S7 Communication with S7-1200

"PUT" and "GET"

FAQ • November 2012

Siemens - Industry Automation and Drive Technologies - Service& Support - Automation Service, A - Windows Internet Explorer

http://support.automation.siemens.com

Siemens - Industry Automation and Drive Techn...

SIEMENS

→ siemens.com → Industry Automation and Drive Technolog

International | Intranet Deutsch | Français | Italiano | Español | 中文 Contact | Sitemap

Industry Automation and Drive Technologies Service & Support

Home | Product Support | Applications & Tools | Services | Information | Forum

Login | Register

[Advanced] Search

Support news →Subscribe to our newsletter now!

→ SIMOTION supports PROFINET V2.2 according to the specifications of PI (PROFIBUS & PROFINET International) from Version V4.1 SP2 onwards

→ SIMATIC HMI HMI device Mobile Panel 277F IWLAN

Self-help

Search Product Support Documents

Enter your specific product information below to quickly extract the latest related entries from our global database.

Product Name or Part Number Go

Browse Support Documents

Go to our global database and explore the product specific FAQs, manuals, downloads and approvals/certifications:

→ Product Support

For automation system interaction and connectivity questions, click below for application examples, samples, demonstration systems and more:

→ Applications & Tools

Support by experts worldwide

Technical Forum

View existing conversations and talk to other users about specific product applications and issues. Please note that you will need to Login to actively participate.

Now start your discussion with other users

Ask the Siemens expert

Send your question directly to a Siemens technical support expert via our online Support Request:

→ Support Request

Our service offer

Repairs, spare parts and consulting

Our Service products are designed to meet your support needs for the entire life cycle of your installation:

→ Spare parts and repairs → Technical Consulting

world-wide support

Choose a country

mySupport

Login

→ To get registered

My Documentation Manager

Start

Contact

Contacts worldwide

Global Business Coordination

→ Field Service

→ Spare parts

Help

Online Help

Guided Tour

Lokales Intranet | Geschützter Modus: Aktiv 100%

Siemens - Industry ... DE 15:42

Service & Support

Answers for industry.

SIEMENS

This entry is from the Siemens Industry Online Support. The general terms of use (http://www.siemens.com/terms_of_use) apply.

Clicking the link below directly displays the download page of this document.

http://support.automation.siemens.com/WW/view/en/65975617

Caution

The functions and solutions described in this article confine themselves predominantly to the realization of the automation task. Furthermore, please take into account that corresponding protective measures have to be taken in the context of Industrial Security when connecting your equipment to other parts of the plant, the enterprise network or the internet. Further information can be found in Entry ID 50203404.

http://support.automation.siemens.com/WW/view/en/50203404

Question

How do you program the instructions “GET” and “PUT” in the user program of the SIMATIC S7-1200 CPU in order to transfer more than 160 bytes of data?

Answer

The instructions and notes listed in this document provide a detailed answer to this question.

Table of Contents

1 Introduction

You must call the instructions below in the user program in order to transfer data with the SIMATIC S7-1200 CPU over a configured S7 connection:

- GET for reading data from the partner CPU
- PUT for writing data to the partner CPU

The S7 connection does not have to be configured on both sides, because the communication over S7 connections using "GET" and "PUT" is based on the server-client principle.

The "PUT" and "GET" instructions are to be found in the "Instructions" task card in the "Communication > S7 Communication" palette.

2 Description of the User Program

Only one job at a time can be triggered by the "GET" and "PUT" instructions over a configured S7 connection.

It is not possible to trigger multiple jobs simultaneously by the "GET" and "PUT" instructions over a configured S7 connection.

The "GET" and "PUT" instructions can only be called sequentially, in other words one after the other. A job is triggered only when the previous job is completed.

In this sample program, the "GET" and "PUT" instructions are called sequentially so that the SIMATIC S7-1200 CPU can receive 400 bytes of data and send 240 bytes of data over a configured S7 connection.

