



Siemens
Industry
Online
Support

APPLICATION EXAMPLE

Control of inspection trigger of SIEMENS Inspekto VIS via SIMATIC PLC

SIEMENS Inspekto VIS / PROFINET / TIA Portal / GSDML / S7-1500 / Unified Comfort Panel

SIEMENS

Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. **The application examples are not subject to standard tests and quality inspections of a chargeable product and may contain functional and performance defects or other faults and security vulnerabilities. You are responsible for the proper and safe operation of the products in accordance with all applicable regulations, including checking and customizing the application example for your system, and ensuring that only trained personnel use it in a way that prevents property damage or injury to persons. You are solely responsible for any productive use.**

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. Any further use of the application examples is explicitly not permitted and further rights are not granted. You are not allowed to use application examples in any other way, including, without limitation, for any direct or indirect training or enhancements of AI models.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice and to terminate your use of the application examples at any time. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (<https://www.siemens.com/global/en/general/terms-of-use.html>) shall also apply.

Cybersecurity information

Siemens provides products and solutions with industrial cybersecurity functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial cybersecurity concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial cybersecurity measures that may be implemented, please visit www.siemens.com/cybersecurity-industry.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Cybersecurity RSS Feed under <https://www.siemens.com/cert>.

Table of contents

1.	Introduction	5
1.1.	Overview	5
1.1.1.	Vision inspection systems	5
1.1.2.	SIEMENS Inspekto VIS	5
1.1.3.	What is in this manual.....	6
2.	Engineering	8
2.1.	Components used.....	8
2.2.	Hardware Setup.....	9
2.2.1.	SIEMENS Inspekto VIS System Connection	9
2.2.2.	Complete setup connection	10
2.3.	Project planning	10
2.4.	Configuration	11
2.4.1.	Inspekto configuration.....	11
2.4.2.	TIA Portal configuration	11
2.5.	Operation.....	18
2.5.1.	Integrating SIEMENS Inspekto VIS function blocks	18
2.5.2.	Testing configuration and PLC-Inspekto connectivity.....	23
2.5.2.1.	Inspekto → PLC feedback screen	24
2.5.2.2.	PLC → Inspekto feedback screen	26
2.5.3.	Trigger control using TIA Portal function blocks.....	27
2.5.3.1.	PLC programming – Trigger inspection	28
2.5.3.2.	HMI programming – Inspection result visualization	36
3.	Additional information	39
3.1.	Inspekto Functions and TIA Portal Function blocks	39
3.1.1.	PLC -> SIEMENS INSPEKTO VIS functions	39
3.1.1.1.	OUT_INSPECTION_MODE (1 byte).....	40
3.1.1.2.	OUT_NEW_OBJECT_TRIGGER (64 bytes null-terminated ASCII string).....	43
3.1.1.3.	OUT_PROFILE_ID (16-bit integer).....	43
3.1.1.4.	OUT_BATCH_ID (64 bytes null-terminated ASCII string)	44
3.1.1.5.	OUT_PLC_SHUTDOWN	44
3.1.2.	SIEMENS Inspekto VIS -> PLC functions	45
3.1.2.1.	IN_CUR_INSPECTION_MODE.....	45
3.1.2.2.	IN_INSPECTION_MODE_RESPONSE	45
3.1.2.3.	IN_CUR_OBJECT_ID	45
3.1.2.4.	IN_NEW_OBJECT_TRIGGER_RESPONSE	46
3.1.2.5.	IN_CUR_PROFILE_ID	46
3.1.2.6.	IN_PROFILE_ID_RESPONSE	46
3.1.2.7.	IN_CUR_BATCH_ID	46

3.1.2.8.	IN_BATCH_ID_RESPONSE.....	46
3.1.2.9.	IN_KEEP_ALIVE	46
3.1.2.10.	IN_OBJECT_INSPECTION_RESULT	47
3.1.2.11.	IN_SYSTEM_ERROR	47
3.1.2.12.	IN_SYSTEM_READY	48
3.1.2.13.	IN_ROIS_DEFECT_LIST	48
3.1.2.14.	IN_PLC_SHUTDOWN_RECEIVED	48
3.2.	SIEMENS Inspekto VIS function blocks	49
3.2.1.	New Batch ID (FBUpdateBatchID)	49
3.2.2.	Change inspection mode (FBUpdateInspectionMode).....	50
3.2.3.	Load profile (FBUpdateProfileID)	51
3.2.4.	Trigger inspection (FBNewObjectTrigger)	52
3.3.	Signal Flow Charts	54
3.3.1.	OUT_NEW_TRIGGER	54
3.3.2.	OUT_PROFILE_ID	55
3.3.3.	OUT_INSPECTION_MODE	56
3.3.4.	OUT_BATCH_ID.....	57
3.4.	System Installation	58
3.4.1.	Updating TIA Portal integration to Inspekto v6.1	58
3.4.2.	System status after Inspekto update to version 6.1	58
3.4.3.	Installing GSDML compatible with version 6.1	58
3.4.4.	Replace Inspekto device	59
4.	Appendix	61
4.1.	Service and support	61
4.2.	Application support	62
4.3.	Links and literature	62
4.4.	Documentation change	62

1. Introduction

1.1. Overview

1.1.1. Vision inspection systems

In today's automated production lines, vision inspection processes are indispensable for ensuring product quality and consistency. Vision inspection processes allow defects detection, verify assembly accuracy, and maintain high standards of quality control.

Historically, vision inspection was mainly performed manually by human operators, nevertheless it is still very frequently present in nowadays highly automated production lines. This method relies heavily on the skill and attention of workers to identify defects and ensure the product quality. Manual inspection is time-consuming, prone to human error, and limited by the inspector's ability to consistently detect subtle issues. As manufacturing processes evolved and the demands towards production are constantly growing, the need for more reliable and efficient inspection methods becomes apparent.

By automating the inspection process, manufacturers can significantly reduce human error, enhance production efficiency, and ensure that each product meets stringent quality requirements. The integration of vision inspection systems into automation lines not only improves product reliability but also adds substantial value by streamlining operations and reducing costs.

Early automated vision systems utilized basic image processing techniques to assist human inspectors. Over time, advancements in technology led to the development of sophisticated AI-based vision inspection systems, capable of performing complex inspections with high accuracy and speed. Autonomous AI-based vision inspection systems, such as SIEMENS Inspekto VIS, play a critical role in modern manufacturing environments. These modern systems have revolutionized quality control in manufacturing, providing real-time feedback and data analysis to quickly identify and correct issues. This proactive approach to quality control helps manufacturers maintain a competitive edge by delivering superior products to the market. The ability to automate inspection tasks also frees up human resources for more complex and value-added activities, further enhancing overall productivity and operational efficiency.

1.1.2. SIEMENS Inspekto VIS

Powered by the Autonomous Machine Vision Artificial Intelligence (AMV-AI) technology, the SIEMENS Inspekto VIS system allows easy fit into dynamic and versatile production environments for a wide range of use cases, while maintaining high visual inspection quality. It is the only self-contained vision inspection product on the market and represents the most user-friendly solution for industrial quality inspection currently available. It can be set up and deployed by the final user in about 45 minutes and can be trained to inspect new products using an average of only 20 good sample of the item. No defective samples are needed for the training process.

The SIEMENS Inspekto VIS system can be installed on several location on the production line, in-line, end-of-line, offline or ad-hoc, and after a simple and short setup process, it captures images of the manufactured items using the connected camera.

The captured image of the inspected item is compared with the good references, using AI capabilities and if there are any defects, the SIEMENS Inspekto VIS system will present them within an user-friendly easy to use user interface. The inspection searches for physical abnormalities, which exceed the shape tolerances and surface variations visible in the reference items. The SIEMENS Inspekto VIS system will only present true defects, not those that can be explained by other phenomena, such as changes to lighting or movement. The AMV-AI technology is highly sensitive to details and identifies even minor differences between defects and permissible variations.

The SIEMENS Inspekto VIS system can also communicate the results of the inspection to the PLC or upper-level control systems over different well-known industrial-grade communication protocols, such as Profinet or Ethernet/IP but also as simple TCP/IP telegrams. What's more, inspection images can be transferred forward to external systems for further processing via FTP.

1.1.3. What is in this manual

This manual describes how to control the trigger of the inspection of the Inspekto system (v6.0 and v6.1) via a SIMATIC S7-1500 PLC over Profinet communication protocol.

Generally, the control of Inspekto is done by exchanging different I/O signals with the PLC. You will also explore all the signals that the PLC can send to and receive from Inspekto and that you can use to execute different actions on the system.

The Inspekto system can also be controlled by the PLC by using TIA Portal function block. This application example focuses on the main functions that needs to be used to trigger a new inspection on the Inspekto system via the PLC. Nevertheless, all the available function blocks are described in a separate section so that you get familiar with all the functions and its usage.

The trigger logic is highly dependent of each automation process and use case and it must be carefully programmed to ensure a proper synchronization between the automation process and the moment when Inspekto system triggers the image capture and performs the inspection. In this document and just with demo purpose, two inspection trigger logics are described:

- **Periodical inspection:** Inspections are triggered periodically every 10 seconds.
- **Manual inspection:** Inspections are triggered manually by the user through the HMI.

As a possible real application scenario of Inspekto, the detection of missing components on the assembly of electronic modules is presented in this example. For this application example an Inspekto profile for detecting missing components on ET200 modules was configured beforehand.

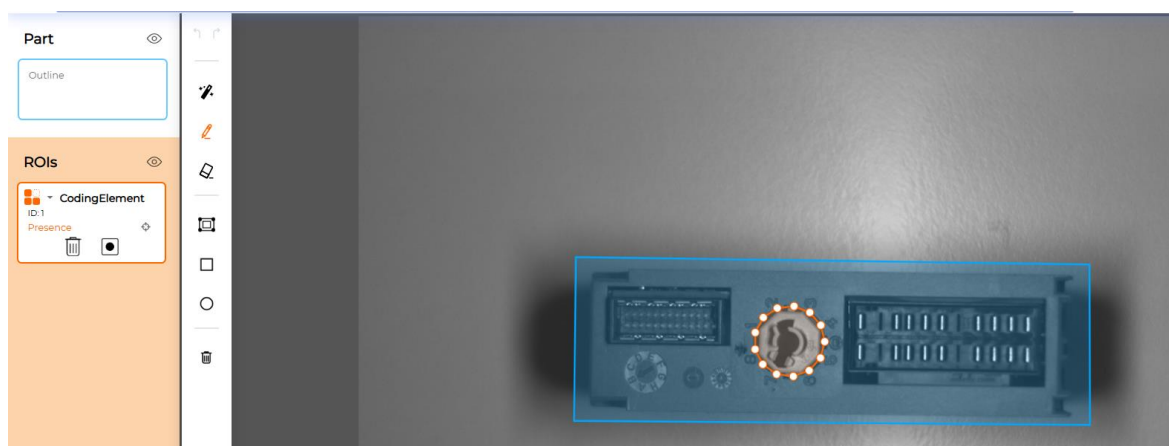


Figure 1-1

In this case, the profile is meant to detect if the mounting element of the ET200SP modules is missing. The part inspected, and the configuration of your profile does not influence the configuration and programming of the function blocks that are described in this manual



Figure 1-2



Figure 1-3

Also, an HMI project is used to change between the operation modes proposed as example and to provide to the user information about the result of the inspection. Within this example, the Inspekto camera is fixed and the electronic module to be inspected is manually placed under the camera.

Nevertheless, Inspekto allows a considerable number of mounting scenarios even integrated with robot arms and cobots.

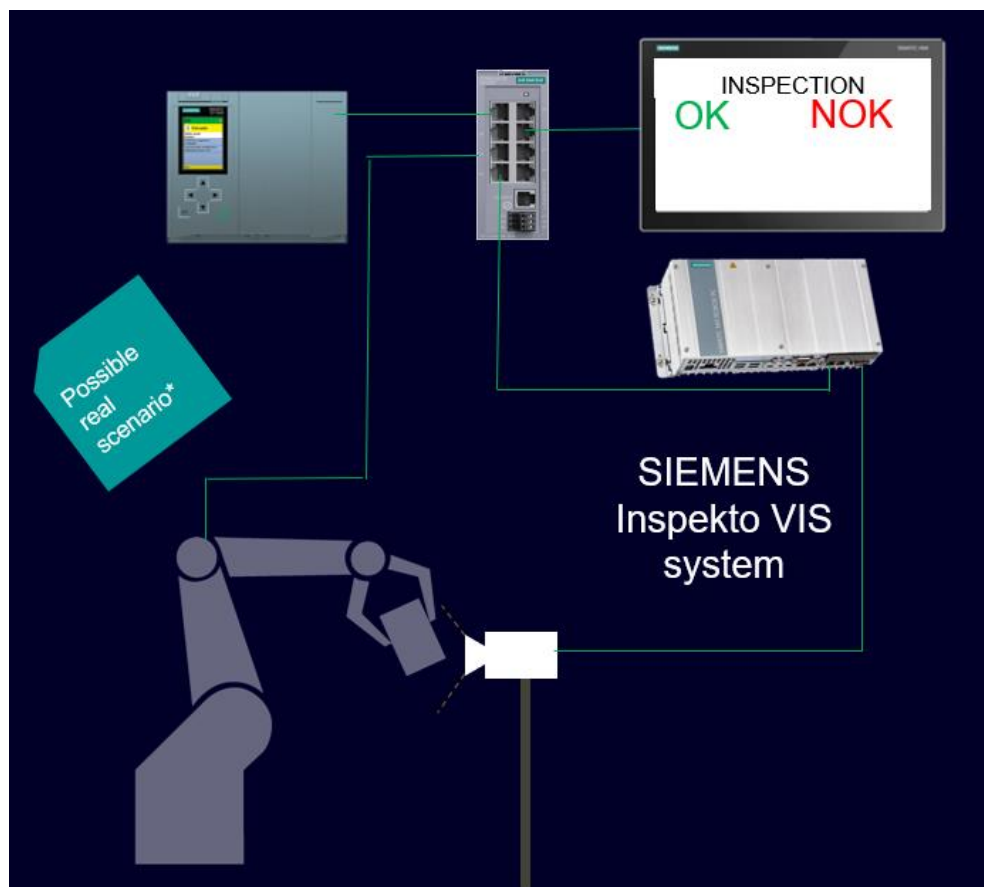


Figure 1-4

Additionally, this document also drives you through the steps that are needed to integrate Inspekto into TIA Portal hardware catalog and later use it to establish a Profinet connection to your SIMATIC PLC. The integration of Inspekto into TIA Portal is done by using the corresponding GSDML file available for download within the entry of this document.

NOTE

A different GSDML file needs to be used based on the version, v6.0 or v6.1, of your Inspekto system.

You will be guided also through the necessary configuration steps on the software of the Inspekto system to enable the communication to the PLC. A brief description about all the other function blocks that are available to control other of Inspekto system, such as profile switching, inspection mode switching or changing batch ID are also included in this manual. Using the function blocks you will have the full control of your Inspekto system, and you will be able to implement a fully automated quality inspection process and get the biggest benefit out of it.

Lastly, you will explore the integrated connectivity feedback feature, and you will learn how to use it to test and ensure that all the I/O signals are properly mapped to ensure a correct communication between your Inspekto system and the SIMATIC PLC.

NOTE

The figures of this document that include information from the Inspekto software are extracted from version v6.0. The look and feel of the user interface of the Inspekto software version v6.1 might have different design and slightly different naming.

2. Engineering

2.1. Components used

This application example has been created with the following hard- and software components:

Component	Qty	Article Number	Note
SIEMENS Inspekto VIS Basic V6	1	6ET2901-0AA00-1BA0	The software SIEMENS Inspekto VIS Professional V6 can also be used for this application example. Additionally, any variant of the version v6.1 can also be used in conjunction with the corresponding GSDML file as described in the next sections of this document.
STEP 7 Professional V19 (TIA Portal)	1	6ES7810-5CE16-0YB5	The screenshots of this application examples were taken from TIA Portal V19. However, for this application examples any higher version of STEP7 Professional (TIA Portal) can be used.
WinCC Unified ES V19 (TIA Portal)		6AV2151-0XB02-3LA5	The screenshots of this application examples were taken from TIA Portal V19. However, for this application examples any higher version of WinCC Unified ES (TIA Portal) can be used.
CPU S7-1513TF-1PN	1	6ES7513-1FM30-0AB0	For this application any S7-1500 / S7-1200 CPU can be used.
Unified Comfort Panel MTP700	1	6AV2128-3GB06-0AX1	For this application any Unified Comfort Panel can be used.
SIEMENS Inspekto VIS GSDML	1	-	The SIEMENS Inspekto GSDML represents the .XML file required to add the SIEMENS Inspekto VIS system in the hardware of TIA Portal. It can be downloaded from the Download section of this SiePortal entry. Please note that a specific GSDML file should be used according to the version of your Inspekto system, v6.0 or v6.1.

Table 2-1

2.2. Hardware Setup

2.2.1. SIEMENS Inspekto VIS System Connection

To start using the SIEMENS Inspekto VIS system, all its hardware components must be connected as shown below

Hardware Connection Diagram

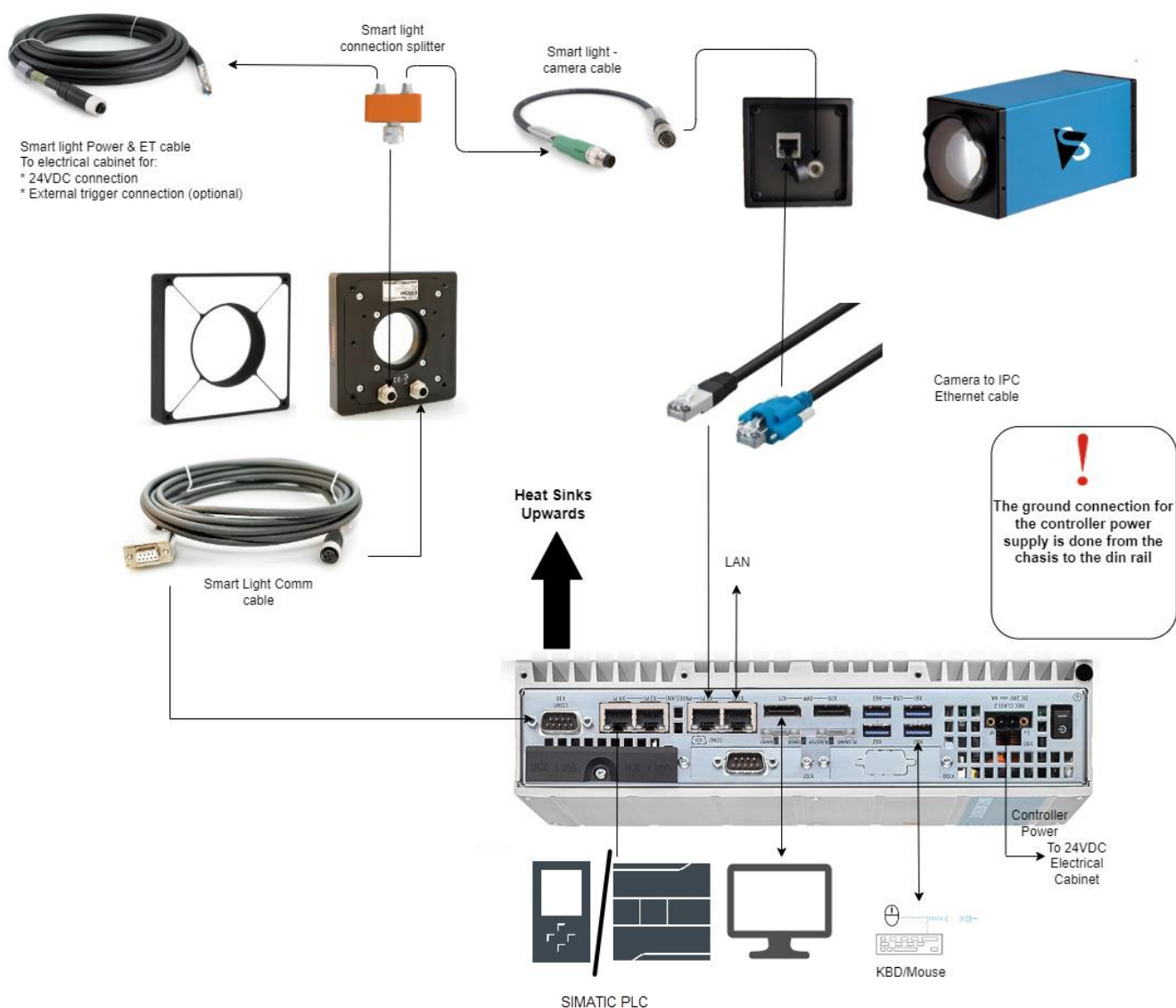


Figure 2-1

NOTE

To allow the PROFINET communication, the SIMATIC PLC must be connected to the X4 Ethernet port of the Inspekto IPC.

2.2.2. Complete setup connection

The connection between Inspekto and the other hardware components used, was done following the next diagram.

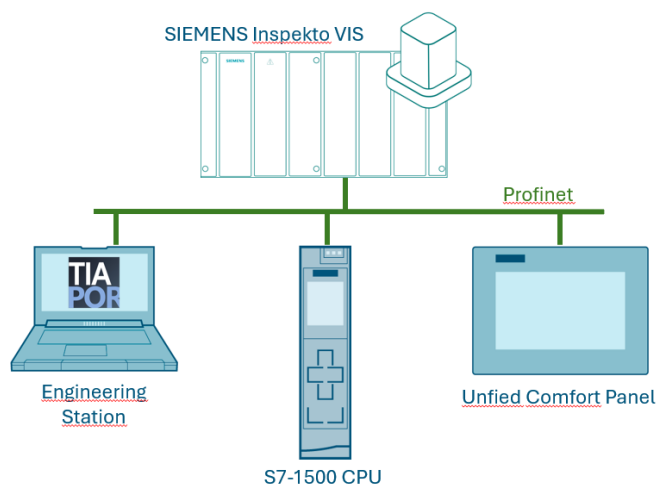


Figure 2-2

2.3. Project planning

To successfully establish a PROFINET communication between the Inspekto system and a SIMATIC PLC there are different steps that must be performed. By doing this, you can integrate Inspekto within the automation environment and control the quality inspection process through the PLC.

For this application example the following steps were followed.

NOTE

Based on your use case or application some of the steps below might not be necessary.

- Configure Inspekto settings.
- Installation of the corresponding GSDML Inspekto file within TIA Portal according to the version of your Inspekto system, v6.0 or v6.1.
- Establish connection Inspekto - PLC connection in TIA Portal.
- Testing connection using Connectivity Feedback Interface.
- Import Inspekto function blocks(FBs) from reference project to TIA Portal.
- Program the *FBNewObjectTrigger* function block to control the trigger of the inspection.
 - Periodical mode : Inspection is triggered periodically every 10 seconds.
 - Manual mode : Inspection is triggered by the user over the HMI.

Additionally if you want to show the results of the inspection also on a HM panel, the next steps needs to be done as well:

- Create HMI connection between the PLC and the HMI.
- Create HMI tags and link them with the PLC tags.
- Create HMI screen where the inspection result is shown
- Create the button on the HMI to select the inspection mode
- Collect the results from the PLC and display them in the HMI screen using HMI controls.

2.4. Configuration

2.4.1. Inspekto configuration

To be able to communicate the Inspekto system with a SIMATIC PLC, the dedicated network card interface for connectivity of the SIMATIC IPC BX-39A where the Inspekto software is running must be enabled. Since the communication is done via the Profinet protocol, select the *PROFINET* connectivity option. To do this, navigate to your Inspekto user interface, hover over the left side of the screen so the navigation menu will open and then click on "Settings".

Under the General tab, scroll down to Connectivity section and chose "Profinet" option from the drop-down menu.

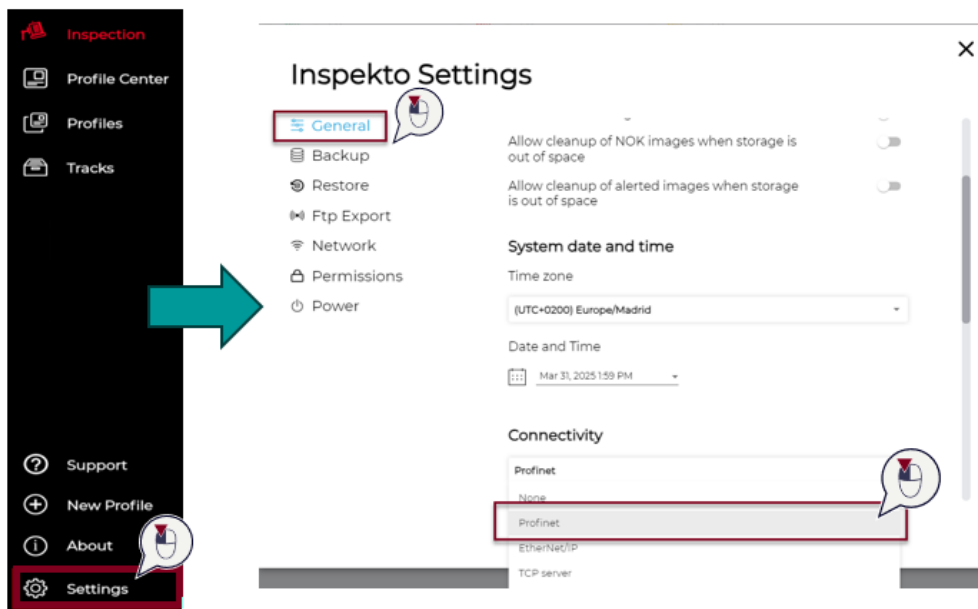


Figure 2-3

At this point, no IP needs to be defined for the Connectivity interface. Once connected to the PLC, it will assign the IP that was configured in the TIA Portal project to the Inspekto system automatically.

