

PLC Examples for Software uniVision 3 DNNF023



Description for PLC Examples

Subject to change without notice
Available as PDF version only
Version 1.3.0
Status: 13.05.2025
www.wenglor.com

Table of contents

- 1. Use for Intended Purpose3
- 2. PROFINET Examples3
- 3. EtherNet/IP Examples and Add-on Instruction3
- 4. EtherCAT Example7
- 5. TCP and UDP Examples8
 - 5.1 Settings in wenglor uniVision 3 Software.....8
 - 5.2 Settings in TIA Software11
 - 5.2.1 Receiving Process Data from Device TCP11
 - 5.2.2 Receiving Process Data from Device UDP14
 - 5.2.3 Sending and Receiving LIMA Commands via TCP.....16
 - 5.3 Settings in Studio 5000.....20
 - 5.3.1 Receiving Process Data from Device TCP20
 - 5.3.2 Sending and Receiving LIMA Commands via TCP.....22
 - 5.4 Settings in TwinCAT324
 - 5.4.1 Receiving Process Data from Device TCP24
 - 5.4.2 Receiving Process Data from Device UDP26
 - 5.4.3 Sending and Receiving LIMA Commands via TCP.....28

1. Use for Intended Purpose

The description contains additional information for the PLC examples. Use the PLC examples in order to understand how to integrate Machine Vision Devices at PLCs.

Supported Machine Vision Devices (with software uniVision 3)

- Smart Camera B60
- Machine Vision Controller MVC

PLC types

- Siemens PLC S7-1200 with TIA Portal V15
- SPS 1769-L18ERM-BB1B from Allen-Bradley with Studio 5000 Logix Designer V32
- Beckhoff TwinCAT3

2. PROFINET Examples

The following PROFINET examples exist:

- PROFINET integration of Smart Camera B60 at TIA (Siemens)
- PROFINET integration of Smart Camera B60 at TwinCAT3 (Beckhoff)
- PROFINET integration of Machine Vision Controller MVC at TIA (Siemens)

NOTE!



- The examples work with the default PROFINET configuration file RTE_Config_H001 for the Smart Camera B60 and with the default PROFINET configuration file RTE_Config_E001 for the Machine Vision Controller MVC.
- For details about PROFINET, check the separate interface protocol “Industrial Ethernet” (www.wenglor.com → DNNF023).

3. EtherNet/IP Examples and Add-on Instruction

The following EtherNet/IP examples exist:

- EtherNet/IP integration of Smart Camera B60 at Studio 5000 (Allen-Bradley)
- EtherNet/IP integration of Machine Vision Controller MVC at Studio 5000 (Allen-Bradley)

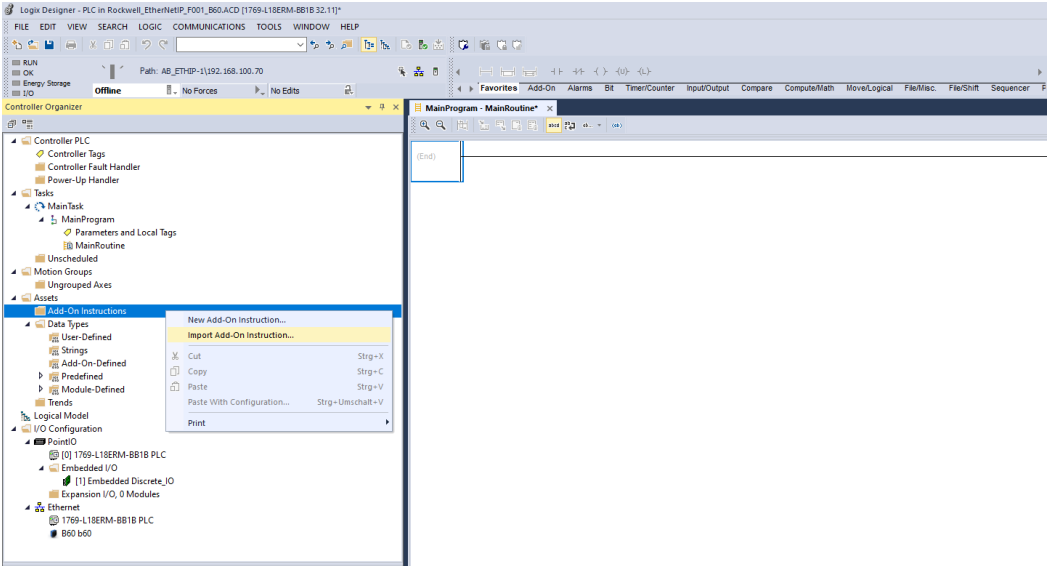
NOTE!



- The examples work with the EtherNet/IP configuration file RTE_Config_F001 for the Smart Camera B60 and with the EtherNet/IP configuration file RTE_Config_G001 for the Machine Vision Controller MVC.
- For details about EtherNet/IP, check the separate interface protocol “Industrial Ethernet” (www.wenglor.com → DNNF023).

The Add-On Instruction for wenglor uniVision 3 helps to integrate B60 Smart Cameras with the configuration file F001 and MVC Machine Vision Controllers with the configuration file G001 at the Rockwell PLC.

- Click on “Import Add-On Instruction...”



- Select the file and continue with the import dialogue.

Import Configuration - AddOnInstruction_wenglor_uniVision3_F001_G001.L5X

Find:

Find/Replace...

Find Within: Final Name

Import Content:

dd-On Instructions

wenglor_uniVision3_F001_G001

Parameters and Local Tags

Routines

References

Tags

Errors/Warnings

Configure Add-On Instruction Properties

Import Name: wenglor_uniVision3_F001_G001

Operation:

Create

References will be imported as configured in the References folders

Final Name:

wenglor_uniVision3_F001_G001

Properties...

Description:

Add-On-Instruction for wenglor uniVision 3:
- Smart Camera B60 with EtherNet/IP for configuration file F001
- Machine Vision Controller MVC

Revision: v1.0

Revision Note:

Vendor: wenglor sensoric GmbH

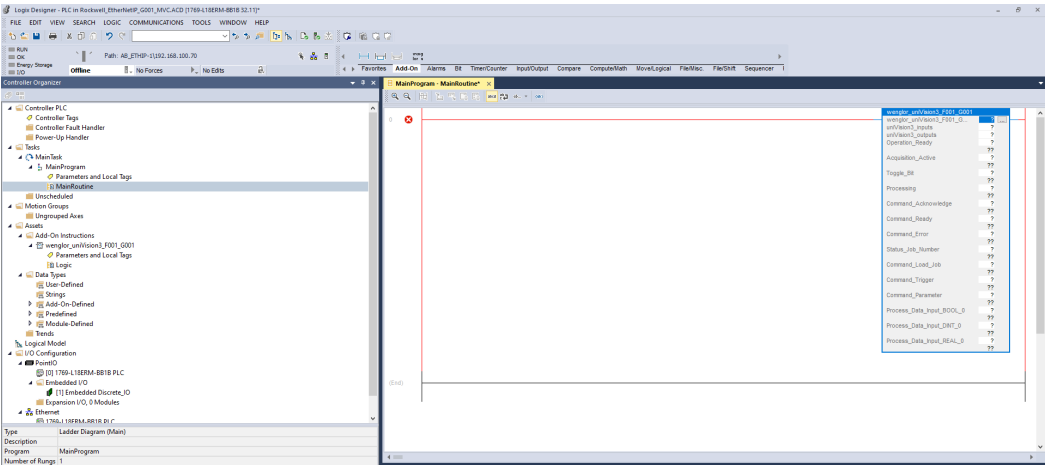
OK

Cancel

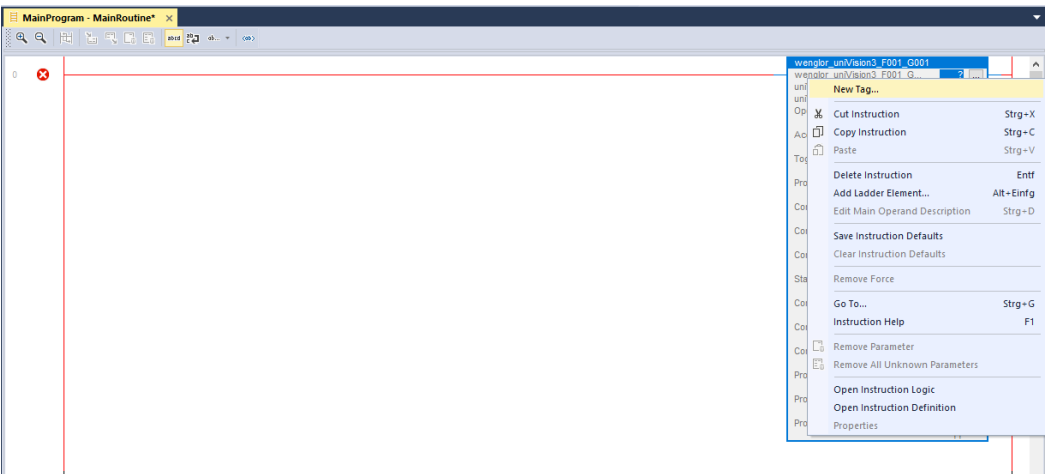
Help

Ready

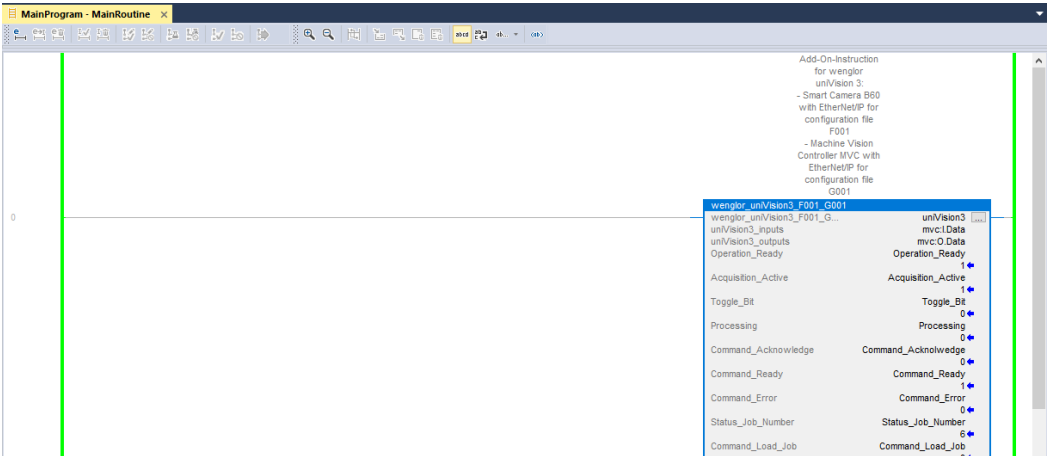
- Add the Add-on Instruction in the Main Routine.



- Create a “New Tag...” for the Add-on Instruction.



- Link all inputs and outputs of the Add-on Instruction so that the program runs without errors. Make sure to link the input and output data for “uniVision3_inputs” and for “uniVision3_outputs”.



4. EtherCAT Example

The following EtherCAT examples exist:

- EtherCAT integration of Machine Vision Controller MVC at TwinCAT3 (Beckhoff)
- EtherCAT integration of Smart Camera B60 at TwinCAT3 (Beckhoff)

NOTE!



- The example works with the EtherCAT configuration file RTE_Config_J001 for the Machine Vision Controller MVC and with the EtherCAT configuration file RTE_Config_I001 for the Smart Camera B60.
- For details about EtherCAT, check the separate interface protocol “Industrial Ethernet” (www.wenglor.com → DNNF023).

5. TCP and UDP Examples

The following TCP and UDP examples exist:

- TCP and UDP integration of uniVision 3 Machine Vision Devices at Siemens PLC
- TCP integration of uniVision 3 Machine Vision Devices at Allen-Bradley (Rockwell) PLC
- TCP and UDP integration of uniVision 3 Machine Vision Devices at Beckhoff PLC

The examples contain the following use cases (if supported by PLC):

- Receiving process data from Device TCP
- Receiving process data from Device UDP
- Sending LIMA commands (e.g. trigger commands) via TCP and receiving LIMA answers



NOTE!

For details about LIMA commands, check the uniVision operating instructions (www.wenglor.com → DNNF023).

5.1 Settings in wenglor uniVision 3 Software

The following settings are necessary in the software wenglor uniVision 3.

- Connect to Machine Vision Device and open job configuration in software uniVision 3.
- Set Trigger Mode to On and Trigger Source to Software in order to use the software LIMA command for triggering.

Job Tree

Module Application

Smart Camera

Image Monochrome

Device Control

Acquisition Control

Image Format Control

Analog Control

Digital IO Control

Error Handling

Optic Control

Light Control

Test Object Status Control

Chunk Data Control

Device Robot Vision

Module Image Code 2D

Add Module

Property	Value	
Exposure Time	200	
Trigger Selector	Exposure Start	
Trigger Mode	On	
Trigger Source	Software	

- Add Device TCP and Device UDP to the job in order to receive the process data at the PLC.

Job Tree

Module Application

Smart Camera

Module Image Code 2D

Device TCP

Device UDP

Add Module

Property	Value	
Process Time [us]	40	
Module State	0	
Output		
Connections	5	
TCP Port	34000	
Preamble		
Postamble		
Delimiter		
String Count	2	
Output Mode	Formatted	
Error Handling	Value Substitution	

- Link results (e.g. Run Counter) to the strings.

Job Tree

Module Image Code 2D

Device TCP

String Count

Error Handling

Formatting Options

Device UDP

String Count

Error Handling

Formatting Options

Add Module

Property	Value	
String#1	Run Counter	
String#2	Result True Count	

- The result “Output” at the device shows a preview of the process data.

Job Tree

Module Application

Smart Camera

Module Image Code 2D

Device TCP

String Count

Error Handling

Formatting Options

Device UDP

Add Module

Property	Value
Process Time [us]	40
Module State	0
Output	+0005888+0000001
Connections	5
TCP Port	34000
Preamble	
Postamble	
Delimiter	
String Count	2
Output Mode	Formatted
Error Handling	Value Substitution

NOTE!

- By default, Output Mode at the devices is set to “Formatted” so that the number of digits is fix independent of the data type of the results. If linking both strings, the default number of characters is 16.
- If needed, adjust the formatting options or further settings (e.g. String Count, Preamble, Delimiter). Check the result “Output” for the number of characters.
- Make sure that the substitution value at the Sub-Module “Error Handling” also has a suitable number of characters.
- For details about the uniVision 3 software, check the uniVision operating instructions (www.wenglor.com → DNNF023).

10

TCP and UDP Examples

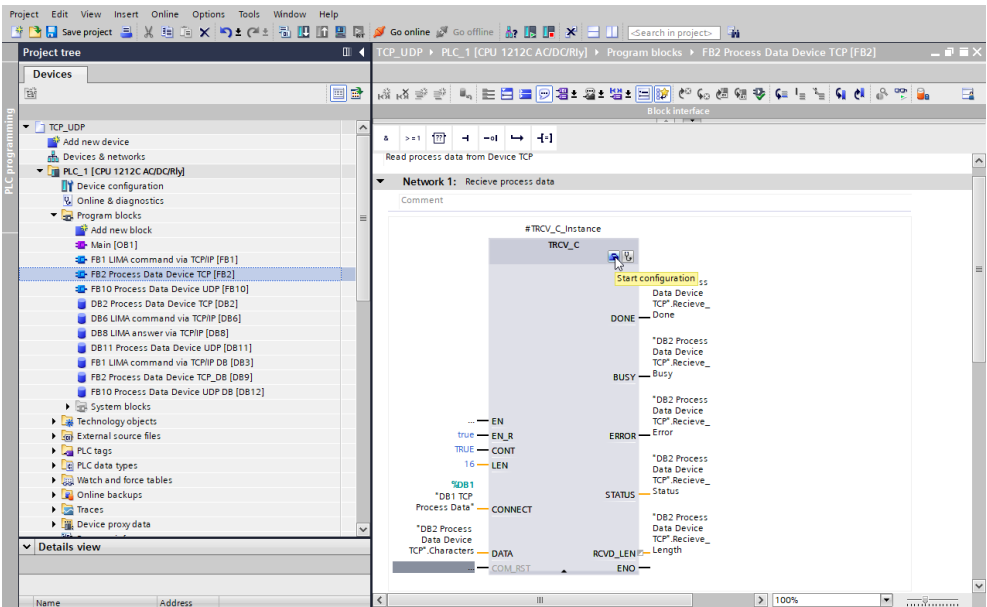
5.2 Settings in TIA Software

5.2.1 Receiving Process Data from Device TCP

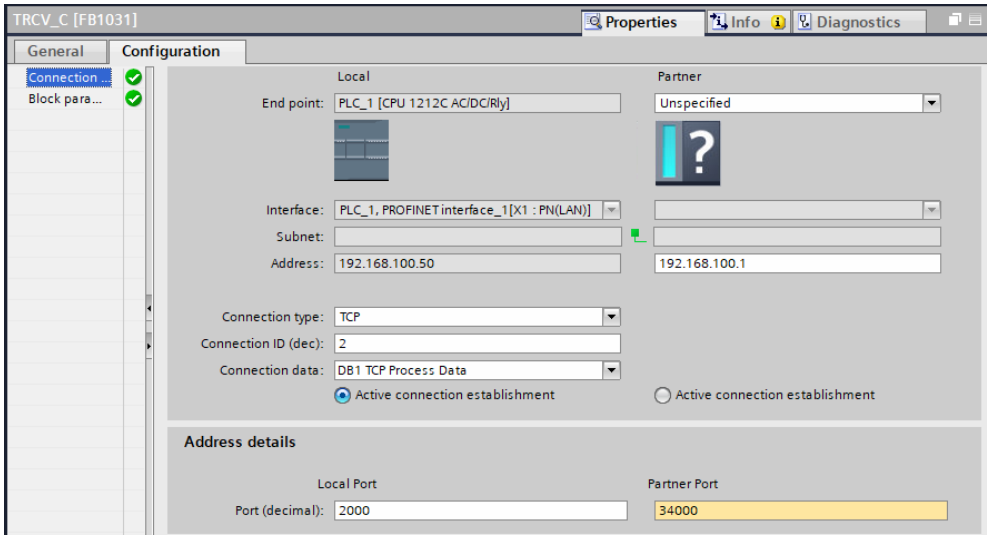
The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 34000

