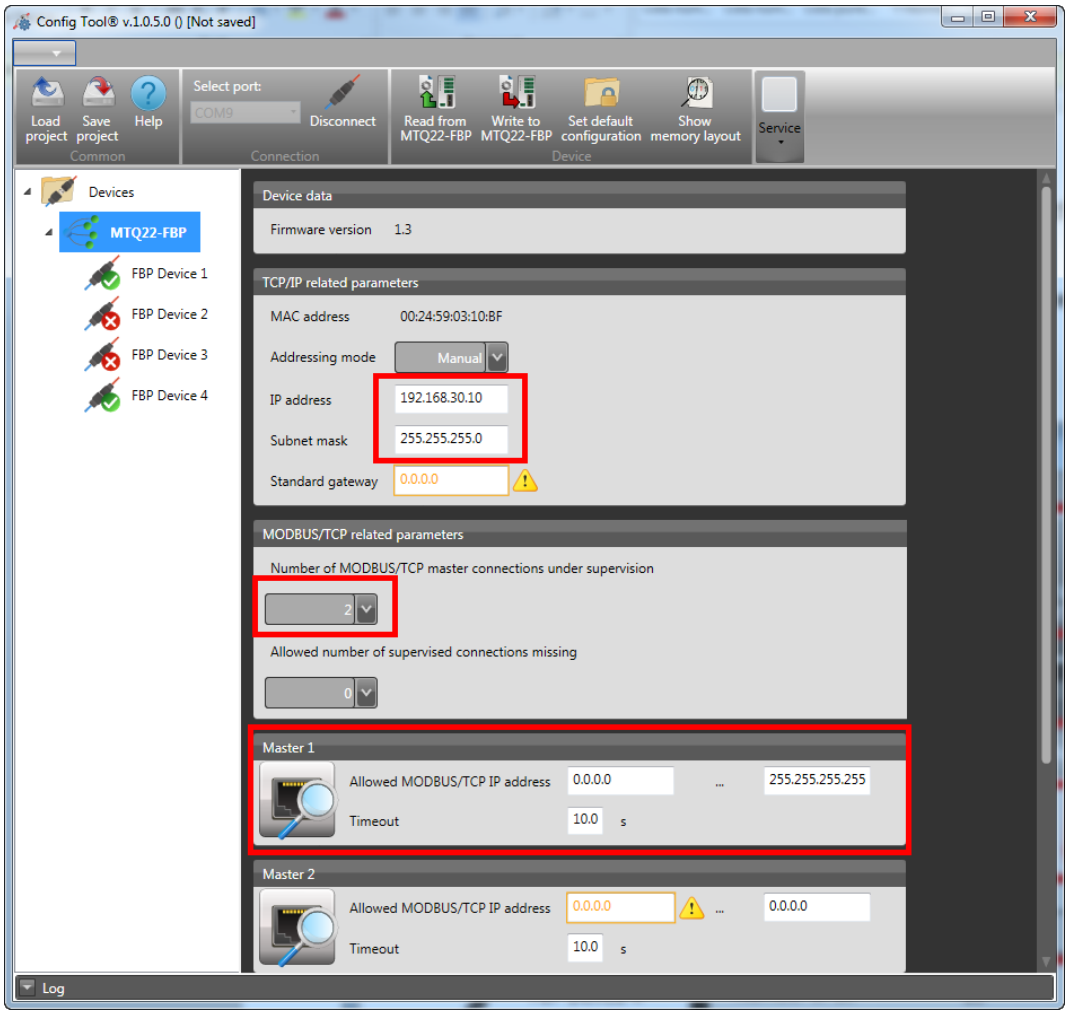
4. Right click on “Devices” to add a new device



5. Select the COM port (see point 3) and click “Connect”

6. Set the IP address and the Subnet mask.

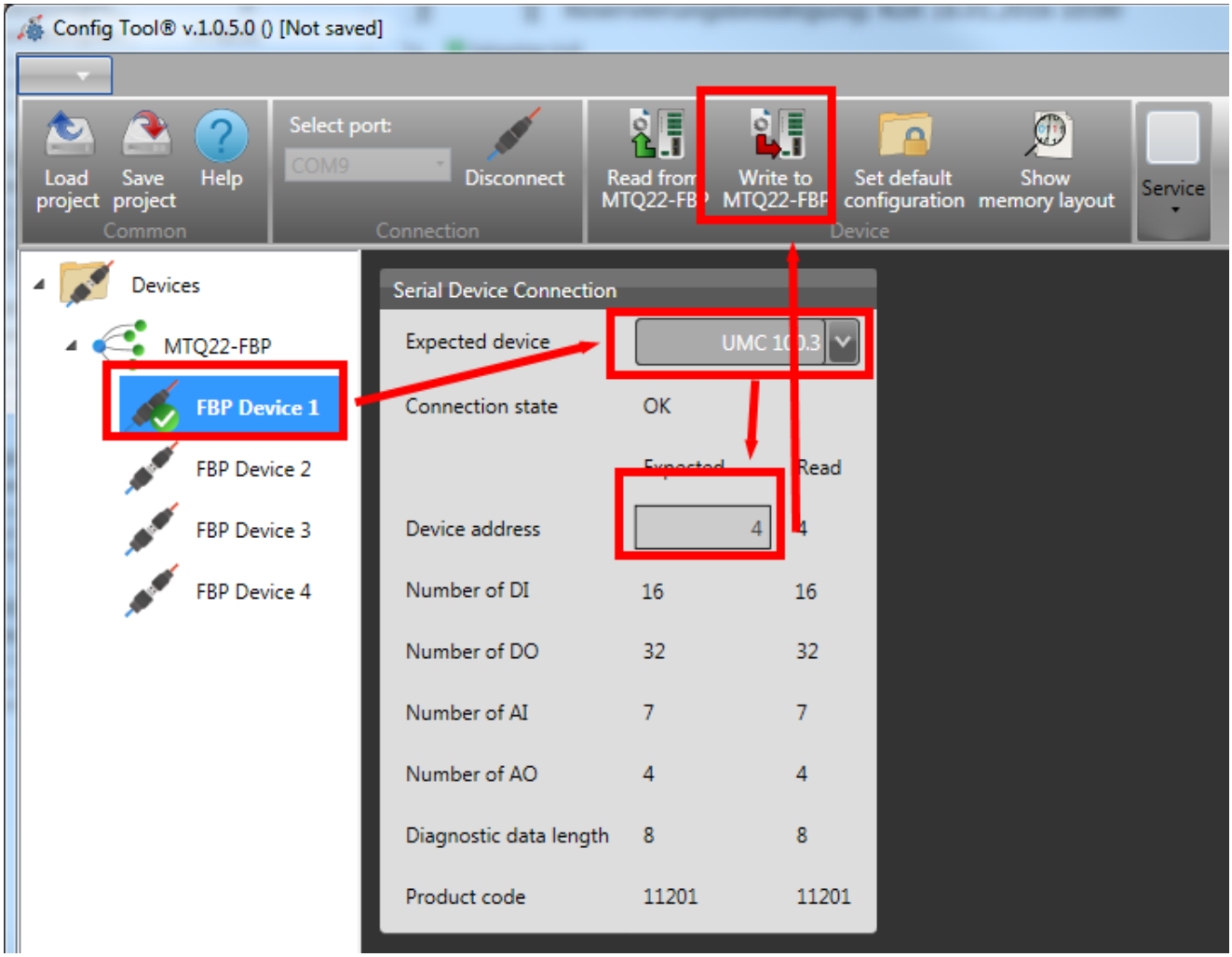
7. Set the allowed IP range for the master and the time out time



8. Click on the first device “FBP Device 1” and select the expected device and the FBP address (FBP address will be set on UMC panel)

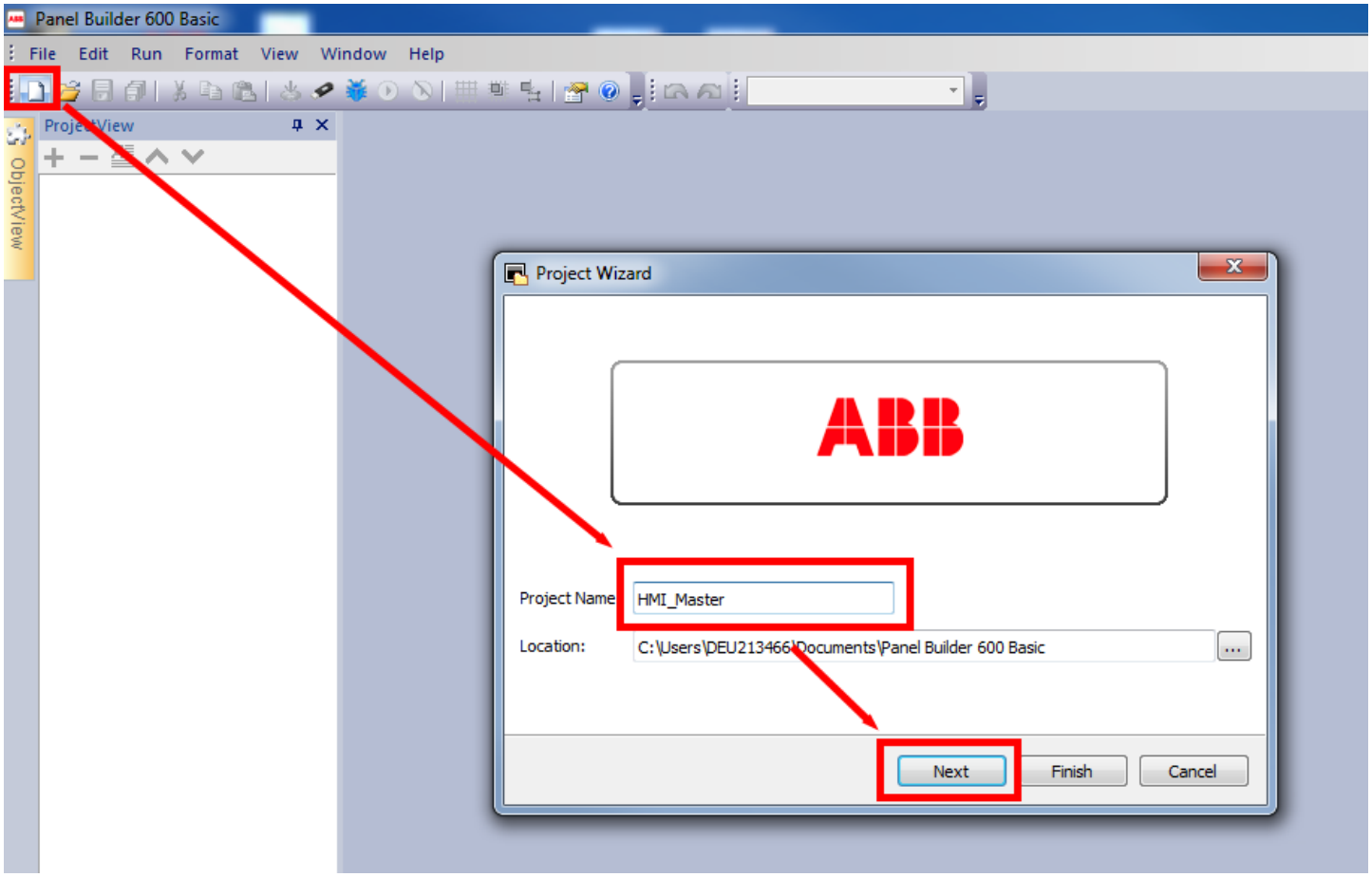
9. Repeat step 7 for all four devices (IMPORTANT: when you use less than four devices you have to select “Port not used” on the empty slot!)

