

Servo Press Kit YJKP - Integration of host function blocks in Mitsubishi GX Works2

This application note describes how you integrate the host function blocks of the servo press kit YJKP in Mitsubishi GX Works2.

YJKP

Supported systems:

- Tested with Q03UDVCPU
- All compatible devices

Title Servo Press Kit YJKP - Integration of host function blocks in Mitsubishi GX Works2
Version 1.10
Document no. 100172
Originalen
AuthorFesto

Last saved 30.09.2021

Copyright Notice

This documentation is the intellectual property of Festo SE & Co. KG, which also has the exclusive copyright. Any modification of the content, duplication or reprinting of this documentation as well as distribution to third parties can only be made with the express consent of Festo SE & Co. KG.

Festo SE & Co KG reserves the right to make modifications to this document in whole or in part. All brand and product names are trademarks or registered trademarks of their respective owners.

Legal Notice

Hardware, software, operating systems and drivers may only be used for the applications described and only in conjunction with components recommended by Festo SE & Co. KG.

Festo SE & Co. KG does not accept any liability for damages arising from the use of any incorrect or incomplete information contained in this documentation or any information missing therefrom.

Defects resulting from the improper handling of devices and modules are excluded from the warranty.

The data and information specified in this document should not be used for the implementation of safety functions relating to the protection of personnel and machinery.

No liability is accepted for claims for damages arising from a failure or functional defect. In other respects, the regulations with regard to liability from the terms and conditions of delivery, payment and use of software of Festo SE & Co. KG, which can be found at www.festo.com and can be supplied on request, shall apply.

All data contained in this document do not represent guaranteed specifications, particularly with regard to functionality, condition or quality, in the legal sense.

The information in this document serves only as basic information for the implementation of a specific, hypothetical application and is in no way intended as a substitute for the operating instructions of the respective manufacturers and the design and testing of the respective application by the user.

The operating instructions for Festo products can be found at www.festo.com/sp.

Users of this document (application note) must verify that all functions described here also work correctly in the application. By reading this document and adhering to the specifications contained therein, users are also solely responsible for their own application.

Table of contents

- 1 Components/Software used 4**
- 2 Application description..... 5**
 - 2.1 Example Project..... 5
 - 2.2 Integration step by step 5
- 3 Device setup in Mitsubishi GX Works2..... 6**
 - 3.1 Adding a Modbus TCP client to your device 6
 - 3.2 Linking the library 20
 - 3.3 Set up a program..... 22

1 Components/Software used

Type/Name	Version Software/Firmware	Date of manufacture
Servo press kit YJKP	general	--
Application software YJKP (GSAY-A4-F0-Z4-1.3.5)	V1.4.1	--
Firmware controller (CECC-X)	V3.8.14	--
Firmware motor controller (CMMP-AS)	V4.0.1501.2.4	--
CODESYS (V3)	V3.5 SP15	--
Library / function blocks	V1.1.1	--

Table 1.1: 1 Components/Software used

2 Application description

2.1 Example Project

The Examples GX Works2 are available on support portal. They can be used as reference or directly imported in your own project. This will save time and first tests can be run quickly.

**Note**

It refers to the functionality of the application software YJKP version GSAY-1.4.1
For lower version use other appropriate application note, which is also available on the Festo website.

2.2 Integration step by step

This application note describes how you integrate the host function blocks of the servo press kit YJKP in Mitsubishi (GX Works2).

Supported systems:

- Tested with Q03UDVCPU
- All compatible devices

Supported fieldbus:

- Modbus-TCP

Following descriptions are part of the application note:

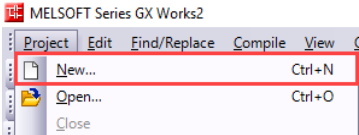
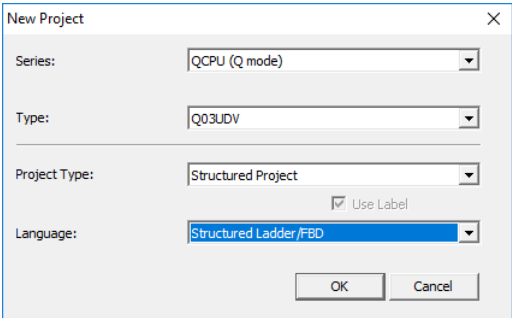
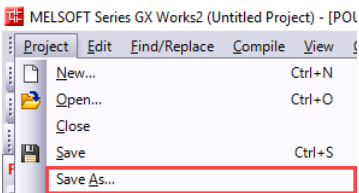
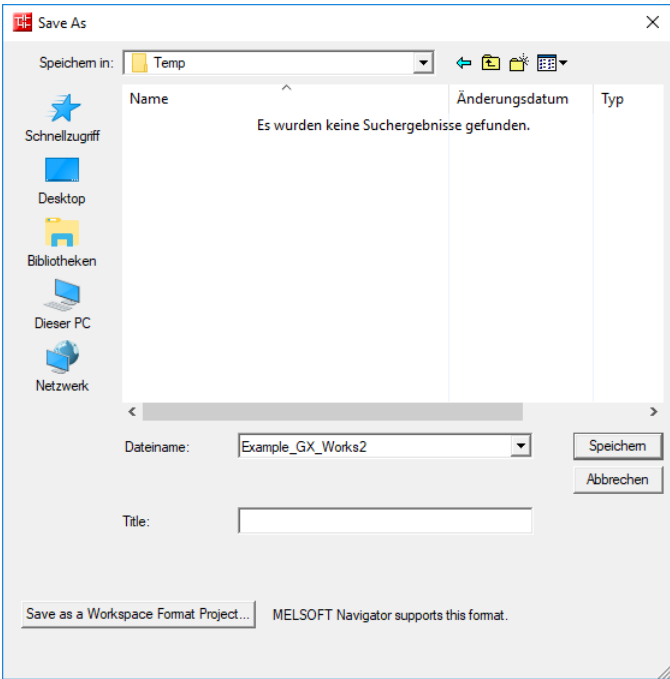
- Device setup in Mitsubishi (GX Works2)
 - Adding a Modbus TCP client to your device
 - Linking the library
 - Set up a program

