EN

B50
C50

Smart Camera weQube with EtherNet/IP

EtherNet/IP™

Interface Protocol

Subject to change without notice
Available as PDF version only
Version: 1.0.3
Status: 03.01.2022
www.wenglor.com

Table of Contents

1. Use for Intended Purpose

The Smart Camera weQube is able to communicate with a PLC via EtherNet/IP. Thus, an exchange of process data between the Smart Camera and the PLC is possible. Furthermore, the Smart Camera sends a status to the PLC, which in turn can send commands to the Smart Camera.

NOTE!
In the manual, the EtherNet/IP integration is shown at an Allen-Bradley PLC 1769-L18E-RM-BB1B with Studio 5000 Logix Designer V32.

2. Basics about the EtherNet/IP interface of the Smart Camera

2.1 Identity Object

- Vendor ID: 1211
- Vendor Name: wenglor sensoric gmbh
- Product Type: 12 (Communications Adapter)
- Product Code: 4096
- Product Name: weqube
- Revision: 1.3

2.2 Assembly Object

- Instance 100: Configuration Assembly (Size: 12 bytes)
- Instance 101: Input Assembly (Data size depends on configuration)
- Instance 102: Output Assembly (Data size depends on configuration)

2.3 Configuration Object

CLASS 1 connection with:

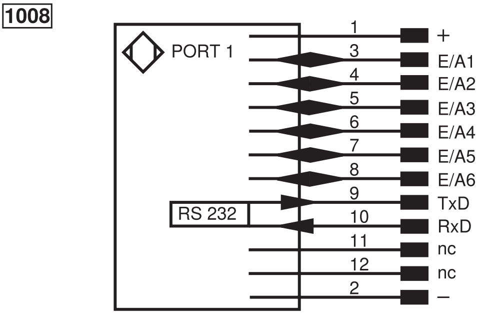
- Config 1: Module ID of Project Number
- Config 2: Module ID of Device Status
- Config 3: Module ID of Slot #3
- Config 4: Module ID of Slot #4
- Config 5: Module ID of Slot #5
- Config 6: Module ID of Slot #6

NOTE!
Only CLASS 1 connections are supported (no CLASS 3 connections)!

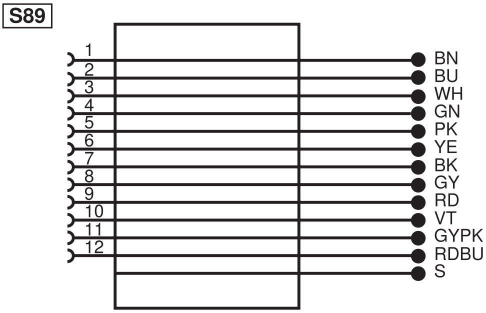
NOTE!
For not used slots, the Module ID must be set to 0.

3. Electrical Connection and Network Overview

Connect port 1 of the Smart Camera to 18…30 V DC. Connect pin 1 (wenglor standard cable: Brown) to the plus pole and pin 2 (wenglor standard cable: Blue) to the minus pole.

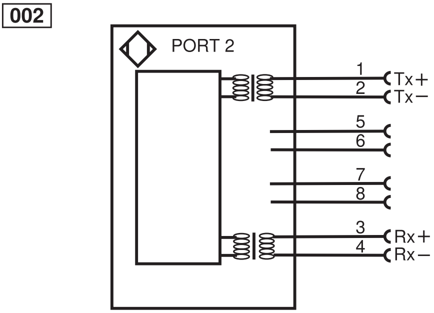


Connection Diagram, weQube Smart Camera, Port 1

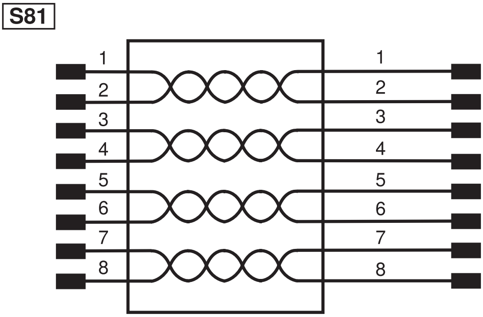


Matching wenglor Connection Equipment

- Connect port 2 of the Smart Camera for EtherNet/IP communication with a PLC – directly or via a switch.



Connection Diagram, weQube Smart Camera, Port 2



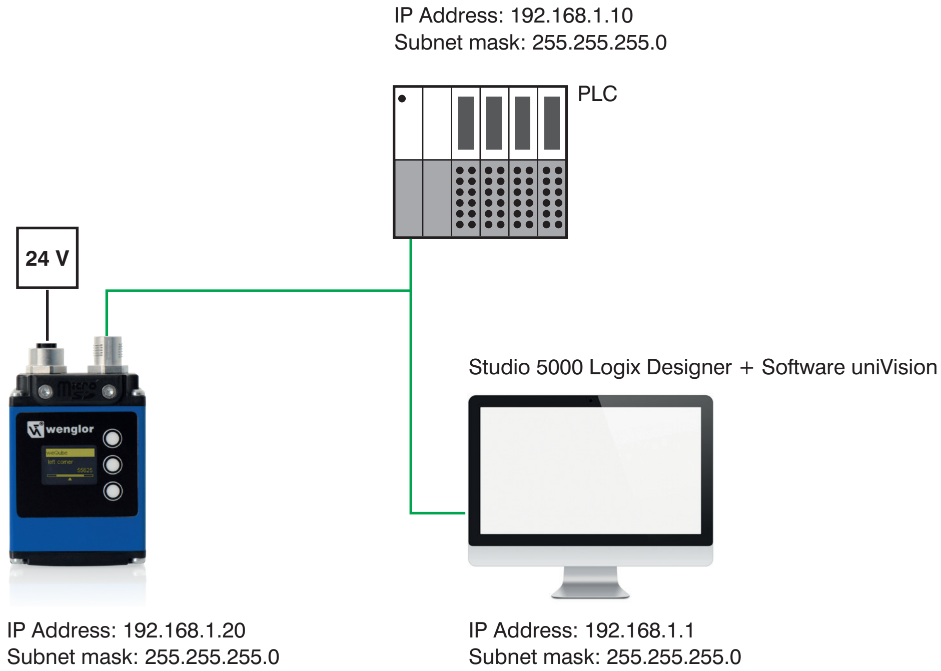
Matching wenglor Connection Equipment

LEDs for EtherNet/IP at the Smart Camera:

LED	Color	State	Meaning
MS (Module Status)	● (Green)	On	Operation Ready
	✹ (Green)	Flashing	Standby
	● (Red)	On	Fatal Error
	✹ (Red)	Flashing	Error
	● (Off)	Off	Switched off
NS (Network Status)	● (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
L/A	● (Green)	On	Ethernet connection is available.
	✹ (Green)	Flashing	Smart Camera sends or receives Ethernet frames.
	● (Off)	Off	No Ethernet connection available.

NOTE!
Port 2 of the Smart Camera supports EtherNet/IP communication and further network functionality (e.g. Software uniVision for Windows, website, process data via TCP, UDP and FTP).

Example: The Smart Camera weQube, the PLC and a PC with the software Studio 5000 Logix Designer and uniVision are in the same network.


IP Address: 192.168.1.10
Subnet mask: 255.255.255.0
PLC
24 V
Studio 5000 Logix Designer + Software uniVision
IP Address: 192.168.1.20
Subnet mask: 255.255.255.0
IP Address: 192.168.1.1
Subnet mask: 255.255.255.0

4. Input and Output Data

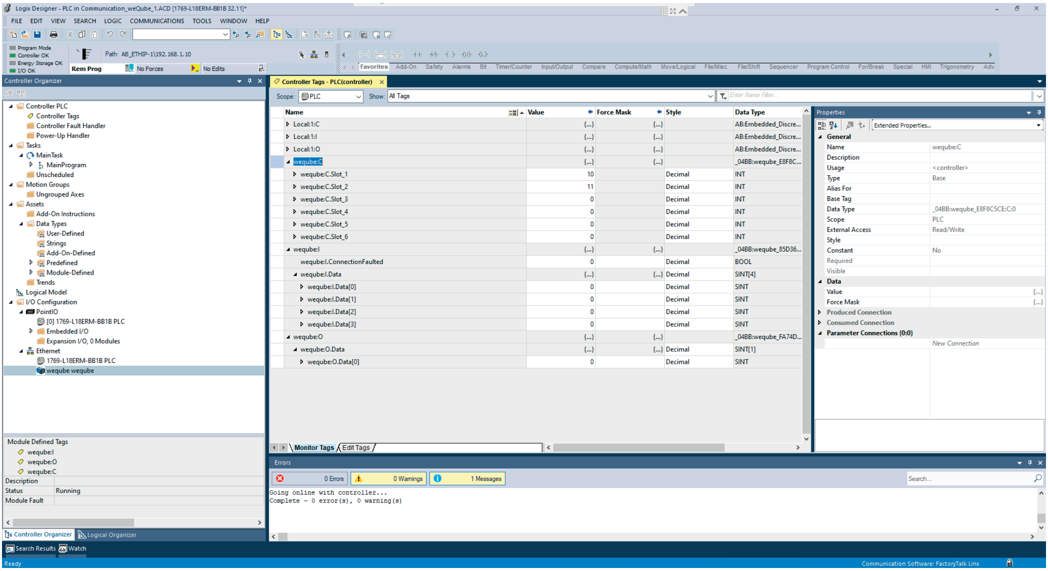
In the view of the PLC, the following input and output data are available for the Smart Camera:

- Slot 1 / weQube:O.Data[0] (fix): Project number (1 Byte Output of PLC)
- Slot 2 / weQube:I.Data[0-3] (fix): Status (4 Bytes Input of PLC)
- Slot 3 – 6 / weQube:I.Data or weQube:O.Data (flexible): User-defined process data (x Bytes Input or Output of PLC)

NOTE!
By default, slot 1 (weQube:O.Data[0]) and slot 2 (weQube:I.Data[0-3]) are always present. Slots 3 to 6 (weQube:I.Data or weQube:O.Data) are optional. The number of optional slots and the data types of such user-defined slots are adjustable.

The following example shows the default input and output configuration of the Smart Camera weQube with slot 1 (weQube:O.Data[0]) and slot 2 (weQube:I.Data[0-3]).

Example:



Description	
Status	Running
Module Fault	

Scope: PLC   Show: All Tags   Enter Name Filter...

Name	Value	Force Mask	Style	Data Type
Local:1:C	{...}	{...}		AB:Embedded_Discre...
Local:1:I	{...}	{...}		AB:Embedded_Discre...
Local:1:O	{...}	{...}		AB:Embedded_Discre...
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...
weqube:C.Slot_1	10		Decimal	INT
weqube:C.Slot_2	11		Decimal	INT
weqube:C.Slot_3	0		Decimal	INT
weqube:C.Slot_4	0		Decimal	INT
weqube:C.Slot_5	0		Decimal	INT
weqube:C.Slot_6	0		Decimal	INT
weqube:I	{...}	{...}		_04BB:weqube_85D36...
weqube:I.ConnectionFaulted	0		Decimal	BOOL
weqube:I.Data	{...}	{...}	Decimal	SINT[4]
weqube:I.Data[0]	0		Decimal	SINT
weqube:I.Data[1]	0		Decimal	SINT
weqube:I.Data[2]	0		Decimal	SINT
weqube:I.Data[3]	0		Decimal	SINT
weqube:O	{...}	{...}		_04BB:weqube_FA74D...
weqube:O.Data	{...}	{...}	Decimal	SINT[1]
weqube:O.Data[0]	0		Decimal	SINT

General	
Name	weqube:C
Description	
Usage	<controller>
Type	Base
Alias For	
Base Tag	
Data Type	_04BB:weqube_E8F8C5CE:C:0
Scope	PLC
External Access	Read/Write
Style	
Constant	No
Required	
Visible	
Data	
Value	{...}
Force Mask	{...}
Produced Connection	
Consumed Connection	
Parameter Connections {0:0}	
	New Connection

```
Going online with controller...
Complete - 0 error(s), 0 warning(s)
```

4.1 Status

The Smart Camera weQube sends a four-byte status information to the PLC. The status gives feedback as to whether the Smart Camera works correctly or is in an error state:

- Status 0: No error
- Status not 0: Error

In case of an error, the binary number indicates the reason of the respective error. The bit number starts with zero. Bits with value true give more information about the error.

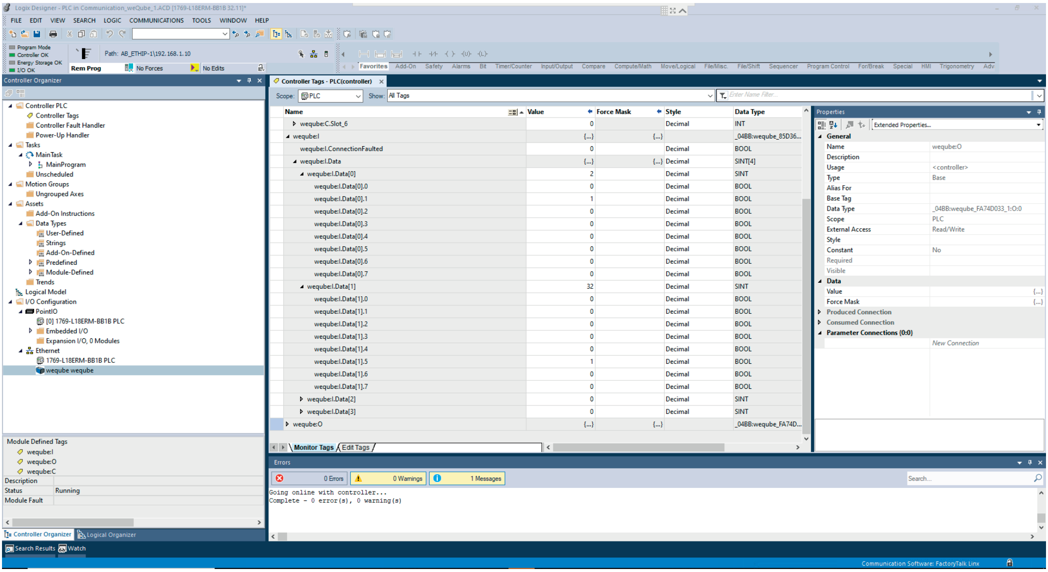
Bit	Section	Signal	Description
0	General	Information	Busy Is high while processing LIMA commands (e.g. because of loading a project or changing any project parameter).
1		Warning	There is at least one bit set, level = Warning
2		Critical Error	There is at least one bit set, level = Critical Error
3		Fatal Error	There is at least one bit set, level = Fatal Error
6	Peripheral	TCP/IP	There is an error concerning the TCP/IP socket
7		UDP	There is an error concerning the UDP socket
8		Industrial Ethernet	There is an error concerning industrial ethernet
12		UART	There is an error concerning the UART device
13		FTP	There is an error concerning the FTP interface.
14	Memory	Flash	There is an error concerning the flash access
15		RAM	There is an error concerning the RAM access
16		SD-Card	There is an error concerning the SD card access
17		File access	There is an error concerning a general file access.
18		Compatibility	There is an error concerning the version of the loaded project
24	Image Processing	Sequencing	There is an error concerning IData vision engine
25		Processing	There is an error concerning a vision module.
26		Trigger	There is an error concerning HW trigger

NOTE!
More details about errors and possible solutions are available in the uniVision software manual.

Example:
The status with the binary number 10 0000 0000 0010 shows an error at bit 1 and bit 13. Consequently, there is a warning that indicates a problem with the FTP interface. An example may be that the Smart Camera is configured to save data on a FTP server, but the FTP server is not available in the network.

Bit	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binary number	1	0	0	0	0	0	0	0	0	0	0	0	1	0

The following screen shows the status bits of the Smart Camera weQube in Studio 5000 Logix Designer in case of the described FTP error.



Path: AB_ETHIP-1\192.168.1.10

Description	
Status	Running
Module Fault	

Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type
weqube:C.Slot_6	0		Decimal	INT
weqube:I	{...}	{...}		_04BB:weqube_85D36...
weqube:I.ConnectionFaulted	0		Decimal	BOOL
weqube:I.Data	{...}	{...}	Decimal	SINT[4]
weqube:I.Data[0]	2		Decimal	SINT
weqube:I.Data[0].0	0		Decimal	BOOL
weqube:I.Data[0].1	1		Decimal	BOOL
weqube:I.Data[0].2	0		Decimal	BOOL
weqube:I.Data[0].3	0		Decimal	BOOL
weqube:I.Data[0].4	0		Decimal	BOOL
weqube:I.Data[0].5	0		Decimal	BOOL
weqube:I.Data[0].6	0		Decimal	BOOL
weqube:I.Data[0].7	0		Decimal	BOOL
weqube:I.Data[1]	32		Decimal	SINT
weqube:I.Data[1].0	0		Decimal	BOOL
weqube:I.Data[1].1	0		Decimal	BOOL
weqube:I.Data[1].2	0		Decimal	BOOL
weqube:I.Data[1].3	0		Decimal	BOOL
weqube:I.Data[1].4	0		Decimal	BOOL
weqube:I.Data[1].5	1		Decimal	BOOL
weqube:I.Data[1].6	0		Decimal	BOOL
weqube:I.Data[1].7	0		Decimal	BOOL
weqube:I.Data[2]	0		Decimal	SINT
weqube:I.Data[3]	0		Decimal	SINT
weqube:O	{...}	{...}		_04BB:weqube_FA74D...

General	
Name	weqube:O
Description	
Usage	<controller>
Type	Base
Alias For	
Base Tag	
Data Type	_04BB:weqube_FA74D033_1:O:0
Scope	PLC
External Access	Read/Write
Style	
Constant	No
Required	
Visible	
Data	
Value	{...}
Force Mask	{...}
Produced Connection	
Consumed Connection	
Parameter Connections {0:0}	
	New Connection

0 Errors 0 Warnings 1 Messages

```
Going online with controller...
Complete - 0 error(s), 0 warning(s)
```

4.2 Commands

Commands (e.g. trigger commands) are sent from the PLC to the Smart Camera. The Smart Camera weQube supports the following commands:

- Trigger
- Load project

It is not allowed to send several commands (e.g. trigger and project load commands) at the same time! Before sending the next command, it is necessary to wait until the processing of the last command has finished completely.

NOTE!
When the PLC sends a command to the Smart Camera, the Smart Camera captures an image or loads the project. Commands are executed immediately in contrast to process data, which is only sent or received by the Smart Camera in case of an evaluation that was started by a trigger signal.

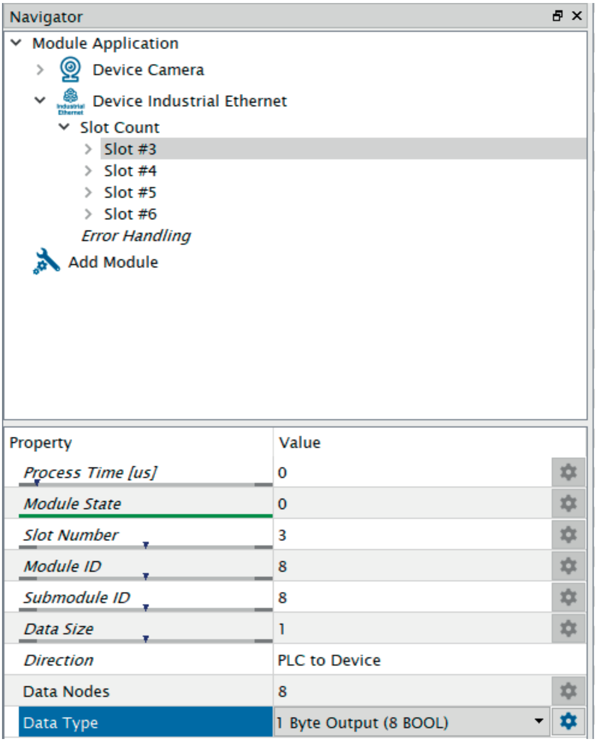
4.2.1 Trigger Command

When the PLC sends a trigger command to the Smart Camera, the Smart Camera captures and evaluates an image and sends the results.

The following steps are necessary to set up a trigger command via EtherNet/IP:

1. Connect to the Smart Camera via the software uniVision for Windows.
2. Add Device Industrial Ethernet to the project.
3. Set Slot Count to define the number of flexible slots (At least one flexible slot is required for the trigger command).
4. Configure one of the flexible slots as 1 Byte Output (8 BOOL)

In the following example, the trigger command via EtherNet/IP is sent at slot 3.



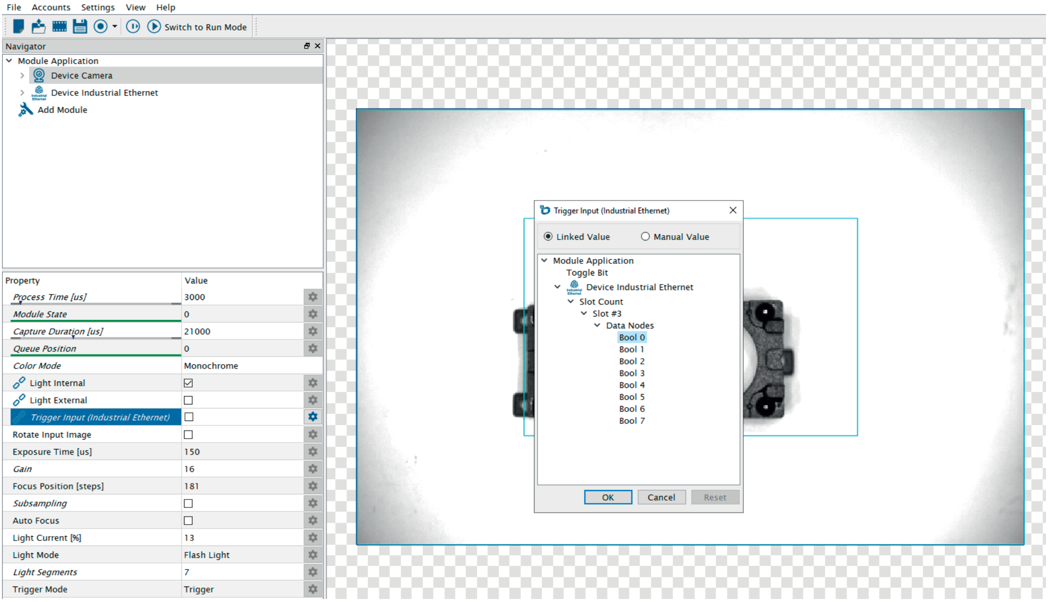
Navigator

- Module Application
  - Device Camera
  - Device Industrial Ethernet
    - Slot Count
      - Slot #3
      - Slot #4
      - Slot #5
      - Slot #6
    - *Error Handling*
  - Add Module

Property	Value
Process Time [us]	0
Module State	0
Slot Number	3
Module ID	8
Submodule ID	8
Data Size	1
Direction	PLC to Device
Data Nodes	8
Data Type	1 Byte Output (8 BOOL)

5. Select Device Camera and link one of the bools of slot 3 to Trigger Input (Industrial Ethernet). In the example, the PLC sends the trigger command at Bool 0 of slot 3.

NOTE!
For triggering via EtherNet/IP, the Trigger Mode of Device Camera must be set to Trigger. Trigger Input (Industrial Ethernet) is only visible in the extended view of the uniVision software.



Property	Value
Process Time [us]	3000
Module State	0
Capture Duration [us]	21000
Queue Position	0
Color Mode	Monochrome
Light Internal	☑
Light External	□
Trigger Input (Industrial Ethernet)	□
Rotate Input Image	□
Exposure Time [us]	150
Gain	16
Focus Position [steps]	181
Subsampling	□
Auto Focus	□
Light Current [%]	13
Light Mode	Flash Light
Light Segments	7
Trigger Mode	Trigger

Trigger Input (Industrial Ethernet)

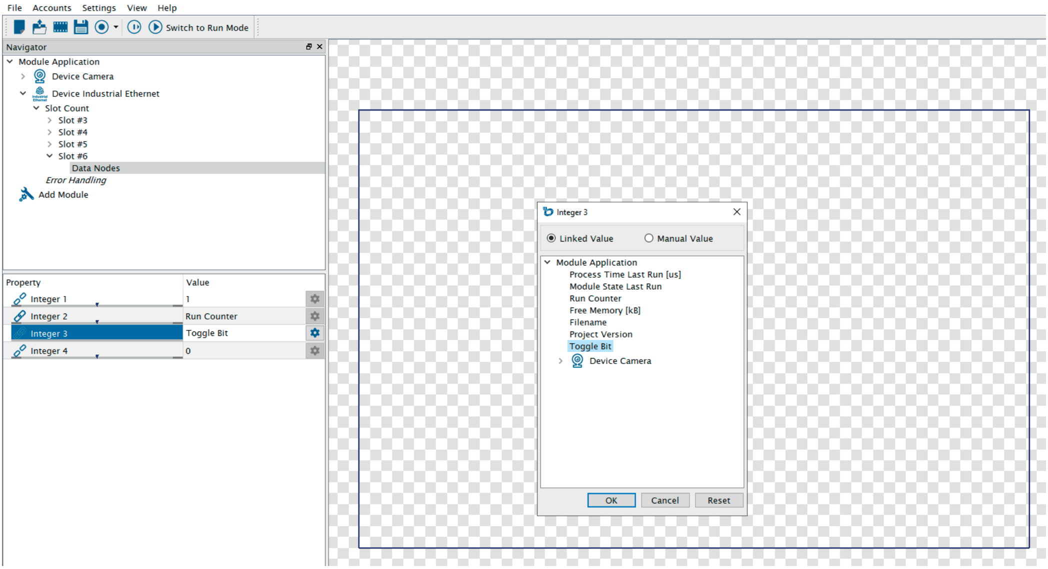
● Linked Value ○ Manual Value

Module Application
Toggle Bit
Device Industrial Ethernet
Slot Count
Slot #3
Data Nodes
Bool 0
Bool 1
Bool 2
Bool 3
Bool 4
Bool 5
Bool 6
Bool 7

OK Cancel Reset

6. Use the last slot in order to verify that all new results of the image evaluation are available on the PLC. For example, configure slot 6 as “16 Byte Input (4 DINT)” and link the toggle bit and the run counter:

- Integer 1: Fix project number
- Integer 2: Run counter
- Integer 3: Toggle bit



Property	Value
Integer 1	1
Integer 2	Run Counter
Integer 3	Toggle Bit
Integer 4	0

Integer 3: ● Linked Value ○ Manual Value — Toggle Bit

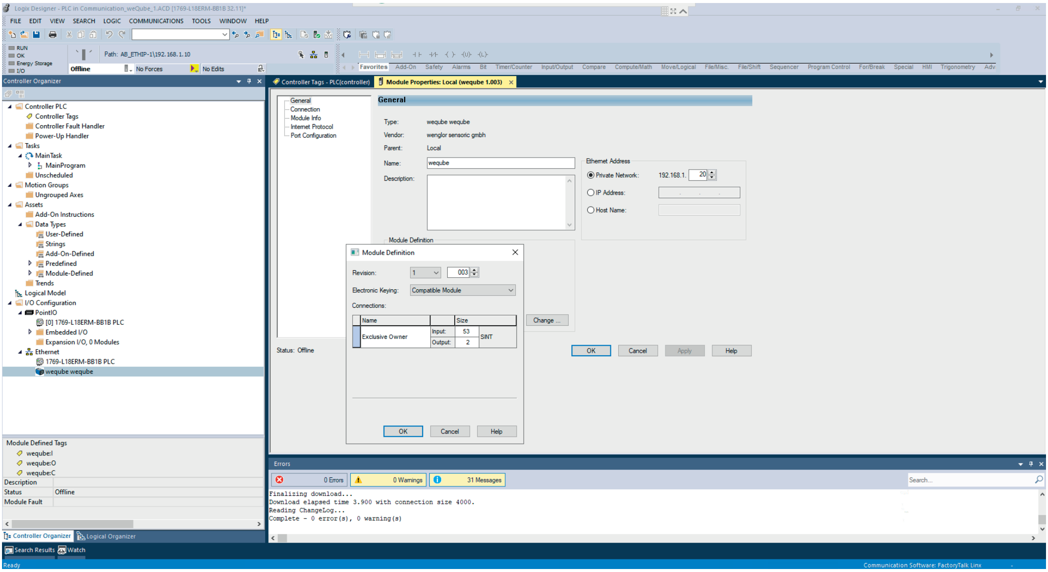
NOTE!
Save the project, set it as startup project and reboot the Smart Camera to apply the settings.

7. Open the software Studio 5000 Logix Designer and adjust the number of input and output addresses of the Smart Camera according to the configuration in the uniVision project.

NOTE!
A detailed description of how to set up the input and output addresses is shown in chapter "7.3 Configure Input and Output Data" on page 40.

