SIEMENS

SIMATIC 505-CP1434-TCP

Industrial Ethernet TCP/IP Module

User Manual

Manual Assembly Number: 2810566-0001
First Edition

Danger

indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

Warning

indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

Caution

used with the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

Caution

used without the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in property damage.

Notice

NOTICE used without the safety alert symbol indicates a potential situation which, if not avoided, may result in an undesirable result or state.

Qualified Personnel

Only **qualified personnel** should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground, and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Correct Usage

Note the following:

Warning

This device and its components may only be used for the applications described in the catalog or the technical descriptions, and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens.

This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

Trademarks

Siemens[r], SIMATIC[r], SINEC[r], and S7[E] are trademarks of Siemens AG. 505 SoftShop[r] is a registered trademark and Series 505[E] and WinCC[E] are trademarks of Siemens Energy & Automation, Inc. Microsoft[r] and Windows[r] are registered trademarks of Microsoft Corportation. CSA[r] is a registered trademark of Canadian Standards Association. Some of other designations used in these documents are also registered trademarks; the owner's rights may be violated if they are used by third parties for their own purposes.

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

MANUAL PUBLICATION HISTORY

SIMATIC 505-CP1434-TCP Industrial Ethernet TCP/IP Module User Manual
Refer to this history in all correspondence and/or discussion about this manual.

Event	Date	Description
Original Issue	12/01	Original Issue (2810566-0001)

LIST OF EFFECTIVE PAGES

Pages	Description	Pages	Description
Cover/Copyright	Original		
History/Effective Pages	Original		
iii — xiv	Original		
1-1 — 1-10	Original		
2-1 — 2-17	Original		
3-1 — 3-32	Original		
4-1 — 4-46	Original		
5-1 — 5-17	Original		
6-1 — 6-40	Original		
A-1 — A-1	Original		
B-1 — B-14	Original		
C-1 — C-21	Original		
D-1 — D-22	Original		
Index-1 — Index-6	Original		
Registration	Original		

Contents

Chapter 3 Using the Configurator

Chapter 4 Configuring Jobs

Chapter 5 Using the Optional OPC Server for TCP/IP

Chapter 6 Examples: Send and Receive Jobs

Appendix D E-mail Jobs

List of Figures

List of Tables

Preface

About This Manual

This manual describes the following products:

- The SIMATIC 505-CP1434-TCP Industrial Ethernet TCP/IP module (PPX:505-CP1434-TCP) for the SIMATIC 505 programmable controller system
- The SIMATIC 505-CP1434-TCP Module Configurator software package, which is used to create configuration files for the SIMATIC 505 Industrial Ethernet TCP/IP module
- The optional SIMATIC 505 OPC Server for TCP/IP software package (PPX:505-6701 single license; PPX:505-6702 site license), which can be used with OPC clients such as SIMATIC WinCC

The SIMATIC 505 Industrial Ethernet TCP/IP module can communicate over the TCP/IP Ethernet network with the following types of device:

- Other SIMATIC 505 TCP/IP modules
- SIMATIC 505-CP2572 modules
- SIMATIC S7 controllers with communications processor modules such as the CP 343-1 Multi-protocol, CP 343-1 IT, CP 443-1 Multi-protocol, or CP 443-1 IT
- Personal computers (PCs) running the optional SIMATIC 505 OPC Server for TCP/IP
- Other 3rd-party units supporting TCP/IP protocol

The following topics are covered in this manual:

- Module features
- Hardware installation
- TCP/IP network communications overview
- OPC server with Report-by-Exception (RBE)
- Software installation and configuration
- Programming examples
- Troubleshooting information

Related Manuals

You can find additional information in the following manuals:

- *SIMATIC 505 Programming Reference Manual* (PPX:505-8104-x)
- *SIMATIC 505 SoftShop for Windows User Manual* (PPX:SS505-8101-x)
- The system manual(s) for your controller(s)

All of the SIMATIC 505 manuals are available on the SIMATIC 505 Electronic Manuals compact disk, PPX:505-CDMANUAL-3 or greater.

Conventions

The following conventions are used in this manual:

- Click means to place the mouse cursor on an object and quickly press and release the mouse button. In most cases only the left mouse button is used.
- Double click means to place the mouse cursor on an object and quickly press and release the left mouse button twice.
- Select means to click on an item so that it is highlighted.
- Text that you should type is shown in plain type.
- Commands that you select from the menu bar are shown in **bold type.**
- Menu items may also be selected by pressing the **Alt** key and the underlined character of the menu item.
- Pressing the **Esc** key is the same as selecting Cancel in a window.

Agency Approvals

Agency approvals for the SIMATIC 505 Industrial Ethernet TCP/IP module are the following:

- Underwriters Laboratory, Inc., UL Listed (Industrial Control Equipment)
- CSA Certified (Process Control Equipment)
- FM (Class I, Div. 2, Hazardous Locations)
- CE Marking (Low Voltage Directive 73/23/EEC and Electro-Magnetic Compatibility Directive 89/336/EEC)

Technical Assistance

For technical assistance, contact your Siemens distributor or sales office. If you need assistance in contacting your U.S. distributor or sales office, call 800-964-4114.

Chapter 1

Product Overview

1.1 Features of the Module

The SIMATIC 505-CP1434-TCP Industrial Ethernet TCP/IP module provides TCP/IP communications between supported Series 505 and S7 programmable logic controllers (PLCs), as well as the optional SIMATIC 505 OPC Server for TCP/IP (described in Section 1.4), over an Ethernet network. The module offers improved update rates through the exclusive SIMATIC 505 report-by-exception (RBE) feature.

System Compatibility

The SIMATIC 505 TCP/IP module is a single-wide Series 505 module. It installs in any Series 505 local or remote base that is hosted by a 545 or 555 PLC (except for a remote base that is connected on PROFIBUS by means of the 505-6870 PROFIBUS-DP remote base controller). In order for the module to use RBE, it must be connected to a 545 or 555 Release 3.0 or higher.

The SIMATIC 505 TCP/IP module can communicate over the TCP/IP Ethernet network with the following types of device:

- Other SIMATIC 505 TCP/IP modules
- Personal computers (PCs) running the optional SIMATIC 505 OPC Server for TCP/IP
- SIMATIC 505-CP2572 modules
- Other third-party units supporting TCP/IP protocol
- SIMATIC S7 controllers with communications processor modules such as the CP 343-1 Multi-Protocol, CP 343-1 IT, CP 443-1 Multi-Protocol, or CP 443-1 IT

NOTE: The SIMATIC 505-CP2572, SIMATIC S7 CP 343, and SIMATIC S7 CP 443 modules close inactive server sockets after a short time if their TCP Keep Alive Interval is set to the default value. To avoid breaks in the connection between a SIMATIC 505 TCP/IP module and these modules, extend or disable their TCP Keep Alive Interval. (For the 505-CP2572, set the TCP Stale Socket equal to or greater than 65,001.)

Module Configurator

The module comes with the SIMATIC 505-CP1434-TCP Module Configurator software package, which offers the following advantages:

- The Configurator is menu driven and easy to use. You do not have to write PLC code to configure module parameters.
- Because the SIMATIC 505 TCP/IP modules have the Configurator and use Job Start bits, communication between the 505 TCP/IP modules is faster than communication between modules that have to fetch data from the PLC.
- Because the configuration does not reside in the PLC, you cannot inadvertently affect the configuration by changes to the PLC program.

Faceplate Features

The faceplate of the module includes the following features:

- MOD GOOD and MOD READY status LEDs
- A recessed RESET button
- An RS-232/423 serial port for loading communication parameters from the Configurator program on your personal computer
- An Ethernet Bus Interface port that links up to the TCP/IP network through a transceiver unit. This 15-pin port conforms to the ANSI/IEEE Std. 802.3 Local Area Network standards (International Standard ISO/IEC 8802-3).

Figure 1-1 shows the faceplate of the module.

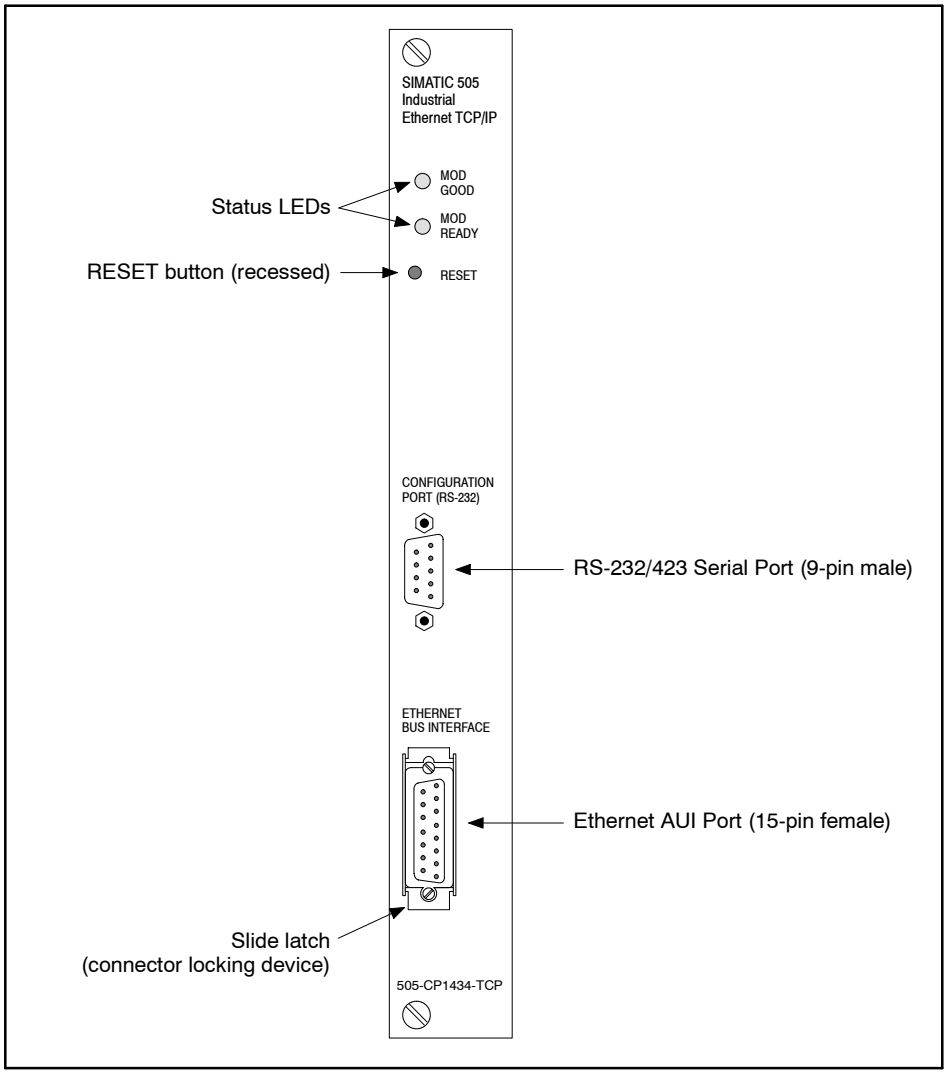


Figure 1-1 SIMATIC 505 TCP/IP Module Faceplate

Memory Storage

The SIMATIC 505 TCP/IP module has non-volatile flash memory that stores your configuration parameters. You create configuration parameters by running the Configurator on a computer with Windows 95 or higher. With a cable connecting the computer to the RS-232/423 serial port on the module, you transfer the configuration file into flash memory on the module (see Figure 1-2). You do not need to remove the module from the base or handle integrated circuits on the board to reprogram the flash memory.

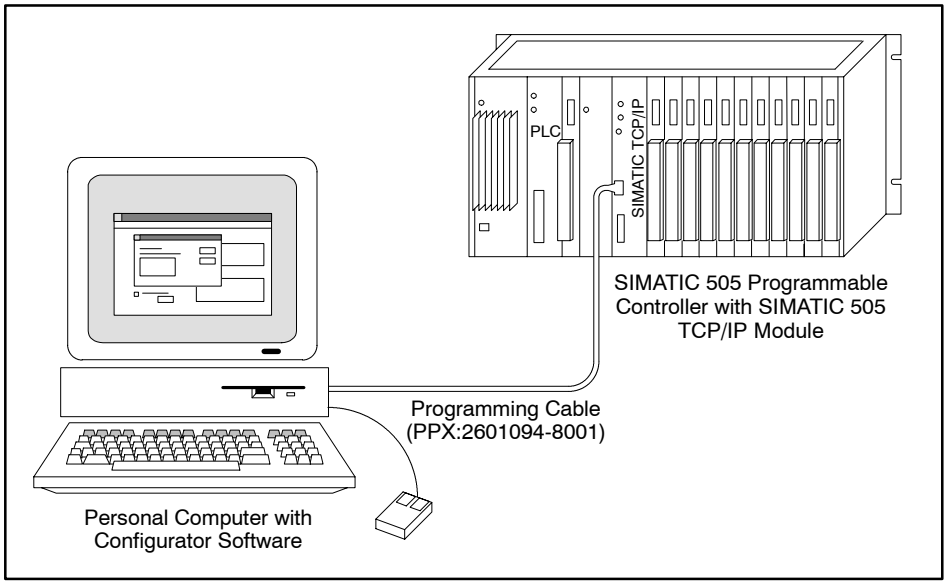


Figure 1-2 Personal Computer Connected to SIMATIC 505 TCP/IP Module

DC-to-DC Converter

The SIMATIC 505 TCP/IP module also includes a DC-to-DC converter that, when enabled, supplies +12 V to the Ethernet communication port that powers the Media Attachment Unit (MAU) transceiver.

1.2 Functional Description of the Module

Communication Functions

The SIMATIC 505 TCP/IP module uses the TCP/IP protocol suite to provide communications between hosts on a TCP/IP programmable controller network.

The following jobs are possible on a SIMATIC 505 TCP/IP module:

- Send/Receive (work as a pair)
- Write Active/Write Passive (work as a pair)
- Read Active/Read Passive (work as a pair)
- TCP Server
- UDP Server
- Write Remote (works with a TCP Server or UDP Server job)
- Read Remote (works with a TCP Server or UDP Server job)
- Memory Exchange (works with a TCP Server or UDP Server job)
- Broadcast Send/Broadcast Receive (one Broadcast Send works with multiple Broadcast Receives)
- E-Mail

For detailed information about these jobs, see Chapter 4.

Number of Jobs Permitted

You can configure a maximum of 64 jobs per module. Section 4.3 describes the guidelines. The individual maximums for the different job types are described in the sections on the different jobs in Chapter 4.

Job Start Bits

The module offers 48 Job Start bits, which means that you can configure a maximum of 48 jobs that use Job Start bits per module. (Some jobs do not require a Job Start bit.)

Compatible Devices

Figure 1-3 illustrates the types of devices that can communicate with the SIMATIC 505 TCP/IP module.

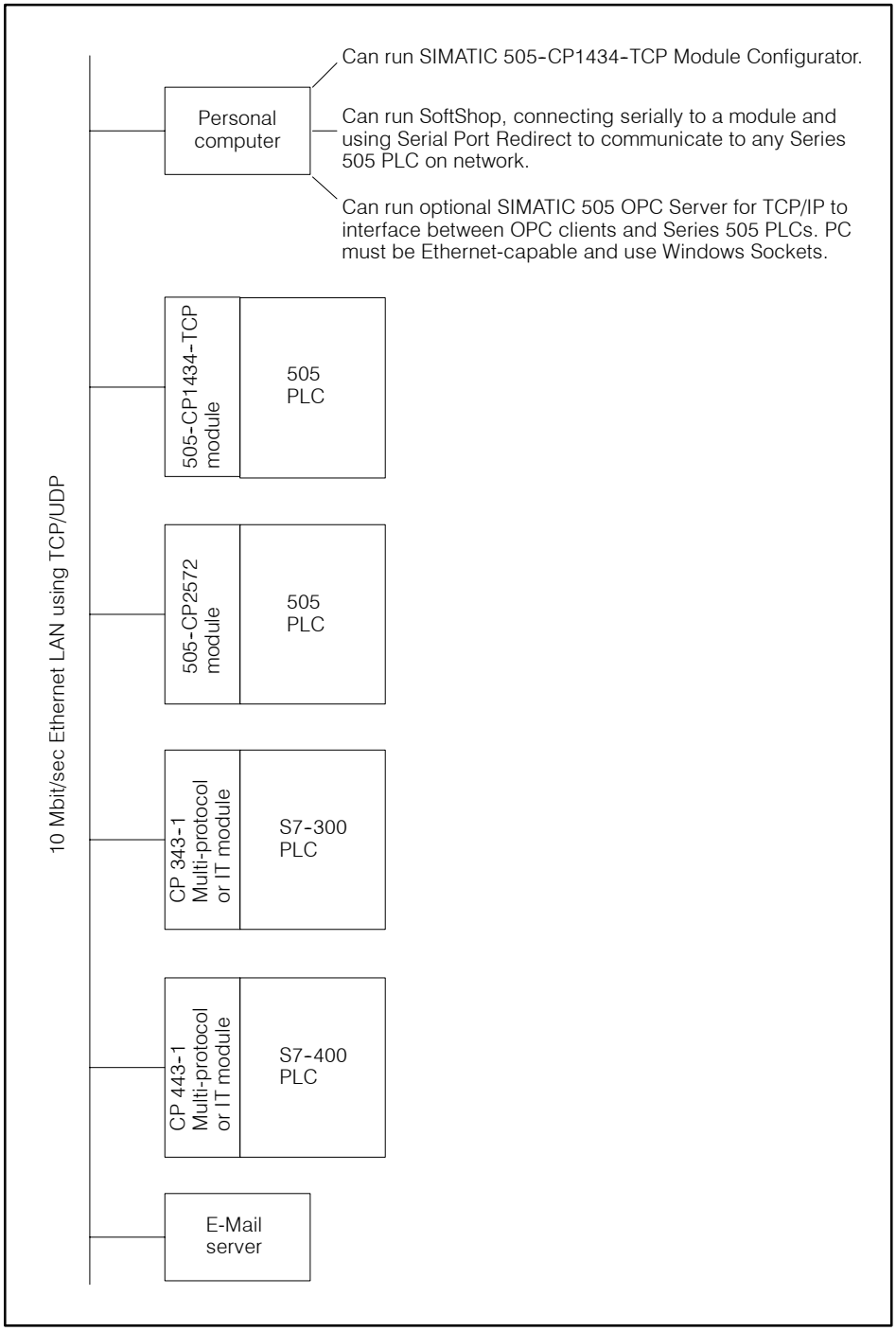


Figure 1-3 Compatible Devices

1.3 SIMATIC 505-CP1434-TCP Module Configurator

Ordering the Configurator

The SIMATIC 505-CP1434-TCP Module Configurator allows you to configure parameters and jobs for SIMATIC 505 TCP/IP modules.

The Configurator operates on a personal computer (abbreviated as PC) running Windows 95 or higher.

To obtain the Configurator, go to the Siemens Energy and Automation web site, or install it from the SIMATIC 505 Electronic Manuals compact disk.

Downloading from the Web Site

To access the Siemens web site where the Configurator is available for download, use one of these addresses:

http://www.sea.siemens.com/automat/product/plc/505/au505dl.html
(the page for the SIMATIC 505 PLC family)

http://www.sea.siemens.com
(the page for Siemens Energy & Automation, U.S.A.)

These web addresses are subject to change over time. If you have difficulty accessing the Siemens web site, contact your Siemens distributor or the Siemens technical support hotline.

Ordering the Compact Disk

To order the SIMATIC 505 Electronic Manuals compact disk, which includes the Configurator Software, ask for PPX:505-CDMANUAL-x, where “x” is 3 or higher.

1.4 SIMATIC 505 OPC Server for TCP/IP

The SIMATIC 505 OPC Server for TCP/IP is an optional software application that provides an interface between Series 505 PLCs (by means of the SIMATIC 505 TCP/IP module) and OPC clients such as SIMATIC WinCC (version 5.0 SP2 or higher).

The SIMATIC 505 OPC server lets you configure variables so that you can use an OPC client product to monitor variables that originate from a Series 505 PLC. Then the client can access that data from the OPC server. When PLC data is cached in the SIMATIC 505 OPC server, the OPC client does not have to query the PLC and can obtain it more quickly. Furthermore, the SIMATIC 505 OPC server supports report-by-exception (RBE) variables, which offer the fastest possible update, for the SIMATIC 505 TCP/IP module.

Ordering the Server Application

To order the optional SIMATIC 505 OPC server, contact your distributor and ask for one of the following order numbers:

Single User License
PPX:505-6701

Site License
PPX:505-6702

1.5 Contacting Siemens

If you need assistance in contacting your Siemens distributor, or need to contact the Siemens technical support hotline, use the phone numbers listed below.

Siemens Distribution/Sales
800-964-4114

Technical Support
800-333-7421
423-262-2522

Chapter 2

Installing the Module

2.1 Overview

Flow of Tasks

Follow the flow of tasks shown in Figure 2-1 when installing this module.

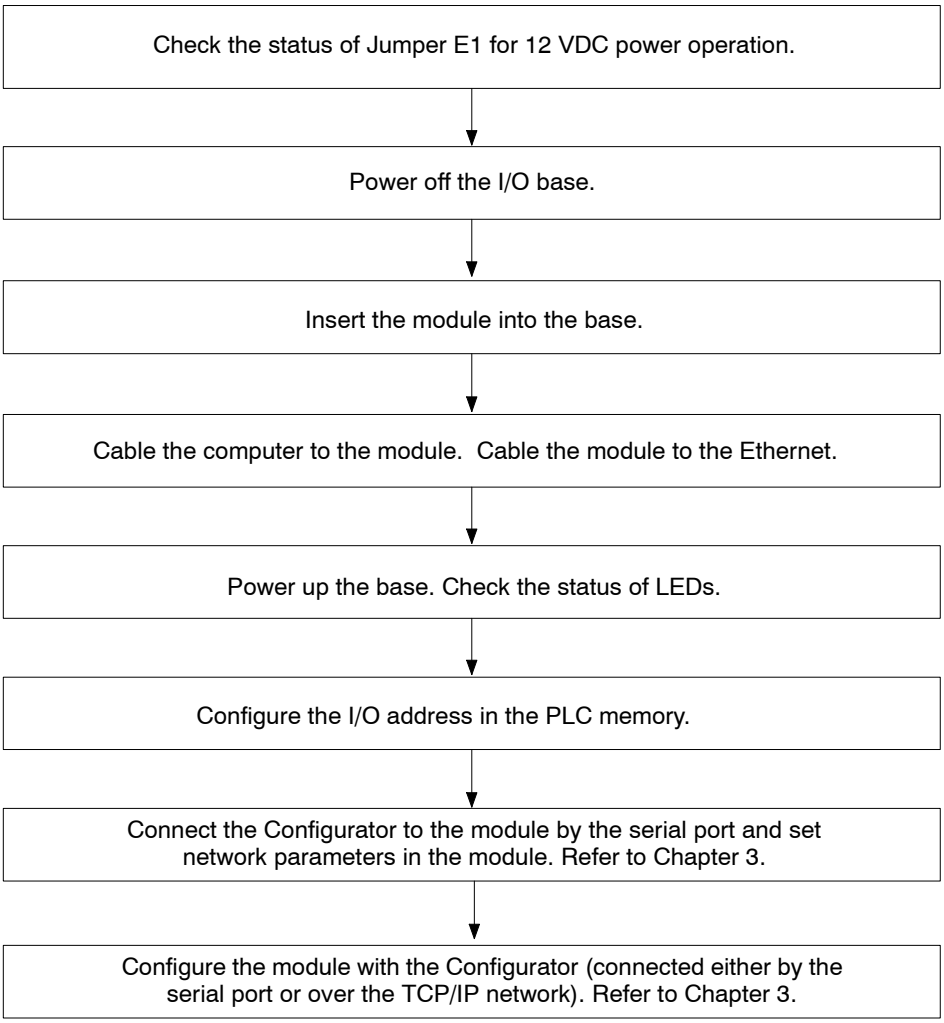


Figure 2-1 Flowchart of Installation

Guidelines for Handling the Module

Integrated circuits are susceptible to damage by the discharge of static electricity. Follow the suggestions listed below to reduce the probability of damage when you handle the SIMATIC 505 TCP/IP module:

- Move the SIMATIC 505 TCP/IP module in an anti-static container or in anti-static material.
- Ensure that the work area has a conductive pad with a lead connecting it to a common ground. Ensure that you and the module are at the same ground potential.
- Ground yourself by touching the conductive pad and/or by wearing a grounded wrist strap.

Visual Inspection

If the SIMATIC 505 TCP/IP module is visibly damaged, contact your vendor for replacement.

2.2 Setting the Jumper for the DC-to-DC Converter

The DC-to-DC converter, when enabled, supplies +12 V to the Ethernet port (which can power an Attachment Unit Interface (AUI)-to-Twisted-Pair converter or a Media Attachment Unit (MAU) transceiver).

A jumper labeled E1 is located near the Series 505 base connector. The jumper is used to enable or disable the DC-to-DC converter. Since the module is shipped with the jumper installed (see Figure 2-2), the DC-to-DC converter is enabled by default.

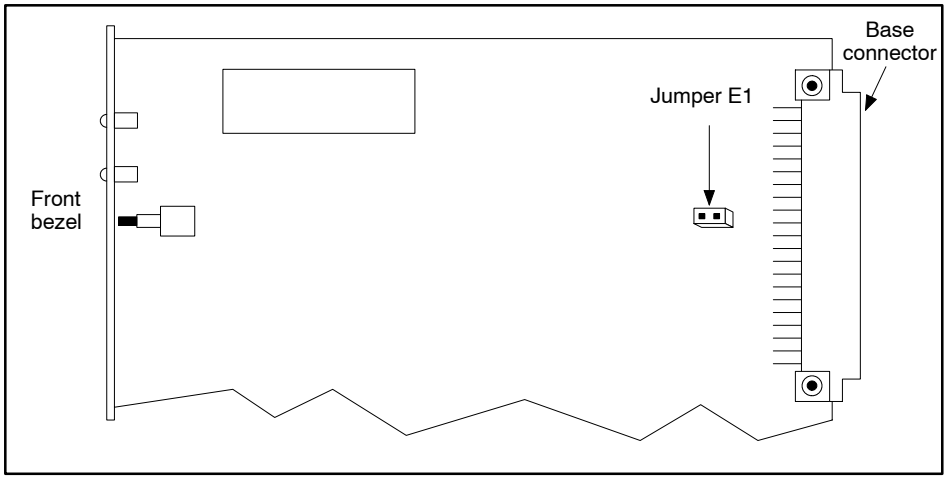


Figure 2-2 Location of Jumper E1 for DC-to-DC Converter

Calculating Power Consumption

SIMATIC 505 bases have a maximum power supply load (typically 55 watts) that cannot be exceeded by the consumption of the specific CPU or Remote Base Controller and modules that are installed in the base. The *SIMATIC 545/555/575 System Manual* (PPX:505–8201–3) provides the information to calculate the total power consumption in a base.

The power consumption of the SIMATIC 505 TCP/IP module can be variable due to the load of the DC-to-DC converter. The maximum DC power consumption is 9 watts with the DC-to-DC converter disabled. When the DC-to-DC converter is enabled the DC power consumption can be determined if the DC-to-DC converter current load is known. The following formula applies:

Max DC Power (watts) = 9 + 20 x (DC–to–DC Converter Load (amps))

For example, at the maximum DC-to-DC Converter load of 0.5 amps:

Max DC Power = (9 + (20x0.5)) watts = 19 watts

For a DC-to-DC Converter load of 0.1 amps, the Max DC power is 11 watts.

To maximize the number of SIMATIC 505 TCP/IP modules in a single base you can disable the DC-to-DC converter on each module installed in the base. With the DC-to-DC converter disabled, AUI port communications must be powered externally.

Disabling the Converter

To disable the DC-to-DC converter, remove Jumper E1 (see Figure 2-3). You can store the jumper on the module itself by plugging it back in any direction on either one of the E1 male header pins.

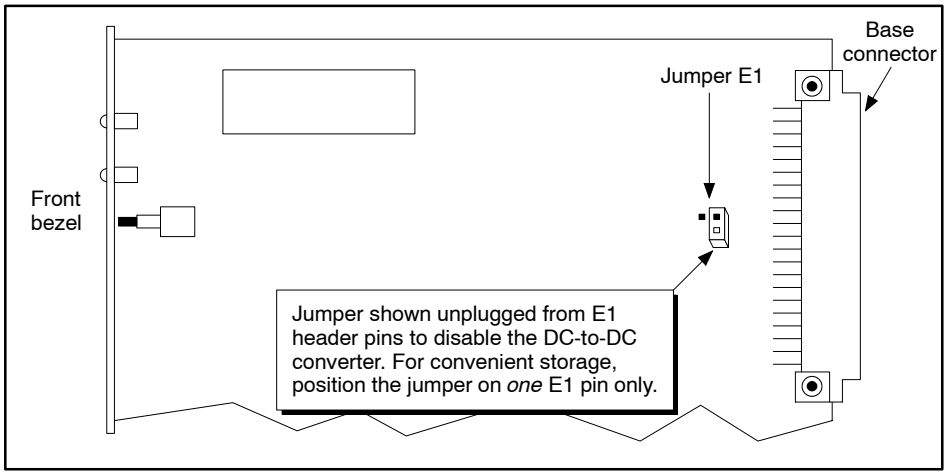


Figure 2-3 Disabling the DC-to-DC Converter

The primary advantage of disabling the converter is that the module power requirement from the base is lower than when the converter is enabled, allowing more SIMATIC 505 TCP/IP modules to be installed in a single I/O base. (See Appendix A for power consumption specifications.)

NOTE: If you disable the converter, you must use a transceiver with an external power supply. This is accomplished by a standard fan-out unit with external power supply, such as the Electrical Switch Module (ESM) manufactured by Siemens, order number 6GK1105-3AA00.

2.3 Inserting the Module into the Base

Remove power from the base before inserting or removing modules.

⚠ **WARNING**

Energized parts could cause electrical shock.

Electrical shock could result in death or serious injury to personnel and/or damage to equipment.

Remove power from the I/O base before inserting or removing modules.

Ensure that the area is non-hazardous before inserting or removing modules.

⚠ **WARNING**

Inserting or removing a module with power on to the I/O base in a National Electric Code Class I, Div. 2, hazardous location could cause an electrical shock and/or an explosion.

Electrical shock and/or explosions could result in death or serious injury to personnel and/or damage to equipment.

Remove power from the I/O base and ensure that the area is non-hazardous before inserting or removing modules.

Inserting the Module

The SIMATIC 505 TCP/IP module is a single-wide module. After powering off the base, insert the module into any I/O slot in any 545/555 local or remote I/O base. Insert the module as shown in Figure 2-4. Note the minimum torque required for the bezel screws to provide specified electromagnetic shielding.

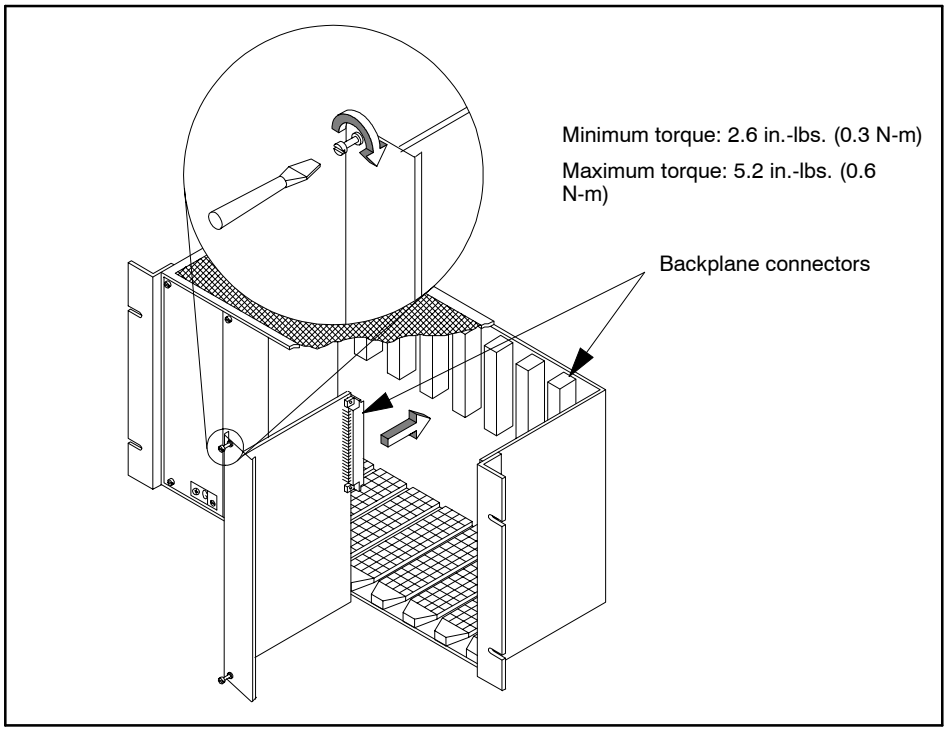


Figure 2-4 Inserting the Module into the I/O Base

2.4 Connecting the Module to Your Computer

Cable Requirements

Before you can program your SIMATIC 505 TCP/IP module with configuration files, you need to connect your module to the PC that is running the Configurator, using a standard 9-pin RS-232/423 serial cable that conforms to the minimum pinouts shown in Figure 2-5.

You can order a standard cable that conforms to the minimum requirements (SIMATIC 545 Programming Cable, PPX:2601094–8001) from your Siemens distributor.

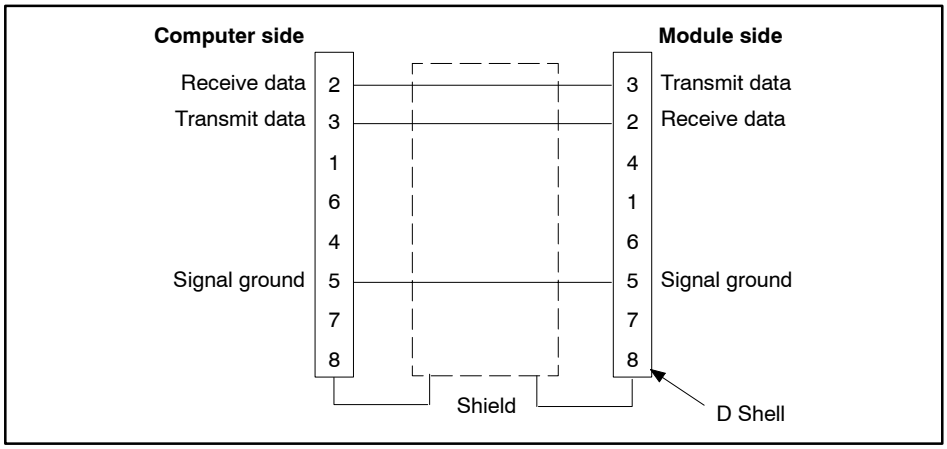


Figure 2-5 RS-232 Serial Port Cable Pinouts

Connecting the Serial Cable

Connect one end of the cable to the appropriate 9-pin RS-232/423 serial port on your computer and the other end to the 9-pin configuration port (RS–232) on the front of the module, as shown in Figure 2-6. If your computer does not have a 9-pin port, use a 25-to-9-pin converter.

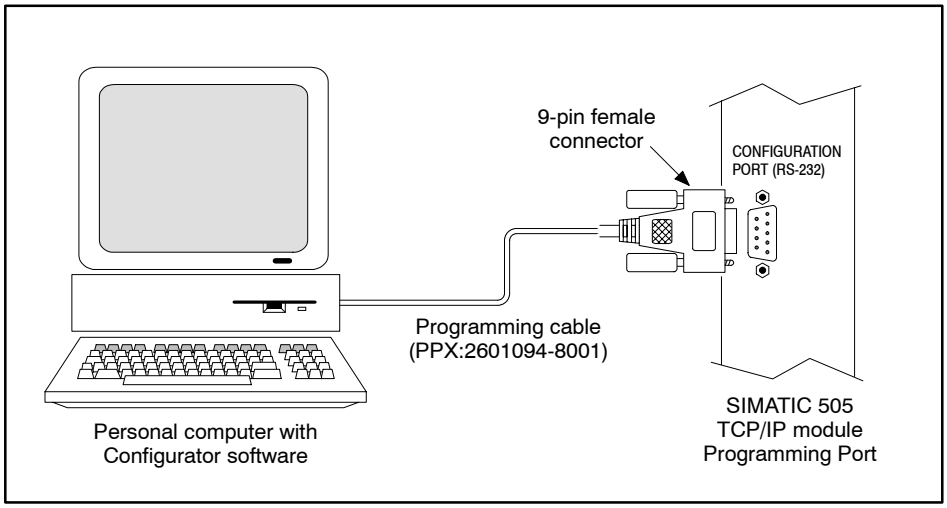


Figure 2-6 Connecting Your Computer to the Module

Transferring Configurations over the Network

After you have transferred the necessary configuration parameters (described in Section 3.10), you can change the configuration transfer mode from Serial to Network and perform subsequent configuration work over the network.

NOTE: To use the SIMATIC 505–CP1434–TCP Module Configurator to communicate with a SIMATIC 505 TCP/IP module over the TCP/IP network, the PC on which the Configurator is running must be Ethernet-capable. If your PC is not Ethernet-capable, you can install a standard Ethernet card such as the CP1613 card, available from Siemens.

2.5 Connecting the Module to the Ethernet

You can connect the SIMATIC 505 TCP/IP module to 10Base5 (thicknet) or 10Base-T (Category 5 cable) installations. In either case, the pinouts for the Ethernet Bus Interface port on the module are the same: see Figure 2-7.

NOTE: Figure 2-7 is provided as a reference only and is not intended as a cable construction guide.

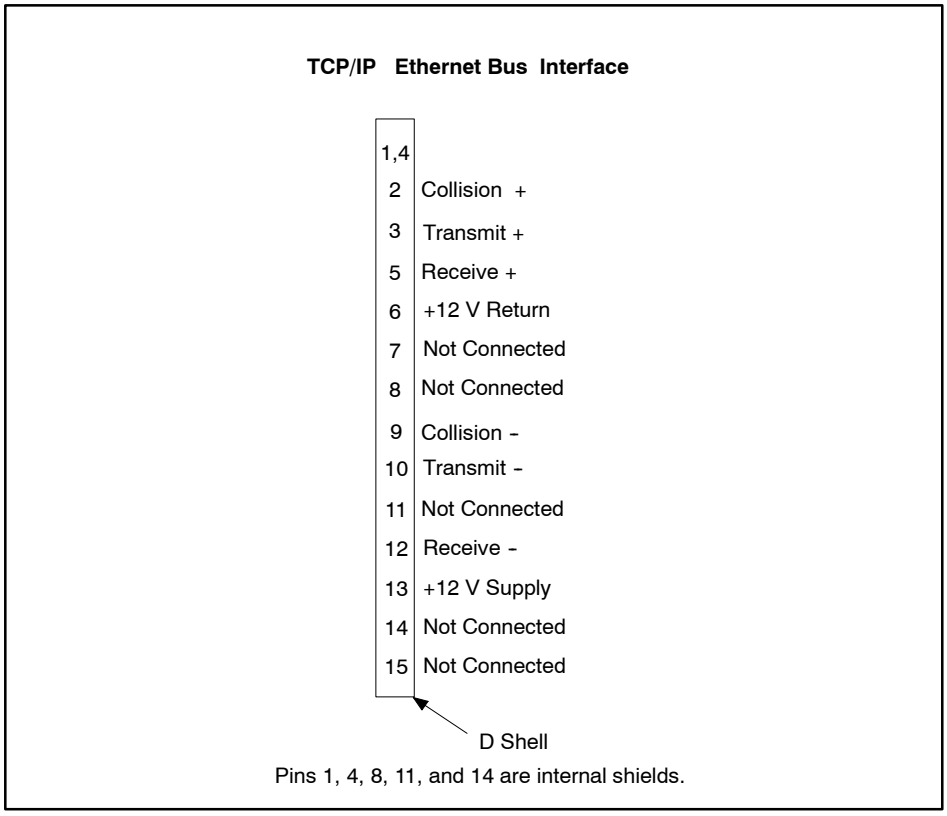


Figure 2-7 Ethernet Port Cable Pinouts

10Base5 Installations

If you are connecting the module to an existing 10Base5 (thicknet, coaxial) installation, you connect the module to the Ethernet transceiver by using a standard 15-pin Ethernet Attachment Unit Interface (AUI) cable that conforms to the IEEE 802.3 standard.

Siemens manufactures a standard AUI cable that conforms to the IEEE 802.3 Ethernet standard: order number 6ES5727-1xxxx (where xxxx = precut length).

Connect one end of the cable to the appropriate 15-pin TCP/IP Ethernet Bus Interface port on the module and the other end to the Ethernet transceiver, as shown in Figure 2-8.

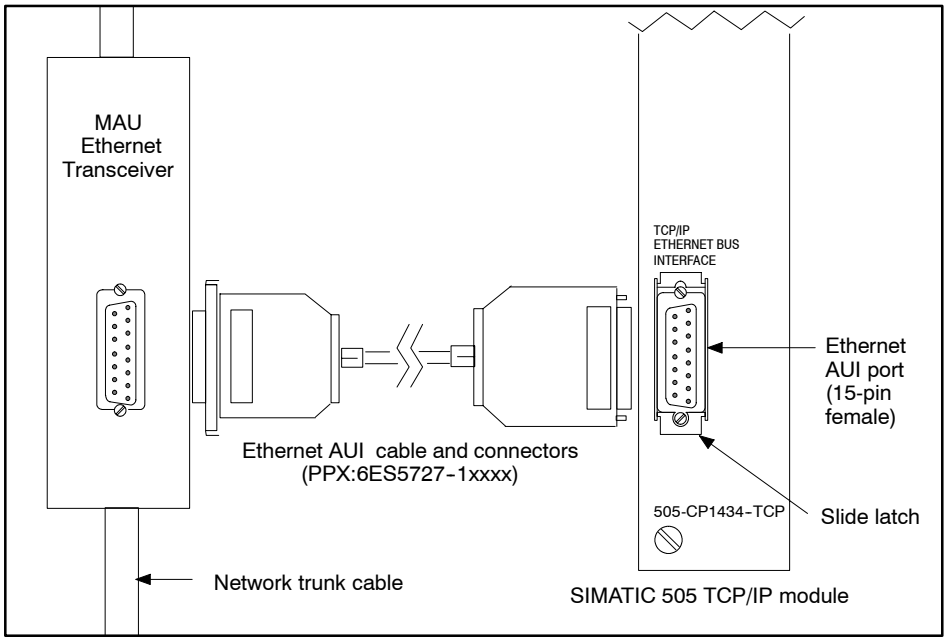


Figure 2-8 Connecting the Module with AUI Cable

10Base-T Installations

If you are connecting the module to a 10Base-T (Category 5, twisted pair) Ethernet installation, the IEEE 802.3 standard offers a number of connection options, depending on the device to which you are connecting the module.

The SIMATIC 505 TCP/IP module can be connected to the following devices in a 10Base-T installation:

- Any standard Ethernet hub or switch
- A second (505-CP1434-TCP) module
- A supported S7 CP (CP 343-1 Multi-Protocol, CP 343-1 IT, CP 443-1 Multi-Protocol, or CP 443-1 IT)
- A PC
- A 505-CP2572

Table 2-1 shows switches and hubs manufactured by Siemens that conform to the IEEE 802.3 standard for Ethernet devices.

You need to use a transceiver on the Ethernet Bus Interface port of the SIMATIC 505 TCP/IP module to connect to a 10Base-T installation. The Mini UTDE RJ45 for Industrial Ethernet (order number HIR:943 270-002) is a 15-pin-to-RJ45 transceiver available from Siemens that conforms to the IEEE 802.3 standard.

Table 2-1 Ethernet Switches and Hubs

Device	Order Number	Description
Electrical Switch Module	6GK1 105-3AA00	Industrial Twisted Pair, 9-pin
	6GK1 105-3AB00	RJ45
Optical Switch Module	6GK1 105-2AA00	Industrial Twisted Pair, 9-pin
	6GK1 105-2AB00	RJ45
Electrical Link Module	6GK1 102-5AA00	Industrial Twisted Pair, 9-pin
Optical Link Module	6GK1 102-4AA00	Industrial Twisted Pair, 9-pin

Table 2-2 through Table 2-6 show the connection options and the cable required to connect the SIMATIC 505 TCP/IP module to 10Base-T Ethernet installations. In all tables, “source” refers to the SIMATIC 505 TCP/IP module.

Table 2-2 Connecting to Switches or Hubs

Destination Port	Source Port	Cable
RJ45 port	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) cord [RJ45/RJ45], Siemens order number 6XV1 850-2GH60 (6m length)
9-pin port	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) cord **9/RJ45** Siemens order number 6XV1 850-2JH60 (6m length)
Note: For simplicity, only one order number per cable is listed. However, the cables can be purchased in other lengths that require different order numbers.		