2.4.2. TIA Portal configuration

The configuration steps described in this section were done using TIA Portal V19. Nevertheless, the same steps apply also TIA Portal V20.

NOTE The GSDML files needed for the configuration in TIA Portal according to the version of your Inspekto system can be downloaded from the download section of this SiePortal entry.

NOTE In case you already integrated Inspekto into your TIA Portal project and you just updated your system to v6.1 check the section [3.5.Updating TIA Portal integration to Inspekto v6.1](#)

Create/Open TIA Portal project

You can create a new project or open an existing one. Open TIA Portal and select the desired option from the Portal View.

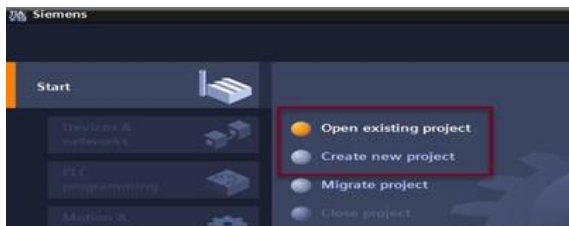


Figure 2-4

Install the GSDML file

1. In the top menu of TIA Portal go to Options tab and click on "Manage general station description files (GSD)".

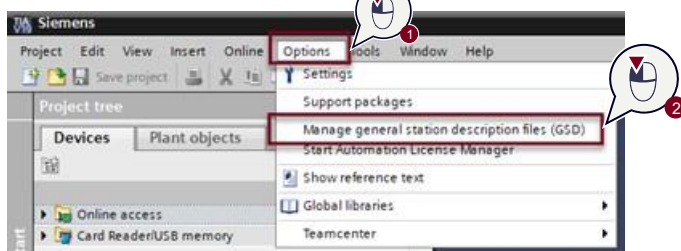


Figure 2-5

In the pop-up window, navigate in your computer to the location of the GSDML that corresponds to the version of your Inspekto system:

Inspekto version

GSDML file

Inspekto VIS Basic/Professional v6.0

GSDML-V2.33-INSPEKTO-VIS-CORE-20250326.xml

Inspekto VIS Basic/Professional v6.1

GSDML-V2.45-INSPEKTOPND-VIS-V61-20250622.xml

Table 2-2

Once done, the GSDML file will be shown in the "Content of imported path" section automatically. Mark the checkbox on the left side of the GSDML file and press "Install":

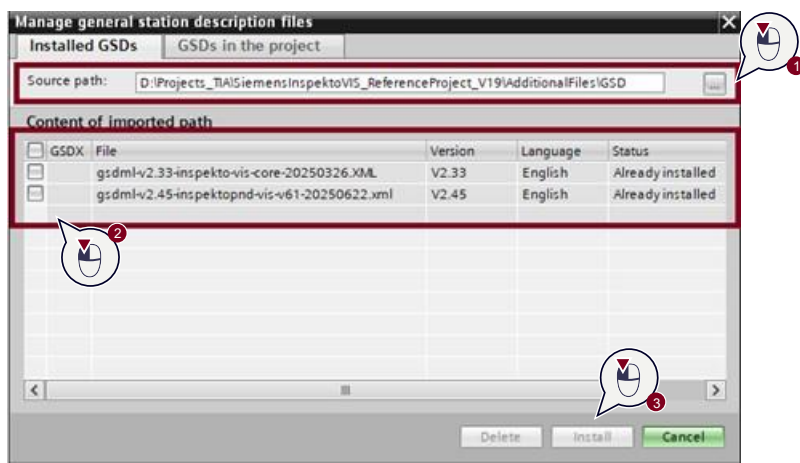


Figure 2-6

2. When the installation is completed the pop-up window will be updated with the installation result. Now the Inspekto system is available in the Hardware Catalog of TIA Portal and can be integrated into your project.

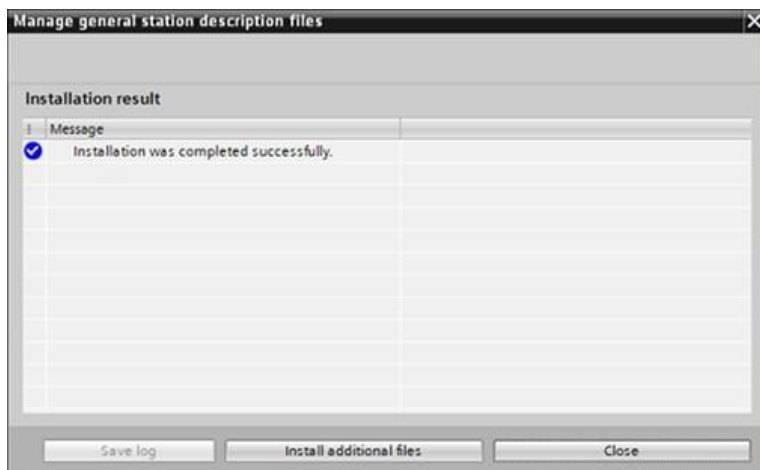


Figure 2-7

Inspekto device configuration

1. In the Project tree double click on "Devices & networks".

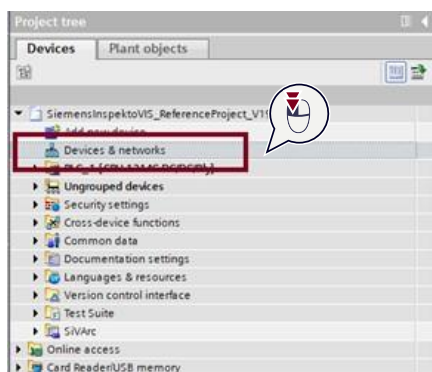


Figure 2-8

2. If you are using Inspekto v6.0, in the right side of the screen, in the "Hardware catalog" go to: "Other field devices" -> "PROFINET IO" -> "Sensors" -> "Inspekto" -> "Inspekto Cameras" and drag&drop "Siemens Inspekto VIS" device to the "Devices & network" working area.

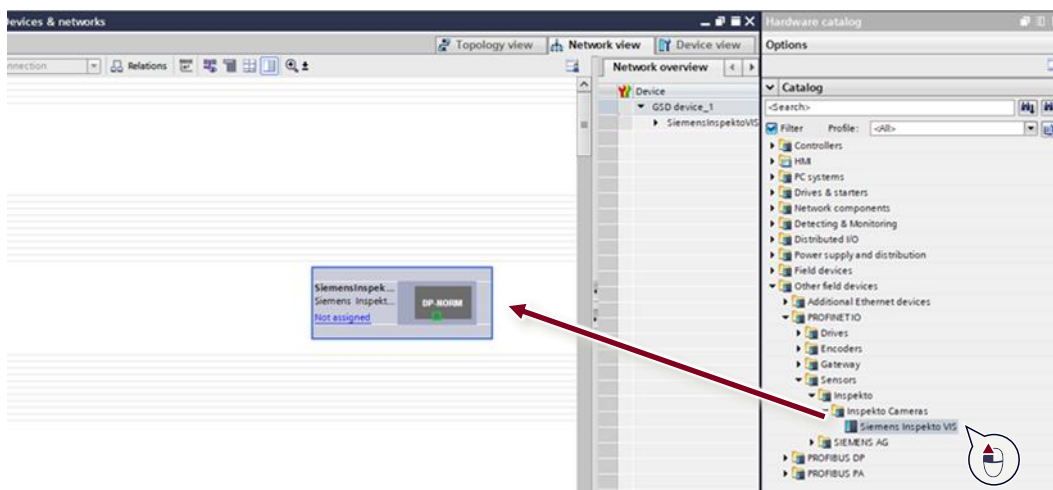


Figure 2-9

If you are using Inspekto v6.1, navigate to "Other field devices" -> "PROFINET IO" -> "Sensors" -> "SIEMENS AG" -> "INSPEKTO" -> "INSPEKTO VIS"

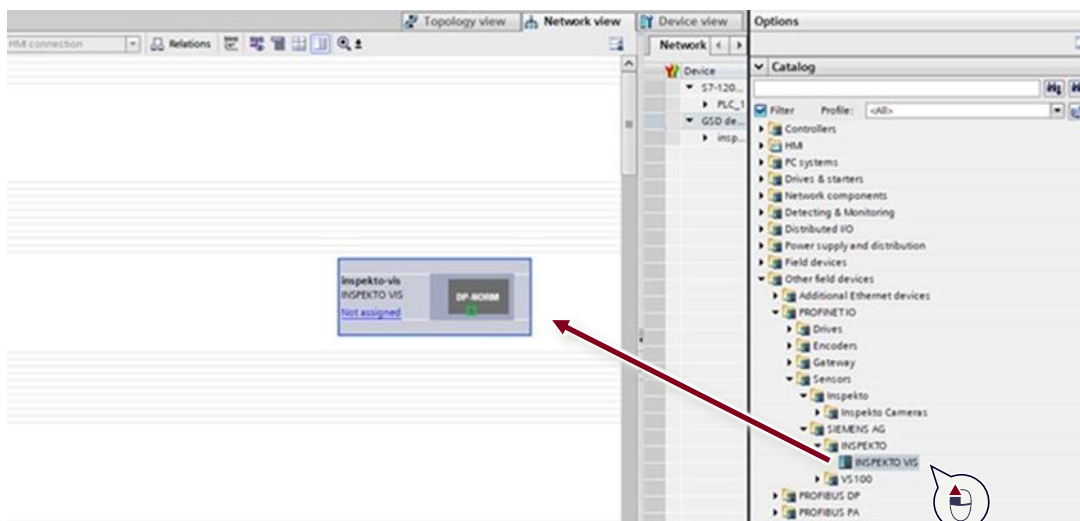


Figure 2-10

3. In the Siemens Inspekto VIS module right click on "Not assigned" and then click on "Assign to new IO controller"

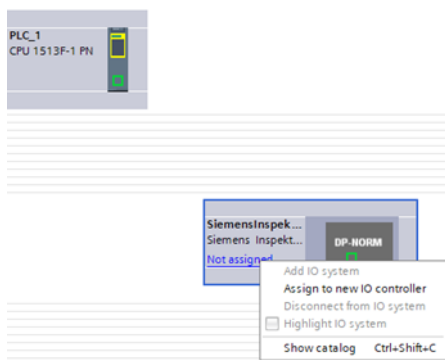


Figure 2-11

4. Select your PLC to assign and press "OK". The Profinet subnet between the PLC and the Inspekto is automatically created.

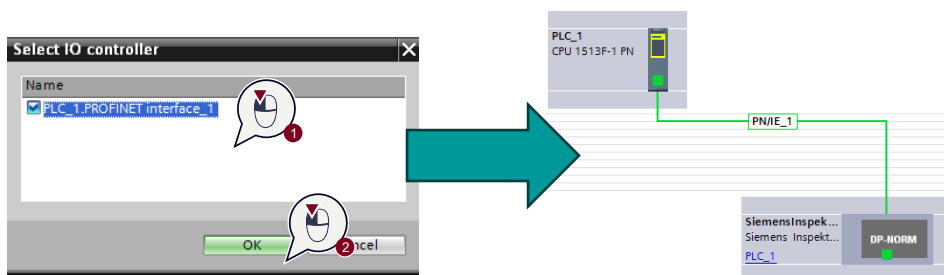


Figure 2-12

By default, TIA Portal assigns a random IP address to the devices on the default subnet 192.168.0.1. Adjust the IP address of your PLC and Inspekto system according to your network configuration.

5. Double click on the icon of Inspekto to open the "Device overview" tab and see its all input and output modules.

Inspekto_ProfinetIntegration > Ungrouped devices > SiemensInspektoVIS [Siemens Inspekto VIS]

SiemensInspektoVIS [SiemensInspektoVIS]

SiemensInspektoVIS

DP-NORM

Device overview

Module	Rack	Slot	I address	Q address	Type	Article no.	Firmware	Comment
SiemensInspektoVIS	0	0			Siemens Inspekto ...	00210	Z1.0	
Interface	0	0 X1			SiemensInspektoVIS			
OUTPUT_1	0	1			OUTPUT		1.0	
OUT_INSPECTION_MODE	0	1 1		0	OUT_INSPECTION...			
OUT_NEW_OBJECT_TRIG...	0	1 2		1...64	OUT_NEW_OBJECT...			
OUT_PROFILE_ID	0	1 3		65...66	OUT_PROFILE_ID			
OUT_BATCH_ID	0	1 4		67...130	OUT_BATCH_ID			
INPUT_1	0	2			INPUT		1.0	
IN_CUR_INSPECTION_MO...	0	2 1	0		IN_CUR_INSPECTIO...			
IN_INSPECTION_MODE_R...	0	2 2	1		IN_INSPECTION_M...			
IN_CUR_OBJECT_ID	0	2 3	2...65		IN_CUR_OBJECT_ID			
IN_NEW_OBJECT_TRIGGE...	0	2 4	66		IN_NEW_OBJECT_T...			
IN_CUR_PROFILE_ID	0	2 5	67...68		IN_CUR_PROFILE_ID			
IN_PROFILE_ID_RESPONSE	0	2 6	69		IN_PROFILE_ID_RES...			
IN_CUR_BATCH_ID	0	2 7	70...133		IN_CUR_BATCH_ID			
IN_BATCH_ID_RESPONSE	0	2 8	134		IN_BATCH_ID_RESP...			
IN_KEEP_ALIVE	0	2 9	135		IN_KEEP_ALIVE			
IN_OBJECT_INSPECTION...	0	2 10	136		IN_OBJECT_INSPEC...			
IN_SYSTEM_ERROR	0	2 11	137		IN_SYSTEM_ERROR			
IN_SYSTEM_READY	0	2 12	138		IN_SYSTEM_READY			
IN_ROIS_DEFECT_LIST	0	2 13	139...202		IN_ROIS_DEFECT_U...			

Figure 2-13

If the "Output" and "Input" modules do not appear in the list, from the "Catalog" section drag the "Output" and "Input" modules to the "Device overview" table.

NOTE

If the “Output” and “Input” modules do not appear in the list. Drag and drop the “Output” and “Input” modules from the catalog to the “Device overview” table f

Review the "I address" and "Q address" column and adjust them if needed them so have the same structure as below:

Device overview						
Module	Rack	Slot	I address	Q address	Type	
Siemens InspektoVIS	0	0			Siemens Inspekto ...	
Interface	0	0 X1			Siemens InspektoVIS	
OUTPUT_1	0	1			OUTPUT	
OUT_INSPECTION_MODE	0	1 1		0	OUT_INSPECTION_...	
OUT_NEW_OBJECT_TRIG...	0	1 2		1...64	OUT_NEW_OBJECT...	
OUT_PROFILE_ID	0	1 3		65...66	OUT_PROFILE_ID	
OUT_BATCH_ID	0	1 4		67...130	OUT_BATCH_ID	
INPUT_1	0	2			INPUT	
IN_CUR_INSPECTION_MO...	0	2 1	0		IN_CUR_INSPECTIO...	
IN_INSPECTION_MODE_R...	0	2 2	1		IN_INSPECTION_M...	
IN_CUR_OBJECT_ID	0	2 3	2...65		IN_CUR_OBJECT_ID	
IN_NEW_OBJECT_TRIGGE...	0	2 4	66		IN_NEW_OBJECT_T...	
IN_CUR_PROFILE_ID	0	2 5	67...68		IN_CUR_PROFILE_ID	
IN_PROFILE_ID_RESPONSE	0	2 6	69		IN_PROFILE_ID_RES...	
IN_CUR_BATCH_ID	0	2 7	70...133		IN_CUR_BATCH_ID	
IN_BATCH_ID_RESPONSE	0	2 8	134		IN_BATCH_ID_RESP...	
IN_KEEP_ALIVE	0	2 9	135		IN_KEEP_ALIVE	
IN_OBJECT_INSPECTION_...	0	2 10	136		IN_OBJECT_INSPEC...	
IN_SYSTEM_ERROR	0	2 11	137		IN_SYSTEM_ERROR	
IN_SYSTEM_READY	0	2 12	138		IN_SYSTEM_READY	
IN_ROIS_DEFECT_LIST	0	2 13	139...202		IN_ROIS_DEFECT_LI...	

Figure 2-14

NOTE

When using the GSDML file for Inspekto v6.1 there is an additional OUTPUT and INPUT signal, OUT_PLG_SHUTDOWN and IN_PLG_SHUTDOWN_RECEIVED respectively.

Device overview						
Module	Rack	Slot	I address	Q address	Type	
inspekto-vis	0	0			INSPEKTO VIS	
Interface	0	0 X1			inspekto-vis	
OUTPUT_1	0	1			OUTPUT	
OUT_INSPECTION_MODE	0	1 1		2	OUT_INSPECTION_...	
OUT_NEW_OBJECT_TRIGGER	0	1 2		3...66	OUT_NEW_OBJECT...	
OUT_PROFILE_ID	0	1 3		67...68	OUT_PROFILE_ID	
OUT_BATCH_ID	0	1 4		69...132	OUT_BATCH_ID	
OUT_PLG_SHUTDOWN	0	1 5		133	OUT_PLG_SHUTDO...	
INPUT_1	0	2			INPUT	
IN_CUR_INSPECTION_MODE	0	2 1	2		IN_CUR_INSPECTIO...	
IN_INSPECTION_MODE_RESPONSE	0	2 2	3		IN_INSPECTION_M...	
IN_CUR_OBJECT_ID	0	2 3	68...131		IN_CUR_OBJECT_ID	
IN_NEW_OBJECT_TRIGGER_RESPONSE	0	2 4	4		IN_NEW_OBJECT_T...	
IN_CUR_PROFILE_ID	0	2 5	5...6		IN_CUR_PROFILE_ID	
IN_PROFILE_ID_RESPONSE	0	2 6	7		IN_PROFILE_ID_RES...	
IN_CUR_BATCH_ID	0	2 7	132...195		IN_CUR_BATCH_ID	
IN_BATCH_ID_RESPONSE	0	2 8	8		IN_BATCH_ID_RESP...	
IN_KEEP_ALIVE	0	2 9	9		IN_KEEP_ALIVE	
IN_OBJECT_INSPECTION_RESULT	0	2 10	10		IN_OBJECT_INSPEC...	
IN_SYSTEM_ERROR	0	2 11	11		IN_SYSTEM_ERROR	
IN_SYSTEM_READY	0	2 12	12		IN_SYSTEM_READY	
IN_ROIS_DEFECT_LIST	0	2 13	196...259		IN_ROIS_DEFECT_LI...	
IN_PLG_SHUTDOWN_RECEIVED	0	2 14	13		IN_PLG_SHUTDO...	

Figure 2-15

When adjusting the addresses, if presented with the following pop-up, select “Do not change tags” option

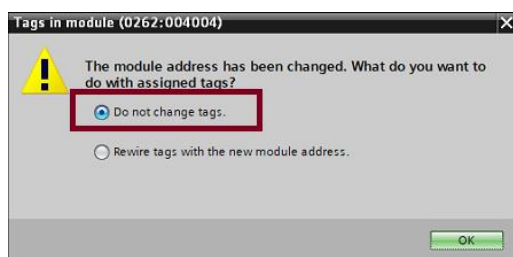


Figure 2-16

6. Assign the PROFINET device name to the Inspekto system.
 - a. Open the "Device & networks" view.
 - b. Right click on the Inspekto device and click on "Assign device name".

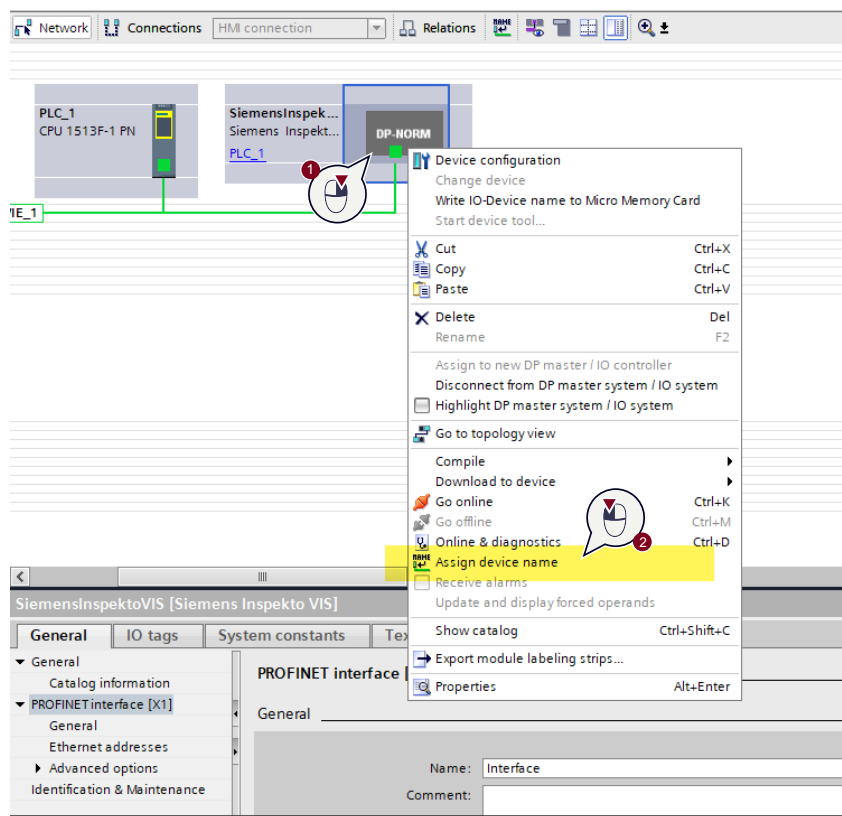


Figure 2-17

- c. Select the Inspekto device within the list of the accesibe device in the network and click on "Assign name".

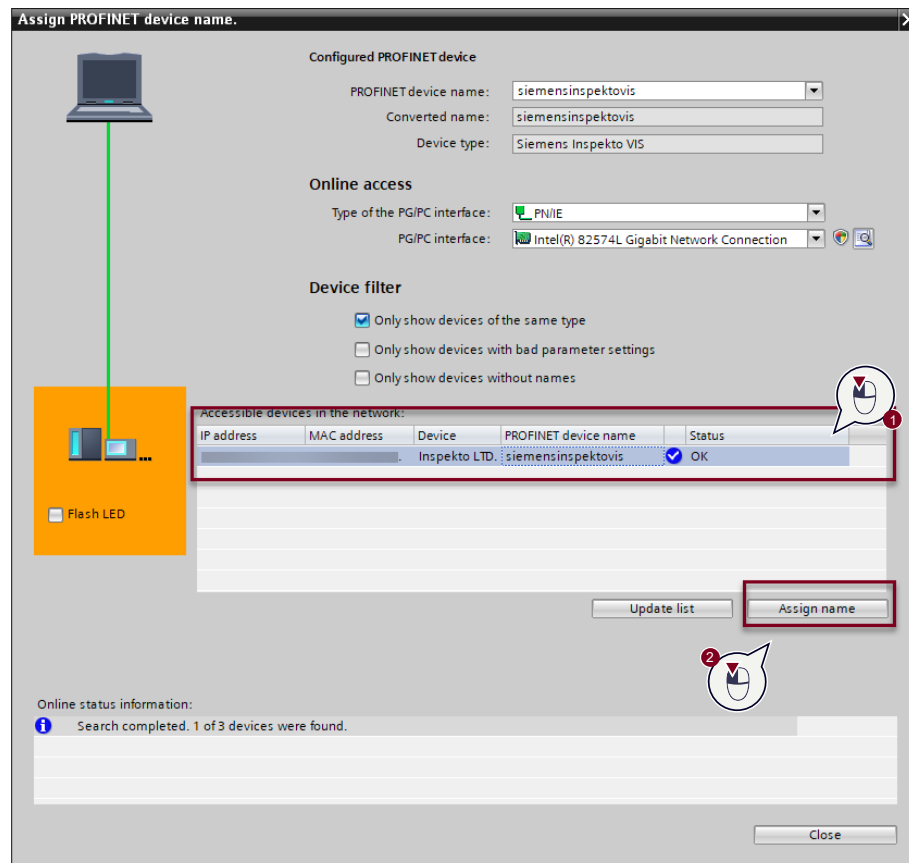


Figure 2-18

- d. If the PROFINET name is assigned successfully you should see the following information on the screen.

