One device
for any room

Programmer MANUAL

UNIVERSAL SENSOR BASED TOUCH PANEL TP10/RA

MODBUS™ INTERFACE

Programmer_UPZIO_0001_TP10_ MODBUS

Index

Integration in Twincat project

Summary

Setting up modbus communication between an Upzio device and a Beckhoff PLC is very easy as all required function blocks are already written.

This document is provided to help people implement the upzio modbus devices into their own TwinCAT2 and TwinCAT3 projects. If required, you can visit our site, www.upzio.com.

The example project that will be created in this chapter is available on our website.

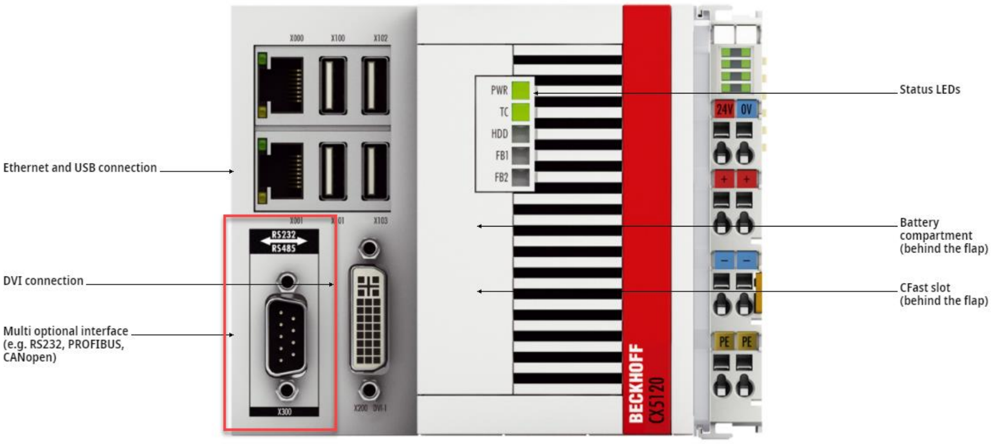
Integration in a Twincat2 project

Short guide to implementing upzio modbus devices into Twincat 2

- Step 1: Use RS485 hardware. The EL6021 or a PLC with an integrated serial com port can be used.
- Step 2: The RS485 hardware must be wired in a way that enables half-duplex communication.
- Step 3: Download all required files and import and add these to your project.
- Step 4: Change settings in your project according to your hardware.
- Step 5: Execute P_ModbusMain() somewhere in your project.
- Step 6: Create instances of the TP10 and execute somewhere.
- Step 7: Set up the system manager.

Step1: pick RS485 hardware

To implement modbus RTU on a Beckhoff PLC, RS485 hardware must be used. To use the serial com port of the PLC, you will need an RS485 connector. (e.g., Subcon 9/M-SH from phoenix contact). If your PLC does not have a serial com port or you need more than one modbus master, the EL6021 can be used instead of the serial com port.

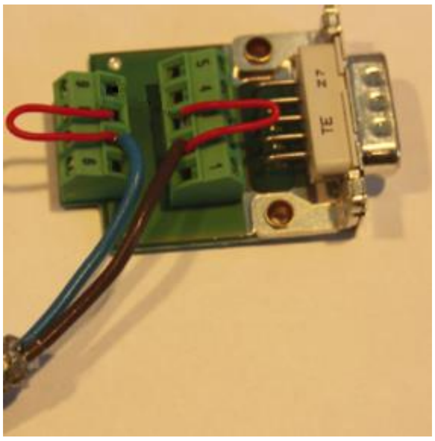

Ethernet and USB connection
DVI connection
Multi optional interface (e.g. RS232, PROFIBUS, CANopen)
Status LEDs
Battery compartment (behind the flap)
CFast slot (behind the flap)
X000
X100
X102
X001
X101
X103
RS232
RS485
X300
X200 DVI-I
PWR
TC
HDD
FB1
FB2
24V
0V
PE
BECKHOFF
CX5120

CX5120 | Embedded PC with Intel Atom® processor

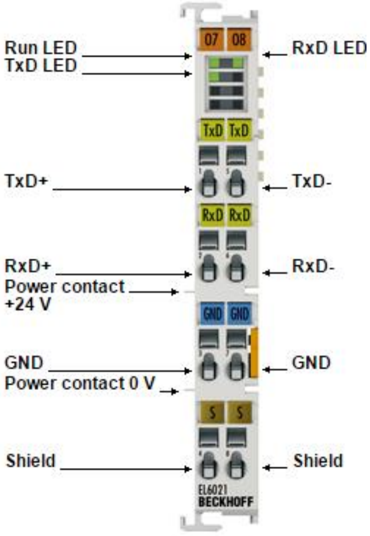
Step2: Wiring the hardware

If the serial com port of the PLC is used with a Subcon 9/M-SH from phoenix contact, you will have to wire the RS485 connector as shown in the pictures below to enable half duplex communication.

The Rx wire must be connected to terminal 2 and 3. The Tx wire must be connected to terminal 7 and 8.



If the EL6021 is used, TxD+ and RxD+ must be connected to each other and to the B-wire. TxD- and RxD- must be connected to each other and to the A-wire.



Top view

Step 3: Download and import

For an easy implementation, the necessary libraries and export files must be downloaded and imported into your project. (see https://www.upzio.com/support)

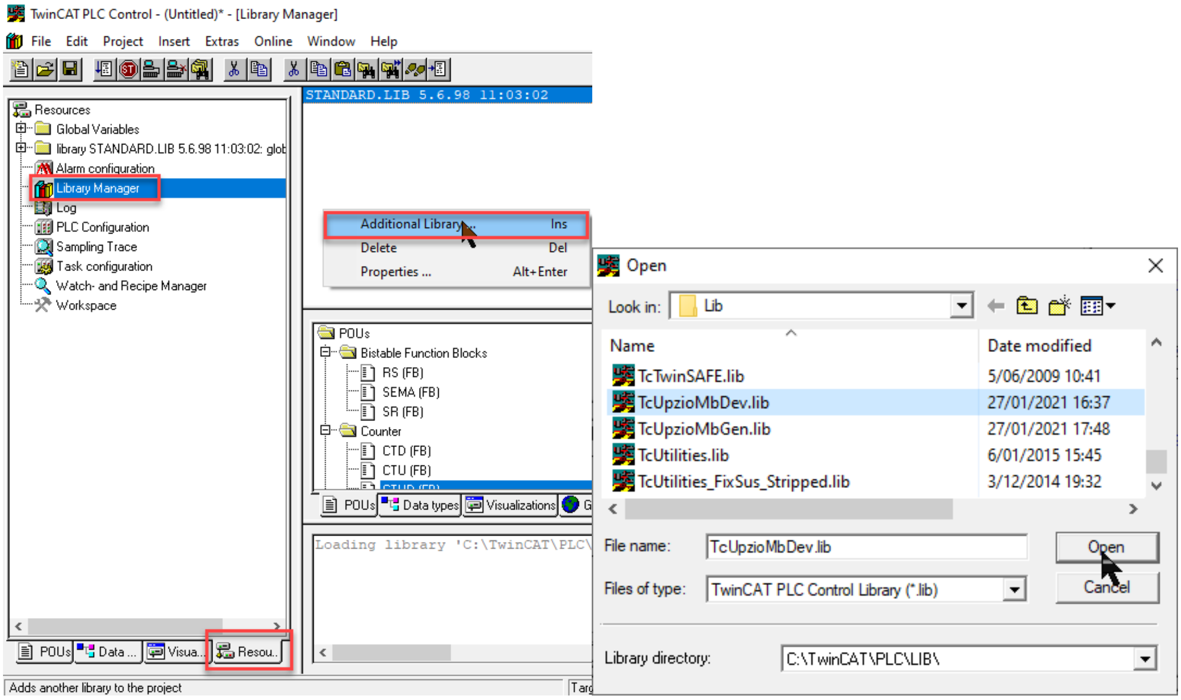
There are 3 libraries required for the modbus devices:

1. ModbusRTU_Upzio.lib , this library works just like the Beckhoff ModbusRTU.lib library, but is a little faster.
2. TcUpzioMbGen.lib , this is the general Upzio modbus library that controls the message buffers of every Modbus master.
3. TcUpzioMbDev.lib , this is the device Upzio modbus library. This library contains all Upzio modbus devices.

There is 1 export file required for the modbus devices:

1. ExampleSettings.EXP , this export file contains a program ‘P_ModbusMain’ and a global variable list ‘Global_ModbusSettings’. Both should be imported in the twincat project.

To import the libraries, put all library files in your twincat library folder (default C:\TwinCAT\Plc\Lib). Then add the libraries to your project by going to resources – library manager -right click -Additional Library… Then select the TcUpzioMbDev.lib library. This library will automatically import the other libraries


TwinCAT PLC Control - (Untitled)* - [Library Manager]

STANDARD.LIB 5.6.98 11:03:02

Loading library 'C:\TwinCAT\PLC\

Adds another library to the project

Open

Look in: Lib

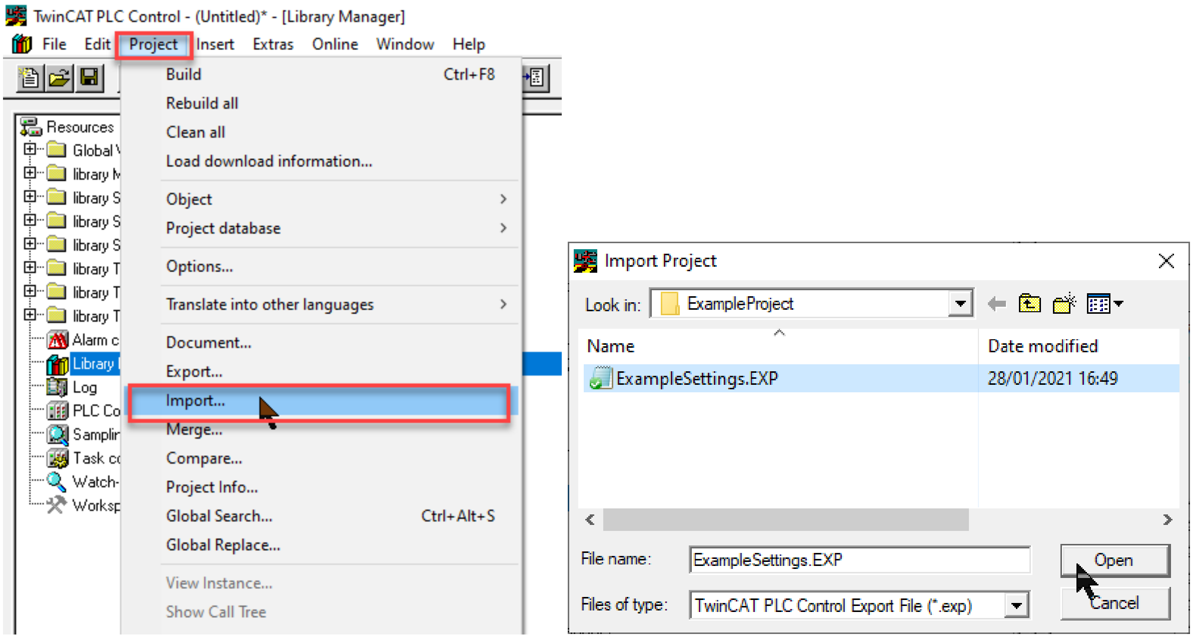
Name	Date modified
TcTwinSAFE.lib	5/06/2009 10:41
TcUpzioMbDev.lib	27/01/2021 16:37
TcUpzioMbGen.lib	27/01/2021 17:48
TcUtilities.lib	6/01/2015 15:45
TcUtilities_FixSus_Stripped.lib	3/12/2014 19:32

File name: TcUpzioMbDev.lib

Files of type: TwinCAT PLC Control Library (*.lib)

Library directory: C:\TwinCAT\PLC\LIB\

Next the export file must be imported into the twincat project. Go to Project – Import… and select the ExampleSettings.Exp file.



Look in: ExampleProject

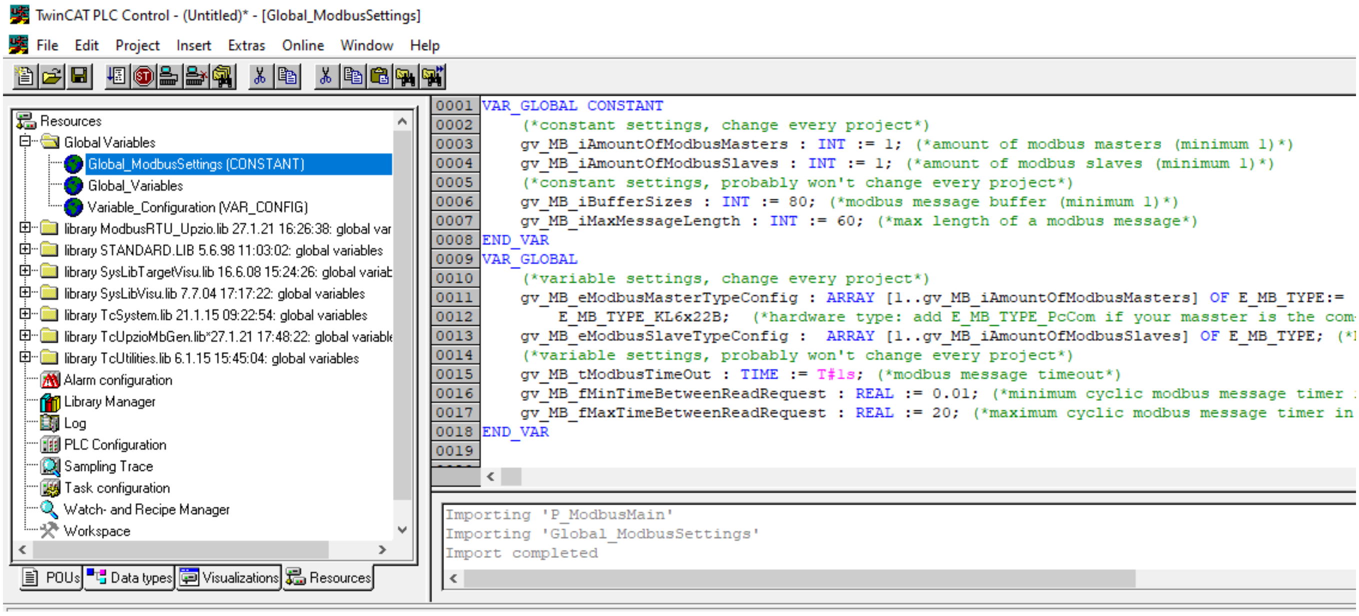
ExampleSettings.EXP 28/01/2021 16:49

File name: ExampleSettings.EXP

Files of type: TwinCAT PLC Control Export File (*.exp)

Step 4: Change settings according to your hardware

The ExampleSettings.Exp contains a global variable list 'Global_ModbusSettings' where the modbus settings can be altered. The most important setting is 'gv_MB_eModbusMasterTypeConfig'. gv_MB_eModbusMasterTypeConfig you must define what kind of hardware you will be using. Eg. If your first modbus master is the serial com port, gv_MB_eModbusMasterTypeConfig[1] must be E_MB_TYPE_PcCom. If your second modbus master is an EL6021, gv_MB_eModbusMasterTypeConfig[1] must be E_MB_TYPE_KL6x22B.

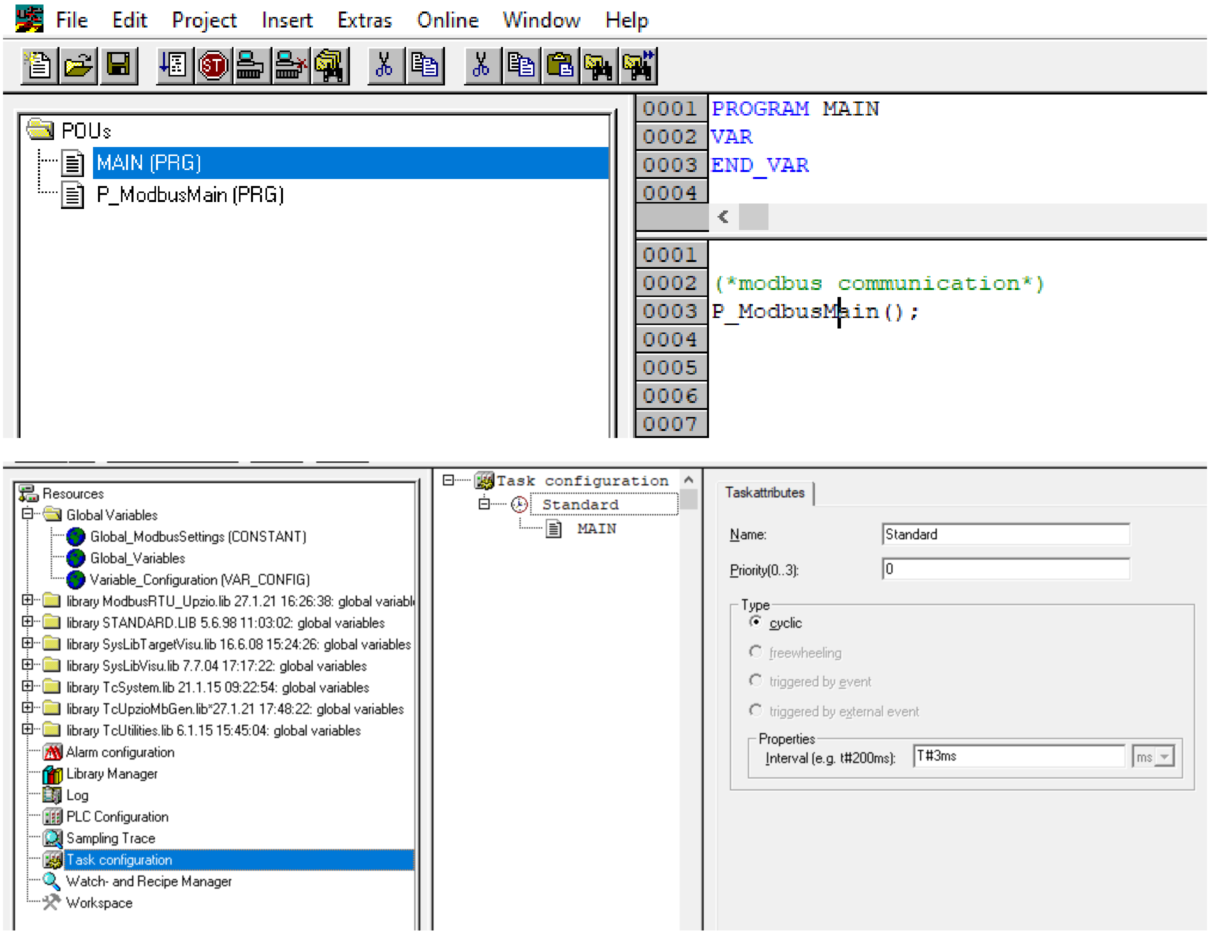

TwinCAT PLC Control - (Untitled)* - [Global_ModbusSettings]
File Edit Project Insert Extras Online Window Help
Resources
Global Variables
Global_ModbusSettings (CONSTANT)
Global_Variables
Variable_Configuration (VAR_CONFIG)
library ModbusRTU_Upzio.lib 27.1.21 16:26:38: global var
library STANDARD.LIB 5.6.98 11:03:02: global variables
library SysLibTargetVisu.lib 16.6.08 15:24:26: global variat
library SysLibVisu.lib 7.7.04 17:17:22: global variables
library TcSystem.lib 21.1.15 09:22:54: global variables
library TcUpzioMbGen.lib*27.1.21 17:48:22: global variable
library TcUtilities.lib 6.1.15 15:45:04: global variables
Alarm configuration
Library Manager
Log
PLC Configuration
Sampling Trace
Task configuration
Watch- and Recipe Manager
Workspace
POUs
Data types
Visualizations
Resources

```
VAR_GLOBAL CONSTANT
    (*constant settings, change every project*)
    gv_MB_iAmountOfModbusMasters : INT := 1; (*amount of modbus masters (minimum 1)*)
    gv_MB_iAmountOfModbusSlaves : INT := 1; (*amount of modbus slaves (minimum 1)*)
    (*constant settings, probably won't change every project*)
    gv_MB_iBufferSizes : INT := 80; (*modbus message buffer (minimum 1)*)
    gv_MB_iMaxMessageLength : INT := 60; (*max length of a modbus message*)
END_VAR
VAR_GLOBAL
    (*variable settings, change every project*)
    gv_MB_eModbusMasterTypeConfig : ARRAY [1..gv_MB_iAmountOfModbusMasters] OF E_MB_TYPE:=
        E_MB_TYPE_KL6x22B;  (*hardware type: add E_MB_TYPE_PcCom if your masster is the com
    gv_MB_eModbusSlaveTypeConfig :  ARRAY [1..gv_MB_iAmountOfModbusSlaves] OF E_MB_TYPE; (*
    (*variable settings, probably won't change every project*)
    gv_MB_tModbusTimeOut : TIME := T#1s; (*modbus message timeout*)
    gv_MB_fMinTimeBetweenReadRequest : REAL := 0.01; (*minimum cyclic modbus message timer
    gv_MB_fMaxTimeBetweenReadRequest : REAL := 20; (*maximum cyclic modbus message timer in
END_VAR
```

```
Importing 'P_ModbusMain'
Importing 'Global_ModbusSettings'
Import completed
```

Step 5: Execute P_ModbusMain()

The export file contained a program 'P_ModbusMain'. This program should be executed somewhere in your project. This program copies all global modbus settings to the Modbus masters and handles all modbus message buffers. The modbus communication will work wherever you execute this program. But the modbus communication will be faster if this program is executed in a task with a lower cycle time. Therefore, it is recommended to put this program in a task with a low cycle time (eg. 3ms).