Table 2-3 Connecting to a Second 505-CP1434-TCP

Destination Port	Source Port	Cable
Ethernet Bus Interface port with Mini UTDE transceiver	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) XP (crossed pair) cord [RJ45/RJ45] Siemens order number 6XV1 850-2HH60 (6m length)
Note: For simplicity, only one order number per cable is listed. However, the cables can be purchased in other lengths that require different order numbers.		

Table 2-4 Connecting to an S7 CP

Destination Port	Source Port	Cable
15-pin port	Ethernet Bus Interface port	AUI cable [15/15], order number 6ES5727-1xxxx (where xxxx = precut length)
15-pin port	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) XP (crossed pair) cord [RJ45/15] Siemens order number 6XV1 850-2SH60 (6m length)
RJ45 port	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) XP (crossed pair) cord [RJ45/RJ45] Siemens order number 6XV1 850-2HH60 (6m length)
Note: For simplicity, only one order number per cable is listed. However, the cables can be purchased in other lengths that require different order numbers.		

Table 2-5 Connecting to a PC

Destination Port	Source Port	Cable
RJ45 port	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) XP (crossed pair) cord [RJ45/RJ45] Siemens order number 6XV1 850-2HH60 (6m length)
Note: For simplicity, only one order number per cable is listed. However, the cables can be purchased in other lengths that require different order numbers.		

Table 2-6 Connecting to a 505-CP2572

Destination Port	Source Port	Cable
15-pin port	Ethernet Bus Interface port	AUI cable [15/15], order number 6ES5727-1xxxx (where xxxx = precut length)
RJ45 port	Ethernet Bus Interface port with Mini UTDE transceiver	TP (twisted pair) XP (crossed pair) cord [RJ45/RJ45] Siemens order number 6XV1 850-2HH60 (6m length)
Note: For simplicity, only one order number per cable is listed. However, the cables can be purchased in other lengths that require different order numbers.		

Figure 2-9 shows how to connect to a 10Base-T Ethernet installation using a Mini UTDE transceiver.

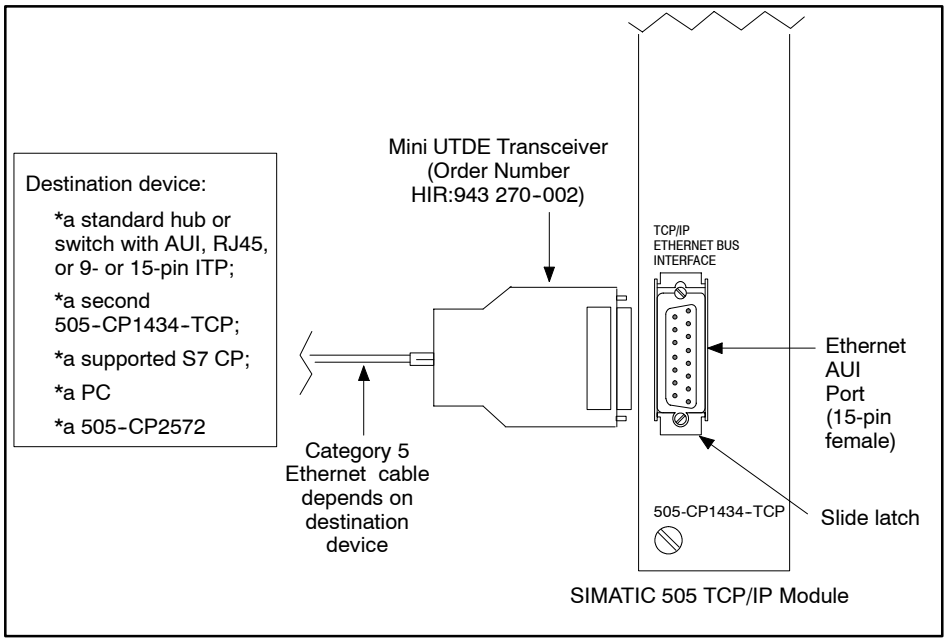


Figure 2-9 Connecting the Module through a Mini UTDE Transceiver

2.6 Powering Up the I/O Base

After installing the SIMATIC 505 TCP/IP module, power up the I/O base and observe the status of the LEDs on the front of the module.

NOTE: Refer to the system manual for your PLC for information on installing and wiring the power supply for your I/O base. Follow all installation guidelines and safety considerations described in your system manual before powering up the system.

LED Status after Power-Up

When you power up the base, the module executes a diagnostic check. The module initialization process generally takes 20 to 25 seconds.

Table 2-7 shows how the combination of both LEDs indicates the status of the module or the Configurator software. Refer also to the Troubleshooting Chart (Table B-3) in Appendix B for additional information on the status LEDs.

Table 2-7 LED Status Indicator Diagnostic Chart

MOD GOOD LED	MOD READY LED	Description
Off	Off	Base is powered down, or hardware has just been reset and power-up diagnostics are in progress.
Flashing	Off	A hardware failure was detected during power-up diagnostics.
Flashing	On	Failure in DC-to-DC converter during power-up diagnostics.
Off	Flashing	Software in memory is not executable.
On	Off	Module initialization is in progress.
On	Flashing	Power-up state after first installation before module is configured; initialization failed due to invalid or missing configuration.
On	On	Module is ready for operation.

NOTE: When you install the module for the first time and power up the base, the MOD GOOD LED indicator comes on, then the MOD READY LED indicator starts flashing. This is normal because you have not yet loaded your configuration file into the flash memory of the module.

2.7 Configuring I/O Memory in the Programmable Logic Controller

After installing the SIMATIC 505 TCP/IP module in the I/O base, you must configure the I/O starting address in your PLC memory. The module does not automatically configure its I/O address in the PLC.

NOTE: Even though the module may appear to be operating correctly, it does not communicate optimally with the PLC unless it is configured in the I/O map.

To configure the I/O map and verify the controller-to-module communications, connect a PC running SoftShop to the PLC. For more information on configuring the I/O, refer to your SoftShop manual. Follow these steps:

1. Apply power to the base and ensure that the module successfully completes power-up initialization.
2. In SoftShop, access the Configure I/O function menu.
3. Select the appropriate channel and base number.
4. Execute the Read Base function.

NOTE: For the example shown in Figure 2-10, the module appears in slot 2. The SIMATIC 505 TCP/IP module uses 8 I/O address locations: 4 WX inputs and 4 WY outputs.

---

5. Assign the starting I/O address.

NOTE: In the example shown in Figure 2-10, the starting address for the module is set for 0017. The starting address you assign here for the module will not be displayed when you configure Job Start and Job Active bits in the SIMATIC 505-CP1434-TCP Module Configurator. Instead, the Configurator uses relative addresses (“WX1” in the Configurator is really whatever you assigned as the starting address of the module in the I/O map of the PLC). So if you assign 0017 as the starting address, then WX1 in the Configurator is actually WX17 in the PLC and WY5 in the Configurator corresponds to WY21 in the PLC. In other words, “WX1” is the relative designation used by the Configurator; “WX17” is the absolute address in the PLC. See the note on page 4-9.

---

6. Write the completed I/O configuration to the PLC memory.

I/O Address:
Because discrete I/O (X, Y) and word I/O (WX, WY) do not share the same image register addresses, the module in this example could use a starting I/O address of 0001, or 0009, or 0025, and so on.

Word I/O Used:
The SIMATIC 505 TCP/IP module has 4 WXs and 4 WYs.

I/O Module Definition

Slot	I/O Addr	X	Y	WX	WY	SF
01	0001	16	00	00	00	No
02	**0017**	00	00	**04**	**04**	**Yes**
03	0000	00	00	00	00	No
04	0025	00	08	00	00	No

Slot Number:
You can install the module into any slot in the I/O base.

SF Module:
The SIMATIC 505 TCP/IP module is a Special Function Module.

Figure 2-10 Sample I/O Module Definition Chart

Chapter 3

Using the Configurator

3.1 Overview

Before you can use the SIMATIC 505 TCP/IP module, you must install and connect it (described in Chapter 2), and configure it (described in this chapter).

Table 3-1 shows the actions necessary in order to install the SIMATIC 505-CP1434-TCP Module Configurator on your personal computer (PC) and use it to configure the SIMATIC 505 TCP/IP module.

Table 3-1 Configuring the Module

Action	Section
Install and open the Configurator software on your PC.	3.2
Set network communication parameters.	3.3
Define jobs.	3.4
Edit the Connection Mode settings to match your COM port.	3.5
Save the configuration file on your PC.	3.6
Transfer (download) the configuration from your PC to the module.	3.7
Print a copy for use when configuring other modules on the network (recommended).	3.8

This chapter also describes the following optional tasks:

- Transferring configurations over the Ethernet network
- Using the Serial Port Redirect feature
- Outputting a configuration to a text file
- Checking the status of the module
- Uploading a configuration
- Loading a firmware upgrade

3.2 Installing and Opening the Configurator Software

Obtaining the Configurator Software

The Configurator software is available from these locations:

- On the World Wide Web at

  http://www.sea.siemens.com/automat/product/plc/505/au505dl.html

  (in the Downloads section)

- On the electronic manual CD for Series 505 (PPX:505-CDMANUAL-3 or higher)

You can copy the SetupCfg.exe file to your personal computer. To install the Configurator, click the Windows Start button, choose the **Run** command, and enter the following path: X:\SetupCfg.exe (where "X:" is the location of the file.) A Setup wizard guides you through the installation procedure.

Opening a Configuration File

To open the Configurator, click the Windows Start button and choose the **Programs > 505-CP1434-TCP Configurator > 505-CP1434-TCP Configurator** command (assuming that you installed the software in the default location).

Figure 3-1 shows the main functional areas of the Configurator.

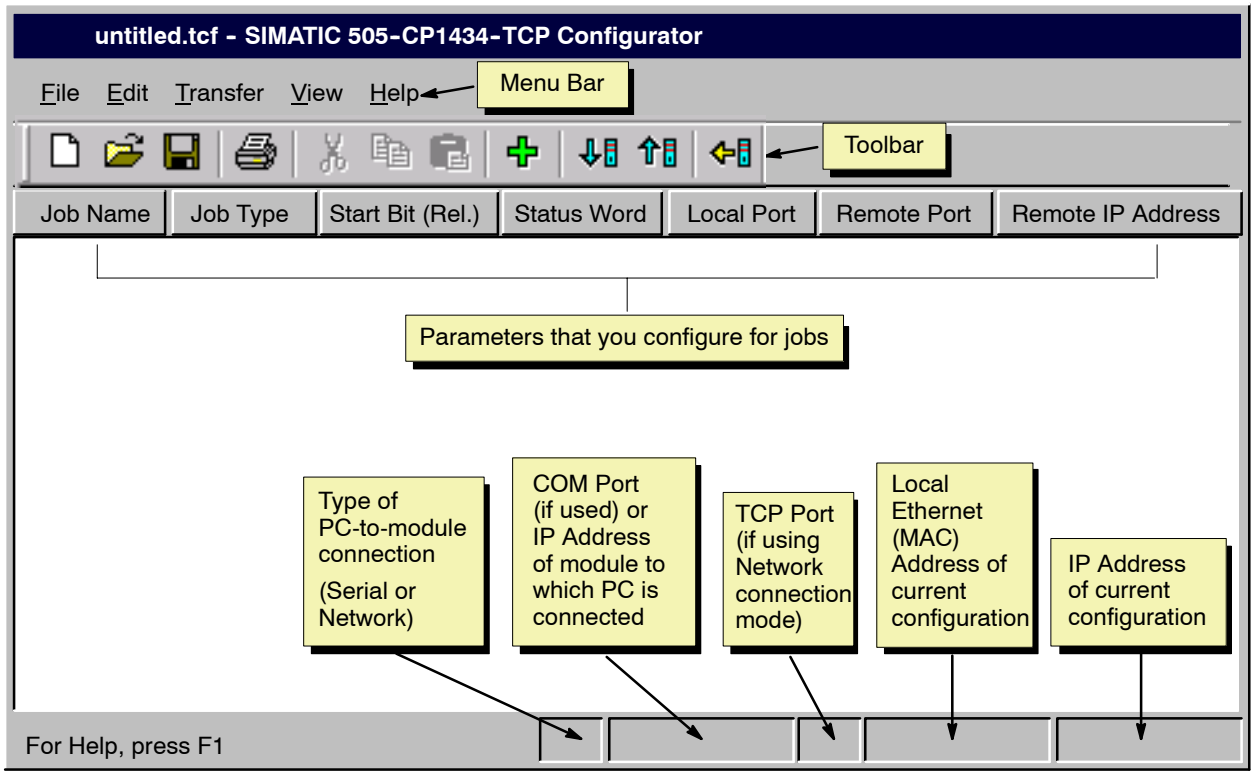


Figure 3-1 Opening the Configurator

3.3 Setting Network Communication Parameters for the Module

You must configure (and transfer) network parameters to the module in order for the module to communicate on the network. To configure network parameters, follow these steps:

1. From the menu bar, choose **Edit > Set Network Parameters** to bring up the Module Network Parameters dialog box.

2. In the Local Ethernet (MAC) Address field, enter a unique Local Area Network address value for this TCP/IP module. You may only need to change the right-most three or four digits.

NOTE: This address is sometimes referred to as the Media Access Control address. It is user-assigned, rather than hard-coded, to the SIMATIC 505 TCP/IP module. It must be unique within your local area network (LAN). This address is different from the address that identifies the module on the Internet, the IP Address, described below in step 4.

The following constraints apply to the Local Ethernet (MAC) Address:

* It must contain exactly 12 hexadecimal digits.

* It cannot be equal to any Remote Ethernet Address on your LAN.

* The address cannot be all zeros or ones (in other words, 000000000000 and FFFFFFFFFFFF are invalid addresses).

See Figure 3-2.

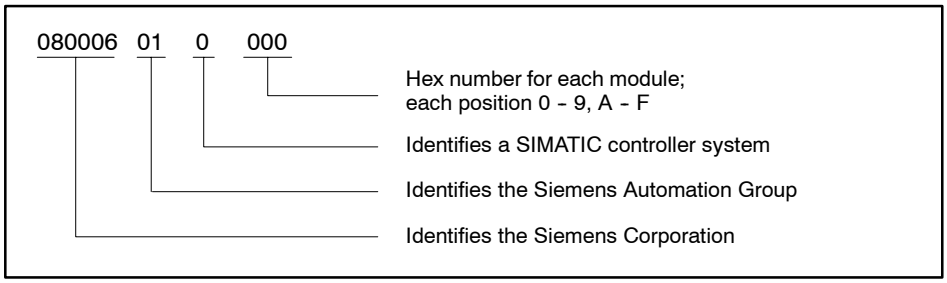


Figure 3-2 Local Ethernet (MAC) Address Format

3. (Optional) To enable serial port redirect, check the Enable Redirect box. (For more information about serial port redirect, see Section 3.11.)

4. Enter the following Network parameters to configure Internet communications.

   IP Address (required) Type the unique Internet address value to be assigned to this TCP/IP module. The valid range is from 1.0.0.0 to 223.255.255.255. See your system administrator if you do not know what value to use.

   Subnet Mask (required) Type the correct value for your network. See your system administrator if you do not know what value to use.

   Gateway (optional) Type the correct value for your network. The valid range is from 1.0.0.0 to 223.255.255.255. A blank or zero IP address means that there is no network gateway. See your system administrator if you do not know what value to use. (A gateway, also called a router, is only necessary if there is communication between the local area network and other networks.)

   Host Name (required) The host name can be up to 20 alphanumeric characters. Accept the default value, or edit it in order to give a different name to each of your SIMATIC 505 TCP/IP modules.

   The host name is automatically inserted in the MAIL FROM block of any e-mail jobs that are sent by this module.

   Domain Name (optional) The domain name can be up to 60 alphanumeric characters. See your system administrator if you do not know what value to use.

   You cannot assign a domain name if there is not a valid value for the host name parameter (described above).

   If you assign a domain name to the module, that domain name is concatenated to the host name of the module (for example, HostName.DomainName) in the MAIL FROM blocks of any e-mail jobs that are sent by this module.

5. Enter the following Broadcast parameters. (These parameters are used by Broadcast Send jobs and the optional SIMATIC 505 OPC Server for TCP/IP package. Broadcast Send jobs are described in Section 4.12, and the 505 OPC Server is described in Chapter 5.)

   Remote IP Address Accept the default value unless you have a reason to change it.

   The Broadcast Remote IP Address is the destination IP address for any data broadcast from the module. Typically, it specifies not a single device but rather a network. The default value, 255.255.255.255, specifies the local area network; otherwise the valid range is from 1.0.0.0 to 223.255.255.255. Broadcast Receive jobs and the optional SIMATIC 505 OPC Server for TCP/IP must be on the specified network in order to receive broadcasts.

   If you do not plan to broadcast to multiple devices, and are only using one SIMATIC 505 OPC Server, you can enter the IP address of the PC on which the SIMATIC 505 OPC Server resides.

   To broadcast to a network other than your local area network, you would put in a network number (left part of IP address, depending on subnet mask).

   Remote UDP Port Accept the default value (1506) unless you have a reason to change it. The valid range is from 1024 to 65535.

   The Broadcast Remote UDP Port value that you assign here is the destination port for all Broadcast Send jobs that you configure on this module. It must be used as the Local UDP Port of all Broadcast Receive jobs receiving data from the Broadcast Send jobs on this module. It is also used as the Module Port in the optional SIMATIC 505 OPC Server for TCP/IP to receive broadcast activity from this module.

   If other network devices are configured to receive their own kind of broadcasts on this port, then they may experience problems during SIMATIC 505 TCP/IP module broadcasts. Work with your system administrator to identify a port value for broadcasting that does not affect other devices on the network.

NOTE: It is permissible to use the same Broadcast Remote UDP Port value when you configure other modules on your network. Assigning the same remote UDP port value to all the modules that use Broadcast Send jobs means that there is only one port being used for broadcasts, which in turn reduces the possibility of interference with any other device on the network.

Delivery Time Accept the default value (1000 ms) unless you have a reason to change it.

The Delivery Time is a tuning parameter. It affects Broadcast Send jobs on the SIMATIC 505 TCP/IP module as well as broadcasts from the module to the optional SIMATIC 505 OPC server. The Delivery Time value is a maximum wait time, once a change of RBE data has occurred, before broadcasts of data changes are sent; they may be sent more frequently. Broadcast data changes are accumulated in a buffer (1472 bytes). RBE data changes are broadcast upon whichever condition occurs first: either when the broadcast data buffer fills up, or when the Delivery Time value is reached. See Figure 3-3.

If the Delivery Time parameter is blank or set to zero, RBE data is sent immediately. Otherwise, the valid range is 1 to 65535.

⚠ WARNING

An extremely low Delivery Time value can cause near-continuous broadcasts of RBE data, thereby flooding the network.

If the network is flooded with broadcasts, other equipment may perform unpredictably. Unpredictable operation of process equipment could cause death or serious injury to personnel, and/or damage to equipment.

Unless you expect very few RBE updates, avoid extremely low Delivery Time values.

⚠ CAUTION

Delivery Time values greater than 5000 ms cause the RBE feature of the optional SIMATIC 505 OPC Server for TCP/IP to be disabled in communications with the module. The SIMATIC 505 OPC server has to obtain data updates by polling instead.

Forcing the SIMATIC 505 OPC server to poll could slow its performance.

Consider using a Delivery Time value that is less than or equal to 5000 ms if the module is used with the SIMATIC 505 OPC server.

Decreasing the Delivery Time value does not automatically cause faster updates: it depends on your unique system configuration and how quickly broadcast data is changing in the PLC.

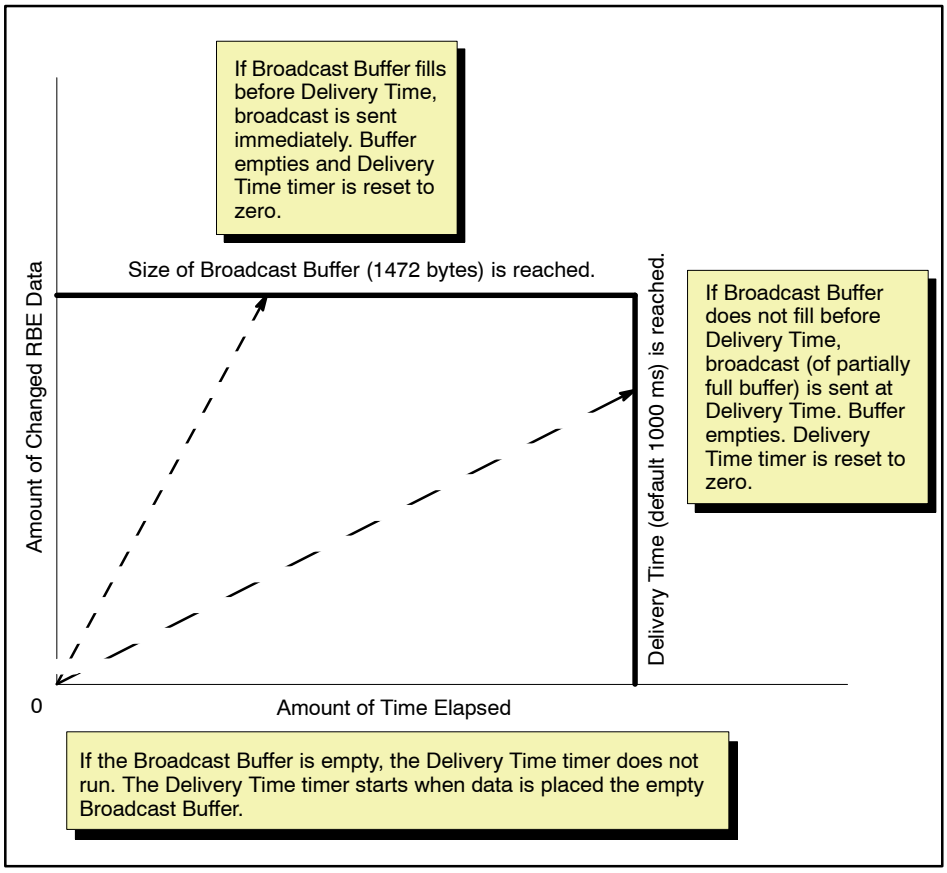


Figure 3-3 Conditions that Trigger Broadcasts

6. Click the OK button to confirm your work and close the dialog box.

NOTE: You may want to back up your configuration work in a file on your PC after defining Module Network Parameters. Choose the **File > Save** menu command when you are ready to save your work.

NOTE: Remember that your network settings cannot take effect until you transfer (download) them to the module. Downloading is described on page 3-14.

3.4 Defining Jobs

You can configure a maximum of 64 jobs in a single configuration. (Section 4.3 provides details.)

The following job types are available:

- Send/Receive (work as a pair)
- Write Active/Write Passive (work as a pair)
- Read Active/Read Passive (work as a pair)
- TCP Server
- UDP Server
- Write Remote (works with a TCP Server or UDP Server job)
- Read Remote (works with a TCP Server or UDP Server job)
- Memory Exchange (works with a TCP Server or UDP Server job)
- Broadcast Send/Broadcast Receive (one Broadcast Send works with multiple Broadcast Receives)
- E-Mail

The basic procedure for configuring jobs is outlined below. See Chapter 4 for details about the parameters required for each job.

1. From the menu bar, choose **Edit > Add Job** to bring up the Add Job Selection dialog box.
2. In the Job Name field, type a unique descriptive name for the job (maximum 39 alphanumeric characters, including punctuation and spaces).
3. From the Job Type drop-down list, select the desired job type.
4. Click the Configure button to specify parameters for the job.
5. Click the OK button when you finish with the Job Configuration dialog box, and click OK again to close the Add Job Selection dialog box.

   The configured job appears in the job list of the main Configurator window.

3.5 Editing the Connection Mode Settings

Understanding Connection Modes

There are two ways to transfer configuration information to the SIMATIC 505 TCP/IP: serial transfers or network transfers. The first time that you transfer a configuration to a module, you must use a direct, serial connection between the PC and the module. This is because you cannot transfer configuration information over the network to the module until after the module has been configured (serially) with network parameters and a TCP server job. (Of course, you do not ever have to change to network connection mode. You can transfer all configuration information, including module network parameters and jobs, serially.)

NOTE: For information about network connection mode, see Section 3.10.

Serial Connection Mode

Figure 3-4 shows a serial connection between a PC running the Configurator and a SIMATIC 505 TCP/IP module.

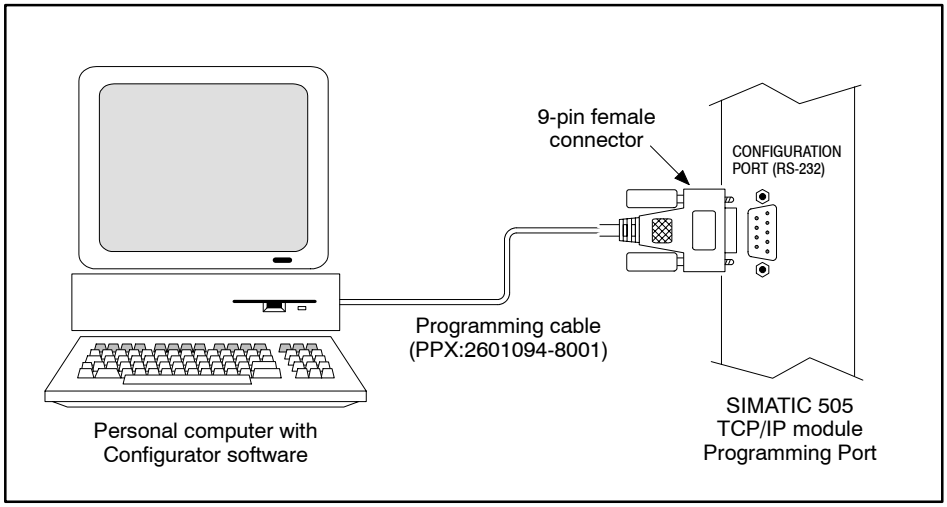


Figure 3-4 PC-to-Module Serial Connection

There are three parts to this direct, serial connection: the PC port, the cable, and the module port.

- The PC port that you use for the connection is a COM port, a 9-pin port that permits RS-232/423 communication.
- An example of a standard serial cable is Siemens' RS-232/423 serial cable (PPX:2601094–8001).
- The module port is labeled Configuration Port (RS-232). This port is also referred to as the serial port.

Checking the COM Port Setting

Before you transfer your communication serially, you need to check the COM port setting in the Configurator, and if necessary, edit it so that it matches the COM port that you are actually using on your PC for the connection to the serial port of the module.

NOTE: The status bar of the Configurator displays the current connection mode (Serial or Network). If the mode is Serial, the status bar also displays your current COM port selection. See Figure 3-5.

Figure 3-5 Status Bar of the Configurator

Changing the COM Port Setting

If the COM port setting does not match what you are actually using on your PC, edit the setting in the Connection Mode dialog box. Follow these steps:

1. Choose the **Transfer > Set Connection Mode** menu command to access the Connection Mode dialog box.
2. Make sure that the option button for Serial Port Selection is selected. If it is not, select it.
3. Choose the option button for the COM port that you are actually using on your PC. See Figure 3-6.
4. Click the OK button to confirm your work and close the dialog box.

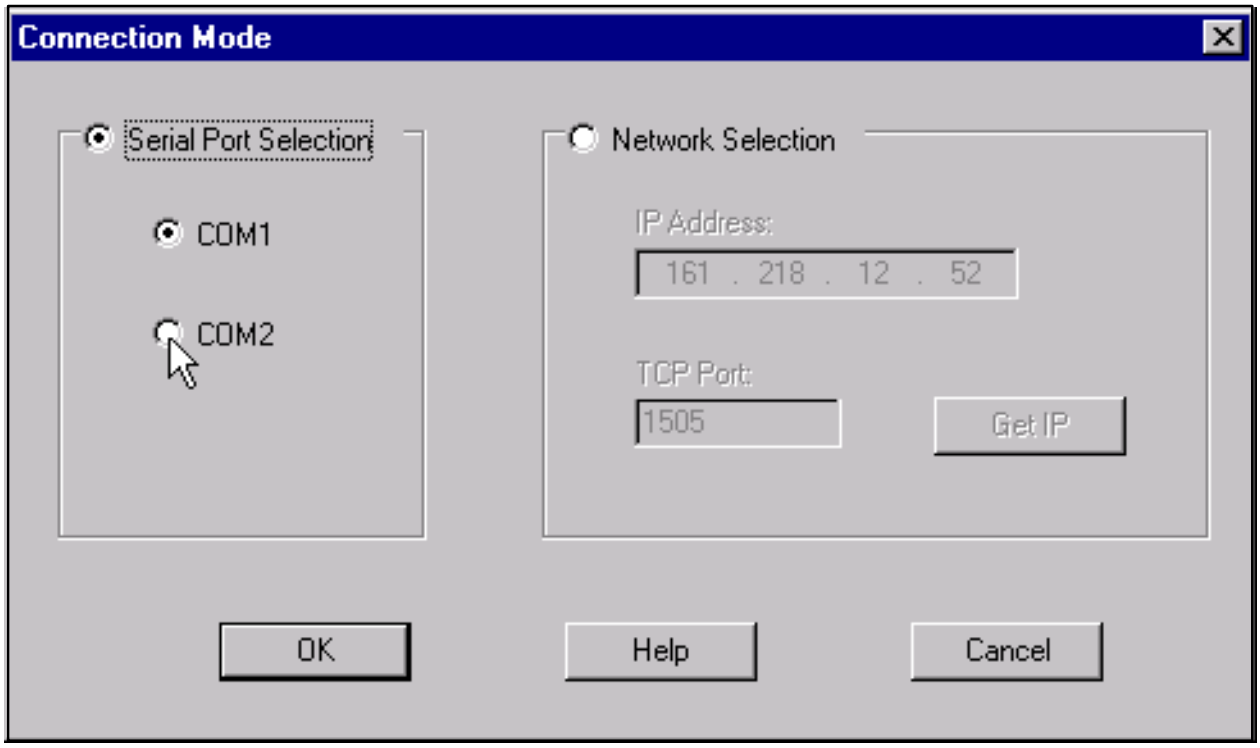


Figure 3-6 Changing COM Port Setting

3.6 Saving the Configuration in a File

Saving your work in a configuration file has the following advantages:

- The saved file provides a backup to reload in case the configuration in the module is lost or corrupted.
- You can open a saved configuration file, modify it, and save the modified file with a different name to quickly create other configurations that use some of the same parameters.
- You can examine some of the configuration parameters later when you configure other modules. For example, if you configure a Broadcast Receive, you have to know the Broadcast Remote UDP Port value of the module where the Broadcast Send is configured.
- You can add, delete, or modify jobs and other configuration settings in the future without having to first upload a copy of the configuration from the module. (However, it is important to know whether there are any differences between the saved configuration file and the actual module configuration before you decide to work with the saved configuration file. See Section 3.9.)

To save your work in a configuration file, choose the **File > Save** menu command. (If you are interested in saving your configuration file as a text file, see Section 3.12.)

3.7 Downloading the Configuration

The work that you do in the Configurator has no effect on the SIMATIC 505 TCP/IP module until you transfer (download) the configuration to the module.

NOTE: You cannot transfer a configuration that lacks valid module network parameters.

NOTE: The first time that you transfer a configuration to a module, you must use a serial connection for the transfer: the COM port of the PC where you are running the Configurator must be cabled to the serial port of the module.
* Section 2.4 describes how to cable your PC to the module serially.
* Page 3-10 describes how to verify that you have the correct COM port settings for serial communication.
* Section 3.10 describes how to use the network for subsequent transfers.

NOTE: Whenever you transfer a configuration to the module, you overwrite any configuration information that is already present in the module. See Section 3.9.

To transfer the configuration to the module, follow these steps (refer to Figure 3-7):

1. Choose the **Transfer > To Module** menu command or the toolbar button to open the Transfer Configuration to Module dialog box.

   The File field displays the name of the currently open configuration file (if it has been saved with a name).

2. Click the Start button to start the download. (The transfer operation can take several minutes.)

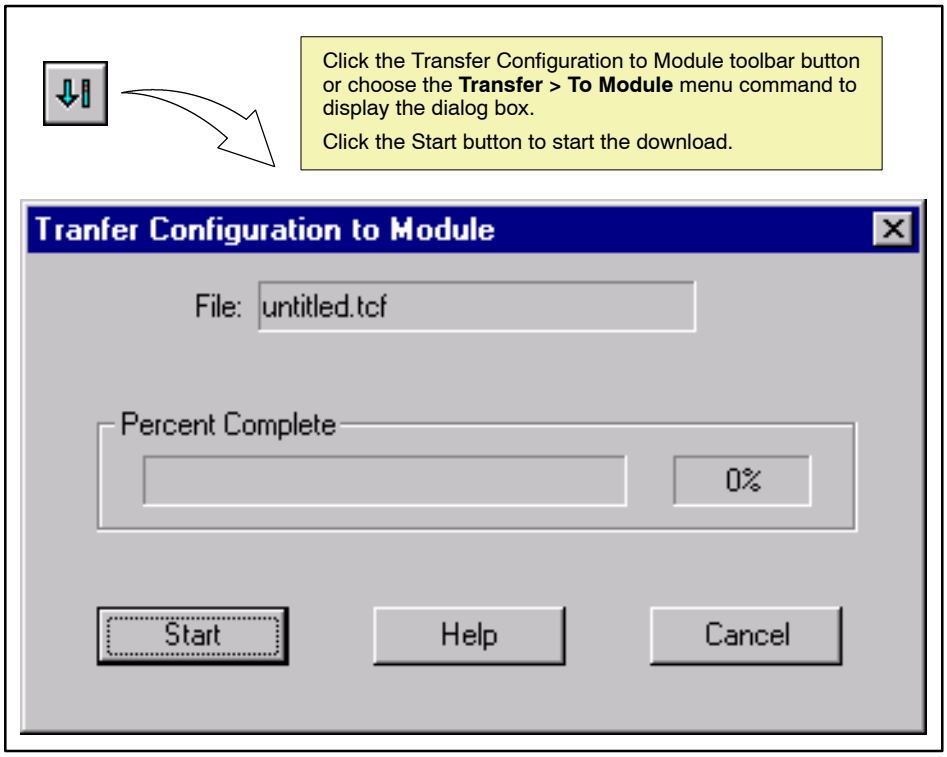


Figure 3-7 Transferring (Downloading) a Configuration

If there is already a configuration in the module, and the name of the configuration differs from the file name that you are transferring, you are prompted to decide whether to overwrite the existing module configuration at this time and continue with the download, or cancel the download to protect the existing module configuration.

NOTE: The download process completely overwrites the current module configuration. For example, if you have already configured the module with network parameters and then you perform a second download of a configuration that contains only a job and no network parameters, you are not only downloading the job, you are erasing the network parameters in the module. Therefore, when you download a configuration, your configuration should contain everything needed in the module.

When the download is complete, the Transfer Configuration to Module dialog box automatically closes.

3.8 Printing the Configuration Information

Printing your configuration information has the following advantages:

- When configuring subsequent devices on the network, you may need to know the parameters that you have assigned to this module. If you print out the configuration of each TCP/IP module, you have a record that you can consult so that you do not inadvertently create configurations that conflict with one another.
- If you have printed the configuration for a module, you do not have to use the Configurator or perform a transfer (upload) from the module to the PC in order to obtain the necessary information.

The printed output of your configuration is formatted to fit the margins used by your printer, and includes a header with the name of the Configurator and page numbering.

To view the file before printing it, choose the **File > Print Preview** menu command.

To change the paper size or the orientation of the page (portrait, landscape), choose the **File > Print Setup** menu command.

To print the job, choose the **File > Print** menu command. You can specify the printer, range of pages to print, and number of copies to print.

3.9 Revising an Existing Module Configuration

It is always good practice to save backup copies of your configuration work; however, if you want to revise an existing module configuration, the safest procedure is to upload the existing configuration from the module, edit it, and then download the revised configuration. (Section 3.14 describes uploading; Section 3.7 describes downloading.) This is the only way to know exactly what configuration data is present in the module.

NOTE: If the configuration of the module has been saved in a file, you can edit the file and then download the revised file to the module. However, the disadvantage of doing so is that if the saved file was not identical to the configuration in the module, then you may inadvertently lose aspects of the previous module configuration without realizing it.

Example:
The first person to configure the module saves the configuration in a file.

Later a second person adds a UDP Server job to the module, in order to enable network configuration work. The second person does not save this revised configuration in a file.

Now you want to add a Receive job to the module. If you add the Receive job to the saved configuration file, and download the revised file, you are overwriting the existing module configuration with one that does not contain a UDP Server job—suddenly, the network connection between the Configurator and the module is going to stop working.

⚠ WARNING

If you edit a configuration that is saved on disk instead of uploading the configuration from the module for edit, configuration data could be lost when you transfer the edited configuration from the disk to the module. If configuration data is lost from a module that is connected to process equipment, it can cause unpredictable operation.

Unpredictable operation of process equipment can cause death or serious injury to personnel, and/or damage to equipment.

To prevent unpredictable operation, always upload the existing configuration from the module and edit that configuration, instead of editing and downloading a saved configuration file (which might not be identical to the existing configuration in the module).

3.10 Transferring Configurations over the Network

Network Connection Mode

The first time that you transfer a configuration to the module, you must use a direct serial connection (described in Section 3.5). After you transfer an initial configuration to the module serially, you can perform subsequent configuration work over the network, if desired.

NOTE: To use the Configurator to communicate with a SIMATIC 505 TCP/IP module over the TCP/IP network, the PC on which the Configurator is running must be Ethernet-capable. If your PC is not Ethernet-capable, you can install a standard Ethernet card such as the CP1613 card, available from Siemens.

Prerequisite Tasks

In order to successfully transfer configuration data over the network, you must first perform these prerequisite tasks:

- Configure module network parameters. See Section 3.3.
- Configure a TCP Server job. See Section 4.7.
- Transfer (serially) the configuration containing module network parameters and the TCP Server job to the module. See Section 3.7.

NOTE: You can select the Network option button in the Connection Mode dialog box at any time, but that does not mean that you can actually transfer configuration data over the network. If you attempt to transfer configuration data over the network, but the module does not already contain a configuration with its Local Ethernet (MAC) Address, IP Address, and a TCP Server job, the network transfer operation fails.

Changing to Network Connection Mode

To change from Serial connection mode to Network connection mode, follow these steps:

1. To access the Connection Mode dialog box, choose the **Transfer > Set Connection Mode** menu command.
2. Click the Network Selection option button.

   Figure 3-8 shows the Connection Mode dialog box.
3. Enter the IP Address (Internet address) of the module on the network to which you wish to communicate.

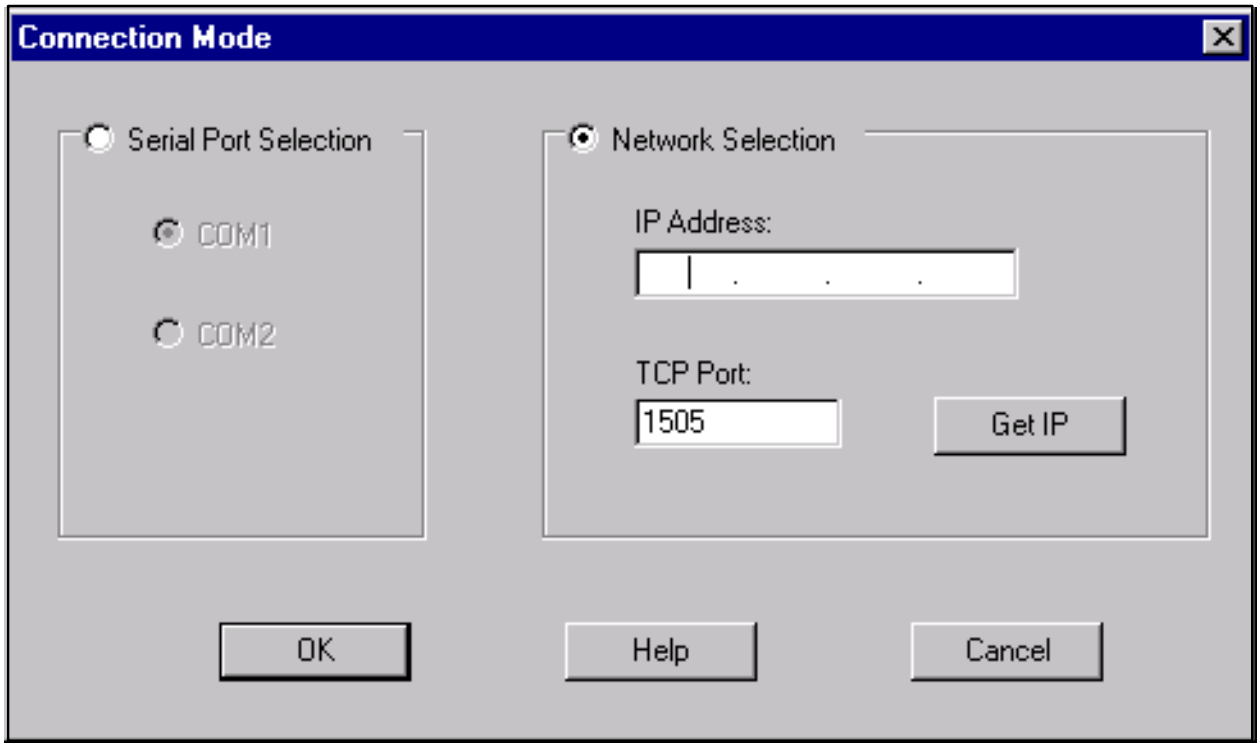


Figure 3-8 Connection Mode Dialog Box

NOTE: To read the IP Address from the current configuration in the Configurator, click the Get IP button. The Local Port value of the first TCP Server job in the current configuration is also read. This makes it easy to configure a network connection when the current configuration matches the configuration in the destination module.

---

4. Enter a TCP Port value that corresponds to the Local Port value of the TCP Server job that you have configured in the module.

5. Click the OK button to confirm your work and close the dialog box.

Now that you have changed the connection mode setting, the Configurator is ready to perform transfers over the network.

3.11 Serial Port Redirect

Overview

The Serial Port Redirect operation allows you to redirect an application from the serial port (labeled Configuration Port) of a SIMATIC 505 TCP/IP module to any Series 505 PLC on the network, as though the application were cabled directly to the RS-232 serial port of that PLC. In this way, you can use the serial port of a SIMATIC 505 TCP/IP module to run an application such as SoftShop and access a PLC.

NOTE: Your Serial Port Redirect setting has no effect on subsequent configuration activity. In other words, you are not redirecting a configuration to another module. You are only redirecting applications such as SoftShop, which use serial port communication, so that they can communicate through a SIMATIC 505 TCP/IP module instead of using a direct serial port connection to a PLC.

Once you set the Serial Port Redirect from the Configurator, you can simply open the desired application (such as SoftShop) and connect in the usual way. But instead of communicating to the module that is serially attached to the PC, you communicate through that module to a remote module. The communication redirection is not visible from within the application.

NOTE: When you use Serial Port Redirect, the application (for instance, SoftShop) is actually operating on a different module/PLC base combination than the one to which it is physically connected. Nothing in the application software gives any indication of whether Serial Port Redirect is in effect. The only way to find out is to examine the Serial Port Redirect setting in the SIMATIC 505 TCP/IP Module Configurator. Consequently, it is possible for a user to mistakenly modify the wrong PLC configuration.

⚠ **WARNING**

When you use Serial Port Redirect, it is possible for a user to mistakenly modify the wrong PLC configuration.

If a user inadvertently modifies the wrong PLC configuration, it could cause unpredictable process operation, which could result in death or serious injury to personnel, and/or damage to equipment.

Only trained, authorized personnel should have access to the PLC configuration. If more than one individual is authorized to modify the PLC configuration, you must ensure that everyone is aware of the safety implications of using Serial Port Redirect, and how to check whether it is in use.

NOTE: When power is cycled to the module, or the module is reset, or a new configuration is downloaded to the module, the Serial Port Redirect connection to the remote module is broken and must be reestablished through the SIMATIC 505 TCP/IP Module Configurator. This could cause confusion if users expect Serial Port Redirect to be in force before it has been reestablished. It is possible under such circumstances for a user to mistakenly modify the wrong PLC configuration.

⚠ **WARNING**

When power is cycled to the module, or the module is reset, or a new configuration is downloaded to the module, the Serial Port Redirect connection to the remote module is broken and must be reestablished through the SIMATIC 505 TCP/IP Module Configurator.

If a user inadvertently modifies the wrong PLC configuration (expecting Serial Port Redirect to be in force), it could cause unpredictable process operation, which could result in death or serious injury to personnel, and/or damage to equipment.

If the connection to the remote module is broken you must redo the Serial Port Redirect: Use the Transfer>Set Redirect command, ensure that the correct IP address is specified in the User Selected Destination, and click the Set Redirect button in order to set the Serial Port Redirect to the remote module again.

To set the Serial Port Redirect to a target module/PLC pair, you must perform several operations:

- Configure the Module Network Parameters of the module(s) involved in the redirect. For the module whose serial port is being redirected, you need to ensure that the "Enable Redirect" option is selected in its Module Network Parameters. (**Edit > Set Network Parameters**)
- Configure the redirect target module with a TCP Server or UDP Server job. (**Edit > Add Job**)
- Download all of the above configurations. (**Transfer > To Module**)
- Configure Serial Port Redirect parameters on the module whose serial port is being redirected. (**Transfer > Set Redirect**)

  (The Set Redirect button causes the redirect to take effect in the module immediately; downloading is not required.)

