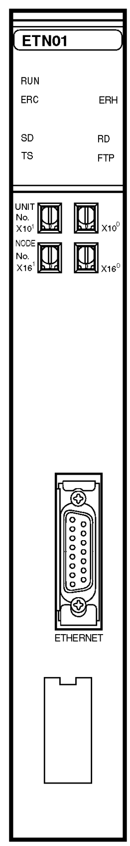
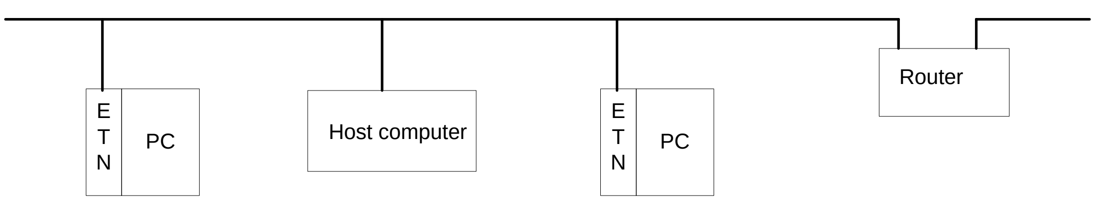
生産終了
DISCONTINUED

SYSMAC CV-series Ethernet

System Manual

Revised July 2000





Read and Understand this Manual

Please read and understand this manual before using the product. Please consult your OMRON representative if you have any questions or comments.

Warranty and Limitations of Liability

WARRANTY

OMRON's exclusive warranty is that the products are free from defects in materials and workmanship for a period of one year (or other period if specified) from date of sale by OMRON.

OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, REGARDING NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR PARTICULAR PURPOSE OF THE PRODUCTS. ANY BUYER OR USER ACKNOWLEDGES THAT THE BUYER OR USER ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE. OMRON DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED.

LIMITATIONS OF LIABILITY

OMRON SHALL NOT BE RESPONSIBLE FOR SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED ON CONTRACT, WARRANTY, NEGLIGENCE, OR STRICT LIABILITY.

In no event shall the responsibility of OMRON for any act exceed the individual price of the product on which liability is asserted.

IN NO EVENT SHALL OMRON BE RESPONSIBLE FOR WARRANTY, REPAIR, OR OTHER CLAIMS REGARDING THE PRODUCTS UNLESS OMRON'S ANALYSIS CONFIRMS THAT THE PRODUCTS WERE PROPERLY HANDLED, STORED, INSTALLED, AND MAINTAINED AND NOT SUBJECT TO CONTAMINATION, ABUSE, MISUSE, OR INAPPROPRIATE MODIFICATION OR REPAIR.

Application Considerations

SUITABILITY FOR USE

OMRON shall not be responsible for conformity with any standards, codes, or regulations that apply to the combination of products in the customer's application or use of the products.

At the customer's request, OMRON will provide applicable third party certification documents identifying ratings and limitations of use that apply to the products. This information by itself is not sufficient for a complete determination of the suitability of the products in combination with the end product, machine, system, or other application or use.

The following are some examples of applications for which particular attention must be given. This is not intended to be an exhaustive list of all possible uses of the products, nor is it intended to imply that the uses listed may be suitable for the products:

- Outdoor use, uses involving potential chemical contamination or electrical interference, or conditions or uses not described in this manual.
- Nuclear energy control systems, combustion systems, railroad systems, aviation systems, medical equipment, amusement machines, vehicles, safety equipment, and installations subject to separate industry or government regulations.
- Systems, machines, and equipment that could present a risk to life or property.

Please know and observe all prohibitions of use applicable to the products.

NEVER USE THE PRODUCTS FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCTS ARE PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

PROGRAMMABLE PRODUCTS

OMRON shall not be responsible for the user's programming of a programmable product, or any consequence thereof.

Disclaimers

CHANGE IN SPECIFICATIONS

Product specifications and accessories may be changed at any time based on improvements and other reasons.

It is our practice to change model numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the products may be changed without any notice. When in doubt, special model numbers may be assigned to fix or establish key specifications for your application on your request. Please consult with your OMRON representative at any time to confirm actual specifications of purchased products.

DIMENSIONS AND WEIGHTS

Dimensions and weights are nominal and are not to be used for manufacturing purposes, even when tolerances are shown.

PERFORMANCE DATA

Performance data given in this manual is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of OMRON's test conditions, and the users must correlate it to actual application requirements. Actual performance is subject to the OMRON Warranty and Limitations of Liability.

ERRORS AND OMISSIONS

The information in this manual has been carefully checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical, or proofreading errors, or omissions.

Notice:

OMRON products are manufactured for use according to proper procedures by a qualified operator and only for the purposes described in this manual.

The following conventions are used to indicate and classify precautions in this manual. Always heed the information provided with them. Failure to heed precautions can result in injury to people or damage to the product.

⚠ 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 Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

OMRON Product References

All OMRON products are capitalized in this manual. The word “Unit” is also capitalized when it refers to an OMRON product, regardless of whether or not it appears in the proper name of the product.

The abbreviation “Ch,” which appears in some displays and on some OMRON products, often means “word” and is abbreviated “Wd” in documentation in this sense.

The abbreviation “PC” means Programmable Controller and is not used as an abbreviation for anything else.

Visual Aids

The following headings appear in the left column of the manual to help you locate different types of information.

Note Indicates information of particular interest for efficient and convenient operation of the product.

1, 2, 3... 1. Indicates lists of one sort or another, such as procedures, checklists, etc.

© OMRON, 1994

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.

No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

TABLE OF CONTENTS

TABLE OF CONTENTS

About this Manual:

This manual describes the installation and operation of a SYSMAC CV-series Ethernet System and includes the sections described below. SYSMAC CV-series PCs are interfaced to the Ethernet System via a CV500-ETN01 Ethernet Unit. In this manual, a SYSMAC CV-series Ethernet System is referred to as simply an Ethernet System; a CV500-ETN01 Ethernet Unit, as an Ethernet Unit.

An Ethernet Unit is classified and treated as a CPU Bus Unit in PC processing.

This manual is based on Ethernet* networks comprised of Ethernet Unit nodes and UNIX* host computer nodes. Although details can vary, theoretically and device support the same Ethernet protocols as the Ethernet Unit can form nodes on the network.

Please read this manual carefully and be sure you understand the information provided before attempting to install and operate an Ethernet System.

Section 1 introduces the overall structure of an Ethernet System, outlines the features of the Ethernet Unit, describes the communications protocols used by an Ethernet System, and provides basic precautions for use of an Ethernet System. Further information is provided in following sections of this manual. Refer to *Appendix B Specifications* for a table of Ethernet Unit specifications.

Section 2 provides information on installing and starting the Ethernet Unit, including descriptions of Unit components, installation, settings, and wiring.

Section 3 provides information on communicating on Ethernet Systems and interconnected networks using FINS commands. The information provided in the section deals only with FINS communications in reference to Ethernet Units. Refer to the *FINS Command Reference Manual* for details on FINS commands for other Units and networks.

FINS commands issued from a PC are sent via the SEND(192), RECV(193), and CMND(194) instructions programmed into the user ladder-diagram program. Although an outline of these instructions is provided in this section, refer to the *CV-series PC Operation Manual: Ladder Diagrams* for further details on ladder-diagram programming.

Section 4 describes the features and use of the FTP (File Transfer Protocol) from the host computer to read and write files to and from a Memory Card inserted in the PC. The Ethernet Unit uses the FTP server for these operations.

Section 5 describes the socket services provided by the Ethernet Unit . Socket services enable data communications between the PC's user program and a host computer program. Refer to documentation for your host computer for details on programming host computer sockets.

Section 6 explains FINS commands that can be sent to Ethernet Units. For information on FINS commands that can be sent to PCs or to other Units that support FINS commands, refer to the *FINS Command Reference Manual*.

Section 7 describes the RAS features of the Ethernet Unit. These features are designed to increase the Reliability, Availability, Serviceability of Ethernet Units.

Section 8 describes the procedures and information required to maintain the Ethernet Unit in an Ethernet system, including periodic maintenance and inspections.

Section 9 describes errors that can occur when using the Ethernet Unit and what to do about them. This section does not deal with errors that can occur in general PC operation. Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for general troubleshooting information.

Appendices are provided of OMRON products related to Ethernet Systems, Ethernet Unit specifications, and other useful reference material (see list in table of contents). There is also a list of recommended reading at the end of *Appendix I IP Network Address Request Form.*

Note *Ethernet is a trademark of the Xerox Corporation. UNIX is an operating system developed by the AT&T Bell Research Laboratories.

⚠WARNING Failure to read and understand the information provided in this manual may result in personal injury or death, damage to the product, or product failure. Please read each section in its entirety and be sure you understand the information provided in the section and related sections before attempting any of the procedures or operations given.

PRECAUTIONS

This section provides general precautions for using C200H-CT001-V1/CT002 High-speed Counter Units and related devices.

The information contained in this section is important for the safe and reliable application of the C200H-CT001-V1/CT002 High-speed Counter Units. You must read this section and understand the information contained before attempting to set up or operate a C200H-CT001-V1/CT002 High-speed Counter Unit.

1 Intended Audience

This manual is intended for the following personnel, who must also have knowledge of electrical systems (an electrical engineer or the equivalent).

- Personnel in charge of installing FA systems.
- Personnel in charge of designing FA systems.
- Personnel in charge of managing FA systems and facilities.

2 General Precautions

The user must operate the product according to the performance specifications described in the relevant manuals.

Before using the product under conditions which are not described in the manual or applying the product to nuclear control systems, railroad systems, aviation systems, vehicles, combustion systems, medical equipment, amusement machines, safety equipment, and other systems, machines, and equipment that may have a serious influence on lives and property if used improperly, consult your OMRON representative.

Make sure that the ratings and performance characteristics of the product are sufficient for the systems, machines, and equipment, and be sure to provide the systems, machines, and equipment with double safety mechanisms.

This manual provides information for programming and operating the Unit. Be sure to read this manual before attempting to use the Unit and keep this manual close at hand for reference during operation.

⚠ WARNING It is extremely important that a PC and all PC Units be used for the specified purpose and under the specified conditions, especially in applications that can directly or indirectly affect human life. You must consult with your OMRON representative before applying a PC system to the above-mentioned applications.

3 Safety Precautions

⚠ WARNING Do not attempt to take any Unit apart while the power is being supplied. Doing so may result in electric shock.

⚠ WARNING Do not touch any of the terminals or terminal blocks while the power is being supplied. Doing so may result in electric shock.

⚠ WARNING Do not attempt to disassemble, repair, or modify any Units. Any attempt to do so may result in malfunction, fire, or electric shock.

4 Operating Environment Precautions

⚠ **Caution** Do not operate the control system in the following locations:

- Locations subject to direct sunlight.
- Locations subject to temperatures or humidity outside the range specified in the specifications.
- Locations subject to condensation as the result of severe changes in temperature.
- Locations subject to corrosive or flammable gases.
- Locations subject to dust (especially iron dust) or salts.
- Locations subject to exposure to water, oil, or chemicals.
- Locations subject to shock or vibration.

⚠ **Caution** Take appropriate and sufficient countermeasures when installing systems in the following locations:

- Locations subject to static electricity or other forms of noise.
- Locations subject to strong electromagnetic fields.
- Locations subject to possible exposure to radioactivity.
- Locations close to power supplies.

⚠ **Caution** The operating environment of the PC system can have a large effect on the longevity and reliability of the system. Improper operating environments can lead to malfunction, failure, and other unforeseeable problems with the PC system. Be sure that the operating environment is within the specified conditions at installation and remains within the specified conditions during the life of the system.

5 Application Precautions

Observe the following precautions when using the PC system.

⚠ **WARNING** Always heed these precautions. Failure to abide by the following precautions could lead to serious or possibly fatal injury.

- Always ground the system to 100 Ω or less when installing the Units. Not connecting to a ground of 100 Ω or less may result in electric shock.
- Always turn OFF the power supply to the PC before attempting any of the following. Not turning OFF the power supply may result in malfunction or electric shock.
  - Mounting or dismounting Power Supply Units, I/O Units, CPU Units, Memory Units, or any other Units.
  - Assembling the Units.
  - Setting DIP switches or rotary switches.
  - Connecting cables or wiring the system.
  - Connecting or disconnecting the connectors.

⚠ **Caution** Failure to abide by the following precautions could lead to faulty operation of the PC or the system, or could damage the PC or PC Units. Always heed these precautions.

- Fail-safe measures must be taken by the customer to ensure safety in the event of incorrect, missing, or abnormal signals caused by broken signal lines, momentary power interruptions, or other causes.

- Interlock circuits, limit circuits, and similar safety measures in external circuits (i.e., not in the Programmable Controller) must be provided by the customer.
- Always use the power supply voltages specified in this manual. An incorrect voltage may result in malfunction or burning.
- Take appropriate measures to ensure that the specified power with the rated voltage and frequency is supplied. Be particularly careful in places where the power supply is unstable. An incorrect power supply may result in malfunction.
- Install external breakers and take other safety measures against short-circuiting in external wiring. Insufficient safety measures against short-circuiting may result in burning.
- Do not apply voltages to the Input Units in excess of the rated input voltage. Excess voltages may result in burning.
- Do not apply voltages or connect loads to the Output Units in excess of the maximum switching capacity. Excess voltage or loads may result in burning.
- Disconnect the functional ground terminal when performing withstand voltage tests. Not disconnecting the functional ground terminal may result in burning.
- Be sure that all the mounting screws, terminal screws, and cable connector screws are tightened to the torque specified in this manual. Incorrect tightening torque may result in malfunction.
- Double-check all wiring and switch settings before turning ON the power supply. Incorrect wiring may result in burning.
- Wire correctly. Incorrect wiring may result in burning.
- Mount Units only after checking terminal blocks and connectors completely.
- Be sure that the terminal blocks, Memory Units, expansion cables, and other items with locking devices are properly locked into place. Improper locking may result in malfunction.
- Check the user program for proper execution before actually running it on the Unit. Not checking the program may result in an unexpected operation.
- Confirm that no adverse effect will occur in the system before attempting any of the following. Not doing so may result in an unexpected operation.
    - Changing the operating mode of the PC.
    - Force-setting/force-resetting any bit in memory.
    - Changing the present value of any word or any set value in memory.
- Resume operation only after transferring to the new CPU Unit the contents of the DM Area, HR Area, and other data required for resuming operation. Not doing so may result in an unexpected operation.
- Do not pull on the cables or bend the cables beyond their natural limit. Doing either of these may break the cables.
- Do not place objects on top of the cables or other wiring lines. Doing so may break the cables.
- Use crimp terminals for wiring. Do not connect bare stranded wires directly to terminals. Connection of bare stranded wires may result in burning.
- When replacing parts, be sure to confirm that the rating of a new part is correct. Not doing so may result in malfunction or burning.
- Before touching a Unit, be sure to first touch a grounded metallic object in order to discharge any static built-up. Not doing so may result in malfunction or damage.

SECTION 1
Introduction

This section introduces the overall structure of an Ethernet System, outlines the features of the Ethernet Unit, describes the communications protocols used by an Ethernet System, and provides basic precautions for use of an Ethernet System. Further information is provided in following sections of this manual. Refer to *Appendix B Specifications* for a table of Ethernet Unit specifications.

1-1 System Configuration

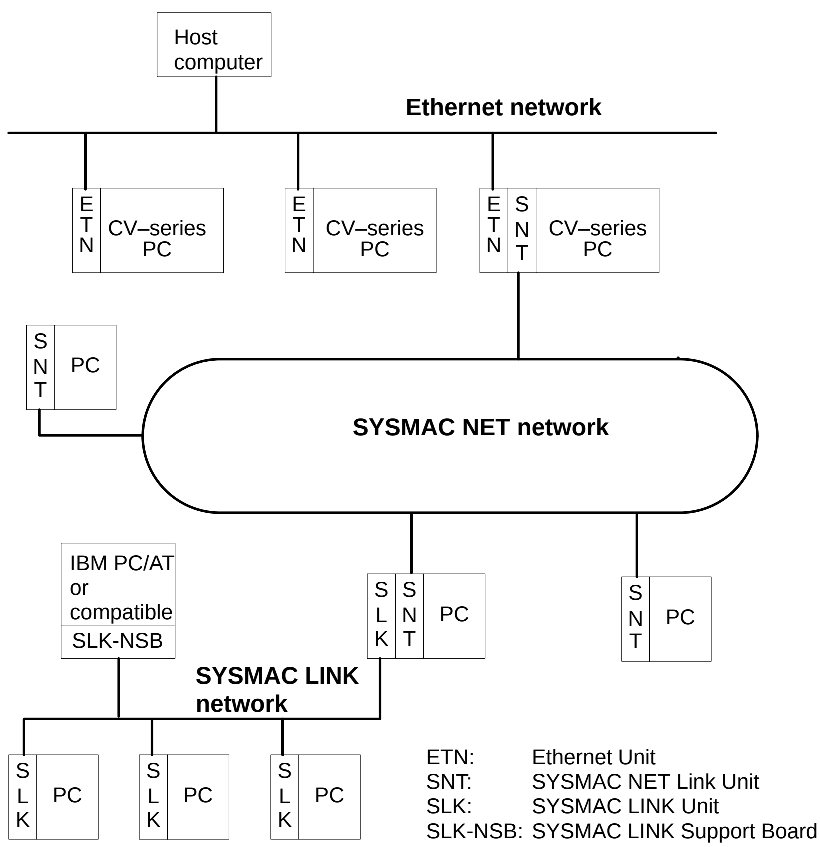
An Ethernet System allows PCs to be connected to an Ethernet network to enable advanced network communications between PCs and other nodes on the Ethernet network or interconnected networks.

PCs are connected to the Ethernet System via an Ethernet Unit, which interfaces communications between the PC and other nodes. The other nodes can exist on the same Ethernet network and can be occupied by other CV-series PCs (also connected by Ethernet Units), Unix-based platforms, IBM PC/AT or compatibles, or any other device that supports Ethernet protocols, or the other nodes can exist on interconnected networks, such as the SYSMAC CV-series SYSMAC NET Link System or the SYSMAC CV-series SYSMAC LINK System, and be occupied by any devices supported by these Systems.

An Ethernet Unit is classified and treated as a CPU Bus Unit in PC processing.

Note Devices on Ethernet System nodes other than those occupied by PCs are simply referred to as host computers in this manual. Example programs for these nodes are written in C language designed for Unix platforms. As already mentioned, other devices can occupy these nodes. In any case, be sure to test programming and communications completely on the actual platforms that you will be using before developing full-scale systems and attempting actual operation.

The following diagram illustrates three interconnected networks: a network in an Ethernet System, a SYSMAC NET network, and a SYSMAC LINK network. Any of the nodes on any of the networks shown can communicate with any other node shown.



1-2 Ethernet Unit Features

A CV500-ETN01 Ethernet Unit provides the following features to the PC to which it is mounted.

Ethernet Communications

An Ethernet Unit supports two standard international communications protocols: TCP/IP and UDP/IP. These protocols enable communications between the PC and any of the many host computers that also support them. To effectively utilize these protocols, an Ethernet Unit also provides eight communications ports so that multiple applications can run concurrently. Ethernet communications on Unix platforms on the network are supported through socket interfaces.

Memory Card File Transfers

In addition to Ethernet communications, an Ethernet Unit also supports an FTP server to enable file transfer between a host computer and a Memory Card mounted in a PC. The FTP server allows simple commands to be executed at a host computer to read or write large quantities of data without requiring any user programming in the PC. (Refer to *Section 4 FTP Server Communications* for details.)

FINS Communications

FINS communications are also supported by an Ethernet Unit, to enable transferring specific data between PCs or between PCs and host computers. FINS commands can be written into host computer programs, or they can be executed from PCs using the SEND(192), RECV(193), and CMND(194) instructions. FINS communications are most useful in internetwork communications, as described in the next feature. (Refer to *Section 3 FINS Communications Services* for details.)

Internetwork Communications

FINS communications are also supported by other SYSMAC CV-series networks, such as those created within SYSMAC NET Link Systems and SYSMAC LINK Systems, enabling communications between nodes on different types of interconnected networks through gateways. For example, a host computer on an Ethernet network can read or write data in the memories of PCs on SYSMAC NET, SYSMAC LINK, or other interconnected networks. PCs on any of these networks can also communicate with any other PC on the networks by using the SEND(192), RECV(193), and CMND(194) instructions. Internetwork communications are also possible for Programming Devices, such as the CVSS, connected to any node on interconnected networks. (Refer to *Section 3 FINS Communications Services* for details.)

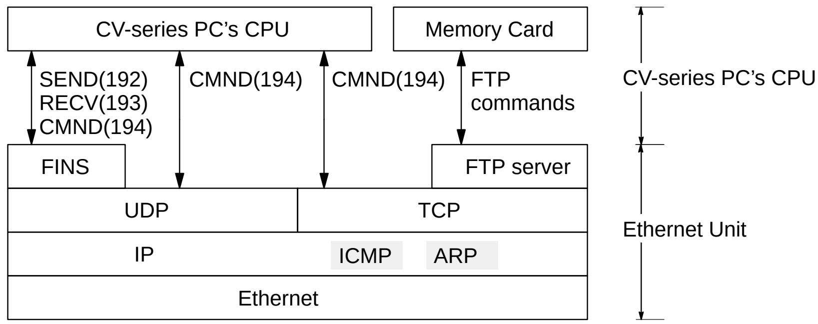
RAS Features

Ethernet connections can be checked by executing the *ping* command from a host computer on the network or by executing a similar command from a PC on the network. (Refer to page 146 for details.)

System performance and reliability are also increased by error logs, internode communications tests, and other diagnostic features. (Refer to *Section 7 RAS Features* for details.)

1-3 Software Configuration

The software supported by the Ethernet Unit runs in the layers shown in the following diagram. The components that form the various layers are defined below the diagram.



Ethernet The version-2 Ethernet frame format is used for communications.

IP Internet Protocol: Transfers datagrams to target nodes using IP addresses.

ICMP Internet Control Message Protocol: Supports IP communications by signalling errors in data transfers.

ARP Address Resolution Protocol: Determines the Ethernet address (i.e., physical address) by broadcasting based on the target IP address.

UDP User Datagram Protocol: Performs datagram communications. Data resends, priority control, flow control, and other measures to ensure communications reliability are not performed for UDP communications, i.e., there is no way of guaranteeing normal communications without programming special measures to do so into the user's application program.

TCP Transmission Control Protocol: Performs communications after establishing a connection (i.e., a virtual circuit) with the target node to provide a highly reliable communications method.

FTP File Transfer Protocol: Transfers data in file units to and from Memory Cards.

FINS Factory Interface Network Service: A protocol that transfers messages between PCs on any of various OMRON FA networks.

1-4 Addresses

Various addresses and ID numbers are used to control communications on an Ethernet network. The addresses and ID numbers actually used depends on the protocol being used. These addresses and ID numbers are provided here for reference.

Unit Number A number used to differentiate CPU Bus Units (including Ethernet Units) mounted to a PC. (See page 10.)

Node Number An address used to differentiate nodes (including Ethernet Units) on OMRON networks for FINS protocol. (See page 10.)

IP Address An address assigned to the Ethernet Unit as a node in an Ethernet network. The IP address consists of a network number, possibly a subnet number, and a host number. (See page 22.)

Network Number The portion of the IP address used to differentiate networks.

Subnet Number The portion of the IP address used differentiate subnetworks in an Ethernet network. This address exists only if the user sets a subnet mask for the IP address to allocate part of the host number as the subnet number. (See page 24.)

Host Number The portion of the IP address used to differentiate nodes on an Ethernet network.

Unit Address An address used to differentiate between various destinations (including Ethernet Units) at the same PC in FINS protocol. (See page 43.)

FINS Network Address An address set in routing tables and used to differentiate OMRON networks for FINS communications. (See page 41.)

Ethernet Address A physical address assigned to Ethernet hardware. (See page 8.)

1-5 Precautions

You must observe the following precautions when installing and using an Ethernet System. Refer to page 14 for details on installation.

1-5-1 Installation

Observe the following precautions when installing an Ethernet System.

1, 2, 3...

1. Use transceiver cable that meets IEEE802.3 standards to ensure high noise resistance.
2. Use a transceiver with a current consumption of 0.35 A or less.
3. Always turn off the power supply to the PC before connecting or disconnecting the transceiver cable.
4. Be sure not to exceed the current capacity of the Power Supply Unit on the Rack to which the Ethernet Unit is mounted. The current consumption of the Ethernet Unit is 1.7 A maximum. This value added to the current consumption of all other Units mounted to the same Rack must not exceed the capacity of the Power Supply Unit.
5. Do no install the transceiver cables or coaxial cables of the Ethernet System near power supply lines. If installation near possible sources of noise is unavoidable, install the cables in grounded metal ducts or take other measure to eliminate noise interference.

1-5-2 Ethernet and IEEE802.3 Standards

The Ethernet Unit was designed based on version-2 Ethernet standards and not on the international IEEE802.3 standards, which were developed based on version-2 Ethernet specifications. Although these two sets of standards are similar, they are not necessarily the same. Particularly, different frame formats are used, making direct communications impossible between systems that do not support the same standards. Standards for equipment used to configure networks are the same, allowing IEEE802.3-standard equipment to be used with the Ethernet Unit. Particularly the transceiver cable for the IEEE802.3 standards provides superior noise resistance and should be used for the Ethernet Unit.

Terminology also differs between version-2 Ethernet and IEEE802.3 standards. These differences are shown in the following table. Version-2 Ethernet terminology is used in this manual.

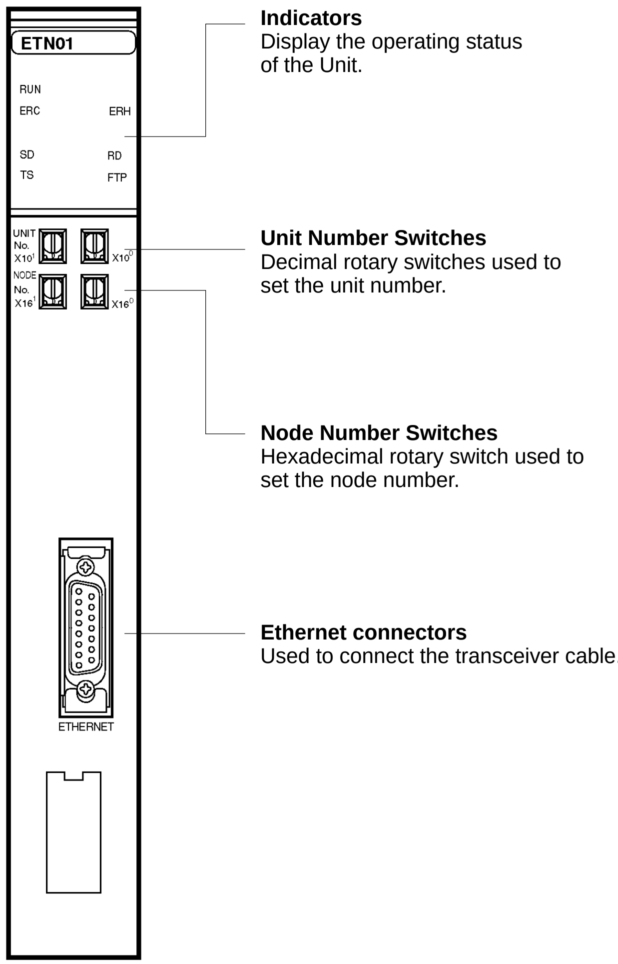
Version-2 Ethernet	IEEE802.3
Transceiver	MAU
Transceiver cable	AUI
Ethernet address	MAC address
Ethernet	10BASE5

SECTION 2
Ethernet Unit Description, Settings, and Installation

This section provides information on installing and starting the Ethernet Unit, including descriptions of Unit components, installation, settings, and wiring.

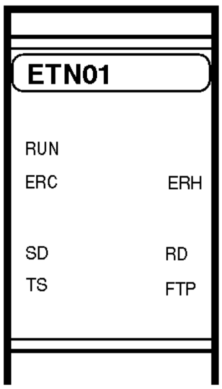
2-1 Description of Parts

The name of each part of the Ethernet Unit and the function of the part are shown in the following diagram.



Note All communications devices connected to Ethernet networks are allocated fixed Ethernet addresses. The Ethernet address of the Ethernet Unit can be found on the right side of the Unit as a 12-digit hexadecimal number. The Ethernet address of the Ethernet Unit can also be read using the CONTROLLER DATA READ command (a FINS command, see page 129 for details).

2-1-1 Indicators



The status of the indicators show the operating status of the Unit.

Indicator	Color	Status	Meaning
RUN	Green	ON	Normal operation.
		OFF	Operation stopped.
ERC (Ethernet Unit error)	Red	ON	Node number not between 1 and 126. A hardware (internal memory or communications unit) error has occurred.
		OFF	Unit normal.
ERH (PC error)	Red	ON	A PC error has occurred. Errors exist in unit numbers or I/O table settings.
		Flashing	IP address is not set or illegal IP address has been set.
		OFF	PC normal.
SD (send data)	Orange	ON	Sending data.
		OFF	Not sending data.
RD (receive data)	Orange	ON	Receiving data (i.e., receiving an normal Ethernet from addressed at this node or broadcast to all nodes).
		OFF	Not receiving data.
TS (test)	Orange	ON	Running internode test.
		OFF	Not running internode test.
FTP (FTP server service)	Orange	ON	FTP server operating.
		OFF	FTP server on stand-by.

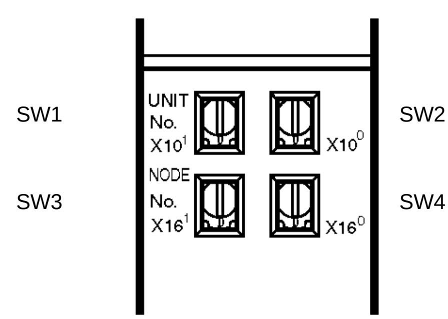
2-2 Rotary Switch Settings

Set the unit number and node number on the rotary switches on the front of the Ethernet Unit.

Note Turn off the PC power before setting the rotary switches.

2-2-1 Location of Switches

The switches for the unit number and node number are found on the front panel of the Ethernet Unit and are arranged as shown in the following diagram. There are two switches used for each setting. The unit number is set on switches SW1 and SW2; the node number is set on switches SW3 and SW4.



2-2-2 Setting Ranges

The unit number and node number must be set in the ranges shown in the following table.

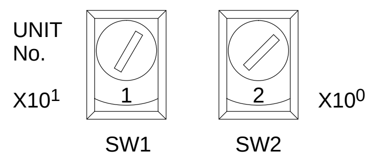
Setting	Range
Unit number (SW1, SW2)	00 to 15 decimal
Node number (SW3, SW4)	01 to 7E hexadecimal, (1 to 126 decimal)

2-2-3 Setting the Unit Number

The unit number is used to identify individual CPU Bus Units when more than one CPU Bus Unit is mounted to the same PC. The left switch (SW1) sets the tens digit and the right switch (SW2) sets the ones digit.

Set the unit number between 00 and 15. Do not set a number that has already been set for another CPU Bus Unit.

Example
Unit #12 is set in the following diagram.



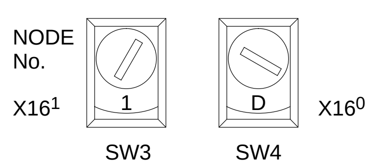
Note
1. Calculate the unit address (10 to 1F Hex) as follows: Unit number expressed in hexadecimal (0 to F) + 10 Hex.
   The address of the Ethernet Unit for the SEND(192), RECV(193), and CMND(194) instructions is the value (10 to 1F Hex) obtained by adding 10 Hex to the unit number expressed in hexadecimal (0 to F).
2. If the unit number is set to a number greater than 15, an error will occur and the ERH indicator will light.
3. Make sure that the same unit number is not set for more than one CPU Bus Unit mounted to the same PC.

2-2-4 Setting the Node Number

The node number identifies individual PC nodes connected to a network supporting FINS commands. The left switch (SW3) sets the sixteens digit (most significant digit) and the right switch (SW4) sets the ones digit (least significant digit).

Set the node number between 01 and 7E hexadecimal (1 to 126 decimal). Do not set a number that has already been set for another node on the same network.

Example
Node #29 (decimal) is set as follows: 29 = 1 x 16 + 13 = 1D (hexadecimal)



Note
1. If the node number is set to a value greater than 7E (126 decimal), an error will occur and the ERC indicator will light.
2. Make sure that the same node number is not set for more than one node on the same network.