POUs
- MAIN (PRG)
- P_ModbusMain (PRG)

```
PROGRAM MAIN
VAR
END_VAR
```

```

(*modbus communication*)
P_ModbusMain();
```

Resources
- Global Variables
  - Global_ModbusSettings (CONSTANT)
  - Global_Variables
  - Variable_Configuration (VAR_CONFIG)
- library ModbusRTU_Upzio.lib 27.1.21 16:26:38: global variabl
- library STANDARD.LIB 5.6.98 11:03:02: global variables
- library SysLibTargetVisu.lib 16.6.08 15:24:26: global variables
- library SysLibVisu.lib 7.7.04 17:17:22: global variables
- library TcSystem.lib 21.1.15 09:22:54: global variables
- library TcUpzioMbGen.lib*27.1.21 17:48:22: global variables
- library TcUtilities.lib 6.1.15 15:45:04: global variables
- Alarm configuration
- Library Manager
- Log
- PLC Configuration
- Sampling Trace
- Task configuration
- Watch- and Recipe Manager
- Workspace

Task configuration
- Standard
  - MAIN

Taskattributes

Name: Standard

Priority(0..3): 0

Type
- ● cyclic
- ○ freewheeling
- ○ triggered by event
- ○ triggered by external event

Properties

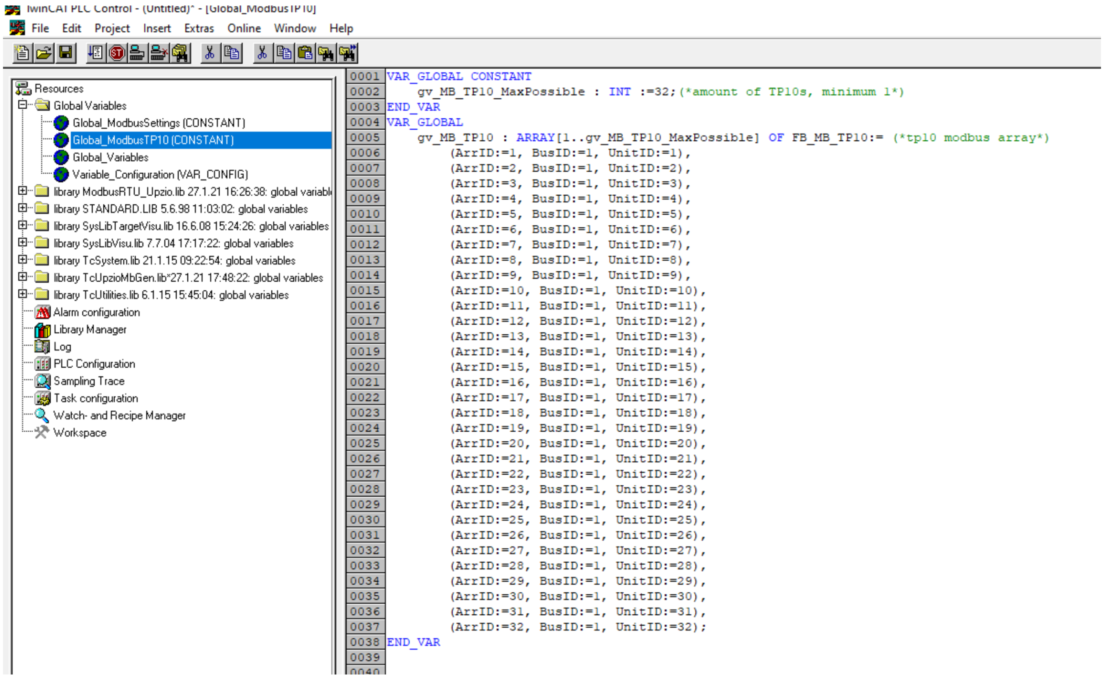
Interval (e.g. t#200ms): T#3ms ms

Step 6: Create instances of the TP10 and execute

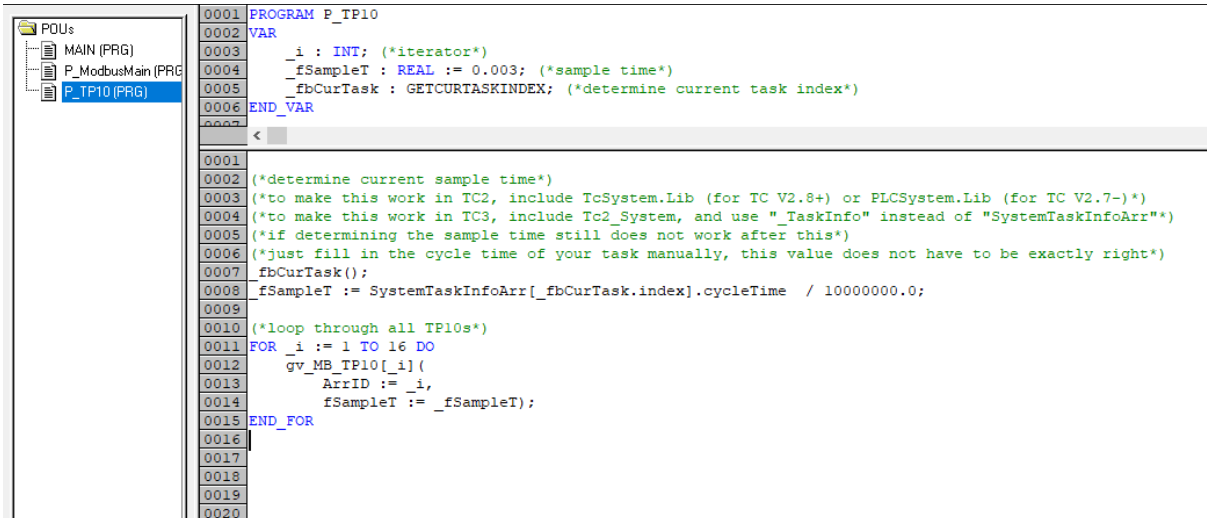
In the example below a global array of 32 TP10s is created. BusID is the Modbus master to which the TP10s are physically wired. UnitID is the modbus address of the TP10s.

The instances of the TP10s must be executed somewhere in your program. In the example below, the TP10s are all executed in the same task as P_ModbusMain, but the TP10s might just as well be executed somewhere else.

The TP10 function blocks need to know the cycle time to time their modbus messages. The example below determines this cycle time automatically.

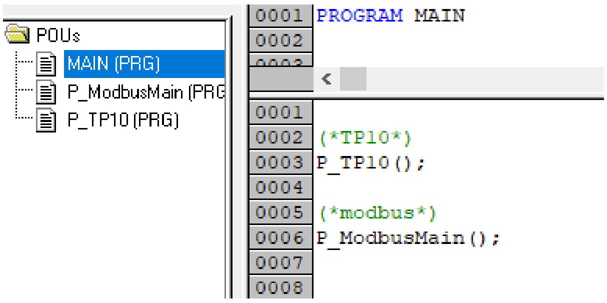

TwinCAT PLC Control - (Untitled)* - [Global_ModbusTP10]
File
Edit
Project
Insert
Extras
Online
Window
Help
Resources
Global Variables
Global_ModbusSettings (CONSTANT)
Global_ModbusTP10 (CONSTANT)
Global_Variables
Variable_Configuration (VAR_CONFIG)
library ModbusRTU_Upzio.lib 27.1.21 16:26:38: global variabl
library STANDARD.LIB 5.6.98 11:03:02: global variables
library SysLibTargetVisu.lib 16.6.08 15:24:26: global variables
library SysLibVisu.lib 7.7.04 17:17:22: global variables
library TcSystem.lib 21.1.15 09:22:54: global variables
library TcUpzioMbGen.lib*27.1.21 17:48:22: global variables
library TcUtilities.lib 6.1.15 15:45:04: global variables
Alarm configuration
Library Manager
Log
PLC Configuration
Sampling Trace
Task configuration
Watch- and Recipe Manager
Workspace

```
VAR_GLOBAL CONSTANT
    gv_MB_TP10_MaxPossible : INT :=32;(*amount of TP10s, minimum 1*)
END_VAR
VAR_GLOBAL
    gv_MB_TP10 : ARRAY[1..gv_MB_TP10_MaxPossible] OF FB_MB_TP10:= (*tp10 modbus array*)
        (ArrID:=1, BusID:=1, UnitID:=1),
        (ArrID:=2, BusID:=1, UnitID:=2),
        (ArrID:=3, BusID:=1, UnitID:=3),
        (ArrID:=4, BusID:=1, UnitID:=4),
        (ArrID:=5, BusID:=1, UnitID:=5),
        (ArrID:=6, BusID:=1, UnitID:=6),
        (ArrID:=7, BusID:=1, UnitID:=7),
        (ArrID:=8, BusID:=1, UnitID:=8),
        (ArrID:=9, BusID:=1, UnitID:=9),
        (ArrID:=10, BusID:=1, UnitID:=10),
        (ArrID:=11, BusID:=1, UnitID:=11),
        (ArrID:=12, BusID:=1, UnitID:=12),
        (ArrID:=13, BusID:=1, UnitID:=13),
        (ArrID:=14, BusID:=1, UnitID:=14),
        (ArrID:=15, BusID:=1, UnitID:=15),
        (ArrID:=16, BusID:=1, UnitID:=16),
        (ArrID:=17, BusID:=1, UnitID:=17),
        (ArrID:=18, BusID:=1, UnitID:=18),
        (ArrID:=19, BusID:=1, UnitID:=19),
        (ArrID:=20, BusID:=1, UnitID:=20),
        (ArrID:=21, BusID:=1, UnitID:=21),
        (ArrID:=22, BusID:=1, UnitID:=22),
        (ArrID:=23, BusID:=1, UnitID:=23),
        (ArrID:=24, BusID:=1, UnitID:=24),
        (ArrID:=25, BusID:=1, UnitID:=25),
        (ArrID:=26, BusID:=1, UnitID:=26),
        (ArrID:=27, BusID:=1, UnitID:=27),
        (ArrID:=28, BusID:=1, UnitID:=28),
        (ArrID:=29, BusID:=1, UnitID:=29),
        (ArrID:=30, BusID:=1, UnitID:=30),
        (ArrID:=31, BusID:=1, UnitID:=31),
        (ArrID:=32, BusID:=1, UnitID:=32);
END_VAR
```



```
PROGRAM P_TP10
VAR
    _i : INT; (*iterator*)
    _fSampleT : REAL := 0.003; (*sample time*)
    _fbCurTask : GETCURTASKINDEX; (*determine current task index*)
END_VAR
```

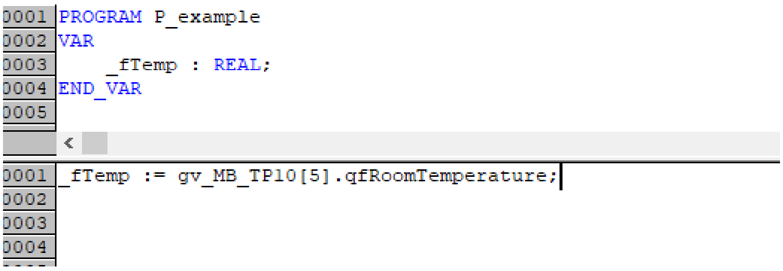
```

(*determine current sample time*)
(*to make this work in TC2, include TcSystem.Lib (for TC V2.8+) or PLCSystem.Lib (for TC V2.7-)*)
(*to make this work in TC3, include Tc2_System, and use "_TaskInfo" instead of "SystemTaskInfoArr"*)
(*if determining the sample time still does not work after this*)
(*just fill in the cycle time of your task manually, this value does not have to be exactly right*)
_fbCurTask();
_fSampleT := SystemTaskInfoArr[_fbCurTask.index].cycleTime  / 10000000.0;

(*loop through all TP10s*)
FOR _i := 1 TO 16 DO
    gv_MB_TP10[_i](
        ArrID := _i,
        fSampleT := _fSampleT);
END_FOR
```



```
PROGRAM MAIN
```

```

(*TP10*)
P_TP10();

(*modbus*)
P_ModbusMain();
```

After doing this, all TP10 functionality will be available everywhere in your project. The example below stores the room temperature measured by TP10 5 to a local variable:

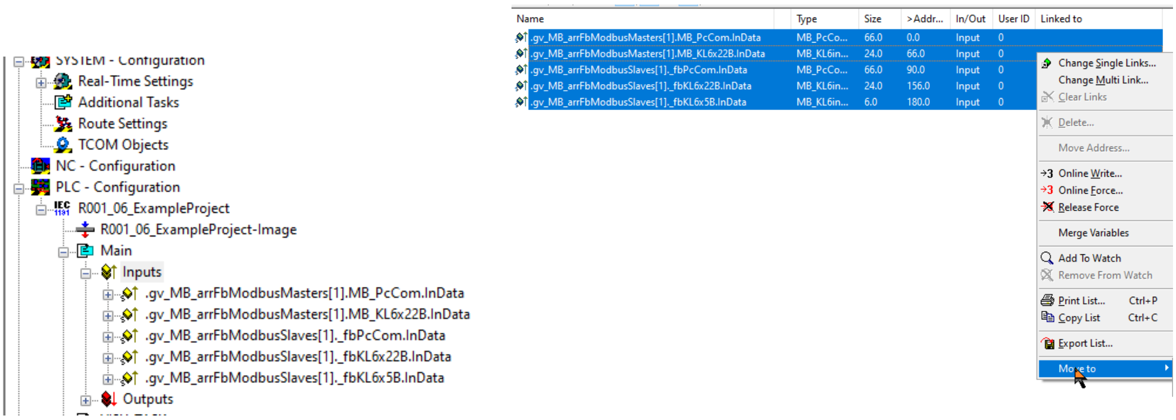


```
PROGRAM P_example
VAR
	_fTemp : REAL;
END_VAR

_fTemp := gv_MB_TP10[5].qfRoomTemperature;
```

Step 7: Setting up the system manager

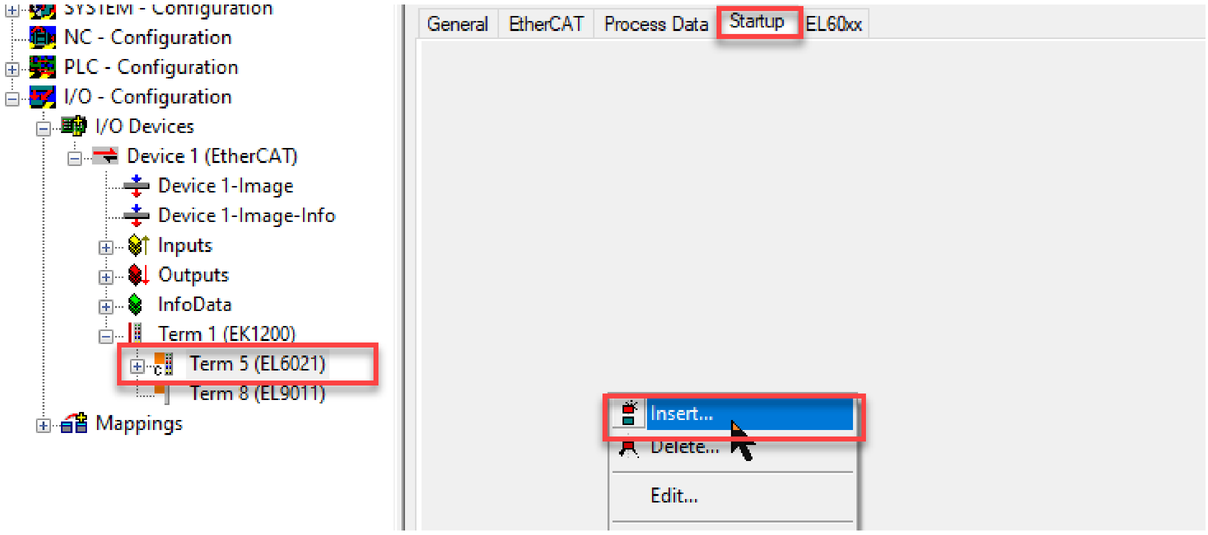
The first step in the system manager is to link your plc project and move your I/O to the right cycle. It may also help to turn on "I/O at task begin" to make sure the I/O is executed just as fast as your program. The example below shows how to move your I/O to the right cycle, but since there was only 1 task in our example, the I/O should not be moved in the example.

The modbus I/Os are always called "gv_MB_arrFbModbusMasters[xx].MB_PcCom" for the serial communication port and "gv_MB_arrFbModbusMasters[xx].MB_KL6x22B" for EL6021.



Name	Type	Size	>Addr...	In/Out	User ID	Linked to
.gv_MB_arrFbModbusMasters[1].MB_PcCom.InData	MB_PcCo...	66.0	0.0	Input	0	
.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData	MB_KL6in...	24.0	66.0	Input	0	
.gv_MB_arrFbModbusSlaves[1]._fbPcCom.InData	MB_PcCo...	66.0	90.0	Input	0	
.gv_MB_arrFbModbusSlaves[1]._fbKL6x22B.InData	MB_KL6in...	24.0	156.0	Input	0	
.gv_MB_arrFbModbusSlaves[1]._fbKL6x5B.InData	MB_KL6in...	6.0	180.0	Input	0	

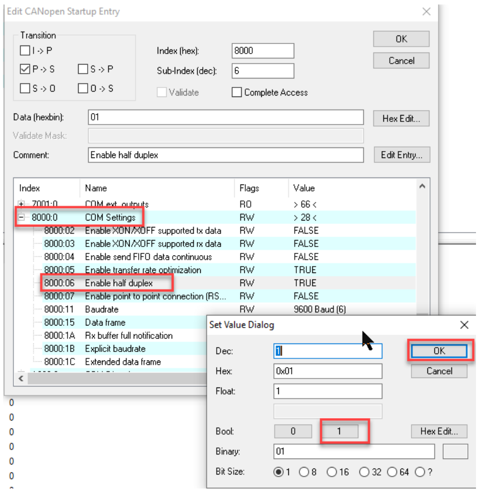
To use an EL6021 as a Modbus master, a few change should be made to the startup list:



Half duplex should be changed to TRUE (Index 8000:06 on EL6021), The baudrate should be changed to the right baud rate (usually 9600) (index 8000:11 on EL6021) and the dataframe should be changed to the right dataframe (usually 8E1) (index 8000:15 on EL6021).

