Function block library

IOL_Basic_6

for PLCnext Engineer

Documentation for
PHOENIX CONTACT function blocks
PHOENIX CONTACT GmbH Co. KG
Flachsmarktstrasse 8
D-32825 Blomberg, Germany

This documentation is available in English only.

Table of Contents

1 Installation hint

If you did not specify a different directory during **library** installation all data in the MSI file will be unpacked to

c:\Users\Public\Documents\Phoenix Contact Libraries\PLCnext Engineer (former: PC Worx Engineer)

Please copy the library data to your PLCnext Engineer (former: PC Worx Engineer) working library directory.

If you did not specify a different directory during **PLCnext Engineer** installation the default PLCnext Engineer working library directory is

c:\Users\Public\Documents\PLCnext Engineer\Libraries (former: PC Worx Engineer\Libraries)

2 General information

IO-Link is a communication system used for the connection of intelligent sensors and actuators to automation systems. An IO-Link system consists of one IO-Link master and one or more IO-Link devices, i.e. sensors or actuators. The IO-Link master provides the interface for the PLC and controls the communication with the connected IO-Link devices.

An IO-Link master can have one or more IO-Link ports, however, only one IO-Link device can be connected to each port.

An IO-Link device has parameter data (objects) that can be read and/or written via asychronous services (commands). The objects are device-specific. Therefore use the current AsynCom library available on the PHOENIX CONTACT website. Integrate both the IOL_Basic and the AsynCom library into your project.

2.1 Articles and function blocks

Article	Local bus	PROFINET
 IB IL 24 IOL 4 DI 12-2MBD-PAC (2692733) IB IL 24 IOL 4 DI 12-PAC (2692717)	Use IOL_COM + AsynCom_IBS.	Use IOL_COM + AsynCom_PN
 AXL F IOL8 2H (1027843)	Do not use IOL_COM, use AsynCom_AXL only.	Do not use IOL_COM, use AsynCom_PN only.

 AXL SE IOL4 (1088132)	Do not use IOL_COM, use AsynCom_AXL only.	Do not use IOL_COM, use AsynCom_PN only.
 AXL E PN IOL8 DI4 M12 6M (2701519)	Not possible. PROFINET only.	Use IOL_COM + AsynCom_PN
 IOL MA8 PN DI8 (1072838)	Not possible. PROFINET only.	Use IOL_COM + AsynCom_PN

3 Change notes

Library version	Library build	PLCnext Engineer version	Change notes	Supported PLCs
6	20200408	>= 2020.0 LTS	Corrupted data during upload	AXC F 1152 (1151412) AXC F 2152 (2404267) AXC F 3152 (1069208)
6	20200330	>= 2020.0 LTS	Hint for AXL SE IOL4 (1088132) added, please refer to “Function blocks”	AXC F 1152 (1151412) AXC F 2152 (2404267) AXC F 3152 (1069208)
6	20200316	>= 2020.0 LTS	Hint for AXL F IOL8 2H (1027843) added, please refer to “Function blocks”	AXC F 1152 (1151412) AXC F 2152 (2404267) AXC F 3152 (1069208)
6	20200206	>= 2020.0 LTS	Released for 2020.0 LTS	AXC F 1152 (1151412) AXC F 2152 (2404267)
6	20191001	2019.0 LTS 2019.3 2019.6 2019.9	Adapted to 2019.9	AXC F 2152 (2404267)
5	20190722	2019.0 LTS 2019.3 2019.6	Adapted to 2019.6	AXC F 2152 (2404267)
4	20190607	2019.0 LTS	Added current build of the AsynCom library (to msi-file)	AXC F 2152 (2404267)
4	20190318	2019.0 LTS	• Adapted to IOL MA8 PN DI8 - 1072838 • Example for IOL MA8 PN DI8 - 1072838 added	AXC F 2152 (2404267)
3	20190225	2019.0 LTS	Supports “Allow extended identifiers” = ON	AXC F 2152 (2404267)
3	20190219	2019.0 LTS	• Adapted to PLCnext Engineer 2019.0 LTS	AXC F 2152 (2404267)

2	20180928	7.2.3	• Approval for IB IL 24 IOL 4 DI 12-2MBD-PAC (2692733) • Timeout changed from 2s to 5s. • Function block also works with cycle times up to 500 ms.	AXC F 2152 (2404267)
1	20180712	7.2.2	• Converted from PC Worx 6	AXC F 2152 (2404267)

New version number: Functional changes of at least one function block, incompatibilities (e.g. change of library format)

New build number: No functional changes, but changes in the MSI file (e.g. documentation update, additional examples)

4 Function blocks

Function block	Description	Version	Supported articles	License
IOL_COM	The block enables the asynchronous communication with the Phoenix Contact IO-Link modules. The function block can be used to write and / or read IO-Link services on the IO-Link-Master or on the IO-Link devices	3	AXL E PN IOL8 DI4 M12 6M (2701519) IB IL 24 IOL 4 DI 12-2MBD-PAC (2692733) IB IL 24 IOL 4 DI 12-PAC (2692717) IOL MA8 PN DI8 (1072838)	none

Access to the AXL F IOL8 2H (1027843) and AXL SE IOL4 (1088132) is done directly by using the function blocks of the AsynCom library. Please refer to the documentation of the AsynCom library. The MSI file of the AsynCom library contains an open example for AXL F IOL8 2H (1027843).

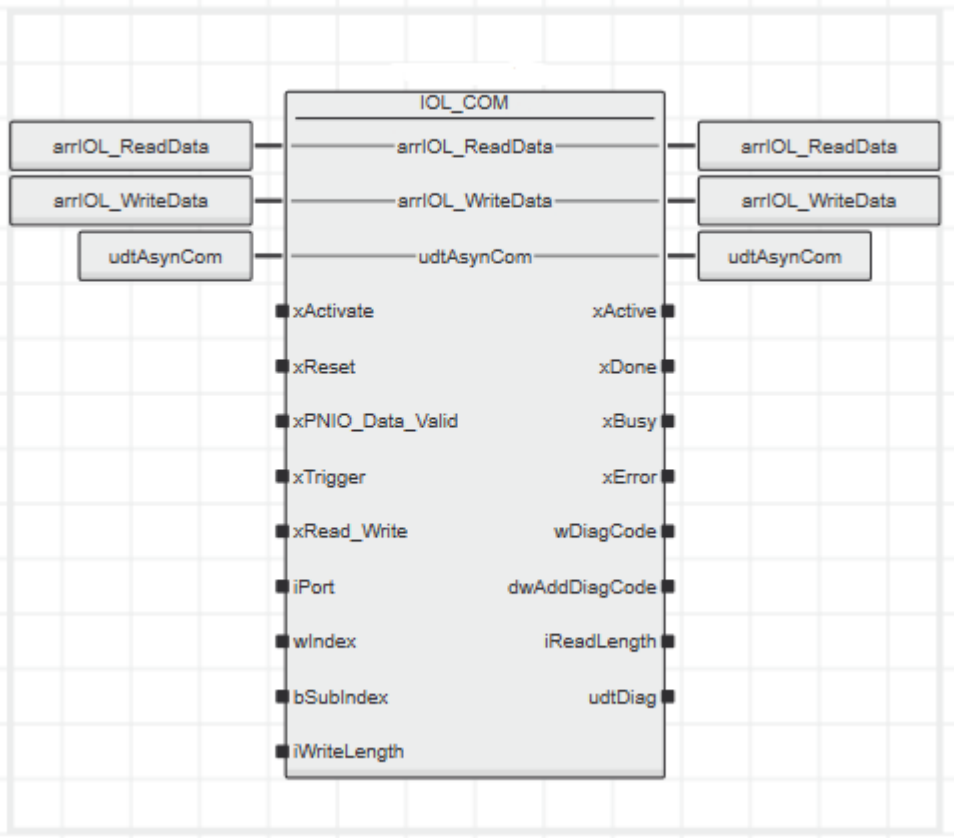
5 IOL_COM

The function block enables the asynchronous communication with the Phoenix Contact IO-Link modules. The function block can be used to write and / or read IO-Link services on the IO-Link master or on the IO-Link devices.

The function block can be used in different bus systems. The AsynCom library contains function blocks for every bus system. The IOL_COM function block communicates with the function blocks from the AsynCom library via the udtAsynCom structure.

If the address (node ID or communication reference) at the AsynCom block has been changed, restart the function block.

