



P2M2HBVL/P2H IO-LINK
FUNCTION BLOCK FOR SIEMENS
PLC S7-300 (TIA-PORTAL) WITH
TURCK TBEN-L4/5-8IOL
PROFINET IO-LINK MASTER
QUICK START GUIDE

CONTENTS

PREFACE.....	3
1. Setting P2M\P2H Slave with TBEN-L4/5-8IOL TURCK IO-Link Master	4
2. Importing function block library.....	5
3. Process Data Function Block Instruction.....	7
3.1. Add instance of function block	7
3.2. Using the instruction	10
3.2.1. Solenoid Valves Control (Cyclic).....	10
3.2.2. Monitoring Status Bits (Cyclic)	11
4. Parameter Data Example Code	12
4.2. Add instance of function block	12
4.3. Using the instruction	13
4.3.1. Reading parameters.	13
4.3.2. Writing Parameters.	16
APPENDIX.....	20



PREFACE

This Quick Start Guide (QSG) is designed to help integrate Parker Hannifin's P2M or P2H IO-link valve manifold into a Siemens PLC environment using TURCK TBEN-L4/5-8IOL Profinet IO-Link Master module. The QSG assumes that you are already using the TURCK network interface Profinet and that it is connecting and configured to the Siemens PLC S7-300 via a Profinet network in Tia portal environment.

The QSG is agnostic to IO Link Device Classification, such that it shall function the same whether you are controlling an A-Class or B-Class P2M / P2H Module. The guide will walk the user through obtaining the necessary files, importing/configuring the FB's, and initiating parameter reads and writes from/to the P2M / P2H IO-Link device.

The "P2M_IOL_TBEN-L45-8IOL_PD" function block facilitates communication and handling of process data between PLC and the IO-Link slave device.

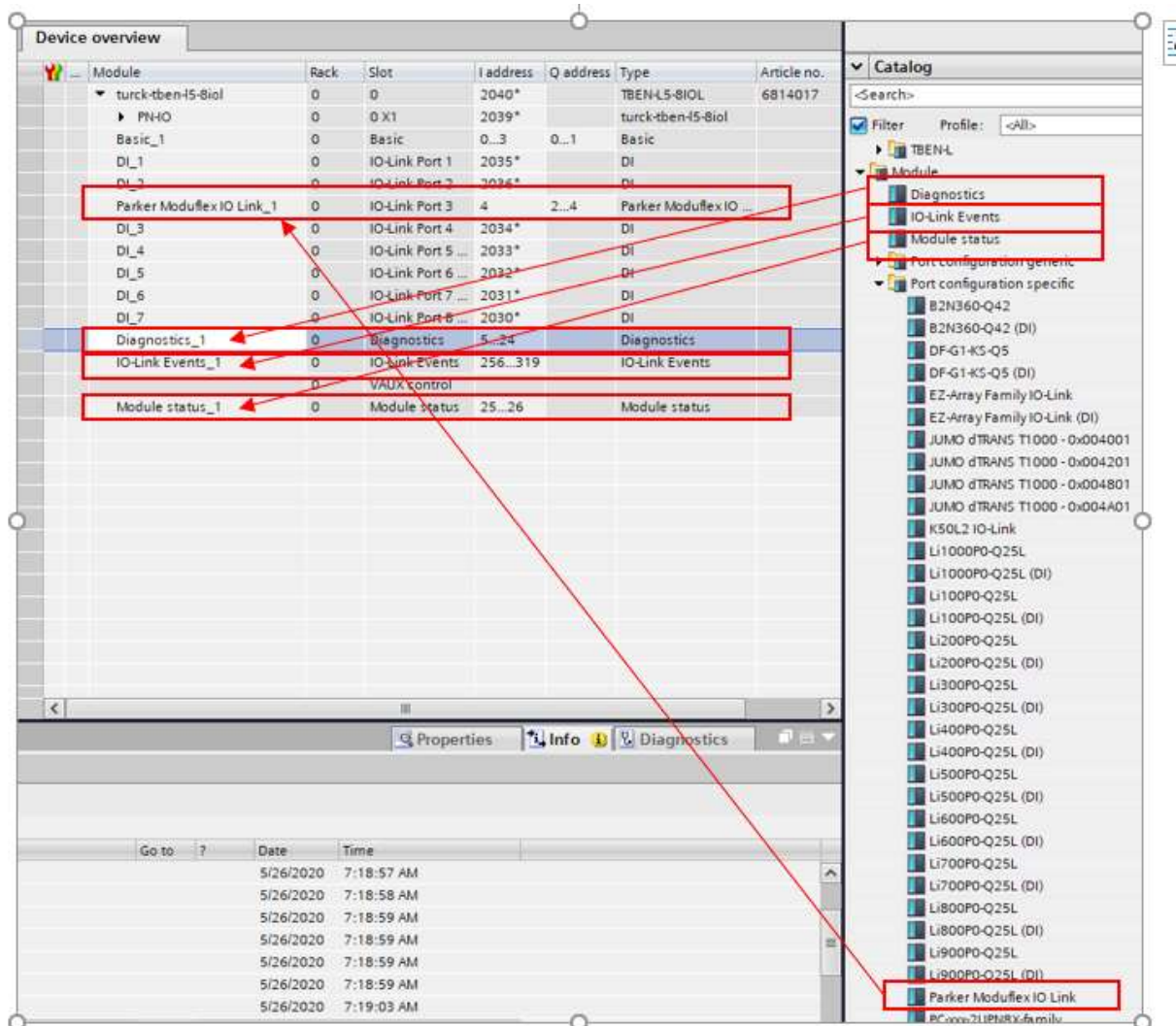
The "P2M_IOL_TBEN-L45-8IOL_PRM" function block facilitates the call-up of the acyclic service data.

You can download resources such as the IODD configuration file, this QSG, a sample TIA portal file "P2M2HBVL_P2HL_TIA_S7300_TBEN-L45-8IOL_R0" and the full P2M manual here:

<http://www.parker.com/pdn/io-Link>

1. SETTING P2M\P2H SLAVE WITH TBEN-L4/5-8IOL TURCK IO-LINK MASTER

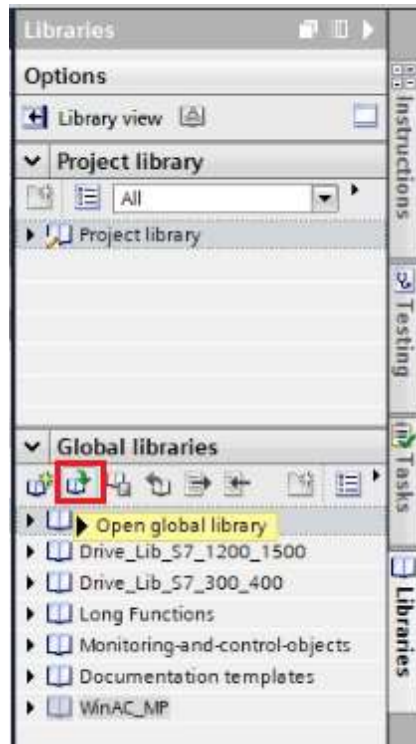
In the project tree double click “Devices & networks” move to “Network view” double click the TURCK IO-Link master, drag and drop “**Parker Moduflex IO Link**”, “**Diagnostics**”, “**IO-Link Events**” and “**Module Status**” from Hardware catalog >> Module to “Device overview”.



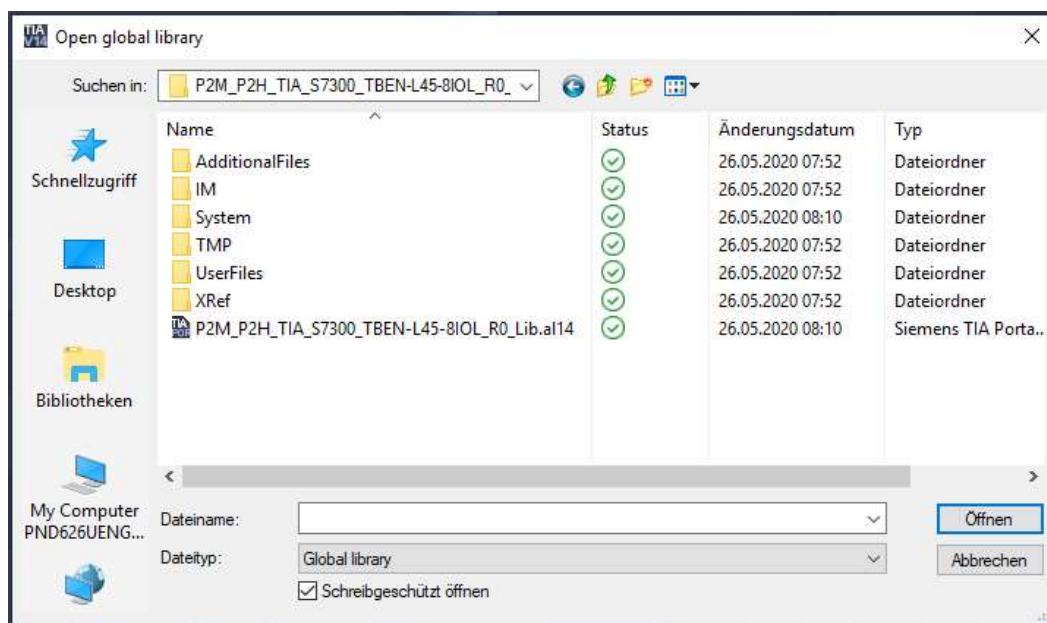
(note: Data from “Diagnostics” and “IO-Link Events” are not used in this guide).

