How to run a Matrix N over PROFINET-IO

Tutorial + Example for PLC Siemens S7-1200

Summary

Master Revision History 2
Overview 3
Making a correct layout 4
Setup with DL.Code 5
Step 1: enabling the protocol 5
Step 2: configuring the Operating Mode 7
Step 3: configuring the Output Bits 9
GSD file installation 10
Designing “Devices & Networks” 11
I/O configuration 11
Device Name 12
PROFINET-IO device name rules 12
Opening the example project 13
OB1: triggering cycle 14
Check the data traffic 16
(Most Common)Troubleshooting 18
Appendix A “Input & Output Bits in DL.Code” 19

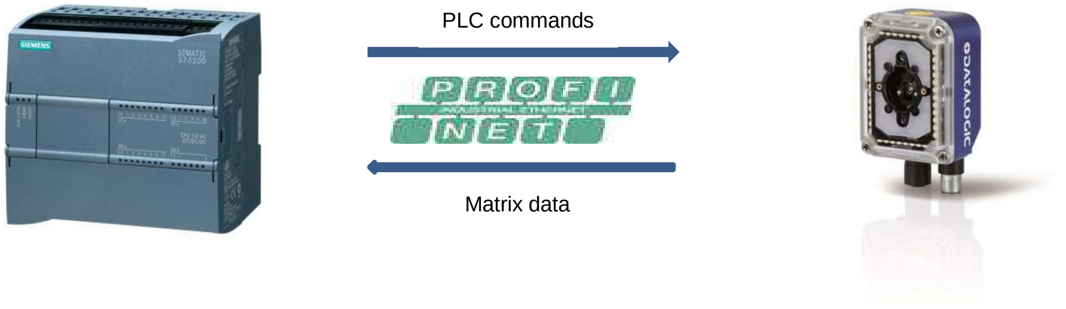
Master Revision History

Revision	Date	Author(s)	Change Description
0	18/7/2016	D.Natati	Preliminary Revision

Overview

This document is intended to lead the user to start and practice the communication between a Datalogic imager **MatrixN** device and a **Siemens S7-1x00 PLC** over **PROFINET-IO**.

Following the simple steps explained below the user will be able to run a complete PLC project, in order to capture barcode data and display them on a PLC.

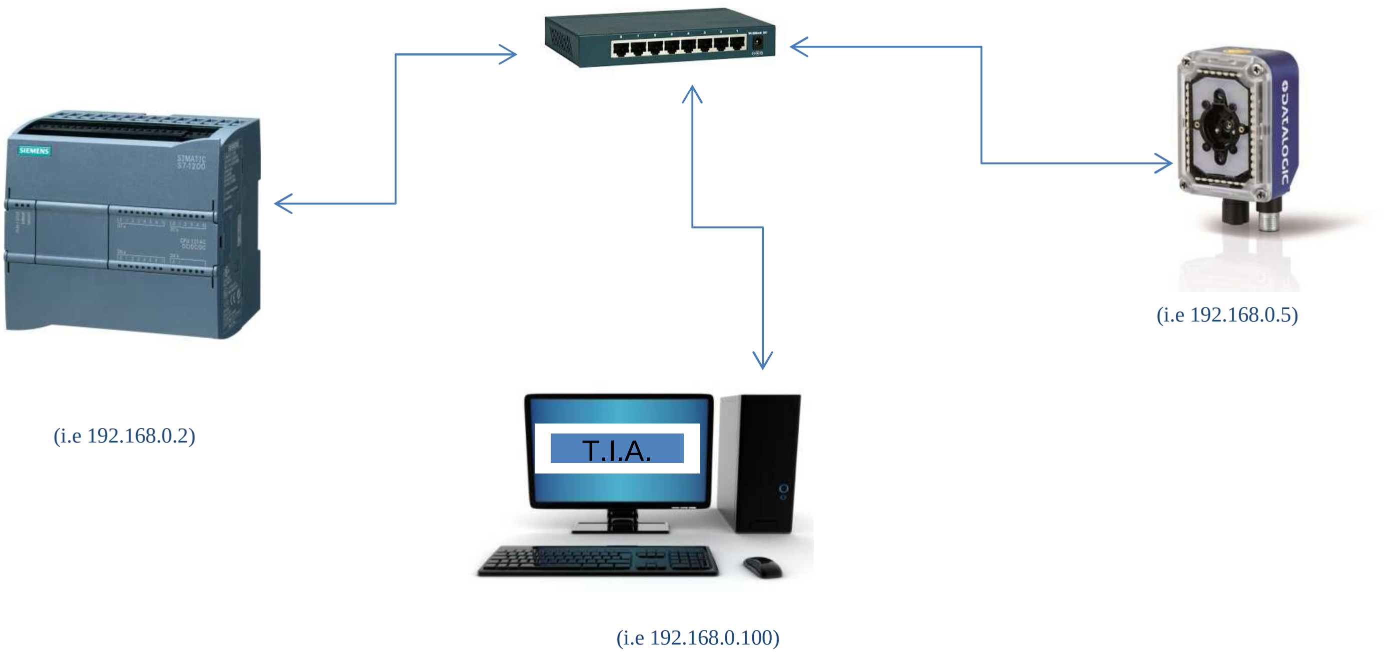


Referred items:

- **Matrix N device**: this document refers a Matrix300N device;
- **Matrix N Configuration Tool**: this document refers the DL.Code v1.4 configuration tool;
- **PLC**: this document refers a S7-1200 PLC;
- **PLC project:** “PNIO_ebd_Matrix_ReadTest”
- **PLC Configuration Tool**: this document refers the Step 7 T.I.A. v.12 configuration tool.

Making a correct layout

Connect a Matrix300N, a PLC and a Pc hosting the STEP-7 T.I.A. configuration tool over the same subnet (i.e through an Ethernet switch)

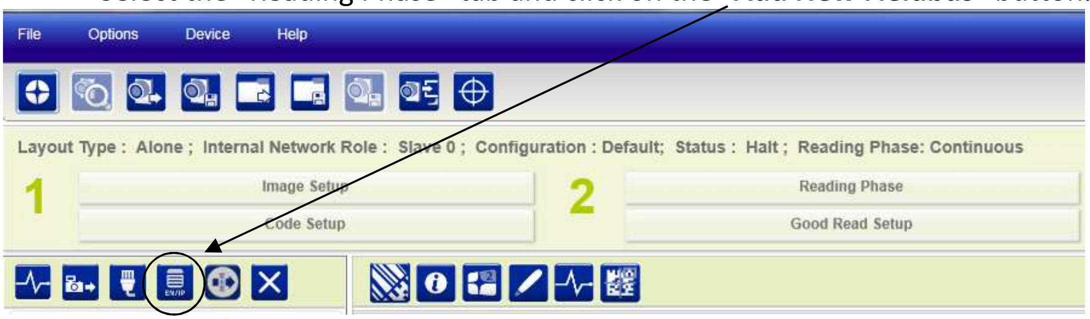


Setup with DL.Code

The picture below shows the needed setup to enable the PROFINET-IO driver on Matrix300N.