Important:

- On an EL6022 terminal you should configure the startup list for both communication channels.
- Do not change any other Com Settings. If you did change other settings, the right com settings are:
    - enable xon/xoff FALSE
    - Enable xon/xoff FALSE (there are 2 settings with the same name)
    - Enable fifo data continuous FALSE
    - Enable data transfer rate optimization TRUE
    - Enable half duplex TRUE
    - Enable point to point connection FALSE



Edit CANopen Startup Entry

Transition: □ I -> P ☑ P -> S □ S -> P □ S -> O □ O -> S

Index (hex): 8000

Sub-Index (dec): 6

□ Validate □ Complete Access

Data (hexbin): 01

Validate Mask:

Comment: Enable half duplex

Index	Name	Flags	Value
7001:0	COM ext. outputs	RO	> 66 <
8000:0	COM Settings	RW	> 28 <
8000:02	Enable XON/XOFF supported tx data	RW	FALSE
8000:03	Enable XON/XOFF supported rx data	RW	FALSE
8000:04	Enable send FIFO data continuous	RW	FALSE
8000:05	Enable transfer rate optimization	RW	TRUE
8000:06	Enable half duplex	RW	TRUE
8000:07	Enable point to point connection (RS...	RW	FALSE
8000:11	Baudrate	RW	9600 Baud (6)
8000:15	Data frame		
8000:1A	Rx buffer full notification		
8000:1B	Explicit baudrate		
8000:1C	Extended data frame		

Set Value Dialog

Dec: 1

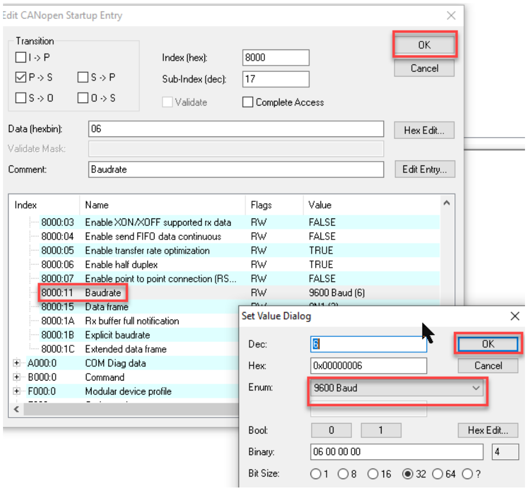
Hex: 0x01

Float: 1

Bool: 0 1

Binary: 01

Bit Size: ● 1 ○ 8 ○ 16 ○ 32 ○ 64 ○ ?



Edit CANopen Startup Entry

Transition: □ I -> P ☑ P -> S □ S -> P □ S -> O □ O -> S

Index (hex): 8000

Sub-Index (dec): 17

□ Validate □ Complete Access

Data (hexbin): 06

Validate Mask:

Comment: Baudrate

Index	Name	Flags	Value
8000:03	Enable XON/XOFF supported rx data	RW	FALSE
8000:04	Enable send FIFO data continuous	RW	FALSE
8000:05	Enable transfer rate optimization	RW	TRUE
8000:06	Enable half duplex	RW	TRUE
8000:07	Enable point to point connection (RS...	RW	FALSE
8000:11	Baudrate	RW	9600 Baud (6)
8000:15	Data frame		
8000:1A	Rx buffer full notification		
8000:1B	Explicit baudrate		
8000:1C	Extended data frame		
A000:0	COM Diag data		
B000:0	Command		
F000:0	Modular device profile		

Set Value Dialog

Dec: 6

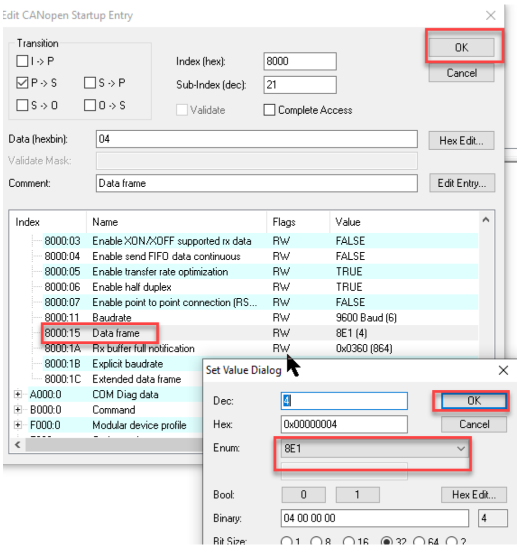
Hex: 0x00000006

Enum: 9600 Baud

Bool: 0 1

Binary: 06 00 00 00 4

Bit Size: ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?



Edit CANopen Startup Entry

Transition: □ I -> P ☑ P -> S □ S -> P □ S -> O □ O -> S

Index (hex): 8000

Sub-Index (dec): 21

□ Validate □ Complete Access

Data (hexbin): 04

Validate Mask:

Comment: Data frame

Index	Name	Flags	Value
8000:03	Enable XON/XOFF supported rx data	RW	FALSE
8000:04	Enable send FIFO data continuous	RW	FALSE
8000:05	Enable transfer rate optimization	RW	TRUE
8000:06	Enable half duplex	RW	TRUE
8000:07	Enable point to point connection (RS...	RW	FALSE
8000:11	Baudrate	RW	9600 Baud (6)
8000:15	Data frame	RW	8E1 (4)
8000:1A	Rx buffer full notification	RW	0x0360 (864)
8000:1B	Explicit baudrate		
8000:1C	Extended data frame		
A000:0	COM Diag data		
B000:0	Command		
F000:0	Modular device profile		

Set Value Dialog

Dec: 4

Hex: 0x00000004

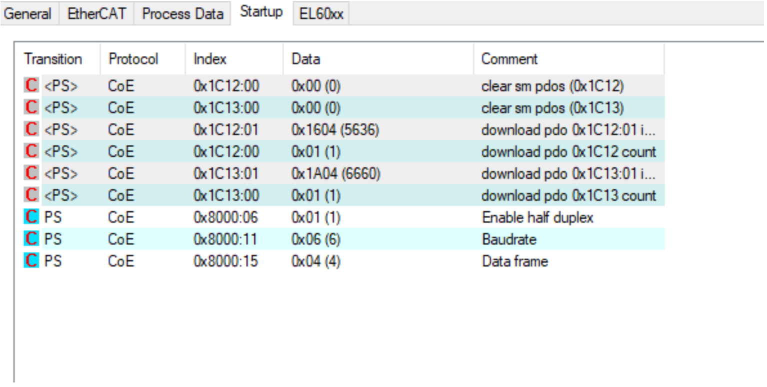
Enum: 8E1

Bool: 0 1

Binary: 04 00 00 00 4

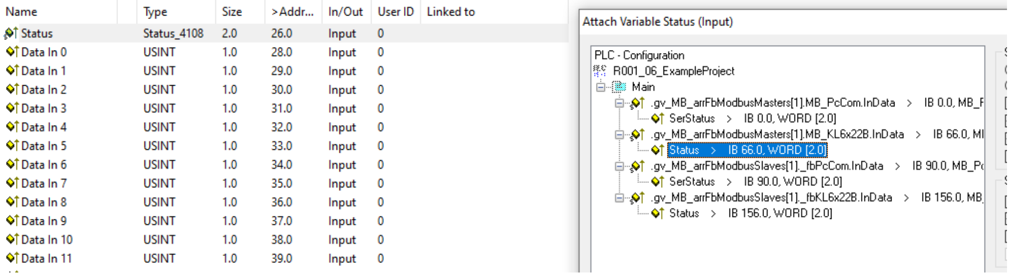
Bit Size: ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

After doing this, the startup list should look like this:



Transition	Protocol	Index	Data	Comment
C <PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
C <PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
C <PS>	CoE	0x1C12:01	0x1604 (5636)	download pdo 0x1C12:01 i...
C <PS>	CoE	0x1C12:00	0x01 (1)	download pdo 0x1C12 count
C <PS>	CoE	0x1C13:01	0x1A04 (6660)	download pdo 0x1C13:01 i...
C <PS>	CoE	0x1C13:00	0x01 (1)	download pdo 0x1C13 count
C PS	CoE	0x8000:06	0x01 (1)	Enable half duplex
C PS	CoE	0x8000:11	0x06 (6)	Baudrate
C PS	CoE	0x8000:15	0x04 (4)	Data frame

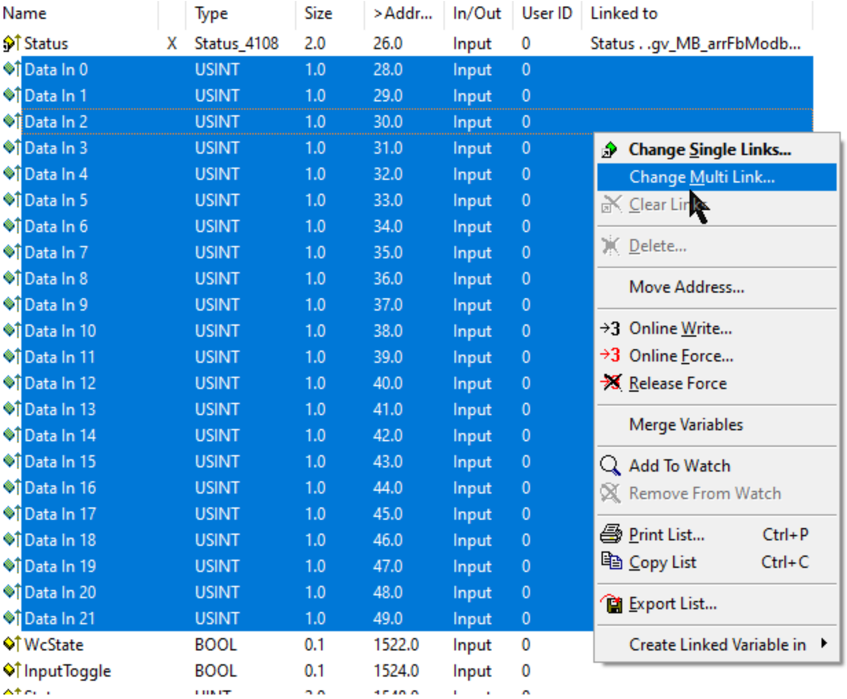
The I/O should be linked by linking the status to status, ctrl to ctrl and the D to data To link the data it is possible to select all data inputs/outputs and clicking on “change multi link”:



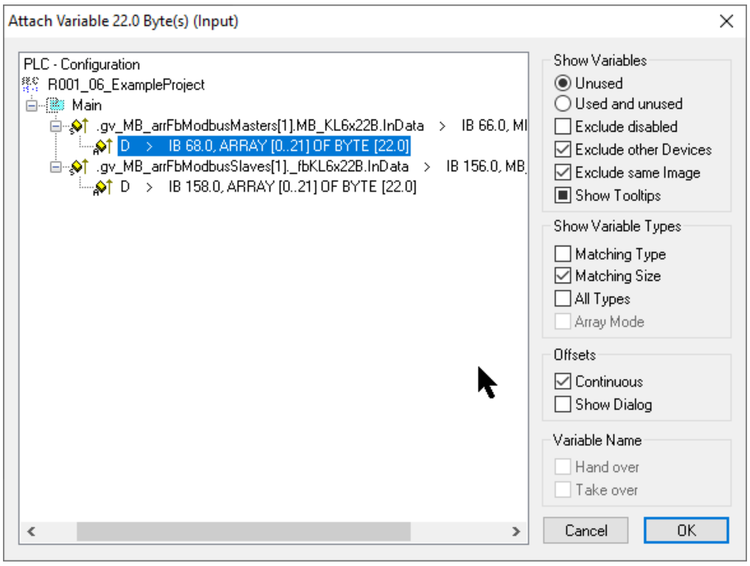
Name	Type	Size	>Addr...	In/Out	User ID	Linked to
Status	Status_4108	2.0	26.0	Input	0	
Data In 0	USINT	1.0	28.0	Input	0	
Data In 1	USINT	1.0	29.0	Input	0	
Data In 2	USINT	1.0	30.0	Input	0	
Data In 3	USINT	1.0	31.0	Input	0	
Data In 4	USINT	1.0	32.0	Input	0	
Data In 5	USINT	1.0	33.0	Input	0	
Data In 6	USINT	1.0	34.0	Input	0	
Data In 7	USINT	1.0	35.0	Input	0	
Data In 8	USINT	1.0	36.0	Input	0	
Data In 9	USINT	1.0	37.0	Input	0	
Data In 10	USINT	1.0	38.0	Input	0	
Data In 11	USINT	1.0	39.0	Input	0	

Attach Variable Status (Input)

```
PLC - Configuration
R001_06_ExampleProject
  Main
    .gv_MB_arrFbModbusMasters[1].MB_PcCom.InData   >   IB 0.0, MB_F
      SerStatus   >   IB 0.0, WORD [2.0]
    .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData   >   IB 66.0, MI
      Status   >   IB 66.0, WORD [2.0]
    .gv_MB_arrFbModbusSlaves[1]._fbPcCom.InData   >   IB 90.0, MB_Pc
      SerStatus   >   IB 90.0, WORD [2.0]
    .gv_MB_arrFbModbusSlaves[1]._fbKL6x22B.InData   >   IB 156.0, MB_
      Status   >   IB 156.0, WORD [2.0]
```



Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status	X	Status_4108	2.0	26.0	Input	0	Status . .gv_MB_arrFbModb...
Data In 0		USINT	1.0	28.0	Input	0	
Data In 1		USINT	1.0	29.0	Input	0	
Data In 2		USINT	1.0	30.0	Input	0	
Data In 3		USINT	1.0	31.0	Input	0	
Data In 4		USINT	1.0	32.0	Input	0	
Data In 5		USINT	1.0	33.0	Input	0	
Data In 6		USINT	1.0	34.0	Input	0	
Data In 7		USINT	1.0	35.0	Input	0	
Data In 8		USINT	1.0	36.0	Input	0	
Data In 9		USINT	1.0	37.0	Input	0	
Data In 10		USINT	1.0	38.0	Input	0	
Data In 11		USINT	1.0	39.0	Input	0	
Data In 12		USINT	1.0	40.0	Input	0	
Data In 13		USINT	1.0	41.0	Input	0	
Data In 14		USINT	1.0	42.0	Input	0	
Data In 15		USINT	1.0	43.0	Input	0	
Data In 16		USINT	1.0	44.0	Input	0	
Data In 17		USINT	1.0	45.0	Input	0	
Data In 18		USINT	1.0	46.0	Input	0	
Data In 19		USINT	1.0	47.0	Input	0	
Data In 20		USINT	1.0	48.0	Input	0	
Data In 21		USINT	1.0	49.0	Input	0	
WcState		BOOL	0.1	1522.0	Input	0	
InputToggle		BOOL	0.1	1524.0	Input	0	



Attach Variable 22.0 Byte(s) (Input)

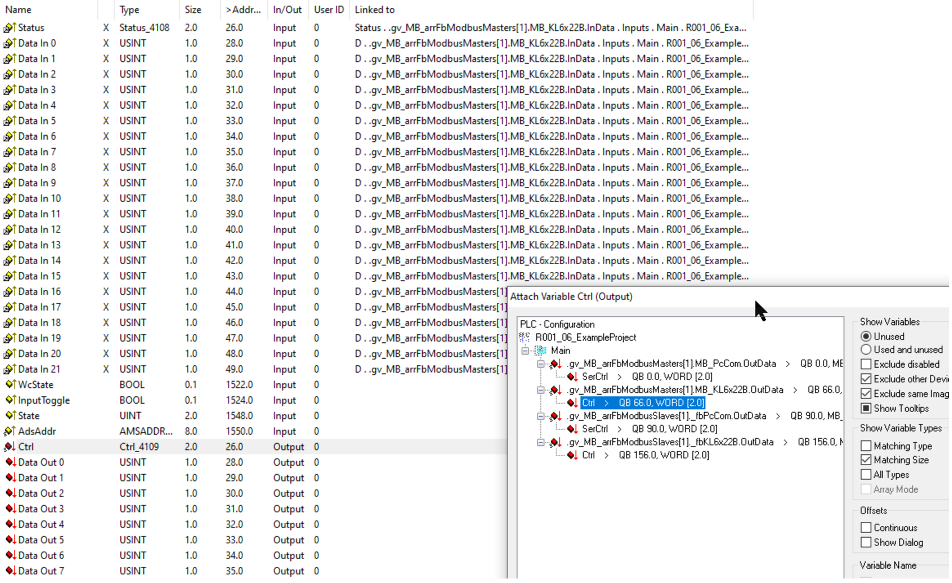
PLC - Configuration
R001_06_ExampleProject
Main
.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData > IB 66.0, MI
D > IB 68.0, ARRAY [0..21] OF BYTE [22.0]
.gv_MB_arrFbModbusSlaves[1]._fbKL6x22B.InData > IB 156.0, MB
D > IB 158.0, ARRAY [0..21] OF BYTE [22.0]

Show Variables
● Unused
○ Used and unused
☐ Exclude disabled
☑ Exclude other Devices
☑ Exclude same Image
■ Show Tooltips

Show Variable Types
☐ Matching Type
☑ Matching Size
☐ All Types
☐ Array Mode

Offsets
☑ Continuous
☐ Show Dialog

Variable Name
☐ Hand over
☐ Take over



Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status	X	Status_4108	2.0	26.0	Input	0	Status . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Exa...
Data In 0	X	USINT	1.0	28.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 1	X	USINT	1.0	29.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 2	X	USINT	1.0	30.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 3	X	USINT	1.0	31.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 4	X	USINT	1.0	32.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 5	X	USINT	1.0	33.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 6	X	USINT	1.0	34.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 7	X	USINT	1.0	35.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 8	X	USINT	1.0	36.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 9	X	USINT	1.0	37.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 10	X	USINT	1.0	38.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 11	X	USINT	1.0	39.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 12	X	USINT	1.0	40.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 13	X	USINT	1.0	41.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 14	X	USINT	1.0	42.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 15	X	USINT	1.0	43.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_Example...
Data In 16	X	USINT	1.0	44.0	Input	0	D . .gv_MB_arrFbModbusMasters[1]
Data In 17	X	USINT	1.0	45.0	Input	0	D . .gv_MB_arrFbModbusMasters[1]
Data In 18	X	USINT	1.0	46.0	Input	0	D . .gv_MB_arrFbModbusMasters[1]
Data In 19	X	USINT	1.0	47.0	Input	0	D . .gv_MB_arrFbModbusMasters[1]
Data In 20	X	USINT	1.0	48.0	Input	0	D . .gv_MB_arrFbModbusMasters[1]
Data In 21	X	USINT	1.0	49.0	Input	0	D . .gv_MB_arrFbModbusMasters[1]
WcState		BOOL	0.1	1522.0	Input	0	
InputToggle		BOOL	0.1	1524.0	Input	0	
State		UINT	2.0	1548.0	Input	0	
AdsAddr		AMSADDR...	8.0	1550.0	Input	0	
Ctrl		Ctrl_4109	2.0	26.0	Output	0	
Data Out 0		USINT	1.0	28.0	Output	0	
Data Out 1		USINT	1.0	29.0	Output	0	
Data Out 2		USINT	1.0	30.0	Output	0	
Data Out 3		USINT	1.0	31.0	Output	0	
Data Out 4		USINT	1.0	32.0	Output	0	
Data Out 5		USINT	1.0	33.0	Output	0	
Data Out 6		USINT	1.0	34.0	Output	0	
Data Out 7		USINT	1.0	35.0	Output	0	

Attach Variable Ctrl (Output)