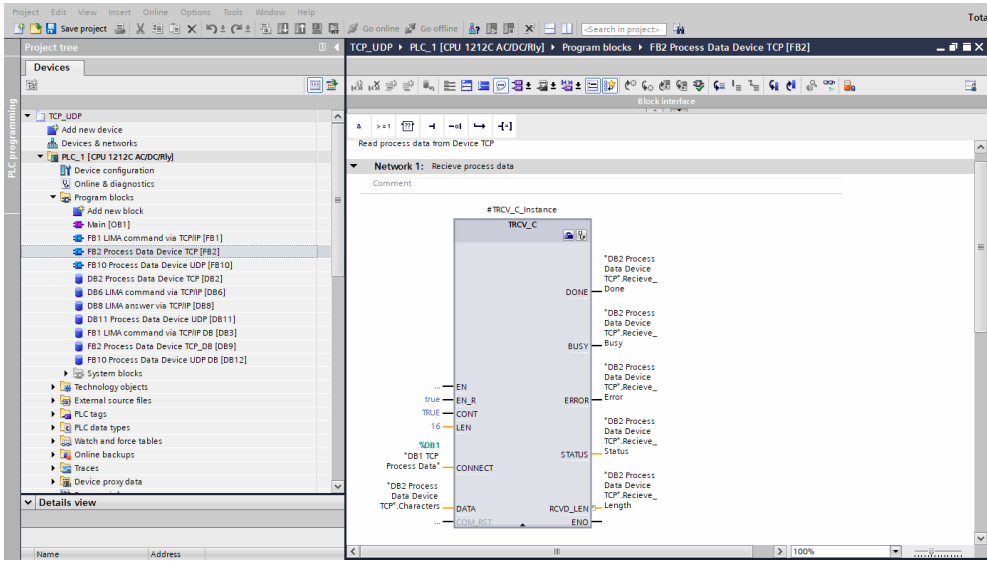
Adjust the example if using another network configuration or another port. To do this, open the function module “FB2 Process Data Device TCP” and click on “Start Configuration” on network 1 “Receive process data”.



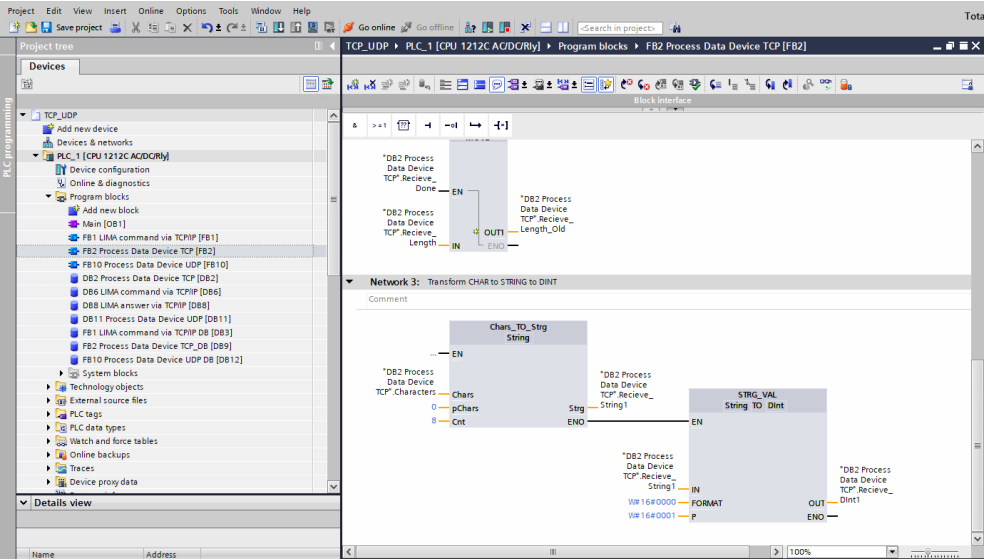
Enter the IP address and the port at "Partner".



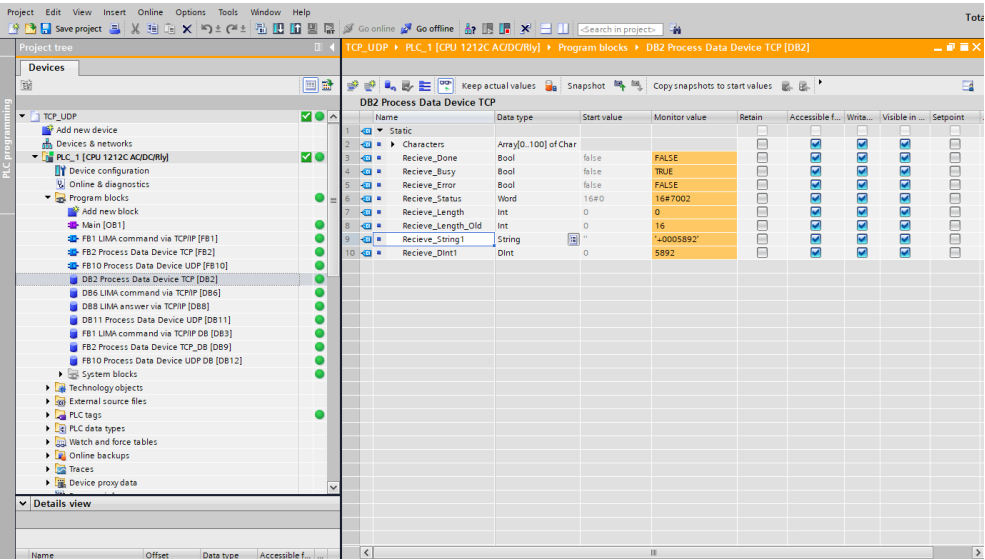
The example is for process data with a length of 16 characters. In case of a different number of characters, adjust the number of characters at "LEN" on network 1 "Receive process data".



The examples include direct conversion for the first eight characters of the string to an integer



Compile the example program, load it on the PLC and go online. Data block “DB2 Process Data Device TCP” receives the process data.



5.2.2 Receiving Process Data from Device UDP

The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 34020

Adjust the example if using another network configuration. To do this, open the function module “DB5 UDP Address” at “System blocks” and enter the IP address of the Machine Vision Device at “REM_IP_ADDR”.

Project tree

PLC programming

Devices

TCP_UDP

Devices & networks

PLC_1 [CPU 1212C AC/DC/PS]

Device configuration

Online & diagnostics

Program blocks

Add new block

Main [OB1]

FB1 LIMA command via TCP/IP [FB1]

FB2 Process Data Device TCP [FB2]

FB10 Process Data Device UDP [FB10]

DB2 Process Data Device TCP [DB2]

DB6 LIMA command via TCP/IP [DB6]

DB8 LIMA answer via TCP/IP [DB8]

DB11 Process Data Device UDP [DB11]

FB1 LIMA command via TCP/IP DB [DB3]

FB2 Process Data Device TCP_DB [DB9]

FB10 Process Data Device UDP DB [DB12]

System blocks

Program resources

TRCV_C [FB1031]

TSEND_C [FB1030]

DB1 TCP Process Data [DB1]

DB4 UDP [DB4]

DB5 UDP Address [DB5]

DB7 TCP LIMA [DB7]

Technology objects

Details view

Name	Offset	Data type	Accessible f...
DB5 UDP Address			
Static			
REM_IP_ADDR		Array(1..4) of U...	
REM_IP_ADDR[1]		USint	192
REM_IP_ADDR[2]		USint	168
REM_IP_ADDR[3]		USint	100
REM_IP_ADDR[4]		USint	1
REM_PORT_NR		UInt	34020
RESERVED		Word	16#0

14

TCP and UDP Examples

Adjust the example if listening on another local port. To do this, open the function module “DB4 UDP” at “System blocks” and enter the port 84E4 in byte (decimal: 34020) at “LOCAL_TSAP_ID”.