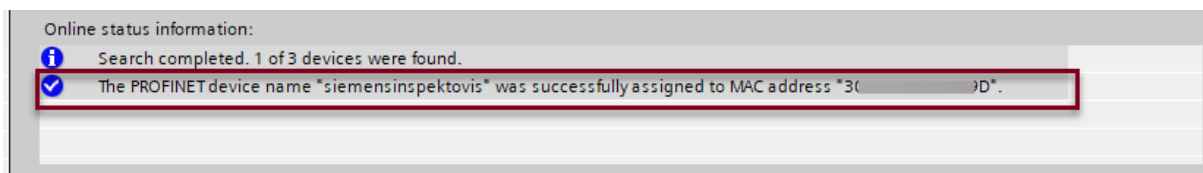


Figure 2-19

7. Compile and download the project to the PLC.

PLC data type configuration

Some of the Inputs and Outputs of Inspekto, such as OUT_NEW_OBJECT_TRIGGER or IN_CUR_OBJECT_ID are 64 bytes long. A new custom PLC data type needs to be created manually that later can be mapped to the corresponding I/Os. In this case, a custom PLC data type composed by an array of 64 chars will be created.

1. From "Project tree" go to PLC data types, double click on "Add new data type" and rename the new PLC data type to "Array64Chars" and click OK.

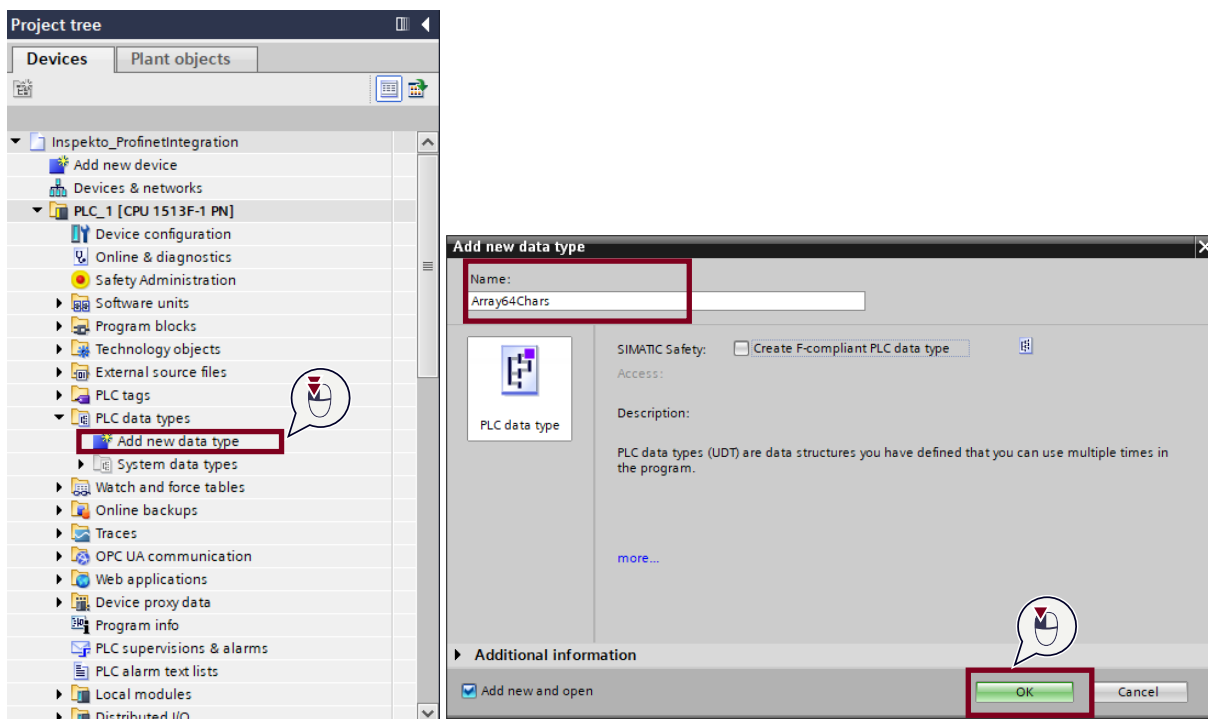


Figure 2-20

2. In the new window, in row 1 under name, type "data" and under "Data type" set it to be Array [0..63] of Char".

Array64Chars								
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint	Supervision
1	data	Array[0..63] of Char		☒	☒	☒	☐	

Figure 2-21

2.5. Operation

2.5.1. Integrating SIEMENS Inspekto VIS function blocks

This section details the steps needed to control the trigger of an inspection on the Inspekto system via the PLC using pre-made TIA Portal function blocks.

These function blocks are based on functions Inspekto supports which are described in more detail in the section "Inspekto Functions and TIA Portal Function blocks" of this manual. It is highly recommended to review and understand all sections of this manual before starting to integrate and control the Inspekto system with the given function blocks.

NOTE

- The function blocks can be used with TIA Portal V19 or higher.
- The instructions below are included in the reference project that you can find within the download section of the SiePortal entry of this application example.

Importing the function blocks to your PLC program

1. Open the project in which you already configured the connection between the Inspekto system and your PLC.
2. Download the reference project from the Download section of the SiePortal entry of this application example.
3. Under "Project tree" expand "Reference projects".

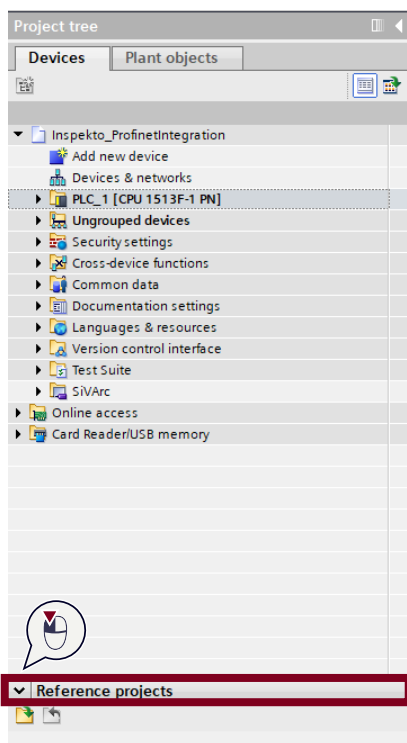


Figure 2-22

If "Reference projects" is not visible – click on "View" in the top menu and check "Reference projects".

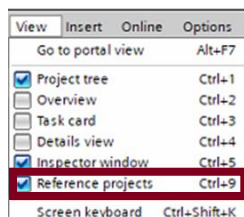


Figure 2-23

- Under "Reference projects" click on "Open reference project".

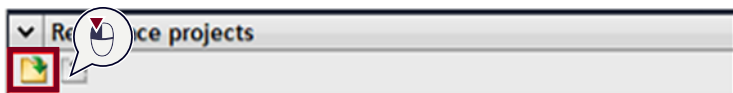


Figure 2-24

- Navigate to the location of the reference project you downloaded and click on "Open". Project name should be "SiemensInspektoVIS_ReferenceProject_V19.zap19"
- Expand the referenced project, the PLC program and the PLC data type tab.

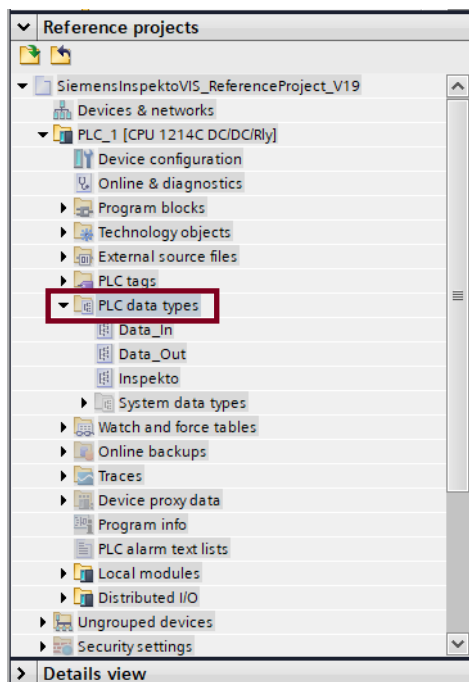


Figure 2-25

- Expand "PLC data types" and mark "Data_In", "Data_Out" and "Inspekto" by left click on the mouse + the Ctrl key in the keyboard. Then, while hovering over one of the marked items right click on the mouse and select "Copy".

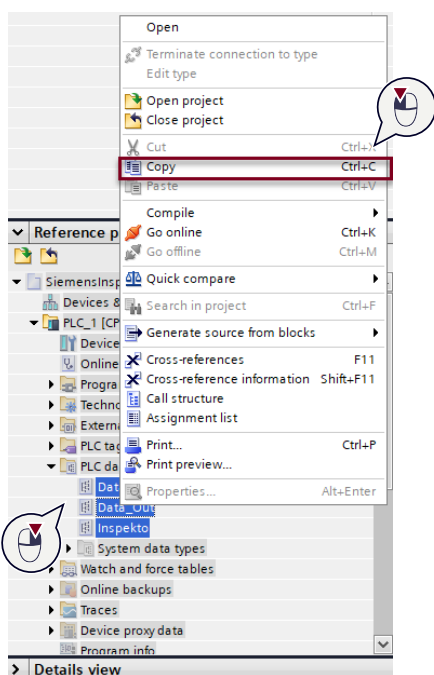


Figure 2-26

8. In "Project tree", under "Devices" expand your PLC project, right click on "PLC data types" and click on "Paste".

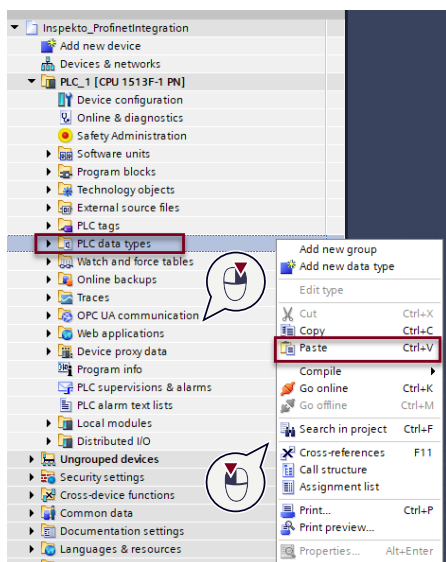


Figure 2-27

9. "PLC data types" should now have the copied data types from the reference project.

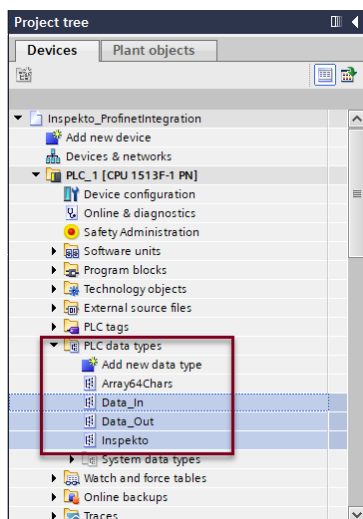


Figure 2-28

- a. "Data_In" content should be as follows:

Project tree									
Inspekto_ProfinetIntegration > PLC_1 [CPU 1513F-1 PN] > PLC data types > Data_In									
Data_In									
Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint	Supervision		
IN_CUR_INSPECTION_MODE	Byte	16#0	☒	☒	☒	☐			
IN_INSPECTION_MODE_RES...	Byte	16#0	☒	☒	☒	☐			
IN_CUR_OBJECT_ID	Array[0..63] of Char		☒	☒	☒	☐			
IN_NEW_OBJECT_TRIGGER_RE...	Byte	16#0	☒	☒	☒	☐			
IN_CUR_PROFILE_ID	Word	16#0	☒	☒	☒	☐			
IN_PROFILE_ID_RESPONSE	Byte	16#0	☒	☒	☒	☐			
IN_CUR_BATCH_ID	Array[0..63] of Char		☒	☒	☒	☐			
IN_BATCH_ID_RESPONSE	Byte	16#0	☒	☒	☒	☐			
IN_KEEP_ALIVE	Byte	16#0	☒	☒	☒	☐			
IN_OBJECT_INSPECTION_RES...	Byte	16#0	☒	☒	☒	☐			
IN_SYSTEM_ERROR	Byte	16#0	☒	☒	☒	☐			
IN_SYSTEM_READY	Byte	16#0	☒	☒	☒	☐			
IN_ROIS_DEFECT_LIST	Array[0..63] of Char		☒	☒	☒	☐			

Figure 2-29

NOTE

In case you are configuring a v6.1 Inspekto system, the IN_PLC_SHUTDOWN_RECEIVED signal is not included by default in the "Data_In" PLC data type available within the reference project provided with this application example. If you need to use this signal you need to add it manually as "Byte" tag.

b. "Data_Out" content should be as follows:

Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint	Supervision
OUT_INSPECTION_MODE	Byte	16#0	☒	☒	☒	☐	☐
OUT_NEW_OBJECT_TRIGGER	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐
OUT_PROFILE_ID	Word	16#0	☒	☒	☒	☐	☐
OUT_BATCH_ID	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐

Figure 2-30

NOTE

In case you are configuring a v6.1 Inspekto system, the OUT_PLC_SHUTDOWN signal is not included by default in the "Data_Out" PLC data type available within the reference project provided with this application example. If you need to use this signal you need to add it manually as "Byte" tag.

c. "Inspekto" PLC data type content should be as follows:

Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint	Supervision
Camera_In	"Data_In"		☒	☒	☒	☒	☒
IN_CUR_INSPECTION_MODE	Byte	16#0	☒	☒	☒	☐	☐
IN_INSPECTION_MODE_RE...	Byte	16#0	☒	☒	☒	☐	☐
IN_CUR_OBJECT_ID	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐
IN_NEW_OBJECT_TRIGGER...	Byte	16#0	☒	☒	☒	☐	☐
IN_CUR_PROFILE_ID	Word	16#0	☒	☒	☒	☐	☐
IN_PROFILE_ID_RESPONSE	Byte	16#0	☒	☒	☒	☐	☐
IN_CUR_BATCH_ID	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐
IN_BATCH_ID_RESPONSE	Byte	16#0	☒	☒	☒	☐	☐
IN_KEEP_ALIVE	Byte	16#0	☒	☒	☒	☐	☐
IN_OBJECT_INSPECTION_R...	Byte	16#0	☒	☒	☒	☐	☐
IN_SYSTEM_ERROR	Byte	16#0	☒	☒	☒	☐	☐
IN_SYSTEM_READY	Byte	16#0	☒	☒	☒	☐	☐
IN_ROIS_DEFECT_LIST	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐
Camera_Out	"Data_Out"		☒	☒	☒	☒	☒
OUT_INSPECTION_MODE	Byte	16#0	☒	☒	☒	☐	☐
OUT_NEW_OBJECT_TRIGG...	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐
OUT_PROFILE_ID	Word	16#0	☒	☒	☒	☐	☐
OUT_BATCH_ID	Array[0..63] of Char	16#0	☒	☒	☒	☐	☐

Figure 2-31

10. Under "Reference project" expand "Program blocks" and mark "Main [OB1]", "HMI [DB2]", and "Inspekto" by left click on the mouse + the Ctrl key in the keyboard. Then, while hovering over one of the marked items right click on the mouse and select "Copy".

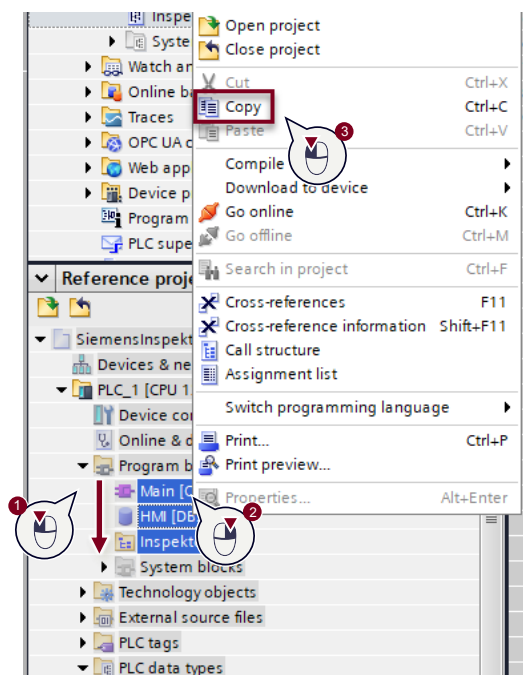


Figure 2-32

11. In "Project tree" under "Devices", right click on "Program blocks" and then click on "Paste".

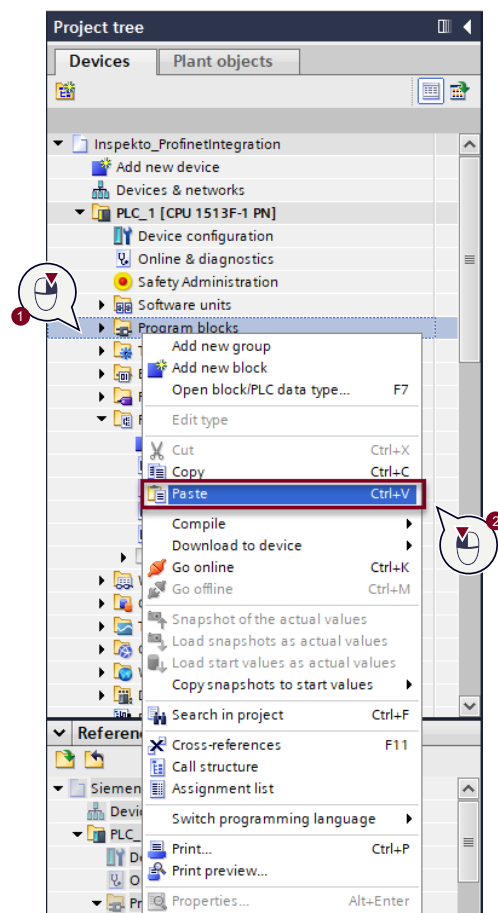


Figure 2-33

In some cases a warning popup might appear.

- a. If this is a new PLC SW select "Replace existing objects and move to this location"
- b. If this is not a new PLC software select the option that fits best according to your needs.



Figure 2-34

12. "Program blocks" should now have the copied program blocks which includes the function blocks from the reference project that will be used to control the Inspekto system.

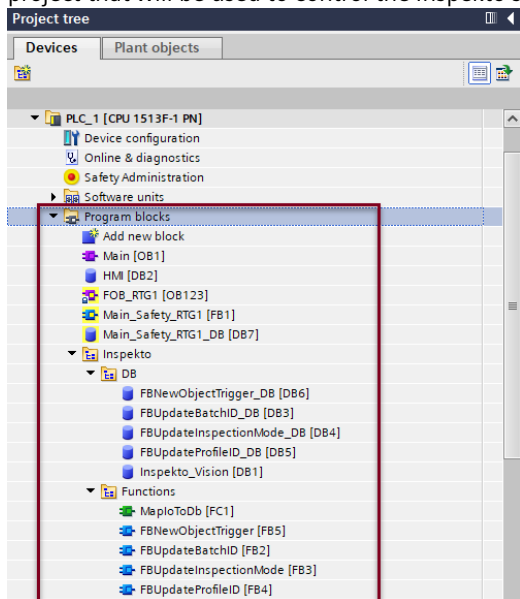


Figure 2-35

13. Compile and Save your project to apply all the operations performed previously.
14. Download the project to the PLC to be able to test the configuration using the Connectivity Feedback Interface later.

2.5.2. Testing configuration and PLC-Inspekto connectivity.

The SIEMENS Inspekto VIS system includes a feature called Connectivity Feedback. This feature is aimed to test and ensure correct communication with all type of connectivity options available in the system .After integrating configuring the SIEMENS Inspekto VIS system within TIA Portal to establish the connection with the SIMATIC PLC using the Profinet, it is highly recommended before start programming the function block for controlling Inspekto to test the configuration this interface to verify that:

- Signals are going back and forth between the SIEMENS Inspekto VIS system and the PLC and,
- Signals are received in the right place and form. E.g, when sending an OUT_PROFILE_ID signal from the PLC with value of "2" we want to ensure that it is received with value of "2" in its corresponding field.

To access the Connectivity Feedback, access your Inspekto system, hover over the left side of the screen, and then click on "Settings. Under general tab scroll down to the connectivity section and click on the "Connectivity Feedback Interface":

Inspekto Settings

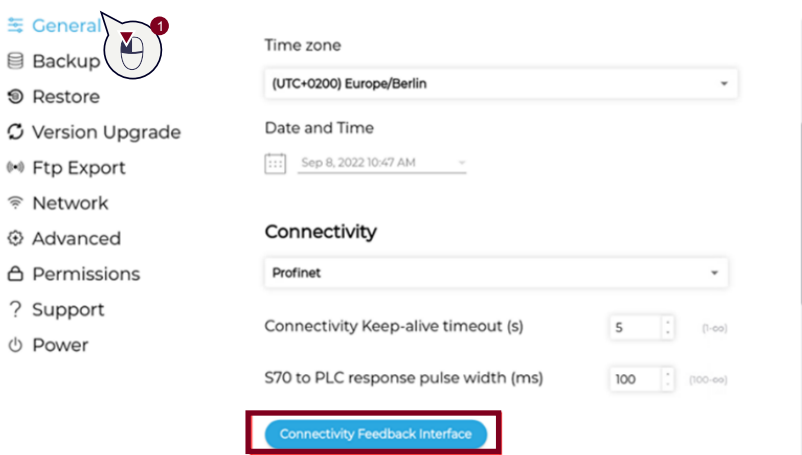


Figure 2-36

Once selected, the system will show a pop-up message stating that the system will now only show signals coming from the PLC but will not act on them. This means that when working in "Connectivity Feedback Interface" all signals coming from the PLC will be received and shown in the interface, but no action will take place. When closing the Connectivity Feedback Interface all PLC communication will return to normal state automatically.

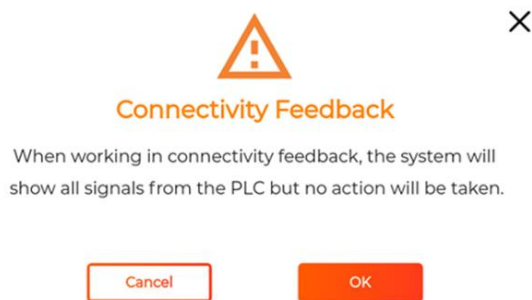


Figure 2-37

The Connectivity Feedback screen has 2 tabs, one for messages going from the SIEMENS Inspekto VIS system to the PLC, and the other from messages going from the PLC to the SIEMENS Inspekto VIS system.

The content of the Connectivity Feedback screen will change according to the selected connectivity method. This manual relates to the PROFINET Connectivity Feedback screen only. For other connectivity methods please see the relevant integration manual.

