EN

Industrial Ethernet (RTE)

For Machine Vision Devices

PROFINET®

EtherNet/IP™

EtherCAT®

Interface Protocol

Subject to change without notice
Available as PDF file only
Version: 1.6.0
Revision level: 19.02.2026
www.wenglor.com

Table of Contents

1. Use for Intended Purpose

Machine Vision Devices can communicate via various Industrial Ethernet (RTE) protocols to communication partners (e.g. PLCs) to exchange data.

Supported Machine Vision Devices:

- Smart Camera B60
- Machine Vision Controller MVC

Supported protocols:

- PROFINET
- EtherNet/IP
- EtherCAT

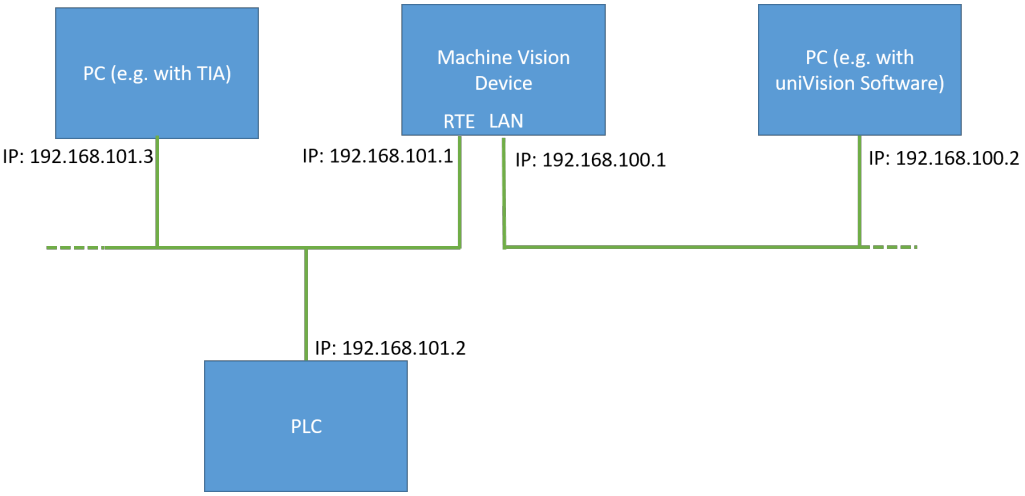
NOTE!
For details about Machine Vision Devices (e.g. conformance class), check the operating instructions of the device.

2. Network Overview

Connect the network cable from the Realtime Ethernet Connector (RTE) of the Machine Vision Device to the communication partner (e.g. PLC) or to a switch in order to work with several devices.

NOTE!

- For details about the Realtime Ethernet Connector (RTE) of the Machine Vision Device, check the operating instructions of the device.
- Do not use any hub in the PROFINET network. A switch may be used if both Priority Tagging and LLPD (100 Mbit/s, Full Duplex) are supported.
- Keep the networks for LAN and RTE separated in order to optimize the performance of the Machine Vision Device.
  - » RTE for communication with e.g. PLC, PC with PLC software and further RTE devices.
  - » LAN for communication with e.g. PC with uniVision software, webbased visualization, LIMA communication and process data via Device TCP or UDP.
- Use unique network configurations for the RTE ports at the Machine Vision Device within the RTE network. Furthermore, the network configuration of the RTE ports must be different to the network configuration of the LAN and CAM ports of the Machine Vision Device.
- Use the connector RTE1 or RTE2 if using PROFINET or EtherNet/IP at the Machine Vision Controller. Use RTE1 as input (and RTE2 only as output) in the perspective of the Machine Vision Device if using EtherCAT at the Machine Vision Controller.



Depending on the protocol, the LEDs at the Smart Camera B60 have different meanings. The following LEDs are not available at the Machine Vision Controller MVC.

PROFINET at Smart Camera B60

LED Labeling (Meaning)	Color	State	Meaning
RN (System error)	(Off)	Off	No error
	(Red)	Flashing (1 Hz, for 3 s)	DCP (discovery and configuration protocol) signal activated by the PLC to find the relevant PROFINET device.
	(Red)	On	System error
ER (Bus error)	(Off)	Off	No error
	(Red)	Flashing (2 Hz)	No data exchange
	(Red)	On	• No PROFINET configuration • Slow or missing physical connection
L/A	(Yellow)	Flashing	Machine Vision Device sends or receives Ethernet frames.
	(Off)	Off	No communication

EtherNet/IP at Smart Camera B60

LED Labeling (Meaning)	Color	State	Meaning
RN	(Green)	On	Ready for EtherNet/IP communication
	(Green)	Flashing	Standby (no configuration)
	(Red)	On	Fatal Error
	(Red)	Flashing	Error
	(Red/green)	Flashing	Self-test
	(Off)	Off	Switched off
ER	(Green)	On	Connected
	(Green)	Flashing	No connection, but valid IP address
	(Red)	On	Network conflict (another device in the network has the same network configuration).
	(Red)	Flashing	Connection time-out
	(Red/green)	Flashing	Self-test
	(Off)	Off	Switched off or no IP address
L/A	(Yellow)	Flashing	Machine Vision Device sends or receives Ethernet frames.
	(Off)	Off	No communication

EtherCAT at Smart Camera B60

LED Labeling (Meaning)	Color	State	Meaning
RN	(Off)	Off	INIT: Device is in state INIT
	(Green)	Blinking (2,5 Hz)	PRE-OPERATIONAL: The device is in state PRE-OPERATIONAL
	(Green)	Single flash	SAFE-OPERATIONAL: The device is in state SAFE-OPERATIONAL
	(Green)	On	OPERATIONAL: The device is in state OPERATIONAL
ER	(Off)	Off	No error
	(Red)	Blinking (2,5 Hz)	Invalid configuration
	(Red)	Single flash	Local error
	(Red)	Double flash	Application watchdog timeout

L/A	(Green)	On	Machine Vision Device is linked to Ethernet, but does not send/receive Ethernet frames.
	(Green)	Flashing	Machine Vision Device sends or receives Ethernet frames.
	(Off)	Off	The Machine Vision Device has no link to the Ethernet.

3. Input and Output Data

In the view of the PLC, the following input and output data is available:

- 8 Bytes Input: Device Status
- For each Processing Instance
  - » Inputs (Device to PLC)
    - 8 Bytes Input: Status of Processing Instance
    - x Bytes Input: User-Defined Process Data
  - » Outputs (PLC to Device)
    - 4 Bytes Output: Commands
    - x Bytes Output: User-Defined Process Data

NOTE!
The size of the User-Defined Process Data is depending on the configuration file. For details, see section “9. Attachments”.

3.1 Device Status

The Machine Vision Device sends the device status info with 8 bytes to the PLC. The following status bits are available (starting with bit and byte number 0). Not used bits are set to FALSE.

Name	Data Type	Bit Number	Byte.Bit	Description
High CPU load	BOOL	0	0.0	Is TRUE in case of high CPU load. Check the device website (Diagnosis).
High RAM	BOOL	1	0.1	Is TRUE in case of high RAM. Check the device website (Diagnosis).
High storage usage	BOOL	2	0.2	Is TRUE in case of high storage usage. Check the device website (Diagnosis).
High LAN Ethernet	BOOL	3	0.3	Is TRUE in case of high LAN Ethernet. Check the device website (Diagnosis).
High CAM Ethernet	BOOL	4	0.4	Is TRUE in case of high CAM Ethernet. Check the device website (Diagnosis). Only supported by Machine Vision Controller.
High temperature	BOOL	6	0.6	Is TRUE in case of high temperature. Check the device website (Diagnosis)

Too big changes in position	BOOL	7	0.7	Is TRUE in case of unwanted position change. Check the device website (→ Settings → Position Sensor). Not supported by all devices (check operating instructions of Machine Vision Device).
Too high acceleration	BOOL	10	1.2	Is TRUE in case of too big acceleration values. Ceck the device website (→ Settings → Position Sensor). Not supported by all devices (check operating instructions of Machine Vision Device).
CPU Throttling	BOOL	12	1.4	Is TRUE in case of CPU throttling. Check the device website (Diagnosis). Not supported by all devices (check operating instructions of Machine Vision Device).
Storage end of life	BOOL	13	1.5	Is TRUE in case of storage end of life. Check the hardware.
Empty battery	BOOL	14	1.6	Is TRUE in case of empty battery (only supported by the Machine Vision Controller MVC). Exchange the battery and reset the device status warning on the device website of the Machine Vision Device at the device status.
Autofocus module	BOOL	20	2.4	Is TRUE in case of any autofocus module error (e.g. blocked autofocus module at B60 Smart Camera). Check the hardware.
Crashed soft-ware service	BOOL	21	2.5	Is TRUE in case of crashed software services. Check device website for details.
Crashed RTU	BOOL	22	2.6	Is TRUE in case of crashed RTU at Smart Camera B60. Only supported by Smart Camera B60.
Uncalibrated analog output	BOOL	23	2.7	Is TRUE in case of uncalibrated analog output. Only supported by Smart Camera B60.
Distance Sensor	BOOL	24	3.0	Is TRUE in case of errors at the distance sensor (only supported by the Smart Camera B60). Check the hardware..
Localize (device is blinking)	BOOL	30	3.6	Is TRUE in case of localizing. Deactivate localize via button on Machine Vision Device or via software wenglor Discovery Tool. Not supported by all devices (check operating instructions of Machine Vision Device).

NOTE!
For details about the device status, check operating instructions of the Software wenglor Discovery Tool, the Machine Vision Device and the Software uniVision.

3.2 Status of Processing Instance

Each Processing Instance sends the status information with 8 bytes to the PLC. The following status bits are available (starting with bit and byte number 0). Not used bits are set to FALSE.

Bytes 0 and 1 provide the following status information:

Name	Data Type	Bit Number	Byte.Bit	Description
Operation Ready	BOOL	0	0.0	Is TRUE if Processing Instance is ready for operation. Possible reasons for operation not ready: • The Machine Vision Device has not yet fully booted or the start job is not yet fully loaded. • A job is currently being loaded in the Processing Instance. • The input device (camera) is not connected or available (only supported by Machine Vision Controller).
Acquisition Active	BOOL	1	0.1	Is TRUE when acquisition is active. Starting/ stopping acquisition is possible via the commands (see section "3.3 Commands").
Toggle Bit	BOOL	2	0.2	Changes between TRUE and FALSE every time a data evaluation is completed. It can be used to check for new results.
Processing	BOOL	3	0.3	Is TRUE when the Processing Instance is evaluating data. The status bit can be TRUE for a short time (in case of a job with small process time) or for a long time (in case of a job with big process time).
Command Acknowledge	BOOL	4	0.4	Is an echo signal of the command signal to verify that the command has been received.
Command Ready	BOOL	5	0.5	Is TRUE if the Processing Instance is ready to receive commands. **NOTE!** Commands to Processing Instances may only be sent if the command ready signal is TRUE. The status of the command ready signal must therefore be checked, before commands are sent.
Command Error	BOOL	6	0.6	Is TRUE if there was an error in the command. Possible reasons for command errors: • Several commands are sent at the same time • Command parameter contains invalid entry • Command load job fails because the job is not available

Byte 2 returns the current job number.

NOTE!
All uniVision jobs must be saved in the following format: “xxx_testjob.u3p” (x = any integer from 0 to 9). A maximum of 254 jobs can be used. The job number can be set between 1 and 254. Every uniVision job file needs an unambiguous number.

Bytes 4 to 7 provide the status of the Processing Instance (starting with bit number 0 and byte number 4). Not used bits are set to FALSE.

Name	Data Type	Bit Number	Byte.Bit	Description
WARNING	BOOL	0	4.0	Is TRUE in case of a warning. Check the reason of the warning
ERROR	BOOL	1	4.1	Is TRUE in case of an error. Check the reason of the error.
Overtrigger	BOOL	2	4.2	Is TRUE in case of overtrigger (trigger signal is sent during trigger prohibited time). It is only supported by the Smart Camera B60. Reduce the trigger frequency or decrease the exposure time.
Data overflow	BOOL	3	4.3	Is TRUE in case of data overflow. Reduce the acquisition frame rate or the processing time of the complete job to avoid data overflow.
Command overflow	BOOL	4	4.4	Is TRUE in case of command overflow. Reduce the frequency to send commands to the Processing Instances.
FTP interface	BOOL	9	5.1	Is TRUE in case of errors at the FTP interface. Check the FTP interface.
Unlicensed module(s)	BOOL	10	5.2	Is TRUE in case of unlicensed modules. Check the modules in the current job.
No connection to input device(s)	BOOL	19	6.3	Is TRUE in case of no connection to the input device at the Machine Vision Controller. Check the power supply and the Ethernet connection of the input device. Only supported at Machine Vision Controller.
Incompatible project	BOOL	20	6.4	Is TRUE in case of incompatible project. Convert the job or install a suitable firmware version.
Project not available	BOOL	21	6.5	Is TRUE in case of a not available job. Check the start job settings on the device website.
Processing	BOOL	22	6.6	Is TRUE in case of an error at processing. Check the job configuration.

Processing Instance initializing	BOOL	30	7.6	Is TRUE in case of initializing of the Processing Instance. Wait until the Processing Instance is created to work with it.
Acquisition Stopped	BOOL	31	7.7	Is TRUE in case of stopped acquisition. Check commands (see section "3.3 Commands") to start or stop the aquisition.

NOTE!