The screenshot shows the Siemens STEP 7 LAD editor with the 'DB4 UDP' system block selected in the 'System blocks' category. The block's parameters are configured as follows:

Name	Data type	Start value	Retain	Accessible f.	Writ...	Visible in	Setpoint	Comment
BLOCK_LENGTH	Unit	64						byte length of SDT
ID	CONN_OUTC	4						reference to the co
CONNECTION_TYPE	USINT	19						17: TCP/IP, 18: ISO
ACTIVE_EST	Bool	false						active/passive con
LOCAL_DEVICE_ID	USINT	1						1: local IE interfa
LOCAL_TSAP_ID_LEN	USINT	2						byte length of loca
REM_SUBNET_ID_LEN	USINT	0						byte length of rem
REM_STADDR_LEN	USINT	0						byte length of rem
NEXT_STADDR_LEN	USINT	0						byte length of next
LOCAL_TSAP_ID	Array[1..16] of Byte							TSAP id/local port n
LOCAL_TSAP_ID[1]	Byte	#16#84						TSAP id/local port n
LOCAL_TSAP_ID[2]	Byte	#16#E4						TSAP id/local port n
LOCAL_TSAP_ID[3]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[4]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[5]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[6]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[7]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[8]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[9]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[10]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[11]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[12]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[13]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[14]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[15]	Byte	#16#00						TSAP id/local port n
LOCAL_TSAP_ID[16]	Byte	#16#00						TSAP id/local port n
REM_SUBNET_ID	Array[1..6] of USINT							remote subnet id
REM_STADDR	Array[1..6] of USINT							remote IP address
REM_TSAP_ID	Array[1..16] of Byte							TSAP id/remote port

The example includes direct conversion of the first eight characters of the string to an integer.

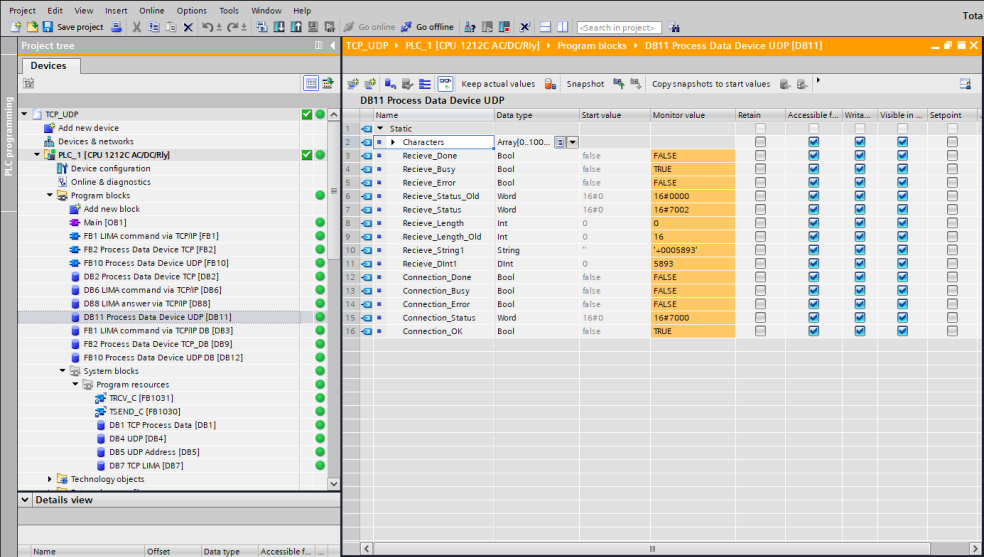
The screenshot shows a network diagram in the Siemens STEP 7 LAD editor. The network is titled 'Network 5: Transform CHAR to STRING to INT'. It illustrates the conversion of the first eight characters of a string into an integer value.

The diagram includes the following components and connections:

- Inputs:** 'Data Device UDP_Receive_Length' (USINT) and 'Data Device UDP_Receive_String' (STRING).
- Block 1:** 'UDF_Receive_Length' (USINT) with 'Length_Old' as an output.
- Block 2:** 'STRG_VAL String to Dint' (STRING to DINT) with 'String' as an input and 'Dint' as an output.
- Block 3:** 'UDF_Receive_String' (STRING) with 'String' as an input and 'Dint' as an output.
- Block 4:** 'UDF_Receive_String' (STRING) with 'String' as an input and 'Dint' as an output.
- Block 5:** 'UDF_Receive_String' (STRING) with 'String' as an input and 'Dint' as an output.

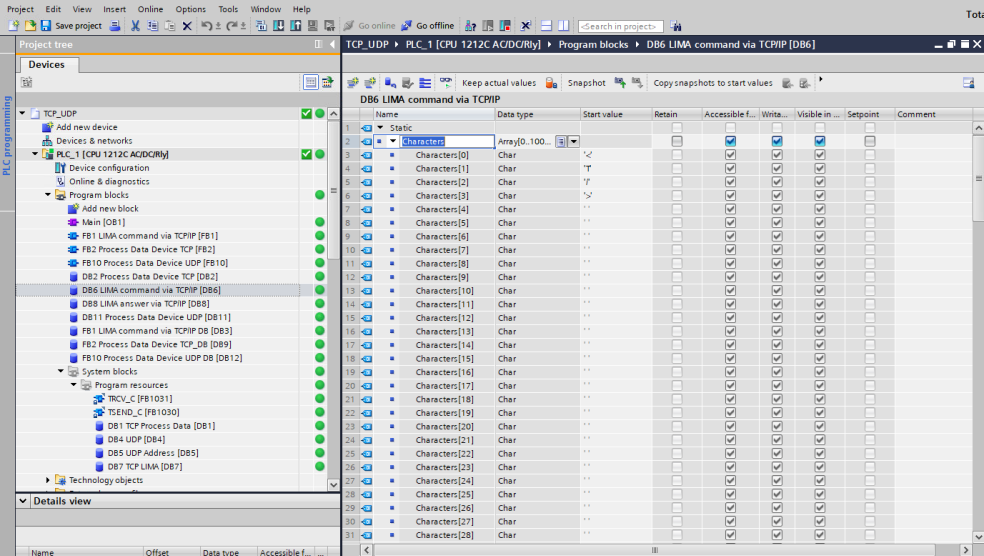
The network is connected to the 'DB11 Process Data Device UDP_Receive' block, which is configured with 'String' as an input and 'Dint' as an output.

Compile the example program, load it on the PLC and go online. Data block “DB11 Process Data Device UDP” receives the process data.



5.2.3 Sending and Receiving LIMA Commands via TCP

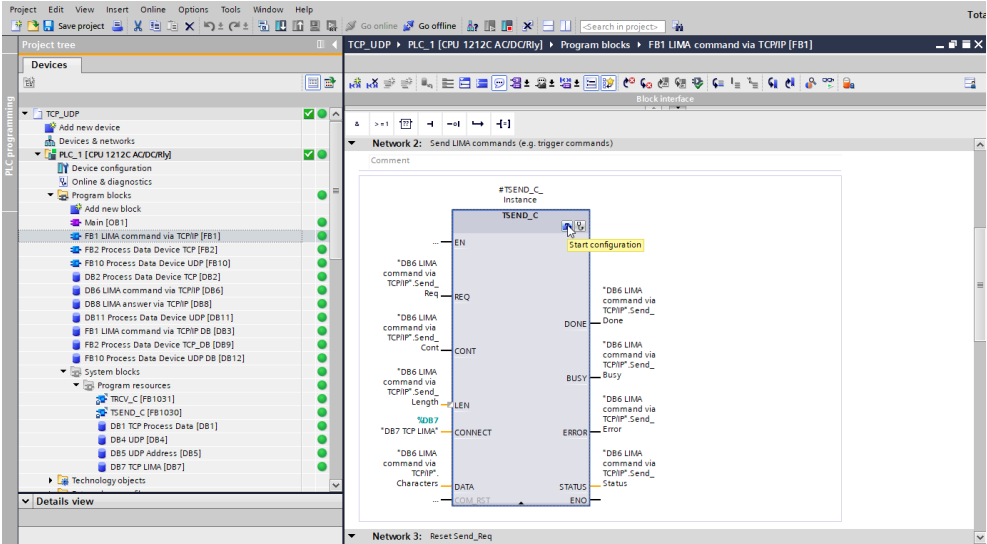
Enter the LIMA command at “DB6 LIMA command via TCP/IP”. The example contains a trigger command by default (<T/>).