The example with 53 bytes input (4 + 1 + 32 + 16) and 2 bytes output (1 + 1) works for the following slot configuration:

- Slot 1 (fix): 1 Byte Output (Project number)
- Slot 2 (fix): 4 Byte Input (Status)
- Slot 3: 1 Byte Output (8 BOOL)
- Slot 4: 1 Byte Input (8 BOOL)
- Slot 5: 32 Byte Input (2 CHAR)
- Slot 6: 16 Byte Input (4 DINT)



Description
Status: Offline
Module Fault

General

Type: weqube weqube
Vendor: wenglor sensoric gmbh
Parent: Local
Name: weqube
Description:

Ethernet Address
● Private Network: 192.168.1. 20
○ IP Address:
○ Host Name:

Module Definition

Status: Offline

Module Definition

Revision: 1 003
Electronic Keying: Compatible Module
Connections:

Name		Size	
Exclusive Owner	Input:	53	SINT
	Output:	2	

OK Cancel Help

Errors

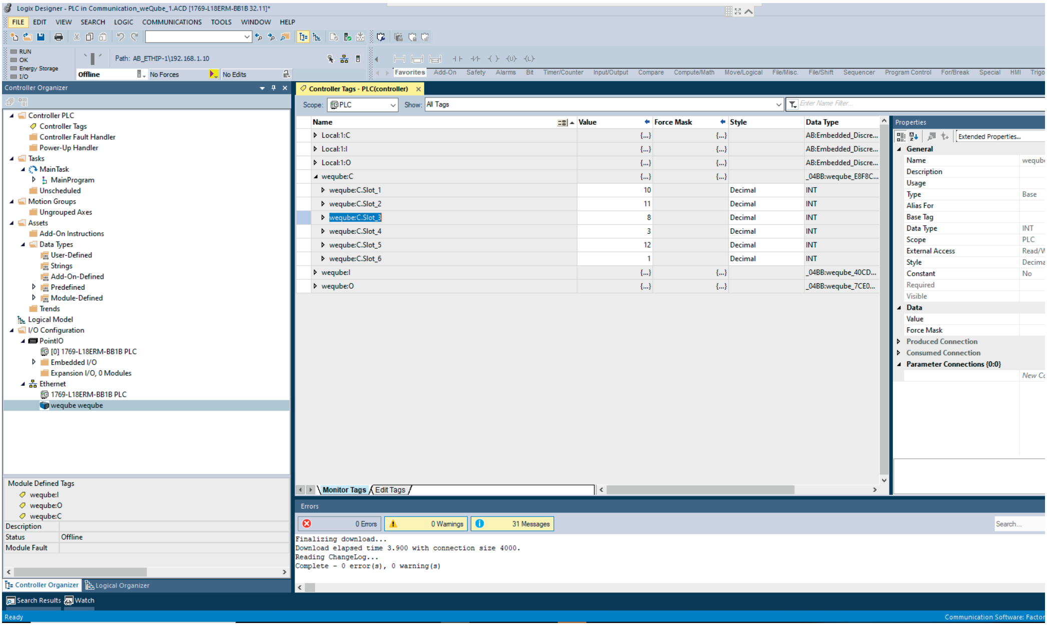
0 Errors 0 Warnings 31 Messages

```
Finalizing download...
Download elapsed time 3.900 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

8. Set the Module IDs in weqube:C according to the configuration in the uniVision project.

NOTE!
A detailed description of all Module IDs is shown in chapter "7.3 Configure Input and Output Data" on page 40.

In the example, the following Module IDs must be used:
- Slot 1 (fix): Module ID 10
- Slot 2 (fix): Module ID 11
- Slot 3: Module ID 8
- Slot 4: Module ID 3
- Slot 5: Module ID 12
- Slot 6: Module ID 1

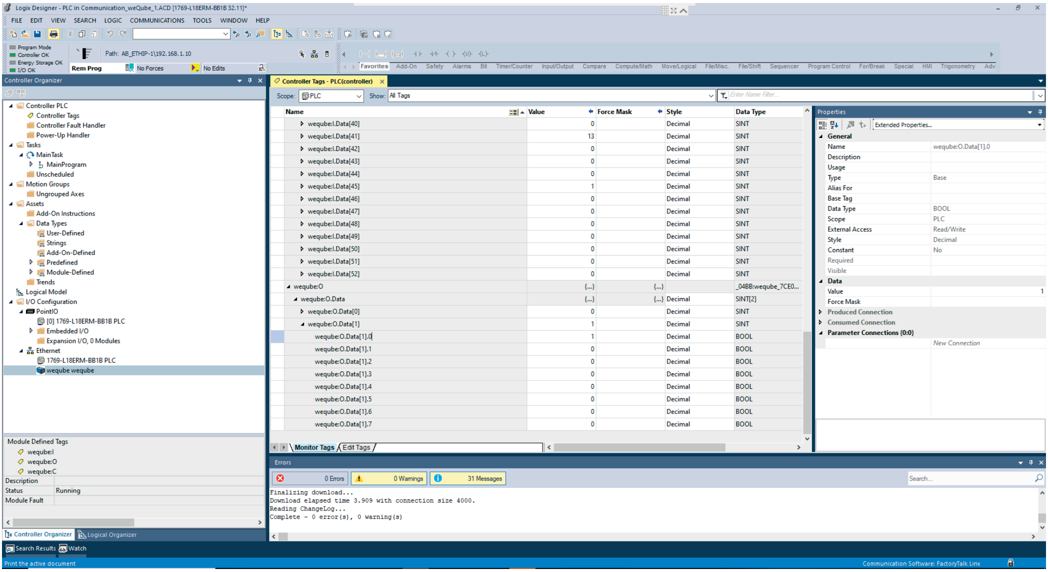

Logix Designer - PLC in Communication_weQube_1.ACD [1769-L18ERM-BB1B 32.11]*
FILE EDIT VIEW SEARCH LOGIC COMMUNICATIONS TOOLS WINDOW HELP
Controller Organizer
Controller Tags - PLC(controller)
Scope: PLC
Show: All Tags
Monitor Tags
Edit Tags
Properties
Errors
0 Errors
0 Warnings
31 Messages

Name	Value	Force Mask	Style	Data Type
Local:1:C	{...}	{...}		AB:Embedded_Discre...
Local:1:I	{...}	{...}		AB:Embedded_Discre...
Local:1:O	{...}	{...}		AB:Embedded_Discre...
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...
weqube:C.Slot_1	10		Decimal	INT
weqube:C.Slot_2	11		Decimal	INT
weqube:C.Slot_3	8		Decimal	INT
weqube:C.Slot_4	3		Decimal	INT
weqube:C.Slot_5	12		Decimal	INT
weqube:C.Slot_6	1		Decimal	INT
weqube:I	{...}	{...}		_04BB:weqube_40CD...
weqube:O	{...}	{...}		_04BB:weqube_7CE0...

```
Finalizing download...
Download elapsed time 3.900 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

9. Download the configuration of the software Studio 5000 Logix Designer to the PLC and go online.

10. Send the value TRUE to the correct address to send a trigger command to the Smart Camera. With every change from FALSE to TRUE for the trigger command bit, the Smart Camera captures and evaluates an image.
    In the example, the trigger signal is sent on weqube:O.Data[1].0. The result of the run counter can be seen on weqube:I.Data[41-44] and the result of the toggle bit is shown on weqube:I.Data[45].0.



Description
Status Running
Module Fault

Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type
weqube:I.Data[40]	0		Decimal	SINT
weqube:I.Data[41]	13		Decimal	SINT
weqube:I.Data[42]	0		Decimal	SINT
weqube:I.Data[43]	0		Decimal	SINT
weqube:I.Data[44]	0		Decimal	SINT
weqube:I.Data[45]	1		Decimal	SINT
weqube:I.Data[46]	0		Decimal	SINT
weqube:I.Data[47]	0		Decimal	SINT
weqube:I.Data[48]	0		Decimal	SINT
weqube:I.Data[49]	0		Decimal	SINT
weqube:I.Data[50]	0		Decimal	SINT
weqube:I.Data[51]	0		Decimal	SINT
weqube:I.Data[52]	0		Decimal	SINT
weqube:O	{...}	{...}		_04BB:weqube_7CE0...
weqube:O.Data	{...}	{...}	Decimal	SINT[2]
weqube:O.Data[0]	0		Decimal	SINT
weqube:O.Data[1]	1		Decimal	SINT
weqube:O.Data[1].0	1		Decimal	BOOL
weqube:O.Data[1].1	0		Decimal	BOOL
weqube:O.Data[1].2	0		Decimal	BOOL
weqube:O.Data[1].3	0		Decimal	BOOL
weqube:O.Data[1].4	0		Decimal	BOOL
weqube:O.Data[1].5	0		Decimal	BOOL
weqube:O.Data[1].6	0		Decimal	BOOL
weqube:O.Data[1].7	0		Decimal	BOOL

Properties – Extended Properties...

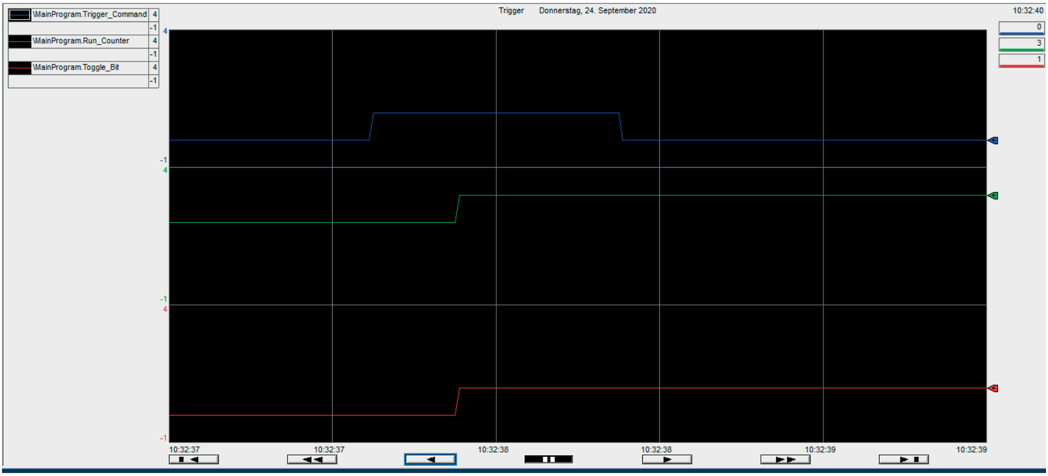
General	
Name	weqube:O.Data[1].0
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	BOOL
Scope	PLC
External Access	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	1
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections {0:0}	
	New Connection

0 Errors 0 Warnings 31 Messages

```
Finalizing download...
Download elapsed time 3.909 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

NOTE!

- After capturing and evaluating the image, the Smart Camera sends results in the form of process data (also via EtherNet/IP if configured accordingly).
- The processing of the trigger signal and the evaluation of the image is completed when the toggle bit has changed and the run counter has increased by one.
- It is not allowed to send several commands at the same time (e.g. trigger and load project commands).
- After a trigger command has been sent from the PLC to the Smart Camera, wait until the results are available on the PLC before sending the next command.



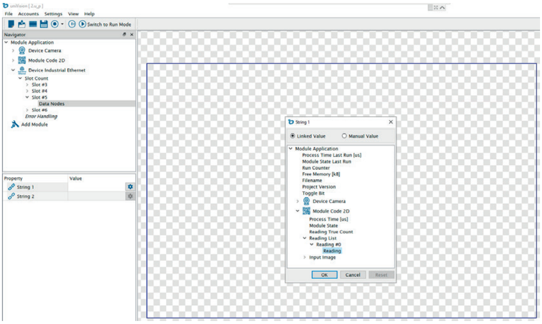
4.2.2 Load Project Command

The load project command allows loading another project onto the Smart Camera. Up to 255 different projects can be loaded via EtherNet/IP.

The following steps are necessary to set up a load project command via EtherNet/IP:
1. Open the software uniVision for Windows and connect to the Smart Camera weQube.
2. Create uniVision projects and save them with a number at the beginning of the filename.

NOTE!
In order to load projects via EtherNet/IP, all projects must be saved in the following format: “xxx_testproject.u_p” (x = any integer from 0 to 9). For example “001_MyProject.u_p”. Project numbers can be set between 1 and 255 (0 is ignored – default value). Use unique numbers for every uniVision project file. The number of slots and the slot configuration must be identical in all uniVision projects in the Smart Camera in order for the project change to be possible from the PLC.

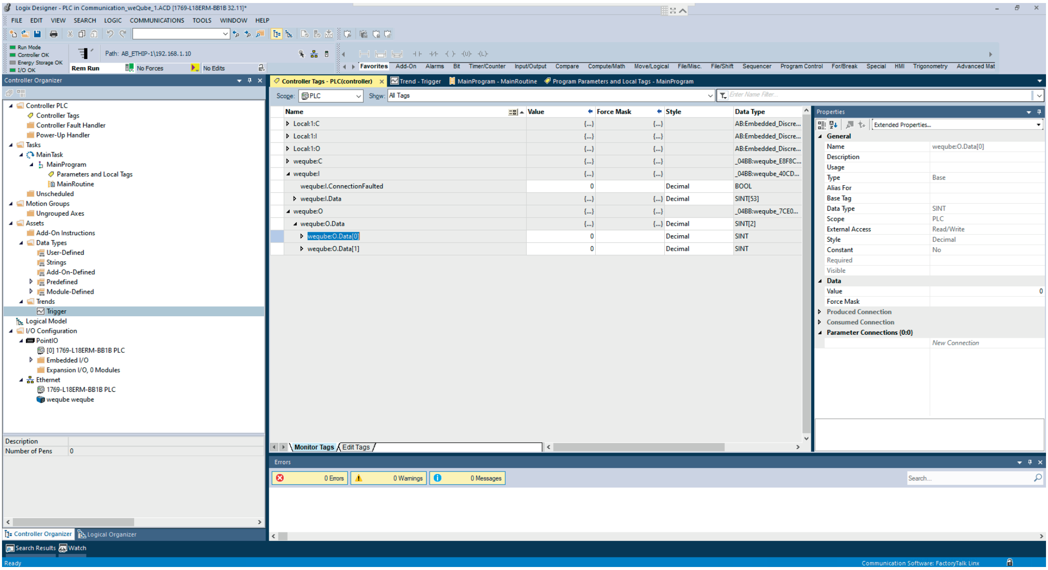
3. Use the last slot in order to send the project number as a fix result from the Smart Camera to the PLC. This value can be used to verify that the project has finished loading.



NOTE!
Save the projects, set one of the projects as startup project and reboot the Smart Camera to apply the settings.

4. Open the software Studio 5000 Logix Designer.

5. Slot 1 / weQube:O.Data[0] (fix) of the Smart Camera is pre-configured for loading uniVision projects.



Logix Designer - PLC in Communication_weQube_1.ACD [1769-L18ERM-BB1B 32.11]*

Path: AB_ETHIP-1\192.168.1.10

Controller Tags - PLC(controller)

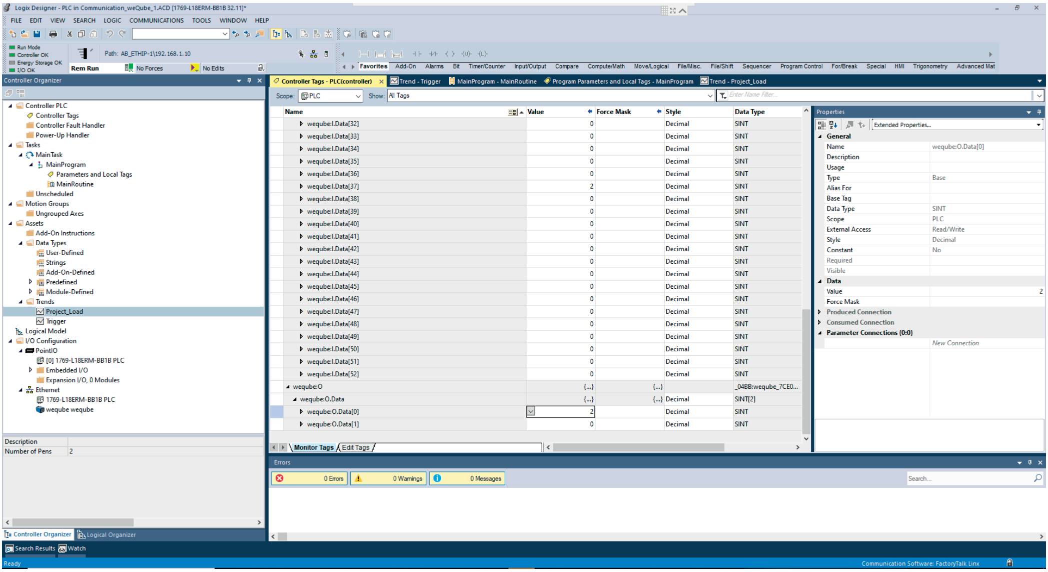
Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type
Local:1:C	{...}	{...}		AB:Embedded_Discre...
Local:1:I	{...}	{...}		AB:Embedded_Discre...
Local:1:O	{...}	{...}		AB:Embedded_Discre...
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...
weqube:I	{...}	{...}		_04BB:weqube_40CD...
weqube:I.ConnectionFaulted	0		Decimal	BOOL
weqube:I.Data	{...}	{...}	Decimal	SINT[53]
weqube:O	{...}	{...}		_04BB:weqube_7CE0...
weqube:O.Data	{...}	{...}	Decimal	SINT[2]
weqube:O.Data[0]	0		Decimal	SINT
weqube:O.Data[1]	0		Decimal	SINT

Properties	
Name	weqube:O.Data[0]
Type	Base
Data Type	SINT
Scope	PLC
External Access	Read/Write
Style	Decimal
Constant	No
Value	0

0 Errors 0 Warnings 0 Messages

6. Send the number defined in the filename of the uniVision project from the PLC to the Smart Camera in order to load the project. With every change of the project number sent by the PLC to the Smart Camera, the Smart Camera loads the project (only if the uniVision project is available and if the project number is different to the currently loaded project).
   In the example, the project number 2 is sent at weqube:O.Data[0]. After the project loading has finished, the weQube sends the current project number back at weqube:I.Data[37-40].



Logix Designer - PLC in Communication_weQube_1.ACD [1769-L18ERM-BB1B 32.11]*

Path: AB_ETHIP-1\192.168.1.10

Controller Tags - PLC(controller)

Scope: PLC Show: All Tags

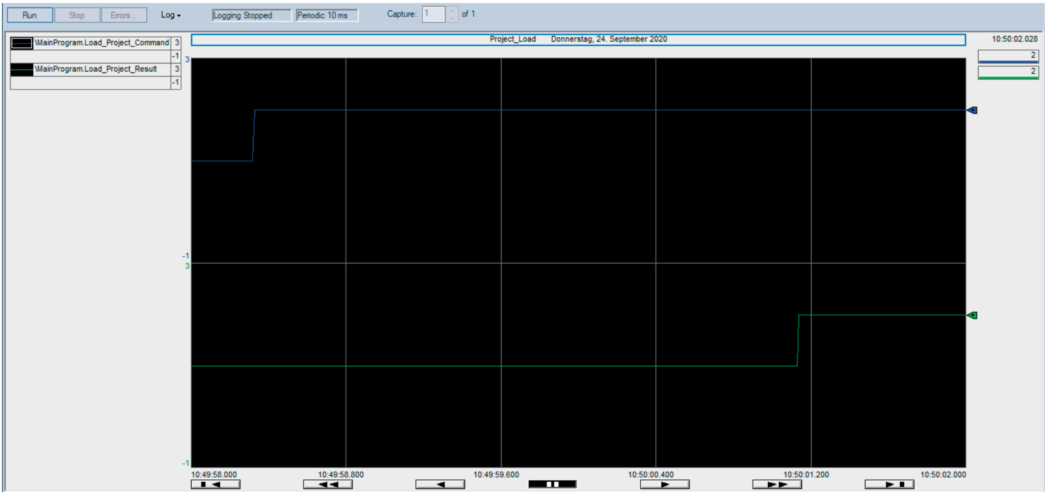
Name	Value	Force Mask	Style	Data Type
weqube:I.Data[32]	0		Decimal	SINT
weqube:I.Data[33]	0		Decimal	SINT
weqube:I.Data[34]	0		Decimal	SINT
weqube:I.Data[35]	0		Decimal	SINT
weqube:I.Data[36]	0		Decimal	SINT
weqube:I.Data[37]	2		Decimal	SINT
weqube:I.Data[38]	0		Decimal	SINT
weqube:I.Data[39]	0		Decimal	SINT
weqube:I.Data[40]	0		Decimal	SINT
weqube:I.Data[41]	0		Decimal	SINT
weqube:I.Data[42]	0		Decimal	SINT
weqube:I.Data[43]	0		Decimal	SINT
weqube:I.Data[44]	0		Decimal	SINT
weqube:I.Data[45]	0		Decimal	SINT
weqube:I.Data[46]	0		Decimal	SINT
weqube:I.Data[47]	0		Decimal	SINT
weqube:I.Data[48]	0		Decimal	SINT
weqube:I.Data[49]	0		Decimal	SINT
weqube:I.Data[50]	0		Decimal	SINT
weqube:I.Data[51]	0		Decimal	SINT
weqube:I.Data[52]	0		Decimal	SINT
weqube:O	{...}	{...}		_04BB:weqube_7CE0...
weqube:O.Data	{...}	{...}	Decimal	SINT[2]
weqube:O.Data[0]	2		Decimal	SINT
weqube:O.Data[1]	0		Decimal	SINT

Properties	
Name	weqube:O.Data[0]
Type	Base
Data Type	SINT
Scope	PLC
External Access	Read/Write
Style	Decimal
Constant	No
Value	2

0 Errors 0 Warnings 0 Messages

NOTE!

- After loading the project, the Smart Camera initializes all results (e.g. run counter is reset to 0).
- The project has finished loading when the result of the project number is available on the PLC.
- It is not allowed to send several commands at the same time (e.g. trigger and load project commands).
- After a load project command is sent from the PLC to the Smart Camera, wait until the result of the project number is available on the PLC before sending the next command.



NOTE!

After starting the Smart Camera weQube, the startup project defined in the global properties is loaded. It is possible to send a command to load another project from the PLC to the Smart Camera before the booting process is finished, but it is necessary to wait until the Smart Camera responds with the correct project number before sending the first trigger command.

4.3 User-Defined Process Data

All other EtherNet/IP values in the uniVision project are process data. Process data is sent from the device to the PLC and vice versa. Details are available in the Smart Camera settings (see section “5.2 Device Industrial Ethernet”, page 24).

NOTE!

Compared to commands and status data that are updated continuously, process data is only evaluated and sent when an image is executed because of a trigger signal.

5. Smart Camera Settings

For EtherNet/IP communication, the following steps are necessary at the Smart Camera.

NOTE!

- The Smart Camera weQube supports EtherNet/IP functionality starting with the Smart Camera firmware version 2.3.0 and the uniVision software version 2.3.0.
- Not all Smart Cameras support EtherNet/IP communication. Check the technical data on the wenglor website for details about every Smart Camera version.

5.1 Set Up uniVision Projects

The following steps are necessary in order to create a communication between the Smart Camera and the PLC:

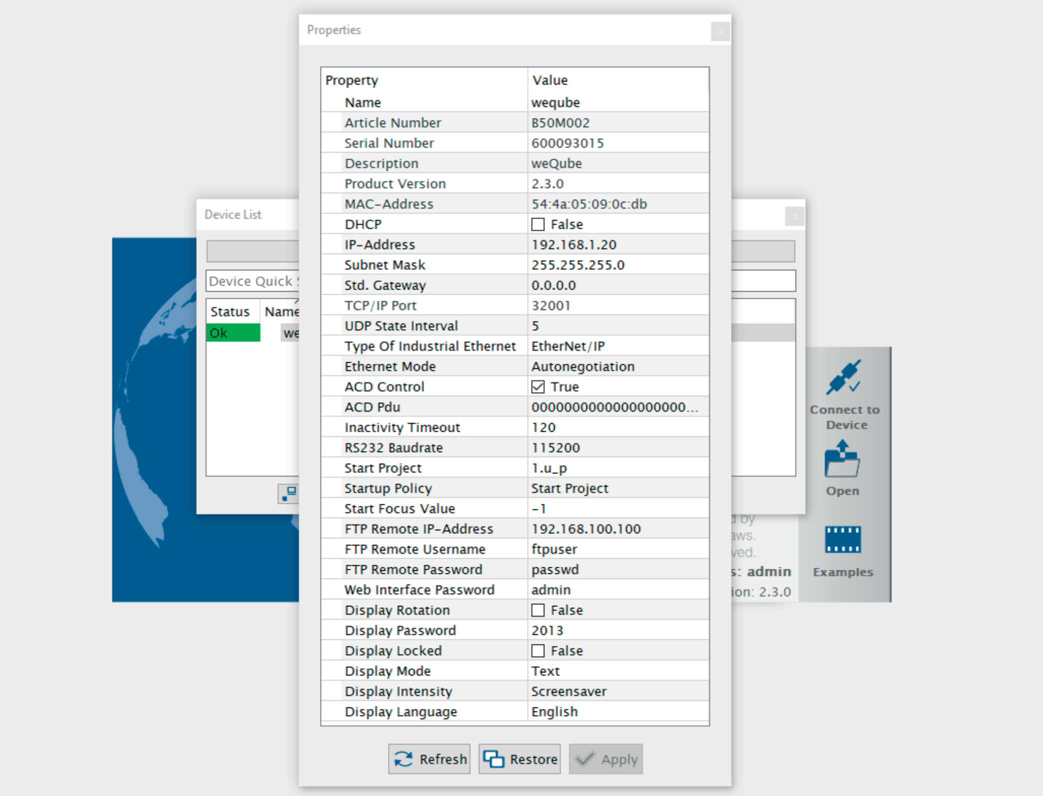
1. Install and open the software uniVision for Windows.
2. Set up the network configuration and the device name of the Smart Camera via the software uniVision.

NOTE!

It is recommended to directly allocate the network configuration and device name to the Smart Camera that is to be used in the EtherNet/IP network later.

NOTE!

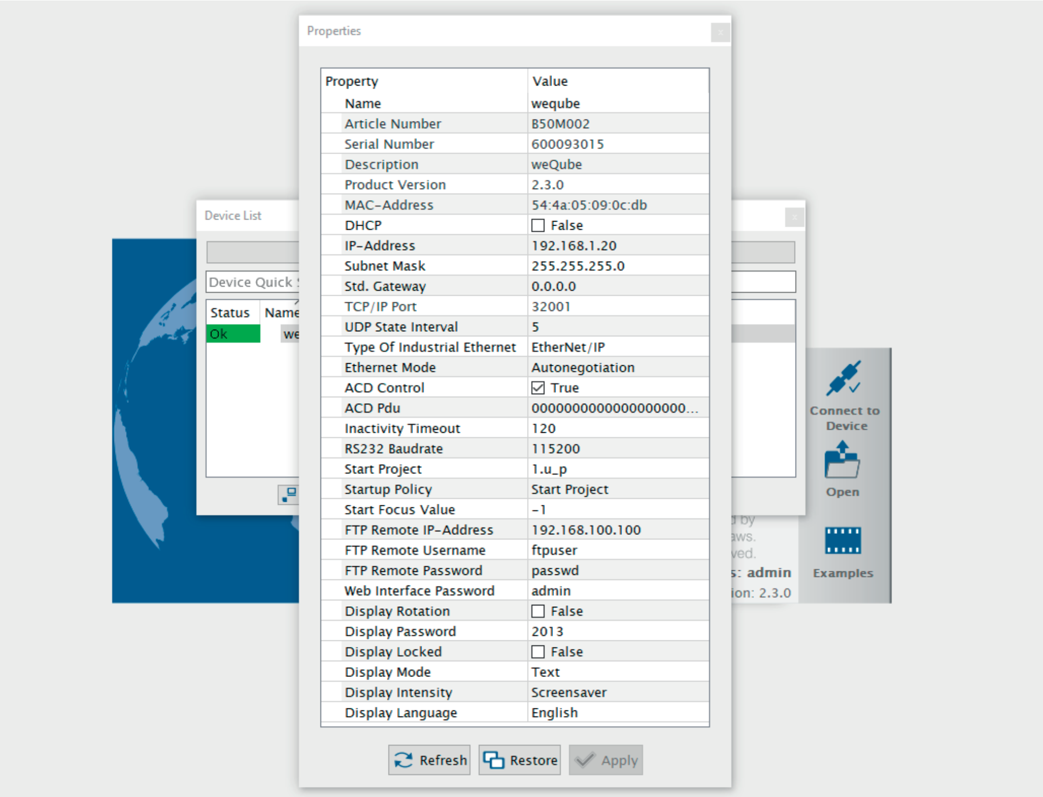
It is possible to use a static network configuration or to activate DHCP at the Smart Camera. In case of activated DHCP, a DHCP server (e.g. BOOTP) is needed in the network so that the Smart Camera gets a valid network configuration (see section "6. EtherNet/IP Network Configuration of the Smart Camera" on page 30). If no DHCP server is available, the Smart Camera will not boot completely. The DHCP setting can be disabled temporarily by pressing the key in the middle of the OLED display on the Smart Camera during start-up.



Property	Value
Name	weqube
Article Number	B50M002
Serial Number	600093015
Description	weQube
Product Version	2.3.0
MAC-Address	54:4a:05:09:0c:db
DHCP	□ False
IP-Address	192.168.1.20
Subnet Mask	255.255.255.0
Std. Gateway	0.0.0.0
TCP/IP Port	32001
UDP State Interval	5
Type Of Industrial Ethernet	EtherNet/IP
Ethernet Mode	Autonegotiation
ACD Control	☑ True
ACD Pdu	0000000000000000000...
Inactivity Timeout	120
RS232 Baudrate	115200
Start Project	1.u_p
Startup Policy	Start Project
Start Focus Value	-1
FTP Remote IP-Address	192.168.100.100
FTP Remote Username	ftpuser
FTP Remote Password	passwd
Web Interface Password	admin
Display Rotation	□ False
Display Password	2013
Display Locked	□ False
Display Mode	Text
Display Intensity	Screensaver
Display Language	English

3. Connect to the Smart Camera.
4. Edit and save uniVision projects on the Smart Camera. All projects must contain Device Industrial Ethernet (see section “5.2 Device Industrial Ethernet”, page 24).

5. Open the Properties of the Smart Camera in the Device List to set up the startup behavior. It is necessary to select a valid startup project with a suitable EtherNet/IP configuration.
   It is also necessary to set “Type of Industrial Ethernet” to “EtherNet/IP” for EtherNet/IP communication (Default: Profinet). Furthermore, you can select if the Address Conflict Detection for EtherNet/IP should be activated or deactivated. Reboot the Smart Camera to apply the settings.

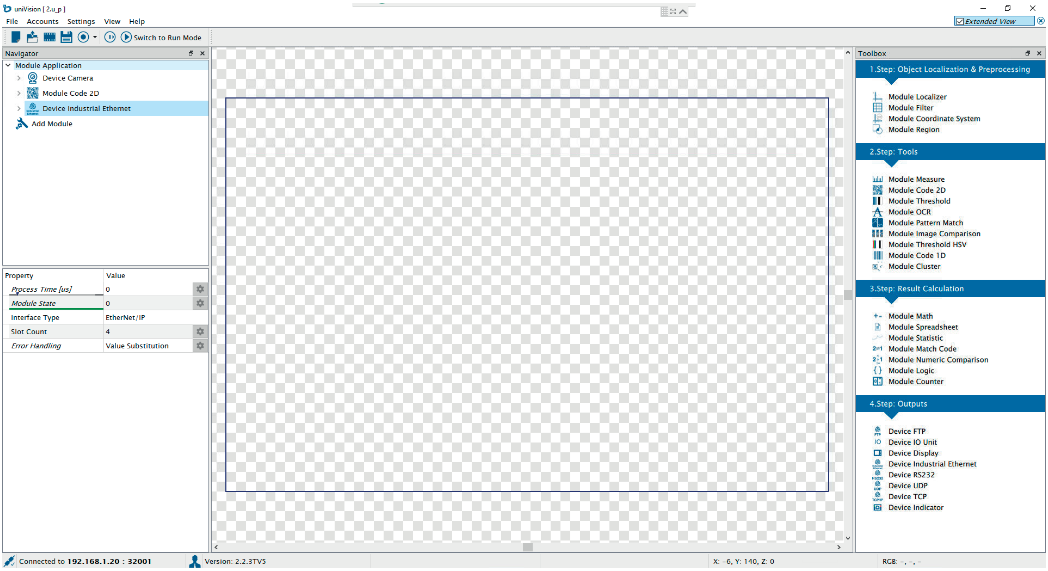


Property	Value
Name	weqube
Article Number	B50M002
Serial Number	600093015
Description	weQube
Product Version	2.3.0
MAC-Address	54:4a:05:09:0c:db
DHCP	☐ False
IP-Address	192.168.1.20
Subnet Mask	255.255.255.0
Std. Gateway	0.0.0.0
TCP/IP Port	32001
UDP State Interval	5
Type Of Industrial Ethernet	EtherNet/IP
Ethernet Mode	Autonegotiation
ACD Control	☑ True
ACD Pdu	0000000000000000000...
Inactivity Timeout	120
RS232 Baudrate	115200
Start Project	1.u_p
Startup Policy	Start Project
Start Focus Value	-1
FTP Remote IP-Address	192.168.100.100
FTP Remote Username	ftpuser
FTP Remote Password	passwd
Web Interface Password	admin
Display Rotation	☐ False
Display Password	2013
Display Locked	☐ False
Display Mode	Text
Display Intensity	Screensaver
Display Language	English

NOTE!
Details about all listed steps are explained in the uniVision software manual.

5.2 Device Industrial Ethernet

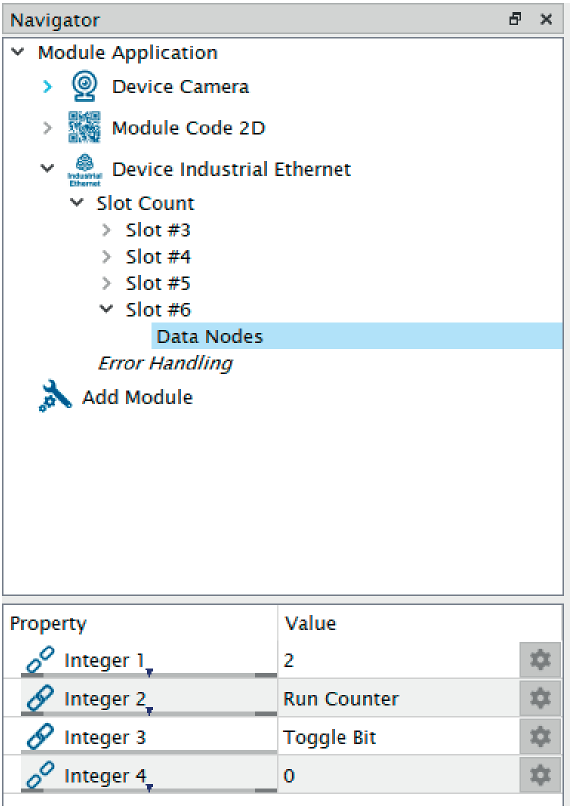
Add Device Industrial Ethernet to the uniVision projects from the toolbox to configure the flexible input and output data. The Smart Camera supports up to four slots to send or receive process data. With one of the flexible slots, it is also possible to send the trigger command to the Smart Camera (see section “4.2.1 Trigger Command”, page 11).



Property	Value
Process Time [us]	0
Module State	0
Interface Type	EtherNet/IP
Slot Count	4
Error Handling	Value Substitution

NOTE!

- Compared to commands and status data that are updated continuously, process data is only evaluated and sent when an image is evaluated because of a trigger signal.
- Only within one slot is the process data sent and received consistently in one EtherNet/IP cycle. The process data from different slots of the Smart Camera to the PLC may be updated in different EtherNet/IP cycles. Use the last slot to make sure that all results are refreshed, e.g. by linking the run counter or the toggle bit to a value from the last slot.
- Furthermore, it is recommended to also send the project number as fixed value in the last slot in order to verify on the PLC side that the project loading process is finished.
- After changing the number of slots or the slot configuration, the Smart Camera must be restarted with a suitable start-up behavior (e.g. fixed start-up project) in order for the settings to be applied correctly.
- The number of slots and the slot configuration must be identical in all uniVision projects in the Smart Camera in order for the project change to be possible from the PLC.



Property	Value
Integer 1	2
Integer 2	Run Counter
Integer 3	Toggle Bit
Integer 4	0

Property	Description
Process Time [us]	Time in μs for processing the module.
Module State	Indicates the status of the module: • 0: No error • Value different to 0: Error (Details about the error code are available in the uniVision software manual)
Interface Type	Indicates the interface type.
Slot Count	Number of flexible slots (Slots 3 – 6). **NOTE!** • By default, the number of flexible slots is set to 0. • Fours flexible slots can be configured at a maximum.
Error handling	If any process data is in error state, it is substituted by a user-defined replacement value (see section "5.2.2 Error Handling", page 29).

5.2.1 Slots

Set up the configuration for every EtherNet/IP slot.

NOTE!
Activate the extended view in the uniVision software in order to be able to see all values (especially the module ID and direction).

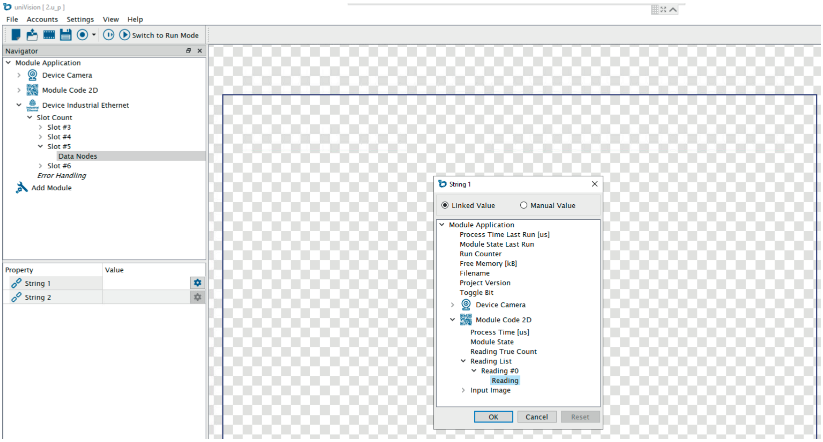
Property	Description
Process Time [us]	Time in μs for processing the module.
Module State	Indicates the status of the module: • 0: No error • Value different to 0: Error (Details about the error code are available in the uniVision software manual)
Slot Number	Indicates the slot number.
Module ID	Indicates the Module ID.
Submodule ID	Indicates the Submodule ID.
Data Size	Indicates the data size in bytes.
Direction	Indicates the direction in which data is sent: • Device to PLC: Values are sent from the Smart Camera to the PLC. • PLC to Device: Values are sent from the PLC to the Smart Camera.
Data Nodes	Indicates the number of data nodes.
Data Type	Select the data type of the slot. The Smart Camera supports the following data types: • 16 Byte Input (4 DINT) • 16 Byte Input (4 REAL) • 1 Byte Input (8 BOOL) • 128 Byte Input (2 CHAR) • 16 Byte Output (4 DINT) • 16 Byte Output (4 REAL) • 1 Byte Output (8 BOOL) • 128 Byte Output (2 CHAR) • 32 Byte Input (2 CHAR) • 64 Byte Input (2 CHAR) • 256 Byte Input (2 CHAR) • 32 Byte Output (2 CHAR) • 64 Byte Output (2 CHAR) • 256 Byte Output (2 CHAR) **NOTE!** • Use BOOL to send or receive true/false results (e.g. toggle bit). • Use REAL to send or receive numbers with positions after decimal point (e.g. x value of a found point). • Use DINT to send or receive numbers without positions after decimal point (e.g. pixel count value of Module Threshold). • Use CHAR to send or receive text information (e.g. code result).

Property	Description
Data Type	Linking results to the different data types works as follows: • BOOL (output) – Link BOOL result: Returns true or false depending on value of bool – Link DINT or REAL result: Returns true if the current value is within thresholds (between the minimum and maximum thresholds) and returns false if the current value is out of tolerance (lower than the minimum or higher than the maximum thresholds) – Link CHAR: Returns true if the text is not empty and returns false if the text is empty. • DINT (output) – Link BOOL result: Returns 0 for bool value false and 1 for bool value true. – Link DINT: Returns current DINT value – Link REAL: Returns a number without decimal places (no rounding!) – Link CHAR: Returns the number of digits of the text • REAL (output) – Link BOOL result: Returns 0 for bool value false and 1 for bool value true. – Link DINT or REAL: Returns a number with decimal places – Link CHAR: Returns the number of digits of the text • CHAR (output) – Link BOOL result: Returns false for bool value false and true for bool value true – Link DINT or REAL: Returns the number – Link CHAR: Returns the text

In the project tree, data nodes appear below the slot.

Double, Integer, BOOL or String (depending on the data type of the slot)

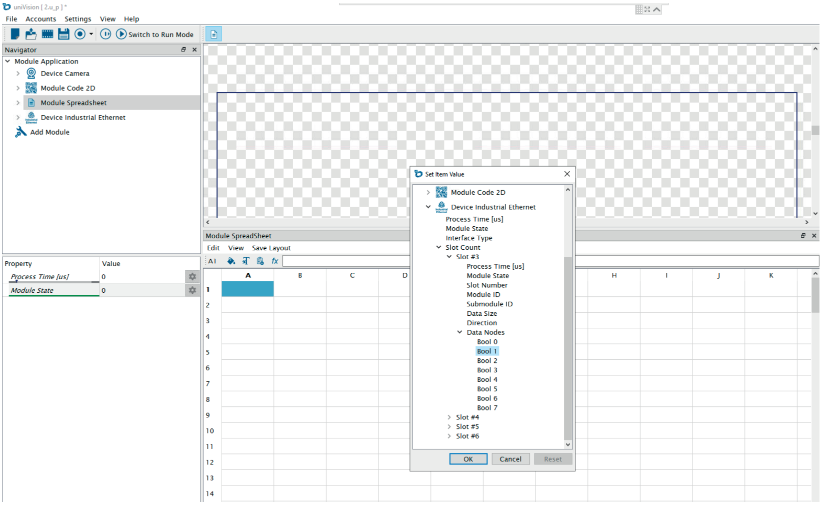
Values (Results) from the device to the PLC can be set to a certain value manually or can be linked with any result of the project (e.g. with the reading result of Module Code 2D).



Values from the PLC to the device are shown as uniVision input values.

NOTE!

- Process data from the PLC to the Smart Camera is only received when an image is evaluated in the Smart Camera because of a trigger signal.
- Process data from the PLC to the Smart Camera is linked as input in other modules (e.g. in the Module Spreadsheet).



5.2.2 Error Handling

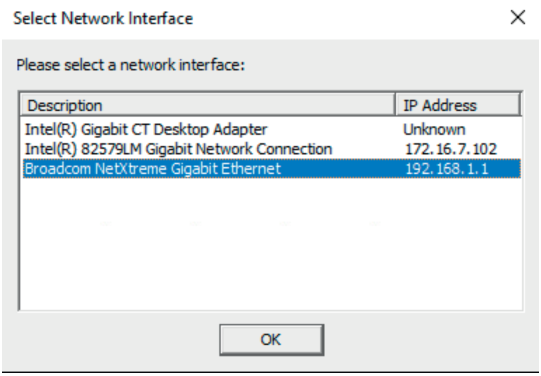
If any process data is in error state, the substitution value can be selected for every data type.

Property	Description
Substitute Bool Types by	If a bool type used in Device Industrial Ethernet is in error state, it is replaced by low or high (Default: low).
Substitute INT Types by	If an INT type used in Device Industrial Ethernet is in error state, it is replaced by any user-defined INT value (Default: 0).
Substitute DOUBLE Types by	If a DOUBLE type 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 type used in Device Industrial Ethernet is in error state, it is replaced by any user-defined STRING value (Default: Error).

6. EtherNet/IP Network Configuration of the Smart Camera

The network settings of the Smart Camera can be set manually or can be allocated via a DHCP server.

In case of DHCP, use e.g. the software BOOTP to allocate the network settings to the Smart Camera:
1. Open software BOOTP.
2. Select the correct LAN adapter and click on OK.

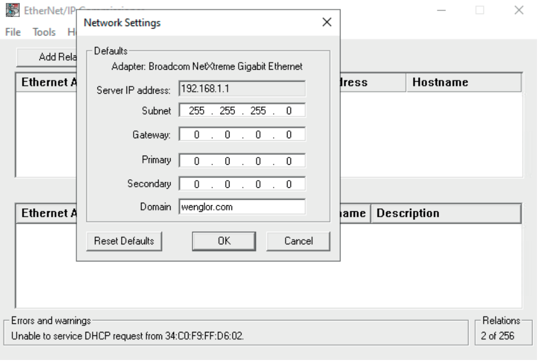


Select Network Interface

Please select a network interface:

Description	IP Address
Intel(R) Gigabit CT Desktop Adapter	Unknown
Intel(R) 82579LM Gigabit Network Connection	172.16.7.102
Broadcom NetXtreme Gigabit Ethernet	192.168.1.1

Click on “Tools” → “Network Settings” to enter the network settings of the PC.



Defaults

Adapter: Broadcom NetXtreme Gigabit Ethernet

Server IP address: 192.168.1.1
Subnet: 255 . 255 . 255 . 0
Gateway: 0 . 0 . 0 . 0
Primary: 0 . 0 . 0 . 0
Secondary: 0 . 0 . 0 . 0
Domain: wenglor.com

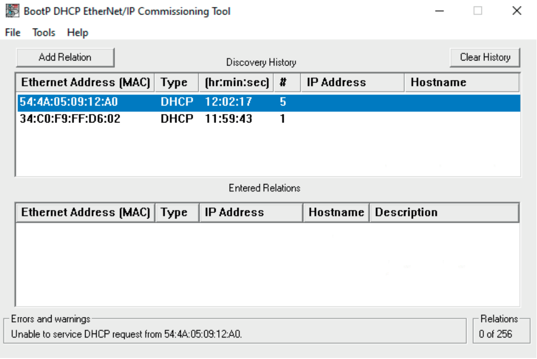
Errors and warnings
Unable to service DHCP request from 34:C0:F9:FF:D6:02.

Relations
2 of 256

All devices with activated DHCP settings connected to the LAN adapter are listed.

3. Select the Smart Camera weQube and click on “Add Relation”.

NOTE!
The MAC address is available on the housing of the Smart Camera.



Discovery History

Ethernet Address (MAC)	Type	(hr:min:sec)	#	IP Address	Hostname
54:4A:05:09:12:A0	DHCP	12:02:17	5		
34:C0:F9:FF:D6:02	DHCP	11:59:43	1		

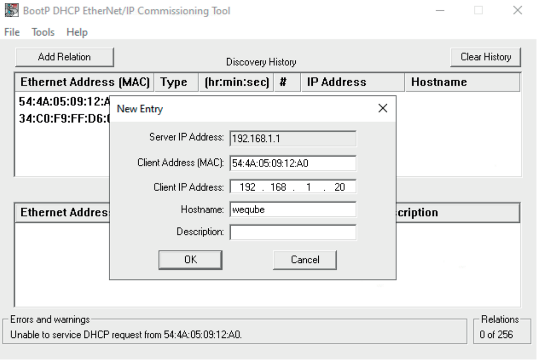
Entered Relations

Ethernet Address (MAC)	Type	IP Address	Hostname	Description

Errors and warnings
Unable to service DHCP request from 54:4A:05:09:12:A0.

Relations
0 of 256

4. Enter IP address and hostname of the Smart Camera and click on OK.



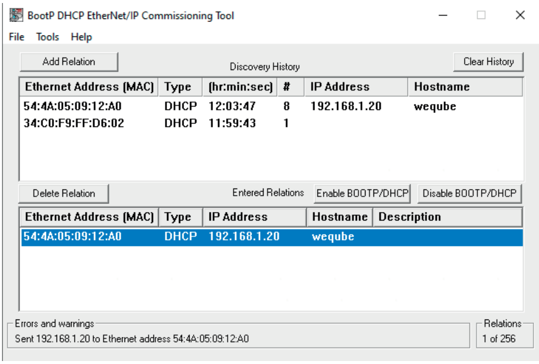
New Entry

Server IP Address: 192.168.1.1
Client Address (MAC): 54:4A:05:09:12:A0
Client IP Address: 192 . 168 . 1 . 20
Hostname: weqube
Description:

Errors and warnings
Unable to service DHCP request from 54:4A:05:09:12:A0.

Relations
0 of 256

5. In order to keep the network settings for the next start-up of the Smart Camera, select weQube and click on "Disable BOOTP/DHCP".



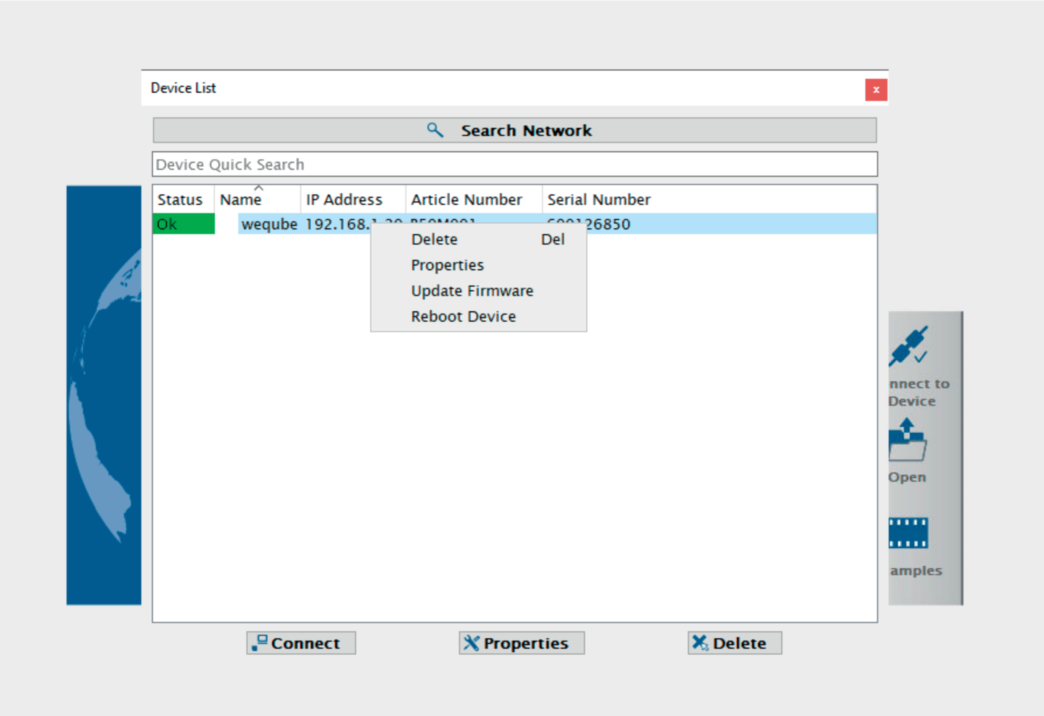
Ethernet Address (MAC)	Type	(hr:min:sec)	#	IP Address	Hostname
54:4A:05:09:12:A0	DHCP	12:03:47	8	192.168.1.20	weqube
34:C0:F9:FF:D6:02	DHCP	11:59:43	1		

Ethernet Address (MAC)	Type	IP Address	Hostname	Description
54:4A:05:09:12:A0	DHCP	192.168.1.20	weqube	

Errors and warnings
Sent 192.168.1.20 to Ethernet address 54:4A:05:09:12:A0

Relations
1 of 256

6. Reboot the Smart Camera to apply the changed network settings (via Software uniVision → Device List).



Status	Name	IP Address	Article Number	Serial Number
Ok	weqube	192.168.		26850

7. PLC Settings at Allen-Bradley PLCs

The following settings are necessary on the Rockwell PLC.

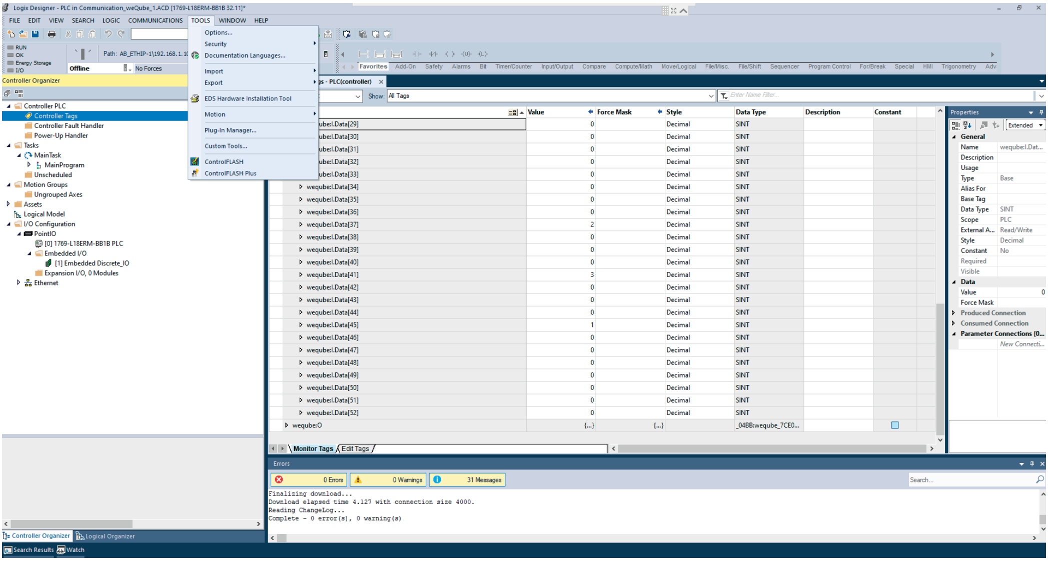
NOTE!
If possible, use the EDS file to integrate the Smart Camera. In case of some old Rockwell PLCs, EDS files are not supported. Consequently the Smart Camera must be integrated as generic device (see section "7.5 Integrate Smart Camera without EDS file" on page 45).

7.1 EDS File

The EDS file is available on the wenglor website in the Download section of the Smart Camera. Download the EDS file, unzip the file and install it on the PLC.

NOTE!
After downloading the zip file, please unzip the file before installing it on the PLC.

In the software Studio 5000 Logix Designer V32, the EDS file is added via "TOOLS" → "EDS Hardware Installation Tool".



Logix Designer - PLC in Communication_weQube_1.ACD [1769-L18ERM-BB1B 32.11]*

Path: AB_ETHIP-1\192.168.1.1

gs - PLC(controller)

Show: All Tags

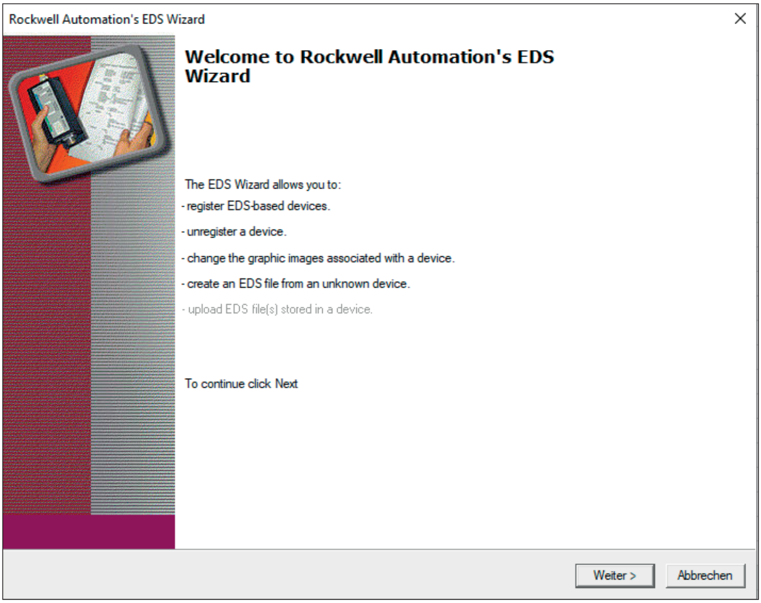
	Value	Force Mask	Style	Data Type	Description	Constant
ube:I.Data[29]	0		Decimal	SINT		
ube:I.Data[30]	0		Decimal	SINT		
ube:I.Data[31]	0		Decimal	SINT		
ube:I.Data[32]	0		Decimal	SINT		
ube:I.Data[33]	0		Decimal	SINT		
weqube:I.Data[34]	0		Decimal	SINT		
weqube:I.Data[35]	0		Decimal	SINT		
weqube:I.Data[36]	0		Decimal	SINT		
weqube:I.Data[37]	2		Decimal	SINT		
weqube:I.Data[38]	0		Decimal	SINT		
weqube:I.Data[39]	0		Decimal	SINT		
weqube:I.Data[40]	0		Decimal	SINT		
weqube:I.Data[41]	3		Decimal	SINT		
weqube:I.Data[42]	0		Decimal	SINT		
weqube:I.Data[43]	0		Decimal	SINT		
weqube:I.Data[44]	0		Decimal	SINT		
weqube:I.Data[45]	1		Decimal	SINT		
weqube:I.Data[46]	0		Decimal	SINT		
weqube:I.Data[47]	0		Decimal	SINT		
weqube:I.Data[48]	0		Decimal	SINT		
weqube:I.Data[49]	0		Decimal	SINT		
weqube:I.Data[50]	0		Decimal	SINT		
weqube:I.Data[51]	0		Decimal	SINT		
weqube:I.Data[52]	0		Decimal	SINT		
weqube:O	{...}	{...}		_04BB:weqube_7CE0...		□

Properties

General	
Name	weqube:I.Dat...
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	SINT
Scope	PLC
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	0
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections (0...	
	New Connecti...