- Status signals of the Processing Instance can be TRUE for a very short time (e.g. if the job has a small process time, then the processing bit is TRUE for a short time). In order to receive all results at the PLC, the RTE cycle time may be maximum half the length of the shortest status signal. E.g. in order to detect a status signal of 2 ms, the RTE cycle time may be maximum 1 ms.
- Status signals of the Processing Instance are sent independently of the user-defined process data. So it can be that toggle bit and processing bit are updated at different RTE cycles than the user defined process data.
- For more details about the status of the Processing Instance, check the operating instructions of the uniVision software.

3.3 Commands

Commands are sent from the PLC to the Processing Instance on the Machine Vision Device. The commands consist of 4 bytes (starting with bit and byte number 0).

NOTE!
Commands to the Processing Instance may only be sent if the command ready signal and the operation ready signal are active. The status of both signals must therefore be checked before commands are sent.

Bytes 0 and 1 are used for the command bits.

Name	Data Type	Bit Number	Byte.Bit	Description
Load Job	BOOL	0	0.0	When the value is changed from FALSE to TRUE, the Processing Instance loads the job that is defined by the command parameter. **NOTE!** For loading jobs via RTE all jobs must be saved in the following format: "xxx_testjob.u3p" (x = any integer from 0 to 9). A maximum of 254 jobs can be used. The job number can be set between 1 and 254. Every uniVision job file needs an unambiguous number.

Name	Data Type	Bit Number	Byte.Bit	Description
Trigger	BOOL	1	0.1	When the value is changed from FALSE to TRUE, the Processing Instance sends a software trigger command to the Machine Vision Device.  **NOTE!** The Trigger Mode of the device must be set to ON and the Trigger Source must be set to Software.
Acquisition Start	BOOL	2	0.2	When the value is changed from FALSE to TRUE, the Processing Instance sends an acquisition start command to the Machine Vision Device.  **NOTE!** Only when acquisition is active (check status bit), the device is ready to receive trigger signals.
Acquisition Stop	BOOL	3	0.3	When the value is changed from FALSE to TRUE, the Processing Instance sends an acquisition stop command to the Machine Vision Device.  **NOTE!** When acquisition is inactive, the device is no longer ready to receive trigger signals.
Reboot	BOOL	4	0.4	When the value is changed from FALSE to TRUE, the Machine Vision Device performs a reboot.  **NOTE!** Reboot is finished after operating ready and command ready are TRUE again.

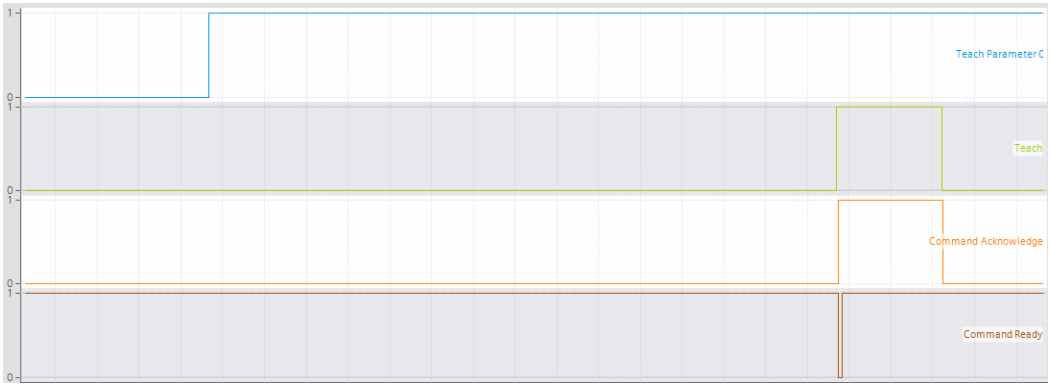
Name	Data Type	Bit Number	Byte.Bit	Description
Teach	BOOL	5	0.5	When the value is changed from FALSE to TRUE, the Processing Instance teaches or resets all values in the job tree linked to the teach parameters that are set to TRUE. Using the Teach command allows to teach or reset values in the job tree even if no data is processed in the Processing Instance (e.g. no trigger signals or acquisition stopped). **NOTE!** • Set the relevant teach parameters to TRUE and then activate the command bit “Teach”. • Make sure that within the uniVision job tree the relevant value (e.g. Reset at Module Counter) is linked to the relevant Teach Parameter (e.g. Data Value of Teach 0). For details, see section “4.2.3 Teach Parameters”. • Only needed if teaching or resetting of values is required without data evaluation - otherwise it is possible to use the user-defined process data.
Localize	BOOL	6	0.6	When the value is changed from FALSE to TRUE, the Machine Vision Device starts or stops localizing (if supported by device). For details, check the operating instructions of the Machine Vision Device and the uniVision software.
Teach Position Sensor	BOOL	7	0.7	When the value is changed from FALSE to TRUE, the Machine Vision Device teaches the current position as new reference and the current position of x and y returns 0° (if supported by device). For details, check the operating instructions of the Machine Vision Device.
Reserved	Byte	0-7	1.0-1.7	Not used

Byte 2 is used for the command parameter.

NOTE!
Make sure to set the command parameter (e.g. job number) first. Then send the command bit (e.g. to load the job).

Byte 3 is used for the teach parameters. 8 parameter bits are available for teaching or resetting of values.

NOTE!
Make sure to set the teach parameter bits first. Then send the teach command bit.

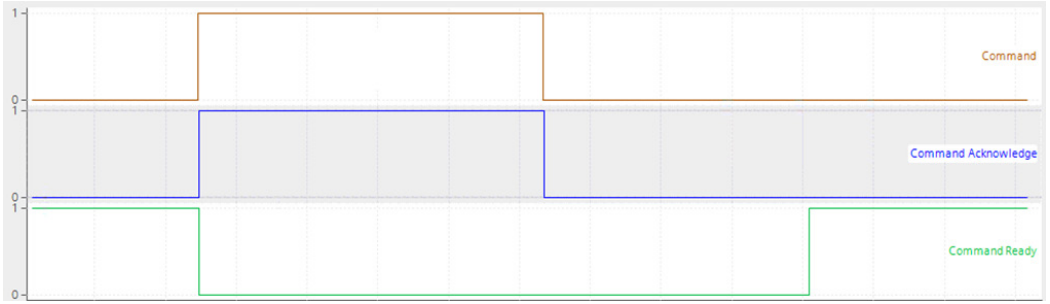


3.3.1 General Commands and Status Behavior

For each command that is sent from the PLC to the Machine Vision Device, an answer is sent back from the Machine Vision Device to the PLC via the status bits.

NOTE!
Commands to the Processing Instance may only be sent if command ready and operating ready signals are active. The status of both signals must therefore be checked before commands are sent.

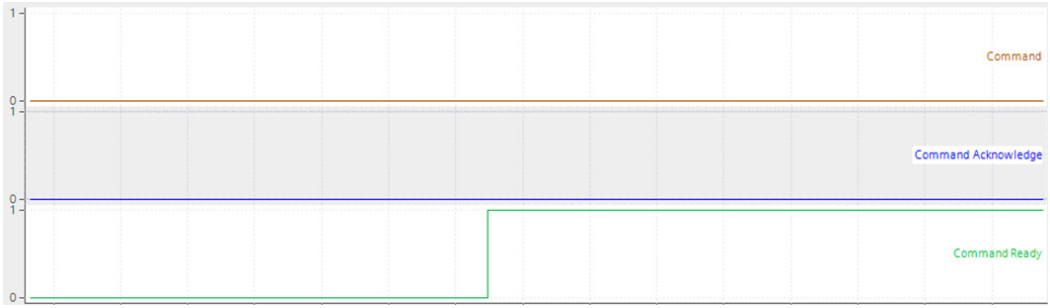
The following example of command and its status bits is valid for all types of commands.



- The command (e.g. trigger command) is sent from the PLC to the Processing Instance on the Machine Vision Device.
- The Processing Instance answers with the status bits after receiving the command:
  - » The command acknowledge signal switches from FALSE to TRUE (echo signal of the command).
  - » The command ready signal switches from TRUE to FALSE.

- When the command that is sent from the PLC to the Machine Vision Device is removed, the command acknowledge signal switches from TRUE to FALSE (echo signal of the command).

- When the execution of the command is finished, the command ready signal switches from FALSE to TRUE and the Processing Instance is ready to receive a new command.

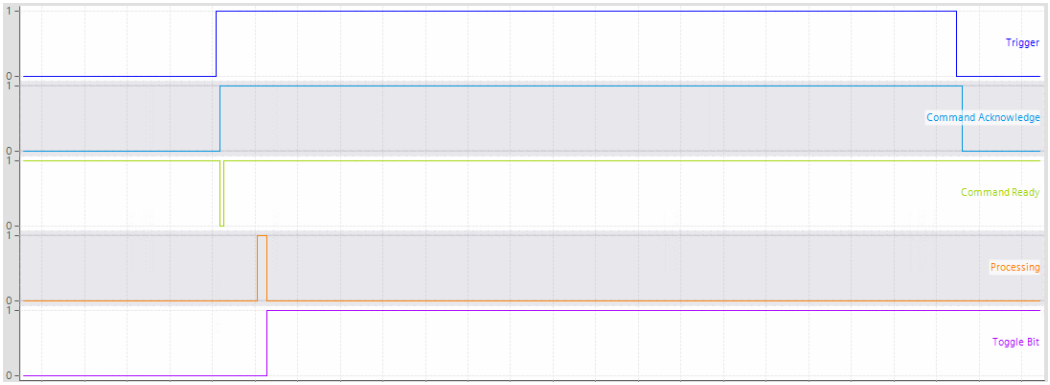


3.3.2 Trigger Command

When sending a trigger command from the PLC to the Processing Instance on the Machine Vision Device, new data is captured (in case of Trigger Source set to Software). While processing such data in the Processing Instance, the Processing bit in the status is active. After processing has been finished, the toggle bit in the status changes and the user-defined process data are available.

NOTE!

- After booting or job loading, a trigger command can be sent as soon as command ready and operating ready signals have switched from FALSE to TRUE.
- Use the toggle bit of the status to identify if the results that belong to the trigger signal are already available.

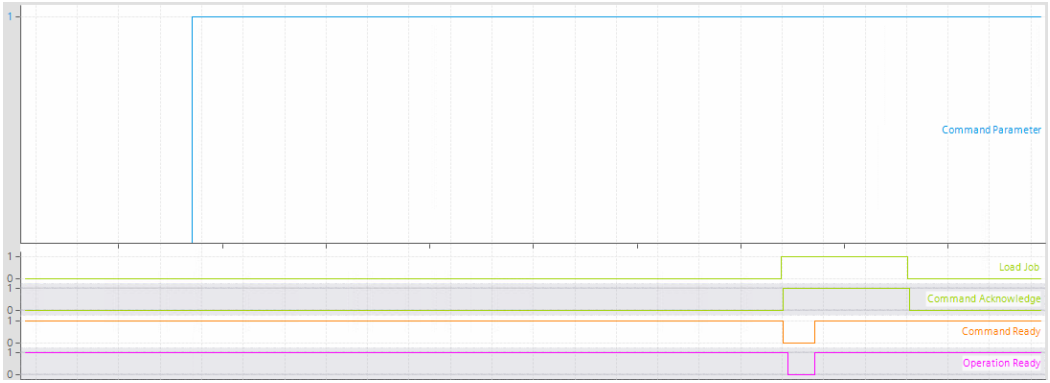


3.3.3 Load Job Command

When sending a load job command from the PLC to the Processing Instance on the Machine Vision Device, the Processing Instance loads the job defined by the command parameter. The number that is used in the job name must be sent by the command parameter.

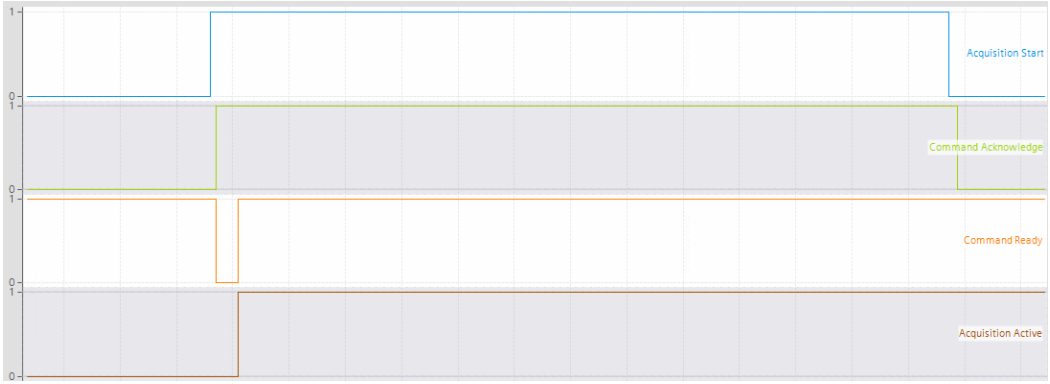
NOTE!