Content of the download package:

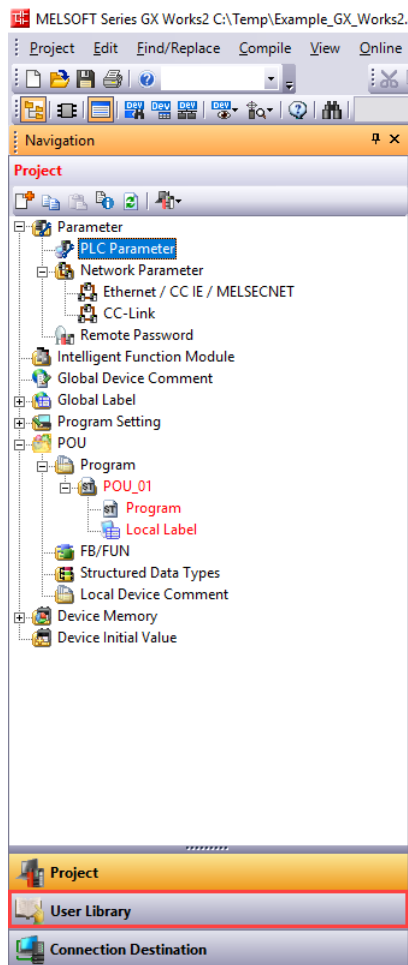
- Link to the Application Notes
- Example
- Library
- Release Note
- Old versions

3 Device setup in Mitsubishi (GX Works2)

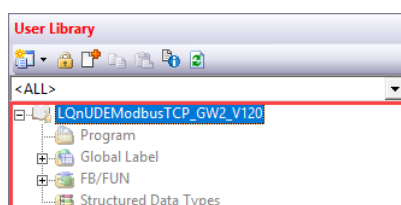
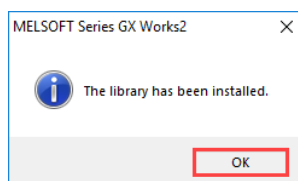
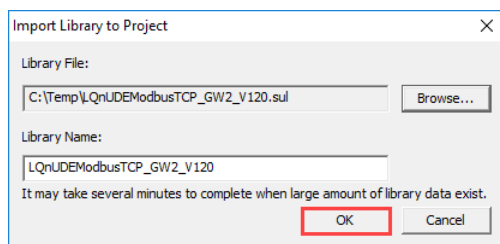
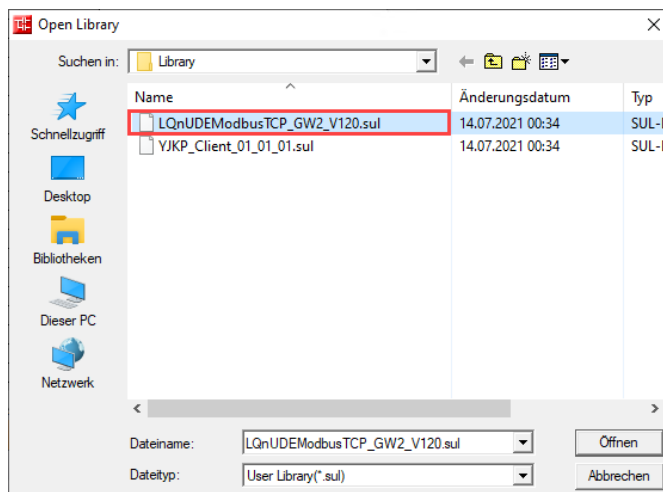
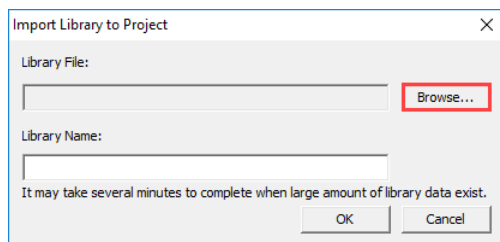
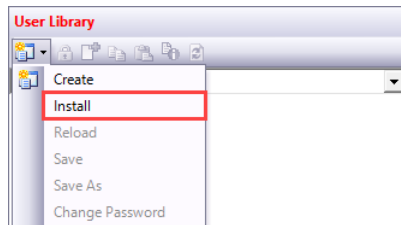
3.1 Adding a Modbus TCP client to your device

No.	Action
1.	Create a new project: “Project” -> “New...” and choose your device and settings. Save your project.    