5.1 Function block call



5.2 Input parameters

Name	Type	Description
xActivate	BOOL	Rising edge: Activates the function block. FALSE: Deactivates the function block.
xReset	BOOL	Rising edge: Resets the function block.
xPNIO_Data_Valid	BOOL	Status bit for the evaluation of a valid PROFINET connection (optional).
xTrigger	BOOL	An IO-Link service is read and / or written with a positive edge.
xRead_Write	BOOL	TRUE: Write access FALSE: Read access
iPort	INT	Port number on which to read or write 0: IO-Link master 1..8: Port number
wIndex	WORD	Index of IOL object to be accessed to
bSubIndex	BYTE	Subindex of IOL object
iWriteLength	INT	Number of bytes to be written

5.3 Output parameters

Name	Type	Description
xActive	BOOL	FALSE: Function block is not active. TRUE: Function block is active.
xDone	BOOL	TRUE: Service written/read successfully. The parameter is True as long as xTrigger is True (for at least one cycle).
xBusy	BOOL	FALSE: The function block is ready for the next access. TRUE: The block is busy with the service execution.
xError	BOOL	TRUE: An error has occurred. For details refer to wDiagCode and wAddDiagCode.
wDiagCode	WORD	Diagnosis code. Refer to diagnostic table.
dwAddDiagCode	DWORD	Additional diagnostic code. Refer to diagnostics table.
iReadLength	INT	Number of data bytes received

5.4 Inout parameters

Name	Type	Description
arrIOL_ReadData	IOL_ARR_B_1_64	The array contains the received data. Every successful read access will overwrite the data in this array by the newly received data.
arrIOL_WriteData	IOL_ARR_B_1_64	The array contains the data to be written.
udtAsynCom	ASYN_UDT_COM	Data exchange structure for asynchronous communication. Connected to AsynCom* function block.

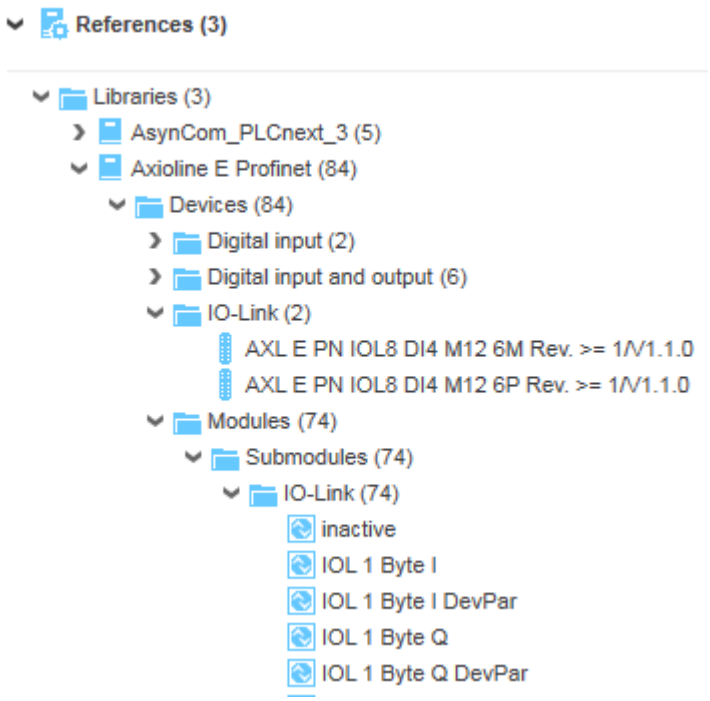
5.5 Diagnosis

<table>
<tr><th>wDiagCode</th><th>wAddDiagCode</th><th>Description</th></tr>
<tr><td>16#0000</td><td>16#00000000</td><td>Function block is deactivated</td></tr>
<tr><td>16#8000</td><td>16#00000000</td><td>Function block is in regular operation</td></tr>
<tr><td>16#C100</td><td></td><td>Communication error</td></tr>
<tr><td></td><td>16#00000000</td><td>Timeout or PROFINET connection error. The parameter xPNIO_Data_Valid (optional) is FALSE.</td></tr>
<tr><td></td><td>16#00000010</td><td>Inline INTERBUS: Invalid communication reference (e.g. strPartner is zero)</td></tr>
<tr><td></td><td>16#09B00005</td><td>Inline INTERBUS: There is no communication reference.</td></tr>
<tr><td></td><td>16#xxxxxxxx</td><td>Inline INTERBUS: Communication error. PCP_CONNECT firmware block indicates an error.</td></tr>
<tr><td>16#C200</td><td>16#00000001</td><td>Configuration error. Invalid parameter at the iWriteLength input.</td></tr>
<tr><td>16#C300</td><td></td><td>IO-Link error</td></tr>
<tr><td></td><td>16#xxxxxxxx</td><td>The parameter is structured as follows:
<table>
<tr><td>WORD 1</td><td colspan="2">WORD 0</td></tr>
<tr><td></td><td>Byte 1</td><td>Byte 0</td></tr>
<tr><td>IOL_M Error_Code</td><td>IOL Error_Code</td><td>IOL Add_Error_Code</td></tr>
</table>
For details refer to appendix.</td></tr>
<tr><td>16#C400</td><td></td><td>Error while sending</td></tr>
<tr><td></td><td>16#xxxxxxxx</td><td>Error in the WRREC or PCP_WRITE firmware block.</td></tr>
<tr><td></td><td>16#DF80A1xx</td><td>Negative acknowledgment when writing to the module: Caused by: Incorrect service content, index, subindex, port number or length of the data to be written.</td></tr>
<tr><td></td><td>16#0000F003</td><td>Invalid node ID</td></tr>
<tr><td></td><td>16#0000F016</td><td>The service used to write the data record could not be run.</td></tr>
<tr><td></td><td>16#06050000</td><td>INTERBUS: Invalid length of the data to be written</td></tr>
<tr><td></td><td>16#06070000</td><td>INTERBUS: Invalid index</td></tr>
<tr><td>16#C500</td><td></td><td>Error while receiving</td></tr>
<tr><td></td><td>16#xxxxxxxx</td><td>Error in the RDREC or PCP_READ firmware block</td></tr>
<tr><td></td><td>16#DE80A0xx</td><td>Negative acknowledgment when reading from the module: Caused by: Incorrect service content, index, subindex or port number</td></tr>
<tr><td></td><td>16#0000F003</td><td>Invalid node ID</td></tr>
<tr><td></td><td>16#0000F015</td><td>The service used to read the data record could not be run.</td></tr>
<tr><td></td><td>16#06070000</td><td>INTERBUS: Invalid index</td></tr>
</table>

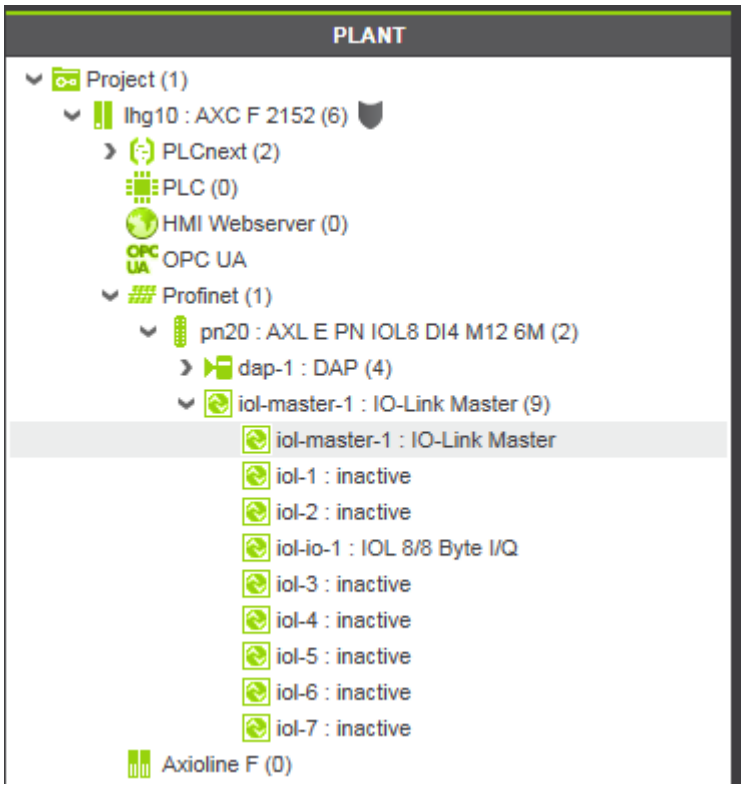
6 Startup instructions

6.1 Axioline PROFINET

Add the firmware library “Axioline E Profinet” to your project.



Insert the AXL E PN IOL8 DI4 M12 6M (2701519) module into the bus configuration.