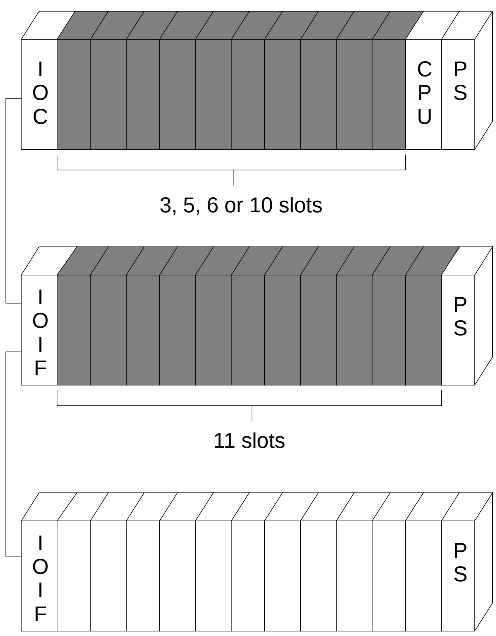
2-3 Mounting to PC Racks

The Ethernet Unit is mounted to either a CV-series CPU Rack or a CV-series Expansion CPU Rack.

2-3-1 Mounting Position

An Ethernet Unit can be mounted to any slot on the CPU Rack or Expansion CPU Rack of a CV-series PC. However, it can be mounted only to the rightmost six slots of a Rack built on a CVM1-BC103 Backplane or to rightmost three slots of a Rack built on a CVM1-BC053 Backplane.

A maximum of four Ethernet Units can be mounted to a single PC, and a maximum total number of 16 CPU Bus Units can be mounted to a single PC. CPU Bus Units include Ethernet Units, BASIC Units, Personal Computer Units, SYSMAC NET Link Units, SYSMAC LINK Units, and SYSMAC BUS/2 Remote I/O Master Units.



CPU Rack
The number of slots to which an Ethernet Unit can be mounted depends on the Backplane used, as follows:

CV500-BC031	3 slots
CV500-BC051	5 slots
CV500-BC101	10 slots
CVM1-BC103	6 slots (rightmost on Rack)
CVM1-BC053	3 slots (rightmost on Rack)

Expansion CPU Rack
An Ethernet Unit can be mounted in any of the 11 slots. An Expansion CPU Rack cannot be connected to a CVM1-BC103 or CVM1-BC053 Backplane, which support only a single Expansion I/O Rack connected directly to it. Expansion I/O Racks do not support Ethernet Units.

Expansion I/O Rack
Ethernet Units cannot be mounted to an Expansion I/O Rack.

PS:	Power Supply Unit
CPU:	CPU
IOC:	I/O Control Unit
IOIF:	I/O Interface Unit

2-3-2 Current Consumption

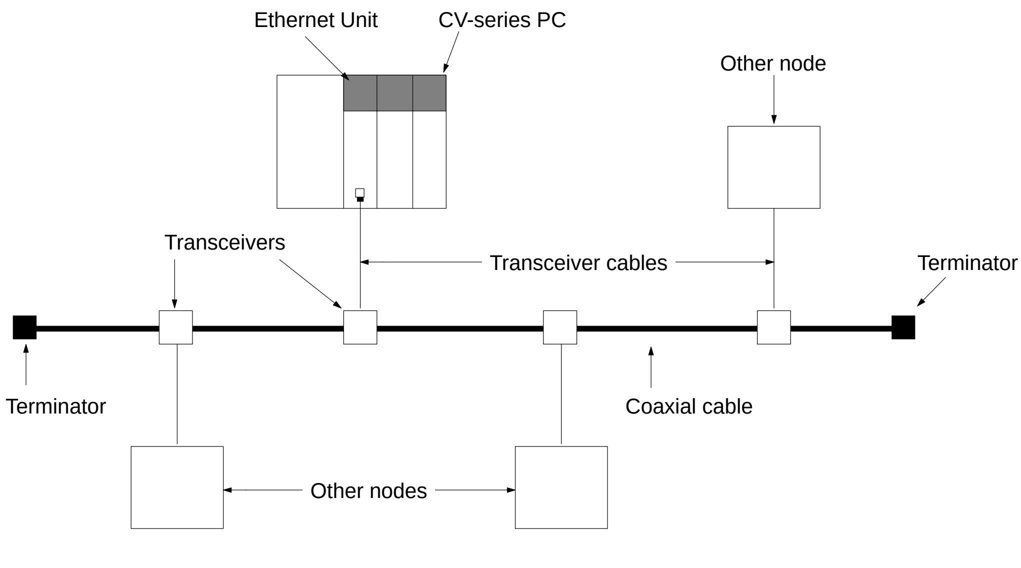
The maximum current consumption of the Ethernet Unit is 1.7 A. This includes the power supplied to the transceiver. The total current consumption for all Units on any Rack must not exceed the maximum supply capacity of the Power Supply Unit. Refer to the *CV-series PC Installation Guide* for details.

2-4 Connecting the Ethernet Network

This section describes how to connect the Ethernet Unit to an Ethernet network.

2-4-1 Ethernet Network Configuration

A basic Ethernet network consists of a coaxial cable to which transceivers have been connected to connect various nodes, as shown in the following diagram. A single length of coaxial cables and the transceivers, transceiver cables, and nodes connected to it are called a segment. The following diagram shows one segment.



Basic Ethernet Network: One Segment

The items shown in the above diagram are required to assemble an Ethernet network. Make sure that you have the sufficient quantities of the required items before attempting to install a system. All items must comply with IEEE802.3 standards.

Item	Description
Transceiver	Forms the interface between the coaxial cable and the node. The Ethernet Unit supplies power to the transceiver. Use a transceiver with a current consumption of 0.35 A or less. Consult the transceiver manufacturer about the current consumption.
Transceiver cable	Connects the transceiver to the node.
Coaxial cable	Forms the trunk line of the Ethernet network.
Terminators for coaxial cable	Connected to both ends of the coaxial cable.

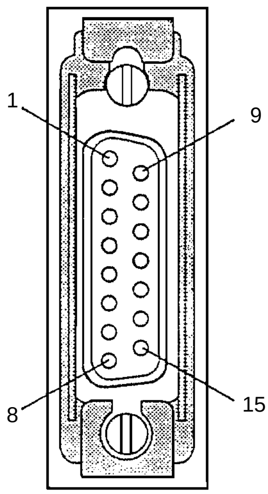
2-4-2 Connector Specifications

Electrical Characteristics Connector electrical characteristics conform to IEEE802.3 standards.

Locking Mechanism The connector locks with a slide latch that conforms to IEEE802.3 standards.

Signals

Connector signals are shown in the following diagram and table. Signal directions (input or output) in the table are described in reference to the Ethernet Unit.



Connector pin	Signal name	Abbr.	Signal direction
1	Signal ground	GND	---
13	Transceiver +12 VDC power supply	+12 V	Output
6	Power ground (common with signal ground)	GND	---
14	Signal ground	GND	---
2	Collision detection signal +	COL+	Input
9	Collision detection signal –	COL–	Input
8	Signal ground	GND	---
5	Receive data +	RX+	Input
12	Receive data –	RX–	Input
4	Signal ground	GND	---
3	Send data +	TX+	Output
10	Send data –	TX–	Output
11	Signal ground	GND	---
7, 15	Not used	---	---
Connector hood	Frame ground	FG	---

2-4-3 Preparing the Transceiver Cable

Required Parts

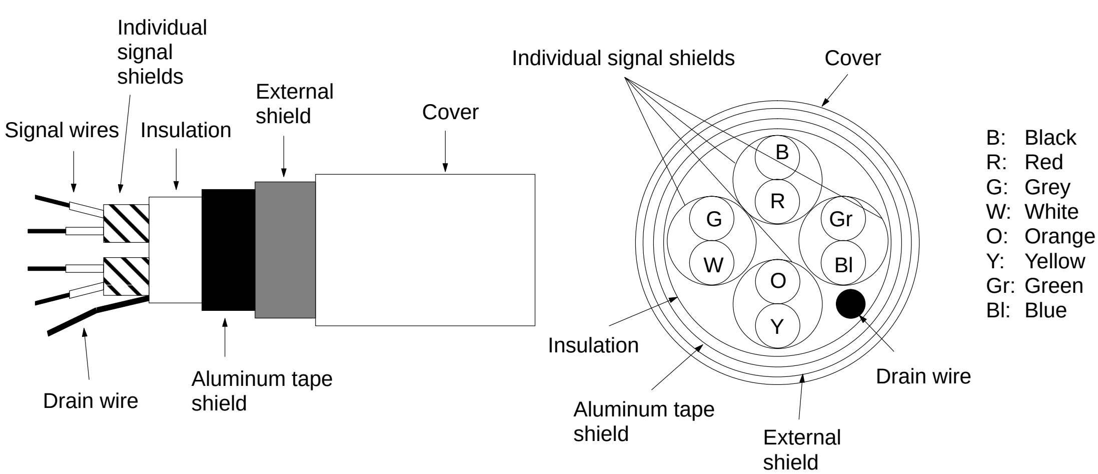
The following parts are required to prepare the transceiver cable.

Cable confirming to IEEE802.3 standards.
Dsub 15-pin male connector with locking posts. (Attached to Ethernet Unit end of the cable.)
Dsub 15-pin female connector with slide latch. (Attached to transceiver end of the cable.)

IEEE802.3 Cable Structure

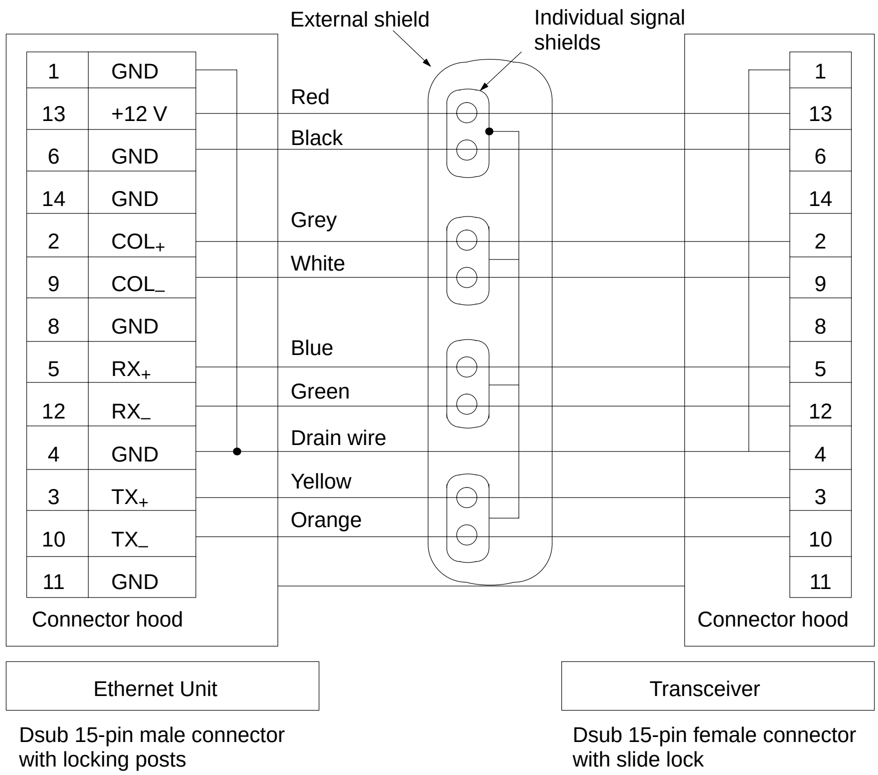
A cable conforming to IEEE802.3 standards includes a shield around each of the individual signal wires, as shown in the following example. In addition, an external shield is included around the insulation between the individual signal wires.

The individual signal shields must be electrically connected to each other and then connected to the drain wire. Always connect the individual signal shields to the drain wire and then connect the drain wire to pin 4 of the connector. The individual shields must be electrically insulated from the external shield.



Connecting the Cable

The following diagram shows an example of connecting the transceiver cable to a transceiver and Ethernet Unit. Connect the signal wires as shown.



Note

1. Connect the drain wire in the transceiver cable to pin #4 of the connector at each end of the cable.
2. Electrically insulate all the individual signal shields from the external shield.

2-4-4 Ethernet Cable Installation

Take the greatest care when installing the Ethernet System, being sure to follow ISO 8802-3 specifications. You must obtain a copy of these specifications and be sure you understand them before attempting to install an Ethernet System. Unless you are already experienced in installing communications systems, we strongly recommend that you employ a professional to install your system.

Caution Do not install Ethernet equipment near sources of noise. If noise-prone environments are unavoidable, be sure to take adequate measures against noise interference, such as installing network components in grounded metal cases, using optical links in the system, etc. Further information on measures that can be taken against noise is given later in this section.

Precautions

Coaxial Cable

Observe the following precautions when laying coaxial cable.

- As a rule, always lay coaxial cables indoors. Laying coaxial cables outdoors requires special measures that should be performed only by a professional. If coaxial cable must be used outdoors, request the work from a professional.
- Attach a terminator to each end of the coaxial cable.
- Cover all exposed metal portions of the coaxial cable except ground points with protective covers or PVC tape to prevent contact with grounds or other metal objects.
- Ground the external shield of the coaxial cable to 100 Ω or less using AWG 14 wire (cross-sectional area of 2 mm^2) or better and a ground cable length of

20 m or less. Ground each segment of the network at one location only. Do not use the same ground for other equipment.

- The physical properties (e.g., minimum bending radius) of coaxial cable vary with the manufacturer. Follow all recommendations and specifications of the manufacturer.
- Do not install coaxial cables together with power cables.
- Do not install coaxial cable near sources of noise.
- Do not install coaxial cable in environments subject to excessive dirt, dust, oil mist, etc.
- Do not install coaxial cable in environments subject to high temperatures or high humidity.

Terminators

Observe the following precautions when attaching terminators.

- You must attach a terminator to each end of the coaxial cable for communications to be possible.
- Insulate terminators when attaching them to be sure they do not come into contact with metal objects.

Transceivers

Observe the following precautions when installing transceivers.

- Transceivers can be attached to coaxial cables only at multiples of 2.5 m. These locations should be marked on the cable.
- Connection methods can vary with the manufacturer of the cable. Follow all recommendations and specifications of the manufacturer.
- Mount the transceiver onto an insulating object, such as a wooden board.
- Ground the transceiver through the shield of the transceiver cable and through the Ethernet Unit.
- Do not install transceivers near sources of noise.
- Do not install coaxial cable in environments subject to excessive dirt, dust, oil mist, etc.
- Do not install coaxial cable in environments subject to high temperatures or high humidity.

Transceiver Cables

Observe the following precautions when laying transceiver cables.

- Make sure that the transceiver cable is firmly and completely connected to both the transceiver and the Ethernet Unit. Be sure that the slide latches are engaged.
- Do not install transceiver cables together with power cables.
- Do not install transceiver cable near sources of noise.
- Do not install transceiver cable in environments subject to high temperatures or high humidity.

Grounds

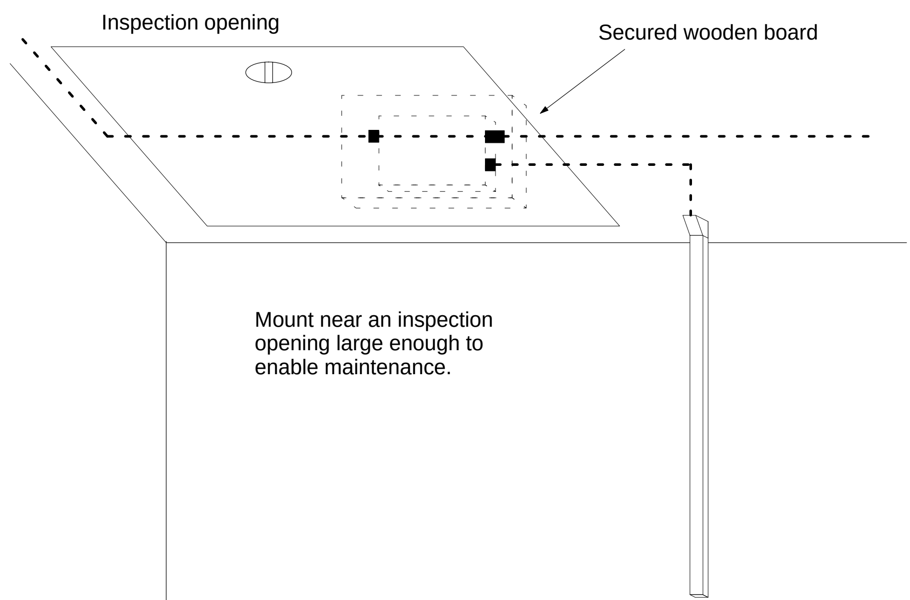
Observe the following precautions when grounding network devices.

- Ground all devices requiring grounding to 100 Ω or less using AWG 14 wire (cross-sectional area of 2 mm^2) or better and a ground cable length of 20 m or less. Electrical shock can occur between ungrounded devices in the system. The lack of proper grounds can also corrupt data.
- Use one safety ground on each segment of the network.
- Inductive voltages can be generated due to lightning surges when coaxial cable is installed outdoors, requiring special measures. Request outdoor installation from a professional.

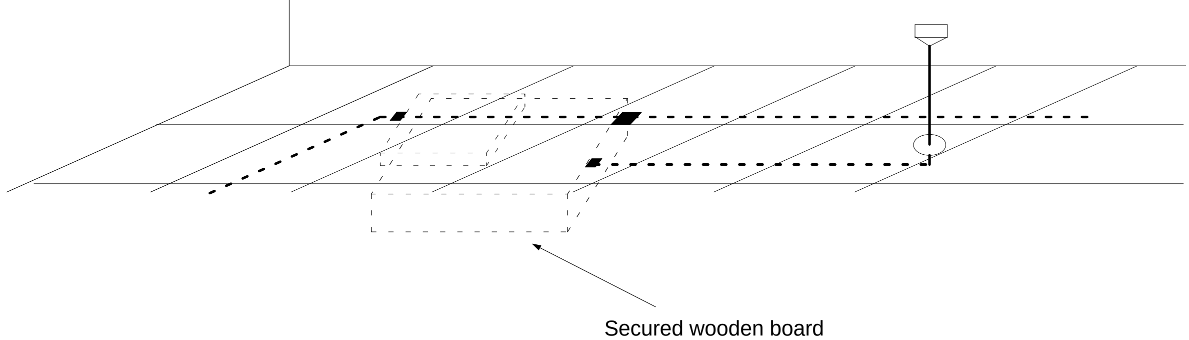
Transceiver Installation Examples

The following illustrations show examples of installing transceivers in the ceiling and under the floor.

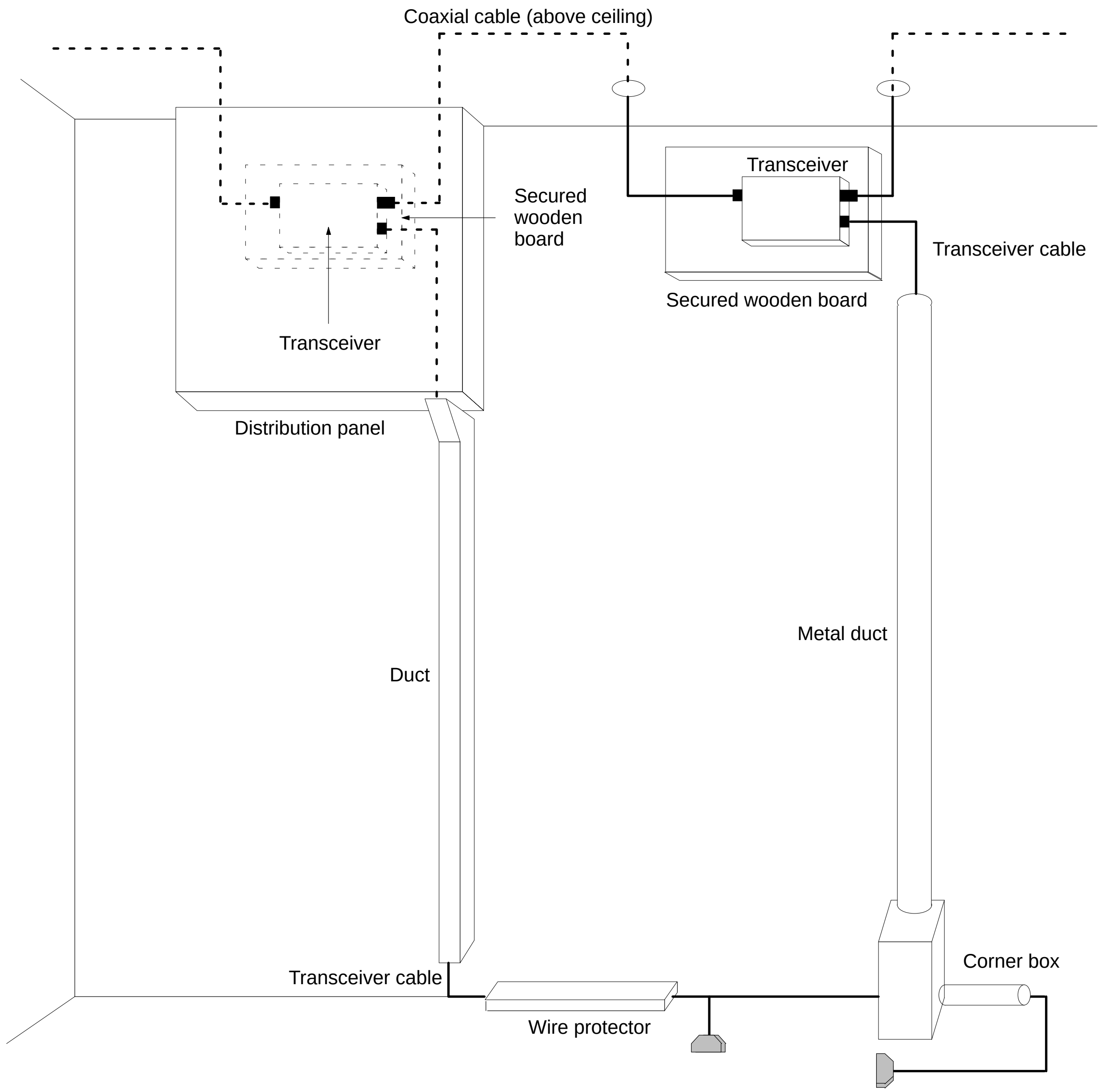
Ceiling Installation



Free-access Floor Installation



Ceiling Installation Examples



Installation Examples for Noise-prone Environments

Ethernet networks do not withstand noise very well and require special installation measures if installed where subject to noise. The remainder of this section describes measures for dealing with noise.

Caution Although the measures described in this section are suitable to certain levels of noise, programming retries in communications will still be required. Although some protocols provide for automatic retries, UDP/IP (FINS communications and UDP sockets) does not, and the user must program retries in user applications.

Metal Ducts

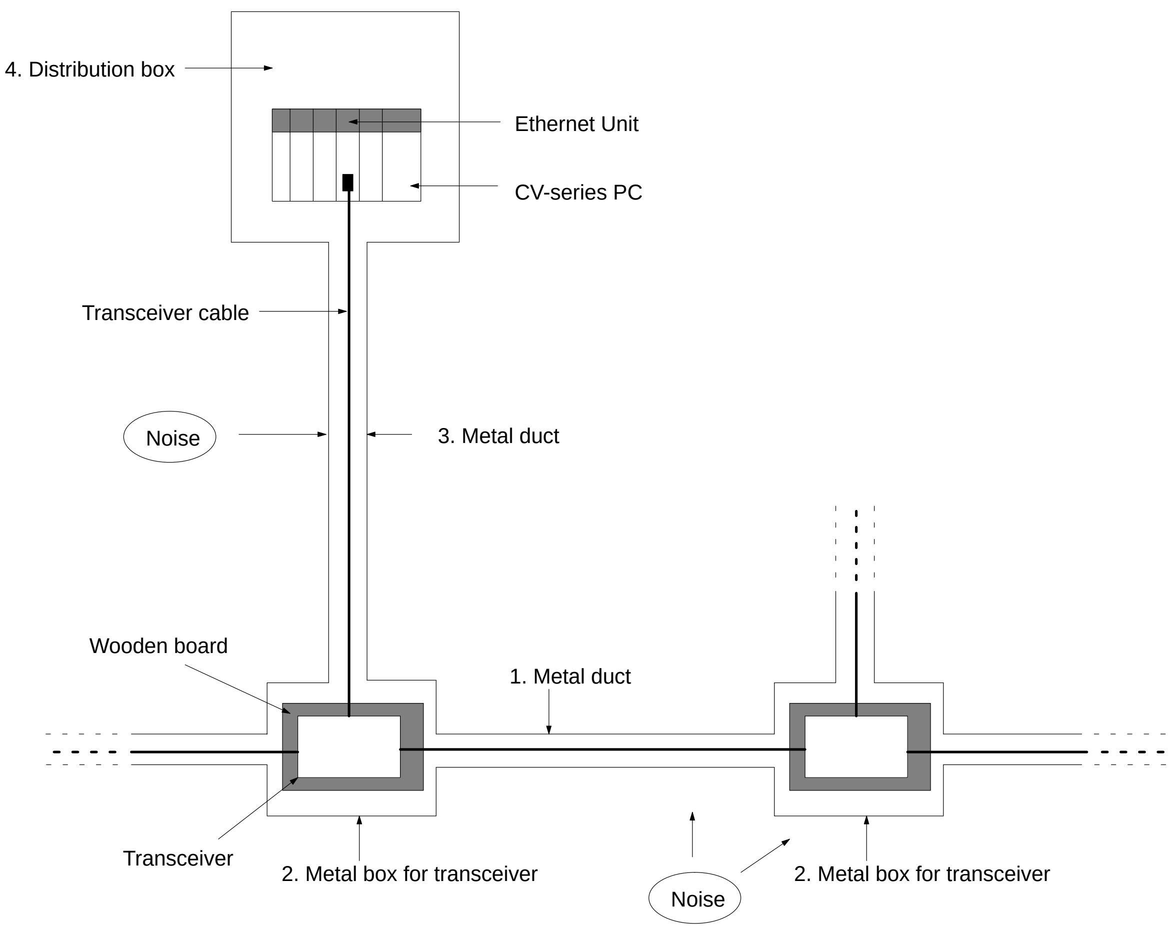
Cables and components can be placed in metal ducts to protect the network from noise.

1, 2, 3...

1. Place the coaxial cable in metal ducts.
2. Place transceivers into metal boxes, placing a wooden board or other insulating board between the box and the transceiver. The transceiver and the metal box must be electrically insulated from each other.
3. If the transceiver cable is also in a noise-prone area, place it into a metal duct as well.
4. Refer to the *CV-series PC Installation Guide* for information on wiring inside distribution boxes.

Note

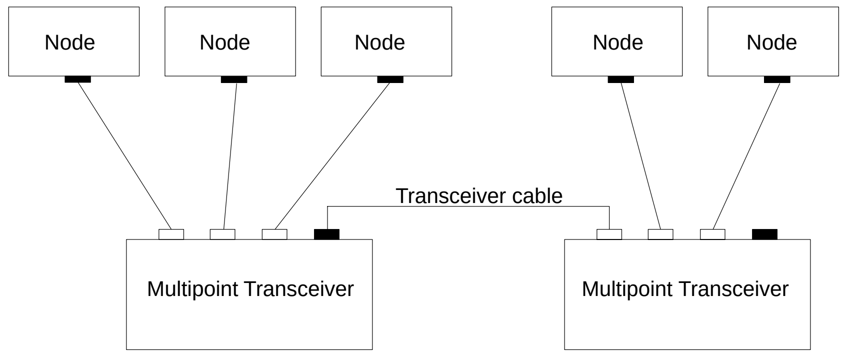
1. Cover all metal portions of cables with insulating tape to be sure they do not come into contact with the metal ducts or boxes.
2. Securely ground all distribution boxes and metal boxes around transceivers. The lower the ground resistance, the more effective the shields will be.



Multipoint Transceivers

Multipoint transceivers can be used to eliminate the need for coaxial cable (which is easily affected by noise) and replace it with transceiver cable as shown in the following diagram. This will increase overall system resistance to noise.

Note Up to two sets of multipoint transceivers can be used in a network. The length of the network will be limited to the maximum length of transceiver cable.



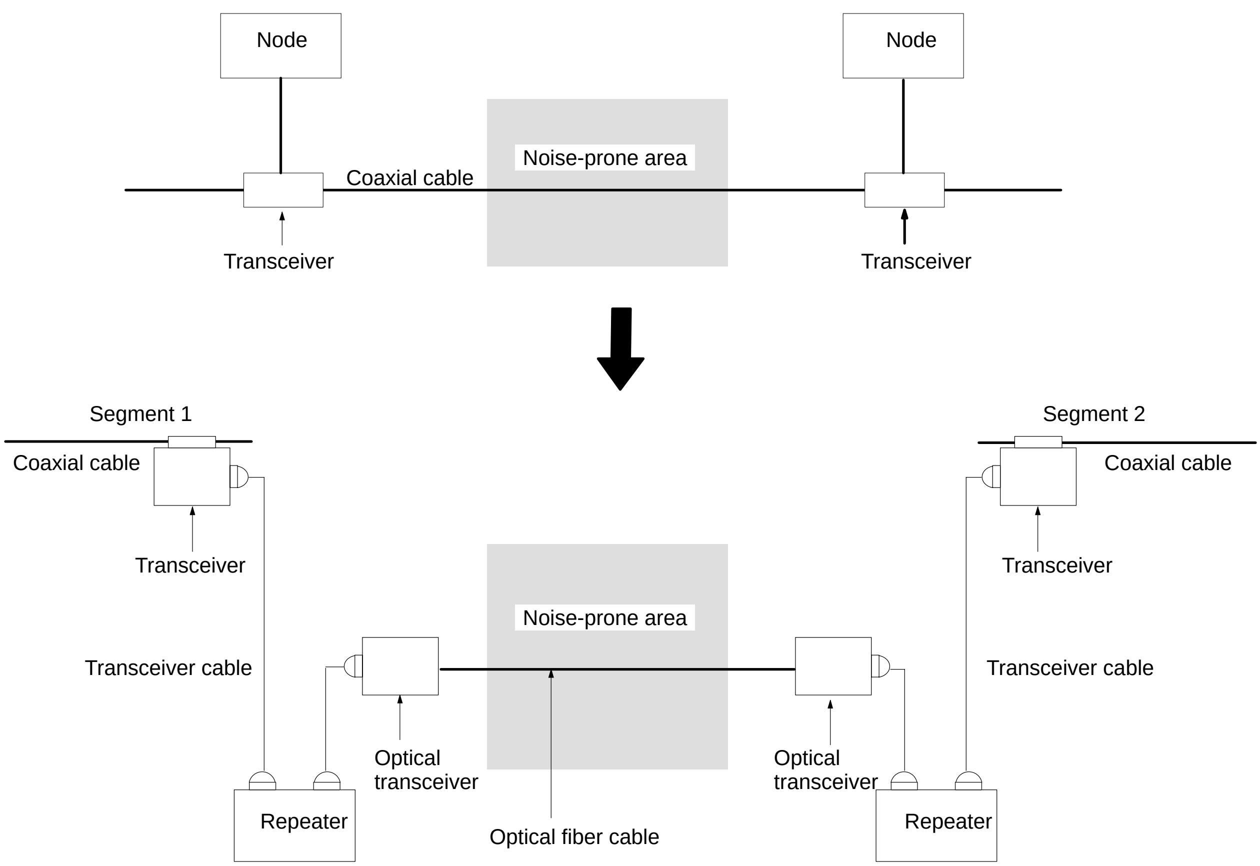
Optical Ethernet Systems

Some manufacturers provide optical fiber components for Ethernet networks to provide high resistance to noise. The following illustrations show some simple examples of optical installations. See the manufacturer for specifics.