2.5.2.1. Inspekto → PLC feedback screen

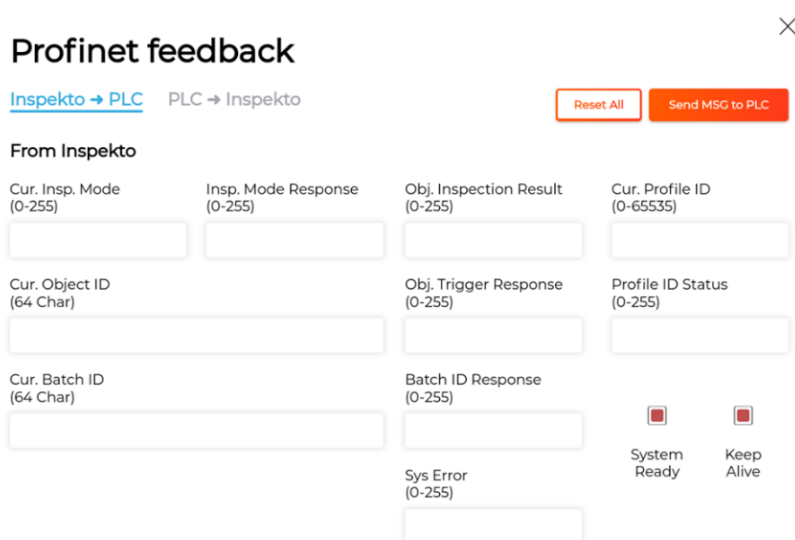


Figure 2-38

This screen contains all the possible signals going from the SIEMENS Inspekto VIS system to the PLC knows also as input signals and are in the global data block under the Camera_In PLC data type:

Inspekto_Vision							
	Name	Data type	Offset	Start value	Retain	Accessible f...	Writa...
1	Static				☐	☐	☐
2	Camera 1	"Inspekto"	0.0		☐	☒	☒
3	Camera_In	"Data_In"	0.0		☐	☒	☒
4	IN_CUR_INSPECTION_MODE	Byte	0.0	16#0	☐	☒	☒
5	IN_INSPECTION_MODE_RESPONSE	Byte	1.0	16#0	☐	☒	☒
6	IN_CUR_OBJECT_ID	Array[0..63] of Char	2.0		☐	☒	☒
7	IN_NEW_OBJECT_TRIGGER_RESPONSE	Byte	66.0	16#0	☐	☒	☒
8	IN_CUR_PROFILE_ID	Word	68.0	16#0	☐	☒	☒
9	IN_PROFILE_ID_RESPONSE	Byte	70.0	16#0	☐	☒	☒
10	IN_CUR_BATCH_ID	Array[0..63] of Char	72.0		☐	☒	☒
11	IN_BATCH_ID_RESPONSE	Byte	136.0	16#0	☐	☒	☒
12	IN_KEEP_ALIVE	Byte	137.0	16#0	☐	☒	☒
13	IN_OBJECT_INSPECTION_RESULT	Byte	138.0	16#0	☐	☒	☒
14	IN_SYSTEM_ERROR	Byte	139.0	16#0	☐	☒	☒
15	IN_SYSTEM_READY	Byte	140.0	16#0	☐	☒	☒
16	IN_ROIS_DEFECT_LIST	Array[0..63] of Char	142.0		☐	☒	☒

Figure 2-39

To send a command to the PLC and test the configuration and the communication:

1. Type a value for a signal you wish to send in the corresponding field and press the "Send MSG to PLC" button.

Profinet feedback

[Inspekto → PLC](#) [PLC → Inspekto](#)

From Inspekto

Cur. Insp. Mode (0-255) 1	Insp. Mode Response (0-255) 	Obj. Inspection Result (0-255) 	Cur. Profile ID (0-65535)
Cur. Object ID (64 Char) 	Obj. Trigger Response (0-255) 	Profile ID Status (0-255) 	
Cur. Batch ID (64 Char) 	Batch ID Response (0-255) 	☐ System Ready ☐ Keep Alive	
	Sys Error (0-255) 		

Figure 2-40

2. Check the signal value is received at the corresponding tag within the PLC data block. If the connection is established successfully, you will be able to see the content of the message in the corresponding tag in the PLC.

Inspekto_Vision						
	Name	Data type	Offset	Start value	Monitor value	Retain
1	Static					
2	Camera 1	"Inspekto"	0.0			
3	Camera_In	"Data_In"	0.0			
4	IN_CUR_INSPECTION_MODE	Byte	0.0	16#0	16#01	
5	IN_INSPECTION_MODE_RESPONSE	Byte	1.0	16#0	16#00	
6	IN_CUR_OBJECT_ID	Array[0..63] of Char	2.0			
7	IN_NEW_OBJECT_TRIGGER_RESPONSE	Byte	66.0	16#0	16#00	
8	IN_CUR_PROFILE_ID	Word	68.0	16#0	16#0008	
9	IN_PROFILE_ID_RESPONSE	Byte	70.0	16#0	16#00	
10	IN_CUR_BATCH_ID	Array[0..63] of Char	72.0			
11	IN_BATCH_ID_RESPONSE	Byte	136.0	16#0	16#00	
12	IN_KEEP_ALIVE	Byte	137.0	16#0	16#01	
13	IN_OBJECT_INSPECTION_RESULT	Byte	138.0	16#0	16#00	
14	IN_SYSTEM_ERROR	Byte	139.0	16#0	16#00	
15	IN_SYSTEM_READY	Byte	140.0	16#0	16#01	
16	IN_ROIS_DEFECT_LIST	Array[0..63] of Char	142.0			
17	Camera_Out	"Data_Out"	206.0			
18	OUT_INSPECTION_MODE	Byte	206.0	16#0	16#00	
19	OUT_NEW_OBJECT_TRIGGER	Array[0..63] of Char	208.0			
20	OUT_PROFILE_ID	Word	272.0	16#0	16#0000	
21	OUT_BATCH_ID	Array[0..63] of Char	274.0			

Figure 2-41

3. Repeat the same with all remaining OUT signals.

NOTE

System Ready, and Keep Alive messages are checkboxes. To check/uncheck them, click in the rectangle, and to pass the signal's value to the PLC press on the "Send MSG to PLC" button

2.5.2.2. PLC → Inspekto feedback screen

Profinet feedback

Inspekto → PLC [PLC → Inspekto](#)

Reset All

Send MSG to PLC

Out Insp. mode
(0-255)

Out profile ID
(0-65535)

Out batch ID
(64 Char)

Out object trigger
(64 Char)

Figure 2-42

This screen contains all the possible signals coming from the PLC to the Inspekto system. These signals are the output signals, OUT_INSPECTION_MODE, OUT_NEW_OBJECT_TRIGGER, OUT_PROFILE_ID and OUT_BATCH_ID which are in the global data block under the Camera_Out PLC data type:

Inspekto_Vision								
	Name	Data type	Offset	Start value	Retain	Accessible f...	Writa...	...
1	Static				☐	☐	☐	
2	Camera1	"Inspekto"	0.0		☐	☒	☒	
3	Camera_In	"Data_In"	0.0		☐	☒	☒	
4	Camera_Out	"Data_Out"	206.0		☐	☒	☒	
5	OUT_INSPECTION_MODE	Byte	206.0	16#0	☐	☒	☒	
6	OUT_NEW_OBJECT_TRIGGER	Array[0..63] of Char	208.0		☐	☒	☒	
7	OUT_PROFILE_ID	Word	272.0	16#0	☐	☒	☒	
8	OUT_BATCH_ID	Array[0..63] of Char	274.0		☐	☒	☒	

Figure 2-43

When the connection is established successfully, the fields will be updated automatically every time a message is being sent from the PLC to the SIEMENS Inspekto VIS system. To test the configuration and the connection are properly setup:

1. Go online with your PLC, open the DB where the output signals are located, activate the "Monitor all" feature and modify the value of every signal.

The screenshot shows the 'Inspekto_Vision' configuration screen with a table of signals. The signal 'OUT_INSPECTION_MODE' is highlighted with a red box. A 'Modify' dialog box is open, showing the operand 'Inspekto_Vision'.Camera1.Camera', data type 'Byte', and a 'Modify value' of '3' in hex format.

Name	Data type	Offset	Start value	Monitor value	Retain	Ac...
Static					☐	
Camera1	"Inspekto"	0.0			☐	
Camera_In	"Data_In"	0.0			☐	
Camera_Out	"Data_Out"	206.0			☐	
OUT_INSPECTION_MODE	Byte	206.0	16#0	16#00	☐	
OUT_NEW_OBJECT_TRIGGER	Array[0..63] of Char	208.0			☐	
OUT_PROFILE_ID	Word	272.0	16#0	16#0000	☐	
OUT_BATCH_ID	Array[0..63] of Char	274.0			☐	

Modify

Operand: "Inspekto_Vision".Camera1.Camera Data type: Byte

Modify value: 3 Format: Hex

OK Cancel

Figure 2-44

2. Check within the Connectivity Feedback screen that the value of the signal that you just modified changed accordingly.

Profinet feedback

Inspekto → PLC [PLC → Inspekto](#)

Reset All Send MSG to PLC

Out Insp. mode (0-255)
3

Out profile ID (0-65535)

Out batch ID (64 Char)

Out object trigger (64 Char)

Figure 2-45

3. Repeat the same with all remaining OUT signals.

2.5.3. Trigger control using TIA Portal function blocks

Once you have integrated all the functions block provided within the reference project into your PLC program, you are ready to call them into the main program and start controlling the Inspekto system via the PLC based on your automation logic. Using the pre-defined function blocks you can perform and control different actions on the Inspekto system.

NOTE

- Before sending any command, the PLC needs to verify that "IN_SYSTEM_READY" =1.
- After sending any command, "IN_SYSTEM_READY" will change to "0" while the command is being performed and change back to "1" when the operation was completed.
- "IN_SYSTEM_ERROR" refers to the status of the last command received, and it is not a system state. If "IN_SYSTEM_ERROR" =1, when "IN_SYSTEM_READY" =1 the system is ready to receive the next command.
- Although using the function blocks, all of the variables detailed in the GSDML are also available for use. It is recommended that the PLC will verify any of the available relevant variables when working with Inspekto system.

This manual is focused on showing how to control the trigger of the inspection from the PLC. For this purpose, in this application example, two operation modes are presented as example:

- Manual: Inspection are triggered manually by the user.
- Periodical : Inspections are triggered periodically every 5 seconds.

NOTE

The operation modes previously described are used just as an example to showcase the trigger of the inspection via the PLC. You must adjust the trigger logic according to your automation process. The operation modes mentioned above does not represent any of the possible trigger inspection modes described in the section "Inspekto Trigger Inspection Modes"

To showcase one of the use cases that Inspekto can be used for, the detection of missing components on the assembly of electronic modules is presented in this example. Additionally, a HMI is also used to provide the user information about the result of the inspection and inform about the total number of good and bad parts inspected. Within this example, the Inspekto camera is fixed and the electronic module to be inspected is manually placed under the camera. Nevertheless, Inspekto allows a considerable number of mounting scenarios even integrated with robot arms and cobots as already described in the first section of this document.

2.5.3.1. PLC programming – Trigger inspection

By default, the main program, OB1, of the PLC from the reference project that was copied during the configuration, includes the call of all the pre-define function blocks for controlling Inspekto. All these function blocks have their inputs and outs already parametrized. The parametrization of the FBs is done using the tags from the "HMI" [DB2] data block.

NOTE

These tags are provided just as reference and all the FBs that you want to use to control your Inspekto system must be adapted according to the user's needs and to the automatic logic.

I/Os mapping

All the provided FBs in the reference project operates over tags from the "Inspekto_VisionDB", therefore, to be able to send the values of the tags from the DB to the Inspekto system and vice versa, the DB's tags must be linked to the inputs and outputs of Inspekto.

1. Open the main PLC program, "Main [OB1]".
2. Use "MaploToDb [FC1]" function to connect between physical I/O addresses and Inspekto_Vision DB.

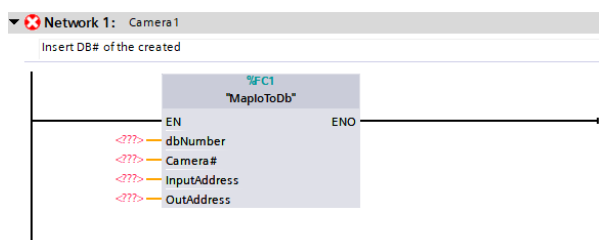


Figure 2-46

3. Parametrize the inputs of the function block as follows:
 - dbNumber: the number of the DB "Inspekto_Vision" as stated in TIA Portal in "DB[x]"
 - Camera# : the number of the camera. For the first camera, must always start at 0.
 - InputAddress: the starting address of the input signals of the corresponding camera. Can be check in the "Device overview" tab of the Inspekto device in the Devices&Networks view of TIA Portal.
 - OutAddress: the starting address of the output signals of the corresponding camera. Can be check in the "Device overview" tab of the Inspekto device in the Devices&Networks view of TIA Portal.

For example, in this case the starting address of the input signals and the output signals are both 0:

Module	Rack	Slot	I address	Q address	Type	Article no.
SiemensInspektoVIS	0	0			Siemens Inspekto ...	00210
Interface	0	0 x1			SiemensInspektoVIS	
OUTPUT_1	0	1		0	OUTPUT	
OUT_INSPECTION_MODE	0	1.1		1..64	OUT_INSPECTION...	
OUT_NEW_OBJECT_TRIG...	0	1.2		65..66	OUT_NEW_OBJECT...	
OUT_PROFILE_ID	0	1.3		67..130	OUT_PROFILE_ID	
OUT_BATCH_ID	0	1.4			OUT_BATCH_ID	
INPUT_1	0	2	0		INPUT	
IN_CUR_INSPECTION_M...	0	2.1	0		IN_CUR_INSPECTIO...	
IN_INSPECTION_MODE...	0	2.2	1		IN_INSPECTION_M...	
IN_CUR_OBJECT_ID	0	2.3	2..65		IN_CUR_OBJECT_ID	
IN_NEW_OBJECT_TRIGGE...	0	2.4	66		IN_NEW_OBJECT_T...	
IN_CUR_PROFILE_ID	0	2.5	67..68		IN_CUR_PROFILE_ID	
IN_PROFILE_ID_RESPON...	0	2.6	69		IN_PROFILE_ID_RES...	
IN_CUR_BATCH_ID	0	2.7	70..133		IN_CUR_BATCH_ID	
IN_BATCH_ID_RESPON...	0	2.8	134		IN_BATCH_ID_RESP...	
IN_KEEP_ALIVE	0	2.9	135		IN_KEEP_ALIVE	
IN_OBJECT_INSPECTION...	0	2.10	136		IN_OBJECT_INSPEC...	
IN_SYSTEM_ERROR	0	2.11	137		IN_SYSTEM_ERROR	
IN_SYSTEM_READY	0	2.12	138		IN_SYSTEM_READY	
IN_NOIS_DEFECTION_LIST	0	2.13	139..202		IN_NOIS_DEFECTION...	

Figure 2-47

Therefore, the function "MaploToDb" results as follows:

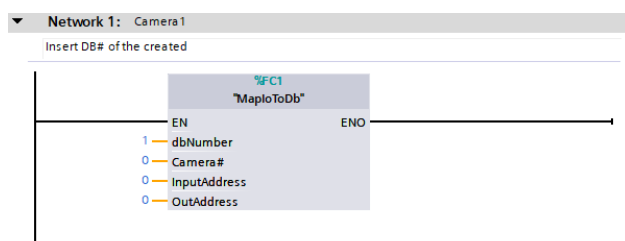


Figure 2-48

NOTE

In case you want to control several Inspekto systems using the same PLC, repeat the same steps for every additional camera adjusting the inputs of the function block accordingly

Load inspection profile

To perform inspection, a profile with the item you want to inspect must be loaded beforehand within the Inspekto system. If there are no profiles created on your Inspekto system you must create a new one. Inspekto profiles, can be loaded via the Inspekto UI but also via the PLC using the corresponding pre-define function block as described in the section "Inspekto functions and TIA Portal function blocks".

In this case, the profile is loaded using the Inspekto UI option:

1. Access the UI of your Inspekto system, screen hover over the left side of the screen so the navigation menu will open, and then click on Profiles menu
2. Select the profile of your interest and click on the "Load" button.

Profiles



Figure 2-49

Setup Inspekto inspection trigger mode

To be able to control the trigger of the inspection via the PLC, the inspection trigger mode of the Inspekto system must be configured accordingly. Inspekto allows 9 different trigger modes as described in more detail in the section "Inspekto Trigger Inspection Modes". To control the inspection trigger via the PLC, the inspection mode must be configured to one of the available "Direct Trigger" based options.

Profile settings

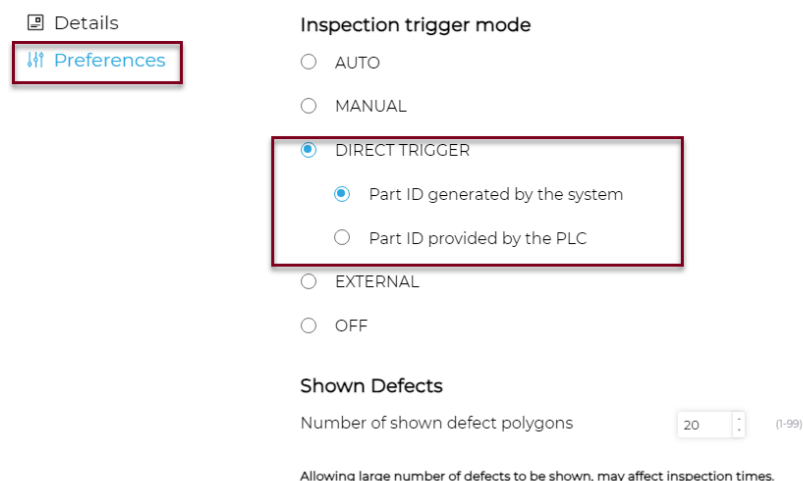


Figure 2-50

The inspection trigger mode can be configured either directly via the Inspekto UI or via the PLC using the dedicated function block in TIA Portal. Each trigger mode has assigned a specific integer value as described in table below:

Inspection Trigger Mode	Value
OFF	0
Auto - Part ID provided by the PLC	1
Auto - Part ID generated by the system	2
Manual	3
External - Part ID generated by the system	4
External - Part ID provided by the PLC	5
Direct Trigger - Part ID generated by the system	6
Direct Trigger - Part ID provided by the PLC	7
Auto - Part ID generated by the external code reader	8

Table 2-3

NOTE The value of each inspection mode does not correspond with its order in the Inspekto UI.

To be able to trigger an inspection via the PLC, the user must select the right inspection mode also via the PLC and using the corresponding function block, FBUpdateInspectionMode, which is described in detail in the “Inspekto functions and TIA Portal function blocks”.

To set the inspection mode:

1. Open the main program, Main[OB1]
2. Navigate to the call of the “FBUpdateInspectionMode” function block.
3. Assign the value “6” to the NewInspectionMode input parameter.

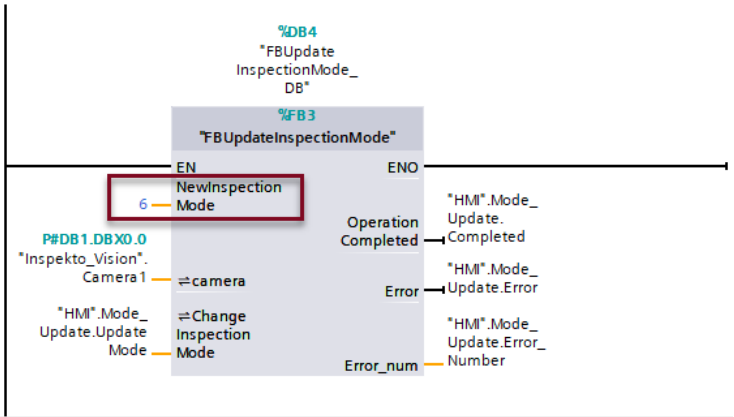


Figure 2-51

4. Compile and download the PLC project.
5. Go online with the PLC.

6. Toggle the value of the input parameter "ChangeInspectionMode" to trigger the change of the inspection mode.

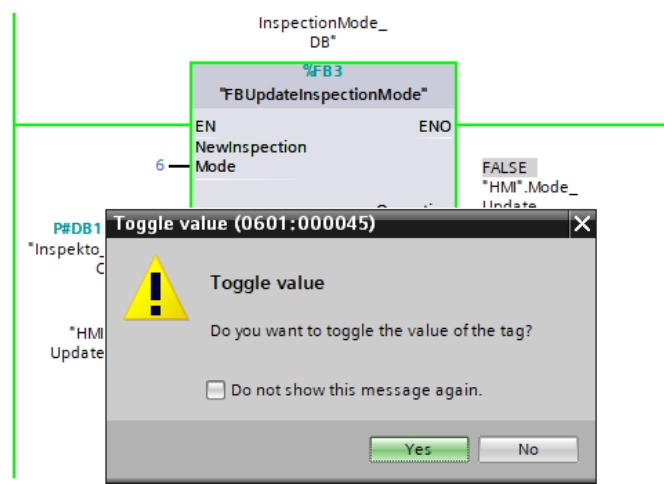


Figure 2-52

7. Check that the output parameter "OperationCompleted" changed to TRUE.

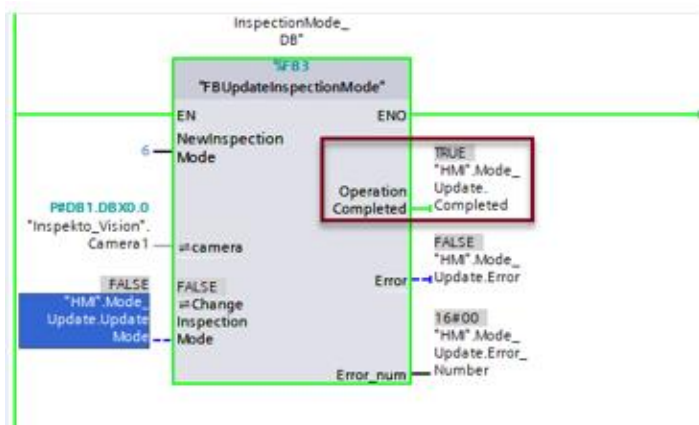


Figure 2-53

8. Check in Inspekto UI that the inspection mode of the profile is now assigned to "EXTERNAL TRIGGER – Part ID provided by the system".

Trigger inspection

The trigger of the inspection is handled using the "FBNewObjectTrigger" function block.

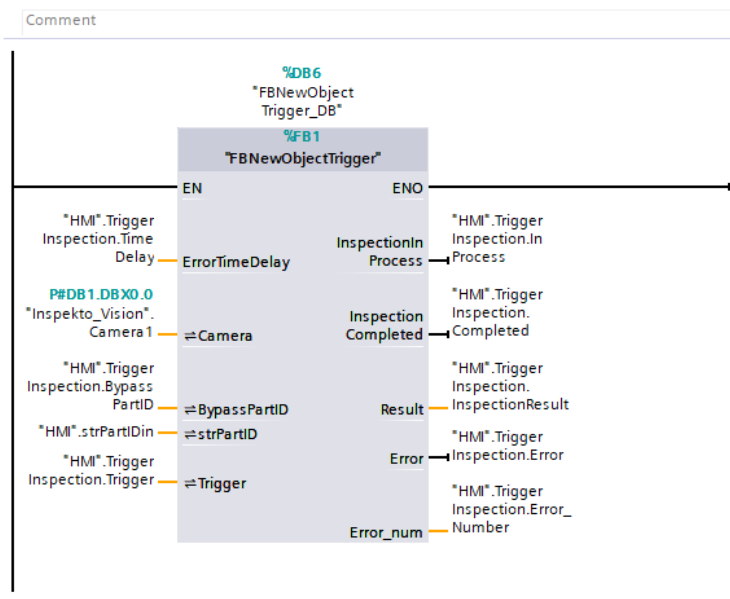


Figure 2-54

To trigger a new inspection following conditions must be met:

- The input parameter "Trigger" receives an rising edge.
- The value of the input parameter "strPartID" has changed from the last inspection.

When working with the inspection mode "External Trigger- Part ID provided by the system", the function block will change automatically the value of the "strPartID" so that the inspection is triggered without changing this parameter manually since the part ID is assigned by the Inspekto system.

As described in the previous section of this document, for the purpose of this application example, two different trigger logics are presented:

- **Manual** : Inspections are triggered manually by the user via the HMI.
- **Periodical**: Inspections are triggered periodically every 10 seconds.

Each mode can be activated by the user from the HMI by using 2 independent buttons.

NOTE The above-mentioned trigger logics are just an example and the user program the trigger logic according to your needs.

NOTE The programming logic presented below is done only for demo purpose and is not meant to be used in a productive environment. The programming logic must be adapted based on your automation process.

1. Open the "HMI DB" within the PLC project and create the tags that will handle the trigger logic for the inspection, manual and periodical respectively.

Inspekto_ProfinetIntegration > PLC_1 [CPU 1513F-1 PN] > Program blocks > HMI [DB2]									
HMI									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	
1	Static								
2	NewBatchID	String[64]	"		☒	☒	☒		
3	InspectioMode	Byte	16#0		☒	☒	☒		
4	NewPprofileID	Word	16#0		☒	☒	☒		
5	strPartID_out	String[64]	"		☒	☒	☒		
6	strPartIDin	String	"	☒	☒	☒	☒		
7	Profile_Update	Struct			☒	☒	☒		
8	Mode_Update	Struct			☒	☒	☒		
9	BatchID_Update	Struct			☒	☒	☒		
10	TriggerInspection	Struct			☒	☒	☒		
11	manuallMode	Bool	false	☒	☒	☒	☒		
12	manualTrigger	Bool	false	☒	☒	☒	☒		
13	periodicalMode	Bool	false	☒	☒	☒	☒		
14	periodicalTrigger	Bool	false	☒	☒	☒	☒		
15	inspectionTrigger	Bool	false	☒	☒	☒	☒		
16	<Add new>								

Figure 2-55

2. Create the corresponding HMI tags for both operation modes and link them to the corresponding tags created previously.

Inspekto_ProfinetIntegration > HMI_2 [MTP700 Unified Comfort] > HMI tags									
HMI tags									
	Name	Tag table	Data type	Connection	PLC name	PLC tag	Addr...	Access mode	Acquisition cycle
	manuallMode	Default tag table	Bool	HMI_Connection_1	PLC_1	HM.manuallMode		<symbolic access>	T1s
	manualTrigger	Default tag table	Bool	HMI_Connection_1	PLC_1	HM.manualTrigger		<symbolic access>	T1s
	periodicalMode	Default tag table	Bool	HMI_Connection_1	PLC_1	HM.periodicalMode		<symbolic access>	T1s

Figure 2-56

3. Create two buttons within the HMI screen to switch between the manual and periodical modes.

Once one of the modes is activated the other must be deactivated. In this case, this operation is implemented by using SetBitInTag and ResetBitInTag functions.