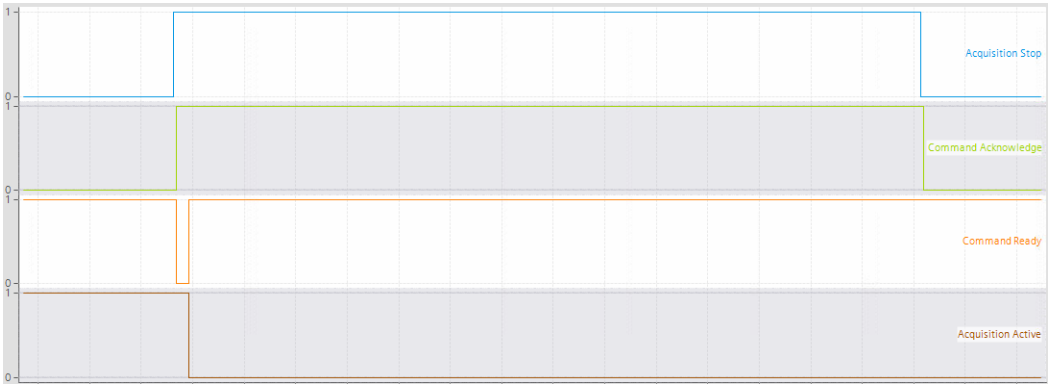
- For loading jobs via Industrial Ethernet all jobs must be saved in the following format: “xxx_testjob.u3p” (x = any integer from 0 to 9). For example 002_myjob.u3p.
- A maximum of 254 jobs can be loaded. The job number can be set between 1 and 254. Every uniVision job file needs an unambiguous number.
- Make sure to set the command parameter first. Only afterwards send the load job command bit.
- The job is loaded completely when the command ready signal has switched back to TRUE and when there was no command error. Hereafter, the next command (e.g. trigger command) can directly be sent to the Processing Instance.
- The job number is also sent back via the status (see section “3.2 Status of Processing Instance”).



3.3.4 Acquisition Start and Stop Command

When sending an acquisition start or stop command, the Processing Instance is ready or no longer ready to receive trigger signals. The trigger signals can be generated by the device itself (if Trigger Mode is set to Off) or via an external interface (e.g. software trigger via PROFINET if Trigger Mode is set to On and Trigger Source is set to Software).





3.4 User-Defined Process Data

User-defined process data are configured in the uniVision job. Process data can be sent from the Machine Vision Device to the PLC and reverse. For details see section “4.2 Device Industrial Ethernet”.

NOTE!
Compared to commands and status data that are updated continuously, process data is only evaluated and sent when data (e.g. image) is executed in the Processing Instance because of a trigger signal.
Use teach command and teach parameters to teach or reset values even if no data evaluation takes place (see section “3.3 Commands”).

NOTE!

- Make sure that the RTE cycle time is fast enough to receive all results. In order to receive all results at the PLC, the RTE sample frequency needs to be at least twice bigger than the uniVision processing frequency. E.g. for a uniVision processing frequency of 200 Hz, the RTE sample frequency needs to be minimum 400 Hz. Furthermore, the uniVision job processing time needs to be at least twice bigger than the RTE cycle time. E.g. for a RTE cycle time of 2 ms, the uniVision processing time needs to be minimum 4 ms.
- The user-defined process data of the Processing Instance are updated at the same time (same RTE cycle). Status signals of the Processing Instance are sent independently. So it can be that toggle bit and processing bit are updated at different RTE cycles than the user-defined process data.

4. Device Website and Software uniVision

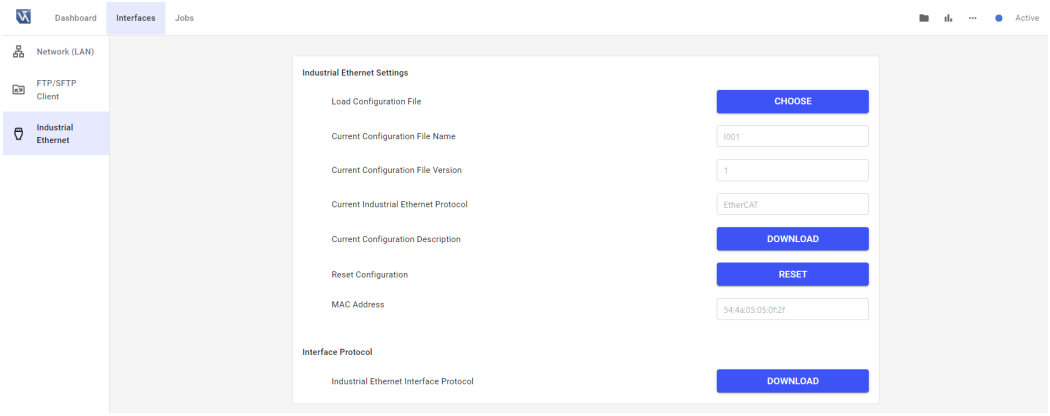
4.1 Installation of Configuration Files

Machine Vision Devices support several fix configuration layouts for the user-defined process data. The default configuration and further available configuration files are available in the attachments (see section “9. Attachments”).

NOTE!
Details about minimum firmware versions of the Machine Vision Device for the various Industrial Ethernet protocols are available in the operating instructions and the release notes of the device. In case of loading another configuration file, certain values are reset (e.g. the PROFINET name and the network configuration of the RTE network connector).

Loading another configuration file:

- Select suitable configuration file (see section “9. Attachments”)
- Download configuration file on the product detail page of the wenglor website (www.wenglor.com).
- Open device website in any supported browser via wenglor Discovery Tool or via entering directly the IP address of the Machine Vision Device.
- Select tab “Interfaces” and side navigation “Industrial Ethernet”.
- Press “Load Configuration File”, select the configuration file and confirm the reboot info.
- After the reboot, enter again the device website and check if the current configuration file name fits.



4.2 Device Industrial Ethernet

User-defined process data are defined within the uniVision job. Open the tab “Jobs” on the device website and click on “Open Job” in order to create a connection from the software wenglor uniVision 3 to the Processing Instance on the Machine Vision Device.

NOTE!
For details about the job configuration, check operating instructions of the uniVision software (DNNF023).

Add Device Industrial Ethernet from the toolbox to the uniVision job tree.

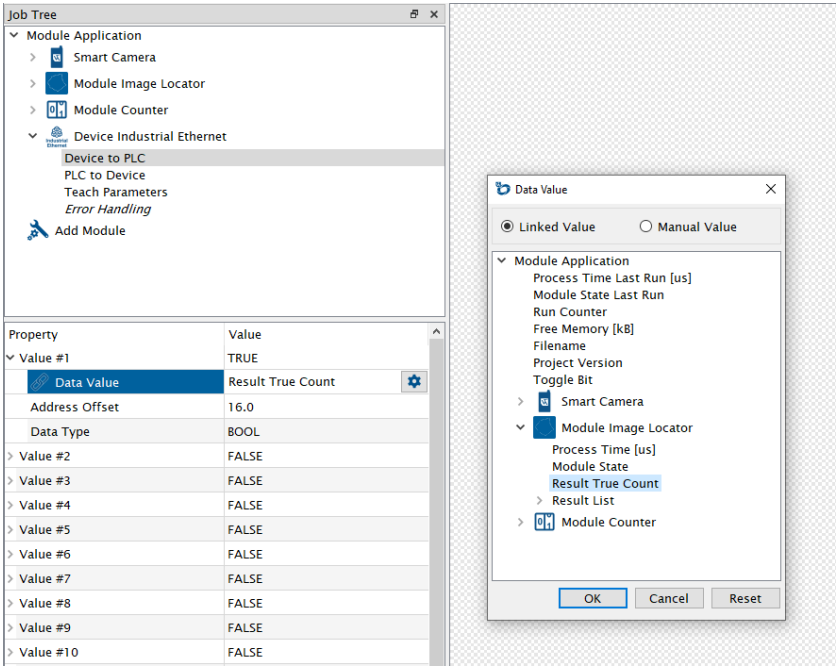
Process time [µs]	Process time to run the module in µs
Module State	Shows state of module: • 0: No error • Different to 0: Error (for error details see operating instructions of uniVision software).
Min Delay Next Run [µs]	Defines the minimum delay time in µs from the last run of the module to the current run. The execution of Device Industrial Ethernet is delayed if the minimum delay time is not reached in order to make sure that the PLC does not miss any result. By default, the value is set to 0 (does not apply any minimum delay time).
Error Handling	If any user-defined process data is in error state, it is substituted by a user-defined replacement value.

NOTE!
Compared to commands and status data that are updated continuously, process data is only evaluated and sent when data (e.g. image) is executed in the Processing Instance because of a trigger signal. Adding Device Industrial Ethernet to a job is only supported on a real device (not at offline jobs).

4.2.1 Device to PLC

Depending on the configuration file, the user-defined process data inputs appear.

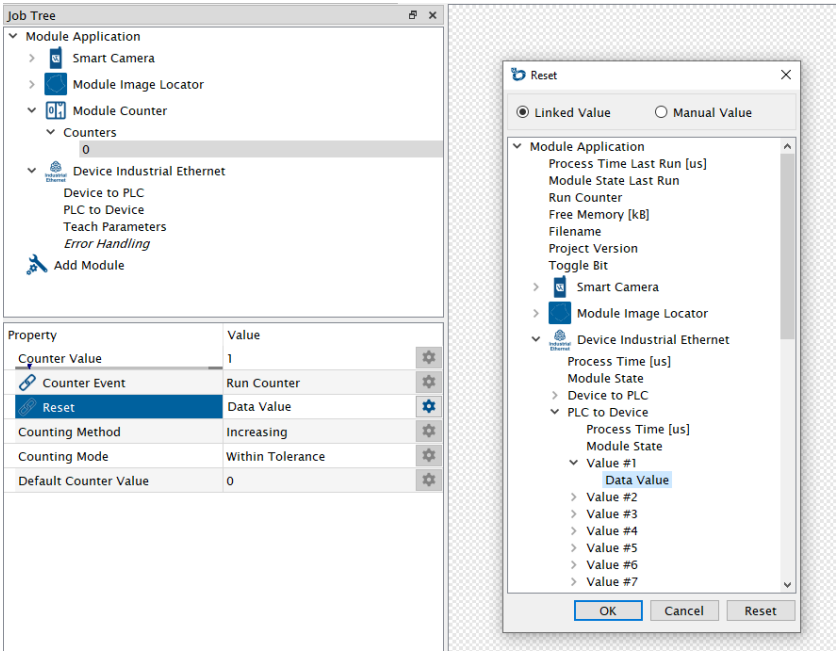
Value	Shows the result of the process data (uniVision job result).
Data Value	Set value manually or link it with any job result.
Address Offset	Shows the address offset for the value. **NOTE!** The address offset has to be added to the first input or output address that is used for the Machine Vision Device.
Data Type	Shows data type of the value.



4.2.2 PLC to Device

Depending on the configuration file, the user-defined process data output appear.

Value	Shows the result of the process data (uniVision input value).
Data Value	Shows the result of the process data (uniVision input value). Link Data Value of “PLC to Device” in other modules to evaluate it if data is processed in the Processing Instance because of a trigger signal. **NOTE!** Process data from the PLC to the Machine Vision Device is received when data is evaluated in the Processing Instance because of a trigger signal.
Address Offset	Shows the address offset for the value. **NOTE!** The address offset has to be added to the first input or output address that is used for the Machine Vision Device.
Data Type	Shows data type of the value.

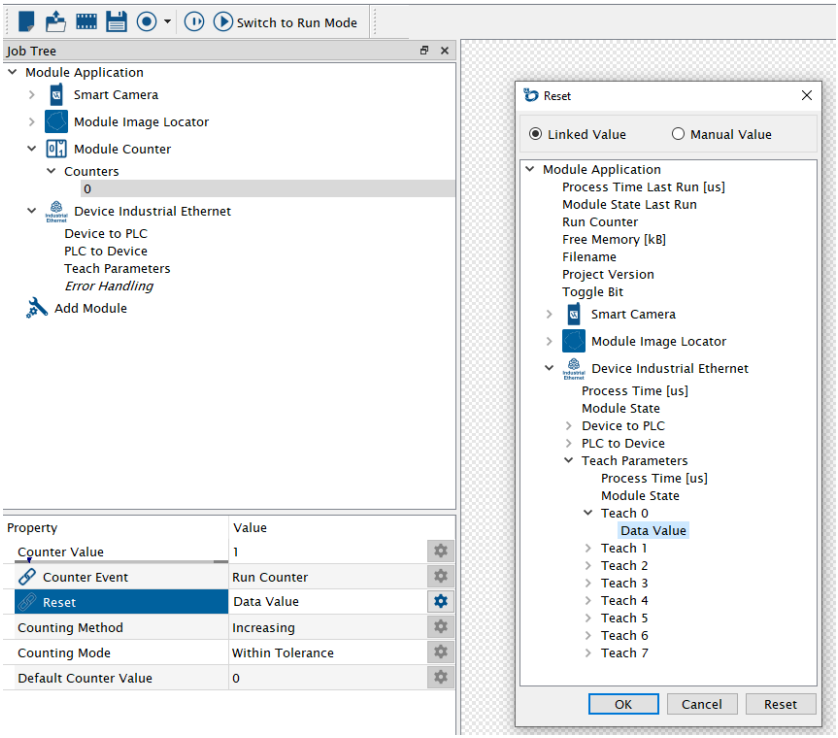


4.2.3 Teach Parameters

Teach parameter bits 0 to 7 appear as Sub-Module of Device Industrial Ethernet. Link Data Value of Teach Parameters in other modules (e.g. at value Reset of Module Counter) to teach or reset values in the job tree even if no data is processed in the Processing Instance (e.g. no trigger signal or acquisition stopped). Only needed if teaching or resetting of values is required without data evaluation – otherwise it is possible to use the user-defined process data.

NOTE!
To teach or reset, set the relevant teach parameter to TRUE and activate the teach command bit at the PLC (see section “3.3 Commands”).

Teach	Shows the result of the teach parameter (uniVision input value).
Data Value	Shows the result of the teach parameter (uniVision input value).
Address Offset	Shows the address offset for the value. **NOTE!** The address offset has to be added to the first PLC output address that is used for the Machine Vision Device.
Data Type	Shows the data type of the value



4.2.4 Error Handling

If any process data is in error state, the substitution value can be set for each data type.