```
Finalizing download...
Download elapsed time 4.127 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

The EDS wizard starts.



Rockwell Automation's EDS Wizard

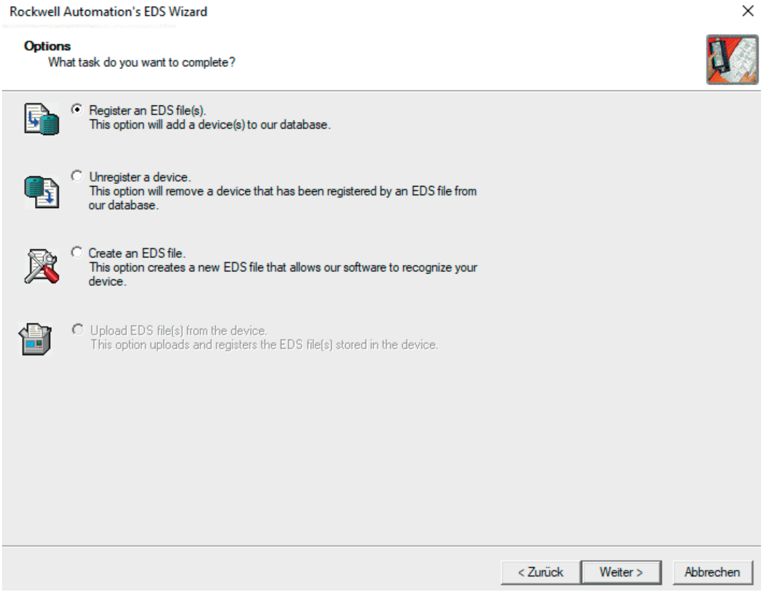
Welcome to Rockwell Automation's EDS Wizard

The EDS Wizard allows you to:

- register EDS-based devices.
- unregister a device.
- change the graphic images associated with a device.
- create an EDS file from an unknown device.
- upload EDS file(s) stored in a device.

To continue click Next

Select “Register an EDS file(s)”.



Rockwell Automation's EDS Wizard

Options
What task do you want to complete?

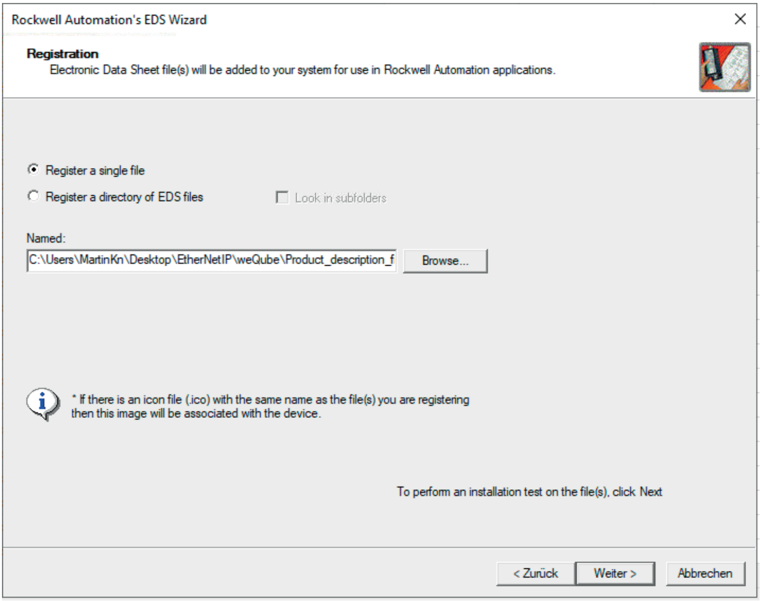
● Register an EDS file(s).
This option will add a device(s) to our database.

○ Unregister a device.
This option will remove a device that has been registered by an EDS file from our database.

○ Create an EDS file.
This option creates a new EDS file that allows our software to recognize your device.

○ Upload EDS file(s) from the device.
This option uploads and registers the EDS file(s) stored in the device.

Select the path of the EDS file.



Rockwell Automation's EDS Wizard

Registration

Electronic Data Sheet file(s) will be added to your system for use in Rockwell Automation applications.

● Register a single file

○ Register a directory of EDS files

□ Look in subfolders

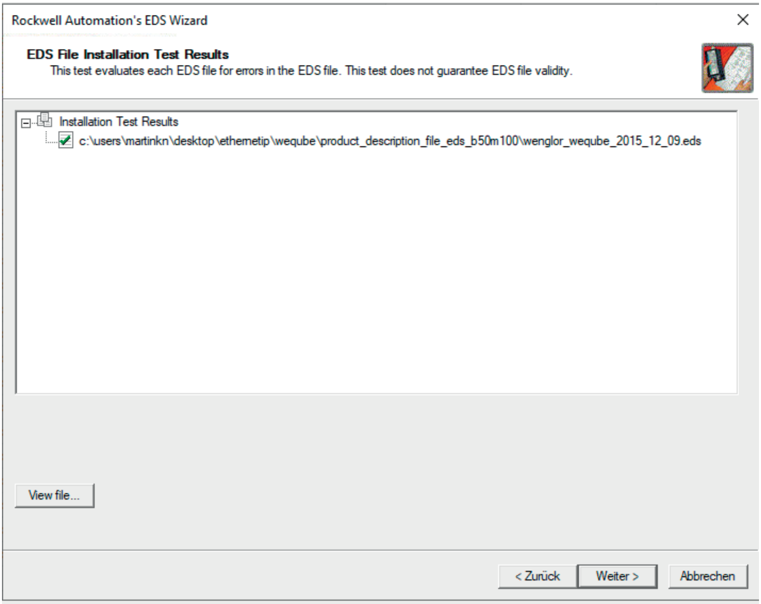
Named:

C:\Users\MartinKn\Desktop\EtherNetIP\weQube\Product_description_f

* If there is an icon file (.ico) with the same name as the file(s) you are registering then this image will be associated with the device.

To perform an installation test on the file(s), click Next

Add the EDS file to the project. It is also possible to view the file.



Rockwell Automation's EDS Wizard

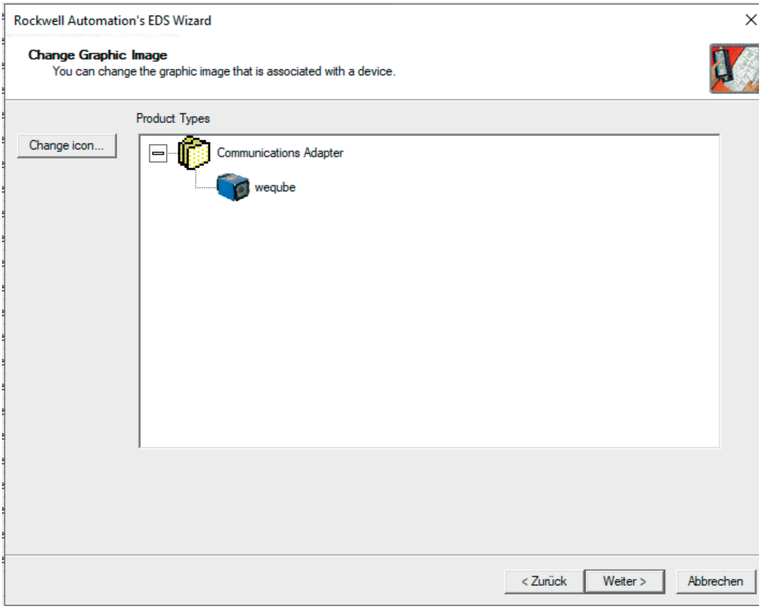
EDS File Installation Test Results

This test evaluates each EDS file for errors in the EDS file. This test does not guarantee EDS file validity.

Installation Test Results

☑ c:\users\martinkn\desktop\ethernetip\weqube\product_description_file_eds_b50m100\wenglor_weqube_2015_12_09.eds

The image associated with the device is shown.