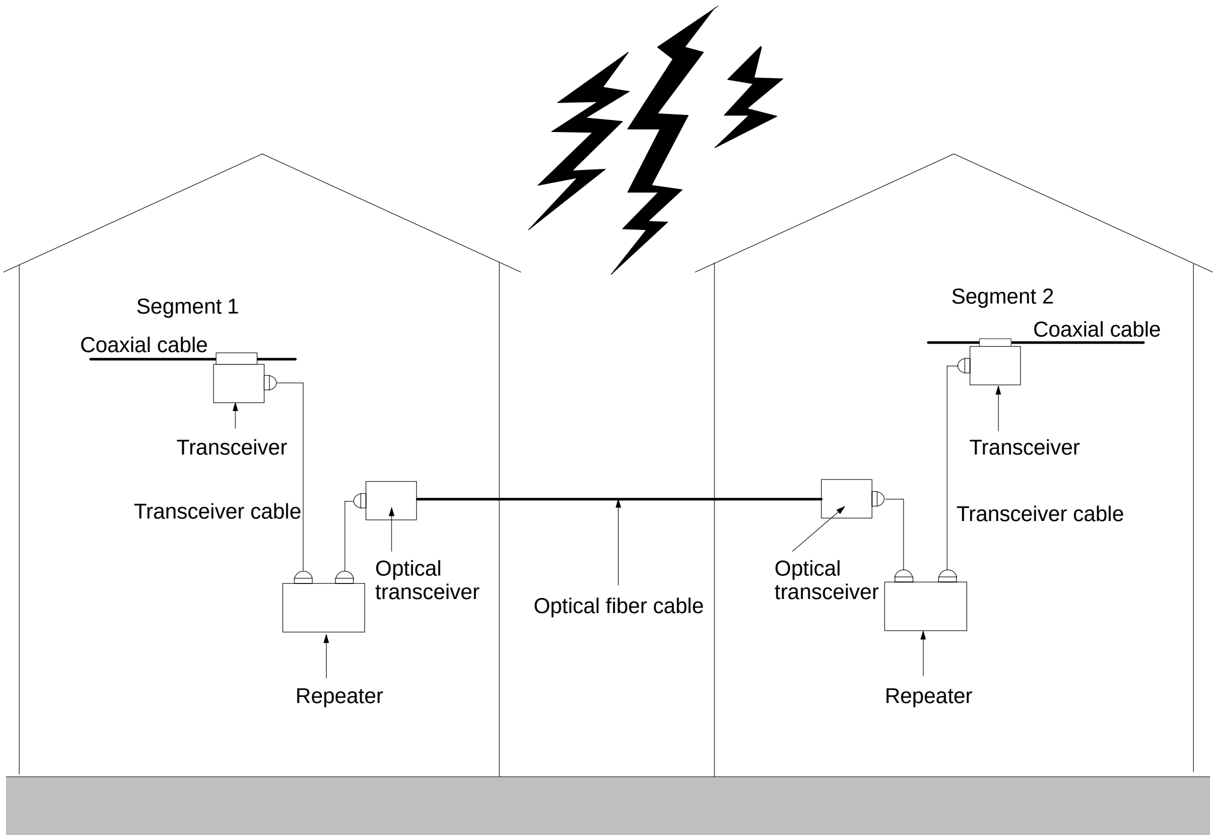
Note Optical transceivers must be installed away from sources of noise or protected from noise.

Use as Noise Countermeasure

The following illustration shows how optical components can be used to convert to optical fiber cable through areas highly prone to noise.



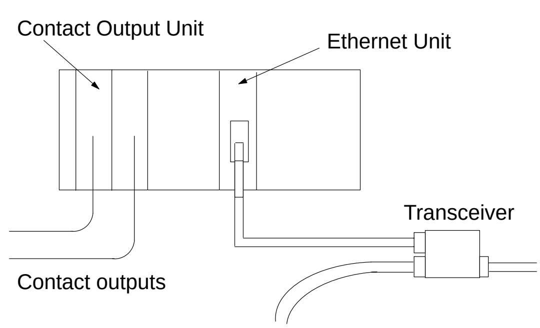
Outdoor Installations
The following example shows how optical fiber cable can be used to connect a network between two buildings, eliminating possible adverse affects of lightning.



Using Contact Outputs

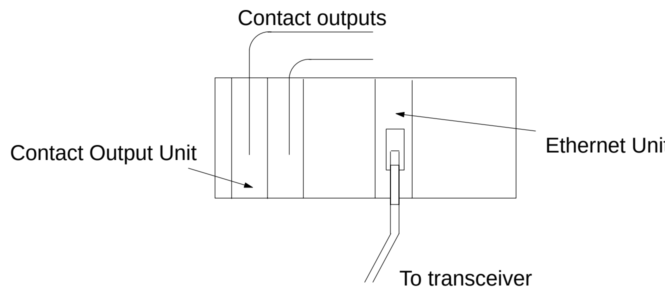
Communications errors can occur when Contact Output Units are mounted to the same Rack as an Ethernet Unit due to noise generated by the contact outputs. Use one or more of the following measures when installing Contact Output Units and Ethernet Units on the same Rack.

Mounting Location
Mount any Contact Output Units as far away from the Ethernet Unit as possible.



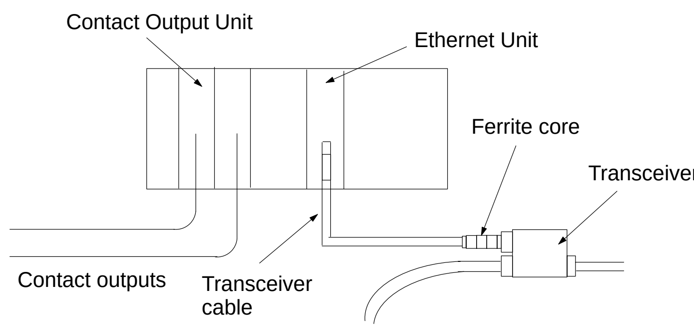
Cable Location

Separate the transceiver cable connecting the Ethernet Unit as far from the wiring to the Contact Output Units as possible. The coaxial cable must also be placed as far away from the Contact Output Units and their wiring as possible.



Transceiver Measures

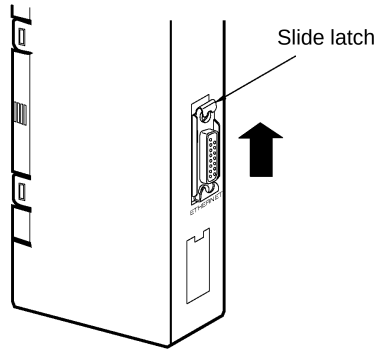
Attach several ferrite cores designed for EMI countermeasures on the transceiver cable near the transceiver. The transceiver must also be placed as far away from the Contact Output Units and their wiring as possible.



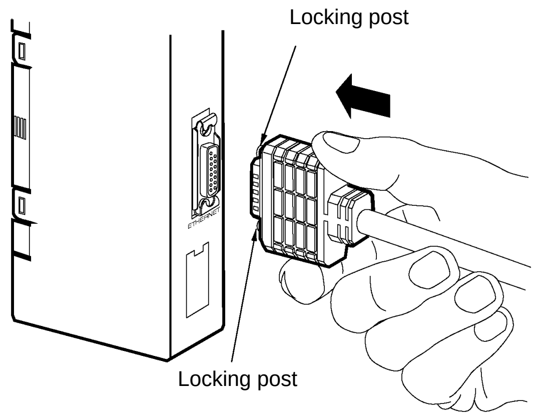
2-4-5 Ethernet Unit Connector

The transceiver cable is attached to the Ethernet Unit connector using the following procedure. The Ethernet Unit connector incorporates a slide latch locking mechanism confirming to IEEE802.3 standards. Operate the slide latch mechanism as described to connect the transceiver cable.

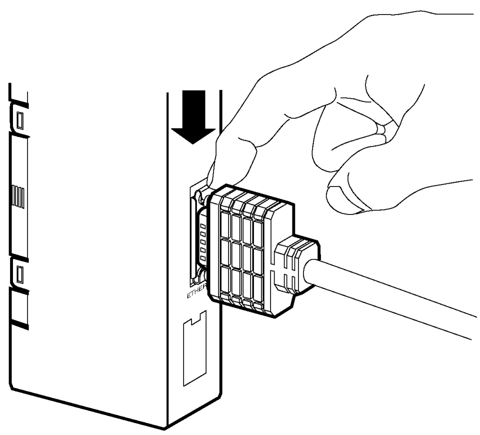
1, 2, 3... 1. Push the slide latch up before inserting the connector.



2. Insert the cable connector into the connector on the Ethernet Unit so that the locking posts on the tip of the cable connector align with the holes in the slide latch.



3. Push down the slide latch to lock the connector.



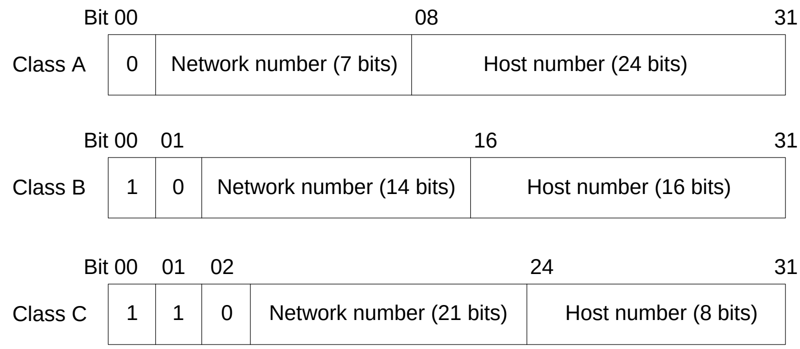
2-5 IP Addresses

Ethernet networks use IP addresses for communications. IP addresses (Internet addresses) identify both the Ethernet network and the node (host computer, Ethernet Unit, etc.) on the Ethernet network. IP addresses must be carefully set and managed to ensure that no addresses are duplicated.

2-5-1 IP Address Configuration

IP addresses are made up of 32 bits of binary data divided into four 8-bit fields called octet. These four octets provide the network number and host number. The network number identifies the network and the host number identifies the node (or host) on the network.

The network numbers in an IP addresses are divided into three classes depending on the scale of the network: Class A, Class B, and Class C. The configuration of the IP address for each of these classes is shown in the following diagram.



The number of networks in each class and the number of nodes possible on the network differ according to the class.

Class	No. of networks	No. of hosts
Class A	Small	2^{24} – 2 max. (16777214 max.)
Class B	Medium	2^{16} – 2 max. (65534 max.)
Class C	Large	2^{8} – 2 max. (254 max.)

IP addresses are represented by the decimal equivalent of each of the four octets in the 32-bit address, each separated by a period. For example, the binary address `10000010 00111010 00010001 00100000` would be represented as 130.58.17.32.

Note
1. The same network number must be set for every node on the same Ethernet network.
2. The network number of the IP address identifies the Ethernet network (IP network segment, see note 3.) and is NOT the same as the network address set in routing tables and used for FINS communications.
3. A network segment is a logical network consisting of all nodes that have the same network number.

2-5-2 Allocating IP Addresses

IP (Internet Protocol) is a standard communications protocol used throughout the world and is designed to enable communications between any Ethernet nodes regardless of the networks on which they exist. To achieve this, network numbers are allocated by the Network Solutions, InterNIC Registration Services, to ensure that all Ethernet networks have unique numbers regardless of where they exist. The local system administrator is left the responsibility of allocating unique host numbers locally. You therefore should obtain a network number from the InterNIC Registration Services to ensure uniqueness and allow for future network expansions if required.

To apply for a network number, request the NETINFO/INTERNET-NUMBER-TEMPLATE.TXT by electronic mail from HOSTMASTER@INTERNIC.NET and you will receive the proper application form. This form (see *Appendix I IP Network Address Request Form*) can then be submitted to receive a network number that is guaranteed to be unique from everyone else to which the InterNIC Registration Services have allocated a number.

You will undoubtedly need only one network number even if you need to establish multiple networks because you can use subnet numbers to establish multiple local networks under the same network number, as described later in this manual.

2-5-3 IP Address Settings

An IP address must be set for the Ethernet Unit before Ethernet communications can proceed. The IP address is set for each Ethernet Unit in the CPU Bus Unit system setup made from the CVSS or other Programming Device. Refer to *2-6 Ethernet Unit Settings* for more details about the setting procedure.

2-5-4 Subnet Numbers

Operation and management of a network may become very difficult if too many nodes are connected on a single network or if a single organization has to manage too many network numbers. It can therefore be convenient to divide a single network up into several subnetworks by using part of the host number as a subnet number. Internally the network can be treated as a number of subnetworks, but from the outside it acts as a single network and uses only a single network number.

To establish subnetworks, the host number in the IP address is divided into a subnet number and a host number by using a setting called the subnet mask. The subnet mask indicates which part of the host number is to be used as the subnet number. The user must first determine the number of bits of the host number to be used as the subnet number and then set the subnet mask accordingly. All bits in the subnet mask that correspond to the bits in the IP address used either as the network number or subnet number are set to "1" and the remaining bits, which will correspond to the bits in the IP address actually used for the host number, are set to "0".

The following example shows the subnet mask for an 8-bit subnet number used in a class-B IP address. This subnet mask is structured as follows:

- The first two octets (16 bits) are set to 1 to indicate the portion of the IP address that corresponds to the network number.
- The next octet (8 bits) is set to 1 to indicate the portion of the host number that is to be used as a subnet number.
- The last octet is set to 0 to indicate the portion of the host number to actually be used as the host number.

This would thus be a class-B IP address masked as a class-C IP address, i.e., externally it would be allocated a class-B IP address but internally it can be addressed using class-C IP addresses.

Subnet mask: `11111111 11111111` `11111111` `00000000` (FFFFFF00)

It is only necessary to set subnet masks if subnetworks are used. If a subnet mask is not set by the user, a default mask will be set automatically according to the IP address class to indicate that the entire host number will be used as the host number, i.e., no bits will be assigned for use as the subnet number.

All nodes on the network that are going to belong to the same subnetwork must have the same subnet mask.

2-6 Ethernet Unit System Setup

To use the Ethernet Unit as a node on an Ethernet network, make the relevant system settings using a Programming Device (SYSMAC-CPT, SYSMAC Support Software, or CV Support Software Version 2 or later).

Note
1. For Programming Device operating instructions, refer to *SYSMAC CVM1/CV–series Support Software Operation Manual Network Version* (Cat. No. W201), or the section on CPU Special I/O Unit system settings in the *C-series and CVM1 PCs SYSMAC Support Software Operation Manual* (Cat. No. W249).
2. After making the system settings, turn OFF the power supply to the PC, and then turn it ON again. (System settings are not enabled merely by saving them.)

2-6-1 System Setup Configuration

The system setup values are stored beginning from the first byte in the system setup area in the order shown in the following table.

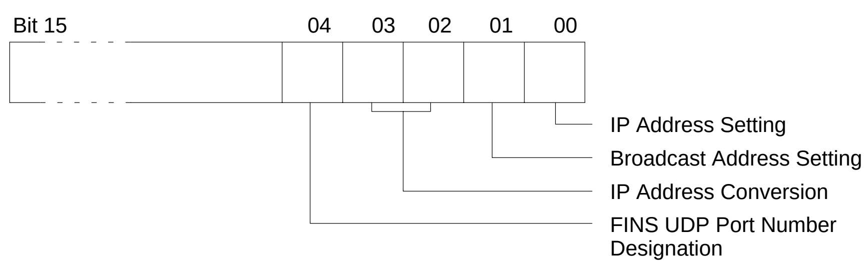
Position relative to first byte	Setting	Size
+0	Mode setting	2 bytes
+2	Not used.	6 bytes
+8	Local IP address	4 bytes
+12	Subnet mask	4 bytes
+16	FINS UDP port #	2 bytes
+18	FTP login name	12 bytes
+30	FTP password	8 bytes
+38	IP address table	194 bytes
+232	IP router table	66 bytes

2-6-2 Individual Settings

The following items can be set for the Ethernet Unit.

Mode Setting

Set bits #0 to #4 as described below.



Bit 00, IP Address Setting: Specifies whether the node number set on the rotary switches on the front panel of the Unit is to be used as the node number in the IP address. Set this bit to 0 if automatic generation is specified as the IP address conversion method.

Bit	Setting
0	Node number setting rotary switch used
1	Node number setting rotary switch not used

Bit 01, Broadcast Address Setting: Specifies when broadcasts are run.

Bit	Setting
0	Broadcast when the host number is all ones (4.3BSD specification)
1	Broadcast when the host number is all zeros (4.2BSD specification)

Note The broadcast address is the address to which the host number is set to send a transmission simultaneously to all nodes on a network. Either all ones or all zeros can be used for the broadcast address. This setting must be the same for all Ethernet Units and other nodes that are to be part of the broadcast. Refer to your host computer's documentation to specify the broadcast address for it.

Bits 02 and 03, Destination IP Address Conversion: Specifies the method for converting the FINS destination node number to an IP address.

Bits		Setting
03	02	
0	0	Automatic conversion: IP addresses will be generated automatically from the FINS node numbers. The IP address table will not be used.
0	1	
1	0	IP address table used.
1	1	IP address table and automatic generation used.

Note Refer to *Section 3 FINS Communications Service* for details about destination IP address conversion.

Bit 04, FINS UDP Port Number Designation: Specifies the method by which the UDP port number is specified for the FINS communications service.

Bit 04	Setting
0	Default value (9600)
1	Use value set in system setup

Local IP Address

Set the IP address for the Ethernet Unit. The Ethernet Unit cannot communicate unless this address is set. The ERH indicator will flash if no IP address is set or if an illegal IP address is set.

The following IP addresses cannot be set:

- Addresses with all bits of the host number set to 0 or 1
- Addresses with all bits of the network number set to 0 or 1
- Addresses with all bits of the subnet number set to 1
- IP addresses starting with 127 (7F hexadecimal), for example 127.35.21.16

If bit 0 of the mode setting is set to 0 (OFF) in the first system setup word, the node number set on the rotary switches on the front panel of the Unit will be automatically input as the host number in the fourth byte (octet) of the local IP address. If the node number set on the Unit is used as the host number, set only the most significant 3 bytes of the address in the local IP address setting; the IP address will automatically be updated each time the node number is changed on the Unit's rotary switches, eliminating the need to change the system setup.

Subnet Mask

Set all bits in the subnet mask that correspond to the bits in the IP address used for the network number and the bits used for the subnet number to "1" and set all bits in the subnet mask that correspond to the bits in the IP address used for the host number to "0".

The subnet mask must be set only when configuring a system which runs using subnetworks. If no subnet mask is set, the subnet mask will be automatically set to one of the following values, depending on the local IP address setting:

- Class-A IP Address: FF000000
- Class-B IP Address: FFFF0000
- Class-C IP Address: FFFFFF00

FINS UDP Port Number

Set the UDP port number used for the FINS communications service. This port number is valid if the FINS UDP port number designation bit (bit #4) in the mode settings is set to 1 in the first system setup word. The FINS UDP port number cannot be set to 0. If a value of 0 is set, the default value (9600) will be used as the FINS UDP port number. We recommend that the FINS UDP port number be set to a value between1024 and 65535.

Note Set the UDP port number of all nodes (Ethernet Units) using FINS communications on the Ethernet network to the same FINS UDP port number. FINS communications is not possible between two nodes with different FINS UDP port numbers.

Sample Setting Display

The screen display produced by the CVSS to set the CPU Bus Unit system setup will appear as shown in the following diagram. This diagram shows data input for the settings given for each item.

```
                  [CPU SIOU Unit System Setup]
unit # 00 ET

BYTE   b7         b0    HEX    BYTE   b7         b0    HEX
+  0   [0000  0000]      00    + 10   [0010  0100]      24
+  1   [0000  0001]      01    + 11   [0000  0101]      05
+  2   [0000  0000]      00    + 12   [0000  0000]      00
+  3   [0000  0000]      00    + 13   [0000  0000]      00
+  4   [0000  0000]      00    + 14   [0000  0000]      00
+  5   [0000  0000]      00    + 15   [0000  0000]      00
+  6   [0000  0000]      00    + 16   [0000  0000]      00
+  7   [0000  0000]      00    + 17   [0000  0000]      00
+  8   [1000  0010]      82    + 18   [0000  0000]      00
+  9   [0001  1001]      19    + 19   [0000  0000]      00
```

Mode Setting (bytes +0 to +1)
Unit number (rotary switch setting) not used as host number in IP address and IP address table used for IP address conversion

Local IP Address (bytes +8 to +9)
130.25...

Local IP Address (contd.) (bytes +10 to +11)
...36.5

Subnet Mask (bytes +12 to +15)
Default (Not set.)

FINS UDP Port Number (bytes +16 to +17)
Default (Not set.)

FTP Login Name

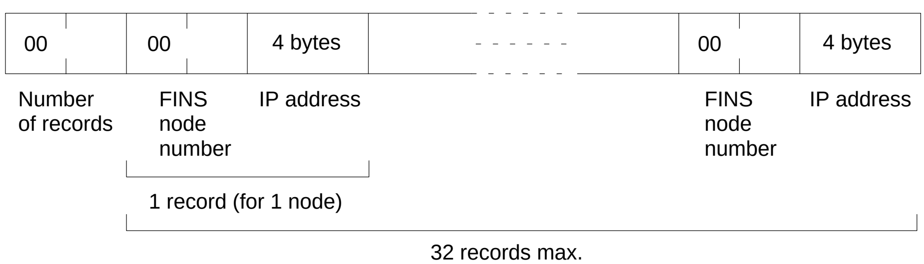
Set the login name of the Unit's FTP server. Refer to *4-2 Setting Login Names and Passwords* for details. If no login name is specified, the default login name of "CONFIDENTIAL" will be used and no FTP password will be required. Any password that is set for the default login name will be ignored.

FTP Password

Set the password for the Unit's FTP server. Refer to 4-2 *Setting Login Names and Passwords* for details.

IP Address Table

This table contains the conversion data used to generate IP addresses from FINS node numbers when using FINS communications. This table is ignored if the IP address conversion method is set to automatic generation only. The configuration of the IP address table is shown in the following diagram.



The first two bytes specify how many records (nodes) are stored in the table. This is followed by a record for each node. Each record consists of an FINS node number (2 bytes with the most significant byte always 00) and the corresponding IP address (4 bytes). The records are repeated for the specified number of records, up to the maximum of 32 records (nodes).

Sample Setting Display

The screen display produced by the CVSS to set the CPU Bus Unit system setup will appear as shown in the following diagram. The display shows data input for the settings shown in the following table.

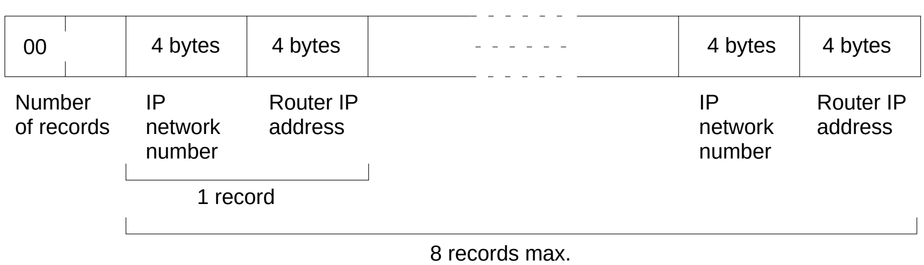
FINS node number	IP address
15	130.25.36.50
16	130.25.36.5
17	130.25.36.88

```
                [CPU SIOU Unit System Setup]
unit # 00 ET

+ 27  [0000  0000]    00   + 37  [0000  0000]    00
+ 28  [0000  0000]    00   + 38  [0000  0000]    00   ← Number of records
+ 29  [0000  0000]    00   + 39  [0000  0011]    03
```

```
                      [CPU SIOU Unit System Setup]
                  unit # 00 ET

                  BYTE  b7         b0  HEX   BYTE  b7         b0  HEX
Node number  →    + 40  [0000  0000]    00   + 50  [0010  0100]    24   ← IP Address (contd.)
                  + 41  [0000  1111]    0F   + 51  [0000  0101]    05
IP Address   →    + 42  [1000  0010]    82   + 52  [0000  0000]    00   ← Node Number
                  + 43  [0001  1001]    19   + 53  [0001  0001]    11
                  + 44  [0010  0100]    24   + 54  [1000  0010]    82   ← IP Address
                  + 45  [0011  0010]    32   + 55  [0001  1001]    19
Node number  →    + 46  [0000  0000]    00   + 56  [0010  0100]    24
                  + 47  [0001  0000]    10   + 57  [0101  1000]    58
IP Address   →    + 48  [1000  0010]    82   + 58  [0000  0000]    00
                  + 49  [0001  1001]    19   + 59  [0000  0000]    00
```

IP Router Table

The IP router table sets how the Ethernet Unit communicates via the IP router with nodes on other IP network segments. The configuration of the IP router table is shown in the following diagram.

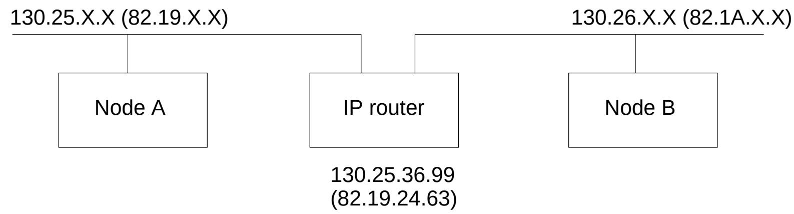


The first two bytes specify how many records (up to the maximum of 8) are stored in the table. This is followed by a record for each network segment with which communications are to be conducted. Each record consists of the IP network number (4 bytes) for another network and the IP address (4 bytes) of the router that communications must first pass through to get to the network segment. The records are repeated for the specified number of records.

The length of the network numbers (in bytes) of the other networks to be communicated with depend on the class of the IP addresses. Although a total of four bytes are allowed for the IP network number, set only the required number of bytes for the network number from the start of the allocated space and fill any remaining bytes with zeros.

Sample Setting Display

The screen display produced by the CVSS to set the CPU Bus Unit system setup will appear as shown in the following diagram for node A. The display shows data input for the settings shown in the following diagram: Node A is set to communicate with the network with the network number 130.26 (hexadecimal: 82.1A) via the IP router with IP address 130.25.36.99 (hexadecimal: 82.19.24.63).



[CPU SIOU Unit System Setup]

unit # 00 ET

BYTE	b7 b0	HEX	BYTE	b7 b0	HEX
+ 220	[0000 0000]	00	+ 230	[0000 0000]	00
+ 221	[0000 0000]	00	+ 231	[0000 0000]	00
+ 222	[0000 0000]	00	+ 232	[0000 0000]	00
+ 223	[0000 0000]	00	+ 233	[0000 0001]	01
+ 224	[0000 0000]	00	+ 234	[1000 0010]	82
+ 225	[0000 0000]	00	+ 235	[0001 1010]	1A
+ 226	[0000 0000]	00	+ 236	[0000 0000]	00
+ 227	[0000 0000]	00	+ 237	[0000 0000]	00
+ 228	[0000 0000]	00	+ 238	[1000 0010]	82
+ 229	[0000 0000]	00	+ 239	[0001 1001]	19

Number of Records

IP Network Number

Router IP Address

[CPU SIOU Unit System Setup]

unit # 00 ET

BYTE	b7 b0	HEX	BYTE	b7 b0	HEX
+ 240	[0010 0100]	24	+ 250	[0000 0000]	00
+ 241	[0110 0011]	63	+ 252	[0000 0000]	00
+ 242	[0000 0000]	00	+ 252	[0000 0000]	00

Router IP Address (contd.)

2-7 CPU Bus Unit Data Areas

The Ethernet Unit is located specific words in the CPU Bus Unit Area and DM Area of the PC to store TCP/UDP socket status information and internode test status information. The configuration of the words allocated to the Unit in these areas is described in this section.

2-7-1 Allocated Words

Words are allocated to the Ethernet Unit according to the unit number of the Unit, as shown in the following tables. Each Unit is allocated 25 words of the CPU Bus Unit Area and 100 words of the DM Area.

CPU Bus Unit Area

Unit number	Allocated words	Unit number	Allocated words
0	CIO 1500 to CIO 1524	8	CIO 1700 to CIO 1724
1	CIO 1525 to CIO 1549	9	CIO 1725 to CIO 1749
2	CIO 1550 to CIO 1574	10	CIO 1750 to CIO 1774
3	CIO 1575 to CIO 1599	11	CIO 1775 to CIO 1799
4	CIO 1600 to CIO 1624	12	CIO 1800 to CIO 1824
5	CIO 1625 to CIO 1649	13	CIO 1825 to CIO 1849
6	CIO 1650 to CIO 1674	14	CIO 1850 to CIO 1874
7	CIO 1675 to CIO 1699	15	CIO 1875 to CIO 1899

DM Area

Unit number	Allocated words	Unit number	Allocated words
0	D2000 to D2099	8	D2800 to D2899
1	D2100 to D2199	9	D2900 to D2999
2	D2200 to D2299	10	D3000 to D3099
3	D2300 to D2399	11	D3100 to D3199
4	D2400 to D2499	12	D3200 to D3299
5	D2500 to D2599	13	D3300 to D3399
6	D2600 to D2699	14	D3400 to D3499
7	D2700 to D2799	15	D3500 to D3599

2-7-2 Area Configuration

The information contained in the CPU Bus Unit Area is described next.

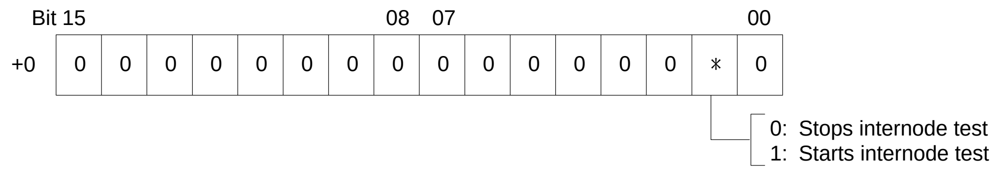
CPU Bus Unit Area

The status information listed below is stored in the CPU Bus Unit Area words allocated to each Unit. The first word is calculated using the following formula: First word = 1500 + (25 x unit number). All other words are calculated by adding the specified offset to the address of the first word.

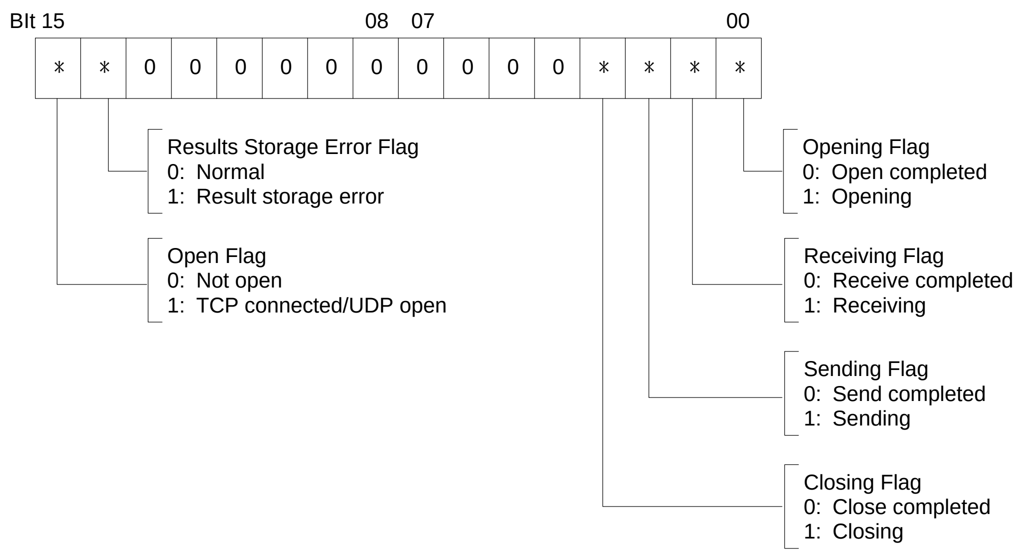
Offset	Content
+0 (first word)	Internode Test Start Bit
+1	Status of UDP socket #1
+2	Status of UDP socket #2
+3	Status of UDP socket #3
+4	Status of UDP socket #4
+5	Status of UDP socket #5
+6	Status of UDP socket #6
+7	Status of UDP socket #7
+8	Status of UDP socket #8
+9	Status of TCP socket #1
+10	Status of TCP socket #2
+11	Status of TCP socket #3
+12	Status of TCP socket #4
+13	Status of TCP socket #5
+14	Status of TCP socket #6
+15	Status of TCP socket #7
+16	Status of TCP socket #8
+17	FTP server status

The bit configurations of each word are described next.

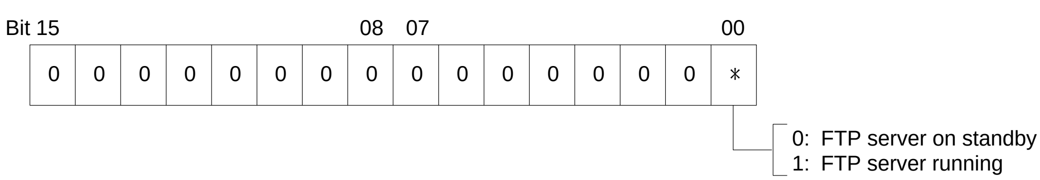
Internode Test Start Bit



UDP/TCP Socket Status



FTP Server Status



DM Area (Software Switches)

The information listed below is stored in the DM Area words allocated to each Unit starting at the first word. The first word is calculated using the following formula: First word = D02000 + (100 x unit number). All other words are calculated by adding the specified offset to the address of the first word.