Substitute BOOL types by	If a BOOL value used in Device Industrial Ethernet is in error state, it is replaced by FALSE or TRUE (Default: FALSE).
Substitute INT types by	If a INT value used in Device Industrial Ethernet is in error state, it is replaced by any user-defined INT (Default: 0).
Substitute DOUBLE types by	If a DOUBLE value used in Device Industrial Ethernet is in error state, it is replaced by any user-defined DOUBLE value (Default: 0.0000).
Substitute STRING types by	If a STRING value used in Device Industrial Ethernet is in error state, it is replaced by any user-defined STRING value (Default: Error###).

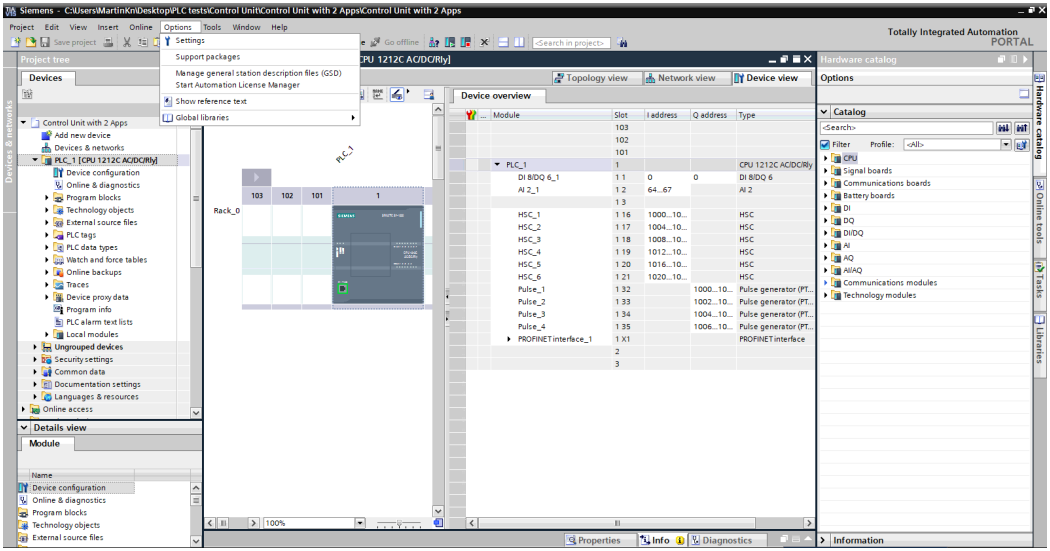
5. PROFINET PLC Settings

The integration for the Machine Vision Device B60 Smart Camera is shown with a Siemens S7-1200 PLC with TIA Portal V15. It works identically for other Machine Vision Devices (e.g. for the Machine Vision Controller).

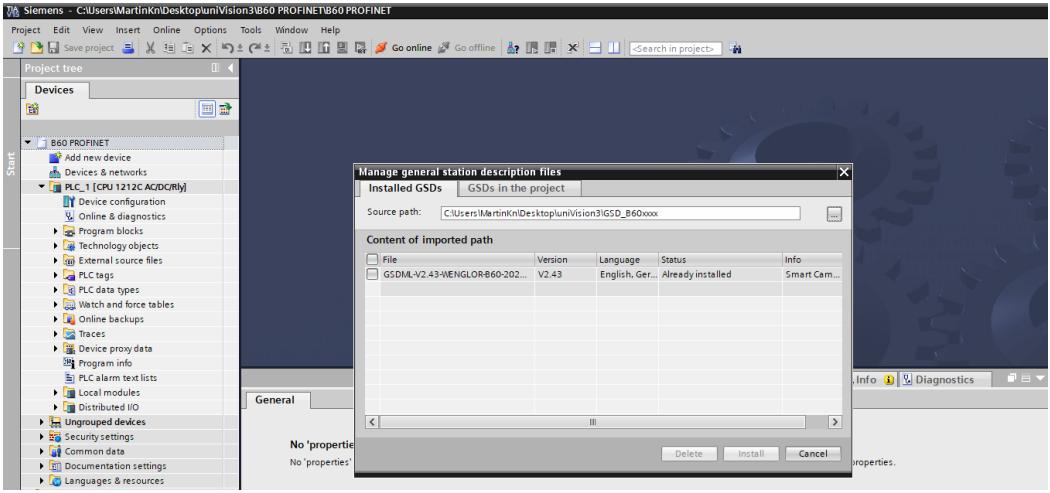
5.1 GSDML File

The GSDML file is available on the wenglor website in the download section of the Machine Vision Device (www.wenglor.com). Download the GSDML file, unzip the file and install it on the PLC.

In the software TIA Portal V15, the GSDML file is added via “Options” → “Manage general station description files (GSD)”.



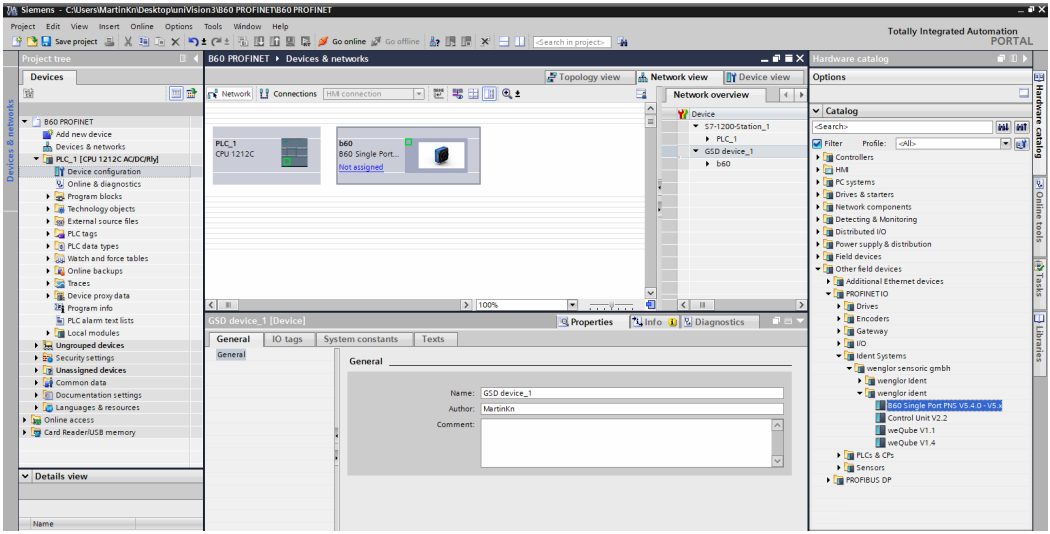
Adjust the correct source path if necessary, select the file and click on “Install”. After the successful installation, the status signals “already installed”.



5.2 Add Machine Vision Device to PLC Network

Search in the hardware catalog for “Other field devices” → “PROFINET IO” → “Ident Systems”. Select “wenglor sensoric gmbh”, “wenglor ident” and add the relevant device to your network.

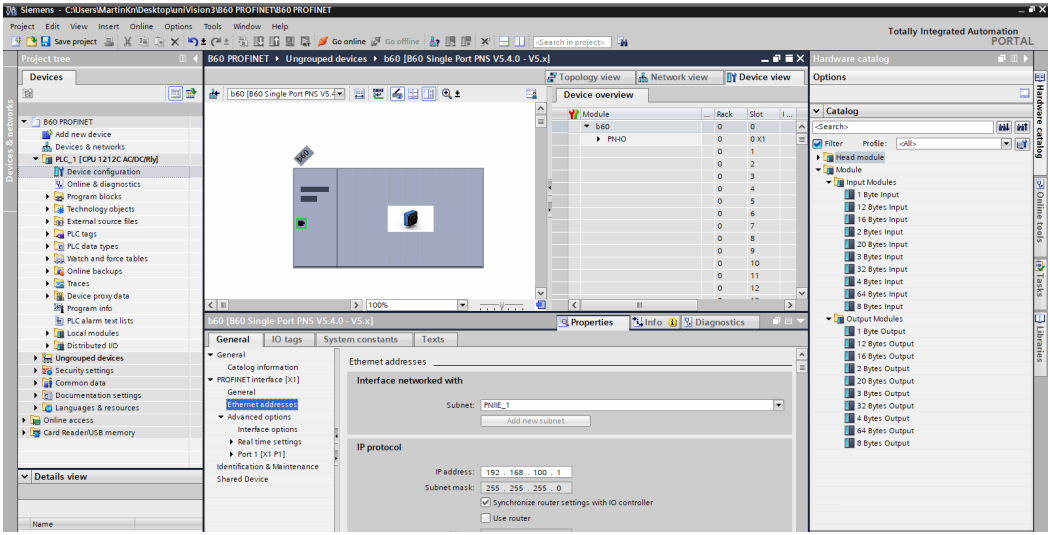
Then connect the Machine Vision Device with the PLC in the network view.



5.3 PROFINET Network Configuration

Switch to the device view of the Machine Vision Device and open the properties. Then setup the network configuration for the PROFINET interface of the Machine Vision Device and select a device name.

NOTE!
Make sure that the PLC cycle time and the real time settings of the device are fast enough to receive all results at the PLC.



If working with e.g. two Processing Instances on the Machine Vision Controller with the configuration file E002, the following slot order must be used:

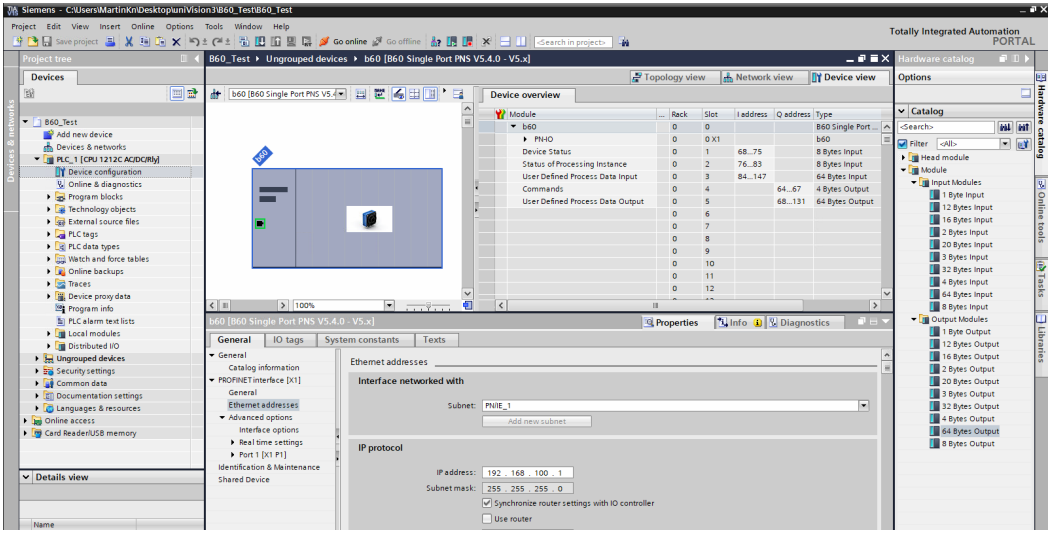
- 8 Bytes Input (Device Status)
- 8 Bytes Input (Status of Processing Instance 1)
- 64 Bytes Input (User-Defined Process Data Input of Processing Instance 1)
- 4 Bytes Output (Commands of Processing Instance 1)
- 64 Bytes Output (User-Defined Process Data Output of Processing Instance 1)
- 8 Bytes Input (Status of Processing Instance 2)
- 64 Bytes Input (User-Defined Process Data Input of Processing Instance 2)
- 4 Bytes Output (Commands of Processing Instance 2)
- 64 Bytes Output (User-Defined Process Data Output of Processing Instance 2)

5.4 Configure Input and Output Data

Add the input and output slots according to the configuration file used on the Machine Vision Device.

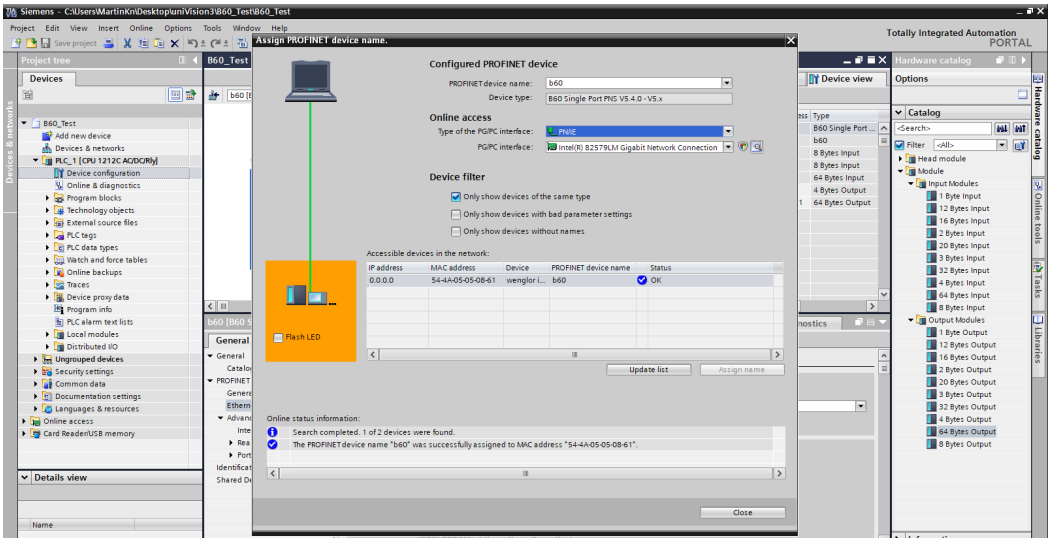
NOTE!
Input and output slots must be added consecutively (without gaps and in the right order) in order to use the address offsets provided in the uniVision job.