Configuring the Serial Port Redirect

After you have configured the necessary module network parameters and a server job on the destination module, you can use the **Transfer > Set Redirect** menu command to configure parameters in the Serial Port Redirect dialog box.

In the User Selected Destination area, you can define up to 50 targets. A target is combination of an IP address and a server job (defined by its port value and protocol).

IP Address The valid range is 1.0.0.0 to 223.255.255.255.

Port The valid range is 1024 to 65535.

UDP, TCP Choose the protocol of the server job on the target module.

Local The Local button enables you to set the Serial Port Redirect to the local module rather than a remote module. It supplies the universal loopback address (127.0.0.1) in the IP Address field, and also sets the protocol to TCP and the port to 1505. The universal loopback address has the same effect as the actual IP address of your local module in the IP Address field.

Get Redirect Click the Get Redirect button to verify the IP Address, Port, and Protocol of the current Redirect selection in the module (if the module contains a Redirect selection). The information appears in the Current Redirect Destination area and overwrites any data in the User Selected Destination area.

Set Redirect Sends the current IP address, port, and protocol selection in the User Selected Destination area to the specified module. It also places the information in the Destination List Selection area and the Current Redirect Destination area.

Destination List Selection Each time you click the Add button, the OK button, or the Set Redirect button, the target information from the User Selected Destination area is added to the Destination List Selection. The Destination List can be used two ways:

- Click an item in the Destination List to display it in the User Selected Destination.
- Double-click an item in the Destination List to set the Serial Port Redirect in the module to that item.

Add Lets you add the information from the User Selected Destination area to the Destination List Selection.

Delete Lets you delete a selected target in the Destination List Selection area from the destination list.

IP Address If you set the redirect or read the redirect setting from a module, the IP address is displayed in this field.

Port If you set the redirect or read the redirect setting from a module, the port of the server job is displayed in this field.

Protocol If you set the redirect or read the redirect setting from a module, the protocol of the server job is displayed in this field.

Setting a Target for Serial Port Redirect

You can specify multiple potential targets (up to 50) for Serial Port Redirect. However, you can only set the Serial Port Redirect to one target at a time. A "target" is not simply a module; rather, it is a specific TCP Server job or UDP Server job on a given module.

To set the Serial Port Redirect to a target, follow these steps:

1. To access the Serial Port Redirect dialog box, choose the **Transfer > Set Redirect** menu command. Figure 3-9 shows an example of the Serial Port Redirect dialog box.

2. In the IP Address field, enter the Internet address of the target module. The valid range is 1.0.0.0 to 223.225.255.255.

3. In the Port field, enter the Local Port value of a TCP Server job or UDP Server job that you have configured on the target module. The valid range is 1024 to 65535.

4. Click the TCP or UDP option button to indicate whether the Server job that you have configured on the module is a TCP Server job or a UDP Server job.

5. Click the Set Redirect button to set the Serial Port Redirect to the target that you have just defined in the User Selected Destination area.

NOTE: When you click the Set Redirect button, your Serial Port Redirect setting takes immediate effect in the module. You do not have to use the **Transfer > To Module** menu command to download the setting.

---

6. Click OK to save your work when you are finished.

Now you can connect your application to the serial port of the module and test the serial port redirect.

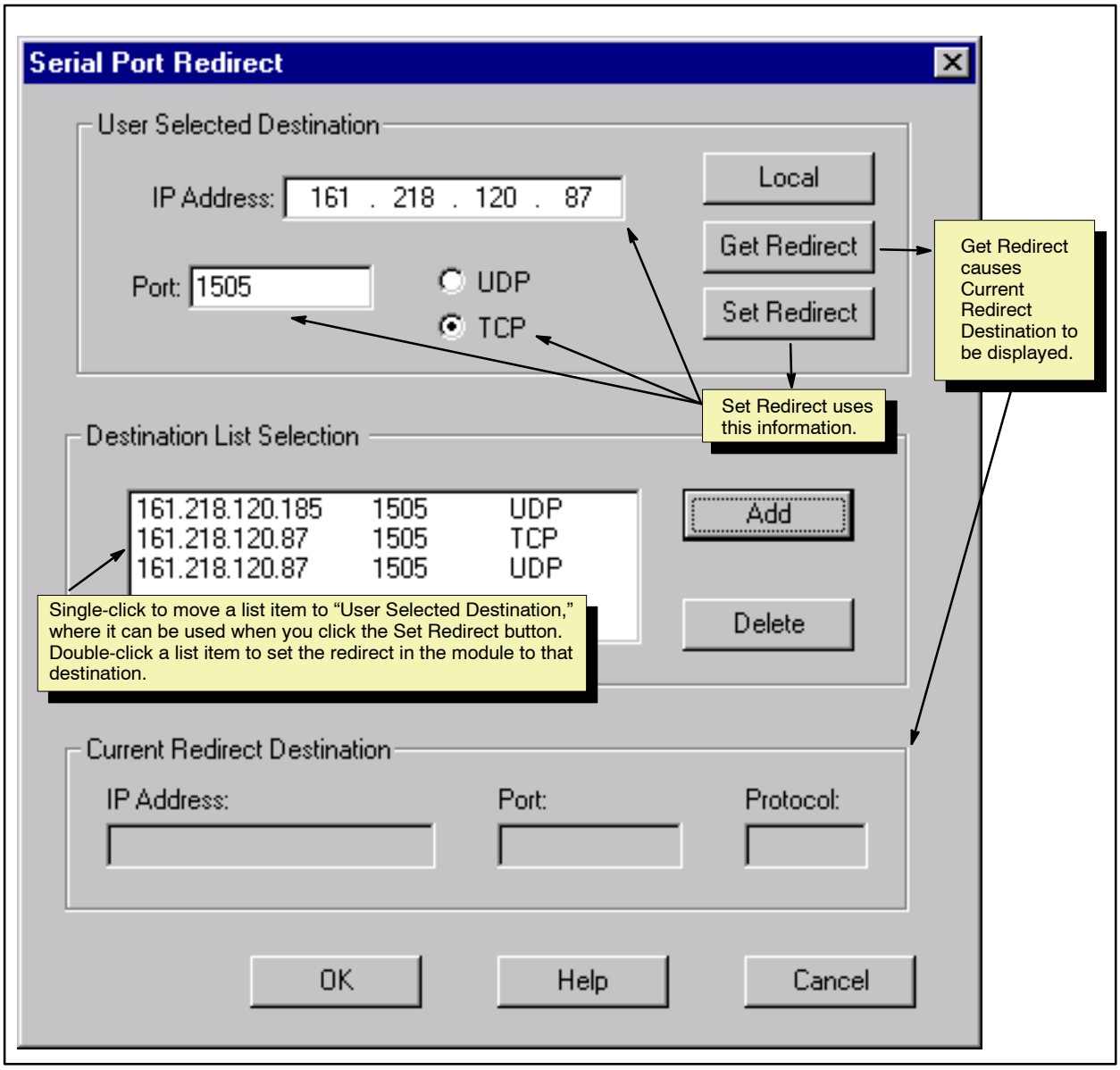


Figure 3-9 Serial Port Redirect

Example: Redirecting to a Remote PLC

This example is typical of many applications:

In your first base, you have Module A and PLC A. In your second base, you have Module B and PLC B. You want to run SoftShop on a PC that is connected to Module A, and use it to program PLC B. You download a configuration to each module as shown in Table 3-2.

Table 3-2 Configurations for Serial Port Redirect

Module A	Module B
Use **Edit > Set Network Parameters** to configure Module Network Parameters: * Local Ethernet (MAC) Address * All Network parameters * Check box for Enable Redirect Page 3-4 describes the Module Network Parameters dialog box in detail.	Use **Edit > Set Network Parameters** to configure Module Network Parameters: * Local Ethernet (MAC) Address * All Network parameters Page 3-4 describes the Module Network Parameters dialog box in detail.
Use **Transfer > Set Redirect** to configure Serial Port Redirect parameters: * IP Address of Module B * Protocol of Server job configured on Module B (TCP or UDP) * Port of Server job configured on Module B Page 3-24 describes in detail how to set the redirect.	Use **Edit > Add Job** to configure a TCP Server or UDP Server job. TCP Server jobs are best for complex or congested networks. UDP Server jobs are fine for simple LANs. See Section 4.7 and Section 4.8 for more information about these jobs.
Note: You can configure either module first, but the Serial Port Redirect cannot function correctly until after Module B has been configured with module network parameters and a server job.	

After you have downloaded the configuration parameters necessary for Serial Port Redirect to each module, you can close the Configurator program if desired. Assuming that your PC is connected to the serial port of Module A, you can start SoftShop in serial connection mode and use it to program PLC B.

NOTE: There is no relationship between Serial Port Redirect and Serial connection mode (described on page 3-10). Serial Port Redirect is used by applications like SoftShop. The Serial connection mode setting, by contrast, only affects the way that the Configurator transfers its parameters to a module (or, in the case of E-mail message data, to a PLC).

The most common reason to use Serial Port Redirect is to access a remote PLC by means of a second SIMATIC 505 TCP/IP module, but it is also possible to use Serial Port Redirect to access the PLC that is associated with the local SIMATIC 505 TCP/IP module.

Example: Redirecting to the Local PLC

Ordinarily it is easiest to connect directly to the serial port of a local PLC. But if you wanted to use Serial Port Redirect to program a local PLC (redirect from Module A to PLC A), you would configure Module A with all of the parameters described in Table 3-2 as follows:

- Configure Module Network Parameters: check the box for Enable Redirect and fill out the Local Ethernet (MAC) Address and all Network parameters.
- Configure a TCP Server or UDP Server job.
- Download the above configuration parameters.
- Configure Serial Port Redirect parameters:

  – Specify the IP Address of Module A. (You can use the Local button, which supplies the universal loopback address, 127.0.0.1, and sets the protocol to TCP and the port to 1505.)

  – Specify the Protocol and Port of the Server job that you configured (if not using the Local settings).

  – Click the Set Redirect button. The Serial Port Redirect parameters take effect in the module immediately.

This procedure allows you to redirect an application such as SoftShop from the serial port of Module A to PLC A.

3.12 Outputting a Configuration to a Text File

When you save a configuration file as a .TCF file, as described on page 3-13, it is a binary file and can only be read by the Configurator. In contrast, if you output a configuration as ASCII text, you can read it with any text editor.

To save your configuration as a text file, choose the **File > Output to Text File** menu command. Browse to the directory where you want to store the file, type the desired name in the File name field, and click the Save button.

You can open the resulting text file with a standard text editor or word processor.

3.13 Checking the Module Status

To check module status, choose the **View > Module Status** menu command.

If you need to update the status information (for instance, because you have kept the Module Status dialog box open for an interval of time), click the Get Status button.

The Module Status dialog box supplies the following information:

Module File Name Name of the configuration file that is currently loaded in the module.

BOOT Release Number Release number of the BOOT software in the module. The BOOT software performs power-on diagnostics and assists in the firmware upgrade process. It typically is not changed when module firmware is upgraded.

Firmware Release Number Release number of the application firmware in the module. Firmware is upgradable in the field; see Section 3.15.

PLC Interface Condition of connection to PLC. See Table 3-3.

Last Transfer Status Whether the most recent download from the Configurator to the module was successful. See Table 3-3.

Hardware Status Condition of chips on the module. See Table 3-3.

RBE Broadcast Enabled RBE broadcasts are enabled for the module if either of the following conditions is true:

- A Broadcast Send job is configured on this module.
- A connection has been established between this module and a SIMATIC OPC Server for TCP/IP that is configured to monitor RBE data from the associated PLC.

"RBE Broadcast Enabled" indicates whether the module is currently capable of sending RBE broadcasts, not whether such activity is currently taking place on the module.

Table 3-3 lists the possible status messages and remedies.

Table 3-3 Possible Status Messages

Field	Possible Messages	Action Required
PLC Interface	Good	No action required.
	Good — No Broadcast	No action required. The PLC associated with this module does not support Broadcast Send (but you can still configure Broadcast Receive jobs). This module will not be able to support broadcasts for the SIMATIC 505 OPC Server for TCP/IP.
	Bad — Base Not Configured	Configure the base.
	Bad	Call the technical service hotline at 800-333-7421 or 423-262-2522.
Last Transfer Status	Good	No action required.
	Bad	The last transfer (download) operation did not succeed. The previous configuration, if any, remains in effect.
	Not Loaded	There is no configuration in the module.
Hardware Status	Good	No action required.
	Bad	Call the technical service hotline at 800-333-7421 or 423-262-2522.
	Bad: Error Code XX	Call the technical service hotline at 800-333-7421 or 423-262-2522.
RBE Broadcast Enabled	Yes	RBE Broadcasts from this module are currently possible.
	No	RBE Broadcasts from this module are currently not enabled.

3.14 Uploading

Transferring a Configuration from Module to PC

Transferring (uploading) a configuration from the module to the PC allows you to recover a configuration that is in a module and copy it to the PC, where it can be viewed and worked on.

The configuration file can be transferred either through a serial or a network connection, based on the connection mode of the module.

If you need to check the file name of the configuration that is currently in the module, choose the **View > Module Status** menu command.

Choose the **Transfer > From Module** menu command to upload the configuration of the module to the Configurator. If you have unsaved work in the Configurator, you are prompted to save your work before the upload proceeds.

3.15 Loading a Firmware Upgrade

From time to time, firmware upgrades for the SIMATIC 505 TCP/IP module may be available. Since the SIMATIC 505 TCP/IP module design includes field-upgradable system memory, commonly known as “flash EPROM” memory, you can perform a firmware upgrade with software you obtain from Siemens.

NOTE: To determine if an upgrade is available, contact your Siemens distributor or the Siemens technical support hotline. If you need assistance in contacting your U.S. distributor, call 800-964-4114. If you need assistance in contacting the Siemens technical support hotline, call 800-333-7421 or 423-262-2522.

To check the revision level of the firmware that is currently in the module, choose the **View > Module Status** menu command.

NOTE: To load a firmware upgrade, the Configurator must be in Serial Connection mode (see Section 3.5).

To load a firmware upgrade to the module, follow these steps:

1. Choose the **Transfer > Load Firmware** menu command.
2. Click the Get File button and browse to the location of the upgrade file, then select the desired file and click the Open button.
3. Click the Start button to start the load operation.

NOTE: It can take approximately 5 minutes to load a firmware upgrade to the module.

Chapter 4

Configuring Jobs

4.1 Understanding the Job Types

Overview

The SIMATIC 505 TCP/IP module offers four categories of jobs: e-mail, broadcast jobs, server jobs, and point-to-point jobs (the majority of jobs).

E-mail jobs allow you to send a text message (it can include variable information) to an e-mail server when a certain condition occurs. See Section 4.13 for more information about this job.

The Broadcast jobs function as a team. You typically configure a Broadcast Send in one module and corresponding Broadcast Receives in multiple modules on the network. You use this job combination when you want to communicate the same information to several PLCs simultaneously. See Section 4.12.

The TCP Server and UDP Server jobs use different Internet protocols. They support a variety of jobs and services. In several cases, you can use either server job. UDP Server jobs are appropriate for local communications with little congestion on the network. TCP Server jobs offer more security on a congested network because of the way they automatically confirm successful communications and retry failures. Section 4.7 describes TCP Server jobs. Section 4.8 describes UDP Server jobs.

The majority of jobs offered by the SIMATIC 505 TCP/IP module are point-to-point data transfer. They follow a common scenario. You have two PLCs, each with a SIMATIC 505 TCP/IP module or other compatible module. The first PLC, PLC A, turns on a Job Start bit, which triggers a job request. Module A communicates the job request to Module B. Module B writes and/or reads data from PLC B and reports back to Module A if necessary.

The following jobs transfer information point-to-point:

- Send/Receive
- Write Active/Write Passive
- Read Active/Read Passive
- Write Remote (to a TCP Server or UDP Server)
- Read Remote (from a TCP Server or UDP Server)
- Memory Exchange (to/from a TCP Server or UDP Server)

Table 4-1 shows what function each point-to-point job or job pair performs (write, read, or both) and the configuration requirements.

Table 4-1 Comparison of Point-to-Point Jobs

Task You Want to Perform	Method of Configuration	Choose...
Send data from PLC A to PLC B	You want to configure data block addresses for PLC B on Module A.	**Write Active** (on Module A) with **Write Passive** (on Module B) -or- **Write Remote** (on Module A) with **TCP Server** or **UDP Server** (on Module B)
	You want to configure data block addresses for PLC B on Module B.	**Send** (Module A) with **Receive** (Module B)
Read from PLC B into PLC A	The only method available is to configure data block addresses for PLC B on Module A.	**Read Active** (on Module A) with **Read Passive** (on Module B) -or- **Read Remote** (on Module A) with **TCP Server** or **UDP Server** (on Module B)
Write to and read from PLC B	The only method available is to configure data block addresses for PLC B on Module A.	**Memory Exchange** (on Module A) with **TCP Server** or **UDP Server** (on Module B)

How do you choose between the point-to-point options? Consider the following factors:

- Do you need a job to write information to a destination PLC, to read information from a destination PLC, or one job that can do both? (Only Memory Exchange can do both.)
- Do you need to configure memory address parameters for the PLCs separately (configure PLC A on Module A and configure PLC B on Module B)? (Only the Send/Receive pair allows you to configure PLC B on Module B; all others require data block addresses for both PLCs to be configured on Module A.)
- Are you using an S7 CP module or a CP2572 with your 505 TCP/IP module? See "Job-to-Module Compatibility," below.

Job-to-Module Compatibility

If both of your modules are SIMATIC 505 TCP/IP modules, you can use any job combination. However, if you are using a SIMATIC 505 TCP/IP module with another supported module, such as an S7 CP (CP 343-1 Multi-Protocol, CP 343-1 IT, CP 443-1 Multi-Protocol, or CP 443-1 IT) or a CP2572, you need to choose a job that works with that module. Table 4-2 shows which jobs are compatible with which modules.

Job-to-Job Matching

All of the jobs except E-mail work in pairs, not alone. For instance, if you configure a Send job, you also have to configure a Receive job. Table 4-2 shows the correspondence between the various jobs.

Table 4-2 Job-to-Job Correspondence

Source	Destination		
SIMATIC 505 TCP/IP	**SIMATIC 505 TCP/IP**	**S7 CP**	**CP2572**
Send	Receive	Receive	N/A
Receive	Send	Send	N/A
Write Active	Write Passive	Write Passive	N/A
Write Passive	Write Active	N/A	N/A
Read Active	Read Passive	Fetch Passive	N/A
Read Passive	Read Active	N/A	N/A
UDP Server	Write Remote, Read Remote, or Memory Exchange	N/A	PLC command block equivalent of Write Remote, Read Remote, or Memory Exchange
TCP Server	Write Remote, Read Remote, or Memory Exchange	N/A	PLC command block equivalent of Write Remote, Read Remote, or Memory Exchange
Write Remote	TCP Server or UDP Server	N/A	PLC command block equivalent of TCP Server or UDP Server
Read Remote	TCP Server or UDP Server	N/A	PLC command block equivalent of TCP Server or UDP Server
Memory Exchange	TCP Server or UDP Server	N/A	PLC command block equivalent of TCP Server or UDP Server
Broadcast Send	Broadcast Receive	N/A	N/A
Broadcast Receive	Broadcast Send	N/A	N/A
E-mail	Communicates to an SMTP mail server, not a module		

Notes:

When a module is configured with Serial Port Redirect, the target module must be configured with either a TCP Server job or a UDP Server job. Serial Port Redirect is described in Section 3.11.

A UDP Server job must be configured on a module that is in communication with the SIMATIC 505 OPC Server for TCP/IP. The SIMATIC 505 OPC Server for TCP/IP is described in Chapter 5.

4.2 Understanding Commonly Used Job Parameters

In order to configure jobs for the SIMATIC 505 TCP/IP module, you need to understand the function of the following common configuration parameters:

- Local and Remote Ports
- Job Start and Job Active bits

Local and Remote Port Parameters

You can configure more than one of a given job type on a SIMATIC 505 TCP/IP module. The Local Port parameter is what differentiates the jobs. Most jobs work as a pair: Send with Receive, Write Active with Write Passive, and so forth. Therefore, the two jobs must be able to connect to one another. They do so by means of port values: the Local Port value on one job must match the Remote Port value on the corresponding job. See Figure 4-1.

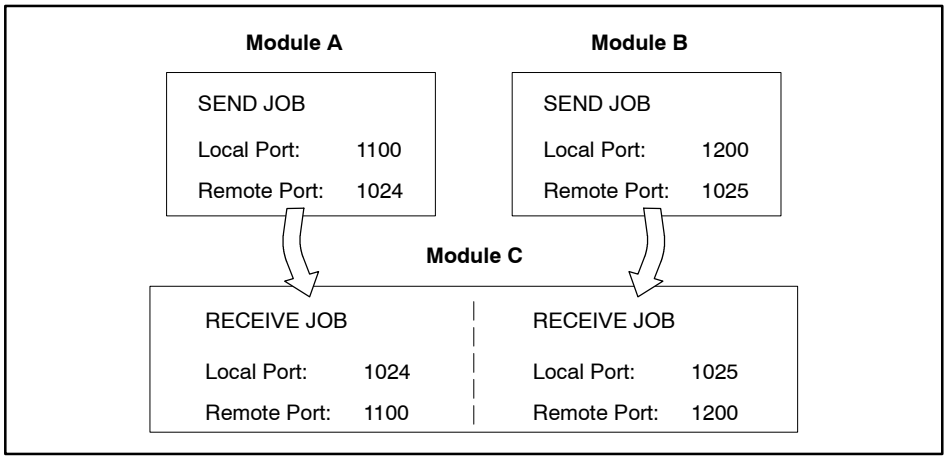


Figure 4-1 Local and Remote Ports

Local Port Assignments

Here are the rules that govern local port assignment.

- The range of local port numbers that are available for user assignment is 1024 to 65,535.
- You cannot reuse a local port number assignment for two different jobs on a module unless the jobs use different Internet protocols (TCP, UDP).

  For instance, if one job uses TCP protocol and one job uses UDP protocol, you could define the Local Port of both jobs as 1505. But if both jobs used the same protocol, you would have to assign each job a different Local Port value. See Table 4-3.
- For some jobs, it is permissible to leave a port parameter unassigned; however, completing all parameter assignments ensures that no unintended connections can be made. Section 4.14 describes which port parameters are mandatory and which are optional.

Protocols Used by Jobs

Table 4-3 shows the Internet protocols that are used by different jobs.

Table 4-3 Protocols Used by Different Jobs

Job Type	Uses TCP	Uses UDP
Send	✓	
Receive	✓	
Write Active	✓	
Write Passive	✓	
Read Active	✓	
Read Passive	✓	
TCP Server (Note 1)	✓	
UDP Server (Note 1)		✓
Write Remote (Note 2)	✓	✓
Read Remote (Note 2)	✓	✓
Memory Exchange (Note 2)	✓	✓
Broadcast Send (Note 2)		✓
Broadcast Receive (Note 3)		✓
E-mail (Note 2)	✓	

Notes:

1) There is no Remote Port parameter on these jobs.

2) These jobs do not have a user-defined Local Port parameter. (The module assigns one internally.)

3) Exception: It is possible to repeat Local Port values for several Broadcast Receive jobs on the same module. These jobs are differentiated by their Broadcast IDs, which must be unique. The Local Port value(s) used for Broadcast Receive jobs cannot be assigned as the Local Port value for another UDP job on the same module; specifically, you cannot use the same Local Port value for a UDP Server job that you are using for Broadcast Receive jobs on the module.

Maximum Number of Jobs Using Job Start Bits

The module offers 48 Job Start bits, and this means that you can configure a maximum of 48 jobs that use Job Start bits on a module. The following jobs use Job Start bits:

- Send job
- Receive job (if Enable Control Bits check box is selected)
- Write Active job
- Read Active job
- Write Remote job
- Read Remote job
- Memory Exchange job
- E-mail job

Job Start and Job Active Bits

The Job Start bit and Job Active bit are used by certain jobs to "handshake" (synchronize) communication between the PLC program and the SIMATIC 505 TCP/IP module, as shown in Figure 4-2.

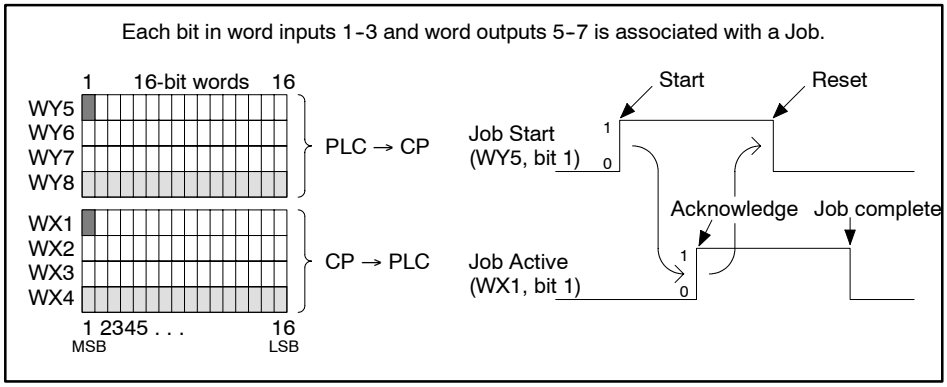


Figure 4-2 Handshaking Communication between PLC and Module

The Job Start bit is used by the PLC to trigger the job, and the Job Active bit is used by the module to indicate that the job is in progress:

- Job Start bit — A user-selected bit in one of the available output words (WY) assigned to the SIMATIC 505 TCP/IP module that is used in the relay ladder logic (RLL) program to trigger a data transfer job. The module must see a 0-to-1 transition on the Job Start bit to begin a data transfer job.

- Job Active bit — A corresponding bit in the input word (WX) set by the SIMATIC 505 TCP/IP module to signal that a requested data transfer job is in progress. When this bit goes back to zero, the PLC knows that the job has been completed and can then initiate another job by setting the corresponding WY bit to one.

Assigning I/O Memory

Each SIMATIC 505 TCP/IP module is registered in the I/O memory map of its PLC as using 4 word inputs (WXs) and 4 word outputs (WYs). You must assign the starting I/O address, as described on page 2-16, in order for the PLC to be able to communicate with the module.

NOTE: The designations WX1, 2, 3, 4 and WY5, 6, 7, 8, shown in the SIMATIC 505-CP1434-TCP Module Configurator software, are relative to the actual starting address you assign in SoftShop (or equivalent) for the slot where the SIMATIC 505 TCP/IP module is installed. For example, if you assign 9 as the starting address of the module in SoftShop, then the addresses WX1, 2, 3, 4, and WY5, 6, 7, 8 in the Configurator are actually WX9, 10, 11, 12 and WY13, 14, 15, 16 in the PLC. Of course, if you configure the starting address as 1, the actual addresses match the relative addresses shown in the Configurator. (See page 2-16.)

Bits in words WY5, WY6, and WY7 (relative addresses) are available as the Job Start bits. Any of the 48 bits in WY5, WY6, and WY7 can be used as the Job Start bit for a job as needed. The corresponding bit in WX1, WX2, or WX3 (the Job Active bit) signals a job in progress.

Example of Synchronizing a Job

The handshaking, or synchronizing, of a job can be summarized as follows:

1. With the WY bit set to 0, the PLC must wait for the corresponding WX bit to become 0.
2. The PLC sets WY bit from 0 to 1 to trigger the job.
3. The module sees the rising edge on the WY bit, sets the corresponding WX bit to 1, and starts the job.

   The module must see the transition of the WY Job Start bit in order to execute the job. Either a reset of the module, or a power cycle of the entire PLC base, can cause the PLC ladder logic to set a WY Job Start bit prior to the module recovery.

⚠ WARNING

If the module does not see the transition of the WY Job Start bit, the module does not execute the job. This could cause unpredictable operation of equipment.

Unpredictable operation of process equipment can cause death or serious injury to personnel, and/or damage to equipment.

To prevent unpredictable operation of the equipment, implement a ladder logic timer (TMR or TMRF) that is enabled when the WY Job Start bit is set and is reset in response to the WX Job Active bit. If the WX Job Active bit never responds to a WY Job Start bit, the timer is allowed to timeout and it clears the current WY Job Start bit. The next execution of the ladder logic starts the WY Job Start bit under normal operating conditions. To allow ample module start-up time, set the timer preset value to 30 seconds. Normal SIMATIC 505 TCP/IP module startup requires approximately 25 seconds.

4. The PLC sees the WX bit go to 1 and then resets WY bit to 0.
5. The module completes the job, stores the job status in PLC V memory, and, when it sees the WY bit reset to 0, resets the WX bit to 0. (The module also resets the WX bit to 0, regardless of the WY state, if the Send, Write Active, or Read Active job times out.)
6. The PLC sees the WX bit return to 0, indicating job is complete; it can check the status word for errors: if none, the PLC knows that data was sent or that it can use any received data.
7. The PLC can start the job again, whenever it is ready, by repeating the sequence from step 2.

4.3 Maximum Number of Jobs per Module

There are 64 job slots available in the module. They are allocated according to the following guidelines:

- The number of job slots taken by a UDP Server job is equal to its Maximum Client parameter. If the module has nothing configured except a UDP Server job, up to 64 UDP simultaneous client transactions are supported; the presence of other jobs in the module reduces the number of available UDP job slots accordingly. Additional simultaneous requests (above the client maximum) are ignored until an existing client finishes the current transaction, thus freeing a job slot. This phenomenon could cause an occasional client request timeout if more clients submit requests simultaneously than are permitted by the "Max Clients" parameter.

- A TCP Server job automatically takes one job slot (before anything connects to it). Each time a client connects to the TCP Server job, another job slot is taken. If the module has nothing configured except a TCP Server job, up to 63 TCP client connections are supported; the presence of other jobs in the module reduces the number of available TCP client connections accordingly. When the TCP client closes the connection, the TCP job slot becomes available for another TCP client connection.

- If you turn on Serial Port Redirect, one job slot is taken.

- The other jobs each take one slot (except for Broadcast Receive, which does not use job slots).

- A maximum of 48 jobs requiring Job Start bits can be configured per module. Page 4-7 lists the jobs that use Job Start bits.

The Configurator does not permit you to download a configuration that would use more than 64 job slots. However, it is possible for other modules to make requests, for instance for connections to a TCP Server job on the module, that the Configurator cannot prevent. In those cases, the module accepts the first 64 requests that are made, and rejects all others until a connection closes and frees a job slot.

4.4 Send and Receive Jobs

Usage

The Send and Receive combination offers point-to-point data transfer using TCP. The Send reads information from its PLC and sends the information to the module specified by the “Remote IP” parameter; the Receive writes the information to its PLC. You can configure the Send job without specifying where the data will be written to on the PLC that is associated with the Receive job. You assign destination memory addresses for that PLC when you configure the Receive job.

NOTE: This job combination has several optional parameters. If you do not specify values for the optional parameters, the module assigns values for them internally. However, the advantage of filling in values for the optional parameters is to prevent unwanted connections in cases where you have more than one of a certain job type on the network. See Section 4.14.

Maximum Number of These Jobs

The Send job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.4, provides the details about job parameters.)

Configuring the Send Job

To configure this job, you must supply the following parameters:

Local TCP Port (optional) Enter a value between 1024 and 65,535. Do not re-use this value for any other job configured on this module (unless the job uses UDP protocol: see Table 4-3). The value that you assign as the Local TCP Port of the Send job corresponds to the value for the Remote TCP Port of the corresponding Receive job. (Local and remote ports are explained on page 4-5.)

NOTE: You can leave the Local TCP Port parameter of the Send job (and the corresponding Remote TCP Port of the Receive job) blank.

When you specify a value for the optional Local TCP Port parameter of the Send job, it does not affect the connection if the optional Remote TCP Port parameter on the Receive job is unspecified. However, when a value is specified for the optional Remote TCP Port parameter on the Receive job, the connection can only succeed if the Local TCP Port parameter of the Send job is also specified (with the same value).

Remote TCP Port Enter a value that matches the Local TCP Port of the corresponding Receive job.

Remote IP Enter the IP Address of the module where the corresponding Receive job is configured.

It is a good practice to print out each configuration file as you configure modules on your network, so that you have a record of the IP Address, the port values assigned to jobs, etc., for each module. If you have no printed record and need to find out the IP Address of a module, you can load its configuration in the Configurator and choose the **Edit > Set Network Parameters** menu command, which causes the Module Network Parameters dialog box to display.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Timeout The range of valid values is 0 to 655 (seconds). A value of zero (0) means that the timeout is disabled. Accept the default, 30 seconds, unless you have a reason to change it. Reducing the value means that the job must complete more quickly or it times out. Increasing the value means that the job has more time to complete (which could affect any logic that you use to retry the job).

Start Bit Assign a single WY word and bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, “6.7” specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Local PLC Data This job can send data from up to 8 different PLC memory locations. The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types, because each data type has its own units, from bits to bytes to words to floats (double words). For instance:

- V memory consists of words, so V100,3 specifies a count of 3 words (6 bytes) of variable memory starting at address V100.
- CP memory consists of packed bits, so CP100,3 specifies a count of 3 bits (packed as 1 byte, with 5 empty bits) of control relay packed memory starting at address CP100.
- C memory consists of bits, so C100,3 specifies a count of 3 bits (sent as 3 bytes) of control relay memory starting at address C100. (Since the bits are not packed, each one is sent as a byte of data, where the byte value is FF if the bit is on and 0 if the bit is off.)
- LMN memory consists of floats, so LMN100,3 specifies a count of 3 floats (12 bytes) of loop output memory starting at address LMN100.

See Appendix C for information about data types that can be used with this module.

Configuring the Receive Job

To configure this job, you must supply the following parameters:

Local TCP Port Enter a value between 1024 and 65,535. Do not reuse this value for any other job configured on this module (unless the job uses UDP protocol: see Table 4-3). The value that you assign as the Local TCP Port of the Receive job corresponds to the value for the Remote TCP Port of the corresponding Send job. (Local and remote ports are explained on page 4-5.)

Remote TCP Port (optional) Enter a value that matches the Local TCP Port of the corresponding Send job.

NOTE: You can leave the Remote TCP Port parameter of the Receive job (and the corresponding Local TCP Port of the Send job) blank.

However, if you specify a value for the optional Remote TCP Port parameter of the Receive job, the connection can only succeed if the Local TCP Port parameter of the Send job is also specified (with the same value).

Remote IP (optional) Enter the IP Address of the module where the corresponding Send job is configured. Specifying a value for this parameter ensures that a connection can only be accepted from the specified module.

Enable Control Bits By default, this selection is off, and data from Send jobs is transferred from the buffer to the PLC as soon as the module receives it. The data, therefore, is received asynchronous to the PLC program. If you select the Enable Control Bits checkbox, you can define a Status Word and Start Bit for the Receive job, and write PLC code to manage the Receive job, to ensure "consistent data."

If you require consistent receive data and you enable control bits for a Receive job, you must ensure that your PLC code handles the Job Start bit and Job Active bit correctly and in a timely manner. Otherwise it is possible for Receive data from a corresponding Send job to deplete the memory buffers of the module, which can cause module operation to "freeze" for a minute and then reset.

⚠ WARNING

If Receive data depletes the memory buffers of the module, the module might "freeze" for a minute and then reset.

If the module freezes when connected to equipment controlling your process, control of the process could be lost, resulting in death or serious injury to personnel, and/or damage to equipment.

If you enable control bits for a Receive job, test your code thoroughly to ensure that it operates as desired before connecting the module to equipment that controls your process.

Status Word (Only available if Enable Control Bits is selected.) Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Start Bit (Only available if Enable Control Bits is selected.) Assign a single WY word and bit that has not been used for any other job on this module. These are relative address assignments (as explained on page 4-9). The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, "6.7" specifies WY6 bit 7. The Job Active bit of the Receive job stays on until either the Send job connects and the Receive job receives the data or else an error or a disconnect occurs: the Job Active bit could be on for a long time.

Local PLC Data This job can receive data from up to 8 different PLC memory locations. The starting addresses, data types, and block sizes in the receiving PLC do not have to match those of the sending PLC, but the total size of the combined blocks must match the total of what you configured for the Send job.

4.5 Write Active and Write Passive Jobs

Usage

The Write Active and Write Passive combination offers point-to-point data transfer using TCP. The Write Active extracts data from its PLC and sends it to the Write Passive job; the Write Passive receives the data and moves it to its PLC. You configure memory assignments for both PLCs within the Write Active job.

NOTE: This job combination has several optional parameters. If you do not specify values for the optional parameters, the module assigns values for them internally. However, the advantage of filling in values for the optional parameters is to prevent unwanted connections in cases where you have more than one of a certain job type on the network. See Section 4.14.

Maximum Number of These Jobs

The Write Active job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.5, provides the details about job parameters.)

Configuring the Write Active Job

To configure this job, you must supply the following parameters:

Local TCP Port (optional) Enter a value between 1024 and 65,535. Do not reuse this value for any other job configured on this module (unless the job uses UDP protocol: see Table 4-3). The value that you assign as the Local TCP Port of the Write Active job corresponds to the value for the Remote TCP Port of the corresponding Write Passive job. (Local and remote ports are explained on page 4-5.)

NOTE: You can leave the Local TCP Port parameter of the Write Active job (and the corresponding Remote TCP Port of the Write Passive job) blank.

When you specify a value for the optional Local TCP Port parameter on the Write Active job, it does not affect the connection if the optional Remote TCP Port parameter on the Write Passive job is unspecified. However, when a value is specified for the optional Remote Parameter on the Write Passive job, the connection can only succeed if the Local TCP Port parameter on the Write Active job is also specified (with the same value).

Remote TCP Port Enter a value that matches the Local TCP Port of the corresponding Write Passive job.

Remote IP Enter the IP Address of the module where the Write Passive job is configured.

It is a good practice to print out each configuration file as you configure modules on your network, so that you have a record of the IP Address, the port values assigned to jobs, etc., for each module. If you have no printed record and need to find out the IP Address of a module, you can load its configuration in the Configurator and choose the **Edit > Set Network Parameters** menu command, which causes the Module Network Parameters dialog box to display.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Timeout The range of valid values is 0 to 655 (seconds). A value of zero (0) means that the timeout is disabled. Accept the default, 30 seconds, unless you have a reason to change it. Reducing the value means that the job must complete more quickly or it times out. Increasing the value means that the job has more time to complete (which could affect any logic that you use to retry the job).

Start Bit Assign a single WY word and bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, "6.7" specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Local PLC Data One block of PLC memory is permitted with this job. The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types, because each data type has its own units, from bits to bytes to words to floats (double words). For instance:

- V memory consists of words, so V100,3 specifies a count of 3 words (6 bytes) of variable memory starting at address V100.
- CP memory consists of packed bits, so CP100,3 specifies a count of 3 bits (packed as 1 byte, with 5 empty bits) of control relay packed memory starting at address CP100.
- C memory consists of bits, so C100,3 specifies a count of 3 bits (sent as 3 bytes) of control relay memory starting at address C100. (Since the bits are not packed, each one is sent as a byte of data, where the byte value is FF if the bit is on and 0 if the bit is off.)
- LMN memory consists of floats, so LMN100,3 specifies a count of 3 floats (12 bytes) of loop output memory starting at address LMN100.

See Appendix C for information about data types that can be used with this module.

Remote PLC Data Choose the option button that describes the remote PLC (SIMATIC 505 or SIMATIC S7). Specify the PLC memory location where the data is to be written. The starting address in the remote PLC does not have to match what you configured for the local PLC, but the size of the block must match what you configured for the local PLC.

Configuring the Write Passive Job

To configure this job, you must supply the following parameters:

Local TCP Port Enter a value between 1024 and 65,535. Do not reuse this value for any other job configured on this module (unless the job uses UDP protocol: see Table 4-3). The value that you assign as the Local TCP Port of the Write Passive job corresponds to the value for the Remote TCP Port of the corresponding Write Active job. (Local and remote ports are explained on page 4-5.)

Remote TCP Port (optional) Enter a value that matches the Local TCP Port of the corresponding Write Active job.

NOTE: You can leave the Remote TCP Port parameter of the Write Passive job (and the corresponding Local TCP Port of the Write Active job) blank.

However, if you specify a value for the optional Remote TCP Port parameter of the Write Passive job, the connection can only succeed if the Local TCP Port parameter of the Write Active job is also specified (with the same value).

Remote IP (optional) Enter the IP Address of the module where the Write Active job is configured. Specifying a value for this parameter ensures that a connection can only be accepted from the specified module. The valid range is 1.0.0.0 to 223.225.255.255.

4.6 Read Active and Read Passive Jobs

Usage

The Read Active and Read Passive combination offers point-to-point data transfer using TCP between two PLCs. The Read Active requests data from the remote PLC; the Read Passive receives the request, reads the data from its PLC, and sends it back to the Read Active job; the Read Active writes the response data to its PLC. You configure memory assignments for both PLCs within the Read Active job.

NOTE: This job combination has several optional parameters. If you do not specify values for the optional parameters, the module assigns values for them internally. However, the advantage of filling in values for the optional parameters is to prevent unwanted connections in cases where you have more than one of a certain job type on the network. See Section 4.14.

Maximum Number of These Jobs

The Read Active job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.6, provides the details about job parameters.)

Configuring the Read Active Job

To configure this job, you must supply the following parameters:

Local TCP Port (optional) Enter a value between 1024 and 65,535. Do not reuse this value for any other job configured on this module (unless the job uses UDP protocol: see Table 4-3). The value that you assign as the Local TCP Port of the Read Active job corresponds to the value for the Remote TCP Port of the corresponding Read Passive job. (Local and remote ports are explained on page 4-5.)

NOTE: You can leave the Local TCP Port parameter of the Read Active job (and the corresponding Remote TCP Port of the Read Passive job) blank.

When you specify a value for the optional Local TCP Port parameter of the Read Active job, it does not affect the connection if the optional Remote TCP Port parameter of the Read Passive job is unspecified. However, when a value is specified for the optional Remote Parameter of the Read Passive job, the connection can only succeed if the Local TCP Port parameter of the Read Active job is also specified (with the same value).

Remote TCP Port Enter a value that matches the Local TCP Port of the corresponding Read Passive job.

Remote IP Enter the IP Address of the module where the Read Passive job is configured.

It is a good practice to print out each configuration file as you configure modules on your network, so that you have a record of the IP Address, the port values assigned to jobs, etc., for each module. If you have no printed record and need to find out the IP Address of a module, you can load its configuration in the Configurator and choose the **Edit > Set Network Parameters** menu command, which causes the Module Network Parameters dialog box to display.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Timeout The range of valid values is 0 to 655 (seconds). A value of zero (0) means that the timeout is disabled. Accept the default, 30 seconds, unless you have a reason to change it. Reducing the value means that the job must complete more quickly or it times out. Increasing the value means that the job has more time to complete (which could affect any logic that you use to retry the job).

Start Bit Assign a single WY word and bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, “6.7” specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Local PLC Data One block of PLC memory is permitted with this job. The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types, because each data type has its own units, from bits to bytes to words to floats (double words). For instance:

- V memory consists of words, so V100,3 specifies a count of 3 words (6 bytes) of variable memory starting at address V100.
- CP memory consists of packed bits, so CP100,3 specifies a count of 3 bits (packed as 1 byte, with 5 empty bits) of control relay packed memory starting at address CP100.
- C memory consists of bits, so C100,3 specifies a count of 3 bits (sent as 3 bytes) of control relay memory starting at address C100. (Since the bits are not packed, each one is sent as a byte of data, where the byte value is FF if the bit is on and 0 if the bit is off.)
- LMN memory consists of floats, so LMN100,3 specifies a count of 3 floats (12 bytes) of loop output memory starting at address LMN100.

See Appendix C for information about data types that can be used with this module.

Remote PLC Data Choose the option button that describes the remote PLC (SIMATIC 505 or SIMATIC S7). Specify the PLC memory location of the data that is to be read. The starting address in the remote PLC does not have to match what you configured for the local PLC, but the size of the block must match what you configured for the local PLC.

Configuring the Read Passive Job

To configure this job, you must supply the following parameters:

Local TCP Port Enter a value between 1024 and 65,535. Do not reuse this value for any other job configured on this module (unless the job uses UDP protocol: see Table 4-3). The value that you assign as the Local TCP Port of the Read Passive job corresponds to the value for the Remote TCP Port of the corresponding Read Active job. (Local and remote ports are explained on page 4-5.)

Remote TCP Port (optional) Enter a value that matches the Local TCP Port of the corresponding Read Active job.

NOTE: You can leave the Remote TCP Port parameter of the Read Passive job (and the corresponding Local TCP Port of the Read Active job) blank.

However, if you specify a value for the optional Remote TCP Port parameter of the Read Passive job, the connection can only succeed if the Local TCP Port parameter of the Read Active job is also specified (with the same value).

Remote IP (optional) Enter the IP Address of the module where the Read Active job is configured. Specifying a value for this parameter ensures that a connection can only be accepted from the specified module.