2. Double-Click on the tab “User library” in the left side bar.
The side bar “User library” will be opened.



3. Click the button “Install” to install a library.
A window “Import Library to Project” will be opened.
Select the library “LQnUDEMdbusTCP_GW2_V120.sul” from the download package.
Confirm the selection with the button “Open”.
Confirm the installation with the button “OK”



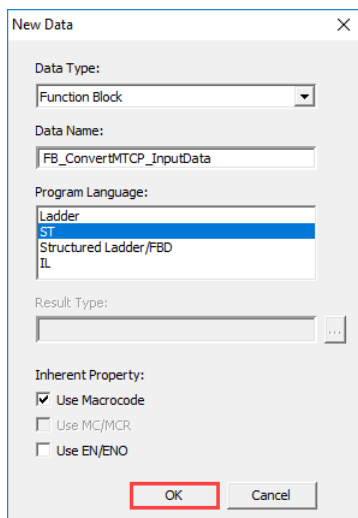
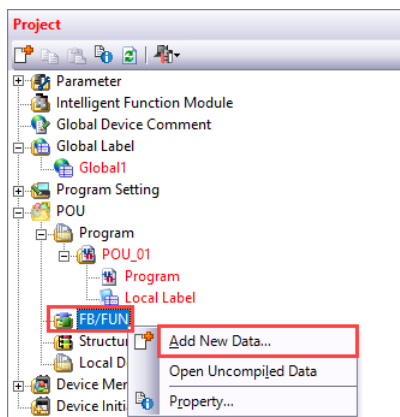
4. Double-Click on “Global1” in the “Project” side bar.
A new window “Global Label Setting Global1” will be opened.
Insert the following code inside the declaration window.

**Important**

For the Modbus TCP communication it must be 64 Bytes IN and OUT!!!

Global Label Setting Global1					
	Class	Label Name	Data Type		Constant
1	VAR_GLOBAL_CONSTANT	gc_uiMaxDataWord	Word[Signed]	...	64
2	VAR_GLOBAL_CONSTANT	gc_uiMaxValueDataWord	Word[Signed]	...	64
3	VAR_GLOBAL	YJKP1	ST_SystemDataRef	...	
4	VAR_GLOBAL	YJKP1_Out	Word[Unsigned]/Bit String[16-bit](1..64)	...	
5	VAR_GLOBAL	YJKP1_In	Word[Unsigned]/Bit String[16-bit](1..64)	...	
6				...	
7				...	

5. Right-Click on “FB/FUN” in the “Project” side bar.
Left-Click on “Add new Data.”
A new window “New Data” will be opened.
Select your settings or use the ones in the example image.
Confirm your selection with the button “OK”.



Project

Parameter

Intelligent Function Module

Global Device Comment

Global Label

Global1

Program Setting

POU

Program

POU_01

Program

Local Label

FB/FUN

FB_ConvertMTCP_InputData

Program

Local Label

Structured Data Types

Local Device Comment

Device Memory

Device Initial Value

6.

Double-Click on “Local Label” of the program “FB_ConvertMTCP_InputData” in the “Project” side bar. A new window “Local Label Setting FB_ConvertMTCP_InputData” will be opened. Insert the following code inside the declaration window.

!

Important

For the Modbus TCP communication it must be 64 Bytes IN and OUT!!!

Function/FB Label Setting F...

	Class	Label Name	Data Type	
1	VAR	swCounter	Word[Signed]	...
2	VAR_INPUT	awInData	Word[Unsigned]/Bit String[16-bit](0..124)	...
3	VAR_OUTPUT	awOutData	Word[Unsigned]/Bit String[16-bit](1..64)	...
4				...
5				...

7.

Double-Click on “Program” of the program “FB_ConvertMTCP_InputData” in the “Project” side bar. A new window “FB_ConvertMTCP_InputData Program” will be opened. Insert the following code inside the implementation window.

!

Important

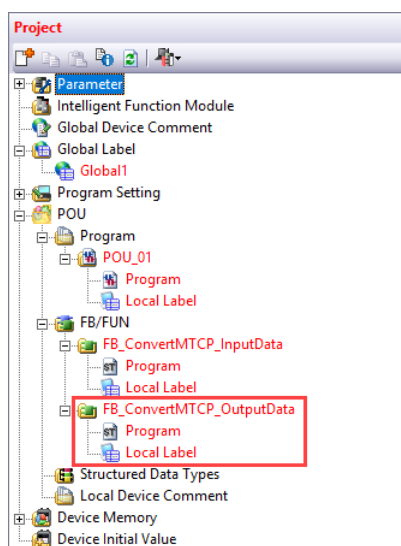
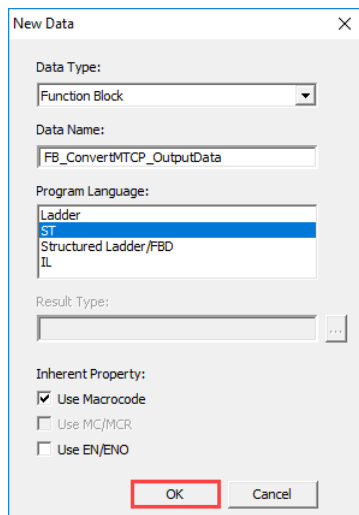
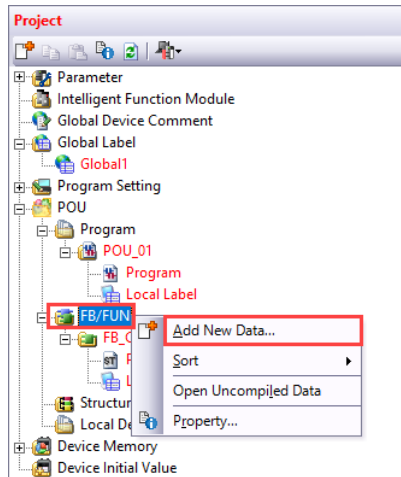
For the Modbus TCP communication it must be 64 Bytes IN and OUT!!!