Rockwell Automation's EDS Wizard

Change Graphic Image

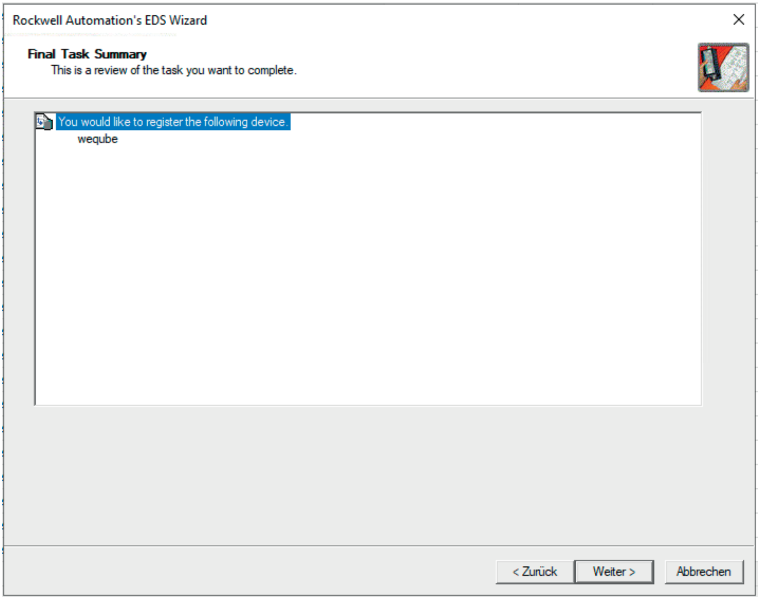
You can change the graphic image that is associated with a device.

Product Types

Communications Adapter

weqube

Add the selected device.



Rockwell Automation's EDS Wizard

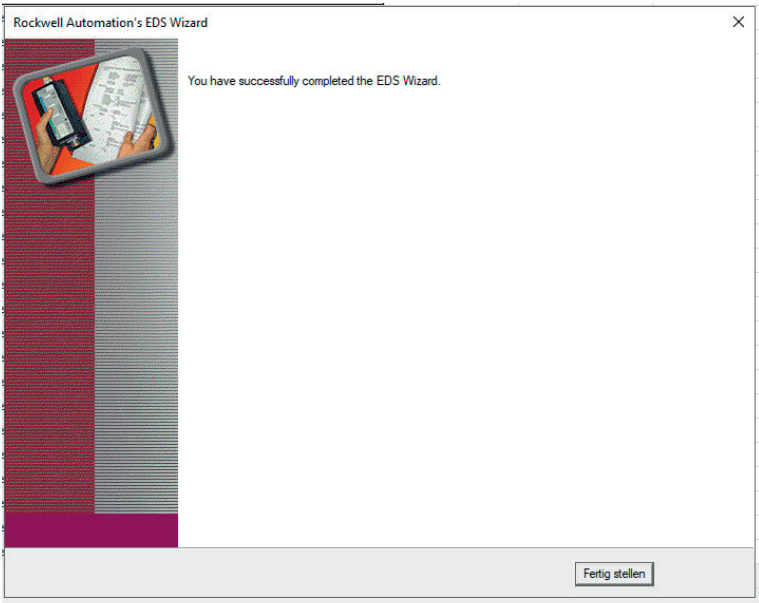
Final Task Summary

This is a review of the task you want to complete.

You would like to register the following device.

weqube

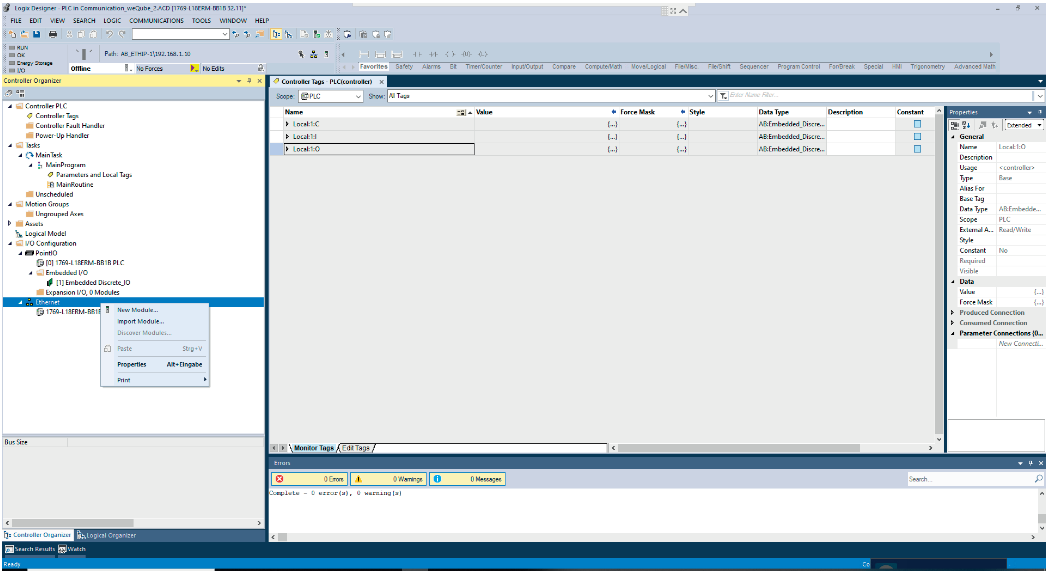
The installation of the EDS file is finished.



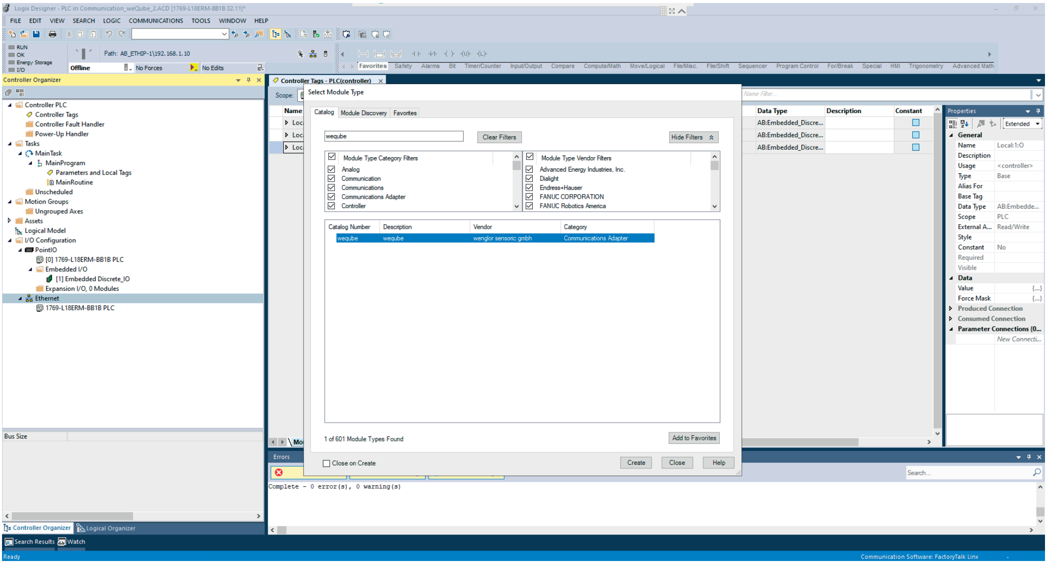
You have successfully completed the EDS Wizard.

7.2 Add Smart Camera to PLC Network

Open the context menu at “Ethernet” with a right click and select “New Module…” to add the Smart Camera to the PLC network.

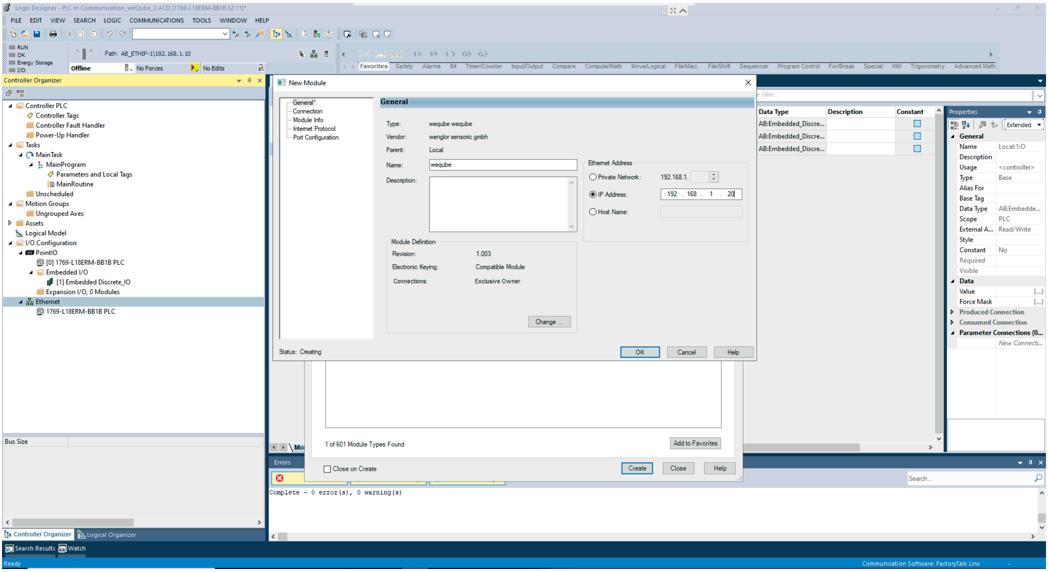


Then search for weqube in the catalog. Select weqube and click on “Create”.



Enter the device name and the network configuration of the device. In the example, the IP address 192.168.1.20 and the name weqube is used.

NOTE!
It is necessary to use the same network configuration and the same device name on the PLC side and on the software for uniVision side.



New Module

- General*
- Connection
- Module Info
- Internet Protocol
- Port Configuration

General

Type: weqube weqube

Vendor: wenglor sensoric gmbh

Parent: Local

Name: weqube

Description:

Ethernet Address

○ Private Network: 192.168.1.

● IP Address: 192 . 168 . 1 . 20

○ Host Name:

Module Definition

Revision: 1.003

Electronic Keying: Compatible Module

Connections: Exclusive Owner

Change ...

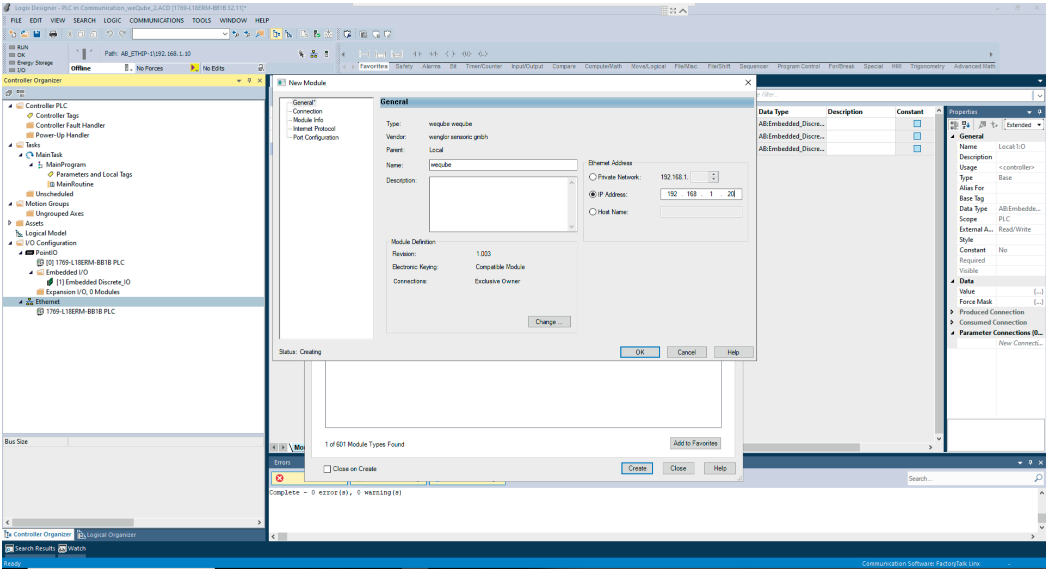
Status: Creating

OK Cancel Help

```
Complete - 0 error(s), 0 warning(s)
```

7.3 Configure Input and Output Data

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



New Module

General

Type: weqube weqube

Vendor: wenglor sensoric gmbh

Parent: Local

Name: weqube

Description:

Ethernet Address

○ Private Network: 192.168.1.

● IP Address: 192 . 168 . 1 . 20

○ Host Name:

Module Definition

Revision: 1.003

Electronic Keying: Compatible Module

Connections: Exclusive Owner

Status: Creating

1 of 601 Module Types Found

□ Close on Create

```
Complete - 0 error(s), 0 warning(s)
```

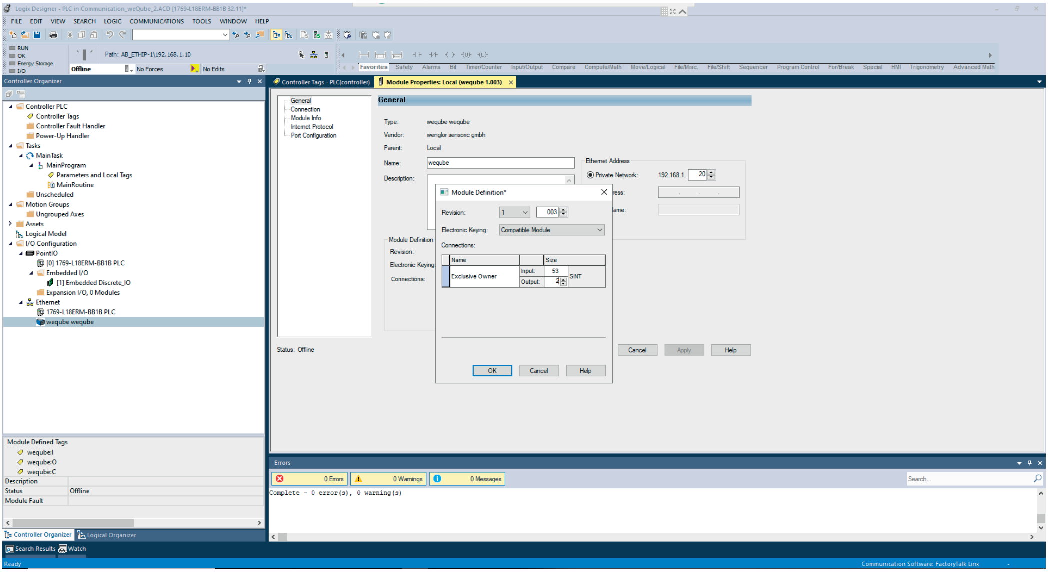
Properties	
General	
Name	Local:1:O
Description	
Usage	<controller>
Type	Base
Alias For	
Base Tag	
Data Type	AB:Embedde...
Scope	PLC
External A...	Read/Write
Style	
Constant	No
Required	
Visible	
Data	
Value	{...}
Force Mask	{...}

Adjust the input and output size according to the current uniVision project configuration.

NOTE!
Slot 1 (1 Byte Output for project number) and Slot 2 (4 Bytes Input for status) are always present and must be added to the user-defined slots 3-6.

The example (Input: 53 Bytes and Output: 2 Bytes) works for the following slot configuration:

- Slot 1 (fix): 1 Byte Output (Project number)
- Slot 2 (fix): 4 Bytes Input (Status)
- Slot 3: 1 Byte Output (8 BOOL)
- Slot 4: 1 Byte Input (8 BOOL)
- Slot 5: 32 Byte Input (2 CHAR)
- Slot 6: 16 Byte Input (4 DINT)



Path: AB_ETHIP-1\192.168.1.10

Offline · No Forces · No Edits

Type: weqube weqube

Vendor: wenglor sensoric gmbh

Parent: Local

Name: weqube

Ethernet Address: ● Private Network: 192.168.1. 20

Revision: 1 003

Electronic Keying: Compatible Module

Connections:

Name		Size	
Exclusive Owner	Input:	53	SINT
	Output:	2	

Status: Offline

Description	
Status	Offline
Module Fault	

0 Errors · 0 Warnings · 0 Messages

```
Complete - 0 error(s), 0 warning(s)
```

Click on Apply and on OK.

Open the “Controller Tags”, select weqube:C and set the Module IDs according to the uniVision project configuration.

The Module IDs of slot 1 and slot 2 are set automatically. The Module IDs of all other slots must be set individually depending on the uniVision project settings.

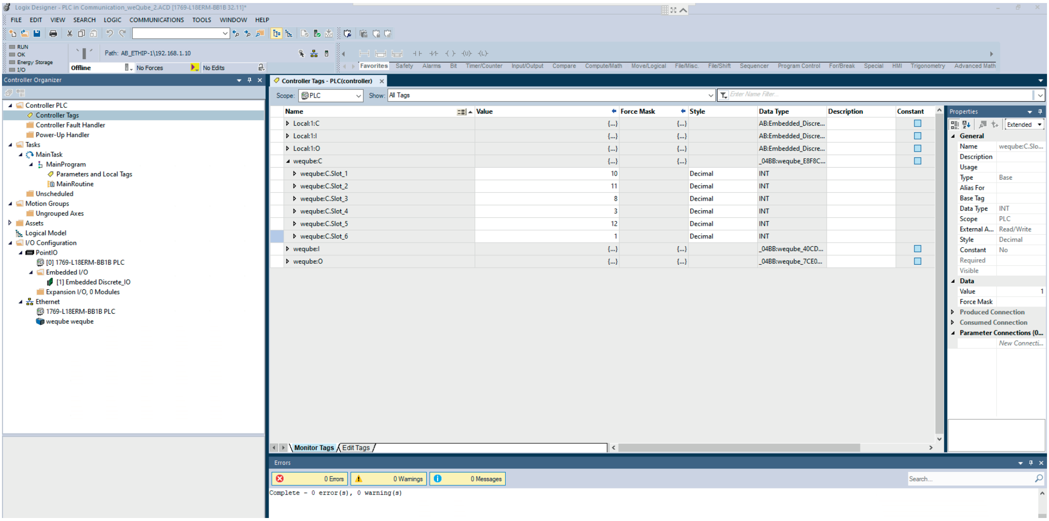
NOTE!
Connect to the Smart Camera via uniVision software in order to check the current Module IDs of the uniVision project (see section "5.2.1 Slots" on page 26)

List of all available Module IDs:

Module ID Hex (Dec)	Description	Direction	Data Length	Representation
0x0000 (0)	Undefined	Undefined	Undefined	Undefined
0x0001 (1)	Process data	weQube → PLC	16 bytes	4 x 4 bytes integer
0x0002 (2)	Process data	weQube → PLC	16 bytes	4 x 4 bytes real
0x0003 (3)	Process data	weQube → PLC	1 byte	8 x 1 bit bool
0x0004 (4)	Process data	weQube → PLC	128 bytes	2 x 64 bytes string
0x0006 (6)	Process data	PLC → weQube	16 bytes	4 x 4 bytes integer
0x0007 (7)	Process data	PLC → weQube	16 bytes	4 x 4 bytes real
0x0008 (8)	Process data	PLC → weQube	1 byte	8 x 1 bit bool
0x0009 (9)	Process data	PLC → weQube	128 bytes	2 x 64 bytes string
0x000A (10)	Project number	PLC → weQube	1 byte	1 x 1 Byte
0x000B (11)	Device State	weQube → PLC	4 bytes	1 x 4 bytes integer
0x000C (12)	Process data	weQube → PLC	32 bytes	2 x 16 bytes string
0x000D (13)	Process data	weQube → PLC	64 bytes	2 x 32 bytes string
0x000E (14)	Process data	weQube → PLC	256 bytes	2 x 128 bytes string
0x0010 (16)	Process data	PLC → weQube	32 bytes	2 x 16 bytes string
0x0011 (17)	Process data	PLC → weQube	64 bytes	2 x 32 bytes string
0x0012 (18)	Process data	PLC → weQube	256 bytes	2 x 128 bytes string

In the example, the following Module IDs must be set:

- Slot 1: 10 (fix)
- Slot 2: 11 (fix)
- Slot 3: 8
- Slot 4: 3
- Slot 5: 12
- Slot 6: 1



Path: AB_ETHIP-1\192.168.1.10

Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type	Description	Constant
Local:1:C	{...}	{...}		AB:Embedded_Discre...		□
Local:1:I	{...}	{...}		AB:Embedded_Discre...		□
Local:1:O	{...}	{...}		AB:Embedded_Discre...		□
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...		□
weqube:C.Slot_1	10		Decimal	INT		
weqube:C.Slot_2	11		Decimal	INT		
weqube:C.Slot_3	8		Decimal	INT		
weqube:C.Slot_4	3		Decimal	INT		
weqube:C.Slot_5	12		Decimal	INT		
weqube:C.Slot_6	1		Decimal	INT		
weqube:I	{...}	{...}		_04BB:weqube_40CD...		□
weqube:O	{...}	{...}		_04BB:weqube_7CE0...		□

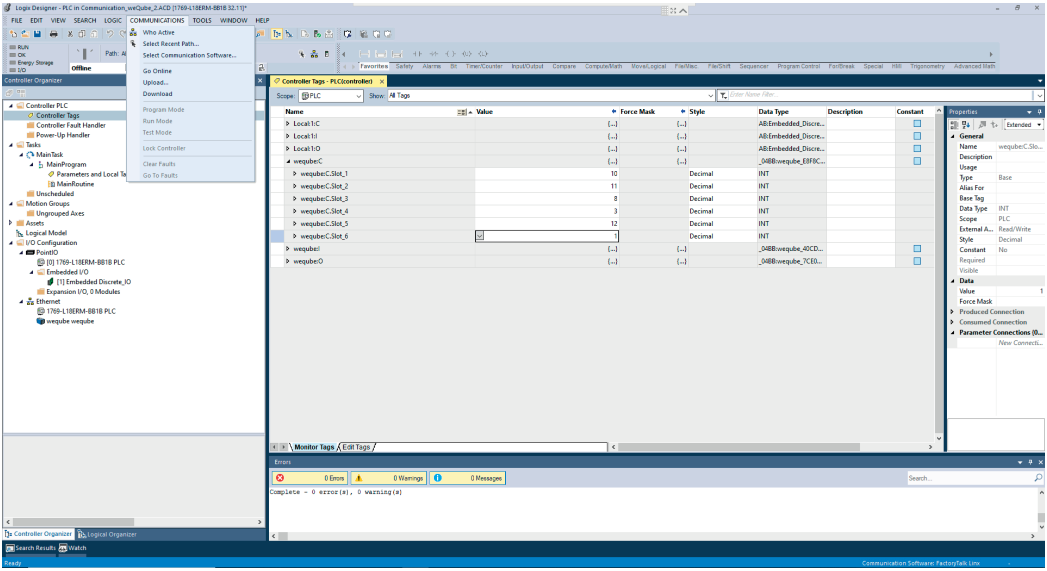
General	
Name	weqube:C.Slo...
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	INT
Scope	PLC
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	1
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections {0...	
	New Connecti...

0 Errors 0 Warnings 0 Messages

```
Complete - 0 error(s), 0 warning(s)
```

7.4 Download Configuration to PLC

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

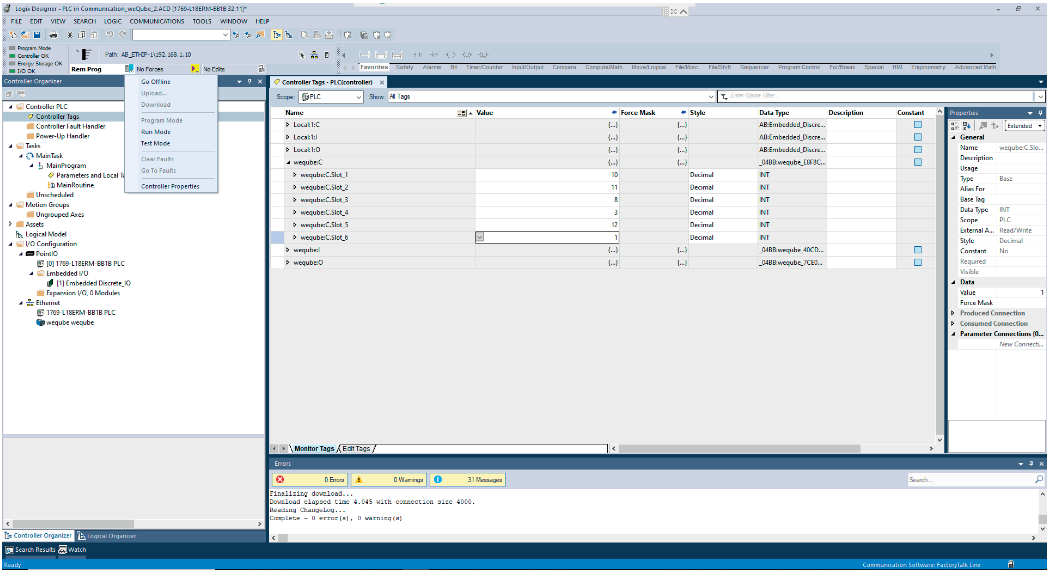


Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type	Description	Constant
Local:1:C	{...}	{...}		AB:Embedded_Discre...		□
Local:1:I	{...}	{...}		AB:Embedded_Discre...		□
Local:1:O	{...}	{...}		AB:Embedded_Discre...		□
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...		□
weqube:C.Slot_1	10		Decimal	INT		
weqube:C.Slot_2	11		Decimal	INT		
weqube:C.Slot_3	8		Decimal	INT		
weqube:C.Slot_4	3		Decimal	INT		
weqube:C.Slot_5	12		Decimal	INT		
weqube:C.Slot_6	1		Decimal	INT		
weqube:I	{...}	{...}		_04BB:weqube_40CD...		□
weqube:O	{...}	{...}		_04BB:weqube_7CE0...		□

```
Complete - 0 error(s), 0 warning(s)
```

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

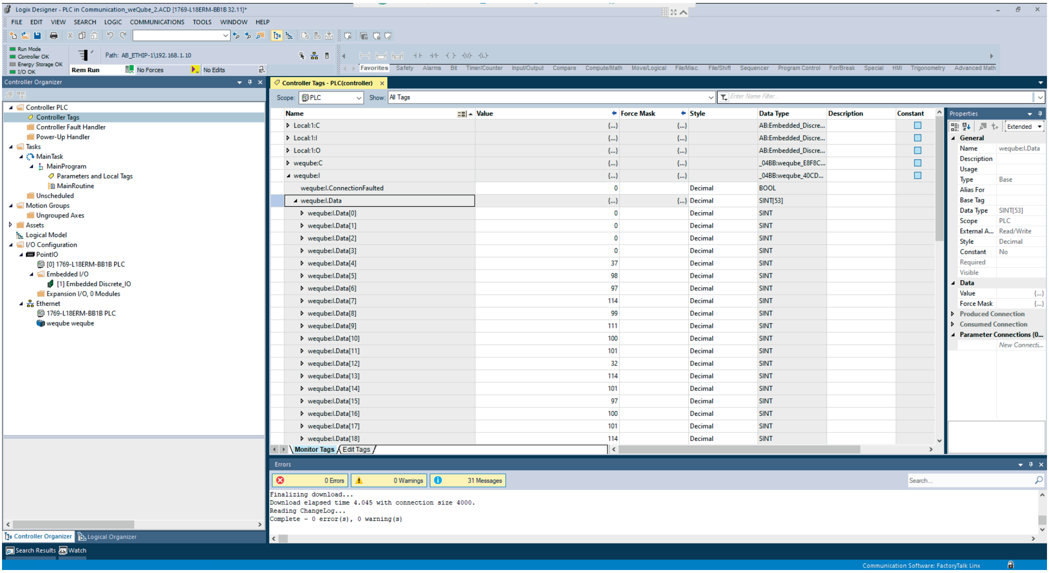

Logix Designer - PLC in Communication_weQube_2.ACD [1769-L18ERM-BB1B 32.11]*
FILE EDIT VIEW SEARCH LOGIC COMMUNICATIONS TOOLS WINDOW HELP
Program Mode
Controller OK
Energy Storage OK
I/O OK
Path: AB_ETHIP-1\192.168.1.10
Rem Prog
No Forces
No Edits
Go Offline
Upload...
Download
Program Mode
Run Mode
Test Mode
Clear Faults
Go To Faults
Controller Properties
Controller Tags - PLC(controller)
Monitor Tags
Edit Tags
Errors
0 Errors
0 Warnings
31 Messages
Communication Software: FactoryTalk Linx
Ready

Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type	Description	Constant
Local:1:C	{...}	{...}		AB:Embedded_Discre...		□
Local:1:I	{...}	{...}		AB:Embedded_Discre...		□
Local:1:O	{...}	{...}		AB:Embedded_Discre...		□
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...		□
weqube:C.Slot_1	10		Decimal	INT		
weqube:C.Slot_2	11		Decimal	INT		
weqube:C.Slot_3	8		Decimal	INT		
weqube:C.Slot_4	3		Decimal	INT		
weqube:C.Slot_5	12		Decimal	INT		
weqube:C.Slot_6	1		Decimal	INT		
weqube:I	{...}	{...}		_04BB:weqube_40CD...		□
weqube:O	{...}	{...}		_04BB:weqube_7CE0...		□

```
Finalizing download...
Download elapsed time 4.045 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

All input and output data of the Smart Camera is available at weqube:I and weqube:O.



Scope: PLC Show: All Tags

Name	Value	Force Mask	Style	Data Type	Description	Constant
Local:1:C	{...}	{...}		AB:Embedded_Discre...		□
Local:1:I	{...}	{...}		AB:Embedded_Discre...		□
Local:1:O	{...}	{...}		AB:Embedded_Discre...		□
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...		□
weqube:I	{...}	{...}		_04BB:weqube_40CD...		□
weqube:I.ConnectionFaulted	0		Decimal	BOOL		
weqube:I.Data	{...}	{...}	Decimal	SINT[53]		
weqube:I.Data[0]	0		Decimal	SINT		
weqube:I.Data[1]	0		Decimal	SINT		
weqube:I.Data[2]	0		Decimal	SINT		
weqube:I.Data[3]	0		Decimal	SINT		
weqube:I.Data[4]	37		Decimal	SINT		
weqube:I.Data[5]	98		Decimal	SINT		
weqube:I.Data[6]	97		Decimal	SINT		
weqube:I.Data[7]	114		Decimal	SINT		
weqube:I.Data[8]	99		Decimal	SINT		
weqube:I.Data[9]	111		Decimal	SINT		
weqube:I.Data[10]	100		Decimal	SINT		
weqube:I.Data[11]	101		Decimal	SINT		
weqube:I.Data[12]	32		Decimal	SINT		
weqube:I.Data[13]	114		Decimal	SINT		
weqube:I.Data[14]	101		Decimal	SINT		
weqube:I.Data[15]	97		Decimal	SINT		
weqube:I.Data[16]	100		Decimal	SINT		
weqube:I.Data[17]	101		Decimal	SINT		
weqube:I.Data[18]	114		Decimal	SINT		

Properties (Extended)

General	
Name	weqube:I.Data
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	SINT[53]
Scope	PLC
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	{...}
Force Mask	{...}
Produced Connection	
Consumed Connection	
Parameter Connections (0...	
	New Connecti...

Errors

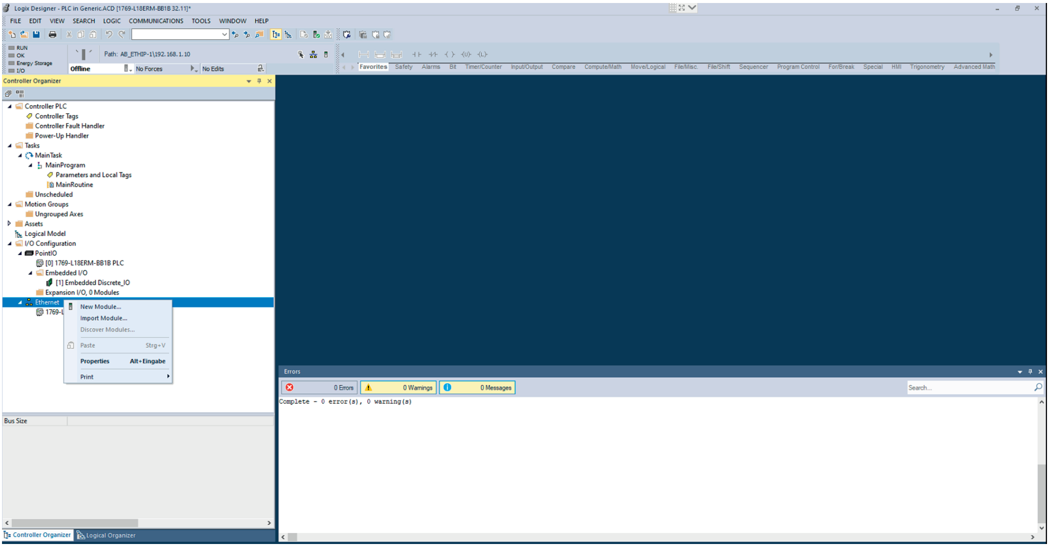
0 Errors 0 Warnings 31 Messages

```
Finalizing download...
Download elapsed time 4.045 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

7.5 Integrate Smart Camera without EDS file

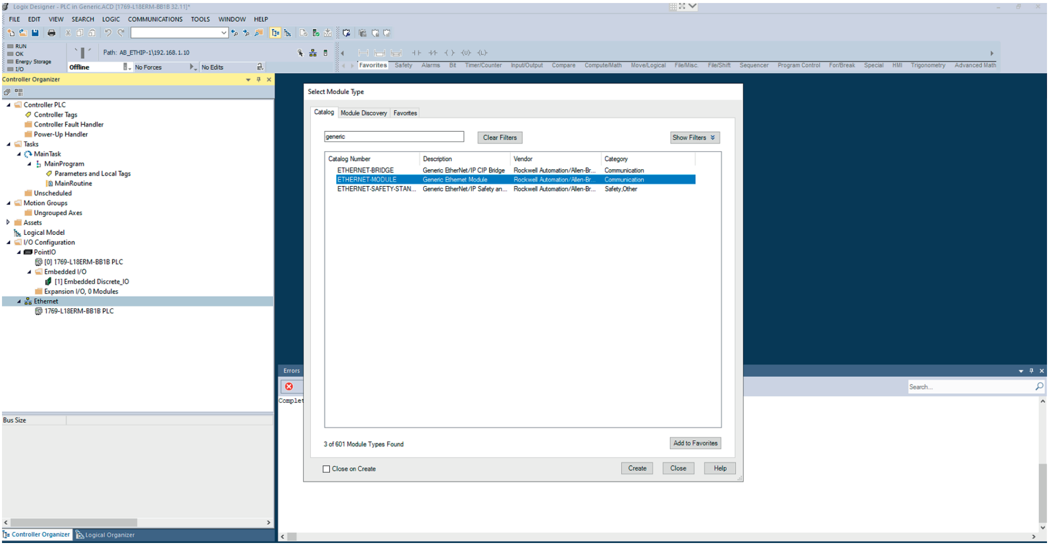
In case of some old Rockwell PLCs, EDS files are not supported. Consequently the Smart Camera must be added as generic device.