4.7 TCP Server Jobs

Usage

This job can serve Write Remote, Read Remote, and Memory Exchange jobs. It can also serve Serial Port Redirect requests and raw NITP task codes sent to the module. One TCP Server job can serve multiple clients.

Maximum Number of These Jobs

The maximum number of TCP Server jobs that can be configured per module is 10. However, since a single TCP Server job can serve multiple clients, under ordinary circumstances you would only need to configure one TCP Server job on a module. Only if the traffic generated by TCP clients caused performance issues might you need to configure more than one TCP Server job.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.7, provides the details about job parameters.)

Configuring the TCP Server Job

To configure this job, you must supply the following parameters:

Local Port Accept the default (1505) unless you have already assigned it as the Local Port value of another job that uses TCP protocol. The range of permissible values is 1024 to 65,535. Do not reuse this value for any other job configured on this module (unless the job uses a different Internet protocol: see Table 4-3). The value that you assign as the Local Port of this TCP Server job corresponds to the value for the Remote Port of all the clients that reference the TCP Server job. (Local and remote ports are explained on page 4-5.)

Send Keepalive When Keepalive is enabled, the module tests any TCP Server connections that have been inactive for 7200 seconds (2 hours) in order to identify whether the connection is broken. For example, a connected client might crash or be reset or powered down without informing the server that the connection has been terminated. The Keepalive feature checks for this and closes the connection, thus freeing the job slot so that it can be used by another TCP client.

4.8 UDP Server Jobs

Usage

This job can serve Write Remote, Read Remote, and Memory Exchange jobs. It can also serve Serial Port Redirect requests, raw NITP task codes sent to the module, and the SIMATIC 505 OPC Server for TCP/IP (described in Chapter 5). One UDP Server job can serve multiple clients.

Maximum Number of These Jobs

The maximum number of UDP Server jobs that can be configured per module is 10. However, since a single UDP Server job can serve multiple clients, under ordinary circumstances you would only need to configure one UDP Server job on a module. Only if the traffic generated by UDP clients caused performance issues might you need to configure more than one UDP Server job.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.8, provides the details about job parameters.)

Configuring the UDP Server Job

To configure this job, you must supply the following parameters:

Local Port Accept the default (1505) unless you have already assigned it as the Local Port value of another job that uses UDP protocol. The range of permissible values is 1024 to 65,535. Do not reuse this value for any other job configured on this module (unless the job uses a different Internet protocol: see Table 4-3). The value that you assign as the Local Port of this UDP Server job corresponds to the value for the Remote Port of all the clients that reference the UDP Server job. (Local and remote ports are explained on page 4-5.)

Maximum Clients The range is 1 to 64, with a default of 10. This value determines the maximum number of UDP clients that can simultaneously access this UDP Server job. A high maximum client value reduces the overall number of slots available on the module for other jobs. A value lower than the actual number of clients means that a client may be ignored during high traffic (although the client can retry the timed-out transaction).

4.9 Write Remote Jobs

Usage

The Write Remote and Server job combination offers point-to-point data transfer. The Write Remote job can work with a TCP Server job, a UDP Server job, or a CP2572 module that has been configured with a corresponding server on the remote PLC. The Write Remote reads data from the local PLC and sends it to the Server job on the remote PLC; the Server job writes the data to the remote PLC. You configure memory assignments for both the local PLC and the remote PLC within the Write Remote job.

Maximum Number of These Jobs

The Write Remote job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.9, provides the details about job parameters.)

Configuring the Write Remote Job

To configure this job, you must supply the following parameters:

UDP, TCP Pick the protocol that is best for your application. This selection must correspond to the TCP or UDP Server job that you configure on the remote module.

Remote Port Enter a value that matches the Local Port of the corresponding TCP or UDP Server job on the remote module.

Remote IP Enter the IP Address of the module where the corresponding TCP or UDP Server job is configured.

It is a good practice to print out each configuration file as you configure modules on your network, so that you have a record of the IP Address, the port values assigned to jobs, etc., for each module. If you have no printed record and need to find out the IP Address of a module, you can load its configuration in the Configurator and choose the **Edit > Set Network Parameters** menu command, which causes the Module Network Parameters dialog box to display.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Timeout The range of valid values is 0 to 655 (seconds). A value of zero (0) means that the timeout is disabled. Accept the default, 30 seconds, unless you have a reason to change it. Reducing the value means that the job must complete more quickly or it times out. Increasing the value means that the job has more time to complete (which could affect any logic that you use to retry the job).

Start Bit Assign a single WY word and bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, "6.7" specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Local PLC Data (Source) This job can write one block of PLC memory. The maximum block size is 512 bytes. The block must be an even number of bytes, and it must match the size of the remote PLC data block. The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types, because each data type has its own units, from bits to bytes to words to floats (double words). For instance:

- V memory consists of words, so V100,3 specifies a count of 3 words (6 bytes) of variable memory starting at address V100.
- CP memory consists of packed bits, so CP100,3 specifies a count of 3 bits (packed as 1 byte, with 5 empty bits) of control relay packed memory starting at address CP100. Because the block for a Write Remote job must be an even number of bytes, you need to increase the count to take a second byte (a count between 9 and 16 would accomplish it) even though you are only going to use 3 of the bits.
- C memory consists of bits, so C100,3 specifies a count of 3 bits (actually 3 bytes) of control relay memory starting at address C100. (Since the bits are not packed, each one is sent as a byte of data, where the byte value is FF if the bit is on and 0 if the bit is off.) Because the block for a Write Remote job must be an even number of bytes, you need to increase the count by one (to a count of 4) even though you are only going to use 3 of the bits.
- LMN memory consists of floats, so LMN100,3 specifies a count of 3 floats (12 bytes) of loop output memory starting at address LMN100.

See Appendix C for information about data types that can be used with this module.

Remote PLC Data (Destination) Specify the PLC memory location where the data is to be written. The starting address in the remote PLC does not have to match what you configured for the local PLC, but the size of the block must match what you configured for the local PLC. The maximum block size is 512 bytes, and the block must be an even number of bytes.

4.10 Read Remote Jobs

Usage

The Read Remote and Server job combination offers point-to-point data transfer. The Read Remote job can work with a TCP Server job, a UDP Server job, or a CP2572 module that has been configured with appropriate PLC logic to act as a server. The Read Remote job requests data from the remote PLC; the Server job moves the data from the remote PLC to the local module; the Read Remote job then places the data in the designated location on the local PLC. You configure memory assignments for both the local PLC and the remote PLC within the Read Remote job.

Maximum Number of These Jobs

The Read Remote job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.10, provides the details about job parameters.)

Configuring the Read Remote Job

To configure this job, you must supply the following parameters:

UDP, TCP Pick the protocol that is best for your application. This selection must correspond to the TCP Server or UDP Server job that you configure on the remote module to serve this Read Remote job.

Remote Port Enter a value that matches the Local Port of the corresponding TCP or UDP Server job on the remote module.

Remote IP Enter the IP Address of the module where the corresponding TCP or UDP Server job is configured.

It is a good practice to print out each configuration file as you configure modules on your network, so that you have a record of the IP Address, the port values assigned to jobs, etc., for each module. If you have no printed record and need to find out the IP Address of a module, you can load its configuration in the Configurator and choose the **Edit > Set Network Parameters** menu command, which causes the Module Network Parameters dialog box to display.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Timeout The range of valid values is 0 to 655 (seconds). A value of zero (0) means that the timeout is disabled. Accept the default, 30 seconds, unless you have a reason to change it. Reducing the value means that the job must complete more quickly or it times out. Increasing the value means that the job has more time to complete (which could affect any logic that you use to retry the job).

Start Bit Assign a single WY word and bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, "6.7" specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Local PLC Data (Destination) This job can read one block of PLC memory. The maximum block size is 512 bytes. The block must be an even number of bytes, and it must match the size of the remote PLC data block. The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types, because each data type has its own units, from bits to bytes to words to floats (double words). For instance:

- V memory consists of words, so V100,3 specifies a count of 3 words (6 bytes) of variable memory starting at address V100.
- CP memory consists of packed bits, so CP100,3 specifies a count of 3 bits (packed as 1 byte, with 5 empty bits) of control relay packed memory starting at address CP100. Because the block for a Read Remote job must be an even number of bytes, you need to increase the count to take a second byte (a count between 9 and 16 would accomplish it) even though you are only going to use 3 of the bits.
- C memory consists of bits, so C100,3 specifies a count of 3 bits (actually 3 bytes) of control relay memory starting at address C100. (Since the bits are not packed, each one is sent as a byte of data, where the byte value is FF if the bit is on and 0 if the bit is off.) Because the block for a Read Remote job must be an even number of bytes, you need to increase the count by one (to a count of 4) even though you are only going to use 3 of the bits.
- LMN memory consists of floats, so LMN100,3 specifies a count of 3 floats (12 bytes) of loop output memory starting at address LMN100.

See Appendix C for information about data types that can be used with this module.

Remote PLC Data (Source) Specify the PLC memory location of the data that is to be read. The starting address in the remote PLC does not have to match what you configured for the local PLC, but the size of the block must match what you configured for the local PLC. The maximum block size is 512 bytes, and the block must be an even number of bytes.

4.11 Memory Exchange

Usage

The Memory Exchange and Server job combination offers point-to-point data transfer and combines the Read Remote and Write Remote into one job. The Memory Exchange job can work with a TCP Server job, a UDP Server job, or a CP2572 module that has been configured with appropriate PLC logic to act as a server. Memory Exchange is the only job that offers both write and read operations. You configure memory assignments for both the local PLC and the remote PLC within the Memory Exchange job.

Maximum Number of These Jobs

The Memory Exchange job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.11, provides the details about job parameters.)

Configuring the Memory Exchange Job

To configure this job, you must supply the following parameters:

UDP, TCP Pick the protocol that is best for your application. This selection must correspond to the TCP Server or UDP Server job that you configure on the remote module to serve this Memory Exchange job.

Remote Port Enter a value that matches the Local Port of the corresponding TCP or UDP Server job on the remote module.

Remote IP Enter the IP Address of the module where the corresponding TCP or UDP Server job is configured.

It is a good practice to print out each configuration file as you configure modules on your network, so that you have a record of the IP Address, the port values assigned to jobs, etc., for each module. If you have no printed record and need to find out the IP Address of a module, you can load its configuration in the Configurator and choose the **Edit > Set Network Parameters** menu command, which causes the Module Network Parameters dialog box to display.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Timeout The range of valid values is 0 to 655 (seconds). A value of zero (0) means that the timeout is disabled. Accept the default, 30 seconds, unless you have a reason to change it. Reducing the value means that the job must complete more quickly or it times out. Increasing the value means that the job has more time to complete (which could affect any logic that you use to retry the job).

Start Bit Assign a single WY word and bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 to 16. For example, "6.7" specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Local PLC Data

Source This block of PLC memory contains data that is read from the local PLC and sent to a destination on the remote PLC. The length of the Local PLC Data Source and Remote PLC Data Destination must match exactly, be limited to a maximum of 508 bytes, and be an even number of bytes.

Destination This block of PLC memory is written with data that is received from a source on the remote PLC. The length of the Local PLC Data Destination and the Remote PLC Data Source must match exactly, be limited to a maximum of 512 bytes, and be an even number of bytes.

The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types, because each data type has its own units, from bits to bytes to words to floats (double words). For instance:

- V memory consists of words, so V100,3 specifies a count of 3 words (6 bytes) of variable memory starting at address V100.
- CP memory consists of packed bits, so CP100,3 specifies a count of 3 bits (packed as 1 byte, with 5 empty bits) of control relay packed memory starting at address CP100. Because the block for a Memory Exchange job must be an even number of bytes, you need to increase the count to take a second byte (a count between 9 and 16 would accomplish it) even though you are only going to use 3 of the bits.
- C memory consists of bits, so C100,3 specifies a count of 3 bits (actually 3 bytes) of control relay memory starting at address C100. (Since the bits are not packed, each one is sent as a byte of data, where the byte value is FF if the bit is on and 0 if the bit is off.) Because the block for a Memory Exchange job must be an even number of bytes, you need to increase the count by one (to a count of 4) even though you are only going to use 3 of the bits.
- LMN memory consists of floats, so LMN100,3 specifies a count of 3 floats (12 bytes) of loop output memory starting at address LMN100.

See Appendix C for information about data types that can be used with this module.

Remote PLC Data

NOTE: You must use the same memory type for the Destination block and the Source block of the Remote PLC Data. There is no such restriction for the Source and Destination blocks on the local PLC.

Destination The data read from the source on the local PLC is written to this destination block on the remote PLC. The starting address and type can differ from what you configured for the Local PLC Data Source block, but the block size must correspond. The length of the Remote PLC Data Destination must match the Local PLC Data Source exactly, be limited to a maximum of 508 bytes, and be an even number of bytes.

Source The data read from this block of remote PLC memory is written to the destination on the local PLC. The value of the starting address cannot exceed 65,535. The starting address can differ from what you configured for the Local PLC Data Destination, but the block size must correspond. The length of the Remote PLC Data Source must match the Local PLC Data Destination exactly, be limited to a maximum of 512 bytes, and be an even number of bytes.

4.12 Broadcast Send and Broadcast Receive Jobs

Usage

The Broadcast jobs function as a team. You configure one Broadcast Send job for multiple Broadcast Receive jobs. You use this job combination when you want to communicate the same information to multiple devices simultaneously, using UDP.

The Broadcast Send job is self-starting, once configured; it does not need a Job Start bit and you do not have to write PLC code to trigger it.

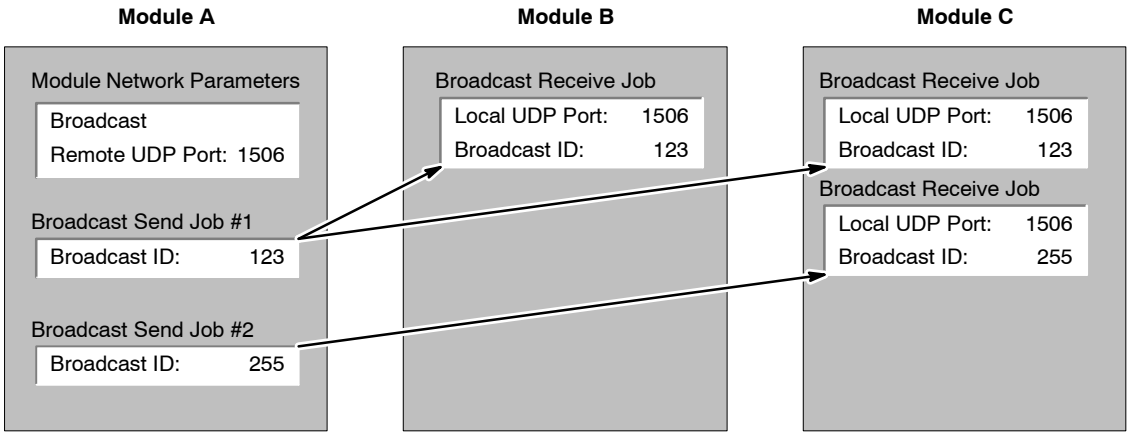
NOTE: Broadcast Send jobs do not run when a PLC is in Program mode.

Broadcast Send jobs only use report-by-exception (RBE) variables, which automatically update whenever the value of a variable changes. When the value of an RBE variable changes, the changed value is placed in the empty broadcast data buffer (which can hold 1472 bytes consisting of RBE variables and their overhead) and the Delivery Time timer starts. Any further changes to RBE variables accumulate in the broadcast data buffer. If the buffer fills up before the Delivery Time value is reached, the buffer data is broadcast at that point. Otherwise, the changed RBE data is not broadcast until the Delivery Time timer expires. (You configure the Delivery Time value in the Module Network Parameters, as described on page 3-7.)

In order to use Broadcast Send and Broadcast Receive, you must have valid Broadcast parameters in the Module Network Parameters dialog box (described on page 3-4). The Broadcast Send jobs use the Broadcast Remote UDP Port that you configure in Module Network Parameters as a destination identifier; they send broadcasts on that port and only jobs listening on that port can receive the broadcasts. Therefore, all the Broadcast Receive jobs that want to listen to a Broadcast Send must have the same value for their Local UDP Port as the Broadcast Remote UDP Port of the module where the Broadcast Send resides.

Furthermore, since you can configure more than one module with the same Broadcast Remote UDP Port, each Broadcast Send job is uniquely identified by its Broadcast ID. Network-wide, you cannot repeat a Broadcast ID for any Broadcast Send jobs that use the same Broadcast Remote UDP Port. (You can reuse a Broadcast ID for another Broadcast Send job only if the two Broadcast Send jobs have different Broadcast Remote UDP Ports.) The combination of Broadcast Remote UDP Port and Broadcast ID makes each Broadcast Send job unique network-wide.

Figure 4-3 shows an example of how to configure parameters for multiple Broadcast Send and Broadcast Receive jobs.



The Broadcast Remote UDP UDP Port of a module is an internal identifier for all the Broadcast Send jobs configured on that module.

There are two Broadcast Send jobs on Module A. The Broadcast Remote UDP UDP Port value of 1506 is used (internally) by both jobs as a destination identifier. So all the Broadcast Receive jobs that are configured for either one of the Broadcast Send jobs must use 1506 as their Local UDP Port value. Furthermore, you could reuse the same Broadcast Remote UDP Port value on other modules, and then all of the Broadcast Send jobs on those modules would also use 1506 as their destination identifier, and all the corresponding Broadcast Receive jobs would use 1506 as their Local UDP Port value.

The combination of Broadcast Remote UDP Port and Broadcast ID is what uniquely identifies a Broadcast Send job and differentiates it from other Broadcast Send jobs that are configured on the same network.

Since both of the Broadcast Send jobs on Module A share the same Broadcast Remote UDP Port, they are further differentiated by their Broadcast ID. Module B and Module C both have Broadcast Receive jobs configured to listen to Broadcast Send Job #1 (which has a Broadcast ID of 123). Module C also has a second Broadcast Receive job, which is configured to listen to Broadcast Send Job #2 (which has a Broadcast ID of 255). The Broadcast Receive jobs that are configured to match a Broadcast Send job must use the Broadcast Remote UDP Port value as their Local UDP Port, and the Broadcast ID as their Broadcast ID.

Figure 4-3 Broadcast Parameters

Maximum Number of These Jobs

The maximum number of Broadcast Send jobs that can be configured on a module is 4. The maximum number of Broadcast Receive jobs that can be configured on a module is 64.

Network-wide, you can configure a maximum of 256 different Broadcast Send jobs on modules that use the same Broadcast Remote UDP Port. This is because each Broadcast Send job that uses the same Broadcast Remote UDP Port must be identified with a unique Broadcast ID (from 0 to 255).

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.12, provides the details about job parameters.)

Configuring the Broadcast Send Job

To configure this job, you must supply the following parameters:

Local PLC Data This job can send data from up to 8 different PLC memory regions (in other words, 8 different blocks). Only the following RBE data types are allowed in Broadcast Send jobs: AVF, CP, DCP, DSC, DSP, G, GA to GZ, K, KF, LRSF, LS, LVF, STW, TCC, TCP, V, VF, WX, WY, XP, YP.

The format is An,x where A is the data type, n is the starting address, and x is the count. The same count value may signify different sizes for different data types. For instance, V100,3 signifies a count of three words. The block size of V100,3 is six bytes. By contrast, CP100,3 signifies a count of three bits. Since the bits are packed, the block size of CP100,3 is one byte.

For Broadcast Send jobs, the maximum size permitted per block is 110 bytes. The maximum item count per block is 272.

NOTE: Word data types cannot exceed an item count of 55 (because that equals the maximum block size of 110 bytes). Packed data types cannot exceed a block size of 34 bytes (because that equals the maximum item count of 272 packed bits).

When you define your data blocks, be aware that only the first sixteen bits (first word) in each of the 8 possible data blocks are monitored for RBE updates. If any of these monitored bits changes, all the data in this block is placed in the broadcast send buffer. However, any remaining data (after the first sixteen bits) in the block is considered “auxiliary”—it is not monitored for changes, and thus no Broadcast Send would occur if it changed.

For example, in the block V100,3, only the first word, V100, is monitored for change. But in the block CP100,3, all three bits, CP100, CP101, and CP102, (corresponding to C100, C101, and C102 in the PLC) are monitored.

For the Broadcast Send job, the amount of data permitted in a block exceeds the amount of data that is actually monitored for RBE. You can use the blocks of Local PLC Data in different ways:

- Use the unmonitored data (the remaining 108 “auxiliary” bytes) in the block like a snapshot of a data table. When any of the 16 monitored bits changes, the entire block is sent, so the monitored bits act as triggers that cause an update of the data table.
- Restrict the size of the data you place in each block to 16 bits or less, ensuring that a change to any data element in that block causes the entire block to be sent. Since each module can have up to four Broadcast Send jobs, with eight data blocks per Broadcast Send job, 32 separate words of data can be monitored by each module.

Max. Interval (Seconds) The range of valid values is 0 to 655 (seconds). Accept the default (a blank field or a value of zero means no forced sends) unless you have a reason to change it.

With a Max. Interval value of zero, Broadcast Sends are only issued when monitored data in one of the data blocks changes. If you assign a non-zero value to the Max. Interval, you are forcing all Local PLC Data to be placed in the broadcast data buffer at the stipulated interval, even if no monitored data has changed.

The Delivery Time timer only takes effect if there is an RBE data change. But if a prolonged period goes by with no RBE data changes, you may want to force a broadcast of all the Local PLC Data in the Broadcast Send job: that is what the Max. Interval timer is for. If a Max. Interval value is configured, the Local PLC Data is automatically placed in the broadcast data buffer every time that the Max. Interval is reached (regardless if there have been many broadcasts or no broadcasts during a given interval). The Local PLC Data that accumulates in the broadcast data buffer is sent only when one of the following conditions is met:

- The buffer becomes full.
- The Delivery Time timer (which starts when the first item is placed in the buffer) reaches the Delivery Time value.

After the broadcast data buffer is sent, the Delivery Time timer is reset to zero. The Delivery Time timer does not resume timing until new data is placed into the emptied broadcast data buffer. However, the Max. Interval timer is independent of the Delivery Time timer. The Max. Interval timer automatically resets when the Max. Interval time value is reached, on a continuous basis.

Typically, if the Max. Interval is used, it is set to a significantly larger value than the Delivery Time parameter. (You configure the Delivery Time value in the Module Network Parameters, as described on page 3-7.)

Broadcast ID The valid range is 0 to 255. The default is the last three digits of the module IP address. However, it is possible to have up to four Broadcast Send jobs on a module, and each job must have a unique Broadcast ID. Therefore, you cannot use the default on more than one Broadcast Send job per module. Furthermore, if you configure Broadcast Sends on multiple modules and those modules use the same Broadcast Remote UDP Port (for example, each module uses 1506), all of the Broadcast Sends must use unique Broadcast IDs.

Configuring the Broadcast Receive Job

To configure this job, you must supply the following parameters:

Local UDP Port Enter a value that matches the Broadcast Remote UDP Port of the module where the Broadcast Send is configured. The valid range is 1024 to 65535. This value must not match the value of any other UDP job in the module configuration. (To find out what the Broadcast Remote UDP Port of the sending module is: Upload or open the stored configuration file for the module where the Broadcast Send was configured. Choose the **Edit > Set Network Parameters** menu command to see what the Broadcast Remote UDP Port value is.)

Local PLC Data This job can receive data in up to 8 different PLC memory locations. The starting addresses and data types of the receiving blocks do not have to match those of the corresponding blocks in the Broadcast Send job. However, the size of each receiving block must match the size of each corresponding block that you have configured in the Broadcast Send job. (Since the Broadcast Send job may be in a different configuration from the Broadcast Receive job(s), it is up to the user to ensure that this requirement is met.) The maximum size permitted per block is 110 bytes.

NOTE: Although Broadcast Send jobs can only transmit data types that are RBE-capable, Broadcast Receive jobs can store the data that they receive in any type of PLC memory, as long as the block size matches the corresponding Broadcast Send block.

Broadcast ID Enter the Broadcast ID of the corresponding Broadcast Send job. (The valid range is 0 to 255.)

See Figure 4-3 (above) for an example of how to configure multiple Broadcast Send and Broadcast Receive jobs on a network.

Rules for Broadcast Blocks

The following rules govern the transmission of data in blocks from Broadcast Send jobs to Broadcast Receive jobs:

- The data types and starting addresses of blocks used by the Broadcast Receive jobs do not have to match those of the blocks used by the Broadcast Send job. The individual block sizes must match.
- Each block in the Broadcast Send job has an internal identifier that associates it with the corresponding block on the Broadcast Receive job(s). When the monitored data (first 16 bits) in a given block changes, the updated block is sent to the corresponding block on the Broadcast Receive job(s). (For instance, if the monitored data (first 16 bits) in block three of the Broadcast Send changes, the entire block (block three) is sent to block three of every corresponding Broadcast Receive job.)
- If a Broadcast Receive job is configured with fewer blocks than the Broadcast Send job, the extra blocks (at the end of the Broadcast Send job) are ignored by that Broadcast Receive job. (Other Broadcast Receive jobs may have been configured to use all the blocks sent.)
- If a Broadcast Receive job is configured with more blocks than the Broadcast Send job, the extra blocks on that Broadcast Receive job are not affected by the Broadcast Send job, and so their contents are indeterminate.

4.13 E-mail Jobs

Usage

The E-mail job allows you to configure a text message that is sent, using TCP, to a Simple Mail Transfer Protocol (SMTP) e-mail server when triggered by a Job Start bit. The e-mail message can also include variables: for information on how to encode a variable in e-mail, see Appendix D.

The SIMATIC 505 TCP/IP module configuration only stores the E-mail job parameters. The actual e-mail message text resides in the PLC and is not a part of the module configuration; however, the Configurator provides a convenient method for entering the e-mail message text into the PLC.

The connection mode of the Configurator affects the prerequisites necessary for e-mail configuration:

- If the Configurator is connected over the network to the module, you need to configure a TCP Server job and download it to the module before writing e-mail message data from the Configurator to the PLC. (The TCP Server job does not need to remain in the module after you successfully write the e-mail data to the PLC.) See Section 4.7 for information about TCP Server jobs.
- If the Configurator has a serial connection to the module, you need to configure a TCP Server job, enable Serial Port Redirect in the module network parameters, set the redirect address to the module IP address (or Local, which has the same effect), and download this configuration to the module before writing e-mail message data from the Configurator to the PLC. (The TCP Server job and Serial Port Redirect do not need to remain in the module after you successfully write the e-mail data to the PLC.) See Section 3.11 for information about Serial Port Redirect.
- If the Configurator is connected directly to the serial port of the PLC, no TCP Server job or Serial Port Redirect is necessary in the module. However, after you write the e-mail message data to the PLC, you need to define the parameters of the e-mail job in the Configurator, reconnect the Configurator to the module, and download the configuration to the module.

Maximum Number of These Jobs

The E-mail job is one of several types of jobs that use a Job Start bit. The maximum number of jobs using a Job Start bit that can be configured (per module) is 48.

Access

To access the configuration dialog box, choose the **Edit > Add Job** menu command. (Page 3-9 gives an overview of the basic procedure for configuring jobs. This section, 4.13, provides the details about job parameters.)

Sending Mail Messages

The easiest way to configure and send an E-mail job from the SIMATIC 505 TCP/IP module is as follows:

- Create an E-mail job and enter the SMTP Server IP address, the TCP Port of the SMTP server, the Status Word for the job, and the Start Bit for the job.
- Use the Message button to open an editor where you can define a Start Address for your e-mail message and enter text into the PLC for all the data blocks (MAIL FROM, RCPT TO, and MESSAGE) of the message.
- After your message is complete, use the Calculate button to automatically determine the size of your message blocks as well as the next available memory address after the e-mail data. (When you confirm your work in the message editor, the e-mail Job Configuration dialog box is automatically configured with the memory locations and counts of the MAIL FROM, RCPT TO, and MESSAGE data blocks.)

NOTE: The Calculate button lets you see if there is a potential addressing conflict so that you can change your Start Address if necessary before writing the e-mail data blocks (MAIL FROM, RCPT TO, and MESSAGE) to the PLC. If you click on Write PLC without using the Calculate button first, the Write PLC operation calculates and displays the block sizes, but it is not possible for you to correct any address conflicts because the e-mail data blocks are immediately written to the PLC and they overwrite any existing data in the PLC.

- Use the Write PLC button to load the MAIL FROM, RCPT TO, and MESSAGE data blocks into PLC memory. (Confirm your work in the message editor, and confirm the E-mail job.)
- Download the configuration with the E-mail job to the module.

NOTE: It is not sufficient to write the e-mail data blocks to the PLC. You must also download the e-mail job parameters from the configuration to the module, or else the module cannot send an e-mail containing the data that you have written to the PLC.

- Write PLC code to trigger the Job Start bit.
- Run your PLC program and test the E-mail job.

You can use the Read PLC button (in the message editor) to upload and display e-mail message blocks that have already been defined in the PLC.

Configuring the E-mail Job

To configure this job, you must supply the following parameters:

IP Address Enter the IP address of your SMTP server. The valid range is 1.0.0.0 to 223.255.255.255. See your system administrator if you do not know what value to use.

TCP Port Use the default of 25 for the remote port of the SMTP server unless your system administrator tells you otherwise. The valid range is 1 to 65535.

Status Word Assign a single word location based on the available V-memory of your PLC. The status word location receives a numeric code when the job ends. You can write PLC logic to check the status word. A value of zero indicates that a job was successful; Appendix B lists error codes for conditions that cause a job to fail.

Start Bit Assign a single WY bit that has not been used for any other job on this module. The range of the word is 5 to 7 and the range of the bit is 1 (most significant bit) to 16. For example, "6.7" specifies WY6 bit 7. (These are relative address assignments, as explained on page 4-9.)

Message Button Click this button to open a message editor where you can build your actual e-mail message and write the message data down to the PLC, and also read up any e-mail message data that has already been configured in the PLC. See Using the Message Editor on page 4-42.

NOTE: If you have already entered your e-mail data in the PLC, you still must use the message editor to read it up from the PLC in order to successfully configure the e-mail job in the module. The Configurator must read the PLC data in order to calculate the sizes of the MAIL FROM, RCPT TO, and MESSAGE data blocks, as well as the Start Address overhead, and that configuration must then be downloaded to the module.

Start Address (Optional) You can enter the start address of the e-mail message data in this field, or enter it in the message editor.

Mail From (Read-only) After you write your e-mail data to the PLC or read it from the PLC using the message editor of the Configurator, and click on the OK button in the message editor, this field displays the V- or K-memory location where the data for the MAIL FROM block of the e-mail message is stored in the PLC. The format is An,x where A is the data type, n is the starting address, and x is the count (in words). For example, V100,20 specifies 20 units (words) of V-memory beginning at address V100. Maximum block size permitted: 464 words.

NOTE: There are 5 words of overhead at the beginning of an e-mail job, so the Mail From block always begins 5 words after the Start Address of the job.

Rcpt To (Read-only) After you write your e-mail data to the PLC or read it from the PLC using the message editor of the Configurator, and click on the OK button in the message editor, this field displays the V- or K-memory location where the data for the RCPT TO block of the e-mail message is stored in the PLC. The format is An,x where A is the data type, n is the starting address, and x is the count (in words). For example, V1000,51 specifies 51 units (words) of V-memory beginning at address V1000. Maximum block size permitted: 16,383 words.

Message (Read-only) After you write your e-mail data to the PLC or read it from the PLC using the message editor of the Configurator, and click on the OK button in the message editor, this field displays the V- or K-memory location where the data for the MESSAGE block of the e-mail message is stored in the PLC. The format is An,x where A is the data type, n is the starting address, and x is the count (in words). For example, V10000,199 specifies 199 units (words) of V-memory beginning at address V10000. Maximum block size permitted: 32,767 words.

Using the Message Editor

To access the message editor, open an e-mail job and click the Message button in the Job Configuration dialog box. This opens the Build E-mail Message for the PLC dialog box.

You do not have to use the message editor of the Configurator to load your e-mail data blocks into the PLC; you can enter them directly. However, the message editor automatically calculates block sizes for you and automates some of the data entry.

Furthermore, if you do load the e-mail blocks into the PLC yourself, you still need to read the data from the PLC to the message editor of the Configurator in order to configure an e-mail job (which requires the block sizes and addresses of the MAIL FROM, RCPT TO, and MESSAGE data blocks).

- If you have not yet written the e-mail message data to the PLC, you can use the message editor to build the e-mail message, and your block sizes and address information are automatically transferred to the e-mail Job Configuration dialog box.
- If you have already placed the message text in the PLC but have not yet configured an e-mail job, you must still use the message editor. You cannot manually enter the message data block locations and sizes into the job configuration fields. The information must be read from the PLC with the Read PLC function.

Building an E-mail Message

Follow these steps to build and download an e-mail message to your PLC:

1. (Optional) The host name of the module, as defined in module network parameters, is the default in the Mail From field. You can edit the field to supply a user-defined Mail From identifier instead if desired.

2. Edit the e-mail address string in the Rcpt To field to identify a recipient of the e-mail. Use the Add Rcpt button to add more RCPT TO statements. (Each RCPT TO statement must begin on a separate line.)

   The recipients listed in the RCPT TO block can be identified later as “To” or “CC” addressees in the Header section of the MESSAGE block. To create the effect of a blind carbon copy, you can include a recipient in the RCPT TO block but not list the recipient as an addressee later in your message header.

3. To add a Header item, select an item from the Header drop-down list box, click on the Add Header button, and edit the item as necessary in the Header Text box. Repeat as desired for other Header statements. (Each Header item must begin on a separate line.)

4. Enter the text of your message in the E-mail Text box.

5. Enter a start address (V- or K- memory) for the e-mail message data.

NOTE: There are 5 words of overhead at the beginning of an e-mail job, so the Mail From block always begins 5 words after the Start Address of the job.

---

6. Click on the Calculate button to see the addresses that will be used in the PLC for your MAIL FROM, RCPT TO, and MESSAGE data blocks.

   The next available address in the PLC is also displayed. If any of the addresses required for the e-mail data have already been assigned in the PLC, change the Start Address of your e-mail message data and click the Calculate button again. Do not proceed until you are sure that there will be no addressing conflict. It is best to leave a gap after the e-mail message data, in case it is revised later and its size increases.

7. Click on the Write PLC button to write your e-mail message data blocks to the specified addresses in the PLC.

NOTE: You are only writing the message data to the PLC. Your e-mail job cannot take effect until you complete the configuration and download the configuration to the module. This is because the e-mail job parameters (Job Start bit, data block counts and Start Address, etc.) must be transferred to the module.

---

8. Click on the OK button to confirm your work and close the Build E-mail Message for the PLC dialog box. The Start Address and block size information are automatically transferred to the Job Configuration dialog box.

9. Click on the OK button to confirm your work and close the Job Configuration dialog box.

You are now finished configuring the e-mail job. For information about downloading your configuration to the module, see Section 3.7.

NOTE: If you place message text in the PLC directly, without using the message editor of the Configurator, you still have to use the Read PLC function in the message editor before you can successfully configure an E-mail job to use the message data. The block addresses and sizes are required parameters of the E-mail job, and you cannot manually enter the addresses and sizes into the job configuration fields. The information must be read from the PLC with the Read PLC function.

Reading a Previously Configured E-mail Message

To access the message editor, open an e-mail job and click the Message button in the Job Configuration dialog box. This opens the Build E-mail Message for the PLC dialog box. Follow these steps to read a previously configured e-mail message up to the module from your PLC:

1. Enter the start address (V- or K- memory) of the e-mail message that you have already configured in the PLC.
2. Click on the Read PLC button to read your e-mail message data blocks starting at the specified address in the PLC.
3. Click on the OK button to confirm your work and close the Build E-mail Message for the PLC dialog box. The Start Address and block size information are automatically transferred to the Job Configuration dialog box.
4. Click on the OK button to confirm your work and close the Job Configuration dialog box.

To complete your configuration of the e-mail job, edit the other fields of the e-mail Job Configuration dialog box as described on page 4-40. After you configure the job, you must download your configuration to the module. For information about downloading, see Section 3.7.

4.14 Using the Optional Parameters to Avoid Competing Jobs

Optional Parameters

Certain job combinations (Send/Receive, Write Active/Write Passive, and Read Active/Read Passive) offer optional parameters. The rationale for making certain parameters optional is to provide leeway in cases where the person configuring one job is not certain about some of the parameters that were used by the corresponding job. However, while that leeway allows you to avoid specifying the wrong parameter, it also introduces the possibility of unwanted connections. The more fully you specify parameters for a job combination, the more likely it is that the jobs will connect with each other as intended.

It may be useful, before configuring jobs, to consider the possible outcomes when optional parameters are left blank.

Jobs Competing for a Connection

In each job pair, one (the Send, Write Active, or Read Active job) may be termed the "Active job," and the corresponding job (Receive, Write Passive, or Read Passive job) termed "Passive job." As long as there is only one of a job combination (one Send, one Receive, for instance) on a network, there is no possibility of unwanted connections. However, when you configure several of the same job pairs on the network (two or more Sends, two or more Receives), then unspecified (blank) parameters could allow unwanted connections.

Competing Passive Jobs

There is no true competition if you mistakenly configure the optional Remote IP and Remote Port parameters of two Passive jobs in such a fashion that they both expect a connection with the same Active job. The Active job must specify the Remote IP and Remote Port parameters of the job it wants to connect to—those parameters are mandatory. Therefore, the Active job can only connect to one of the two Passive jobs. The other Passive job never receives a connection.

Competing Active Jobs

By contrast, if you mistakenly configure the mandatory Remote IP and Remote Port parameters of two or more Active jobs in such a fashion that they both expect to connect to the same Passive job, there are a number of possible outcomes. There is a real likelihood of unwanted connections. The outcome is influenced by whether the competing Active jobs reside on a single module or separate modules, and by how many, if any, of the optional parameters in the Passive job have been specified.

Table 4-4 illustrates the possible outcomes if two or more Active jobs are configured to connect to the same Passive job.

Table 4-4 Competing Active Jobs

Parameters Specified by the Passive Job	Possible Scenarios with the Active Jobs	
	Active jobs are on same module. (Note 1)	**Active jobs are on different modules.** (Note 2)
Passive job has specified the Remote IP and Remote Port.	Assuming that these Active jobs reside on the correct module, the first Active job that is defined with the correct Local Port can establish a connection. Active jobs with the wrong parameters, or with no Local Port specified, are denied.	The Active job that resides on the correct module can establish a connection only if it has the correct Local Port defined. Active jobs with the wrong parameters, or with no Local Port specified, are denied.
Passive job has specified the Remote IP only.	Assuming that the Active jobs are on the specified module, the first Active job to attempt connection succeeds, and all others are denied.	The first Active job that resides on the correct module can establish a connection, whether the optional Local Port of the Active job has been defined or not. All Active jobs from modules that do not match the specified IP are denied.
Passive job has specified the Remote Port only.	Only one of the Active jobs on this module can possibly possess a Local Port that corresponds to the Remote Port specified (and hence establish a connection). Active jobs with the wrong Local Port or with no Local Port specified are denied.	Since the Active jobs are on different modules, it is possible for their Local Ports to be the same. If both Active jobs have defined a Local Port that corresponds to the Remote Port specified, the first Active job to attempt connection succeeds and all others are denied. Active jobs with the wrong Local Port or with no Local Port specified are denied.
Passive job has specified neither the Remote IP nor the Remote Port.	The first Active job to attempt connection succeeds, and all others are denied.	The first Active job to attempt connection succeeds, and all others are denied.

Notes:

1) The Active jobs share the same IP address, because they are on the same module. Consequently their Local Port values (whether user-defined or assigned by the module) must be different.

2) The Active jobs have different IP addresses, because they are on different modules. It is possible for their Local Port values (whether user-defined or assigned by the modules) to be the same.

Chapter 5

Using the Optional OPC Server for TCP/IP

5.1 Overview

The optional SIMATIC 505 OPC Server for TCP/IP (PPX:505-6701; site license PPX:505-6702) lets you configure variables so that you can use an OPC client product, such as SIMATIC WinCC (version 5.0 SP2 or higher), to monitor variables that originate from a Series 505 PLC. The SIMATIC 505 OPC server is COM/DCOM-compliant, so it can be accessed by OPC clients on local or remote PCs. For information about DCOM (Distributed Component Object Model), see the Microsoft web site:

http://www.microsoft.com/com/tech/dcom.asp

When data is cached in the SIMATIC 505 OPC server, the OPC client does not have to query the PLC and can obtain it more quickly. Furthermore, the SIMATIC 505 OPC server supports report-by-exception (RBE) variables for the SIMATIC 505 TCP/IP module, which offer the fastest possible update.

Supported Devices

The SIMATIC 505 OPC server can obtain PLC data several different ways:

- From a network connection to a supported module. The 505-CP1434-TCP and the 505-CP2572 are supported.
- From a direct serial connection to a supported PLC. The following PLCs are supported:

Model Number	Release
545-1102	3.0 or higher
545-1103	3.1 or higher
545-1104	4.0 or higher
545-1105	5.0 or higher
545-1106	5.0 or higher
555-1101	3.0 or higher
555-1102	3.0 or higher
555-1103	4.0 or higher
555-1104	4.0 or higher
555-1105	5.0 or higher
555-1106	5.0 or higher

All releases of these PLCs that are at or above the specified release level are supported by the SIMATIC 505 OPC server. Earlier releases have slow updates and the XP, YP, CP, LS, LM, G, GA to GZ, VMM, and VMS data types are not supported.

The most productive way to use the SIMATIC 505 OPC server is with a network connection to the 505-CP1434-TCP (the SIMATIC 505 TCP/IP module). The other connections obtain PLC data by polling, whereas with a network connection to the SIMATIC 505 TCP/IP module, the SIMATIC 505 OPC server obtains the PLC data through RBE updates, which are more efficient and which decrease network traffic and increase communication speed in a large installation.

RBE Variables

You use the SIMATIC 505 OPC server to define variables from your PLC(s). Then your OPC client can access that data from the SIMATIC 505 OPC server. RBE variables offer the fastest possible updates. The SIMATIC 505 OPC server supports the following RBE variables from the SIMATIC 505 TCP/IP module:

AVF, CP, DCP, DSC, DSP, G, GA to GZ, K, KL, KF, LRSF, LS, LVF, STW, TCC, TCP, V, VL, VF, WX, WY, XP, YP

These data types are described in more detail in Appendix C. The maximum number of RBE variables supported is system dependent, and varies based upon the model of PLC, the installed programs, and the installed I/O modules.

Using the TCP/IP Module with the OPC Server

In order to take advantage of RBE updates, you need to use a SIMATIC 505 TCP/IP module and follow these guidelines:

- Configure a SIMATIC 505 TCP/IP module with network Broadcast parameters (see page 3-6) and a UDP Server job to send data to the SIMATIC 505 OPC server.
- Configure the SIMATIC 505 OPC server to connect over the network to the PLC through the SIMATIC 505 TCP/IP module.
- Use the OPC client to access the data from a saved OPC server configuration file, with the SIMATIC 505 OPC server closed, rather than using an open instance of the SIMATIC 505 OPC server. (This causes the server configuration file to be freshly written to the PLC(s), ensuring that all RBE variable identifiers are up to date.)

More about OPC

More information about the OPC specification, a non-proprietary technical specification that defines a set of standard interfaces based on Microsoft's OLE/COM technology, is available on the Internet at www.opcfoundation.org.

5.2 Using the SIMATIC 505 OPC Server

The SIMATIC 505 OPC server is simple to use. Figure 5-1 shows the process:

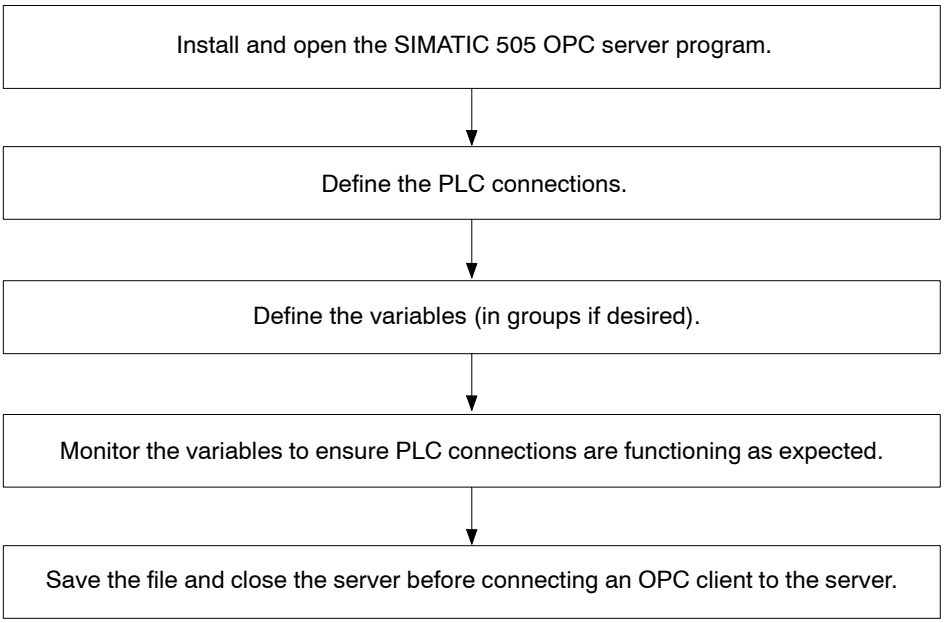


Figure 5-1 Getting Started with the OPC Server

Installing the Server Software

After you install the floppy disk that contains the SIMATIC 505 OPC server software, click the Windows Start button, choose the **Run** command, and enter the following path:

X:\SetupOPC.exe

where "X" represents the letter of the appropriate disk drive on your PC.