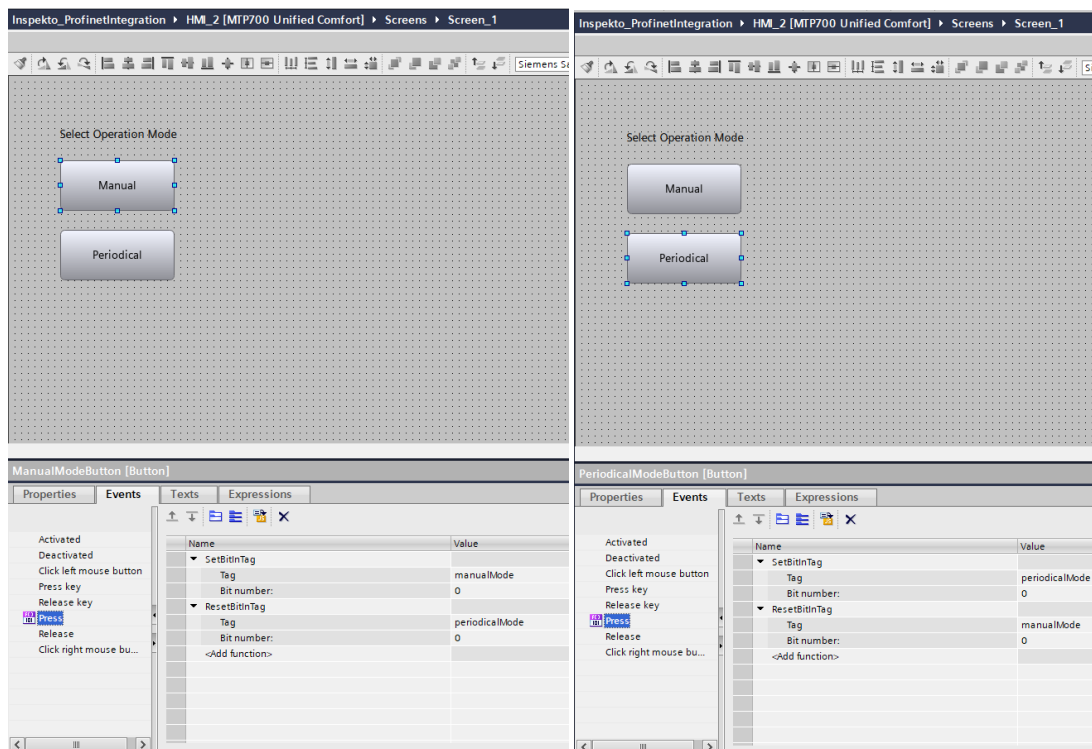


Figure 2-57

4. Create a new button that will be used to trigger the inspection while the system is in manual mode.

Pressing this button the tag that represents the manual trigger will change to True. In addition, the button will only be visible when the Manual mode is activated. For this purpose, the visibility feature of the button will be dynamic.

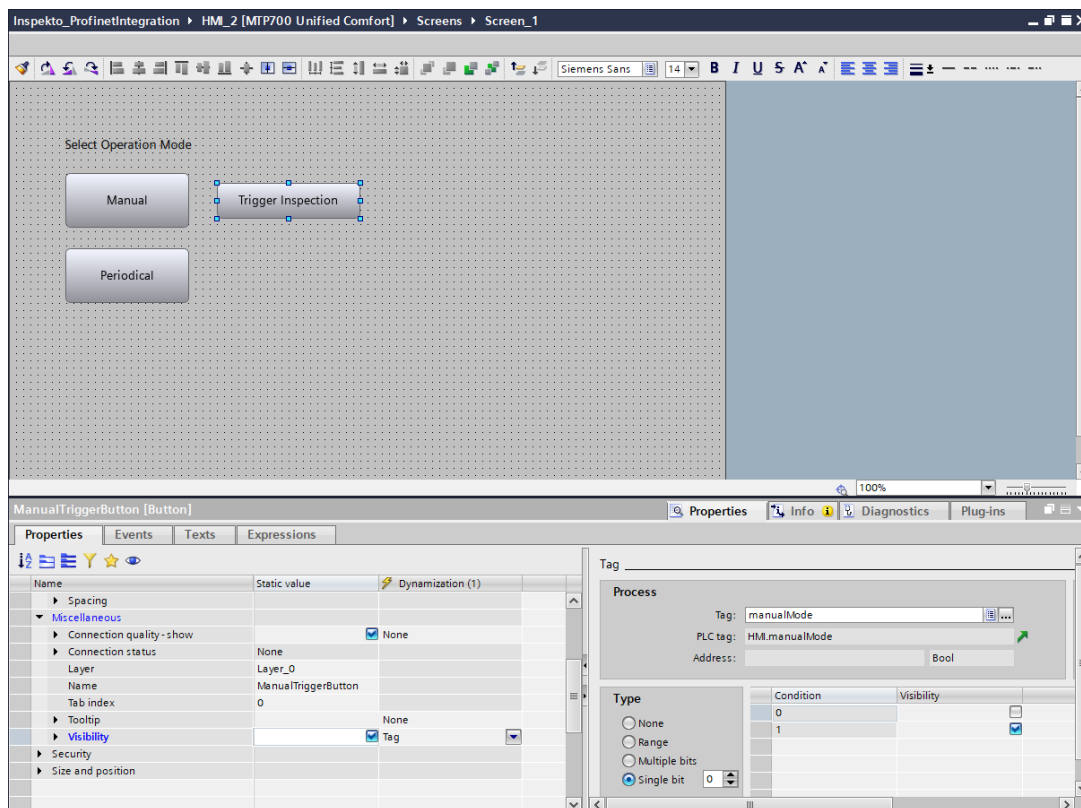


Figure 2-58

5. Create a new network in the main program to generate the periodical trigger every 10 seconds.

For this purpose, in this case a TON timer is used.

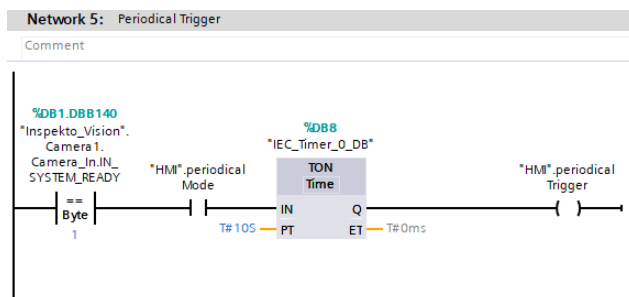


Figure 2-59

6. Create a new network in the main program to include the manual trigger and handle both operation modes.

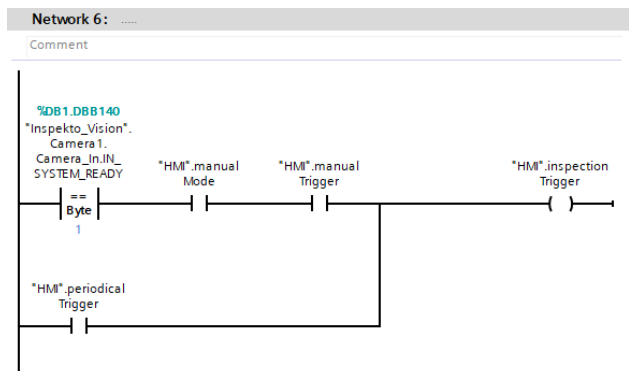


Figure 2-60

7. Connect the trigger inspection tag to the corresponding input parameter of the "FBNewObjectTrigger" function block.

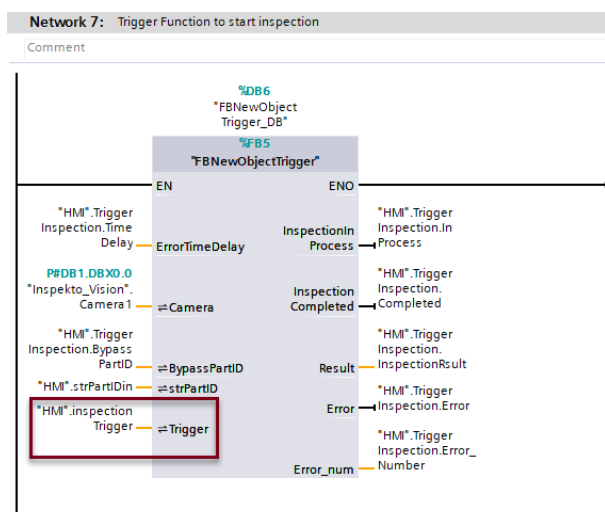


Figure 2-61

8. Compile and download the PLC program and the HMI program. If you do not have a physical Unified Comfort Panel you can use the simulation mode.
9. Open the HMI runtime and select the manual mode. The trigger inspection button should appear.

10. Press the trigger inspection button and check within the Inspekto UI that a new inspection was triggered.

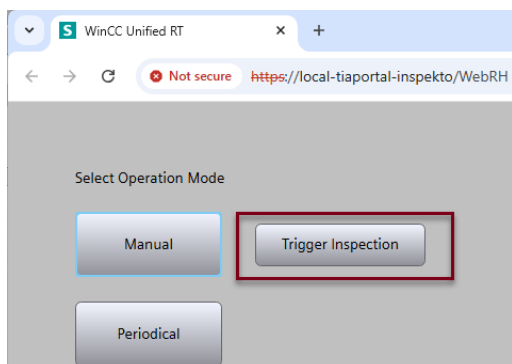


Figure 2-62

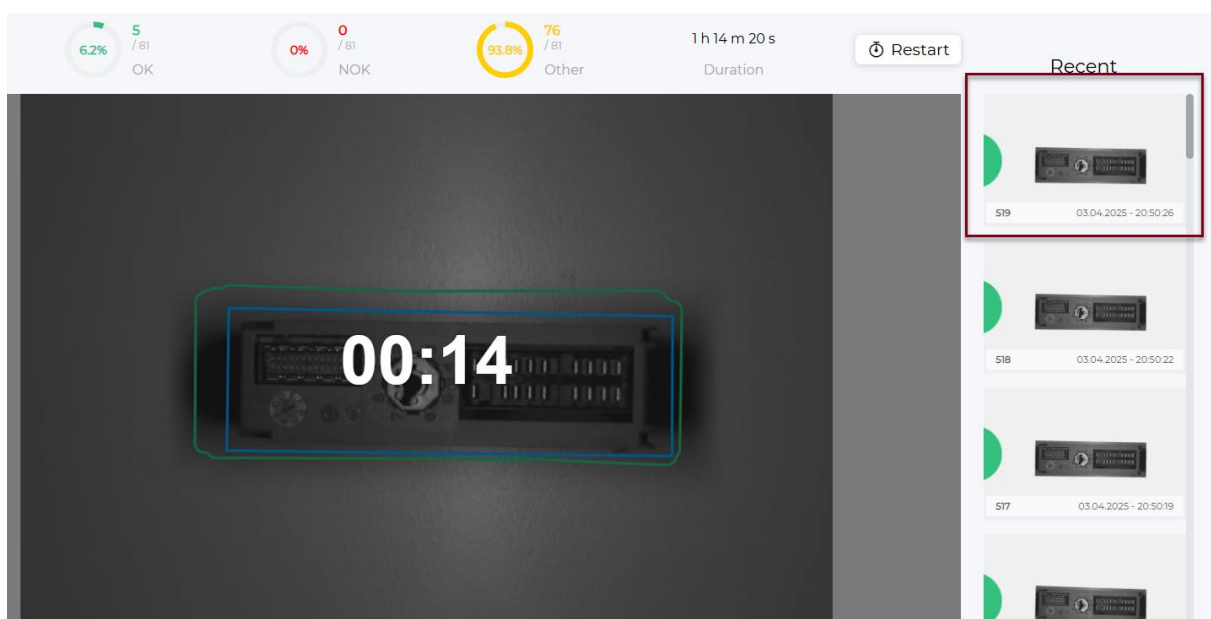


Figure 2-63

11. Press the periodical mode button to switch to the periodical operation mode.

12. Check within the Inspekto UI that a new inspection is triggered every 10 seconds.

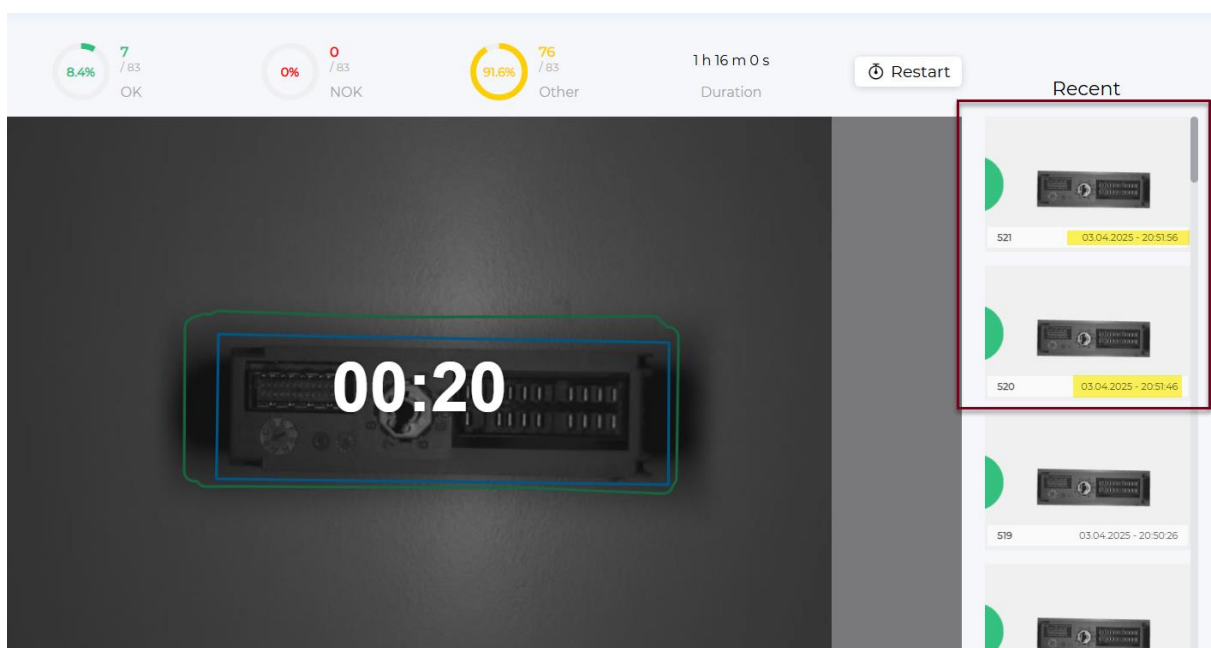


Figure 2-64

2.5.3.2. HMI programming – Inspection result visualization

Once the inspection is finished on the Inspekto system, it will send back to the PLC the result over the IN_OBJECT_INSPECTION_RESULT tag.

This section provides an example of how the user could the result of the inspection within the HMI program in a very simple way.

1. Create a new HMI tag and link it with the PLC tag IN_OBJECT_INSPECTION_RESULT from the "Inspekto_Vision" DB.

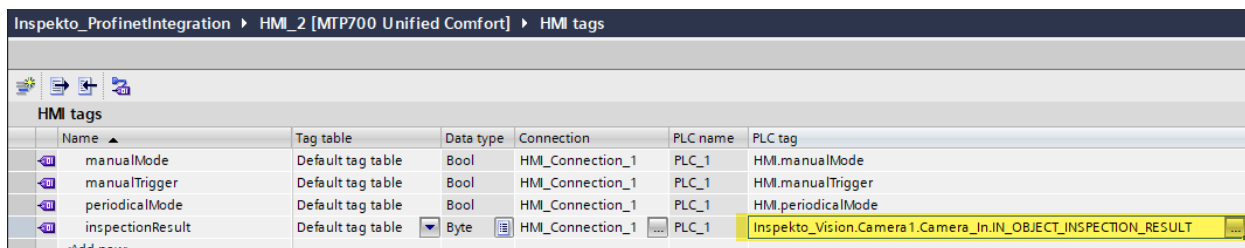


Figure 2-65

2. Add an I/O Field element to the HMI screen and use the previous created HMI tag as process value.

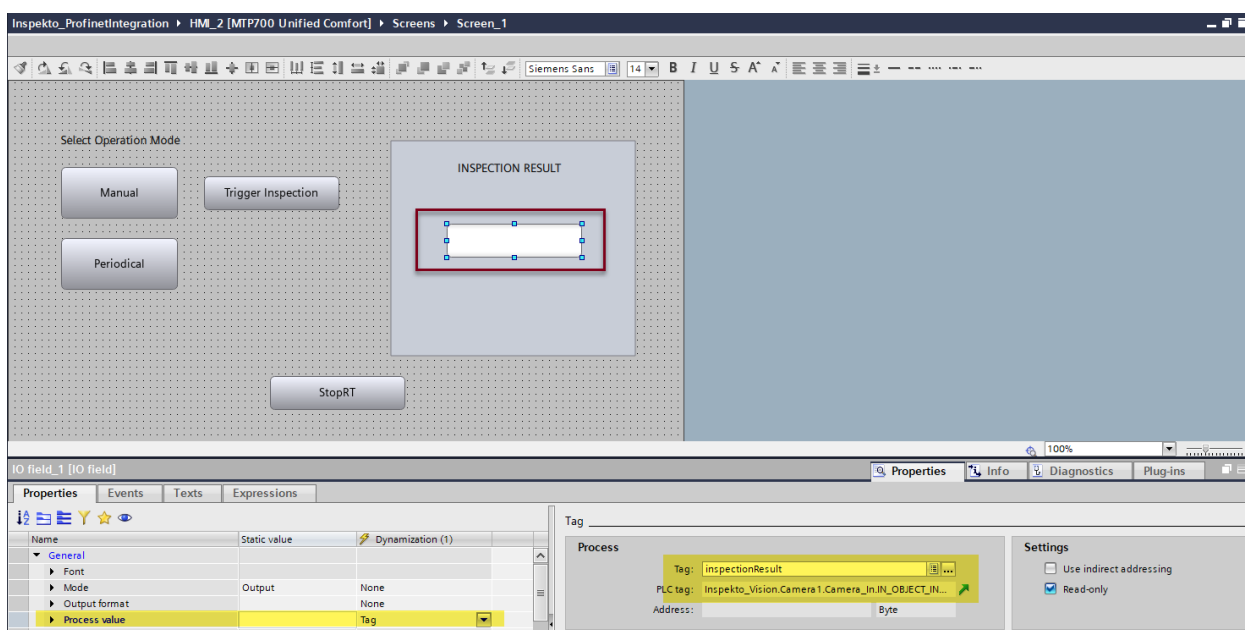


Figure 2-66

3. Define the range of the possible value the HMI tag might have and assign the corresponding text.

0= IDLE; 1= INSPECTION PASS; 2= INSPECTION FAILED; 3= OBJECT NOT INSPECTED;255= OBJECT NOT DETECTED

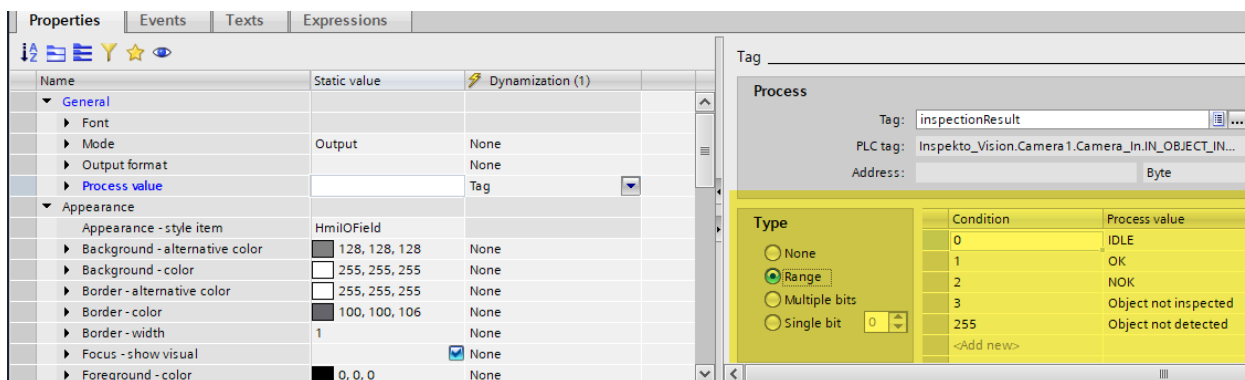


Figure 2-67

4. Trigger a new inspection and check that the inspection result is displayed correctly in the I/O field.

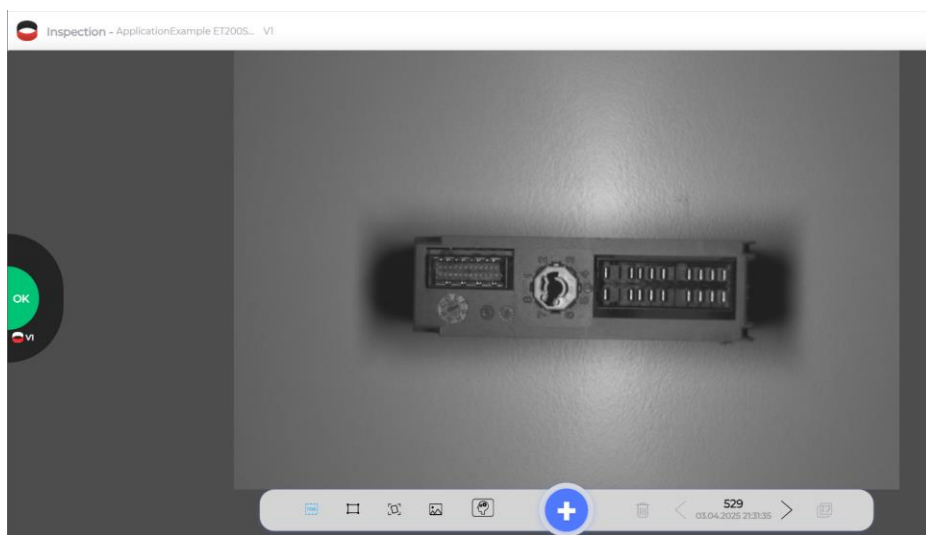


Figure 2-68

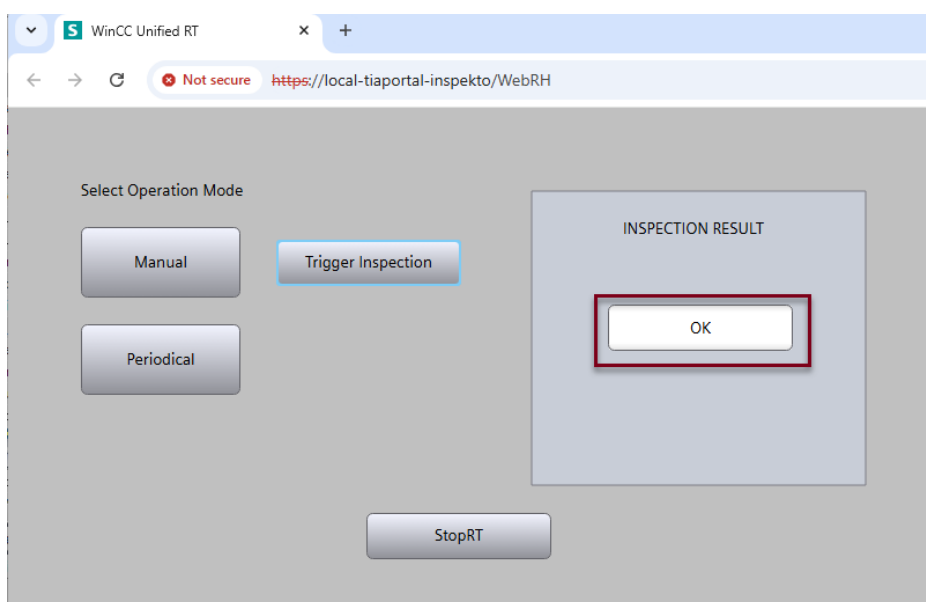


Figure 2-69

5. Change the part with a defective one, trigger a new inspection and check the inspection result.

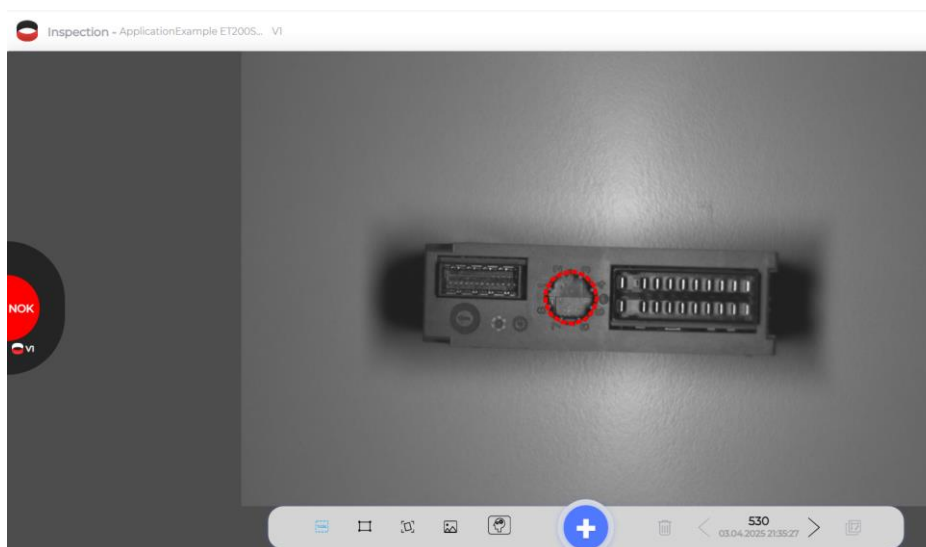


Figure 2-70

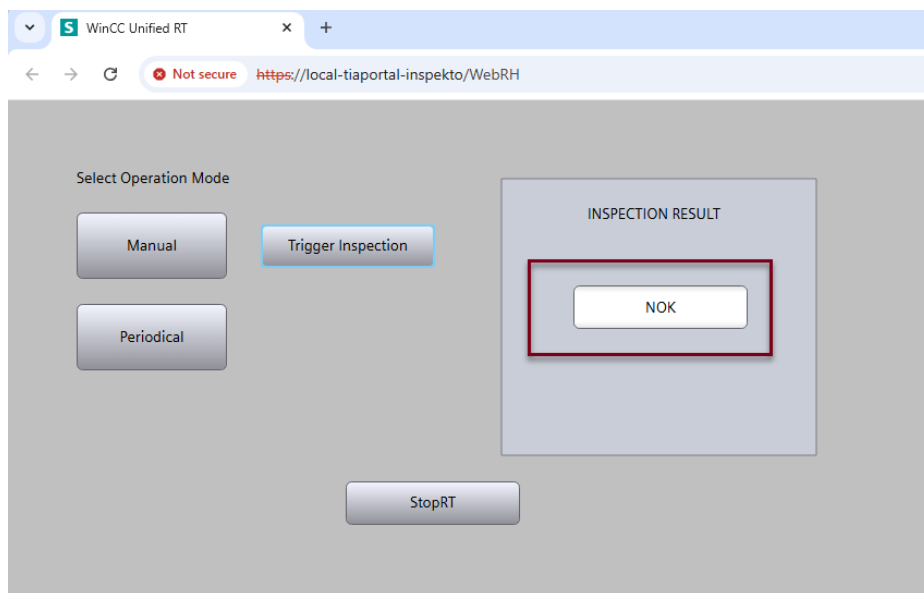


Figure 2-71

3. Additional information

3.1. Inspekto Functions and TIA Portal Function blocks

The SIEMENS Inspekto VIS system v6.0 supports the following functions:

Function	Data Type	Direction	Explanation (see below for detailed explanations)
OUT_INSPECTION_MODE	Byte	PLC -> INSPEKTO	Sets the system's triggering mode.
OUT_NEW_OBJECT_TRIGGER	Array64Chars	PLC -> INSPEKTO	Sets a new inspection trigger
OUT_PROFILE_ID	Word	PLC -> INSPEKTO	Load inspection profile with profile connectivity ID XX
OUT_BATCH_ID	Array64Chars	PLC -> INSPEKTO	Start a new batch with batch ID XX
IN_CUR_INSPECTION_MODE	Byte	INSPEKTO -> PLC	Report the current inspection mode (trigger mode)
IN_INSPECTION_MODE_RESPONSE	Byte	INSPEKTO -> PLC	a bit to acknowledge that the OUT_INSPECTION_MODE command has been performed
IN_CUR_OBJECT_ID	Array64Chars	INSPEKTO -> PLC	Report the current object ID
IN_NEW_OBJECT_TRIGGER_RESPONSE	Byte	INSPEKTO -> PLC	a bit to acknowledge that the OUT_NEW_OBJECT_TRIGGER command has been performed
IN_CUR_PROFILE_ID	Word	INSPEKTO -> PLC	Report the current inspection profile ID
IN_PROFILE_ID_RESPONSE	Byte	INSPEKTO -> PLC	a bit to acknowledge that the OUT_PROFILE_ID command has been performed
IN_CUR_BATCH_ID	Array64Chars	INSPEKTO -> PLC	Report the current Batch ID
IN_BATCH_ID_RESPONSE	Byte	INSPEKTO -> PLC	a bit to acknowledge that the OUT_BATCH_ID command has been performed
IN_KEEP_ALIVE	Byte	INSPEKTO -> PLC	a bit that periodically being updated between values of 0 and 1 to indicate the system is online. The flipping period can be configured.
IN_OBJECT_INSPECTION_RESULT	Byte	INSPEKTO -> PLC	Report the last inspection cycle result
IN_SYSTEM_ERROR	Byte	INSPEKTO -> PLC	report an error in case there is one, values may differ given the error state and the command that was sent to the INSPEKTO S70
IN_SYSTEM_READY	Byte	INSPEKTO -> PLC	report whether the system is ready to receive a new command or not
IN_ROIS_DEFECT_LIST	Array64Chars	INSPEKTO -> PLC	reports defects per ROI, up to 64 ROIs

Figure 3-1

The version v6.1 of Inspekto supports additionally the function to shut down the system from the PLC

Function	Data Type	Direction	Explanation (see below for detailed explanations)
OUT_INSPECTION_MODE	Byte	PLC -> S70	Sets the system's triggering mode.
OUT_NEW_OBJECT_TRIGGER	Array64Chars	PLC -> S70	Sets a new inspection trigger
OUT_PROFILE_ID	Word	PLC -> S70	Load inspection profile with profile connectivity ID XX
OUT_BATCH_ID	Array64Chars	PLC -> S70	Start a new batch with batch ID XX
OUT_PLD_SHUTDOWN	Byte	PLC -> S70	Shut down the Inspekto VIS system
IN_CUR_INSPECTION_MODE	Byte	S70 -> PLC	Report the current inspection mode (trigger mode)
IN_INSPECTION_MODE_RESPONSE	Byte	S70 -> PLC	a bit to acknowledge that the OUT_INSPECTION_MODE command has been performed
IN_CUR_OBJECT_ID	Array64Chars	S70 -> PLC	Report the current object ID
IN_NEW_OBJECT_TRIGGER_RESPONSE	Byte	S70 -> PLC	a bit to acknowledge that the OUT_NEW_OBJECT_TRIGGER command has been performed
IN_CUR_PROFILE_ID	Word	S70 -> PLC	Report the current inspection profile ID
IN_PROFILE_ID_RESPONSE	Byte	S70 -> PLC	a bit to acknowledge that the OUT_PROFILE_ID command has been performed
IN_CUR_BATCH_ID	Array64Chars	S70 -> PLC	Report the current Batch ID
IN_BATCH_ID_RESPONSE	Byte	S70 -> PLC	a bit to acknowledge that the OUT_BATCH_ID command has been performed
IN_KEEP_ALIVE	Byte	S70 -> PLC	a bit that periodically being updated between values of 0 and 1 to indicate the system is online. The flipping period can be configured.
IN_OBJECT_INSPECTION_RESULT	Byte	S70 -> PLC	Report the last inspection cycle result
IN_SYSTEM_ERROR	Byte	S70 -> PLC	report an error in case there is one, values may differ given the error state and the command that was sent to the INSPEKTO S70
IN_SYSTEM_READY	Byte	S70 -> PLC	report whether the system is ready to receive a new command or not
IN_ROIS_DEFECT_LIST	Array64Chars	S70 -> PLC	reports defects per ROI, up to 64 ROIs
IN_PLD_SHUTDOWN_RECEIVED	Byte	S70->PLC	reports that a PLC shutdown command has been received

Figure 3-2

3.1.1. PLC -> SIEMENS INSPEKTO VIS functions

All of the PLC -> SIEMENS Inspekto VIS commands (also listed as OUT_XXXX commands) will be considered as new commands only after the previous value has changed. Therefore, in order to issue a command, the PLC must update the value. For example – when working with several profiles, if the part ID for the first profile was "12345" the part ID for the next profile cannot be "12345"

In cases where the same ID would be needed it is recommended to add another character that will imply for the profile it is being used, for example – "12345A" for profile 1 and "12345B" for profile 2 where profile A can represent one side of the part to be inspected and profile B can represent another side of the part to be inspected.

3.1.1.1. OUT_INSPECTION_MODE (1 byte)

Inspection mode controls the SIEMENS Inspekto VIS system's trigger mode out of the following available options:

Inspection Mode	Value
OFF	0
Auto - Part ID provided by the PLC	1
Auto - Part ID generated by the system	2
Manual	3
External - Part ID generated by the system	4
External - Part ID provided by the PLC	5
Direct Trigger - Part ID generated by the system	6
Direct Trigger - Part ID provided by the PLC	7
Auto - Part ID generated by the external code reader	8

Table 3-1

Users can also modify inspection modes using the system user interface by going to menu "Profile settings -> Preferences".

Profile settings

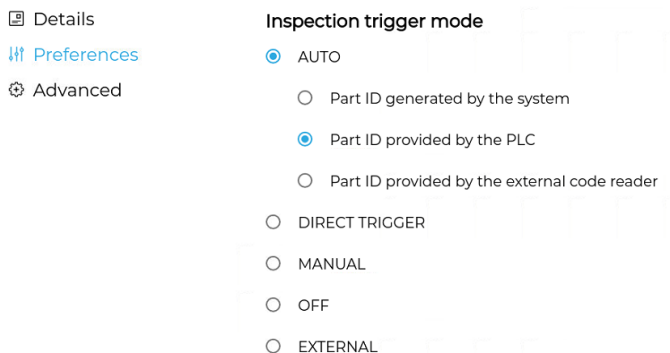


Figure 3-3

NOTE

The order in which trigger modes appear in the user interface is different from the associated value needed when using the corresponding function block to change the inspection mode from the PLC.

- **OFF** : in this mode the system will not respond to trigger signals and will not perform inspections. Use this mode if the system should not be active.
- **Auto – Part ID provided by the PLC**: in this mode the system will automatically identify the part for inspection, wait until it is stationary, and without vibrations and will perform the inspection.

In order to enable the system to identify the part the PLC needs to send a new part ID that will be assigned to the inspection image. If a new part ID will not be sent, a new inspection will not take place.

When using this option users should set "Inspection Timeout" that will fit to the line restrictions. Once a trigger signal has been sent, and a part was not identified before the timeout has been reached, the system will issue an "Alert"

result stating "Part not found". It is recommended that the PLC will treat "Alert" result as NOK. In order to set "Inspection Timeout" navigate to Profile settings -> Preferences -> scroll down and set the desired value (minimum value is 500ms).

Profile settings

Details | **Preferences** | Advanced

☐ Part ID provided by the external code reader

☐ DIRECT TRIGGER

☐ MANUAL

☐ OFF

☐ EXTERNAL

Inspection Interval

Auto Trigger Inspection Interval (ms) (33-999)

Inspection Timeout

Inspection Timeout (ms) (500-999)

Figure 3-4

- **Auto – Part ID generated by the system:** in this mode the system will automatically identify the part for inspection, wait until it is stationary, and without vibrations and will perform the inspection. In this mode there is no need to send a trigger signal from the PLC, the system will trigger automatically and assign a unique part ID to the inspection image.
- **Manual:** when selecting this mode, an "Inspect" button will appear in the center bottom of the screen:



Figure 3-5

- **External – Part ID generated by the system:** this mode requires connection of an industrial sensor (e.g. photoelectric sensor) to the Smart Light Power & ET cable (see connection diagram above). A new inspection will be triggered when the connected sensor will issue a 24VDC signal. In this mode there is no need to send a trigger signal from the PLC, the system will trigger automatically once a signal is issued from the external sensor.

NOTE

When selecting this inspection mode, it is also required to specify whether the connected sensor is of type PNP/NPN. In order to set the sensor type - in the inspection modes screen, scroll down and select the wanted type:

Profile settings

Details | **Preferences** | Advanced

Inspection trigger mode

☐ AUTO

☐ DIRECT TRIGGER

☐ MANUAL

☐ OFF

☒ EXTERNAL

☒ Part ID generated by the system

☐ Part ID provided by the PLC

Sensor type

☒ PNP ☐ NPN

Figure 3-6

NOTE

When selecting External trigger inspection mode, the system will not show a live video stream. Instead, the system will present a timer stating the time passed since the last inspection

- **External – Part ID provided by the PLC:** this mode requires connection of an industrial sensor (i.e. photoelectric sensor) to the Smart Light Power & ET cable (see connection diagram above). In order to start a new inspection, the PLC needs to send a new part ID that will be assigned to the inspection image. **The system will react to the connected external sensor signal only after receiving a new part ID from the PLC**, the received part ID will be assign to the inspection image.

NOTE

When selecting this inspection mode, it is also required to specify whether the connected sensor is of type PNP/NPN. In order to set the sensor type - in the inspection modes screen, scroll down and select the wanted type:

Profile settings

Figure 3-7

- **Direct Trigger – Part ID generated by the system:** this mode works in the exact same way as "External – Part ID generated by the system", but instead of receiving the trigger signal from an external sensor, it will receive it from the PLC. In order to generate a new inspection cycle the PLC needs to send a new part ID. In this mode the new part ID will **not** be assigned to the inspection image, it is only used to initiate a new inspection cycle and the system will assign a unique part ID to the inspection image.

NOTE

When selecting Direct Trigger inspection mode, the system will not show a live video stream. Instead, the system will present a timer stating the time passed since the last inspection.

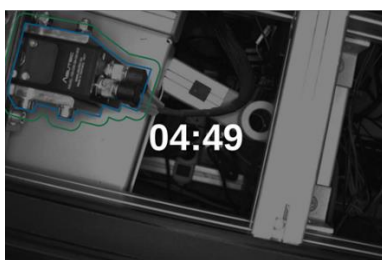


Figure 3-8

- **Direct Trigger – Part ID generated by the PLC** this mode works in the exact same way as "External – Part ID provided by the PLC", but instead of receiving the trigger signal from an external sensor, it will receive it from the PLC. In order to generate a new inspection cycle the PLC needs to send a new part ID. In this mode the new part ID will be assigned to the inspection image.

NOTE

When selecting Direct Trigger inspection mode, the system will not show a live video stream. Instead, the system will present a timer stating the time passed since the last inspection.

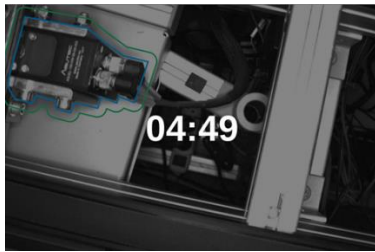


Figure 3-9

- **Auto -Part ID generated by the external code reader:** this mode requires a connection of a USB code reader to the SIEMENS Inspekto VIS system controller. When this mode is selected, a new inspection cycle will start only after the connected code reader will stream the read string to the system.

Once a new string has been streamed from the code reader a new inspection will start and the streamed string will be assigned as the inspection image part ID.

The system can work with any USB code reader that has an option to act as UID, please refer to your code reader manual to verify this option is available.

NOTE

In order to inspect moving objects, the system will need to be set to one of the following inspection modes:

- External – Part ID generated by the system.
- External – Part ID provided by the PLC.
- Direct Trigger – Part ID generated by the system.
- Direct Trigger – Part ID provided by the PLC.

3.1.1.2. OUT_NEW_OBJECT_TRIGGER (64 bytes null-terminated ASCII string)

When selecting a corresponding inspection mode (any of the inspection modes that ends with "Part ID provided by the PLC), updating this value will start a new inspection cycle. In the SIEMENS Inspekto VIS system, in order to issue a trigger, the PLC needs to send a new object ID (sometimes referred to as "Part ID") to the SIEMENS Inspekto VIS system.

- The provided Part ID needs to be a different value than the previous part ID the system received.
- In order for the system to receive the new Part ID and start a new inspection cycle:
 - The system needs to have an active profile loaded. If a new Part ID is sent when there is no active profile loaded, the system will return an Error (see below for errors details).
 - The "System Ready" signal has to be with value "1" stating "System Ready" (value "0" states "System Not Ready"). If a new part ID is sent when "System Ready" = "0" the system will return an Error (see below for details).
 - The maximum length of the provided Part ID is 64 characters.
 - The allowed characters are all numbers and letters.

3.1.1.3. OUT_PROFILE_ID (16-bit integer)

When updating this value, the system will load a profile with the corresponding PLC integration ID.

There are 2 ways to view a profile PLC integration ID:

1. When the profile is loaded by clicking on the settings profile icon in the top left corner of the screen.



Figure 3-10

2. When the profile is not loaded:
 - a. Click on the "Profile" button on the left side panel

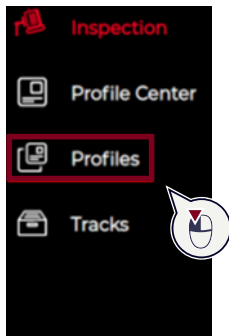


Figure 3-11

- b. Click on the "Settings" button of the relevant profile and the PLC integration ID will appear under "Details" tab:

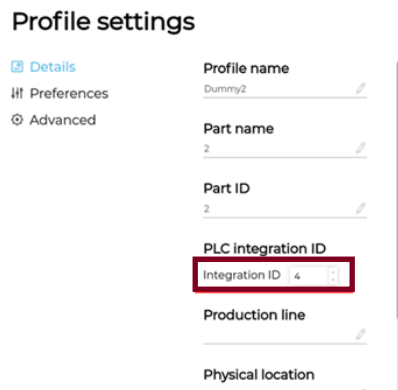


Figure 3-12

Users can modify the PLC integration ID of a profile to any number ranging from 1 – 99.999.

NOTE PLC integration ID's must be unique per profile.

In order for the system to receive the Profile ID value and load a new profile:

1. The "System Ready" signal must be with value "1" stating "System Ready" (value "0" states "System Not Ready"). If a new part ID is sent when "System Ready" = "0" the system will return an Error (see below for details).
2. The system must have a profile with the received PLC integration ID value. If received a value that does not apply to any profile in the system, the system will send a "Profile not loaded" error (see below for details).

3.1.1.4. OUT_BATCH_ID (64 bytes null-terminated ASCII string)

When updating this value the system will reset all counters and assign a new batch ID.

- Batch ID can not be seen in the SIEMENS Inspekto VIS system screen.
- The system needs to have an active profile loaded. If a new Part ID is sent when there is no active profile loaded, the system will return an Error (see below for errors details).
- The "System Ready" signal has to be with value "1" stating "System Ready" (value "0" states "System Not Ready"). If a new part ID is sent when "System Ready" = "0" the system will return an Error (see below for details).
- The maximum length of the provided Part ID is 64 characters.
- The allowed characters are all numbers and letters.

3.1.1.5. OUT_PLC_SHUTDOWN

NOTE This signal can only be used with Inspekto v6.1

When setting this value to "1" the Inspekto VIS will perform a system shutdown.

- Re-activating the system requires manual power on via the system's IPC.
- Note that if the value stays at "1" when powering on the system, when the system automatically reconnect to the PLC it will read this value and shut down again.

3.1.2. SIEMENS Inspekto VIS -> PLC functions

After receiving an "OUT" command from the PLC the SIEMENS Inspekto VIS system will respond with a corresponding "IN" response. The corresponding "IN" response will change for a timed period which can be determined in the system settings by setting the "SIEMENS Inspekto VIS to PLC response pulse width(ms)" for some responses and stay in the same state until the next command has been received for others (see details below).

In order to change "SIEMENS Inspekto VIS to PLC response pulse width(ms)" value:

- Open the system settings by pressing on the  icon in the bottom left corner of the screen.
- Under "General" tab, scroll down to the "Connectivity" section:

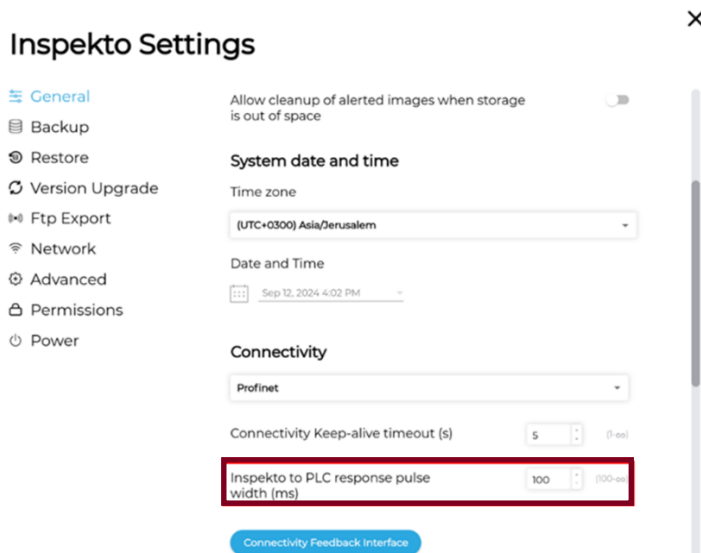


Figure 3-13

3.1.2.1. IN_CUR_INSPECTION_MODE

After successfully receiving a new OUT_INSPECTION_MODE command, the system will update this value to the current inspection mode.

This value will stay constant until there is a change in inspection mode (either done by sending OUT_INSPECTION_MODE command from the PLC or by changing the inspection mode in the system user interface).

3.1.2.2. IN_INSPECTION_MODE_RESPONSE

This value indicates to the PLC regarding the status of a new OUT_INSPECTION_MODE command that was received.

- This value will change for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".
- Possible values:
 - 0 = IDLE – the system will show this value when there is no new OUT_INSPECTION_MODE command or after the pulse width stated in "SIEMENS Inspekto VIS to PLC response pulse width(ms)" has passed.
 - 1 = SUCCESS – stating the last OUT_INSPECTION_MODE command received was performed successfully, and the current inspection mode is updated in "IN_CUR_INSPECTION_MODE".
 - 2 = FAIL – stating that the last OUT_INSPECTION_MODE command received was not performed.

When getting this value – a description of the reason can be seen in the "System Error" signal (see details below).

3.1.2.3. IN_CUR_OBJECT_ID

After successfully receiving a new OUT_NEW_OBJECT_TRIGGER command and the inspection cycle is completed, the system will update this value to the current Object ID (also refers to as PART ID).

This value will stay constant until there is a change in object ID a new object ID will be received after the next inspection cycle, whether it was triggered by the PLC or by the system user interface.

3.1.2.4. IN_NEW_OBJECT_TRIGGER_RESPONSE

This value indicates to the PLC regarding the status of a new OUT_NEW_OBJECT_TRIGGER command that was received. This value will change for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

- Possible values:
 - 0 = IDLE – the system will show this value when there is no new OUT_NEW_OBJECT_TRIGGER command or after the pulse width stated in "SIEMENS Inspekto VIS to PLC response pulse width(ms)" has passed.
 - 1 = SUCCESS – stating the last OUT_NEW_OBJECT_TRIGGER command received was performed successfully, and the current object ID is updated in "IN_CUR_OBJECT_ID".
 - 2 = FAIL – stating that the last OUT_NEW_OBJECT_TRIGGER command received was **not** performed.

When getting this value – a description of the reason can be seen in the "System Error" signal (see details below).

3.1.2.5. IN_CUR_PROFILE_ID

After successfully receiving a new "OUT_PROFILE_ID" command and the new profile has been loaded, the system will update this value to the current active profile.

This value will stay constant until there is a change in the active profile of the system (either done by sending OUT_PROFILE_ID command from the PLC or by changing the profiles in the system user interface).

3.1.2.6. IN_PROFILE_ID_RESPONSE

This value indicates to the PLC regarding the status of a new OUT_PROFILE_ID command that was received. This value will change for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

- Possible values:
 - 0 = IDLE – the system will show this value when there is no new OUT_PROFILE_ID command or after the pulse width stated in "SIEMENS Inspekto VIS to PLC response pulse width(ms)" has passed.
 - 1 = SUCCESS – stating the last OUT_PROFILE_ID command received was performed successfully, and the current profile ID is updated in "IN_CUR_PROFILE_ID".
 - 2 = FAIL – stating that the last OUT_PROFILE_ID command received was **not** performed.

When getting this value – a description of the reason can be seen in the "System Error" signal (see details below).

3.1.2.7. IN_CUR_BATCH_ID

After successfully receiving a new "OUT_BATCH_ID" command, the counters have been reset and the new batch ID was received, the system will update this value to the current batch ID.

This value will stay constant until there is a change in the batch ID (either done by sending OUT_BATCH_ID command from the PLC or by resetting the batch in the system user interface).