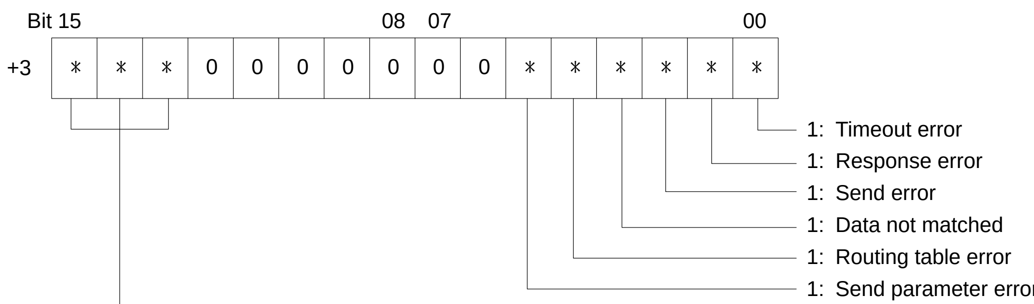
Offset	Contents	
+0	Bits 8 to 15: Target network address	Bits 0 to 7: Target node number
+1	Number of send bytes	
+2	Response monitor time	
+3	Test status	
+4	Number of internode test runs	
+5	Number of timeout errors	
+6	Number of response errors	
+7	Number of send errors	
+8	Number of times data did not match	

Words +0 through +2 must be set when running an internode test. The set parameters and their setting ranges are shown in the following table.

Parameter	Range
Target network address	01 to 7F (1 to 127 decimal)
Target node number	01 to 7E (0 to 126 decimal)
Number of send bytes	0001 to 07CC (1 to 1996 decimal) 0000 specifies the max. length of 1,996 bytes.
Response monitor time (Unit: 0.1 s)	0001 to FFFF (1 to 65535 decimal) 0000 specifies 2 s

Note The network address used here is the address used by the FINS communications service. It is NOT the IP address network number.

The following diagram shows the bit configuration of the test status (word+3), which indicates the results of the internode test.



Bits			Meaning
15	14	13	
0	0	0	Normal
0	0	1	Timeout error
0	1	0	Response (response code) error
0	1	1	Send error
1	0	0	Data not matched
1	0	1	Routing table error
1	1	0	Send parameter error

2-8 Initial Startup Procedures

The following procedures must be followed when installing and starting an Ethernet Unit for the first time. Refer to the specified pages for details. Refer to the CVSS Operation Manuals for details on procedures for setting the system setup.

Note Always restart the PC after changing any system setup parameters. All settings are stored in the CPU Unit. You must make the settings again each time you replace the CPU Unit.

2-8-1 Ethernet Unit

1, 2, 3...

1. Set the node number and unit number switches on each Ethernet Unit in the system. (Page 9.)
2. Set the IP address for each Ethernet Unit in the system. (Pages 9 and 22.)
3. Mount all Ethernet Units to the appropriate Racks and turn ON power to the PCs. (Page 12.)
4. Connect a computer running the CVSS to each PC and generate I/O tables. (Refer to CVSS operation manuals for details.)

2-8-2 Setting the CPU Bus Unit System Setup

This section describes how to set the system setup settings required for Ethernet Unit operation. The *General Settings* must be set for any Ethernet System. The other settings are required only if the feature indicated is going to be used.

General Settings

1, 2, 3...

1. Set bit 0 in the Ethernet Unit modes in the first word of the system setup to specify if the unit number setting on the Ethernet Unit is to be used as the host number. (Page 25.)
2. Set the IP address. Only the network number (and subnet number, if subnetworks are being used) needs to be set if the unit number setting on the Ethernet Unit has been enabled for use as the host number in the mode settings in step 1. (Page 26.)
3. Set a subnet mask if subnetworks are going to be used. (Pages 24 and 26.)

Settings for FINS Communications (Using SEND(192)/RECV(193)/CMND(194))

1, 2, 3...

1. Set bit 4 in the Ethernet Unit modes in the first word of the system setup to specify if the default FINS port number is to be used. (Page 25.)
2. If the default FINS port number is not to be used, set the desired port number. (Page 26.)
3. Set the IP Address Conversion method in bits 2 and 3 in the Ethernet Unit modes in the first word of the system setup. (Page 26.)
4. If an IP address table is specified for use in IP address conversion in step 3. (setting of “10” or “11”), input the IP address table. (Page 27.)
5. Set routing tables. If communications are to be conducted only within the Ethernet System, set only the local routing table. (Page 41.)

or If communications are to be conducted with other OMRON networks (e.g., other Ethernet network, SYSMAC NET networks, or SYSMAC LINK networks), set both local and relay routing tables. (Page 41.)

Settings for Socket Services

Set a local routing table. (Page 41.)

Settings for FTP Server Services

Set a login name and password if a login name other than the default (CONFIDENTIAL) is to be used. (Page 27.)

Settings for Communications with Other Network Segments via IP Routers

Set the IP router table (Pages 28 and 40.)

SECTION 3
FINS Communications Service

This section provides information on communicating on Ethernet Systems and interconnected networks using FINS commands. The information provided in the section deals only with FINS communications in reference to Ethernet Units. Refer to the *FINS Command Reference Manual* for details on FINS commands for other Units and networks.

FINS commands issued from a PC are sent via the SEND(192), RECV(193), and CMND(194) instructions programmed into the user ladder-diagram program. Although an outline of these instructions is provided in this section, refer to the *CV-series PC Operation Manual: Ladder Diagrams* for further details on ladder-diagram programming.

3-1 Introduction

The FINS communications service was developed by OMRON for its factory automation networks. FINS communications allow PCs on these networks to be controlled by reading or writing memory area data without the need to program these operations into the PC user program.

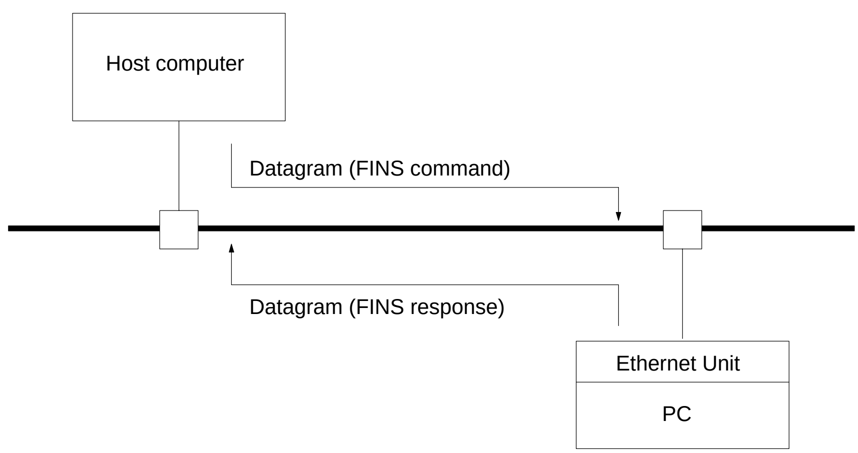
FINS communications use a unique set of addresses which differ from the address system of the Ethernet network. This different addressing system was implemented to provide a consistent communications method that can be used regardless of whether the PC at the target node is on an Ethernet network or is on another FA network, such as a SYSMAC NET or SYSMAC LINK network.

UDP/IP Communications

Communications Units that support FINS communications provide the FINS communications service through the features of the individual networks. Ethernet Units provide the FINS communications service via UDP/IP ports. Therefore, when a host computer communicates with a PC, for example, it is able to read or write the PC memory data and control the PC simply by sending a datagram containing the FINS commands to the UDP port for FINS communications with the Ethernet Unit.

A datagram is the unit of data handled by UDP/IP communications through the port specified for UDP communications. A communications service using UDP/IP normally allocates unique communications ports to run the service.

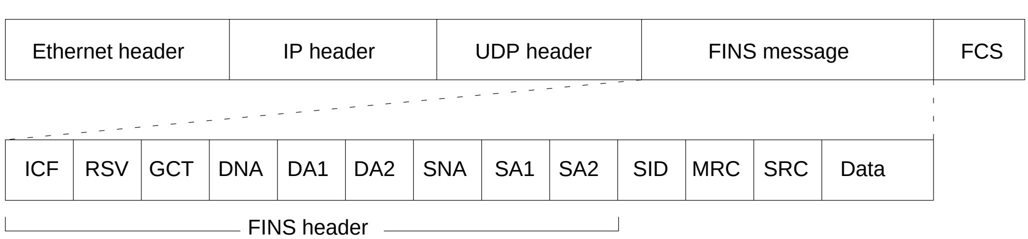
The basic flow of an FINS command (and the response to it) is shown in the following diagram.



Note The UDP/IP protocol does not provide communications control to ensure communications reliability. Consequently, the FINS communications services using the UDP/IP protocols cannot guarantee that any message arrived safely and unaltered at the destination. Methods to ensure reliability must be programmed into the user application, e.g., programming reties and appropriate processing of FINS responses.

Frame Format

Datagrams of the following format are sent and received during FINS communications.



3-2 Address Conversion

Nodes are specified by IP addresses for Ethernet communications, whereas the FINS node numbers are used for FINS communications. Therefore, the Ethernet Unit must be able to convert between IP addresses and FINS node numbers.

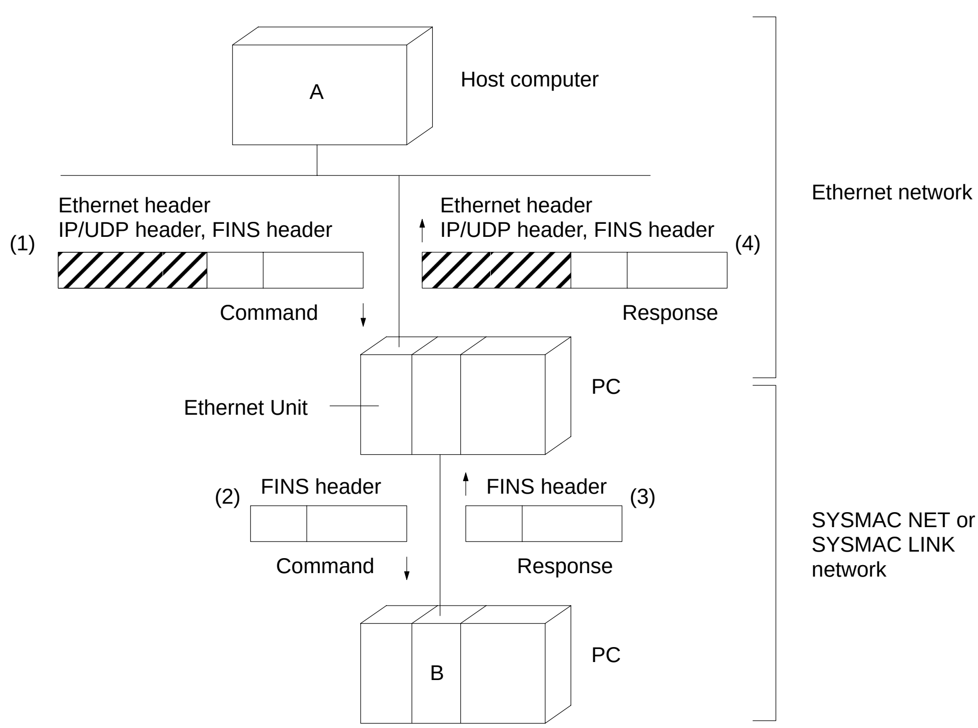
The following procedure is used to transfer datagrams when an FINS command is sent from node A on an Ethernet network to node B on another network, as shown in the diagram following the procedure.

Note Refer to *2-6 Ethernet Unit System Setup* for details about setting the conversion method.

1, 2, 3...

1. The Ethernet Unit on the middle PC receives the Ethernet message (datagram) sent from host A, saves the FINS message only, and disposes of the unneeded header information.
2. The FINS message is then delivered to the destination node, node B, by using routing tables.
3. The destination node returns a response for the FINS command it received. This response is delivered to the Ethernet Unit on the middle PC, again by using routing tables.
4. The FINS response received by the Ethernet Unit contains the FINS node number of the remote node (host computer) but no IP address, which is required to return the response over the Ethernet network.

   Without the IP address, the response cannot be delivered to the remote node, and that is where address conversion comes in. To achieve this, the Ethernet Unit uses one of the conversion methods described next to convert the addresses. Which of the conversion methods is used is set by the user in the CPU Bus Unit system setup.



3-2-1 Automatic Address Generation

Automatic address generation converts addresses using the FINS node number as the host number of the IP address. The IP address of the target node is thus calculated from a logical AND between local Ethernet Unit's IP address and subnet mask, to which is added the FINS node number of the target node:

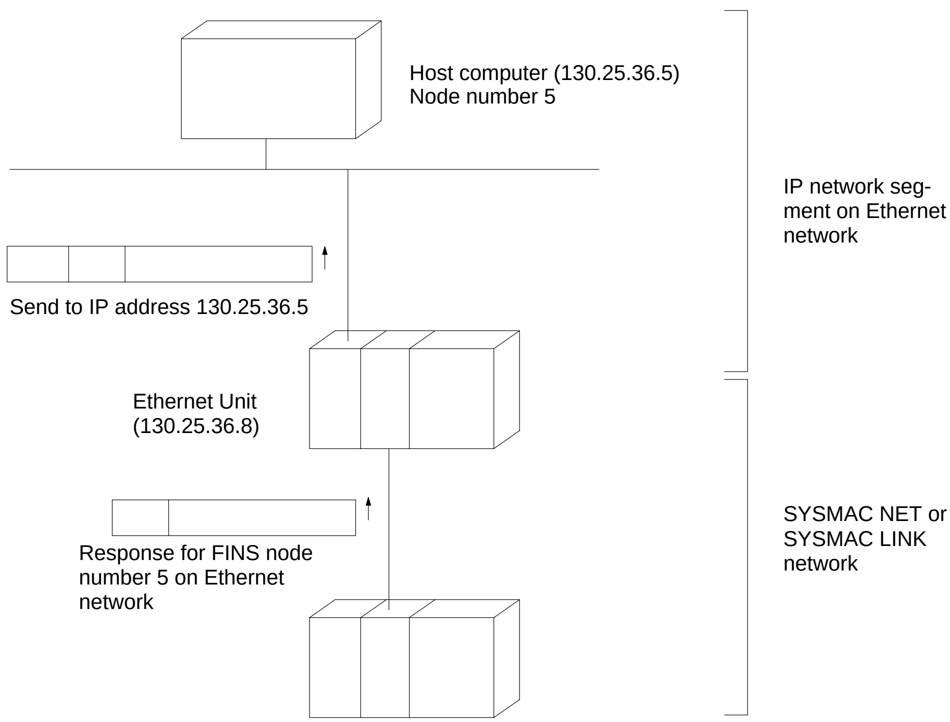
IP address of target node =
(local IP address AND subnet mask) + FINS node number of target node

Example

The IP address is calculated for the node using the following settings:

Local IP address:	130.25.36.8 (decimal)
Subnet mask:	FF.FF.FF.00 (hexadecimal)
FINS node number:	5

```
    130.25.36. 8          Local IP address
 AND FF.FF.FF.00          Subnet mask
-----------------------------------------------

    130.25.36.00
 +             05         FINS node number
-----------------------------------------------
    130.25.36.05          IP address
```



Note Before using the automatic generation method to convert IP addresses, the Ethernet network must be configured so that the host number of the IP address of each node in the network matches the FINS node number. The setting range of the FINS node numbers is from 1 to 126, so that the host numbers of the IP addresses of the Ethernet Units must also fall within this range.

- Example 1
  The IP address for FINS node number 75 on a network with network number 130.25.??.?? (subnet mask: FFFF0000) is 130.25.0.75.
- Example 2
  The IP address for FINS node number 75 on a network with network number 130.25.40.?? (subnet mask: FFFFFF00) is 130.25.40.75.

3-2-2 IP Address Table

An IP address table can be used to convert between FINS node numbers and IP addresses. The IP address table is set from a Programming Device, such as the CVSS. Refer to *2-6 Ethernet Unit System Setup* for details.

Example
We'll consider the IP address table for the following record in the IP address table.

FINS node number: 11
IP address: 150.31.2.83

For this setting, an FINS message for FINS node #11 will be sent to the node at IP address 150.31.2.83.

3-2-3 Combined Addresses Conversion

Both automatic generation and use of an IP address table can be specified together. In this case, the IP address table is accessed first and if the required node number is not contained in the table, the IP address is calculated using automatic generation.

3-3 FINS UDP Port Number

The default value for the UDP port number used by the FINS communications is 9600 (decimal). This UDP port number can be changed in the system setup.

A datagram received at this port is processed as an FINS communications message. Also, when sending data with the SEND(192)/RECV(193) instructions from the PC or when relaying an FINS message received from another network, the data is sent from this port to the destination port using the same port number. Therefore, set the UDP port number of all nodes (Ethernet Units) using FINS communications on the Ethernet network to the same FINS UDP port number. FINS communications is not possible between two nodes with different FINS UDP port numbers.

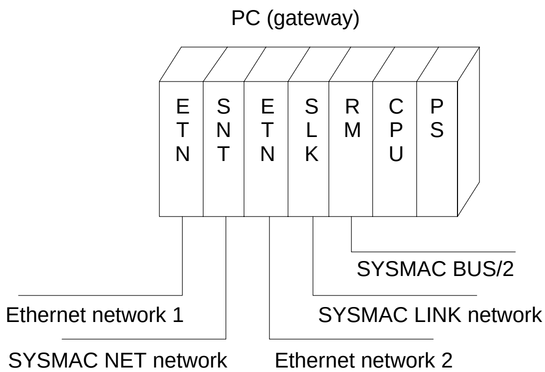
Note
1. FINS communications are not possible if an FINS message is received from a UDP port not specified for FINS communications.
2. The FINS UDP port number cannot be set to zero. If zero is set, the default value of 9600 will be used.

3-4 Internetwork Connections

Multiple networks can be interconnected via a PC or IP router to allow communications between nodes on remote networks.

3-4-1 PC Gateways

Mounting multiple Communications Units (which include the Ethernet Unit) on a PC permits FINS communications service to operate between nodes on an Ethernet networks and node on other types of FA networks. A PC that belongs to two networks and can thus function as an interface between them is called a PC gateway. Nodes on two different networks interconnected using a PC gateway can communicate only by using the FINS communications service.

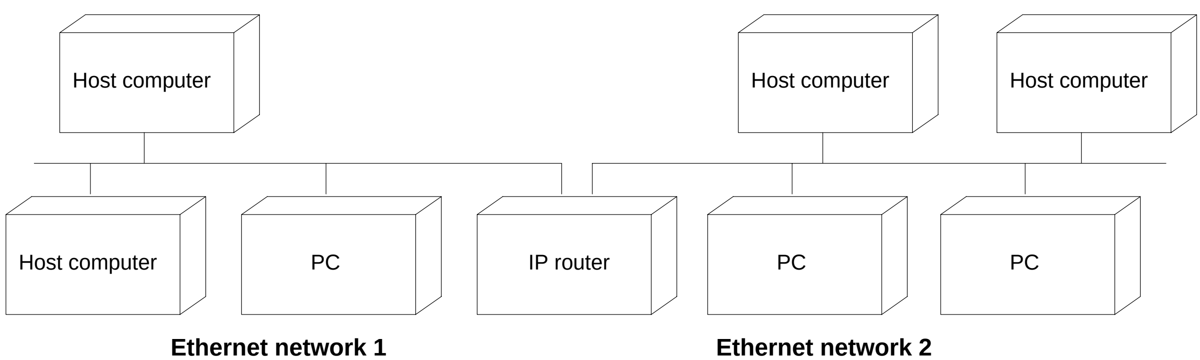


PS: Power Supply Unit
SLK: SYSMAC LINK Unit
SNT: SYSMAC NET Link Unit
ETN: Ethernet Unit
RM: SYSMAC BUS/2 Remote I/O Master Unit

Note
1. The PC gateway is provided using the FINS commands for the FINS communications service only. PC gateways cannot be used for the socket service.
2. A maximum data length of 2 Kbytes is possible for communications between Ethernet and SYSMAC NET networks, but the length is restricted to 552 bytes for communications between Ethernet and SYSMAC LINK or SYSMAC BUS/2 networks.
3. The maximum FINS message length that can be handled by an Ethernet Unit, including the FINS header, is 2,002 bytes.

3-4-2 Ethernet Connections Via IP Routers

Two Ethernet networks can be connected together via an IP router. Nodes on two network segments connected via an IP router can communicate with each other using all of the features provided by the Ethernet Unit. The FINS communications service is also possible between nodes on two networks connected via an IP router, and for the purpose of FINS communications, these two networks are actually considered as one network level and are set to the same FINS network address.



Note Set the same FINS network address for all Ethernet networks connected via IP routers (but, the network number and/or subnet number in the IP address will be different). For example, all PCs (Ethernet Units) in the diagram above must have the same FINS network address (i.e., same network level) to enable FINS communications on the same network level.

IP Router Table

To communicate between nodes on interconnected networks, IP router tables must be set in the Ethernet Units. Refer to 2-6 *Ethernet Unit System Setup* for details.

Broadcasting is not possible between two Ethernet networks connected by an IP router, such as the two shown in the above diagram.

3-5 Routing Tables

Routing tables must be set in the PC to use SEND(192), RECV(193), or CMND(194) or to otherwise exchange data between two networks. The routing tables defines the data transmission path between the local network and remote networks.

There are two kinds of routing tables, local network tables and relay network tables.

Local Network Tables

A local network table provides the network address and corresponding unit number for each Ethernet Unit, SYSMAC NET Link Unit, or SYSMAC LINK Unit that is mounted to the PC. You can set up to 16 local networks for a single PC. Table entries are not necessary for SYSMAC BUS/2 Remote I/O Systems.

Relay Network Tables

A relay network table provides the transmission path from the local network to the designated network. There can be up to 20 entries in a relay network table for a single PC. You need not set a relay network table if there will be no data exchanges with remote networks.

3-5-1 Setting Routing Tables

Routing tables can be set from a computer on the network or from a CV-series PC by sending a PARAMETER AREA WRITE command from a node that supports it. For details, refer to *FINS Command Reference Manual* for details on the PARAMETER AREA WRITE command.

The routing tables can also be set using CVSS. Refer to the *CVSS Operation Manual: Online* for details.

Note After you set the routing tables with PARAMETER AREA WRITE, be sure to restart the PC. You do not need to restart the PC if you have set the routing tables with CVSS because CVSS automatically resets the PC.

3-5-2 Local Network Tables

A local network table consists of a list of unit numbers and the corresponding network addresses for all local networks. If a single PC is mounted with more than one Ethernet Unit, SYSMAC NET Link Unit and/or SYSMAC LINK Unit, the local network table will need to list the corresponding network for each Unit.

Note The SYSMAC BUS/2 Remote I/O System does not have a network address and is not registered in local network tables.

Setting Data

Set the following data for each Unit.

Item	Range	Contents
Local network address	1 to 127	The address of the network to which the Ethernet Unit, SYSMAC NET Link, or SYSMAC LINK Unit belongs
CPU Bus Unit unit number	0 to 15	The unit number of the Ethernet Unit, SYSMAC NET Link Unit, or SYSMAC LINK Unit

The local network table setting display will appear on the CVSS as follows:

[Local Network Table]

#	Loc Netwk	SIOU unit #
1	001	00
2		
3		
4		
5		
6		
7		
8		

#	Loc Netwk	SIOU unit #
9		
10		
11		
12		
13		
14		
15		
16		

3-5-3 Relay Network Tables

To exchange data between two networks, you must set a relay network table that specifies the data transmission path for remote networks. You can register up to 20 destination networks.

Setting Data

Data transmission paths include the following.

Item	Range	Contents
Destination network	1 to 127	The network address of the designated node
Relay network	1 to 127	The address of the network through which the destination network can be reached
Relay node	Ethernet Units or SYSMAC NET Link Units: 1 to 126 SYSMAC LINK Units: 1 to 62	The address of the node on the relay network through which the destination network can be reached

The relay network table setting display will appear on the CVSS as shown below.

[Relay Network Table]

#	End Netwk	Relay PC ID	Relay Netwk	Relay node
1	002		001	001
2				
3				
4				
5				
6				
7				
8				
9				
10				

#	End Netwk	Relay PC ID	Relay Netwk	Relay node
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Each node can be assigned a unique PC name (PC ID). If a node has a PC name (the name of the PC on the node), you need not designate the network address or node number. For details, refer to the *CVSS Operation Manual: Online*.

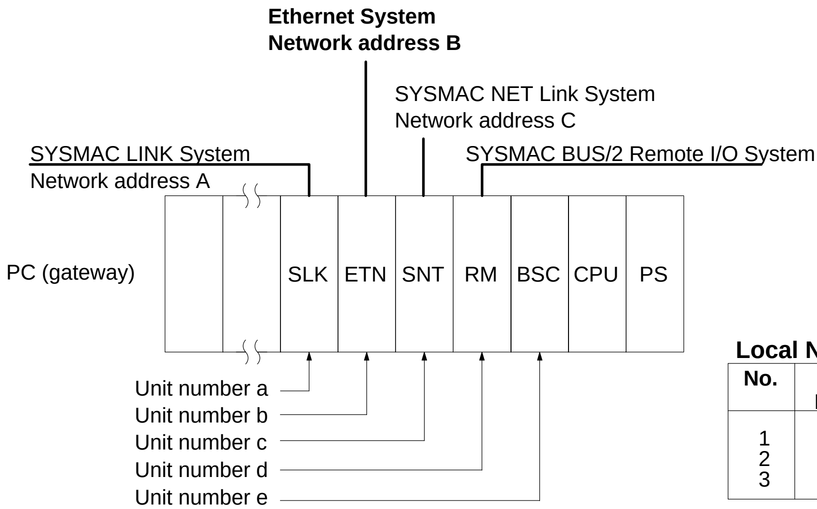
Note

1. You must set a relay network table that will not conflict with the local network table or a routing table error will result.
2. Each network in the entire interconnected system must have a unique address.

3-5-4 Routing Table Examples

Local Network Table with More Than One Unit

The following illustration shows an example of routing table settings for a single PC mounted with more than one Unit. The PC shown here belongs to three networks that require routing tables, and there are thus two entires in the local network table. The SYSMAC BUS/2 System does not have a network address and is not registered on the local network table. Here, letters are used to indicate network addresses and unit numbers in stead of the normal values.



PS: Power Supply Unit
CPU: CPU
ETN; Ethernet Unit
SNT: SYSMAC NET Link Unit
SLK: SYSMAC LINK Unit
RM: SYSMAC BUS/2 Remote I/O Master
BSC: Basic Unit

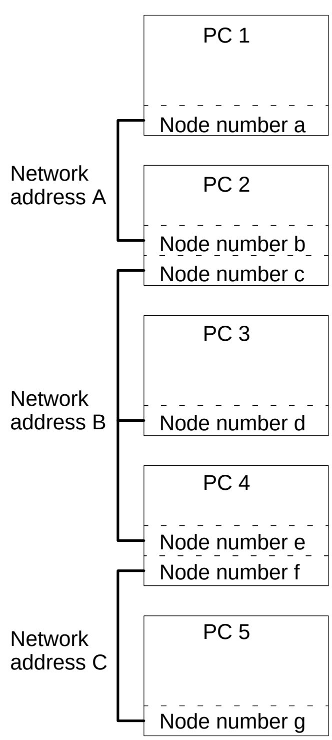
Local Network Table

No.	Loc Netwk	SIOU unit #
1	A	(a)
2	B	(b)
3	C	(c)

Three Connected Networks

The following shows an example of routing table setting with three networks connected to one another. Take a look at the relay network table for PC 3. When network A is the destination network for PC 3, B is the relay network and c is the relay node. When network C is the destination network, B is the relay network and e is the relay node.

Relay Network Tables



No.	End Netwk	Relay		
		PC ID	Netwk	node
1	B		A	b
2	C		A	b

No.	End Netwk	Relay		
		PC ID	Netwk	node
1	C		B	e

No.	End Netwk	Relay		
		PC ID	Netwk	node
1	A		B	c
2	C		B	e

No.	End Netwk	Relay		
		PC ID	Netwk	node
1	A		B	c

No.	End Netwk	Relay		
		PC ID	Netwk	node
1	A		C	f
2	B		C	f

3-6 Unit Addresses

Unit addresses are used to designate the actual Units that are to participate in FINS communications. Unit addresses are required because communications are possible with more than one Unit at each node.

When a network node is occupied by a PC, communications are possible with the PC's CPU or with a CPU Bus Unit mounted to the PC. The unit addresses used to distinguish between these are given in the following table.

Destination	Unit address
PC's CPU	00
CPU Bus Units (see note 1)	10 + unit number or FE (see note 2)
SYSMAC BUS/2 Group-2 Slave (commands cannot be sent to Group-1 Slaves)	\$90 to \$CF: Unit number+90+10×Master address

Note
1. Communications are possible with the following CPU Bus Units: Ethernet Units, SYSMAC NET Link Units, SYSMAC LINK Units, SYSMAC BUS/2 Units, Personal Computer Units, and BASIC Units.
2. FE is used for Ethernet, SYSMAC NET Link, and SYSMAC LINK Units to specify the Unit actually connecting the destination node to the network and can be used to communicate with such Units without having to worry about unit addresses.
3. Ranges are as follows:

   Unit numbers: 0 to F (0 to 15)
   Master addresses: 0 to 3 (0 to 3)

When a network node is occupied by an IBM PC/AT or compatible connected via a SYSMAC LINK Support Board, the unit addresses given in the following table are used.

Destination	Unit address
User application on the computer	01
SYSMAC LINK Support Board	10 or FE (see note 1)

Note
1. The unit number of the SYSMAC LINK Support Board is fixed at 0, i.e., "10" is 10 plus the unit number. FE is used to specify the Unit actually connecting the destination node to the network, i.e., the SYSMAC LINK Support Board, without regard to the unit number.
2. User applications must be written to return responses if desired or commands must be set without requests for responses.

There are no specific restrictions to unit addresses for FINS communications with host computers. Be sure to use unique unit addresses on each network.

3-7 PC-originating Commands

FINS commands can be sent from the user's ladder-diagram program in the PC by using the SEND(192), RECV(193), and CMND(194) instructions.

3-7-1 Communications Specifications

The PC's communications specifications are given in the following table.

Item	Specifications
Destination	1:1 SEND(192), RECV(193), CMND(194) instructions 1:N SEND(192), CMND(194) instructions (broadcasting) There will be no response from the nodes for broadcasting. N is 126 max. in SYSMAC NET Link Systems; 62 max. in SYSMAC LINK Systems.
Data length	SEND(192): 990 words (1,980 bytes) max. (broadcasting: 727 words) RECV(193): 990 words (1,980 bytes) max. CMND(194): 1,990 bytes max. after command code (broadcasting: 1,462 bytes)
Contents of transmission	SEND(192): Instruction requesting the destination node to receive data RECV(193): Instruction requesting the destination node to send data CMND(194): Any available FINS commands
Communications port number	Ports 0 to 7 (8 transmissions can occur simultaneously)
Response monitor time	0000: 2 s (default) 0001 to FFFF: 0.1 to 6,553.5 s in 0.1-s increments (specified by user)
Number of retries	0 to 15 times

Note The maximum data length is limited to 512 bytes for data exchange between the PC and SYSMAC LINK Systems or the PC and SYSMAC BUS/2 Remote I/O Systems.

3-7-2 Data Areas

The available data areas and words for data exchange using the SEND(192) and RECV(193) instructions vary with the CV-series PC model as follows:

Data area	CV1000/CV2000/ CVM1-CPU11-E	CV500/CVM1-CPU01-E
CIO Area	0000 to 2555	
Auxiliary Area	A000 to A511	
CPU Bus Link Area	G000 to G255	
DM Area	D00000 to D24575	D00000 to D08191
EM Area	E00000 to E32765 (8 banks)	---
Timer Area	T0000 to T1023	T0000 to T0511
Counter Area	C0000 to C1023	C0000 to C0511

Note
1. Areas other than those listed above must not be used.
2. G000 to G007 and A256 to A511 are read-only.
3. The EM Area is available only when you mount the Expansion Memory Unit to the PC. Up to 8 banks (32 K words per bank) can be used (the number of available banks varies with the model of the Expansion Memory Unit). For details, refer to the *CV-series PC Operation Manual: Ladder Diagrams*.