Opening the Server

To open the SIMATIC 505 OPC server, click the Windows Start button and choose the **Programs > 505 OPC Server > 505 OPC Server** command (assuming that you installed the software in the default location).

Figure 5-2 shows an example of a SIMATIC 505 OPC server configuration.

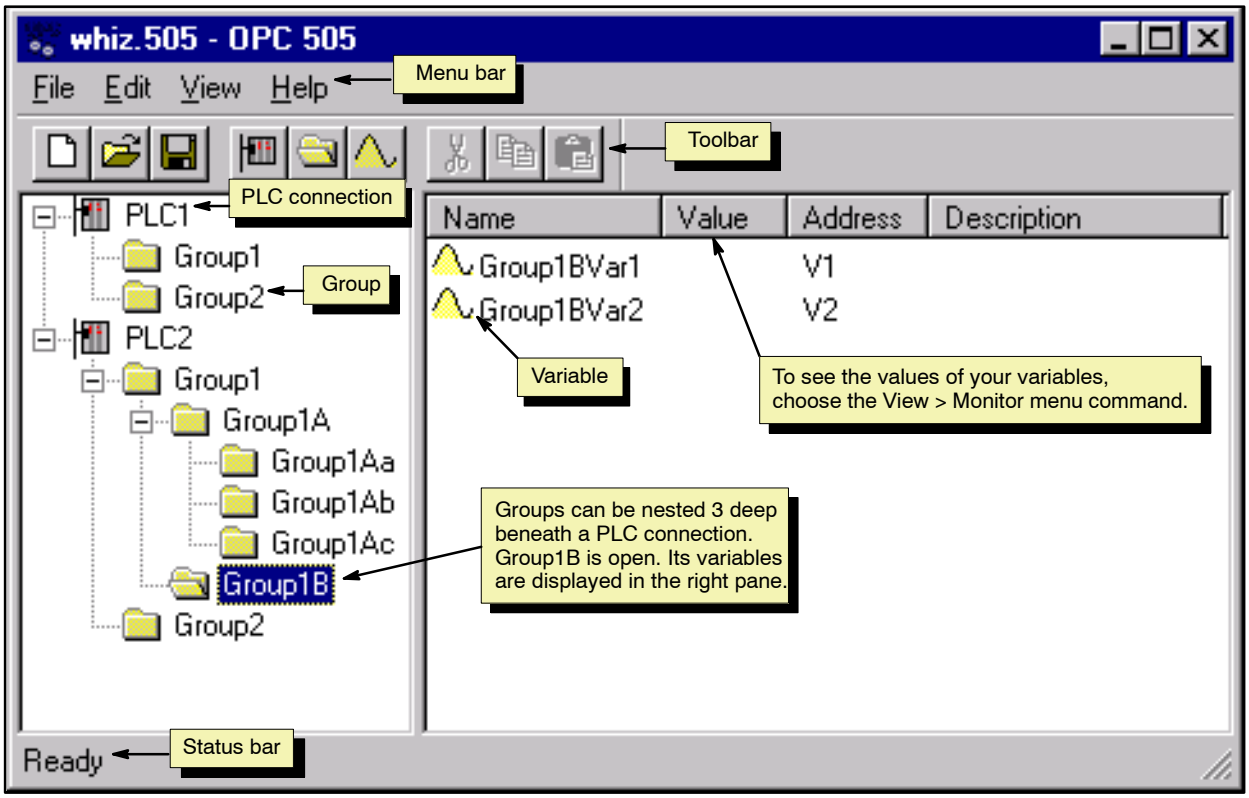


Figure 5-2 Example of an OPC Server File

Defining a PLC Connection

To define a PLC connection, you must supply a unique name for the connection, and specify whether you are connecting over the network (by means of a supported module) or directly to the PLC (serial communication).

NOTE: You can only use RBE (fastest possible updates) if you communicate over the network using the SIMATIC 505 TCP/IP module.

To define a PLC connection, follow these steps:

1. To access the Connection dialog box, choose the **Edit > New PLC Connection** menu command.

2. In the Name field, enter an alphanumeric string to identify this PLC connection.

3. Fill out the appropriate parameters that define how the SIMATIC 505 OPC server is connected:

   - **Network Connection** Click this option button if the PC running the SIMATIC 505 OPC server application is connected to a supported module. Then fill out the Module Address and Module Port fields as described below:

     Module Address Enter the IP Address of the module. (Page 3-5 describes how to configure or view the IP Address of the module.)

     Module Port Enter the Local Port value of the UDP Server job on the module. (A UDP Server job must be configured on the SIMATIC 505 TCP/IP module in order for you to run the SIMATIC 505 OPC server with the module. The default value for UDP Server jobs is 1505.)

   - **Serial Connection** If the PC running the SIMATIC 505 OPC server application is connected directly to the PLC or module, click this option button and then click the option button (PC Port 1 through PC Port 4) that corresponds to the serial port that is used on the PC end of the connection.

4. Accept the default in the Timeout field unless you have a reason to change it.

   The timeout default is 3 seconds. Increase it only if you are experiencing frequent communications failures. (With a slower PC, e.g., 90 MHz, you might need a larger timeout, e.g., 32 seconds.) Decrease it only if you are experiencing noticeable communications lag time; for instance, if you have PLCs with scan times lower than 50 ms.

5. Click the OK button to confirm your work and close the dialog box.

After you create one or more PLC connections, you can define groups and variables.

NOTE: When you configure your PLC(s), in addition to configuring the I/O, you should specify a rate of 8 task codes/scan for the PLC.

Defining Groups

A group is an optional, logical (not physical) collection of variables. Grouping allows you to display related variables together and separate them from other variables or groups. (Variables do not have to be associated with groups. Variables can be associated directly with a given PLC connection.)

You can nest groups a maximum of three levels deep beneath a given PLC connection. In order to add a group, you must first have defined at least one PLC connection.

To add a group, follow these steps:

1. Click the PLC connection or existing group (in the left pane of the SIMATIC 505 OPC server) under which you want to place the group.

2. Choose the **Edit > New Group** menu command.

3. In the Name field, enter an alphanumeric string to identify this group.

4. Click the OK button to confirm your work and close the Group dialog box.

After you define a group, you can add variables to it.

Defining Variables

Variables can be associated directly with a PLC connection, or they can be associated with a group that belongs to a PLC connection.

To add a variable, follow these steps:

1. Click the PLC connection or group (in the left pane of the SIMATIC 505 OPC server) under which you want to place the variable.
2. Choose the **Edit > New Variable** menu command.
3. In the Name field, enter an alphanumeric string to identify this variable.
4. In the PLC Address area, choose a memory type from the drop-down list, and type a starting address in the field beside it.

NOTE: The size of the variable depends on its data type. All variables are one unit of memory, but different data types use different units. For instance, V-memory uses words, but VF memory uses double words, so a variable defined as starting at V100 would occupy V100 (only), whereas a variable defined as starting at VF100 would occupy V100 and V101.

See page 5-3 of this manual for a list of the data types that are RBE capable. Data types are also discussed in Appendix C of this manual, and in the programming reference manual for your PLC.

5. (optional) In the Description field, enter an application-specific description of the variable.

NOTE: If you configure an invalid address for any variable, then performance for all SIMATIC 505 server variable updates can be significantly decreased. To prevent such a situation, verify that all the variable addresses are valid for a given PLC application.

Monitoring Variables

After you have defined at least one variable, you can verify that the PLC connection operates correctly by monitoring the variable as its value updates in the right pane of the SIMATIC 505 OPC server. Follow these steps:

1. Click the PLC or group that is associated with the variable(s) that you want to monitor.
2. Choose the **View > Monitor** menu command.

NOTE: Most commands are not available when in Monitor mode.

Troubleshooting Invalid Variable Values

Here are some tips for troubleshooting if the SIMATIC 505 OPC server or the OPC client displays unexpected/invalid values:

- The following codes indicate a calculation or range error:

  "NAN" (Not A Number) The value is invalid, typically because of a round-off error or division by zero.

  "+INF" The calculation produced a larger number than can be represented in single-precision IEEE format.

  "-INF" The calculation produced a smaller number than can be represented in single-precision IEEE format.

- If the Value column in the SIMATIC 505 OPC server does not display valid values, check the PLC Connection parameters. A network connection to the SIMATIC 505 TCP/IP module must use the correct module IP address and module port value (Local Port of the UDP Server job on the module). A serial connection to the PLC must specify the correct PC port. Also check connections and cabling.

- The "Invalid Value" message indicates that a variable address is not valid. The address may not have been configured for your PLC. Invalid variable addresses are likely to cause a significant slowdown in performance for the SIMATIC 505 OPC server. To prevent such a situation, verify that all the variable addresses are valid for a given PLC application.

- If the OPC client is not displaying expected values, check the values in the SIMATIC 505 OPC server. If the SIMATIC 505 OPC server shows the expected values, then the OPC client is probably not configured properly.

- If the OPC server seems to update slowly, one possible explanation is that up-to-date identifiers for the RBE variables in the configuration have not been written from the configuration file to the PLC. In order to ensure that RBE identifiers in the PLC are up to date, use one of these strategies:

  –After saving your configuration file, close the SIMATIC 505 OPC server and reboot the server computer before connecting to the OPC client. This ensures that the PLC reads the configuration file from disk (not from an open instance of the SIMATIC 505 OPC server) and therefore all RBE variable identifiers are updated in the PLC.

  –If the SIMATIC 505 OPC server is open, use the **File > Open** command to open (or re-open) the configuration file. This causes the SIMATIC 505 OPC server to read the configuration file from disk, and therefore all RBE variable identifiers in the PLC are updated. (However, it does not update OPC client connections. You must reboot to update OPC client connections.)

Saving Your Work

Before you exit the SIMATIC 505 OPC server, you need to save your work in a configuration file (*.505 extension).

Choose the **File > Save** menu command to save your work before you exit.

Typically, you would only have one configuration file for each instance of the SIMATIC 505 OPC server (unless it becomes necessary to create a temporary file for troubleshooting purposes). Running multiple active instances of the SIMATIC 505 OPC server that contain connections to the same PLC can cause unpredictable fluctuation in the values reported by each SIMATIC 505 OPC server to its OPC client. This is because each SIMATIC 505 OPC server configuration file assigns unique identifiers to the RBE variables in the PLC; if two or more configuration files use connections to the same PLC, they are liable to continually overwrite one another's assignments and might read values from the wrong variables

⚠ WARNING

Running multiple active instances of the SIMATIC 505 OPC server that contain connections to the same PLC can cause unpredictable fluctuation in the values reported by each SIMATIC 505 OPC server to its OPC client.

Any machinery affected by this unpredictable data fluctuation could perform erratically, which could result in death or serious injury to personnel, and/or damage to equipment.

If you run multiple active instances of the SIMATIC 505 OPC server that contain connections to the same PLC, ensure that each instance of the SIMATIC 505 OPC server uses an identical configuration file. Likewise, identical configuration files must also be used if a PLC has several SIMATIC 505 TCP/IP modules connected to separate instances of the SIMATIC 505 OPC server. The configuration files must be identical because although the SIMATIC 505 OPC server configuration files are connecting through different modules, they are connecting to the same PLC.

Connecting OPC Clients

In order to obtain RBE updates from the PLC, you need to ensure that the SIMATIC 505 OPC server is closed before you connect the OPC client to it.

NOTE: The unique RBE identifiers are only written to the variables in the PLC when the OPC server configuration file is opened. Connecting the OPC client to a closed instance of the SIMATIC 505 OPC server automatically ensures that the configuration file is reopened and the RBE identifiers are written to the PLC.

For guidelines about revising a configuration after you establish connections from OPC clients to the SIMATIC 505 OPC server, see Section 5.3.

5.3 Editing Your File

Revising an Existing File

The techniques for editing a PLC connection, group, or variable are the same whether you are working in a new file or revising an existing one. However, if you want to revise an existing file, you must ensure that any OPC clients connected to the server configuration are updated with your changes, and that any connected PLCs are updated with changes that you make to RBE variables.

⚠ WARNING

It is possible for an OPC client to connect to a different instance of the SIMATIC 505 OPC server than the instance that you are editing. In this case, the OPC client does not receive the editing changes, and so its operation is different from what you intended.

Unpredictable operation of software connected to process equipment can cause death or serious injury to personnel, and/or damage to equipment.

To prevent unpredictable operation of the software, reboot the PC where the SIMATIC 505 OPC server resides after making editing changes.

The SIMATIC 505 OPC server requires you to disconnect any currently connected OPC clients before you make changes to a server configuration. However, certain clients, such as WinCC, automatically attempt to re-establish connection if the connection is terminated. For instance, if you open the SIMATIC 505 OPC server and respond Yes to the prompt, “An open client connection exists. Do you want to close the connection?”, the SIMATIC OPC server terminates the connection, but WinCC immediately opens a new instance of the OPC server. This happens at the Windows level and may not be visible from the user interface. (You can see it from the Windows Task Manager, Processes tab.) Consequently, while you are editing the configuration in one instance of the SIMATIC 505 OPC server, the OPC client has re-connected to a second instance, and it does not receive your editing changes.

Updating by Rebooting

To ensure that all connected OPC clients are updated with editing changes, reboot after you edit. Follow the sequence described below:

1. Open the SIMATIC 505 OPC server.
2. Edit your configuration.
3. Save the configuration.
4. Exit the SIMATIC 505 OPC server.
5. Reboot the PC where the SIMATIC 505 OPC server resides.
6. If your OPC clients do not automatically reconnect (WinCC automatically reconnects), you must manually reconnect them.

In addition to updating OPC clients, this procedure also ensures that the PLCs connected to the SIMATIC 505 OPC server are updated with any changes that you have made to RBE variables.

Editing Tips

To edit a PLC connection, group, or variable in the SIMATIC 505 OPC server, keep the following tips in mind:

- The **Edit > Properties** menu command lets you edit the properties of the PLC connection, group, or variable that is currently selected.
- You cannot cut, copy, or paste a PLC connection; you can only delete it.
- You cannot cut, copy, or paste a group; you can only delete it.
- To select more than one variable from a group for cutting, copying, or deletion, hold down the <CTRL> key while you select the variables. (You cannot select variables from more than one level in the SIMATIC 505 OPC server hierarchy at the same time.)
- To paste variables, click the PLC connection or group under which you want to place the variables.

- To overwrite or duplicate one variable with another that you have previously cut or copied, click the PLC connection or group that contains the variable and issue the Paste command. You are prompted to choose from the following options:

  Rename Lets you specify a new name so that the variable in the paste buffer is pasted as a duplicate of a variable that is present.

  Overwrite Lets you overwrite the selected variable with the variable that is in the paste buffer.

  Skip When you have more than one variable in the paste buffer, lets you skip the current variable but continue the paste operation.

  Cancel Cancels the paste operation for any variables remaining in the paste buffer.

NOTE: The editing commands are not available from Monitor mode.

5.4 Moving the SIMATIC 505 OPC Server from Installed Location

When you install the SIMATIC 505 OPC server, it is automatically entered in the Windows registry. If you want to move it to a different location after installation, you need to update the registry entry to reflect the new location.

For example, if you moved the OPC505.exe file to the d:\temp directory, you would use the following procedure to update the registry entry for the new location:

1. Click the Windows Start button and choose the **Start > Run** menu command. The Run dialog box opens.

2. Enter the following command to update the registry with the new location of the OPC505 file:

   “d:\temp\OPC505” /regserver

NOTE: The path to the file must be fully specified and enclosed in quotations. The /regserver parameter causes the registry to be updated with the new location.

---

3. Click the OK button in the Run dialog box to run the OPC505 executable and update the registry.

Chapter 6

Examples: Send and Receive Jobs

6.1 Overview

This chapter describes how to configure Send and Receive jobs. The process differs depending on whether both jobs are configured on SIMATIC 505 TCP/IP modules, or one job is configured on a SIMATIC S7 CP, such as the CP 343-1 Multi-protocol, CP 343-1 IT, CP 443-1 Multi-protocol, or CP 443-1 IT.

For the purpose of these examples, it is assumed that you have already performed the following tasks:

- Installed the SIMATIC 505 TCP/IP module in a Series 505 base
- Connected the module to a PC running the SIMATIC 505-CP1434-TCP Module Configurator
- Powered up the module and checked LEDs
- Configured I/O addresses for the module in the Series 505 PLC

Likewise, for a SIMATIC S7 CP, it is assumed that you have already performed the following tasks:

- Mounted the S7 PLC onto a DIN rail and attached the CP
- Connected the CP to a PC running STEP 7
- Powered up the S7 PLC and checked LEDs

6.2 Sending between Two SIMATIC 505 Modules

In this example, the Send job is configured on Module A, the Receive job is configured on Module B, and both modules are SIMATIC 505 TCP/IP modules. It does not matter which job or module is configured first. For the purpose of this example, the Send job on Module A is shown first. Network parameters must also be configured for both modules.

NOTE: Before proceeding with this example, consult with your network administrator to ensure that the IP Address and Local Ethernet (MAC) Address values used in the example do not cause conflicts on your network. If necessary, your network administrator can suggest substitute values for the example.

Table 6-1 shows the important parameters of both modules.

Table 6-1 505 Module Parameters

Parameters	SEND Module A (505)	RECEIVE Module B (505)
Local Ethernet (MAC) Address	08 00 06 01 01 01	08 00 06 01 01 02
IP Address	161.218.124.222	161.218.124.223
Local TCP Port (of job on module)	5556	5555
Status Word	V100 (Note 1)	V100 (Note 1)
Job Start Bit	5.1 (Note 1)	5.1 (Note 1)
Job Active Bit	1.1	1.1
Local PLC Data	K1, 50 CP1, 9 V250, 50 X100, 1 XP1, 3	K51, 50 V10, 1 K1, 50 C200, 1 CP10, 3 (Note 2)

Notes:

1) It does not matter whether you use the same Status Word values on Module A and Module B, since they are using the memory of different PLCs. Likewise, the Job Start bits of the Send job on Module A and the Receive job on Module B can be the same, as long as the two modules are connected to different PLCs.

2) The starting addresses of blocks in the sending PLC do not have to match the starting address of blocks in the receiving PLC. Nor do the memory types have to match. The gross size of the Local PLC Data blocks must match the gross size of the Remote PLC Data blocks (the gross size of each is 204 bytes in this example). Different data types use different units, and this can result in different counts. For instance, two bytes of CP memory (CP1, 9) can fit into one word of V memory (V10, 1). See Section C.3 for more information about matching gross data block sizes (“byte balancing”) when data is transferred between a local and a remote PLC.

Configure Network Parameters for Module A

To configure network parameters for the module, follow these steps:

1. Open the SIMATIC 505-CP1434-TCP Module Configurator software application.
2. From the menu bar, choose **Edit > Set Network Parameters** to bring up the Module Network Parameters dialog box.
3. In the Local Ethernet (MAC) Address field, enter the following value:

   080006010101

4. Ignore the Enable Redirect check box. For the purpose of this example, it makes no difference whether the box is selected.
5. In the Network IP Address field, enter the following value:

   161.218.124.222

6. In the Subnet Mask field, enter the following value:

   255.255.0.0

7. Leave the Gateway field empty. For the purpose of this example, it is assumed that you are communicating on the local area network only and do not need a gateway (router) value.
8. Edit the Host Name field to differentiate your first SIMATIC 505 TCP/IP module from your second one. Enter the following value:

   cp1434-tcpModuleA

   The host name parameter is used by e-mail jobs, not send or receive jobs, but it is good practice to have unique host names for each module: for identification, and in case e-mail jobs are configured on the module(s) later.

9. Leave the Domain Name field empty.
10. Accept the default Broadcast parameters. (If you decided to add Broadcast jobs later, you could edit Module Network Parameters and change the Broadcast settings at that point.)
11. Click the OK button to confirm your work and close the Module Network Parameters dialog box.

Configure the Send Job on Module A

To configure the Send job, follow these steps:

1. From the menu bar of the Configurator, choose the **Edit > Add Job** command. The Add Job Selection dialog box opens.
2. In the Job Name field, type the following:

   Send to Module B
3. From the Job Type drop-down list box, choose Send.
4. Click the Configure button. The Job Configuration dialog box opens.
5. In the Local TCP Port field, type the following:

   5556
6. In the Remote TCP Port field, type the following:

   5555
7. In the Remote IP field, type the following:

   161.218.124.223
8. In the Status Word field, type the following:

   100
9. Accept the default Timeout value of 30.
10. In the Start Bit field, type the following:

    5.1
11. In the Local PLC Data field, type the following:

    K1, 50
    CP1, 9
    V250, 50
    X100, 1
    XP1, 3
12. Click the OK button to confirm your work and close the Job Configuration dialog box.
13. Click the OK button to confirm your work and close the Add Job Selection dialog box.

Download the Configuration to Module A

To download your configuration to Module A, follow these steps:

1. From the menu bar of the Configurator, choose the **Transfer > To Module** command.

2. Click the Start button.

   If there is already a configuration in the module, and the name of the configuration differs from the file name that you are transferring, you are prompted to decide whether to overwrite it at this time and continue with the example procedure, or cancel the download to protect the existing module configuration. (You could upload the "real" module configuration and save it as a file, then download the "example" configuration temporarily, restoring the "real" configuration later.)

   When the download is complete, the Transfer Configuration to Module dialog box automatically closes. You can now save the example configuration to a file on your PC if desired.

Connect the Configurator to the Second Module

The remaining work is done on Module B. If your Configurator is using a serial connection to the module, you must physically disconnect the cable from Module A and connect to Module B. (See Section 2.4 for information about connecting the Configurator to the serial port of the module.) If you have already configured network parameters for Module B (as described below) and your Configurator is using a network connection to the modules, you must edit your Network Selection in the Connection Mode dialog box to change the IP address from Module A to Module B. (See Section 3.10 for information about network connections.)

Configure Network Parameters for Module B

To configure network parameters for the module, follow these steps:

1. Open a configuration file for Module B in the Configurator.

2. From the menu bar, choose **Edit > Set Network Parameters** to bring up the Module Network Parameters dialog box.

3. In the Local Ethernet (MAC) Address field, enter the following value:

   080006010102

4. Ignore the Enable Redirect check box. For the purpose of this example, it makes no difference whether the box is selected.

5. In the Network IP Address field, enter the following value:

   161.218.124.223

6. In the Subnet Mask field, enter the following value:

   255.255.0.0

7. Leave the Gateway field empty. For the purpose of this example, it is assumed that you are communicating on the local area network only and do not need a gateway (router) value.

8. Edit the Host Name field to differentiate your second SIMATIC 505 TCP/IP module from your first one. The host name parameter is used by e-mail jobs, not send or receive jobs, but it is good practice to have unique host names for each module: for identification, and in case e-mail jobs are configured on the module(s) later. Enter the following value:

   cp1434–tcpModuleB

9. Leave the Domain Name field empty.

10. Accept the default Broadcast parameters. (If you decided to add Broadcast jobs later, you could edit Module Network Parameters and change the Broadcast settings at that point.)

11. Click the OK button to confirm your work and close the Module Network Parameters dialog box.

Configure the Receive Job on Module B

To configure the Receive job, follow these steps:

1. From the menu bar of the Configurator, choose the **Edit > Add Job** command. The Add Job Selection dialog box opens.

2. In the Job Name field, type the following:

   Rcv from Module A

3. From the Job Type drop-down list box, choose Receive.

4. Click the Configure button. The Job Configuration dialog box opens.

5. In the Local TCP Port field, type the following:

   5555

6. In the Remote TCP Port field, type the following:

   5556

7. In the Remote IP field, type the following:

   161.218.124.222

8. In the Status Word field, type the following:

   100

   It does not matter whether you use the same Status Word values on Module A and Module B, since they are using the memory of different PLCs.

9. Accept the default Timeout value of 30.

10. In the Start Bit field, type the following:

    5.1

    It does not matter whether you use the same Start Bit values on Module A and Module B, since they are using the memory of different PLCs.

11. In the Local PLC Data field, type the following:

    K51, 50
    V10, 1
    K1, 50
    C200, 1
    CP10, 3

    The starting addresses of blocks in the sending PLC do not have to match the starting address of blocks in the receiving PLC. Nor do the memory types have to match. The sizes of the blocks must match. Since different data types use different units, this could result in different counts in order to achieve matching block sizes: for instance, two bytes of CP memory (CP1, 9) can fit into one word of V memory (V10, 1).

12. Click the OK button to confirm your work and close the Job Configuration dialog box.

13. Click the OK button to confirm your work and close the Add Job Selection dialog box.

Download the Configuration to Module B

To download your configuration to Module B, follow these steps:

1. From the menu bar of the Configurator, choose the **Transfer > To Module** command.

2. Click the Start button.

   If there is already a configuration in the module, and the name of the configuration differs from the file name that you are transferring, you are prompted to decide whether to overwrite it at this time and continue with the example procedure, or cancel the download to protect the existing module configuration. (You could upload the “real” module configuration and save it as a file, then download the “example” configuration temporarily, restoring the “real” configuration later.)

   When the download is complete, the Transfer Configuration to Module dialog box automatically closes. You can now save the example configuration to a file on your PC if desired.

At this point, you are ready to run and monitor the operation of your program. You can connect the module(s) to the Ethernet using AUI cables or a transceiver with 10BaseT cables (see Section 2.5).

RLL Code for Send Job at PLC A

The Relay Ladder Logic (RLL) example in Figure 6-1 demonstrates a typical scenario for enabling a Send request. The WX and WY locations used in this RLL example correspond to module I/O addresses that have been configured in the memory of PLC A using the configuration function of a tool like SoftShop.

Table 6-2 explains the program logic that is shown in Figure 6-1:

Table 6-2 Program Logic

C98	Enables condition logic defined by the application (X1) which must remain active until job completion.
MOVW1	Sets status word V100 to a positive value (1) prior to each transaction. This positive value is used for job completion check in CMP1.
BITP3/BITP4	Assures that no pending active Send jobs exist prior to starting a Send job (defined for Bit 1 of the SIMATIC 505 TCP/IP module, located at I/O address WY5).
BITC3	Clears the Job Start bit defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 after the Job Active bit WX1 is set.
BITS2	Sets the Job Start bit defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 to invoke a data transfer function.
CMP1	Tests the completion status of the Send data transfer by the same SIMATIC 505 TCP/IP module, placed in V100, against a zero constant stored at location V15. When the Send job has completed, the value of status word V100 is either zero if the transaction was good, or else a non-zero value (see Table B-2) if the transaction was bad. You must take appropriate action in the USER ERROR CONTROL LOGIC ladder rung at this time. If the Send job is to be executed only one time, you take must take action to disable X1 after the Status word changes value from its preset condition.
TMR2 BITC4	The parallel net of C41 and C43 act together to start and hold TMR2 on until the SIMATIC 505 TCP/IP module recognizes the Job Start bit WY5, 1 has been set and responds by setting the Job Active bit WX1, 1 or TMR2 time expires. TMR2 and BITC4 are used to clear an unrecognized Job Start bit WY5, 1. Either a reset of the module or a power cycle of the entire base may result in the programmable logic controller ladder setting a WY Job Start bit prior to module recovery. When this condition occurs the module will not execute the job.

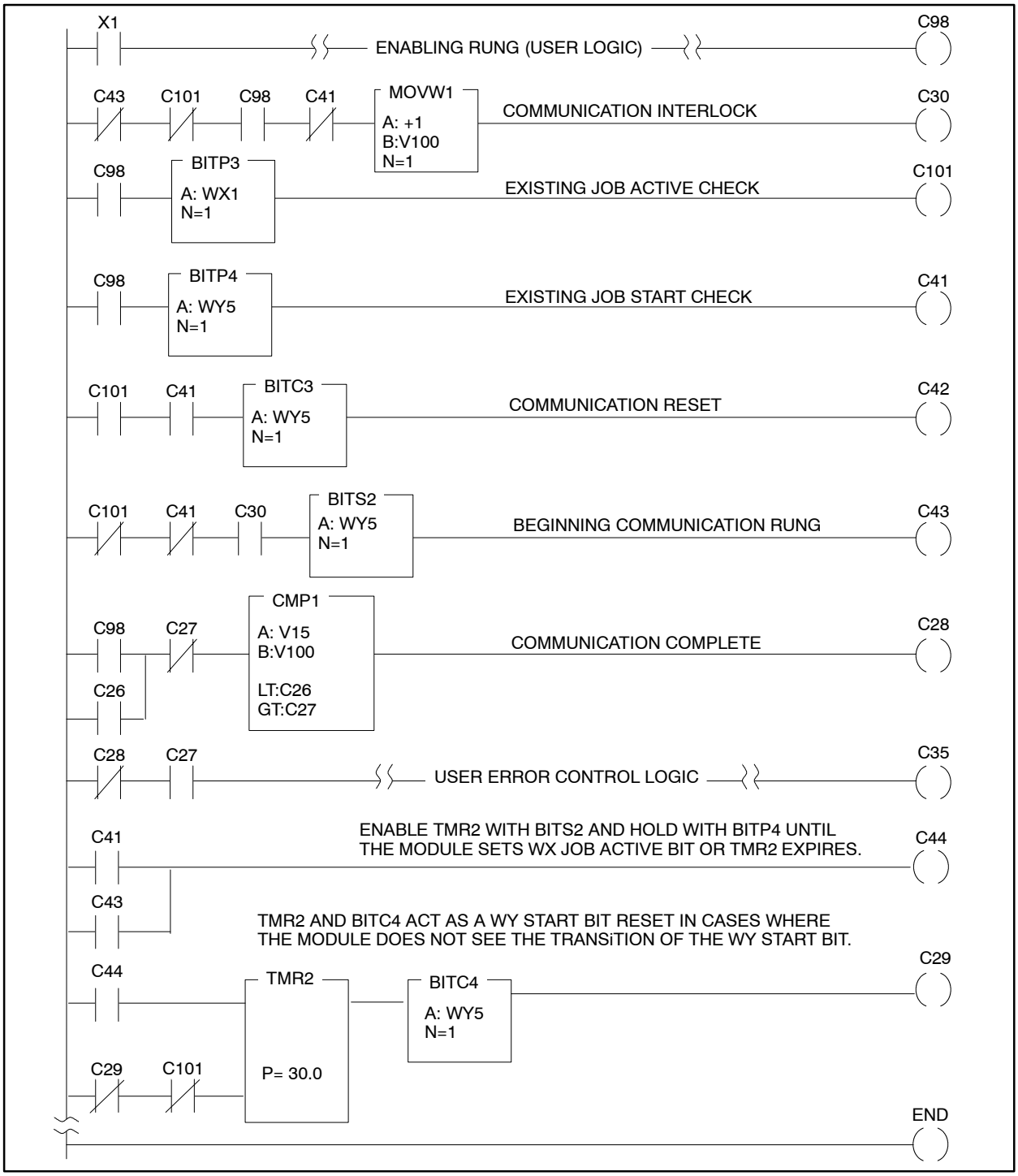


Figure 6-1 Send RLL for PLC A

RLL Code for Receive Job at PLC B

The RLL example in Figure 6-2 demonstrates a typical scenario for enabling a Receive acknowledgement. The WX and WY locations used in this RLL example correspond to module I/O addresses that have been configured in the memory of PLC B using the configuration function of a tool like SoftShop.

Table 6-3 explains the program logic that is shown in Figure 6-2:

Table 6-3 Program Logic

C49	Enables condition logic defined by the application (X2) which must remain active until job completion.
MOVW1	Sets status word V100 to a positive value (1) prior to each transaction. This positive value is used for job completion check in CMP1.
BITP1/BITP2	Assures that no pending active Receive jobs exist prior to starting a Receive job defined for Bit 1 of the SIMATIC 505 TCP/IP module, located at I/O address WY5.
BITC1	Clears the Receive job defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 after the Receive Job Active bit WX1 is set.
BITS1	Sets the Job Start Bit defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 to invoke a data transfer function.
CMP1	Tests the completion status of the Receive data transfer by the same SIMATIC 505 TCP/IP module, placed in V100, against a zero constant stored at location V25. When the Receive job has completed, the value of status word V100 is either zero if the transaction was good, or else a non-zero value (see Table B-2) if the transaction was bad. You must take appropriate action in the USER ERROR CONTROL LOGIC ladder rung at this time. If the Receive job is to be executed only one time, you take must take action to disable X2 after the Status word changes value from its preset condition.
TMR1 BITC2	The parallel net of C31 and C33 act together to start and hold TMR1 on until the module recognizes the Job Start bit WY5, 1 has been set and responds by setting the Job Active bit WX1, 1 or TMR1 time expires. TMR1 and BITC2 are used to clear an unrecognized Job Start bit WY5, 1. Either a reset of the module or a power cycle of the entire base may result in the programmable logic controller ladder setting a WY Job Start bit prior to module recovery. When this condition occurs the module will not execute the job.

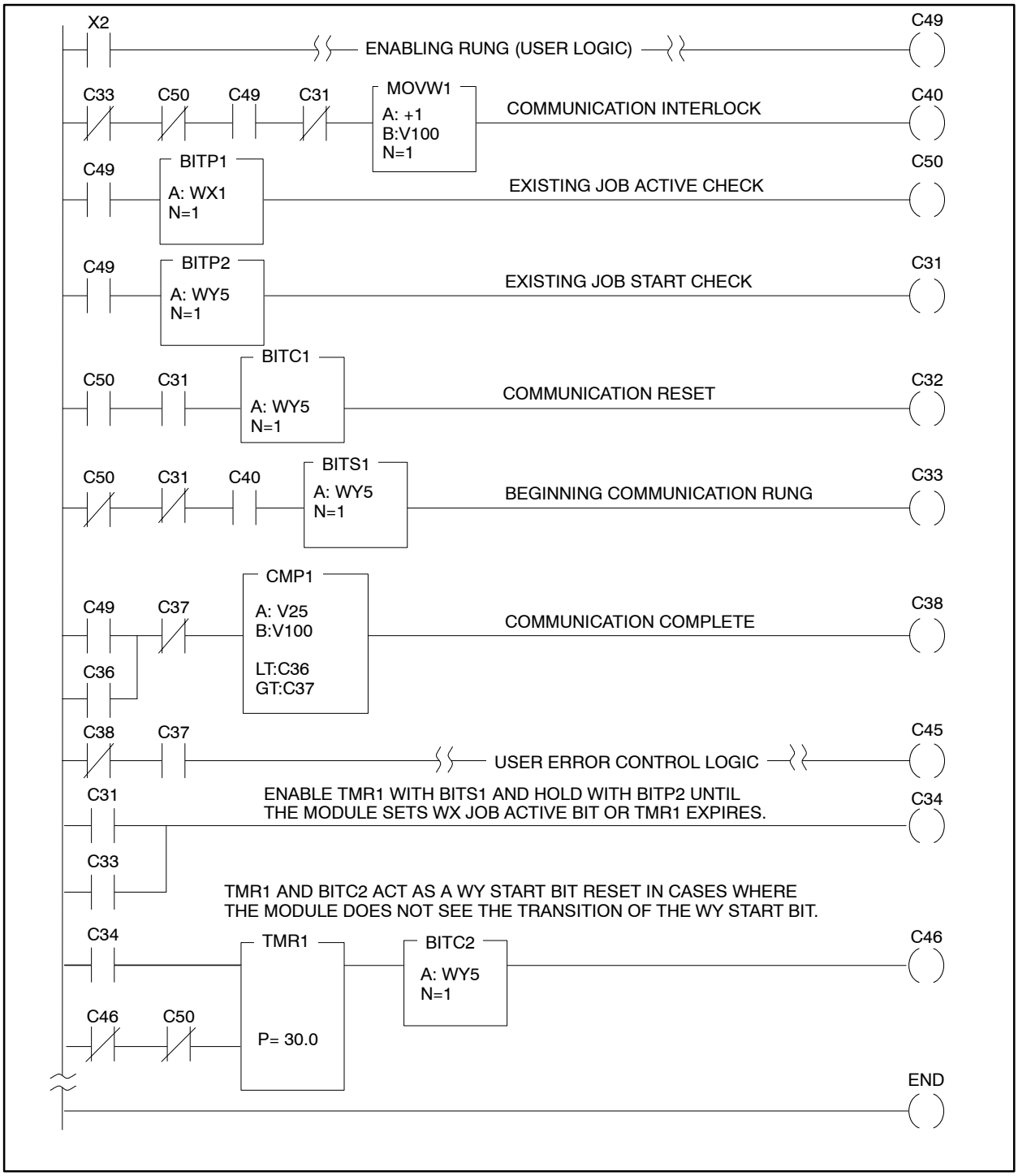


Figure 6-2 Receive RLL for PLC B

6.3 Sending between SIMATIC 505 Modules and S7 CPs

In this example there are three modules: two SIMATIC 505 TCP/IP modules and one S7 CP 343–1 IT. Module A, the first 505 module, sends to Module C, the S7 CP, and Module C in turn sends to Module B, the second 505 module. It does not matter which job or module is configured first. For the purpose of this example, the 505 modules are configured first.

NOTE: Before proceeding with this example, consult with your network administrator to ensure that the IP Address and Local Ethernet (MAC) Address values used in the example do not cause conflicts on your network. If necessary, your network administrator can suggest substitute values for the example.

Table 6-4 shows the important parameters of all three modules.

NOTE: In this example, the Send and Receive transmit the same amount of data to the same location. Otherwise, the buffer address and length of the Send would differ from the Receive.

Table 6-4 505 and S7 Module Parameters

Parameters	SEND Module A (505)	RECEIVE Module C (S7)	SEND Module C (S7)	RECEIVE Module B (505)
IP Address	161.218.124.222	161.218.124.230	161.218.124.230	161.218.124.223
Local Ethernet (MAC) Address	08.00.06.01.01.01	08.00.06.01.01.05	08.00.06.01.01.05	08.00.06.01.01.02
Local TCP Port (of job on module)	5556	5557	5558	5555
Status Word	V100 (Note1)	not applicable (Note2)	not applicable (Note2)	V100 (Note1)
Job Start Bit	5.1 (Note1)	not applicable (Note3)	not applicable (Note4)	5.1 (Note1)
Job Active Bit	1.1	not applicable (Note5)	not applicable (Note6)	1.1
Local PLC Data	V10000, 200	(Note 7)	(Note 8)	V20000, 200

Notes:

1) It does not matter whether you use differing Status Word values on Module A and Module B, since they are using the memory of different PLCs. Likewise, the Job Start bits of the Send job on Module A and the Receive job on Module B do not have to differ, as long as the two modules are connected to different PLCs.
2) The equivalent in a STEP 7 program is the STATUS variable, defined in the FC as MW44 for this example, and the ERROR bit, M40.1.
3) There is no direct equivalent in STEP 7 for a Job Start bit on the Receive side. (OB1 initiates the Receive job by calling MY_RECEIVE_CALL, which in turn calls the standard Siemens function AG_LRECV.)
4) For the Send job, the equivalent of a Job Start bit is the ACT bit, defined in this example as M41.0.
5) For the Receive job, the equivalent of a Job Active bit is the NDR bit (New Data Received), defined in this example as M40.2.
6) For the Send job, the equivalent of a Job Active bit is the DONE bit (M40.4).
7) Local PLC Data for the Receive (FC41) is stored in P#DB40.DBX2.0. The buffer length (RECV_SEND_BUFFERLENGTH, MW48) is 400 bytes.
8) Local PLC Data for the Send (FC40) is stored in P#DB40.DBX2.0. The buffer length (RECV_SEND_BUFFERLENGTH, MW48) is 400 bytes.

Configure Network Parameters for Module A

To configure network parameters for the module, follow these steps:

1. Open the SIMATIC 505-CP1434-TCP Module Configurator software application.
2. From the menu bar, choose **Edit > Set Network Parameters** to bring up the Module Network Parameters dialog box.
3. In the Local Ethernet (MAC) Address field, enter the following value:

   080006010101

4. Ignore the Enable Redirect check box. For the purpose of this example, it makes no difference whether the box is selected.
5. In the Network IP Address field, enter the following value:

   161.218.124.222

6. In the Subnet Mask field, enter the following value:

   255.255.0.0

7. Leave the Gateway field empty. For the purpose of this example, it is assumed that you are communicating on the local area network only, and do not need a gateway (router) value.
8. The host name parameter is used by e-mail jobs, not send or receive jobs, but it is good practice to have unique host names for each SIMATIC 505 TCP/IP module: for identification, and in case e-mail jobs are configured on the module(s) later. Enter the following value:

   cp1434-tcpModuleA

9. Leave the Domain Name field empty.
10. Accept the default Broadcast parameters. (If you decided to add Broadcast jobs later, you could edit Module Network Parameters and change the Broadcast settings at that point.)
11. Click the OK button to confirm your work and close the Module Network Parameters dialog box.

Configure the Send Job on Module A

To configure the Send job, follow these steps:

1. From the menu bar of the Configurator, choose the **Edit > Add Job** command. The Add Job Selection dialog box opens.
2. In the Job Name field, type the following:

   Send to Module C
3. From the Job Type drop-down list box, choose Send.
4. Click the Configure button. The Job Configuration dialog box opens.
5. In the Local TCP Port field, type the following:

   5556
6. In the Remote TCP Port field, type the following:

   5557
7. In the Remote IP field, type the following:

   161.218.124.230
8. In the Status Word field, type the following:

   80
9. Accept the default Timeout value of 30.
10. In the Start Bit field, type the following:

    5.1
11. In the Local PLC Data field, type the following:

    V10000, 200
12. Click the OK button to confirm your work and close the Job Configuration dialog box.
13. Click the OK button to confirm your work and close the Add Job Selection dialog box.

Download the Configuration to Module A

To download your configuration to Module A, follow these steps:

1. From the menu bar of the Configurator, choose the **Transfer > To Module** command.

2. Click the Start button.

   If there is already a configuration in the module, you are prompted to decide whether to overwrite it at this time and continue with the example procedure, or cancel the download to protect the existing module configuration. (You could upload the “real” module configuration and save it as a file, then download the “example” configuration temporarily, restoring the “real” configuration later.)

   When the download is complete, the Transfer Configuration to Module dialog box automatically closes. You can now save the example configuration to a file on your PC if desired.

Connect the Configurator to the Second Module

Next, switch the Configurator connection to Module B. If your Configurator is using a serial connection to the module, you must physically disconnect the cable from Module A and connect to Module B. (See Section 2.4 for information about connecting the Configurator to the serial port of the module.) If you have already configured network parameters for Module B (as described below) and your Configurator is using a network connection to the modules, you must edit your Network Selection in the Connection Mode dialog box to change the IP address from Module A to Module B. (See Section 3.10 for information about network connections.)

Configure Network Parameters for Module B

To configure network parameters for the module, follow these steps:

1. Open a configuration file for Module B in the Configurator.

2. From the menu bar, choose **Edit > Set Network Parameters** to bring up the Module Network Parameters dialog box.

3. In the Local Ethernet (MAC) Address field, enter the following value:

   080006010102

4. Ignore the Enable Redirect check box. For the purpose of this example, it makes no difference whether the box is selected.

5. In the Network IP Address field, enter the following value:

   161.218.124.223

6. In the Subnet Mask field, enter the following value:

   255.255.0.0

7. Leave the Gateway field empty. For the purpose of this example, it is assumed that you are communicating on the local area network only and do not need a gateway (router) value.

8. Edit the Host Name field to differentiate your second SIMATIC 505 TCP/IP module from your first one. The host name parameter is used by e-mail jobs, not send or receive jobs, but it is good practice to have unique host names for each module: for identification, and in case e-mail jobs are configured on the module(s) later. Enter the following value:

   cp1434-tcpModuleB

9. Leave the Domain Name field empty.

10. Accept the default Broadcast parameters. (If you decided to add Broadcast jobs later, you could edit Module Network Parameters and change the Broadcast settings at that point.)

11. Click the OK button to confirm your work and close the Module Network Parameters dialog box.

Configure the Receive Job on Module B

To configure the Receive job, follow these steps:

1. From the menu bar of the Configurator, choose the **Edit > Add Job** command. The Add Job Selection dialog box opens.

2. In the Job Name field, type the following:

   Rcv from Module C

3. From the Job Type drop-down list box, choose Receive.

4. Click the Configure button. The Job Configuration dialog box opens.

5. In the Local TCP Port field, type the following:

   5555

6. In the Remote TCP Port field, type the following:

   5556

7. In the Remote IP field, type the following:

   161.218.124.230

8. In the Status Word field, type the following:

   100

   It does not matter whether you use the same Status Word values on Module A and Module B, since they are using the memory of different PLCs.

9. Accept the default Timeout value of 30.

10. In the Start Bit field, type the following:

    5.1

    It does not matter whether you use the same Start Bit values on Module A and Module B, since they are using the memory of different PLCs.