Step 1: enabling the protocol

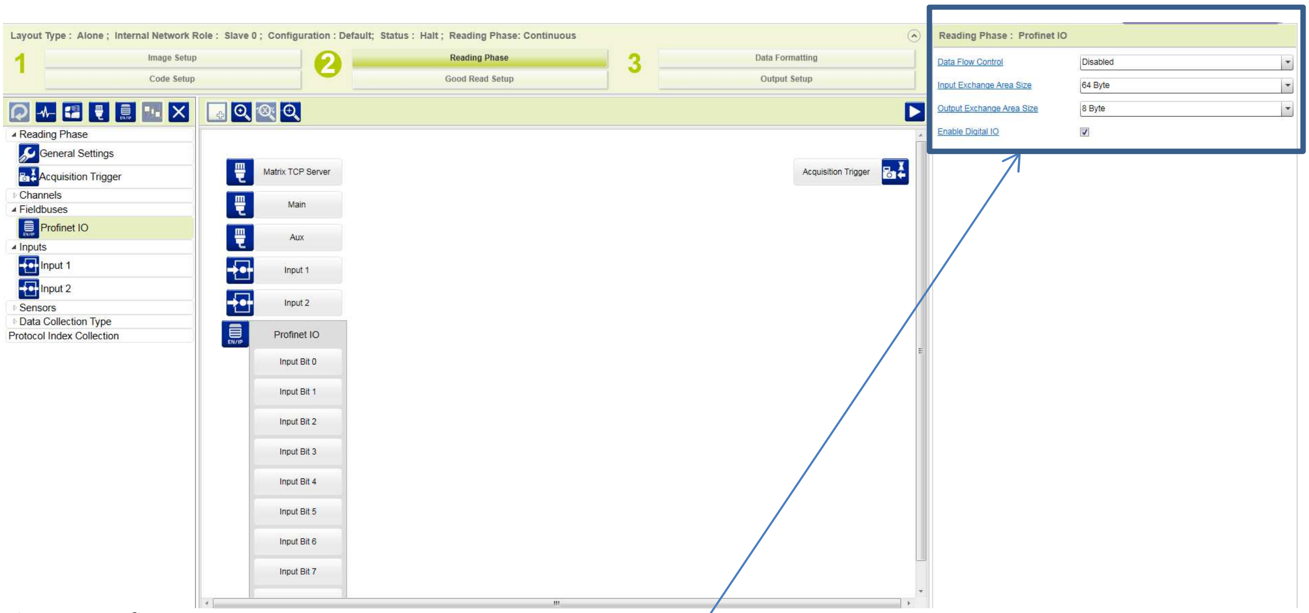
- Select the “Reading Phase” tab and click on the “**Add New Fieldbus**” button.



Layout Type : Alone ; Internal Network Role : Slave 0 ; Configuration : Default; Status : Halt ; Reading Phase: Continuous

Figure 1. adding a new fieldbus

- Select the “Profinet IO” option:



Layout Type : Alone ; Internal Network Role : Slave 0 ; Configuration : Default; Status : Halt ; Reading Phase: Continuous

Reading Phase : Profinet IO

Data Flow Control	Disabled
Input Exchange Area Size	64 Byte
Output Exchange Area Size	8 Byte
Enable Digital IO	☑

Figure 2. Profinet IO parameters

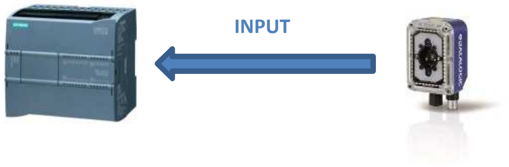
On the right side the “Reading Phase: **PROFINET IO**” section shows the default values:

- Data Flow Control = Disabled
- Input Exchange Area Size = 64 Byte ← “INPUT” means data incoming into the PLC
- Output Exchange Area Size = 8 Byte← “OUTPUT” means data coming out from the PLC
- Enable Digital IO: *<checked>*

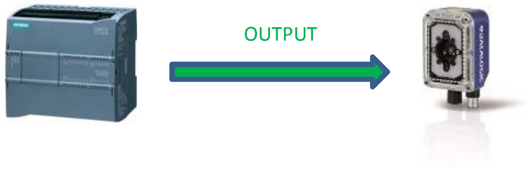
Mind that:

- “Data Flow Control” has two possible options:
    - Disabled → a control-free communication
    - DAD Flow Control → the communication is managed by the DAD protocol

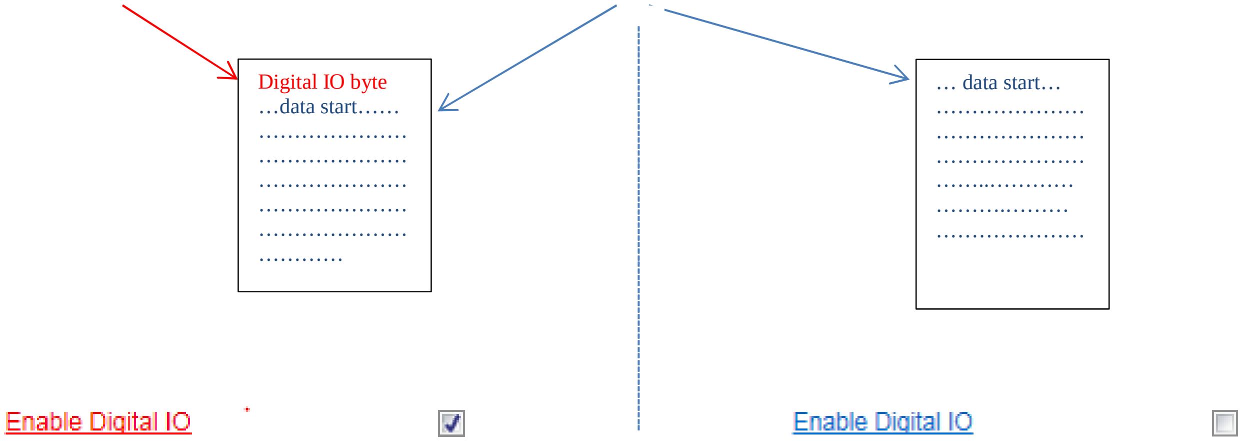
- ***Input** Exchange Area Size = 64 byte*
  → “**INPUT**” refers the point-of-view of the PLC, meaning data incoming into the PLC.
  “64” states the maximum amount of data PLC can get - in 1 CPU cycle - is 64 bytes.



- ***Output** Exchange Area Size = 8 byte*
  → “**OUTPUT**” refers the point-of-view of the PLC, meaning data out coming from the PLC.
  “8” states the maximum amount of data that PLC can send, in 1 CPU cycle, is 8 bytes.



- ***Enable Digital IO**: <checked>*
  This option enables/disables the fieldbus-control function of digital inputs, digital outputs and software trigger. <u>If enabled it reserves the first byte of the PLC data areas to the IO control functions and moves **1-byte down** the data start</u>.



Step 2: configuring the Operating Mode

- Selecting the “Reading Phase” tab and click on the “**Phase Mode**” button, in order to define an “on-off controlled” operating mode

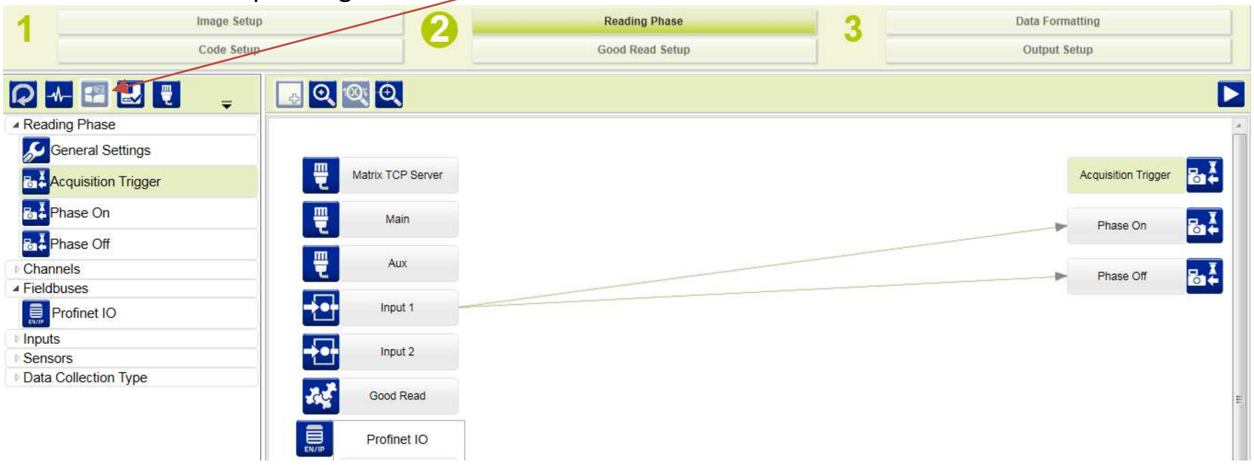
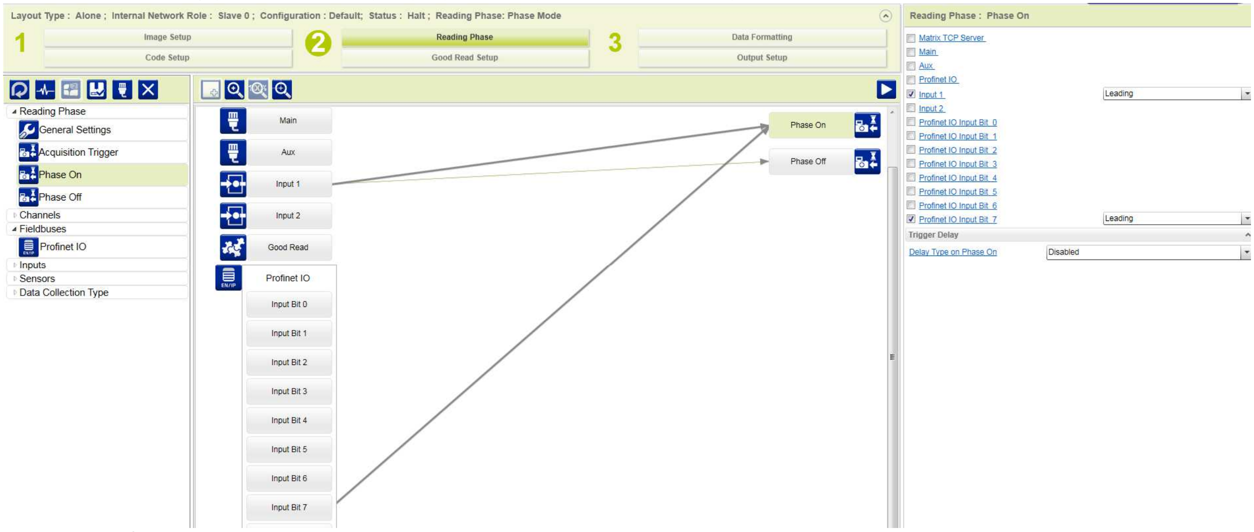