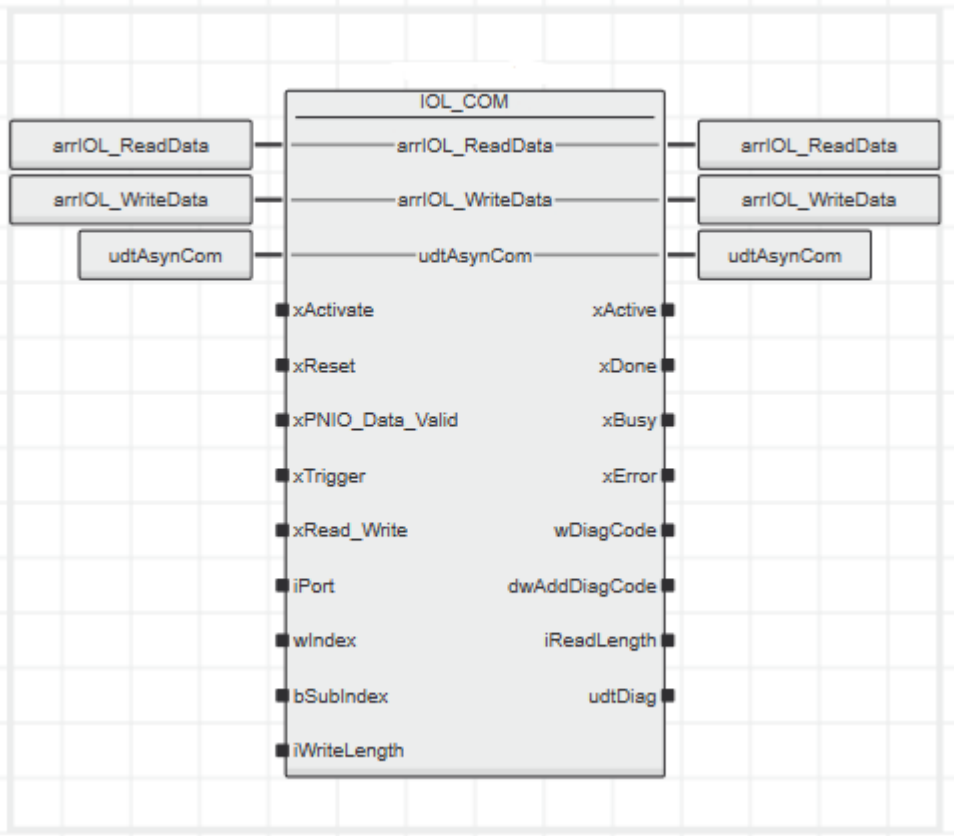
Extend the bus configuration with the IO-Link devices from the Axioline IO-Link module catalog.

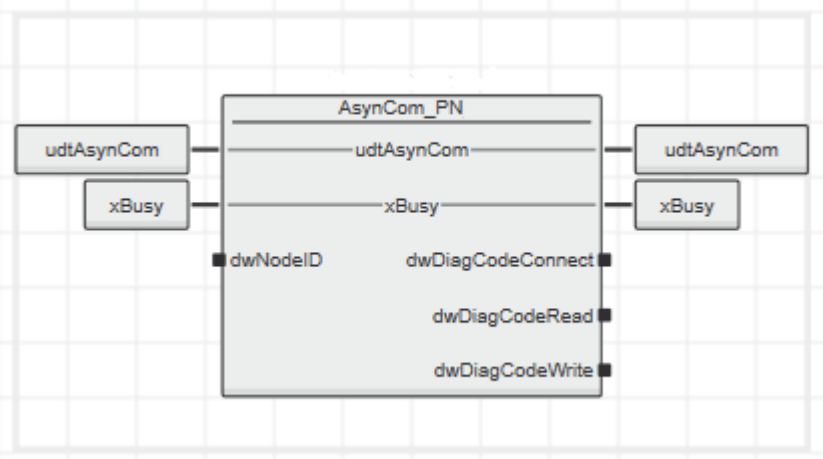
When you use third-party devices, insert a non-specific device with the same process data size as the third-party device (here: IOL 8/8 Byte I/Q).

Make sure that the module operates despite the configuration difference.

In the program, call the IOL_COM function block and the AsynCom_PN function block.

At the AsynCom_PN function block, enter the node ID of the master slot.





6.2 IO-Link communication with IOL MA8 PN DI8 (1072838) IO-Link-Master

For the startup instruction of IO-Link communication please find the following example:

- IOL_6_EXA_IOL_MA8_PN_DI8.pcwex

This example is located in the “Examples” folder of the unzipped msi-file of the library.

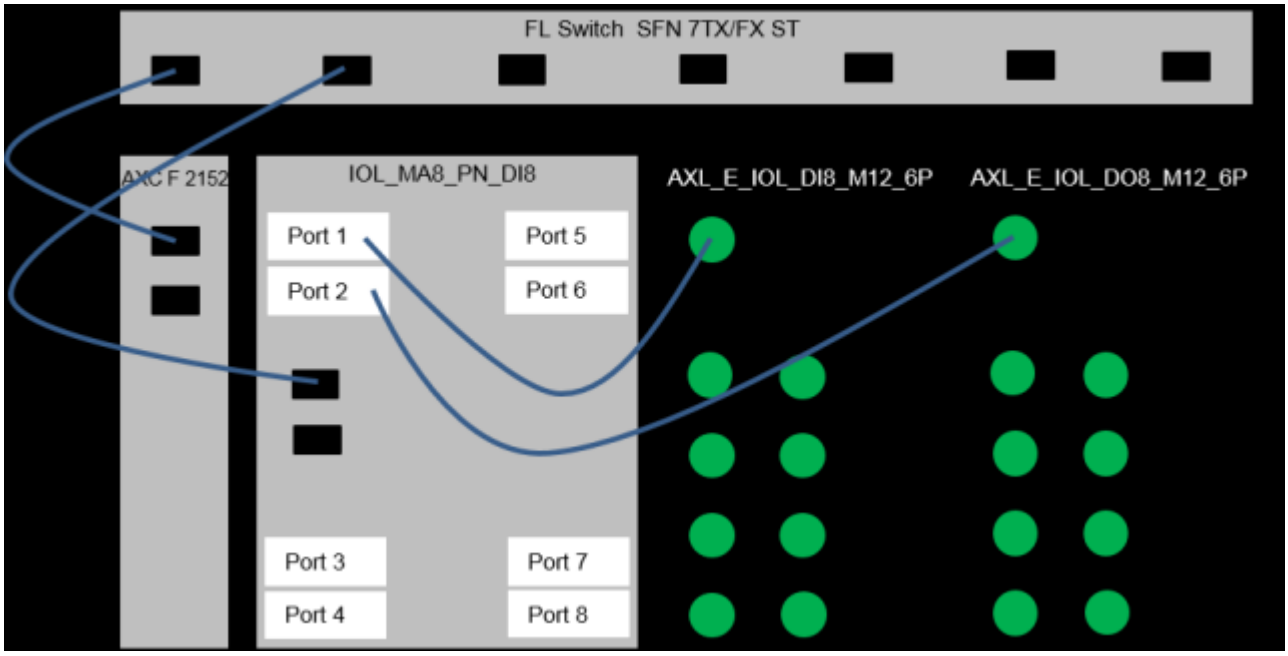
It represents IO-Link communication between IO-Link-Master (IOL MA8 PN DI8 (1072838)) and IO-Link-Devices by using the IOL_Com function block and asynchronous communication.

6.2.1 Plant

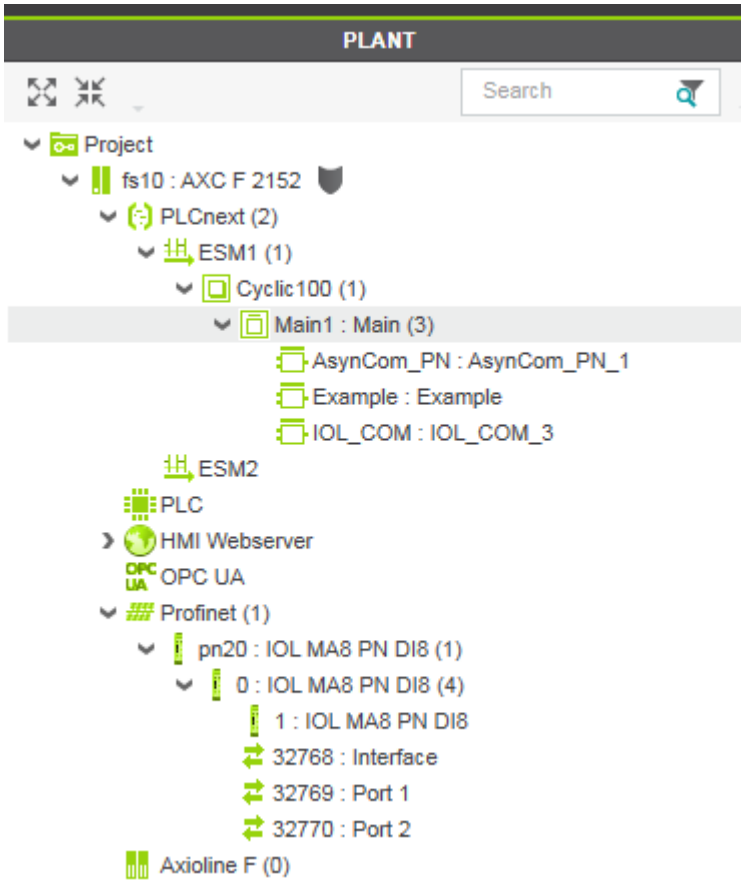
For this example, the following hardware is used:

- AXC F 2152 (2404267)
- IOL MA8 PN DI8 (1072838)
- AXL E IOL DI8 M12 6P (2702658)
- AXL E IOL DO8 M12 6P (2702659)

Hardware:



PLCnext Engineer:

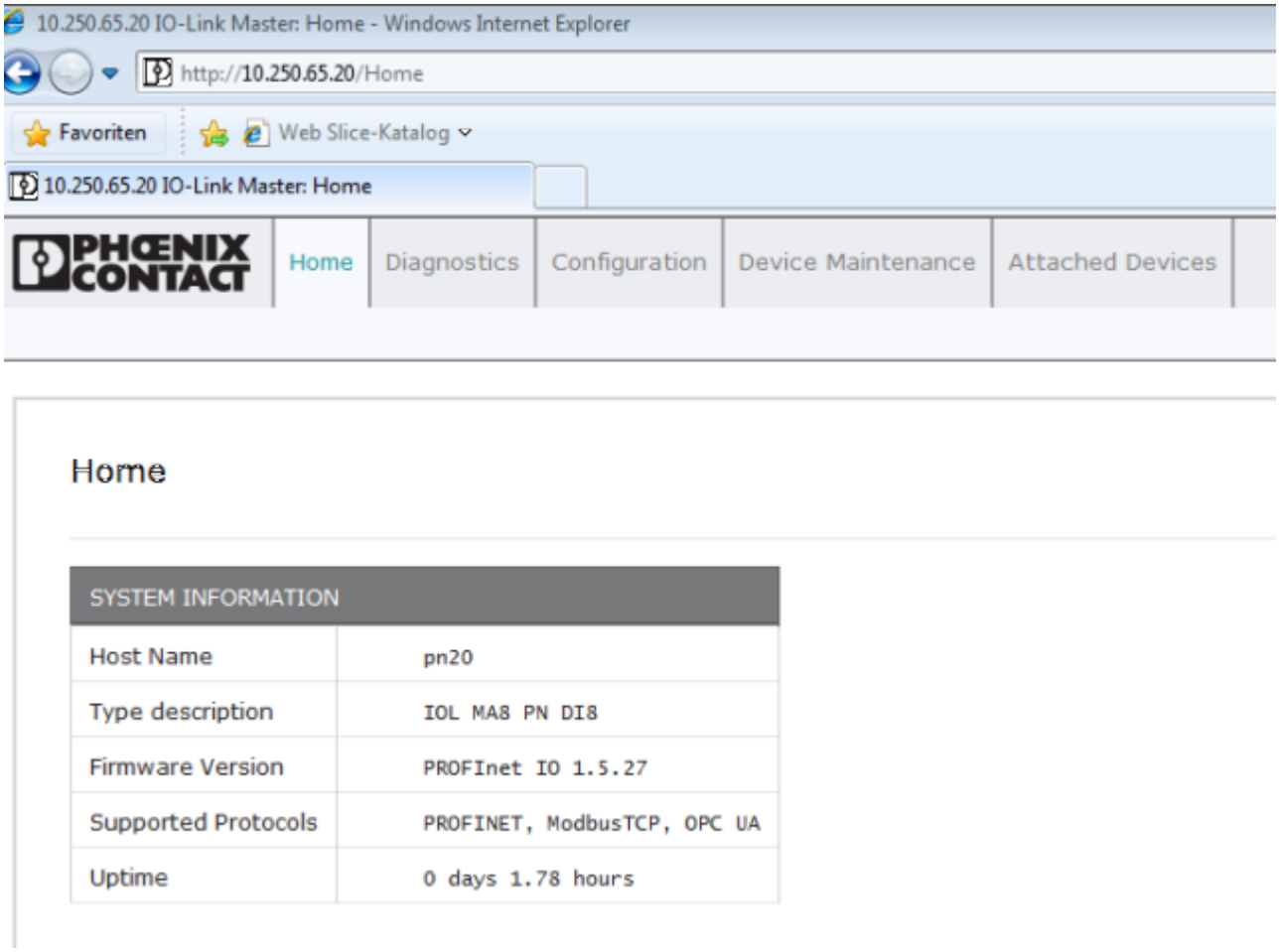


6.2.2 Example IO_Link communication with IOL MA8 PN DI8 (1072838) IO-Link-Master

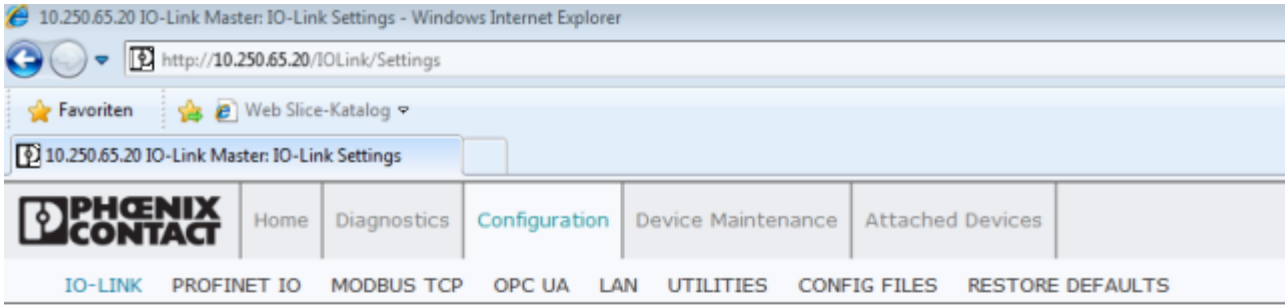
The function block IOL_COM can be used to read out IO-Link services on the IO-Link devices using the IO-Link master.

In this example, the article numbers of the IO-Link devices (AXL E IOL DI8 M12 6P (2702658)) are read out.

Before you start, it is necessary to configure the IO-Link-Master. Please open your web browser and enter the IP address of the IO-Link-Master. After a short time the start page will appear.



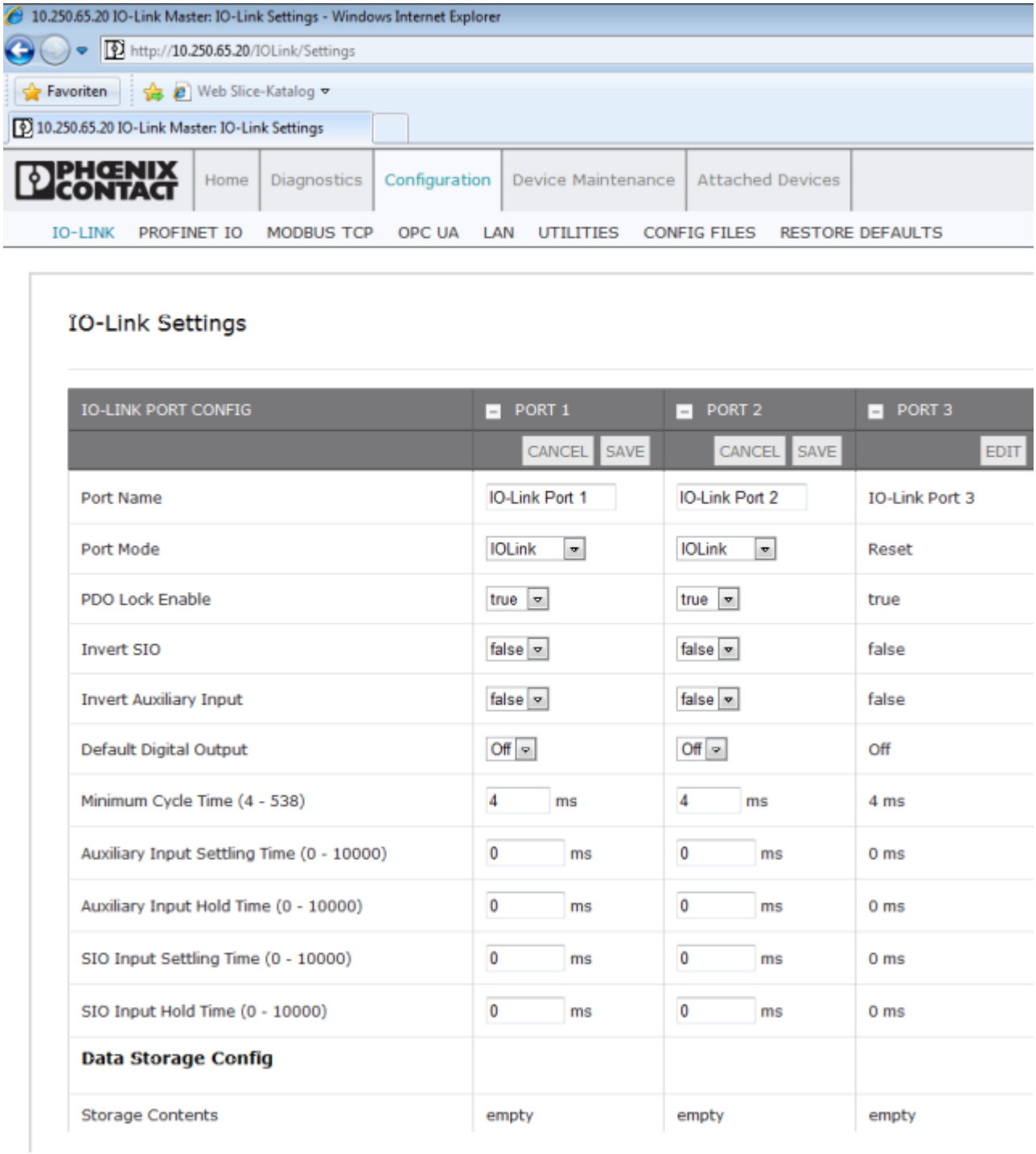
Now please select the Configuration tab to open the IO-Link-Settings window.