The following example shows the default slot configuration for the Smart Camera B60 (H001).

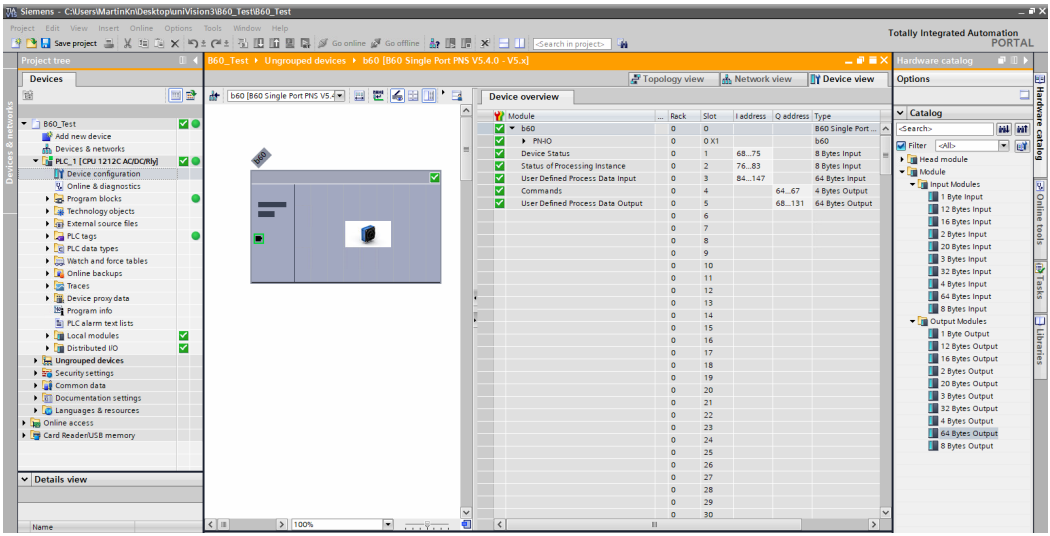


5.5 Download Configuration to PLC

Compile and download the configuration to the PLC. Then assign the network configuration and the device name to the Machine Vision Device via the context menu.
Click on “Update list” to see all Machine Vision Devices in the network. Select the available Machine Vision Device, assign the name and close the window.



Click on “Go online”. Check the status in TIA portal to analyse if the configuration of the PLC and the Machine Vision Device fit together.



5.6 PLC Tags

Select PLC tags to add the input and output data to your PLC project. Use the address offsets and the data types provided within the uniVision job for the process data.

NOTE!
Data types and address offsets are available for all user-defined process data. The address offset must be added to the starting address of the first input or output address of the Machine Vision Device.

The following example shows status, commands and user-defined process data with the default configuration of the Smart Camera B60 (H001).

	Name	Address	Display format	Monitor value
1	"Device Status"	%IB68	Hex	16#00
2	"Device Status(1)"	%IB69	Hex	16#00
3	"Device Status(2)"	%IB70	Hex	16#00
4	"Device Status(3)"	%IB71	Hex	16#00
5	"Device Status(4)"	%IB72	Hex	16#00
6	"Device Status(5)"	%IB73	Hex	16#00
7	"Device Status(6)"	%IB74	Hex	16#00
8	"Device Status(7)"	%IB75	Hex	16#00
9	"Operation Ready"	%I76.0	Bool	TRUE
10	"Acquisition Active"	%I76.1	Bool	FALSE
11	"Toggle Bit"	%I76.2	Bool	FALSE
12	"Processing"	%I76.3	Bool	FALSE
13	"Command Acknowledge"	%I76.4	Bool	FALSE
14	"Command Ready"	%I76.5	Bool	TRUE
15	"Command Error"	%I76.6	Bool	FALSE
16	"Job Number"	%IB78	Hex	16#01
17	"Status Processing Instance"	%IB80	Hex	16#00
18	"Status Processing Instance(1)"	%IB81	Hex	16#00
19	"Status Processing Instance(2)"	%IB82	Hex	16#00
20	"Status Processing Instance(3)"	%IB83	Hex	16#00
21	"User Defined Process Data Input BOOL 0"	%I84.0	Bool	FALSE
22	"User Defined Process Data Input DINT 0"	%ID88	DEC+/-	0
23	"User Defined Process Data Input REAL 0"	%ID100	Floating-point nu...	0.0
24	"User Defined Process Data Input CHAR 0"	%IB116	Character	'$00'
25	"Load Job"	%Q64.0	Bool	FALSE
26	"Trigger"	%Q64.1	Bool	FALSE
27	"Acquisition Start"	%Q64.2	Bool	FALSE
28	"Acquisition Stop"	%Q64.3	Bool	FALSE
29	"Reboot"	%Q64.4	Bool	FALSE
30	"Teach"	%Q64.5	Bool	FALSE
31	"Command Parameter"	%QB66	Hex	16#00
32	"Teach Parameter 0"	%Q67.0	Bool	FALSE
33	"User Defined Process Data Output BOOL 0	%Q68.0	Bool	FALSE
34	"User Defined Process Data Output DINT 0"	%QD72	DEC+/-	0
35	"User Defined Process Data Output REAL 0"	%QD84	Floating-point nu...	0.0
36	"User Defined Process Data Output CHAR 0"	%QB100	Character	'$00'

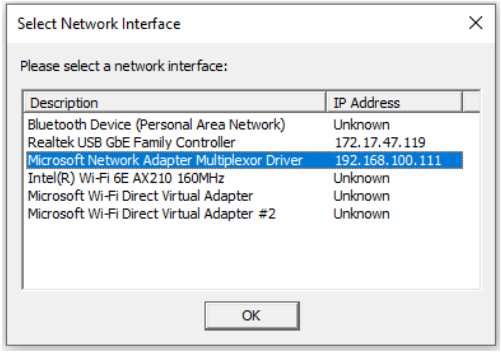
6. EtherNet/IP PLC Settings

The integration for the Machine Vision Device B60 Smart Camera is shown with an Allen-Bradley PLC 1769-L18ERM-BB1B with Studio 5000 Logix Designer V32. It works identically for other Machine Vision Devices (e.g. for the Machine Vision Controller).

6.1 RTE Network Configuration

By default, the RTE network is set to DHCP if loading an EtherNet/IP configuration file. Use a DHCP server (e.g. BOOTP) to allocate network settings to the RTE connector of the Machine Vision Device.

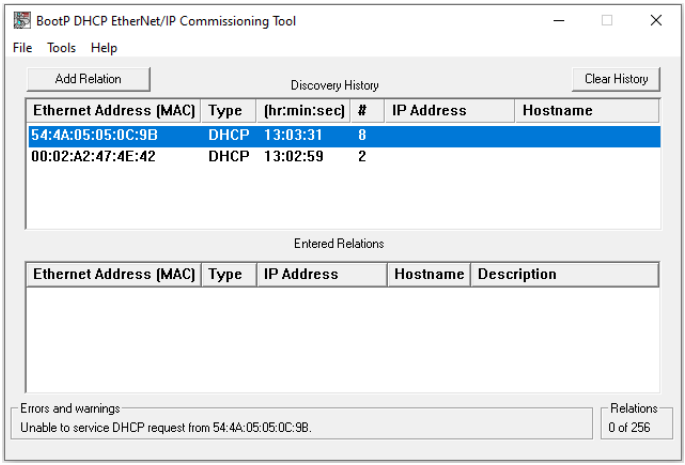
Open BOOTP and select the LAN adapter of the PC.



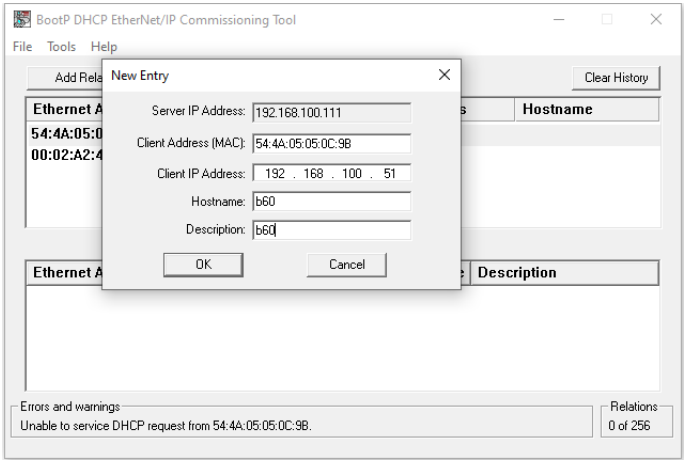
BOOTP lists all devices with activated DHCP settings within the network.

NOTE!

Open the device website (tab “Interfaces” → “Industrial Ethernet”) to see the MAC address of the RTE connector. It allows to identify the relevant device in case of several devices within the network (see section “4.1 Installation of Configuration Files”).

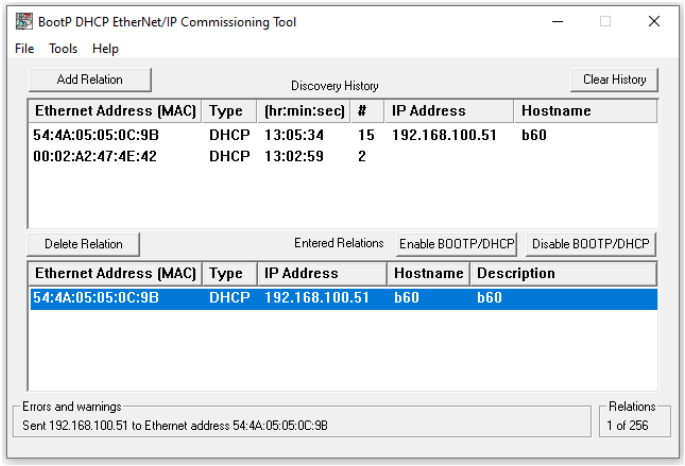


Select the Machine Vision Device, click on “Add Relation” and enter the IP address and the hostname.



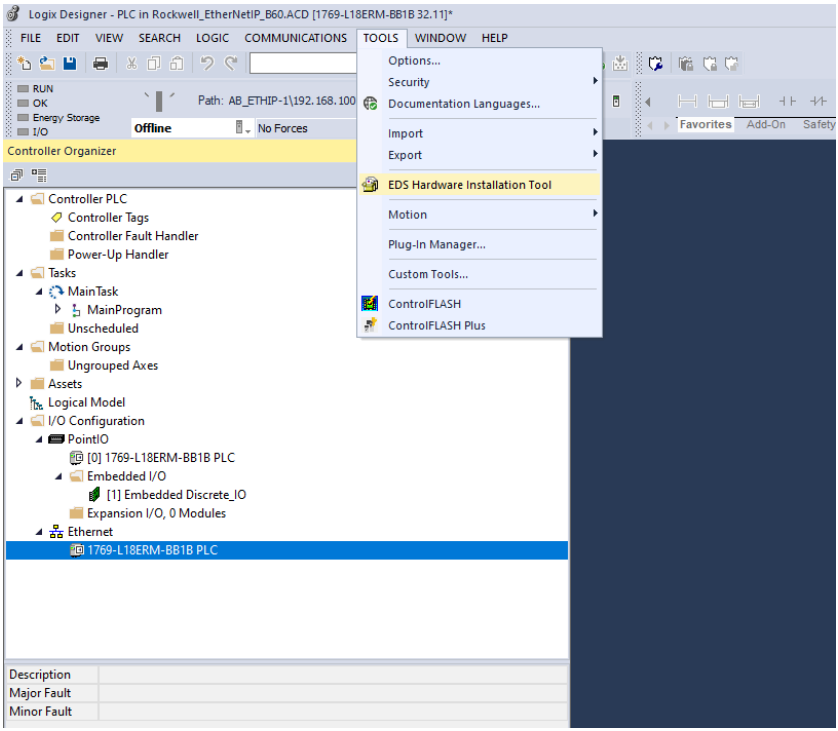
After a manual reboot of the Machine Vision Device, the network configuration is sent to the device.
In order to keep the RTE network settings for the next startup of the Machine Vision Device, select the device and click on “Disable BOOTP/DHCP”.

NOTE!
Enabling or disabling BOOTP/DHCP only works if the PLC is not connected to the Machine Vision Device.

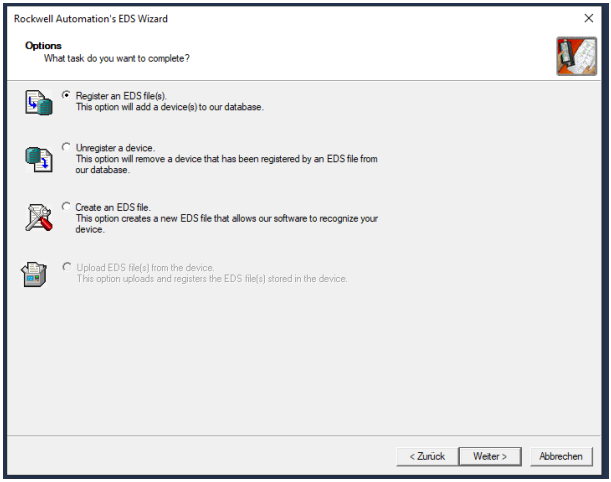


6.2 EDS File

The EDS file is available on the wenglor website in the download section of the Machine Vision Device (www.wenglor.com). Download the EDS file, unzip it and install it on the PLC.
In the software Studio 5000 Logix Designer, install the EDS file via “TOOLS” → “EDS Hardware Installation Tool”.