3-7-3 Communications Ports

When you have executed the SEND(192), RECV(193), or CMND(194) instruction, the status information of the instruction for each communications port is held in the Auxiliary Area of the PC. Using this status information, the PC can adjust the execution timing of the SEND(192), RECV(193), and CMND(194) instructions while the user's program is running.

Timing

You can use eight communications ports simultaneously to execute the SEND(192), RECV(193), and/or CMND(194) instructions. You cannot, however, execute more than one instruction per port at the same time. If you want to execute more than one instruction in sequence at the same port, use network communications Enabled Flags to be sure that only one instruction is being executed for each port at any one time. Network communications Enabled Flags are contained in the status data.

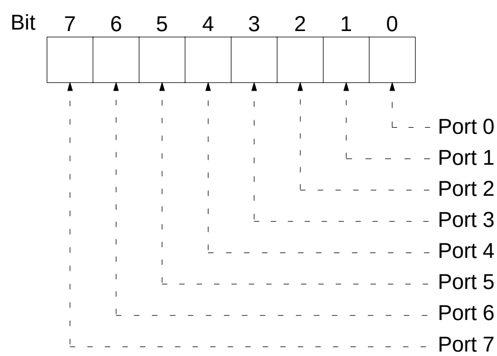
Status Data

The following table shows the configuration of the status data in the Auxiliary Area.

Word	Bit 15 to 8	Bit 7 to 0
A502	Execute Error Flags	Enabled Flags
A503	Completion code for port 0	
A504	Completion code for port 1	
A505	Completion code for port 2	
A506	Completion code for port 3	
A507	Completion code for port 4	
A508	Completion code for port 5	
A509	Completion code for port 6	
A510	Completion code for port 7	

Enabled Flags (A502)

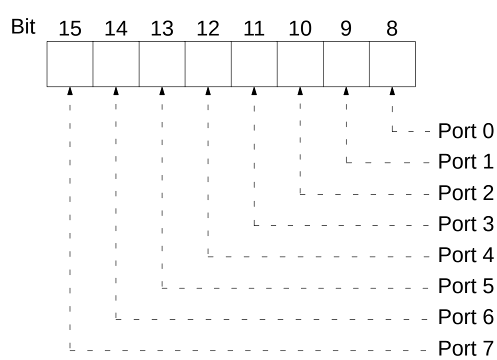
Each bit in word A502 corresponds to a port, as shown below. A port is available for instruction execution if its Enabled Flag is ON (1).



0 (OFF): Executing
1 (ON): Not executing

Execute Error Flags (A502)

Each bit in word A502 corresponds to a port as shown below. The ON/OFF condition of the bits will be retained until execution of the next instruction.



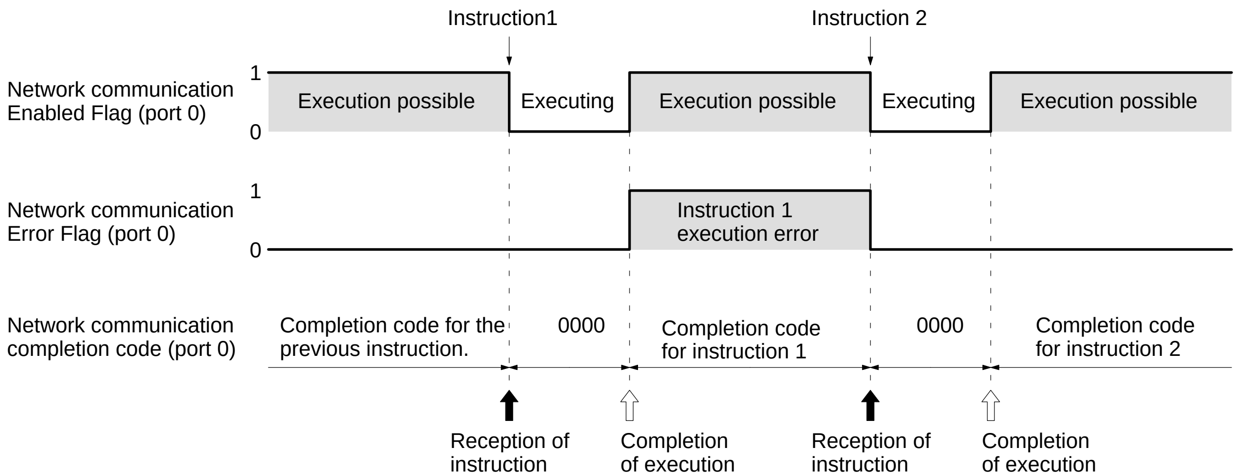
0 (OFF): Executed without error
1 (ON): Error occurs

Completion Codes (A503 to A510)

The status after execution of an instruction will be shown by a completion code. The completion code will be retained until the PC executes the next instruction for that port. The completion code is 0000 while an instruction is being executed. There is no difference in content between completion codes and response codes. For details, refer to *Section* 9 *Troubleshooting.*

Flag Timing

The following timing chart shows the timing of the Enabled Flag, Execute Error Flag, and Completion Code. In this example, two instructions are executed at port 0 and there was an error when instruction 1 was being executed.



Caution Be sure to confirm that the Execution Enabled Flag has turned back ON following communications before changing the contents of data that is being sent. It will not be possible to determine what data was sent if the send data is changed during a transmission.

Execute Error Flag and Completion Codes with CMND(194)

The following will cause the Execute Error Flag to turn ON and/or will be recorded as the completion code when the CMND(194) instruction is executed.

1, 2, 3...

1. A response time-out error.
2. The number of transmission data bytes exceeding the maximum permissible range.
3. The number of response data bytes exceeding the number of reception data bytes requested (in this case, no response will be recorded).

Errors other than the above will be output as the response code and will not be recorded as completion codes.

3-8 Using SEND(192)/RECV(193)/CMND(194)

3-8-1 NETWORK SEND Instruction – SEND(192)

The SEND(192) instruction allows a PC to write data to the memory of a device located at a node on a local or remote network.

Format

The format of the SEND(192) instruction is as follows:

Ladder Symbol

```
     (192)
───[ SEND     S        D        C ]
```

Variations

↑ SEND(192)

Operand Data Areas

S: 1st source word	CIO, G, A, T, C, DM, DR*, IR*
D: 1st destination word	CIO, G, A, T, C, DM, DR*, IR*
C: 1st control word	CIO, G, A, T, C, DM, DR*, IR*

*Indirect addressing only

Control Data

Set the destination node number to $FF to broadcast the data to all nodes in the designated network or to $00 to send to a destination within the node of the PC executing the send.

Word	Bits 00 to 07	Bits 08 to 15
C	Number of words (1 to 990 in 4-digit hexadecimal, i.e., $0001 to $03DE)	
C+1	Destination network address (1 to 127, i.e., $01 to $7F)	Bits 08 to 15: Set to 0.
C+2	Destination unit address	Destination node number
C+3	Bits 00 to 03: No. of retries (0 to 15 in hexadecimal, i.e., $0 to $F) Bits 04 to 07: Set to 0.	Bits 08 to 10: Communications port number ($0 to $7) Bit 11 to 14: Set to 0. Bit 15: Response Request Bit ON: Don't request response OFF: Request response
C+4	Response monitor time ($0001 to $FFFF = 0.1 to 6553.5 seconds)	

Note Transmissions cannot be sent to the PC executing the send.

1, 2, 3...

1. Number of Words
   Set the total number of words to be transferred.
2. Destination Network Address
   Set the destination network address. Do not set 00 to designate the local network; this is not possible with an Ethernet Unit.
3. Destination Node Number
   Set the destination node number to FF for broadcasting within the local network. Set to 00 for transmission to a device at the local PC.
4. Destination Unit Address
   Set the destination unit address to 00 if the destination is a PC. If the destination is a user application on an IBM PC/AT-compatible computer, the unit address will depend on the board. Refer to documentation for the board being addressed. For a CPU Bus Unit, add 10 to the unit number (0 to F) so that the destination unit address is from 10 to 1F.

Destination	Address (hexadecimal)
PC (CPU)	00
NSB, NSU	See documentation for NSB or NSU.
CPU Bus Unit	Unit no. + 10 (10 to 1F for Unit 0 to F (0 to 15))

5. Response Request Bit (Bit 15 of C+3)
   Usually set this bit to OFF (0: Request response). If you do not need any response, set the bit to ON (1: Don't request response).
6. Communications Port No.
   Set the communications port number that the SEND(192) instruction will be sent from.
7. Number of Retries
   Set the number of retries for unsuccessful transmissions. You can set the number between 1 to 15.
8. Response Monitor Time (Unit: 0.1 S)
   Designates the length of time that the PC retries transmission when bit 15 of C+3 is OFF and no response is received. The default value is $0000, which indicates 2 seconds. The response function is not used when the destination node number is set to $FF, broadcasting to all nodes in the network.

Range of Control Data The permissible ranges of control data to be set are as follows:

Item	Value
Number of words (see note 1.)	0001 to 03DE (1 to 990 words)
Destination network address	01 to 7F: Destination network address
Destination node number	00: Transmission within local PC 01 to 7E: Destination node number (1 to 126) (Ethernet or SYSMAC NET) 01 to 3E: Destination node number (1 to 62) (SYSMAC LINK) FF: Broadcasting
Destination unit address (see note 2.)	00: PC 10 to 1F: CPU Bus Unit (unit numbers 0 to F, respectively)
Response request bit	0 (OFF): Request response 1 (ON): Don't request response
Communications port number	0 to 7 (0 to 7)
Number of retries	0 to F (0 to 15)
Response monitor time	0000: 2 s (default) 0001 to FFFF: 0.1 to 6,553.5 s with 0.1-s increments (set by the user)

Note 1. A maximum of 727 words can be sent for broadcasts.
2. Refer to documentation for the board being addressed if the destination is a user application on an IBM PC/AT-compatible computer.

3-8-2 NETWORK RECEIVE Instruction – RECV(193)

The RECV(193) instruction enables the local node to write to its memory data from a node either on a local or remote network.

Format The format of the RECV(193) instruction is as follows:

Ladder Symbol

```
       (193)
 ——[ RECV    S     D     C ]
```

Variations

↑ RECV(193)

Operand Data Areas

S: 1st source word CIO, G, A, T, C, DM, DR*, IR*

D: 1st destination word CIO, G, A, T, C, DM, DR*, IR*

C: 1st control word CIO, G, A, T, C, DM, DR*, IR*

*Indirect addressing only

Control Data Set the source node number to $00 to send data within the PC executing the instruction.

Note The source is the node that sends data as a result of executing RECV(193), i.e., the node to which RECV(193) is being sent.

Word	Bits 00 to 07	Bits 08 to 15
C	Number of words (1 to 990 in 4-digit hexadecimal, i.e., $0001 to $03DE)	
C+1	Source network address (1 to 127, i.e., $01 to $7F)	Bits 08 to 15: Set to 0.
C+2	Source unit address	Source node number
C+3	Bits 00 to 03: No. of retries (0 to 15 in hexadecimal, i.e., $0 to $F) Bits 04 to 07: Set to 0.	Bits 08 to 10: Communications port number ($0 to $7) Bit 11 to 15: Set to 0.
C+4	Response monitor time ($0001 to $FFFF = 0.1 to 6553.5 seconds)	

1, 2, 3...

1. Number of Words
   Set the total number of words to be transferred.
2. Source Network Address
   Set the source network address. Do not set 00 to designate the local network; this is not possible with an Ethernet Unit.

3. Source Node Number
   Set the source node number to 00 for transmission to devices at the local PC.

4. Source Unit Address
   Set the source unit address to 00 if the source is a PC. If it is a user application on the IBM PC/AT or compatible computer, the unit address will depend on the board. Refer to documentation for the board being addressed. For a CPU Bus Unit, add 10 to the unit number (0 to F) so that the destination unit address is from 10 to 1F.

Destination	Address (hexadecimal)
PC (CPU)	00
NSB, NSU	See documentation for NSB or NSU.
CPU Bus Unit	Unit no. + 10 (10 to 1F for Unit 0 to F (0 to 15))

5. Communications Port Number
   Set the communications port number that the RECV(193) instruction will be sent to.

6. Number of Retries
   Set the number of retries for unsuccessful transmissions. You can set the number between 1 to 15.

7. Response Monitor Time
   Designates the length of time that the PC retries transmission when bit 15 of C+3 is OFF and no response is received. The default value is $0000, which indicates 2 seconds.

Range of Control Data

The permissible ranges of control data to be set are as follows:

Note For RECV(193), the source is the node that will be sending data in response to the command, i.e., the node that is the source of the data being transferred as a result of executing RECV(193). The source is thus the node to which RECV(193) is being sent.

Item	Value
Number of words	0001 to 03DE (1 to 990 words)
Source network address	01 to 7F: Address of the network from which data is being requested (source)
Source node number	00: Transmission within local PC 01 to 7E: Source node number (1 to 126) (Ethernet or SYSMAC NET) 01 to 3E: Source node number (1 to 62) (SYSMAC LINK) FF: Broadcasting
Source unit address (see note)	00: PC 10 to 1F: CPU Bus Unit (for unit numbers 0 to F, respectively)
Communications port number	0 to 7 (0 to 7)
Number of retries	0 to F (0 to 15)
Response monitor time	0000: 2 s (default) 0001 to FFFF: 0.1 to 6,553.5 s in 0.1-s increments (set by the user)

Note Refer to documentation for the board being addressed if the destination is a user application on an IBM PC/AT-compatible computer.

3-8-3 DELIVER COMMAND Instruction – CMND(194)

The CMND(194) instruction can be used to send any available FINS command to a desired node.

Format

The format of the CMND(194) instruction is as follows:

Ladder Symbol

(194)
[CMND S D C]

Variations

↑ CMND(194)

Operand Data Areas

S: 1st command word	CIO, G, A, T, C, DM, DR*, IR*
D: 1st response word	CIO, G, A, T, C, DM, DR*, IR*
C: 1st control word	CIO, G, A, T, C, DM, DR*, IR*

*Indirect addressing only

The CMND(194) instruction transmits a command beginning at word S to the designated Unit at the destination node number in the designated network and receives the response beginning at word D.

Control Data

The control words, beginning with C, specify the number of bytes of control data to be sent, the number of bytes of response data to be received, the destination node, and other parameters.

Word	Bits 00 to 07	Bits 08 to 15
C	Number of bytes to send (2 to 1,990, i.e., $0002 to $07C6)	
C+1	Number of bytes to receive (2 to 1,990, i.e., $0002 to $07C6)	
C+2	Destination network address (1 to 127, i.e., $01 to $7F)	Bits 08 to 15: Set to 0.
C+3	Destination unit address	Destination node number
C+4	Bits 00 to 03: No. of retries (0 to 15 in hexadecimal, i.e., $0 to $F) Bits 04 to 07: Set to 0.	Bits 08 to 10: Communications port number ($0 to $7) Bit 11 to 14: Set to 0. Bit 15: Response Request Bit ON: Don’t request response OFF: Request response
C+5	Response monitoring time ($0001 to $FFFF = 0.1 to 6553.5 seconds)	

1, 2, 3...

1. Number of Bytes to Send
   Set the total number of bytes of command data (including the command code) stored at the first command and following words of the command block.
2. Number of Bytes to Receive
   Set the total number of bytes of response data (including the response code) to be stored at the first response and following words of the response block.

   Note Maximum number of bytes that can be sent or received is as follows:

System	Max. number of bytes
Ethernet or SYSMAC NET Link	$07C6 (1,990)
SYSMAC LINK	$021E (542)
SYSMAC BUS/2	$021E (542)

3. Destination Network Address
   Set the destination network address. Do not set 00 to designate the local network; this is not possible with an Ethernet Unit.
4. Destination Node Number
   Set the destination node number to FF for broadcasting within the local net-

work. Set to 00 for transmission to a device at the local PC. The destination node number can have the following values:

System/type of transmission	Possible values
Ethernet or SYSMAC NET Link System	$01 to $7E (nodes 1 to 126)
SYSMAC LINK System	$01 to $3E (nodes 1 to 62)
Broadcast to all nodes in network	$FF
Transmit within the PC	$00

5. Destination Unit Address
   Set the destination unit address as follows:

Destination	Designation (hexadecimal)
PC (CPU)	00
NSB, NSU	See documentation for NSB or NSU.
SYSMAC NET Link Unit	FE
CPU Bus Unit	Unit no. + 10 (10 to 1F for Unit 0 to F (0 to 15))

6. Response Request Bit
   Usually set this bit to OFF (0: Request response). If no response is required, set the bit to ON (1: Don't request response).
7. Communications Port Number
   Set the communications port number that the CMND(194) instruction will be sent to.
8. Number of Retries
   Set the number of retries for unsuccessful transmissions. You can set the number between 1 to 15.
9. Response Monitor Time (Unit: 0.1 S)
   Designates the length of time that the PC retries transmission when bit 15 of C+3 is OFF and no response is received. The default value is $0000, which indicates 2 seconds.

Note If more than the *number of bytes to receive* is received, no response will be stored. If fewer bytes are received, the response data will be stored and the unused memory area of the PC will remain unchanged.

Range of Control Data

The permissible ranges of control data are as follows:

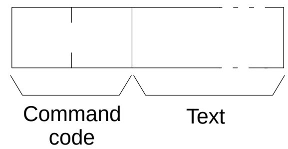
Item	Value
Number of bytes to send (see note 1.)	0002 to 07C6 (2 to 1,990 bytes) (after FINS command code)
Number of bytes to receive	0002 to 07C6 (2 to 1,990 bytes)
Destination network address	01 to 7F: Destination network address
Destination node number	00: Transmission within local PC 01 to 7E: Destination node number (1 to 126) (SYSMAC NET) 01 to 3E: Destination node number (1 to 62) (SYSMAC LINK) FF: Broadcasting
Destination unit address (see note 2.)	00: PC 10 to 1F: CPU Bus Unit (unit number: 0 to F, respectively)
Response request bit (see note 3.)	0 (OFF): Request response 1 (ON): Don't request response
Communications port number	0 to 7 (0 to 7)
Number of retries	0 to F (0 to 15)
Response monitor time	0000: 2 s (default) 0001 to FFFF: 0.1 to 6,553.5 s with 0.1-s increments (set by the user)

Note
1. A maximum of 1,462 bytes can be sent for broadcasts.
2. Refer to documentation for the board being addressed if the destination is a user application on an IBM PC/AT-compatible computer.

3. In the case of broadcasting, set the Response request bit to ON (1: Don't request response).

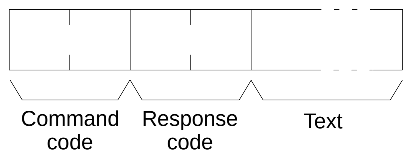
Command Format

The command stored beginning at word S (specified in instruction) must be in the following format. The text can be up to 1,988 bytes long.



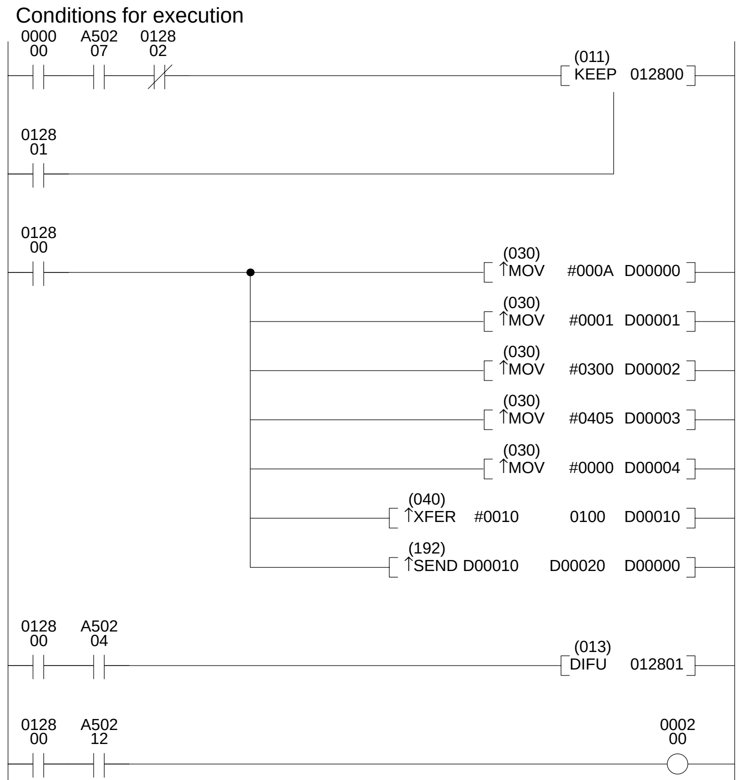
Response Format

The response returned for the command and stored beginning at word D (specified in instruction) will be in the following format. The text can be up to 1,986 bytes long.



3-8-4 Program Example

The following program example shows use of the SEND(192) and RECV(193) instructions. Remember, you must be sure that only one communications process is being executed simultaneously for each port.



The SEND(192) program section will run when 000000 is ON, provided that the Enabled Flag for port 4 is ON and the RECV(193) instruction is not being executed.

Bit 012800 is ON while the SEND(192) instruction is being executed, and it goes OFF when the execution of the instruction has been completed.

The control data is set in consecutive words starting from D00000 as shown in the following table.

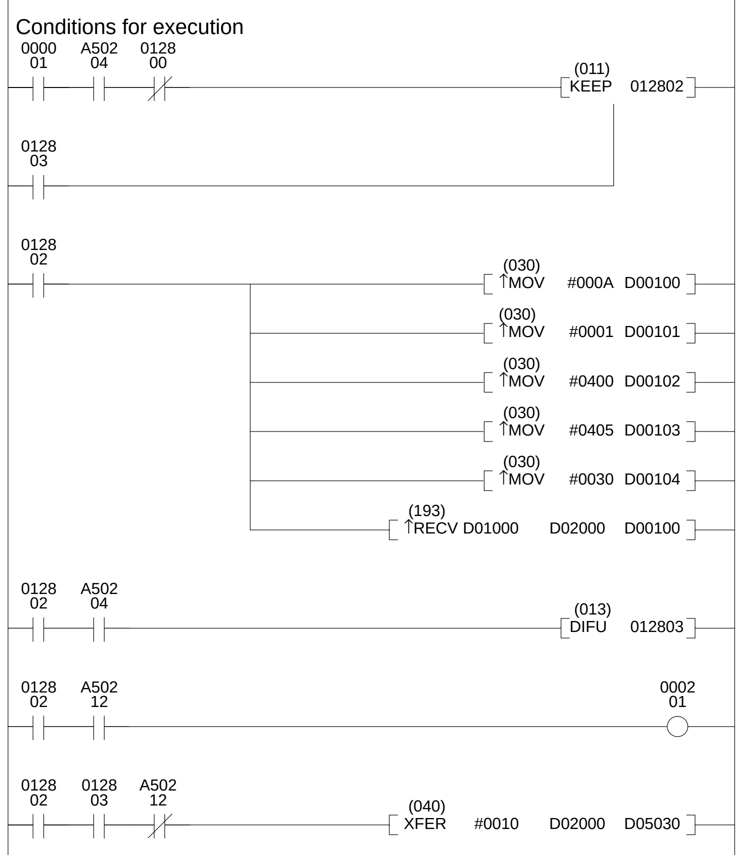
Word	Content		Meaning
D00000	00	0A	Number of words: 10
D00001	00	01	Destination network address: 1
D00002	03	00	Destination node number: 3 Destination unit address: 0 (PC)
D00003	04	05	Communications port no.: 4 Number of retries: 5
D00004	00	00	Response monitor time: 2s (default)

The 10 words of data starting from 0100 are transferred to words starting at D00010.

The 10 words of data starting from D00010 on the local PC are sent to D00020 and following words on the PC with a unit address of 0, node number of 3, and network address of 1 (as specified in control data).

Sending is finished when the Port Enable Flag for port #4 (A50204) comes ON to turn ON bit 012801.

Bit 000200 is turned ON to indicate an error if the Port Execution Error Flag for port #4 (A50212) turns ON.



The RECV(193) program will run when 000001 is ON, provided that the Enabled Flag for port 4 is ON and the SEND(192) instruction is not being executed.

Bit 012802 is ON while the RECV(193) instruction is being executed, and it goes OFF when the execution of the instruction has been completed.

The control data is set in consecutive words starting from D00100 as shown in the following table.

Word	Content		Meaning
D00100	00	0A	Number of words: 10
D00101	00	01	Destination network address: 1
D00102	04	00	Destination node number: 04 Destination Unit address: 0
D00103	04	05	Communications port no.: 4 Number of retries: 5
D00104	00	30	Response monitor time: 4.8 s

The 10 words of data starting from D01000 on the PC with a unit address of 0, node number of 04, and a network address of 1 are stored at D02000 and following words in the local PC.

Bit 000201 is turned ON to indicate an error if the Port Execution Error Flag for port #4 (A50212) turns ON.

When the data is received without an error, the 10 words of data received at D02000 and following words are stored in D05030 and following words on the same PC.

Note To execute more than one instruction in sequence at the same port, you must use the Enabled Flags to be sure that only one instruction is being executed at the same time for any one port (refer to page 45).

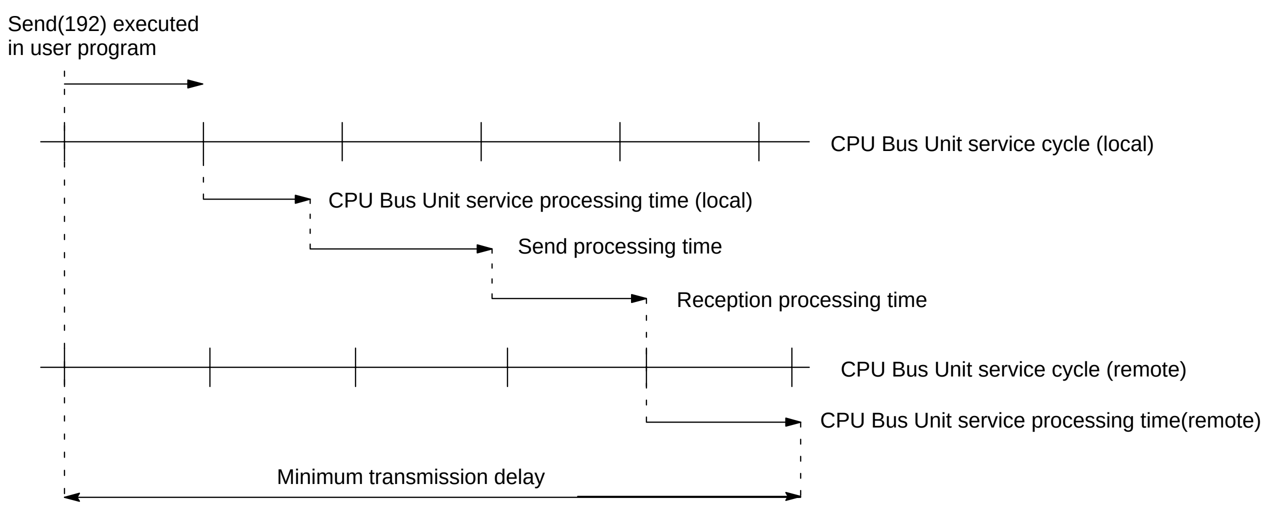
3-8-5 Transmission Delays

The methods of computing the minimum time required from execution of the SEND(192), RECV(193), and CMND(194) instructions until data arrives are described in this section. These are, however, the minimum transmission delays and they can be greatly increased depending on the conditions under which the instructions are executed.

SEND(192)

The transmission delay for the SEND(192) instruction can be calculated using the following equation, which is illustrated in the following diagram.

Min. delay = local node service cycle + local node service processing time + send processing time + reception processing time + remote node service processing time



CPU Bus Unit Service Cycle

The CPU Bus Unit service cycle is either one cycle for synchronous processing or the peripheral device processing time for asynchronous processing. Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for details.

CPU Bus Unit Service Processing Time

This is the time required to process CPU Bus Units and is approximately 1 ms for Ethernet Units.

Send and Reception Processing Times

The send and reception processing times are each equivalent to 0.047 ms/word times the number of words transferred, plus 10 ms. For synchronous PC execution processing, the reception processing time can be much larger than calculated with this equation. When the remote node is set for synchronous execution processing, use the first integral multiple of the minimum cycle time of the PC that is larger than 0.047 ms/word x number of transfer words + 10 ms.

Note The operating environment can cause transmission delays larger than those calculated with the methods given here. Among the causes of longer delays are the following: traffic on the network, window sizes at network nodes, traffic through the Ethernet Unit (e.g., socket servicing, FTP server communications, etc.), and the system configuration.

Example Calculations

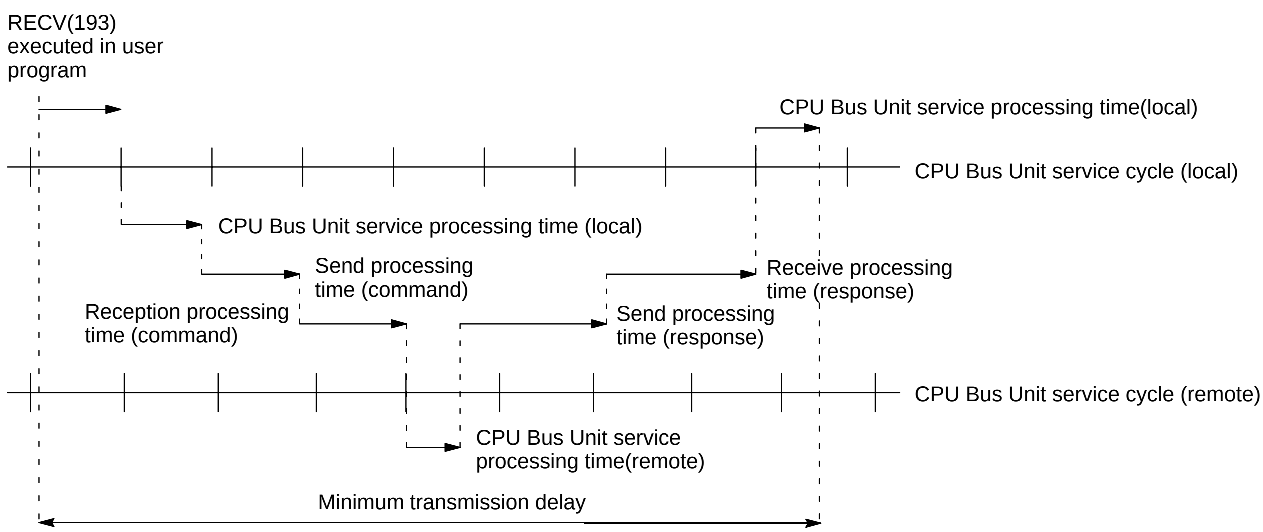
The following example shows calculations for sending 256 words between two PC nodes using SEND(192). Both nodes are set for synchronous processing, the local node's CPU cycle time is 40 ms, and the remote node's CPU cycle time is 15 ms. Calculations are shown in the following table.

Item	Calculation
Local node service cycle	PC cycle time = 40 ms
Local node service processing time	1 ms
Send processing time	256 x 0.047 + 10 ≅ 22 ms
Reception processing time	256 x 0.047 + 10 ≅ 22 ms (Use 30 ms to compensate for synchronous processing)
Remote node service processing time	1 ms
Total	40 + 1 + 22 + 30 + 1 = **94 ms**

RECV(193)

The transmission delay for the RECV(193) instruction can be calculated using the following equation, which is illustrated in the following diagram.

Min. delay = local node service cycle + local node service processing time + send processing time (command) + reception processing time (command) + remote node service processing time + send processing time (response) + reception processing time (response) + local node service processing time



CPU Bus Unit Service Cycle

The CPU Bus Unit service cycle is either one cycle for synchronous processing or the peripheral device processing time for asynchronous processing Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for details.

CPU Bus Unit Service Processing Time

This is the time required to process CPU Bus Units and is approximately 1 ms for Ethernet Units.

Send and Reception Processing Times

The send and reception processing times are each equivalent to 10 ms for commands and to 0.047 ms/word times the number of words transferred, plus 10 ms for responses. For synchronous PC execution processing, the processing times can be much longer than calculated with this equation. When the PC is set for synchronous execution processing, use the first integral multiple of the minimum cycle time of the PC that is larger than 0.047 ms/word x number of transfer words + 10 ms.

Note The operating environment can cause transmission delays larger than those calculated with the methods given here. Among the causes of longer delays are the following: traffic on the network, window sizes at network nodes, traffic through the Ethernet Unit (e.g., socket servicing, FTP server communications, etc.), and the system configuration.