First you configure your SIMATIC S7-1200 CPU and the S7 connection.

Configure marker byte 10 as clock marker. The send request is triggered by this clock marker.

Save the project and compile the hardware of the SIMATIC S7-1200 CPU. Then load the hardware configuration into the SIMATIC S7-1200 CPU.

The user program consists of the components below.

Table 2-1

Block	Description
OB100	Startup OB for restart (warm restart)
OB1	The operating system of the CPU processes OB1 cyclically. The cyclic processing of OB1 begins when the startup finishes.
FB111	FB111 calls the "GET" and "PUT" instructions sequentially to receive 400 bytes of data and send 240 bytes of data over an S7 connection.
DB111	Instance data block of the FB111
DB200	The sent data is stored in the DB200 data block.
DB201	The received data is stored in the DB201 data block.

2.1 OB100

The OB100 is a restart OB and is run when the CPU is restarted (warm start). In this OB the first communication trigger is enabled with the marker M0.3.

2.2 OB1

OB1 is called cyclically. The FB111 is called in OB1 with the instance data block DB111 and the marker M0.3 as INIT_COM parameter. The marker M0.3 is reset in OB1 after the FB111 has been called.

Figure 2-1

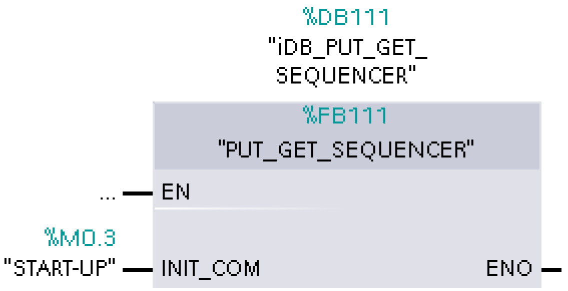
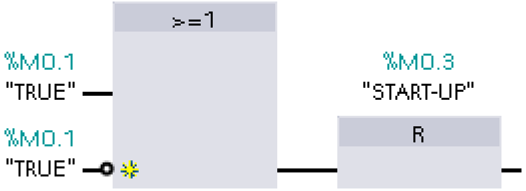


Figure 2-2



2.3 FB111

FB111 is called cyclically in OB1.

A sequencer is used to call the "GET" instructions and two "PUT" instructions sequentially that is one after the other in FB111.

Defining connection numbers

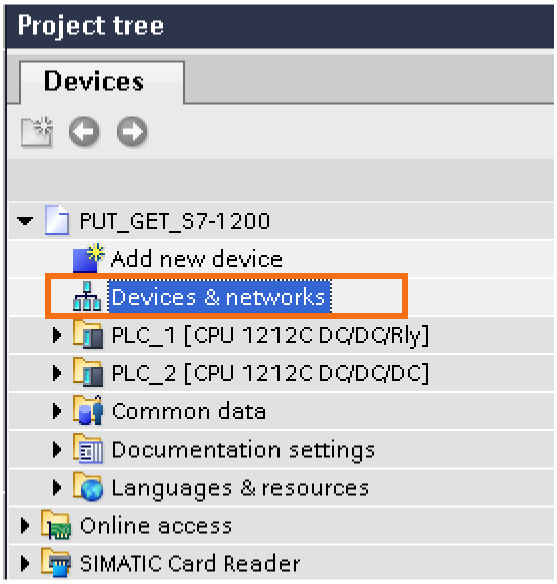
Using the "ID" input parameter you assign the same connection number to all the "GET" and "PUT" instructions. This means that all jobs are triggered one after the other over the same S7 connection.

You determine the connection number in the hardware and network editor.