10. Click on “Write to MTQ22-FBP”

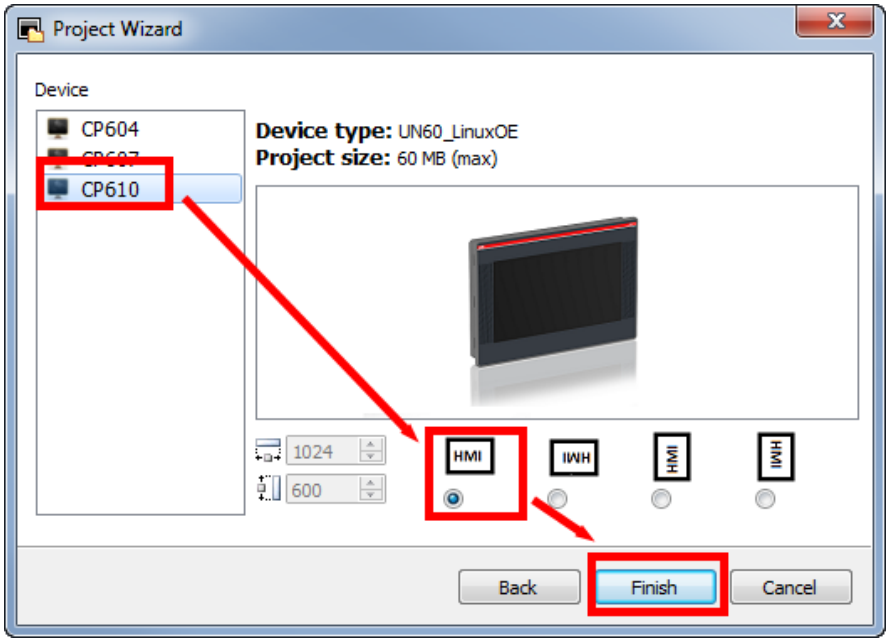


Creating an HMI project

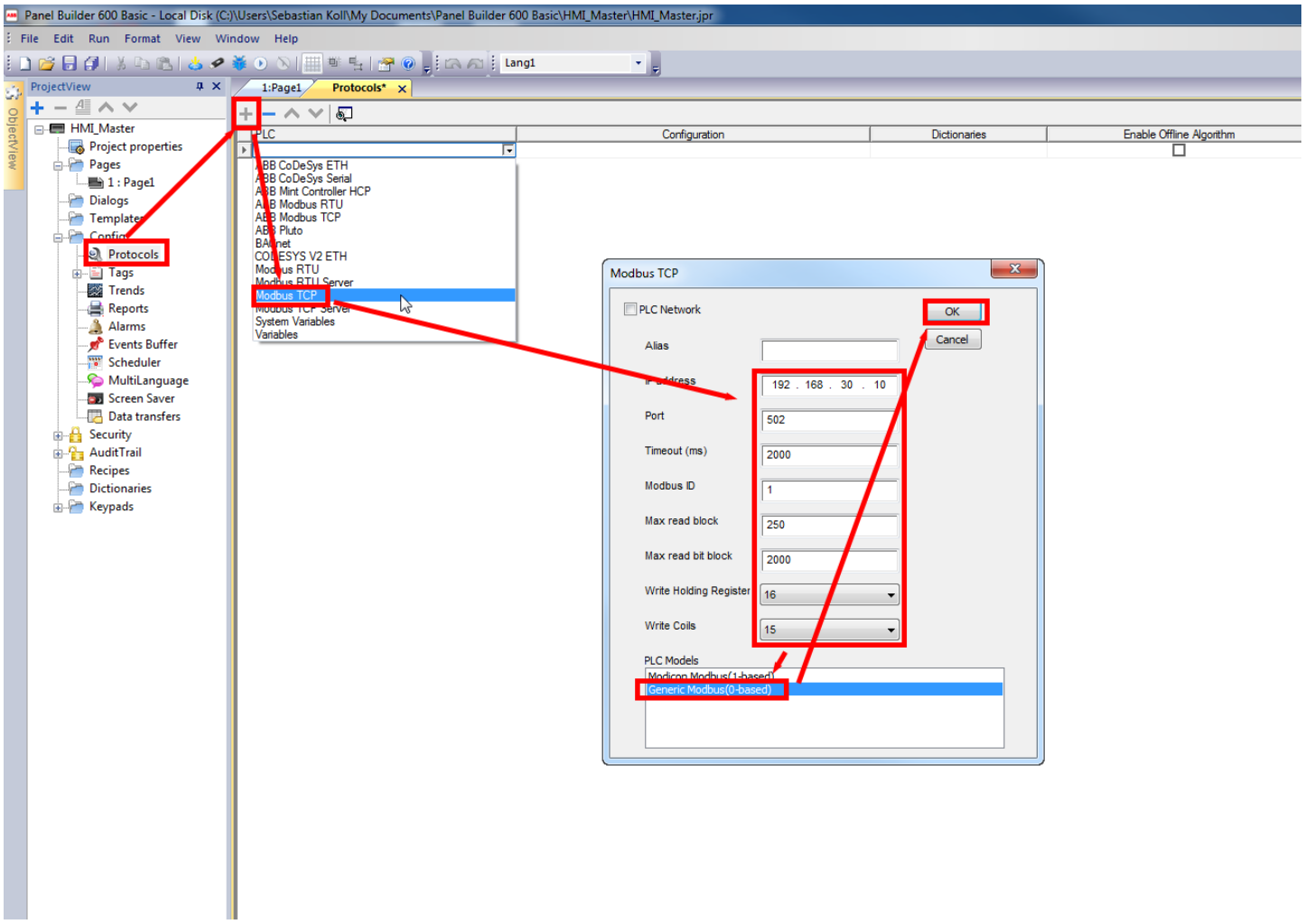
1. Open Panel Builder 600 Basic
2. Click on “New”
3. Insert the project name and click on “Next”



4. Select the used device (in this application note a CP610 is used) and choose the orientation. Click on “Finish”



5. After clicking on “Finish” the “ProjectView” is filled with a tree and an empty page is shown in the main window.
6. Open “Config” in the tree and double click on “Protocols”
7. Click on “+”
8. Select “Modbus TCP” from the list
9. A new window opens. In case only one slave is connected, just insert the **slave** parameters and choose the “Generic Modbus(0-based)” model. In case several slaves are connected, tick the box “PLC Network” and click on “Add” to insert each slave. Confirm with “OK”.



Monitoring data

The following data are used and necessary in this project. The information, like register addresses and data mapping, used in this chapter are taken from the MTQ22.0 manual (2CDC194003D0202):

- Feedback running reverse (Word 0, Byte 0, Bit 0)
- Feedback motor stopped (Word 0, Byte 0, Bit 1)
- Feedback running forward (Word 0, Byte 0, Bit 2)
- Local control (Word 0, Byte 0, Bit 5)
- Summary fault (Word 0, Byte 0, Bit 6)
- Summary warning (Word 0, Byte 0, Bit 7)
- Motor current in percent (Word 1)

Monitoring Data

	Word	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0	
Profile 1	0	0	Summary Warning	Summary Fault	Local Control[3]	Reverse Lockout Time[3]	Overload warning	Run Forward[1,3] / Opening[2,3]	Off[3]	Run Reverse[1,3] / Closing[2,3]	Profile 2
		1	UMC100 DI5	UMC100 DI4	UMC100 DI3	UMC100 DI2	UMC100 DI1	UMC100 DI0	Run Fast Forward4)	-	
	1	2, 3	Motor Current in % of I_e (0% - 800%)								
	2	4, 5	Analogue Word (Thermal Load: 0% - 100%)								
	3	6, 7	Analogue Word (Time to trip in seconds)								
	4	8, 9	Analogue Word (Time to restart in seconds)								
	5	10.11	Analogue Word (Active power in selected scale)								
	6	12	DX1xx DI7	DX1xx DI6	DX1xx DI5	DX1xx DI4	DX1xx DI3	DX1xx DI2	DX1xx DI1	DX1xx DI0	
		13	-	-	Run Time Exceeded[2]	Out of Position[2]	Torque Open[2]	Torque Closed[2]	End Pos Open[2]	End Pos Closed[2]	
	7	14	U Imbal. warn	U Imbal. trip	Under-voltage warn	Under-voltage trip	Under-power warn	Under-power trip	Over-power warn	Over-power trip	
		15	Earth fault warning	Earth fault trip	Cooling time running	-	THD warning	No start possible[5)]	1 start left [5)]	More than 1 start left[5)]	

For the data access via Modbus TCP we'll request the data in Word 0 with a bit oriented access:

Bit oriented access to the monitoring data of UMC100 acc. to the UMC100 manual section A1 Parameters and Data Structures on a Fieldbus

UMC at port 1	UMC at port 2	UMC at port 3	UMC at port 4	Bit Information	
Address					
0x1000 dec 4096	0x1100 dec 4352	0x1200 dec 4608	0x1300 dec 4864	Run Reverse / Closing	Bit 0
0x1001 dec 4097	0x1101 dec 4353	0x1201 dec 4609	0x1301 dec 4865	Off	
...	...	...	...	...	
0x100F dec 4111	0x130F dec 4867	0x130F dec 4823	0x130F dec 4879	UMC100 DI5	Bit 15

In the picture above the bit oriented access to monitoring word 0 is shown with addresses for each bit and each port.

For the monitoring word 1 with the motor current in percent the word oriented access will be used to get one word containing the current:

Analog Inputs of UMC100 acc. to the UMC100 manual section A1 Parameters and Data Structures on a Fieldbus

UMC at port 1	UMC at port 1	UMC at port 1	UMC at port 1	Data
Address				
0x2000 dec 8192	0x2100 dec 8448	0x2200 dec 8704	0x2300 dec 8960	Motor Current in % of I_e (0% - 800%
0x2001 dec 8193	0x2101 dec 8448	0x2201 dec 8704	0x2301 dec 8961	Analogue Word (Thermal Load: 0% - 100%)
0x2002 dec 8194	0x2102 dec 8450	0x2202 dec 8704	0x2302 dec 8962	Analogue Word (Time to trip in seconds)
0x2003 dec 8195	0x2103 dec 8451	0x2203 dec 8704	0x2303 dec 8963	Analogue Word (Time to restart in seconds)
0x2004 dec 8196	0x2104 dec 8452	0x2204 dec 8704	0x2304 dec 8964	Analogue Word (Active power in selected scale)

Command data

- Command start reverse (Word 0, Byte 0, Bit 0)
- Command stop (Word 0, Byte 0, Bit 1)
- Command start forward (Word 0, Byte 0, Bit 2)

- Auto mode (Word 0, Byte 0, Bit 5)
- Fault reset (Word 0, Byte 0, Bit 6)

Command Data

	Word	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0	
Profile 1	0	0	-	Fault Reset	Auto Mode[3]	Prepare Emergency Start	-	Run Forward[1,3] / Opening[2,3]	Off[3]	Run Reverse[1,3] / Closing[2,3]	Profile 2
		1	UMC100 DO2	UMC100 DO1	UMC100 DO0	UMC100 24 V DC Out	-	-	Run Fast Forward [4)]	-	
	1	2	VI15x DO0	-	-	-	DX1xx DO3	DX1xx DO2	DX1xx DO1	DX1xx DO0	
		3	-	-	-	-	-	-	-	-	
	2	4, 5	Analogue Word								
	3	6, 7	Analogue Word								
	4	8,9	Analogue Word								
	5	10,11	Analogue Word								

For the writing the command data on the Modbus TCP network we use the bit oriented access:

Bit oriented access to the command data of UMC100 acc. to the UMC100 manual section A1 Parameters and Data Structures on a Fieldbus

UMC at port 1	UMC at port 2	UMC at port 3	UMC at port 4	Bit Information	
Address					Bit 0
0x1800 dec 6144	0x1900 dec 6400	0x1A00 dec 6656	0x1B00 dec 6912	Run Reverse / Closing	
0x1801 dec 6145	0x1901 dec 6401	0x1A01 dec 6657	0x1B01 dec 6913	Off	
...	...	...	...	...	Bit 31
0x1820 dec 6175	0x 191F dec 6431	0x1A1F dec 6687	0x1B1F dec 6943	-	

Diagnostic data UMC

All four diagnostic words are needed

Diagnosis Data

Word	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0
0	0	Checkback missing	PTC wiring failure	PTC hot	Pre-waring thermal model	Locked rotor during start-up (stall)	Phase imbalance[1]	Phase loss[1]	Thermal overload trip
	1	Actuator problem[1]	UMC self-test error	Earth fault pre-warning	Eart fault trip (internal or externally triggered)	I above high current warning threshold	I above high current trip threshold	I below low current warning threshold	I below low current trip threshold
1	2	Trip/Warning from AuxFault function block input 5[2)]	Trip/Warning from AuxFault function block input 4[2)]	Trip/Warning from AuxFault function block input 3[2)]	Trip/Warning from AuxFault function block input 2[2)]	Trip/Warning from AuxFault function block input 1[2)]	HW fault on IO module	Custom application error	IO module missing
	3	Warning triggered from AM2	Trip triggered from AM2	Warning triggered from AM1	Trip triggered from AM1	Trip triggered from Multifunction input DI2	Trip triggered from Multifunction input DI1	Trip triggered from Multifunction input DI0	Trip / Warning from AuxFault function block input 6[2)]
2	4	-	-	THD Warning	Voltage out of spec[1]	Overload power	Underload power[1]	-	-
	5	-	-	Cooling Time Running	Just one start left	Num Starts Overrun	-	-	-
3	6	Extended diagnosis is available[1)].	Parameter out of range	-	-	-	-	-	-
	7	Fault code. See section "Error Handling, Maintenance and Service-> Fault Messages" for a description of the code.							

The diagnostic registers for each connected slave can be accessed with the following register addresses:

Diagnosis Data of UMC100 according to the UMC100 manual section
A1 Parameters and Data Structures on a Fieldbus

UMC at port 1	UMC at port 2	UMC at port 3	UMC at port 4	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit 0
Adress											
0x0C00 dec 3072	0x0C20 dec 3104	0x0C40 dec 3136	0x0C60 dec 3168	Check-back missing	PTC wiring failure	PTC hot	Pre-waring thermal model	Locked rotor during start-up (stall)	Phase imbalance[1]	Phase loss[1]	Thermal overload trip
				Actuator problem[1]	UMC self-test error	Earth fault pre-warning	Eart fault trip (internal or externally triggered)	I above high current warning threshold	I above high current trip threshold	I below low current warning threshold	I below low current trip threshold
0x0C01 dec 3073	0xC021 dec 3105	0x0C41 dec 3137	0x0C61 dec 3167	Trip/Warning from AuxFault function block input 5[2)]	Trip/Warning from AuxFault function block input 4[2)]	Trip/ Warning from AuxFault function block input 3[2)]	Trip/Warning from AuxFault function block input 2[2)]	Trip/Warning from AuxFault function block input 1[2)]	HW fault on IO module	Custom application error	IO module missing
				-	-	-	-	Trip triggered from Multifunction input DI2	Trip triggered from Multifunction input DI1	Trip triggered from Multi-function input DI0	Trip / Warning from AuxFault function block input 6[2)]
0x0C02 dec 3074	0x0C22 dec 3106	0x0C42 dec 3138	0x0C62 dec 3168	-	-	THD Warning	Voltage out of spec[1]	Overload power	Underload power[1]	-	-
				-	-	Cooling Time Running	Just one start left	Num Starts Overrun	-	-	-
0x0C03 dec 3075	0x0C23 dec 3107	0x0C43 dec 3139	0x0C63 dec 3169	Extended diagnosis is available[1)].	Parameter out of range	-	-	-	-	-	-
				Fault code. See section "Error Handling, Maintenance and Service-> Fault Messages" for a description of the code.							

Diagnostic Data MTQ22

Four words are needed, one for each port.