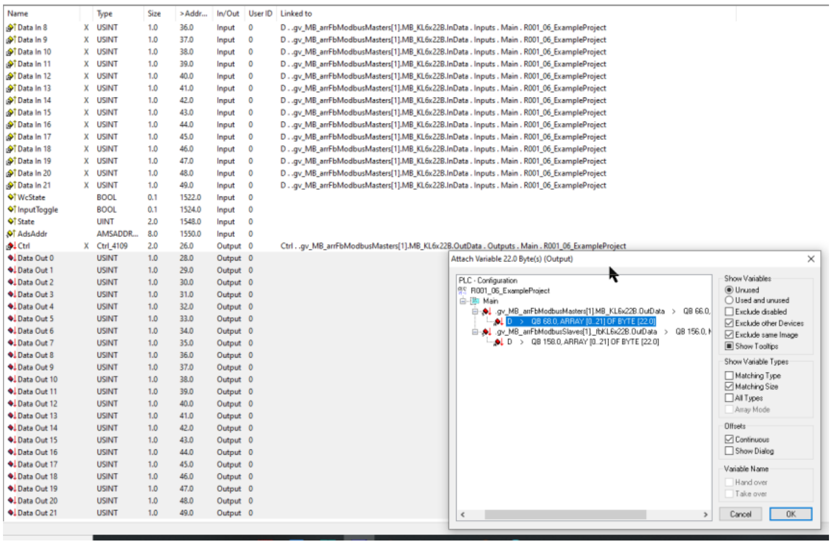
PLC - Configuration
R001_06_ExampleProject
Main
.gv_MB_arrFbModbusMasters[1].MB_PcCom.OutData > QB 0.0, MB
SerCtrl > QB 0.0, WORD [2.0]
.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData > QB 66.0,
Ctrl > QB 66.0, WORD [2.0]
.gv_MB_arrFbModbusSlaves[1]._fbPcCom.OutData > QB 90.0, MB_
SerCtrl > QB 90.0, WORD [2.0]
.gv_MB_arrFbModbusSlaves[1]._fbKL6x22B.OutData > QB 156.0, M
Ctrl > QB 156.0, WORD [2.0]

Show Variables
● Unused
○ Used and unused
☐ Exclude disabled
☑ Exclude other Devi
☑ Exclude same Imag
■ Show Tooltips

Show Variable Types
☐ Matching Type
☑ Matching Size
☐ All Types
☐ Array Mode

Offsets
☐ Continuous
☐ Show Dialog

Variable Name



Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Data In 8	X	USINT	1.0	36.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 9	X	USINT	1.0	37.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 10	X	USINT	1.0	38.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 11	X	USINT	1.0	39.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 12	X	USINT	1.0	40.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 13	X	USINT	1.0	41.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 14	X	USINT	1.0	42.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 15	X	USINT	1.0	43.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 16	X	USINT	1.0	44.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 17	X	USINT	1.0	45.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 18	X	USINT	1.0	46.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 19	X	USINT	1.0	47.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 20	X	USINT	1.0	48.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 21	X	USINT	1.0	49.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
WcState		BOOL	0.1	1522.0	Input	0	
InputToggle		BOOL	0.1	1524.0	Input	0	
State		UINT	2.0	1548.0	Input	0	
AdsAddr		AMSADDR...	8.0	1550.0	Input	0	
Ctrl	X	Ctrl_4109	2.0	26.0	Output	0	Ctrl . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 0		USINT	1.0	28.0	Output	0	
Data Out 1		USINT	1.0	29.0	Output	0	
Data Out 2		USINT	1.0	30.0	Output	0	
Data Out 3		USINT	1.0	31.0	Output	0	
Data Out 4		USINT	1.0	32.0	Output	0	
Data Out 5		USINT	1.0	33.0	Output	0	
Data Out 6		USINT	1.0	34.0	Output	0	
Data Out 7		USINT	1.0	35.0	Output	0	
Data Out 8		USINT	1.0	36.0	Output	0	
Data Out 9		USINT	1.0	37.0	Output	0	
Data Out 10		USINT	1.0	38.0	Output	0	
Data Out 11		USINT	1.0	39.0	Output	0	
Data Out 12		USINT	1.0	40.0	Output	0	
Data Out 13		USINT	1.0	41.0	Output	0	
Data Out 14		USINT	1.0	42.0	Output	0	
Data Out 15		USINT	1.0	43.0	Output	0	
Data Out 16		USINT	1.0	44.0	Output	0	
Data Out 17		USINT	1.0	45.0	Output	0	
Data Out 18		USINT	1.0	46.0	Output	0	
Data Out 19		USINT	1.0	47.0	Output	0	
Data Out 20		USINT	1.0	48.0	Output	0	
Data Out 21		USINT	1.0	49.0	Output	0	

Attach Variable 22.0 Byte(s) (Output)

```
PLC - Configuration
R001_06_ExampleProject
  Main
    .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData  >  QB 66.0,
      D  >  QB 68.0, ARRAY [0..21] OF BYTE [22.0]
    .gv_MB_arrFbModbusSlaves[1]._fbKL6x22B.OutData  >  QB 156.0, I
      D  >  QB 158.0, ARRAY [0..21] OF BYTE [22.0]
```

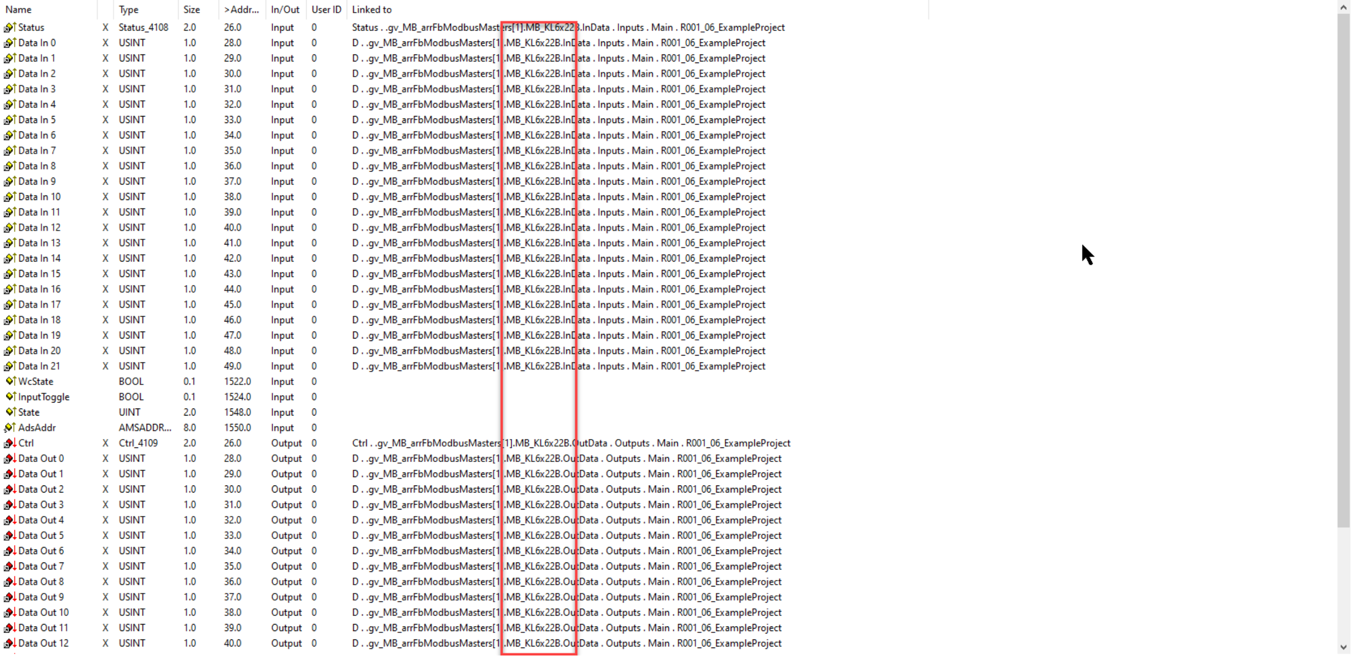
Show Variables
● Unused
○ Used and unused
☐ Exclude disabled
☑ Exclude other Devices
☑ Exclude same Image
■ Show Tooltips

Show Variable Types
☐ Matching Type
☑ Matching Size
☐ All Types
☐ Array Mode

Offsets
☑ Continuous
☐ Show Dialog

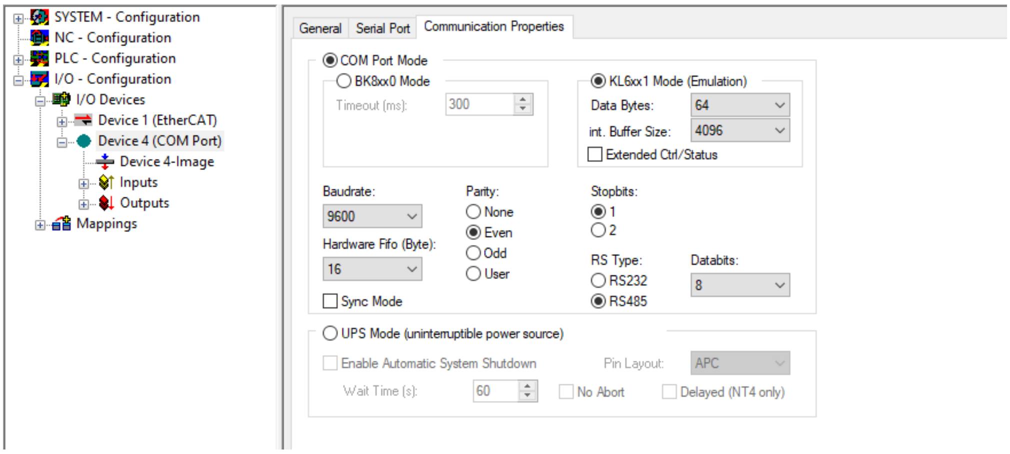
Variable Name
☐ Hand over
☐ Take over

Make sure you always link MB_KL6x22B when using an EL6021:



Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status	X	Status_4108	2.0	26.0	Input	0	Status . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 0	X	USINT	1.0	28.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 1	X	USINT	1.0	29.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 2	X	USINT	1.0	30.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 3	X	USINT	1.0	31.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 4	X	USINT	1.0	32.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 5	X	USINT	1.0	33.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 6	X	USINT	1.0	34.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 7	X	USINT	1.0	35.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 8	X	USINT	1.0	36.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 9	X	USINT	1.0	37.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 10	X	USINT	1.0	38.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 11	X	USINT	1.0	39.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 12	X	USINT	1.0	40.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 13	X	USINT	1.0	41.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 14	X	USINT	1.0	42.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 15	X	USINT	1.0	43.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 16	X	USINT	1.0	44.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 17	X	USINT	1.0	45.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 18	X	USINT	1.0	46.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 19	X	USINT	1.0	47.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 20	X	USINT	1.0	48.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
Data In 21	X	USINT	1.0	49.0	Input	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . Inputs . Main . R001_06_ExampleProject
WcState		BOOL	0.1	1522.0	Input	0	
InputToggle		BOOL	0.1	1524.0	Input	0	
State		UINT	2.0	1548.0	Input	0	
AdsAddr		AMSADDR...	8.0	1550.0	Input	0	
Ctrl	X	Ctrl_4109	2.0	26.0	Output	0	Ctrl . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 0	X	USINT	1.0	28.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 1	X	USINT	1.0	29.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 2	X	USINT	1.0	30.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 3	X	USINT	1.0	31.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 4	X	USINT	1.0	32.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 5	X	USINT	1.0	33.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 6	X	USINT	1.0	34.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 7	X	USINT	1.0	35.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 8	X	USINT	1.0	36.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 9	X	USINT	1.0	37.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 10	X	USINT	1.0	38.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 11	X	USINT	1.0	39.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject
Data Out 12	X	USINT	1.0	40.0	Output	0	D . .gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . Outputs . Main . R001_06_ExampleProject

If the serial communication port is used, the communication properties of the port must be set as shown below. Change the baud rate, parity and stop bits according to your setup.



```
SYSTEM - Configuration
NC - Configuration
PLC - Configuration
I/O - Configuration
  I/O Devices
    Device 1 (EtherCAT)
    Device 4 (COM Port)
      Device 4-Image
      Inputs
      Outputs
  Mappings
```

General | Serial Port | Communication Properties

● COM Port Mode

○ BK8xx0 Mode — Timeout (ms): 300

● KL6xx1 Mode (Emulation) — Data Bytes: 64; int. Buffer Size: 4096; □ Extended Ctrl/Status

Baudrate: 9600

Hardware Fifo (Byte): 16

□ Sync Mode

Parity: ○ None ● Even ○ Odd ○ User

Stopbits: ● 1 ○ 2

RS Type: ○ RS232 ● RS485

Databits: 8

○ UPS Mode (uninterruptible power source)

□ Enable Automatic System Shutdown — Pin Layout: APC

Wait Time (s): 60 — □ No Abort — □ Delayed (NT4 only)

Linking the serial com port is similar to linking the EL6021, but this time, make sure you always link MB_PcCom when using a serial com port.

Integration in a Twincat 3 project

Short guide to implementing upzio modbus devices into Twincat 3

- Step 1: Use RS485 hardware. The EL6021 or a PLC with an integrated serial com port can be used.
- Step 2: The RS485 hardware must be wired in a way that enables half-duplex communication.
- Step 3: Download all required files, import, and add these to your project.
- Step 4: Change settings in your project according to your hardware.
- Step 5: Execute P_ModbusMain() somewhere in your project.
- Step 6: Create instances of the TP10 and execute somewhere.
- Step 7: Set up the I/O.

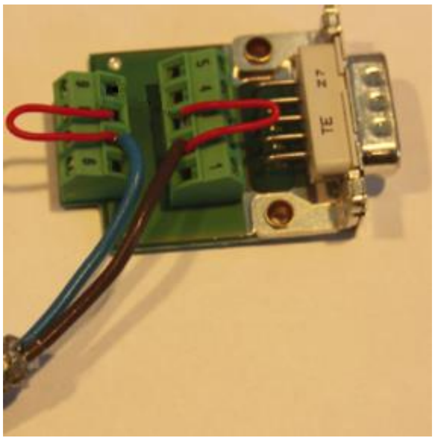
Step1: pick RS485 hardware

To implement modbus RTU on a Beckhoff PLC, RS485 hardware must be used. To use the serial com port of the PLC, you will need an RS485 connector. (e.g., Subcon 9/M-SH from phoenix contact). If your PLC does not have a serial com port or you need more than one modbus master, the EL6021 can be used instead of the serial com port.

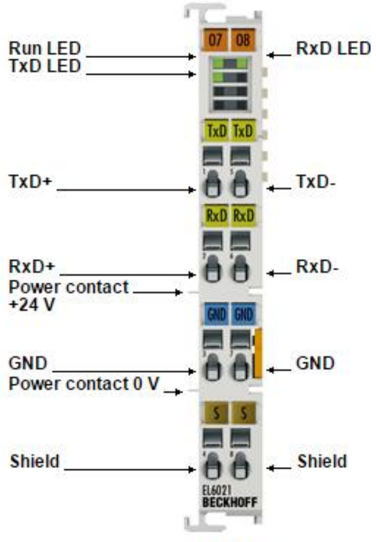
Step2: Wiring the hardware

If the serial com port of the PLC is used with a Subcon 9/M-SH from phoenix contact, you will have to wire the RS485 connector as shown in the pictures below to enable half duplex communication.

The Rx wire must be connected to terminal 2 and 3. The Tx wire must be connected to terminal 7 and 8.



If the EL6021 is used, TxD+ and RxD+ must be connected to each other and to the B-wire. TxD- and RxD- must be connected to each other and to the A-wire.



Top view

Step 3: Download and import

For an easy implementation, the necessary libraries and export files must be downloaded and imported into your project. (see https://www.upzio.com/support)

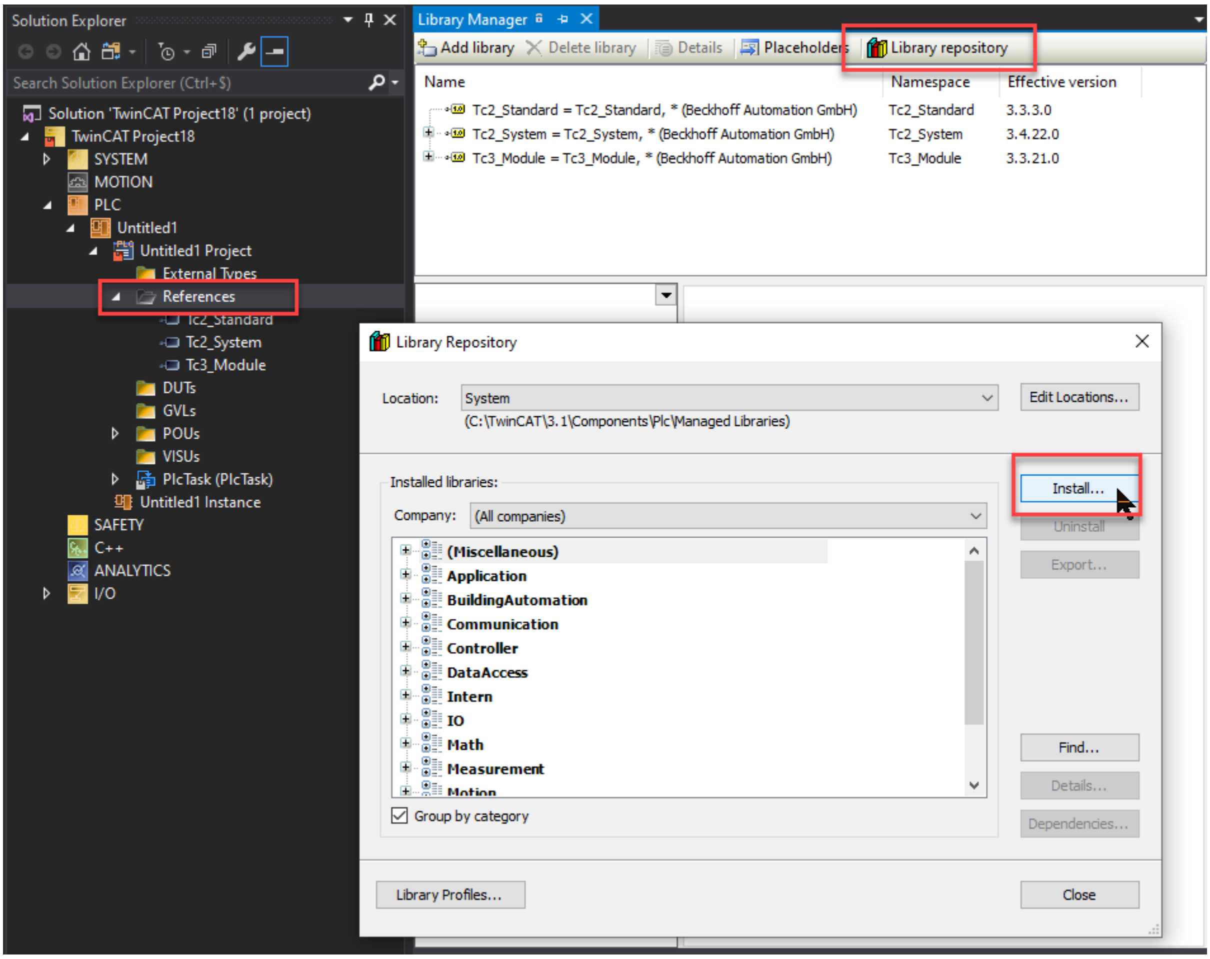
There are 3 libraries required for the modbus devices:

1. ModbusRTU_Upzio.library , this library works just like the TC2 Beckhoff ModbusRTU library, but is a little faster.
2. TcUpzioMbGen.library , this is the general Upzio modbus library that controls the message buffers of every Modbus master.
3. TcUpzioMbDev.library , this is the device Upzio modbus library. This library contains all Upzio modbus devices.

There is 1 xml file required for the modbus devices, which should both be imported in the project.

1. ExampleSettings.xml, this contains ‘Global_ModbusSettings’ which is a global variable list with all modbus settings and ‘P_ModbusMain’ which is a program that should be executed somewhere in your project.

To install the libraries (or to update your libraries), open a twincat project and go to ‘References’. Then click on ‘Library repository’, next click on ‘Install..’.