Open with a right click the context menu at "Ethernet" and select "New Module..." to add the Smart Camera to the PLC network.



```
Complete - 0 error(s), 0 warning(s)
```

Search for generic and select the "Generic Ethernet Module".



Search: generic

Catalog Number	Description	Vendor	Category
ETHERNET-BRIDGE	Generic EtherNet/IP CIP Bridge	Rockwell Automation/Allen-Br...	Communication
ETHERNET-MODULE	Generic Ethernet Module	Rockwell Automation/Allen-Br...	Communication
ETHERNET-SAFETY-STAN...	Generic EtherNet/IP Safety an...	Rockwell Automation/Allen-Br...	Safety,Other

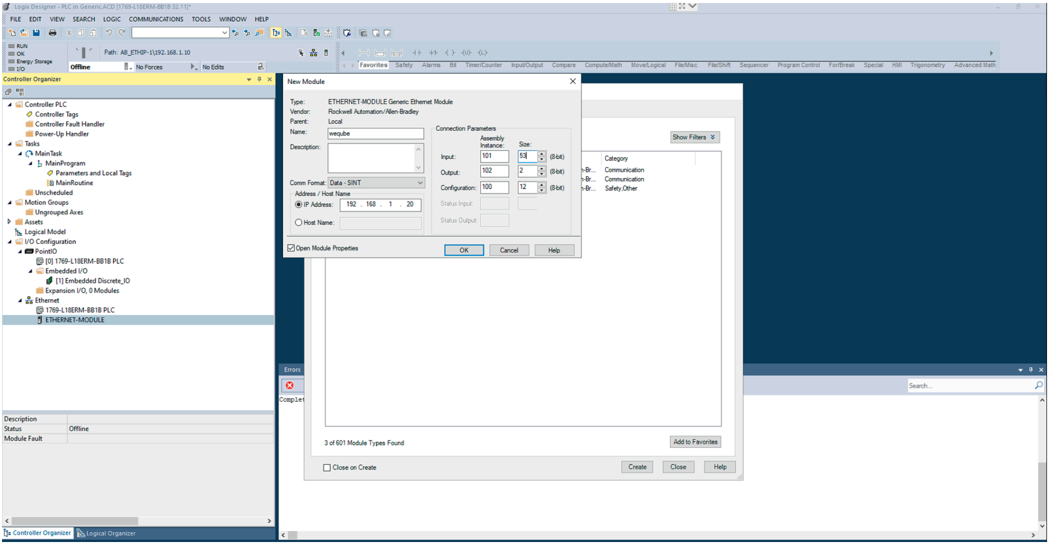
3 of 601 Module Types Found

☐ Close on Create

Enter the name and the IP address of the Smart Camera. Set the Communication Format to "Data - SINT". Furthermore the instances must be configured accordingly:

- Input: Instance 101 with x bytes
- Output: Instance 102 with x bytes
- Configuration: Instance 100 with 12 bytes

NOTE!
The input and output size must fit to the uniVision project configuration! Slot 1 and 2 are always present and must be added to the user-defined slots 3-6.



New Module

Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Rockwell Automation/Allen-Bradley
Parent: Local
Name: weqube
Description:
Comm Format: Data - SINT

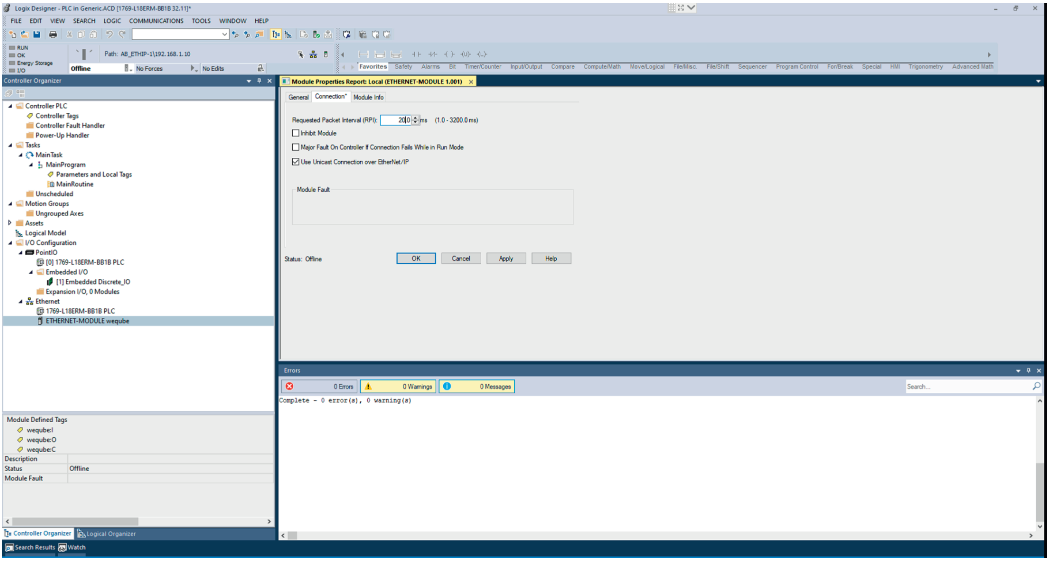
Address / Host Name
● IP Address: 192 . 168 . 1 . 20
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	101	53	(8-bit)
Output:	102	2	(8-bit)
Configuration:	100	12	(8-bit)
Status Input:			
Status Output:			

☑ Open Module Properties

Set the requested packet interval (RPI) to minimum 20 ms and click on ok.



Module Properties Report: Local (ETHERNET-MODULE 1.001)

General | Connection* | Module Info

Requested Packet Interval (RPI): 20.0 ms (1.0 - 3200.0 ms)

☐ Inhibit Module

☐ Major Fault On Controller If Connection Fails While in Run Mode

☑ Use Unicast Connection over EtherNet/IP

Module Fault

Status: Offline

OK | Cancel | Apply | Help

Description	
Status	Offline
Module Fault	

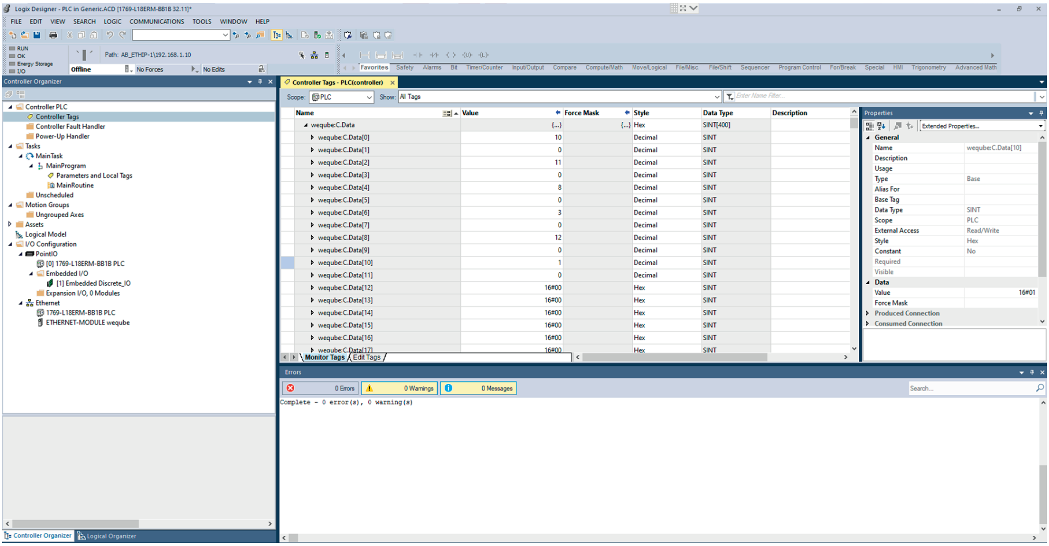
```
Complete - 0 error(s), 0 warning(s)
```

Open the "Controller Tags", select weqube:C and set the Module IDs according to the uniVision project configuration.

NOTE!
Connect to the Smart Camera via uniVision software in order to check the current Module IDs of the uniVision project (see section "5.2.1 Slots" on page 26).

In the example, the following Module IDs are used:

- weqube:C.Data[0]: 10 (Project number)
- weqube:C.Data[2]: 11 (Status)
- weqube:C.Data[4]: 8 (Module ID of slot 3)
- weqube:C.Data[6]: 3 (Module ID of slot 4)
- weqube:C.Data[8]: 12 (Module ID of slot 5)
- weqube:C.Data[10]: 1 (Module ID of slot 6)



Scope: PLC Show: All Tags

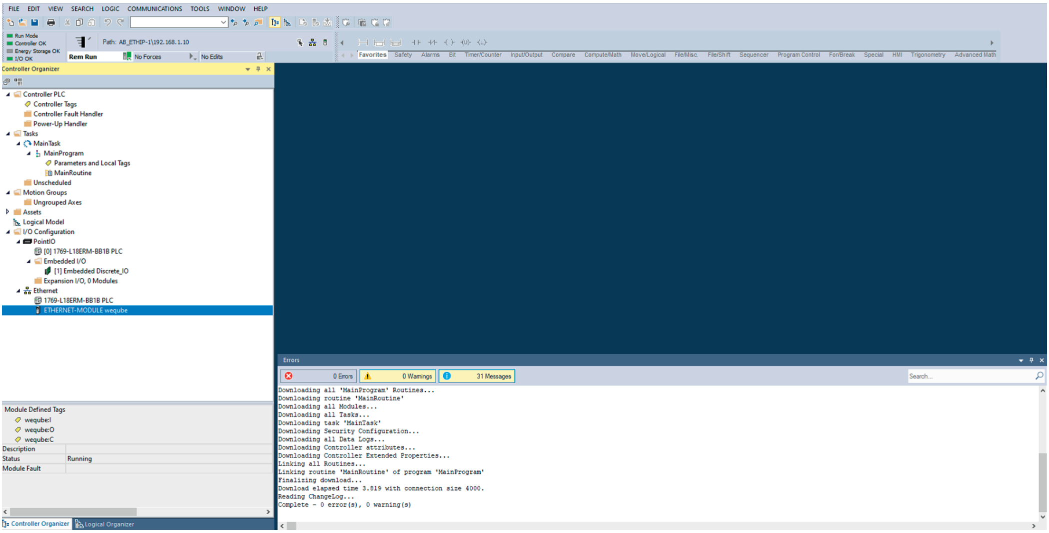
Name	Value	Force Mask	Style	Data Type	Description
weqube:C.Data	{...}	{...}	Hex	SINT[400]	
weqube:C.Data[0]	10		Decimal	SINT	
weqube:C.Data[1]	0		Decimal	SINT	
weqube:C.Data[2]	11		Decimal	SINT	
weqube:C.Data[3]	0		Decimal	SINT	
weqube:C.Data[4]	8		Decimal	SINT	
weqube:C.Data[5]	0		Decimal	SINT	
weqube:C.Data[6]	3		Decimal	SINT	
weqube:C.Data[7]	0		Decimal	SINT	
weqube:C.Data[8]	12		Decimal	SINT	
weqube:C.Data[9]	0		Decimal	SINT	
weqube:C.Data[10]	1		Decimal	SINT	
weqube:C.Data[11]	0		Decimal	SINT	
weqube:C.Data[12]	16#00		Hex	SINT	
weqube:C.Data[13]	16#00		Hex	SINT	
weqube:C.Data[14]	16#00		Hex	SINT	
weqube:C.Data[15]	16#00		Hex	SINT	
weqube:C.Data[16]	16#00		Hex	SINT	
weqube:C.Data[17]	16#00		Hex	SINT	

Properties — Extended Properties...

General	
Name	weqube:C.Data[10]
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	SINT
Scope	PLC
External Access	Read/Write
Style	Hex
Constant	No
Required	
Visible	
Data	
Value	16#01
Force Mask	
Produced Connection	
Consumed Connection	

```
Complete - 0 error(s), 0 warning(s)
```

Download the configuration to the PLC and go online.



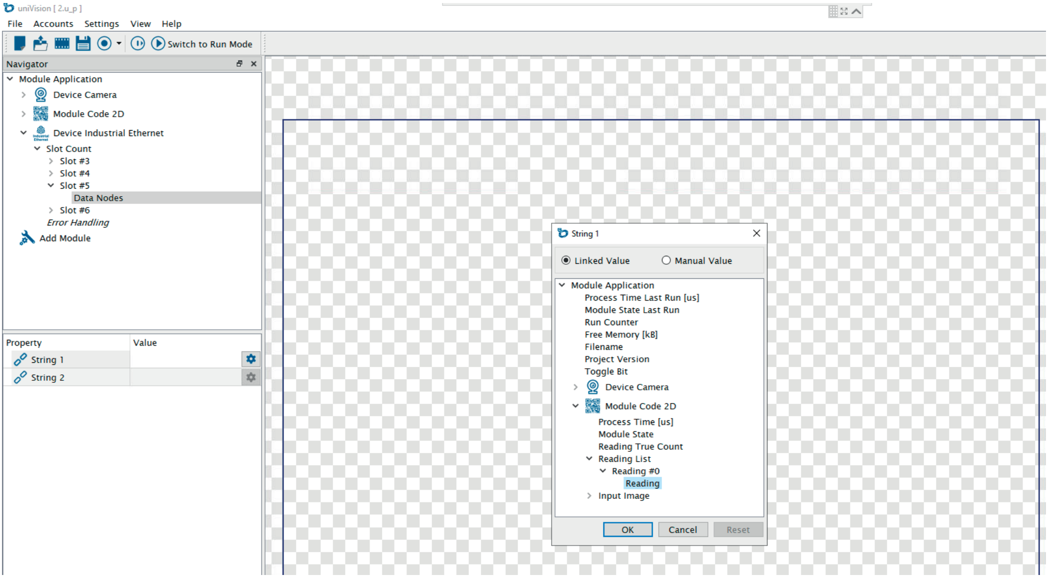
Description	
Status	Running
Module Fault	

```
Downloading all 'MainProgram' Routines...
Downloading routine 'MainRoutine'
Downloading all Modules...
Downloading all Tasks...
Downloading task 'MainTask'
Downloading Security Configuration...
Downloading all Data Logs...
Downloading Controller attributes...
Downloading Controller Extended Properties...
Linking all Routines...
Linking routine 'MainRoutine' of program 'MainProgram'
Finalizing download...
Download elapsed time 3.819 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

7.6 PLC Parameters and Local Tags

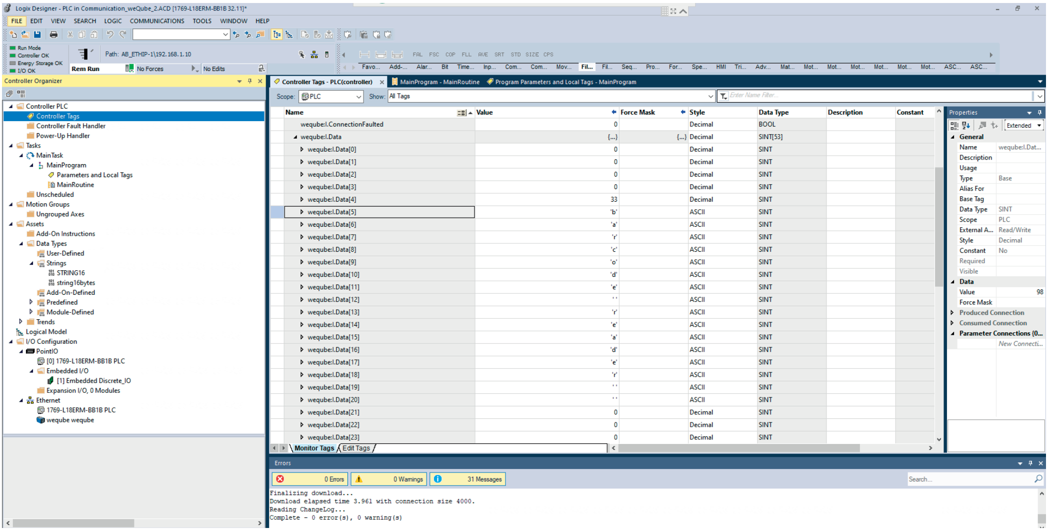
7.6.1 String data

The example shows the sending process of the code result as string 1 in slot 5 (Size: 16 bytes).



Property	Value
String 1	
String 2	

In the software Studio 5000 Logix Designer, the code result is available starting with byte 5 and ending with byte 20.



Scope: PLC Show: All Tags

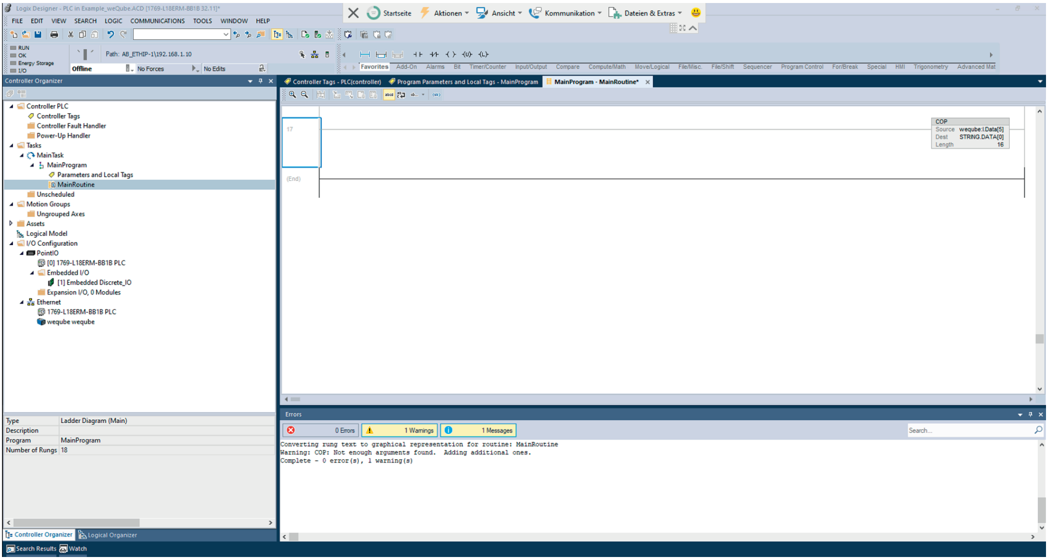
Name	Value	Force Mask	Style	Data Type	Description	Constant
weqube:I.ConnectionFaulted	0		Decimal	BOOL		
weqube:I.Data	{...}	{...}	Decimal	SINT[53]		
weqube:I.Data[0]	0		Decimal	SINT		
weqube:I.Data[1]	0		Decimal	SINT		
weqube:I.Data[2]	0		Decimal	SINT		
weqube:I.Data[3]	0		Decimal	SINT		
weqube:I.Data[4]	33		Decimal	SINT		
weqube:I.Data[5]	'b'		ASCII	SINT		
weqube:I.Data[6]	'a'		ASCII	SINT		
weqube:I.Data[7]	'r'		ASCII	SINT		
weqube:I.Data[8]	'c'		ASCII	SINT		
weqube:I.Data[9]	'o'		ASCII	SINT		
weqube:I.Data[10]	'd'		ASCII	SINT		
weqube:I.Data[11]	'e'		ASCII	SINT		
weqube:I.Data[12]	' '		ASCII	SINT		
weqube:I.Data[13]	'r'		ASCII	SINT		
weqube:I.Data[14]	'e'		ASCII	SINT		
weqube:I.Data[15]	'a'		ASCII	SINT		
weqube:I.Data[16]	'd'		ASCII	SINT		
weqube:I.Data[17]	'e'		ASCII	SINT		
weqube:I.Data[18]	'r'		ASCII	SINT		
weqube:I.Data[19]	' '		ASCII	SINT		
weqube:I.Data[20]	' '		ASCII	SINT		
weqube:I.Data[21]	0		Decimal	SINT		
weqube:I.Data[22]	0		Decimal	SINT		
weqube:I.Data[23]	0		Decimal	SINT		

Properties

General	
Name	weqube:I.Dat...
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	SINT
Scope	PLC
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	98
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections (0...	
	New Connecti...

```
Finalizing download...
Download elapsed time 3.961 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

To copy the bytes in a string, use the FILE COPY (COP) function in the Main Routine. In the example the source is linked to weqube:I.Data[5] and a new string tag with 16 bytes length is created. The destination is set to STRING.Data[0] and the size of the string is set fix to 16 bytes.



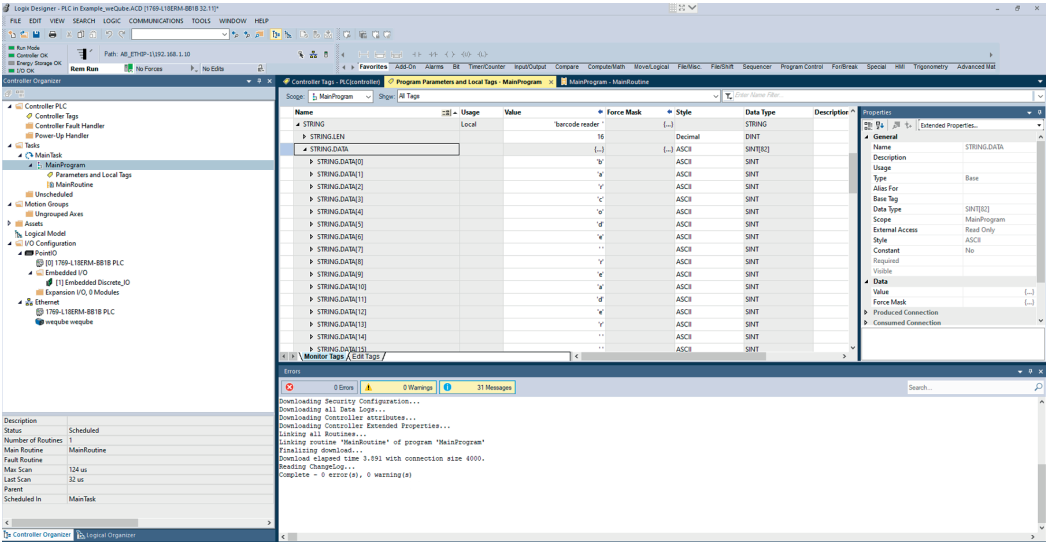
COP	
Source	weqube:I.Data[5]
Dest	STRING.DATA[0]
Length	16

Type	Ladder Diagram (Main)
Description	
Program	MainProgram
Number of Rungs	18

Errors: 0 Errors, 1 Warnings, 1 Messages

```
Converting rung text to graphical representation for routine: MainRoutine
Warning: COP: Not enough arguments found.  Adding additional ones.
Complete - 0 error(s), 1 warning(s)
```

The STRING result is available in the Parameteres and Local Tags.



Program Parameters and Local Tags - MainProgram

Scope: MainProgram Show: All Tags

Name	Usage	Value	Force Mask	Style	Data Type	Description
STRING	Local	'barcode reader '	{...}		STRING	
STRING.LEN		16		Decimal	DINT	
STRING.DATA		{...}	{...}	ASCII	SINT[82]	
STRING.DATA[0]		'b'		ASCII	SINT	
STRING.DATA[1]		'a'		ASCII	SINT	
STRING.DATA[2]		'r'		ASCII	SINT	
STRING.DATA[3]		'c'		ASCII	SINT	
STRING.DATA[4]		'o'		ASCII	SINT	
STRING.DATA[5]		'd'		ASCII	SINT	
STRING.DATA[6]		'e'		ASCII	SINT	
STRING.DATA[7]		' '		ASCII	SINT	
STRING.DATA[8]		'r'		ASCII	SINT	
STRING.DATA[9]		'e'		ASCII	SINT	
STRING.DATA[10]		'a'		ASCII	SINT	
STRING.DATA[11]		'd'		ASCII	SINT	
STRING.DATA[12]		'e'		ASCII	SINT	
STRING.DATA[13]		'r'		ASCII	SINT	
STRING.DATA[14]		' '		ASCII	SINT	
STRING.DATA[15]		' '		ASCII	SINT	

Properties

General	
Name	STRING.DATA
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	SINT[82]
Scope	MainProgram
External Access	Read Only
Style	ASCII
Constant	No
Required	
Visible	
Data	
Value	{...}
Force Mask	{...}

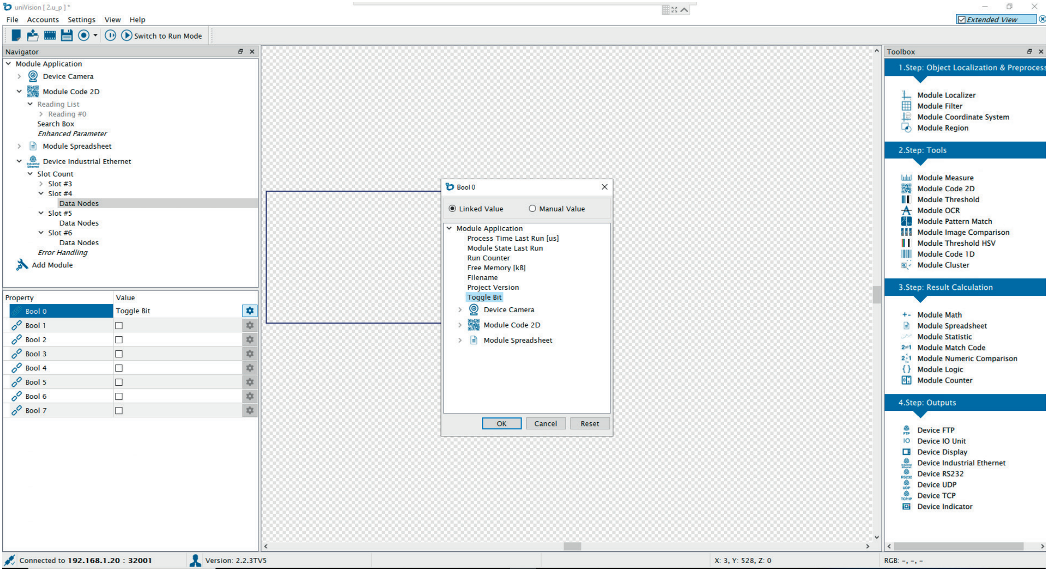
Description	
Status	Scheduled
Number of Routines	1
Main Routine	MainRoutine
Fault Routine	
Max Scan	124 us
Last Scan	32 us
Parent	
Scheduled In	MainTask