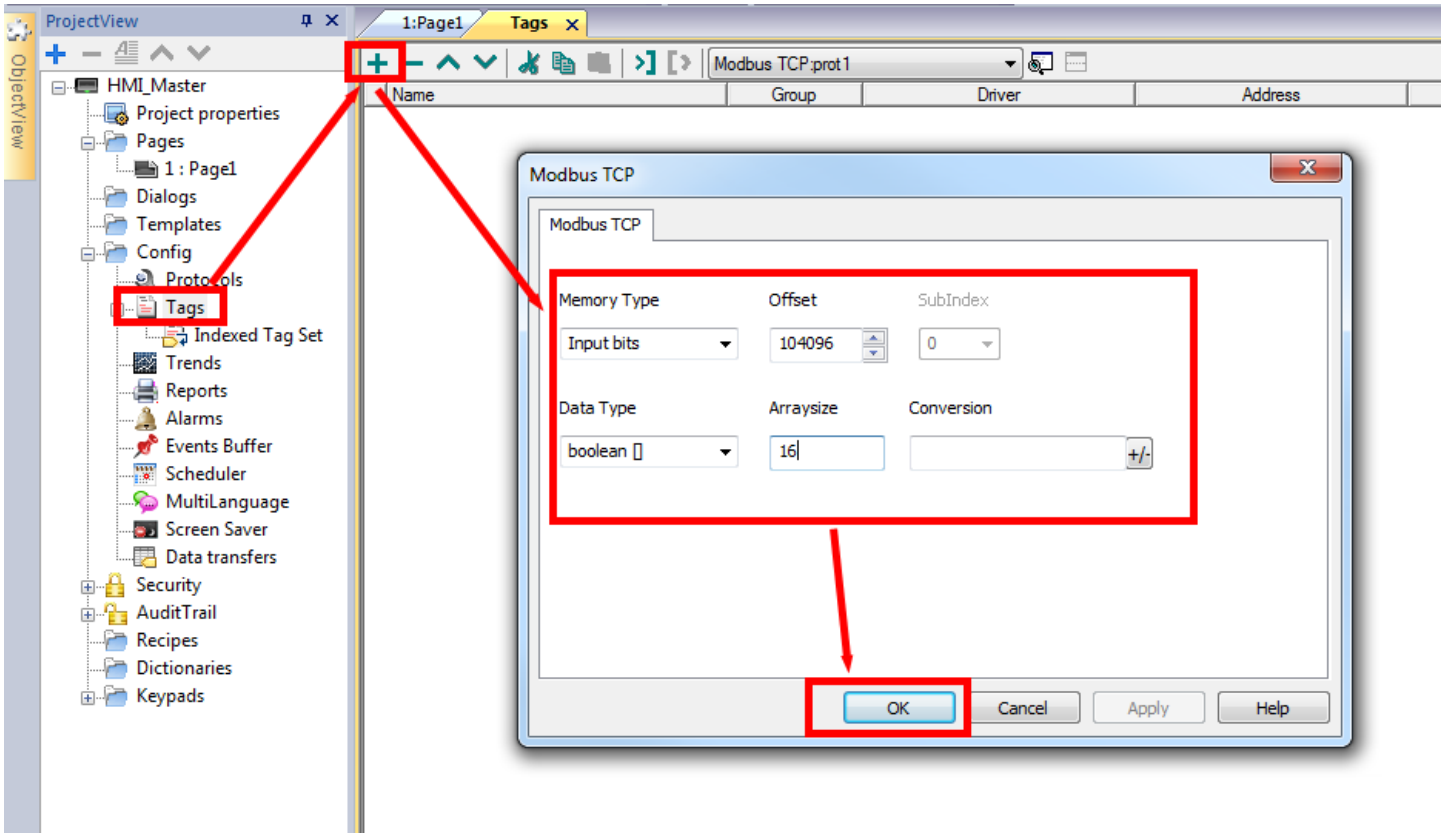
MTQ22 Status Data

Adress	Data	Size	Access
0x0000 - 0x007F (HEX) 0 - 127 (DEC)	General Parameter for MTQ22-FBP (reserved for further use)	128 Words	Read, Write
0x0080 - 0x0083 (HEX) 128 - 131 (DEC)	Diagnostics (one block)	4 Words	Read
0x0080 - 0x0083 (HEX) 128 - 131 (DEC)	FBP Port 1 ... Port 4 communication status	1 Word	Read
0x0090 - 0x0093 (HEX) 144 - 147 (DEC)	Number of connection losses between FBP device (e.g. UMC100) and MTQ22-FBP	1 Word	Read / Write

Tags

Tags are used for the communication. With a connected slave it's necessary to exchange data for monitoring and giving commands to the slave. With the MTQ22.0 as slave, there will be the possibility to connect up to four UMC100.3 and each of this sub slaves has its own tags. In this chapter will be explained how to access the monitoring data and how to create command telegrams.

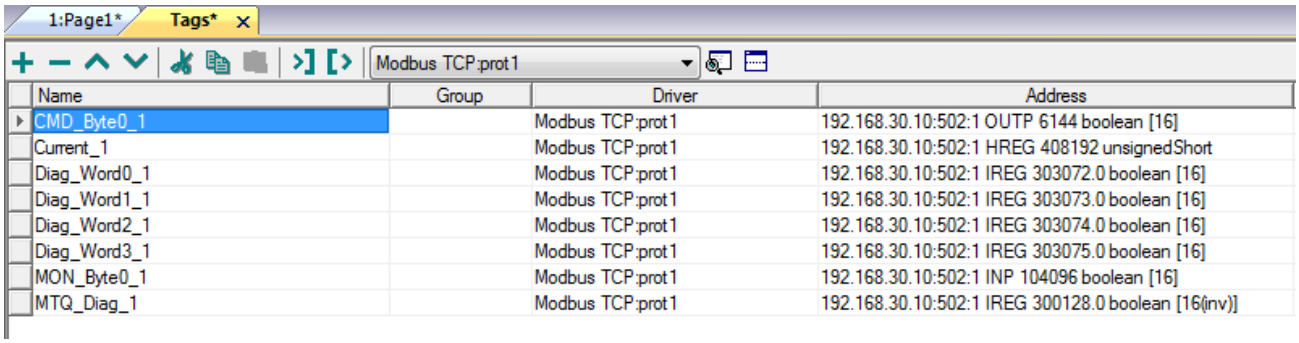
1. Open "Tags" in the ProjectView.
2. Click on "+" to add a tag.
3. Add all tags according to the table below. In the table is only shown the data for one port as example. In the previous chapter the addresses for the three other ports are shown.



TAG SETTINGS

Tag name	Memory Type	Offset	Data Type	Arraysize	Conversion
Mon_Byte0_1	Input bits	104096	Boolean []	16	-
Current_1	Holding registers	408192	unsignedShort	-	-
Cmd_Byte0_1	Output coils	6144	Boolean[]	16	-
Diag_Word0_1	Input registers	303072	Boolean[]	16	-
Diag_Word1_1	Input registers	303073	Boolean[]	16	-
Diag_Word2_1	Input registers	303074	Boolean[]	16	-
Diag_Word3_1	Input registers	303075	Boolean[]	16	-
MTQ_Diag_1	Input registers	300128	Boolean[]	16	INV

4. After completing the tags for port 1 it should look like this:



5. For the 3 other ports this will be done the same way with the addresses according the previous chapter.

Modbus TCP:prot1

Name	Group	Driver	Address
CMD_Byte0_1		Modbus TCP:prot1	192.168.30.10:502:1 OUTP 6144 boolean [16]
Current_1		Modbus TCP:prot1	192.168.30.10:502:1 HREG 408192 unsignedShort
Diag_Word0_1		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303072.0 boolean [16]
Diag_Word1_1		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303073.0 boolean [16]
Diag_Word2_1		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303074.0 boolean [16]
Diag_Word3_1		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303075.0 boolean [16]
MON_Byte0_1		Modbus TCP:prot1	192.168.30.10:502:1 INP 104096 boolean [16]
MTQ_Diag_1		Modbus TCP:prot1	192.168.30.10:502:1 IREG 300128.0 boolean [16(inv)]
CMD_Byte0_2		Modbus TCP:prot1	192.168.30.10:502:1 OUTP 6400 boolean [8]
CMD_Byte0_3		Modbus TCP:prot1	192.168.30.10:502:1 OUTP 6656 boolean [8]
CMD_Byte0_4		Modbus TCP:prot1	192.168.30.10:502:1 OUTP 6912 boolean [8]
Current_2		Modbus TCP:prot1	192.168.30.10:502:1 HREG 408448 unsignedShort
Current_3		Modbus TCP:prot1	192.168.30.10:502:1 HREG 408704 unsignedShort
Current_4		Modbus TCP:prot1	192.168.30.10:502:1 HREG 408960 unsignedShort
Diag_Word0_2		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303104.0 boolean [16]
Diag_Word0_3		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303136.0 boolean [16]
Diag_Word1_4		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303168.0 boolean [16]
Diag_Word1_2		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303105.0 boolean [16]
Diag_Word1_3		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303137.0 boolean [16]
Diag_Word0_4		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303167.0 boolean [16]
MON_Byte0_2		Modbus TCP:prot1	192.168.30.10:502:1 INP 104352 boolean [8]
MON_Byte0_3		Modbus TCP:prot1	192.168.30.10:502:1 INP 104608 boolean [8]
MON_Byte0_4		Modbus TCP:prot1	192.168.30.10:502:1 INP 104864 boolean [8]
MTQ_Diag_2		Modbus TCP:prot1	192.168.30.10:502:1 IREG 300129.0 boolean [16(inv)]
MTQ_Diag_3		Modbus TCP:prot1	192.168.30.10:502:1 IREG 300130.0 boolean [16(inv)]
MTQ_Diag_4		Modbus TCP:prot1	192.168.30.10:502:1 IREG 300131.0 boolean [16(inv)]
Diag_Word2_2		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303106.0 boolean [16]
Diag_Word3_2		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303107.0 boolean [16]
Diag_Word2_3		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303138.0 boolean [16]
Diag_Word3_3		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303139.0 boolean [16]
Diag_Word2_4		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303169.0 boolean [16]
Diag_Word3_4		Modbus TCP:prot1	192.168.30.10:502:1 IREG 303170.0 boolean [16]

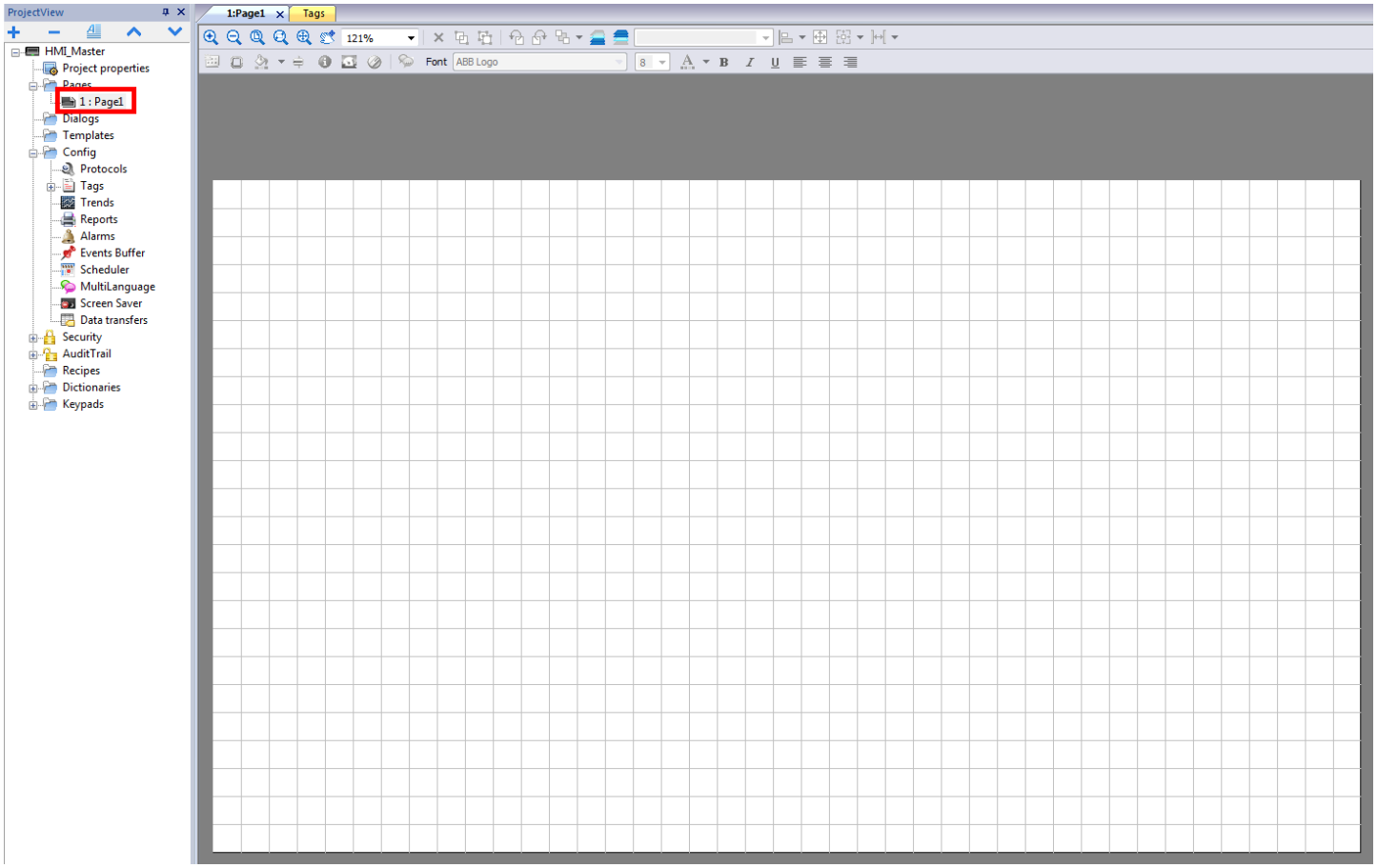
Page layout

The page layout shows what will be shown on the panel after downloading it and using it. As a user you are completely free to decide which information will be shown and which commands can be given.

In this example we'll create one cover page with a picture of a motor control center with several drawer. Four of the drawers can be entered for detailed information of the connected UMCs.