3.1.2.8. IN_BATCH_ID_RESPONSE

This value indicates to the PLC regarding the status of a new OUT_BATCH_ID command that was received. This value will change for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

- Possible values:
 - 0 = IDLE – the system will show this value when there is no new OUT_BATCH_ID command or after the pulse width stated in "SIEMENS Inspekto VIS to PLC response pulse width(ms)" has passed.
 - 1 = SUCCESS – stating the last OUT_BATCH_ID command received was performed successfully, and the current batch ID is updated in "IN_CUR_BATCH_ID".
 - 2 = FAIL – stating that the last OUT_BATCH_ID command received was **not** performed.

When getting this value – a description of the reason can be seen in the "System Error" signal (see details below).

3.1.2.9. IN_KEEP_ALIVE

The system periodically updates a keep-alive, the IN_KEEP_ALIVE value flips between 0 and 1, to indicate the system is online.

- In order to change the flipping period:

- Open the system settings by clicking on the  icon in the bottom left corner of the screen.
- Under "General" tab, scroll down to the "Connectivity" section:

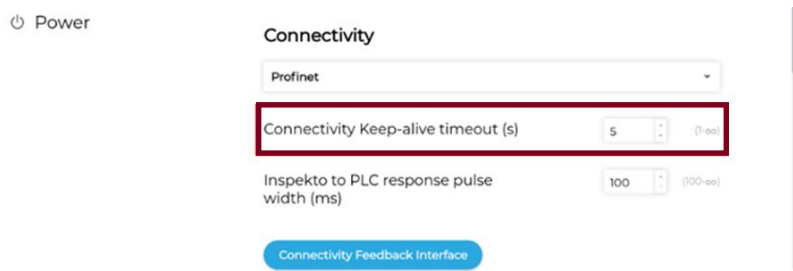


Figure 3-14

3.1.2.10. IN_OBJECT_INSPECTION_RESULT

This value will update after an inspection cycle has been completed with the following possible values:

- 0 = IDLE – the system will show this value if there is no new OUT_NEW_OBJECT_TRIGGER command or after the pulse width stated in "SIEMENS Inspekto VIS to PLC response pulse width(ms)" has passed.
- 1 = Inspection result PASS.
- 2 = Inspection result FAIL.
- 3 = Object detected but not inspected – when received, this value means that the system was able to detect the part but not to inspect it, due to its position in the image (see profile setup manual for detail explanation).

NOTE

It is recommended that the PLC will treat this value as a FAIL result.

- 255 = Object not detected, timeout reached – when received, this value means that the system was not able to detect the part for inspection.

NOTE

It is recommended that the PLC will treat this value as a FAIL result.

This value will stay constant until there is a new inspection result, whether it was triggered by the PLC or by the system user interface.

3.1.2.11. IN_SYSTEM_ERROR

This value will update in case that there is an error in a command received from the PLC.

• Possible values:

- **0 = IDLE** – the system will show this value if there is no new command received or after the pulse width stated in "S70 to PLC response pulse width(ms)" has passed.
- **1 = Camera disconnected** – this error means that the system does not identify a camera connected to the controller.

This value will remain constant as long as there is no camera identified by the system.

- **2 = Connectivity sync problem** – this error means that the system identifies a communication problem with the PLC.

This value will remain constant as long as the system cannot identify the PLC communication.

- **3 = Trigger mode OFF** – this error means that a trigger command was sent from the PLC while the system is in inspection mode = OFF (see above for detailed explanation).

This value will appear after receiving a trigger while inspection mode is set to OFF, and remain for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

- **4 = No profile loaded** - this error means that a command that requires an active profile in order to perform was sent while there was no active profile running (see above for the various commands requirements)

This value will appear after receiving a OUT_PROFILE_ID command, with an ID that does not exist, and remain for the duration stated in the "S70 to PLC response pulse width(ms)".

- **5 = Trigger stopped but not started** – internal error, if encountered repeatedly please contact support.
- **6 = System not ready** – this error means that a command was sent from the PLC while "SYSTEM_READY" was with value of "0".

This value will appear when receiving a command while SYSTEM_READY = 0, and will remain until SYSTEM_READY will change its value to "1".

- **7 = Multiple commands received** – this error means that the PLC sent multiple commands at once. In this situation the system will respond with an error because it will have no ability to know which command to perform first.

This value will appear if the PLC will send more than one command in a single point of time, and remain for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

- **8 = Profile incompatible with requested operation** – this error means that the requested operation could not be performed with the current active profile.

This value will remain for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

This value will change for the duration stated in the "SIEMENS Inspekto VIS to PLC response pulse width(ms)".

IN_SYSTEM_ERROR describes an error regarding the last command that was received from the PLC, there is no need to perform a reset operation after receiving an error as the value will change automatically back to IDLE ("0") according to the details stated above.

3.1.2.12. IN_SYSTEM_READY

This value indicates that the system is ready to receive a new command from the PLC.

- Possible values:

- 1 = System ready.
- 0 – System **not** ready.

When IN_SYSTEM_READY = "0", any command sent from the PLC will result with an IN_SYSTEM_ERROR = 6 and the requested command will not be performed.

3.1.2.13. IN_ROIS_DEFECT_LIST

The Inspekto system reports inspections results not only as a general OK/NOK result (as described in IN_INSPECTION_RESULT above), but also a result dedicated per ROI (**for detailed explanation regarding profiles and ROIs please see profile setup manual**).

ROI_DEFECT_LIST is represented with an array of 64 characters, where the position in the array represents the ROI ID.

- Value of "0" = ROI PASS.
- Value of "1" = Defect in ROI.

In cases where there are less than 64 ROIs in the active profile, only the characters representing the existing ROIs in the profile will update according to the inspection result.

For example when receiving: 00101000....00 it means that there are defects in ROI 3 and 5.

The default values are "0" in all of the characters, and the values will change only after an inspection took place **and** the inspection result was FAIL (IN_OBJECT_INSPECTION_RESULT = 2).

The system will update these values from the time the inspection has been completed until the "SIEMENS Inspekto VIS to PLC response pulse width(ms)" period has passed, after which the values will return to "0" in all characters.

3.1.2.14. IN_PLC_SHUTDOWN_RECEIVED

This value will update to 1 to acknowledge that the system has received an OUT_PLC_SHUTDOWN command and starting to shut down.

3.2. SIEMENS Inspekto VIS function blocks

NOTE

- Before sending any command, the PLC needs to verify that "IN_SYSTEM_READY" =1.
- After sending any command, "IN_SYSTEM_READY" will change to "0" while the command is being performed and change back to "1" when the operation was completed.
- "IN_SYSTEM_ERROR" refers to the status of the last command received, and it is not a system state. If "IN_SYSTEM_ERROR" =1, when "IN_SYSTEM_READY" =1 the system is ready to receive the next command.
- Although using the function blocks, all of the variables detailed in the GSDML are available for use. It is recommended that the PLC will verify any of the available relevant variables when working with Inspekto.

3.2.1. New Batch ID (FBUpdateBatchID)

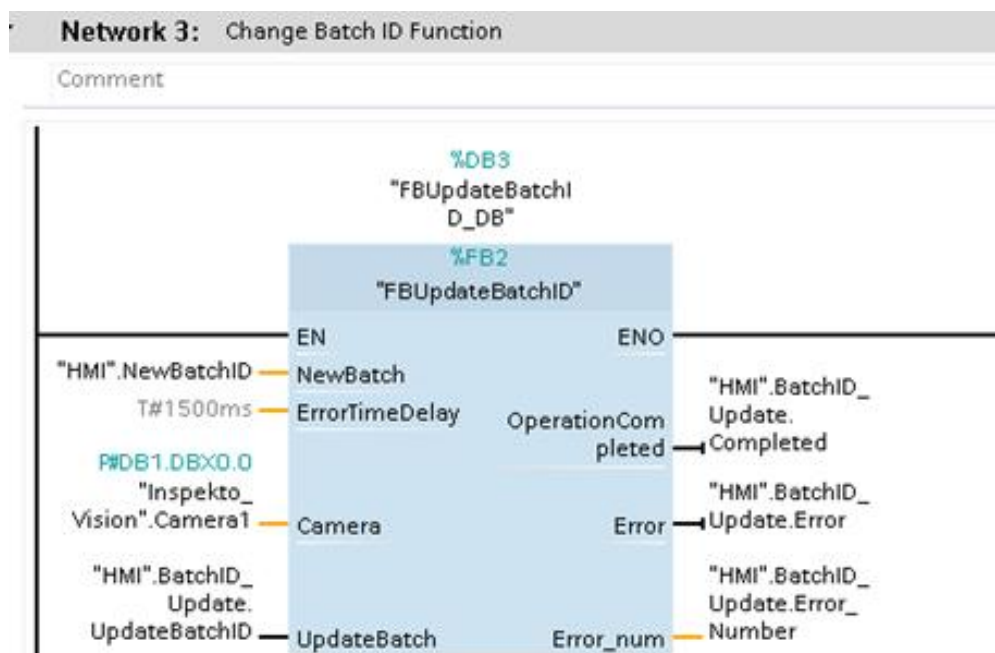


Figure 3-15

Parameter type	Name	Data type	Details
Input	NewBatch	String	Will hold the ID of the new batch to be sent to the PLC
Input	ErrorTimeDelay	Time	Default set to 1.5Sec. Will issue Error output if operation was not complete in set time
Input	Camera	Struct	Camera# from struct
Input	UpdateBatch	Bool	When set to true, will send command to the SIEMENS Inspekto VIS System
Outputs	OperationCompleted	Bool	Indicates that the operation was completed
Output	Error	Bool	Indicates that the operation resulted with an error
Output	Errorn_num	Byte	Will detail the error type in case Error bit was true

Table 3-2

Error bit will set to "true" if:

- After ErrorTimeDelay time "NewBatchID" is not equal to IN_CURRENT_BATCH_ID.
- IN_SYSTEM_ERROR bit is set to "true".

This function will change OUT_BATCH_ID to the value set to "NewBatchID".

The workflow for using the FBUpdateBatchID FB is as follows:

1. Assign camera from struct into the "Camera" input.
2. Set the new batch ID value to "NewBatch"
3. Set "UpdateBatch" to "true" to send the new batch ID to the Inspekto system.
4. Verify:
 - "OperationCompleted" changes to "True"
 - "Error" remains at "False".
 - Counters reset in the Inspekto system UI.
 - IN_SYSTEM_READY changed to "1".
5. Optional: Set "ErrorTimeDelay" to the wanted value (default 1.5s).

3.2.2. Change inspection mode (FBUpdateInspectionMode)

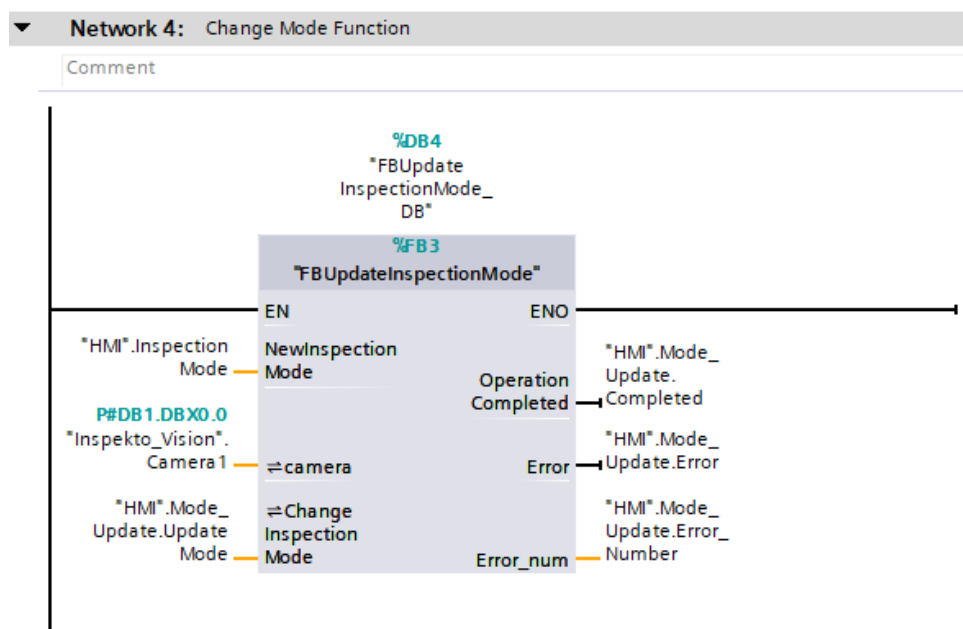


Figure 3-16

Parameter type	Name	Data type	Details
Input	NewInspectionMode	Byte	Will hold the value of the new inspection mode to be sent to the system
Input	Camera	Struct	Camera# from struct
Input	ChangeInspectionMode	Bool	When set to True will send the command to the Inspekto system assigning the value of "NewInspectionMode" to the Inspekto system.
Output	OperationCompleted	Bool	Indicates that the operation was completed
Output	Error	Bool	Indicates that the operation resulted with an error
Output	Error_num	Byte	Will detail the error type in case Error bit was "True"

Table 3-3

Error bit will set to "true" if:

- Set "NewInspectionMode" value does not exist.
- IN_CUR_INSPECTION_MODE did not change.
- IN_SYSTEM_ERROR bit is set to "true".

This function will change OUT_INSPECTION_MODE to the value set to "NewInspectionMode"

The workflow for using the FBUpdateInspectionModeID FB is as follows:

1. Assign camera from struct into the "Camera" input.
2. Assign the wanted inspection mode to "NewInspectionMode".
3. Set "ChangeInspectionMode" to "true" to send the new inspection mode to the Inspekto system.
4. Verify:
 - "OperationCompleted" changes to "true".
 - "Error" remains at "false".
 - IN_CUR_INSPECTION_MODE changed to the value set to "NewInspectionMode".
 - IN_SYSTEM_READY changed to "1".

3.2.3. Load profile (FBUpdateProfileID)

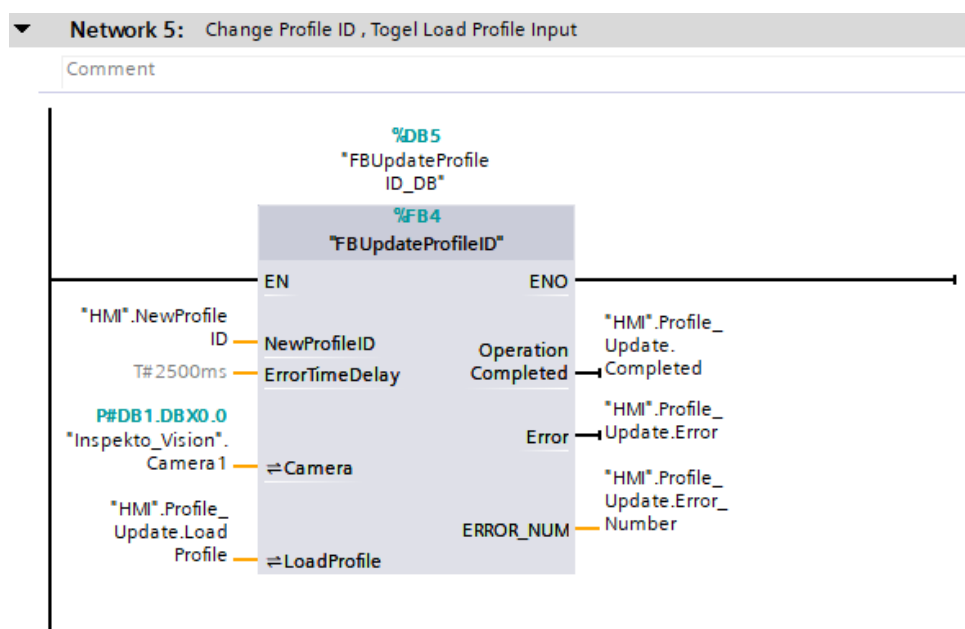


Figure 3-17

Parameter type	Name	Data type	Details
Input	NewProfileID	Word	Will hold the ID of the profile we want the system to load
Input	ErrorTimeDelay	Time	Default set to 2.5s. Will issue Error output if operation was not complete in set time.
Input	Camera	Struct	Camera # from struct
Input	LoadProfile	Bool	When set to "True" will send the command to the Inspekto system
Output	OperationCompleted	Bool	Indicates that the operation was completed
Output	Error	Bool	Indicates that the operation resulted with an error
Output	Error_num	Byte	Will detail the error type in case Error bit was "True"

Table 3-4

Error bit will set to "true" if:

- After ErrorTimeDelay time "NewProfileID" is not equal to IN_CURRENT_PROFILE_ID.
- IN_SYSTEM_ERROR bit is set to "true".

The function will change OUT_PROFILE_ID to the value set in "NewProfileID".

The workflow for using the FBUpdateProfileID FB is as follows:

1. Assign camera from struct into the "Camera" input.
2. Set the new profile ID value to "NewProfileID"
3. Set "LoadProfile" to "true" to send the new Profile ID to the Inspekto system.
4. Verify:
 - "OperationCompleted" changes to "true"
 - "Error" remains at "false".
 - IN_CUR_PROFILE_ID changed to the value set in "NewProfileID"
 - IN_SYSTEM_READY changed to "1".
5. Optional: set "ErrorTimeDelay" to the wanted value (default 2.5s).

3.2.4. Trigger inspection (FBNewObjectTrigger)

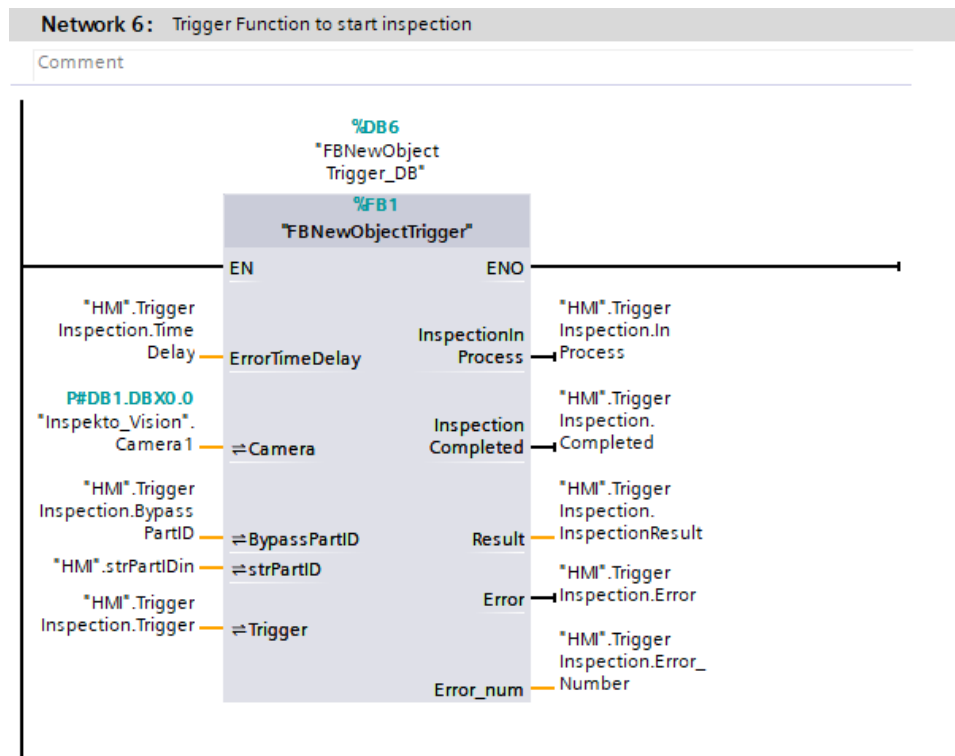


Figure 3-18

Parameter Type	Name	Data type	Details
Input	ErrorTimeDelay	Time	Default set to 2.5s. Will issue Error output if operation was not complete in set time
Input	Camera	Struct	Camera # from struct
Input	BypassPartID	Bool	When working in "Part ID is generated by the system" trigger modes – needs to be set to "True". When set to true will toggle Part ID between "A" and "B"
Input	strPartID	String	Will hold the Part ID that will be assigned to the next trigger

Parameter Type	Name	Data type	Details
Input	Trigger	Bool	When set to "True" will send to command to Inspekto system to start new inspection.
Output	InspectionInProgress	Bool	Will set to "True" while inspection is being performed
Output	InspectionCompleted	Bool	Indicates that the operation was completed
Output	InspectionResults	Byte	Will specify the inspection results.
Output	Error	Bool	Indicates that the operation resulted with an error
Output	Error_num	Byte	Will detail the error type in case Error bit was "True"

Table 3-5

Error bit will set to "true" if:

- After ErrorTimeDelay time "strPartID" is not equal to IN_CURRENT_OBJECT_ID.
 - Valid for "Part ID provided by the PLC" inspection modes only.
- IN_SYSTEM_ERROR bit is set to "true".

This function will change OUT_NEW_OBJECT_TRIGGER to the value set in "strPartID".

The workflow for using the "FBNewObjectTrigger" FB is as follows:

1. Assign camera from struct into the "Camera" input.
2. Set "BypassPartID" to "true" or "false" according to the current inspection mode (see inspection modes details).
3. Set "strPartID" to the wanted value of the next inspection's ID.
 - Relevant only for "Part ID provided by the PLC" inspection modes only (see inspection modes details).
4. Set "Trigger" to "true" to send the command to the Inspekto system.
5. Verify:
 - "OperationCompleted" changes to "true"
 - "Error" remains at "false".
 - InspectionResults will represent the last inspection's result (see inspection results details)
 - IN_CUR_OBJECT_ID changed to the value set in "strPartID"
 - IN_SYSTEM_READY changed to "1".
6. Optional: set "ErrorTimeDelay" to the wanted value (default 2.5s).

NOTE

- Method of adding the GSDML file and connecting the PLC to the Inspekto system might change according to the PLC model and vendor and/or software version in use.
- Please make sure to add an internal timeout function for commands issued to the Inspekto system. Using this will allow internal PLC monitoring of any issue that might occur.

3.3. Signal Flow Charts

3.3.1. OUT_NEW_TRIGGER

This section demonstrates the signal behaviour when an OUT_NEW_OBJECT_TRIGGER command is received, and how the relevant signals going from the SIEMENS Inspekto VIS system to the PLC will behave.

In the chart below each section relates to a different scenario (zoom in in order to view with more details).

Please see "Inspekto functions and TIA Portal function blocks" section of this document for detailed explanation regarding this command.

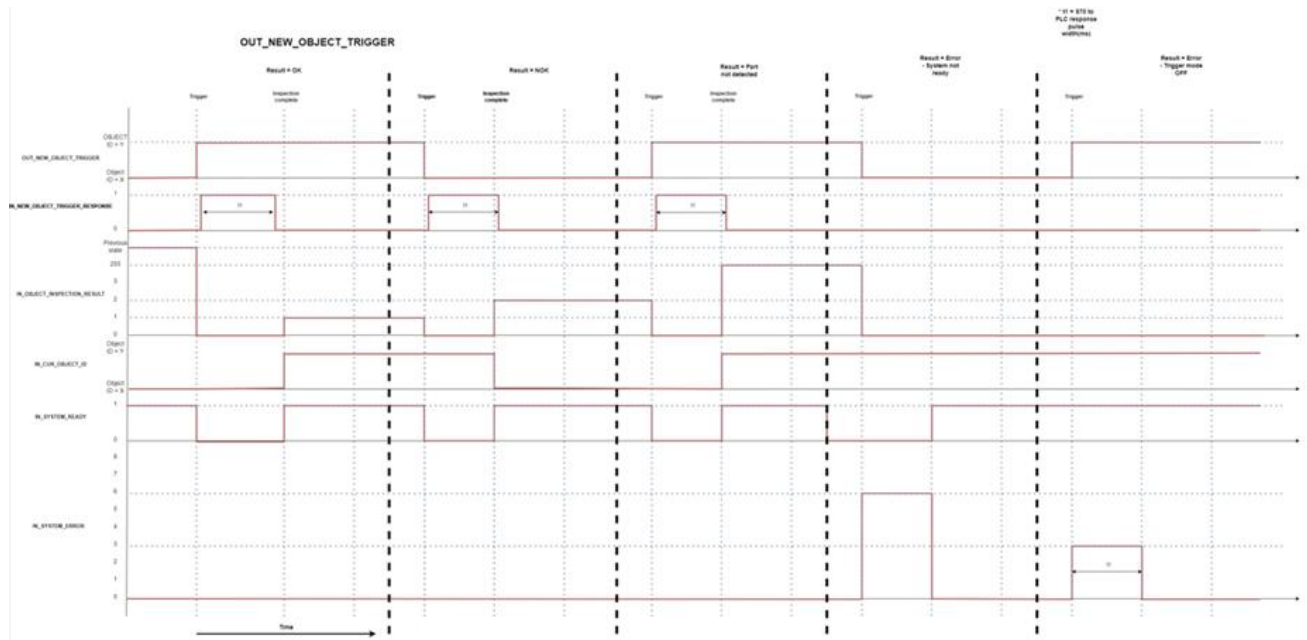


Figure 3-19

3.3.2. OUT_PROFILE_ID

This section demonstrates the signal behaviour when an OUT_PROFILE_ID command is received, and how the relevant signals going from the SIEMENS Inpekto VIS system to the PLC will behave.