2. IMPORTING FUNCTION BLOCK LIBRARY

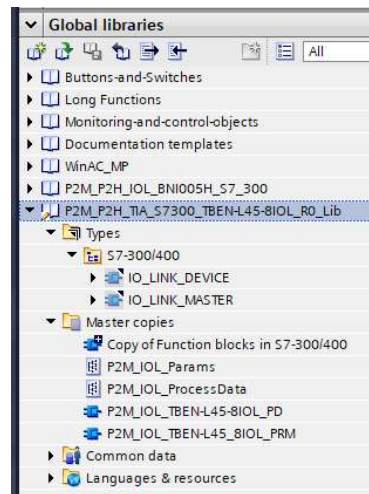
In Tia portal environment at right side, open the tab “Global Libraries” by clicking on “Libraries”, then select “open global library”.



In “Open global Library” view move to your stored Library function block folder “P2M_P2H_TIA_S7300_TBEN-L45-8IOL_R0_Lib” and select “.al14” file then click **open**



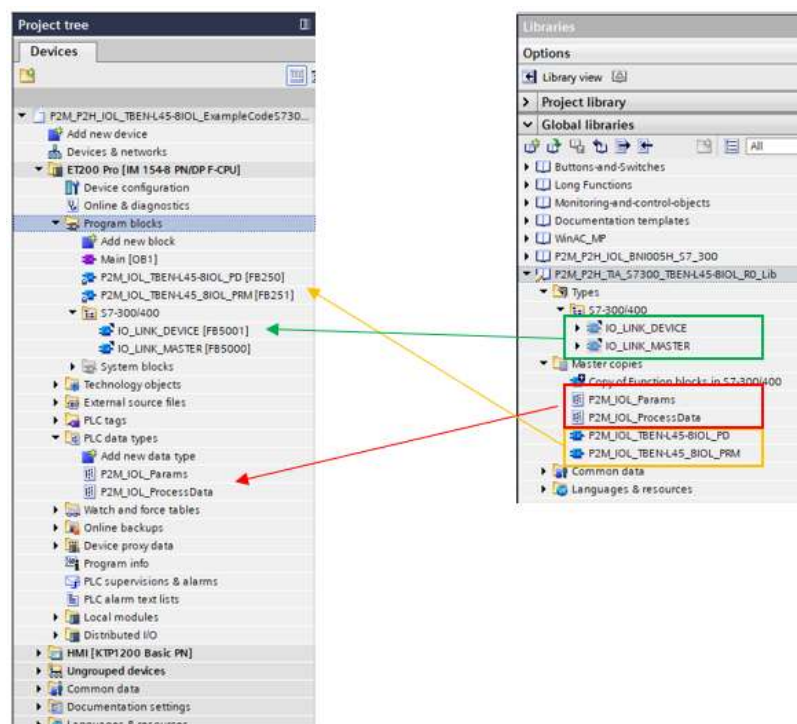
You should then see the new Library “P2M_P2H_IOL_TBEN-L45-8IOL_S7_300” in the Global Libraries list, click on your imported Library title to view your function block from drop-down menu.



Drag and drop type data “P2M_IOL_Params” and the “P2M_IOL_ProcessData” from Global libraries to the project tree “PLC data types” folder.

Then drag and drop function block “P2M_IOL_TBEN-L45-8IOL_PD” and “P2M_IOL_TBEN-L45-8IOL_PRM” to the project tree folder “Program blocks”.

Drag and drop also IO-Link system functions “IO_Link_Device” and “IO_Link_Master” to folder “Program blocks” → “S7-300/400”

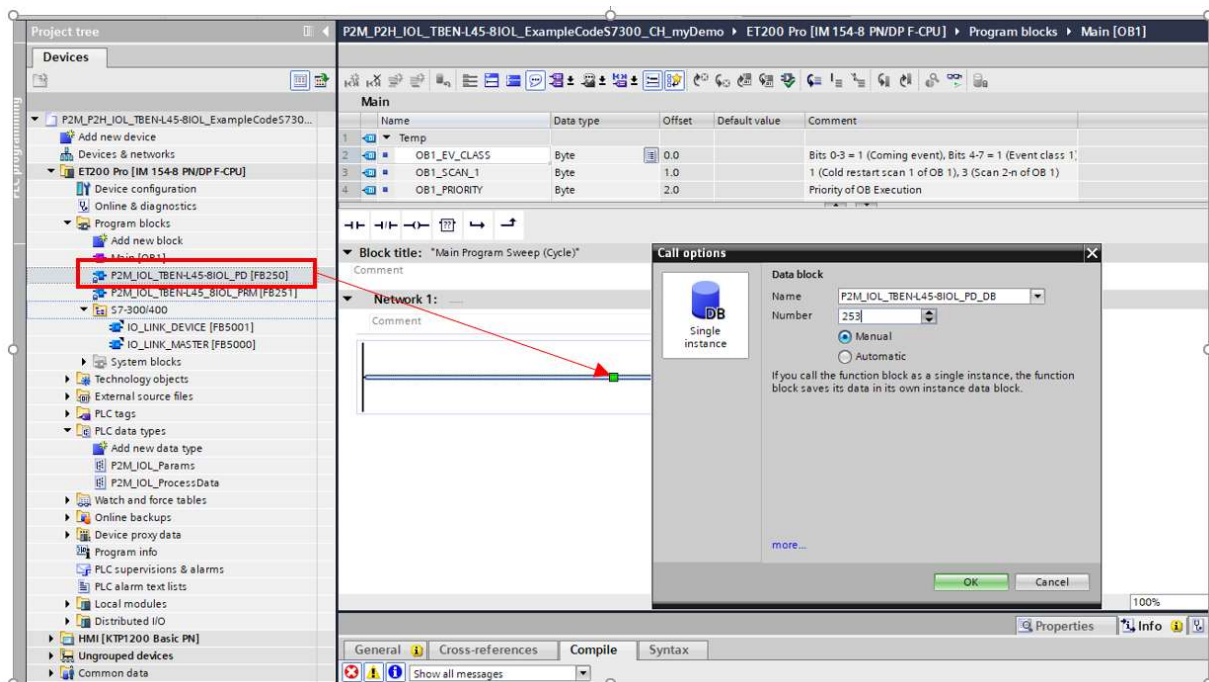


3. PROCESS DATA FUNCTION BLOCK INSTRUCTION

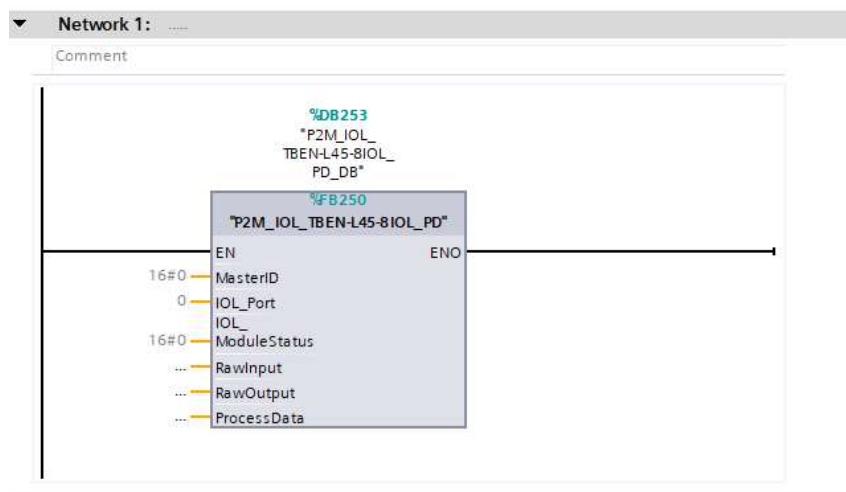
The “P2M_IOL_TBEN-L45-8IOL_PD” function block simplifies the usage of Parker P2M and P2H IO-link devices with Siemens S7-300/400 PLCs when connected via Profinet to TURCK TBEN-L4/5-8IOL IO-Link Master. Data is mapped to user-friendly control and diagnostic tags on the PLC side.