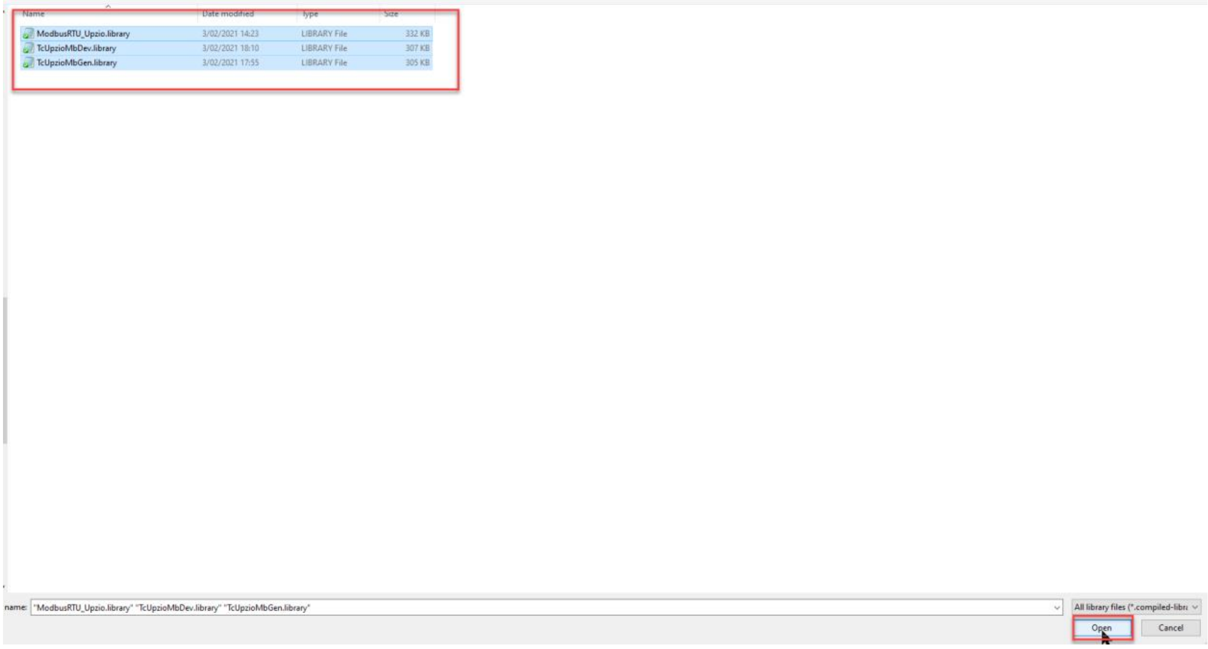
Name	Namespace	Effective version
Tc2_Standard = Tc2_Standard, * (Beckhoff Automation GmbH)	Tc2_Standard	3.3.3.0
Tc2_System = Tc2_System, * (Beckhoff Automation GmbH)	Tc2_System	3.4.22.0
Tc3_Module = Tc3_Module, * (Beckhoff Automation GmbH)	Tc3_Module	3.3.21.0

Location: System
(C:\TwinCAT\3.1\Components\Plc\Managed Libraries)

Company: (All companies)

☑ Group by category

Navigate to your 3 library files, select them, and click on open.

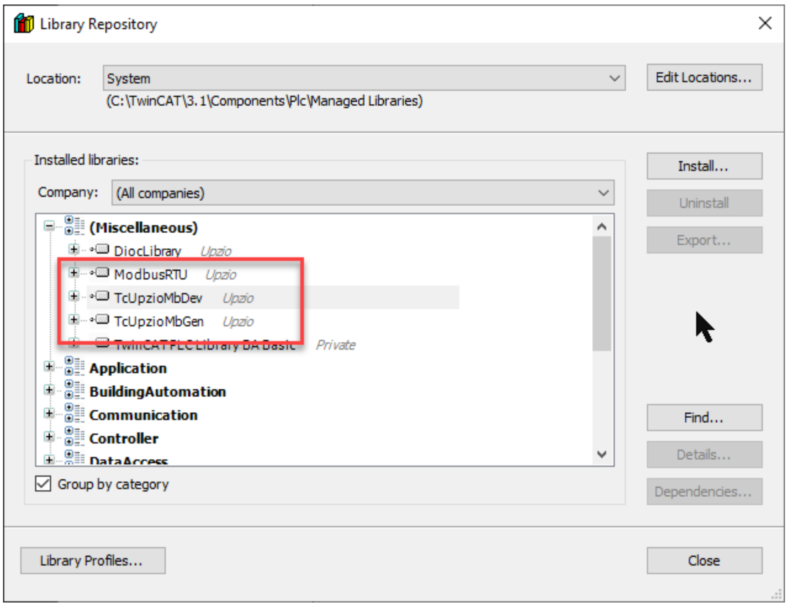


Name	Date modified	Type	Size
ModbusRTU_Upzio.library	3/02/2021 14:23	LIBRARY File	332 KB
TcUpzioMbDev.library	3/02/2021 18:10	LIBRARY File	307 KB
TcUpzioMbGen.library	3/02/2021 17:55	LIBRARY File	305 KB

name: "ModbusRTU_Upzio.library" "TcUpzioMbDev.library" "TcUpzioMbGen.library"

All library files (*.compiled-libr

If the installation was successful, you will notice three new libraries in your library repository:



Location: System
(C:\TwinCAT\3.1\Components\Plc\Managed Libraries)

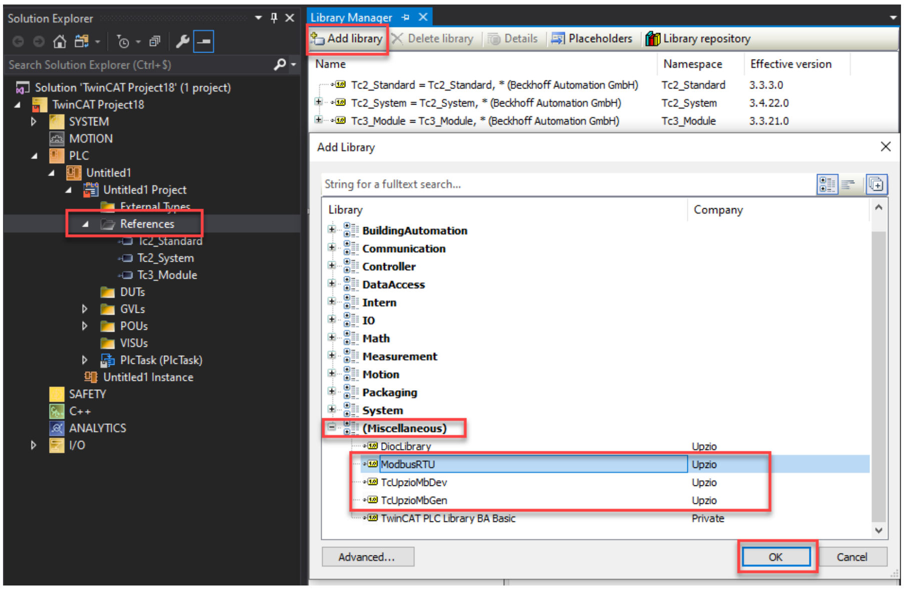
Installed libraries:

Company: (All companies)

(Miscellaneous)	
DiocLibrary	Upzio
ModbusRTU	Upzio
TcUpzioMbDev	Upzio
TcUpzioMbGen	Upzio
TwinCAT PLC Library BA Basic	Private
Application	
BuildingAutomation	
Communication	
Controller	
DataAccess	

☑ Group by category

After installing the libraries, all libraries should be included in your project by going to ‘References’ – ‘Add library’ – ‘Miscellaneous’ – ‘ModbusRTU’, ‘TcUpzioMbDev’ and ‘TcUpzioMbGen’ and clicking on ‘OK’.

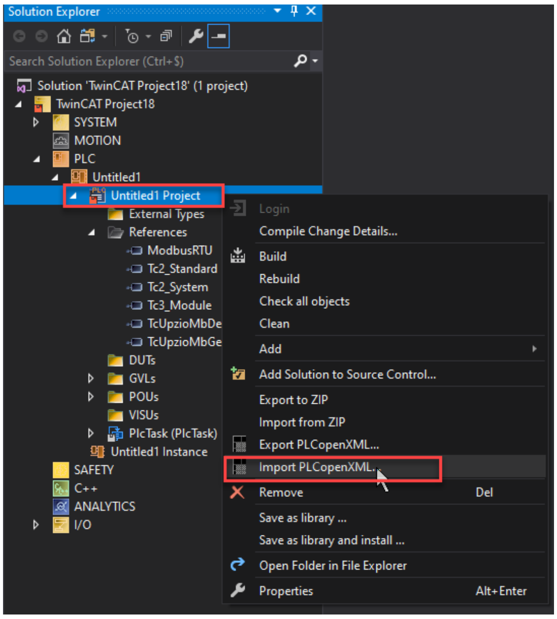


Solution 'TwinCAT Project18' (1 project)
TwinCAT Project18
SYSTEM
MOTION
PLC
Untitled1
Untitled1 Project
External Types
References
Tc2_Standard
Tc2_System
Tc3_Module
DUTs
GVLs
POUs
VISUs
PlcTask (PlcTask)
Untitled1 Instance
SAFETY
C++
ANALYTICS
I/O

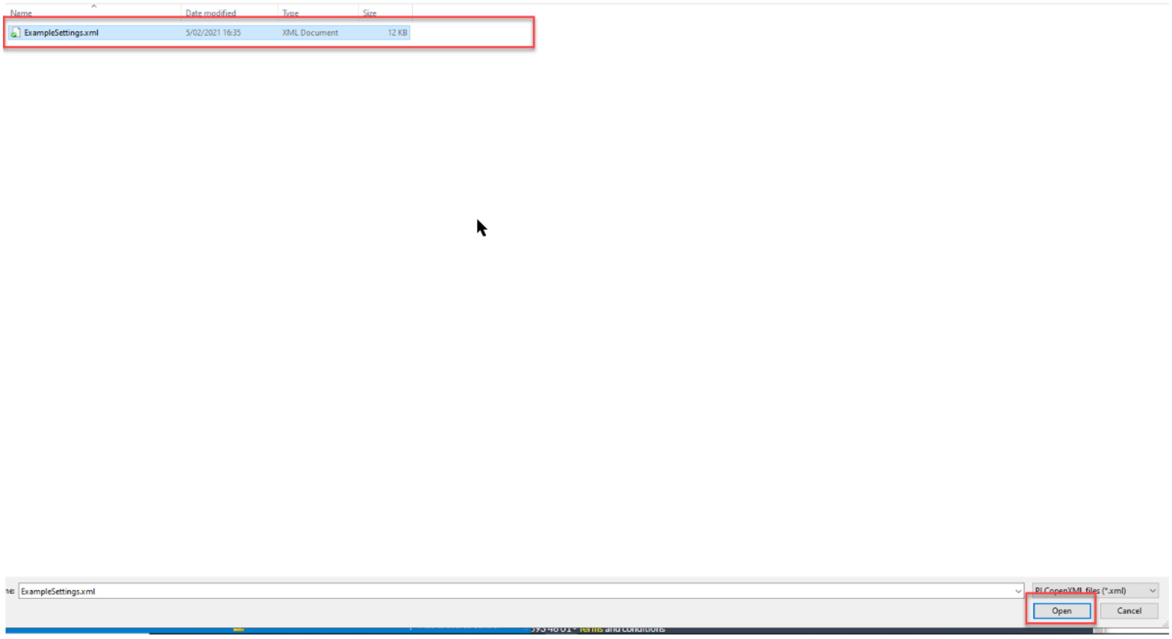
Name	Namespace	Effective version
Tc2_Standard = Tc2_Standard, * (Beckhoff Automation GmbH)	Tc2_Standard	3.3.3.0
Tc2_System = Tc2_System, * (Beckhoff Automation GmbH)	Tc2_System	3.4.22.0
Tc3_Module = Tc3_Module, * (Beckhoff Automation GmbH)	Tc3_Module	3.3.21.0

Library	Company
BuildingAutomation	
Communication	
Controller	
DataAccess	
Intern	
IO	
Math	
Measurement	
Motion	
Packaging	
System	
(Miscellaneous)	
DiocLibrary	Upzio
ModbusRTU	Upzio
TcUpzioMbDev	Upzio
TcUpzioMbGen	Upzio
TwinCAT PLC Library BA Basic	Private

The XML file can be imported by right clicking on your project, and selecting ‘Import PLCOpenXml…’



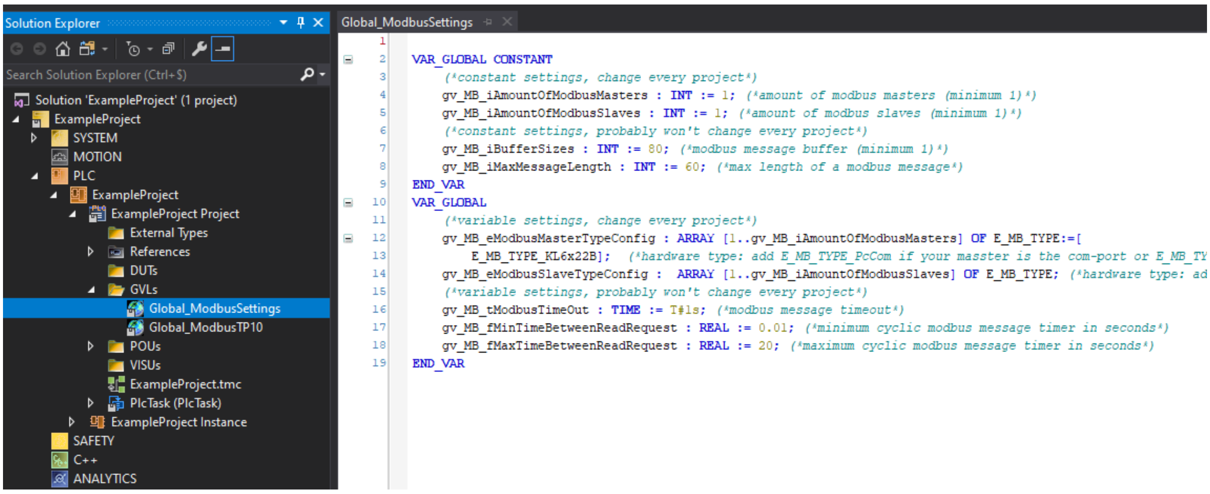
Navigate to your ExampleSettings.xml file, select it and click on 'open'.



This should add 'Global_ModbusSettings' and 'P_ModbusMain' to your project.

Step 4: Change settings according to your hardware

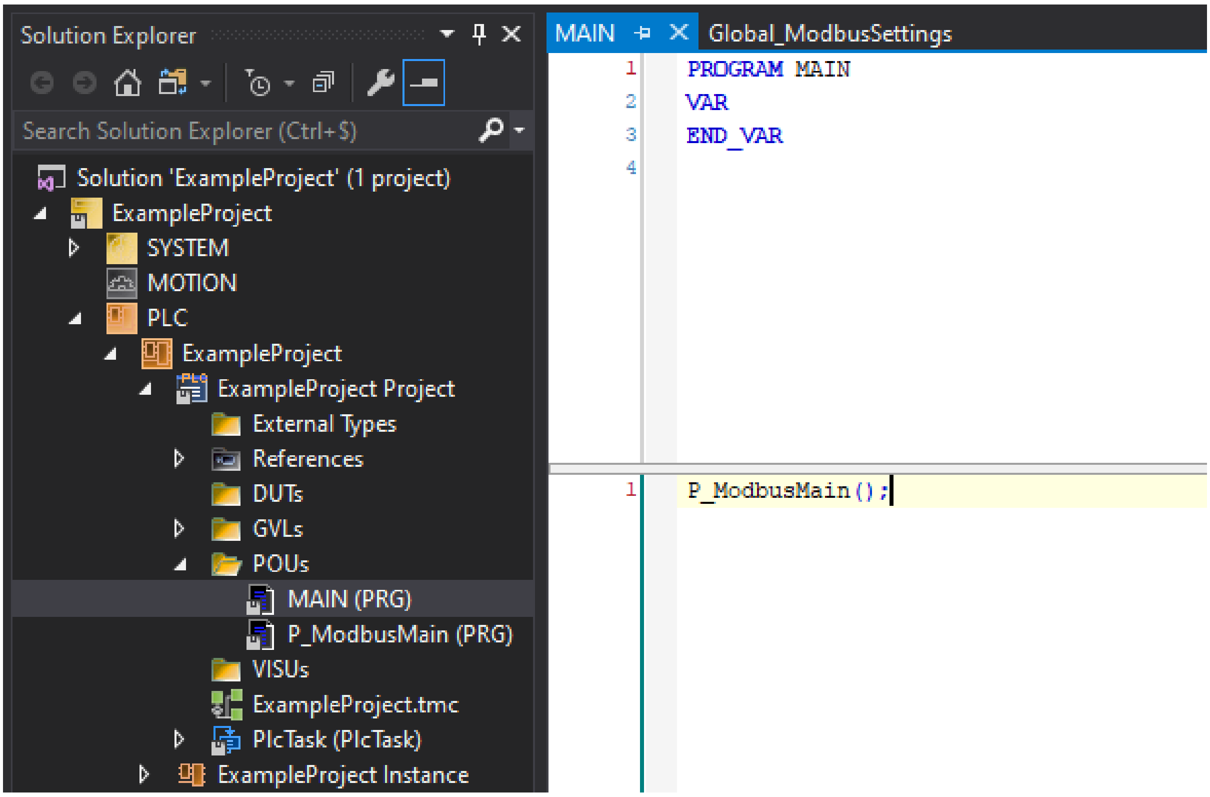
The xml file contained a global variable list 'Global_ModbusSettings' where the modbus settings can be altered. The most important setting is 'gv_MB_eModbusMasterTypeConfig'. In gv_MB_eModbusMasterTypeConfig you must define what kind of hardware you will be using. Eg. If your first modbus master is the serial com port, gv_MB_eModbusMasterTypeConfig[1] must be E_MB_TYPE_PcCom. If your second modbus master is an EL6021, gv_MB_eModbusMasterTypeConfig[1] must be E_MB_TYPE_KL6x22B.



```

VAR_GLOBAL CONSTANT
    (*constant settings, change every project*)
    gv_MB_iAmountOfModbusMasters : INT := 1; (*amount of modbus masters (minimum 1)*)
    gv_MB_iAmountOfModbusSlaves : INT := 1; (*amount of modbus slaves (minimum 1)*)
    (*constant settings, probably won't change every project*)
    gv_MB_iBufferSizes : INT := 80; (*modbus message buffer (minimum 1)*)
    gv_MB_iMaxMessageLength : INT := 60; (*max length of a modbus message*)
END_VAR
VAR_GLOBAL
    (*variable settings, change every project*)
    gv_MB_eModbusMasterTypeConfig : ARRAY [1..gv_MB_iAmountOfModbusMasters] OF E_MB_TYPE:=[
        E_MB_TYPE_KL6x22B];  (*hardware type: add E_MB_TYPE_PcCom if your masster is the com-port or E_MB_TY
    gv_MB_eModbusSlaveTypeConfig :  ARRAY [1..gv_MB_iAmountOfModbusSlaves] OF E_MB_TYPE; (*hardware type: ad
    (*variable settings, probably won't change every project*)
    gv_MB_tModbusTimeOut : TIME := T#1s; (*modbus message timeout*)
    gv_MB_fMinTimeBetweenReadRequest : REAL := 0.01; (*minimum cyclic modbus message timer in seconds*)
    gv_MB_fMaxTimeBetweenReadRequest : REAL := 20; (*maximum cyclic modbus message timer in seconds*)
END_VAR
```