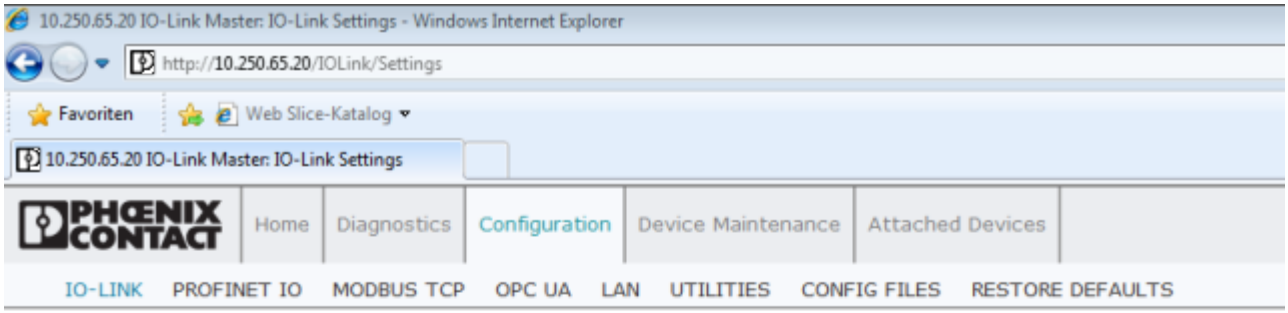


IO-Link Settings

IO-LINK PORT CONFIG	PORT 1	PORT 2	PORT 3
	EDIT	EDIT	EDIT
Port Name	IO-Link Port 1	IO-Link Port 2	IO-Link Port 3
Port Mode	Reset	Reset	Reset
PDO Lock Enable	true	true	true
Invert SIO	false	false	false
Invert Auxiliary Input	false	false	false
Default Digital Output	Off	Off	Off
Minimum Cycle Time (4 - 538)	4 ms	4 ms	4 ms
Auxiliary Input Settling Time (0 - 10000)	0 ms	0 ms	0 ms
Auxiliary Input Hold Time (0 - 10000)	0 ms	0 ms	0 ms
SIO Input Settling Time (0 - 10000)	0 ms	0 ms	0 ms
SIO Input Hold Time (0 - 10000)	0 ms	0 ms	0 ms
Data Storage Config			
Storage Contents	empty	empty	empty
Automatic Upload Enable	Off	Off	Off
Automatic Download Enable	Off	Off	Off
Data Storage Manual Ops			

To configure the required ports 1 and 2, click on the Edit button. Now you can select IO-Link under Port Mode and save it with the Save button.


10.250.65.20 IO-Link Master: IO-Link Settings - Windows Internet Explorer
http://10.250.65.20/IOLink/Settings
Favoriten
Web Slice-Katalog
10.250.65.20 IO-Link Master: IO-Link Settings
PHOENIX CONTACT
Home
Diagnostics
Configuration
Device Maintenance
Attached Devices
IO-LINK
PROFINET IO
MODBUS TCP
OPC UA
LAN
UTILITIES
CONFIG FILES
RESTORE DEFAULTS
IO-Link Settings
IO-LINK PORT CONFIG
PORT 1
PORT 2
PORT 3
CANCEL
SAVE
CANCEL
SAVE
EDIT
Port Name
IO-Link Port 1
IO-Link Port 2
IO-Link Port 3
Port Mode
IOLink
IOLink
Reset
PDO Lock Enable
true
true
true
Invert SIO
false
false
false
Invert Auxiliary Input
false
false
false
Default Digital Output
Off
Off
Off
Minimum Cycle Time (4 - 538)
4 ms
4 ms
4 ms
Auxiliary Input Settling Time (0 - 10000)
0 ms
0 ms
0 ms
Auxiliary Input Hold Time (0 - 10000)
0 ms
0 ms
0 ms
SIO Input Settling Time (0 - 10000)
0 ms
0 ms
0 ms
SIO Input Hold Time (0 - 10000)
0 ms
0 ms
0 ms
Data Storage Config
Storage Contents
empty
empty
empty