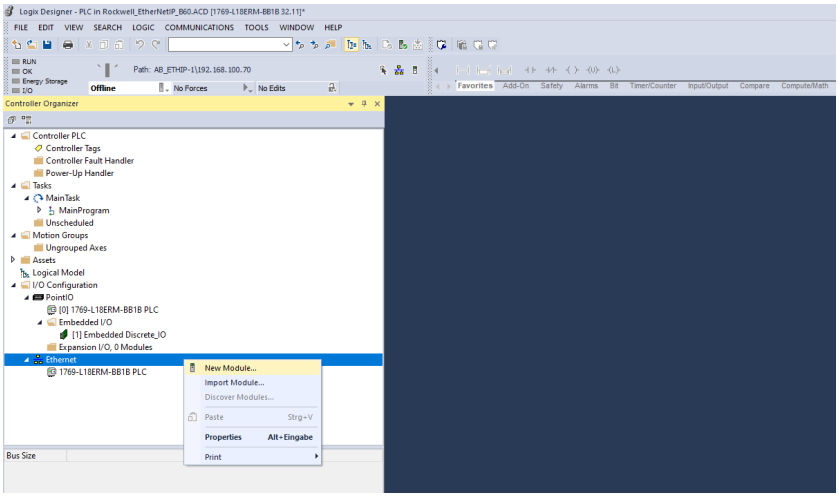
Select to “Register an EDS file(s)”.



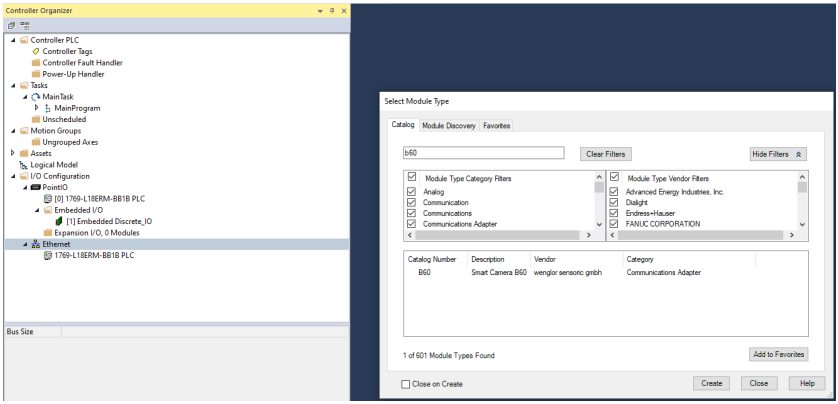
Select the path of the EDS file in order to install it.

6.3 Add Machine Vision Device to PLC Network

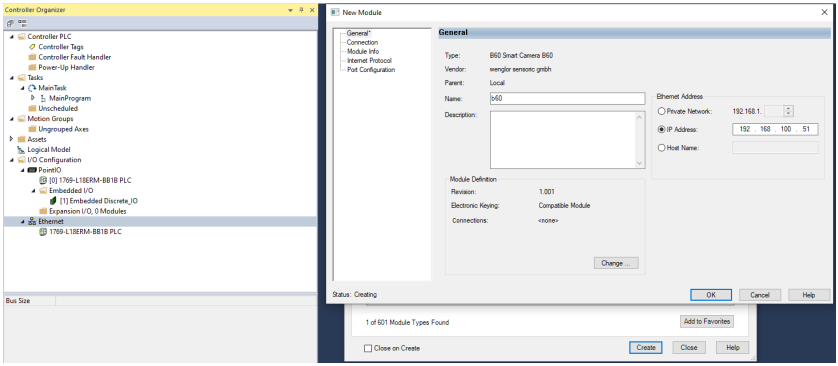
Open the context menu at “Ethernet” with a right click and select “New Module...” to add the Machine Vision Device to the PLC network.



Then search for the Machine Vision Device in the catalog. Select it and click on “Create”.



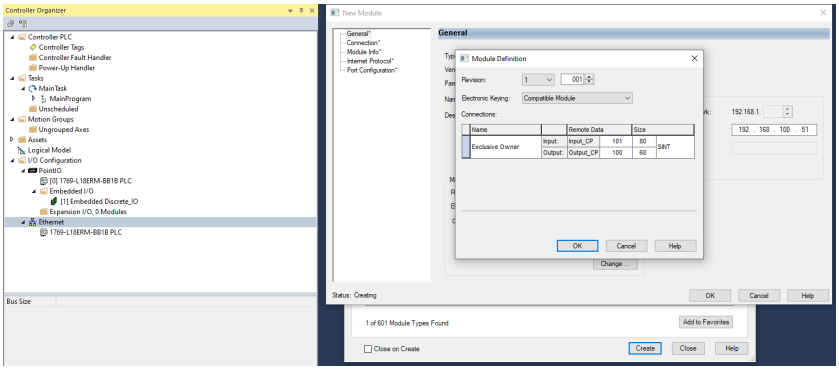
Enter the device name and the network configuration of the RTE connector of the Machine Vision Device (see section “6.1 RTE Network Configuration”). Make sure that the RTE cycle time (via Requested Packet Interval RPI) is fast enough to receive all results at the PLC.



6.4 Configure Input and Output Data

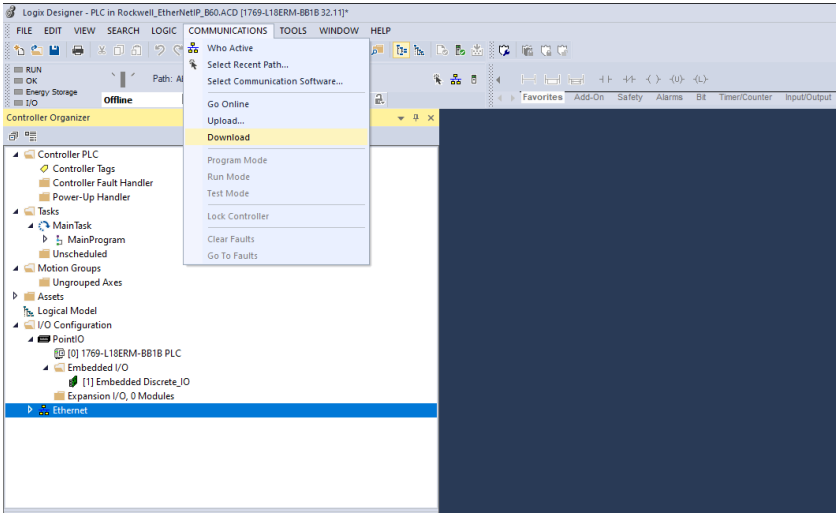
Click on “Change” in order to setup input and output data.

Adjust the input and output size according to the configuration file (see section “9. Attachments”).

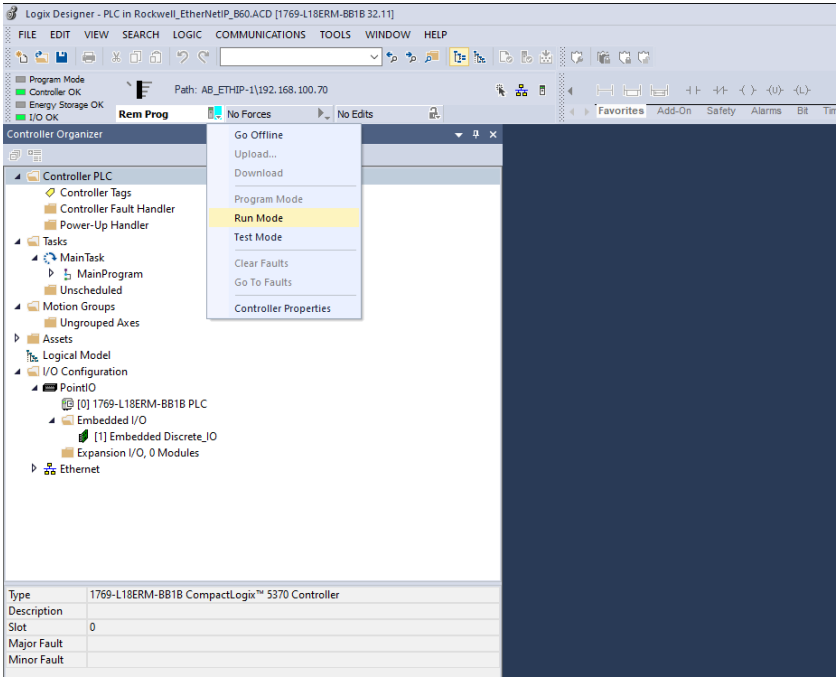


6.5 Download Configuration to PLC

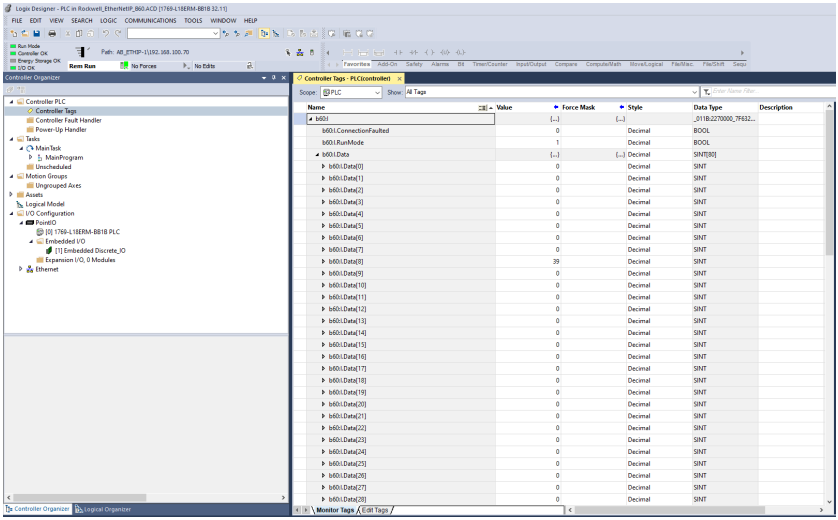
Click on “Communications” → “Download” in order to download the current configuration to the PLC.



Select “Run Mode” to update the input and output data.



Input and output data of the Machine Vision Device is available at b60:I and b60:O.



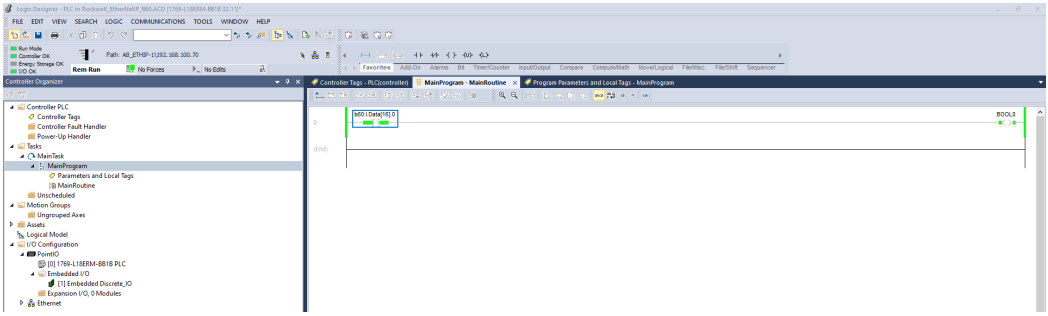
If using a configuration file with e.g. two Processing Instances on the Machine Vision Controller MVC, the input and output data look like this:

- Inputs (Device to PLC):
  - » 8 Bytes Input: Device Status
  - » Processing Instance 1:
    - 8 Bytes Input: Status of Processing Instance 1
    - x Bytes Input: User-Defined Process Data of Processing Instance 1
  - » Processing Instance 2:
    - 8 Bytes Input: Status of Processing Instance 1
    - x Bytes Input: User-Defined Process Data of Processing Instance 2
- Outputs (PLC to Device)
  - » Processing Instance 1:
    - 4 Bytes Output: Commands of Processing Instance 1
    - x Bytes Output: User-Defined Process Data of Processing Instance 1
  - » Processing Instance 2:
    - 4 Bytes Output: Commands of Processing Instance 2
    - x Bytes Output: User-Defined Process Data of Processing Instance 2

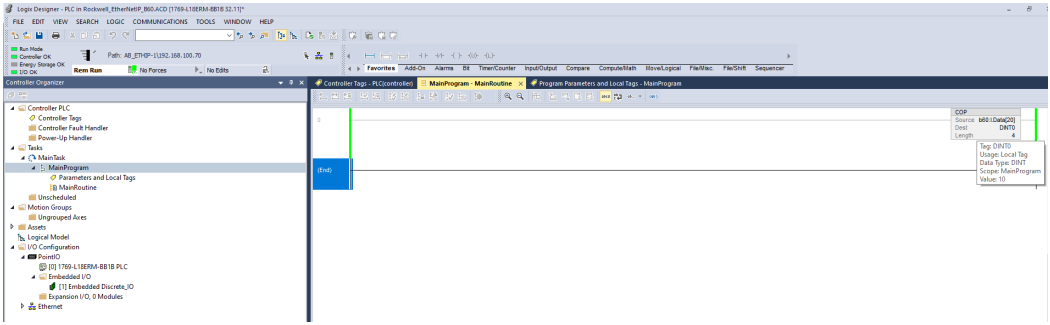
6.6 PLC Parameters and Local Tags

Select the Main Routine and the PLC Paramters and Local Tags in order to add input and output data to your PLC project. Use the address offsets and the data types provided within the uniVision job for the user-defined process data.