3.1. Add instance of function block

Add instance of instruction to an empty Network of ladder, drag and drop the function block “P2M_IOL_TBEN-L45-8IOL_PD” onto the selected Network, create data instance associated with function block then click Ok.

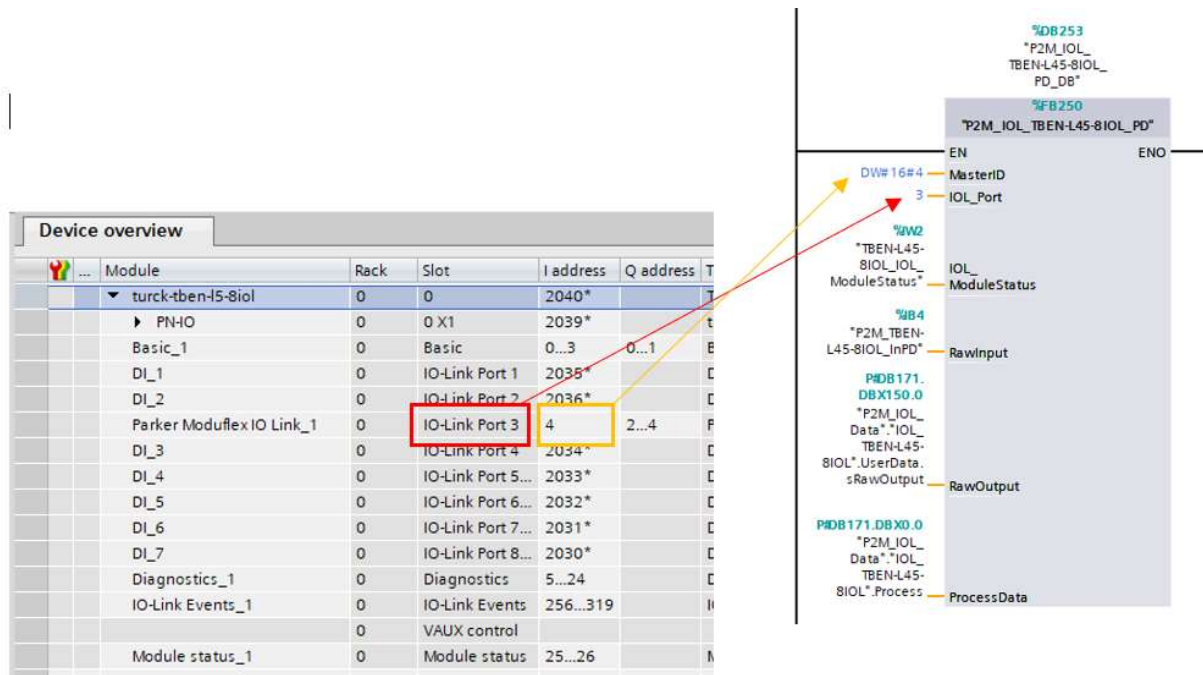


function block view with default data in the Network

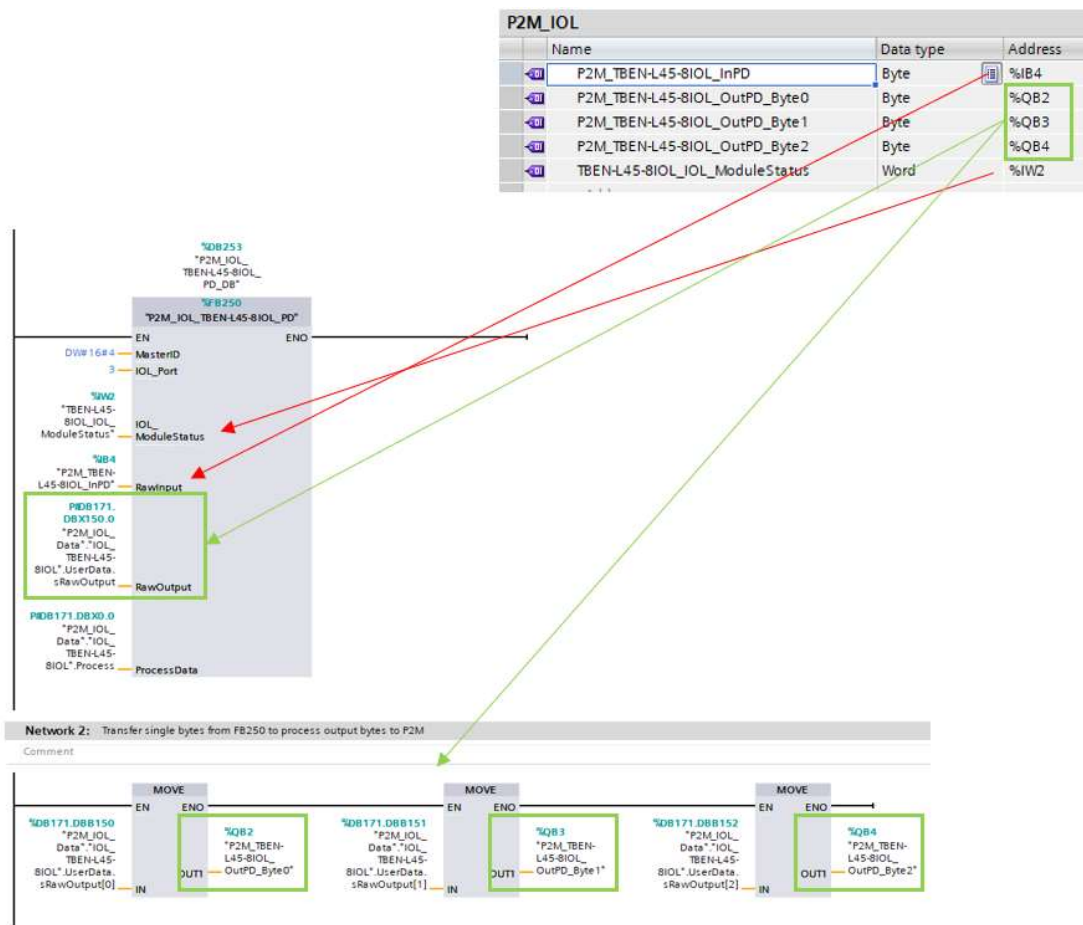


Read **MasterID** (convert from decimal to hexadecimal) and **IOL_Port** from “Devices & networks” > “Device Overview”. **The IOL Port is dependent on the physical port the P2M / P2H is connected to!**

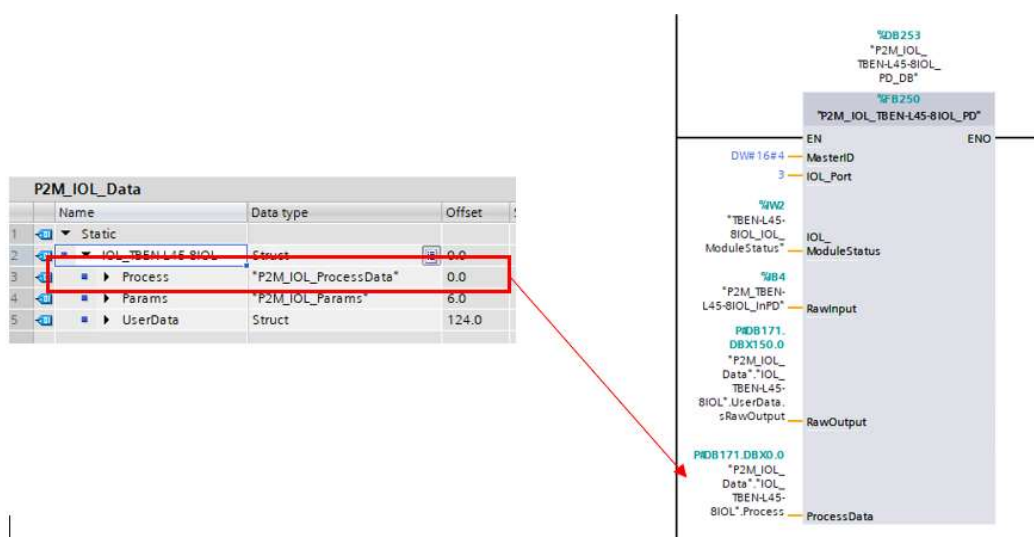
If you choose the incorrect port the function block will not work, and undesired results are likely. IO-Link is sensitive to the port which it has been assigned to communicate on.