FB_ConvertMTCP_InputDat...

```
FOR swCounter := 1 TO gc_uiMaxDataWord DO
    awOutData[swCounter] := awInData[swCounter-1];
END_FOR;
```

Page 10 of 23 Application Note – Servo Press Kit YJKP - Integration of host function blocks in Mitsubishi GX Works2 – 1.10

8. Right-Click on “FB/FUN” in the “Project” side bar.
 Left-Click on “Add new Data.”
 A new window “New Data” will be opened.
 Select your settings or use the ones in the example image.
 Confirm your selection with the button “OK”.



9. Double-Click on “Local Label” of the program “FB_ConvertMTCP_OutputData” in the “Project” side bar. A new window “Local Label Setting FB_ConvertMTCP_OutputData” will be opened. Insert the following code inside the declaration window.



Important

For the Modbus TCP communication it must be 64 Bytes IN and OUT!!!

Function/FB Label Setting F...			
	Class	Label Name	Data Type
1	VAR	swCounter	Word[Signed]
2	VAR_INPUT	awInData	Word[Unsigned]/Bit String[16-bit](1..64)
3	VAR_OUTPUT	awOutData	Word[Unsigned]/Bit String[16-bit](0..122)
4			
5			

10. Double-Click on “Program” of the program “FB_ConvertMTCP_OutputData” in the “Project” side bar. A new window “FB_ConvertMTCP_OutputData Program” will be opened. Insert the following code inside the implementation window.



Important

For the Modbus TCP communication it must be 64 Bytes IN and OUT!!!

```

FOR swCounter := 1 TO gc_uiMaxValueDataWord DO
  (* Copy internal array data byte to output data *)
  awOutData[swCounter-1] := awInData[swCounter];
END_FOR;

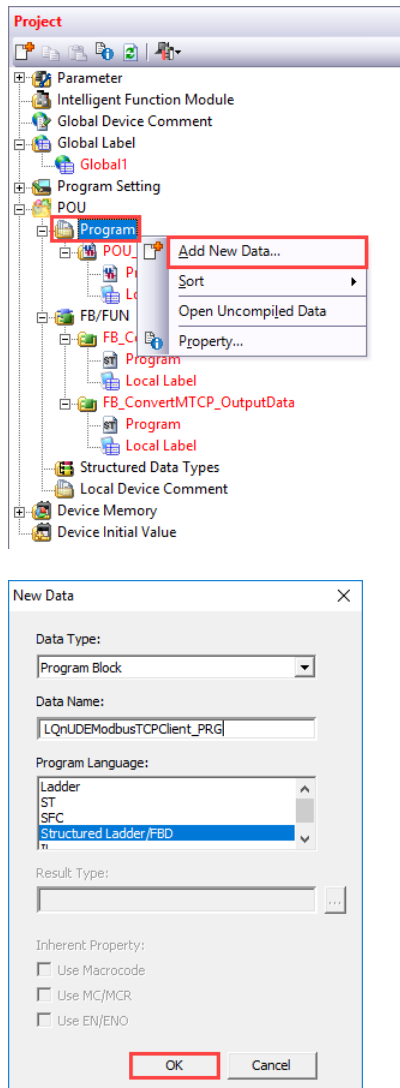
```

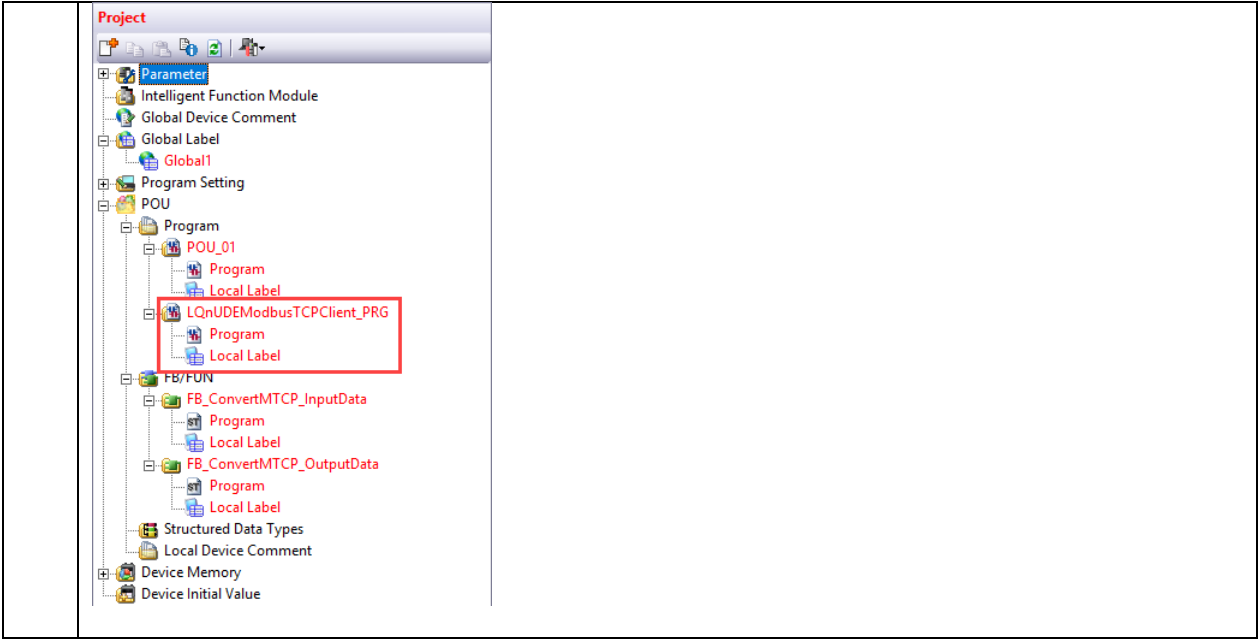
11. Right-Click on “Program” in the “Project” side bar.
 Left-Click on “Add new Data.”
 A new window “New Data” will be opened.
 Select your settings or use the ones in the example image.
 Confirm your selection with the button “OK”.