Determining the connection number in the hardware and network editor

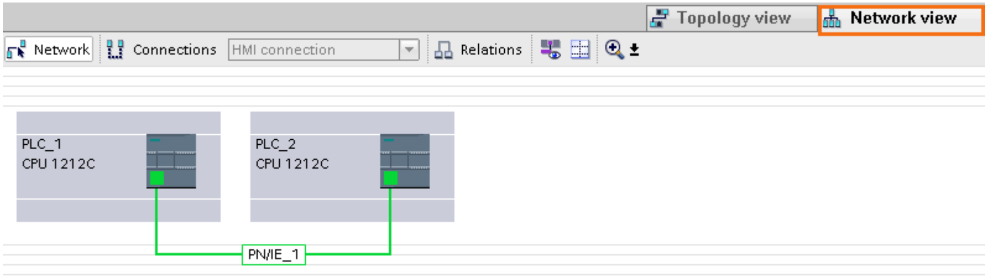
In the project tree, double-click the "Devices & networks" item to open the hardware and network editor.

Figure 2-3



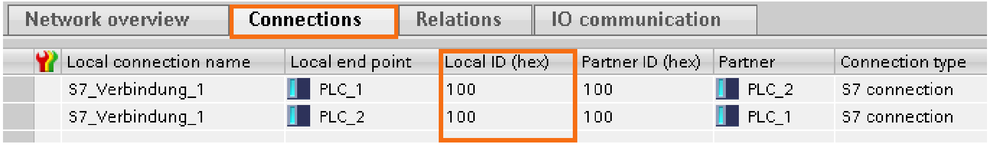
In the hardware and network editor you select the Network view.

Figure 2-4



In the Network view area you open the "Connections" table. Here you take the connection number of the configured S7 connection.

Figure 2-5



Local connection name	Local end point	Local ID (hex)	Partner ID (hex)	Partner	Connection type
S7_Verbindung_1	PLC_1	100	100	PLC_2	S7 connection
S7_Verbindung_1	PLC_2	100	100	PLC_1	S7 connection

First job trigger with the "GET" instruction

When the clock marker M10.5 has a positive edge and no other job is already running, then:

- The first "GET" instruction is activated by means of the "REQ" input parameter.
- The sequencer is initialized.

The input parameter "ADDR_1" refers to a data area in the partner CPU from which the data is read. The first 160 bytes of data are read from DB200 starting with address 0.

The input parameter "RD_1" refers to a data area in the local CPU in which the data read is stored. The data read is stored in the local CPU starting with address 0 in DB201.

The output parameters NDR, ERROR and STATUS are required for the job evaluation and are only valid in the same cycle.

When the first "GET" instruction terminates successfully or with an error, then the "REQ" input parameter of the first "GET" instruction is reset to the value "0" so that the next job can be triggered.