Errors: 0 Errors, 0 Warnings, 31 Messages

```
Downloading Security Configuration...
Downloading all Data Logs...
Downloading Controller attributes...
Downloading Controller Extended Properties...
Linking all Routines...
Linking routine 'MainRoutine' of program 'MainProgram'
Finalizing download...
Download elapsed time 3.891 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

7.6.2 BOOL data

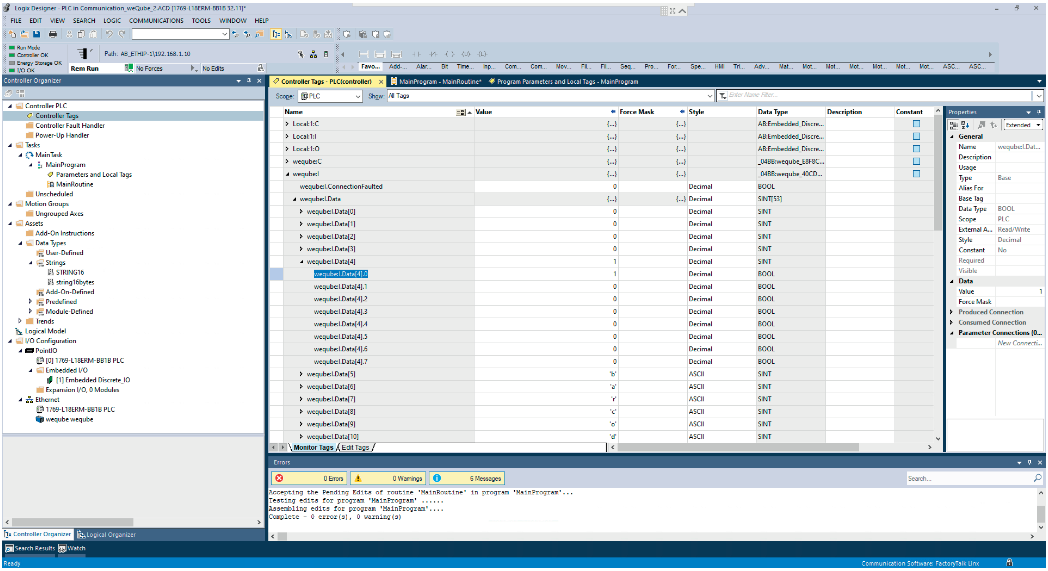
The following example shows linking a boolean result in slot 4 (Bool 0).



Property	Value
Bool 0	Toggle Bit
Bool 1	□
Bool 2	□
Bool 3	□
Bool 4	□
Bool 5	□
Bool 6	□
Bool 7	□

Connected to 192.168.1.20 : 32001 Version: 2.2.3TV5 X: 3, Y: 528, Z: 0 RGB: -, -, -

On the PLC, the boolean result is shown at weqube:I.Data[4].0.

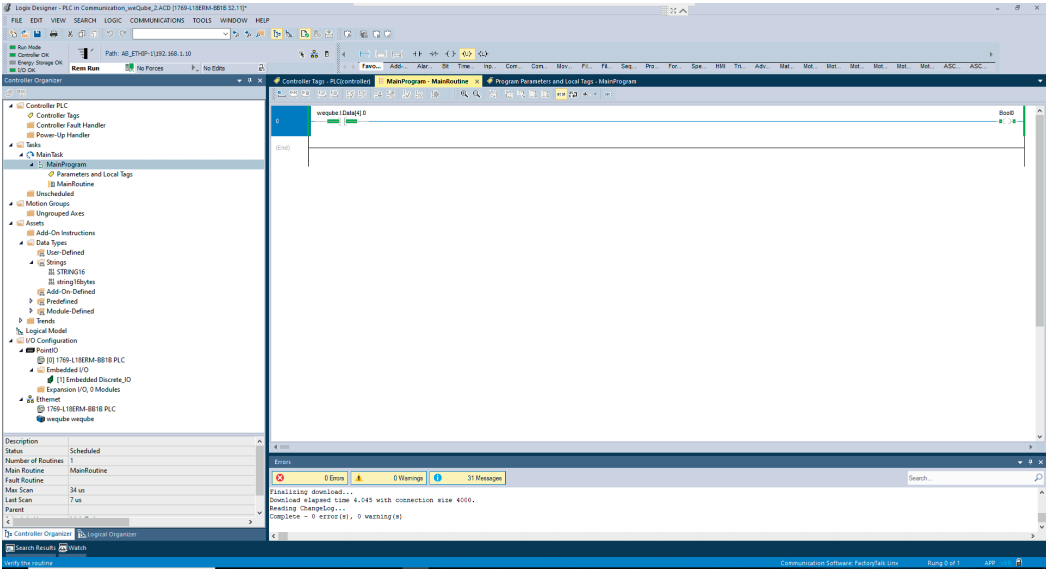


Name	Value	Force Mask	Style	Data Type	Description	Constant
Local:1:C	{...}	{...}		AB:Embedded_Discre...		□
Local:1:I	{...}	{...}		AB:Embedded_Discre...		□
Local:1:O	{...}	{...}		AB:Embedded_Discre...		□
weqube:C	{...}	{...}		_04BB:weqube_E8F8C...		□
weqube:I	{...}	{...}		_04BB:weqube_40CD...		□
weqube:I.ConnectionFaulted	0		Decimal	BOOL		
weqube:I.Data	{...}	{...}	Decimal	SINT[53]		
weqube:I.Data[0]	0		Decimal	SINT		
weqube:I.Data[1]	0		Decimal	SINT		
weqube:I.Data[2]	0		Decimal	SINT		
weqube:I.Data[3]	0		Decimal	SINT		
weqube:I.Data[4]	1		Decimal	SINT		
weqube:I.Data[4].0	1		Decimal	BOOL		
weqube:I.Data[4].1	0		Decimal	BOOL		
weqube:I.Data[4].2	0		Decimal	BOOL		
weqube:I.Data[4].3	0		Decimal	BOOL		
weqube:I.Data[4].4	0		Decimal	BOOL		
weqube:I.Data[4].5	0		Decimal	BOOL		
weqube:I.Data[4].6	0		Decimal	BOOL		
weqube:I.Data[4].7	0		Decimal	BOOL		
weqube:I.Data[5]	'b'		ASCII	SINT		
weqube:I.Data[6]	'a'		ASCII	SINT		
weqube:I.Data[7]	'r'		ASCII	SINT		
weqube:I.Data[8]	'c'		ASCII	SINT		
weqube:I.Data[9]	'o'		ASCII	SINT		
weqube:I.Data[10]	'd'		ASCII	SINT		

Properties	
Name	weqube:I.Dat...
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	BOOL
Scope	PLC
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Value	1
Force Mask	

```
Accepting the Pending Edits of routine 'MainRoutine' in program 'MainProgram'...
Testing edits for program 'MainProgram' ......
Assembling edits for program 'MainProgram'....
Complete - 0 error(s), 0 warning(s)
```

To copy the boolean result in a local tag, add the “Examine On” and the “Output Energize” in the Main Routine. Link the “Examine On” to byte 4 bool 0 and link the “Output Energize” to a new boolean tag.

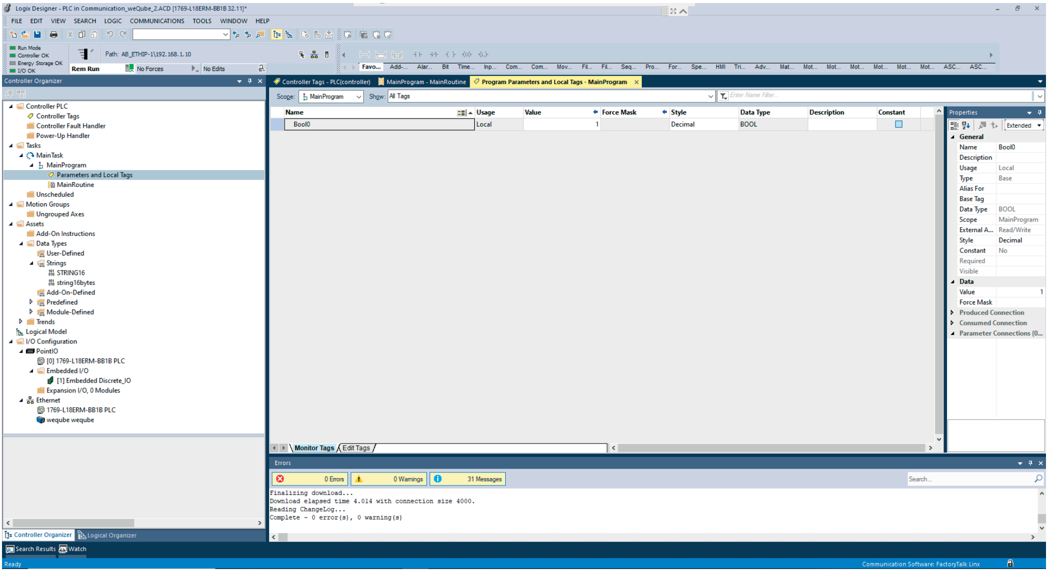

Logix Designer - PLC in Communication_weQube_2.ACD [1769-L18ERM-BB1B 32.11]*
Controller Tags - PLC(controller)
MainProgram - MainRoutine
Program Parameters and Local Tags - MainProgram
weqube:I.Data[4].0
Bool0
(End)

Description	
Status	Scheduled
Number of Routines	1
Main Routine	MainRoutine
Fault Routine	
Max Scan	34 us
Last Scan	7 us
Parent	

0 Errors 0 Warnings 31 Messages

```
Finalizing download...
Download elapsed time 4.045 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

The boolean result is available in the Parameters and Local Tags.


Logix Designer - PLC in Communication_weQube_2.ACD [1769-L18ERM-BB1B 32.11]*
Controller Tags - PLC(controller)
MainProgram - MainRoutine
Program Parameters and Local Tags - MainProgram
Monitor Tags
Edit Tags

Scope: MainProgram Show: All Tags

Name	Usage	Value	Force Mask	Style	Data Type	Description	Constant
Bool0	Local	1		Decimal	BOOL		□

Properties	
General	
Name	Bool0
Description	
Usage	Local
Type	Base
Alias For	
Base Tag	
Data Type	BOOL
Scope	MainProgram
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	1
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections {0...	

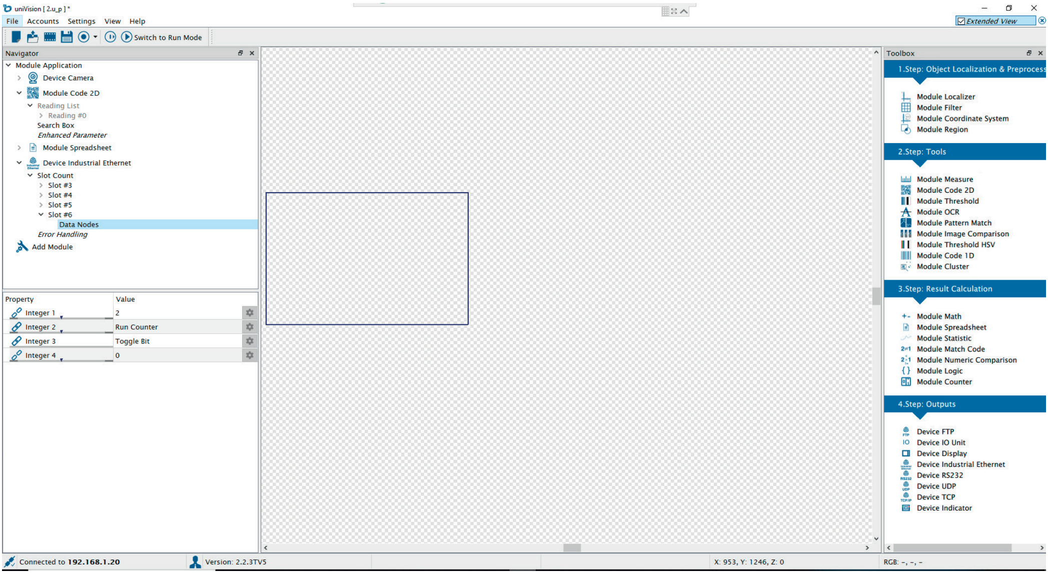
0 Errors 0 Warnings 31 Messages

```
Finalizing download...
Download elapsed time 4.014 with connection size 4000.
Reading ChangeLog...
Complete - 0 error(s), 0 warning(s)
```

7.6.3 DINT and REAL data

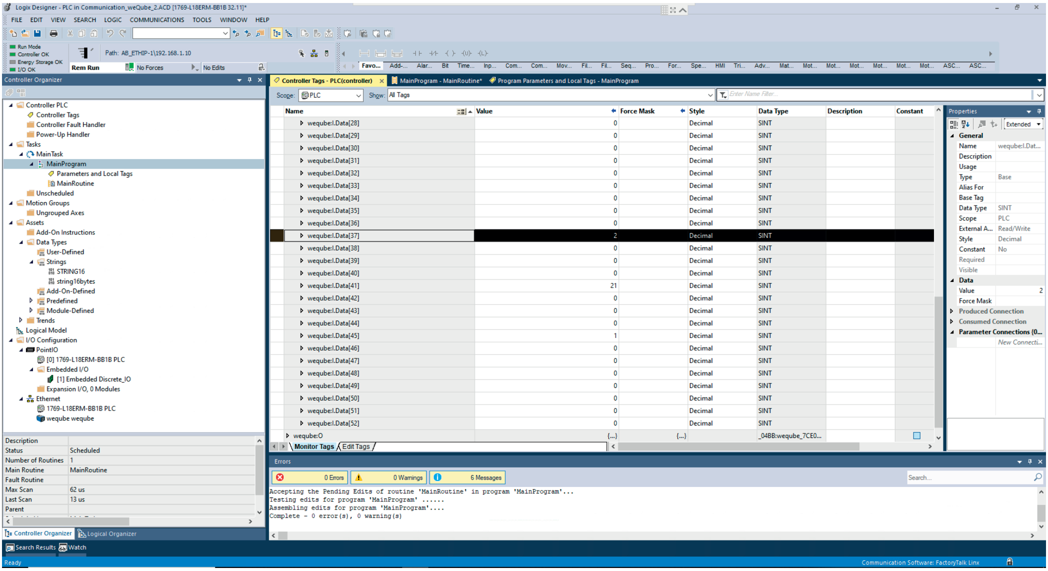
The following example shows linking DINT results in slot 6.

- DINT 1: Project number
- DINT 2: Run Counter
- DINT 3: Toggle Bit



Property	Value
Integer 1	2
Integer 2	Run Counter
Integer 3	Toggle Bit
Integer 4	0

On the PLC, the DINT results are shown in single bytes.



Name	Value	Force Mask	Style	Data Type	Description	Constant
weqube:I.Data[28]	0		Decimal	SINT		
weqube:I.Data[29]	0		Decimal	SINT		
weqube:I.Data[30]	0		Decimal	SINT		
weqube:I.Data[31]	0		Decimal	SINT		
weqube:I.Data[32]	0		Decimal	SINT		
weqube:I.Data[33]	0		Decimal	SINT		
weqube:I.Data[34]	0		Decimal	SINT		
weqube:I.Data[35]	0		Decimal	SINT		
weqube:I.Data[36]	0		Decimal	SINT		
weqube:I.Data[37]	2		Decimal	SINT		
weqube:I.Data[38]	0		Decimal	SINT		
weqube:I.Data[39]	0		Decimal	SINT		
weqube:I.Data[40]	0		Decimal	SINT		
weqube:I.Data[41]	21		Decimal	SINT		
weqube:I.Data[42]	0		Decimal	SINT		
weqube:I.Data[43]	0		Decimal	SINT		
weqube:I.Data[44]	0		Decimal	SINT		
weqube:I.Data[45]	1		Decimal	SINT		
weqube:I.Data[46]	0		Decimal	SINT		
weqube:I.Data[47]	0		Decimal	SINT		
weqube:I.Data[48]	0		Decimal	SINT		
weqube:I.Data[49]	0		Decimal	SINT		
weqube:I.Data[50]	0		Decimal	SINT		
weqube:I.Data[51]	0		Decimal	SINT		
weqube:I.Data[52]	0		Decimal	SINT		
weqube:O	{...}	{...}		_04BB:weqube_7CE0...		□

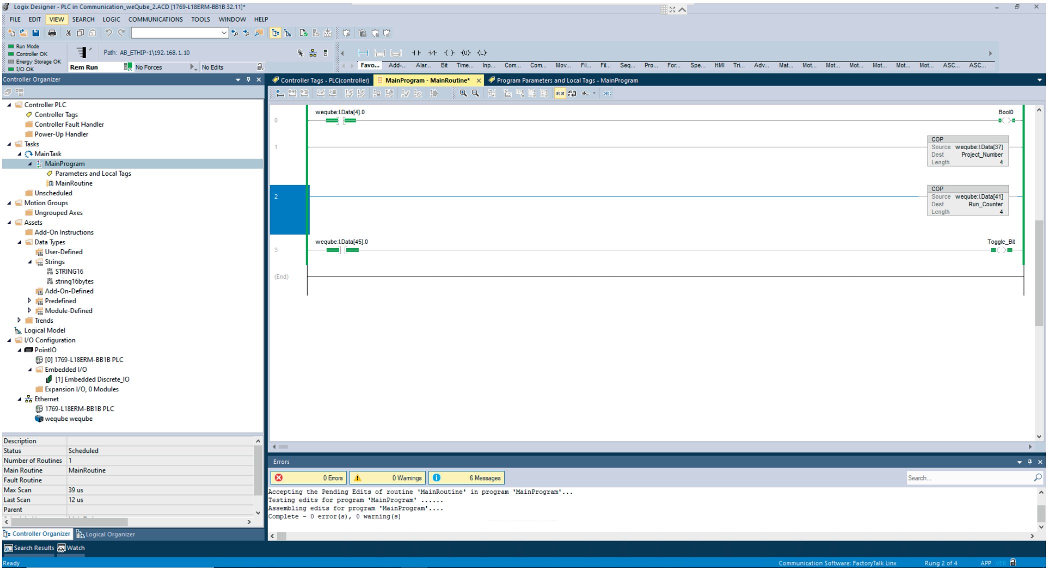
General	
Name	weqube:I.Dat...
Description	
Usage	
Type	Base
Alias For	
Base Tag	
Data Type	SINT
Scope	PLC
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	2
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections (0...	
	New Connecti...

Description	
Status	Scheduled
Number of Routines	1
Main Routine	MainRoutine
Fault Routine	
Max Scan	62 us
Last Scan	13 us
Parent	

0 Errors | 0 Warnings | 6 Messages

```
Accepting the Pending Edits of routine 'MainRoutine' in program 'MainProgram'...
Testing edits for program 'MainProgram' ......
Assembling edits for program 'MainProgram'....
Complete - 0 error(s), 0 warning(s)
```

To create DINT results out of the single bytes, use the FILE COPY (COP) function in the Main Routine. In the example, the source is linked to byte 37 for the project number and to byte 41 for the run counter. Create new tags for the destination (with data type DINT and a length of 4 bytes).



COP	
Source	weqube:I.Data[37]
Dest	Project_Number
Length	4

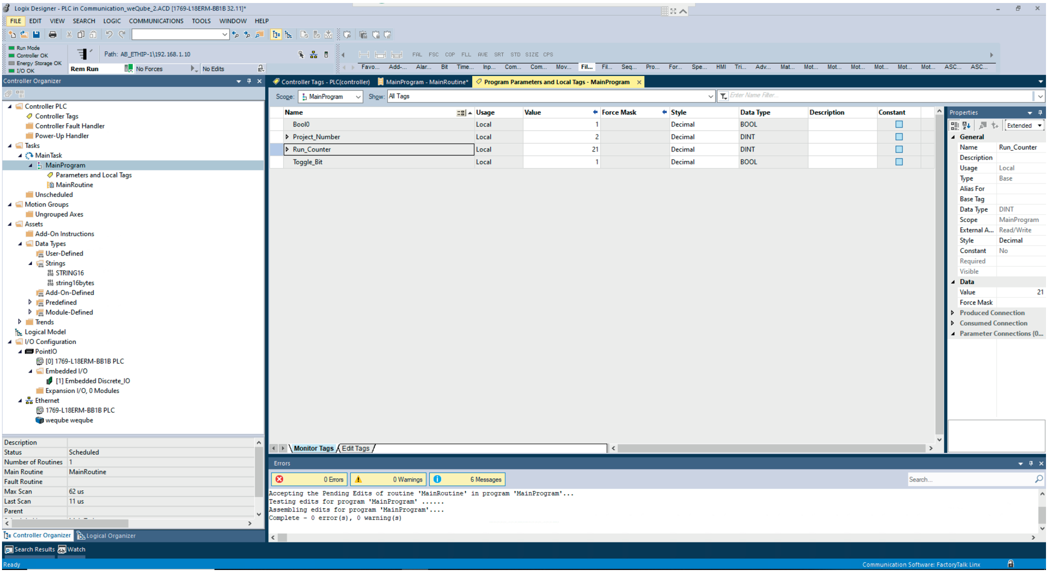
COP	
Source	weqube:I.Data[41]
Dest	Run_Counter
Length	4

Description	
Status	Scheduled
Number of Routines	1
Main Routine	MainRoutine
Fault Routine	
Max Scan	39 us
Last Scan	12 us
Parent	

0 Errors | 0 Warnings | 6 Messages

```
Accepting the Pending Edits of routine 'MainRoutine' in program 'MainProgram'...
Testing edits for program 'MainProgram' ......
Assembling edits for program 'MainProgram'....
Complete - 0 error(s), 0 warning(s)
```

The DINT results are available in the Parameters and Local Tags.



Scope: MainProgram Show: All Tags

Name	Usage	Value	Force Mask	Style	Data Type	Description	Constant
Bool0	Local	1		Decimal	BOOL		□
Project_Number	Local	2		Decimal	DINT		□
Run_Counter	Local	21		Decimal	DINT		□
Toggle_Bit	Local	1		Decimal	BOOL		□

Properties (Extended)

General	
Name	Run_Counter
Description	
Usage	Local
Type	Base
Alias For	
Base Tag	
Data Type	DINT
Scope	MainProgram
External A...	Read/Write
Style	Decimal
Constant	No
Required	
Visible	
Data	
Value	21
Force Mask	
Produced Connection	
Consumed Connection	
Parameter Connections (0...	

Description	
Status	Scheduled
Number of Routines	1
Main Routine	MainRoutine
Fault Routine	
Max Scan	62 us
Last Scan	11 us
Parent	

0 Errors 0 Warnings 6 Messages

```
Accepting the Pending Edits of routine 'MainRoutine' in program 'MainProgram'...
Testing edits for program 'MainProgram' ......
Assembling edits for program 'MainProgram'....
Complete - 0 error(s), 0 warning(s)
```

NOTE!
REAL data can be copied the same way to Parameters and Local Tags. The only difference is that the data type of the destination must be set to REAL.

NOTE!
Make sure to check the change of the toggle bit and/or the run counter in the last slot in order to check if all measurement results are already updated.

8. PLC Settings at Omron PLCs

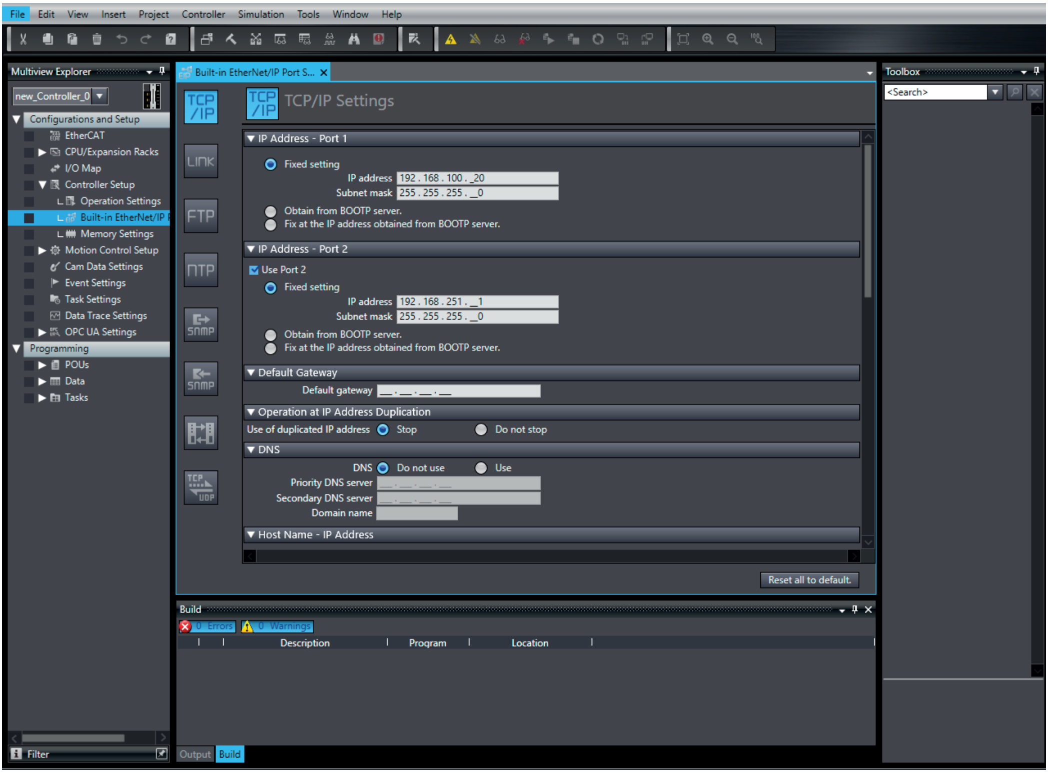
The Smart Camera can also be integrated via EtherNet/IP at Omron PLCs. The following description shows the relevant steps for a NX102-1200 PLC from Omron with Sysmac Studio Version 1.41.0.10.

8.1 Network Settings

In the example, the following network settings are used:

- PC with Sysmac Studio and uniVision software: IP address 192.168.100.1
- Smart Camera weQube: IP address 192.168.100.15
- PLC: IP address 192.168.100.20

Open Sysmac Studio and define the network settings of the PLC.



TCP/IP Settings

▼ IP Address - Port 1

● Fixed setting

IP address 192 . 168 . 100 . _20

Subnet mask 255 . 255 . 255 . _0

○ Obtain from BOOTP server.

○ Fix at the IP address obtained from BOOTP server.

▼ IP Address - Port 2

☑ Use Port 2

● Fixed setting

IP address 192 . 168 . 251 . _1

Subnet mask 255 . 255 . 255 . _0

○ Obtain from BOOTP server.

○ Fix at the IP address obtained from BOOTP server.

▼ Default Gateway

Default gateway ___ . ___ . ___ . ___

▼ Operation at IP Address Duplication

Use of duplicated IP address ● Stop ○ Do not stop

▼ DNS

DNS ● Do not use ○ Use

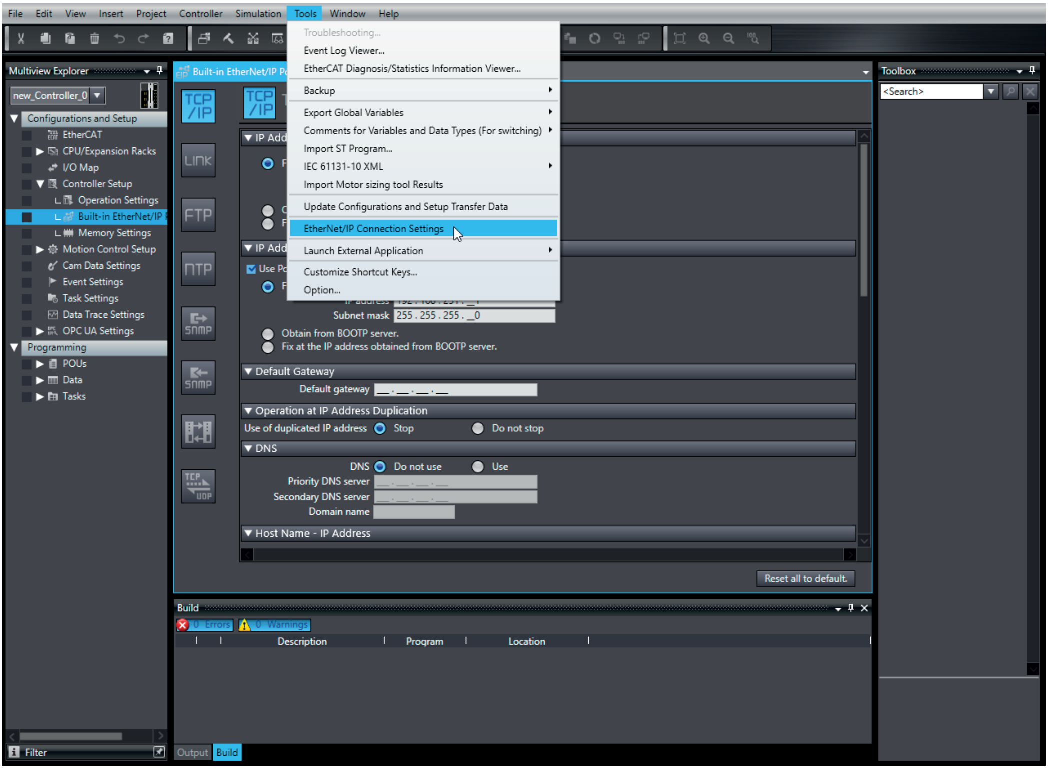
Priority DNS server ___ . ___ . ___ . ___

Secondary DNS server ___ . ___ . ___ . ___

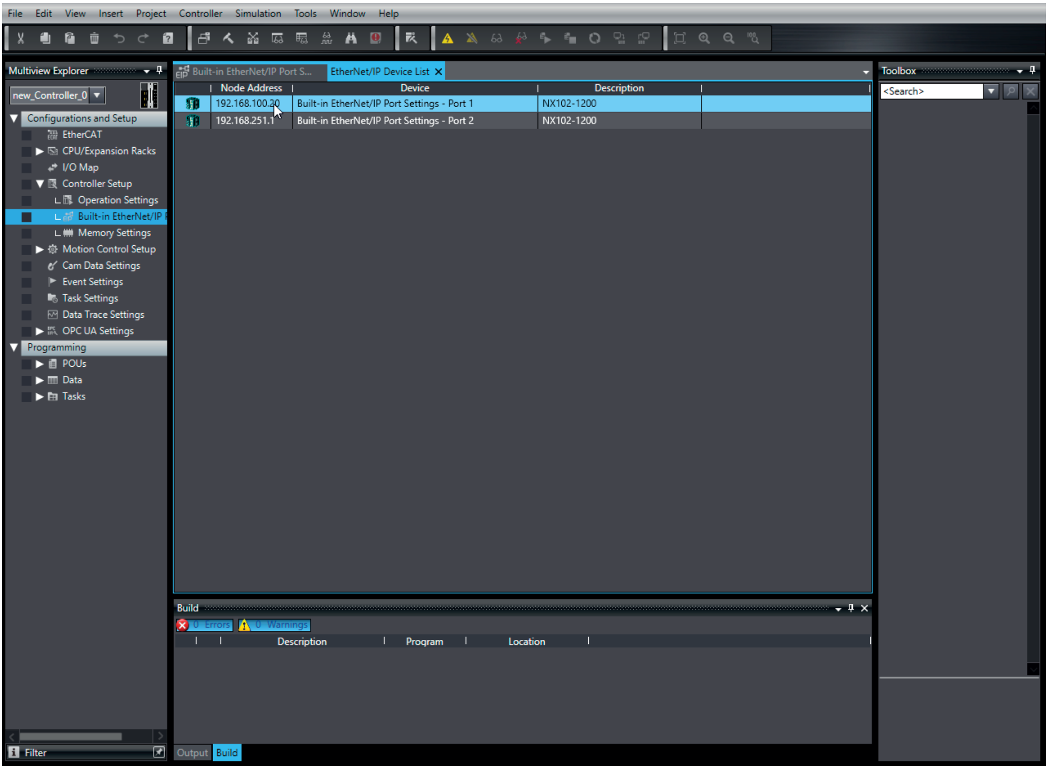
Domain name

▼ Host Name - IP Address

Click on "Tools" → "EtherNet/IP Connection Settings".