Example Calculations

The following example shows calculations for receiving 256 words between two PC nodes using RECV(193). Both nodes are set for synchronous processing, the local node’s CPU cycle time is 40 ms, and the remote node’s CPU cycle time is 50 ms. Calculations are shown in the following table.

Item	Calculation
Local node service cycle	PC cycle time = 40 ms
Local node service processing time	1 ms
Send processing time (command)	10 ms
Reception processing time (command) + Remote node service processing time + Send processing time (response)	10 ms +1 ms +256 x 0.047 + 10 = 33 ms (Use 50 ms to compensate for synchronous processing)
Reception processing time (response) + Remote node service processing time	256 x 0.047 + 10 +1 = 23 ms (Use 40 ms to compensate for synchronous processing)
Total	40 + 1 + 10 + 50 + 40 = **141 ms**

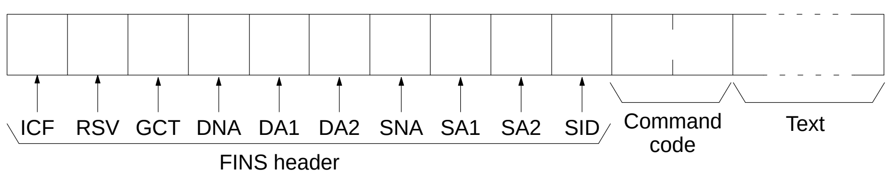
3-9 Host Computer-originating Commands

Commands and responses sent from host computers must be in the formats described in this section and must provide the proper FINS header information. These formats can also be used to decode commands and responses received from other network nodes.

3-9-1 Command/Response Formats

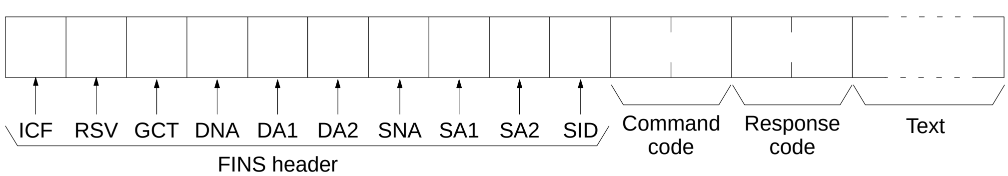
Commands

Commands consist of an FINS header, command code, and text. The length and content of the text depends on the particular command.



Responses

A response block includes a response code (a 2-byte code in binary) added to the command format. In the FINS header information, DNA DA1, and DA2 are interchanged with SNA, SA1, and SA2 (in comparison to the command header) and SID is the same as that in the command header.



Text Length

The maximum length of the text is as follows:

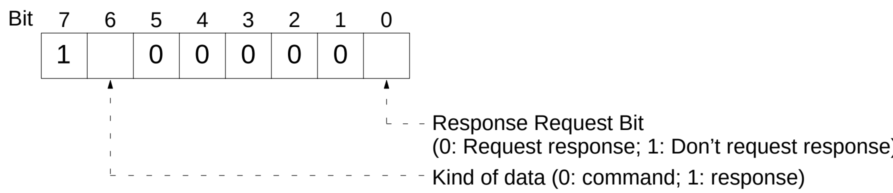
Commands: 1,988 bytes
Responses: 1,986 bytes

Text length, however, must meet the specifications of the remote node if internetwork communications are conducted with other types of networks, such as SYS-

MAC NET or SYSMAC LINK network. Refer to the system manual for each network for details.

FINS Header Information

ICF (Information Control Field)
The configuration of the ICF is as follows:



RSV (Reserved by system)
Set to 00.

GCT (Permissible Number of Gateways)
Set to 02.

DNA (Destination Network Address)

01 to 7F: Destination network address (1 to 127)

DA1 (Destination Node Number)

00: Local PC Unit
01 to 7E: Node in Ethernet or SYSMAC NET Link System (1 to 126)
01 to 3E: Node in SYSMAC LINK System (1 to 62)
FF: Broadcasting

DA2 (Destination Unit Address)

00: PC (CPU)
FE: Communications Unit (i.e., the Unit at the node receiving FINS data)
10 to 1F: CPU Bus Unit

SNA (Source Network Address)

01 to 7F: Local network address (1 to 127)

SA1 (Source Node Number)

00: Local PC Unit
01 to 7E: Node in Ethernet or SYSMAC NET Link System (1 to 126)
01 to 3E: Node in SYSMAC LINK System (1 to 62)

SA2 (Source Node Address)

00: PC (CPU)
FE: Communications Unit
10 to 1F: CPU Bus Unit

SID (Service ID)
The SID is used to identify the process that data is sent from. Set any desired number from 00 to FF for the SID. The same number will be returned in the response, allowing you to match commands and responses in your application.

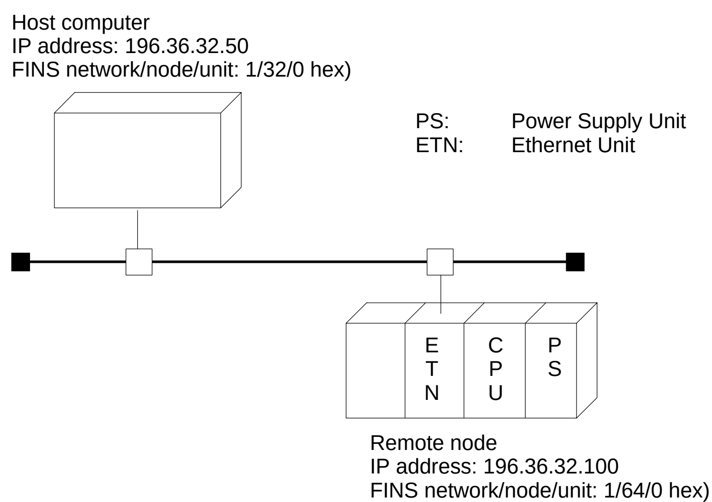
3-9-2 Designating Remote Addresses

UDP sockets are used when sending FINS commands from a host computer to a PC. This section provides an example of addressing remote PCs from the host computer for communications. In the first example, the PC and the host computer are on the same network. In the second example, the PC and host computer are on different networks connected by a PC gateway, i.e., a PC that belongs to both networks.

Intranetwork Addressing

The communications parameters specified from the host computer would be as follows:

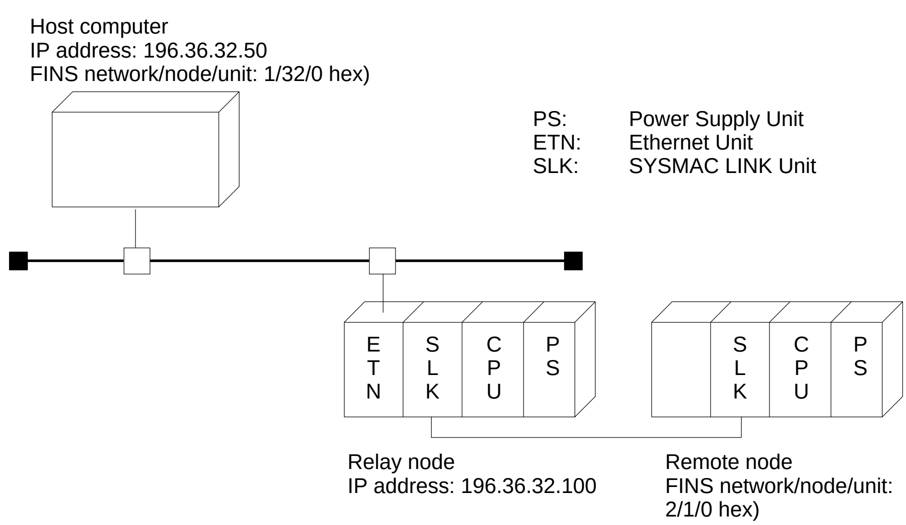
Destination IP Address:	196.36.32.100 (Ethernet Unit of remote node)
UDP port number:	FINS UDP port No. (Ethernet Unit of remote node)
FINS addresses:	DNA, DA1, DA2 = 1, 64, 0 (hex) (PC)
	SNA, SA1, SA2 = 1, 32, 0 (hex) (host computer)



Internetwork Addressing

The communications parameters specified from the host computer would be as follows:

Destination IP Address:	196.36.32.100 (Ethernet Unit of relay node)
UDP port number:	FINS UDP port number (Ethernet Unit of relay node)
FINS addresses:	DNA, DA1, DA2 = 2, 1, 0 (hex) (PC of remote node)
	SNA, SA1, SA2 = 1, 32, 0 (hex) (host)



3-9-3 Sample Program

Operation of the Program

This program reads 150 words of the PC memory beginning at D00100 by sending an FINS command (MEMORY AREA READ, command code 0101) from a UNIX workstation to the PC on the Ethernet network. If no response is received within 2 seconds of sending the FINS command, the command will be resent.

Settings

The Ethernet Unit IP address is 196.36.32.100, and the FINS node number is 100. IP address conversion is set to automatic generation.

The workstation’s IP address is 196.36.32.50 and it’s FINS node number is 50.

The FINS UDP port number is 9600 (default).

Sample Program

```
#include <errno.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <signal.h>

#define FINS_UDP_PORT 9600
#define SERV_IP_ADDR ”196.36.32.100”     /*ETN IP ADDRESS*/
#define MAX_MSG      2010
#define RESP_TIMEOUT 2


/*
* FINS COMMUNICATIONS SAMPLE PROGRAM
*/
main(argc,argv)
int   argc;
char  *argv[];
{
     int    sockfd;
     struct sockaddr_in , ws_addr,cv_addr;
     char   fins_cmnd[MAX_MSG],fins_resp[MAX_MSG];
     int    sendlen,recvlen,addrlen;
     char   sid=0;
     extern recv_fail();

     /*GENERATE UDP SOCKET*/
     if((sockfd=socket(AF_INET,SOCK_DGRAM,0))<0)
          err_exit(”can’t open datagram socket”);

     /*ALLOCATE IP ADDRESS AND PORT # TO SOCKET*/
     bzero((char*)&ws_addr,sizeof(ws_addr));
     ws_addr.sin_family=AF_INET;
     ws_addr.sin_addr.s_addr=htonl(INADDR_ANY);
     ws_addr.sin_port=htons(FINS_UDP_PORT);
     if(bind(sockfd,(struct sockaddr*)&ws_addr,sizeof(ws_addr))<0)
          err_exit(”can’t bind local address”);

     /*
     *GENERATE MEMORY AREA READ COMMAND
     *(READ 150 WORDS FROM D00100.)
     /*
     fins_cmnd[0]=0x80;  /*ICF*/
     fins_cmnd[1]=0x00;  /*RSV*/
```

```
        fins_cmnd[2]=0x02;  /*GCT*/
        fins_cmnd[3]=0x01;  /*DNA*/
        fins_cmnd[4]=0x64;  /*DA1*/      /*WS FINS NODE NUMBER*/
        fins_cmnd[5]=0x00;  /*DA2*/
        fins_cmnd[6]=0x01;  /*SNA*/
        fins_cmnd[7]=0x32;  /*SA1*/      /*ETN FINS NODE NUMBER*/
        fins_cmnd[8]=0x00;  /*SA2*/
        fins_cmnd[9]=++sid; /*SID*/
        fins_cmnd[10]=0x01; /*MRC*/
        fins_cmnd[11]=0x01; /*SRC*/
        fins_cmnd[12]=0x82; /*VARIABLE TYPE: DM*/
        fins_cmnd[13]=0x00; /*READ START ADDRESS: 100*/
        fins_cmnd[14]=0x64;
        fins_cmnd[15]=0x00;
        fins_cmnd[16]=0x00; /*WORDS READ: 150*/
        fins_cmnd[17]=0x96;


        /*SEND FINS COMMAND*/
        bzero((char*)&cv_addr,sizeof(cv_addr));
        cv_addr.sin_family=AF_INET;
        cv_addr.sin_addr.s_addr=inet_addr(SERV_IP_ADDR);
        cv_addr.sin_port=htons(FINS_UDP_PORT);

        singnal((SIGALRM,recv_fail);

    CMND_SEND:
        sendlen = 18;
        if(sendto(sockfd,fins_cmnd,sendlen,0,&cv_addr,sizeof(cv_addr))
        ==sendlen){
              alarm(RESP_TIMEOUT); /*START RESPONSE MONITOR TIMER*/
              printf(”send length %d¥n”,sendlen);
        }
        else{
              err_exit(”send error”);
        }

        /*RECEIVE FINS RESPONSE*/
        if((recvlen = recvfrom(sockfd,fins_resp,MAX_MSG,0,&cv_addr,&addrlen))
        <0){
              if(errno == EINTR)
                    goto CMND_SEND; /*RE-SEND FINS COMMAND*/
              err_exit(”receive error”);
        }
        else{
              alarm(0); /*STOP RESPONSE MONITOR TIMER*/
              printf(”recv length %d¥n”,recvlen);
              if(recvlen<14) /*ILLEGAL RESPONSE LENGTH CHECK*/
                    err_exit(”FINS length error”);
              if((fins_cmnd[3]!=fins_resp[6])||(fins_cmnd[4]!=fins_resp[7])
                    ||(fins_cmnd[5]!=fins_resp[8])){ /*DESTINATION ADDRESS CHECK*/
                    err_exit(”illegal source address error”);
              }
              if(fins_cmnd[9]!=fins_resp[9]) /*SID CHECK*/
                    err_exit(”illegal SID error”);
```

```
       }


       /*CLOSE SOCKET*/
       close(sockfd);
}
/*
* ERROR PROCESSING FUNCTIONS
*/
err_exit(err_msg)
char   *err_msg;
{
       printf("client: %s %x¥n",err_msg,errno);
       exit(1);
}

/*
*SIGNAL CAPTURE FUNCTIONS
*/
recv_fail()
{
       printf("response timeout error ¥n");
}
```

3-9-4 Minimum Delays for Accessing PC Memory

The time for the response to be received after a remote node on the Ethernet network sends a memory area read or write command to a PC can be calculated using the following formula. This time may be extended under some operating conditions.

Write command delay time = remote node communications processing time + 25 + (0.047 x words written) (ms)

Read command delay time = remote node communications processing time + 29 + (0.032 x words read) (ms)

PC CPU Execution Setting

If the PC CPU is set to synchronous processing, the actual delay time may be longer than the delay time calculated with the formulas above. Therefore, when the PC is set to run synchronously, replace the calculated delay time with the first integer multiple of the cycle time which exceeds the calculated delay time.

Note The transfer time may exceed the calculated value due to the actual operating environment. Factors affecting the transfer time are network traffic, the window size of each node, Ethernet Unit traffic (e.g., socket services, FTP server communications, etc.), and the system configuration.

Example

This example shows calculations for a host computer sending a write command for 256 words to the PC. The minimum transfer delay time is given below for the following conditions:

PC CPU operation setting = synchronous
PC CPU cycle time = 40 ms

Minimum transfer delay time

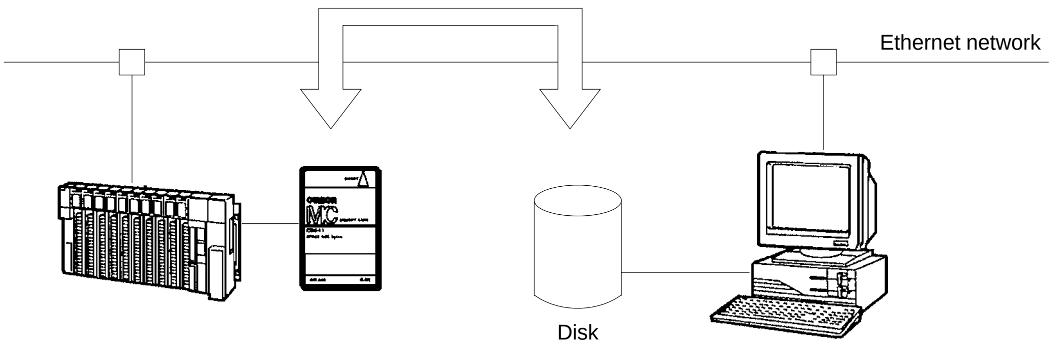
= host computer communications processing time + 25 + (0.047 x 250)
= host computer communications processing time + 36.75
→ host computer communications processing time + **40 (ms)**

SECTION 4
FTP Server Communications

This section describes the features and use of the FTP (File Transfer Protocol) from the host computer to read and write files to and from a Memory Card inserted in the PC. The Ethernet Unit uses the FTP server for these operations.

4-1 Introduction

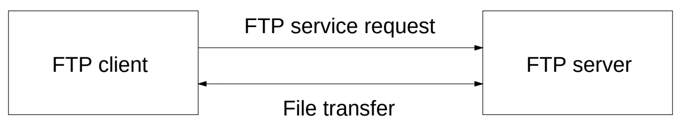
The Ethernet Unit supports the server functions of FTP (File Transfer Protocol), a standard protocol TCP/IP. FTP is used to transfer data by files. FTP allows a host computer on the Ethernet network to read and write files to and from a Memory Card inserted in the PC's CPU. The PC, however, is unable to read or write files at other nodes using FTP because the Ethernet Unit does not support FTP client functions.



Note The FTP server manipulates files in the Memory Card inserted in the PC. A formatted Memory Card must be inserted in the PC before the FTP server will attempt to read or write files there. Refer to *4-3 Using Memory Cards* for details on initializing Memory Cards.

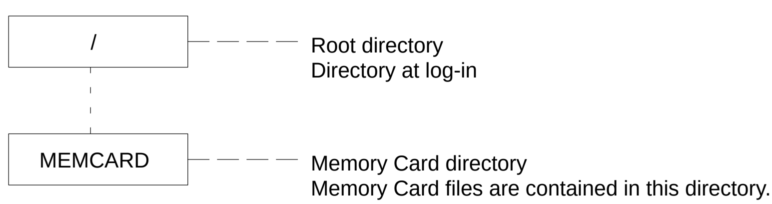
FTP Clients and the FTP Server

When the FTP service is started, the node requesting file transfer to/from another node is called the FTP client, and the node responding to the request is called the FTP server. The Ethernet Unit supports FTP server functions only.



4-1-1 Directory Structure and Files

The current directory when a host computer logs onto the Ethernet Unit is the root directory (/). The files in the Memory Card inserted in the PC are contained in the MEMCARD directory under the root directory. The MEMCARD directory is not a physical directory in the Memory Card, rather a virtual directory in the Ethernet Unit. This directory exists only when a Memory Card is correctly inserted.



Note No MEMCARD directory will exist after logging in if no Memory Card is inserted in the PC or if the Memory Card Power indicator is not lit.

Files in Memory Card

Files in the Memory Card are identified by an 8-character name and a 3-character extension. The size of the files and the number that can be stored depends on the type of Memory Card. Refer to *4-3 Using Memory Cards* for details.

Note When a file of the same name as a file already on the Memory Card is written using FTP, the existing file will be overwritten. If a write error occurs due to insufficient space or some other reason when writing a file to the Memory Card, the entire file will be deleted. To prevent the loss of important data, we recommend that you make regular back-ups of the files on Memory Cards.

4-1-2 Connecting to the FTP Server

The host computer must connect to the FTP server before the FTP server functions can be used. The login name and, if necessary, the password must be input to connect to the FTP server.

Login Name and Password

The login name and password to connect the FTP are set in the CPU Bus Unit system setup. If no login name is specified, the default name of "CONFIDENTIAL" will be used and no password will be required or can be used. Refer to *4-2 Setting Login Names and Passwords* for details.

Connection Message

Only a single client can be connected to the Ethernet Unit's FTP server at any one time, i.e., multiple clients cannot be simultaneously connected to an Ethernet Unit FTP server. If a connection request is received from a client while the FTP server is already connected to a client, a message will be returned to refuse the connection request.

Message for Normal Connection
The following message will be returned if connection to the FTP server was successfully completed.

220 XXX.XX.XX.XX FTP server (Version y.yy zzzzzzzzzzzzzz) ready.

Here,
XXX.XX.XX.XX: Ethernet Unit's IP address
y.yy: Ethernet Unit's FTP version
zzzzzzzzzzzzzz: Date FTP server was created

Message to Refuse Connection
The following message will be returned if connection was refused because the FTP server is already connected to another host.

221 FTP server busy. Goodbye.

Note

1. The FTP server does not lock files when accessing them, i.e., a file can be accessed by more than one host at the same time. File contents cannot be guaranteed if a file is accessed by more than one host in either of the following two ways:

   - If two or more Ethernet Unit simultaneously access the same file when multiple Ethernet Units are mounted and each is connected to a different FTP client (host).
   - If the FTP server accesses a file which is currently being accessed by another Communications Unit, such as SYSMAC NET Link or SYSMAC LINK Unit.

2. If the FTP server is started (by a connection request) immediately after FTP server services are ended or the Unit is booted, a busy response may be returned and connection may fail. This occurs because the Ethernet Unit is executing the FTP server shut-down procedure or is preparing for boot-up. Wait a short time before sending the connection request again.

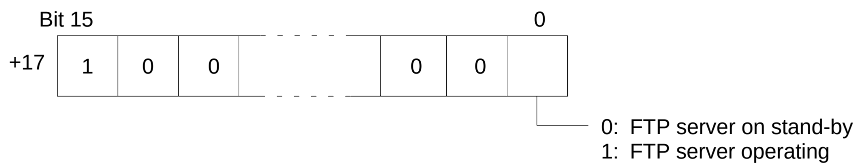
4-1-3 FTP Server Status

The FTP server status can be checked in the FTP Status words kept in the CPU Bus Unit Area or from the status of the FTP indicator on the front of the Ethernet Unit.

FTP Status

The FTP server status is stored in the the CPU Bus Unit Area. The address where this status is stored is calculated by the following formula:

Word = 1500 + (25 x unit number) + 17



FTP Indicator

Status	Meaning
Not lit	FTP server on stand-by
Lit	FTP server operating

Note

1. The FTP indicator lights to indicate that a Memory Card file operation is underway. Do not pull out the Memory Card or turn OFF the PC power supply when this indicator is lit; doing so may destroy the files stored in the Card.
2. While the FTP client and FTP server are connected (i.e., from when an Open command is sent to when a Close command is sent), the FTP status is 1: FTP server operating.
3. If executing file processing instructions, such as READ DATA FILE (FILR(180)) or WRITE DATA FILE (FILW(181)), in the program on the CPU Unit, the FTP status can be used for exclusive control.

4-1-4 File Transfer Time

It may take several minutes to transfer a file using the FTP server, depending on the contents of the file. The table below shows the approximate file transfer times required for various Memory Cards.

Card capacity (bytes)		512 K	128 K				
Memory Card type		**EPROM**	**EEPROM**	**RAM**			
PC	**Operating mode**	**PROGRAM**	**PROGRAM**	**PROGRAM**	**Asynchronous RUN**	**Synchronous RUN**	**Synchronous RUN**
	Cycle time	**---**	**---**	**---**	**variable**	**20 ms**	**50 ms**
put execution (writing)	**Bytes**						
	1 K	---	3 s	2 s	2 s	2 s	2 s
	30 K	---	19 s	16 s	16 s	9 s	8 s
	60 K	---	40 s	34 s	36 s	20 s	17 s
	120 K	---	100 s	88 s	90 s	46 s	42 s
get execution (reading)	**Bytes**						
	1 K	3 s	3 s	2 s	2 s	2 s	2 s
	30 K	5 s	6 s	5 s	6 s	4 s	4 s
	60 K	10 s	12 s	12 s	12 s	8 s	8 s
	120 K	22 s	30 s	28 s	30 s	18 s	16 s

Note EEPROM Memory Cards cannot be written with the *put* (write) operation unless a V1 CPU is used and the PC is in PROGRAM mode.

4-2 Setting Login Names and Passwords

The default FTP login name is “CONFIDENTIAL” and no password is required nor can one be used if this login name is used. Login is completed by typing only “CONFIDENTIAL”. A different login name and a password can be set as required using the CPU Bus Unit system setup.

4-2-1 Setting Method

Set the login name and password in the system setup for the Ethernet Unit. Refer to Section 2.6 *Ethernet Unit System Setup* for details about other items in this area.

System Setup

Offset	Contents
+0 to 1	Mode setting (2 bytes)
+2 to 7	Not used (6 bytes)
+8 to 11	Local IP address (4 bytes)
+12 to 15	Subnet mask (4 bytes)
+16 to 17	FINS UDP port # (2 bytes)
+18 to 29	FTP login name (12 bytes)
+30 to 37	FTP password (8 bytes)
+38 to 231	IP address table (194 bytes)
+232 to 297	IP router table (66 bytes)

The CPU Bus Unit system setup can be changed from the CVSS.

If no login name is set, that is, if the login name begins with a null-code ($00), the default name of “CONFIDENTIAL” will be used and the password setting will be ignored.

4-2-2 Setting Conditions

The login name can be up to 12 bytes long. If the name occupies less than 12 bytes, fill the remaining bytes with null-codes ($00) to bring the total length to 12 bytes. The password can be up to 8 bytes long. If the name occupies less than 8 bytes, fill the remaining bytes with null-codes ($00) to bring the total length to 8 bytes. If a login name is set, a password must also be set, or login will not be possible.

The login name and password can contain any alphanumerics plus “-” and “_”. Use only ASCII characters. Refer to *Appendix E ASCII Characters* for a list of ASCII characters.

Note If the login name contains a character which is not an alphanumeric, “-” or “_”, the default name of “CONFIDENTIAL” will be used and the password setting will be ignored.

4-2-3 Sample Setting

An example of the CPU Bus Unit system setup made with the CVSS is shown below. In the example, the login name is “ABCD-123” and the password is “OMRON”. As the login name is less than 12 bytes and the password is less than 8 bytes, the remaining bytes are filled with null-codes ($00).

Character	ASCII (hexadecimal)
‘A’	$41
‘B’	$42
‘C’	$43
‘D’	$44
‘-’	$2D
‘1’	$31
‘2’	$32
‘3’	$33
‘O’	$4F
‘M’	$4D
‘R’	$52
‘O’	$4F
‘N’	$4E

CVSS Displays

```
                [CPU SIOU Unit System Setup]
unit # 00 ET

BYTE  b7         b0    HEX   BYTE  b7         b0    HEX
+  0  [0000  0000]     00    + 10  [0010  0100]     24
+  1  [0001  0001]     11    + 11  [0000  1000]     08
+  2  [0000  0000]     00    + 12  [1111  1111]     FF
+  3  [0000  0000]     00    + 13  [1111  1111]     FF
+  4  [0000  0000]     00    + 14  [1111  1111]     FF
+  5  [0000  0000]     00    + 15  [0000  0000]     00
+  6  [0000  0000]     00    + 16  [0000  0000]     00
+  7  [0000  0000]     00    + 17  [0000  0000]     00
+  8  [1000  0010]     82    + 18  [0100  0001]     41
+  9  [0001  1001]     19    + 19  [0100  0010]     42
```

Login Name Setting (contd.) (bytes +18 to +19)

```
                [CPU SIOU Unit System Setup]
unit # 00 ET

BYTE  b7         b0    HEX   BYTE  b7         b0    HEX
+ 20  [0100  0011]     43    + 30  [0100  1111]     4F
+ 21  [0100  0100]     44    + 31  [0100  1101]     4D
+ 22  [0010  1101]     2D    + 32  [0101  0010]     52
+ 23  [0011  0001]     31    + 33  [0100  1111]     4F
+ 24  [0011  0010]     32    + 34  [0100  1110]     4E
+ 25  [0011  0011]     33    + 35  [0000  0000]     00
+ 26  [0000  0000]     00    + 36  [0000  0000]     00
+ 27  [0000  0000]     00    + 37  [0000  0000]     00
+ 28  [0000  0000]     00    + 38  [0100  0001]     00
+ 29  [0000  0000]     00    + 39  [0100  0010]     00
```

Login Name Setting (contd.) (bytes +20 to +25)

Password Setting (bytes +30 to +34)

4-3 Memory Cards

Memory Cards can be used to store data or programs as files to expand the memory storing capacity of the PC. Memory Cards fit into the slot located on the lower left side of the CPU. Memory Cards are not provided with the PC and must be ordered separately and installed in the CPU. There are three types of Memory Cards that can be used for the CV-series PCs: RAM, EEPROM, and EPROM.

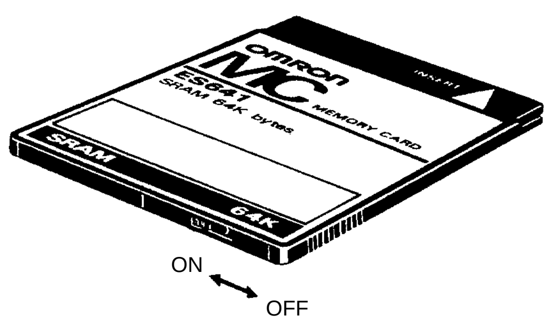
4-3-1 Types of Memory Cards

Memory type	Model	Capacity	Remarks
RAM	HMC-ES641	64K bytes	All FTP commands can be used.
	HMC-ES151	128K bytes	
	HMC-ES251	256K bytes	
	HMC-ES551	512K bytes	
EEPROM	HMC-EE641	64K bytes	All FTP commands can be used in PROGRAM mode as long as a V1 CPU is used.
	HMC-EE151	128K bytes	Only FTP read commands can be used in any other mode or with pre-V1 CPUs. (CV500-MCW□□ Memory Card Writer can also be used to write files.)
EPROM	HMC-EP551	512K bytes	CV500-MCW□□ Memory Card Writer required to write files (cannot be written with FTP commands).
	HMC-EP161	1M bytes	

RAM and EEPROM Cards

Data can be randomly written to and read from RAM using FTP commands. The memory of a RAM Cards is erased, however, when power is not supplied to the CPU or when the RAM Card is removed from the CPU without a backup battery. EEPROM Cards can also be written while mounted in the CPU, but only when the PC is in PROGRAM mode and only when a V1 CPU is used. (See *Appendix A Standard Models* for specific model numbers.) EEPROM Cards cannot be written while mounted in the PC in any other PC operating mode or with any of the older CPUs.

Both the RAM and EEPROM Memory Cards are equipped with write-protect switches. Setting the write-protect switch to ON prevents data from being written to or erased from the Card. Setting the write-protect switch to OFF allows data to be written to or erased from the Card.



RAM Card Backup Battery

Insert a battery into a RAM Memory Card before mounting the Card into the CPU. Leave the battery in its holder. Battery life expectancies are given in the following table. Replace the battery within the time listed. Refer to the *CV-series PC Installation Guide* for the battery replacement procedure.

Card	Capacity	Life
HMC-ES641	64K bytes	5 years
HMC-ES151	128K bytes	3 years
HMC-ES251	256K bytes	1 year
HMC-ES551	512K bytes	6 months

EPROM Cards

Data contained in the ROM Card is stored on EPROM chips and cannot be altered or erased during the CPU's operation. The EPROM chip is mounted to the Memory Card and the entire pack is installed in the CPU. Once data is written to the chip, the data will not be lost when the power to the PC is OFF. ROM Cards are shipped unprogrammed. The ROM Card can be programmed using a CV500-MCW☐☐ Memory Card Writer.

The procedure for erasing EPROM Memory Card data is as follows:

1, 2, 3...

1. Open the memory card cover by pressing at an angle on the catch at the bottom edge of the card using a pointed object, such as a pen.
2. Erase the data from the EPROM chips by exposing the window of the EPROM chips to ultraviolet light. Commercially available EPROM erasers may be used. To ensure complete erasure, the Memory Card should be subject to a minimum exposure as specified by the eraser manufacturer.
3. Close the cover and slide the catch back into place.

4-3-2 Memory Card Files

Capacities

The data capacities and maximum numbers of files possible for Memory Cards are listed in the following table.

Card capacity	Actual data capacity	Max. number of files
64K bytes	61K bytes	64
128K bytes	125K bytes	64
256K bytes	251K bytes	112
512K bytes	506K bytes	112
1M bytes	1,016K bytes	112

File Names

The file names and extensions determine the file type of files stored on Memory Cards, as shown in the following table. All user names must be 8 ASCII characters or less and must be uppercase alphanumerics. File access will not be possible from some Programming Devices and from the ladder-diagram program if lowercase names are used.