Important

The implementation in the image is only an example! Please implement it according to your own standard!!!





12. Double-Click on “Local Label” of the program “LQnUDEMdbusTCPClient_PRG” in the “Project” side bar.
A new window “Local Label Setting LQnUDEMdbusTCPClient_PRG” will be opened.
Insert the following code inside the declaration window.

**Important**

The implementation in the image is only an example! Please implement it according to your own standard!!!

Local Label Setting LQnUDE... x				
	Class	Label Name	Data Type	Co
1	VAR	BitOrRegister	Bit	...
2	VAR	CommandCompleted	Bit	...
3	VAR	CommandCompletedOK	Bit	...
4	VAR	CommandCompletedOKCount	Word[Signed]	...
5	VAR	EnableConnection	Bit	...
6	VAR	FB_ConvertMTCP_InputData_1	FB_ConvertMTCP_InputData	...
7	VAR	FB_ConvertMTCP_OutputData_1	FB_ConvertMTCP_OutputData	...
8	VAR	FirstAddress	Bit	...
9	VAR	GeneralError	Bit	...
10	VAR	GeneralErrorCount	Word[Signed]	...
11	VAR	InitComplete	Bit	...
12	VAR	InitError	Bit	...
13	VAR	InitErrorCode	Word[Unsigned]/Bit String[16-bit]	...
14	VAR	InitErrorLog	Word[Signed]	...
15	VAR	InvalidCommand	Bit	...
16	VAR	InvalidCommandCount	Word[Signed]	...
17	VAR	InvalidResponse	Bit	...
18	VAR	InvalidResponseCount	Word[Signed]	...
19	VAR	LinkError	Bit	...
20	VAR	ModbusException	Bit	...
21	VAR	ModbusExceptionCode	Word[Signed]	...
22	VAR	ModbusExceptionCodeLog	Word[Signed]	...
23	VAR	ModbusExceptionCount	Word[Signed]	...
24	VAR	ModbusTcpClient	LQnUDEMdbusTCPClient	...
25	VAR	NoOfReadBits	Word[Signed]	...
26	VAR	NoOfReadRegisters	Word[Signed]	...
27	VAR	NoOfWriteBits	Word[Signed]	...
28	VAR	NoOfWriteRegisters	Word[Signed]	...
29	VAR	OpenComplete	Bit	...
30	VAR	OpenError	Bit	...
31	VAR	OpenErrorCode	Word[Unsigned]/Bit String[16-bit]	...
32	VAR	OpenErrorCount	Word[Signed]	...
33	VAR	OpenErrorLog	Word[Signed]	...
34	VAR	Read	Bit	...
35	VAR	ReadData	Word[Unsigned]/Bit String[16-bit](0..124)	...
36	VAR	ReadDataByteCount	Word[Signed]	...
37	VAR	ReadDataWordCount	Word[Signed]	...
38	VAR	ReadInputRegister_Storage	Word[Unsigned]/Bit String[16-bit](0..124)	...
39	VAR	ReadNoOfDevs	Word[Signed]	...
40	VAR	ReadOnlyDevices	Bit	...
41	VAR	ReadStartAddr	Word[Unsigned]/Bit String[16-bit]	...
42	VAR	ResponseTimeoutError	Bit	...
43	VAR	ResponseTimeoutErrorCount	Word[Signed]	...
44	VAR	Sequence	Word[Signed]	...
45	VAR	SingleDevices	Bit	...
46	VAR	Start	Bit	...
47	VAR	StartSend	Bit	...
48	VAR	Write	Bit	...
49	VAR	WriteData	Word[Unsigned]/Bit String[16-bit](0..122)	...
50	VAR	WriteNoOfDevs	Word[Signed]	...
51	VAR	WriteStartAddr	Word[Unsigned]/Bit String[16-bit]	...

13.

Double-Click on “Program” of the program “LQnUDEMdbusTCPClient_PRG” in the “Project” side bar. A new window “LQnUDEMdbusTCPClient_PRG Program” will be opened. Insert the following code inside the implementation window. Change the IP-Address “RemotelPAddress” of the function block “ModbusTcpClient” to your IP-Address of your device.



Important

The implementation in the image is only an example! Please implement it according to your own standard!!!

Copyright © Mitsubishi Electric, 2011

All examples are only intended to improve understanding of the functionality and handling of the product. In view of the wide range of applications for this product, users must acquire sufficient knowledge themselves in order to ensure that it is correctly used in their specific application. Persons responsible for the application and the product must themselves ensure that each application is in compliance with all relevant requirements, standards and legislation in respect to configuration and safety.