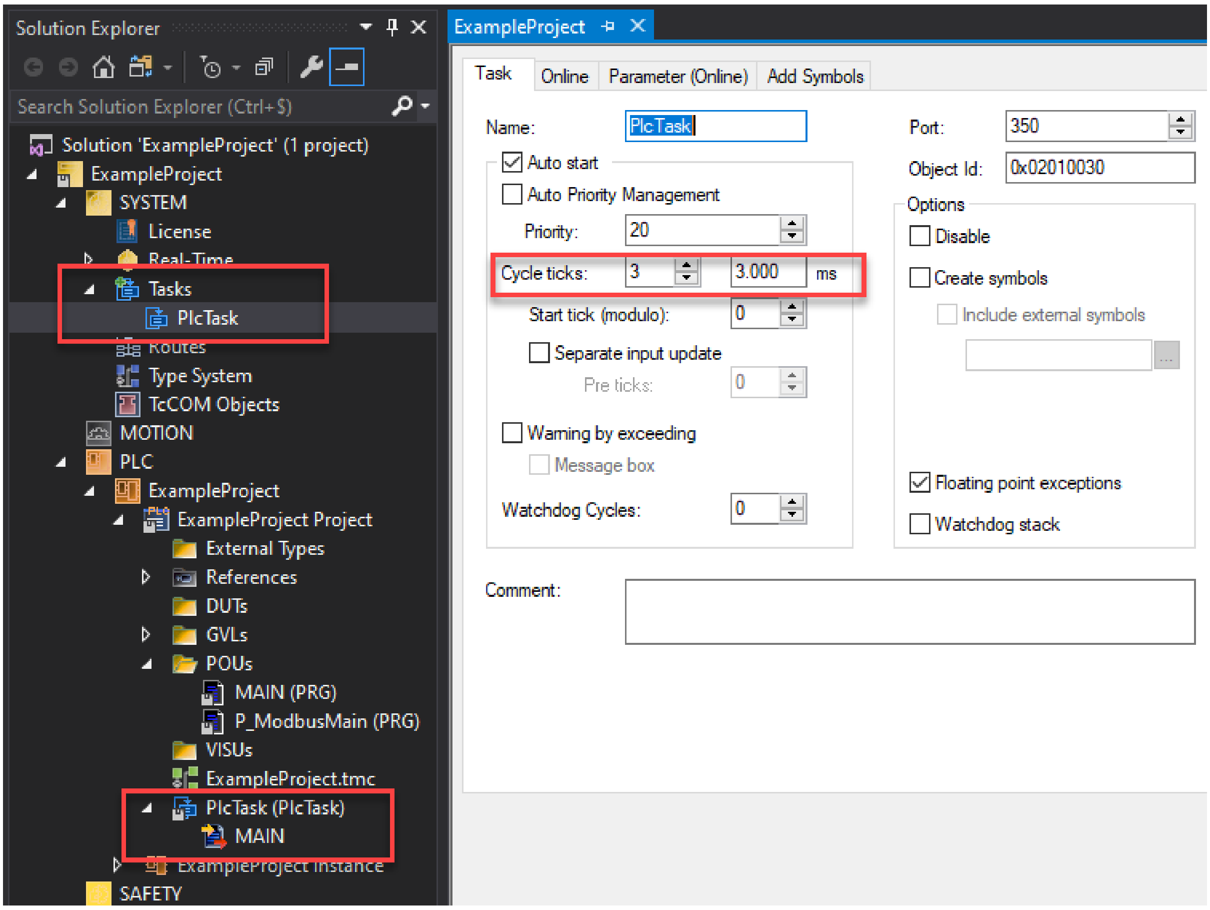
Step 5: Execute P_ModbusMain()

The xml file contained a program ‘P_ModbusMain’. This program should be executed somewhere in your project. This program copies all global modbus settings to the Modbus masters and handles all modbus message buffers. The modbus communication will work wherever you execute this program. But the modbus communication will be faster if this program is executed in a task with a lower cycle time. Therefore, it is recommended to put this program in a task with a low cycle time (eg. 3ms).



```
PROGRAM MAIN
VAR
END_VAR

```

```
P_ModbusMain();
```



Solution Explorer

Search Solution Explorer (Ctrl+$)

- Solution 'ExampleProject' (1 project)
  - ExampleProject
    - SYSTEM
      - License
      - Real-Time
      - Tasks
        - PlcTask
      - Routes
      - Type System
      - TcCOM Objects
    - MOTION
    - PLC
      - ExampleProject
        - ExampleProject Project
          - External Types
          - References
          - DUTs
          - GVLs
          - POUs
            - MAIN (PRG)
            - P_ModbusMain (PRG)
          - VISUs
          - ExampleProject.tmc
          - PlcTask (PlcTask)
            - MAIN
        - ExampleProject Instance
    - SAFETY

ExampleProject

Task | Online | Parameter (Online) | Add Symbols

Name: PlcTask

Port: 350

Object Id: 0x02010030

☑ Auto start

□ Auto Priority Management

Priority: 20

Cycle ticks: 3 3.000 ms

Start tick (modulo): 0

□ Separate input update

Pre ticks: 0

□ Warning by exceeding

□ Message box

Watchdog Cycles: 0

Options

□ Disable

□ Create symbols

□ Include external symbols

☑ Floating point exceptions

□ Watchdog stack

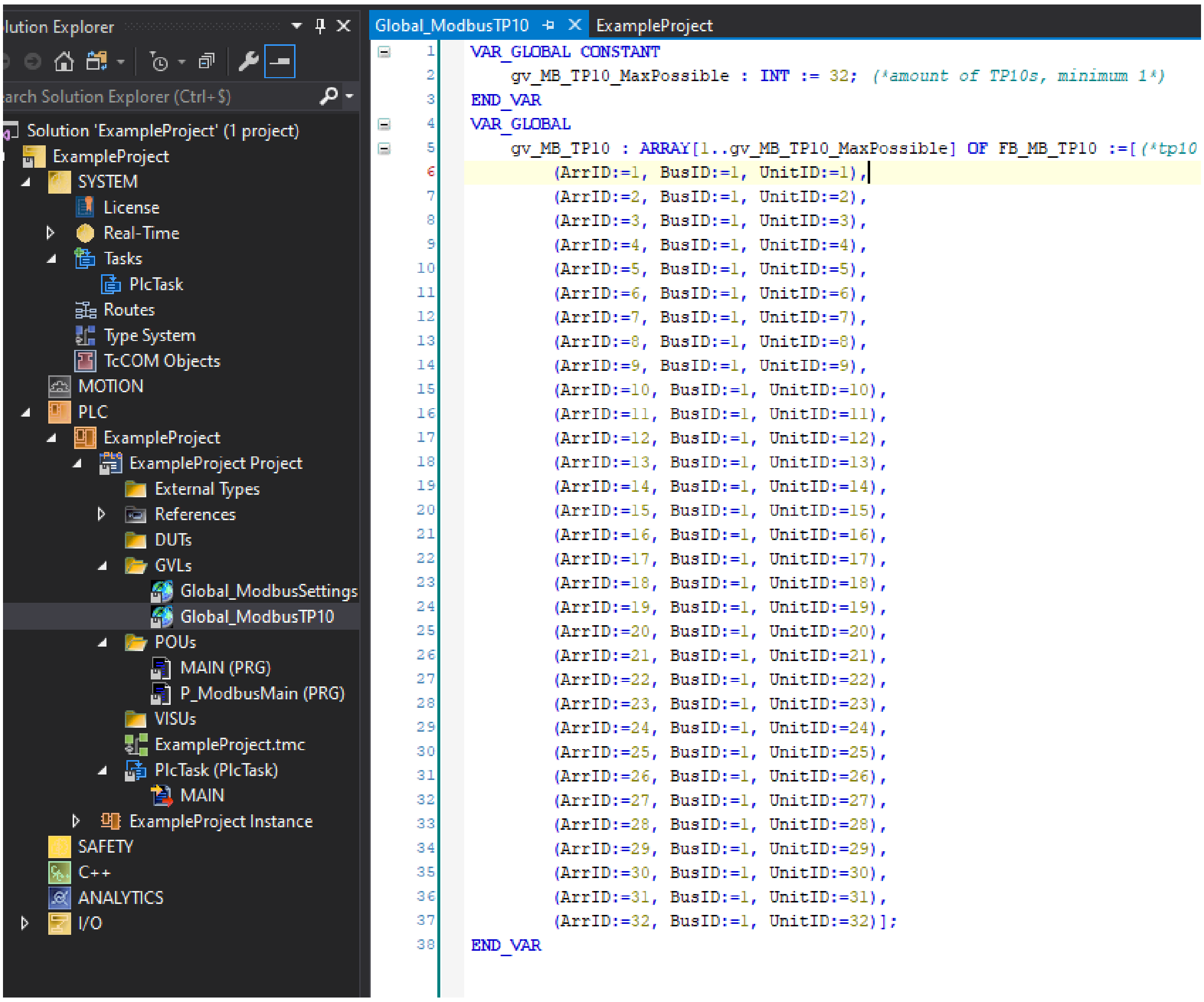
Comment:

Step 6: Create instances of the TP10 and execute

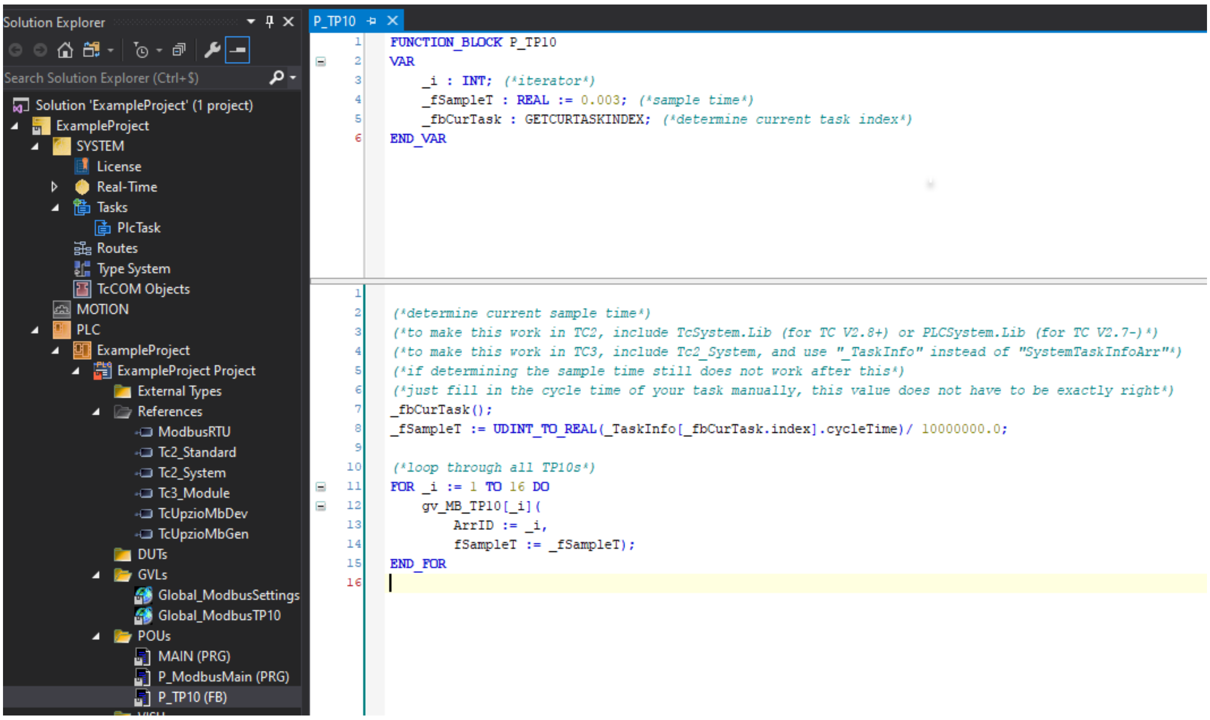
In the example below a global array of 32 TP10s is created. BusID is the Modbus master to which the TP10s are physically wired. UnitID is the modbus address of the TP10s.

The instances of the TP10s must be executed somewhere in your program. In the example below, the TP10s are all executed in the same task as P_ModbusMain, but the TP10s might just as well be executed somewhere else.

The TP10 function blocks need to know the cycle time to time their modbus messages. The example below determines this cycle time automatically.

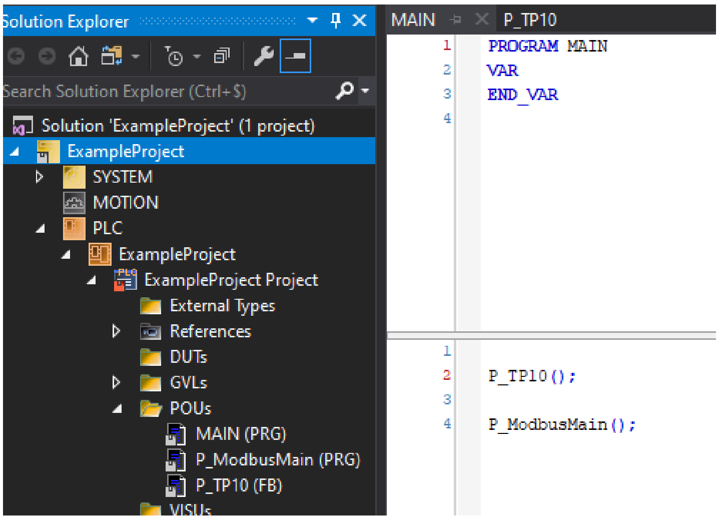


```
VAR_GLOBAL CONSTANT
    gv_MB_TP10_MaxPossible : INT := 32; (*amount of TP10s, minimum 1*)
END_VAR
VAR_GLOBAL
    gv_MB_TP10 : ARRAY[1..gv_MB_TP10_MaxPossible] OF FB_MB_TP10 :=[(*tp10
        (ArrID:=1, BusID:=1, UnitID:=1),
        (ArrID:=2, BusID:=1, UnitID:=2),
        (ArrID:=3, BusID:=1, UnitID:=3),
        (ArrID:=4, BusID:=1, UnitID:=4),
        (ArrID:=5, BusID:=1, UnitID:=5),
        (ArrID:=6, BusID:=1, UnitID:=6),
        (ArrID:=7, BusID:=1, UnitID:=7),
        (ArrID:=8, BusID:=1, UnitID:=8),
        (ArrID:=9, BusID:=1, UnitID:=9),
        (ArrID:=10, BusID:=1, UnitID:=10),
        (ArrID:=11, BusID:=1, UnitID:=11),
        (ArrID:=12, BusID:=1, UnitID:=12),
        (ArrID:=13, BusID:=1, UnitID:=13),
        (ArrID:=14, BusID:=1, UnitID:=14),
        (ArrID:=15, BusID:=1, UnitID:=15),
        (ArrID:=16, BusID:=1, UnitID:=16),
        (ArrID:=17, BusID:=1, UnitID:=17),
        (ArrID:=18, BusID:=1, UnitID:=18),
        (ArrID:=19, BusID:=1, UnitID:=19),
        (ArrID:=20, BusID:=1, UnitID:=20),
        (ArrID:=21, BusID:=1, UnitID:=21),
        (ArrID:=22, BusID:=1, UnitID:=22),
        (ArrID:=23, BusID:=1, UnitID:=23),
        (ArrID:=24, BusID:=1, UnitID:=24),
        (ArrID:=25, BusID:=1, UnitID:=25),
        (ArrID:=26, BusID:=1, UnitID:=26),
        (ArrID:=27, BusID:=1, UnitID:=27),
        (ArrID:=28, BusID:=1, UnitID:=28),
        (ArrID:=29, BusID:=1, UnitID:=29),
        (ArrID:=30, BusID:=1, UnitID:=30),
        (ArrID:=31, BusID:=1, UnitID:=31),
        (ArrID:=32, BusID:=1, UnitID:=32)];
END_VAR
```



```
FUNCTION_BLOCK P_TP10
VAR
    _i : INT; (*iterator*)
    _fSampleT : REAL := 0.003; (*sample time*)
    _fbCurTask : GETCURTASKINDEX; (*determine current task index*)
END_VAR
```

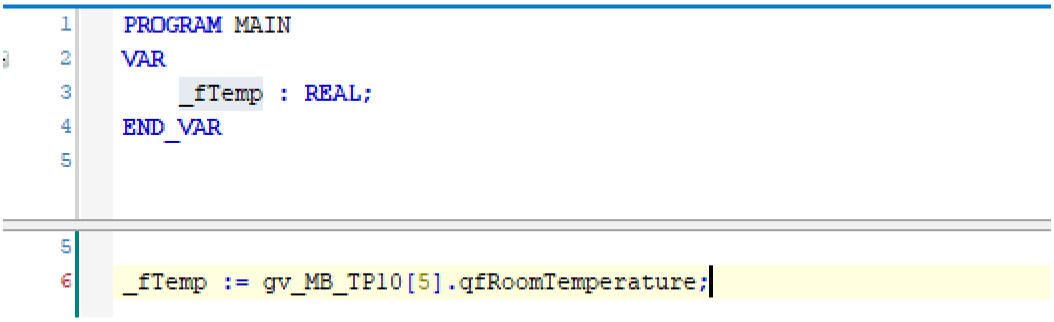
```

(*determine current sample time*)
(*to make this work in TC2, include TcSystem.Lib (for TC V2.8+) or PLCSystem.Lib (for TC V2.7-)*)
(*to make this work in TC3, include Tc2_System, and use "_TaskInfo" instead of "SystemTaskInfoArr"*)
(*if determining the sample time still does not work after this*)
(*just fill in the cycle time of your task manually, this value does not have to be exactly right*)
_fbCurTask();
_fSampleT := UDINT_TO_REAL(_TaskInfo[_fbCurTask.index].cycleTime)/ 10000000.0;

(*loop through all TP10s*)
FOR _i := 1 TO 16 DO
    gv_MB_TP10[_i](
        ArrID := _i,
        fSampleT := _fSampleT);
END_FOR

```



```
PROGRAM MAIN
VAR
END_VAR

```

```

P_TP10();

P_ModbusMain();
```

After doing this, all TP10 functionality will be available everywhere in your project. The example below stores the room temperature measured by TP10 5 to a local variable:



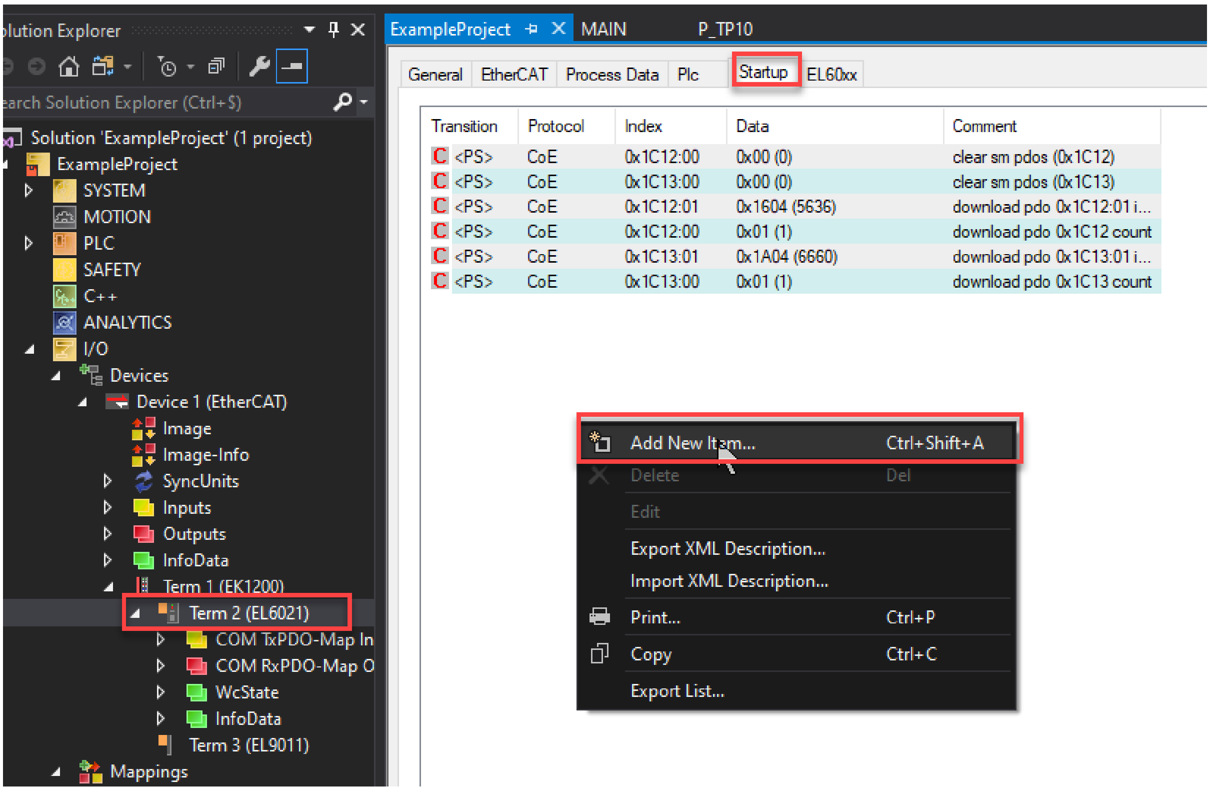
```
PROGRAM MAIN
VAR
    _fTemp : REAL;
END_VAR

```

```

_fTemp := gv_MB_TP10[5].qfRoomTemperature;
```

Step 7: Setting up the I/O

The first step is to scan your I/O in config mode or to add the I/O manually according to your setup.