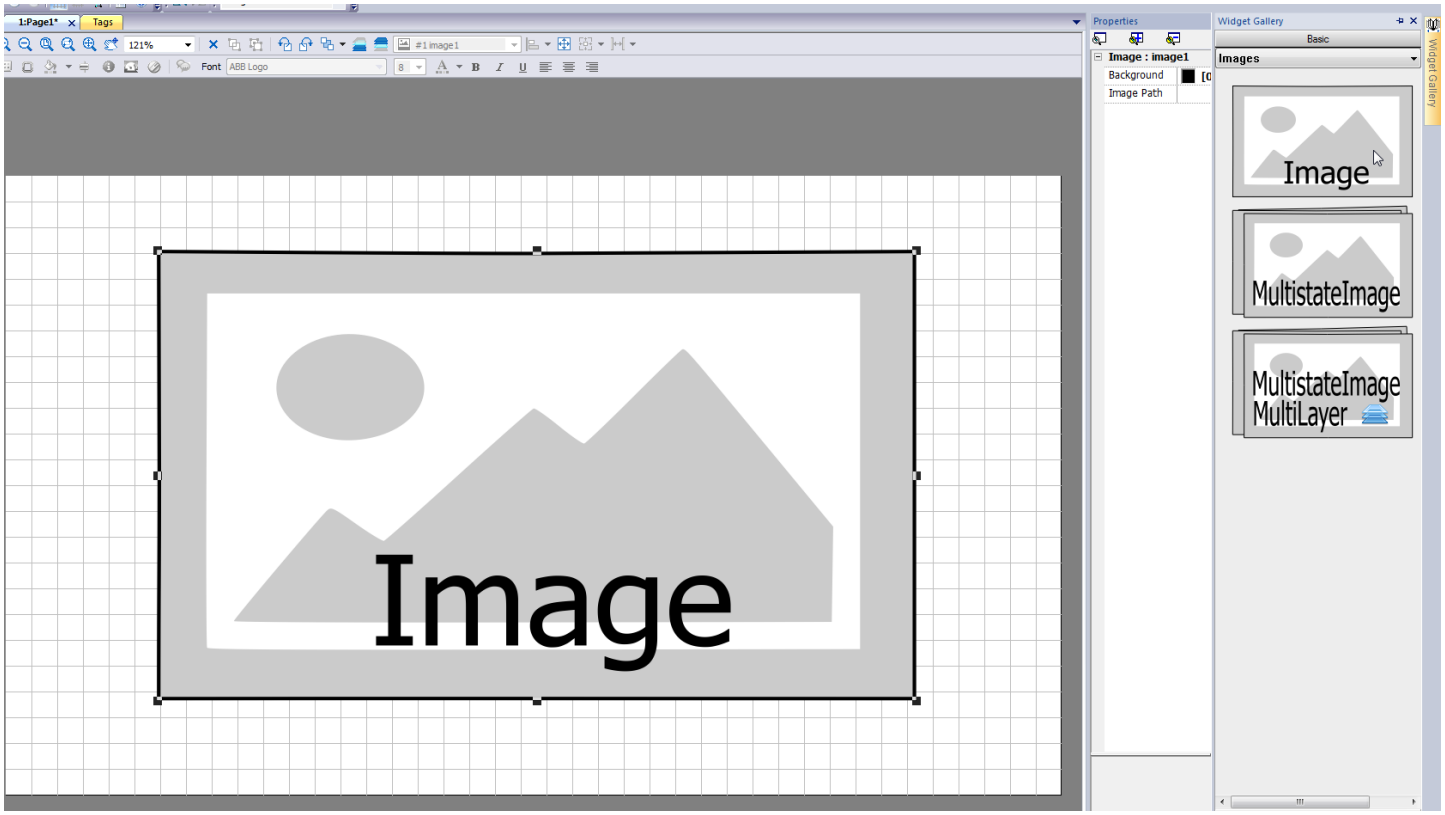
Cover page

1. Open the "Pages" folder in the ProjectView.
2. Open the page which is already there.

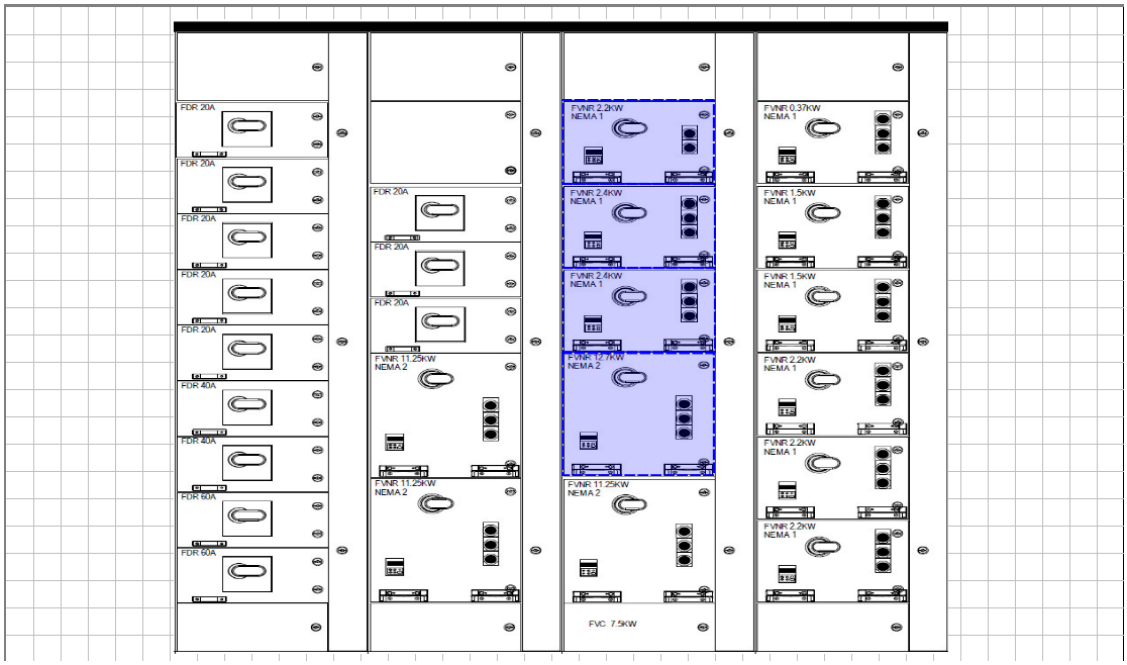


3. Open the widget gallery and select “Basic” → “Images” and insert it into the empty window by drag and drop.

4. Adapt the size of the image.



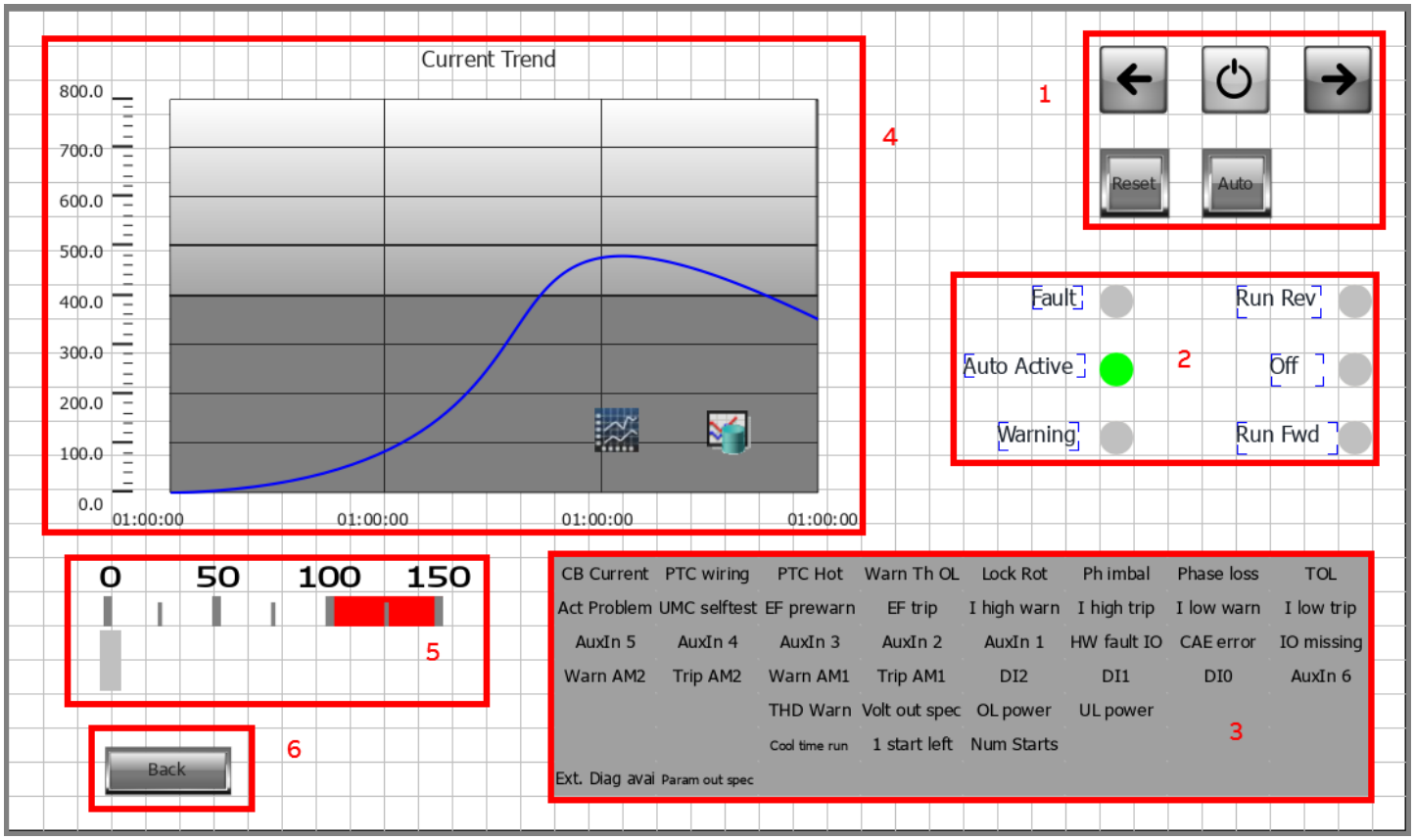
5. Click on “Image Path” in the properties and select the picture you want to use.
6. Open the widget gallery go on “Buttons” → “Buttons – others” and select the blue shape.
7. Repeat this three times. These buttons will open additional pages. Place them on the drawers where you want to operate the UMCs later.



UMC page

For the design of the UMC pages we include some buttons for controlling the UMC and also some monitors for getting the motor current, the diagnostics and also the motor state.

All used components can be found in the widget library. In or around the red boxes are numbers for a detailed description



1. Buttons for motor control. With the “Reset” button faults can be reset and with the “Auto” button the remote bit will be set.
2. Shows the actual state of the motor and also if faults or warnings are active.
3. In case of a fault the fault will be indicated by a red box.
4. Here the current history is shown in a settable scale and also the settable time range.
5. The actual current is shown in percent with a bar graph.
6. By pushing this button the user comes back to the main page.

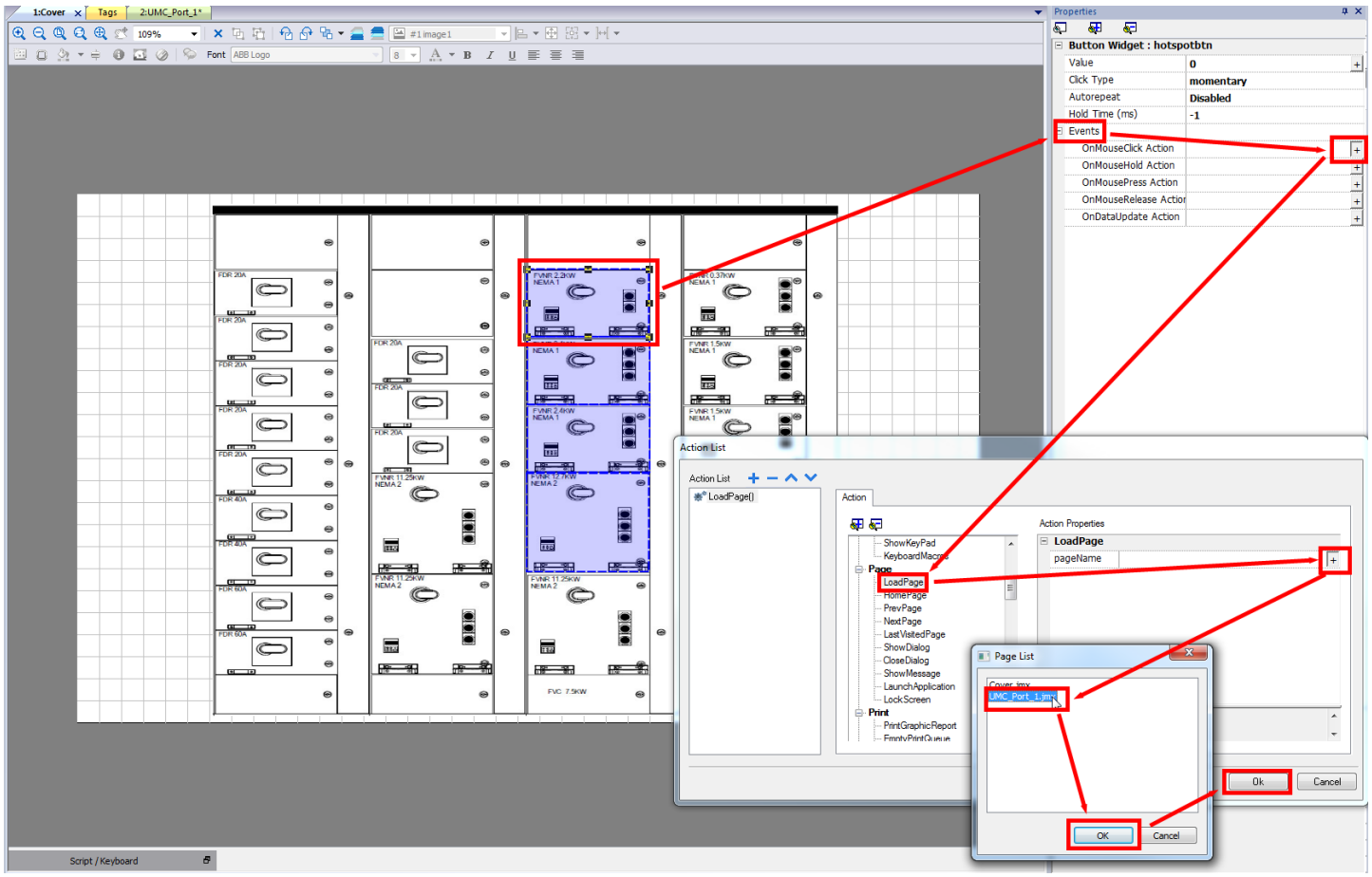
After finishing the page layout it can be saved and copied three times to get four pages with the same layout for each of the four ports of the MTQ.

Connecting data with the page layout

In this chapter the whole data will be linked to the widgets.