Point IOL_ModuleStatus, RawInput and RawOutput fields to the “PLC Tags” associated with IO-Link Basic module that the P2M / P2H is connected to.



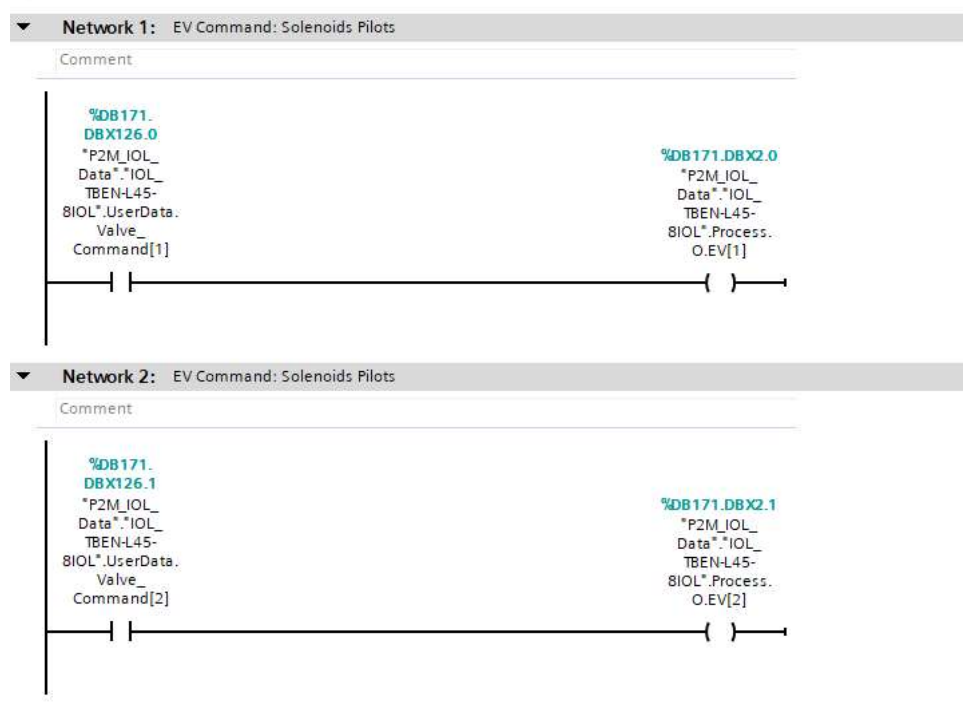
Point “ProcessData” to a variable type “P2M_IOL_ProcessData” created in a user data block



3.2. Using the instruction

It is important to note the difference between cyclic and acyclic data. Process Data (cyclic) is updated without a request; whereas Parameter Data (acyclic) requires the program to toggle a bit to pull data. Cyclic data includes input module status, and valve output control. This means that created variable “P2M_IOL_Data”.PD.I.xxx and “P2M_IOL_Data”.PD.O.EV[xx] are live tags with real data just by calling the Function Block. See appendix for all data points available. See ladder logic examples below:

3.2.1. Solenoid Valves Control (Cyclic)



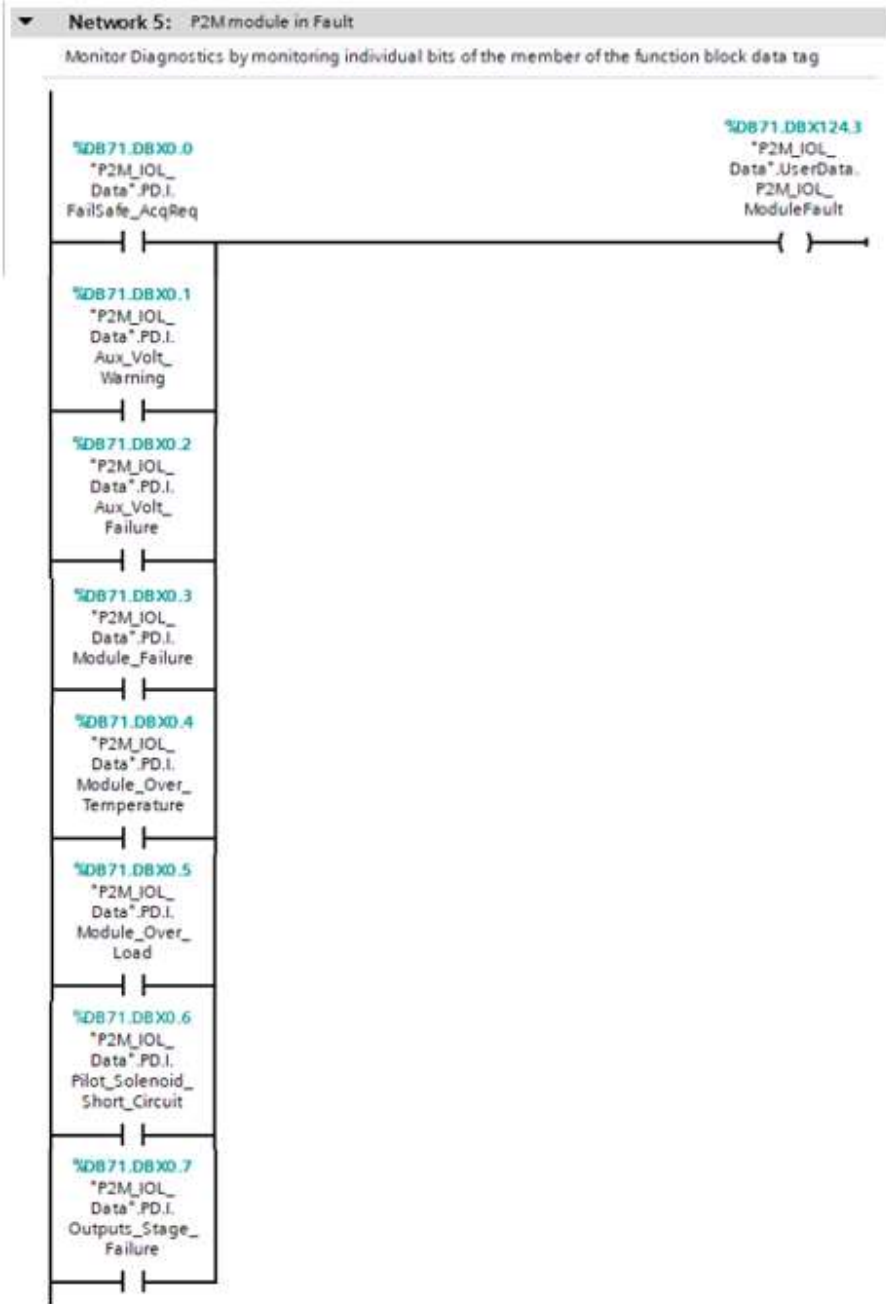
Example command

With the “P2M_IOL_Data.IOL_TBEN-L45-8IOL'.UserData.Valve_Command[x] you can control each valve by a single bit.