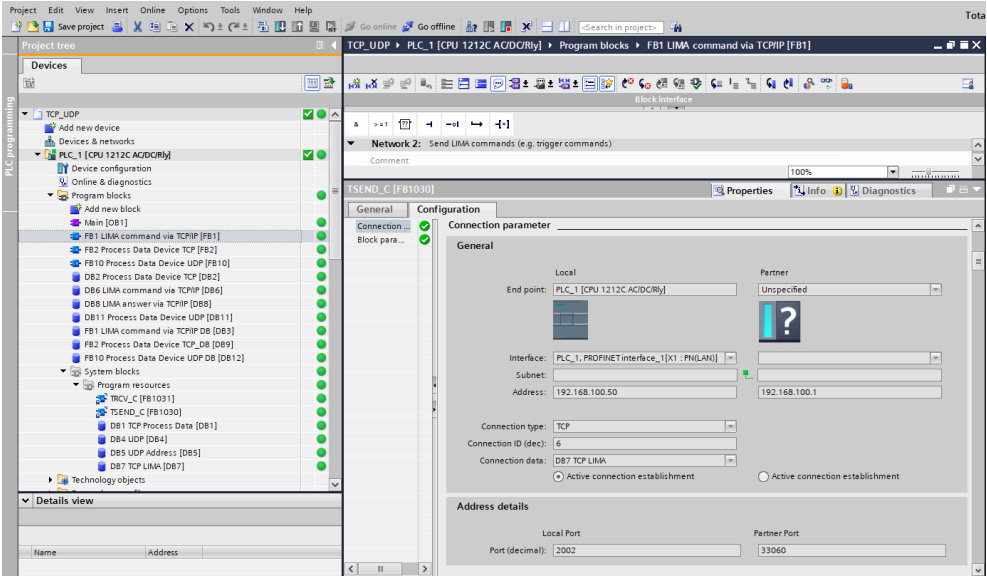
The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 33060 (LIMA Read/Write Limited)

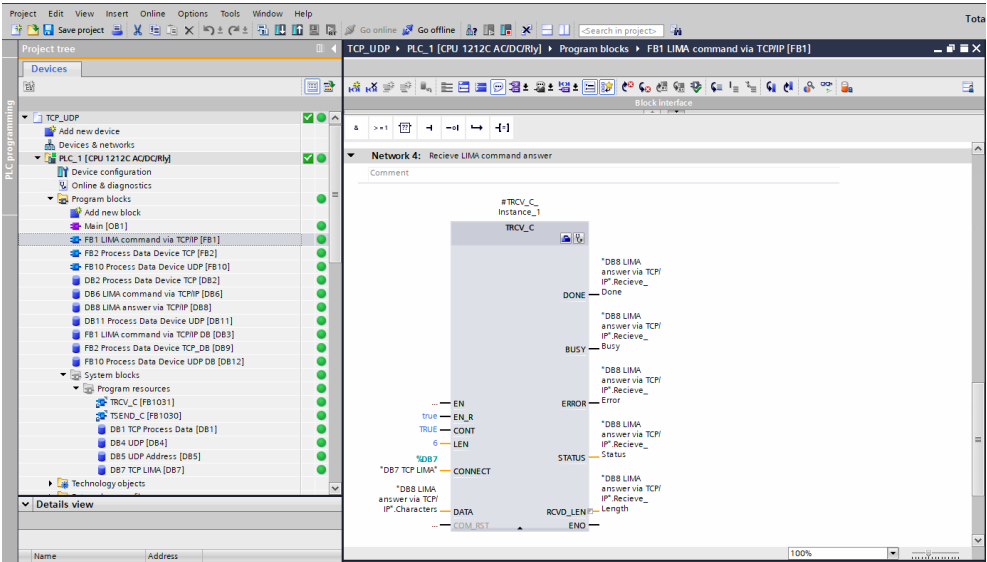
Adjust the example if using another network configuration or another port. To do this, open the function module “FB1 LIMA command via TCP/IP” and click on “Start Configuration” on network 2 “Send LIMA commands (e.g. trigger commands)”.



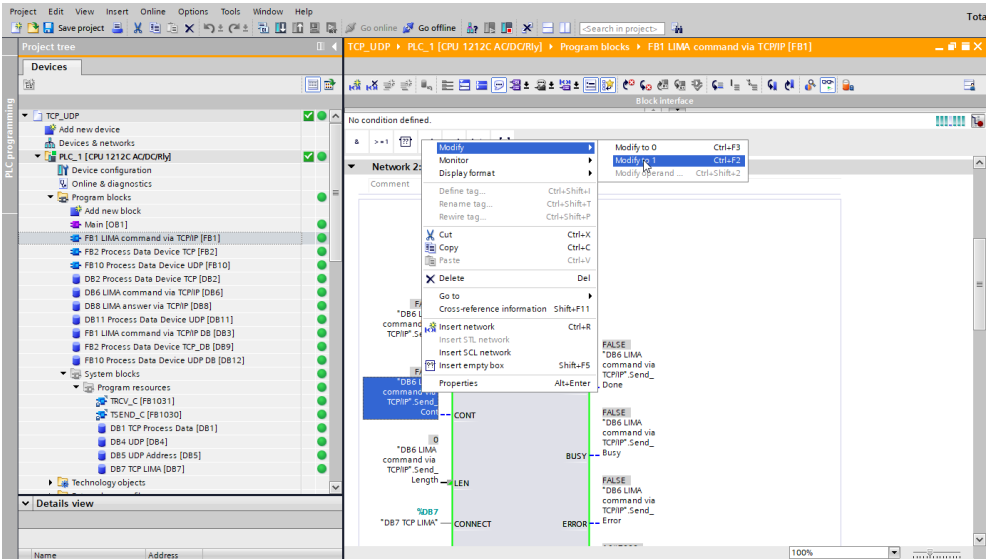
Adjust IP address and port at “Partner”.



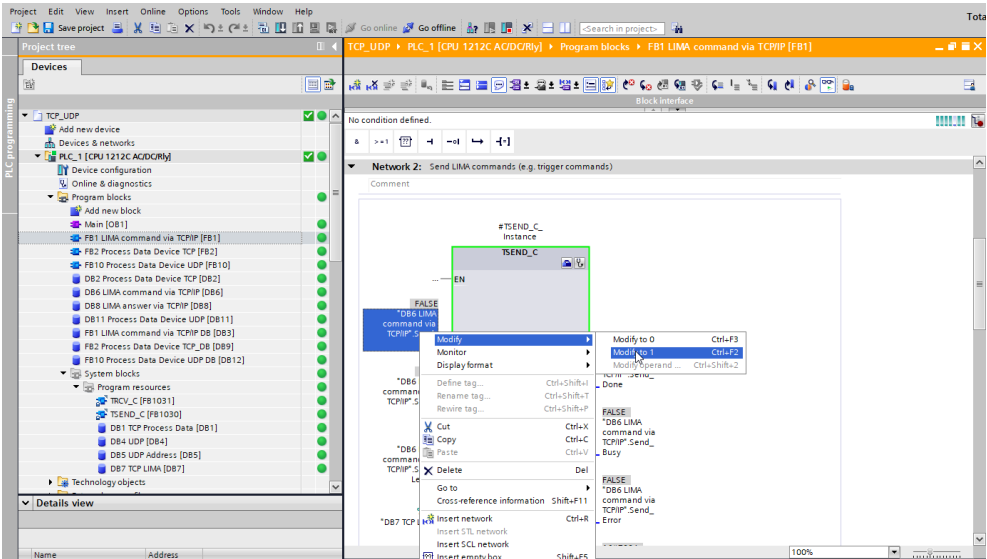
Similarly, click on “Start Configuration” on network 4 “Receive LIMA command answer” and enter the IP address and the port again. Additionally, adjust the number of characters for the LIMA answer on network 4 at “LEN”. The trigger command answer contains six characters (<Tok>).



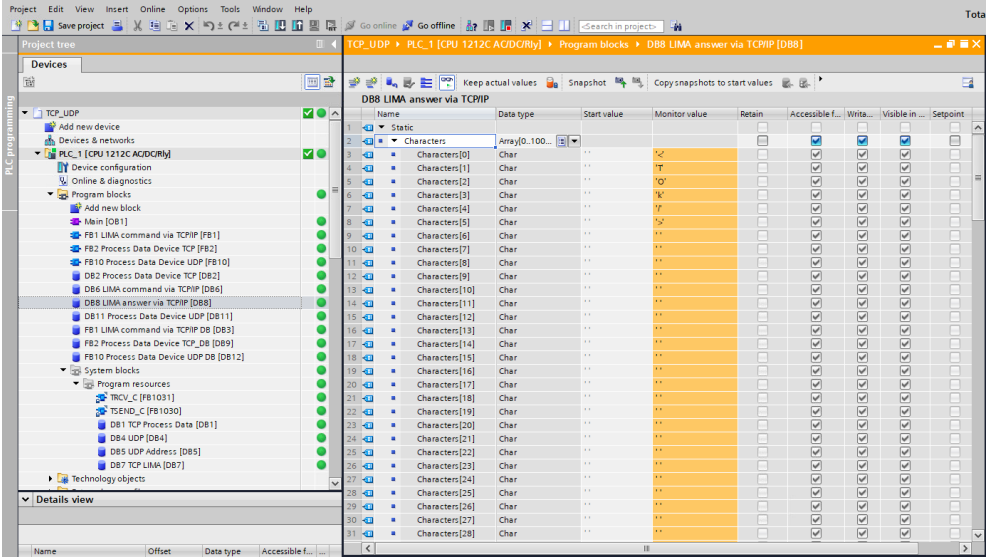
Compile the example program, load it on the PLC and go online. Establish the connection to the Machine Vision Device. To do this, open the function module “FB1 LIMA command via TCP/IP” and set CONT to 1 at network 2 “Send LIMA commands (e.g. trigger commands)”.