11. In the Local PLC Data field, type the following:

    V20000, 200

    The starting addresses of blocks in the sending PLC do not have to match the starting address of blocks in the receiving PLC. Nor do the memory types have to match. The sizes of the blocks must match. Since different data types use different units, this could result in different counts in order to achieve matching block sizes: for instance, two bytes of CP memory (CP1, 9) can fit into one word of V memory (V10, 1).

12. Click the OK button to confirm your work and close the Job Configuration dialog box.

13. Click the OK button to confirm your work and close the Add Job Selection dialog box.

Download the Configuration to Module B

To download your configuration to Module B, follow these steps:

1. From the menu bar of the Configurator, choose the **Transfer > To Module** command.

2. Click the Start button.

   If there is already a configuration in the module, you are prompted to decide whether to overwrite it at this time and continue with the example procedure, or cancel the download to protect the existing module configuration. (You could upload the “real” module configuration and save it as a file, then download the “example” configuration temporarily, restoring the “real” configuration later.)

   When the download is complete, the Transfer Configuration to Module dialog box automatically closes. You can now save the example configuration to a file on your PC if desired.

RLL Code for Send Job at PLC A

The RLL example in Figure 6-3 demonstrates a typical scenario for enabling a Send request. The WX and WY locations used in this RLL example correspond to module I/O addresses that have been configured in the memory of PLC A using the configuration function of a tool like SoftShop.

Table 6-5 explains the program logic that is shown in Figure 6-3:

Table 6-5 Program Logic

C98	Enables condition logic defined by the application (X1) which must remain active until job completion.
MOVW1	Sets status word V100 to a positive value (1) prior to each transaction. This positive value is used for job completion check in CMP1.
BITP3/BITP4	Assures that no pending active Send jobs exist prior to starting a Send job (defined for Bit 1 of the SIMATIC 505 TCP/IP module, located at I/O address WY5).
BITC3	Clears the Job Start bit defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 after the Job Active bit WX1 is set.
BITS2	Sets the Job Start bit defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 to invoke a data transfer function.
CMP1	Tests the completion status of the Send data transfer by the same SIMATIC 505 TCP/IP module, placed in V100, against a zero constant stored at location V15. When the Send job has completed, the value of status word V100 is either zero if the transaction was good, or else a non-zero value (see Table B-2) if the transaction was bad. You must take appropriate action in the USER ERROR CONTROL LOGIC ladder rung at this time. If the Send job is to be executed only one time, you take must take action to disable X1 after the Status word changes value from its preset condition.
TMR2 BITC4	The parallel net of C41 and C43 act together to start and hold TMR2 on until the SIMATIC 505 TCP/IP module recognizes the Job Start bit WY5, 1 has been set and responds by setting the Job Active bit WX1, 1 or TMR2 time expires. TMR2 and BITC4 are used to clear an unrecognized Job Start bit WY5, 1. Either a reset of the module or a power cycle of the entire base may result in the programmable logic controller ladder setting a WY Job Start bit prior to module recovery. When this condition occurs the module will not execute the job.

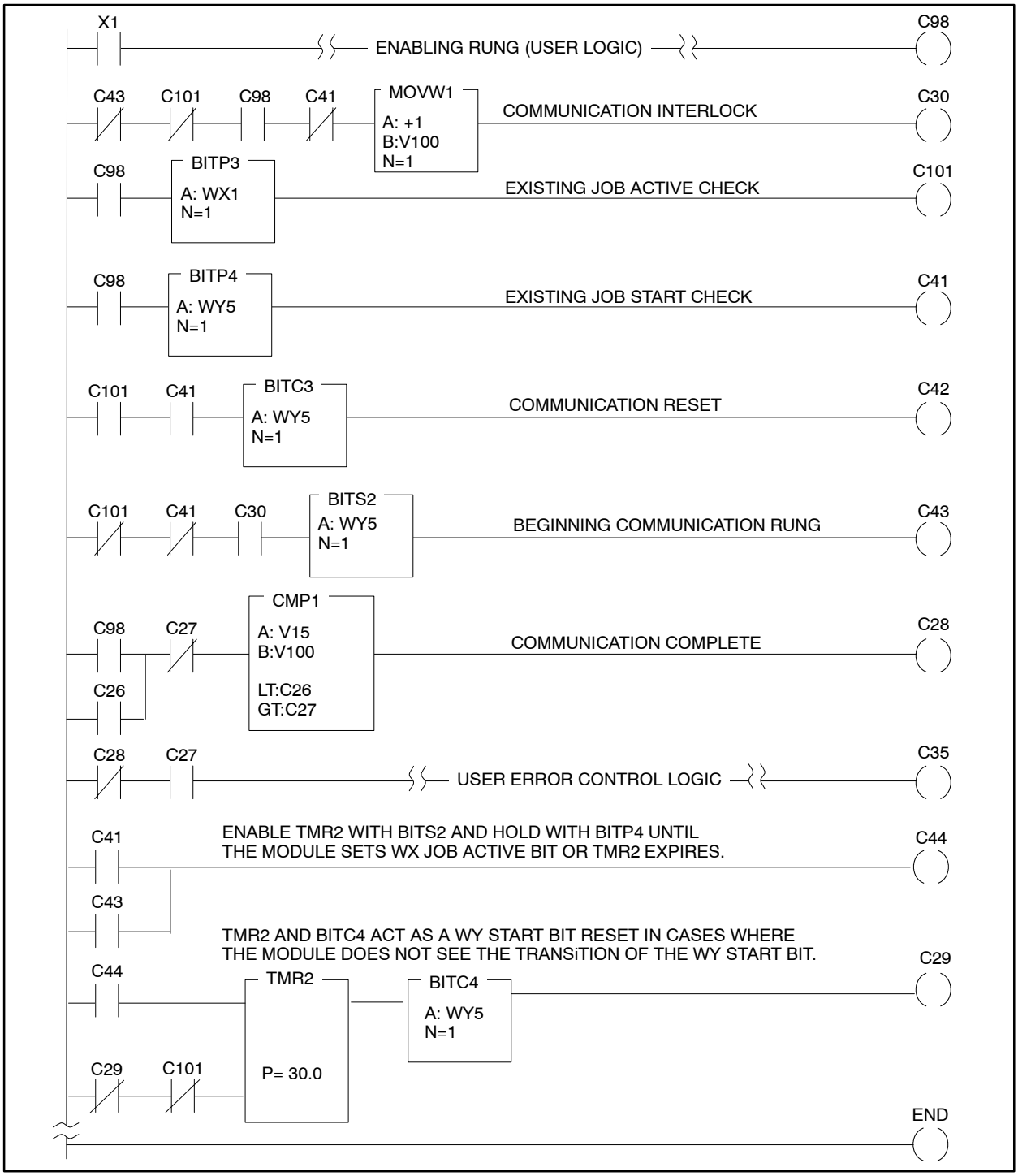


Figure 6-3 Send RLL for PLC A

RLL Code for Receive Job at PLC B

The RLL example in Figure 6-4 demonstrates a typical scenario for enabling a Receive acknowledgement. The WX and WY locations used in this RLL example correspond to module I/O addresses that have been configured in the memory of PLC B using the configuration function of a tool like SoftShop.

Table 6-6 explains the program logic that is shown in Figure 6-4:

Table 6-6 Program Logic

C49	Enables condition logic defined by the application (X2) which must remain active until job completion.
MOVW1	Sets status word V100 to a positive value (1) prior to each transaction. This positive value is used for job completion check in CMP1.
BITP1/BITP2	Assures that no pending active Receive jobs exist prior to starting a Receive job defined for Bit 1 of the SIMATIC 505 TCP/IP module, located at I/O address WY5.
BITC1	Clears the Receive job defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 after the Receive Job Active bit WX1 is set.
BITS1	Sets the Job Start Bit defined for Bit 1 of the SIMATIC 505 TCP/IP module located at WY5 to invoke a data transfer function.
CMP1	Tests the completion status of the Receive data transfer by the same SIMATIC 505 TCP/IP module, placed in V100, against a zero constant stored at location V25. When the Receive job has completed, the value of status word V100 is either zero if the transaction was good, or else a non-zero value (see Table B-2) if the transaction was bad. You must take appropriate action in the USER ERROR CONTROL LOGIC ladder rung at this time. If the Receive job is to be executed only one time, you take must take action to disable X2 after the Status word changes value from its preset condition.
TMR1 BITC2	The parallel net of C31 and C33 act together to start and hold TMR1 on until the module recognizes the Job Start bit WY5, 1 has been set and responds by setting the Job Active bit WX1, 1 or TMR1 time expires. TMR1 and BITC2 are used to clear an unrecognized Job Start bit WY5, 1. Either a reset of the module or a power cycle of the entire base may result in the programmable logic controller ladder setting a WY Job Start bit prior to module recovery. When this condition occurs the module will not execute the job.

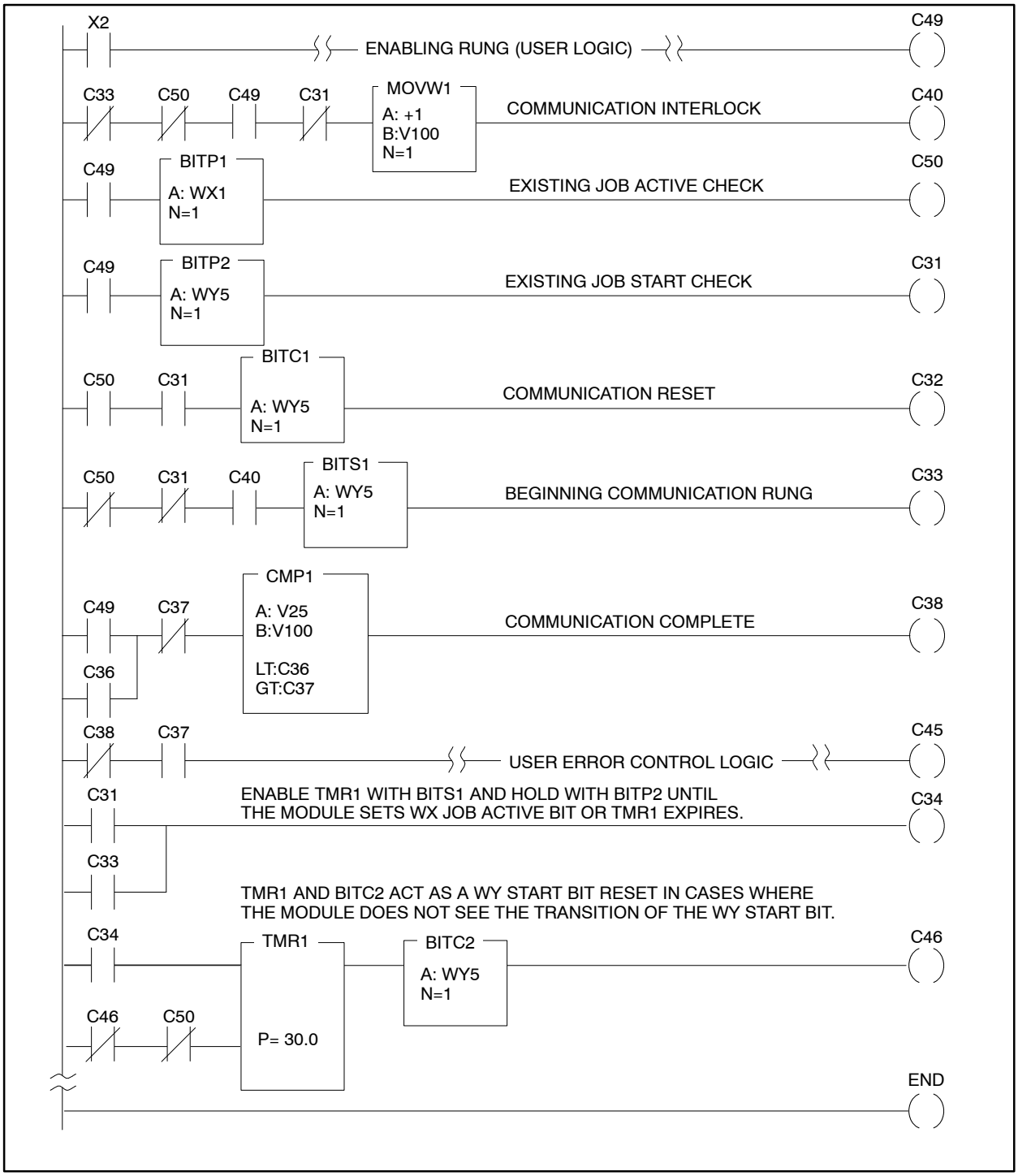


Figure 6-4 Receive RLL for PLC B

Open the SIMATIC Manager

The remaining work is done on Module C. Closing the Configurator is optional.

Ensure that Module A and Module B are connected to the Ethernet (Section 2.5 describes connecting 505 modules to the Ethernet). Open the SIMATIC Manager on the programming station that is connected to your S7 CP (Module B), and open the project that you want to use.

NOTE: There are multiple ways to perform tasks in STEP 7. The directions below represent only one possibility. The basic tasks remain constant, but you may use different steps to perform them. The details below are valid for STEP 7 V5.1.

Create the S7 Project, Station, and Ethernet Network

To create and name an S7 project, with a SIMATIC 300 station and an Ethernet network, follow these steps:

1. In the SIMATIC Manager, choose the **File > New...** menu command. The New dialog box opens.
2. In the Name field, type 505_Demo.
3. Click the OK button to confirm your work and close the dialog box.
4. To add an Ethernet network to your project, choose the **Insert > Subnet > Industrial Ethernet** menu command.
5. In the highlighted name field of the new subnet object, delete Ethernet(1) and type My_Ethernet.
6. To add a station, select the 505_Demo project icon and choose the **Insert > Station > SIMATIC 300 Station** menu command.
7. Right-click on the SIMATIC 300(1) icon and choose Object Properties from the pop-up menu.
8. In the Name field, type My_300_Station. Click the OK button to close the dialog box.

Configure the Station Hardware

To configure hardware parameters for the SIMATIC 300 station, including a CP 343–1 IT module, follow these steps:

1. Open the 505_Demo project icon, then click the My_300_Station station icon (in the left pane of the SIMATIC Manager) to reveal the Hardware icon (in the right pane).

2. Double-click the Hardware icon to open the HW Config editor.

3. In the Hardware Catalog, open the SIMATIC 300 folder.

4. Select a rail from the RACK-300 folder of the Hardware Catalog, drag the rack into the upper half of the station window (upper pane of the HW Config editor), and drop it.

5. Drag a power supply from the PS-300 folder of the Hardware Catalog and drop it onto slot 1 of the rack.

6. Drag a CPU from the CPU-300 folder of the Hardware Catalog and drop it onto slot 2 of the rack. For this example, choose a CPU 315–2 DP.

   When you drop the CPU onto the rack, the Properties – PROFIBUS Interface DP Master dialog box opens. Choose the Cancel button.

7. Drag a CP from the CP-300 folder of the Hardware Catalog and drop it onto slot 4 of the rack. For this example, choose an S7 CP 343–1 IT (inside the Industrial Ethernet folder).

   When you drop the CP onto the rack, the Properties – Ethernet Interface CP 343–1 IT dialog box opens.

8. In the MAC Address field, type the following:

   08.00.06.01.00.05

   Note: In a real project, you need to make sure that all of the devices on your local area network have unique local Ethernet (MAC) addresses.

9. In the IP Address field, type the following:

   161.218.124.230

   Note: In a real project, you would consult your network administrator to obtain a valid IP address.

10. In the Subnet Screen Form field, accept the default: 255.255.0.0
11. Under Gateway, accept the default: Do not use router.
12. Under Subnet, choose My_Ethernet from the list.
13. Click OK to confirm your work and close the dialog box.

Save Your Work

Performing incremental saves and error checks helps to prevent data loss or project errors. Follow these steps:

1. From the menu bar of the HW Config editor, choose the **Station > Save and Compile** command.
2. If any errors are detected, choose the **Station > Consistency Check** menu command to obtain more information about them.
3. Close the HW Config window after you are satisfied that the station configuration contains no errors.

Add the 505 Modules to the Project

To add the 505 modules to your STEP 7 project, follow these steps:

1. Click the 505_Demo project icon.
2. From the menu bar of the SIMATIC Manager, choose the **Insert > Station > Other station** command.
3. In the highlighted name field of the new station object, delete Other station(1) and type My_505_A.
4. From the menu bar of the SIMATIC Manager, choose the **Insert > Station > Other station** command.
5. In the highlighted name field of the new station object, delete Other station(2) and type My_505_B.

Connect Module A to the Ethernet

To configure the Ethernet connection for Module A, follow these steps:

1. Right-click on the My_505_A icon and choose Object properties from the pop-up menu. The Properties – Other station dialog box opens.

2. Click the Interfaces tab, then click the New button. The New Interface – Type Selection dialog box opens.

3. In the Type list, select IP interface, then click OK to close the dialog box. The Properties – Ethernet Interface dialog box opens.

4. In the MAC Address field, type the following:

   08.00.06.01.00.01

   This is the MAC address of Module A, the sending 505 module.

5. In the IP Address field, type the following:

   161.218.124.222

6. In the Subnet Screen Form field, accept the default: 255.255.0.0

7. Under Gateway, accept the default: Do not use router.

8. Under Subnet, choose My_Ethernet from the list.

9. Click OK to confirm your work and close the Properties – Ethernet interface dialog box.

   The Interfaces tab of the Properties – Other station dialog box now displays the name of the Ethernet network to which Module A is connected and the IP address of Module A.

10. Click the OK button to confirm your work and close the dialog box.

Connect Module B to the Ethernet

To configure the Ethernet connection for Module B, follow these steps:

1. Right-click on the My_505_B icon and choose Object properties from the pop-up menu. The Properties – Other station dialog box opens.
2. Click the Interfaces tab, then click the New button. The New Interface – Type Selection dialog box opens.
3. In the Type list, select IP interface, then click OK to close the dialog box. The Properties – Ethernet Interface dialog box opens.
4. In the MAC Address field, type the following:

   08.00.06.01.00.02

   This is the MAC address of Module B, the receiving 505 module.
5. In the IP Address field, type the following:

   161.218.124.223
6. In the Subnet Screen Form field, accept the default: 255.255.0.0
7. Under Gateway, accept the default: Do not use router.
8. Under Subnet, choose My_Ethernet from the list.
9. Click OK to confirm your work and close the Properties – Ethernet interface dialog box.

   The Interfaces tab of the Properties – Other station dialog box now displays the name of the Ethernet network to which Module B is connected and the IP address of Module B.
10. Click the OK button to confirm your work and close the dialog box.

Connect the S7 CPU to Module A

To connect the S7 CPU to the 505 sending module (Module A), follow these steps:

1. Open the My_300_Station station icon, then click the CPU 315 icon (in the left pane) to reveal the Connections icon (in the right pane).
2. Double-click the Connections icon to open the NetPro editor.

3. Inside the My_300_Station object, select the CPU square and choose the **Insert > New Connection** menu command.

   The New Connection dialog box opens.

4. Under Connection Partner, choose My_505_A from the Station drop-down list.

5. Under Connection, choose TCP connection from the Type drop-down list.

   If the Display Properties Dialog check box is selected, de-select it.

6. Click the Apply button. The New Connection dialog box closes and the Connection Table of the NetPro editor displays the details of the connection that you have just made between the S7 CPU and the 505 station (Module A).

Connect the S7 CPU to Module B

To connect the S7 CPU to the 505 receiving module (Module B), follow these steps:

1. From the NetPro menu bar, choose the **Insert > New Connection** menu command.

   The New Connection dialog box opens.

2. Under Connection Partner, choose My_505_B from the Station drop-down list.

3. Under Connection, choose TCP connection from the Type drop-down list.

   Make sure the Display Properties Dialog check box is selected. If it is not, select it.

4. Click the Apply button. The New Connection dialog box closes and the Connection Table of the NetPro editor displays the details of the connection that you have just made between the S7 CPU and the 505 station.

   After you click the Apply button, the Properties - TCP Connection dialog box opens.

Configure Properties for the TCP Connection to Module B

The Properties – TCP Connection dialog box should be open, displaying parameters for the TCP connection between the S7 CPU and Module B. To configure the parameters, follow these steps:

1. On the General page, examine the following parameters:
   - The check box for Active connection establishment should be selected (because the S7 module is in Send mode). If it is not selected, select it.
   - Take note of the Block Parameter values: they are used in your Send FC (as the ID and LADDR values).
2. Click the Addresses tab.
3. Under Local, in the PORT field, type the following:

   5558

   This is the TCP port value for the Send job on the S7 CP.
4. Under Remote, in the PORT field, type the following:

   5555

   This is the TCP port value for the Receive job on the 505 module (Module B).
5. Click the Overview tab if you want to print a hard copy of this configuration information (recommended).
6. Click the OK button to confirm your work and close the dialog box.

Configure Properties for the TCP Connection to Module A

To configure properties for the TCP connection between the SIMATIC 300 station and the 505 module, follow these steps:

1. In the Connection Table of the NetPro editor, right-click on the TCP connection between the S7 CPU and the Send module (Module A) and choose Object Properties from the pop-up menu. The Properties – TCP Connection dialog box opens.
2. On the General page, examine the following parameters:
   - The check box for Active connection establishment should not be selected (because the S7 module is in Receive mode, receiving the job from the 505 module; the connection is passive). If it is selected, de-select it.
   - Take note of the Block Parameter values: they are used in your Receive FC (as the ID and LADDR values).

3. Click the Addresses tab.

4. Under Local, in the PORT field, type the following:

   5557

   This is the TCP port value for the Receive job on the S7 CP.

5. Under Remote, in the PORT field, type the following:

   5556

   This is the TCP port value for the Send job on the 505 module (Module A).

6. Click the Overview tab if you want to print a hard copy of this configuration information (recommended).

7. Click the OK button to confirm your work and close the dialog box.

Set the Data Length

To set the data length, follow these steps:

1. Inside the My_300_Station object, select the CP square and choose the **Edit > Object Properties** menu command. The Properties - CP 343-1 IT dialog box opens.

2. Click the Options tab.

3. If the Data length > 240 bytes check box is not selected, select it.

4. Click the OK button to confirm your work and close the dialog box.

Save and Download Your Work

Performing incremental saves and error checks helps to prevent data loss or project errors. Follow these steps:

1. From the menu bar of the NetPro editor, choose the **Network > Save and Compile** command.

2. In the Save and Compile dialog box, select the Compile and Check Everything, and then click the OK button. You are automatically notified as to whether the network configuration contains any errors.

3. Select the My_300_Station object (click in the square with the station name) and choose the **PLC > Download > Selected Stations** menu command.

4. Click the OK button to download the configuration to the PLC.

   If there is already a configuration in the S7 PLC, you are prompted to decide whether to overwrite it at this time and continue with the example procedure, or cancel the download to protect the existing module configuration.

   When the download is complete, the download dialog box automatically closes.

5. Close the NetPro window.

This concludes the configuration work for the S7 CP and 505 modules. Next, you need to create program and data blocks and symbol and variable tables.

Edit OB1 to Call Your FCs

Edit OB1 to call MY_RECEIVE_CALL (FC41) and MY_SEND_CALL (FC40), as shown in Figure 6-5.

Address	Declaration	Name	Type	Initial value	Comment
0.0	temp	OB1_EV_CLASS	BYTE		Bits 0-3=1 (Coming event), Bits 4-7=1 (Event class 1)
1.0	temp	OB1_SCAN_1	BYTE		1 (Cold restart scan 1 of OB1), 3 (Scan 2-n of OB1)
2.0	temp	OB1_PRIORITY	BYTE		1 (Priority of 1 is lowest)
3.0	temp	OB1_OB_NUMBR	BYTE		1 (Organization block 1, OB1)
4.0	temp	OB1_RESERVED_1	BYTE		Reserved for system
5.0	temp	OB1_RESERVED_2	BYTE		Reserved for system
6.0	temp	OB1_PREV_CYCLE	INT		Cycle time of previous OB1 scan (milliseconds)
8.0	temp	OB1_MIN_CYCLE	INT		Minimum cycle time of OB1 (milliseconds)
10.0	temp	OB1_MAX_CYCLE	INT		Maximum cycle time of OB1 (milliseconds)
12.0	temp	OB1_DATE_TIME	DATE_AND_TIME		Date and time OB1 started

Block: OB1

Network: 1

```
//
//  example for send/receive with 400 bytes of data
//
        CALL        "MY_RECEIVE_CALL"                    // FC41
//
        CALL        "MY_SEND_CALL"                       // FC40
```

Figure 6-5 OB1 Calls FCs

Create an FC to Call AG_LRECV

OB1 calls this FC, which in turn calls the standard Siemens function AG_LRECV (FC 60). Fill out the function call (FC41, MY_RECEIVE_CALL) as shown in Figure 6-6.

Address	Declaration	Name	Type	Start Value	Comment
	in				
	out				
	in_out				
	temp				

Block: FC41

Network: 1

```
//
        CALL      "AG_LRECV"
          ID      :=1                                  // ID of connection
          LADDR   :=W#16#100                           // logic module address of CP
          RECV    :=P#DB40.DBX2.0 BYTE 400             // ANY-pointer to receive buffer; use
                                                       // BYTE value
          NDR     :="RECV_NDR"                         // new data received
          ERROR   :="RECV_ERROR"
          STATUS  :="RECV_STATUS"                      // actual  function status
          LEN     :="RECV_BUFFERLENGTH"
//
        U         "RECV_ERROR"                         // if function error, jump
        SPB       err
//
        L         "RECV_BUFFERLENGTH"                  // save length of receive data
        T         "DATA_BUFFER".send_bufferlength
//
        U         "RECV_NDR"                           // received new data ?
        S         "SEND_ACTIV"                         // call send function
        BE
//
err:    L         "RECV_STATUS"                        // save actual (error) send status
        T         "PARAMETER/ERROR_DB".receive_error_status
```

Figure 6-6 FC41 "MY_RECEIVE_CALL"

Create an FC to Call AG_LSEND

OB1 calls this FC, which in turn calls the standard Siemens function AG_LSEND (FC 50). Fill out the function call (FC40, MY_SEND_CALL) as shown in Figure 6-7.

Address	Declaration	Name	Type	Start Value	Comment
	in				
	out				
	in_out				
	temp				

Block: FC40

Network: 1

```
//
        CALL      "AG_LSEND"
          ACT     :="SEND_ACTIV"                    // start send after receive
          ID      :=2                               // ID of connection
          LADDR   :=W#16#100                        // logic module address of CP
          SEND    :=P#DB40.DBX2.0 BYTE 400          // ANY-pointer to receive buffer; use
                                                    // BYTE value
          LEN     :="RECV_BUFFERLENGTH"             // length of send data
          DONE    :="SEND_DONE"
          ERROR   :="SEND_ERROR"
          STATUS  :="SEND_STATUS"                   // actual  function status
//
        UN        "SEND_ERROR"                      // if no error, end
        U         "SEND_DONE"
        R         "SEND_ACTIV"
        BEB
//
        L         "SEND_STATUS"                     // save actual (error) send status
        T         "PARAMETER/ERROR_DB".send_error_status
//
```

Figure 6-7 FC40 "MY_SEND_CALL"

Complete the Symbol Table

In this example, symbolic names are used in the code. Symbolic names help the original programmer remember the purpose of variables and they are useful to future programmers who have to modify or debug the code. To open the Symbol Editor in the SIMATIC Manager and define symbols, follow these steps:

1. Select the S7 Program(1) program icon (left pane) and double-click on the Symbols icon (right pane). The Symbol Editor opens.

2. Fill out the symbol table as shown in Figure 6-8.

Symbol	Address		Data type	Comment
AG_LRECV	FC	60	FC 60	Long receive function for SEND/RECEIVE-Services with SIMATIC NET CPs
AG_LSEND	FC	50	FC 50	Long send function for SEND/RECEIVE-Services with SIMATIC NET CPs
DATA_BUFFER	DB	40	DB 40	Send and receive buffer
FIRST_RUN	M	40.0	BOOL	First program step after stop->run
LAST_ERROR_SAVE	FC	43	FC 43	This FC is called to save the last error
MY_RECEIVE_CALL	FC	41	FC 41	This FC calls the AG_RECV function
MY_SEND_CALL	FC	40	FC 40	This FC calls the AG_SEND function
PARAMETER/ERROR_DB	DB	43	DB 43	DB for last send/receive error
RECEIVE	M	40.6	BOOL	Wait for receive function call
RECEIVE_DATA	MW	50	WORD	For received data
SEND_DATA	MW	52	WORD	For sent data
RECV_BUFFERLENGTH	MW	48	INT	Length of received data
RECV_ERROR	M	40.1	BOOL	Indicates incorrect execution
RECV_NDR	M	40.2	BOOL	Confirmation of execution
RECV_STATUS	MW	44	WORD	Detailed error and status decoding
SEND	M	40.7	BOOL	Wait for send function call
SEND_ACTIV	M	41.0	BOOL	Send is active
SEND_DONE	M	40.4	BOOL	Confirmation of execution
SEND_ERROR	M	40.3	BOOL	Indicates incorrect execution
SEND_STATUS	MW	42	WORD	Detailed error and status decoding
SEND_STATUS_ACTUAL	M	40.5	BOOL	Display new send status
VAT2_BfLn	VAT	1		

Figure 6-8 Example Symbol Table

Define a VAT This example uses VAT2_BfLn. Fill it out as shown in Figure 6-9.

	Address	Symbol	Display format
1	//SEND:		
2	//DATA:		
3	DB40.DBD 2		DEC
4	//DONE:		
5	M 40.4	“SEND_DONE”	BIN
6	//ERROR:		
7	M 40.3	“SEND_ERROR”	BIN
8	//STATUS:		
9	MW 42	“SEND_STATUS”	HEX
10	MW 46	“SEND_BUFFERLENGTH”	HEX
11	//RECEIVE:		
12	//DATA:		
13	DB40.DBD 2		HEX
14	//NDR:		
15	M 40.2	“RECV_NDR”	BIN
16	//ERROR:		
17	M 40.1	“RECV_ERROR”	BIN
18	//STATUS:		
19	MW 44	“RECV_STATUS”	HEX
20	MW 48	“RECV_BUFFERLENGTH”	HEX
21	//last error in program		
22	DB43.DBW 2	“PARAMETER/ERROR_DB”.receive_error_status	HEX
23	DB43.DBW 0	“PARAMETER/ERROR_DB”.send_error_status	HEX
24			

Figure 6-9 VAT2_BfLn

Define Data Blocks

This example uses three data blocks.

DB40 contains the send buffer. Fill it out as shown in Figure 6-10.

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	send_bufferlength	INT	0
+2.0	send_data	ARRAY [0..399]	B#16#0
*1.0		BYTE	
=404.0		END_STRUCT	

Figure 6-10 DB40

DB41 contains the receive buffer. Fill it out as shown in Figure 6-11.

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	recv_bufferlength	INT	0
+2.0	recv__data	ARRAY [0..399]	W#16#0
*2.0		WORD	
=402.0		END_STRUCT	

Figure 6-11 DB41

DB43 contains the parameter/error buffer. Fill it out as shown in Figure 6-12.

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	send_error_status	WORD	W#16#0
+2.0	receive_error_status	WORD	W#16#0
=4.0		END_STRUCT	

Figure 6-12 DB43

Compile and Download Your Program

Once you have configured your project and written the necessary program blocks and tables, compile and download the project to your PLC.

Appendix A
Specifications

Operating temperature	0 to 60 °C (32 to 140 °F)
Storage temperature	-40 to +70 °C (-40 to 158 °F)
Shock	IEC 68-2-27; Test Ea; half sine, 15 g, 11 ms
Pollution degree	2, IEC 664, 664A
Noise immunity, conducted	IEC 801, Part 4, Level 3 MIL STD 461B, Part 4 CS01, CS02, CS06 IEC 255-4, Appendix E EMA DC33 NEMA ICS 2-230.45 IEC 255
Noise immunity, radiated	IEC 801 Part 3, Level 3 MIL STD 461B, Part 4 RS01, RS02
Electrostatic discharge	IEC 801, Part 2, Level 4, (15 kV)
Minimum torque for bezel screws	2.6 in.-lbs. (0.3 N-m)
Maximum torque for bezel screws	4.12 in.-lbs. (0.6 N-m)
Module power required from base	19 W or less of +5 VDC with jumper E1 installed (see Section 2.2) 9 W of +5 VDC without jumper 0.1 W of -5 VDC with or without jumper
Relative humidity	5% to 95% noncondensing
Vibration	Sinusoidal IEC 68-2-6, Test Fc 0.15 mm peak-to-peak, 10-57 Hz; 1.0 g, 57-150 Hz Random IEC 68-2-34, Test Fdc, equivalent to NAVMAT P-9492 0.04 g^2/Hz, 80-350 Hz
Corrosion protection	All parts of corrosion-resistant material or plated or painted as corrosion protection
Agency approvals	Underwriters Laboratories: UL Listed (Industrial Control Equipment) Canadian Standards Association: CSA Certified (Process Control Equipment) Factory Mutual Approved; Class I, Div. 2 Hazardous Locations CE Marking (Low Voltage Directive 73/23/EEC and Electro-Magnetic Compatibility Directive 89/336/EEC)

Appendix B

Error Codes and Troubleshooting

B.1 Configurator Error Messages

Table B-1 shows error messages that may be displayed by the Configurator and suggests corrective actions.

Table B-1 Configurator Error Messages

Error Message	Possible Causes and Corrective Actions
"A host name must be specified."	Enter an alphanumeric string between 1 and 20 characters in length.
"A job has not been selected from the list."	Select the job that you wish to modify.
"A job type has not been selected."	Select a job type from the drop-down list.
"A maximum of 4 jobs are allowed for this type."	Examine your configuration and adjust your plans in order to stay within the number of jobs allowed.
"A maximum of 10 jobs are allowed for this type."	Examine your configuration and adjust your plans in order to stay within the number of jobs allowed.
"A valid job name must be specified."	You cannot save an unnamed job. Enter a valid job name.
"Access to this module has been locked out by another operation."	Indicates simultaneous Configurator access by means of serial and/or TCP connections. Try again, or disconnect other Configurator sources.
"An error has occurred while opening the network connection."	Check network parameters, cables, and connections. Attempt to reconnect. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"An error has occurred while sending or receiving data over the network."	Try again; troubleshoot your cables and connectors. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"At least one local PLC data entry is required."	Enter at least one data block entry. Use the form datatype address,count . For example, V100,10 defines 10 words of V memory beginning at V100.
"Communications with the module has timed out."	Check cable connections. Try again. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"Communications with the PLC could not be made. Communications with the PLC are needed in order to save the message. Check your communications connections. A TCP Server job must be running in the module if you are going through the module to the PLC. For serial communications through the module, serial port redirect must be enabled and set to the destination PLC."	You must correct any conditions that prevent communications with the PLC before you can access the message editor.

Table B-1 Configurator Error Messages (continued)

Error Message	Possible Causes and Corrective Actions
”Connection to remote module has been lost. Use **Transfer > Set Redirect** to re-establish connection.”	If the **Transfer > Set Redirect** command does not take effect, check your cables and connectors. Also check the server job on the remote module. If the remote module is using a TCP Server job for the redirect connection, and the Keepalive parameter of the TCP Server job is enabled, the TCP Server job automatically dis-connects if the connection is inactive for more than 7200 s.
“Error setting the baud rate.”	Make sure cables are connected, module is powered on, and MOD GOOD indicator is on.
“Invalid parameter encountered.”	Unexpected protocol. Call the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
“Invalid PC port.”	Check serial cable connection on PC and use menu command **Transfer > Set Connection Mode** to examine Connection Mode dialog box. The COM port setting must match the port that is actually used on the PC. For more information, see page 3-11.
“Module communications error.”	This can be caused by phenomena such as a checksum error, invalid data, or an invalid connection. Try again. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
“No connection to remote module. Check IP address, port, and protocol.”	Use the menu command **Edit > Set Network Parameters** to access the Module Network Parameters dialog box, where you can examine these parameters.
“No header type has been specified. Select a type from the drop down list.”	You cannot use the “Add Header” button unless you have already specified a header type from the drop down list.
“No *‘Mail From | Rcpt To | Message Text’* line was found.”	You must supply at least one line in each of the e-mail data blocks (MAIL FROM, RCPT TO, and MESSAGE).
“Redirect has not been enabled on the module. See the “edit > set network parameters” operation.”	Use the menu command **Edit > Set Network Parameters** to access the Module Network Parameters dialog box, where you can check the Enable Redirect box. For more information about setting Serial Port Redirect, see Section 3.11.
“SCI layer error.”	Call the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
“The address of the PLC remote data source field cannot be greater than 65535.”	Enter an address value that is less than 65,535.
“The block id field contains invalid data. Valid range is 0-255.”	Enter a value that is in range.
“The broadcast delivery time field contains invalid data. Valid range is 0-65535.”	Enter a value that is in range. Avoid setting extremely low values for the broadcast delivery timer. See page 3-7.
“The broadcast ID is already assigned to this local port in another job.”	Choose another value for the Broadcast ID (range is 0 to 255).

Table B-1 Configurator Error Messages (continued)

Error Message	Possible Causes and Corrective Actions
"The broadcast remote address field is invalid. Valid range is 1.0.0.0 – 223.225.255.255 or 255.255.255.255 ."	For instance, you cannot enter this value for a Broadcast Remote Address: 0.0.0.0
"The broadcast remote port field contains invalid data. Valid range is 1024-65535."	Enter a value that is in range for the Broadcast Remote UDP Port.
"The configuration contains a broadcast send job, but the PLC does not support it."	Only PLCs that support RBE support Broadcast Send jobs.
"The configurator must be in serial mode to perform this operation."	Use the menu command **Transfer > Set Connection Mode** to change the Configurator to serial mode.
"The copy job operation has failed."	Try again. You did not succeed in placing anything on the Clipboard.
"The cut job operation has failed."	Try again. You did not succeed in placing anything on the Clipboard.
"The domain name cannot be greater than 60 characters."	Enter an alphanumeric value that is between 1 and 60 characters in length, or leave the field blank.
"The file name cannot be greater than 29 characters."	Enter a shorter name.
"The file version is unknown and cannot be accessed."	The configuration file was created by a newer version of the Configurator. Upgrade your Configurator.
"The firmware file is not a known type."	If the firmware file is more recent than the Configurator, upgrade your Configurator.
"The firmware load operation failed due to incorrect checksum."	Try again. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"The firmware load operation has failed."	Try again. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"The firmware load operation is not allowed. Make sure the module firmware and the load file name are correct."	Try again. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"The gateway IP address field contains invalid data."	Valid range is 1.0.0.0 to 223.255.255.255 or 0.0.0.0.
"The host name cannot be greater than 20 characters."	Enter an alphanumeric value between 1 and 20 characters in length.
"The IP address field contains invalid data. Valid range is 1.0.0.0 - 223.255.255.255 or 0.0.0.0."	Enter an IP address that is in range, or enter 0.0.0.0 to set the IP address parameter to null.

Table B-1 Configurator Error Messages (continued)

Error Message	Possible Causes and Corrective Actions
"The job local port is already defined."	Choose a different value for the Local Port.
"The job name cannot be greater than 39 characters."	Shorten the name.
"The job start bit is already defined."	Enter an unused bit address. Use the correct form: for example, WY5.1 .
"The length of the local destination and/or remote source fields are invalid. The local destination and remote source fields must be equal in length. Valid length range is 2–512 bytes (1–256 words). Data types that use bits or packed bits must have a count that uses an even number of bytes (word boundary)."	Correct your block size so that it is an even value within the valid range.
"The length of the local source and/or remote destination fields are invalid. The local source and remote destination fields must be equal in length. Valid length range is 2–508 bytes (1–254 words). Data types that use bits or packed bits must have a count that uses an even number of bytes (word boundary)."	Correct your block size so that it is an even value within the valid range.
"The length of the local source and/or remote destination fields are invalid. The local source and remote destination fields must be equal in length. Valid length range is 2–512 bytes (1–256 words). Data types that use bits or packed bits must have a count that uses an even number of bytes (word boundary)."	Correct your block size so that it is an even value within the valid range.
"The local data length does not match the remote data length."	Adjust parameters for the job so that the local data block and the remote data block are equal in size.
"The *local* field length has been exceeded. Valid range is 1–110 bytes."	Enter a value within the range 1 to 110 bytes.

Table B-1 Configurator Error Messages (continued)

Error Message	Possible Causes and Corrective Actions
"The local port field contains invalid data. Valid range is 1024–65535."	Enter a value within the range 1,024 to 65,535.
"The *local/remote* field contains invalid data. Use the form "datatype address, count". Other possible causes include invalid memory type, missing comma, or an address or count out of range."	An example of a valid entry would be V100,10 . This defines 10 words of V memory beginning at V100. Or if the remote PLC were an S7 PLC, you would need to enter an S7 data type. S7 data types are described in Table C-4.
"The *'Mail From | Rcpt To | Message Text'* field address count is too large. The Mail From field has a maximum of 928 characters (464 words). The Rcpt To field has a maximum of 32766 characters (16383 words). The Message field has a maximum of 65534 characters (32767 words)."	Reduce the content of the blocks as necessary to fit within range. Bear in mind that this count includes hidden characters such as line feeds and carriage returns.
"The mail message has not been written to the PLC."	You can write the mail message to the PLC now, or you can click the Cancel button. Cancel gives you the option to save your message data temporarily (until you close the Configurator).
"The max clients field contains invalid data. Valid range is 1–64."	Enter a value that is in range.
"The max send interval field contains invalid data. Valid range is 0–655."	Enter a value (in seconds) that is in range.
"The maximum data length of the job has been exceeded."	Refer to Chapter 4 or the online help system to find out the maximum data length permitted for the job.
"The maximum data length of the mail from field has been exceeded."	The maximum count permitted is 464 words of V- or K-memory.
"The maximum number of entries has been exceeded."	The maximum number of entries permitted in the Serial Port Redirect Destination list is 50. If you want to add more entries, you must remove existing entries in order to make room.
"The maximum number of 64 jobs has been exceeded."	A configuration file can contain a maximum of 64 jobs or 64 job slots used.
"The message start address is invalid or contains incorrect data."	Either the e-mail overhead block is not present at the e-mail start address, or else the e-mail overhead block contains invalid data in one or more of its five words.
"The module firmware is not a known type."	Verify that you are connected to a module that supports the TCP/IP firmware.

Table B-1 Configurator Error Messages (continued)

Error Message	Possible Causes and Corrective Actions
"The module has returned an error."	Try again. If the problem persists, contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"The network IP address has not been set."	Use the **Edit > Set Network Parameters** menu command to enter a valid, non-null IP address (between 1.0.0.0 and 223.225.225.225) for the module.
"The paste job operation has failed."	Try again. This error usually occurs when the Clipboard is empty.
"The PLC remote data source memory type must match the PLC remote data destination memory type."	Select matching memory types for the source and destination.
"The PLC returned error code: 80*xx*."	Examine the list of error codes in Table B-2.
"The port field contains invalid data. Valid range is 1-65535."	Enter a value within the range 1 to 65,535.
"The port field contains invalid data. Valid range is 1024-65535."	Enter a value within the range 1,024 to 65,535.
"The remote IP field contains invalid data. Valid range is 1.0.0.0 - 223.255.255.255."	Enter an IP address that is in range.
"The remote IP field contains invalid data. Valid range is 1.0.0.0 - 223.255.255.255 or 0.0.0.0."	Enter an IP address that is in range, or enter 0.0.0.0 to set the IP address parameter to null.
"The remote port field contains invalid data. Valid range is 1024-65535."	Enter a value within the range 1,024 to 65,535.
"The selected firmware file contains invalid data."	The firmware upgrade file is corrupt or the wrong format. Contact the Siemens Technical Support hotline at 800-333-7421 or 423-262-2522.
"The selected job must be configured."	Click the Configure button in the dialog box and configure the job.
"The selected job name is already defined."	Use a different name.
"The '*start*' address field must be specified."	You cannot calculate e-mail message data, read it from the PLC, or write it to the PLC without first specifying a start address.

Table B-1 Configurator Error Messages (continued)

Error Message	Possible Causes and Corrective Actions
"The start bit field contains invalid data. Use the form "address.bit". Valid range of address is 5–7 and bit 1–16."	Enter a start bit that is in range. Use the correct form: for example, 5.1 for WY5.1.
"The '*start*' data type or address is invalid. Use the form 'datatype address'."	Only V- and K-memory are supported for e-mail. Valid examples are V100, K100.
"The timeout field contains invalid data. Valid range is 0–655."	Enter a value within the range 0 to 655.
"The transfer operation was unsuccessful."	The transfer (download) from the PC to the module did not succeed. The previous configuration, if any, remains in effect in the module.
"The V status field contains invalid data."	Enter a value within the range V1 to V999424.

NOTE: Error messages other than those listed in this table may be displayed; however, they are self-explanatory.

B.2 Status Word Error Codes

Many of the jobs available on the SIMATIC 505 TCP/IP module include a status word. Table B-2 shows the values that can appear in the status word after a job terminates.