Figure 3. selecting the “Phase Mode” operating mode

- Configuring the “**Phase On**” option as controlled both by “**Input1**” and “**Input bit 7**” events, to drive the reading phase start by a physical input or a PLC Output bit[1]



Layout Type : Alone ; Internal Network Role : Slave 0 ; Configuration : Default; Status : Halt ; Reading Phase: Phase Mode

Reading Phase : Phase On

☐ Matrix TCP Server
☐ Main
☐ Aux
☐ Profinet IO
☑ Input 1 — Leading
☐ Input 2
☐ Profinet IO Input Bit 0
☐ Profinet IO Input Bit 1
☐ Profinet IO Input Bit 2
☐ Profinet IO Input Bit 3
☐ Profinet IO Input Bit 4
☐ Profinet IO Input Bit 5
☐ Profinet IO Input Bit 6
☑ Profinet IO Input Bit 7 — Leading

Trigger Delay

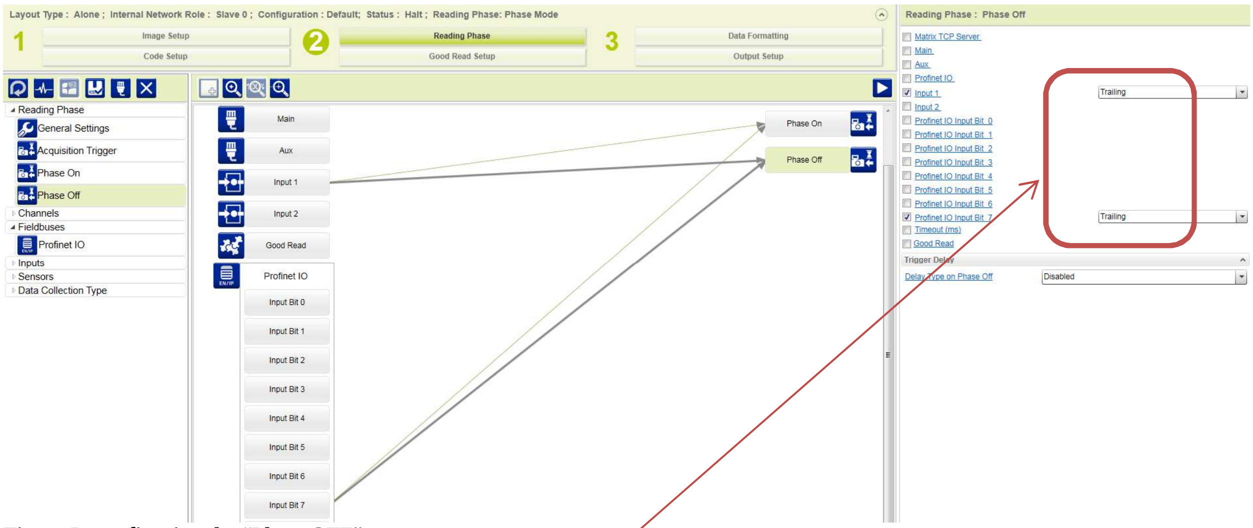
Delay Type on Phase On — Disabled

Figure 4. configuring the “Phase ON”

This setup configures the leading edge of bit7 in the first Output byte of PLC as Phase On start.

[1] Go to Appendix A “Input & Output Bits in DL.Code” to find out more.

- Configuring the "**Phase Off**" option as controlled both by the "**Input1**" and "**Input bit 7**" events, to drive the reading phase stop by a physical input or a PLC Output bit[2]



Layout Type : Alone ; Internal Network Role : Slave 0 ; Configuration : Default; Status : Halt ; Reading Phase: Phase Mode

Reading Phase : Phase Off

- ☐ Matrix TCP Server
- ☐ Main
- ☐ Aux
- ☐ Profinet IO
- ☑ Input 1 — Trailing
- ☐ Input 2
- ☐ Profinet IO Input Bit 0
- ☐ Profinet IO Input Bit 1
- ☐ Profinet IO Input Bit 2
- ☐ Profinet IO Input Bit 3
- ☐ Profinet IO Input Bit 4
- ☐ Profinet IO Input Bit 5
- ☐ Profinet IO Input Bit 6
- ☑ Profinet IO Input Bit 7 — Trailing
- ☐ Timeout (ms)
- ☐ Good Read

Trigger Delay

Delay Type on Phase Off: Disabled

Figure 5. configuring the "Phase OFF"

This setup configures the trailing edge of bit7 in the first Output byte of PLC as Phase Off start.

Take care about the "**Leading**" or "**Trailing**" edge option, it must be set not to cause an activation conflict.

Do not set the same edge on the same de/activation event for related parameters.

Examples:

~~"Phase ON" from "Input1-**Leading** Edge"~~
~~"Phase OFF" from "Input1-**Leading** Edge"~~

NO!

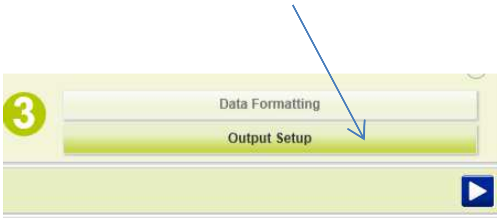
"Phase ON" from "Input1-**Leading** Edge"
"Phase OFF" from "Input1-**Trailing** Edge"

YES

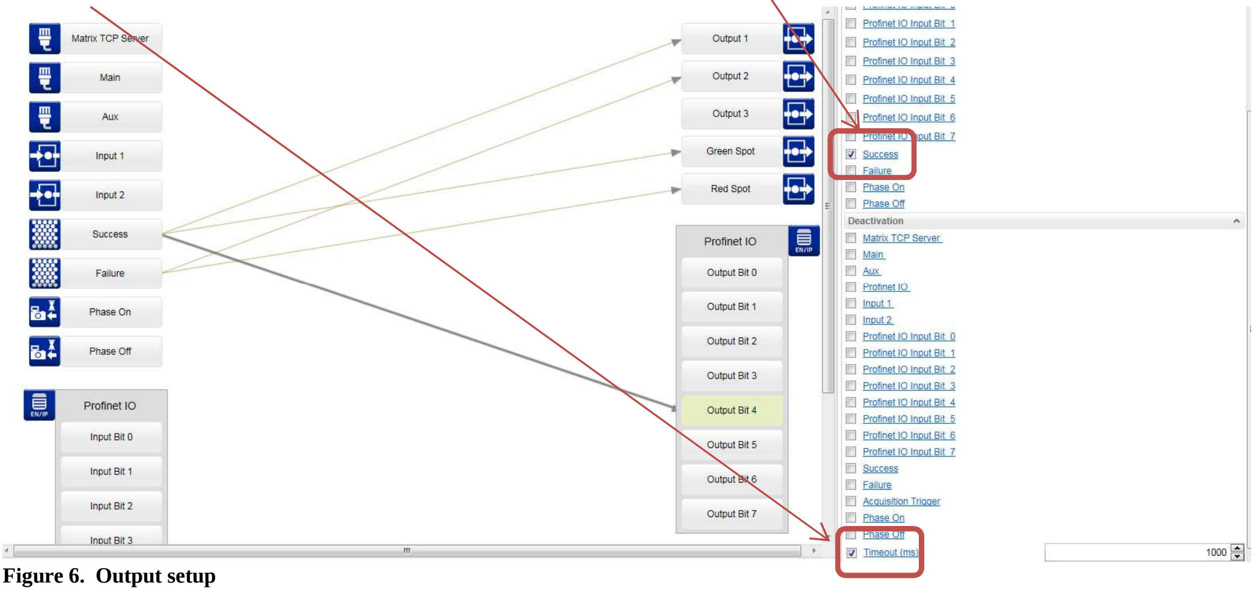
[2] Go to Appendix A "Input & Output Bits in DL.Code" to find out more