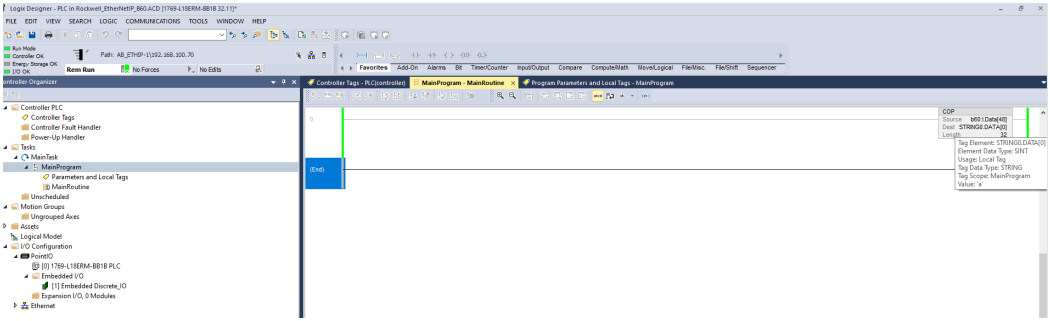
In order to copy boolean results in a local tag, add the “Examine On” and the “Output Energize” in the Main Routine. Link the “Examine On” to the suitable bit and link “Output Energize” to a boolean tag.



In order to create DINT or REAL results, use the FILE COPY (COP) function in the Main Routine. Link the source to the suitable start byte and the destination to a DINT or REAL tag (Length: 4 bytes).



To copy bytes into a string, use the FILE COPY(COP) function in the Main Routine. Link the source to the suitable start byte, the destination to a STRING tag and set the size accordingly (e.g. 32 bytes).



7. EtherCAT PLC Settings

The integration for the Machine Vision Controller MVC is shown with the software TwinCAT3 from Beckhoff. It works identically for other Machine Vision Devices (e.g. for the Smart Camera B60).

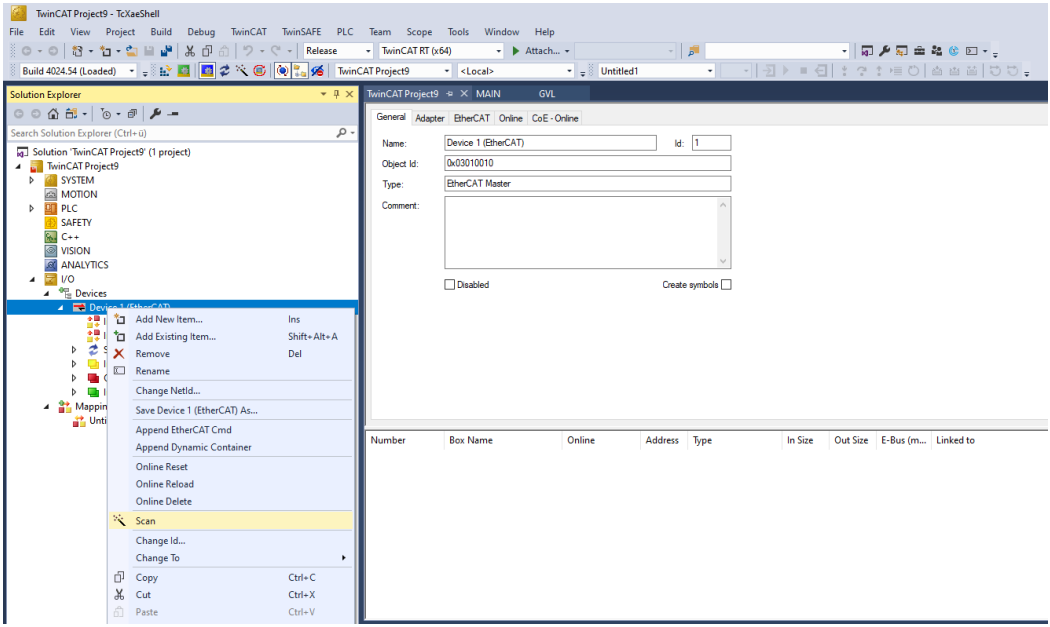
7.1 ESI File

The ESI file is available on the wenglor website in the download section of the Machine Vision Device (www.wenglor.com). Download the ESI file, unzip it and install it on the PLC. In the software TwinCAT3 3.1, just copy the ESI file on Windows PCs in the following file location:

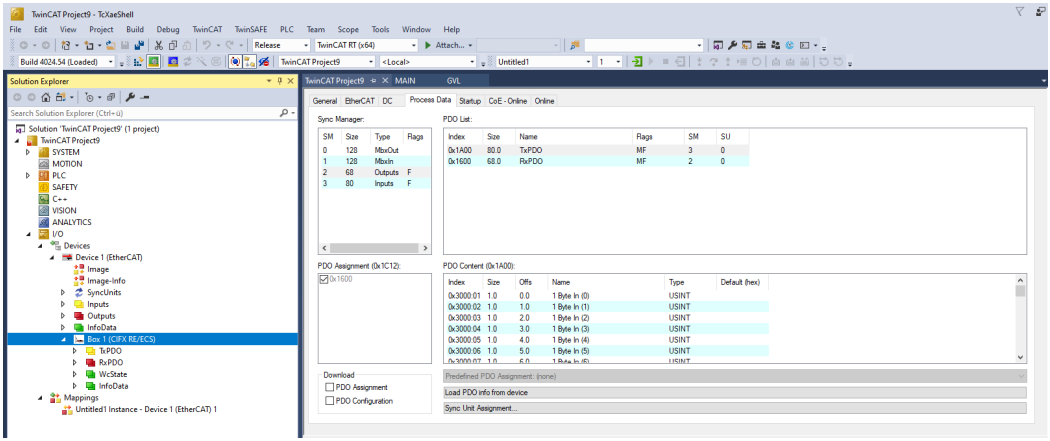
C:\TwinCAT\3.1\Config\Io\EtherCAT

7.2 Add Machine Vision Device to PLC Network

Start the software TwinCAT and create a solution with an EtherCAT Master. Open the context menu at “Device 1 (EtherCAT)” (EtherCAT Master) and click on “Scan” to find EtherCAT Slave devices in the network.



TwinCAT3 adds the Machine Vision Device to the network, if connected via the RTE port. The following image shows the Machine Vision Controller with the configuration file J001.

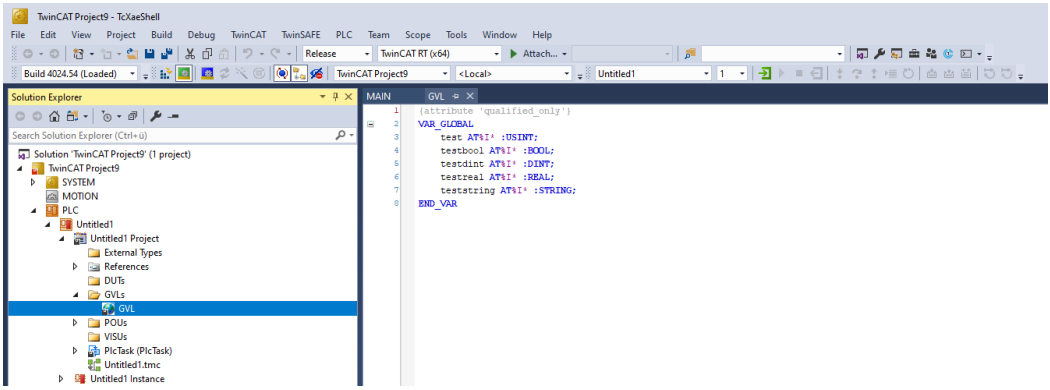


If using a configuration file with e.g. two Processing Instances on the Machine Vision Controller MVC, the input and output data look like this:

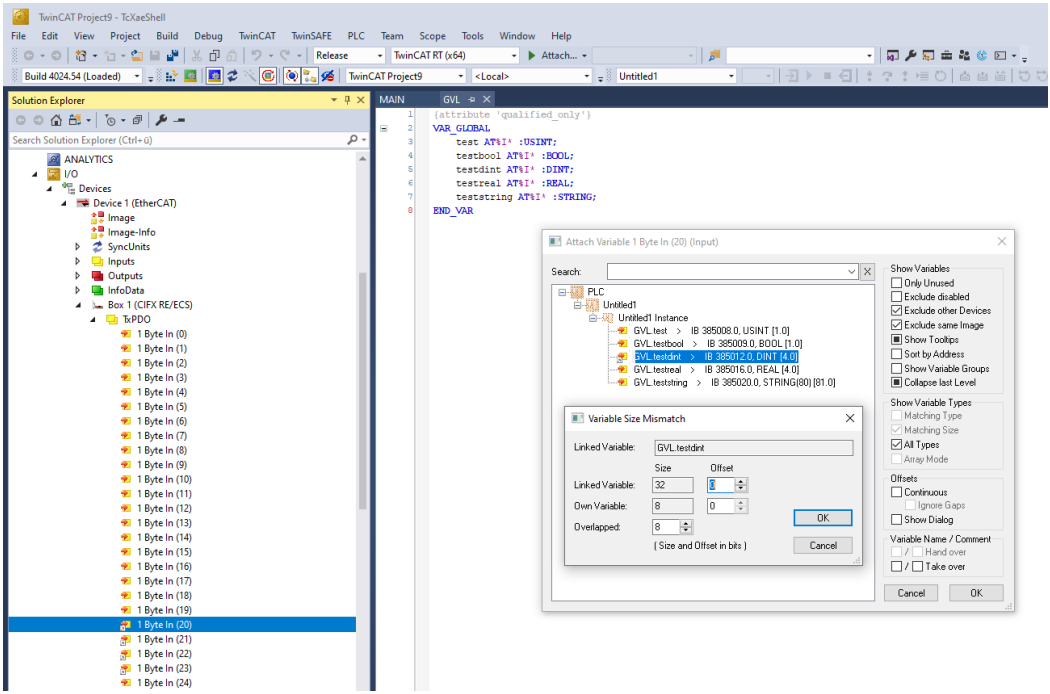
- Inputs (Device to PLC):
  - » 8 Bytes Input: Device Status
  - » Processing Instance 1:
    - 8 Bytes Input: Status of Processing Instance 1
    - x Bytes Input: User-Defined Process Data of Processing Instance 1
  - » Processing Instance 2:
    - 8 Bytes Input: Status of Processing Instance 1
    - x Bytes Input: User-Defined Process Data of Processing Instance 2
- Outputs (PLC to Device)
  - » Processing Instance 1:
    - 4 Bytes Output: Commands of Processing Instance 1
    - x Bytes Output: User-Defined Process Data of Processing Instance 1
  - » Processing Instance 2:
    - 4 Bytes Output: Commands of Processing Instance 2
    - x Bytes Output: User-Defined Process Data of Processing Instance 2

7.3 PLC Variables

Create variables to convert USINTs to the relevant datatype.

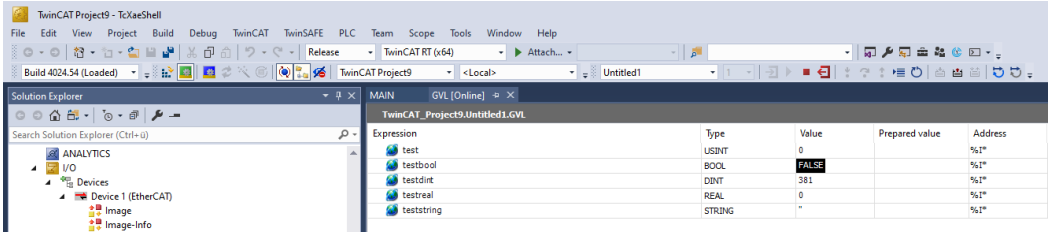


Change the link of the input or output byte at the Machine Vision Device in order to assign it to a variable. In the following example, the example DINT (“testdint”) links to the bytes 20 to 23.



7.4 Activate PLC Configuration

Activate the configuration, login and start the PLC program to see the variables.



8. Change Index of Operating Instructions

Version	Date	Description	Associated software version
1.0.0	29.11.2024	Initial version for the market introduction	Software uniVision 3.1.0
1.1.0	15.05.2024	Updated description for version 3.2.0 of Software wenglor uniVision (incl. EtherNet/IP).	Software uniVision 3.2.0
1.2.0	25.07.2024	Updated description for version 3.3.0 of Software wenglor uniVision (incl. EtherCAT and Machine Vision Controller MVC support)	Software uniVision 3.3.0
1.3.0	30.01.2025	Bugfixes: • Loading jobs via RTE works for maximum 254 jobs (not 255). • Order of input and output slots at PROFINET for several processing instances at the Machine Vision Controller.	Software uniVision 3.4.0
1.4.0	12.05.2025	Updated description for version 3.5.0 of Software wenglor uniVision 3 (incl. EtherCAT on Smart Camera B60).	Software uniVision 3.5.0
1.5.0	20.08.2025	Minor fixes	Software uniVision 3.6.0
1.6.0	19.02.2026	Updated description for version 3.7.0 of Software wenglor uniVision 3.	Software uniVision 3.7.0

9. Attachments

The configuration files are available on the wenglor website at the product detail page of the Machine Vision Device (see www.wenglor.com). For further info about the installation of configuration files see section “4.1 Installation of Configuration Files”.