Mitsubishi Electric cannot assume any liability if these examples are used in real applications.

Program example for a Modbus TCP/IP Client Q06UDEH PLC system

On the first scan provide write data initial values and reset the sequence counter to the initial value

MOV

EN

ENO

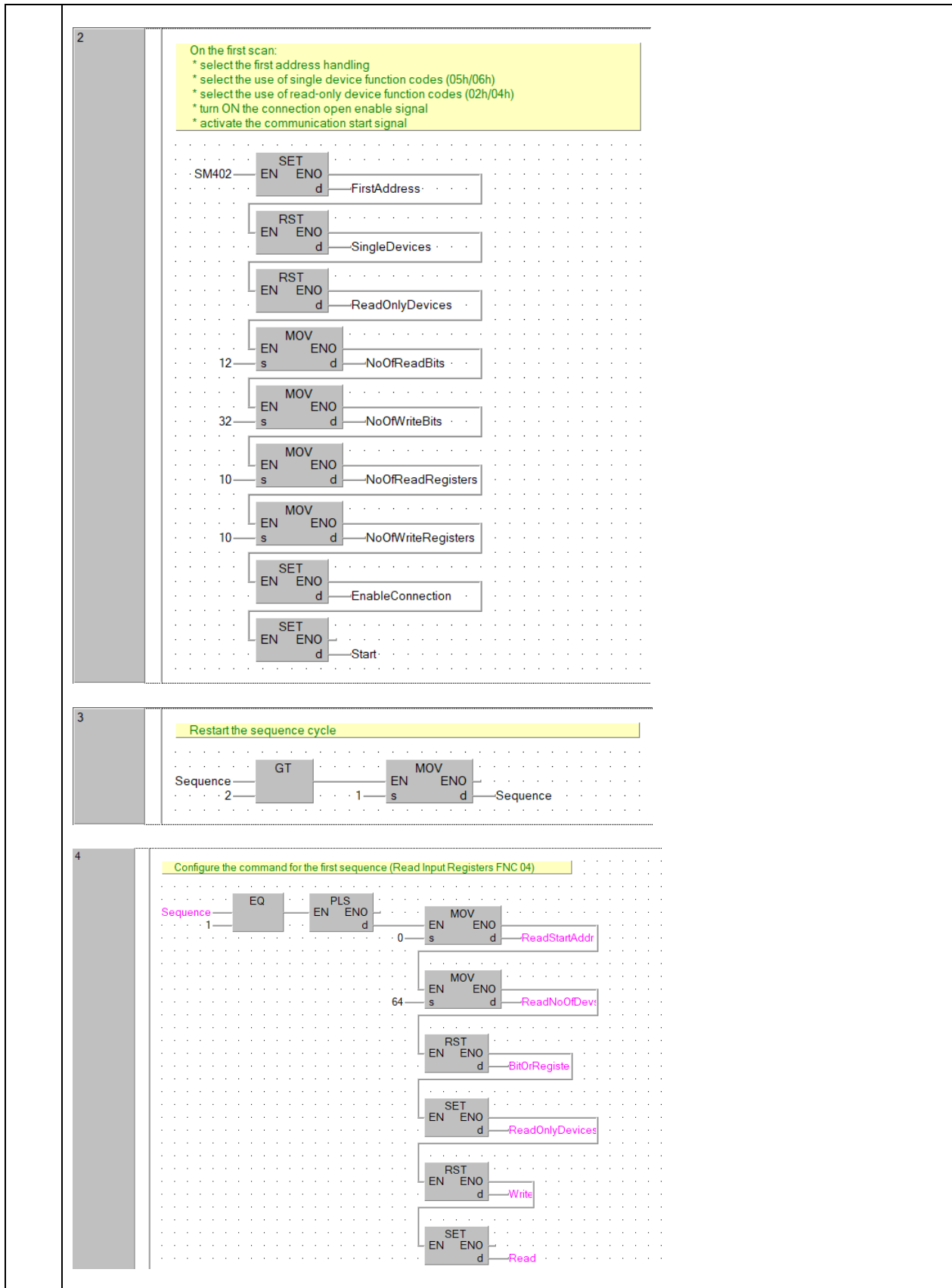
s

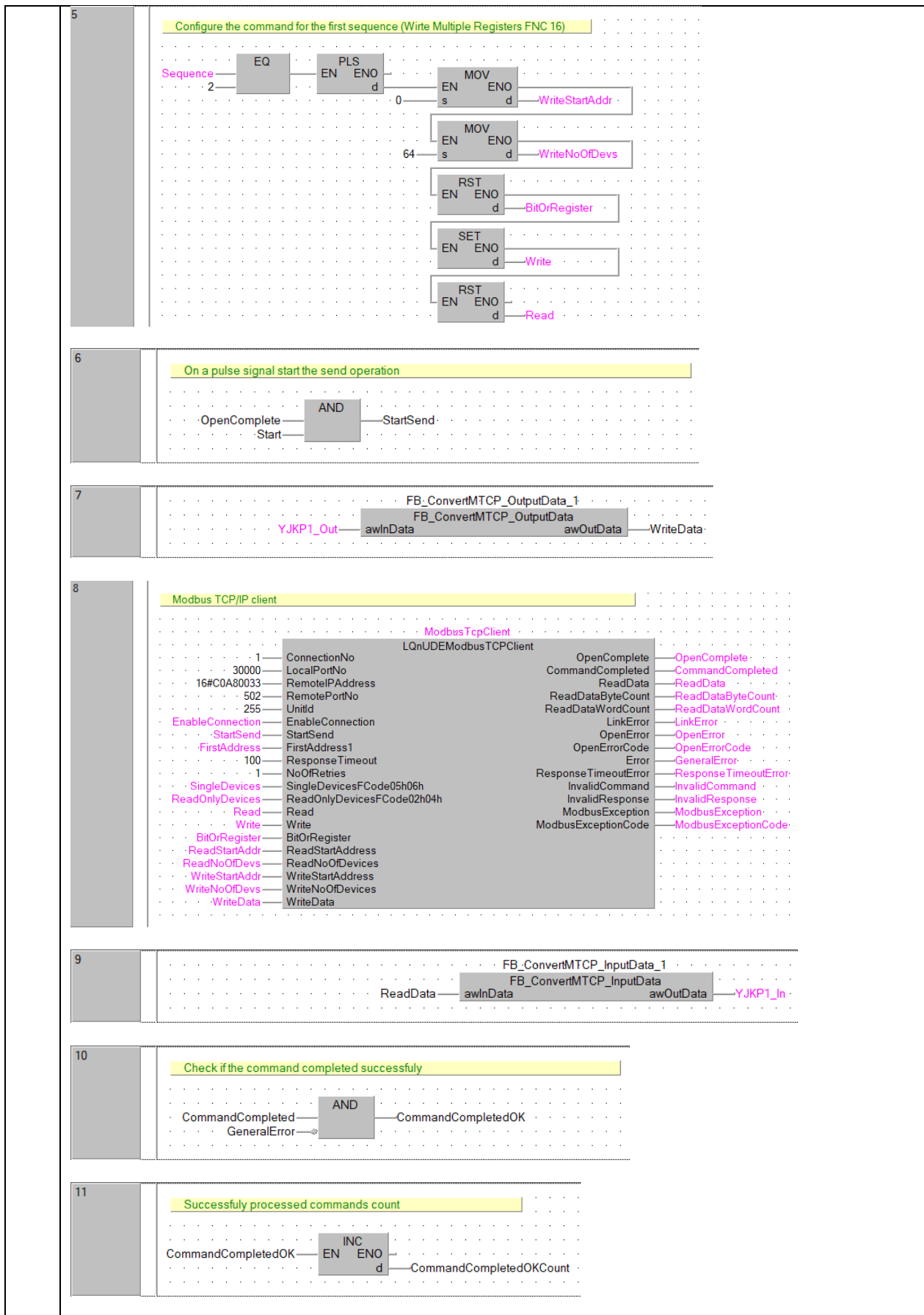
d

Sequence

SM402

1





12	Open Error handling