Step 3: configuring the Output Bits

Selecting the “Output Setup” tab to drive a PLC input bit by the reading result.



The setup below links the Profinet-IO Output bit 4[3] to the “Success” event, the deactivation event is a “Timeout”.



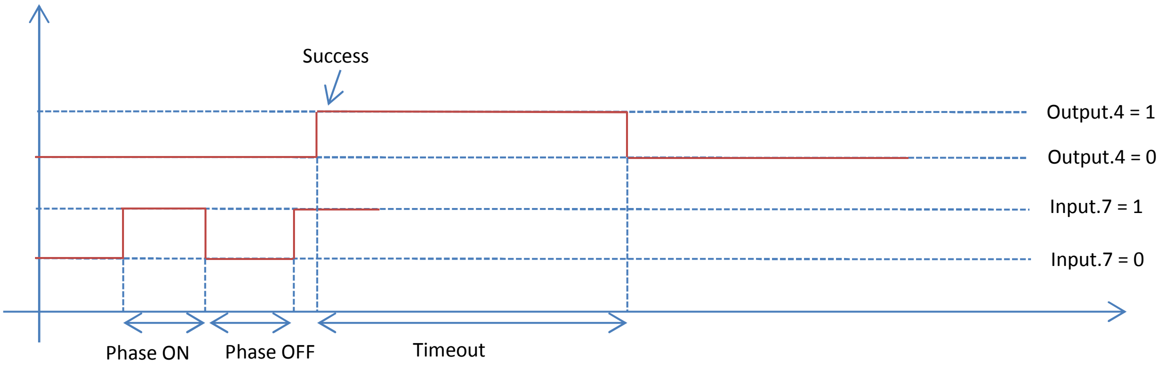
☐ Profinet IO Input Bit 1
☐ Profinet IO Input Bit 2
☐ Profinet IO Input Bit 3
☐ Profinet IO Input Bit 4
☐ Profinet IO Input Bit 5
☐ Profinet IO Input Bit 6
☐ Profinet IO Input Bit 7
☑ Success
☐ Failure
☐ Phase On
☐ Phase Off

Deactivation

☐ Matrix TCP Server
☐ Main
☐ Aux
☐ Profinet IO
☐ Input 1
☐ Input 2
☐ Profinet IO Input Bit 0
☐ Profinet IO Input Bit 1
☐ Profinet IO Input Bit 2
☐ Profinet IO Input Bit 3
☐ Profinet IO Input Bit 4
☐ Profinet IO Input Bit 5
☐ Profinet IO Input Bit 6
☐ Profinet IO Input Bit 7
☐ Success
☐ Failure
☐ Acquisition Trigger
☐ Phase On
☐ Phase Off
☑ Timeout (ms) 1000

Figure 6. Output setup

IO configuration summary diagram:



[3] Go to Appendix A “PROFINET-IO Input & Output in DL.Code” to find out more.

GSD file installation

Copy the content of the zip package “GSDML_Matrix_20140430” on a local directory.
Select on the main bar “Options-Install General Station Description file (GSD)”.

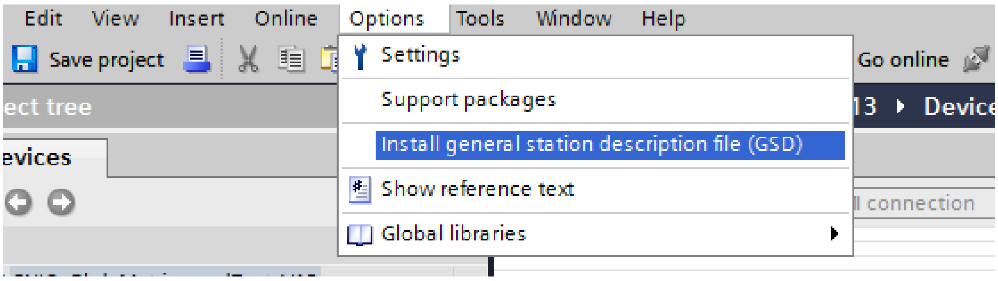


Figure 7. Installing the GSD file

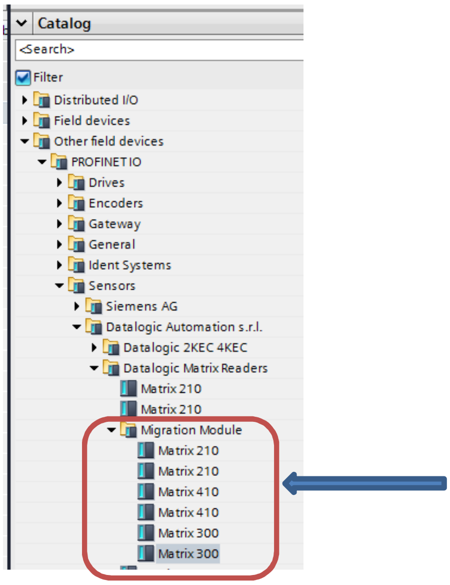
From that local directory select the file: **GSDML-V2.1-Datalogic-Matrix-PNIO-20140430**[4] then confirm the loading.

After the installation has been completed, the new Matrix nodes and related icons appear in the PLC HW catalogue under the folders:

- Other Field Devices/PROFINET IO/Sensors/Datalogic Automation s.r.l./Datalogic Matrix Readers

and

- Other Field Devices/PROFINET IO/Sensors/Datalogic Automation s.r.l./Datalogic Matrix Readers/Migration Module



Select the nodes from the “Migration Module” sub-folder

Figure 8. updated HW catalogue

[4] This is the latest available GSDML file for MatrixN series devices. It could be replaced by Datalogic with next file versions.

Designing "Devices & Networks"

Select the node (matrix) and connect it to the PLC (plc_1) over the PROFINET IO network (PN/IE_1)

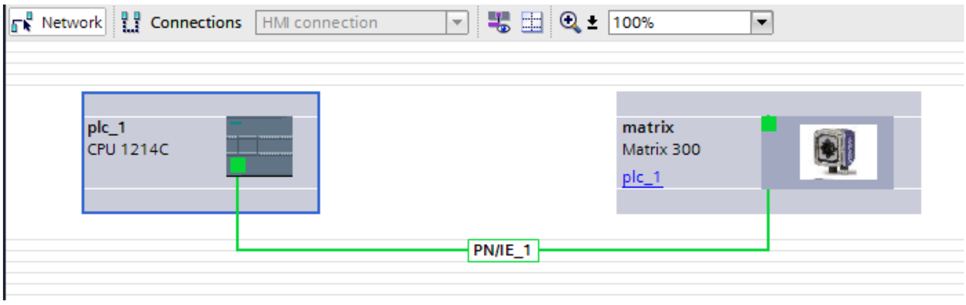
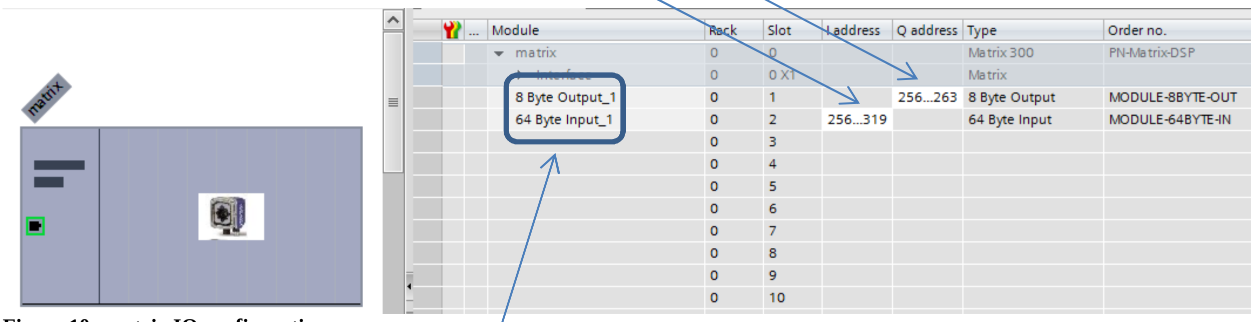