Value correlation to solenoids address	
“P2M_IOL_Data”.UserData.Valve_Command[1] = 1	“P2M_IOL_Data”.PD.O.EV[1]=1
“P2M_IOL_Data”.UserData.Valve_Command[2] = 1	“P2M_IOL_Data”.PD.O.EV[2]=1
“P2M_IOL_Data”.UserData.Valve_Command[3] = 1	“P2M_IOL_Data”.PD.O.EV[3]=1
..	..
“P2M_IOL_Data”.UserData.Valve_Command[24] = 1	“P2M_IOL_Data”.PD.O.EV[24]=1



3.2.2. Monitoring Status Bits (Cyclic)



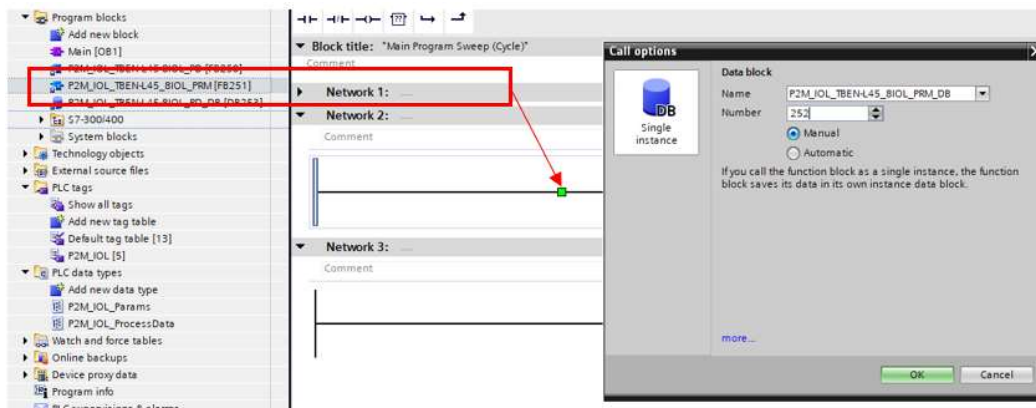
4. PARAMETER DATA EXAMPLE CODE

Note:** It is important to not initiate multiple read or write requests at the same time. Write your logic such that only one of the request bits is turned on at a time and wait for the done bit to go high before executing the next request.

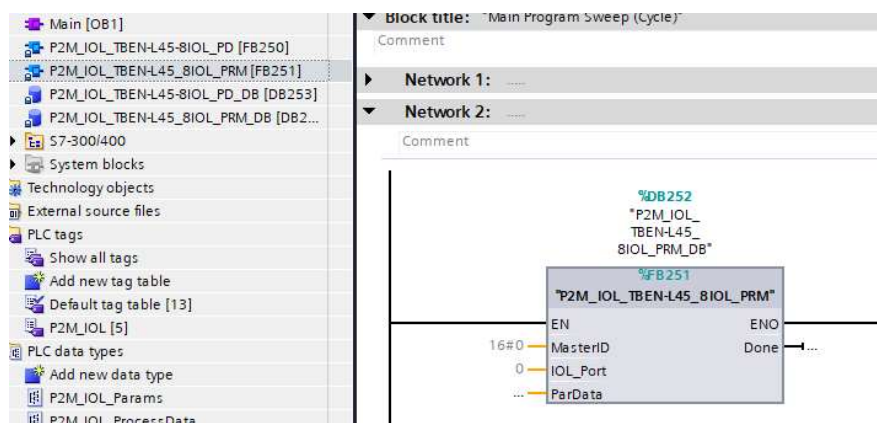
The “P2M_IOL_TBEN-L45-8IOL_PRM” function block simplifies the usage of Parker P2M and P2H IO-link devices with Siemens S7-300/400 PLCs when connected via Profinet to TURCK TBEN-L4/5-8IOL IO-Link Master. The Function block facilitates the read/write acyclic data between the PLC and the Parker P2M or P2H IO-Link device. We assume that the Global Libraries importing steps are already done by following the described steps above.

4.2. Add instance of function block

From “Program Blocks” Drag and drop the function block “P2M_IOL_TBEN-L4/5-8IOL_PRM” to add an instance of instruction to an empty network of ladder

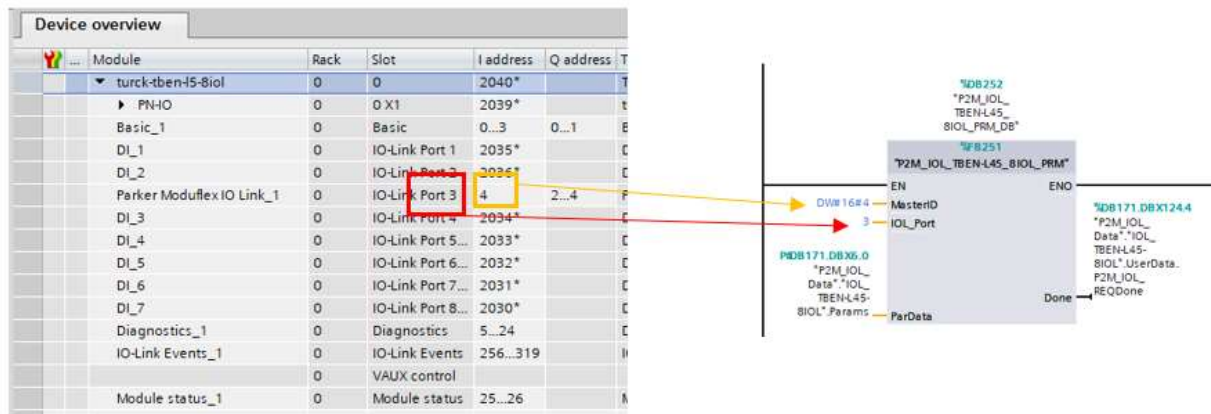


Assign a data block to the instance, define the name and the number. View the create instance below with default data.

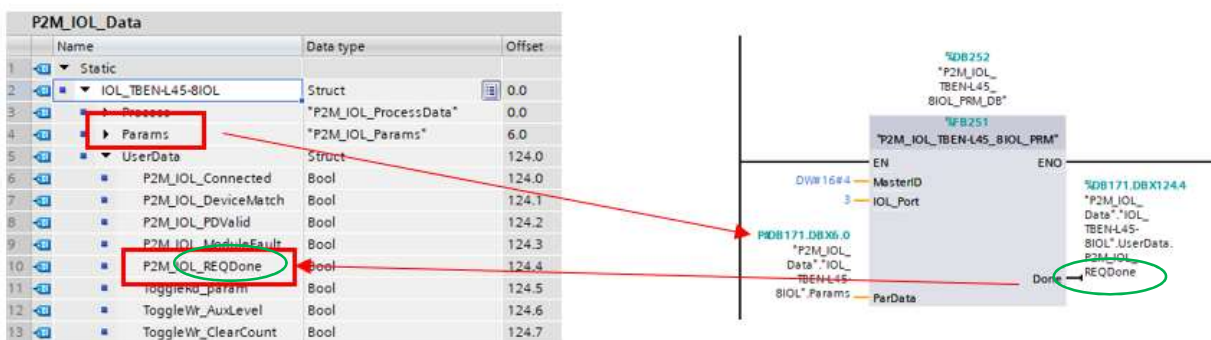


1. Create Tags necessary for operation. Read and set MasterID (convert from decimal to hexadecimal) and IOL_Port from “Devices & networks” > “Device Overview”. The **IOL_Port** is **dependent on the physical port the P2M / P2H is connected to!**

If you choose the incorrect port the function block will not work, and undesired results are likely. IO-Link is sensitive to the port which it has been assigned to communicate on.



Point "ParData" to a variable type "P2M_IOL_Params" created in a user data block and "Done" to a variable type BOOL created in an user data block.