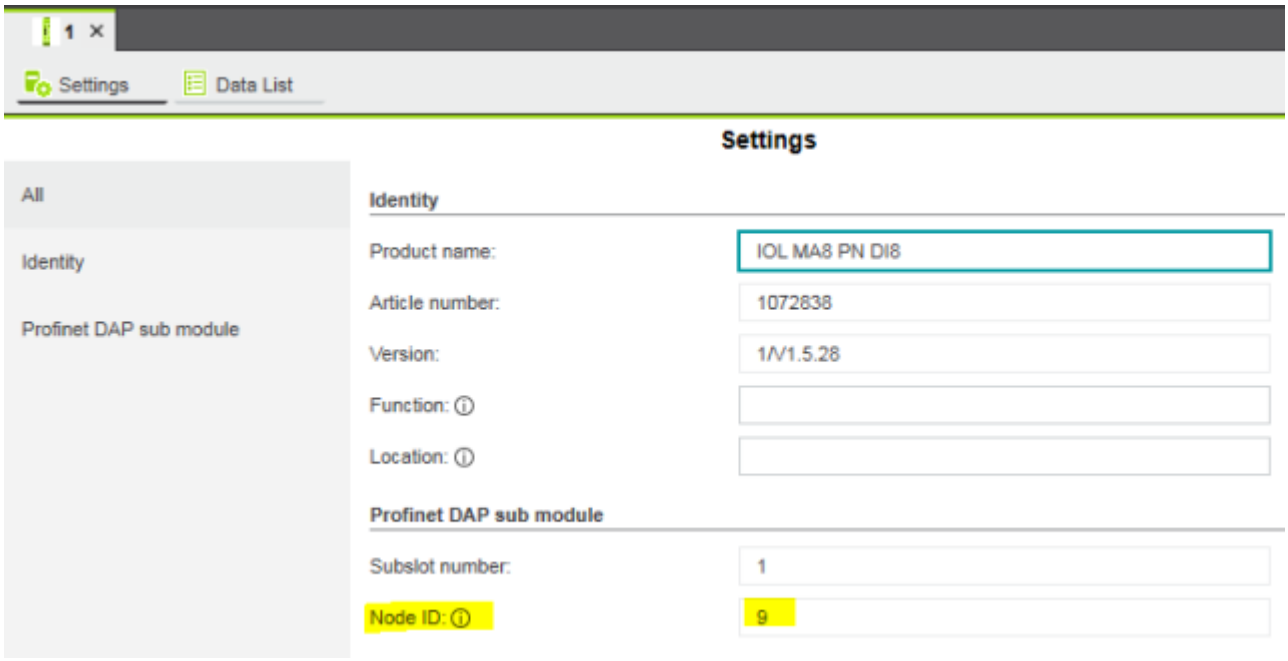
IO-Link Settings

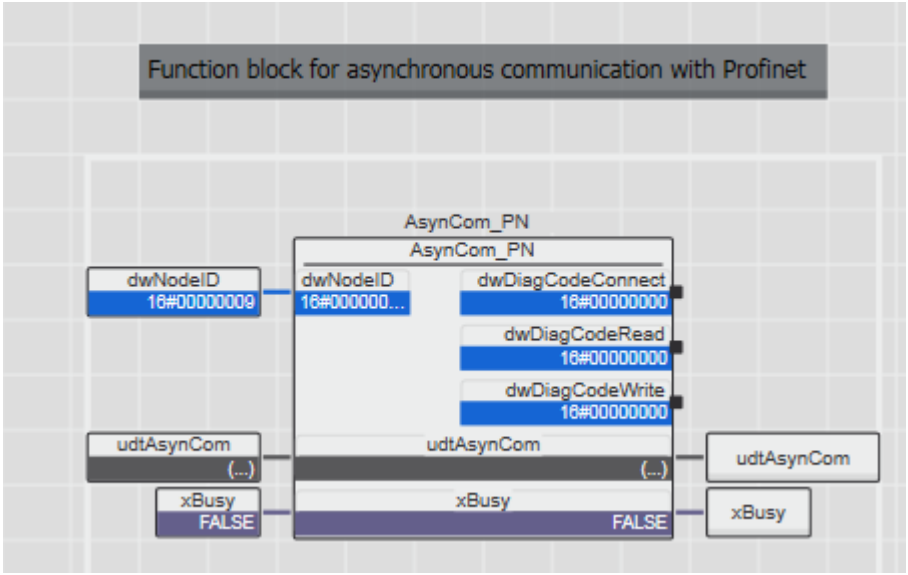
IO-LINK PORT CONFIG	PORT 1	PORT 2	PORT 3
	EDIT	EDIT	EDIT
Port Name	IO-Link Port 1	IO-Link Port 2	IO-Link Port 3
Port Mode	IOLink	IOLink	Reset
PDO Lock Enable	true	true	true
Invert SIO	false	false	false
Invert Auxiliary Input	false	false	false
Default Digital Output	Off	Off	Off
Minimum Cycle Time (4 - 538)	4 ms	4 ms	4 ms
Auxiliary Input Settling Time (0 - 10000)	0 ms	0 ms	0 ms
Auxiliary Input Hold Time (0 - 10000)	0 ms	0 ms	0 ms
SIO Input Settling Time (0 - 10000)	0 ms	0 ms	0 ms
SIO Input Hold Time (0 - 10000)	0 ms	0 ms	0 ms
Data Storage Config			
Storage Contents	empty	empty	empty
Automatic Upload Enable	Off	Off	Off
Automatic Download Enable	Off	Off	Off
Data Storage Manual Ops			

If the configuration is stored, Port Mode has to changed from Reset to IO_Link and the LEDs on the IO-Link-Master for ports 1 and 2 light up.

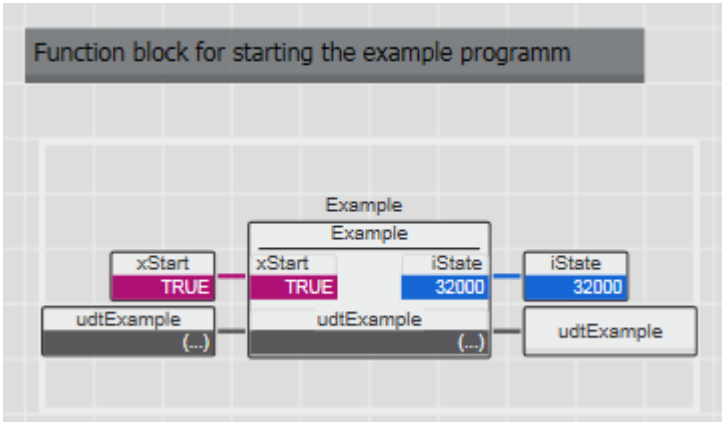
The example program can now be called up.

Therefore please add the Node-ID of the IOL_MA8_PN_DI8 IO-Link-Master at the dwNodeID input parameter of the AsynCom_PN function block for Profinet communication. You can find the Node-ID under IOL MA8 PN DI8/Settings/Node_ID





Now, in order to start the example program, the input parameter xStart must be set to TRUE at the Example function block.



This starts the following state machine:

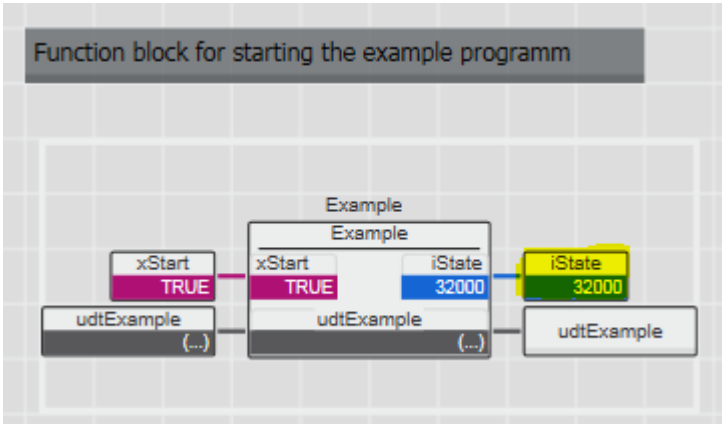
CASE iState OF

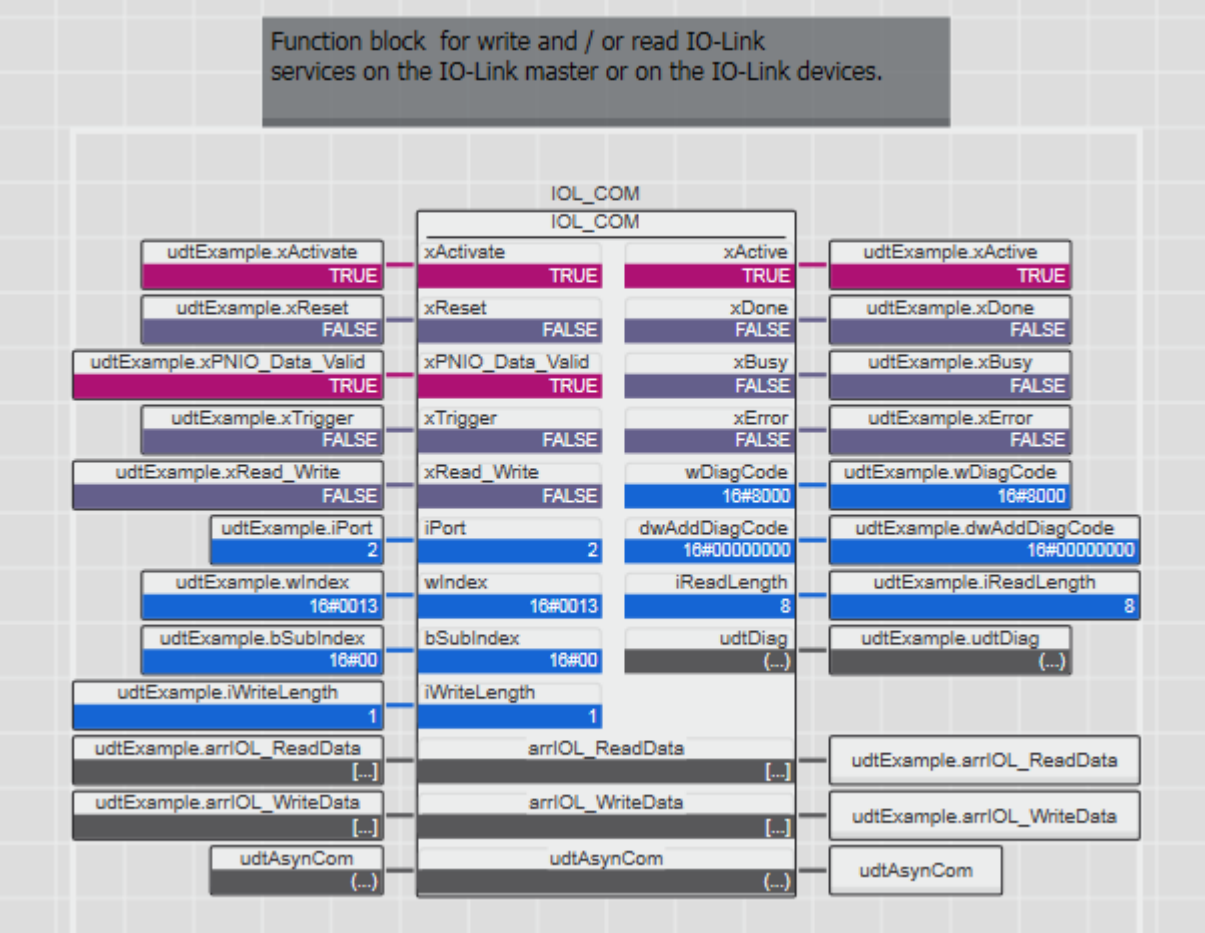
```
0: (* Wait for start of example program *).
    iIndex := 0;
    IF xStart = TRUE THEN
        iState := 1;
    END_IF;

1: (* Reset state *)
    (* Initial value of iteration *)
    iIteration                  := 0;
    (* Max value of iteration *)
    iIterationMax               := 1;
    (* Activation of function block *)
    udtExample.xActivate        := FALSE;
    (* Reset of function block *)
    udtExample.xReset           := FALSE;
    (* Status bit for the evaluation of a valid Profinet connection *)
    udtExample.xPNIO_Data_Valid := TRUE;
    (* An IO-Link service is read with a positive edge *)
    udtExample.xTrigger         := FALSE;
    (* Read instruction *)
    udtExample.xRead_Write      := FALSE;
    (* Port number on which to read *)
    udtExample.iPort            := 0;
    (* Index of IOL object to be accessed to *)
    udtExample.wIndex           := WORD#16#13; (* Article number *)
    (* Subindex of IOL object to be accessed to *)
    udtExample.bSubIndex        := BYTE#0;
    (* Number of bytes to be written *)
    udtExample.iWriteLength     := 1;

    IF udtExample.xActive = FALSE THEN
        iState := 10;
    END_IF;

10:
    IF(iIteration = 0) THEN
        (* AXL E IOL DO8 M12 6P - 2702659 *)
        udtExample.iPort    := 1; (* Port number on which to read *)
        strRef := '2702659'; (* Reference value *)
    ELSE
        (* AXL E IOL DI8 M12 6P - 2702658 *)
        udtExample.iPort    := 2; (* Port number on which to read *)
        strRef := '2702658'; (* Reference value *)
    END_IF;
    (* Activating of function block and starting read instruction *)
    udtExample.xActivate    := TRUE;
    udtExample.xTrigger     := TRUE;
    (* If all conditions are met...*)
    IF (
        udtExample.xActive = TRUE
        AND udtExample.xDone = TRUE
        AND udtExample.xBusy = FALSE
        AND udtExample.xError = FALSE
        AND udtExample.wDiagCode = WORD#16#8000
        AND udtExample.dwAddDiagCode = DWORD#16#00000000
    (*...then copy elementary data types from a byte stream to a variable *)
    ) THEN
```

```
        BUF_TO_STRING.REQ       := TRUE;
        BUF_TO_STRING.BUF_OFFS  := DINT#0;
        BUF_TO_STRING.BUF_CNT   := DINT#7;
        iState                  := 20;
    END_IF;

20: (* When copy operation is complete....*)
    IF (
        BUF_TO_STRING.DONE = TRUE
        (*...and strTemp = strRef *)
        AND EQ_STRING(strTmp, strRef)
    ) THEN
        (*...then reset udtExample.xTrigger and BUF_TO_STRING.REQ *)
        udtExample.xTrigger := FALSE;
        BUF_TO_STRING.REQ   := FALSE;
        (*...and go to next state *)
        iState              := 100;
    END_IF;

100: (* If iteration is finished then go to next state... *)
    IF(iIteration = iIterationMax) THEN
        iState      := 32000;
    ELSE
        (*...else start next iteration with read of port number 2 *)
        iIteration  := iIteration + 1;
        iState      := 10;
    END_IF;

32000: (* End *)
    iState  := 32000; (* End of demo *)
```