Table B-2 Error Codes Written to Status Word in V-Memory

Status Code (Hex)	Description
0	Job executed successfully, no error.
80**xx**	**xx** is task code error response from attached PLC. These errors usually indicate a problem in accessing data in the attached PLC. Most of these will never occur, but for completeness all are shown. **01** - Reset current transaction **02** - Address out of range; other than ladder logic **03** - Requested data not found **04** - Illegal task code request (for example, task code not supported) **05** - Request exceeds available memory **06** - Diagnostic fail upon power up **07** - Fatal error detected **08** - Keylock protect error **09** - Incorrect amount of data sent with request **0A** - Illegal request in current operational mode **0B** - Network was not deleted **0C** - Attempted write operation did not verify **0D** - Illegal number of ASCII characters received **0E** - Illegal request when running from EPROM **0F** - Data not inserted **10** - Data not written **11** - Invalid data sent with the command **12** - Invalid operation with NIM local/remote mode (obsolete) **13** - The store and forward buffer is busy **14** - No response from the Special Function module **15** - Illegal instruction found in program memory on a Program to Run transition **16** - Attempted write to a protected variable (for example, TCC, TCP) **17** - No response from PLC (for example, single scan not performed) **18** - Requested memory size exceeds total available memory **19** - Requested memory size is not multiple of block allocation size **1A** - Requested memory size is less than minimum defined value **1B** - Requested memory size is larger than maximum defined value **1C** - PLC busy - cannot complete the requested operation **1D** - Comm error in HOLD mode - transition to Run not allowed **1E** - Port Lockout is active (see Task Code 48) **1F** - Attempting to delete active program via reconfiguration **20** - Program load in progress or invalidated. **21** - I/O configuration error - too many points **22** - I/O configuration conflict **23** through **3E** - Spare **3F** - Bus error detected

Table B-2 Error Codes Written to Status Word in V-Memory (continued)

Status Code (Hex)	Description
80**xx**, continued	**40** - Operating system error detected **41** - Invalid control block type **42** - Control block number out of range **43** - Control block does not exist **44** - Control block already exists **46** - Offset out of range **47** - Arithmetic error detected while writing Loop or Analog-Alarm parameters **48** - Invalid SF Program type **49** - Instruction number or RAMP/SOAK Step Number out of range **4A** - Attempt to access an integer-only variable as a real **4B** - Attempt to access a real-only variable as an integer **4C** - Task code buffer overflow — too much data requested **4D** - Control block size error **4E** - Attempt to write a read-only variable (for example, X, WX, or DCC) **4F** - Invalid variable data type for this operation **50** - Task code request buffer too large **51** - Invalid SF Statement size specified **52** - Invalid return value **53** - Attempt to execute a cyclic statement in a non-cyclic SF Program **54** - Control block is disabled **55** - Control block is not disabled **56** - Attempt to perform an FTSR-OUT SF Statement on an empty FIFO **57** - Attempt to perform an FTSR-IN SF Statement on a full FIFO **58** - Stack overflow while evaluating a MATH, IF-THEN, or IMATH expression **59** - Maximum SF Subroutine nesting level exceeded **5A** - Arithmetic overflow **5B** - Invalid operator in an IF, MATH, or IMATH expression **5C** - S-Memory overflow **5D** - Attempt to divide by zero (IMATH statement) **5E** - FIFO is incompatible with FTSR-IN or FTSR-OUT statement **5F** - FIFO is invalid **60** - Invalid data type code **61** - RAMP/SOAK step type mismatch **62** - Invalid code **63** - Application is unassigned **64** - State change failed **65** - Required application not present **66** - Error compiling an SFPGM or SFSUB **67** - Ladder memory corrupted, reload program **68** through 6**F** - Spare **70** through **8F** - Reserved for 575 inter-board communication **90** through **FF** - Spare

Table B-2 Error Codes Written to Status Word in V-Memory (continued)

Status Code (Hex)	Description
9000	Invalid request to PLC. This is a software error in the module rather than a user error, and should never occur.
9001	Memory type request from PLC is not defined for any SIMATIC 505 PLC. This error can only be caused by a Read/Write access from a 3rd party device that requests an undefined plcTT type. Otherwise, the Configurator guards against the use of undefined data types when configuring communications between 505 PLCs. If the accessed plcTT type is unsupported or not configured in the attached PLC, error code 9003 occurs (see below).
9002	Reserved for invalid request for PLC mode. Currently, module does not produce this error.
9003	Memory type requested from PLC is not supported or not configured. For example, some PLCs do not support loop data types. It also occurs if the requested data type is not configured in the PLC (for example, unconfigured analog alarm data types). Access of data types undefined for any 505 PLC results in error 9001. (Active station: F002)
9004	Starting memory address accessed in PLC is invalid. This occurs if memory type is defined, supported, and configured, but the starting address is out of range for the data type. For example, if the starting address is V27000 and only 52K bytes of V-memory are configured (allowing a maximum V-address of 26,624), then this error occurs. If you reconfigured the PLC memory after installing and initializing the SIMATIC 505 TCP/IP module, reset the SIMATIC 505 TCP/IP module so that it can read the new memory configuration in the PLC. (Active station: F003)
9005	Memory type cannot be written since it is read-only. See the Access column of Table C-1. (Active station: F00C)
9006	PLC communication time-out. Due to extremely heavy load on the PLC or module, the data access to the PLC took longer than expected. This error rarely occurs, but if it does, retry the job.
9007	Starting memory address OK, but the number of items requested are causing invalid memory access. For example, this error occurs if 2,000 words of V-memory starting at V26000 are requested but only 52K bytes of V-memory are configured (V26624 is highest configured address). If you reconfigured PLC memory after installing and initializing the SIMATIC 505 TCP/IP module, reset the SIMATIC 505 TCP/IP module so that it can read the new memory configuration in the PLC. (Active station: F005)
9008	Reserved for error “Slot into which SIMATIC 505 TCP/IP module is placed is not configured.” This error does not explicitly appear for any job. An unconfigured base is only indicated on the Configurator Module Status display. If the base is not configured, the module is unable to detect the WY Job Start Bits required for Read Active, Write Active, Send, and other jobs; however, Read Passive and Write Passive jobs still respond, although possibly at a greatly reduced throughput.

Table B-2 Error Codes Written to Status Word in V-Memory (continued)

Status Code (Hex)	Description
A00x	The A00x error codes indicate an internal software error in the module which should never occur during normal module operation.
B220–B599	Error codes 220–599 returned from SMTP server as defined in RFC 821, Section 4.2.1 (for E-Mail jobs only).
B600	Unable to open socket for connection to SMTP server.
B604	Unable to connect to SMTP server.
B605	No response after connect from SMTP server.
B607	MAIL FROM: not terminated correctly with <cr-lf> (hex sequence 0D0A).
B608	RCPT TO: not terminated correctly with <cr-lf> (hex sequence 0D0A).
B612	Error when sending SMTP data to SMTP server.
B613	Error when receiving SMTP response from SMTP server.
B614	Response from SMTP server did not end in <cr-lf> (hex sequence 0D0A).
BB01	Active job failed, due possibly to loss of connection to remote host or other transient errors. (Remote host may have timed out the connection if data is not transferred often enough.) The active job should be triggered again so that it attempts to re-establish the connection and send the data.
BD02	Could not currently establish a successful connection with the corresponding remote host. The error can result from a number of causes: an unattached Ethernet cable or a configuration error at either end of the connection is the usual cause. This error can also occur when a re-connect occurs too soon (less than 60 seconds) after a disconnect, especially if the job specifies a fixed local port. (The disconnect could be due to a previous job error, such as 804E.) The error can also occur if numerous jobs attempt simultaneously to connect.
BD03	The upper limit for number of jobs and/or connections has been reached. Remove unneeded jobs, if possible, from your configuration, or install another 505-CP1434-TCP module to handle additional jobs/connections.
C080	Loss of connection (for example, remote host has disconnected). Try again.
D002	Unexpected connection indication received. This error could occur if an active job receives a connection request; for example, if two active jobs are incorrectly configured as partners.
D003	Unexpected confirmation received. Typically this indicates an internal software problem in the module. It may also be transient in nature; retrying the job may clear the error.
D004	More data received on job than expected. This occurs on a Receive job if the total number of bytes entered in the Configurator data list does not match the total number of bytes entered for the corresponding Send job. (The Configurator does not enforce a byte balance between Send and Receive jobs since they are specified in separate configuration files.)
D005	Less data received on job than expected. Same explanation as D004 above.
D006	Attempted to access an unopened SCI device. This error would indicate an internal software error in the module and should never occur.
D007	Unable to allocate sufficient memory in SIMATIC 505 TCP/IP module to perform the job. This error should only occur in a very heavily loaded module. Retrying the job may clear the error.

Table B-2 Error Codes Written to Status Word in V-Memory (continued)

Status Code (Hex)	Description
D008	Mismatch of bytes processed and expected size during job execution.
D00D	Bad HTB header on read or write request. This error is usually seen on the passive side when a Read Passive job is paired with a Write Active job or if a Write Passive job is paired with a Read Active job.
E001	Send, Read/Write Active, Read Remote, Write Remote, or Memory Exchange job timed out. This error occurs at the Active unit when the job does not complete in the specified time (1 to 655 seconds).
F00**x**	These errors appear only at the active side of Read or Write jobs, transmitted back by the corresponding passive job. The value of **x** indicates one of the following errors: **1** - Memory type accessed in the remote PLC is unknown. **2** - Memory type requested from PLC is not supported or not configured. **3** - Starting memory address accessed in PLC is invalid. **4** - A PLC access time-out has occurred at the remote PLC. **5** - Starting memory address OK—excessive length causing invalid memory access. **6** - Error reserved for unconfigured base at the remote PLC. **7** - Invalid request performed at the remote PLC. **8** - Undefined error occurred at the remote PLC. **9** - General error code for all non-900x errors at the remote passive job. No specific information is implied. **B** - Invalid request for PLC mode change occurred at the remote passive job. **C** - Invalid PLC access, for example, trying to write a read-only memory type.
F0**xx**	The **xx** value is returned by the TCP or UDP server to indicate a remotely detected error (xx-range from 6E to C7). The following values are returned by the SIMATIC 505 TCP/IP module in the Read Remote, Write Remote, and Memory Exchange jobs (which use CAMP protocol) as V-memory status words. Other implementations of TCP or UDP server (for example, SIMATIC 505 CP2572) may return an expanded set. See the specific camp server documentation. The Configurator should preclude most of these errors. **74** - Bad checksum on CAMP job response **75** - Invalid Type character in CAMP message **76** - Non-hex-ASCII (0-9, A-F) character in message **77** - Odd number of characters in received CAMP message **81** - No data written or read **82** - Invalid word count in message **83** - Invalid memory address 0 used **8F** - Invalid request to read 0 words **90** - Unsupported address class (tt) in message **91** - Too much data requested to fit in CAMP message **AC** - Bad response for reading CAMP data from remote CAMP server **AD** - Bad response when writing CAMP data to remote CAMP server

B.3 Troubleshooting the LED Indicators of the Module

Module Status

Table B-3 provides a summary of the module status according to the LED indicators under power-up or operating conditions, and the action required to correct the error conditions. (If any LEDs are still off or blinking more than 25 seconds after power-up, the module is non-functional. For full information about LED indicators, see Table 2-7.)

Table B-3 Troubleshooting Chart

○ Off	● On	☀ Flashing

LED Status	Module Status	Corrective Action
MOD GOOD ☀ MOD READY ○	At power-up or reset of the SIMATIC 505 TCP/IP module (during first 25 seconds): MOD GOOD flashing and MOD READY off indicate a module hardware error detected.	Turn off power to the module. Reconnect power to determine if the problem is recurrent. If problem recurs on successive power-ups or resets, turn off power and remove module from base. Check for correct positioning of jumper (refer to Chapter 2 for correct jumper position). Reinstall module and power up again. If error condition continues, module has a hardware fatal error. Return module for repair or replacement.
MOD GOOD ○ MOD READY ☀	At power-up or reset of the SIMATIC 505 TCP/IP module (during first 25 seconds): MOD GOOD off and MOD READY flashing indicate a module software error detected.	The firmware in the flash memory of the module is corrupted. Use the Configurator to reload the firmware. You may need to contact your distributor to obtain the most recent firmware. If the error condition persists, call your Siemens distributor for technical assistance. If you need assistance in contacting your distributor, phone 800-964-4114.
MOD GOOD ● MOD READY ☀	At power-up or reset of the SIMATIC 505 TCP/IP module (after first 25 seconds): MOD GOOD on and MOD READY flashing indicate a module initialization error detected.	The module failed to initialize because configuration parameters of the module are invalid or not present. (Note: A new module installed for the first time powers up this way until it has been configured by the Configurator.) Connect the module to a PC with the Configurator installed. Create a configuration and download it to the module using the Transfer function. After the file has transferred, the module automatically resets. If this condition persists, there may be a hardware error. Connect the Configurator to module and select Module Status. If the status screen displays a hardware failure, contact your distributor for technical assistance.
MOD GOOD ☀ MOD READY ●	At power-up or reset of the SIMATIC 505 TCP/IP module (after first 25 seconds): MOD GOOD flashing and MOD READY on indicate a fault condition in the DC power supply detected.	Disconnect the cable from the Ethernet Bus Interface port, and reset the module. If the module resets successfully and no longer displays the LED pattern described, it indicates that the Ethernet port has a DC power overload. Take action to clear up power abnormalities on the network to ensure proper communication with the SIMATIC 505 TCP/IP module.

Appendix C

Data Types and NITP Communications

C.1 SIMATIC 505 Data Types

Table C-1 shows the general data types supported by the SIMATIC 505 TCP/IP module. (For loop data types, see Table C-2. For analog alarm data types, see Table C-3.)

Table C-1 General Data Types

Mnemonic	Name	Syntax	Type	Notes	PLCtt	Access	RBE
C	Control Relay	C <address>	Boolean		05	read/ write	no
CP	Control Relay Packed	CP <address>	packed 8 bits	1	08	r/w	yes
DCC	Drum Counter Current	DCC <drum>	unsigned 32	2	1B	read	no
DCP	Drum Counter Preset	DCP <drum> . <step>	unsigned 16	3	12	r/w	yes
DSC	Drum Step Current	DSC <drum>	unsigned 16		11	r/w	yes
DSP	Drum Step Preset	DSP <drum>	unsigned 16		10	r/w	yes
G	Global Variable (575 only)	G <address>	signed 16	4	EF	r/w	yes
GA	Global Variable App. A (575 only)	GA <address>	signed 16	5	EE	r/w	yes
GB	Global Variable App. B (575 only)	GB <address>	signed 16	5	ED	r/w	yes
GC	Global Variable App. C (575 only)	GC <address>	signed 16	5	EC	r/w	yes
GD	Global Variable App. D (575 only)	GD <address>	signed 16	5	EB	r/w	yes
GE	Global Variable App. E (575 only)	GE <address>	signed 16	5	EA	r/w	yes
GF	Global Variable App. F (575 only)	GF <address>	signed 16	5	E9	r/w	yes
GG	Global Variable App. G (575 only)	GG <address>	signed 16	5	E8	r/w	yes
GH	Global Variable App. H (575 only)	GH <address>	signed 16	5	E7	r/w	yes
GI	Global Variable App. I (575 only)	GI <address>	signed 16	5	E6	r/w	yes
GJ	Global Variable App. J (575 only)	GJ <address>	signed 16	5	E5	r/w	yes
GK	Global Variable App. K (575 only)	GK <address>	signed 16	5	E4	r/w	yes
GL	Global Variable App. L (575 only)	GL <address>	signed 16	5	E3	r/w	yes
GM	Global Variable App. M (575 only)	GM <address>	signed 16	5	E2	r/w	yes
GN	Global Variable App. N (575 only)	GN <address>	signed 16	5	E1	r/w	yes
GO	Global Variable App. O (575 only)	GO <address>	signed 16	5	E0	r/w	yes
GP	Global Variable App. P (575 only)	GP <address>	signed 16	5	DF	r/w	yes
GQ	Global Variable App. Q (575 only)	GQ <address>	signed 16	5	DE	r/w	yes
GR	Global Variable App. R (575 only)	GR <address>	signed 16	5	DD	r/w	yes
GS	Global Variable App. S (575 only)	GS <address>	signed 16	5	DC	r/w	yes
GT	Global Variable App. T (575 only)	GT <address>	signed 16	5	DB	r/w	yes
GU	Global Variable App. U (575 only)	GU <address>	signed 16	5	DA	r/w	yes
GV	Global Variable App. V (575 only)	GV <address>	signed 16	5	D9	r/w	yes
GW	Global Variable App. W (575 only)	GW <address>	signed 16	5	D8	r/w	yes
GX	Global Variable App. X (575 only)	GX <address>	signed 16	5	D7	r/w	yes
GY	Global Variable App. Y (575 only)	GY <address>	signed 16	5	D6	r/w	yes
GZ	Global Variable App. Z (575 only)	GZ <address>	signed 16	5	D5	r/w	yes

Table C-1 General Data Types (continued)

Mnemonic	Name	Syntax	Type	Notes	PLCtt	Access	RBE
K	Constant Memory	K <address>	signed 16		02	r/w	yes
KF	Constant Memory (2 addresses)	KF <address>	real 32	6	1D	r/w	yes
KL	Constant Memory (2 addresses)	KL <address>	signed 32	6	n/a	r/w	yes
STW	Status Word	STW <address>	unsigned 16		1A	r/w	yes
TCC	Timer/Counter Current	TCC <timer>	unsigned 16		0F	r/w	yes
TCP	Timer/Counter Preset	TCP <timer>	unsigned 16		0E	r/w	yes
V	Variable Memory	V <address>	signed 16		01	r/w	yes
VF	Variable Memory (2 addresses)	VF <address>	real 32	6	1C	r/w	yes
VL	Variable Memory (2 addresses)	VL <address>	signed 32	6	n/a	r/w	yes
VMM	VME A24 Space (575 only)	VMM <hex-address>	signed 16	7	D3	r/w	no
VMS	VME A16 Space (575 only)	VMS <hex-address>	signed 16	7	D4	r/w	no
WX	Word Input	WX <address>	signed 16		09	r/w	yes
WY	Word Output	WY <address>	signed 16		0A	r/w	yes
X	Discrete Input	X <address>	Boolean		03	r/w	no
XP	Discrete Input Packed	XP <address>	packed 8 bits	1	06	r/w	yes
Y	Discrete Output	Y <address>	Boolean		04	r/w	no
YP	Discrete Output Packed	YP <address>	packed 8 bits	1	07	r/w	yes

Note 1 The CP, XP, and YP data types are 8 control relay bits packed as an unsigned 8-bit integer. For example, if you assign values to CP1 through CP5, the values are sent as a single byte: the first five bits contain the values that you assigned, and the remaining three bits are set to zero. By contrast, if you use C, X, or Y memory, each bit assignment is sent as a Boolean byte. So it would take 5 bytes to send C1 through C5. (For each C, X, or Y bit, if the bit is on, it is sent as a byte with a value of FF; if the bit is off, it is sent as a byte with a value of 0.)

Note 2 This is a read-only data type. If a job attempts to write to read-only memory, the PLC breaks the connection to the module. Do not configure jobs to write to read-only data types.

Note 3 The DCP address is an address containing a drum number and a step number. The format is as follows:

<DCP> <drum address>.<step number>

For example: DCP 5.1 indicates the Drum Count Preset value of Drum 5, Step 1.

Note 4 The G memory partitions (in the 575 PLC) are global memories accessible by the local PLC as well as by other bus masters. Access to the G memory partition associated with the local PLC is provided through PLCtt code EF (hex). The addressing of this partition is 24-bit zero relative; (that is, the first element of G memory is G1, which has an offset of 0).

Note 5 Each PLC in a 575 system may contain one or more application process. These applications are identified by a single letter ranging from Application A through Application Z. Associated with each application is a G Memory partition. Access to the G memory partitions (in the 575 PLC) associated with a PLC application is provided through PLCtt codes D5 through EE (hex). Addressing (PLCtt offset) of these partitions is 24-bit zero relative.

Note 6 The VF and KF types are overlaid on the V and K memory directly. Note that VF and KF addressing makes use of word-oriented addressing even though VF and KF refer to double-word (real) data elements. For example, VF100 references the IEEE single-precision floating-point value stored at locations V100 and V101. The VL and KL data types are only available as a selection in the SIMATIC 505 OPC Server; they cannot be entered in the Configurator. VL and KL refer to double-word data elements accessed as 32-bit signed integer values. For example, VL100 references the 32-bit signed integer value stored at locations V100 and V101.

Note 7 VMS corresponds to VME address modifier 29 (short non-privileged access). VMM corresponds to VME address modifier 39 (standard non-privileged data access). The access size is always 16 bits. The offset field contains the relative word of the specified address space. VMM and VMS addresses must be even.

Table C-2 shows the loop data types supported by the SIMATIC 505 TCP/IP module.

Table C-2 Loop Data Types

Mnemonic	Name	Syntax	Type	Notes	PLCtt	Access	RBE
LACK	Loop Acknowledge Flag	LACK <loop>	unsigned 16		4F	read/ write	no
LADB	Loop Alarm Deadband	LADB <loop>	real 32		37	r/w	no
LADBR	Loop Alarm Deadband Raw	LADBR <loop>	unsigned 16		47	r/w	no
LCF	Loop C-Flags	LCF <loop>	unsigned 32		2E	r/w	no
LCFH	Loop C-Flags High Word	LCFH <loop>	unsigned 16		4B	r/w	no
LCFL	Loop C-Flags Low Word	LCFL <loop>	unsigned 16		4C	r/w	no
LERR	Loop Error	LERR <loop>	real 32	1	30	read	no
LERRR	Loop Error Raw	LERRR <loop>	signed 16	1	3F	read	no
LHA	Loop High Alarm Limit	LHA <loop>	real 32		23	r/w	no
LHAR	Loop High Alarm Limit Raw	LHAR <loop>	unsigned 16		38	r/w	no
LHHA	Loop High-High Alarm Limit	LHHA <loop>	real 32		32	r/w	no
LHHAR	Loop High-High Alarm Limit Raw	LHHAR <loop>	unsigned 16		40	r/w	no
LKC	Loop Gain	LKC <loop>	real 32		20	r/w	no
LKD	Loop Derivative Gain Limiting Coefficient	LKD <loop>	real 32		4D	r/w	no
LLA	Loop Low Alarm Limit	LLA <loop>	real 32		24	r/w	no
LLAR	Loop Low Alarm Limit Raw	LLAR <loop>	unsigned 16		39	r/w	no
LLLA	Loop Low-Low Alarm Limit	LLLA <loop>	real 32		33	r/w	no
LLLAR	Loop Low-Low Alarm Limit Raw	LLLAR <loop>	unsigned 16		41	r/w	no
LM	Loop Mode	LM <loop>	unsigned 16		1F	r/w	no
LMN	Loop Output (percent)	LMN <loop>	real 32		2C	r/w	no
LMNR	Loop Output Raw	LMNR <loop>	unsigned 16		3D	r/w	no
LMX	Loop Bias	LMX <loop>	real 32		31	r/w	no
LMXR	Loop Bias Raw	LMXR <loop>	signed 16		48	r/w	no
LODA	Loop Orange Deviation Alarm Limit	LODA <loop>	real 32		28	r/w	no
LODAR	Loop Orange Deviation Alarm Limit Raw	LODAR <loop>	unsigned 16		3B	r/w	no
LPV	Loop Process Variable	LPV <loop>	real 32		25	r/w	no
LPVH	Loop Process Variable High Limit	LPVH <loop>	real 32		26	r/w	no
LPVL	Loop Process Variable Low Limit	LPVL <loop>	real 32		27	r/w	no

Table C-2 Loop Data Types (continued)

Mnemonic	Name	Syntax	Type	Notes	PLCtt	Access	RBE
LPVR	Loop Process Variable Raw	LPVR <loop>	unsigned 16		3A	r/w	no
LRCA	Loop Rate-of-Change Alarm Limit	LRCA <loop>	real 32		34	r/w	no
LRSF	Loop Ramp/Soak Status Flags	LRSF <loop>	unsigned 16	2	2F	r/w	yes
LRSN	Loop Ramp/Soak Step Number	LRSN <loop>	unsigned 16		4E	r/w	no
LS	Loop Status	LS <loop>	unsigned 16	1	1E	read	yes
LSP	Loop Setpoint	LSP <loop>	real 32		2B	r/w	no
LSPH	Loop Setpoint High Limit	LSPH <loop>	real 32		35	r/w	no
LSPHR	Loop Setpoint High Limit Raw	LSPHR <loop>	unsigned 16		4A	r/w	no
LSPL	Loop Setpoint Low Limit	LSPL <loop>	real 32		36	r/w	no
LSPLR	Loop Setpoint Low Limit Raw	LSPLR <loop>	unsigned 16		49	r/w	no
LSPR	Loop Setpoint Raw	LSPR <loop>	unsigned 16		3E	r/w	no
LTD	Loop Rate Time (minutes)	LTD <loop>	real 32		22	r/w	no
LTI	Loop Reset Time (minutes)	LTI <loop>	real 32		21	r/w	no
LTS	Loop Sample Rate (seconds)	LTS <loop>	real 32		2A	r/w	no
LVF	Loop V-Flags	LVF <loop>	unsigned 16	2	2D	r/w	yes
LYDA	Loop Yellow Deviation Alarm Limit	LYDA <loop>	real 32		29	r/w	no
LYDAR	Loop Yellow Deviation Alarm Limit Raw	LYDAR <loop>	unsigned 16		3C	r/w	no

Note 1 This is a read-only data type. If a job attempts to write to read-only memory, the PLC breaks the connection to the module. Do not configure jobs to write to read-only data types.

Note 2 These flags have a “control” part and a “status” part. When written, only the control part is written. When read, the status part is returned with the control part set to zero.

Table C-3 shows the analog alarm data types supported by the SIMATIC 505 TCP/IP module.

Table C-3 Analog Alarm Data Types

Mnemonic	Name	Syntax	Type	Notes	PLCtt	Access	RBE
AACK	Alarm Acknowledge Flag	AACK <alarm>	unsigned 16		73	read/ write	no
AADB	Alarm Deadband	AADB <alarm>	real 32		61	r/w	no
AADBR	Alarm Deadband Raw	AADBR <alarm>	unsigned 16		68	r/w	no
ACF	Alarm C-Flags	ACF <alarm>	unsigned 32		5A	r/w	no
ACFH	Alarm C-Flags High Word	ACFH <alarm>	unsigned 16		71	r/w	no
ACFL	Alarm C-Flags Low Word	ACFL <alarm>	unsigned 16		72	r/w	no
AERR	Alarm Error	AERR <alarm>	real 32	1	5B	read	no
AERRR	Alarm Error Raw	AERRR <alarm>	signed 16	1	69	read	no
AHA	Alarm High Alarm Limit	AHA <alarm>	real 32		50	r/w	no
AHAR	Alarm High Alarm Limit Raw	AHAR <alarm>	unsigned 16		62	r/w	no
AHHA	Alarm High-High Alarm Limit	AHHA <alarm>	real 32		5C	r/w	no
AHHAR	Alarm High-High Alarm Limit Raw	AHHAR <alarm>	unsigned 16		6A	r/w	no
ALA	Alarm Low Alarm Limit	ALA <alarm>	real 32		51	r/w	no
ALAR	Alarm Low Alarm Limit Raw	ALAR <alarm>	unsigned 16		63	r/w	no
ALLA	Alarm Low-Low Alarm Limit	ALLA <alarm>	real 32		5D	r/w	no
ALLAR	Alarm Low-Low Alarm Limit Raw	ALLAR <alarm>	unsigned 16		6B	r/w	no
AODA	Alarm Orange Deviation Alarm Limit	AODA <alarm>	real 32		55	r/w	no
AODAR	Alarm Orange Deviation Alarm Limit Raw	AODAR <alarm>	unsigned 16		65	r/w	no
APV	Alarm Process Variable	APV <alarm>	real 32		52	r/w	no
APVH	Alarm Process Variable High Limit	APVH <alarm>	real 32		53	r/w	no
APVL	Alarm Process Variable Low Limit	APVL <alarm>	real 32		54	r/w	no
APVR	Alarm Process Variable Raw	APVR <alarm>	unsigned 16		64	r/w	no
ARCA	Alarm Rate-of-Change Alarm Limit	ARCA <alarm>	real 32		5E	r/w	no
ASP	Alarm Setpoint	ASP <alarm>	real 32		58	r/w	no
ASPH	Alarm Setpoint High Limit	ASPH <alarm>	real 32		5F	r/w	no
ASPHR	Alarm Setpoint High Limit Raw	ASPHR <alarm>	unsigned 16		70	r/w	no
ASPL	Alarm Setpoint Low Limit	ASPL <alarm>	real 32		60	r/w	no
ASPLR	Alarm Setpoint Low Limit Raw	ASPLR <alarm>	unsigned 16		6F	r/w	no
ASPR	Alarm Setpoint Raw	ASPR <alarm>	unsigned 16		67	r/w	no
ATS	Alarm Sample Rate (seconds)	ATS <alarm>	real 32		57	r/w	no

Table C-3 Analog Alarm Data Types (continued)

Mnemonic	Name	Syntax	Type	Notes	PLCtt	Access	RBE
AVF	Alarm V-Flags	AVF <alarm>	unsigned 16	2	59	r/w	yes
AYDA	Alarm Yellow Deviation Alarm Limit	AYDA <alarm>	real 32		56	r/w	no
AYDAR	Alarm Yellow Deviation Alarm Limit Raw	AYDAR <alarm>	unsigned 16		66	r/w	no

Note 1 This is a read-only data type. If a job attempts to write to read-only memory, the PLC breaks the connection to the module. Do not configure jobs to write to read-only data types.

Note 2 This flag has a “control” part and a “status” part. When written, only the control part is written. When read, the status part is returned with the control part set to zero.

C.2 SIMATIC S7 Data Types

Table C-4 shows the S7 data types that are supported for the SIMATIC 505 TCP/IP module.

Table C-4 S7 Data Types

International Mnemonic	German Mnemonic	Name	Syntax	Type
DB	DB	Data Block	DB <block> DW <address>	unsigned 16
IB	EB	Input Byte	IB <address>	unsigned 8
MB	MB	Memory Byte	MB <address>	unsigned 8
QB	AB	Output Byte	QB <address>	unsigned 8

Reading or Writing Discrete Data Types

When discrete data types (X, Y, and C) are read or written from the PLC, they can be accessed as either “Boolean” or “packed” (see Table C-1).

If you access a discrete data type as Boolean (X, Y, or C), each discrete item is transferred as a separate byte, as shown in Figure C-1:

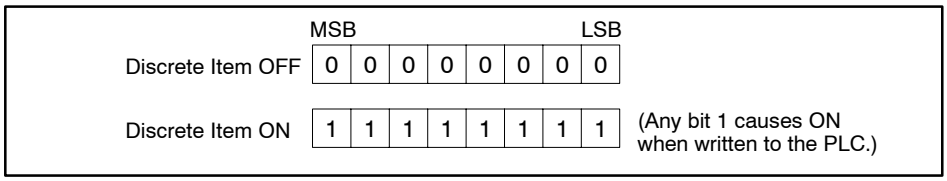


Figure C-1 Boolean Discrete Data Type Format

If you access a discrete data type as packed (XP, YP, or CP), up to 8 discrete items can occupy each byte transferred, as shown in Figure C-2:

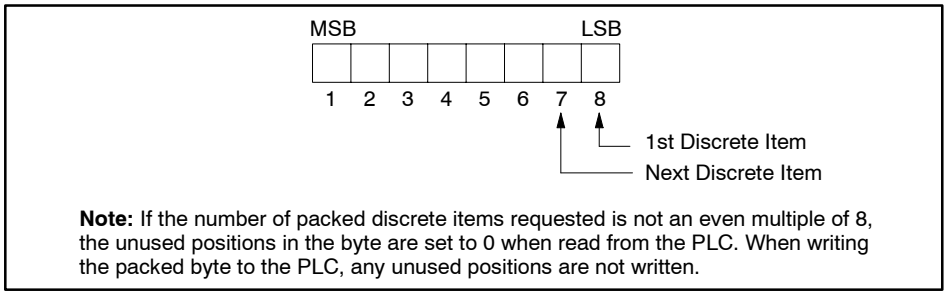


Figure C-2 Packed Discrete Data Type Format

When packed data items are transferred to/from integer memory (e.g., V), the following mapping results. For example, a Write Active job specifies Local V10, 1 and Remote CP100, 14, as shown in Figure C-3.

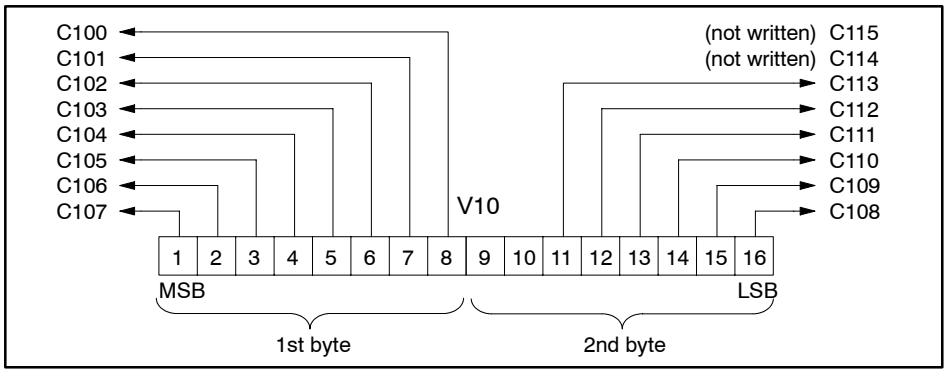


Figure C-3 Packed Discrete Data Type Transfer

Byte Balance between Local and Remote PLCs

When defining a point-to-point job configuration between a 505 local PLC and another 505 remote PLC, or a remote S7, or a 505 system with a 505-CP2572 module, you must make sure that the total number of bytes entered for the Local PLC matches the number of bytes entered for the Remote PLC. The Configurator does not allow a job that is not "byte balanced" (except for Send/Receive and Broadcast Send/Broadcast Receive jobs).

Any data type from Section C.1 can be used for either the local or remote PLC on the Configurator. If you select an S7 system for the remote PLC, you can use any data type listed in Section C.2. Each S7 word-based data type consists of 2 bytes.

To help illustrate the byte balance enforced by the Configurator, Table C-5 gives the number of bytes for each data type listed in Section C.1.

Table C-5 Number of Bytes per Data Type

PLC Representation	Bytes for N Items
Boolean	N x 1
Integer	N x 2
16-bit	N x 2
32-bit	N x 4
Real	N x 4
Packed	(N + 7)/8 (with remainder ignored)

As an example, if you select a quantity of 4 integer-sized data items for the Local PLC and if you choose a real data type for the Remote PLC, then the Configurator requires you to enter the quantity of 2 for the real data type to obtain a byte balance (4 x 2 integer items = 2 x 4 real items, i.e., 8 = 8).

When packed data types (XP, YP, or CP) are used, maintaining byte balance is somewhat more complicated. For example, if 29 CPs are to be transferred, the number of bytes is (29 + 7)/8 = 4. You can balance this with 4 Boolean data types, 2 integer data types, or 1 real data type, etc. If you transfer 19 CPs, (19 + 7)/8 = 3 bytes, you can balance this with 3 Boolean data types or from 17 to 24 packed items.

NOTE: Write Remote, Read Remote, and Memory Exchange jobs require that their data blocks are an even number of bytes (word boundaries).

Maximum Data Block Sizes

When you specify the data to be transferred in the Job Configuration dialog box, the format is the following:

<memory type> <start address> , <number of items> (e.g. V101, 20)

The maximum number of items allowed depends on the memory type. A maximum total of 65,535 bytes can be transferred for most jobs. (Write Remote, Read Remote, Memory Exchange, Broadcast Send, and Broadcast Receive jobs have more restrictive limits.) Smaller data types (packed discretes or Boolean types) allow more items per transfer than larger data units (integer or real). The maximum limits for the different types are shown in Table C-6.

Table C-6 Maximum Number of Locations per Data Type

PLC Representation	Maximum Number of Items per Job
Packed	65535 bytes x 8 items per byte = 524,280
Boolean	65535 bytes x 1 item per byte = 65,535
16-bit, Integer	65535 bytes / 2 bytes per item = 32,767
32-bit, Real	65535 bytes / 4 bytes per item = 16,383

Of course, the number of items that can be transferred also depends on the memory capacity and configuration of the local and remote PLCs.

C.4 Data Entry Examples

Table C-7 gives examples of how to enter a starting address and count for different data types, and explains what size block results. Since different data types use different units (bits, bytes, words, double words), the same count value can result in blocks of differing sizes.

Table C-7 Data Entry Examples

Example	Description
C18, 10	Specifies 10 C bits starting at address 18 for a total of 10 bits (sent as 10 bytes that are Boolean, where the byte value is FF if the bit is on and 0 if the bit is off).
CP18, 10	Specifies 10 C bits that are packed starting at address 18 for a total of 10 bits (sent as 2 bytes, with 6 empty bits).
DCP12.6, 2	Specifies 2 DCP words starting at drum 12 step 6 for a total of 2 words (4 bytes).
V23, 4	Specifies 4 V-memory words starting at address 23 for a total of 4 words (8 bytes).
VF23, 4	Specifies 4 V-memory double words starting at address 23 for a total of 8 words (16 bytes).
VMM0AB24, 6	Specifies 6 VMM words starting at address 0AB24 hex for a total of 6 words (12 bytes).
LMN7, 4	Specifies 4 LMN reals starting at loop 7 (LMN7 to LMN10) for a total of 4 double words (16 bytes).
ASPR14, 4	Specifies 4 ASPR words starting at alarm 14 (ASPR14 to ASPR17) for a total of 4 words (8 bytes).
DB23 DW8, 6	Specifies 6 memory words starting at block 23 address 8 for a total of 6 words (12 bytes).
MB23, 12	Specifies 12 memory bytes starting at address 23 for a total of 12 bytes (12 bytes).

C.5 Using NITP in Network Applications

The SIMATIC 505 TCP/IP module is backward compatible with all SIMATIC 505–CP2572 network server protocols. All existing network client applications will work with no modification.

NOTE: For future applications, consider using the SIMATIC 505 OPC Server for TCP/IP (described in Chapter 5), in order to take advantage of the broadcast capability and report-by-exception variable updates.

A SIMATIC 505 TCP/IP module implements several complex socket protocols. Detailing all of them is beyond the scope of this appendix. The information in this section is offered to programmers familiar with the SIMATIC Non Intelligent Terminal Protocol (NITP), PLC programming, and Berkeley or Windows sockets.

The 575 Task Code manual on the SIMATIC 505 Electronic Manuals CD (PPX:505–CDMANUAL–x, where x=2 or higher) gives information about NITP task codes.

This appendix (Section C.5) discusses NITP and presents examples of two NITP task codes (9E and 9D) that are not documented in the 575 Task Code manual.

NITP Format

NITP is a simple, character-oriented method of data link communications using standard 7-bit ASCII codes. Both command and response messages consist of starting and ending delimiters, a character count or message length field, the body of the message, and an error-checking code field as shown in Figure C-4.

:	Character Count	Message Body	ECC	;

Figure C-4 Command and Response Messages

NITP Character Set

NITP uses a subset of standard ASCII codes (see Table C-8) to communicate with a wide variety of host devices, from ASCII terminals to more intelligent machines.

Table C-8 Standard ASCII Codes

7-bit ASCII Code	Displayed Character	7-bit ASCII Code	Displayed Character
30	0	39	9
31	1	3A	:
32	2	3B	;
33	3	41	A
34	4	42	B
35	5	43	C
36	6	44	D
37	7	45	E
38	8	46	F

NOTE: ASCII characters, except those in the NITP character set (for example, a carriage return or line feed), may be sent to control special network devices between the ending delimiter and the next beginning delimiter. The controller ignores these characters.

Hexadecimal values must be translated into two ASCII codes or characters: For example, 0E (hex) equals 30 (ASCII 0) 45 (ASCII E).

Message Delimiters

A colon (:) marks the beginning of a message and a semicolon (;) marks the end of a message. Any characters between a colon and the next semicolon are interpreted as valid; any characters between a semicolon and the next colon are ignored. This allows the host to use any parameters required by its software between lines of output. When transmitting data to the host, the PLC sends carriage return and line feed characters after the terminating semicolon in order to scroll the response on ASCII terminals. More intelligent host devices can be set to filter out the carriage return and line feed.

Character Count

To aid in error control, the colon (:) at the beginning of the message is followed by a two-character count field representing the total number of printable characters in the message, including the colon, character count, message body, error checking code, and the terminating semicolon. The two characters represent an 8-bit hexadecimal value so that a count of 50 characters is represented as 3332 (32 hex). The maximum length of NITP messages is 72 characters.

Message Body

The message body consists of ASCII character pairs from the NITP character set, each representing a single binary byte value.

Error-Checking Code

Following the message body is an ASCII four-character error-checking code (ECC) in the form of a 16-bit hexadecimal number that is included at the end of the message just before the semicolon terminator. The ECC is a checksum that is computed by both the sending and receiving stations according to the following procedure:

- Divide the character count and the message body into blocks of four characters, left-justified and zero-filled. The beginning and ending delimiters are not included in the calculation.
- Treat each block as a four-digit hexadecimal number.
- Sum the resulting numbers (blocks).
- Take the two's complement of the sum to get the ECC.

For example, consider a message whose body is the Read Word Memory Random Task Code (TC01) reading V9:

	01		00		00		00		08	
ASCII Codes	30	31	30	30	30	30	30	30	30	38

The total character count is calculated by adding together the number of characters in the message, the four characters for the ECC, the two characters for the character count, and the two characters used to delimit the message. The total character count for a message containing "0100000008" :

Starting Delimiter	:	=	01	(1)*
Character Count		=	02	(2)
Message Body - 0100000008		=	0A	(10)
Error-checking Code		=	04	(4)
Ending Delimiter	;	=	01	(1)
Total Character Count			12	(18)

* Numbers in parentheses are the corresponding decimal values.

After determining the total character count, insert the character count at the beginning of the message body as “120100000008”. The ECC is given by:

Two’s complement of (1201 + 0000 + 0008) = EDF7

so that the complete message is the character string:

:120100000008EDF7;

or

ASCII Codes	3A	31	32	30	31	30	30	30	30	30	30	30	38	45	44	46	37	3B
Characters	:	1	2	0	1	0	0	0	0	0	0	0	8	E	D	F	7	;

NOTE: If the number of characters in the concatenated character count and message body string is not evenly divisible by 4, then the fill characters “00” are added to the end of the string for use in the ECC calculation. These fill characters are not actually placed in the final message.

NITP Buffer Size Limitations

More than one block can be read or written in a single NITP command. The amount is limited by an NITP buffer size of 72 characters. This translates into the following limits.

For read commands:

- 6 blocks of data or fewer
- 15 integer values or fewer
- 7 real values or fewer
- 31 discrete values or fewer
- 248 packed discrete values or fewer

For write commands:

- 13 integers or fewer
- 6 real values or fewer
- 26 discrete values or fewer
- 208 packed discrete values or fewer

For example, if the NITP buffer contains 13 integers, the integers would consume 52 characters (since 4 is the size of an integer); the overhead from the data element count, PLCtt code, and offset consumes 10 characters; the NITP task code overhead consumes 10 characters.

The NITP buffer size maximum of 72 characters applies to both send and receive buffers. If you send a request to read more data than can fit in the receive buffer, the data is truncated to fit within the 72-character buffer maximum.

Table C-9 summarizes the NITP message structure:

Table C-9 NITP Message Structure

Field	Contents	No. of Characters
Beginning delimiter	Colon (:)	1
Character count	Hexadecimal	2
Body of Message	Hexadecimal	64 (maximum)
Error Checking Code	Hexadecimal	4
Terminator	Semicolon (;)	1

The message body mentioned above consists of command codes (task codes) followed by relevant parameters. White-space characters, carriage-control characters, and nulls should not be sent.

Table C-10 gives an example of the command codes and parameters for writing and reading NITP messages.

Table C-10 Command Codes and Parameters

Codes/Parameters	Description
9E	task code for write command or response to write
9D	task code for read command or response to read
nn	data element count
tt	data element PLCtt code; see Table C-1 through Table C-3
aaaaaaaa	data element offset [the data element address is equal to the offset plus 1]

Reading a Block

The following example shows a task code sent to the module in order to read V8 through V15 using NITP, and the response that the module makes.

Figure C-5 shows a read request for V8 through V15.