9.1 Configuration Files for Smart Camera B60

Select the suitable configuration file according to your application, for example:

- H001, F001, I001: Includes all kinds of datatypes
- H101, F101, I101: Includes many REAL (e.g. for pick and place applications)
- H201, F201, I201: Includes many STRINGS with 32 bytes (e.g. for code reading)
- H301, F301, I301: Includes STRING with 64 bytes (e.g. for code reading of long codes)

9.1.1 Configuration File H001 (Default for B60)

Basics

- Device Type: B60
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- 8 Bytes Input: Status of Processing Instance
- 64 Bytes Input: User-Defined Process Data
- 4 Bytes Output: Commands
- 64 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 12 Bytes: 3 DINT
- 16 Bytes: 4 REAL
- 32 Bytes: 1 STRING with 32 Bytes

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 12 Bytes: 3 DINT
- 16 Bytes: 4 REAL
- 32 Bytes: 1 STRING with 32 Bytes

9.1.2 Configuration File H101

Basics

- Device Type: B60
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- 8 Bytes Input: Status of Processing Instance
- 64 Bytes Input: User-Defined Process Data
- 64 Bytes Input: User-Defined Process Data
- 4 Bytes Output: Commands
- 32 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 60 Bytes: 15 REAL
- 64 Bytes: 16 REAL

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 28 Bytes: 7 REAL

9.1.3 Configuration File H201

Basics

- Device Type: B60
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- 8 Bytes Input: Status of Processing Instance
- 16 Bytes Input: User-Defined Process Data
- 32 Bytes Input: User-Defined Process Data
- 64 Bytes Input: User-Defined Process Data
- 4 Bytes Output: Commands
- 4 Bytes Output: User-Defined Process Data
- 32 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 4 Bytes: 1 DINT
- 8 Bytes: 2 REAL
- 32 Bytes: 1 STRING with 32 Bytes
- 32 Bytes: 1 STRING with 32 Bytes
- 32 Bytes: 1 STRING with 32 Bytes

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 32 Bytes: 1 STRING with 32 Bytes

9.1.4 Configuration File H301

Basics

- Device Type: B60
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- 8 Bytes Input: Status of Processing Instance
- 16 Bytes Input: User-Defined Process Data
- 64 Bytes Input: User-Defined Process Data
- 4 Bytes Output: Commands
- 4 Bytes Output: User-Defined Process Data
- 64 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 4 Bytes: 1 DINT
- 8 Bytes: 2 REAL
- 64 Bytes: 1 STRING with 64 Bytes

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 64 Bytes: 1 STRING with 64 Bytes

9.1.5 Configuration File F001

Basics

- Device Type: B60
- Protocol: EtherNet/IP

Input and Output Size

Input Bytes	Output Bytes
80	68

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 64 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 12 Bytes: 3 DINT
    - 16 Bytes: 4 REAL
    - 32 Bytes: 1 STRING with 32 Bytes
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 64 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 12 Bytes: 3 DINT
    - 16 Bytes: 4 REAL
    - 32 Bytes: 1 STRING with 32 Bytes

9.1.6 Configuration File F101

Basics

- Device Type: B60
- Protocol: EtherNet/IP

Input and Output Size

Input Bytes	Output Bytes
144	36

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 128 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 124 Bytes: 31 REAL
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 32 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 28 Bytes: 7 REAL

9.1.7 Configuration File F201

Basics

- Device Type: B60
- Protocol: EtherNet/IP

Input and Output Size

Input Bytes	Output Bytes
128	40

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 112 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 4 Bytes: 1 DINT
    - 8 Bytes: 2 REAL
    - 96 Bytes: 3 STRING with 32 Bytes
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 36 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 32 Bytes: 1 STRING with 32 Bytes

9.1.8 Configuration File F301

Basics

- Device Type: B60
- Protocol: EtherNet/IP

Input and Output Size

Input Bytes	Output Bytes
96	72

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 80 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 4 Bytes: 1 DINT
    - 8 Bytes: 2 REAL
    - 64 Bytes: 1 STRING with 64 Bytes
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 68 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 64 Bytes: 1 STRING with 64 Bytes

9.1.9 Configuration File I001

Basics

- Device Type: B60
- Protocol: EtherCAT

Input and Output Size

Input Bytes	Output Bytes
80	68

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 64 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 12 Bytes: 3 DINT
    - 16 Bytes: 4 REAL
    - 32 Bytes: 1 STRING with 32 Bytes
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 64 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 12 Bytes: 3 DINT
    - 16 Bytes: 4 REAL
    - 32 Bytes: 1 STRING with 32 Bytes

9.1.10 Configuration File I101

Basics

- Device Type: B60
- Protocol: EtherCAT

Input and Output Size

Input Bytes	Output Bytes
144	36

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 128 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 124 Bytes: 31 REAL
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 32 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 28 Bytes: 7 REAL

9.1.11 Configuration File I201

Basics

- Device Type: B60
- Protocol: EtherCAT

Input and Output Size

Input Bytes	Output Bytes
128	40

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 112 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 4 Bytes: 1 DINT
    - 8 Bytes: 2 REAL
    - 96 Bytes: 3 STRING with 32 Bytes
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 36 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 32 Bytes: 1 STRING with 32 Bytes

9.1.12 Configuration File I301

Basics

- Device Type: B60
- Protocol: EtherCAT

Input and Output Size

Input Bytes	Output Bytes
96	72

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » 8 Bytes Input: Status of Processing Instance
  - » 80 Bytes Input: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 4 Bytes: 1 DINT
    - 8 Bytes: 2 REAL
    - 64 Bytes: 1 STRING with 64 Bytes
- Outputs (PLC to Device)
  - » 4 Bytes Output: Commands
  - » 68 Bytes Output: User-Defined Process Data
    - 4 Bytes: 32 BOOL
    - 64 Bytes: 1 STRING with 64 Bytes

9.2 Configuration Files for Machine Vision Controller MVC

Select the suitable configuration file according to your application, for example:

- E0xx, G0xx, J0xx: Includes all kinds of datatypes.
- E1xx, G1xx, J1xx: Includes many REAL (e.g. for pick and place applications)
- E2xx, G2xx, J2xx: Includes many STRINGS with 32 bytes (e.g. for code reading)
- E3xx, G3xx, J3xx: Includes STRING with 64 bytes (e.g. for code reading of long codes)

9.2.1 Configuration File E0xx (E001 is default for MVC)

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. E002 is for 2 Processing Instances)
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- Slots for each Processing Instance:
  - » 8 Bytes Input: Status of Processing Instance
  - » 64 Bytes Input: User-Defined Process Data
  - » 4 Bytes Output: Commands
  - » 64 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 12 Bytes: 3 DINT
- 16 Bytes: 4 REAL
- 32 Bytes: 1 STRING with 32 Bytes

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 12 Bytes: 3 DINT
- 16 Bytes: 4 REAL
- 32 Bytes: 1 STRING with 32 Bytes

9.2.2 Configuration File E1xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. E102 is for 2 Processing Instances)
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- Slots for each Processing Instance:
  - » 8 Bytes Input: Status of Processing Instance
  - » 64 Bytes Input: User-Defined Process Data
  - » 64 Bytes Input: User-Defined Process Data
  - » 4 Bytes Output: Commands
  - » 32 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 60 Bytes: 15 REAL
- 64 Bytes: 16 REAL

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 28 Bytes: 7 REAL

9.2.3 Configuration File E2xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. E202 is for 2 Processing Instances)
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- Slots for each Processing Instance:
  - » 8 Bytes Input: Status of Processing Instance
  - » 16 Bytes Input: User-Defined Process Data
  - » 32 Bytes Input: User-Defined Process Data
  - » 64 Bytes Input: User-Defined Process Data
  - » 4 Bytes Output: Commands
  - » 4 Bytes Output: User-Defined Process Data
  - » 32 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 4 Bytes: 1 DINT
- 8 Bytes: 2 REAL
- 32 Bytes: 1 STRING with 32 Bytes
- 32 Bytes: 1 STRING with 32 Bytes
- 32 Bytes: 1 STRING with 32 Bytes

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 32 Bytes: 1 STRING with 32 Bytes

9.2.4 Configuration File E3xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. E302 is for 2 Processing Instances)
- Protocol: PROFINET

Slot Configuration

- 8 Bytes Input: Device Status
- Slots for each Processing Instance:
  - » 8 Bytes Input: Status of Processing Instance
  - » 16 Bytes Input: User-Defined Process Data
  - » 64 Bytes Input: User-Defined Process Data
  - » 4 Bytes Output: Commands
  - » 4 Bytes Output: User-Defined Process Data
  - » 64 Bytes Output: User-Defined Process Data

User-Defined Process Data Input (Device to PLC)

- 4 Bytes: 32 BOOL
- 4 Bytes: 1 DINT
- 8 Bytes: 2 REAL
- 64 Bytes: 1 STRING with 64 Bytes

User-Defined Process Data Output (PLC to Device)

- 4 Bytes: 32 BOOL
- 64 Bytes: 1 STRING with 64 Bytes

9.2.5 Configuration File G0xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. G002 is for 2 Processing Instances)
- Protocol: EtherNet/IP

Input and Output Size

Configuration File	Input Bytes	Output Bytes
G001	80	68
G002	152	136
G003	224	204
G004	296	272
G005	368	340
G006	440	408

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 64 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 12 Bytes: 3 DINT
      - 16 Bytes: 4 REAL
      - 32 Bytes: 1 STRING with 32 Bytes

- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 64 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 12 Bytes: 3 DINT
      - 16 Bytes: 4 REAL
      - 32 Bytes: 1 STRING with 32 Bytes

9.2.6 Configuration File G1xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. G102 is for 2 Processing Instances)
- Protocol: EtherNet/IP

Input and Output Size

Configuration File	Input Bytes	Output Bytes
G101	144	36
G102	280	72
G103	416	108

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 128 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 124 Bytes: 31 REAL
- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 32 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 28 Bytes: 7 REAL

9.2.7 Configuration File G2xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. G202 is for 2 Processing Instances)
- Protocol: EtherNet/IP

Input and Output Size

Configuration File	Input Bytes	Output Bytes
G201	128	40
G202	248	80
G203	368	120
G204	488	160

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 112 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 4 Bytes: 1 DINT
      - 8 Bytes: 2 REAL
      - 96 Bytes: 3 STRING with 32 Bytes

- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 36 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 32 Bytes: 1 STRING with 32 Bytes

9.2.8 Configuration File G3xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. G302 is for 2 Processing Instances)
- Protocol: EtherNet/IP

Input and Output Size

Configuration File	Input Bytes	Output Bytes
G301	96	72
G302	184	144
G303	272	216
G304	360	288
G305	448	360

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 80 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 4 Bytes: 1 DINT
      - 8 Bytes: 2 REAL
      - 64 Bytes: 1 STRING with 64 Bytes
- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 68 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 64 Bytes: 1 STRING with 64 Bytes

9.2.9 Configuration File J0xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. J002 is for 2 Processing Instances)
- Protocol: EtherCAT

Input and Output Size

Configuration File	Input Bytes	Output Bytes
J001	80	68
J002	152	136
J003	224	204

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 64 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 12 Bytes: 3 DINT
      - 16 Bytes: 4 REAL
      - 32 Bytes: 1 STRING with 32 Bytes
- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 64 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 12 Bytes: 3 DINT
      - 16 Bytes: 4 REAL
      - 32 Bytes: 1 STRING with 32 Bytes

9.2.10 Configuration File J1xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. J101 is for 1 Processing Instance)
- Protocol: EtherCAT

Input and Output Size

Configuration File	Input Bytes	Output Bytes
J101	144	36

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - ◦ 8 Bytes Input: Status of Processing Instance
    - ◦ 128 Bytes Input: User-Defined Process Data
      - ▪ 4 Bytes: 32 BOOL
      - ▪ 124 Bytes: 31 REAL
- Outputs (PLC to Device)
  - » For each Processing Instance
    - ◦ 4 Bytes Output: Commands
    - ◦ 32 Bytes Output: User-Defined Process Data
      - ▪ 4 Bytes: 32 BOOL
      - ▪ 28 Bytes: 7 REAL

9.2.11 Configuration File J2xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. J202 is for 2 Processing Instance)
- Protocol: EtherCAT

Input and Output Size

Configuration File	Input Bytes	Output Bytes
J201	128	40
J202	248	80

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 112 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 4 Bytes: 1 DINT
      - 8 Bytes: 2 REAL
      - 96 Bytes: 3 STRING with 32 Bytes
- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 36 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 32 Bytes: 1 STRING with 32 Bytes

9.2.12 Configuration File J3xx

Basics

- Device Type: MVC
- Number of Processing Instances: xx (coded in the name of the configuration file, e.g. J302 is for 2 Processing Instance)
- Protocol: EtherCAT

Input and Output Size

Configuration File	Input Bytes	Output Bytes
J301	96	72
J302	184	144

Configuration

- Inputs (Device to PLC)
  - » 8 Bytes Input: Device Status
  - » For each Processing Instance
    - 8 Bytes Input: Status of Processing Instance
    - 80 Bytes Input: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 4 Bytes: 1 DINT
      - 8 Bytes: 2 REAL
      - 64 Bytes: 1 STRING with 64 Bytes
- Outputs (PLC to Device)
  - » For each Processing Instance
    - 4 Bytes Output: Commands
    - 68 Bytes Output: User-Defined Process Data
      - 4 Bytes: 32 BOOL
      - 64 Bytes: 1 STRING with 64 Bytes