Figure 2-6

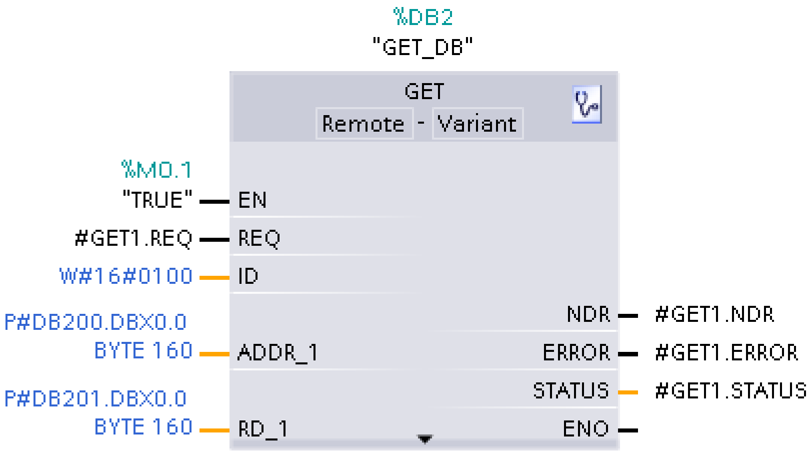
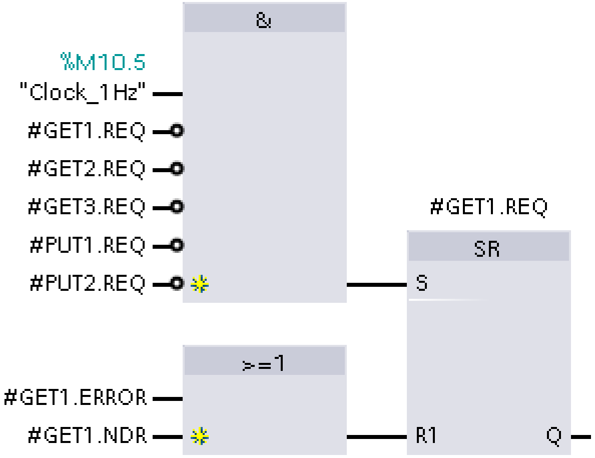
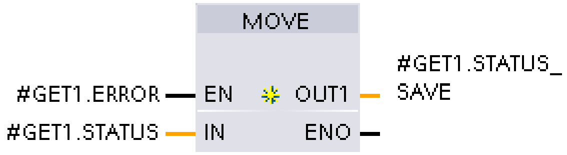


Figure 2-7



If the first "GET" instruction is completed with an error, then the value of the "STATUS" output parameter is saved for error analysis.

Figure 2-8



Second job trigger with the "GET" instruction

When the first "GET" instruction is completed successfully and no other job is running, then the second "GET" instruction is activated with the "REQ" input parameter.

The "ADDR_1" input parameter refers to a data area in the partner CPU from which the data is read. After the first 160 bytes of data have been received successfully, another 160 bytes of data are read out of DB200 as from address 160.

The "RD_1" input parameter refers to a data area in the local CPU in which the data read is stored. The data read is stored in the local CPU starting with address 160 in DB201.

The output parameters NDR, ERROR and STATUS are required for the job evaluation and are only valid in the same cycle.

When the second "GET" instruction terminates successfully or with an error, then the "REQ" input parameter of the second "GET" instruction is reset to the value "0" so that the next job can be triggered.

Figure 2-9

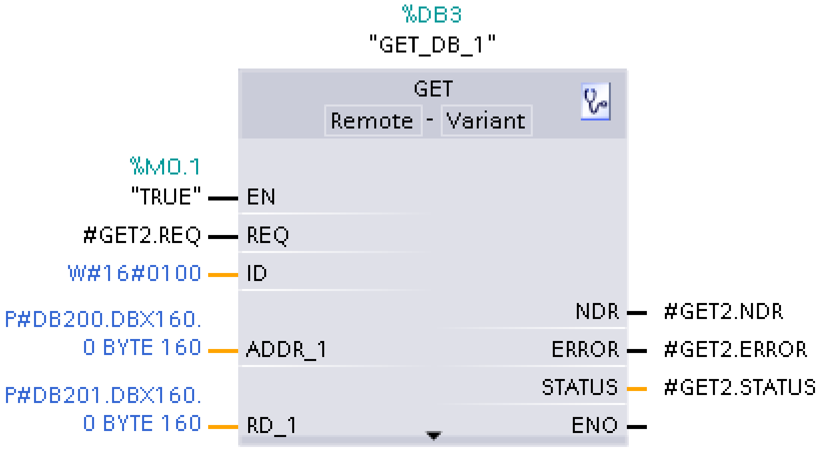
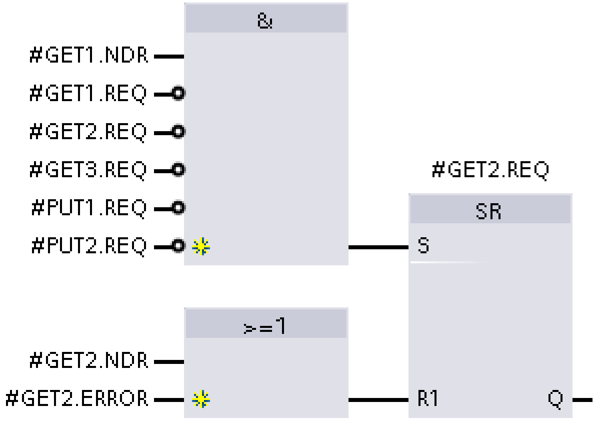
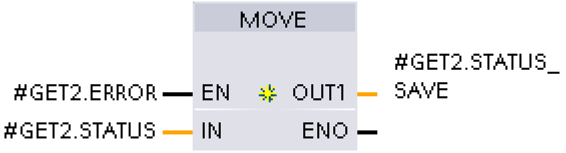