Figure 9. Network view

I/O configuration

- The (default) I/O configuration is as follows:
- 8 byte Output, addressed at 256..263
- 64 byte Input, addressed at 256..319

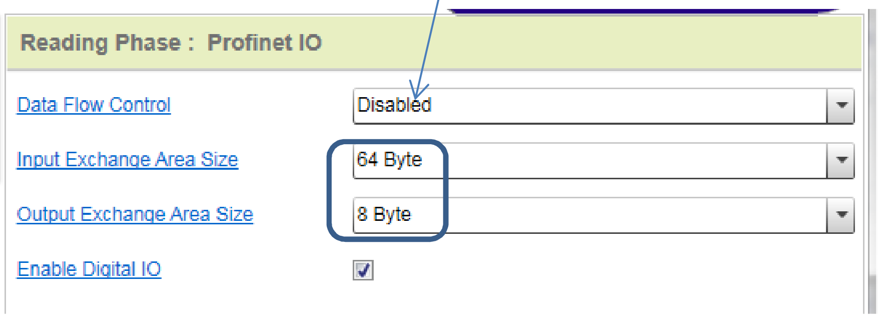


	...	Module	Rack	Slot	I address	Q address	Type	Order no.
		matrix	0	0			Matrix 300	PN-Matrix-DSP
		Interface	0	0 X1			Matrix	
		8 Byte Output_1	0	1		256...263	8 Byte Output	MODULE-8BYTE-OUT
		64 Byte Input_1	0	2	256...319		64 Byte Input	MODULE-64BYTE-IN
			0	3				
			0	4				
			0	5				
			0	6				
			0	7				
			0	8				
			0	9				
			0	10				

Figure 10. matrix IO configuration

Take care:

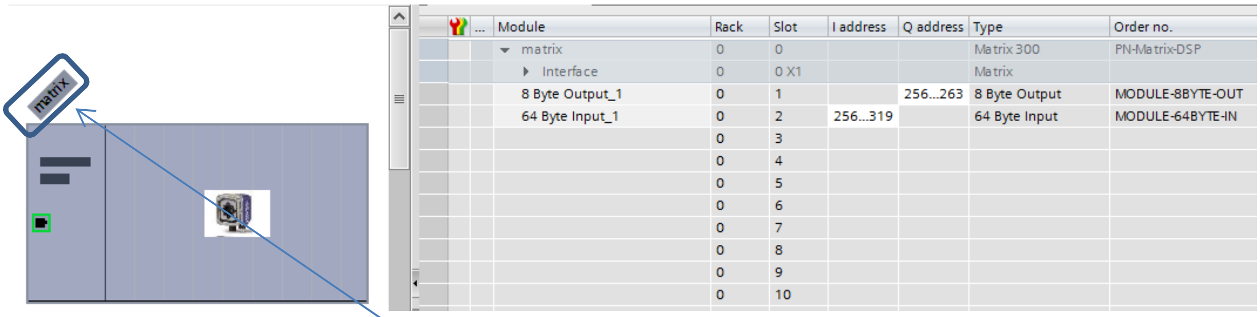
- the IO configuration size – **64 & 8** – must MATCH the DL.Code Profinet-IO size configuration, else the PLC signals a configuration error.



Reading Phase : Profinet IO

Data Flow Control	Disabled
Input Exchange Area Size	64 Byte
Output Exchange Area Size	8 Byte
Enable Digital IO	☑

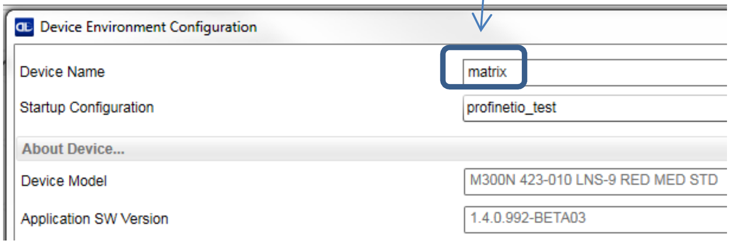
Device Name



	Module	Rack	Slot	I address	Q address	Type	Order no.
	matrix	0	0			Matrix 300	PN-Matrix-DSP
	Interface	0	0 X1			Matrix	
	8 Byte Output_1	0	1		256...263	8 Byte Output	MODULE-8BYTE-OUT
	64 Byte Input_1	0	2	256...319		64 Byte Input	MODULE-64BYTE-IN
		0	3				
		0	4				
		0	5				
		0	6				
		0	7				
		0	8				
		0	9				
		0	10				

Take care:

- The device name (here it is “**matrix**”) MUST match the “Device name” set by DL.Code, else the PLC signals a configuration error



Device Environment Configuration

Device Name	matrix
Startup Configuration	profinetio_test

About Device...

Device Model	M300N 423-010 LNS-9 RED MED STD
Application SW Version	1.4.0.992-BETA03

Figure 11. DL.Code matrix settings

PROFINET-IO device name rules

xxxxxxx.xxxxxxxxx-xxxxxxxx

- allowed characters: lower case letters, numbers, “.”, “-”
- “.” and “-” cannot be used at the beginning or at the end of the name
- Maximum length = 240 characters

Examples:

1. *this-is.a.good.name*
2. *this is-not-a-good-name*
3. *device-123-is-ok*
4. *.device123-is-not-ok*

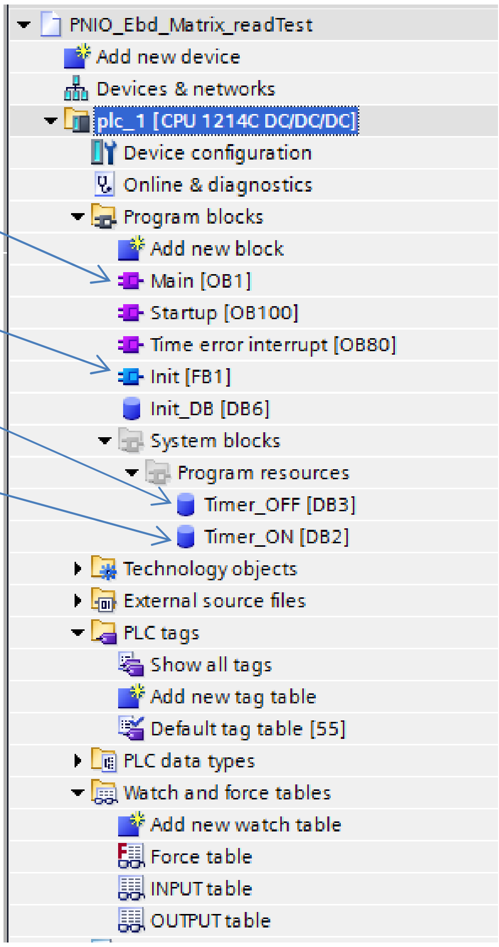
Opening the example project

This section shows how to use the referred PLC project “PNIO_ebd_Matrix_ReadTest”; it implements an endless reading loop for devices of the “Matrix 2x0/3x0/4x0” series.

The PLC is a **Siemens S7-1200 (CPU 1214 DC/DC/DC)**, the fieldbus in use is **PROFINET-IO**.