13	Processed commands count
14	Invalid commands count
15	Response time-out error handling
16	Invalid commands count
17	Modbus exception count and Modbus exception code logging
18	Increase sequence counter after the completion of a command

Table 3-1: Adding a Modbus TCP client to your device

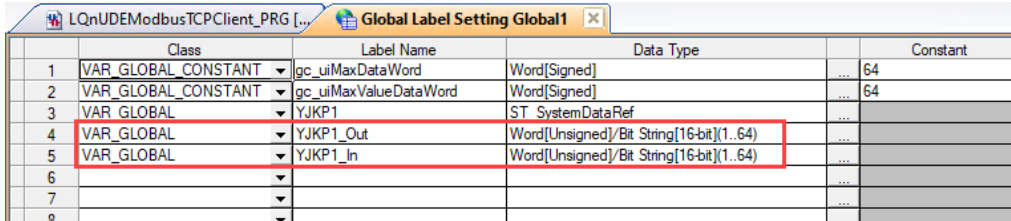
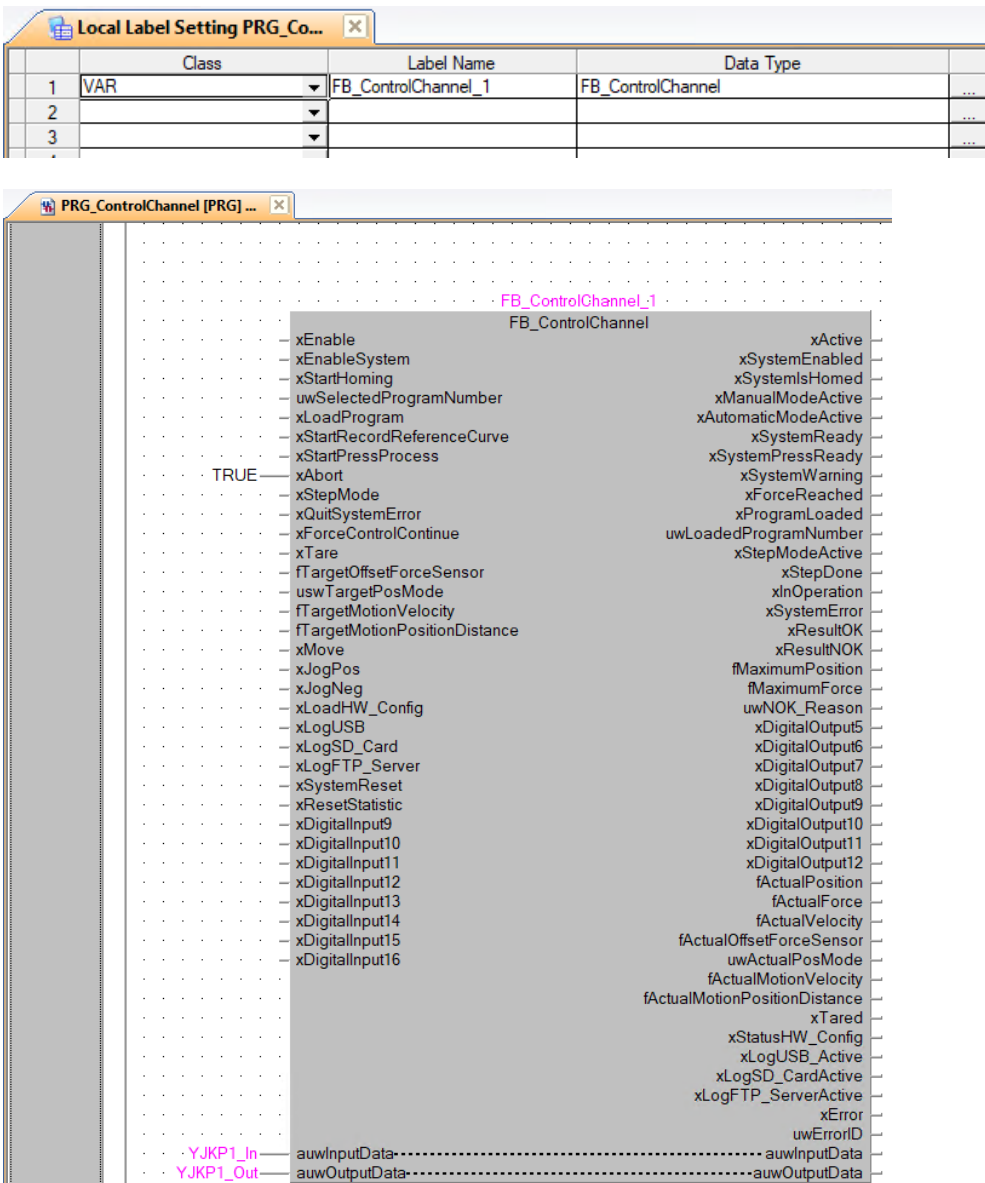
3.2 Linking the library