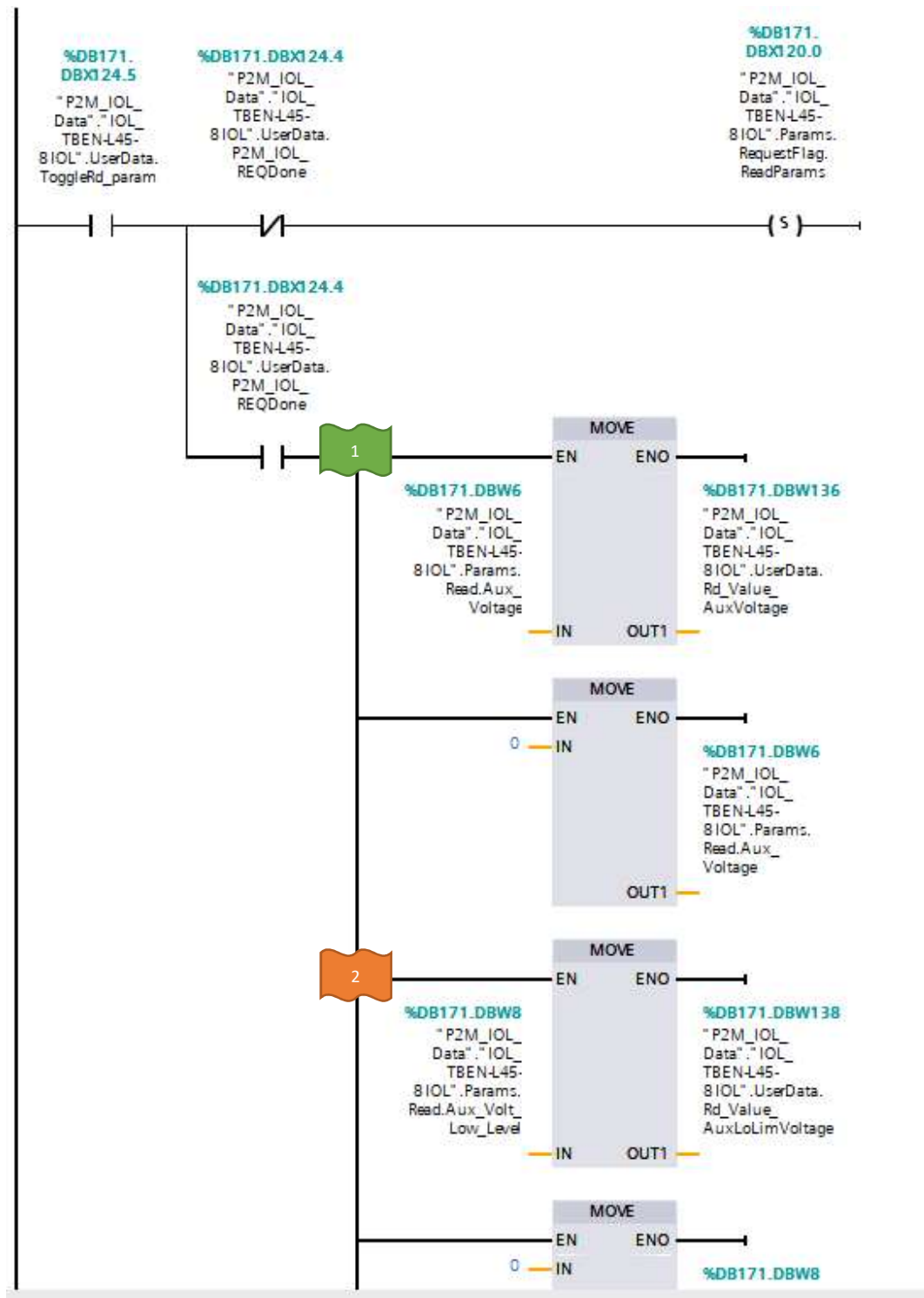


4.3. Using the instruction

The parameter data(acyclic) requires the program to toggle a bit to read or write data contained inside the slave device.

4.3.1. Reading parameters.

see below for logic example



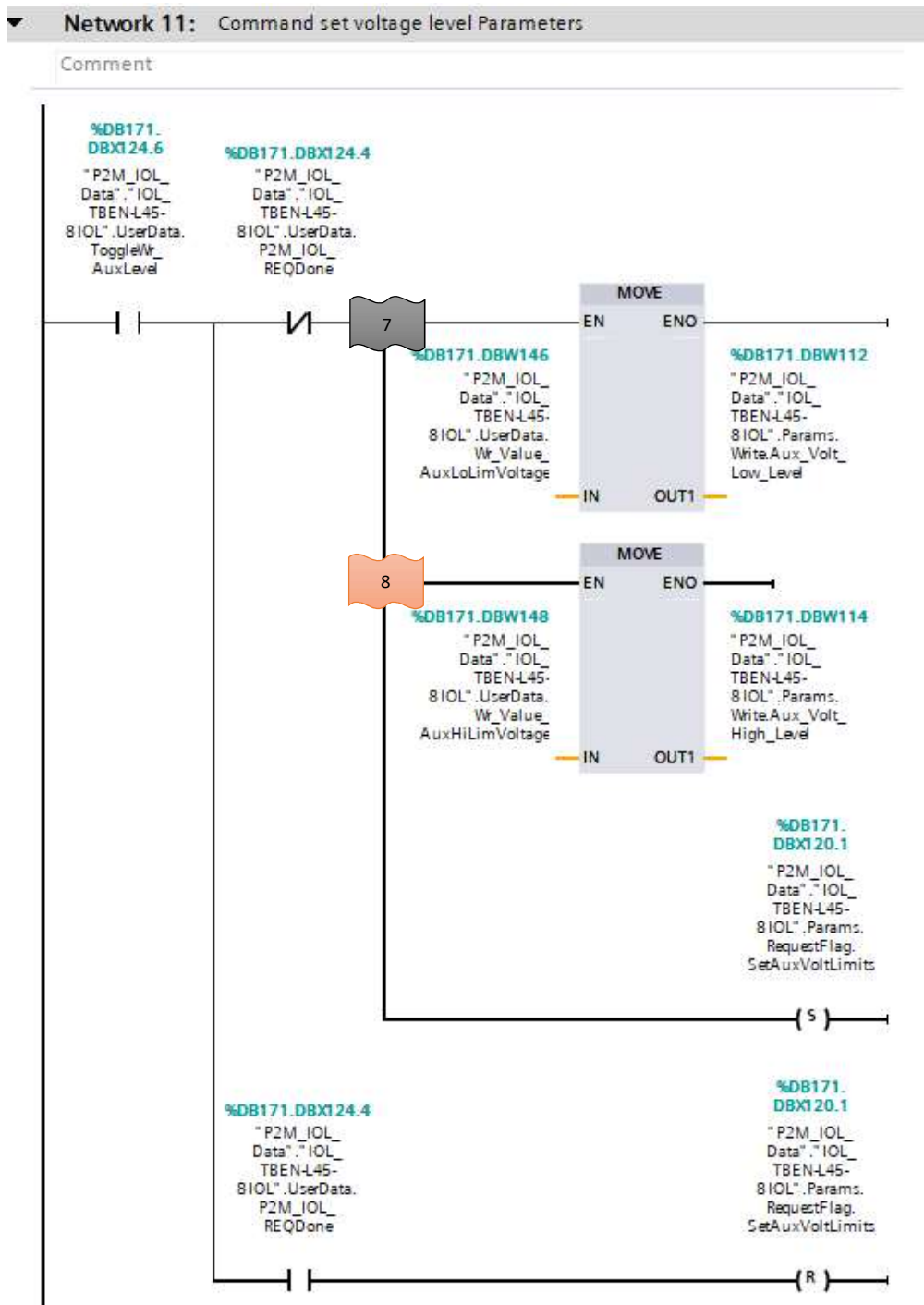


Description	Index	Destination Element
Read Auxiliary Voltage	67	P2M_IOL_Data.PRM.Read.AuxVoltage
Read Auxiliary Voltage Low Limit Warning	70	P2M_IOL_Data.PRM.Aux_Volt_Low_Lev
Read Auxiliary Voltage High Limit Warning		P2M_IOL_Data.PRM.Aux_Volt_High_Lev
Read Channel Errors	66	P2M_IOL_Data.PRM.Channel_Error
Read Cycle Counters	64	P2M_IOL_Data.PRM.Switching_Cycle

4.3.2. Writing Parameters.

Writing Auxiliary Voltage Low and High Limits, clearing cycle counters example, send systems command.

Note: **When sending new limit setpoints to the P2M module, the values will not be written unless the high limit is more than one volt greater than the low limit, and greater than zero.**



Network 13: Command Clear switching Cycle count

NOTE: Bit Correlation: Valve Command Bits and Cycle Counter Clear Bits doesn't correlate in 1:1 relation:

Means: Valve Commands correlate to DB171.DBD 2:

DB171.DBB 2 → V01-V08

DB171.DBB 3 → V09-V16

DB171.DBB 4 → V17-V24

DB171.DBB 5 → not occupied

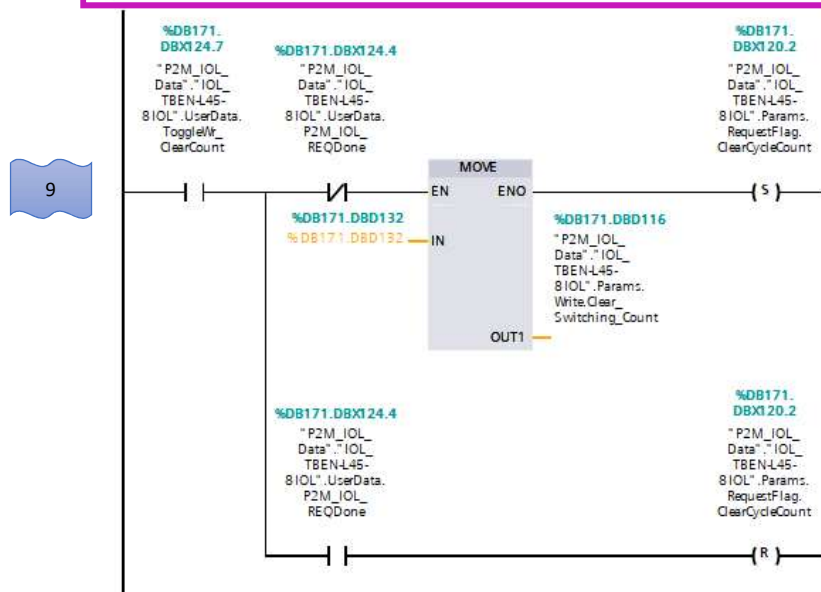
Cycle Counter Commands correlate to: DB171.DBD 116