Cover

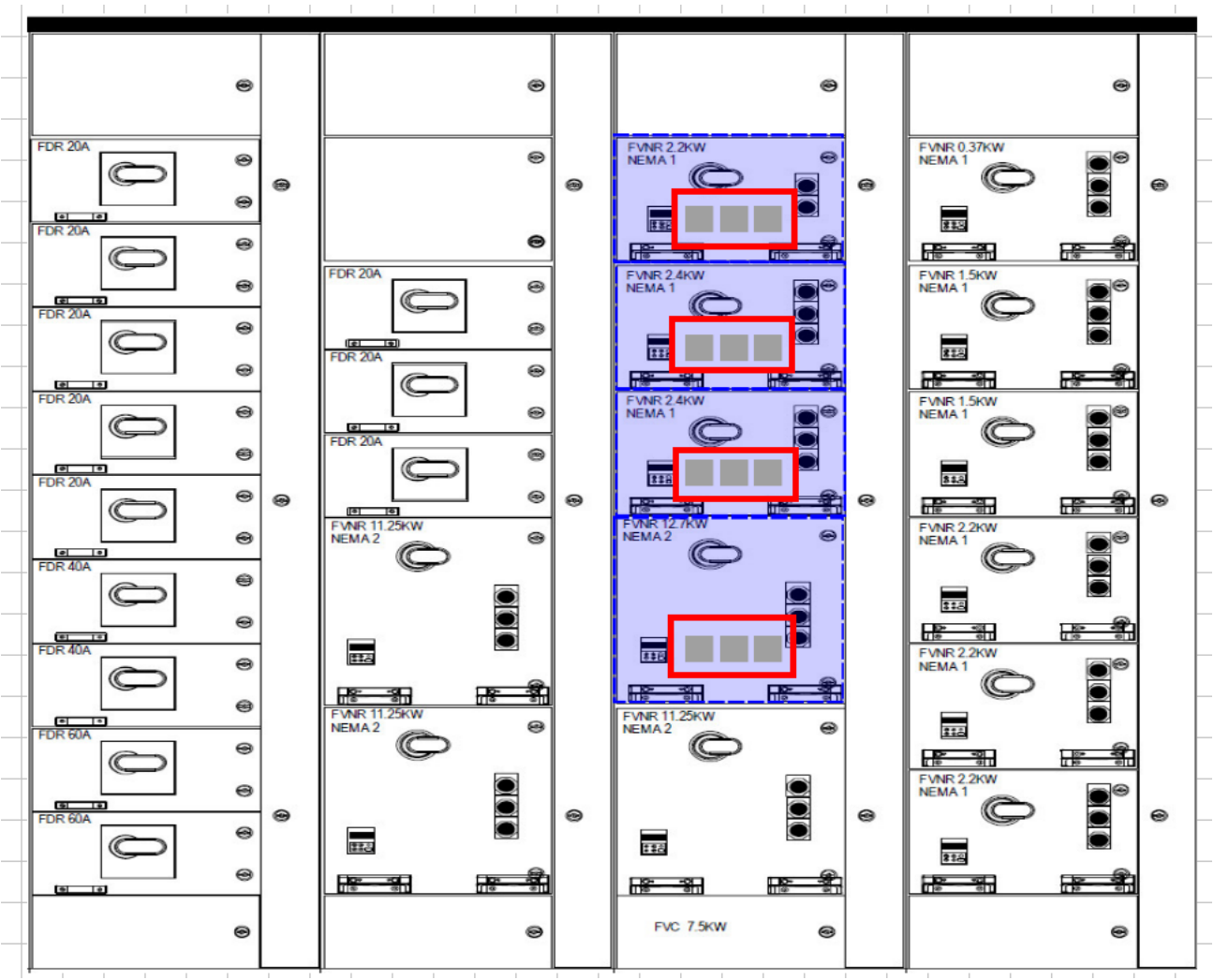
1. Select one of the blue shades on the cover and expand “Events” in the properties.
2. Click on the “+” next to “OnMouseClick Action”.
3. Scroll down to select “LoadPage” as action.
4. Click on “+” next to “pageName” and select the UMC page you want to link to this shade.
5. Repeat this for all pages.



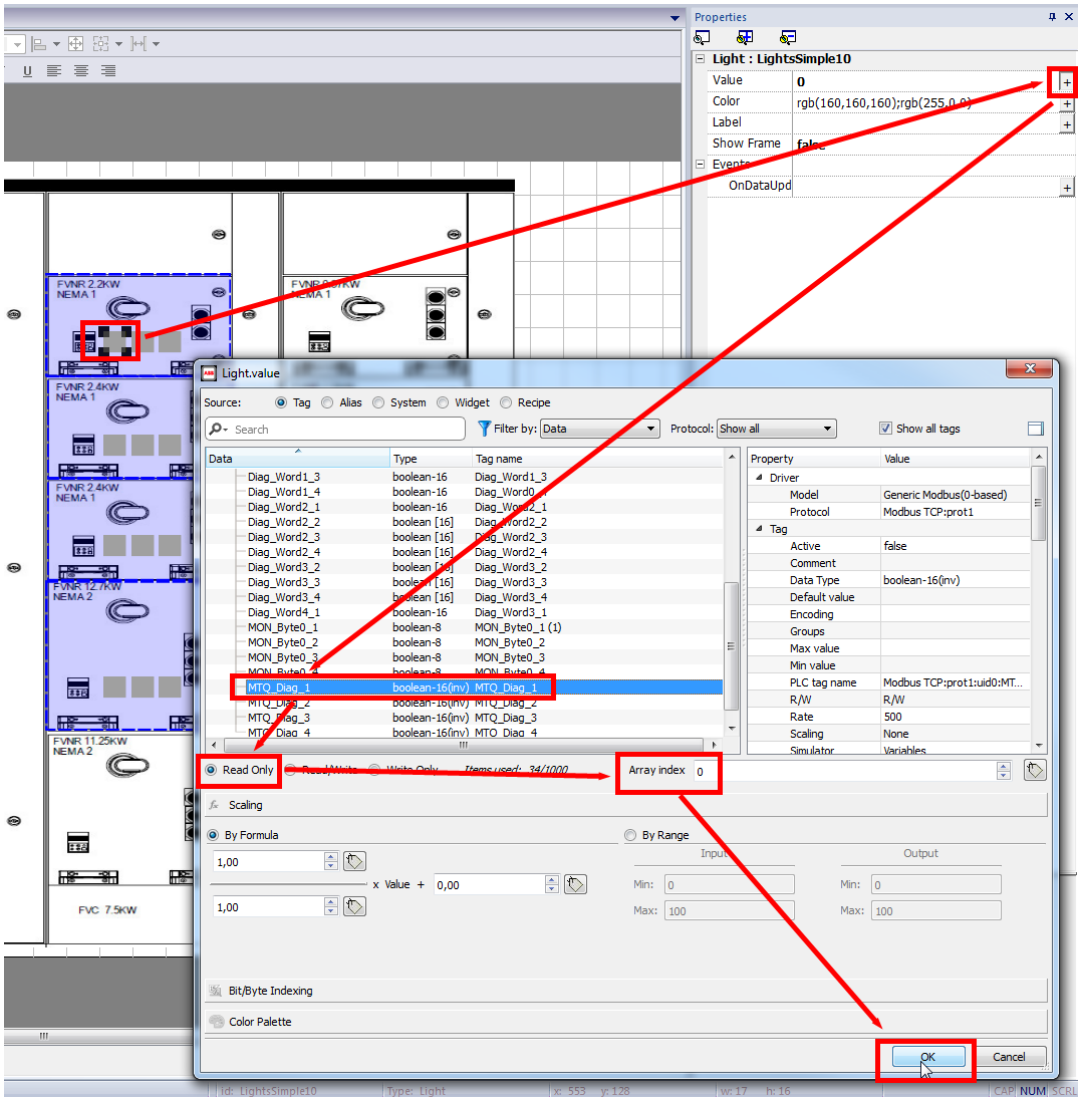
Furthermore there shall be an indication on the cover page for following information:

- UMC is powered on and reachable
- Current is flowing through the UMC
- Fault state of the UMC

1. For each drawer three indicators will be placed on the cover page



2. Select the left indicator. This will indicate if the UMC is reachable.
3. Click on “+” next to “Value”
4. Select the “MTQ_Diag_1” tag
5. Set “Read only” and set array index to “0”
6. Apply with “OK”



7. For the second indicator a workaround is necessary, because it's not possible to use logic operators. The current is flowing in case the UMC is reachable and the UMC is not stopped. Therefore we'll set the LED to visible only when the UMC is available.

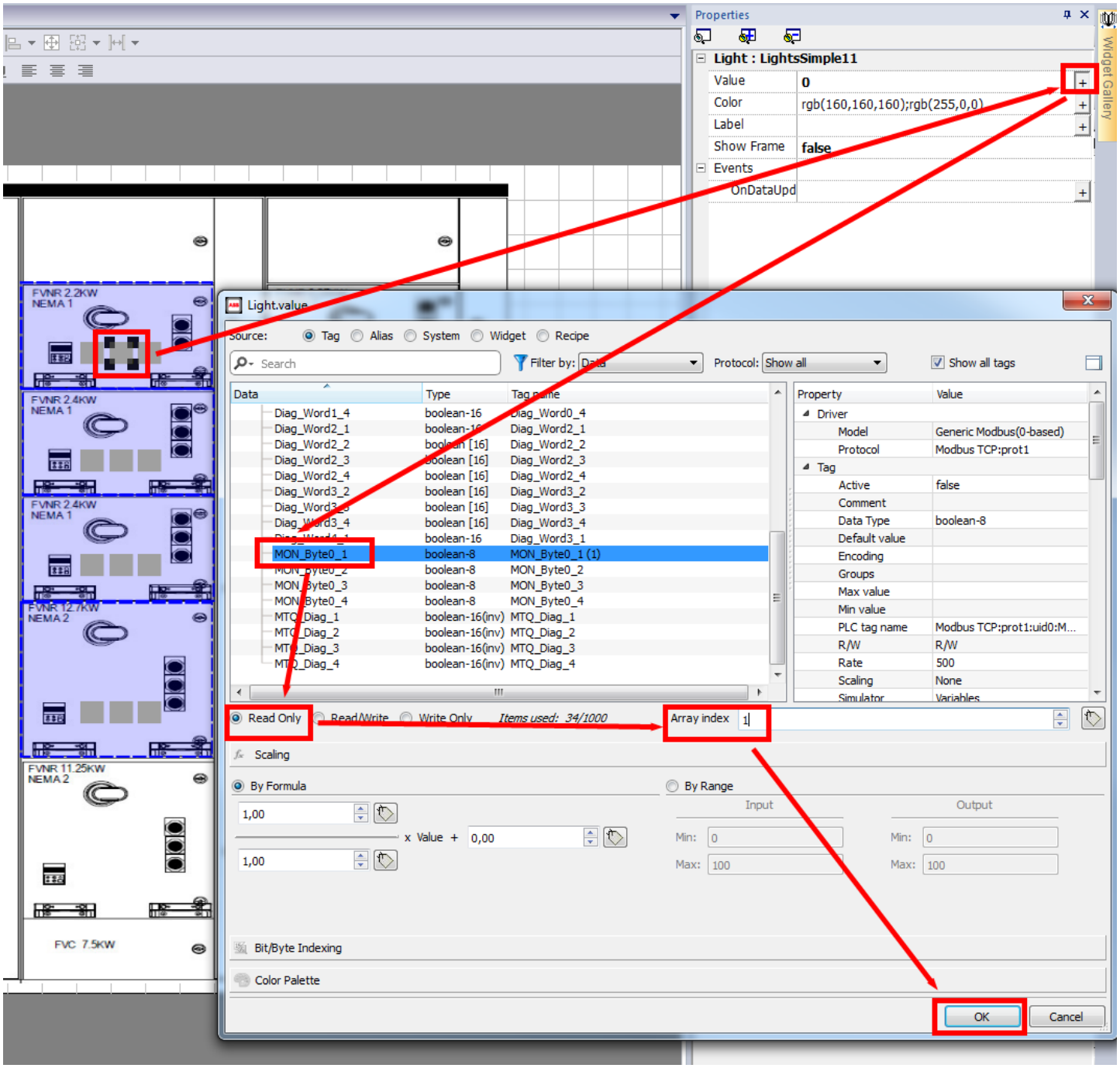
8. Select the second indicator

9. Click on "+" next to "Value"

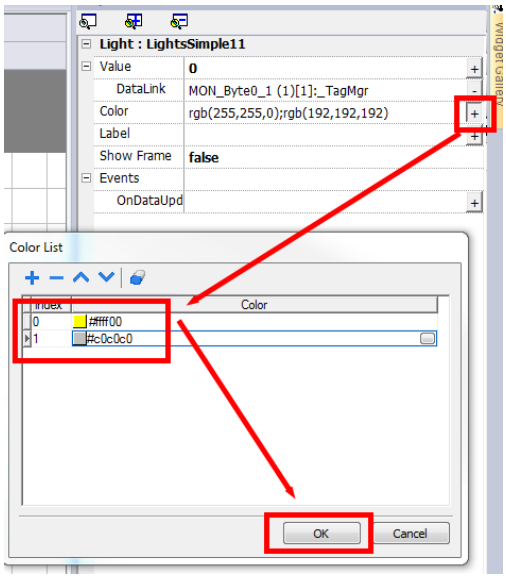
10. Select "MON_Byte0_1"