The project basic items are:

- Main(OB1):
  It includes the trigger cycle
- Init(FB1):
  Timers initialization
- Timer_OFF(DB3):
  It sets the phase off interval
- Timer_ON(DB2):
  It sets the phase on interval



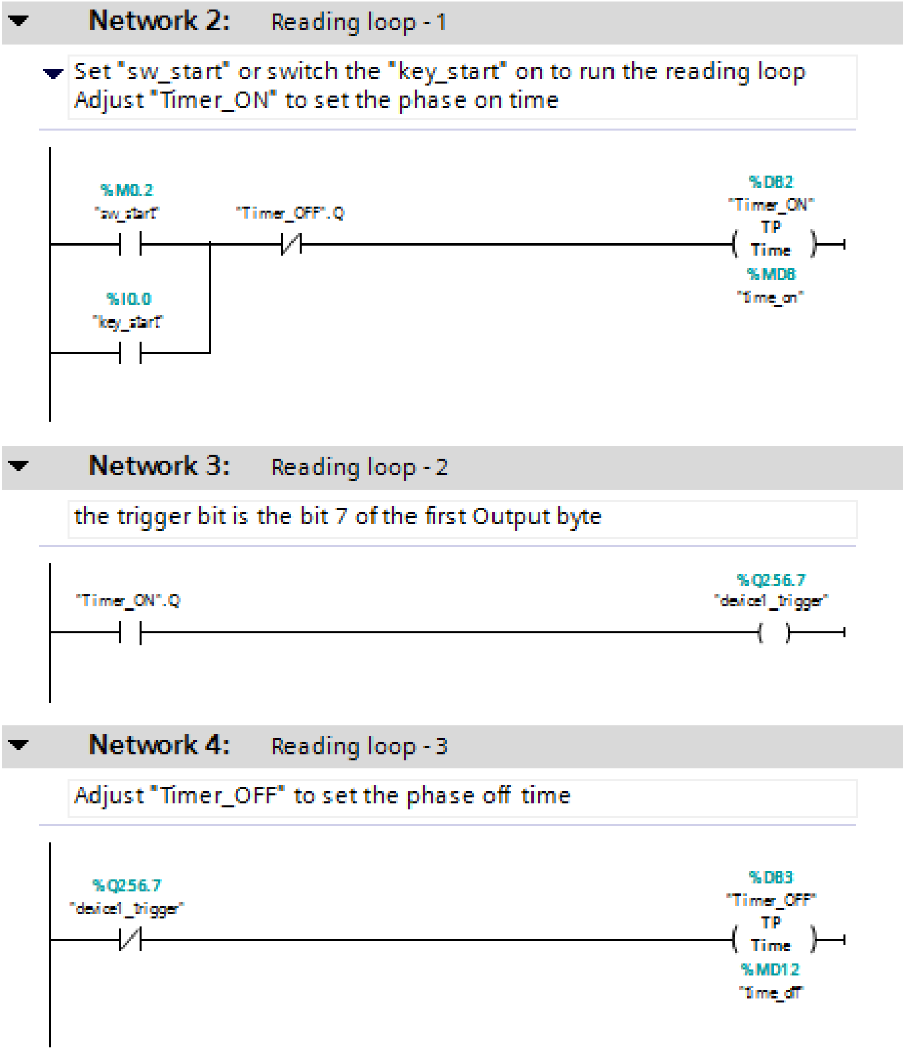
```
PNIO_Ebd_Matrix_readTest
    Add new device
    Devices & networks
    plc_1 [CPU 1214C DC/DC/DC]
        Device configuration
        Online & diagnostics
        Program blocks
            Add new block
            Main [OB1]
            Startup [OB100]
            Time error interrupt [OB80]
            Init [FB1]
            Init_DB [DB6]
            System blocks
                Program resources
                    Timer_OFF [DB3]
                    Timer_ON [DB2]
        Technology objects
        External source files
        PLC tags
            Show all tags
            Add new tag table
            Default tag table [55]
        PLC data types
        Watch and force tables
            Add new watch table
            Force table
            INPUT table
            OUTPUT table
```

Figure 12. PLC project logical blocs

OB1: triggering cycle

Basically the Main block provides a triggering cycle to the device in order to control the reading phase.

See networks 2, 3 and 4 of OB1:



Network 2: Reading loop - 1

Set "sw_start" or switch the "key_start" on to run the reading loop
Adjust "Timer_ON" to set the phase on time

Network 3: Reading loop - 2

the trigger bit is the bit 7 of the first Output byte

Network 4: Reading loop - 3

Adjust "Timer_OFF" to set the phase off time

Figure 12. Project reading loop

These networks implement an endless cycle which set and reset the “**device1_trigger**” signal. It is the **Q256.7 bit**, it’s the software trigger for the device according to the DL.Code setup.

The phase ON and phase OFF intervals are controlled by the **Timer_ON** and **Timer_OFF** timer, their value are set by the “Init” function block (FB1).

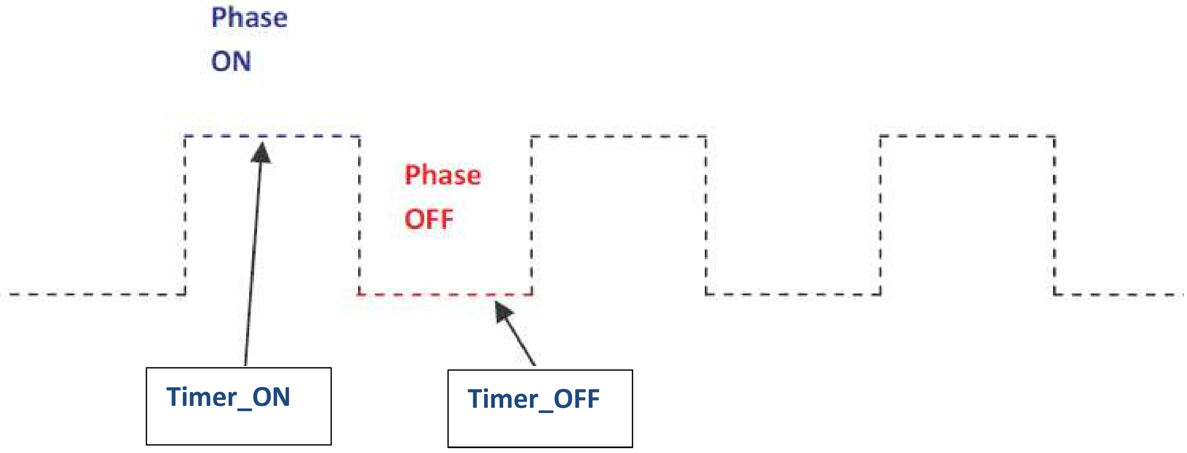
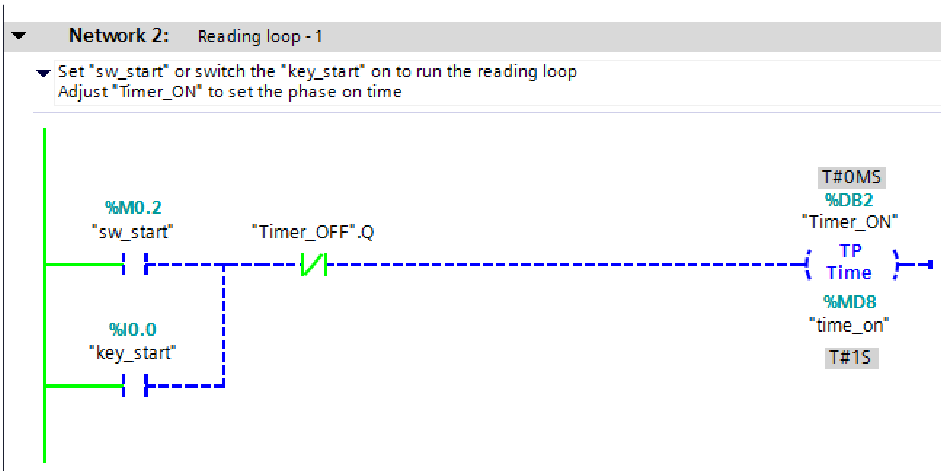


Figure 13. Reading loop timing

In order to start the reading loop, set the “sw_start” flag or switch the digital input “key_start”(**I0.0**).