DB171.DBB 116 → not occupied

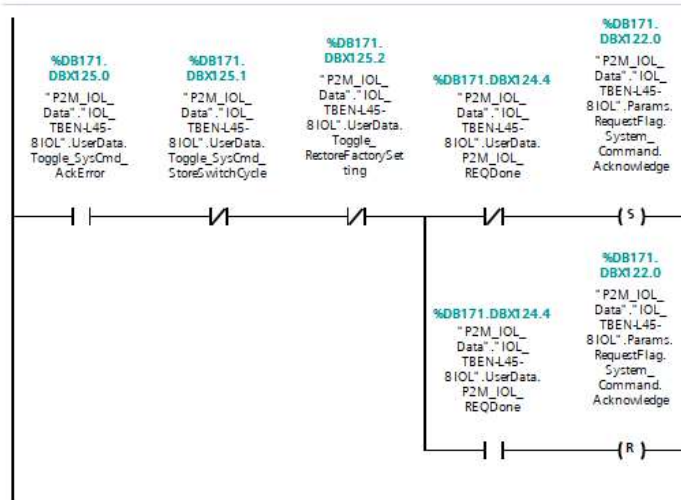
DB171.DBB 117 → C17-C24

DB171.DBB 118 → C09-C16

DB171.DBB 119 → C01-C08

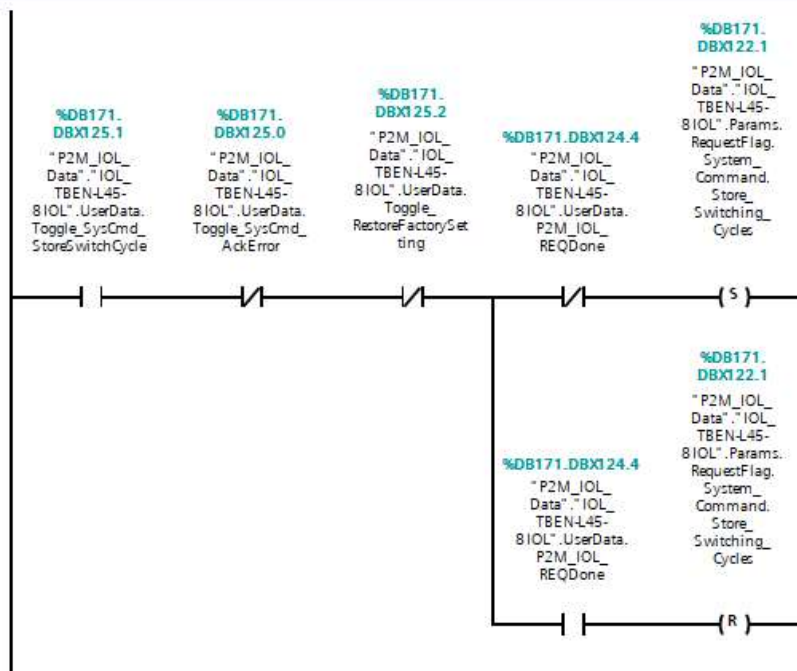

Network 15: Acknowledge Error

Comment

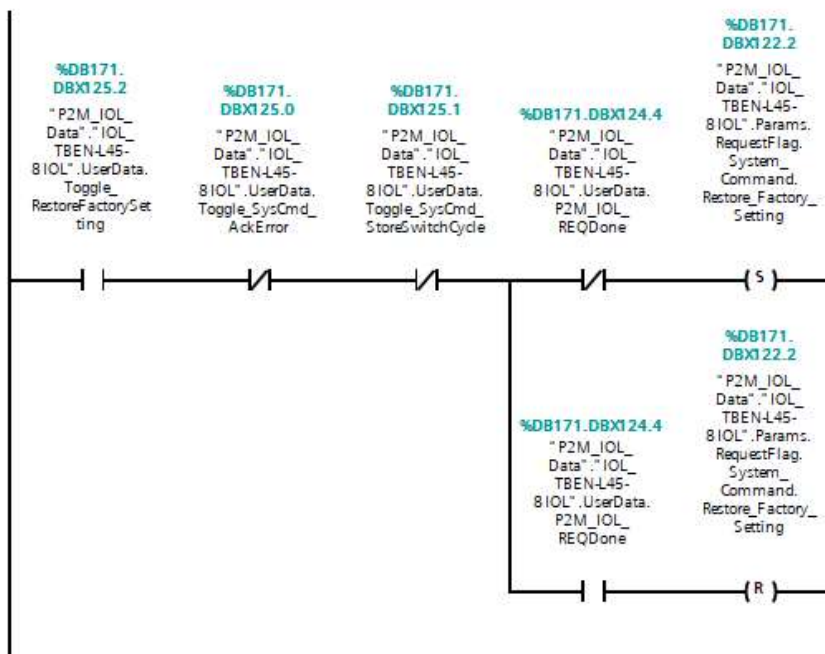


Network 17: The current Value of the switching Cycle Counters are stored into EEPROM.

Comment

**Network 19:** Reset Parameters to default Values

Comment

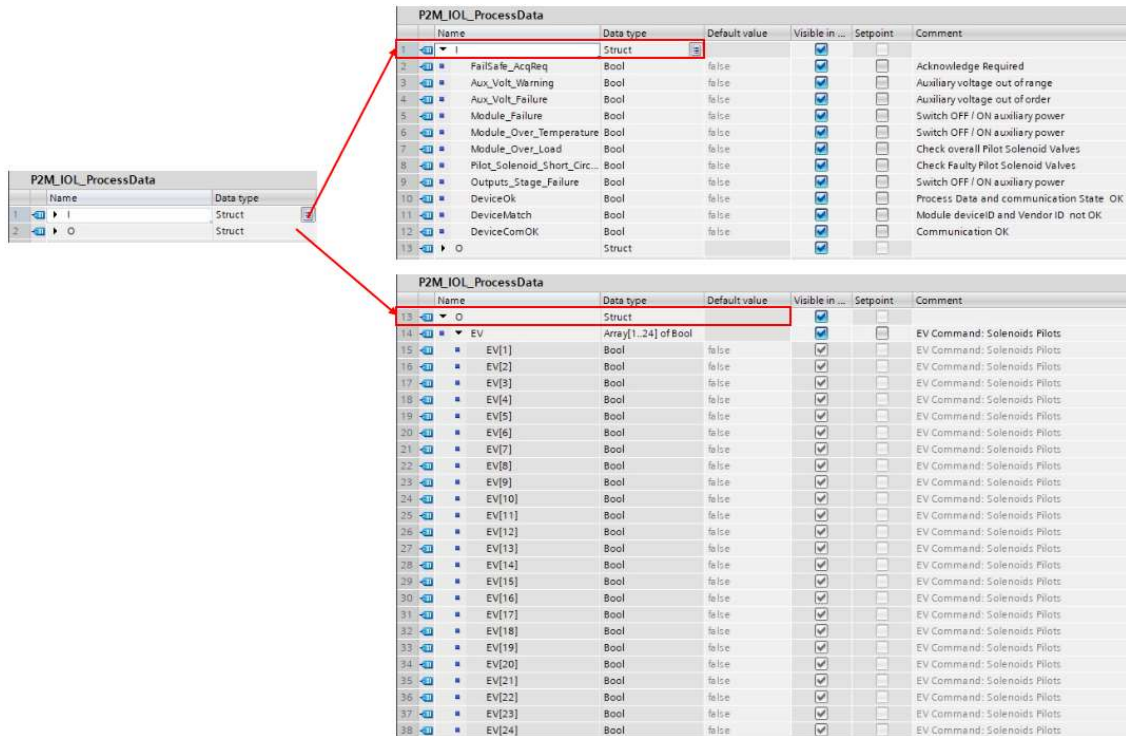


	Description	Index	Source Element		Source Length (bytes)
7	Write Auxiliary Voltage Low Limit Warning	70	P2M_IOL_Data.PRM.Write.Aux_Volt_Low_Level		4
8	Write Auxiliary Voltage High Limit Warning		P2M_IOL_Data.PRM.Write.Aux_Volt_High_Level		Note**: first 16bits=Low level, second 16bits=high level
9	Clear Cycle Counters	65	P2M_IOL_Data.PRM.Write.Clear_Switching_Count		4
	System Command	2	P2M_IOL_Data.PRM.RequestFlag.System_Command: Note: **By sending a system command request these values are written		1
			Command Value	Command Send	
			160	Acknowledge	
			161	Store Switching Cycles	
			130	Restore Factory Setting	

APPENDIX

- **Process Data Structures:** User Defined Data Structures utilized by FB “P2M_IOL_TBEN-L45-8IOL_PD”

From project tree\PLC click on “PLC data types” then double click to open “P2M_IOL_ProcessData” data types.