11. Set "Read only" and set the array index to "1"

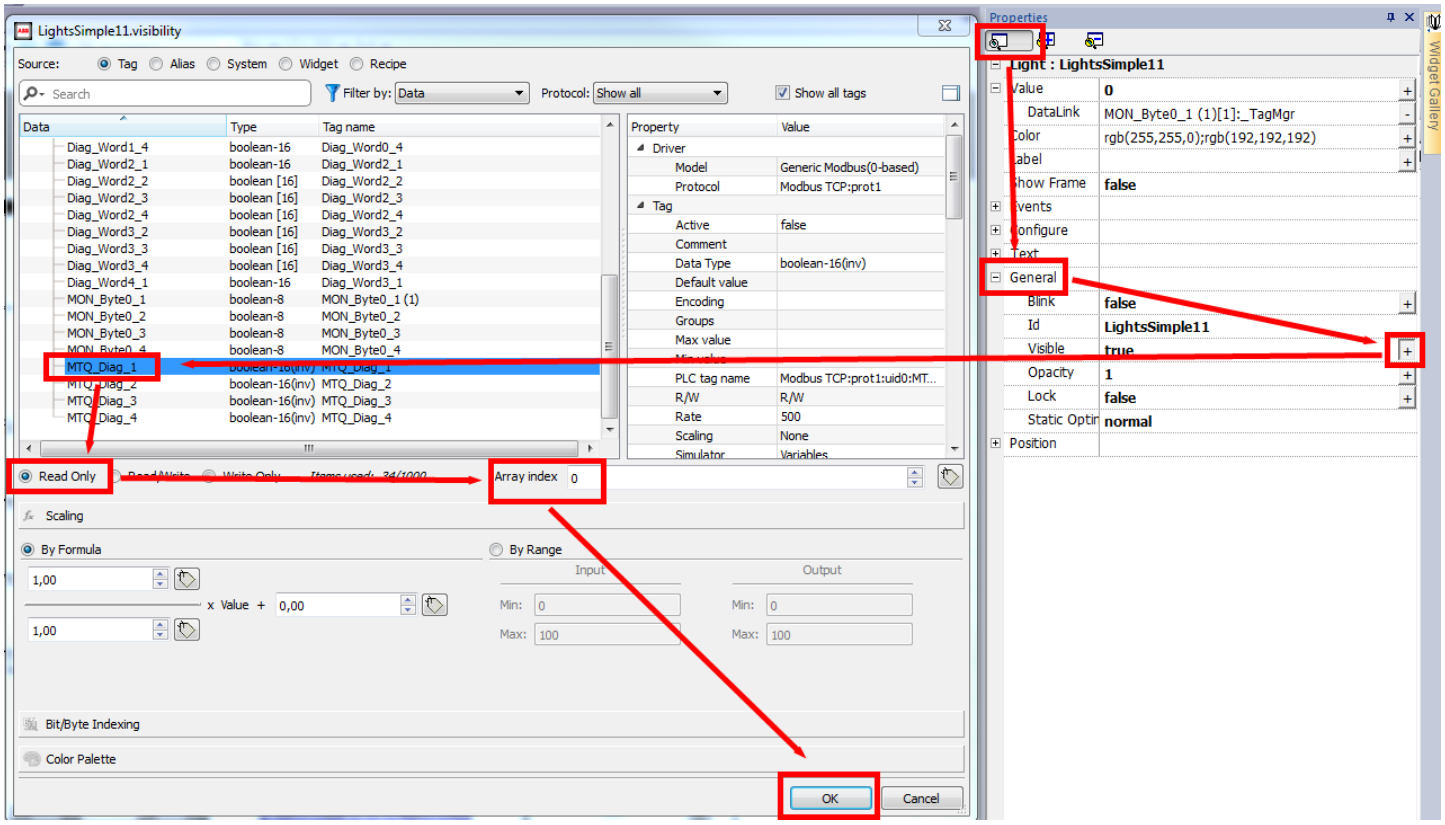
12. Confirm with "OK"



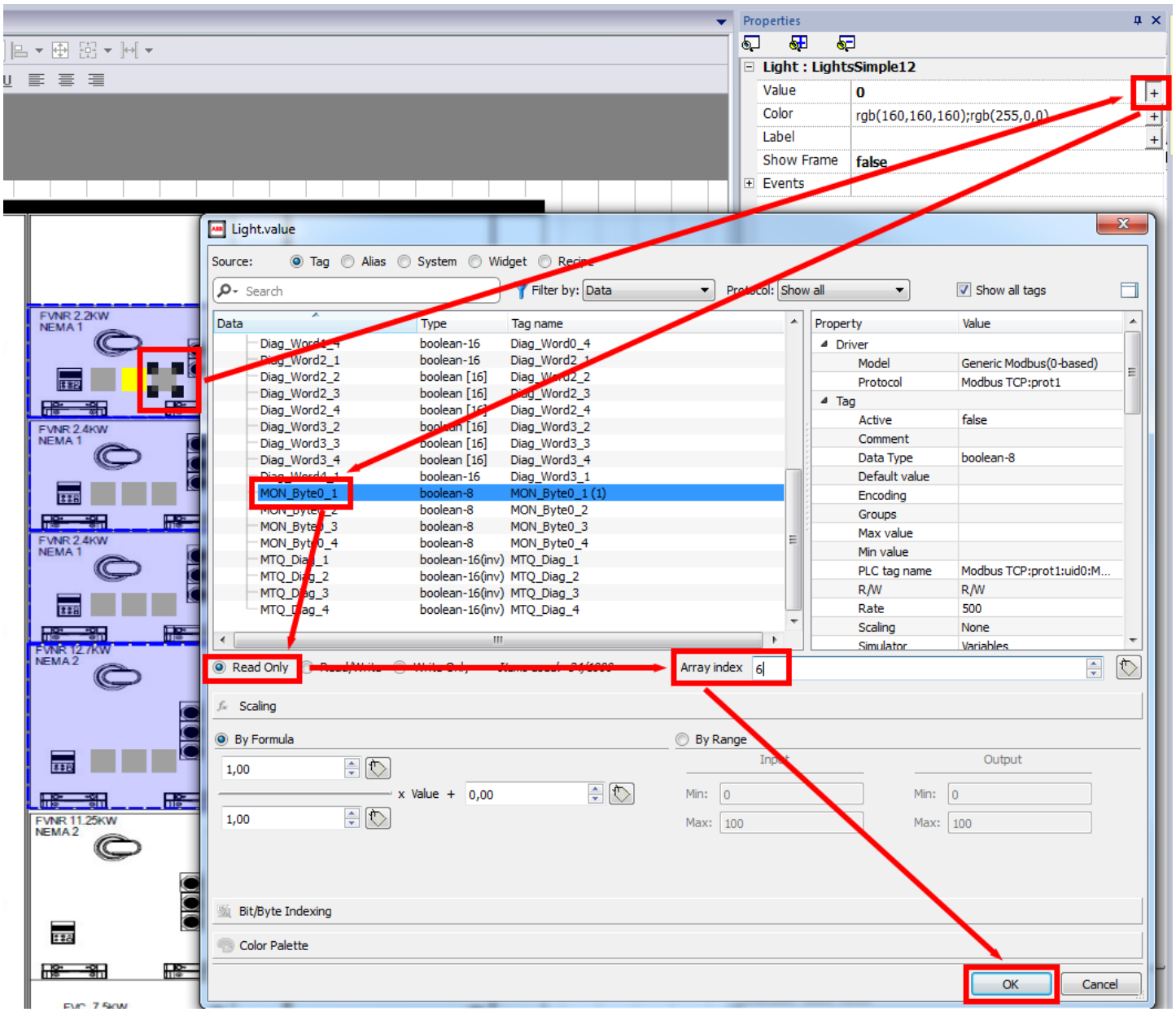
13. Click on “+” next to “Color”
14. Select yellow for index “0” and grey for “1”
15. Confirm with “OK”



16. Click on “Show Advanced Properties” and expand “General”
17. Click on “+” next to “Visible”
18. Select “MTQ_Diag_1”
19. Set “Read only” and array index to “0”
20. Confirm with “OK”



21. Select the third indicator
22. Click on “+” next to “Value”
23. Select “MON_Byte0_1” from the list
24. Select “Read only” and set index to “6”
25. Confirm with “OK”



Repeat the previous steps for each UMC / drawer.

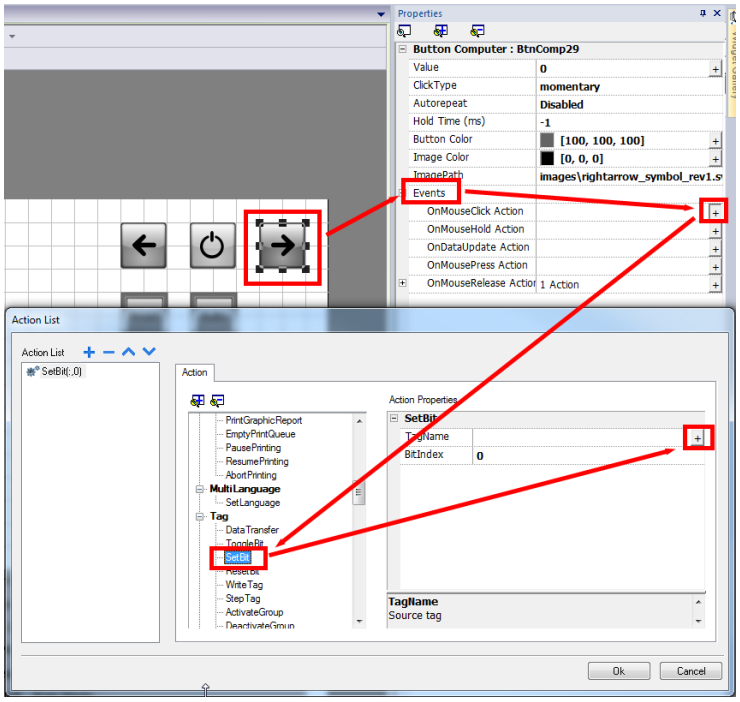
UMC page

After creating the layout and all tags, those have to be connected to send and get data over Modbus:

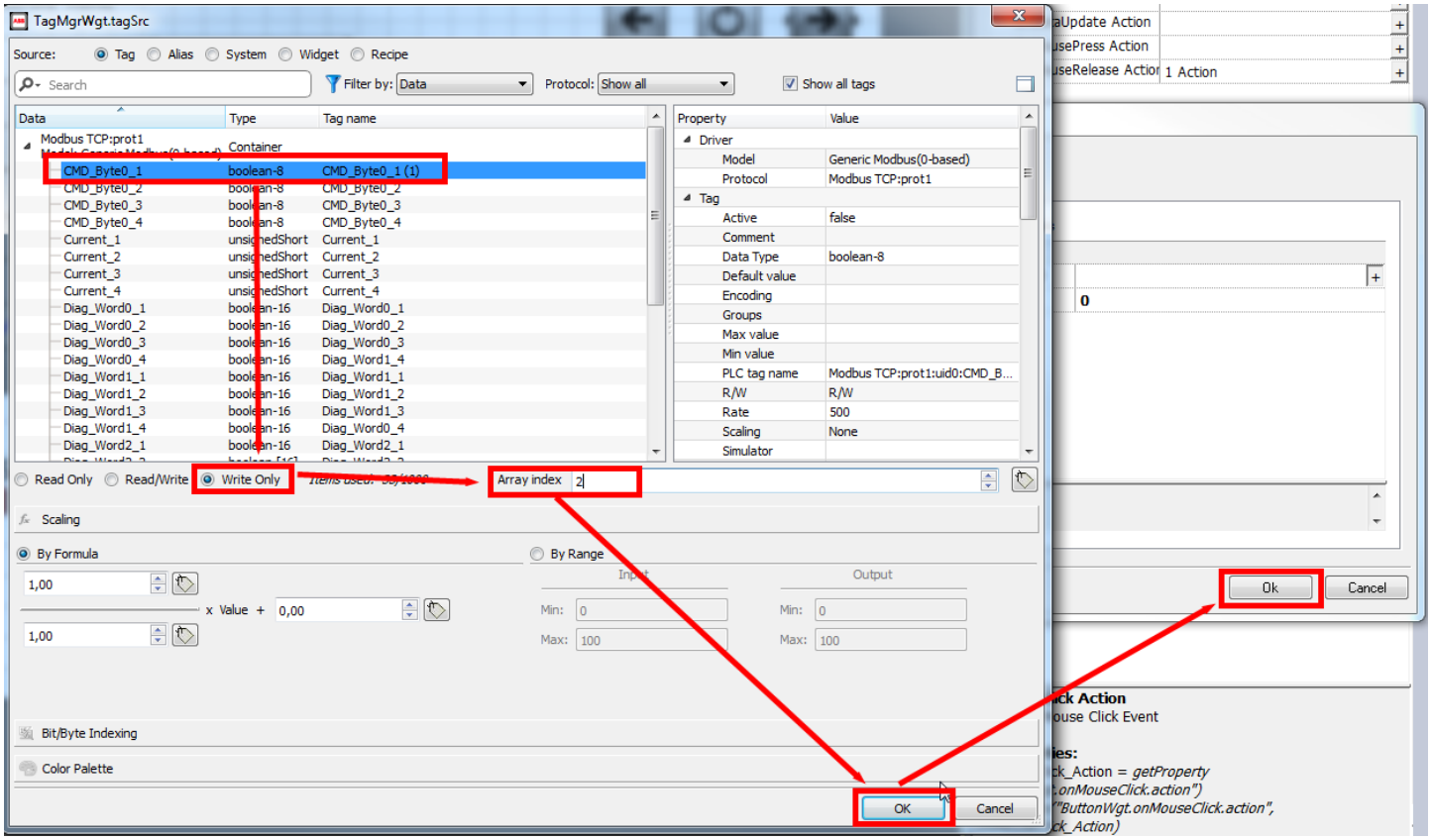
Buttons for motor control

For the motor control buttons two events have to be created. The first is for “OnMouseClick Action” and one for “OnMouseRelease Action”. That is necessary to only send a pulse for motor start and stop.

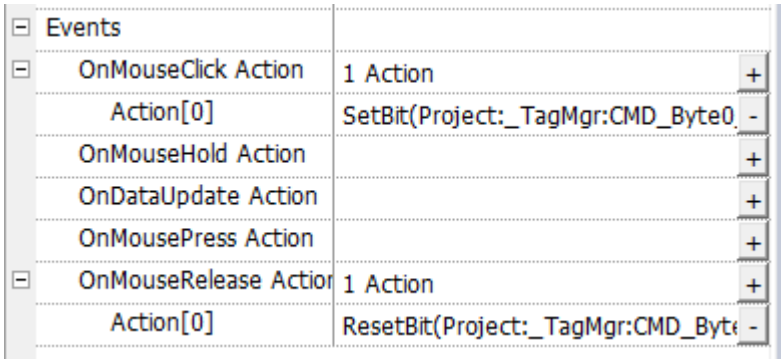
1. Select a push button and expand “Events”.
2. Click on the “+” next to “OnMouseClick Action”.
3. Select “SetBit” and click on “+” next to “TagName”.



4. For the selected “Run Forward” button select “CMD_Byte0_1” from the list.
5. Select “Write Only”.
6. For “Array index” select the bit position within the array. For “Run forward” this is index “2”.
7. Click on “OK” in this and the previous window.



8. Click on “+” next to “OnClickRelease Action”
9. Select “ResetBit” from the list and repeat the steps 4-7.



This procedure is the same for “Start reverse”, “Stop”, and “Reset” with a different array index.