In the chart below each section relates to a different scenario (zoom in in order to view with more details).

Please see " Inpekto functions and TIA Portal function blocks " section in this document for detailed explanation regarding this command.

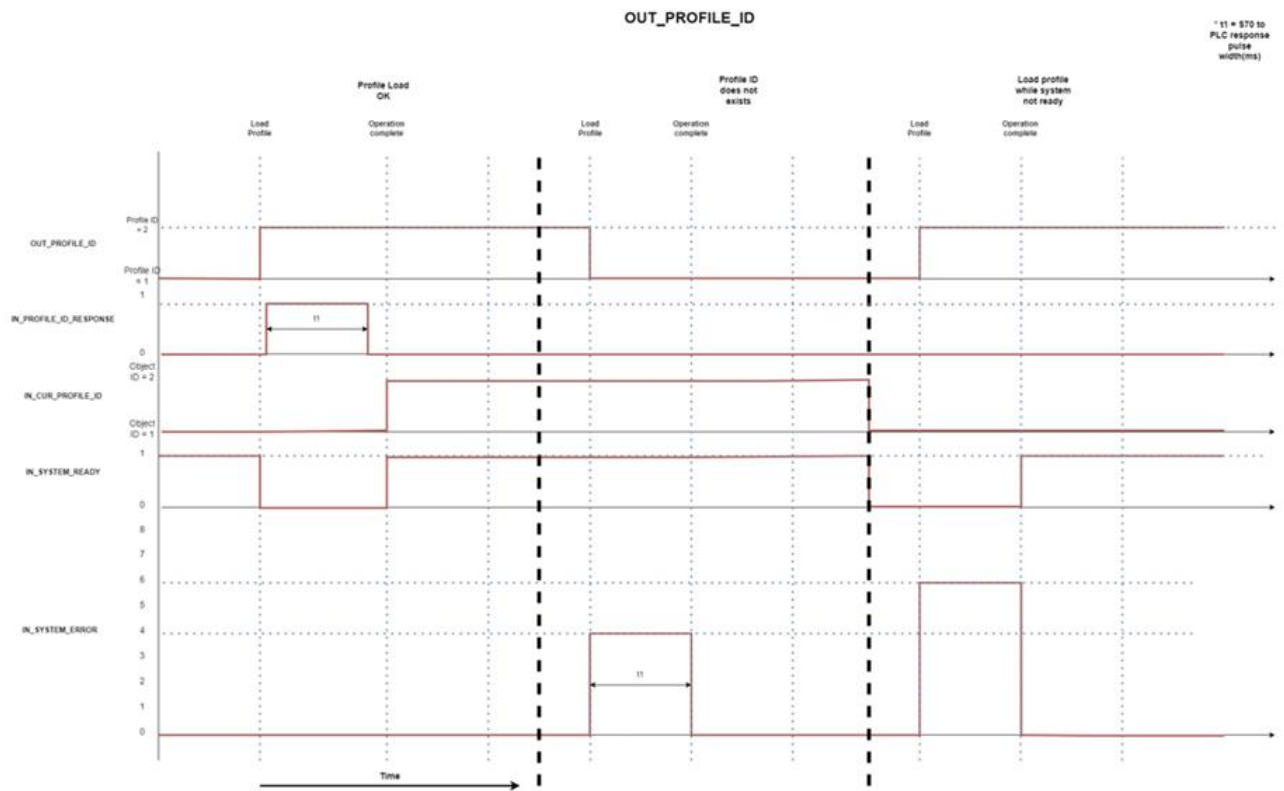


Figure 3-20

3.3.3. OUT_INSPECTION_MODE

This section demonstrates the signal behaviour when an OUT_INSPECTION_MODE command is received, and how the relevant signals going from the SIEMENS Inspekto VIS system to the PLC will behave.

In the chart below each section relates to a different scenario (zoom in in order to view with more details).

Please see " Inspekto functions and TIA Portal function blocks " section in this document for detailed explanation regarding this command.

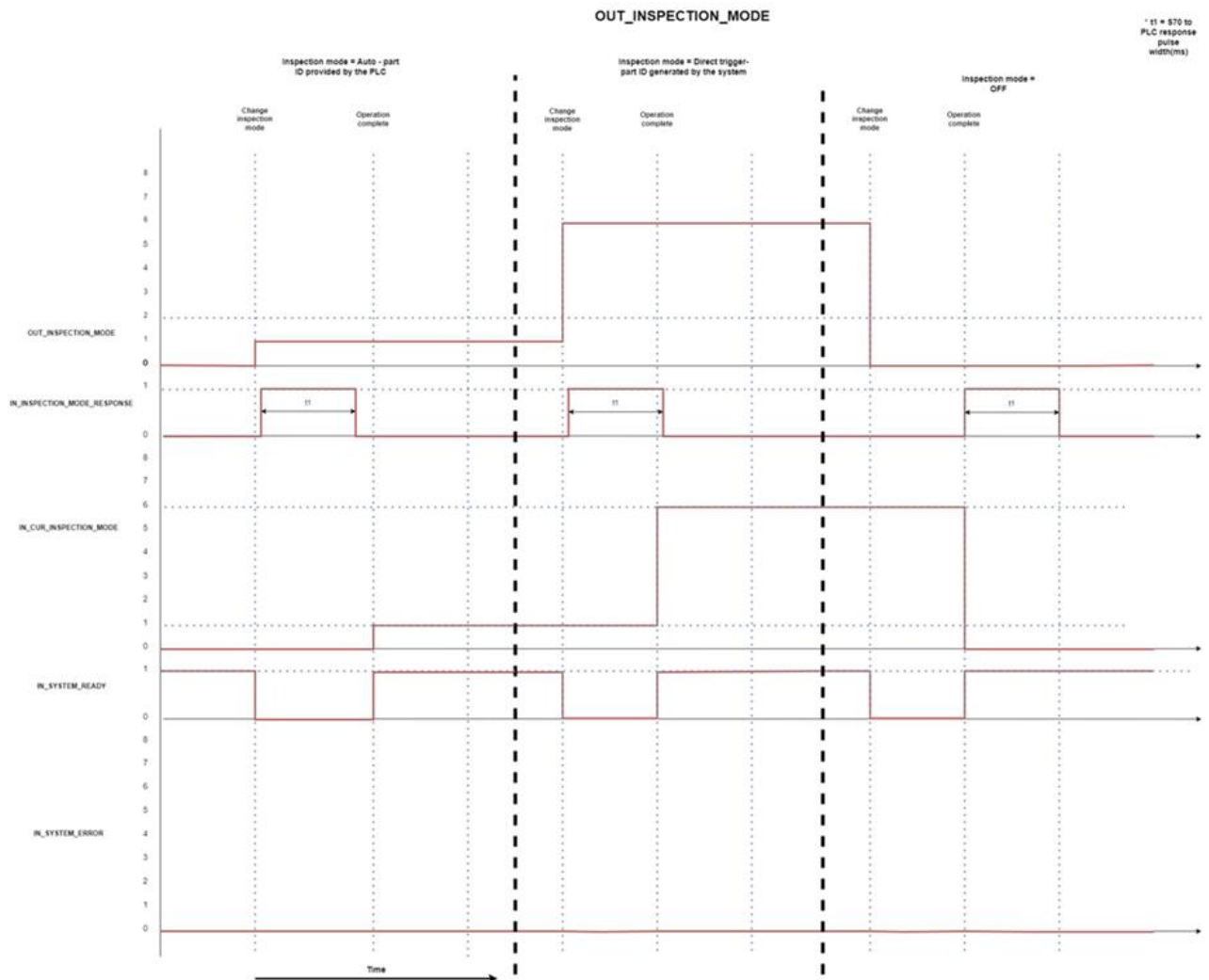


Figure 3-21

3.3.4. OUT_BATCH_ID

This section demonstrates the signal behaviour when an OUT_BATCH_ID command is received, and how the relevant signals going from the SIEMENS Inspekto VIS system to the PLC will behave.

Please see " Inspekto functions and TIA Portal function blocks " section in this document for detailed explanation regarding this command.

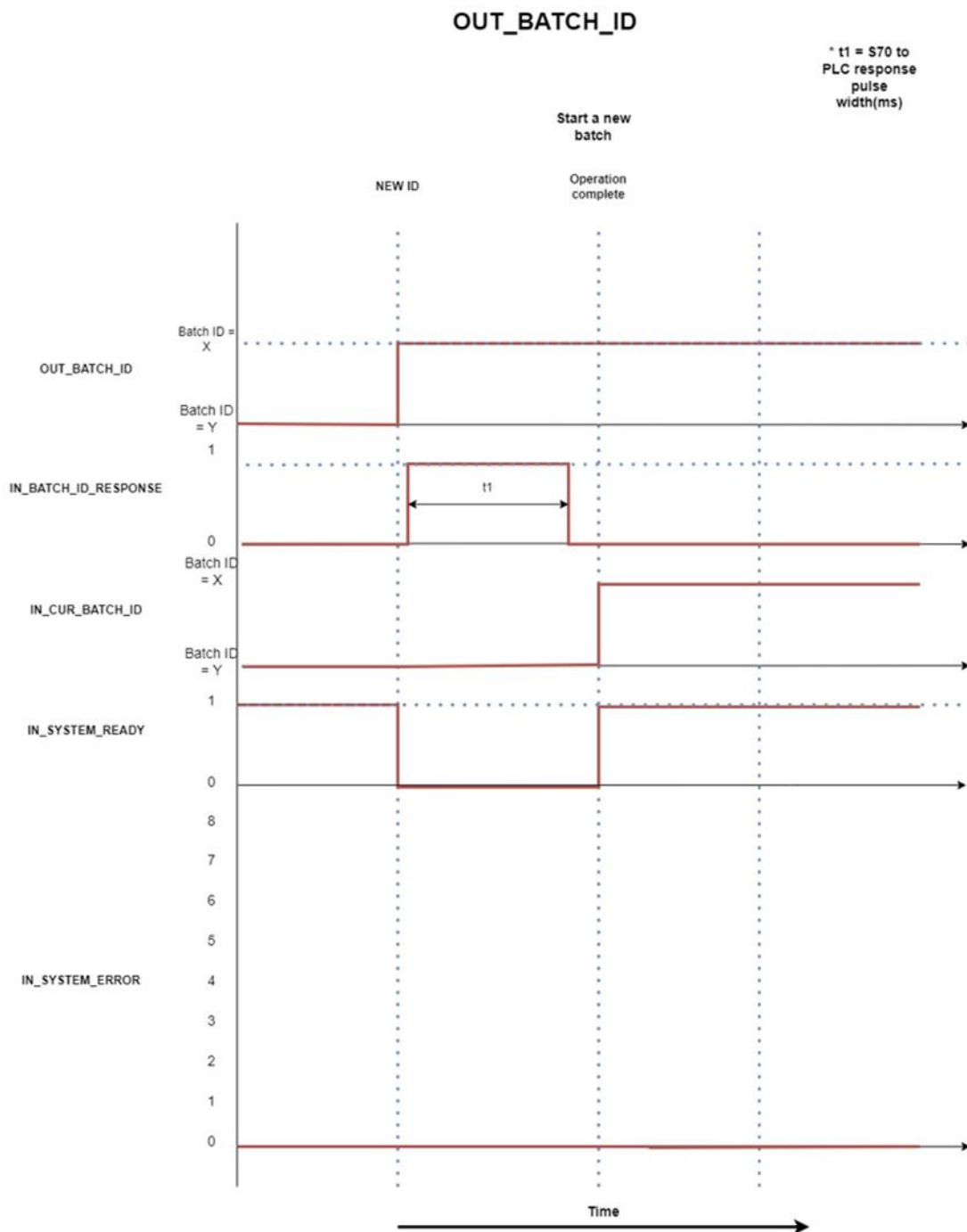


Figure 3-22

3.4. System Installation

Detailed information about the steps needed in order to install the SIEMENS Inspketo hardware are available under the following link.

- [SIEMENS Inspketo System Installation](#)

3.4.1. Updating TIA Portal integration to Inspketo v6.1

As already described in the initial sections of this document, in the new version v6.1 of Inspketo there is a need to install a new GSDML file and to update the device in TIA Portal to ensure the PLC to be able to communicate with the Inspketo system. In addition to that, in V6.1 there is an additional command allowing to shut down the Inspketo system using a command sent from the PLC.

This particular section describes the steps needed to be taken in order to update your already configured TIA Portal project so the PLC can still communicate with the Inspketo system after you updated it to v6.1.

3.4.2. System status after Inspketo update to version 6.1

After your Inspketo system has been updated to v6.1, given the changes introduced in relation to the PROFINET communication, the PLC will not be able to communicate with Inspketo using the old configuration and GSDML file.

When going online with the PLC and accessing the “Network view” in TIA Portal, Inspketo will be shown as unreachable.

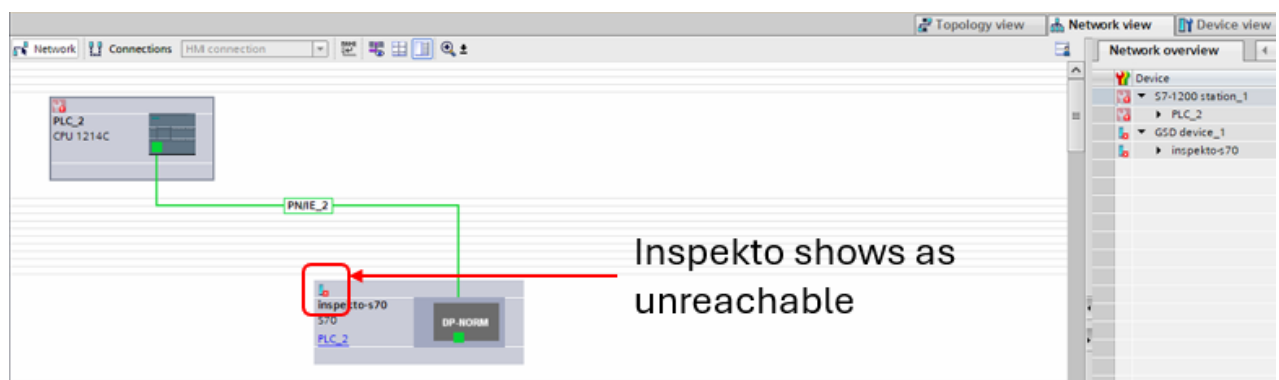


Figure 3-23

At this point, when right-clicking on the Inspketo device and selecting “Assign device name”, the Inspketo system will not be found if the option “Only show devices of the same type”.

If the “Only show devices of the same type” option is removed, the Inspketo system will become visible but registered as “Device type is different”.

To be able to restore communication between the PLC and the Inspketo system, the old GSDML file needs to be replaced with the new one that is compatible with version v6.1.

3.4.3. Installing GSDML compatible with version 6.1

To install the new GSDML file, proceed according to following instructions:

1. Download the Profinet Integration package that contains new GSDML file named GSDML-V2.45-INSPEKTOPND-VIS-V61-20250622.xml from the SiePortal entry of this [application example](#).
2. Install the downloaded GSDML file according to the steps described in the section [2.4.2 – TIA Portal Configuration: Install the GSDML file](#).

NOTE

Double check you are installing the correct GSDML file.

3.4.4. Replace Inspekto device

Once the new GSDML file has been installed, the old Inspekto device configured within the “Network” view of the menu “Device & Networks” needs to be replaced by the new Inspekto device.

1. Navigate to “Device & Networks -> Network view” and delete the old Inspekto device from the network configuration.

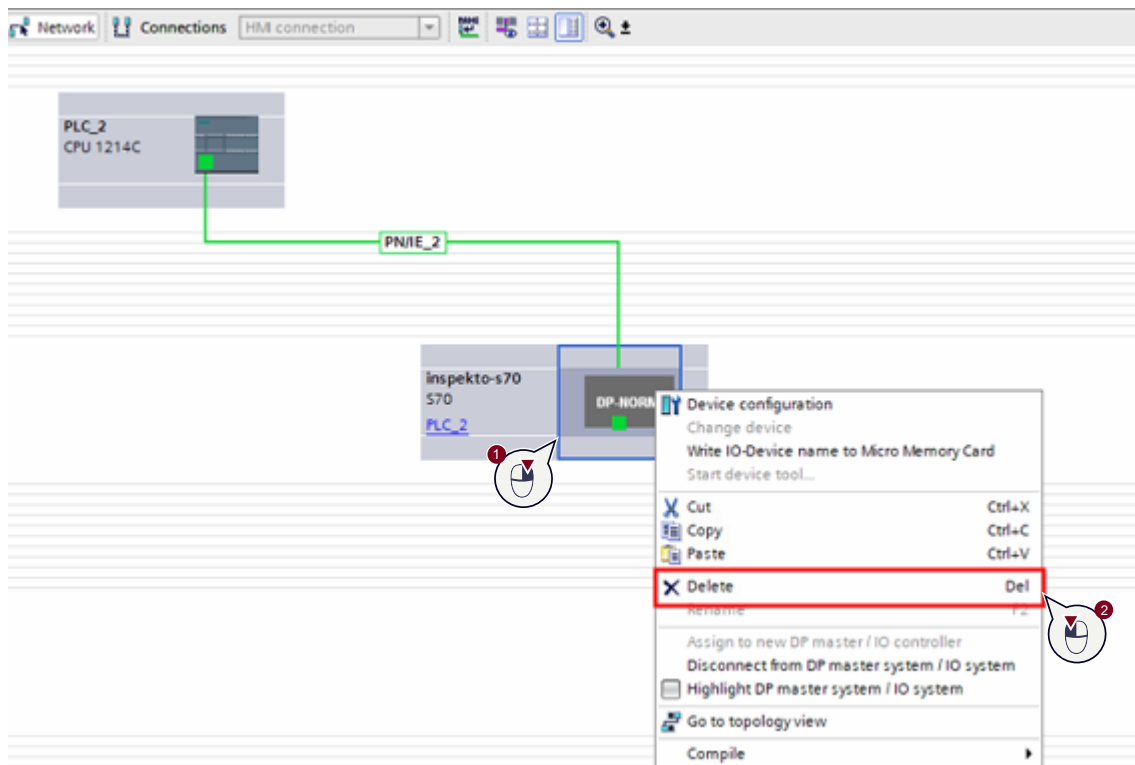


Figure 3-24

2. In the “Catalog” view, navigate to “Other field devices” -> “PROFINET IO” -> “Sensors” -> “SIEMENS AG” -> “INSPEKTO” -> “INSPEKTO VIS” and drag and drop the new Inspekto device to the “Network view” working area

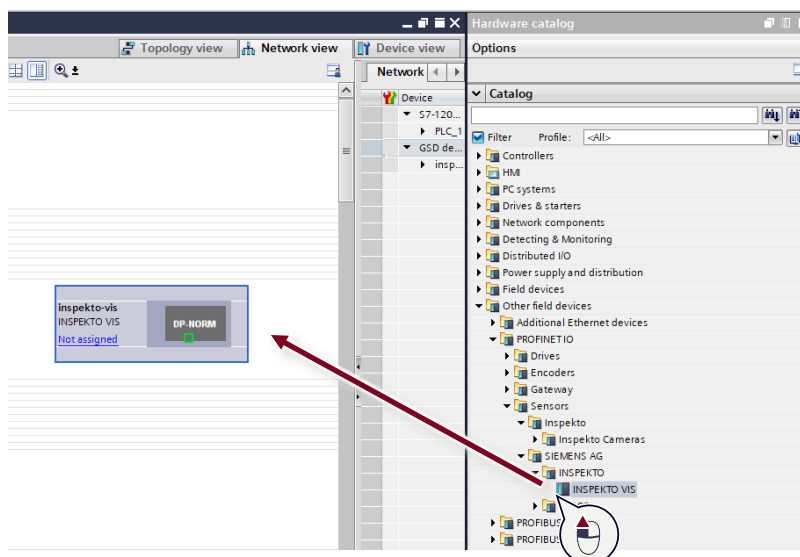


Figure 3-25

3. Assign the Inspekto device to your PLC to create the PROFINET network.

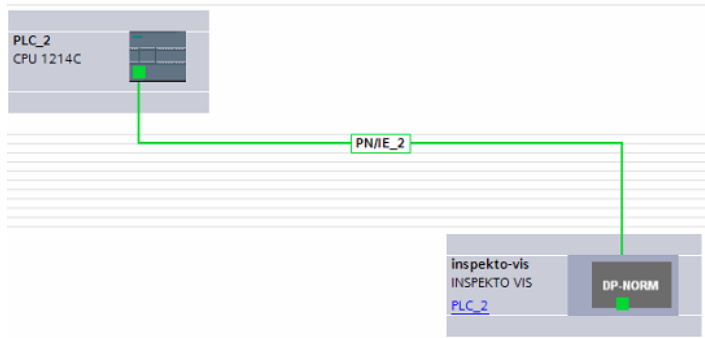


Figure 3-26

4. Double click on the Inspekto device and verify that I and Q address columns have still been assigned the same addresses as with the previous device.

NOTE

An additional OUT command and IN response for has been introduced in the new version to control the shutdown of the Inspekto system from the PLC. If you want to use them, you need to assign an Q and I address according to your PLC configuration.

When adjusting the addresses, if presented with the following pop-up window, select the option “Do not change tags”.

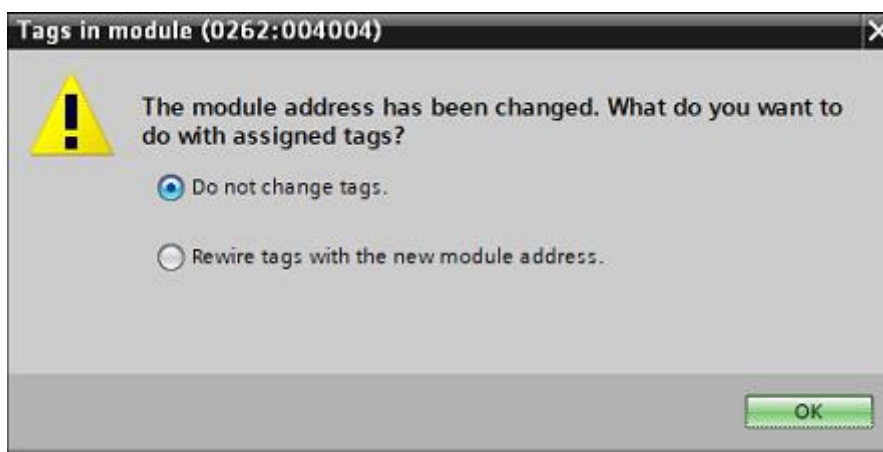


Figure 3-27

5. (Optional) Add PLC shutdown variables

If you want to use the PLC to shutdown the Inspekto system

- a. Add a Byte tag for OUT_PLC_SHUTDOWN command within Data_Out PLC data type
- b. Add a Byte tag for IN_PLC_SHUTDOWN command within Data_In PLC data type

The previously generated tags will be available within the Inspekto_Vision datablock after compiling your project.

6. Assign the PROFINET device name to the new Inspekto device following the instructions described in section [2.4.2 TIA Portal Configuration : Inspekto device configuration](#)
7. Compile (SW & HW) and download the project to the PLC.

4. Appendix

4.1. Service and support

SiePortal

The integrated platform for product selection, purchasing and support - and connection of Industry Mall and Online support. The SiePortal home page replaces the previous home pages of the Industry Mall and the Online Support Portal (SIOS) and combines them.

- **Products & Services**
In Products & Services, you can find all our offerings as previously available in Mall Catalog.
- **Support**
In Support, you can find all information helpful for resolving technical issues with our products.
- **mySieportal**
mySiePortal collects all your personal data and processes, from your account to current orders, service requests and more. You can only see the full range of functions here after you have logged in.

You can access SiePortal via this address: sieportal.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts.

Please send queries to Technical Support via Web form: support.industry.siemens.com/cs/my/src

SITRAIN – Digital Industry Academy

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page: siemens.com/sitrain

Industry Online Support app

You will receive optimum support wherever you are with the "Industry Online Support" app. The app is available for iOS and Android:



4.2. Application support

Siemens AG
Digital Factory Division
Factory Automation
SIMATIC System Support
DI FA S SUP
Gleiwitzer Str. 555
90475 Nürnberg, Germany

mailto: inspektosupport.industry@siemens.com

4.3. Links and literature

No.	Topic
-----	-------

\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link of this application example https://support.industry.siemens.com/cs/ww/en/view/109977467
\3\	

4.4. Documentation change

Version	Date	Modification
V1.3	01/2026	Included configuration steps for Inspekto v6.1
V1.2	05/2025	Included "Assign PROFINET" device name within configuration
V1.1	04/2025	Included description of assembly verification use case, trigger logics examples, inspection result visualization via HMI.
V1.0	12/2024	First version