END_CASE;

Once the state machine has arrived at state 32000, the program has run successfully.

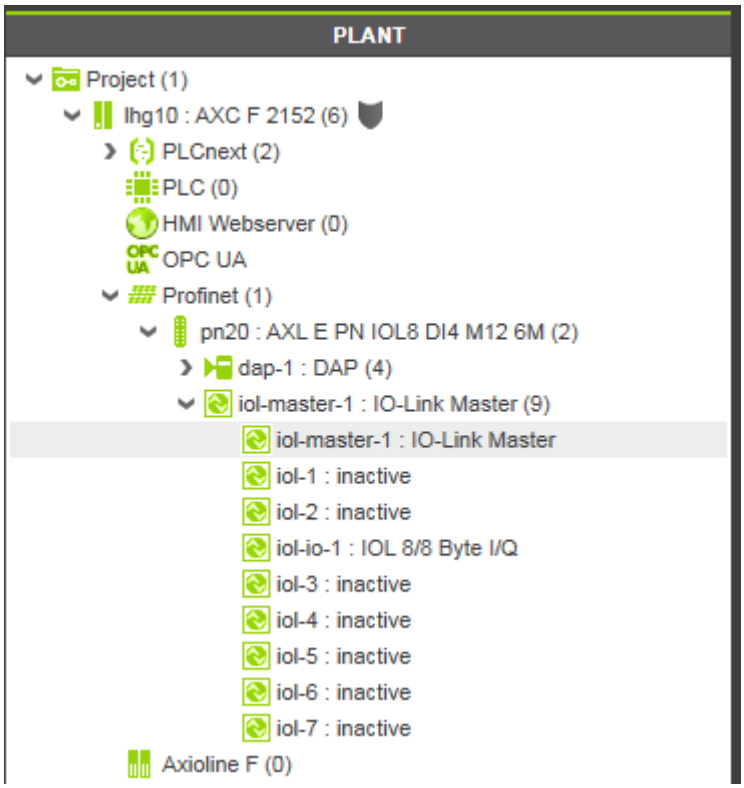



Function block for write and / or read IO-Link services on the IO-Link master or on the IO-Link devices.
IOL_COM
IOL_COM
udtExample.xActivate
TRUE
xActivate
TRUE
xActive
TRUE
udtExample.xActive
TRUE
udtExample.xReset
FALSE
xReset
FALSE
xDone
FALSE
udtExample.xDone
FALSE
udtExample.xPNIO_Data_Valid
TRUE
xPNIO_Data_Valid
TRUE
xBusy
FALSE
udtExample.xBusy
FALSE
udtExample.xTrigger
FALSE
xTrigger
FALSE
xError
FALSE
udtExample.xError
FALSE
udtExample.xRead_Write
FALSE
xRead_Write
FALSE
wDiagCode
16#8000
udtExample.wDiagCode
16#8000
udtExample.iPort
2
iPort
2
dwAddDiagCode
16#00000000
udtExample.dwAddDiagCode
16#00000000
udtExample.wIndex
16#0013
wIndex
16#0013
iReadLength
8
udtExample.iReadLength
8
udtExample.bSubIndex
16#00
bSubIndex
16#00
udtDiag
(...)
udtExample.udtDiag
(...)
udtExample.iWriteLength
1
iWriteLength
1
udtExample.arrIOL_ReadData
[...]
arrIOL_ReadData
[...]
udtExample.arrIOL_ReadData
udtExample.arrIOL_WriteData
[...]
arrIOL_WriteData
[...]
udtExample.arrIOL_WriteData
udtAsynCom
(...)
udtAsynCom
(...)
udtAsynCom

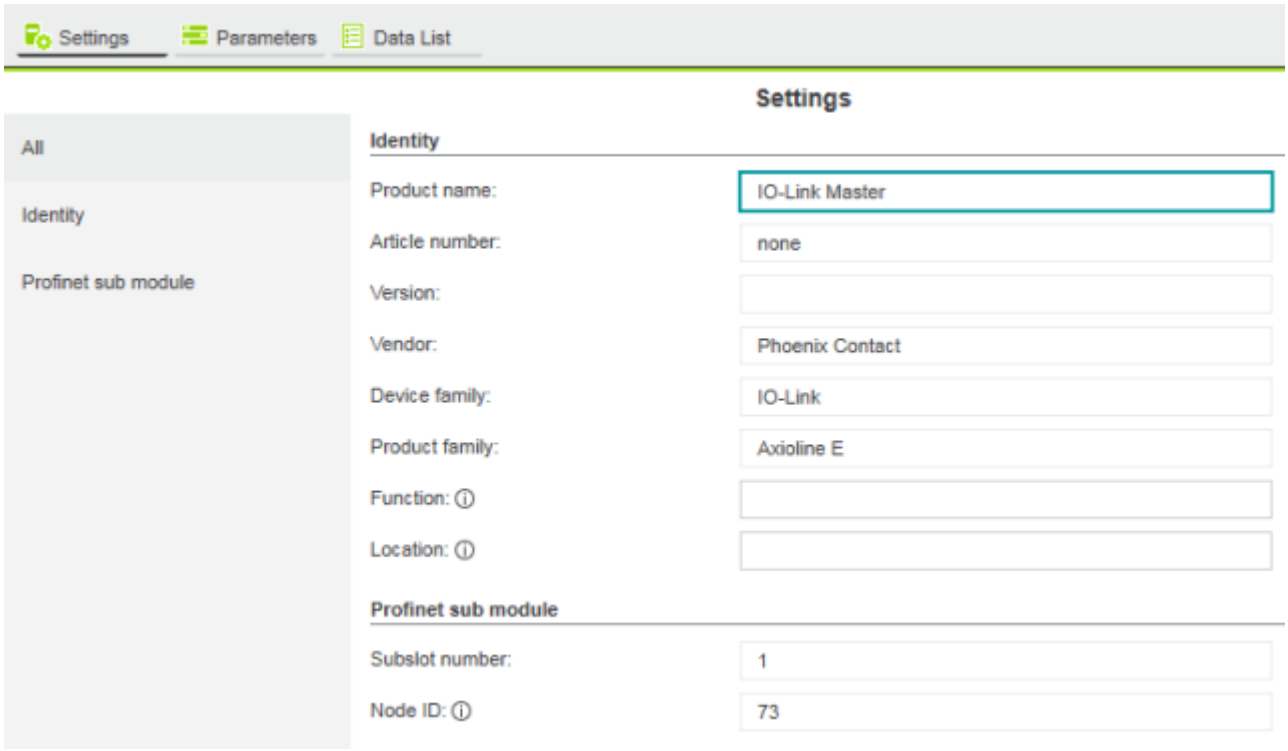
7 FAQs

7.1 Obtaining the node ID

In the "Plant" window, select "IO-Link Master".