For the “Auto” bit button the procedure is:

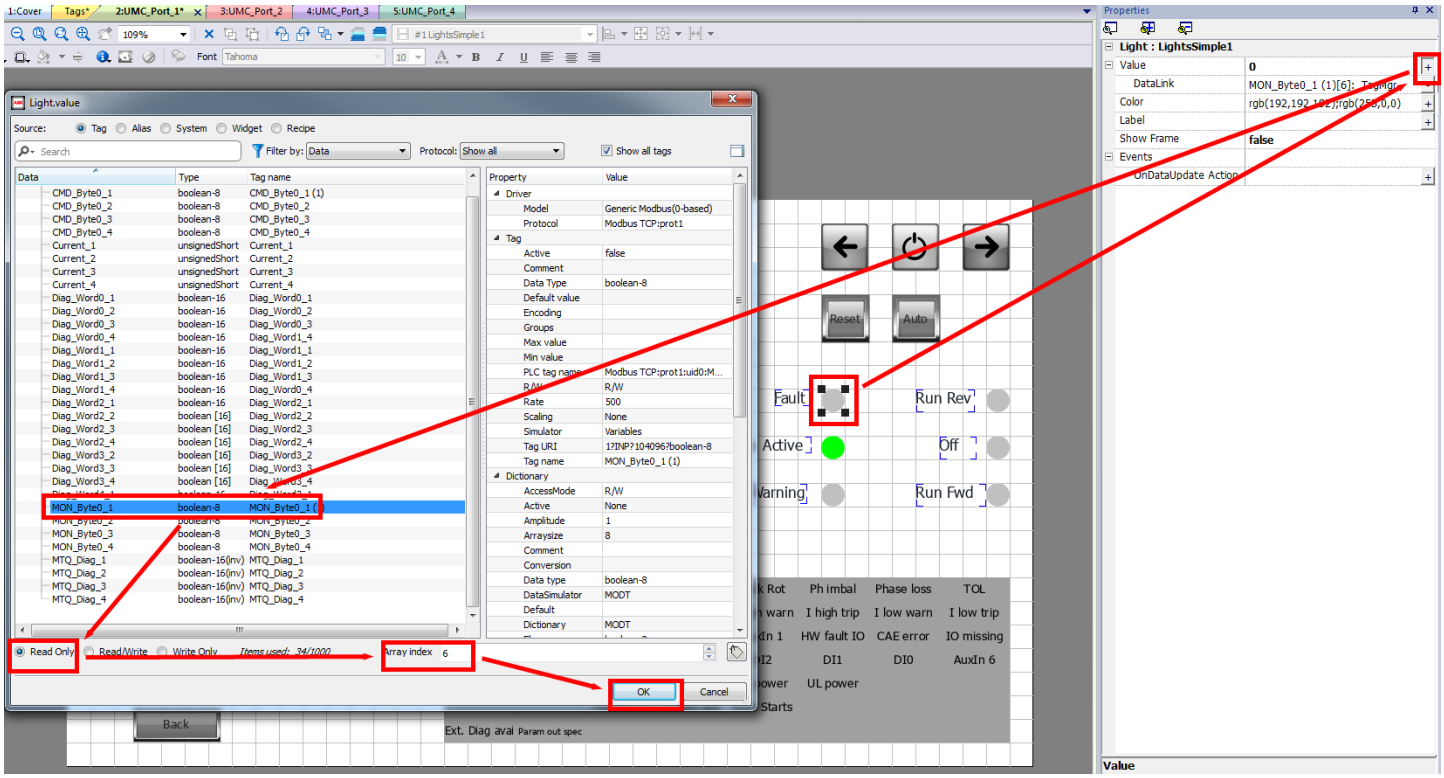
1. Select the button
2. Expand “Events” in the properties
3. Click on the “+” next to “OnMouseClick Action”

4. Select “ToggleBit” and click on “+” next to “TagName”
5. Select “CMD_Byte0_1” from the list.
6. Select “Read/Write”
7. Set the array index to “5”
8. Apply with “OK”

Indicators for UMC state

In total we have six indicators, showing the motor run state and the fault state. These are directly mapped to the monitoring telegram.

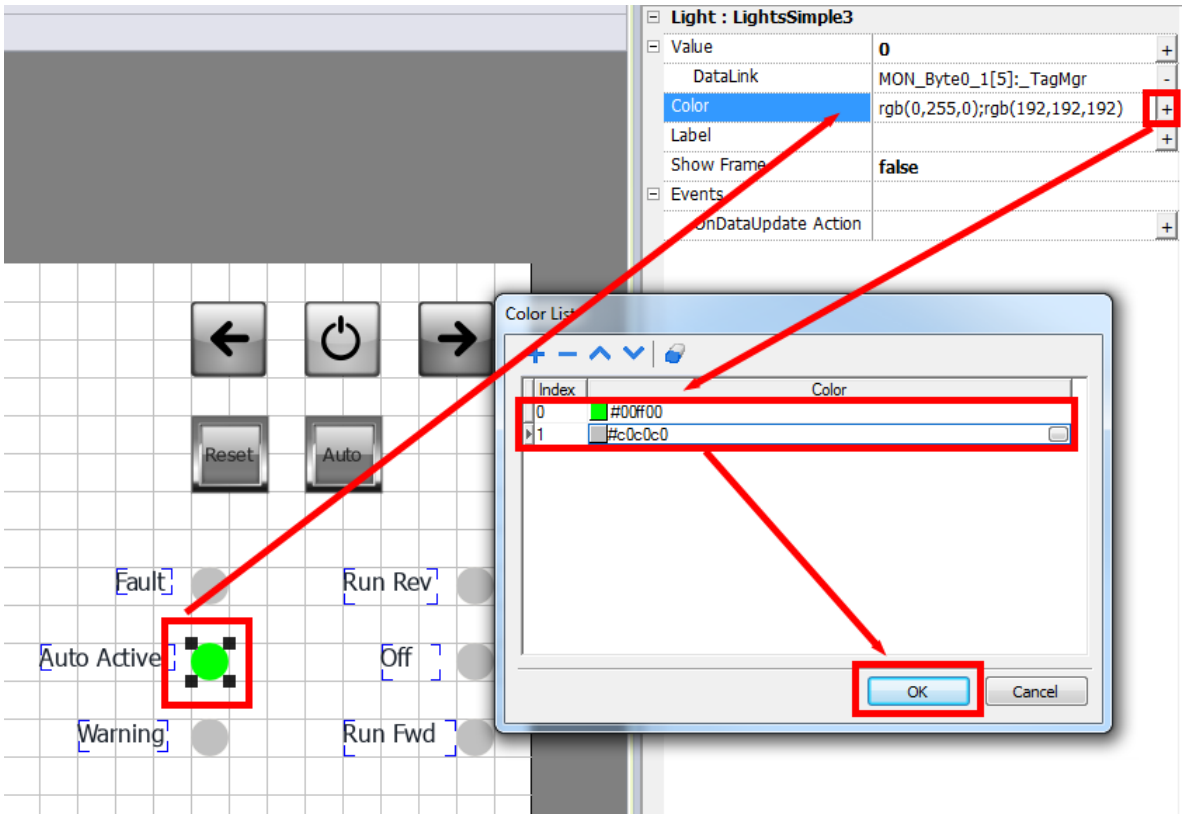
1. Select one indicator
2. Click on “+” next to “Value” in the properties
3. Select “MON_Byte0_1” from the list
4. Select “Read only”
5. Set the array index to its actual index (this example “Fault” is index “6”
6. Apply with “OK”



7. Repeat this for all indicators

For “Auto Active” it’s necessary to change the color of the LED because it shall be green in case the monitoring bit is zero. The bit transferred over the bus is the local state, so in case it’s “true” it means the UMC is controlled locally.

1. Click on the indicator for “Auto Active”
2. Click on “+” next to “Color”
3. Select green for “0” and grey for ”1”

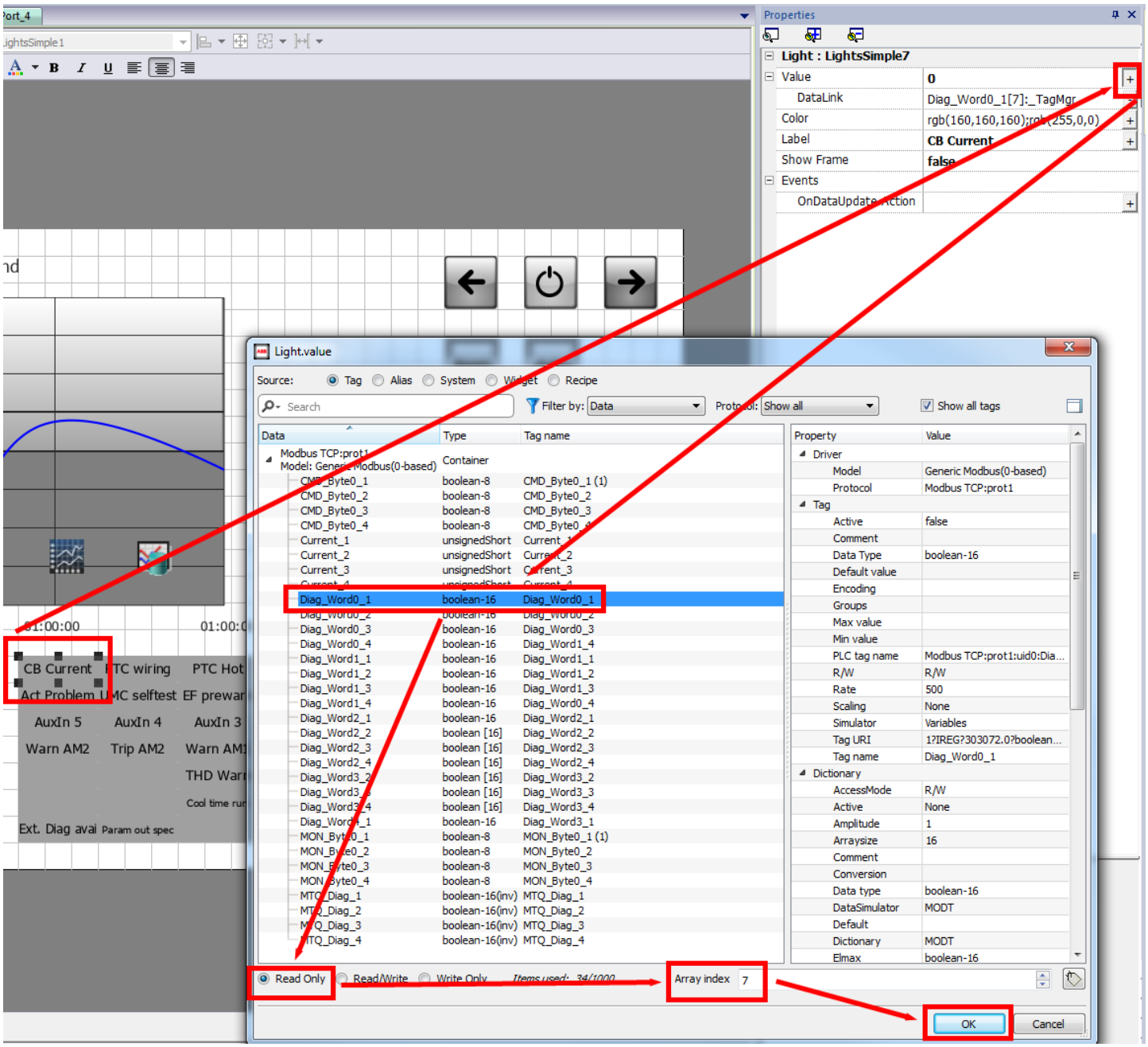


UMC Diagnostics

In the bottom right corner of the page layout the whole UMC diagnostic is displayed. The area consists of single lights labeled with the corresponding diagnostic message. For mapping the data the following example has to be used for each filled field.

1. Select one field
2. Click on “+”next to “Value” in the properties
3. Select the correct diagnostic word (for checkback current and port 1 it’s “Diag_Word0_1”)
4. Select “Read only” and for array index set the index (for checkback current it’s index “7”)

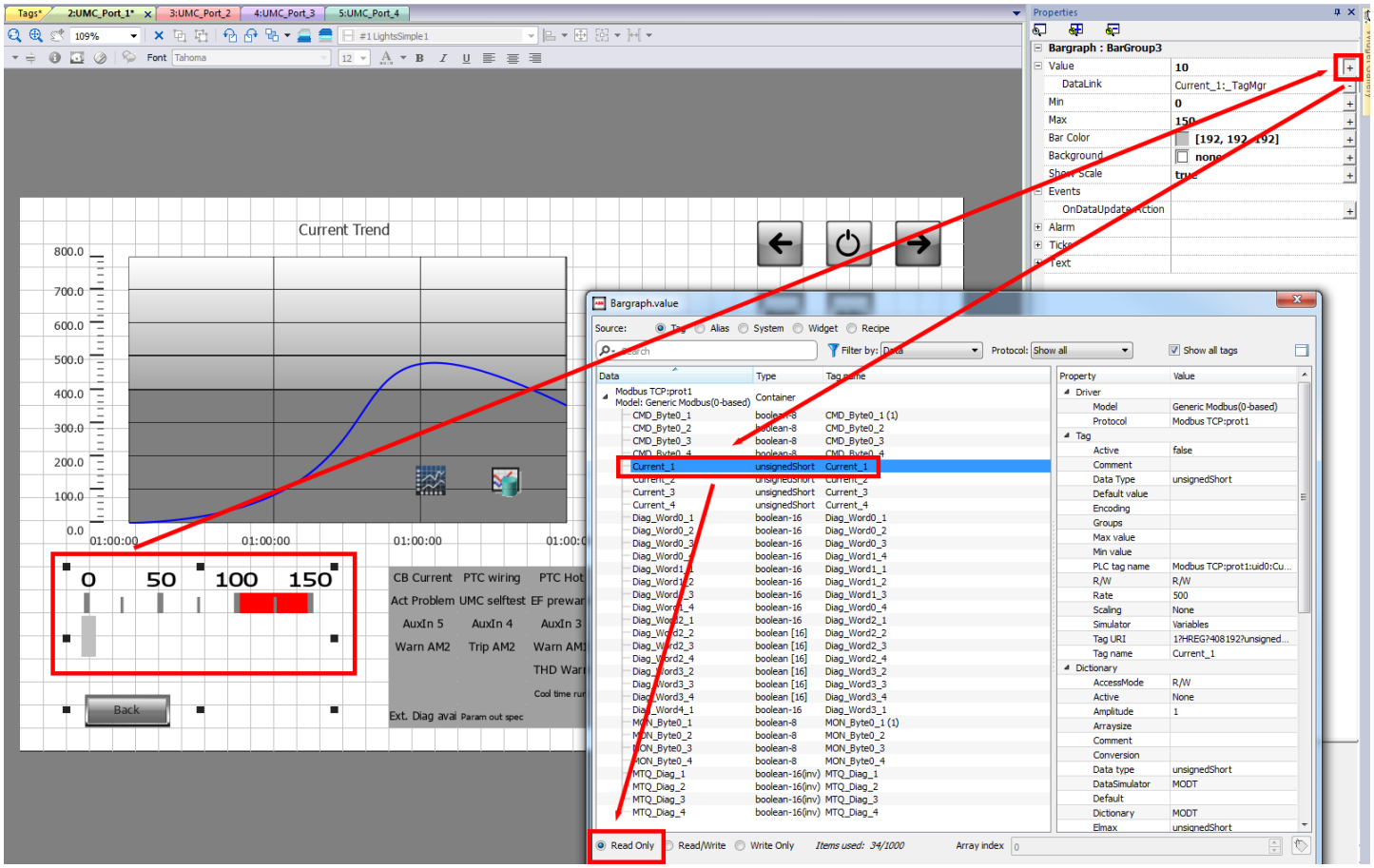
5. Apply changes with “OK”
6. Repeat the previous steps for each field with the correct combination of diagnostic word and array index