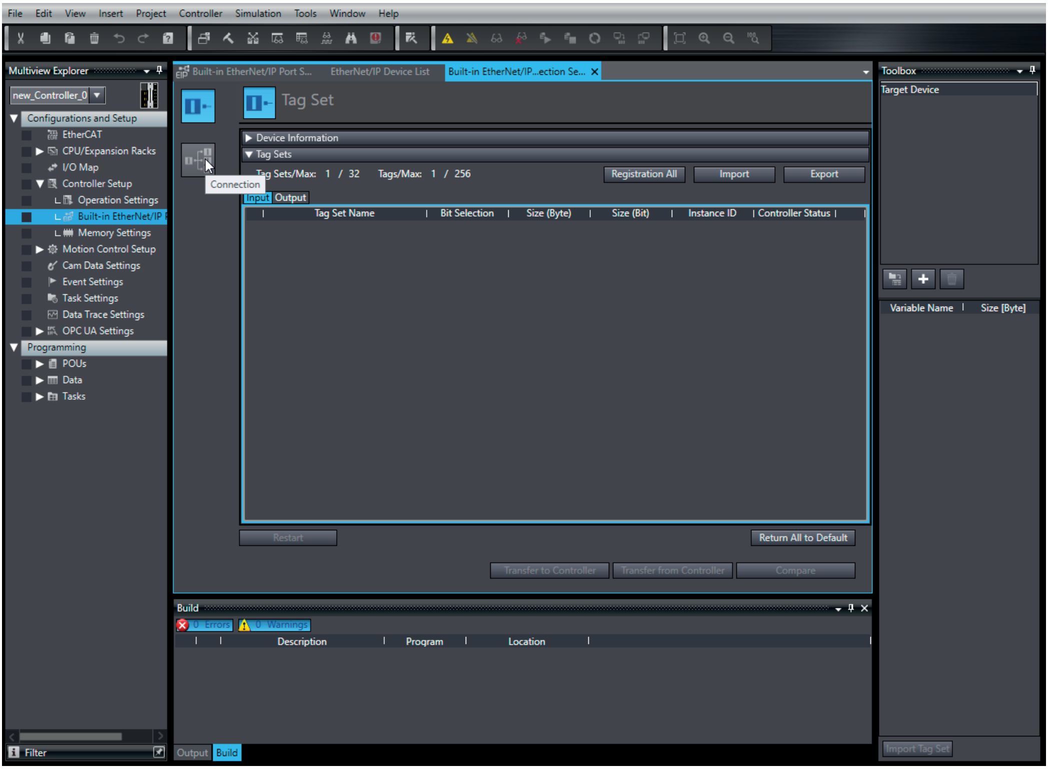


Double click on the relevant IP address.



Node Address	Device	Description
192.168.100.20	Built-in EtherNet/IP Port Settings - Port 1	NX102-1200
192.168.251.1	Built-in EtherNet/IP Port Settings - Port 2	NX102-1200

Select the connection button.



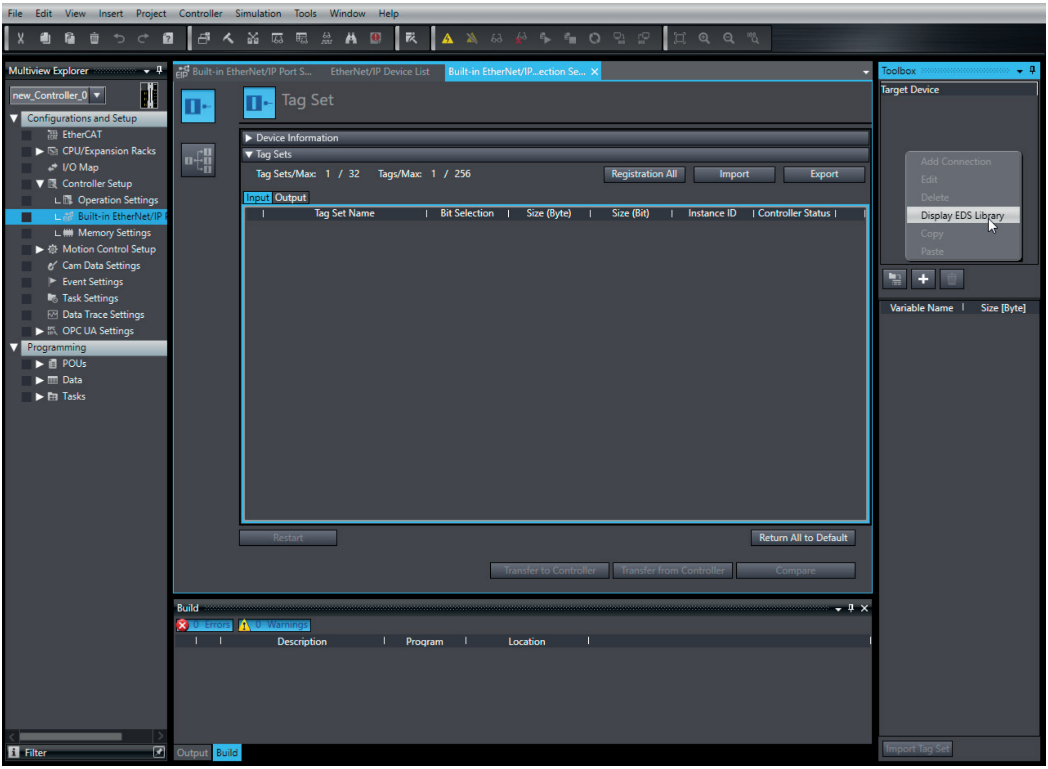
Tag Sets/Max: 1 / 32 Tags/Max: 1 / 256

Tag Set Name	Bit Selection	Size (Byte)	Size (Bit)	Instance ID	Controller Status

Description	Program	Location

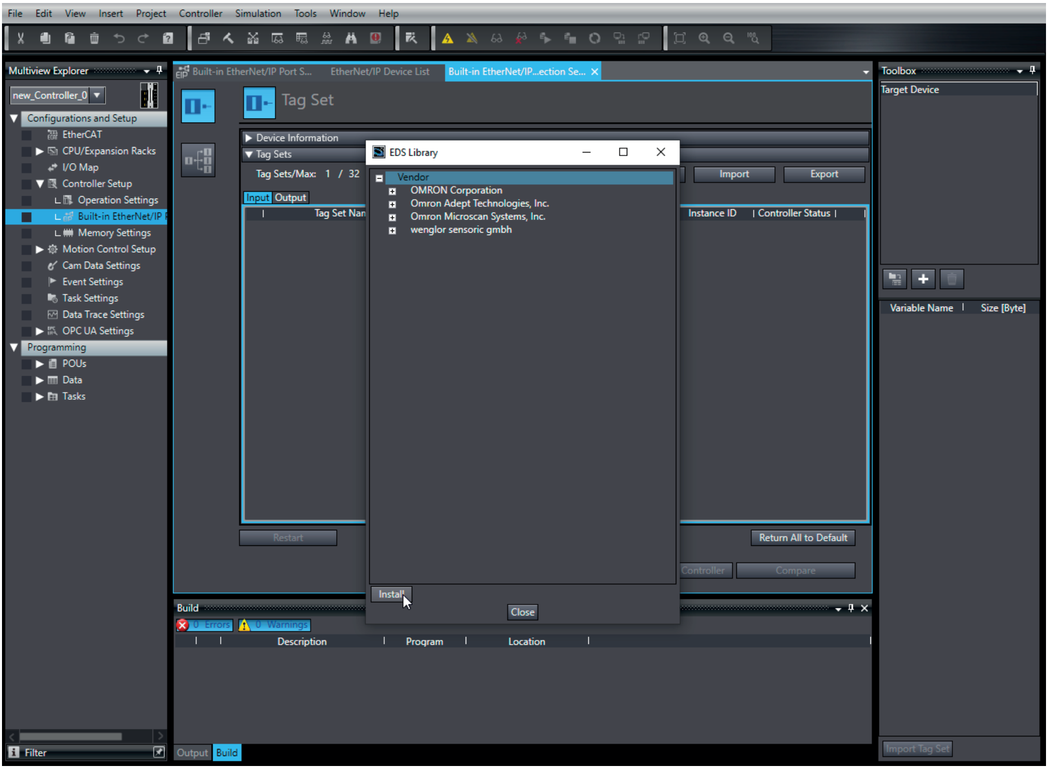
8.2 EDS file

Open the context menu at the toolbox with a right click and select "Display EDS Library".

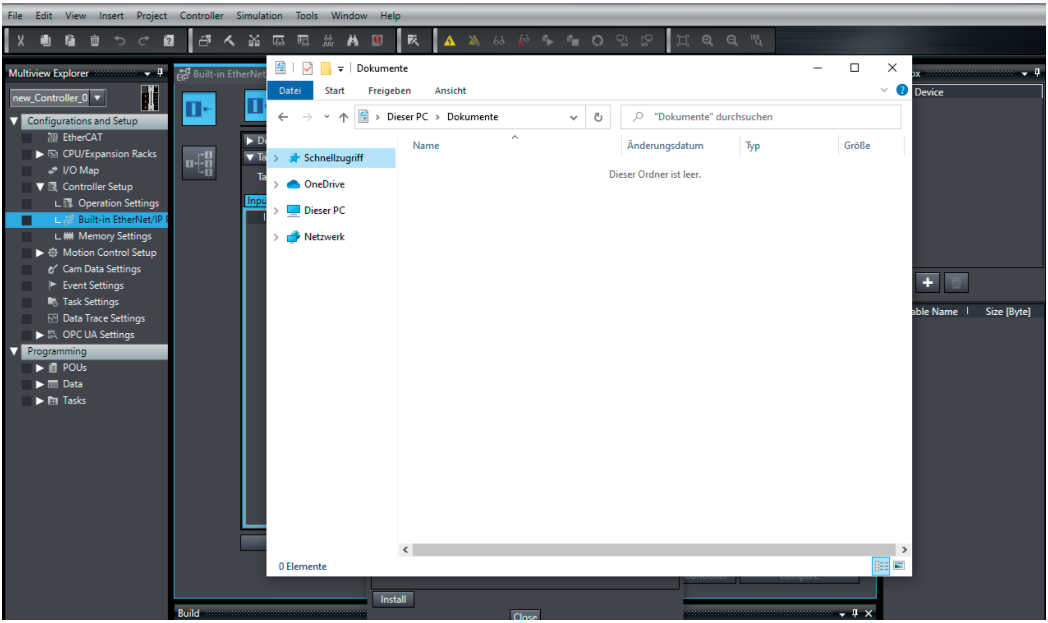


Tag Sets/Max: 1 / 32 Tags/Max: 1 / 256

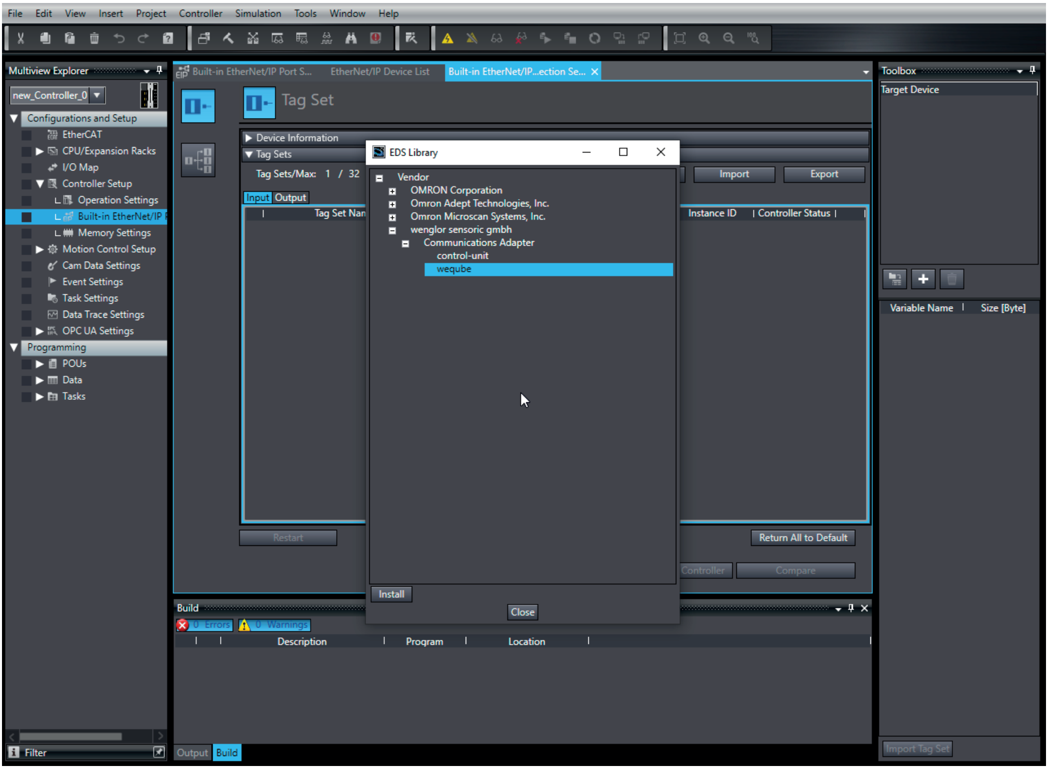
Click on "Install" to install an EDS file.



Select the EDS file of the Smart Camera weQube. Visit www.wenglor.com and search for the article number of the Smart Camera in order to download the EDS file.

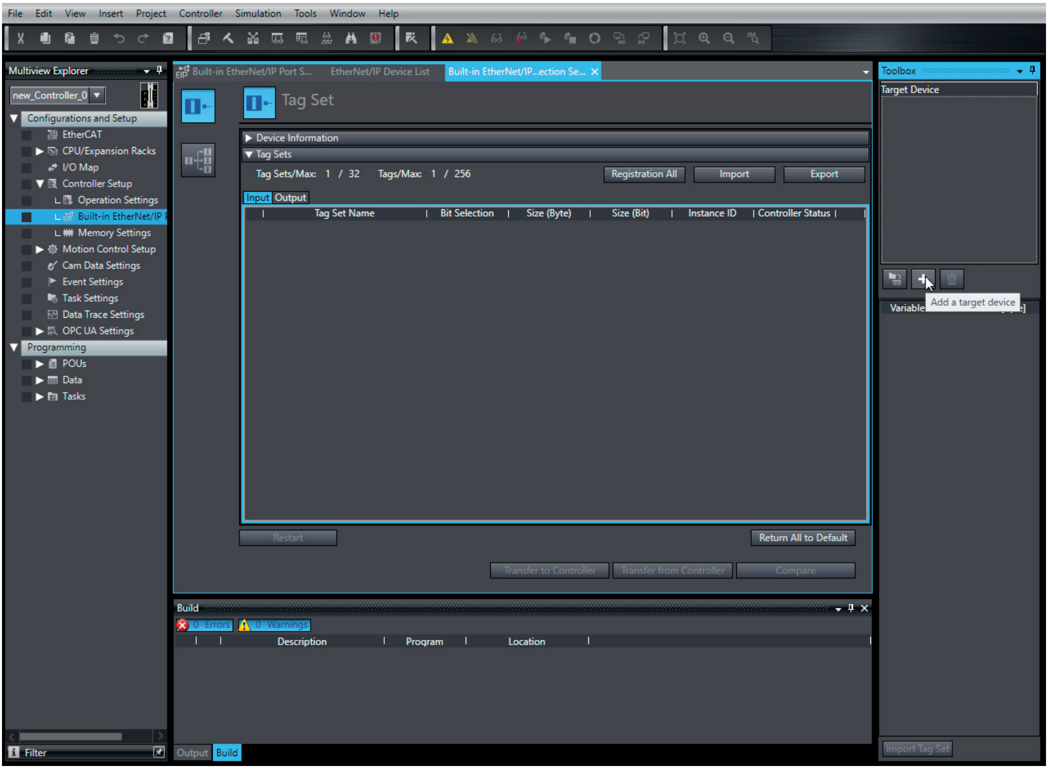


After the installation of the EDS file, it is shown at "wenglor sensoric gmbh" -> "Communications Adapters".

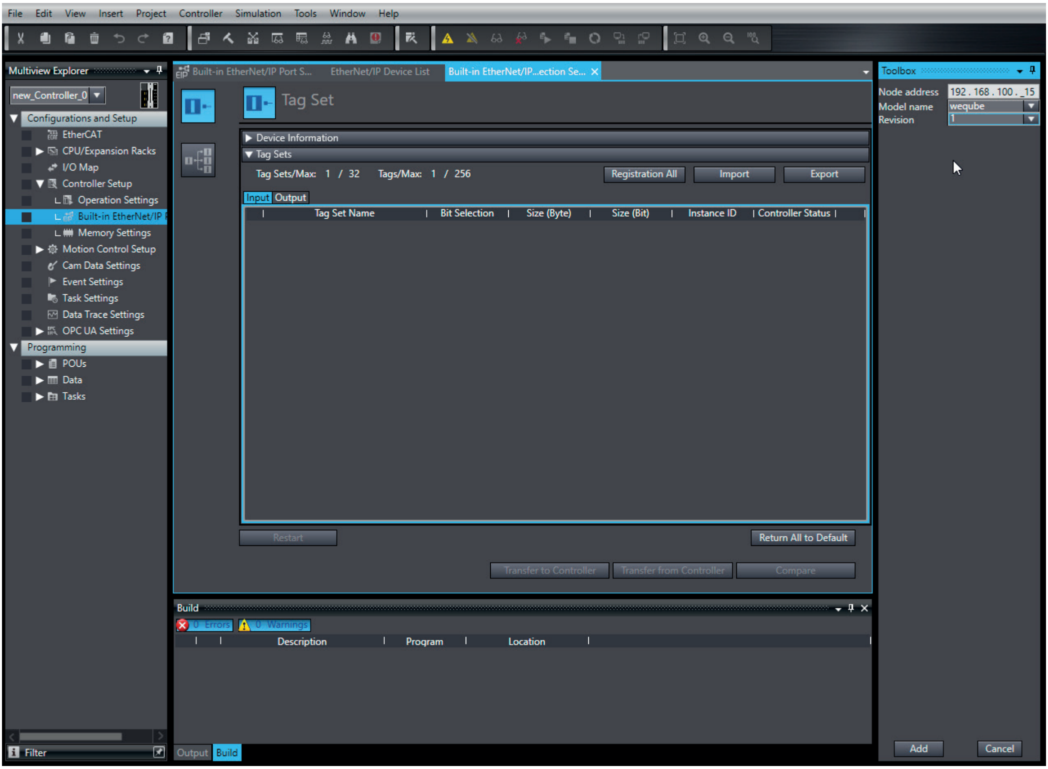


8.3 Add Smart Camera to PLC network

Close the window and click on the add symbol in the "Target Device" window.

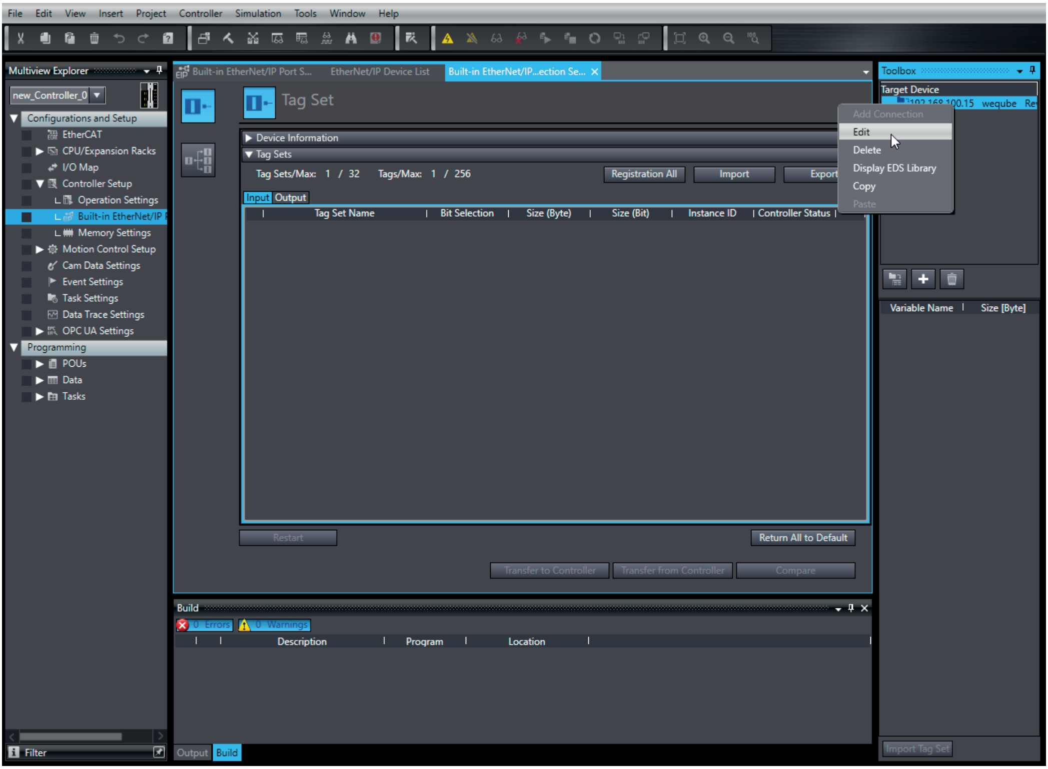


Enter the IP address, the name "weqube" and the revision of the Smart Camera and click on "Add".



Node address 192 . 168 . 100 . _15
Model name weqube
Revision 1

Open the context menu of the device in order to edit the settings.



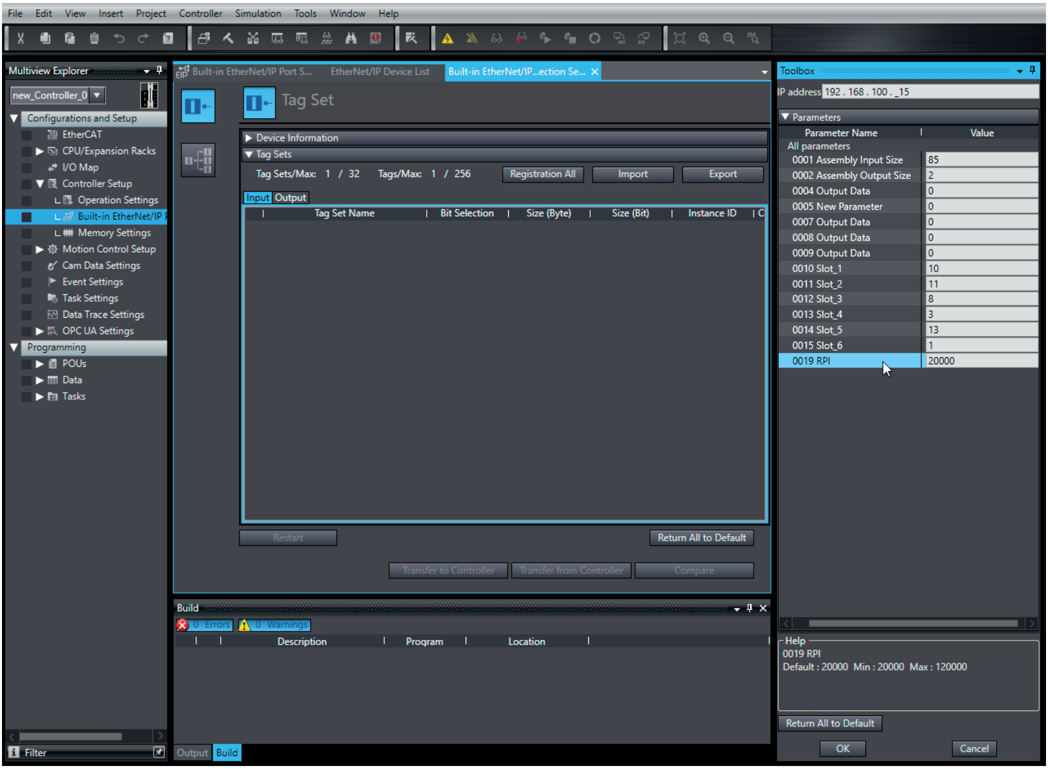
Tag Sets/Max: 1 / 32 Tags/Max: 1 / 256

Depending on the settings of the uniVision project, the size and the Module IDs for all slots must be set. In the example, the following slots are used:

- Slot 1: Project number (fix)
- Slot 2: Module Status (fix)
- Slot 3: 1 Byte Output (8 BOOL)
- Slot 4: 1 Byte Input (8 BOOL)
- Slot 5: 64 Byte Input (2 CHAR)
- Slot 6: 16 Byte Input (4 DINT)

Consequently, the following settings must be done:

- Assembly Input Size: 85 bytes
- Assembly Output Size: 2 bytes
- Slot_1 (Module ID): 10
- Slot_2 (Module ID): 11
- Slot_3 (Module ID): 8
- Slot_4 (Module ID): 3
- Slot_5 (Module ID): 13
- Slot 6 (Module ID): 1
- RPI: 20000 µs



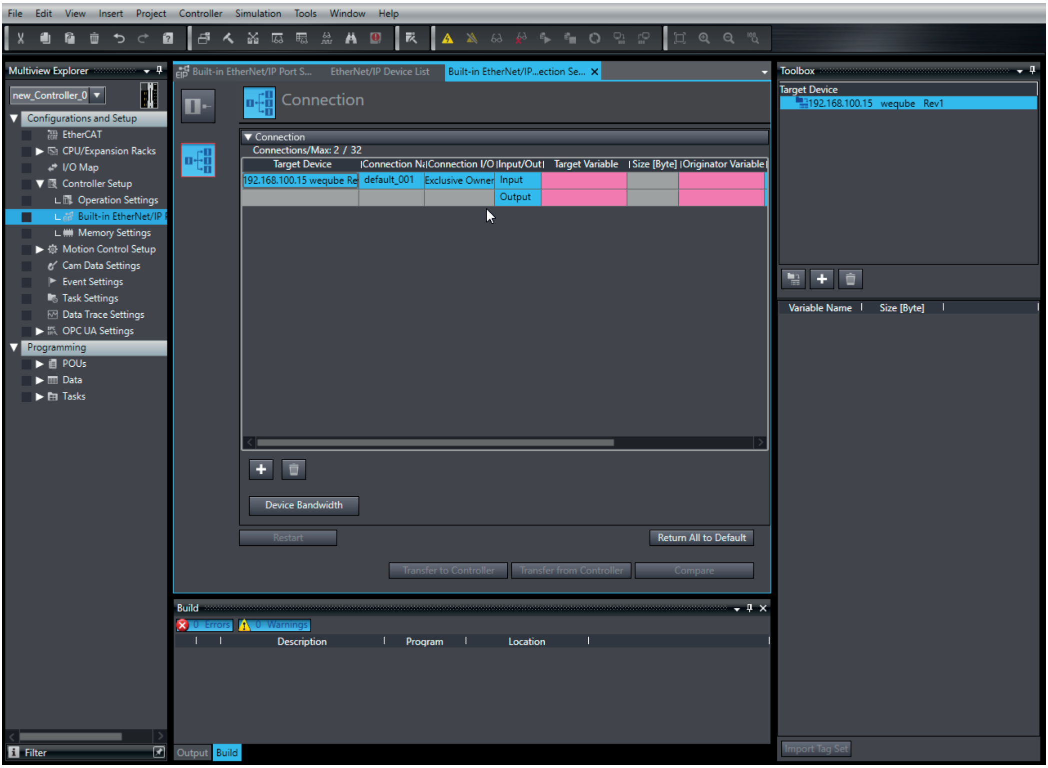
IP address 192 . 168 . 100 . _15

Parameters

Parameter Name	Value
All parameters	
0001 Assembly Input Size	85
0002 Assembly Output Size	2
0004 Output Data	0
0005 New Parameter	0
0007 Output Data	0
0008 Output Data	0
0009 Output Data	0
0010 Slot_1	10
0011 Slot_2	11
0012 Slot_3	8
0013 Slot_4	3
0014 Slot_5	13
0015 Slot_6	1
0019 RPI	20000

Help
0019 RPI
Default : 20000 Min : 20000 Max : 120000

Add the weQube to the connections.



Connection

Connections/Max: 2 / 32

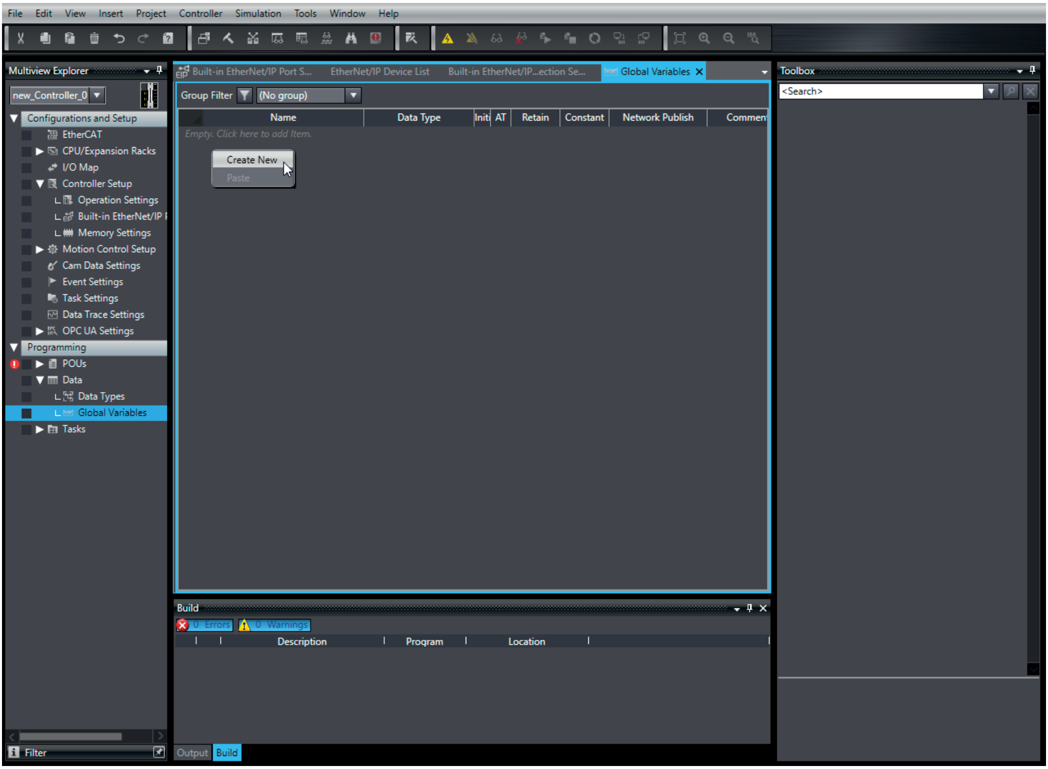
Target Device	Connection N	Connection I/O	Input/Out	Target Variable	Size [Byte]	Originator Variable
192.168.100.15 weqube Re	default_001	Exclusive Owner	Input			
			Output			

Target Device: 192.168.100.15 weqube Rev1

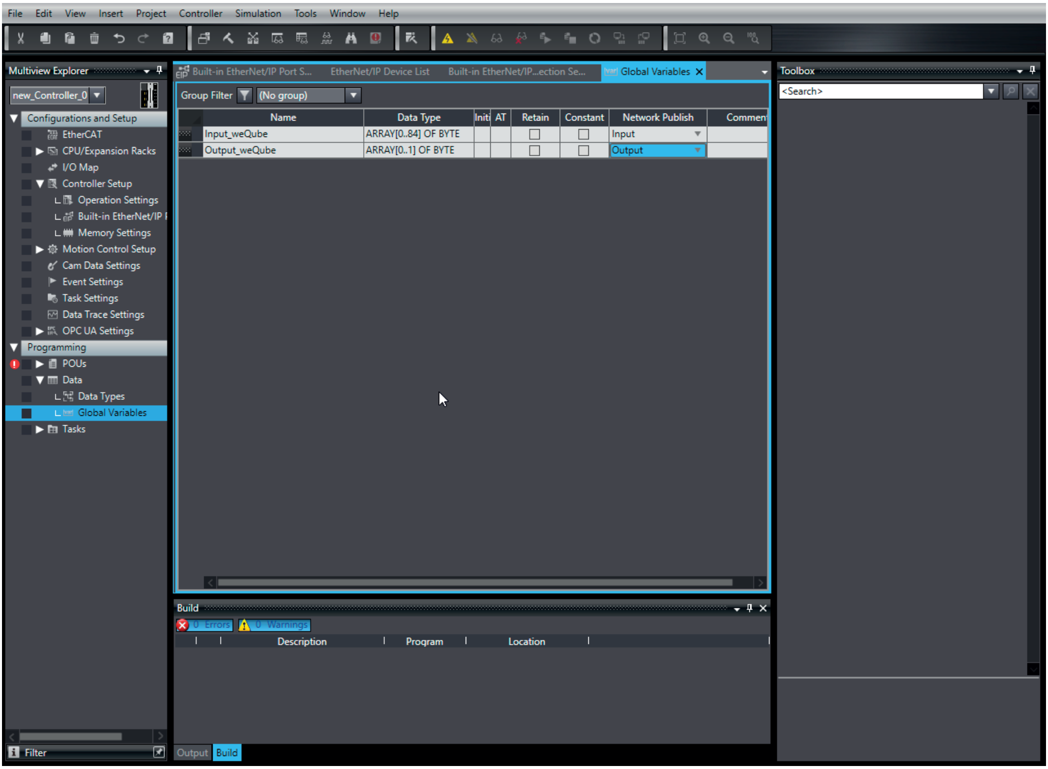
0 Errors 0 Warnings

8.4 Configure Input and Output Data

Open the global variables and create new variables.