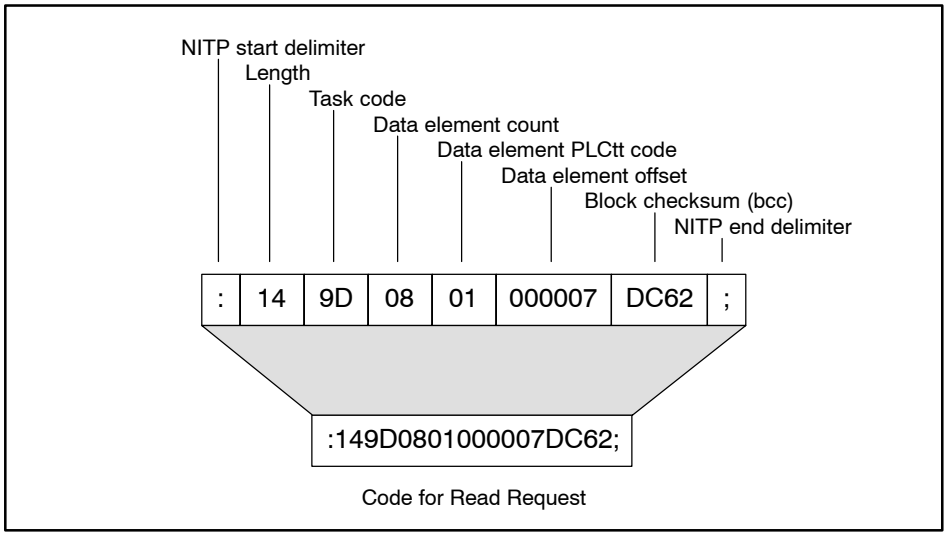


Figure C-5 Example Read Request

Figure C-6 shows the response to the read request (assuming V8 through V15 are zero).

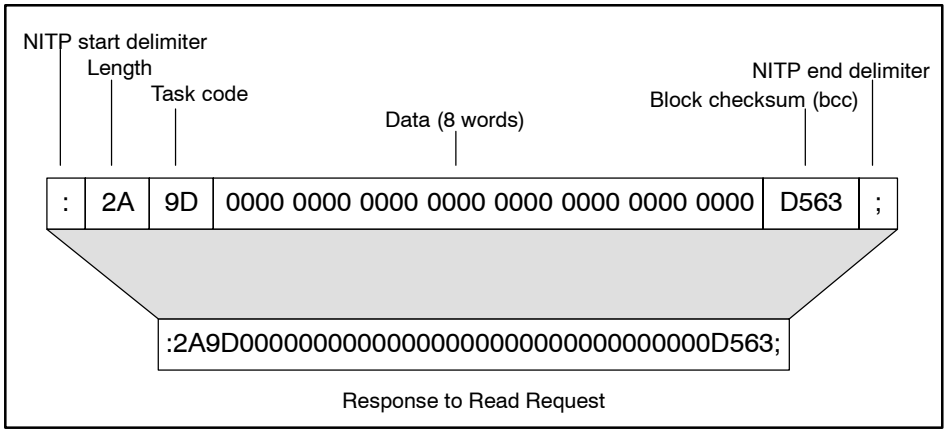


Figure C-6 Example Response

Writing a Block

The following example shows a task code to write hexadecimal values 1123 2123 and so on through 8123 to addresses V8 through V15 using NITP, and the response that the module makes.

Figure C-7 shows the write request to the module.

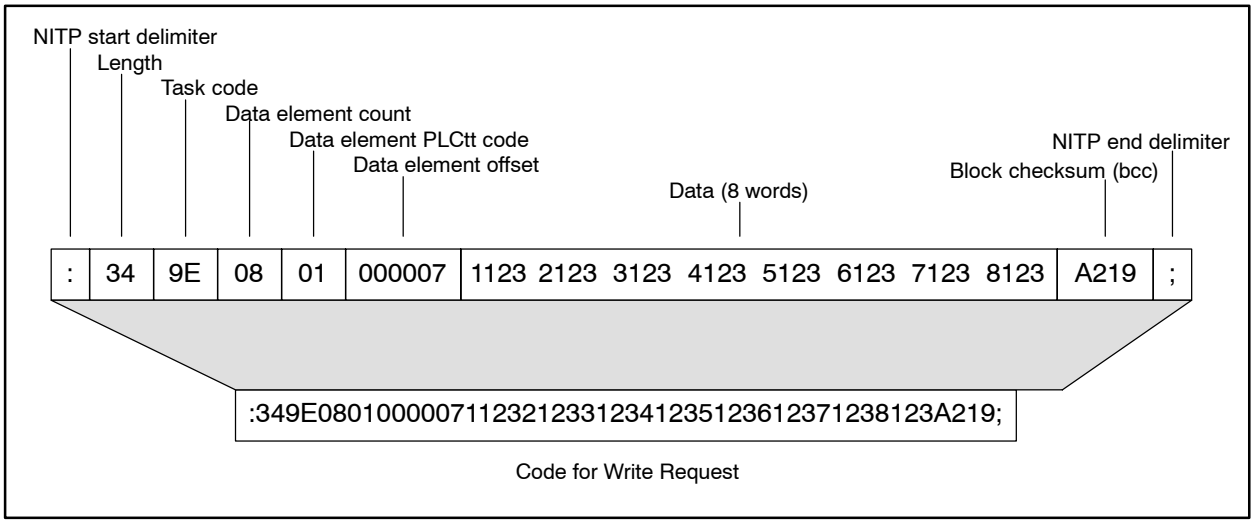


Figure C-7 Example Write Request

Figure C-8 shows the response from the module.

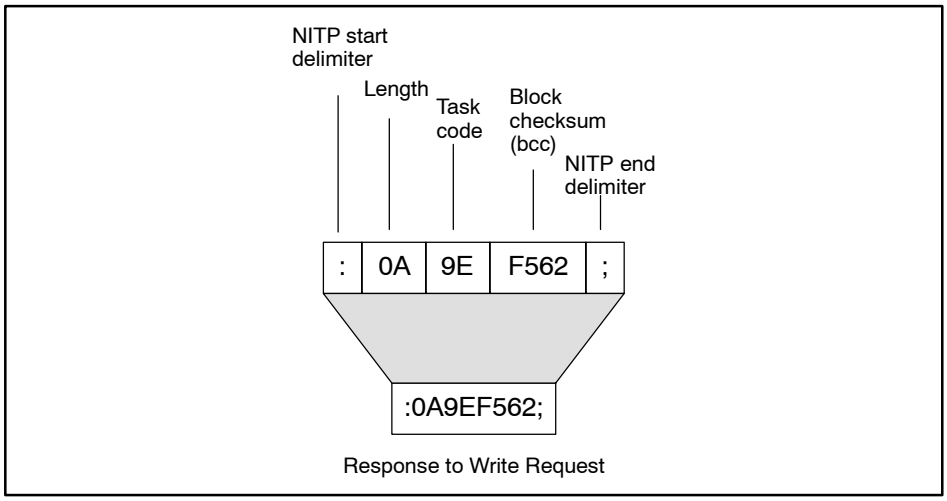


Figure C-8 Example Response

Reading Multiple Blocks

The following example shows a task code for reading multiple blocks or data areas.

In Figure C-9, the three blocks are V1 (zero offset), V3 (offset of 2 from V1), and V5 through V6 (offset of 4, count of 2).

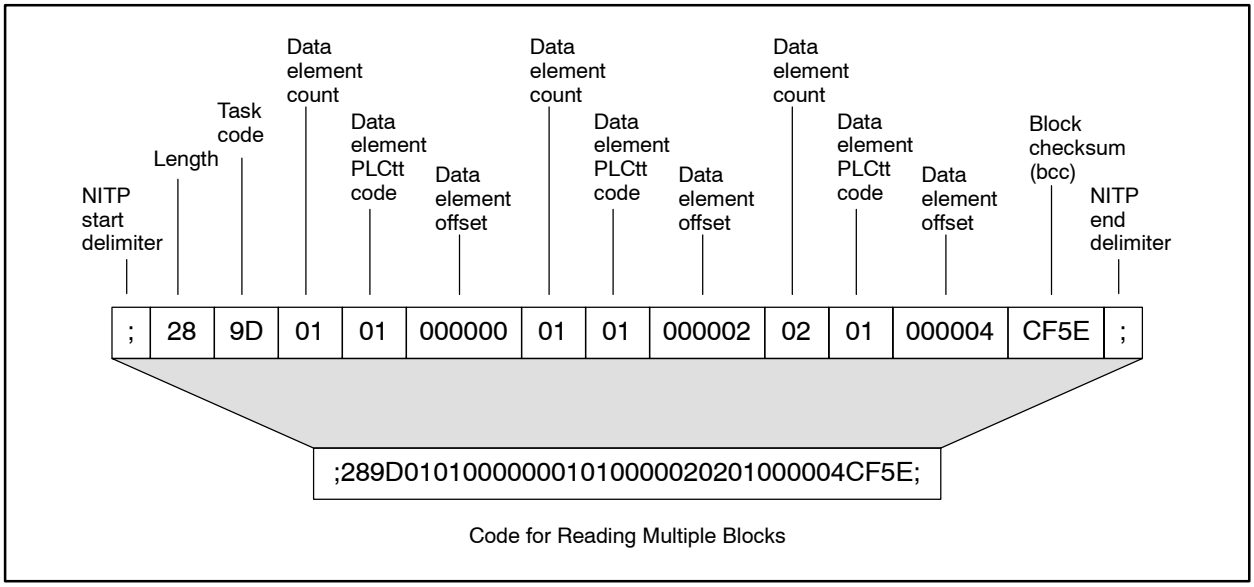


Figure C-9 Example of Reading Multiple Blocks

Figure C-10 shows the response from the module. V1 has a value of 29 (decimal) while V3, V5, and V6 are zero.

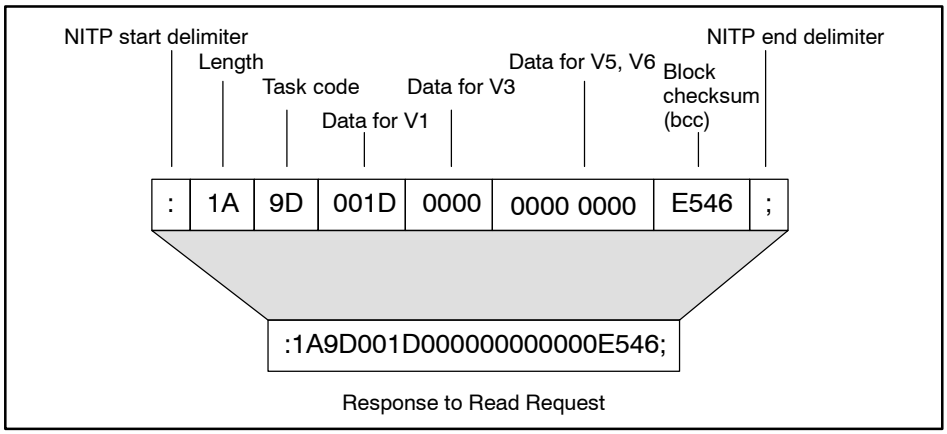


Figure C-10 Example Response

Appendix D

E-mail Jobs

D.1 Guidelines for Building Your Own E-Mail Messages

The E-mail job, as described in Section 4.13, allows you to configure a text message that is sent to a Simple Mail Transfer Protocol (SMTP) e-mail server when triggered by a Job Start bit. The SIMATIC 505 TCP/IP Module Configurator can automatically write your e-mail text to the PLC and count the data block sizes for you. It is not necessary for you to build your own message in a programming tool such as SoftShop and download it directly to the PLC, although you can do that. This section describes how to build and count the size of mail data blocks.

Locating the Starting Address of a Placeholder

The term “placeholder” refers to any text string that you use in an e-mail message block to reserve memory locations. The placeholder allows you to set aside a fixed amount of space within an e-mail message, space which can later be overwritten by a variable. You can use the Configurator to build an e-mail message that contains placeholder text and download the message data to the PLC. Then you create a program that writes variable information over the placeholder text. In order to write the variable information in the location of the placeholder, you must be able to locate the starting address of the placeholder within a given e-mail block. The counting information in this section can help you quickly locate the starting addresses of placeholders, and ensure that you do not violate any guidelines if you construct your own e-mail message data without using the Configurator.

Section D.2 provides a sample e-mail message with placeholders for variables, and shows how to use the counting information from this section to locate the starting addresses of the placeholders. Section D.3 provides sample SFSUBs that show how to write variables for the date (SW141–143), an integer value, and a real number to the sample e-mail message.

E-mail Overhead Block

The Configurator stores overhead information about the e-mail message in a block that precedes the three standard e-mail message data blocks (MAIL FROM, RCPT TO, and MESSAGE) that are specified by the Internet Request For Comment (RFC) specifications 821 and 822. This overhead block is used internally by the Configurator; it is not specified by RFC 821/822 and is not sent to the SMTP server. The e-mail overhead block begins at the Start Address of the e-mail job, and it is always 5 words long. All four blocks are contiguous in the Series 505 PLC.

You can use the Configurator to define your e-mail message, but if instead you choose to enter message data directly into the PLC, you need to know how to create the overhead block, which is a required component of E-mail jobs in the SIMATIC 505 TCP/IP module. Table D-1 describes the structure of the overhead block:

Table D-1 Structure of the Overhead Block

Position	Contents
First word	Size of overhead block: always 5.
Second word	Reserved for future use: always 0.
Third word	Contains the size, counted in words, of the MAIL FROM block of this e-mail message.
Fourth word	Contains the size, counted in words, of the RCPT TO block of this e-mail message.
Fifth word	Contains the size, counted in words, of the MESSAGE block of this e-mail message.

Creating MAIL FROM, RCPT TO, and MESSAGE Blocks

Table D-2 describes guidelines for creating the three standard blocks (MAIL FROM, RCPT TO, and MESSAGE) that comprise an e-mail message. Refer to the Internet Request For Comment (RFC) specifications 821 and 822 for full details. The RFCs, in conjunction with your local SMTP server, determine the required syntax of the three blocks. The module merely sends the data entered by the user in the three blocks to the server.

NOTE: Using a standard Telnet client, you can connect to your local SMTP server (at TCP port 25) and manually send e-mail messages to experiment with the various formats, following the guidelines outlined in RFC 821. For example, at the command line of your operating system, you would type the following command:

telnet 161.218.9.43 25

where 161.218.9.43 is the address of your SMTP server and 25 is the server TCP port.

Table D-2 E-mail Guidelines

Block	Syntax	Comments
MAIL FROM	MAIL FROM:<*HostNameofPLC*>[CR][LF]	Use this line to identify the sender of the e-mail (the fact that it comes from the SIMATIC 505 TCP/IP module). Recipients do not reply to this address. Terminate with a carriage return-line feed (hex value 0D0A).
RCPT TO	RCPT TO:<*EmailAddress*>[CR][LF]	Use this line to identify a recipient of the e-mail. Multiple RCPT TO statements are permitted. Each must be terminated by a carriage return-line feed (hex value 0D0A). If an e-mail address is specified here but not listed in a TO or CC statement in the mail header, it has the effect of a “blind” carbon copy to that recipient.

Table D-2 E-mail Guidelines (continued)

Block	Syntax	Comments
MESSAGE (Header)	DATE:*dd mon yy hh:mm:ss*[LF]	You can enter a date value (which is effectively "frozen" and may not correspond to the time that the e-mail is actually sent) or you can encode the string based on the time and date value as read from your PLC. Terminate with a line feed (hex value 0A).
	FROM:*Name<EmailAddress>*[LF]	This is the e-mail address of the nominal sender of the e-mail. Terminate with a line feed (hex value 0A).
	TO:*Name<EmailAddress>*[LF]	Multiple TO statements are permitted. Each must be terminated by a line feed (hex value 0A). The TO lines must correlate to recipients in the RCPT TO block.
	CC:*Name<EmailAddress>*[LF]	Multiple CC statements are permitted. Each must be terminated by a line feed (hex value 0A). The CC lines must correlate to recipients in the RCPT TO block.
	REPLY-TO:*Name<EmailAddress>*[LF]	This line does not have to match the FROM line (the Reply recipient can differ from the nominal sender of the e-mail). Terminate with a line feed (hex value 0A).
	SUBJECT:*Description*[LF]	This line describes the content of the e-mail. Terminate with a line feed (hex value 0A).
MESSAGE (Body)	[LF]	The body of the message text must be preceded by a blank line (hex value 0A).
	The body of the message is not preceded by a keyword.[CRLF] .[CRLF]	You can type the message body as one long line or use line feeds (hex value 0A) to create line breaks. Terminate the message body with a carriage return-line feed (hex value 0D0A) followed by a period and a second carriage return-line feed (hex value 0D0A).

Notes:

The MESSAGE block is the same as the DATA section described in RFC 821.

The Header lines are optional and can be entered in any order. Also, other Header lines, as described in RFC 821/822, may be accepted by your SMTP server.

The angle brackets around the e-mail addresses are literal and are entered as part of the mail data. The square brackets around the carriage returns and line feeds are not entered: only the carriage returns and/or line feeds are entered.

Counting Characters in an E-mail

If you enter your e-mail message into the PLC directly, without using the Configurator, you need to count the sizes of the MAIL FROM, RCPT TO, and MESSAGE blocks, and put those counts into the overhead block, as described on page D-3. Use these guidelines:

- Alphanumeric characters, punctuation, and spaces each count as a character.
- A line feed (hex value 0A) counts as one character.
- A carriage return (hex value 0D) counts as one character.
- A character uses a byte of memory. Divide the number of characters by two to obtain the count in words. (E-mail data blocks are stored in V- or K-memory, and these memory types are allocated in units of words.)

  If the block size is an odd number of bytes and you are entering the message data directly into the PLC, you must add an extra space (hex value 20) before the termination (the final line feed or carriage return-line feed). For instance, if the block size is 39 characters, you add a space to increase it to 40, and then divide by two to obtain the count in words, which is 20. If you use the Configurator to build and write the message data to the PLC, the space is added for you automatically.

Rules for Terminating Blocks

The RFC specifications for SMTP e-mail stipulate that each block must be terminated by certain delimiting characters. This is done for you automatically if you build your e-mail message in the Configurator. If you enter the message data in the PLC directly, without using the Configurator, you must ensure that the blocks are properly terminated.

- The MAIL FROM block and RCPT TO block are terminated by a carriage return-line feed sequence. The carriage return must occupy the first byte of a word and the line feed must occupy the second byte; they cannot be split across two words. If the message text does not end on a word boundary, a blank space (hex value 20) must be added in the last byte of the message so that the carriage return will not occupy the second-byte position in the word.

- The MESSAGE block must be terminated by a five-character sequence (carriage return, line feed, period, carriage return, line feed). If the user-entered text ends on a word boundary, a blank space must be inserted before the first terminating character (the first carriage return) to fill up that word. In other words, the first of the five terminating characters (the first carriage return) must always occupy the second byte of a word. It must not occupy the first byte.

Figure D-1 shows how to count the characters in a typical line of an e-mail and convert the count from characters to words.

R	C	P	T		T	O	:	<	A	n	y	b	o	d	y	@	A	d	d	r	e	s	s	.	c	o	m	>	CR	LF

This block has 31 characters, which is an invalid block size. An additional space needs to be added before the terminating CR LF to bring the number of characters to 32 (resulting in a count of 16 words). However, if the block length had been an even number of characters (ending on a word boundary), it would not be necessary to insert a blank space before the terminating CR LF.

Figure D-1 Counting Characters, Converting to Words

D.2 Sample E-mail Message with Placeholders for Variables

Using Placeholders for Variables in E-mail

In order to be able to use variables in e-mail, you have to be able to set aside a fixed area in memory for the variable within a given message block. You do this by creating a placeholder, a text string that is the same size as your variable. To locate the starting address of the placeholder for your variable within the e-mail block (which becomes the starting address of your variable in the PLC code), you count from the beginning of the block where the placeholder is located to the first character of the placeholder itself.

This section provides a sample e-mail message with placeholders for variables. Section D.3 provides three sample SFSUBs that insert variables into an e-mail message. The first SFSUB overwrites the DATE header field (dd mon yy hh:mm:ss) with date and time values from the PLC. The second and third SFSUBs write integer and real values into the e-mail message when the e-mail job is triggered.

Entering Variables in an E-mail Message

To send an E-mail job containing variables from the SIMATIC 505 TCP/IP module, you must perform the following tasks:

- Load the text for the MAIL FROM, RCPT TO, and MESSAGE blocks (described below) into PLC memory. Use a placeholder for the variable. The placeholder must take the same number of memory locations as the variable and it should be easy to differentiate from the rest of the message text.
- Identify the starting location of the placeholder within the e-mail data block. To do this, you must be able to recognize the beginning byte of your placeholder, and you must be able to count the bytes of the block in order to identify what the starting address of the placeholder is within the block.
- Create PLC code to write the variable to the location that you have set aside (by means of your placeholder, which gets overwritten) in the e-mail message.
- Configure an E-mail job with the memory locations and counts of the MAIL FROM, RCPT TO, and MESSAGE data blocks.
- Download the configuration to the module.
- Write PLC code to trigger the Job Start bit.
- Run your PLC program and test the E-mail job.

Figure D-2 shows a sample e-mail message with placeholders for the variables used in SFSUB1, SFSUB2, and SFSUB3. You can copy this message verbatim to create an example e-mail job for use with the SFSUBs provided in Section D.3.

Mail From

E-mail Address:

MAIL FROM: <Host Name of PLC>

Rcpt To

Add Rcpt

E-mail Address:

RCPT TO: <Recipient@MyMail.com>
RCPT TO: <CarbonCopy@MyMail.com>

Message

Header Type:

Add Header

Header Text:

FROM: <Me@MyMail.com>
DATE: dd mon yy hh:mm:ss
TO: RecipientName
CC: CarbonCopyName
SUBJECT: This e-mail uses variables for the date and for some values in the body of the message.

E-mail Text:

The "dd mon yy hh:mm:ss" string, in Header area (above), is the placeholder for the Date variable of SFSUB1.
Here is the placeholder for the variable used in SFSUB2: SDDDDD. Here is the placeholder for the integer variable used in the first call of SFSUB3: SDDDDDDD.DDD. Here is the placeholder for the real variable used in the second call-- SDDDDDDD.DDD.
If you program your PLC with the example code provided for SFSUB1, SFSUB2, and SFSUB3, and provide values for the variables, at runtime when the e-mail is sent, the date and number placeholders will be replaced with the values of the variables. Remember to download this configuration to the module after you write the message data to the PLC!!

Figure D-2 Sample E-mail Message

Figure D-3 shows the word counts for the e-mail message blocks, given a starting address of V3000 and a five-word overhead block. (The e-mail overhead block, which is a feature of SIMATIC 505 TCP/IP E-mail jobs rather than the RFC 821/822 e-mail specifications, is described in Section D.1.)

If you use the sample e-mail message provided in Figure D-2, your block counts should be identical to the counts shown in Figure D-3. If there are any discrepancies, consult Table D-3, which explains the contents of various V-memory addresses within the e-mail message. In several places, blank spaces and non-printing characters (the block terminators described in Section D.1) are required in the e-mail message.

After you construct your sample e-mail job (using the message content shown in Figure D-2, and your own chosen values for the SMTP server IP address and other e-mail job parameters), you can write PLC code using the SFSUB samples in Section D.3. The SFSUBs will overwrite the placeholders in your e-mail message with the date and time values from your PLC and whatever integer and real number values you provide as parameters.

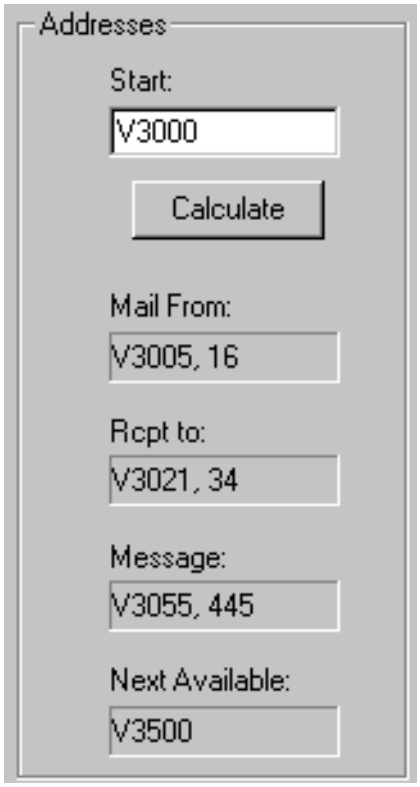


Figure D-3 Block Counts for Sample E-mail Message

Table D-3 explains the contents of various V-memory addresses within the sample e-mail message that was shown in Figure D-2.

Table D-3 V-Memory Address Contents in Sample Message

Location	Contents
OVERHEAD BUFFER. Count is 5 words.	
V3000	Starting address. First word of overhead buffer. Contains “5” (which is the size, in words, of the overhead block).
V3001	Second word of overhead buffer. Contains “0” (default value).
V3002	Third word of overhead buffer. Contains “16” (count of MAIL FROM block, in words).
V3003	Fourth word of overhead buffer. Contains “34” (count of RCPT TO block, in words).
V3004	Fifth word of overhead buffer. Contains “445” (count of MESSAGE block, in words).
MAIL FROM block. Count is 16 words (32 characters).	
V3005	First word of MAIL FROM block. Contains the letters “M” and “A”.
V3006	“I” and “L”
V3007	“ ” (space) and “F”
V3008	“R” and “O”
V3009	“M” and “ : ” (colon character)
V3010	“ ” (space) and “<”
V3011	“H” and “o”
V3012	“s” and “t”
. . .	
V3019	“>” and “ ” (space) The last character of the user-entered text is an angle bracket. The Configurator adds a blank space to fill the other byte of the word.
V3020	[CR] (non-printing character) and [LF] (non-printing character)
RCPT TO block. Count is 34 words (68 characters).	
V3021	First word of RCPT TO block. Contains the letters “R” and “C”.
. . .	
V3053	“m” and “>”
V3054	[CR] (non-printing character) and [LF] (non-printing character)

Table D-3 V-Memory Address Contents (continued)

Location	Contents
MESSAGE block. Count is 445 words (890 characters).	
V3055	First word of MESSAGE block. Contains the letters “F” and “R”.
. . .	
V3069	“d” and “d” (first word of **dd mon yy hh:mm:ss** placeholder for SFSUB1)
. . .	
V3229	“S” and “D” (first word of **SDDDDD** placeholder for SFSUB2)
. . .	
V3274	“ ” (space) and “ ” (space) Note: the second blank space is padding so that the upcoming placeholder will be the first byte rather than the second byte of a word address. The sample SFSUBs in Section D.3 write to the first byte of word addresses. User code could be written, however, to write a variable value to the second byte of a word address.
V3275	“S” and “D” (first word of **SDDDDDDD.DDD** placeholder for first call of SFSUB3)
. . .	
V3318	“S” and “D” (first word of **SDDDDDDD.DDD** placeholder for second call of SFSUB3)
. . .	
V3495	“L” and “C”
V3496	“ ! ” (exclamation point) and “ ! ” (exclamation point) These are the final characters of the user-entered message text.
V3497	“ ” (space) and [CR] (non-printing character) Note: As explained on page D-7, a blank space must be inserted so that the carriage return (the first of five characters required for termination of the MESSAGE block) occupies the second, not first, byte of the word.
V3498	[LF] (non-printing character) and “ . ” (period)
V3499	[CR] (non-printing character) and [LF] (non-printing character)

D.3 SFSUB Samples for Use with E-mail Jobs

Using Variables in E-mail Jobs

In order to use a variable in an E-mail job, you need to perform the following tasks:

- Configure an e-mail job (Section 4.13 provides guidelines)
- Write the e-mail message data, including a placeholder for the variable, to the PLC (Section 4.13 provides guidelines; Section D.2 provides sample message text for use with these SFSUBs)
- Determine the starting address of the variable placeholder within the relevant MAIL FROM, RCPT TO, or MESSAGE block (Section D.2 shows how to locate the addresses of the placeholders used with these SFSUBs)

After you perform these tasks, you can create Special Function subroutines (SFSUBs) to place variable information in the address area that is reserved by the placeholder in your e-mail.

Figure D-4 shows calls for three SFSUBs.

- SFSUB1, shown in Table D-4, calculates the current date, using Status Word 141–143. (The PLC clock must be set to the correct time in order for you to calculate dates.)
- SFSUB2, shown in Table D-5, calculates a signed integer value. It left-justifies the value (SDDDDD, where S is the sign) and has an output of 6 characters (3 words).
- SFSUB3, shown in Table D-6, calculates signed integer and real values. It right-justifies the values (SDDDDDDD.DDD, where S is the sign) and has an output of 12 characters (6 words).

Notice that the addresses of the output parameters in the subroutine calls correspond to the starting addresses listed in Table D-3 for the variable placeholders in the sample e-mail message. The addresses for the input parameters (V400 and V401.) are where you would store the source values for the integer and real number values that you want to use for the example.

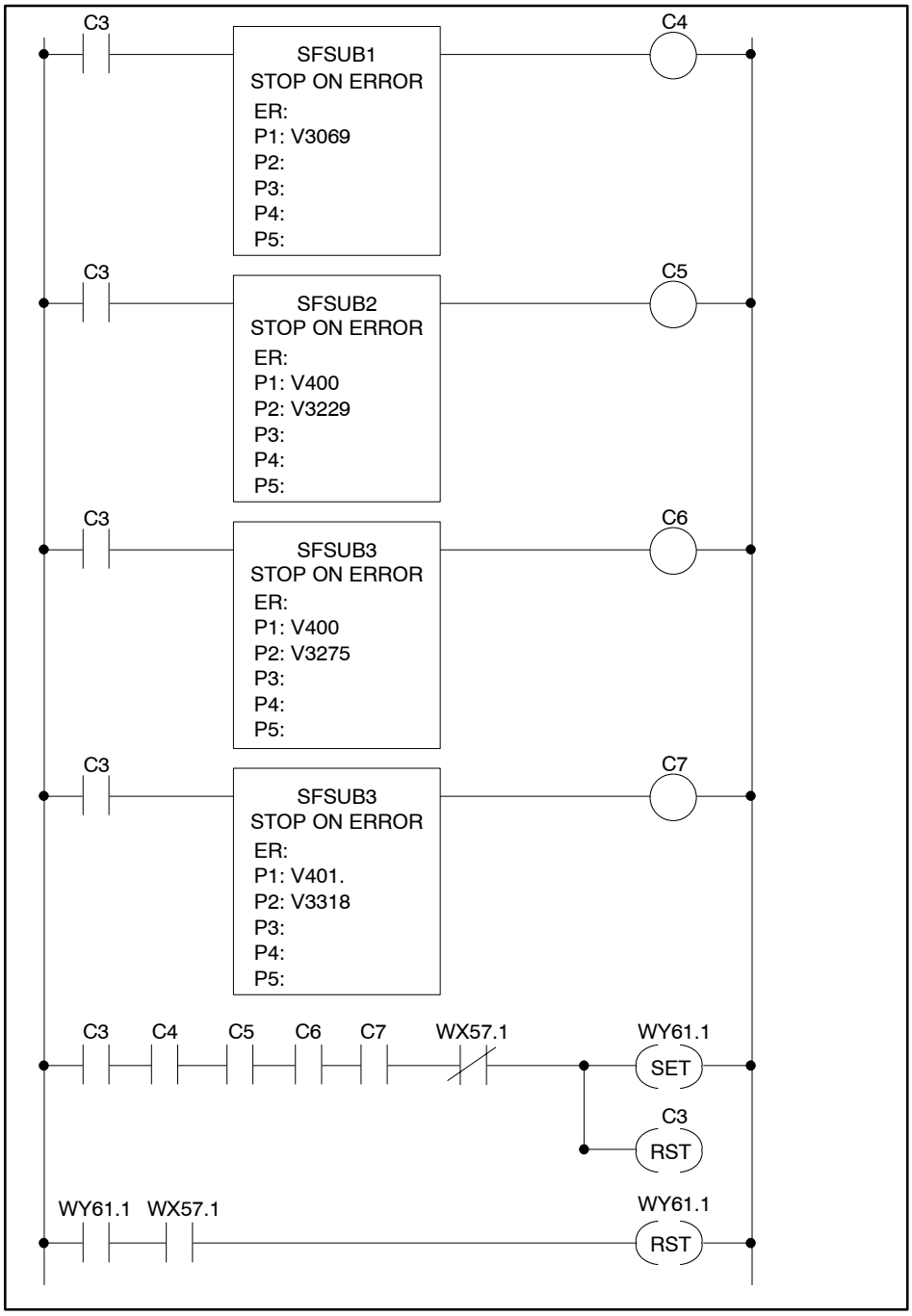


Figure D-4 SFSUBs for E-mail Jobs

Table D-4 SFSUB1: Calculate Date

```
00001     *     FORMAT DATE FOR E-MAIL
00002     *       P1:LOCATION TO STORE DATE
00003     *
00004     *     DAY
00005     IMATH   T8 := STW142 & 0HFF00
00006     IMATH   T8 := ( T8 >> 4 | T8 >> 8 ) & 0H0F0F
00007     IMATH   P1( 1 ) := T8 | 0H3030
00008     *
00009     *     MONTH
00010     *
00011     IMATH   T8 := STW141 & 0HFF
00012     *     JAN
00013     IF     T8 = 0H01
00014     IMATH   P1( 2 ) := 0H204A
00015     IMATH   P1( 3 ) := 0H414E
00016     ENDIF
00017     *     FEB
00018     IF     T8 = 0H02
00019     IMATH   P1( 2 ) := 0H2046
00020     IMATH   P1( 3 ) := 0H4542
00021     ENDIF
00022     *     MAR
00023     IF     T8 = 0H03
00024     IMATH   P1( 2 ) := 0H204D
00025     IMATH   P1( 3 ) := 0H4152
00026     ENDIF
00027     *     APR
00028     IF     T8 = 0H04
00029     IMATH   P1( 2 ) := 0H2041
00030     IMATH   P1( 3 ) := 0H5052
00031     ENDIF
00032     *     MAY
00033     IF     T8 = 0H05
00034     IMATH   P1( 2 ) := 0H204D
00035     IMATH   P1( 3 ) := 0H4159
00036     ENDIF
00037     *     JUN
00038     IF     T8 = 0H06
00039     IMATH   P1( 2 ) := 0H204A
00040     IMATH   P1( 3 ) := 0H554E
00041     ENDIF
```

Table D-4 SFSUB1: Calculate Date (continued)

```
00042     *     JUL
00043     IF    T8 = 0H07
00044     IMATH  P1( 2 ) := 0H204A
00045     IMATH  P1( 3 ) := 0H554C
00046     ENDIF
00047     *     AUG
00048     IF    T8 = 0H08
00049     IMATH  P1( 2 ) := 0H2041
00050     IMATH  P1( 3 ) := 0H5547
00051     ENDIF
00052     *     SEP
00053     IF    T8 = 0H09
00054     IMATH  P1( 2 ) := 0H2053
00055     IMATH  P1( 3 ) := 0H4550
00056     ENDIF
00057     *     OCT
00058     IF    T8 = 0H10
00059     IMATH  P1( 2 ) := 0H204F
00060     IMATH  P1( 3 ) := 0H4354
00061     ENDIF
00062     *     NOV
00063     IF    T8 = 0H11
00064     IMATH  P1( 2 ) := 0H204E
00065     IMATH  P1( 3 ) := 0H4F56
00066     ENDIF
00067     *     DEC
00068     IF    T8 = 0H12
00069     IMATH  P1( 2 ) := 0H2044
00070     IMATH  P1( 3 ) := 0H4543
00071     ENDIF
00072     *
00073     *     YEAR
00074     IMATH  T8 := STW141 & 0HFF00
00075     IMATH  T9 := T8 >> 12 & 0H0F
00076     IMATH  T8 := T8 & 0H0F00
00077     IMATH  P1( 4 ) := T9 | 0H2030
00078     IMATH  P1( 5 ) := T8 | 0H3020
00079     *
00080     *
```

Table D-4 SFSUB1: Calculate Date (continued)

```
00081        *      TIME
00082        *
00083        *      HOUR
00084        IMATH   T8 := STW142 & 0HFF
00085        IMATH   T8 := ( T8 << 4 | T8 ) & 0H0F0F
00086        IMATH   P1( 6 ) := T8 | 0H3030
00087        *
00088        *      MINUTE
00089        IMATH   T8 := STW143 & 0HFF00
00090        IMATH   T9 := T8 >> 12 & 0H0F
00091        IMATH   T8 := T8 & 0H0F00
00092        IMATH   P1( 7 ) := T9 | 0H3A30
00093        IMATH   P1( 8 ) := T8 | 0H303A
00094        *
00095        *      SECOND
00096        IMATH   T8 := STW143 & 0HFF
00097        IMATH   T8 := ( T8 << 4 | T8 ) & 0H0F0F
00098        IMATH   P1( 9 ) := T8 | 0H3030
00099        *
             ****  END ****
```

Table D-5 SFSUB2: Calculate Integer

```
00001    *     FORMAT INTEGER FOR E-MAIL
00002    *       P1:VALUE TO FORMAT
00003    *       P2:OUTPUT LOCATION
00004    *
00005    *     SPECIAL CASE -32768
00006    IF     -32768 = P1
00007    IMATH   P2 := 0H2D33
00008    IMATH   P2( 2 ) := 0H3237
00009    IMATH   P2( 3 ) := 0H3638
00010    RETURN
00011    ENDIF
00012    *     COPY VALUE TO WORK VARIABLE AND INITIALIZE INDEX
00013    IMATH   T8 := P1
00014    IMATH   T9 := 1
00015    *
00016    *     OUTPUT SIGN AND INITIALIZE BYTE MARKER
00017    IF     0 > T8
00018    IMATH   T8 := 0 - T8
00019    IMATH   P2 := 0H2D00
00020    ELSE
00021    IMATH   P2 := 0H2000
00022    ENDIF
00023    IMATH   T7 := 1
00024    *
00025    *     INITIALIZE DIVISOR AND SUPPRESS-LEADING-ZEROS FLAG
00026    IMATH   T6 := 10000
00027    IMATH   T4 := 1
00028    *
00029    *     OUTPUT DIGITS
00030    *
00031    LABEL    LABEL.........: 1
00032    *
00033    *     CALCULATE DIGIT
00034    IMATH   T5 := T8 / T6
00035    *
00036    *     UPDATE WORK VALUE AND DIVISOR
00037    IMATH   T8 := T8 - T5 * T6
00038    IMATH   T6 := T6 / 10
00039    *
```

Table D-5 SFSUB2: Calculate Integer (continued)

```
00040          *      SUPPRESS LEADING ZEROS
00041          IF     0 = T5 AND 0 <> T4 AND 0 < T6
00042          GOTO     LABEL.........: 1
00043          ENDIF
00044          IMATH   T4 := 0
00045          *
00046          *      MAKE ASCII
00047          IMATH   T5 := 0H30 | T5
00048          *
00049          *      OUTPUT DIGIT
00050          IF     0 <> T7
00051          IMATH   T5 := T5 | P2( T9 )
00052          IMATH   P2( T9 ) := T5
00053          IMATH   T9 := T9 + 1
00054          IMATH   T7 := 0
00055          ELSE
00056          IMATH   P2( T9 ) := T5 << 8
00057          IMATH   T7 := 1
00058          ENDIF
00059          *
00060          *      NEXT DIGIT
00061          IF     0 < T6
00062          GOTO     LABEL.........: 1
00063          ENDIF
00064          *
00065          *      FINISH LAST
00066          IF     0 <> T7
00067          IMATH   P2( T9 ) := P2( T9 ) | 0H20
00068          IMATH   T9 := T9 + 1
00069          ENDIF
00070          *
00071          *      PAD WITH BLANKS
00072          LABEL    LABEL.........: 2
00073          IF     3 < T9
00074          GOTO     LABEL.........: 3
00075          ENDIF
00076          IMATH   P2( T9 ) := 0H2020
00077          IMATH   T9 := T9 + 1
00078          GOTO     LABEL.........: 2
00079          LABEL    LABEL.........: 3
00080          *
****  END ****
```

Table D-6 SFSUB3: Calculate Integer or Real

```
00001    *     FORMAT VALUE FOR E-MAIL
00002    *       P1:VALUE TO FORMAT
00003    *       P2:OUTPUT LOCATION
00004    *
00005    *     INITIALIZE WORK VALUES, DIVISOR, BYTE FLAG, AND INDEX
00006    MATH    T8. := ABS ( P1 )
00007    MATH    T12 := FRAC ( T8. ) * 1000.0
00008    MATH    T6. := 1.0E6
00009    IMATH   T10 := 0
00010    IMATH   T11 := 1
00011    *
00012    *     VALIDATE SIZE (OTHERWISE FILL WITH ***)
00013    IF      1.0E7 <= T8. OR 0.0 <> T8. AND 1.0E-3 > T8.
00014    LABEL    LABEL.........: 1
00015    IMATH   P2( T11 ) := 0H2A2A
00016    IMATH   T11 := T11 + 1
00017    IF      6 >= T11
00018    GOTO     LABEL.........: 1
00019    ENDIF
00020    RETURN
00021    ENDIF
00022    *
00023    *     OUTPUT LEADING BLANKS
00024    LABEL    LABEL.........: 2
00025    IF      1.0 < T6. AND T6. > T8.
00026    IF      0 = T10
00027    IMATH   T5 := 0H2000
00028    IMATH   T10 := 1
00029    ELSE
00030    IMATH   P2( T11 ) := 0H2020
00031    IMATH   T11 := T11 + 1
00032    IMATH   T10 := 0
00033    ENDIF
00034    MATH    T6. := T6. / 10.0
00035    GOTO     LABEL.........: 2
00036    ENDIF
00037    *
00038    *     OUTPUT SIGN
00039    IF      0 = T10
00040    IF      0.0 > P1
00041    IMATH   T5 := 0H2D00
00042    ELSE
00043    IMATH   T5 := 0H2000
00044    ENDIF
00045    IMATH   T10 := 1
```

Table D-6 SFSUB3: Calculate Integer or Real (continued)

```
00046        ELSE
00047        IF     0.0 > P1
00048        IMATH   P2( T11 ) := 0H202D
00049        ELSE
00050        IMATH   P2( T11 ) := 0H2020
00051        ENDIF
00052        IMATH   T11 := T11 + 1
00053        IMATH   T10 := 0
00054        ENDIF
00055        *
00056        *      OUTPUT DIGITS
00057        *
00058        LABEL    LABEL.........: 3
00059        MATH    T4 := TRUNC ( T8. / T6. )
00060        MATH    T8. := T8. - T4 * T6.
00061        MATH    T6. := T6. / 10.0
00062        IMATH   T4 := T4 | 0H30
00063        IF     0 = T10
00064        IMATH   T5 := T4 << 8
00065        IMATH   T10 := 1
00066        ELSE
00067        IMATH   P2( T11 ) := T5 | T4
00068        IMATH   T11 := T11 + 1
00069        IMATH   T10 := 0
00070        ENDIF
00071        *
00072        *      CONTINUE IF NOT FRACTION
00073        IF     0.1 < T6.
00074        GOTO    LABEL.........: 3
00075        ENDIF
00076        *
00077        *      CHECK FOR DONE
00078        IF     0.0 >= T12
00079        GOTO    LABEL.........: 4
00080        ENDIF
00081        *      INITIALIZE FOR FRACTION AND INSERT DECIMAL
00082        IF     0 < T12
00083        MATH    T8. := T12
00084        MATH    T6. := 100.0
00085        IMATH   T12 := -1
00086        IF     0 = T10
00087        IMATH   T5 := 0H2E00
00088        IMATH   T10 := 1
```

Table D-6 SFSUB3: Calculate Integer or Real (continued)

```
00089        ELSE
00090        IMATH   P2( T11 ) := T5 | 0H2E
00091        IMATH   T11 := T11 + 1
00092        IMATH   T10 := 0
00093        ENDIF
00094        GOTO    LABEL.........: 3
00095        ENDIF
00096        *
00097        *      PAD WITH BLANKS
00098        LABEL   LABEL.........: 4
00099        IF      0 <> T10
00100        IMATH   P2( T11 ) := T5 | 0H20
00101        IMATH   T11 := T11 + 1
00102        ENDIF
00103        LABEL   LABEL.........: 5
00104        IF      6 >= T11
00105        IMATH   P2( T11 ) := 0H2020
00106        IMATH   T11 := T11 + 1
00107        GOTO    LABEL.........: 5
00108        ENDIF
00109        *
             ****  END ****
```

Index

Symbols

Numbers

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

Customer Registration

We would like to know what you think about our user manuals so that we can serve you better. How would you rate the quality of our manuals?

	Excellent	Good	Fair	Poor
Accuracy	_____	_____	_____	_____
Organization	_____	_____	_____	_____
Clarity	_____	_____	_____	_____
Completeness	_____	_____	_____	_____
Overall design	_____	_____	_____	_____
Size	_____	_____	_____	_____
Index	_____	_____	_____	_____

Would you be interested in giving us more detailed comments about our manuals?

☐ **Yes!** Please send me a questionnaire.

☐ **No.** Thanks anyway.

Your Name: _____

Title: _____

Telephone Number: (_____) _____

Company Name: _____

Company Address: _____

_____

_____

Manual Name: SIMATIC 505-CP1434-TCP Industrial Ethernet TCP/IP User Manual **Edition:** Original

Manual Assembly Number: 2810566-0001 **Date:** 12/01

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO.3 JOHNSON CITY, TN

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN: Technical Communications M/S 519
SIEMENS ENERGY & AUTOMATION, INC.
3000 BILL GARLAND RD
P O BOX 1255
JOHNSON CITY TN 37605-1255

FOLD