P2M_IOL_ProcessData						
	Name	Data type	Default value	Visible in ...	Setpoint	Comment
1	I	Struct		☒	☐	
2	FailSafe_AcqReq	Bool	false	☒	☐	Acknowledge Required
3	Aux_Volt_Warning	Bool	false	☒	☐	Auxiliary voltage out of range
4	Aux_Volt_Failure	Bool	false	☒	☐	Auxiliary voltage out of order
5	Module_Failure	Bool	false	☒	☐	Switch OFF / ON auxiliary power
6	Module_Over_Temperature	Bool	false	☒	☐	Switch OFF / ON auxiliary power
7	Module_Over_Load	Bool	false	☒	☐	Check overall Pilot Solenoid Valves
8	Pilot_Solenoid_Short_Circ...	Bool	false	☒	☐	Check Faulty Pilot Solenoid Valves
9	Outputs_Stage_Failure	Bool	false	☒	☐	Switch OFF / ON auxiliary power
10	DeviceOk	Bool	false	☒	☐	Process Data and communication State OK
11	DeviceMatch	Bool	false	☒	☐	Module deviceID and Vendor ID not OK
12	DeviceComOK	Bool	false	☒	☐	Communication OK
13	O	Struct		☒	☐	

P2M_IOL_ProcessData						
	Name	Data type	Default value	Visible in ...	Setpoint	Comment
14	O	Struct		☒	☐	
15	EV	Array[1..24] of Bool		☒	☐	EV Command: Solenoids Pilots
16	EV[1]	Bool	false	☒	☐	EV Command: Solenoids Pilots
17	EV[2]	Bool	false	☒	☐	EV Command: Solenoids Pilots
18	EV[3]	Bool	false	☒	☐	EV Command: Solenoids Pilots
19	EV[4]	Bool	false	☒	☐	EV Command: Solenoids Pilots
20	EV[5]	Bool	false	☒	☐	EV Command: Solenoids Pilots
21	EV[6]	Bool	false	☒	☐	EV Command: Solenoids Pilots
22	EV[7]	Bool	false	☒	☐	EV Command: Solenoids Pilots
23	EV[8]	Bool	false	☒	☐	EV Command: Solenoids Pilots
24	EV[9]	Bool	false	☒	☐	EV Command: Solenoids Pilots
25	EV[10]	Bool	false	☒	☐	EV Command: Solenoids Pilots
26	EV[11]	Bool	false	☒	☐	EV Command: Solenoids Pilots
27	EV[12]	Bool	false	☒	☐	EV Command: Solenoids Pilots
28	EV[13]	Bool	false	☒	☐	EV Command: Solenoids Pilots
29	EV[14]	Bool	false	☒	☐	EV Command: Solenoids Pilots
30	EV[15]	Bool	false	☒	☐	EV Command: Solenoids Pilots
31	EV[16]	Bool	false	☒	☐	EV Command: Solenoids Pilots
32	EV[17]	Bool	false	☒	☐	EV Command: Solenoids Pilots
33	EV[18]	Bool	false	☒	☐	EV Command: Solenoids Pilots
34	EV[19]	Bool	false	☒	☐	EV Command: Solenoids Pilots
35	EV[20]	Bool	false	☒	☐	EV Command: Solenoids Pilots
36	EV[21]	Bool	false	☒	☐	EV Command: Solenoids Pilots
37	EV[22]	Bool	false	☒	☐	EV Command: Solenoids Pilots
38	EV[23]	Bool	false	☒	☐	EV Command: Solenoids Pilots
39	EV[24]	Bool	false	☒	☐	EV Command: Solenoids Pilots

- **Parameter Data Structures:** User Defined Data Structures utilized by FB “P2M_IOL_TBEN-L45-8IOL_PRM”

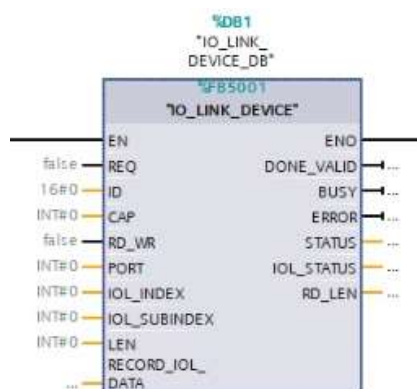
From project tree\PLC click on “PLC data types” then double click to open “P2M_IOL_Params” data types.

P2M_IOL_Params								
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Read	Struct		☒	☒	☒	☐	List of the parameters to Read
2	Aux_Voltage	Int	0	☒	☒	☒	☐	Current Auxiliary Voltage
3	Aux_Volt_Low_Level	Int	0	☒	☒	☒	☐	Auxiliary voltage warning low level
4	Aux_Volt_High_Level	Int	0	☒	☒	☒	☐	Auxiliary voltage warning High level
5	Channel_Error	Dint	0	☒	☒	☒	☐	Shows which Channels Caused the error
6	Switching_Cycle	Array[1..24] of Dint		☒	☒	☒	☐	Switching Cycle Counter
7	Write	Struct		☒	☒	☒	☐	List of the parameters to Write
8	Aux_Volt_Low_Level	Int	0	☒	☒	☒	☐	Auxiliary voltage warning low level
9	Aux_Volt_High_Level	Int	0	☒	☒	☒	☐	Auxiliary voltage warning High level
10	Clear_Switching_Count	Dint	0	☒	☒	☒	☐	Clear switching Counters
11	RequestFlag	Struct		☒	☒	☒	☐	List of the request command to read/write Value
12	ReadParams	Bool	false	☒	☒	☒	☐	Command read parameters
13	SetAuxVoltLimits	Bool	false	☒	☒	☒	☐	Command set voltage level Parameters
14	ClearCycleCount	Bool	false	☒	☒	☒	☐	Command Clear switching Cycle count
15	System_Command	Struct		☒	☒	☒	☐	
16	Acknowledge	Bool	false	☒	☒	☒	☐	Acknowledge Error
17	Store_Switching_Cycles	Bool	false	☒	☒	☒	☐	The current Value of the switching Cycle Counters are
18	Restore_Factory_Setting	Bool	false	☒	☒	☒	☐	Reset Parameters to default Values

➤ Use siemens function block “IO_LINK_DEVICE” to read and write acyclic data

IO_LINK_DEVICE is a function block wrote by siemens to access acyclic data of an IO-Link module. The function blocks “P2M_IOL_TBEN-L45-8IOL_PD” and “P2M_IOL_TBEN-L45-8IOL_PRM” of this guide work with IO_LINK_DEVICE to read and write each single parameter of the P2M/P2H IO-Link slave.

The function block is included in the Parker library but can also be download from siemens website.



Parameter	Data Type	Description
REQ	BOOL	Positive edge: Trigger data transfer
ID	DWORD	HW_ID of IO-Link submodule see paragraph Setting Up P2M/P2H IO-Link slave above
CAP	INT	Access point of the IO-Link communication (TURCK CAP = 254 decimal)
RD_WR	BOOL	Read or write access (0: Read, 1: Write)
PORT	INT	Port number at which the IO-Link device is operated. Add one to port number read from device view (Ex: if the connected port is 0, the port input to function block is equal to 1)
IOL_INDEX	INT	Parameter index (see P2M/P2H IO-Link manual for detail) 2: System Command 64: Switching cycle counter

		65: Clear Switching Cycles 66: Channel Error 67: Current Auxiliary Voltage 70: Auxiliary voltage warning levels
IOL_SUBINDEX	INT	Parameter subindex (see P2M/P2H IO-Link manual for detail) 1 to 24: Switching cycle counter, sub index 1 refers to cycle counter of the first output
LEN	INT	Length of the data to be read/written
RECORD_IOL_DATA	ARRAY [0..231] OF BYTE	Source/target area for the data to be read/written