The LIMA command is sent to the Machine Vision Device by setting REQ to 1. The example resets REQ immediately to 0 in order to send only one trigger signal. Receive the associated results for the trigger at the process data. For example, use the run counter to check for new results.



The data block “DB8 LIMA answer via TCP/IP” contains the LIMA answer. For the trigger command, the successful answer is <TOK/>



5.3 Settings in Studio 5000

The example “Rockwell_ProcessDataTCP_uniVision3.ACD” shows how to receive process data of Device TCP. The example “Rockwell_LIMA_uniVision3.ACD” shows how to send LIMA commands and how to receive LIMA answers.



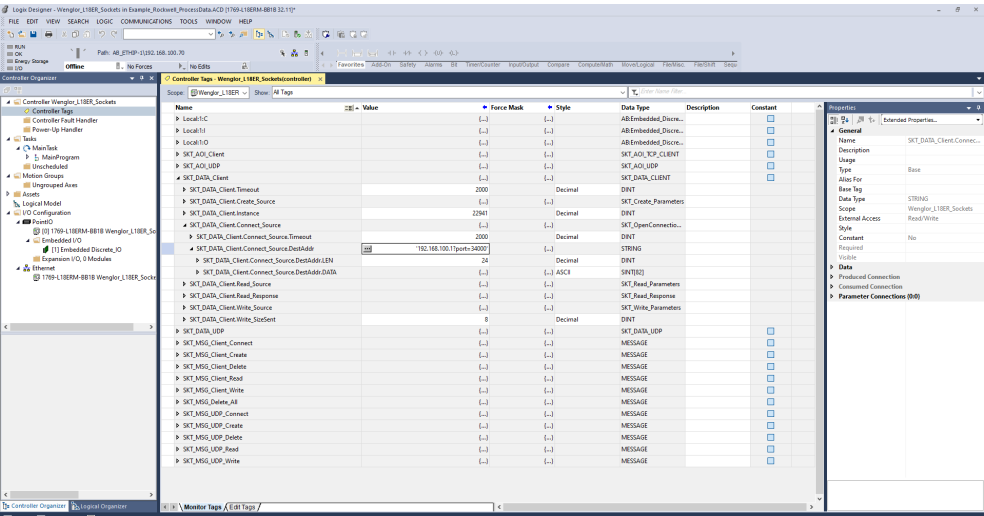
NOTE!
Receiving process data from Device UDP is not supported by Studio 5000, because Rockwell does not support the port 34020 (or higher) for listening on UDP messages.

5.3.1 Receiving Process Data from Device TCP

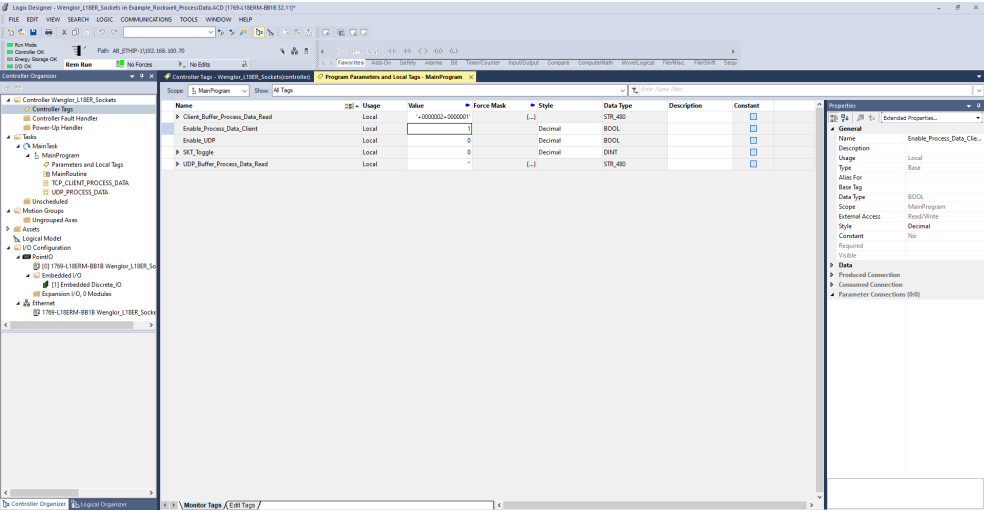
The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 34000

Adjust the example if using another network configuration or another port. To do this, open the Controller Tags and adjust “SKT_DATA_Client.Connect_Source.DestAddr”.

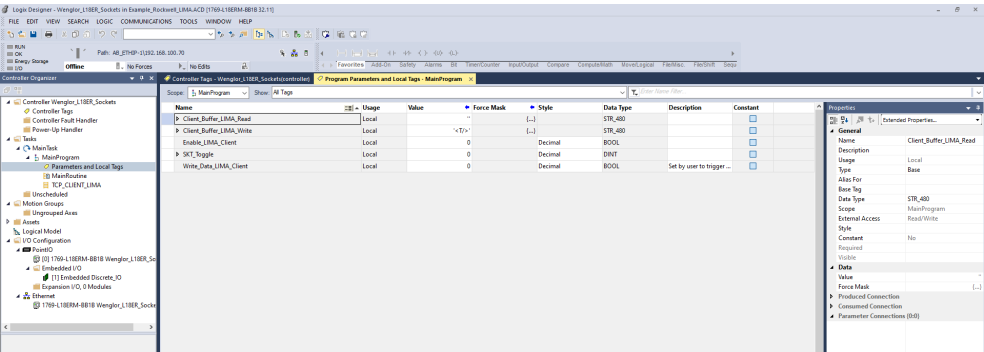


Transfer the example program to the controller and go online. Activate the TCP connection by setting the value “Enable_Process_Data_Client” at “Parameters and Local Tags” to 1. The process data of Device TCP appear at “Client_Buffer_Process_Data_Read”.



5.3.2 Sending and Receiving LIMA Commands via TCP

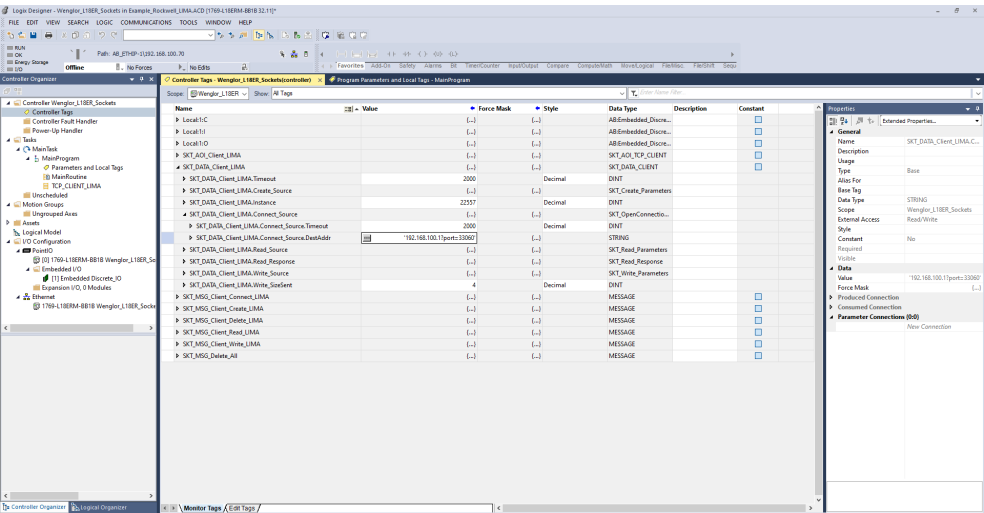
Enter the LIMA command at “Client_Buffer_LIMA_Write” in “Parameters and Local Tags”. The example contains a trigger command by default (<T/>).