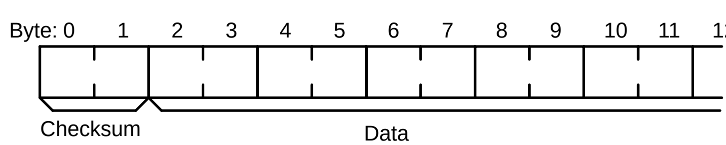
File	File name	Extension
System Setup File (see note)	(User name)	.STD
CIO/DM Area File		.IOM
Ladder Program Part File		.LDP
SFC Program Step File		.SFC
User Program File (entire user program)		.OBJ
System Setup File (automatically loaded at startup)	AUTOEXEC	.STD
User Program File (automatically loaded at startup)	AUTOEXEC	.OBJ

Note

1. Contains PC System Setup, I/O table, routing tables, data link tables, BASIC Unit System Setup, function code settings, and custom area settings.
2. Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for information on autoloading files and settings.

CIO/DM Area Files

The format of CIO/DM Area Files (.IOM) is shown in the following diagram.



The first two bytes of the above format are a checksum calculated from the data that follows it. This checksum is calculated by added two bytes at a time (i.e., word by word) from the beginning of the data to the end of the data and then using the least-significant two bytes of the total as the checksum. Therefore, there should normally be an even number of bytes of data in the file.

The following shows an example calculation of the checksum for a file that contains only 10 bytes total.

Data bytes: 13 3A E4 F3 CC 0B 3C 5F A2 77
Data words: 133A E4F3 CC0B 3C5F A277
Calculation: 133A + E4F3 + CC0B + 3C5F + A277 = 2<u>A30E</u>

The checksum would thus be the last two bytes of the total 2A30E, or A30E.

The checksum is always based on an even number of bytes. If an odd number of bytes is specified, the checksum will be calculated after adding “00,” as shown in the following example.

Data bytes: 13 3A E4 F3 CC 0B 3C 5F A2
Data words: 133A E4F3 CC0B 3C5F A200
Calculation: 133A + E4F3 + CC0B + 3C5F + A200 = 2<u>A297</u>

The checksum would thus be the last two bytes of the total 2A297, or A297.

Precautions

If an odd number of data bytes are written to a Memory Card using FTP commands, you will not be able to read the last byte from the user ladder-diagram program using the FILR(180) instruction. You should thus always write an even number of bytes to Memory Cards.

For example, if the data used in the second example, above, was written to the file DMDATA.IOM in a Memory Card via the FTP using the `put` command and then read back into PC memory using FILR(180) starting from D00000, the results would be as shown below. The control data for FILR(180) is also given. Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for details on the FILR(180) instruction.

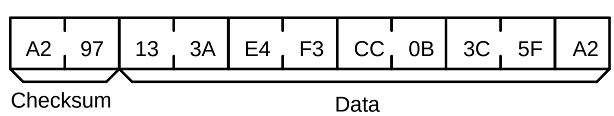
Command

Operands

Words to transfer: 5
1st destination word: D00000
1st control word: D00100 (see table)

Word	Contents	Meaning
D00100	0000	---
D00101	444D	DM
D00102	4441	DA
D00103	5441	TA
D00104	2020	“ ”
D00105	0000	---

Memory Card File: DMDATA.IOM



Results in Memory

The following table shows memory contents resulting from executing FILR(180) as described. Notice that the last byte in the Memory Card has not been read out.

Word	Contents
D00000	133A
D00001	E4F3
D00002	CC0B
D00003	3C5F
D00004	0000

Note UM and DM Area data is binary. Always be sure the data type is set to binary using the `type` command when reading or writing UM or DM Area data via FTP commands.

4-3-3 Handling Memory Cards

Formatting Cards

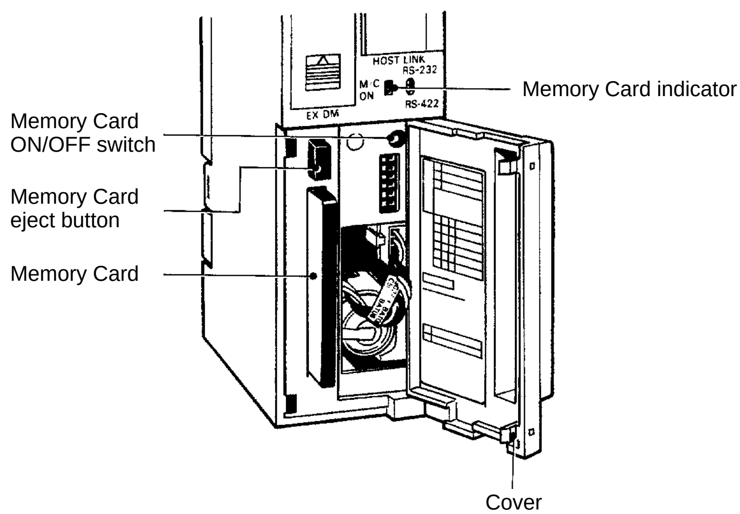
Memory Cards must be formatted before they can be used. Any card can be formatted using a Memory Card Writer, normally using CVSS operations. RAM Cards or EEPROM Cards can also be formatted after they are mounted to the PC by using an FINS command (code: 2204). EEPROM Cards, however, can be formatted mounted to the PC only in PROGRAM mode and only if a V1 CPU is being used.

Mounting a Memory Card

Mount a Memory Card to the CPU using the following procedure.

1, 2, 3...

1. If the Memory Card is RAM or EEPROM, set the write-protect switch to OFF so that data can be written to the Card.
2. Open the cover of the Memory Card compartment.
3. Insert the Memory Card into its compartment. In doing so, a slight resistance will be felt as the connector on the Memory Card mates with the connector on the CPU. Continue pushing until the Memory Card is inserted completely into the CPU. If the Memory Card ON/OFF switch is ON, the Memory Card indicator will light.
4. Close the cover.



Removing a Memory Card

1, 2, 3...

1. Open the cover of the Memory Card compartment.
2. Confirm that the FTP indicator on the Ethernet Unit is not lit. If the FTP indicator is lit, disconnect the FTP and be sure that the FTP indicator is not lit before continuing.
3. Press the Memory Card ON/OFF switch once if the Memory card indicator is lit. The Memory Card indicator will turn OFF.
4. Press the Memory Card eject button. The Memory Card will be released allowing it to be removed.
5. Pull out the Memory Card.
6. Close the cover.

Note

1. Do not remove the Card while the Memory Card indicator is lit; doing so may result in data errors in the memory.
2. Do not turn off the Memory Card ON/OFF switch when the FTP indicator is lit. Memory Card reading/writing may be taking place whenever the FTP indicator is lit.

3. Do not expose the Memory Card to high temperature, humidity, or direct sunlight.
4. Do not bend the Card or subject it to shock.
5. Do not apply excess force to the Card when inserting or removing it.

4-4 Using FTP Commands

This section describes the FTP commands which the host computer (FTP client) can send to the Ethernet Unit's FTP server. The description is for a workstation manufactured by OMRON. The descriptions should also apply to UNIX workstations made by other manufacturers, but slight differences may arise. Refer to your workstation's operation manuals for details.

4-4-1 Table of Commands

The FTP commands which can be sent to the Ethernet Unit are listed in the following table.

Command	Description
`open`	Connects the specified host FTP server.
`user`	Specifies user name for the remote FTP server.
`ls`	Displays the Memory Card file names.
`dir`	Display the Memory Card file names and details.
`cd`	Changes the Ethernet Unit work directory to the specified directory.
`pwd`	Displays the Ethernet Unit work directory.
`type`	Specifies the data type of transferred files.
`get`	Transfers the specified file from the Memory Card to the local host.
`mget`	Transfers multiple files from the Memory Card to the local host.
`put`	Transfers the specified local file to the Memory Card.
`mput`	Transfers multiple local files to the Memory Card.
`delete`	Deletes the specified file from the Memory Card.
`mdelete`	Deletes multiple files from the Memory Card.
`close`	Disconnects the FTP server.
`bye`	Closes the FTP (client).
`quit`	Closes the FTP (client).

Note The Ethernet Unit is considered to be the remote host and the host computer (FTP client) is considered to be the local host.

4-4-2 Using the Commands

open

Format

`open [`*`IP_address`* or *`host_name_of_FTP_server`*`]`

Function

Connects the FTP server. Normally when the FTP client is booted, the FTP server IP address is specified to execute this command automatically.

user

Format

`user [`*`user_name`*`]`

Function

Specifies the user name. Specify the FTP login name set in the Ethernet Unit system setup. The default FTP login name is "CONFIDENTIAL".

If a non-default login name is used, it must be followed by the password. In this case, enter the FTP password set in the system setup.

The user name is automatically requested immediately after connection to the FTP server.

ls

Format `ls [-l] [`*`REMOTE_FILE_NAME`*` [`*`local_file_name`*`]]`

Function Displays the remote host (Ethernet Unit) file names.

Set the switch [-l] to display not only the file names but the creation date and size as well. If the switch is not set, only the file names will be displayed.

You can specify a file name in the Memory Card if desired.

If a local file name is specified, the file information will be stored in the specified file in the host computer.

When this command is executed, the system checks whether a Memory Card is inserted. If a Memory Card is inserted after the FTP is connected, Memory Card file operations are possible only after executing this command or the `dir` command on the root directory.

Example `ls /`

dir

Format `dir [`*`REMOTE_FILE_NAME`*` [`*`local_file_name`*`]]`

Function Displays the file names, date created, and size of the the files in the remote host (Ethernet Unit). It displays the same information as command “ls –l”.

Specify a file name in the Memory Card as the remote file name.

If a local file name is specified, the file information is stored in the specified file in the host computer.

When this command is executed, the system checks whether a Memory Card is inserted. If a Memory Card is inserted after the FTP is connected, Memory Card file operations are possible only after executing this command or the `ls` command on the root directory.

Example `dir /`

pwd

Format `pwd`

Function Displays the remote host’s (Ethernet Unit) current work directory.

cd

Format `cd [`*`directory_name`*`]`

Function Changes the remote host (Ethernet Unit) work directory to the specified remote directory.

The files in the Memory Card are contained in the MEMCARD directory under the root directory (/). The root directory (/) is the directory used when logging into the Ethernet Unit. No MEMCARD directory exists if a Memory Card is not inserted in the PC or if the Memory Card power indicator is not lit.

type

Format `type `*`data_type`*

Function Specifies the file data type. The following data types are supported:

ascii: Files are transferred as ASCII data
binary (image): Files are transferred as binary data.

All files are treated by the PC as binary files. Before reading or writing any files, always use the `type` command to set the file type to binary. File contents cannot be guaranteed if transferred as ASCII data.

The default file type is ASCII.

get

Format `get `*`FILE_NAME`*` [`*`receive_file_name`*`]`

Function Transfers the specified remote file from the Memory Card to the local host.

A receive file name can be used to specify the name of the file in the local host.

mget

Format `mget` *`FILE_NAME`*

Function Allows the use of a wildcard character (*) to transfer multiple remote files from the Memory Card to the local host.

put

Format `put` *`file_name`* `[`*`DESTINATION_FILE_NAME`*`]`

Function Transfers the specified local file to the remote host (Memory Card).

A destination file name can be used to specify the name the file is stored under in the Memory Card.

Any existing file with the same name in the remote host (Memory Card) will be overwritten by the contents of the transferred file.

If an error occurs during file transfer, the file being transferred will be deleted and the transmission will end in an error.

mput

Format `mput` *`FILE_NAME`*

Function Allows the use of a wildcard character (*) to transfer multiple local files to the remote host (Memory Card).

Any existing file with the same name in the remote host (Memory Card) will be overwritten by the contents of the transferred file.

If an error occurs during file transfer, the file being transferred will be deleted and the transmission of that file will end in an error. However, `mput` execution will continue and remaining files will be transferred.

delete

Format `delete` *`FILE_NAME`*

Function Deletes the specified remote file from the Memory Card.

mdelete

Format `mdelete` *`FILE_NAME`*

Function Allows the use of a wildcard character (*) to delete multiple remote files from the Memory Card.

close

Format `close`

Function Disconnects the Ethernet Unit's FTP server.

bye

Format `bye`

Function Ends the FTP (client).

quit

Format `quit`

Function Ends the FTP (client).

4-4-3 Error Messages

The following error messages may be returned by the Ethernet Unit FTP server to the FTP client.

Error message	Description
File name length exceeded max size	The file name or the extension exceeded the maximum length. The maximum file name length is 8 characters and the maximum extension length is 3 characters.
FINS error, MRES 10 command format error, SRES XX.	FINS response error: command format error (see note).
FINS error, MRES 11 parameter error, SRES XX.	FINS response error: parameter error (see note). Insufficient free space in Memory Card when writing a file.
FINS error, MRES 20 read error, SRES XX.	FINS response error: read error (see note).
FINS error, MRES 21 write error, SRES XX.	FINS response error: write error (see note). An attempt was made to write to a write-protected Memory Card.
FINS error, MRES 23 no such Unit error, SRES XX.	FINS response error: No Memory Card inserted (see note) or CPU Rack not serving Ethernet Unit
FINS error, MRES XX; SRES YY.	FINS response error: MRES=XX, MRES=YY (see note).
No Such File or Directory	The specified file or directory name is incorrect or does not exist. No Memory Card inserted (see note). CPU Rack not servicing Ethernet Unit.
Timeout (900 seconds): closing control connection.	Command input timed out. FTP connection is broken if no command is received from the client within 15 minutes.
Write access denied.	Could not write file. This error message will be returned if an attempt is made to write a file to the root directory.

Note MRES and SRES are FINS command response codes. Refer to the *FINS Command Reference Manual* for details.

4-4-4 Example of Using Commands

In this example the host computer (a Unix workstation) runs the following procedure. The following settings are required for this example.

- The Ethernet Unit IP address is stored as the host name "cv500" in /etc/hosts on the workstation.
- The FTP login name is the default name: CONFIDENTIAL
- The production results data in the Memory Card is stored under the file name RESULT.IOM
- The work instructions data in the Memory Card is stored under the file name PLAN.IOM

The following operations transfer the RESULT.IOM production results file from the PC Memory Card to the workstation, then transfer the PLAN.IOM work instructions file from the workstation to the PC Memory Card. Comments are placed after semicolons. User inputs are underlined.

1, 2, 3...

1. Boot FTP and connect to the Ethernet Unit.

```
$ ftp cv500                                ; Boots FTP
connected to cv500
220 **IPaddress** FTP server(**version**date)ready
Name(cv500:root):■
```

2. Input the login name.

```
Name(cv500:root):CONFIDENTIAL              :Input login name
230 Guest logged in.
ftp>■
```

3. A check is made to confirm that the Memory Card is correctly inserted. The MEMCARD directory will be displayed if the card is correctly inserted.

```
ftp>ls                                  ;Memory Card checked.
200 PORT command successful.
150 opening data connection for ls(**IPaddress**port#**)(0 bytes).
MEMCARD
226 Transfer complete.
15 bytes received in 0 seconds(**bytes/s)
ftp>■
```

4. Change to the MEMCARD directory.

```
ftp>cd MEMCARD                          ;Changes directory.
250 CD command successful.
ftp>■
```

5. Specify the file data type as binary data.

```
ftp>type binary                         ;Specifies binary data.
200 Type set to I.
ftp>■
```

6. Read the RESULT.IOM file and transfer it to the workstation.

```
ftp>get RESULT.IOM                      ;Reads file to host.
200 PORT command successful.
150 opening data connection for
result.iom(**IPaddress**port#**)(**bytes).
226 Transfer complete.
**bytes received in*.***seconds (**bytes/s)
ftp>■
```

7. Write the PLAN.IOM file to the Memory Card.

```
ftp>put PLAN.IOM                        ;Writes file to Memory Card.
200 PORT command successful.
150 opening data connection for plan.iom(**IPaddress**port#**).
226 Transfer complete.
**bytes received in*.***seconds (**bytes/s)
ftp>■
```

8. End FTP.

```
ftp>bye                                 ;Quits FTP.
221 Goodbye.
$ ■
```

SECTION 5
Socket Service

This section describes the socket services provided by the Ethernet Unit. Socket services enable data communications between the PC's user program and a host computer program. Refer to documentation for your host computer for details on programming host computer sockets.

5-1 About Socket Services

5-1-1 Sockets

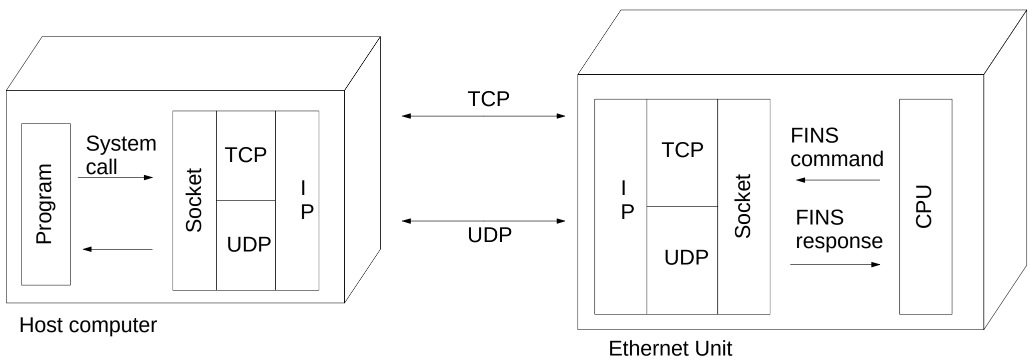
A socket is an interface which allows a user program to directly use TCP (Transmission Control Protocol) and UDP (User Datagram Program). Socket services allow arbitrary data to be sent to and from other nodes.

Socket services are supplied for most host computers (or some personal computers) as a C-language interface library. This interface library allows user programs to communicate using TCP and UDP.

The socket interface is supported for a UNIX computer in the form of system calls. Socket services are achieved for CV-series PCs by sending FINS commands from the user program to an Ethernet Unit. Refer to *5-2 Using Socket Services* and *Section 6 FINS Commands to Ethernet Units* for details.

5-1-2 Socket Operation

A socket operates as shown in the following diagram when communicating between a host computer and an Ethernet Unit. Data from the computer or PC passes through the socket and is communicated using the TCP or UDP protocol.



5-1-3 Differences between TCP and UDP

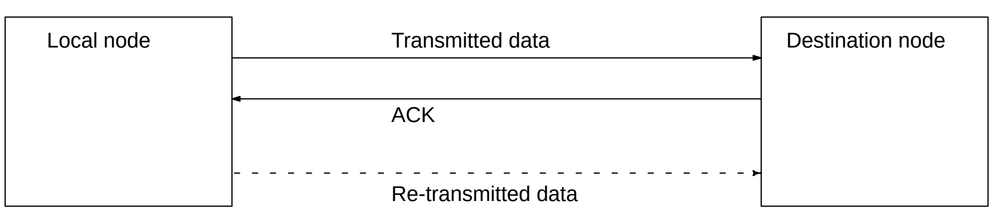
There are differences in the socket services between TCP and UDP.

TCP Communications

The following procedure is followed each time data is transmitted to ensure that the data arrives normally at the destination node:

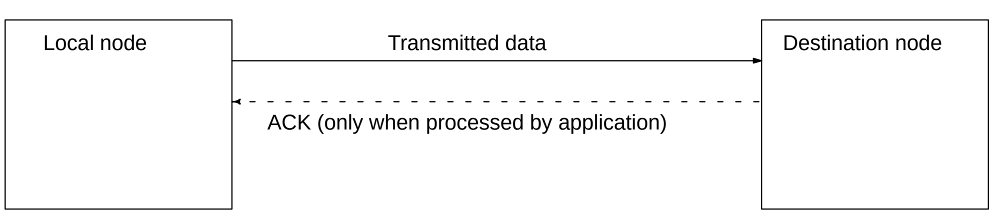
1, 2, 3...

1. The destination node returns ACK when data is received normally.
2. The local node sends the next data after it receives ACK, or it resends the same data if ACK is not returned within the specified time.



UDP Communications

Data is simply sent to the destination. Unlike TCP, the receipt of data is not checked and data is not re-transmitted. To increase communication reliability, data resends must be programmed by the user in user applications.

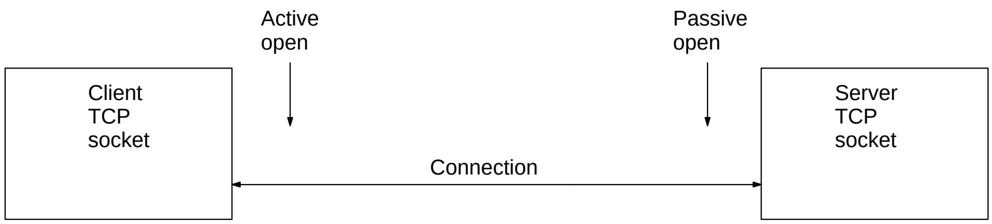


5-1-4 Opening TCP Sockets

To achieve highly reliable data communication, TCP establishes a virtual communications circuit between the two nodes before starting data transmissions. The virtual communications circuit is know as a "connection."

Passive OPEN and Active OPEN

An open command is executed for a node to establish a connection. The open method differs depending on whether the node is a client or server. A passive open method is used to open the node as a server and the active open method is used to open the node as a client.



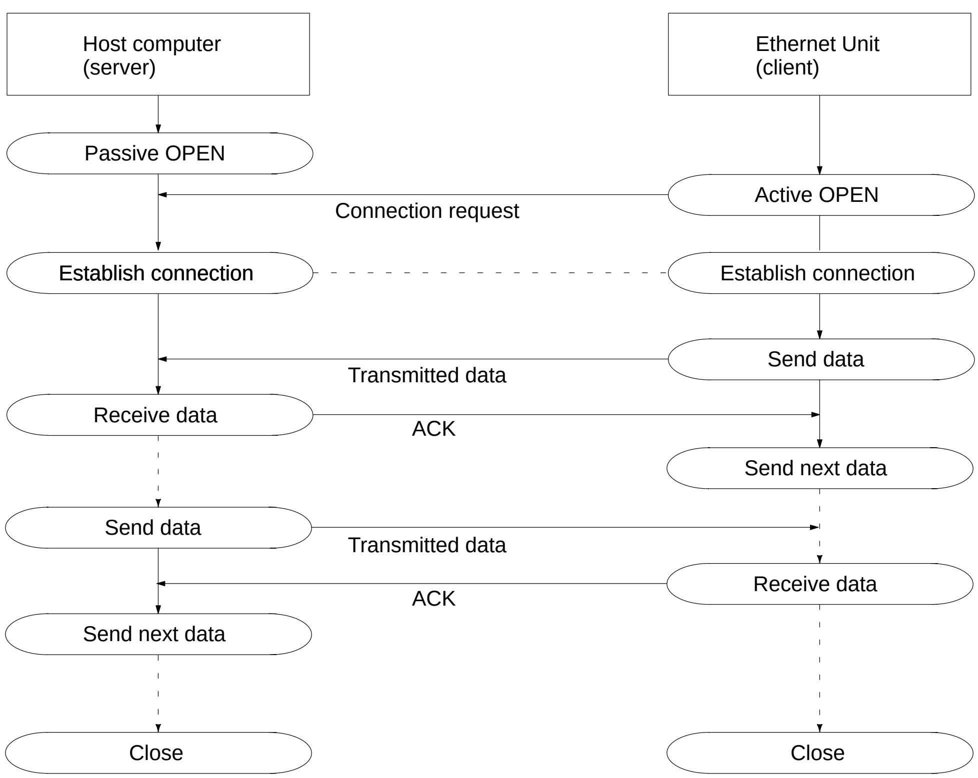
Client and Server

When an application which includes data communications processing between two nodes is executed, the node which provides the service is known as the "server" and the node which requests the service is known as the "client." The server is booted first and waits for a service request from a client. Data is transmitted only after the client requests the server to establish a connection. If the TCP protocol is used, this process is carried out automatically at the protocol level; it is not necessary to run an application program.

Note TCP communications with other TCP sockets will not be possible until the connection for the first socket opened has been closed (8 TCP sockets are available). Communications on multiple UDP sockets is possible.

TCP Communications Procedure

The communications procedure is shown below for communications between a host computer and Ethernet Unit using a TCP socket.



5-1-5 Socket Port Numbers

Port numbers up to 1023 on a UNIX workstation can be used by the superuser only. Port numbers 0 to 255 are reserved for well-known ports. Consequently, port numbers 1024 to 65535 should be used for socket services. The Ethernet Unit does not support port #0.

Some port numbers from 1024 and over may be reserved on some workstations (for example, the X-window server is port #6000). Do not use port numbers that are already reserved for other processes.

The setting status of the Unix workstation port numbers can be checked using the contents of /etc/services.

5-1-6 Fragmentation of Transmitted Data

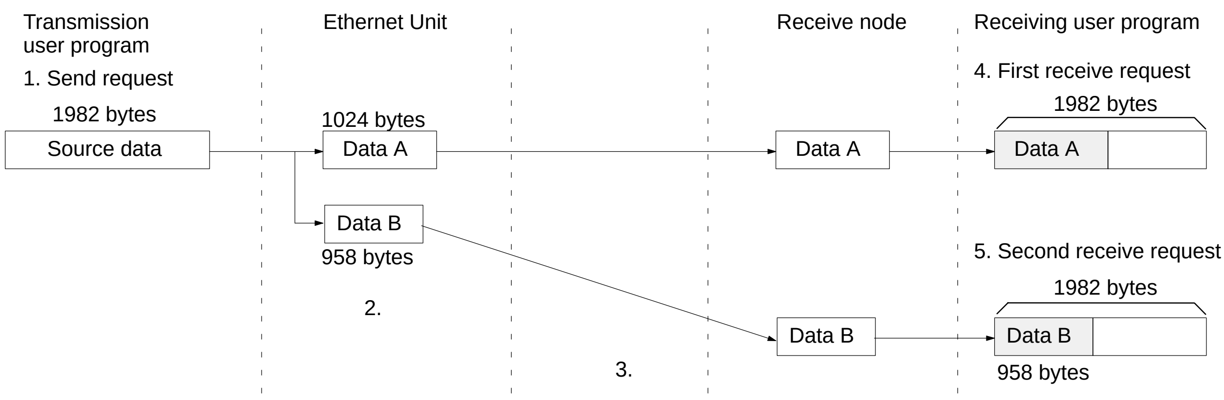
The Ethernet Unit fragments data for TCP transmission into units of 1,024 bytes and data for UDP transmission into units of 1,472 bytes. TCP requires one reception request to receive each unit of data. UDP, however, restores the original data before passing it to the user process, allowing all the data in a single transmission to be received with one reception request. Examples of these are given next.

Cautions when Using TCP

An example of the fragmentation and transmission of data using the TCP is shown in the following illustration.

1, 2, 3...

1. The transmission user program sends a request to send 1,982 bytes of data.
2. The Ethernet Unit fragments the transmission data into Data A with 1,024 bytes and Data B with 958 bytes.
3. Data A and Data B are sent consecutively.
4. The receiving user program sends a request to receive 1,982 bytes of data. However, only data A is sent in the first packet; data B is not received.
5. Another receive request to receive data must be made before the remaining data, Data B, is sent.



When using TCP protocol, the fragmented data is passed to the user program. Therefore, the receiving user program must be able to evaluate the end of the data transmission, and repeatedly send receive requests until all data has been received. The receive request is sent twice in the example shown above, but the data would be even more fragmented if a router were included in the communications path, and the number of receive requests would need to be increased accordingly.

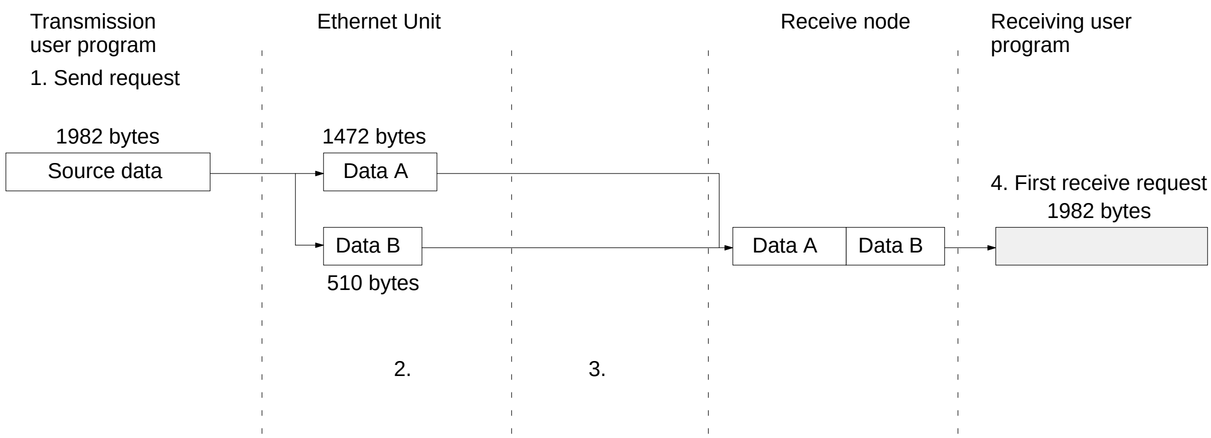
When making the receive request, it is not necessary to specify the same data length as the transmitted data length. For example, if the length setting is shorter than the actual length of the data, all the data can be received by repeating the receive requests.

Cautions when Using UDP

An example of fragmentation and transmission of data using the UDP is shown in the following illustration.

1, 2, 3...

1. The transmission user program sends a request to send 1,982 bytes of data.
2. The Ethernet Unit fragments the transmission data into Data A with 1,472 bytes and Data B with 510 bytes.
3. Data A and Data B are sent consecutively.
4. When the receiving user program sends a request to receive 1,982 bytes of data, Data A and Data B are linked to restore the original data which is passed to the user program.



As shown above, the UDP protocol handles data communications as datagrams, so that the transmitted data is restored to the original data before being passed to the user program. Consequently, if the data length in the receive request is set to the length of the transmitted data, the entire data can be received using a single receive data request. However, if the data length in the receive data request is set smaller than the actual length of the data, all received data exceeding the set data length will be discarded.

5-2 Using Socket Services

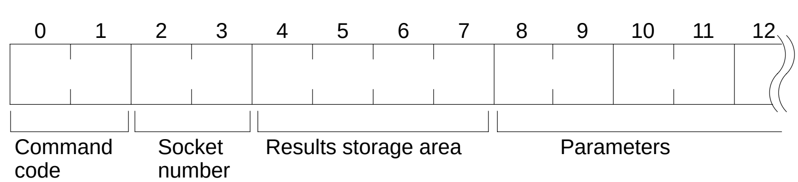
Socket services for CV-series PCs are controlled using FINS commands sent to the Ethernet Unit.

5-2-1 FINS Commands

Socket services are run by executing the DELIVER COMMAND instruction CMND(194) from the PC program to send FINS commands to the Ethernet Unit.

Basic FINS Command Format

The basic format for FINS commands used for socket services is shown in the following diagram.



Command Code:
Specifies the process code requested for the socket.

Socket Number
Specifies the socket number for the process, between 1 and 8.

Results Storage Area
Specifies the area to store the results of the requested process.

Parameters
Specifies the parameters defined for the command code.

Refer to *Section 6 FINS Commands to Ethernet Units* for details about commands.

Table of FINS Commands The following FINS commands can be used for socket services to Ethernet Units.

Command code		Name	Description	Page
MRC	SRC			
27	01	UDP OPEN REQUEST	Opens the UDP socket.	135
	02	UDP RECEIVE REQUEST	Receives data at the UDP socket.	136
	03	UDP SEND REQUEST	Sends data from the UDP socket.	137
	04	UDP CLOSE REQUEST	Closes the UDP socket and ends communications.	138
	10	TCP OPEN REQUEST (PASSIVE)	Opens a TCP socket and waits for connection to another node.	139
	11	TCP OPEN REQUEST (ACTIVE)	Opens a TCP socket and connects to another node.	140
	12	TCP RECEIVE REQUEST	Receives data at the TCP socket.	142
	13	TCP SEND REQUEST	Sends data from the TCP socket.	143
	14	TCP CLOSE REQUEST	Closes the TCP socket and ends communications.	145

5-2-2 Socket Status Area

Each Ethernet Unit has eight TCP sockets and eight UDP sockets. The status of each of these sockets can be confirmed from the CPU Bus Unit Area words allocated to each unit. The address of the first word of the status area is calculated using the following formula. All other words are calculated by adding the specified offset to the address of the first word.