You will find the node ID in the settings.



The node ID is specified in decimal format (here: 73). However, the node ID for the input of the AsynCom_PN function block has to be in hexadecimal format.

Note that the node ID is generated automatically by PC Worx and may therefore vary in the different projects.

8 Appendix

8.1 IO-Link diagnostics[1,2]

[1] IO-Link Community, IOL-Interface-Spec_10002_V112_Jul13, page 85
[2] IO-Link Community, IOL-Comm-Spec_10002_V10_090118.doc, pages 115-116

8.1.1 IOL-M Error_Codes

IOL-M Error_Code	Coding	Definitions
No error	16#0000	No IOL-M or IOL-Server errors.
-	16#0001 - 16#06FF	Reserved for future use.
IOL_Call conflict	16#7000	Unexpected write req instead of read req.
Wrong IOL_Call	16#7001	Decode error.
Port blocked	16#7002	Port occupied by another task.
-	16#7003 - 16#7FFF	Reserved for future use.
Timeout	16#8000	Timeout when IOL-Ds or IOL-M ports are busy.
Wrong index	16#8001	IOL_Index >32767 or <65535
Wrong port address	16#8002	Port address beyond defined maximum.
Wrong port function	16#8003	Port function not supported.
-	16#8004 - 16#FFFF	Reserved for future use.

8.1.2 Error codes

Type	Origin	Name	Category	Mode	Inst.	Value	Remark
PDU buffer overflow	remote	S_PDU_BUFFER	ERROR	SINGLE SHOT	DL	16#52	Device buffer is too small for storing the complete PDU.
PDU checksum error (master)	local	M_PDU_CHECK	ERROR	SINGLE SHOT	DL	16#56	Calculated PDU checksum in master does not match actual received SPDU.
PDU checksum error (device)	remote	S_PDU_CHECK	ERROR	SINGLE SHOT	DL	16#56	Calculated PDU checksum in device does not match actual received SPDU.
PDU flow control error	remote	S_PDU_FLOW	ERROR	SINGLE SHOT	DL	16#56	Violation of flow control rule during transfer of SPDU between master and device.
Illegal PDU service primitive (master)	local	M_PDU_ILLEGAL	ERROR	SINGLE SHOT	AL	16#57	Unknown service primitive or wrong response e.g. Read response or Write request
Illegal PDU service primitive (device)	remote	S_PDU_ILLEGAL	ERROR	SINGLE SHOT	AL	16#58	Unknown service primitive e.g. different protocol revision.
Communication error	local / remote	COM_ERR	ERROR	SINGLE SHOT	unknown	16#10	Negative service response initiated by a communication error, e.g. IO-Link connection interrupted
Device / application error	remote	S_APP_DEV	ERROR	SINGLE SHOT	APP	16#80	Service PDU transferred, but not processed due to device error. See error details in Additional Code.

8.1.3 Additional codes for error code S_APP_DEV

Type	Value	Remark
No details	16#00	-
Index not available	16#11	-
Subindex not available	16#12	-
Service temporarily not available	16#20	-
Service temporarily not available, local control	16#21	local operation at device inhibits access
Service temporarily not available, device control	16#22	device state does not allow access, e.g. during teach or calibration
Access denied	16#23	e.g. write on read-only parameters
Parameter value out of range	16#30	-
Parameter value above limit	16#31	-
Parameter value below limit	16#32	-
Interfering parameter	16#40	collision with other parameter values
Application failure	16#81	e.g. separate application and communication processors
Application not ready	16#82	e.g. separate application and communication processors

8.2 Diagnosis of used firmware function blocks

8.2.1 PDI_READ

for PLCnext Engineer

ERROR = TRUE

STATUS[0]	STATUS[1]	Meaning
16#09B0	16#000C	The variable connected to RD_1 is invalid (no array or invalid array type).
16#09B0	16#000B	The array connected to RD_1 is too small to save the requested receive data.
16#09B0	16#000E	Timeout. No response to the sent PDI READ request received.
16#09B0	16#000F	An internal error has occurred.

When receiving a negative confirmation as response to a PDI_READ request, the Axioline module directly copies the received error code (Error_Code and Add_Info) to STATUS[0] or STATUS[1]. These error codes are module-specific. For a description see the respective module documentation.

8.2.2 PDI_WRITE

for PLCnext Engineer

ERROR = TRUE

STATUS[0]	STATUS[1]	Meaning
16#09B0	16#000A	The variable connected to SD_1 is invalid (no array or invalid array type).
16#09B0	16#0009	Invalid value at DATA_CNT input. The value is either greater than the array connected to SD_1, greater than the maximum allowed length (245 bytes) or equal to zero.
16#09B0	16#000E	Timeout. No response to the sent PDI WRITE request received.
16#09B0	16#000F	An internal error has occurred.

When receiving a negative confirmation as response to a PDI_WRITE request, the Axioline module directly copies the received error code (Error_Code and Add_Info) to STATUS[0] or STATUS[1]. These error codes are module-specific. For a description see the respective module documentation.

8.2.3 RDREC

for PLCnext Engineer

Error code (hex)	Meaning
16#0000	No error occurred.
16#F001	Too many instances used.
16#F002	Error during initialization of the function block.
16#F003	Invalid ID.
16#F004	Invalid HANDLE/ID.
16#F005	Resources conflict.
16#F006	A function block internal task could not be generated.
16#F007	Too many instances used.
16#F008	Invalid type of a parameter.
16#F009	Invalid parameter value.
16#F00A	Unallowed parameter.
16#F00B	Invalid length specified.
16#F00C	ID could not be created (too many IDs).
16#F00D	No entry found that matches the specified ID.
16#F00F	No further entries found.
16#F010	Entry in use.
16#F011	Alarm acknowledgement could not be done.
16#F012	Error reading the AR parameters (1st time).
16#F013	Negative acknowledgement received for the execution of a PROFINET service.
16#F014	Invalid length for parameter LEN/MLEN or/and RECORD data record too short.
16#F015	The service used to read the RECORD data record could not be run.
16#F016	The service used to write the RECORD data record could not be run.
16#F017	Service acknowledgement not received.
16#F018	Invalid INDEX used to access the RECORD data record of the IO device, for example, INDEX greater than 16#7FFF.
16#F019	Unknown command code.
16#F01A	Error starting the Application Relation (AR).
16#F01B	Error stopping the Application Relation (AR).
16#F01C	Notification of stopped Application Relation (AR) failed.
16#F01D	Setting the "Drive BF" flag failed.
16#F01E	Error reading the AR parameters (2nd time).

8.2.4 WRREC

for PLCnext Engineer

Error code (hex)	Meaning
16#0000	No error occurred.
16#F001	Too many instances used.
16#F002	Error during initialization of the function block.
16#F003	Invalid ID.
16#F004	Invalid HANDLE/ID.
16#F005	Resources conflict.
16#F006	A function block internal task could not be generated.
16#F007	Too many instances used.
16#F008	Invalid type of a parameter.
16#F009	Invalid parameter value.
16#F00A	Unallowed parameter.
16#F00B	Invalid length specified.
16#F00C	ID could not be created (too many IDs).
16#F00D	No entry found that matches the specified ID.
16#F00F	No further entries found.
16#F010	Entry in use.
16#F011	Alarm acknowledgement could not be done.
16#F012	Error reading the AR parameters (1st time).
16#F013	Negative acknowledgement received for the execution of a PROFINET service.
16#F014	Invalid length for parameter LEN/MLEN or/and RECORD data record too short.
16#F015	The service used to read the RECORD data record could not be run.
16#F016	The service used to write the RECORD data record could not be run.
16#F017	Service acknowledgement not received.
16#F018	Invalid INDEX used to access the RECORD data record of the IO device, for example, INDEX greater than 16#7FFF.
16#F019	Unknown command code.
16#F01A	Error starting the Application Relation (AR).
16#F01B	Error stopping the Application Relation (AR).
16#F01C	Notification of stopped Application Relation (AR) failed.
16#F01D	Setting the "Drive BF" flag failed.
16#F01E	Error reading the AR parameters (2nd time).

9 Support

For technical support please contact your local PHOENIX CONTACT agency

at https://www.phoenixcontact.com

Owner:

PHOENIX CONTACT Electronics GmbH
Business Unit Automation Systems
System Services
Library Services