Network 2: Reading loop - 1

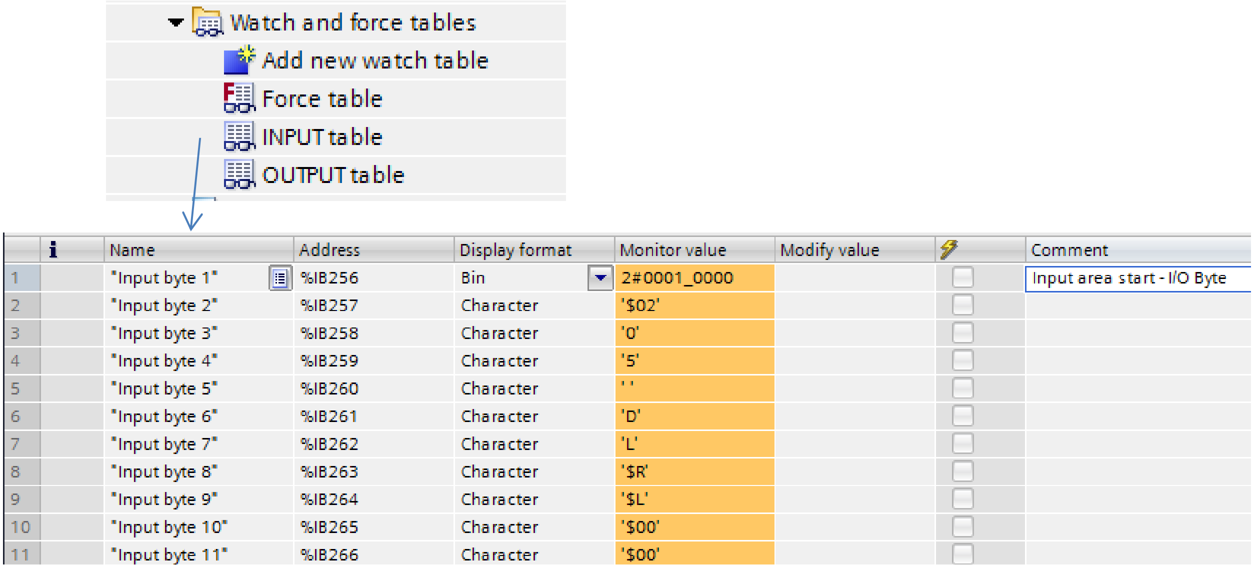
Set "sw_start" or switch the "key_start" on to run the reading loop
Adjust "Timer_ON" to set the phase on time

Figure 14. Reading loop start – monitoring ON

Check the data traffic

In order to check the current status of the I/O data exchange between reader and PLC, watch tables are available and very useful. On the project block list select and open:

- "**InputTable**", already designed to check the first 64 input bytes
- "**OutputTable**" ,already designed to check the first 8 output bytes



	i	Name	Address	Display format	Monitor value	Modify value	⚡	Comment
1		"Input byte 1"	%IB256	Bin	2#0001_0000		□	Input area start - I/O Byte
2		"Input byte 2"	%IB257	Character	'$02'		□	
3		"Input byte 3"	%IB258	Character	'0'		□	
4		"Input byte 4"	%IB259	Character	'5'		□	
5		"Input byte 5"	%IB260	Character	' '		□	
6		"Input byte 6"	%IB261	Character	'D'		□	
7		"Input byte 7"	%IB262	Character	'L'		□	
8		"Input byte 8"	%IB263	Character	'$R'		□	
9		"Input byte 9"	%IB264	Character	'$L'		□	
10		"Input byte 10"	%IB265	Character	'$00'		□	
11		"Input byte 11"	%IB266	Character	'$00'		□	

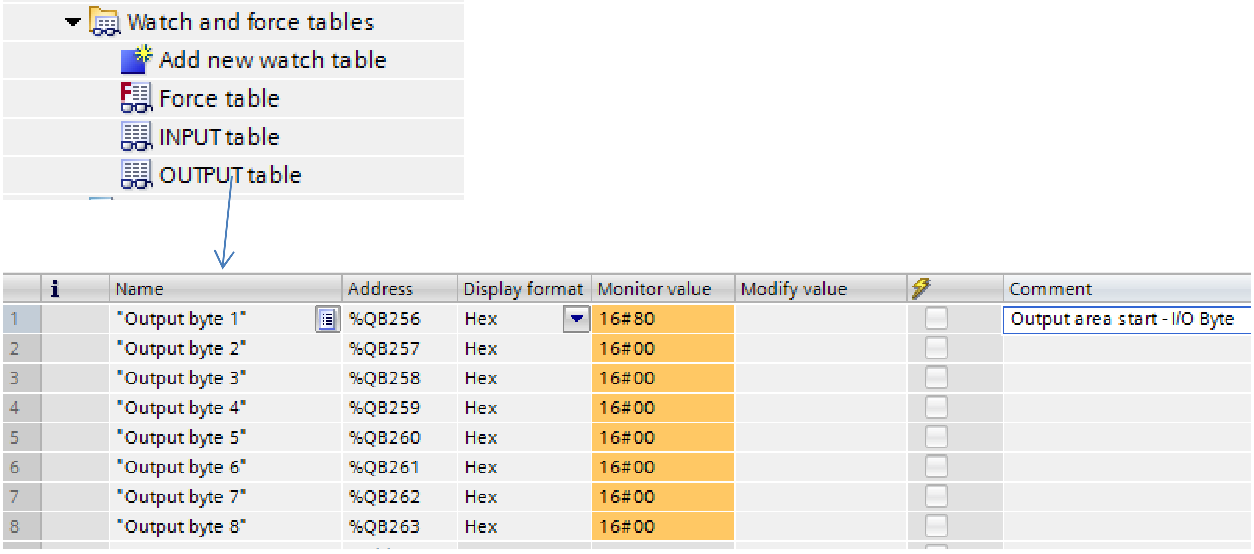
Figure 15. INPUT table

Above a snapshot of the Input table during the I/O handshake.
Note that:

1. **IB256: I/O byte** : it is reserved for the I/O control functions. See pag. 6 "Enable Digital IO" option and Appendix A to find out more.

2. **IB257…..: data bytes:** data string the device sent to the PLC.

The bar code data are: *<STX>05<space> DL<CR><LF>*



- Watch and force tables
  - Add new watch table
  - Force table
  - INPUT table
  - OUTPUT table

	i	Name	Address	Display format	Monitor value	Modify value	⚡	Comment
1		"Output byte 1"	%QB256	Hex	16#80		☐	Output area start - I/O Byte
2		"Output byte 2"	%QB257	Hex	16#00		☐	
3		"Output byte 3"	%QB258	Hex	16#00		☐	
4		"Output byte 4"	%QB259	Hex	16#00		☐	
5		"Output byte 5"	%QB260	Hex	16#00		☐	
6		"Output byte 6"	%QB261	Hex	16#00		☐	
7		"Output byte 7"	%QB262	Hex	16#00		☐	
8		"Output byte 8"	%QB263	Hex	16#00		☐	

Figure 16. OUTPUT table

Above a snapshot of the Output table during the I/O handshake.
Note that:

1. **QB256: I/O byte** : it is reserved for the I/O control functions. See pag. 6 "Enable Digital IO" option to find out more.
   - QB256.7 = 1: it's the trigger bit, set to "1" activates the reading phase
2. **QB257.....: command bytes:** data string the PLC (eventually) sends to the node.

(Most Common)Troubleshooting

Trouble	Cause	Action
PLC does not recognize the node	GSD file not correct	Delete the node, install the correct gsd file and draw the HW configuration again
PLC does not recognize the node	Node NOT selected from the folder “Migration Module”	Delete the node and replace it with a node from the correct folder, then draw the HW configuration again
Reader not triggered	Trigger bit not mapped on the DL.Code configured bit	Check the DL.Code configuration and re-assign the trigger bit
Reader not triggered	Trigger bit not mapped on the correct Output area	Check the PLC HW configuration and re-assign the trigger bit

Appendix A “Input & Output Bits in DL.Code”

The picture below shows an example of the “Output Setup” page.