The example uses the default network configuration for the Machine Vision Device:

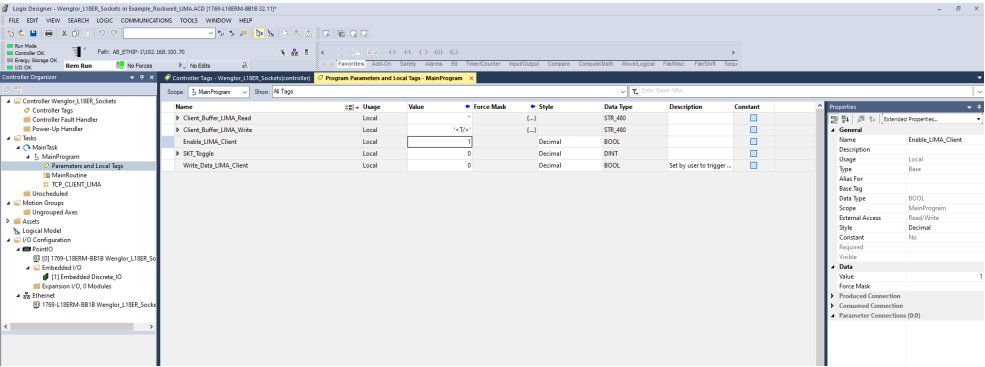
- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 33060 (LIMA Read/Write Limited)

Adjust the example if using another network configuration or another port. To do this, open the “Controller Tags” and enter the IP address and the port at “SKT_Data_Client_LIMA.Connect_Source.DestAddr”.



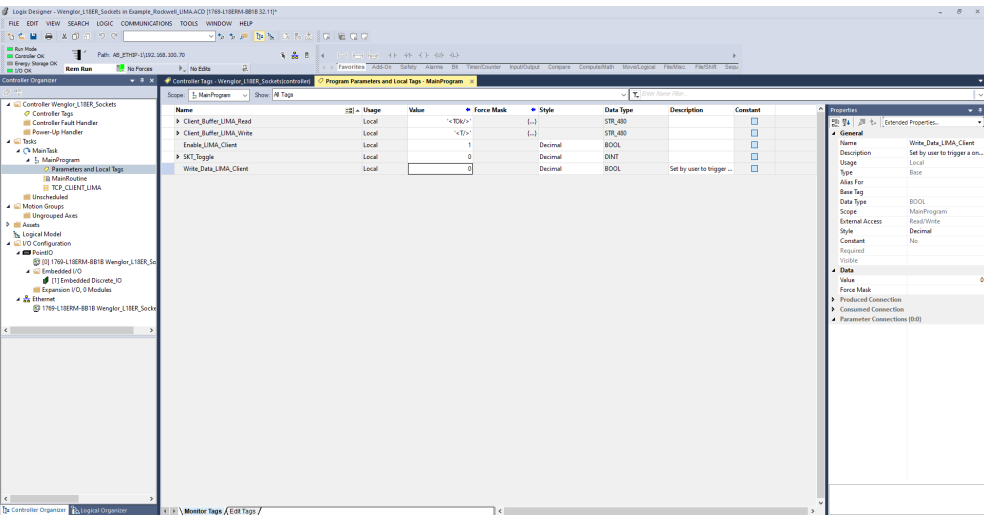
Transfer the example program to the controller and go online.

Activate the TCP connection by setting the value “Enable_LIMA_Client” at “Parameters and Local Tags” to 1.



The LIMA command is sent to the Machine Vision Device by setting “Write_Data_LIMA_Client” to 1. The example resets the value immediately to 0 in order to send only one trigger signal. Receive the associated results for the trigger at the process data. For example, use the run counter to check for new results.

“Client_Buffer_LIMA_Read” contains the LIMA answer. For the trigger command, the successful answer is <TOK>.



5.4 Settings in TwinCAT3

Use the latest TwinCAT3 version (including TF6310 TC3 TCP/IP module). For details, please contact the Beckhoff support.

5.4.1 Receiving Process Data from Device TCP

The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 34000

Adjust the example if using another network configuration or another port. To do this, open "fbClient_ProcessData" at "Main" of "TCP_Communication" and enter the IP address at "sRemoteHost" and the port at "nRemotePort".

```
PROGRAM MAIN
VAR
  sRemoteHost : T_AnsiString := '192.168.100.1'; (* IP address of univision 3 Device *)
  nRemotePort : T_Int := 34000; (* Port number of univision 3 Device - by default 34000 *)
  sPrefix : T_AnsiString := 'CLIENT_ProcessData'; (* TCP/IP client (connection) instance *)
  sProcessDataTCP : T_AnsiString;
  sLINA_Command : T_AnsiString;
  sLINA_Answer : T_AnsiString;
  nRemotePort : T_Int := 34000; (* Default port 34000 *)
  sPrefix : T_AnsiString := 'CLIENT_LINA'; (* TCP/IP client (connection) instance *)

  fbCloseAll : FB_SocketCloseAll;
  bInitialize : BOOL := TRUE;

  // Variables for receiving process data via TCP
  sProcessDataTCP : T_AnsiString;
  sLINA_Command : T_AnsiString;
  sLINA_Answer : T_AnsiString;

  // Variables for sending and receiving LINA commands via TCP/IP
  bSend_LINA_Command : BOOL;
  sLINA_Command : T_AnsiString := 'C';
  sLINA_Answer : T_AnsiString;
END_VAR

// Close open sockets
fbCloseAll(sRemoteHost, nRemotePort, bInitialize);
IF fbCloseAll.Error THEN
  RETURN;
END_IF
bInitialize := FALSE;

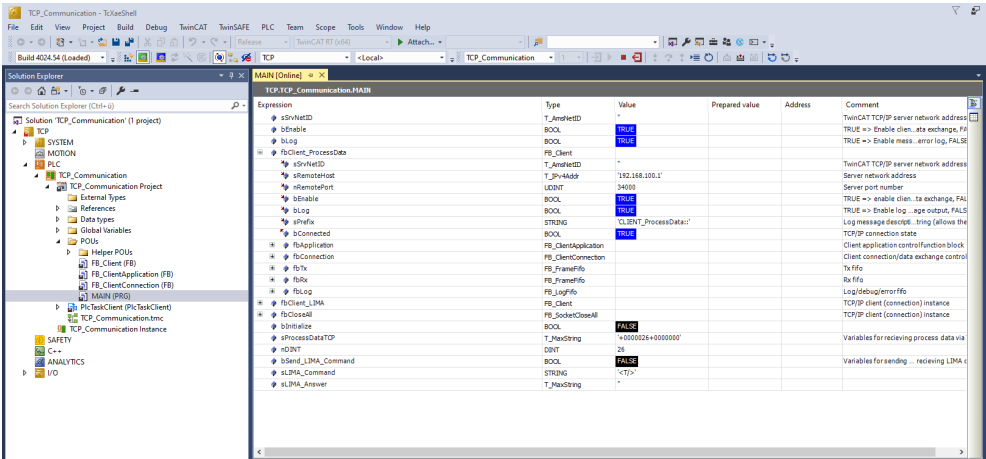
// Receive process data via Device TCP
fbClient_ProcessData.fbx(sRemoteHost, nRemotePort, sPrefix, sProcessDataTCP);
nRemotePort := sProcessDataTCP.Left(4);
sProcessDataTCP := sProcessDataTCP.Right(Length(sProcessDataTCP) - 4);

// Send LINA command via TCP/IP
fbClient_LINA.fbx(sRemoteHost, nRemotePort, sPrefix, sLINA_Command);
bSend_LINA_Command := FALSE;
fbClient_LINA_Answer.fbx(sRemoteHost, nRemotePort, sPrefix, sLINA_Answer);
END_IF

// Receive LINA command via TCP/IP
fbClient_LINA_Answer.fbx(sRemoteHost, nRemotePort, sPrefix, sLINA_Answer);
fbClient_LINA_Answer.RemoveHead(sLINA_Answer);
```

The example includes direct conversion of the first eight characters to an integer.

Enable the program, log in and start it. Process data of Device TCP appear at the variable “sProcessDataTCP”. The integer appears at the variable “nDINT”.