Figure 2-10



If the second "GET" instruction is completed with an error, then the value of the "STATUS" output parameter is saved for error analysis.

Figure 2-11



Third job trigger with the "GET" instruction

When the second "GET" instruction is completed successfully and no other job is running, then the third "GET" instruction is activated with the "REQ" input parameter.

The "ADDR_1" input parameter refers to a data area in the partner CPU from which the data is read. After 320 bytes of data have been received successfully, another 80 bytes of data are read out of DB200 as from address 320.

The "RD_1" input parameter refers to a data area in the local CPU in which the data read is stored. The data read is stored in the local CPU starting with address 320 in DB201.

The output parameters NDR, ERROR and STATUS are required for the job evaluation and are only valid in the same cycle.

When the third "GET" instruction terminates successfully or with an error, then the "REQ" input parameter of the third "GET" instruction is reset to the value "0" so that the next job can be triggered.

Figure 2-12

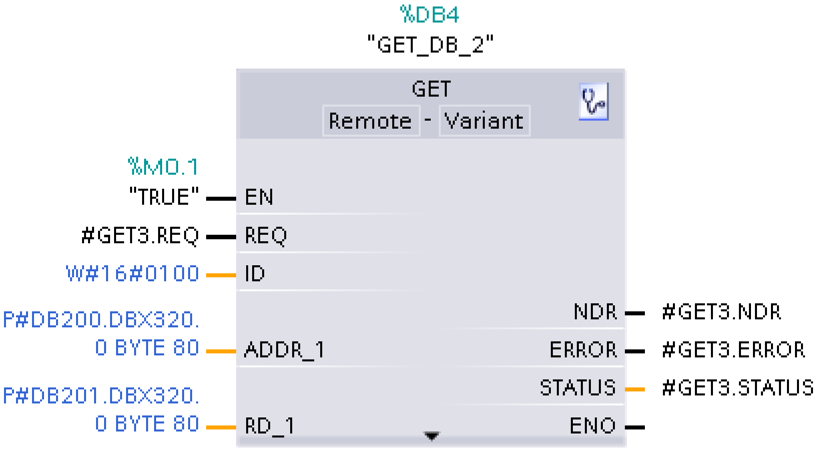
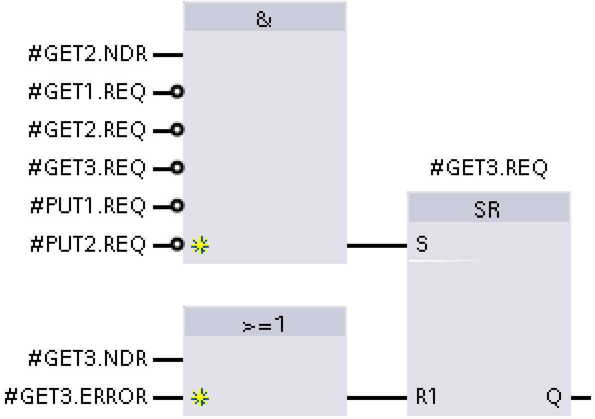
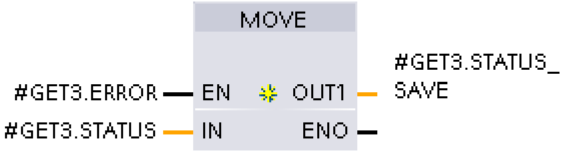


Figure 2-13



If the third "GET" instruction is completed with an error, then the value of the "STATUS" output parameter is saved for error analysis.