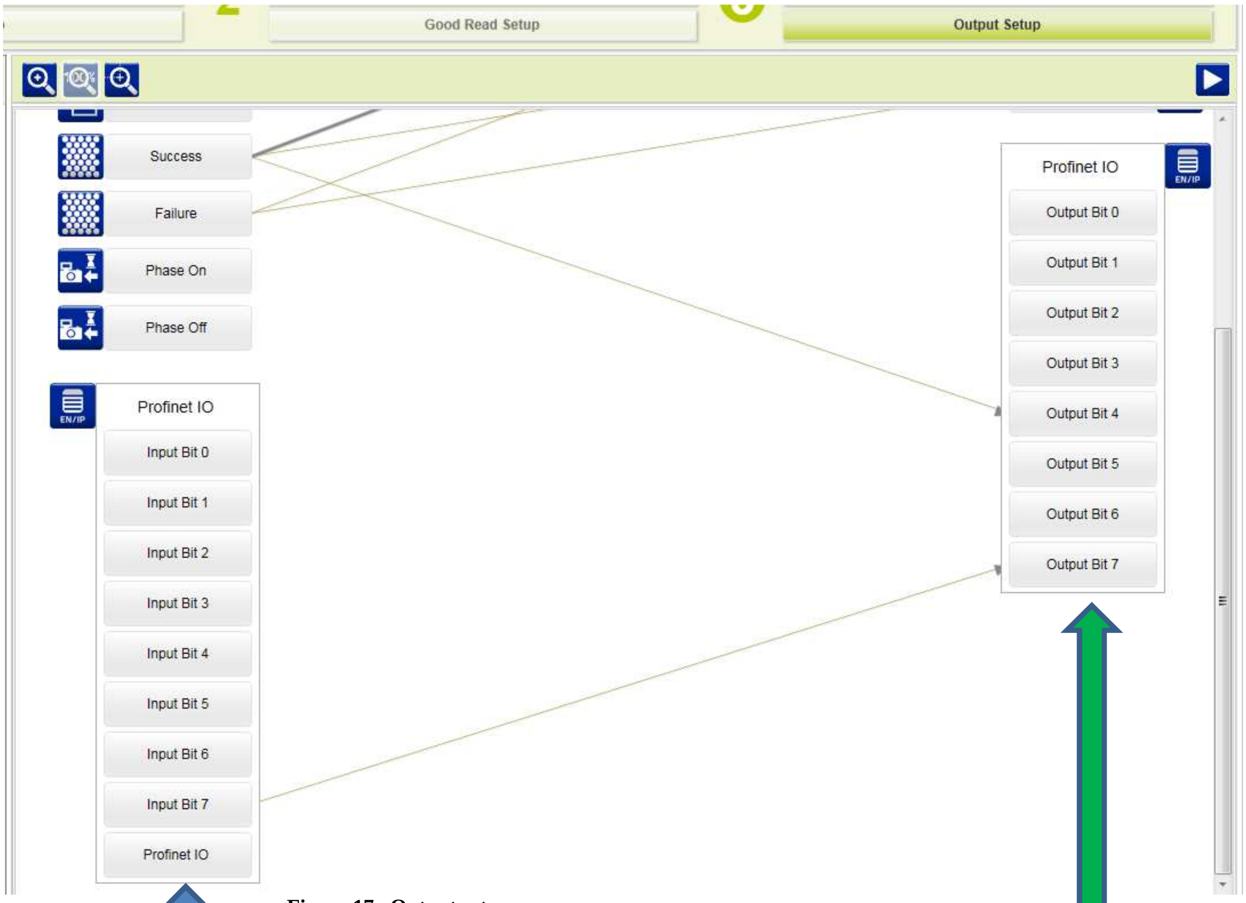


Figure 17. Output setup

The “Profinet IO” column on the left side contains 8 “**Input Bits**”.

The “Profinet IO” column on the right side contains 8 “**Output Bits**”.

The Profinet IO Input bits are mapped one-to-one on the 1st byte of the PLC Output area if:

- “Enable Digital IO = *<checked>*”
- the correct activation/deactivation events have been set (note Input Bit 7 linked to Output Bit 7 in the picture 17)

A PLC can write the Input bits through the bits of 1st byte of the Output area.

Example:
the user needs to have the 1st Input byte.bit 7 (incoming data) as echo of the 1st Output byte.bit 7 (command out from PLC)

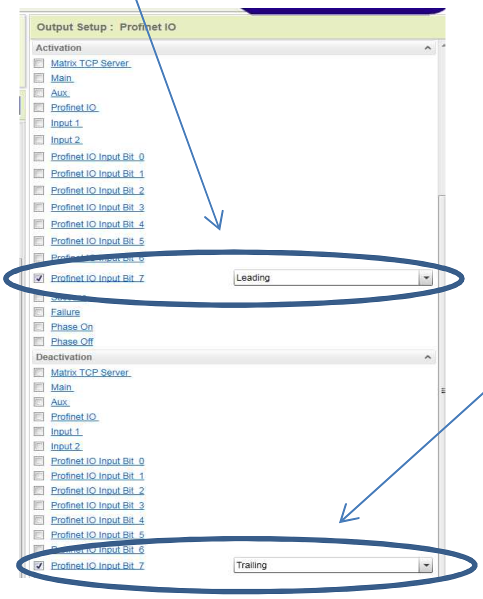
Steps:

Enable Digital IO ☑

1. set: "Enable Digital IO = *<checked>*"

2. "Output Bit 7" activation = Profinet IO Input Bit 7 - leading

3. "Output Bit 7" deactivation = Profinet IO Input Bit 7 - trailing

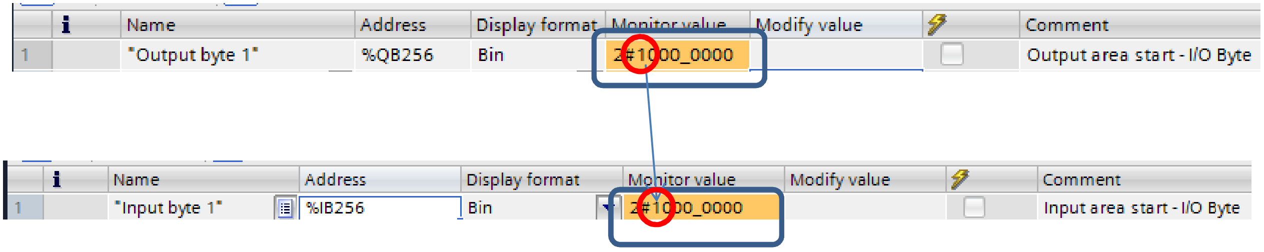


Output Setup : Profinet IO

Activation
- □ Matrix TCP Server
- □ Main
- □ Aux
- □ Profinet IO
- □ Input 1
- □ Input 2
- □ Profinet IO Input Bit 0
- □ Profinet IO Input Bit 1
- □ Profinet IO Input Bit 2
- □ Profinet IO Input Bit 3
- □ Profinet IO Input Bit 4
- □ Profinet IO Input Bit 5
- □ Profinet IO Input Bit 6
- ☑ Profinet IO Input Bit 7 — Leading
- □ Success
- □ Failure
- □ Phase On
- □ Phase Off

Deactivation
- □ Matrix TCP Server
- □ Main
- □ Aux
- □ Profinet IO
- □ Input 1
- □ Input 2
- □ Profinet IO Input Bit 0
- □ Profinet IO Input Bit 1
- □ Profinet IO Input Bit 2
- □ Profinet IO Input Bit 3
- □ Profinet IO Input Bit 4
- □ Profinet IO Input Bit 5
- □ Profinet IO Input Bit 6
- ☑ Profinet IO Input Bit 7 — Trailing

So if PLC sets the bit 7 of the 1st Output byte….



	i	Name	Address	Display format	Monitor value	Modify value	⚡	Comment
1		"Output byte 1"	%QB256	Bin	2#1000_0000		□	Output area start - I/O Byte

	i	Name	Address	Display format	Monitor value	Modify value	⚡	Comment
1		"Input byte 1"	%IB256	Bin	2#1000_0000		□	Input area start - I/O Byte

….. the bit 7 of the 1st Input byte gets the "1" value

Likewise if PLC resets the bit 7 of the 1st Output byte, the bit 7 of the 1st Input byte gets the "0" value.