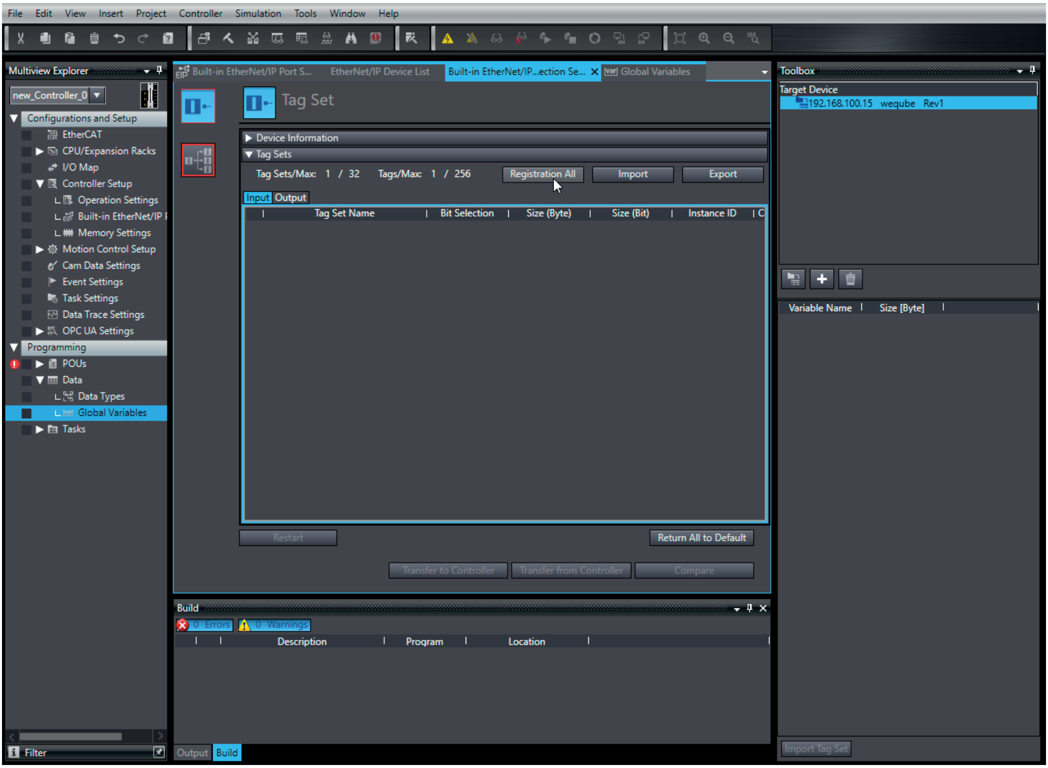


One array of bytes is necessary for the input data and one output array of bytes is needed for the output data.

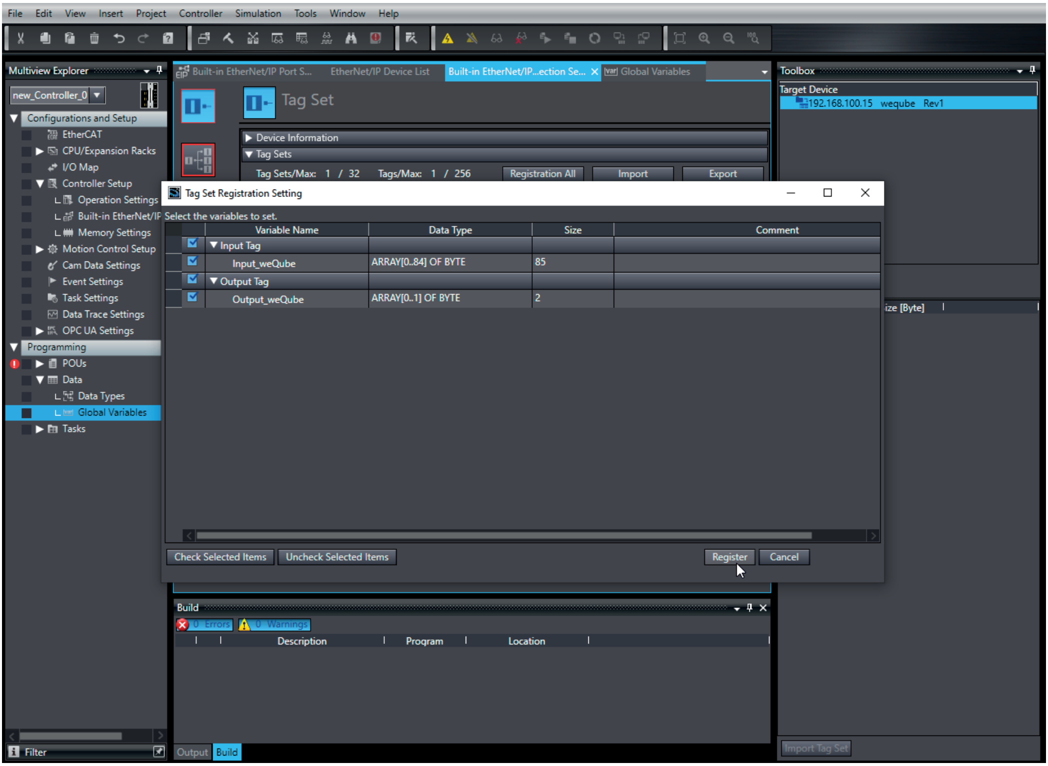


Name	Data Type	Initi	AT	Retain	Constant	Network Publish	Comment
Input_weQube	ARRAY[0..84] OF BYTE			□	□	Input	
Output_weQube	ARRAY[0..1] OF BYTE			□	□	Output	

Switch to the tag definition and click on "Registration All.



Select all and click on "Register".



Tag Set Registration Setting

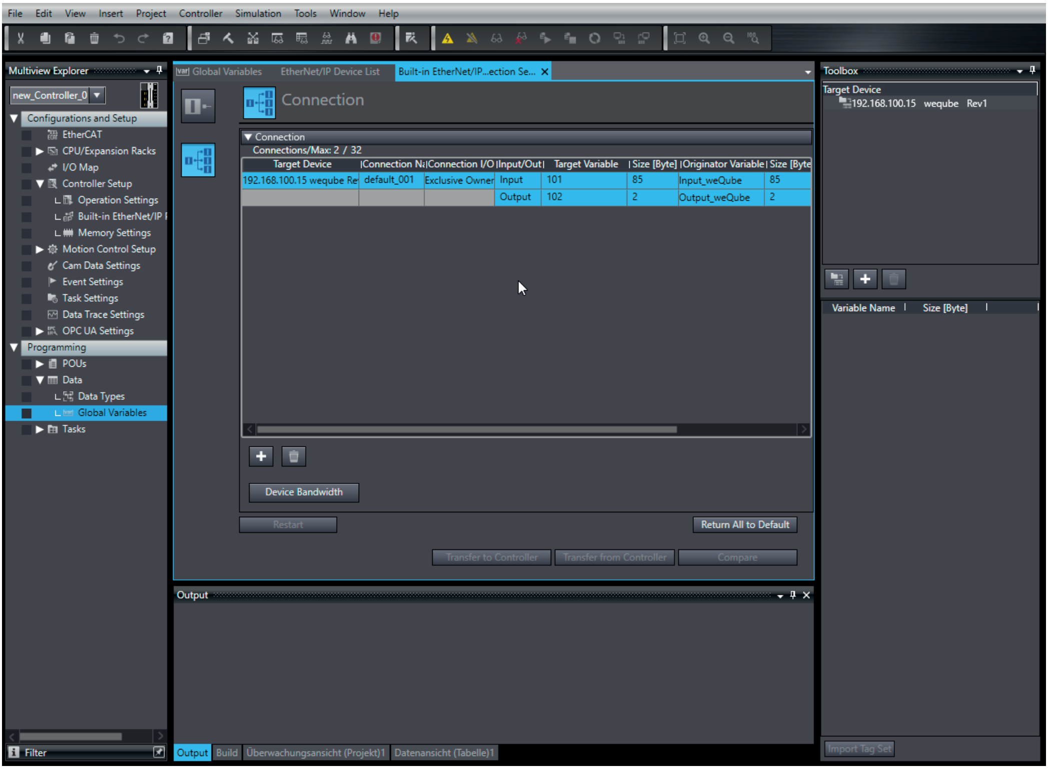
Select the variables to set.

	Variable Name	Data Type	Size	Comment
☑	▼ Input Tag			
☑	Input_weQube	ARRAY[0..84] OF BYTE	85	
☑	▼ Output Tag			
☑	Output_weQube	ARRAY[0..1] OF BYTE	2	

Tag Sets/Max: 1 / 32 Tags/Max: 1 / 256

Target Device: 192.168.100.15 weqube Rev1

Select in the connections the input and output variables.



Connection

Connections/Max: 2 / 32

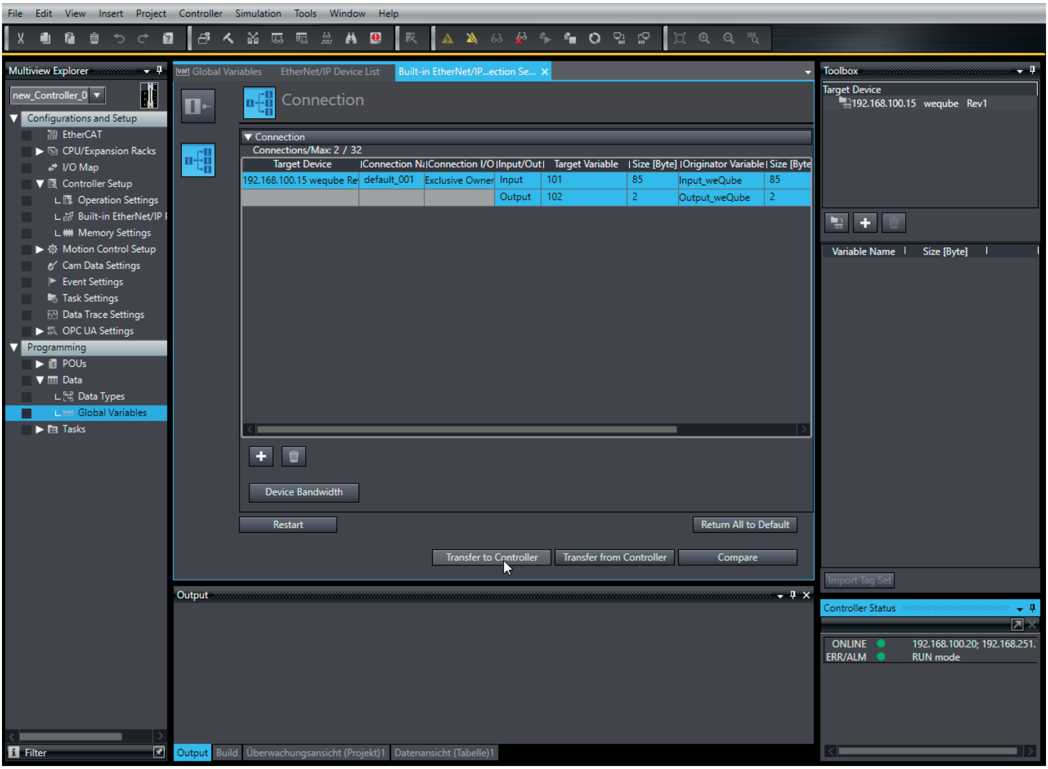
Target Device	Connection N	Connection I/O	Input/Out	Target Variable	Size [Byte]	Originator Variable	Size [Byte
192.168.100.15 weqube Re	default_001	Exclusive Owner	Input	101	85	Input_weQube	85
			Output	102	2	Output_weQube	2

Target Device
192.168.100.15 weqube Rev1

Variable Name	Size [Byte]

8.5 Download Configuration to PLC

Go online and select "Transfer to Controller" to send the configuration to the PLC.



Connection

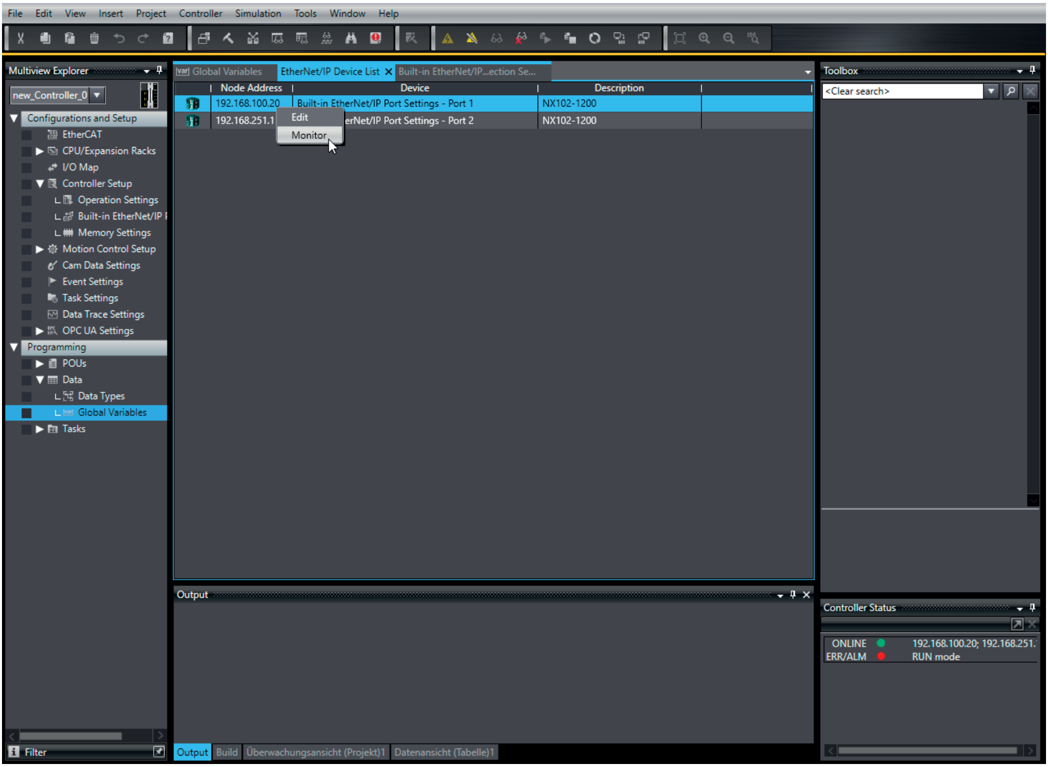
Connections/Max: 2 / 32

Target Device	Connection N	Connection I/O	Input/Out	Target Variable	Size [Byte]	Originator Variable	Size [Byte
192.168.100.15 weqube Re	default_001	Exclusive Owner	Input	101	85	Input_weQube	85
			Output	102	2	Output_weQube	2

Target Device
192.168.100.15 weqube Rev1

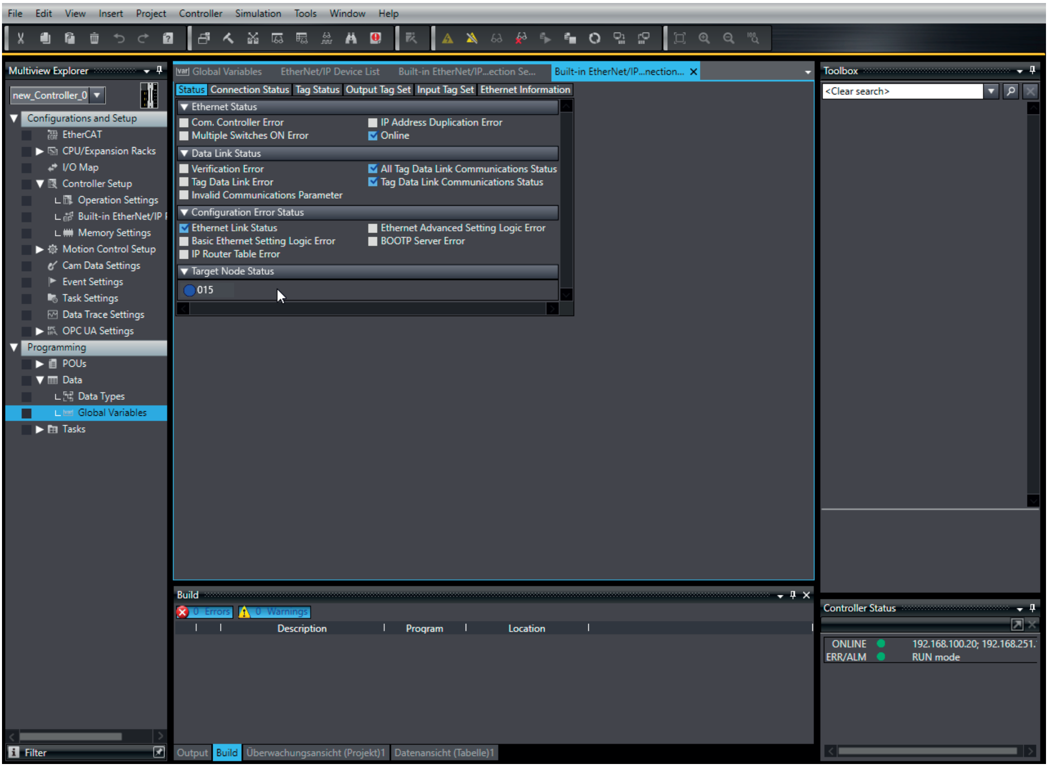
ONLINE ● 192.168.100.20; 192.168.251.
ERR/ALM ● RUN mode

Click on "Monitor" at the relevant IP address to check the connection status.



Node Address	Device	Description
192.168.100.20	Built-in EtherNet/IP Port Settings - Port 1	NX102-1200
192.168.251.1	erNet/IP Port Settings - Port 2	NX102-1200

The blue LED shows that the connection from the PLC to the Smart Camera is ok.



Ethernet Status

☐ Com. Controller Error ☐ IP Address Duplication Error
☐ Multiple Switches ON Error ☑ Online

Data Link Status

☐ Verification Error ☑ All Tag Data Link Communications Status
☐ Tag Data Link Error ☑ Tag Data Link Communications Status
☐ Invalid Communications Parameter

Configuration Error Status

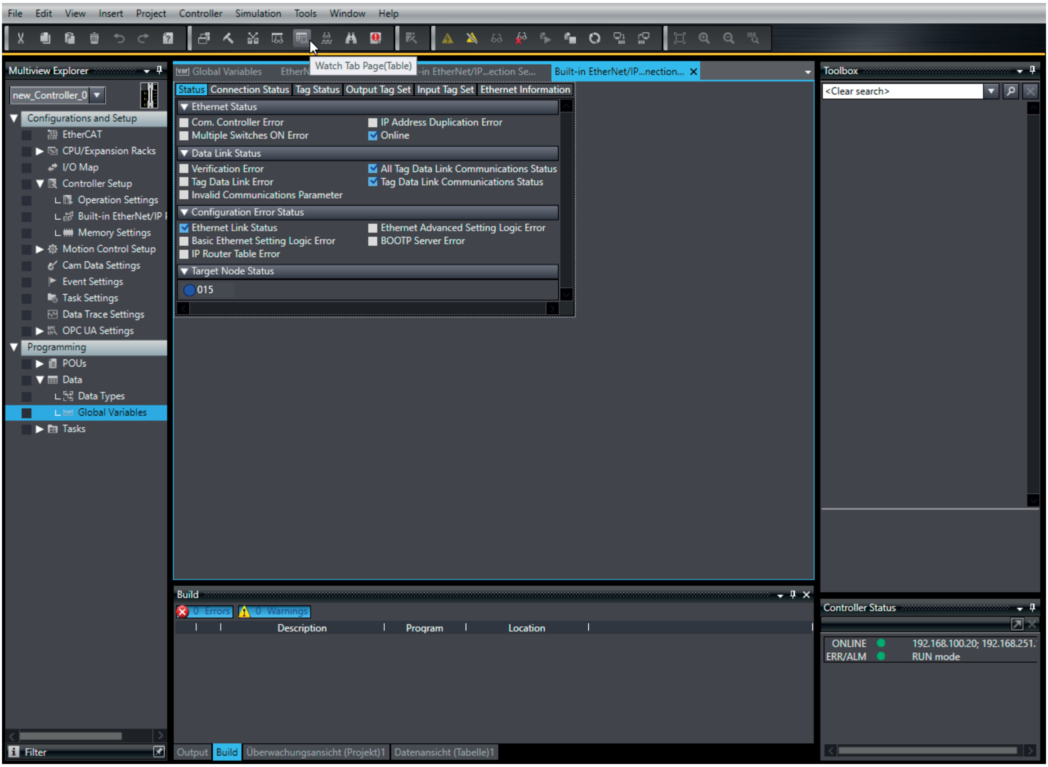
☑ Ethernet Link Status ☐ Ethernet Advanced Setting Logic Error
☐ Basic Ethernet Setting Logic Error ☐ BOOTP Server Error
☐ IP Router Table Error

Target Node Status

● 015

ONLINE ● 192.168.100.20; 192.168.251.
ERR/ALM ● RUN mode

Open "Watch Tab Page (Table)" in order to see all input or output bytes.



Ethernet Status

- □ Com. Controller Error
- □ IP Address Duplication Error
- □ Multiple Switches ON Error
- ☑ Online

Data Link Status

- □ Verification Error
- ☑ All Tag Data Link Communications Status
- □ Tag Data Link Error
- ☑ Tag Data Link Communications Status
- □ Invalid Communications Parameter

Configuration Error Status

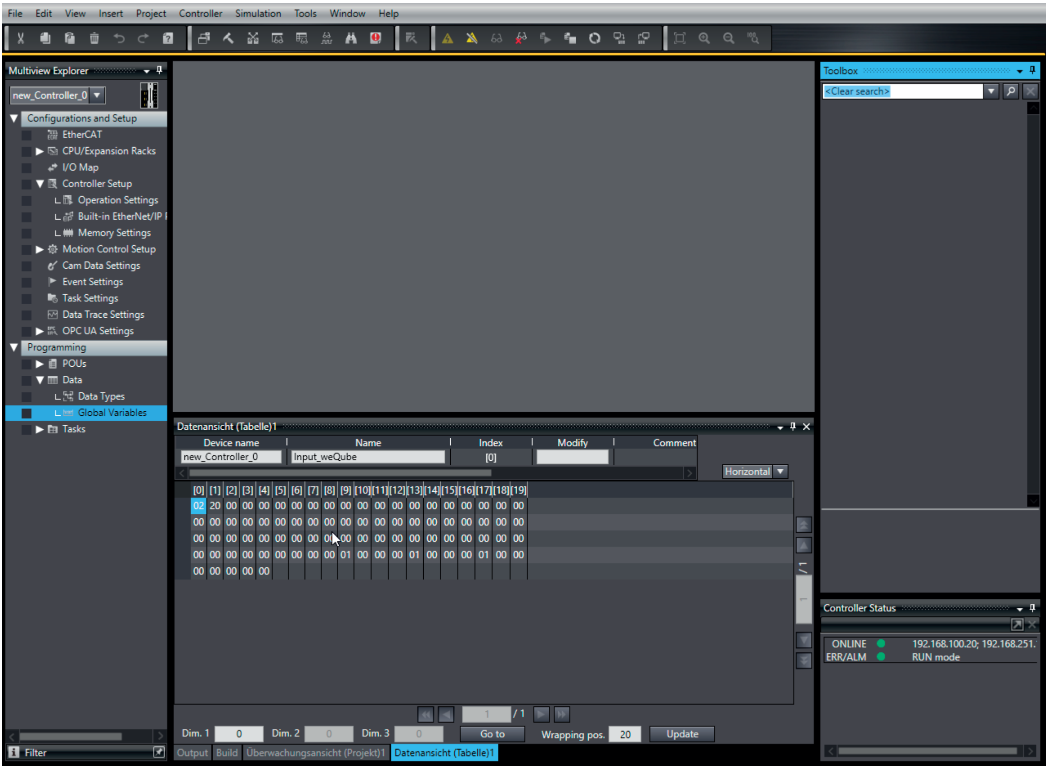
- ☑ Ethernet Link Status
- □ Ethernet Advanced Setting Logic Error
- □ Basic Ethernet Setting Logic Error
- □ BOOTP Server Error
- □ IP Router Table Error

Target Node Status

● 015

ONLINE	●	192.168.100.20; 192.168.251.
ERR/ALM	●	RUN mode

Enter the input or output byte array and check the single byte values.



Datenansicht (Tabelle)1

Device name	Name	Index	Modify	Comment
new_Controller_0	Input_weQube	[0]		

Horizontal

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]
02	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	01	00	00	00	01	00	00	00	01	00	00
00	00	00	00	00															

1 / 1

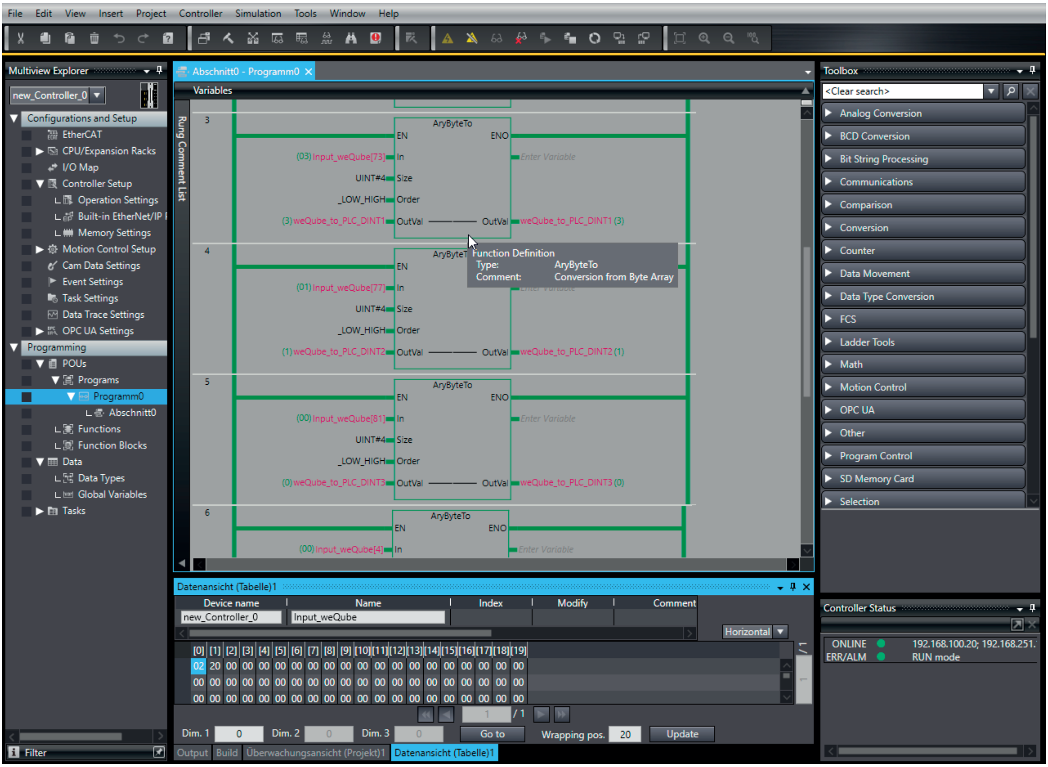
Dim. 1 0 Dim. 2 0 Dim. 3 0 Go to Wrapping pos. 20 Update

ONLINE ● 192.168.100.20; 192.168.251.
ERR/ALM ● RUN mode

8.6 PLC Variables

In order to create variables out of the single bytes open "Programs" -> "Program0" -> "Section0". Add the function "AnyByteTo" in order to create DINT, REAL or string values out of the byte array.

The following example shows how to create a DINT result for the Run Counter. The input (In) is set to the start byte value, the size of DINT is UINT#4, the order is "_LOW_HIGH" and a new global variable must be created with data type DINT.



Rung 3: AryByteTo — EN / ENO; (03) Input_weQube[73] — In; UINT#4 — Size; _LOW_HIGH — Order; (3) weQube_to_PLC_DINT1 — OutVal; OutVal — weQube_to_PLC_DINT1 (3); Enter Variable

Rung 4: AryByteTo — EN / ENO; (01) Input_weQube[77] — In; UINT#4 — Size; _LOW_HIGH — Order; (1) weQube_to_PLC_DINT2 — OutVal; OutVal — weQube_to_PLC_DINT2 (1)

Function Definition
Type: AryByteTo
Comment: Conversion from Byte Array

Rung 5: AryByteTo — EN / ENO; (00) Input_weQube[81] — In; UINT#4 — Size; _LOW_HIGH — Order; (0) weQube_to_PLC_DINT3 — OutVal; OutVal — weQube_to_PLC_DINT3 (0); Enter Variable

Rung 6: AryByteTo — EN / ENO; (00) Input_weQube[4] — In; Enter Variable

Device name	Name	Index	Modify	Comment
new_Controller_0	Input_weQube			

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]
02	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

1 / 1

Dim. 1: 0 Dim. 2: 0 Dim. 3: 0 Wrapping pos.: 20

ONLINE 192.168.100.20; 192.168.251.
ERR/ALM RUN mode

9. Sample PLC program

The download area for the Smart Camera at www.wenglor.com contains sample PLC projects for various controllers. The projects show examples of the required settings on the controller side for Ethernet/IP communication with the weQube Smart Camera.

Samples are available for the following controllers:

- Allen-Bradley 1769-L18ERM BB1B PLC with Studio 5000 Logix Designer V32
- Omron NX102-1200 PLC with Sysmac Studio Version 1.41.0.10
- Beckhoff TwinCAT 3

How to use the sample PLC programs:

1. Download the sample file from the wenglor website and unzip it.
2. Open the associated uniVision project file in the Smart Camera, save it as a start-up project, and restart the Smart Camera. The following slot configuration is used in uniVision projects:

For Studio 5000 (Rockwell):

- Slot 3: 1-byte output
- Slot 4: 1-byte input
- Slot 5: 32-byte input
- Slot 6: 16-byte input (4 DINT)

For Sysmac Studio (Omron):

- Slot 3: 1-byte output
- Slot 4: 1-byte input
- Slot 5: 64-byte input
- Slot 6: 16-byte input (4 DINT)

For TwinCAT 3 (Beckhoff):

- Slot 3: 1-byte output
- Slot 4: 1-byte input
- Slot 5: 16-byte input (4 REAL)
- Slot 6: 16-byte input (4 DINT)

3. Open the sample PLC program, adjust the network configuration, and transfer the program to the PLC, or activate it on the PLC.