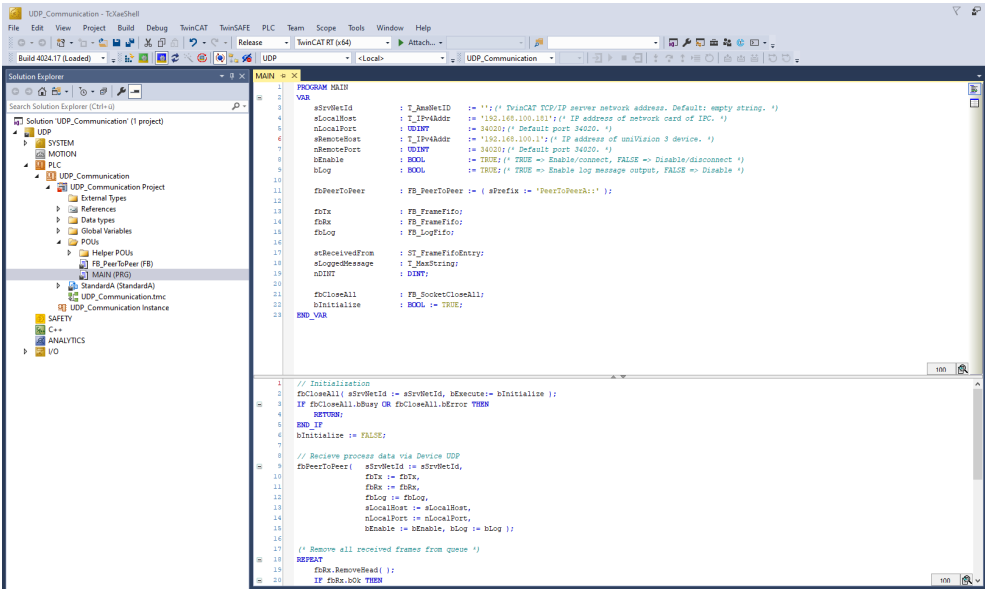


5.4.2 Receiving Process Data from Device UDP

The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 34020

Adjust the example if using another network configuration or another port. To do this, open “Main” of “UDP_Communication” and enter the IP address at “sRemoteHost” and the port at “nLocalPort”.



The example includes direct conversion of the first eight characters to an integer.

Enable the program, log in and start it. Process data of Device UDP appear at the variable “stReceivedFrom” → “sMessage”. The integer appears at the variable “nDINT”.

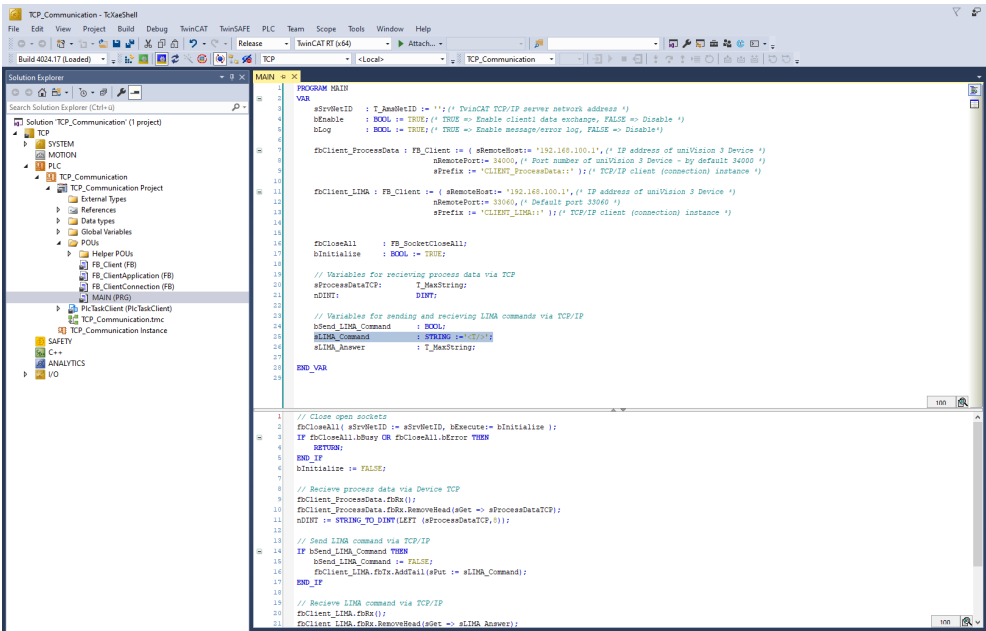
The screenshot displays the TwinCAT 3 IDE interface for the 'UDP_Communication' project. The left pane shows the 'Solution Explorer' with the project structure, including 'UDP', 'SYSTEM', 'MOTION', 'PLC', and 'UDP_Communication'. The main pane shows the 'MAIN' program with a variable declaration table and a ladder logic program.

Expression	Type	Value	Prepared value	Address	Comment
sSrvNetId	T_AmbNetID	*			TwinCAT TCP/IP serve...etwork address
sLocalHost	T_IPv4Addr	'192.168.100.25'			IP address of network card of IPC.
sLocalPort	UDINT	34020			Default port 34020.
sRemoteHost	T_IPv4Addr	'192.168.100.1'			IP address of uni/vision 3 device.
sRemotePort	UDINT	34020			Default port 34020.
sEnable	BOOL	TRUE			TRUE => EnableIcom. FALSE => D
sLog	BOOL	TRUE			TRUE => Enable log...age output, f...
fbPeerToPeer	FB_PeerToPeer				
fbTx	FB_FrameInfo				
fbRx	FB_FrameInfo				
fbLog	FB_LogInfo				
stReceivedFrom	ST_FrameInfoEntry				
sRemoteAddr	ST_SockAddr				Remote address. Strin...aining an (
sMessage	T_AmbString	'+0000002+0000000'			UDP packet data
sLoggedMessage	T_AmbString	'PeerToPeer::Connectionless...			
nDINT	DINT	32			
fbCloseAll	FB_SocketCloseAll				
bInitiate	BOOL	FALSE			

```
1 // Initialization
2 @ fbCloseAll( sSrvNetId := sSrvNetId, bExecut := FALSE, bInitiate := FALSE );
3 IF fbCloseAll.bError := FALSE ON fbCloseAll.bError := FALSE THEN
4   RETURN;
5 END_IF
6 bInitiate := FALSE := FALSE;
7
8 // Receive process data via Device UDP
9 @ fbPeerToPeer( sSrvNetId := sSrvNetId,
10 fbTx := fbTx,
11 fbRx := fbRx,
12 fbLog := fbLog,
13 sLocalHost := sLocalHost, sLocalPort := sLocalPort, sRemoteHost := sRemoteHost, sRemotePort := sRemotePort, sEnable := sEnable, sLog := sLog, sMessage := sMessage, nDINT := nDINT, fbCloseAll := fbCloseAll, bInitiate := bInitiate );
```

5.4.3 Sending and Receiving LIMA Commands via TCP

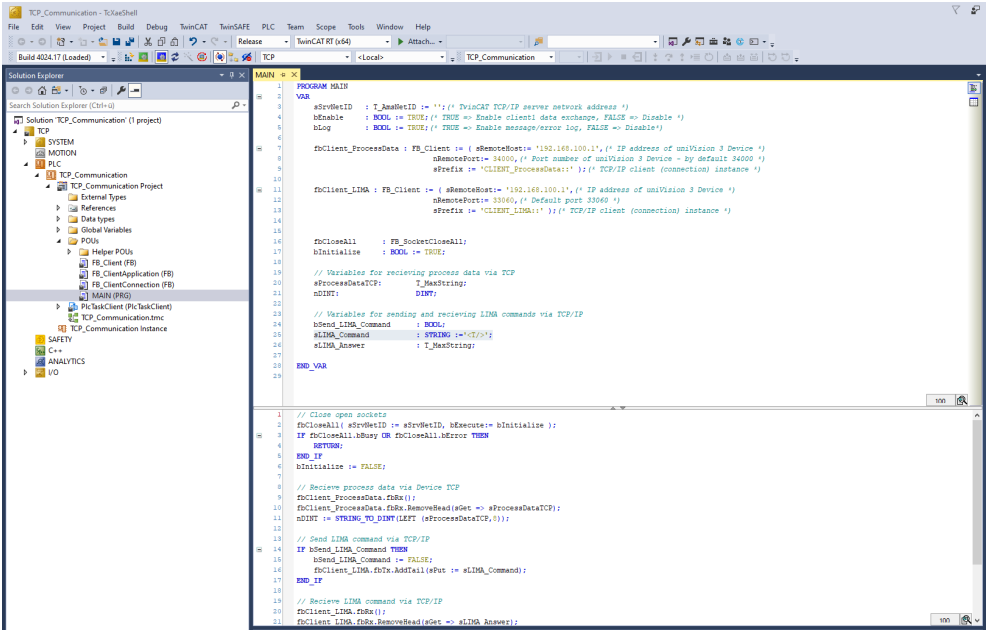
Enter the LIMA command at “sLIMA_Command” in the “Main” of “TCP_Communication”. The example contains a trigger command by default (<T/>).



The example uses the default network configuration for the Machine Vision Device:

- IP address: 192.168.100.1
- Subnet mask: 255.255.255.0
- Port: 33060 (LIMA Read/Write Limited)

Adjust the example if using another network configuration or another port. To do this, open “Main” of “TCP_Communication” and enter the IP address and the port at “fbClient_LIMA”.



Enable the program, log in and start it. Send the LIMA command to the Machine Vision Device by setting “bSend_LIMA_Command” to TRUE. The command may only be sent once, not sent constantly, so that only one trigger is sent. A new command must not be sent, until the LIMA answer to the previous command has been received.

Receive the associated results for the trigger at the process data. For example, use the run counter to check for new results.

“sLIMA_Answer” contains the LIMA answer. For the trigger command, the successful answer is <Tok>.