No.	Action									
1.	Click the button “Install” to install a library. A window “Import Library to Project” will be opened. Select the library “LQnUDEMdbusTCP_GW2_V120.sul” from the download package. Confirm the selection with the button “Open”. Confirm the installation with the button “OK” User Library Create Install Reload Save Save As Change Password Import Library to Project Library File: Browse... Library Name: It may take several minutes to complete when large amount of library data exist. OK Cancel Open Library Suchen in: Library <table><tr><th>Name</th><th>Änderungsdatum</th><th>Typ</th></tr><tr><td>LQnUDEMdbusTCP_GW2_V120.sul</td><td>14.07.2021 00:34</td><td>SUL-I</td></tr><tr><td>YJKP_Client_01_01_01.sul</td><td>14.07.2021 00:34</td><td>SUL-I</td></tr></table> Dateiname: YJKP_Client_01_01_01.sul Offnen Dateityp: User Library(*.sul) Abbrechen Import Library to Project Library File: C:\Users\DE4U3528\Desktop\kjk\GSAY-A4-F7-Z14-V1-1.1\G Browse... Library Name: YJKP_Client_01_01_01 It may take several minutes to complete when large amount of library data exist. OK Cancel MELSOFT Series GX Works2 The library has been installed. OK	Name	Änderungsdatum	Typ	LQnUDEMdbusTCP_GW2_V120.sul	14.07.2021 00:34	SUL-I	YJKP_Client_01_01_01.sul	14.07.2021 00:34	SUL-I
Name	Änderungsdatum	Typ								
LQnUDEMdbusTCP_GW2_V120.sul	14.07.2021 00:34	SUL-I								
YJKP_Client_01_01_01.sul	14.07.2021 00:34	SUL-I								



Table 3-2: Linking the library

3.3 Set up a program

No.	Action
1.	Double-Click on “Global1” in the “Project” side bar. A new window “Global Label Setting Global1” will be opened. Create two word variables with the name of your choice. In this example the names are “YJKP1_Out” and “YJKP1_In”. These two variables will be mapped in the next steps to YJKP FBs. 
2	Create a new variable of the function block “FB_ControlChannel” with the name of your choice. In this example the name is “FB_ControlChannel_1”. Insert the function block “FB_ControlChannel_1” and assign the global variables “YJKP1_In” and “YJKP1_Out”. 

3.

Create a new variable of the function block “FB_ParameterChannel” with the name of your choice. In this example the name is “FB_ ParameterChannel _1”. Insert the function block “FB_ ParameterChannel _1” and assign the global variables “YJKP1_In” and “YJKP1_Out”.

Local Label Setting PRG_Par...

	Class	Label Name	Data Type
1	VAR	FB_PrarameterChannel_1	FB_PrarameterChannel
2			
3			

PRG_ParameterChannel [PR...

FB_PrarameterChannel_1

FB_PrarameterChannel

xExecute

uwAccessType

uwAccessOption

uwObjectIndex

uwObjectSubIndex

uwObjectDataType

uwDataLengthToBeWritten

auwDataToBeWritten

auwInputData

auwOutputData

xDone

xBusy

uwAcknowledge

uwReadObjectDataType

uwReadDataLength

auwReadData

xError

uwErrorID

auwInputData

auwOutputData

Table 3-3: Set up a program