Figure 2-14



First job trigger with the "PUT" instruction

When the third "GET" instruction has terminated successfully and no other job is running, then:

- A total of 400 bytes of data have been read and stored as received data in DB201 starting with address 0.
- Another job for sending data is triggered by the "PUT" instruction. The first "PUT" instruction is activated by the "REQ" input parameter.

The "ADDR_1" input parameter refers to a data area in the partner CPU to which the data is written. The data written is stored in the partner CPU starting with address 0 in DB201.

The "SD_1" input parameter refers to a data area in the local CPU which contains the data to be sent. 160 bytes of data are sent from DB200 starting with address 0.

The output parameters DONE, ERROR and STATUS are required for the job evaluation and are only valid in the same cycle.

When the first "PUT" instruction terminates successfully or with an error, then the "REQ" input parameter of the first "PUT" instruction is reset to the value "0" so that the next job can be triggered.

Figure 2-15

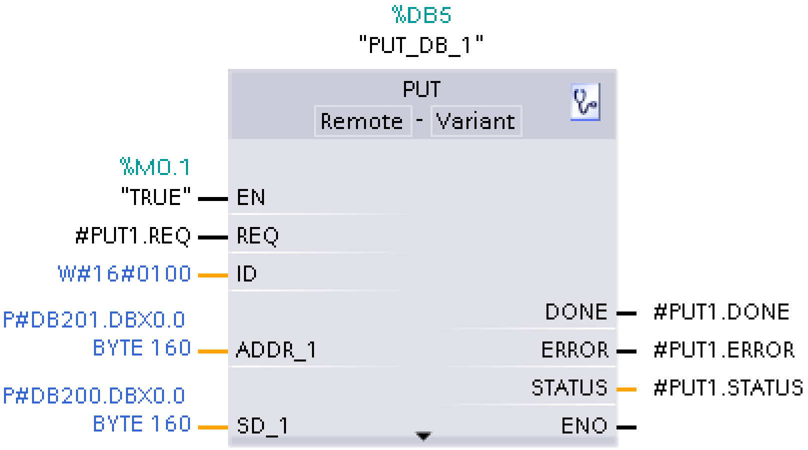
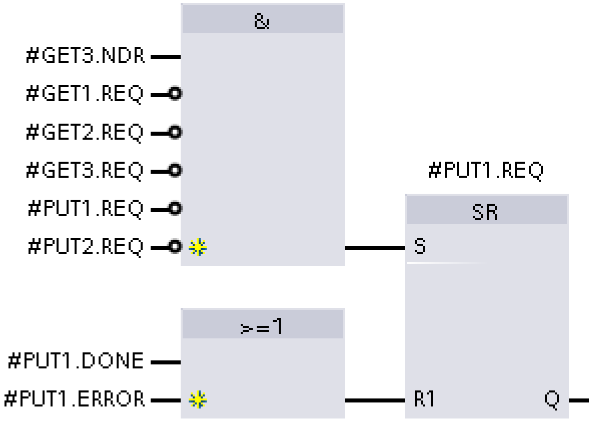
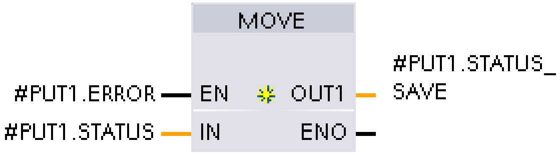


Figure 2-16



If the first "PUT" instruction is completed with an error, then the value of the "STATUS" output parameter is saved for error analysis.