To use an EL6021 as a Modbus master, a few change should be made to the startup list:

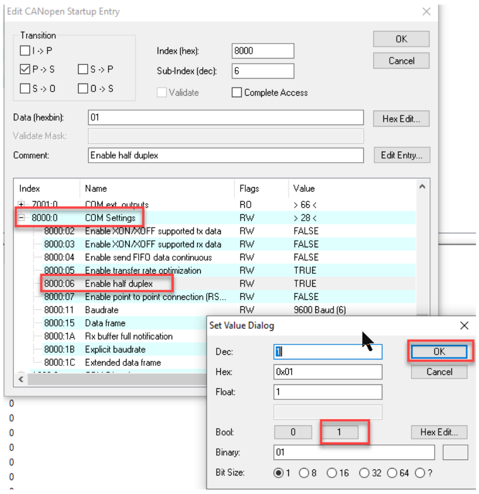


Transition	Protocol	Index	Data	Comment
C <PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
C <PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
C <PS>	CoE	0x1C12:01	0x1604 (5636)	download pdo 0x1C12:01 i...
C <PS>	CoE	0x1C12:00	0x01 (1)	download pdo 0x1C12 count
C <PS>	CoE	0x1C13:01	0x1A04 (6660)	download pdo 0x1C13:01 i...
C <PS>	CoE	0x1C13:00	0x01 (1)	download pdo 0x1C13 count

Half duplex should be changed to TRUE (Index 8000:06 on EL6021), The baudrate should be changed to the right baud rate (usually 9600) (index 8000:11 on EL6021) and the dataframe should be changed to the right dataframe (usually 8E1) (index 8000:15 on EL6021).

Important:

- On an EL6022 terminal you should configure the startup list for both communication channels.
- Do not change any other Com Settings. If you did change other settings, the right com settings are:
    - enable xon/xoff FALSE
    - Enable xon/xoff FALSE (there are 2 settings with the same name)
    - Enable fifo data continuous FALSE
    - Enable data transfer rate optimization TRUE
    - Enable half duplex TRUE
    - Enable point to point connection FALSE



Edit CANopen Startup Entry

Transition: □ I -> P ☑ P -> S □ S -> P □ S -> O □ O -> S

Index (hex): 8000

Sub-Index (dec): 6

□ Validate □ Complete Access

Data (hexbin): 01

Validate Mask:

Comment: Enable half duplex

Index	Name	Flags	Value
7001:0	COM ext. outputs	RO	> 66 <
8000:0	COM Settings	RW	> 28 <
8000:02	Enable XON/XOFF supported tx data	RW	FALSE
8000:03	Enable XON/XOFF supported rx data	RW	FALSE
8000:04	Enable send FIFO data continuous	RW	FALSE
8000:05	Enable transfer rate optimization	RW	TRUE
8000:06	Enable half duplex	RW	TRUE
8000:07	Enable point to point connection (RS...	RW	FALSE
8000:11	Baudrate	RW	9600 Baud (6)
8000:15	Data frame		
8000:1A	Rx buffer full notification		
8000:1B	Explicit baudrate		
8000:1C	Extended data frame		

Set Value Dialog

Dec: 1

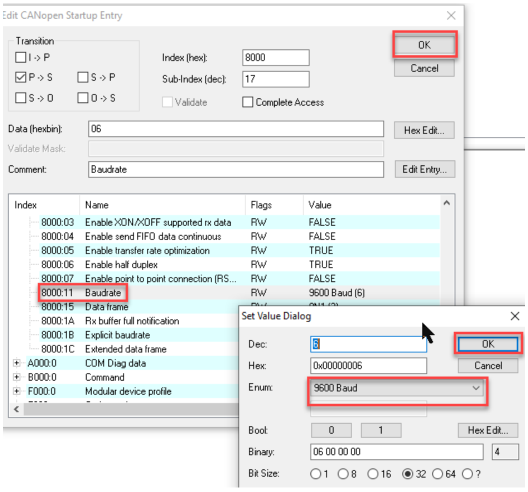
Hex: 0x01

Float: 1

Bool: 0 | 1

Binary: 01

Bit Size: ● 1 ○ 8 ○ 16 ○ 32 ○ 64 ○ ?



Edit CANopen Startup Entry

Transition: □ I -> P ☑ P -> S □ S -> P □ S -> O □ O -> S

Index (hex): 8000

Sub-Index (dec): 17

□ Validate □ Complete Access

Data (hexbin): 06

Validate Mask:

Comment: Baudrate

Index	Name	Flags	Value
8000:03	Enable XON/XOFF supported rx data	RW	FALSE
8000:04	Enable send FIFO data continuous	RW	FALSE
8000:05	Enable transfer rate optimization	RW	TRUE
8000:06	Enable half duplex	RW	TRUE
8000:07	Enable point to point connection (RS...	RW	FALSE
8000:11	Baudrate	RW	9600 Baud (6)
8000:15	Data frame		
8000:1A	Rx buffer full notification		
8000:1B	Explicit baudrate		
8000:1C	Extended data frame		
A000:0	COM Diag data		
B000:0	Command		
F000:0	Modular device profile		

Set Value Dialog

Dec: 6

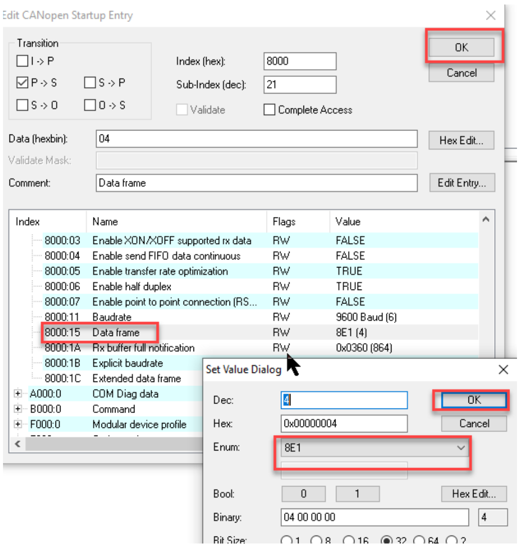
Hex: 0x00000006

Enum: 9600 Baud

Bool: 0 | 1

Binary: 06 00 00 00 | 4

Bit Size: ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?



Edit CANopen Startup Entry

Transition: □ I -> P ☑ P -> S □ S -> P □ S -> O □ O -> S

Index (hex): 8000

Sub-Index (dec): 21

□ Validate □ Complete Access

Data (hexbin): 04

Validate Mask:

Comment: Data frame

Index	Name	Flags	Value
8000:03	Enable XON/XOFF supported rx data	RW	FALSE
8000:04	Enable send FIFO data continuous	RW	FALSE
8000:05	Enable transfer rate optimization	RW	TRUE
8000:06	Enable half duplex	RW	TRUE
8000:07	Enable point to point connection (RS...	RW	FALSE
8000:11	Baudrate	RW	9600 Baud (6)
8000:15	Data frame	RW	8E1 (4)
8000:1A	Rx buffer full notification	RW	0x0360 (864)
8000:1B	Explicit baudrate		
8000:1C	Extended data frame		
A000:0	COM Diag data		
B000:0	Command		
F000:0	Modular device profile		

Set Value Dialog

Dec: 4

Hex: 0x00000004

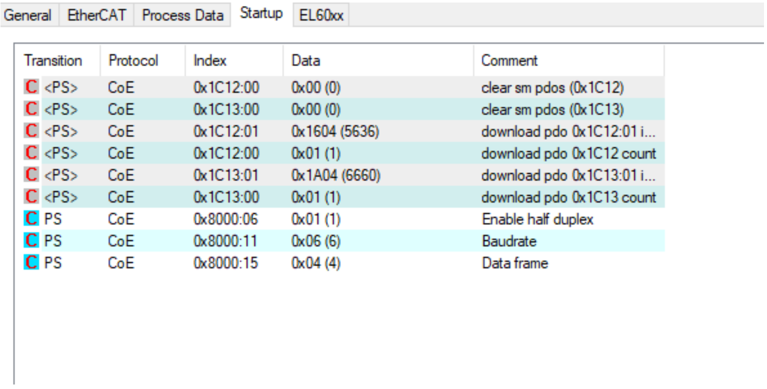
Enum: 8E1

Bool: 0 | 1

Binary: 04 00 00 00 | 4

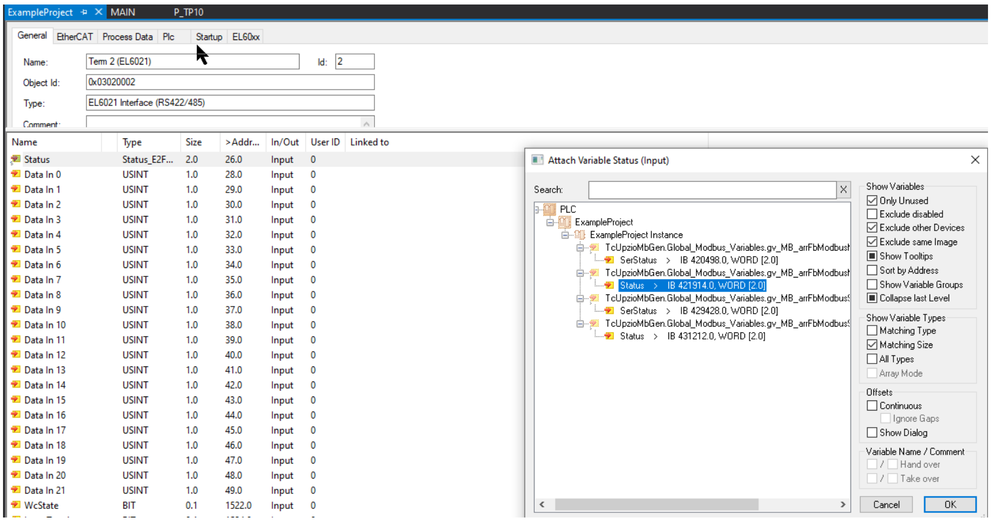
Bit Size: ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

After doing this, the startup list should look like this:



Transition	Protocol	Index	Data	Comment
C <PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
C <PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
C <PS>	CoE	0x1C12:01	0x1604 (5636)	download pdo 0x1C12:01 i...
C <PS>	CoE	0x1C12:00	0x01 (1)	download pdo 0x1C12 count
C <PS>	CoE	0x1C13:01	0x1A04 (6660)	download pdo 0x1C13:01 i...
C <PS>	CoE	0x1C13:00	0x01 (1)	download pdo 0x1C13 count
C PS	CoE	0x8000:06	0x01 (1)	Enable half duplex
C PS	CoE	0x8000:11	0x06 (6)	Baudrate
C PS	CoE	0x8000:15	0x04 (4)	Data frame

The I/O should be linked by linking the status to status, ctrl to ctrl and the D to data To link the data it is possible to select all data inputs/outputs and clicking on “change multi link”:



ExampleProject | MAIN | P_TP10

General | EtherCAT | Process Data | Plc | Startup | EL60xx

Name: Term 2 (EL6021) Id: 2

Object Id: 0x03020002

Type: EL6021 Interface (RS422/485)

Comment:

Name	Type	Size	>Addr...	In/Out	User ID	Linked to
Status	Status_E2F...	2.0	26.0	Input	0	
Data In 0	USINT	1.0	28.0	Input	0	
Data In 1	USINT	1.0	29.0	Input	0	
Data In 2	USINT	1.0	30.0	Input	0	
Data In 3	USINT	1.0	31.0	Input	0	
Data In 4	USINT	1.0	32.0	Input	0	
Data In 5	USINT	1.0	33.0	Input	0	
Data In 6	USINT	1.0	34.0	Input	0	
Data In 7	USINT	1.0	35.0	Input	0	
Data In 8	USINT	1.0	36.0	Input	0	
Data In 9	USINT	1.0	37.0	Input	0	
Data In 10	USINT	1.0	38.0	Input	0	
Data In 11	USINT	1.0	39.0	Input	0	
Data In 12	USINT	1.0	40.0	Input	0	
Data In 13	USINT	1.0	41.0	Input	0	
Data In 14	USINT	1.0	42.0	Input	0	
Data In 15	USINT	1.0	43.0	Input	0	
Data In 16	USINT	1.0	44.0	Input	0	
Data In 17	USINT	1.0	45.0	Input	0	
Data In 18	USINT	1.0	46.0	Input	0	
Data In 19	USINT	1.0	47.0	Input	0	
Data In 20	USINT	1.0	48.0	Input	0	
Data In 21	USINT	1.0	49.0	Input	0	
WcState	BIT	0.1	1522.0	Input	0	

Attach Variable Status (Input)

Search:

```
PLC
  ExampleProject
    ExampleProject Instance
      TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusI
        SerStatus  >  IB 420498.0, WORD [2.0]
      TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusI
        Status  >  IB 421914.0, WORD [2.0]
      TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusS
        SerStatus  >  IB 429428.0, WORD [2.0]
      TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusS
        Status  >  IB 431212.0, WORD [2.0]
```

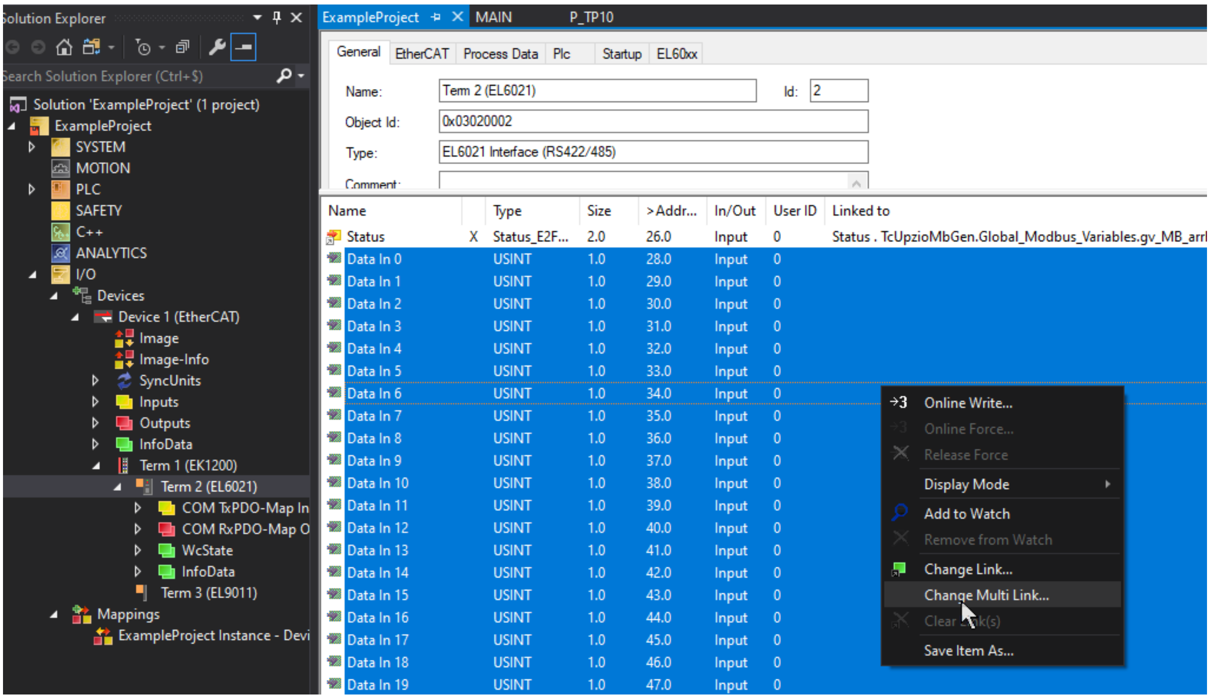
Show Variables
- ☑ Only Unused
- □ Exclude disabled
- ☑ Exclude other Devices
- ☑ Exclude same Image
- ■ Show Tooltips
- □ Sort by Address
- □ Show Variable Groups
- ■ Collapse last Level

Show Variable Types
- □ Matching Type
- ☑ Matching Size
- □ All Types
- □ Array Mode

Offsets
- □ Continuous
- □ Ignore Gaps
- □ Show Dialog

Variable Name / Comment
- □ / □ Hand over
- □ / □ Take over

Cancel | OK



Solution Explorer

Search Solution Explorer (Ctrl+$)

```
Solution 'ExampleProject' (1 project)
  ExampleProject
    SYSTEM
    MOTION
    PLC
    SAFETY
    C++
    ANALYTICS
    I/O
      Devices
        Device 1 (EtherCAT)
          Image
          Image-Info
          SyncUnits
          Inputs
          Outputs
          InfoData
          Term 1 (EK1200)
            Term 2 (EL6021)
              COM TxPDO-Map In
              COM RxPDO-Map O
              WcState
              InfoData
            Term 3 (EL9011)
      Mappings
        ExampleProject Instance - Devi
```

ExampleProject | MAIN | P_TP10

General | EtherCAT | Process Data | Plc | Startup | EL60xx

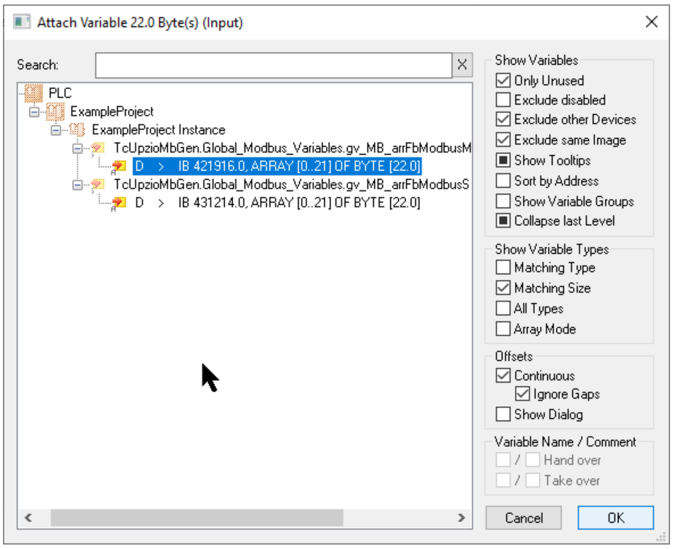
Name: Term 2 (EL6021) Id: 2

Object Id: 0x03020002

Type: EL6021 Interface (RS422/485)

Comment:

Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status	X	Status_E2F...	2.0	26.0	Input	0	Status . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrl
Data In 0		USINT	1.0	28.0	Input	0	
Data In 1		USINT	1.0	29.0	Input	0	
Data In 2		USINT	1.0	30.0	Input	0	
Data In 3		USINT	1.0	31.0	Input	0	
Data In 4		USINT	1.0	32.0	Input	0	
Data In 5		USINT	1.0	33.0	Input	0	
Data In 6		USINT	1.0	34.0	Input	0	
Data In 7		USINT	1.0	35.0	Input	0	
Data In 8		USINT	1.0	36.0	Input	0	
Data In 9		USINT	1.0	37.0	Input	0	
Data In 10		USINT	1.0	38.0	Input	0	
Data In 11		USINT	1.0	39.0	Input	0	
Data In 12		USINT	1.0	40.0	Input	0	
Data In 13		USINT	1.0	41.0	Input	0	
Data In 14		USINT	1.0	42.0	Input	0	
Data In 15		USINT	1.0	43.0	Input	0	
Data In 16		USINT	1.0	44.0	Input	0	
Data In 17		USINT	1.0	45.0	Input	0	
Data In 18		USINT	1.0	46.0	Input	0	
Data In 19		USINT	1.0	47.0	Input	0	