Bar graph for actual current

To display the actual current in a bar graph the following settings have to be done:

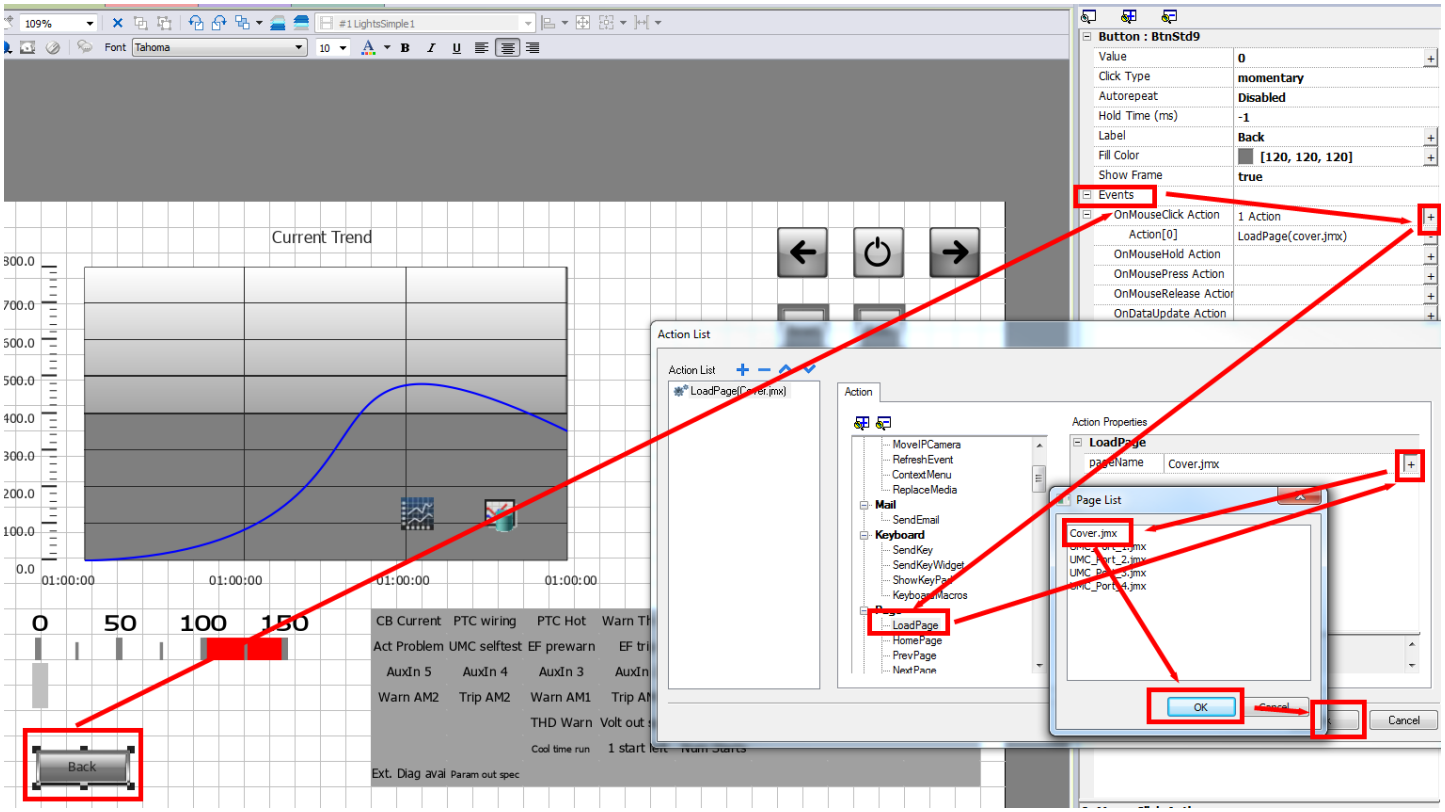
1. Click on the bar graph
2. Click on “+” next to “Value” in the properties
3. Select the current register from the list
4. Set “Read only” and apply with “OK”



5. Set other settings in the properties for “Min”, “Max” and “Bar Color”

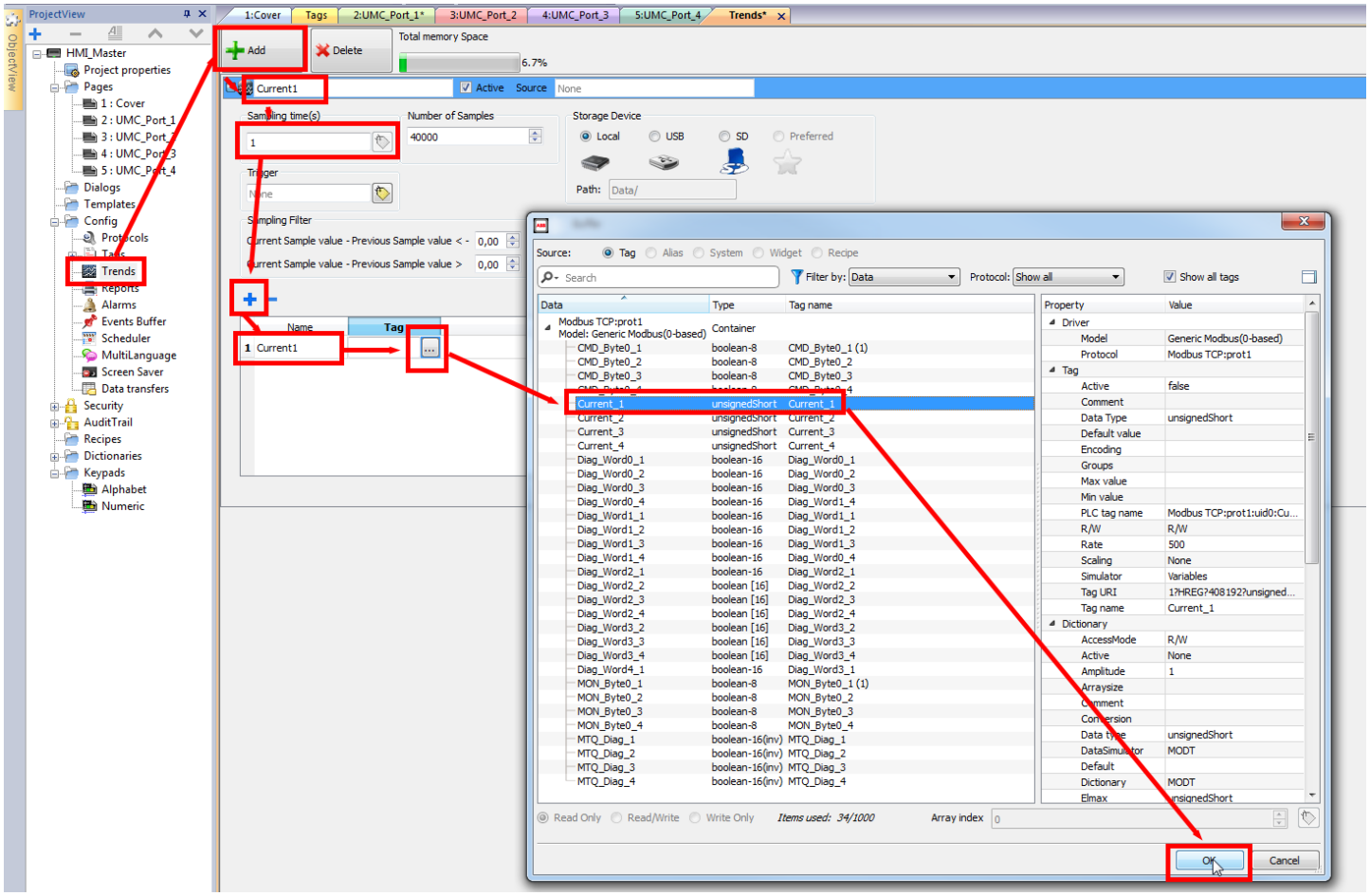
Connecting the “Back” button

1. Click on the “Back” button
2. Open “Events” in the properties
3. Click on “+” next to “OnMouseClick Action”
4. Select “LoadPage” and click on “+” next to “pageName”
5. Select the Cover page and apply both windows with “OK”

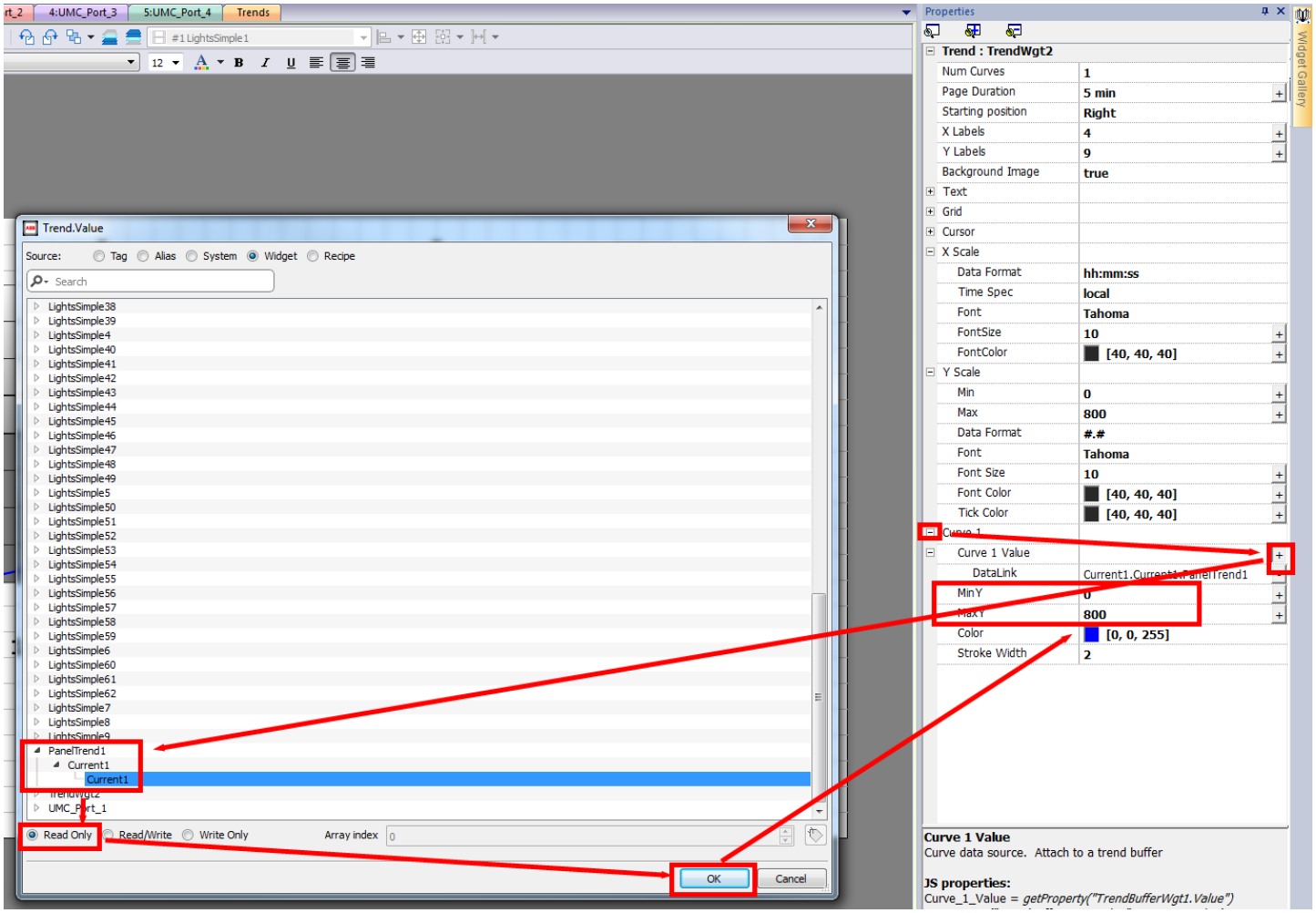


Current Trend

1. Open “Trends” in the “ProjectView”
2. Click on “Add”
3. Set a name (e.g. “Current1”)
4. Set the sampling time in seconds (e.g. 1s)
5. Click on “+”
6. Insert a name and click on the button in the “Tag” field
7. Select the Current tag from the list and click “OK”



8. Go on the UMC page and select the current trend
9. Set the properties for “Page Duration”, “X Scale” and “Y Scale”
10. Open “Curve 1”
11. Click on “+” next to “Curve 1 Value”
12. Select the trend (“PanelTrend1” → “Current1” → “Current1”)
13. Set “Read only” and apply with “OK”
14. Set “MinY” and “MaxY”



The same procedure has to be completed for each connected UMC.

After downloading everything to the panel, it behaves totally like a master and the user is able to give start and stop commands, to receive diagnostic information and so on.

Contact us

ABB STOTZ-KONTAKT GmbH
Eppelheimer Straße 82
69123 Heidelberg
Germany

You can find the address of your
local sales organization on the
ABB home page
http://www.abb.com/contacts
-> Low Voltage Products and Systems

Legal note
We reserve the right to make technical changes or modify the contents of this document without prior notice. With regard to purchase orders, the agreed particulars shall prevail. ABB AG does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

We reserve all rights in this document and in the subject matter and illustrations contained therein. Any reproduction, disclosure to third parties or utilization of its contents – in whole or in parts – is forbidden without prior written consent of ABB AG.

Copyright© 2017 ABB

All rights reserved