Figure 2-17



Second job trigger with the "PUT" instruction

When the first "PUT" has terminated successfully and no other job is running, then another job for sending data is triggered by the "PUT" instruction. The second "PUT" instruction is activated by the "REQ" input parameter.

The "ADDR_1" input parameter refers to a data area in the partner CPU to which the data is written. The data written is stored in the partner CPU starting with address 160 in DB201.

The "SD_1" input parameter refers to a data area in the local CPU which contains the data to be sent. After the first 160 bytes of data have been transferred successfully, another 80 bytes of data are sent from DB200 starting with address 160.

The output parameters DONE, ERROR and STATUS are required for the job evaluation and are only valid in the same cycle.

When the second "PUT" instruction terminates successfully or with an error, then the "REQ" input parameter of the first instruction is reset to the value "0" so that once again the first job can be triggered with the "GET" instruction in the next cycle of FB111. The "GET" and "PUT" instructions are once again called sequentially for receiving and sending new data.

Figure 2-18

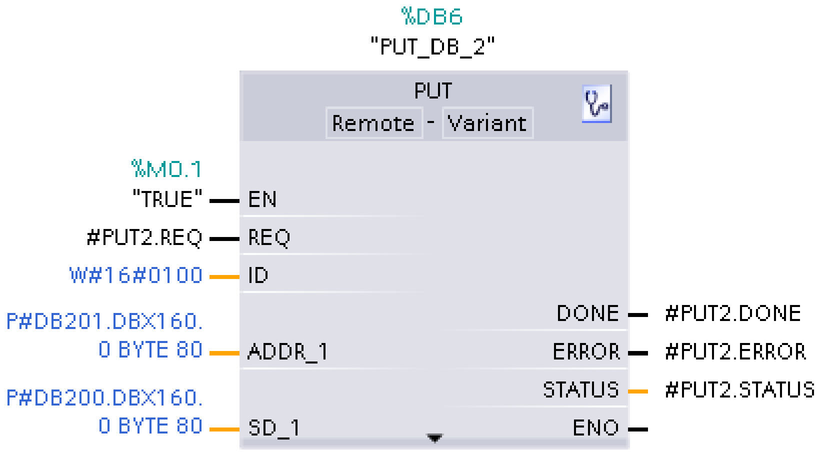
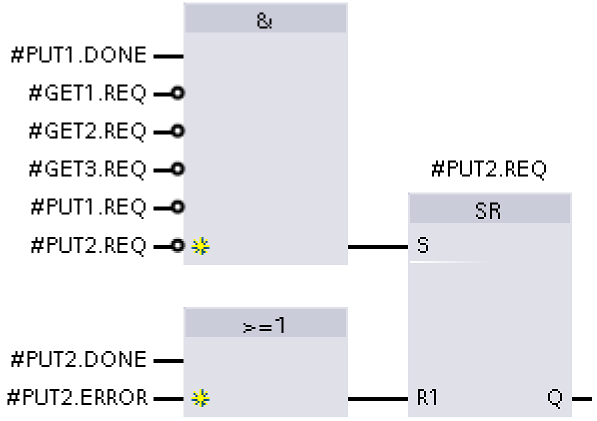
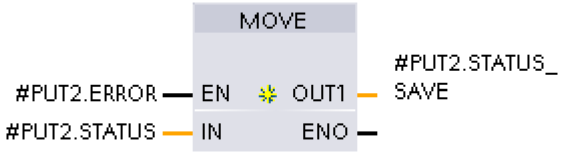


Figure 2-19



If the second "PUT" instruction is completed with an error, then the value of the "STATUS" output parameter is saved for error analysis.

Figure 2-20


MOVE
#PUT2.ERROR
EN
OUT1
#PUT2.STATUS_
SAVE
#PUT2.STATUS
IN
ENO