Attach Variable 22.0 Byte(s) (Input)

Search:

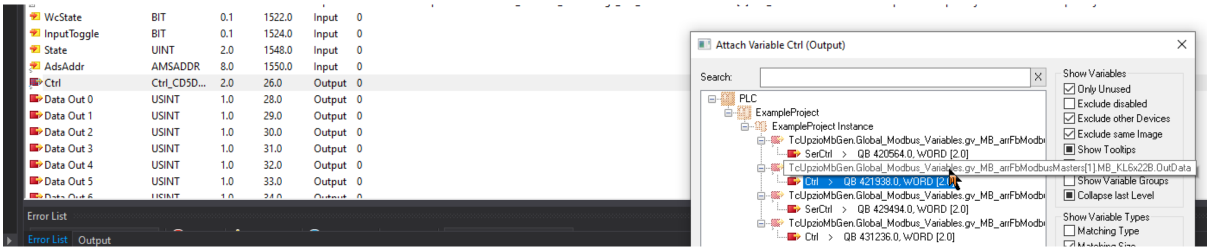
- PLC
  - ExampleProject
    - ExampleProject Instance
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusM
        - D > IB 421916.0, ARRAY [0..21] OF BYTE [22.0]
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusS
        - D > IB 431214.0, ARRAY [0..21] OF BYTE [22.0]

Show Variables
☑ Only Unused
☐ Exclude disabled
☑ Exclude other Devices
☑ Exclude same Image
■ Show Tooltips
☐ Sort by Address
☐ Show Variable Groups
■ Collapse last Level

Show Variable Types
☐ Matching Type
☑ Matching Size
☐ All Types
☐ Array Mode

Offsets
☑ Continuous
☑ Ignore Gaps
☐ Show Dialog

Variable Name / Comment
☐ / ☐ Hand over
☐ / ☐ Take over



WcState	BIT	0.1	1522.0	Input	0
InputToggle	BIT	0.1	1524.0	Input	0
State	UINT	2.0	1548.0	Input	0
AdsAddr	AMSADDR	8.0	1550.0	Input	0
Ctrl	Ctrl_CD5D...	2.0	26.0	Output	0
Data Out 0	USINT	1.0	28.0	Output	0
Data Out 1	USINT	1.0	29.0	Output	0
Data Out 2	USINT	1.0	30.0	Output	0
Data Out 3	USINT	1.0	31.0	Output	0
Data Out 4	USINT	1.0	32.0	Output	0
Data Out 5	USINT	1.0	33.0	Output	0
Data Out 6	USINT	1.0	34.0	Output	0

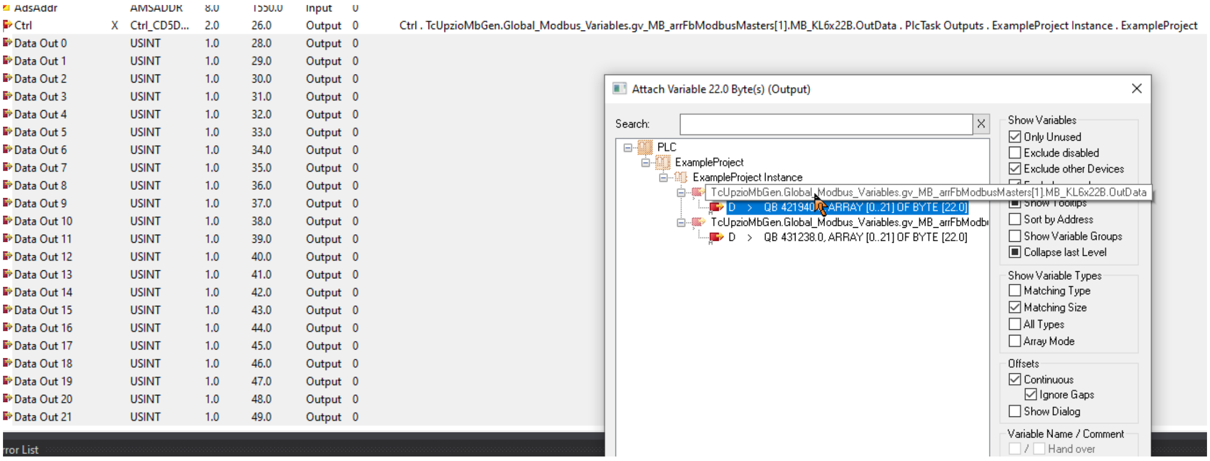
Attach Variable Ctrl (Output)

Search:

- PLC
  - ExampleProject
    - ExampleProject Instance
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbı
        - SerCtrl > QB 420564.0, WORD [2.0]
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData
        - Ctrl > QB 421938.0, WORD [2.0]
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbı
        - SerCtrl > QB 429494.0, WORD [2.0]
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbı
        - Ctrl > QB 431236.0, WORD [2.0]

Show Variables
☑ Only Unused
☐ Exclude disabled
☑ Exclude other Devices
☑ Exclude same Image
■ Show Tooltips
☐ Show Variable Groups
■ Collapse last Level

Show Variable Types
☐ Matching Type



AdsAddr		AMSADDR	8.0	1550.0	Input	0	
Ctrl	X	Ctrl_CD5D...	2.0	26.0	Output	0	Ctrl . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 0		USINT	1.0	28.0	Output	0	
Data Out 1		USINT	1.0	29.0	Output	0	
Data Out 2		USINT	1.0	30.0	Output	0	
Data Out 3		USINT	1.0	31.0	Output	0	
Data Out 4		USINT	1.0	32.0	Output	0	
Data Out 5		USINT	1.0	33.0	Output	0	
Data Out 6		USINT	1.0	34.0	Output	0	
Data Out 7		USINT	1.0	35.0	Output	0	
Data Out 8		USINT	1.0	36.0	Output	0	
Data Out 9		USINT	1.0	37.0	Output	0	
Data Out 10		USINT	1.0	38.0	Output	0	
Data Out 11		USINT	1.0	39.0	Output	0	
Data Out 12		USINT	1.0	40.0	Output	0	
Data Out 13		USINT	1.0	41.0	Output	0	
Data Out 14		USINT	1.0	42.0	Output	0	
Data Out 15		USINT	1.0	43.0	Output	0	
Data Out 16		USINT	1.0	44.0	Output	0	
Data Out 17		USINT	1.0	45.0	Output	0	
Data Out 18		USINT	1.0	46.0	Output	0	
Data Out 19		USINT	1.0	47.0	Output	0	
Data Out 20		USINT	1.0	48.0	Output	0	
Data Out 21		USINT	1.0	49.0	Output	0	

Attach Variable 22.0 Byte(s) (Output)

Search:

- PLC
  - ExampleProject
    - ExampleProject Instance
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData
        - D > QB 421940.0, ARRAY [0..21] OF BYTE [22.0]
      - TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbı
        - D > QB 431238.0, ARRAY [0..21] OF BYTE [22.0]

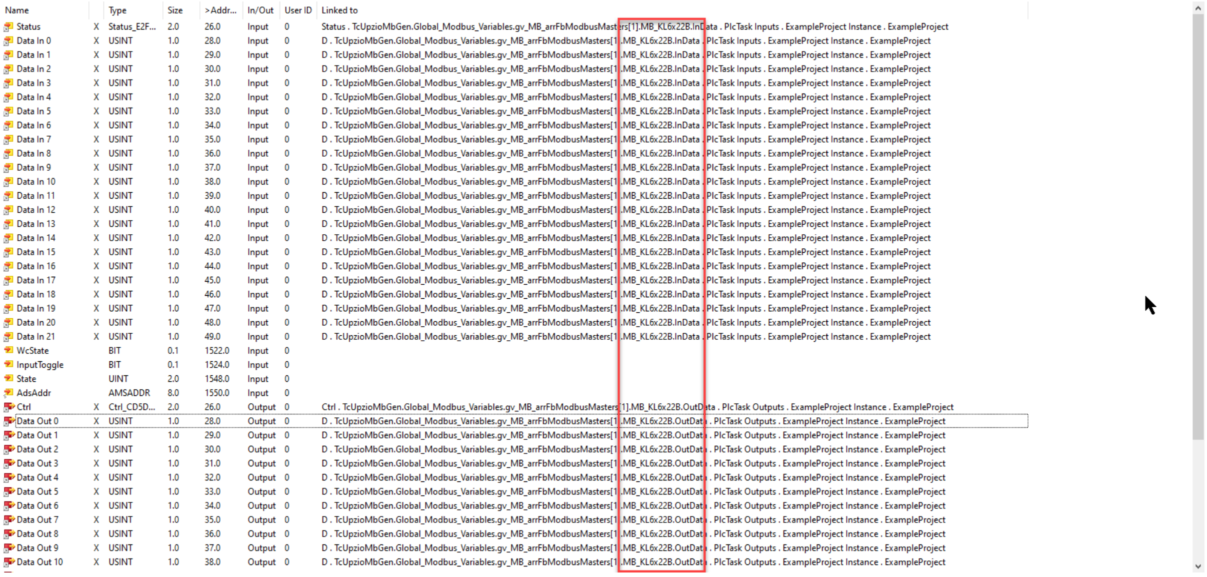
Show Variables
☑ Only Unused
☐ Exclude disabled
☑ Exclude other Devices
■ Show Tooltips
☐ Sort by Address
☐ Show Variable Groups
■ Collapse last Level

Show Variable Types
☐ Matching Type
☑ Matching Size
☐ All Types
☐ Array Mode

Offsets
☑ Continuous
☑ Ignore Gaps
☐ Show Dialog

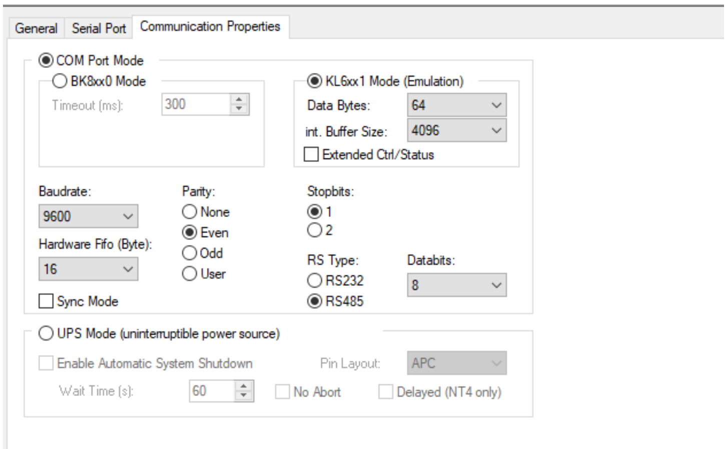
Variable Name / Comment
☐ / ☐ Hand over

Make sure you always link MB_KL6x22B when using an EL6021:



Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status	X	Status_E2F...	2.0	26.0	Input	0	Status . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 0	X	USINT	1.0	28.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 1	X	USINT	1.0	29.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 2	X	USINT	1.0	30.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 3	X	USINT	1.0	31.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 4	X	USINT	1.0	32.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 5	X	USINT	1.0	33.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 6	X	USINT	1.0	34.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 7	X	USINT	1.0	35.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 8	X	USINT	1.0	36.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 9	X	USINT	1.0	37.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 10	X	USINT	1.0	38.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 11	X	USINT	1.0	39.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 12	X	USINT	1.0	40.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 13	X	USINT	1.0	41.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 14	X	USINT	1.0	42.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 15	X	USINT	1.0	43.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 16	X	USINT	1.0	44.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 17	X	USINT	1.0	45.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 18	X	USINT	1.0	46.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 19	X	USINT	1.0	47.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 20	X	USINT	1.0	48.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
Data In 21	X	USINT	1.0	49.0	Input	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.InData . PlcTask Inputs . ExampleProject Instance . ExampleProject
WcState		BIT	0.1	1522.0	Input	0	
InputToggle		BIT	0.1	1524.0	Input	0	
State		UINT	2.0	1548.0	Input	0	
AdsAddr		AMSADDR	8.0	1550.0	Input	0	
Ctrl	X	Ctrl_CD5D...	2.0	26.0	Output	0	Ctrl . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 0	X	USINT	1.0	28.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 1	X	USINT	1.0	29.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 2	X	USINT	1.0	30.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 3	X	USINT	1.0	31.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 4	X	USINT	1.0	32.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 5	X	USINT	1.0	33.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 6	X	USINT	1.0	34.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 7	X	USINT	1.0	35.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 8	X	USINT	1.0	36.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 9	X	USINT	1.0	37.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject
Data Out 10	X	USINT	1.0	38.0	Output	0	D . TcUpzioMbGen.Global_Modbus_Variables.gv_MB_arrFbModbusMasters[1].MB_KL6x22B.OutData . PlcTask Outputs . ExampleProject Instance . ExampleProject

If the serial communication port is used, the communication properties of the port must be set as shown below. Change the baud rate, parity and stop bits according to your setup.



● COM Port Mode
○ BK8xx0 Mode
Timeout (ms): 300
● KL6xx1 Mode (Emulation)
Data Bytes: 64
int. Buffer Size: 4096
□ Extended Ctrl/Status

Baudrate: 9600
Hardware Fifo (Byte): 16
□ Sync Mode

Parity:
○ None
● Even
○ Odd
○ User

Stopbits:
● 1
○ 2

RS Type:
○ RS232
● RS485

Databits: 8

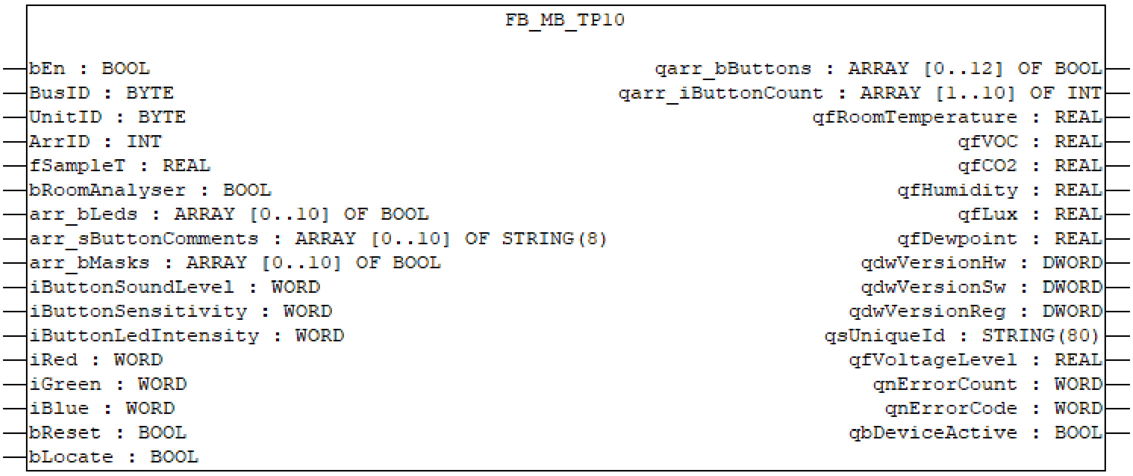
○ UPS Mode (uninterruptible power source)
□ Enable Automatic System Shutdown
Pin Layout: APC
Wait Time (s): 60
□ No Abort
□ Delayed (NT4 only)

Linking the serial com port is similar to linking the EL6021, but this time, make sure you always link MB_PcCom when using a serial com port.

Overview function block inputs/outputs

Overview

The in- and outputs of the FB_MB_TP10 are shown in the image below. More details about the in and outputs are described in the table below.



```
                                  FB_MB_TP10

bEn : BOOL                                    qarr_bButtons : ARRAY [0..12] OF BOOL
BusID : BYTE                               qarr_iButtonCount : ARRAY [1..10] OF INT
UnitID : BYTE                                           qfRoomTemperature : REAL
ArrID : INT                                                          qfVOC : REAL
fSampleT : REAL                                                      qfCO2 : REAL
bRoomAnalyser : BOOL                                            qfHumidity : REAL
arr_bLeds : ARRAY [0..10] OF BOOL                                    qfLux : REAL
arr_sButtonComments : ARRAY [0..10] OF STRING(8)                qfDewpoint : REAL
arr_bMasks : ARRAY [0..10] OF BOOL                            qdwVersionHw : DWORD
iButtonSoundLevel : WORD                                      qdwVersionSw : DWORD
iButtonSensitivity : WORD                                    qdwVersionReg : DWORD
iButtonLedIntensity : WORD                               qsUniqueId : STRING(80)
iRed : WORD                                                qfVoltageLevel : REAL
iGreen : WORD                                                qnErrorCount : WORD
iBlue : WORD                                                  qnErrorCode : WORD
bReset : BOOL                                               qbDeviceActive : BOOL
bLocate : BOOL
```

Details inputs

Name	Type	Default value	Description
bEn	BOOL	TRUE	Enable function block
BusID	BYTE	1	Master id for modbus
UnitID	BYTE	247	Slave id for modbus
ArrID	INT	0	Id in the TP10 array, only indicational, has no purpose
fSampleT	REAL	0.012	Cycle time of this function block in seconds
bRoomAnalyser	BOOL	FALSE	Set this to false if this is a TP10, set this to true if this is a roomanalyser. Setting this to true will disable the button readout and will reduce the amount of modbus messages significantly.
arr_bLeds	ARRAY[0..10] OF BOOL	FALSE	Used to turn on/off the button leds. (e.g.: arr_bLeds[2] := TRUE; will turn on the 2nd button led)
arr_sButtonComments	ARRAY[0..10] OF STRING(8)		Comments for every button, only indicational, has no purpose
arr_bMasks	ARRAY[0..10] OF BOOL	FALSE	Used to turn off the touch buttons. (e.g.: arr_bMasks[4] := TRUE; will turn off the 4th touch button.)
iButtonSoundLevel	WORD	255	Intensity of the sound when pressing a button (0..255)

iButtonSensitivity	WORD	55	Button sensitivity (1..100). Lower values will give a higher sensitivity, higher values will give a lower sensitivity.
iButtonLedIntensity	WORD	128	Intensity of the white leds (0..255)
iRed	WORD	0	Red value of the RGB button leds. (0..255)
iGreen	WORD	0	Green value of the RGB button leds (0..255)
iBlue	WORD	0	Blue value of the RGB button leds (0..255)
bReset	BOOL	FALSE	Resets the TP10 on a rising edge
bLocate	BOOL	FALSE	If true, the TP10 will go to 'locate mode' and start flashing in every color

Details outputs

Name	Type	Description
qarr_bButtons	ARRAY [0..12] OF BOOL	Button statuses, True if a button is being pressed. (e.g.: qarr_bButtons[3] means button 3 is being pressed)
qarr_iButtonCount	ARRAY[1..10] OF INT	Button counters counts the number of times a button was pressed
qfRoomTemperature	REAL	Measured room temperature in °C
qfVOC	REAL	Measured VOC in ppb
qfCO2	REAL	Measured CO2 in ppm
qfHumidity	REAL	Measured relative humidity in %
qfLux	REAL	Measured illumination in lux
qfDewpoint	REAL	Measured dewpoint in °C
qdwVersoinHw	DWORD	Hardware version, this unsigned integer represents a date. In the 'yyyyMMdd' format (e.g.: 2020-11-24 would be 20201124)
qdwVersionSw	DWORD	Firmware version, this unsigned integer represents a date. In the 'yyyyMMdd' format (e.g.: 2020-11-24 would be 20201124)
qdwVersionReg	DWORD	Register version, this unsigned integer represents a date. In the 'yyyyMMdd' format (e.g.: 2020-11-24 would be 20201124)
qsUniqueId	STRING	Unique id of the device
qfVoltageLevel	REAL	Voltage level of the power supply in V
qnErrorCount	WORD	Error count read from the device
qnErrorCode	WORD	Error code read from the device
qbDeviceActive	BOOL	True if modbus communication with the device is okay