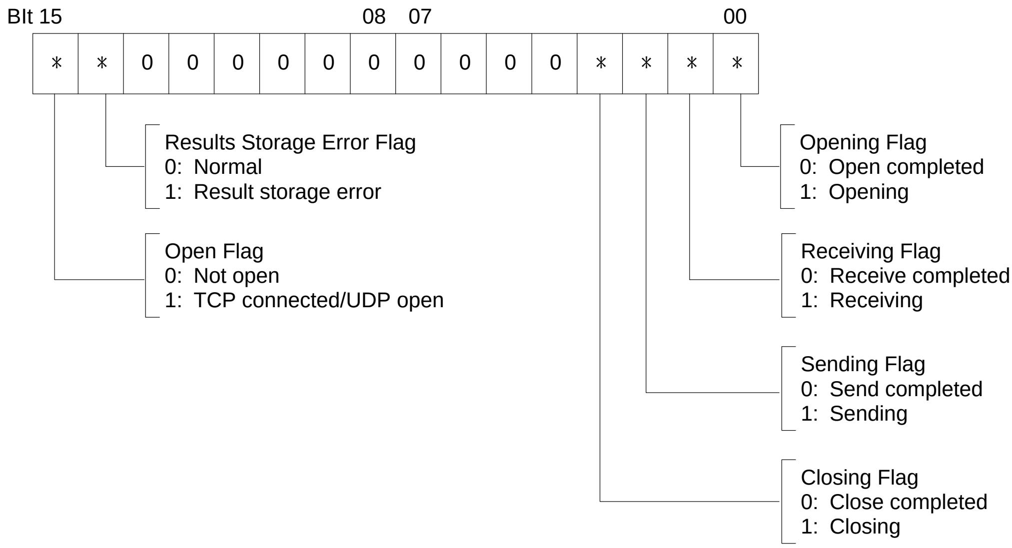
Word = 1500 + (25 x unit number)

Offset	Content
+0 (first word)	Internode Test Start Bit
+1	Status of UDP socket #1
+2	Status of UDP socket #2
+3	Status of UDP socket #3
+4	Status of UDP socket #4
+5	Status of UDP socket #5
+6	Status of UDP socket #6
+7	Status of UDP socket #7
+8	Status of UDP socket #8
+9	Status of TCP socket #1
+10	Status of TCP socket #2
+11	Status of TCP socket #3
+12	Status of TCP socket #4
+13	Status of TCP socket #5
+14	Status of TCP socket #6
+15	Status of TCP socket #7
+16	Status of TCP socket #8
+17	FTP server status

Note The first socket status area word contains the Start Bit for the internode test. Refer to 7-2 *Internode Test* for details.

Bit Configuration

The meaning of individual bits in each status word (except the first) is shown in the following diagram.



Bit	Flag	Value	Status	Description
Bit 0	Opening Flag	1	Opening	ON when an open request is received.
		0	Open complete	OFF when opening has been completed.
Bit 1	Receiving Flag	1	Receiving	ON when a receive request is received.
		0	Receive complete	OFF when receive has been completed.
Bit 2	Sending Flag	1	Sending	ON when a send request is received.
		0	Send complete	OFF when send has been completed.
Bit 3	Closing Flag	1	Closing	ON when a close request is received.
		0	Close complete	OFF when close has been completed.
Bit 4 to 13	Not used			
Bit 14	Results Storage Error Flag	1	Results storage error	ON when the results storage area is incorrectly defined for a FINS command sent to an Ethernet Unit. This flag will not turn ON until the Opening, Receiving, Sending, and Closing Flags turn OFF.
		0	Results storage normal	OFF when the next service is requested at the socket.
Bit 15	Open Flag	1	Open (connected)	ON when opening has been completed. This flag indicates the TCP socket is connected.
		0	Closed	OFF when closing has been completed. Remains OFF if an error occurs during opening.

5-2-3 Using Socket Service

Each Ethernet Unit has eight TCP sockets and eight UDP sockets. Open, close, send, and receive processes are available for communications with sockets.

Open
Enables communications on a specified socket. A socket must be opened before it can be used for socket services. Opening a TCP socket establishes a connection.

Close
Ends use of the socket. Breaks the connection for a TCP socket.

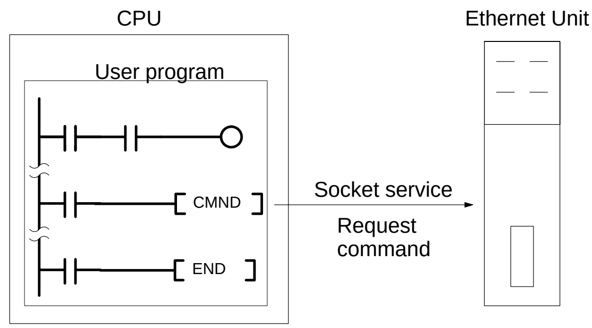
Send
Sends data from a specified open socket.

Receive
Specifies an open socket and receives data from that socket.

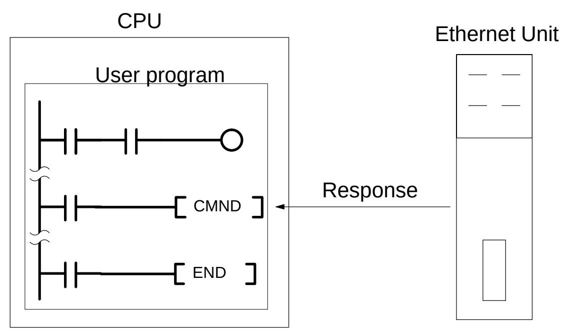
These processes are carried out by sending FINS commands to the Ethernet Unit. The process from sending a request for processing to completion is shown in the following illustrations.

1, 2, 3...

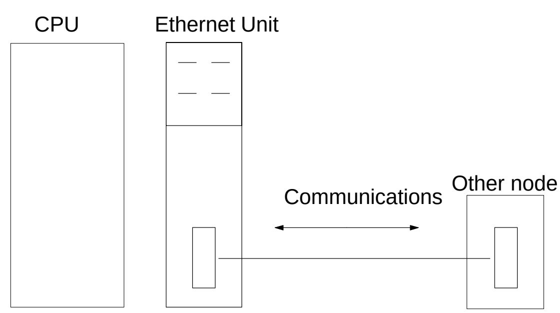
1. Execute a socket service request command (MRC: 27) for the Ethernet Unit using CMND(194).



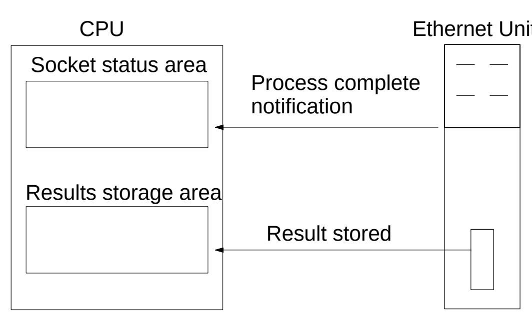
2. CMND(194) ends normally when the socket service request command is received and a response is returned (response code: 0000).



3. The Ethernet Unit starts the process requested by the parameters in the socket service request command.



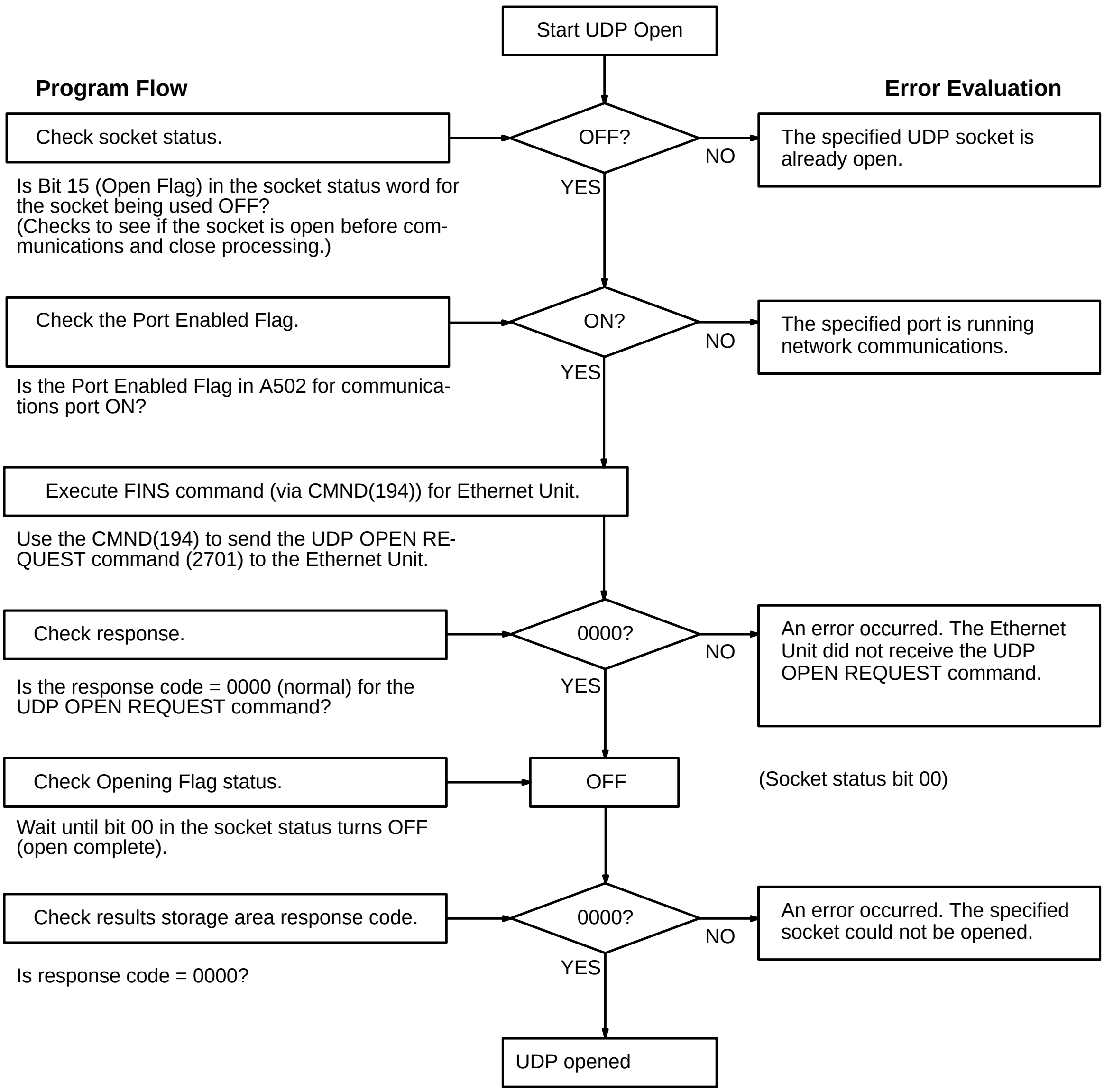
4. When the process has been completed, the result is stored in the results storage area defined in the socket service request command and the socket status will indicate completion of processing.



5-2-4 Socket Services and Socket Status

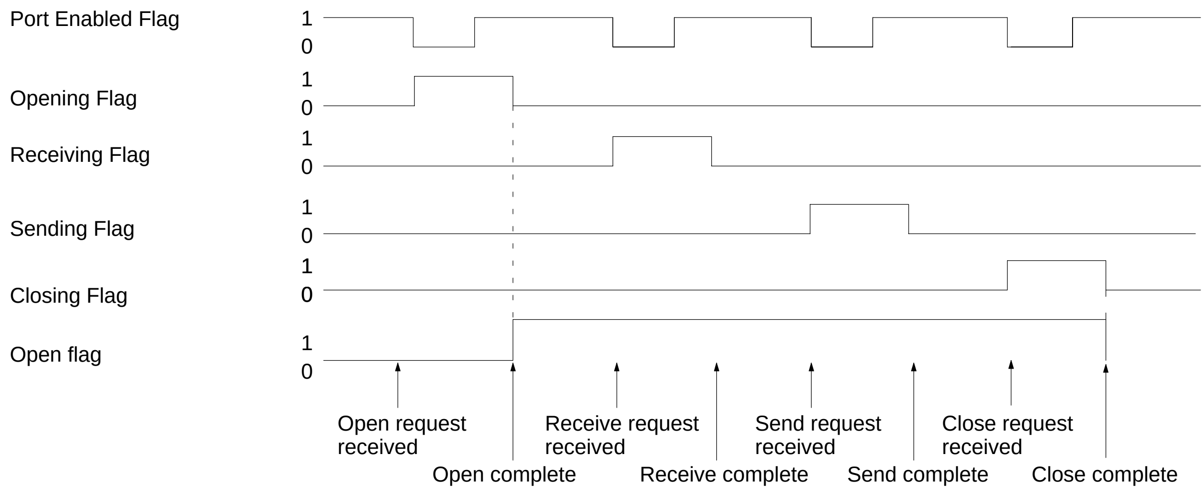
When using socket services, it is important to consider the timing of the status changes in the socket status area. The diagram below shows a program flowchart for opening UDP.

Program flow is similar for other socket services. Replace the names of the appropriate flags in the flowchart to adapt it to other socket services.



5-2-5 Communications Timing Chart

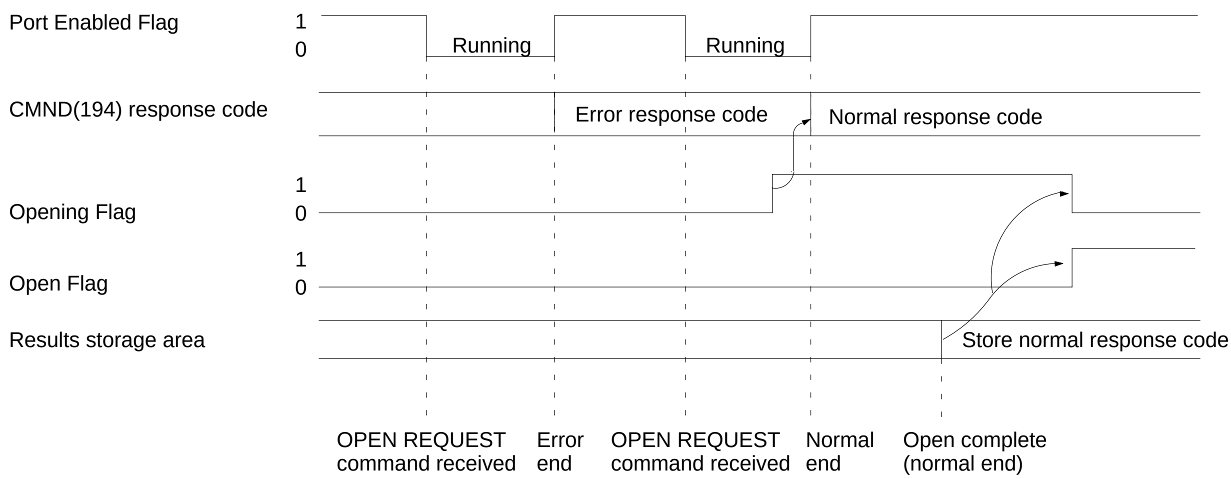
The timing of the status changes of the bits in the socket status area and the Port Enabled Flag is shown in the following diagram.



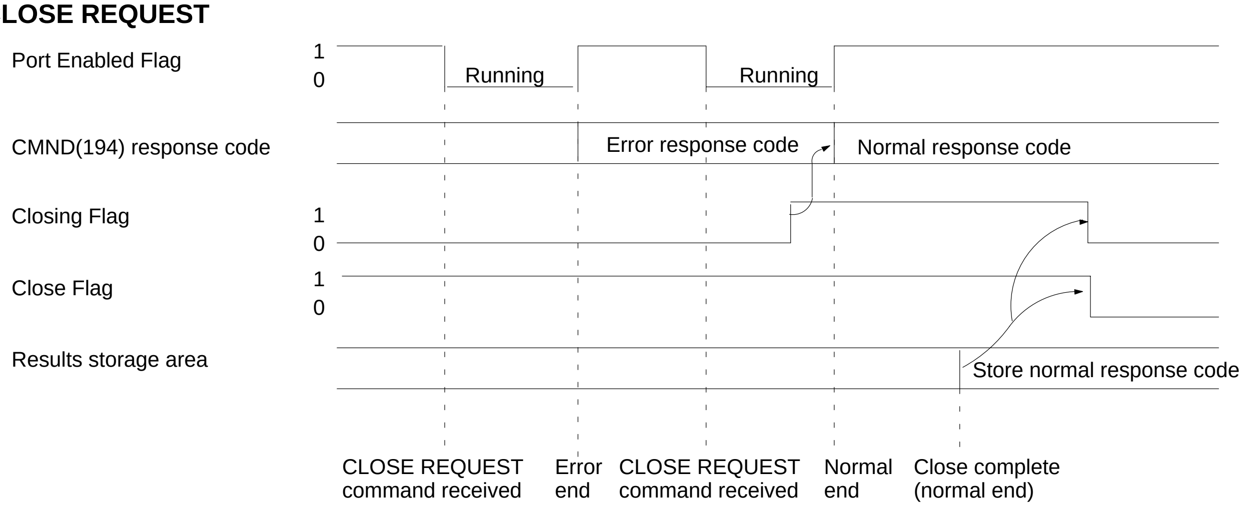
5-2-6 Socket Service Timing Chart

The timing of the socket service open, send, receive and close request commands are shown in the following diagrams.

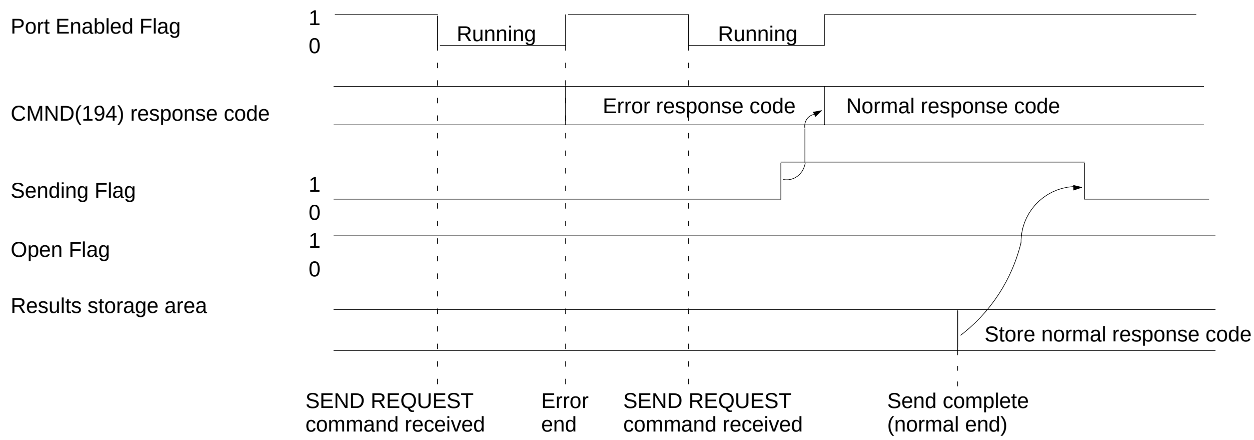
OPEN REQUEST



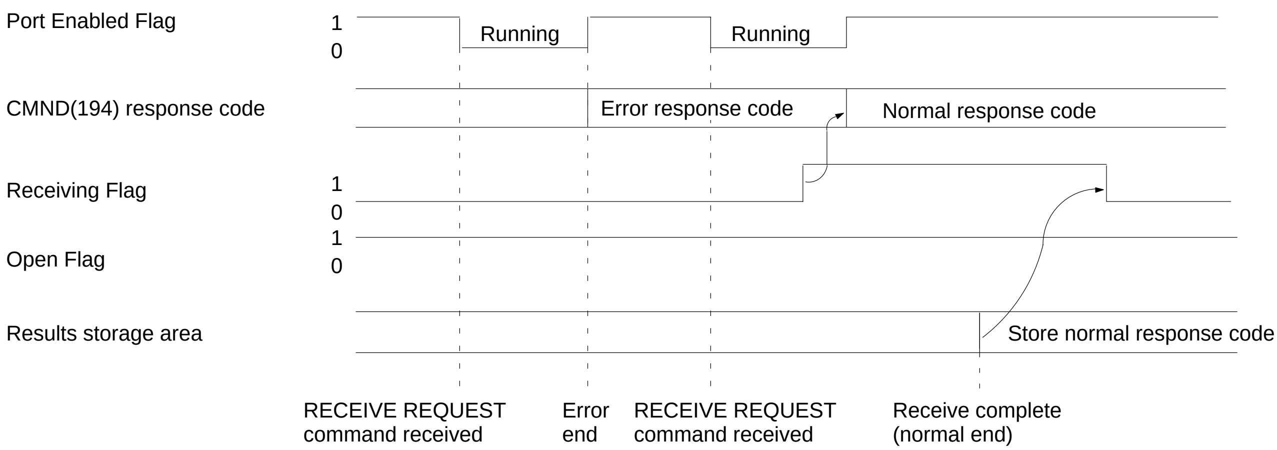
CLOSE REQUEST



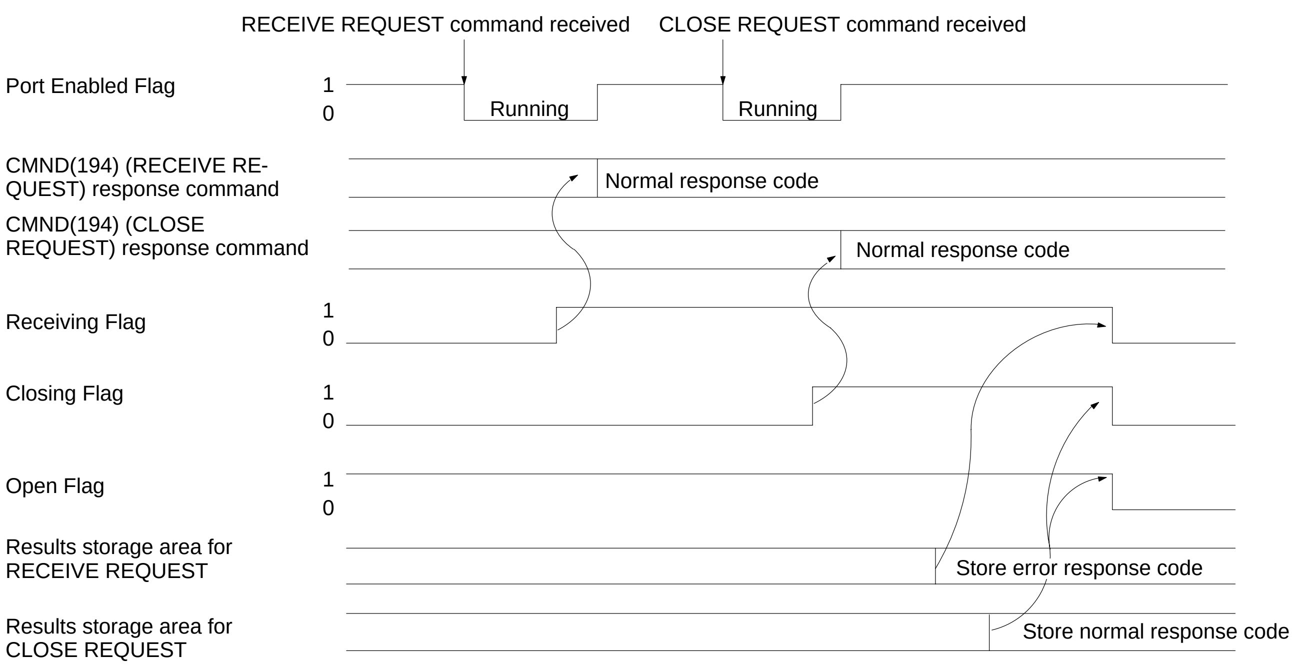
SEND REQUEST



RECEIVE REQUEST



CLOSE REQUEST during RECEIVE REQUEST



Note The timing shown in the above diagram occurs if a CLOSE REQUEST command is executed during SEND REQUEST command execution. The timing shown in the diagram also applies if a CLOSE REQUEST command is executed during OPEN REQUEST command execution, with the exception of the status of the Open Flag.

5-2-7 Precautions in Using Socket Service

UDP and TCP Socket Services

- If a short response monitor time is specified in CMND(194) control data and the Ethernet Unit is operating under a high load, a result may be stored even if the response code indicates a time-out. If this occurs, increase the monitor time specified with CMND(194).
- The socket status area (from CIO 1500) is zeroed when the PC’s operating mode is changed (e.g., from PROGRAM to RUN). The actual Ethernet Unit socket status, however, will remain unchanged after the socket status area is zeroed. To avoid this problem, define the socket status area as a hold area in the PC Setup. Refer to the CVSS operation manuals for details on settings.
- The Results Storage Error Flag will turn ON in the socket status to indicate that the specified results storage area does not exist in the PC. Correct the user program.
- Communications time may increase if multiple Ethernet Unit functions are used simultaneously or due to the contents of the user program.
- Communications efficiency may decrease due to high communications loads on the network.
- All data is flushed from the socket’s communications buffer when a socket is closed with the CLOSE REQUEST command. In some cases, the transmit data for the SEND REQUEST command issued just before the socket was closed may not be sent.
- When sockets are open, the Ethernet Unit provides 60 receive buffers of 2 Kbytes each to allow data to be received at any time. These buffers are shared by all open sockets. Socket service communications are interrupted if all these buffers become full. The user application must therefore issue RECEIVE REQUEST commands frequently enough to prevent the internal buffers from becoming full.

UDP Socket Services Only

- When data is received by a UDP socket, one receive buffer is used for each packet of data received. Therefore, if receive processing is not carried out, all receive buffers will become full after data is received 60 times, regardless of the actual amount of data received.
- The UDP socket sets a broadcast address for the destination node address to broadcast data to all nodes of the network simultaneously. The maximum length of broadcast data is 1,472 bytes. Data in multiple fragments (over 1,472 bytes for a UDP socket) cannot be broadcast.
- The UDP socket does not check the transmitted data to ensure communications reliability. To increase communication reliability, communications checks and retries must be included in the user application program.

TCP Socket Services Only

- When data is received by a TCP socket, each packet of data sent is appended to the previous packet, so that even without receive processing being carried out, a single receive buffer is used until it becomes full with 2 Kbytes of data before another buffer is used.
- In the TCP send processing, the end code (0000) is stored in the results storage area when the data is stored in the send buffer of the local node, regardless of the receive status of the remote socket. Actual data is sent to the receive buffer in the remote socket using the TCP protocol processing. Consequently, pay attention to the following two points.
    - If the TCP socket of the remote node is closed (the connection is broken) during communications, the TCP socket at the local node must also be closed. The communications results storage area can be used to check if

the connection has been broken. Close the local socket immediately after detecting that the remote TCP socket has closed. The following situations indicate that the remote socket has closed.

TCP receive results storage area: Response code = 0000 (normal), number of bytes received = 0000

TCP send results storage area: Response code = 0020 (connection with remote socket is broken)

- Even if the remote node TCP socket has been closed unilaterally, the data that needs to be sent may still be in the buffer on the local node. (The remaining data will be deleted when the local node performs TCP close processing.) To avoid the data being deleted, you must consider appropriate measures according to the application, such as sending data to permit closing from the local node, and then closing after the remote note has performed a check.

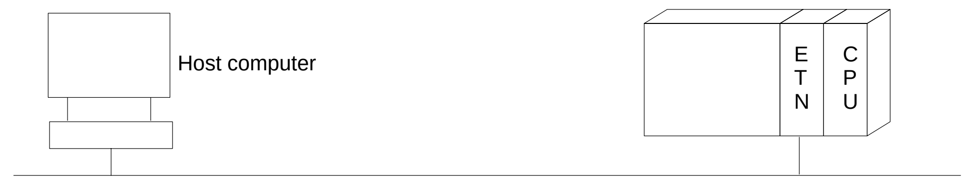
- After closing the port of a connected TCP/IP socket, the port cannot be used for 60 seconds after being closed. However, this restriction does not apply for a port opened using the TCP OPEN REQUEST (ACTIVE) command with a local TCP port number of 0 (port number automatically assigned) which is closed from the side that actively opened the socket.
- A connection is established for a passively opened socket by actively opening it from another socket. You cannot actively open multiple connections to a socket passively opened at the Ethernet Unit.
- The Ethernet Unit TCP sockets have no KEEP ALIVE function to check that the connection is normal if communications do not occur for a set time period through a communications line for which a connection has been established. The Ethernet Unit's TCP sockets make no checks to socket at the other node. Checks made by the remote node, however, are received as responses, so that it is not necessary for the user program to consider the KEEP ALIVE function.

5-2-8 Ladder Programming for TCP/IP Communications

The following program example sends and receives 100 bytes of data between the Ethernet Unit and the host computer using TCP/IP communications. The system and data area applications are described before the program example and details of program operation are described following each section of the example.

System Configuration

The system configuration for the program example and the Ethernet Unit system setup are shown below. To establish a TCP connection, the Ethernet Unit must be passively opened and the host computer actively opened.



IP address: 196.36.32.55, Port #4096

IP address: 196.36.32.101
Set using the CPU Bus Unit system setup bytes 8 to 11 (local IP address). (Refer to *2-6 Ethernet Unit System Setup* for details.)

Port: #4096
Set using the local port number to open in TCP PASSIVE open (2710 hex) using CMND(194).

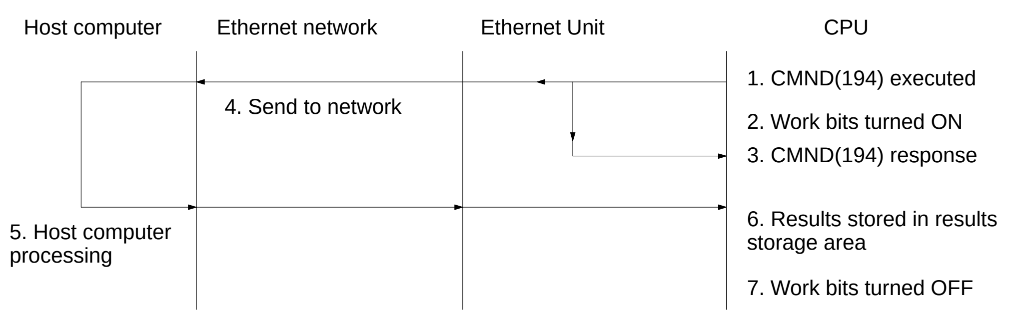
FINS Network address: 01 (hex)
You must set the routing tables in advance. (Refer to *3-5 Routing Tables* for details.)

Node number: 01 (hex)
Set using the rotary switches on the front panel of the Unit. (Refer to *2-2 Rotary Switch Settings* for details.)

Unit address: 10 (hex)
Calculate the Unit address as follows: Unit number rotary switch setting as set on the front panel of the Unit (= 0 hex) + 10 hex.

Data Flow of CMND(194) to the Ethernet Unit

The data flow to send CMND(194) to the Ethernet Unit is shown in the following diagram.



Note These work bits (CIO 000000 to CIO 000003) are set by the program to control CMND(194) execution and are not system flags, such as the Port Enabled Flags (A50200 to A50207).

Data Area Application

The data area words and bits used in the communications program are allocated as shown in the following diagrams. These words and bits are used for CMND(194) control data, command data, and results storage.

Example
The following diagrams indicates that D00000 is the first word used in CMND(194) for a TCP OPEN REQUEST and it contains 0012_{hex}, that D00001

contains 0004_{hex}, that CIO 00000 is used to control TCP open processing, and that CIO 00001 is used to control TCP close processing.

	0	1	2
D00000	TCP OPEN REQUEST via CMND(194)		
	$0012	$0004	$00

	15 to 8	1	0
CIO 00000		TCP close	TCP open

DM Area

	0	1	2	3	4	5	6	7	8	9
D00000	CMND(194) control data for TCP OPEN REQUEST (PASSIVE)									
	0012H	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Response monitor time: 5 s
Port number: Port #0
Destination Ethernet Unit designation
Network address: 01_{hex}
Node number: 01_{hex}
Unit address: 10_{hex}
Number bytes to receive: 4 bytes
Number of bytes to send: 18 (0012_{hex})

	0	1	2	3	4	5	6	7	8	9
D00010	CMND(194) control data for TCP CLOSE REQUEST									
	0008H	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Number bytes to receive: 4 bytes
Number of bytes to send: 8 bytes

	0	1	2	3	4	5	6	7	8	9
D00020	CMND(194) control data for TCP SEND REQUEST									
	006EH	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Number bytes to receive: 4 bytes
Number of bytes to send: 110 ($006E_{hex}$) bytes

Command format = 10 bytes + 100 bytes transmission data

	0	1	2	3	4	5	6	7	8	9
D00030	CMND(194) control data for TCP RECEIVE REQUEST									
	000CH	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Number bytes to receive: 4 bytes
Number of bytes to send: 12 bytes ($000C_{hex}$)

Number of bytes received specified in command data.

	0	1	2	3	4	5	6	7	8	9
D01000	TCP OPEN REQUEST (PASSIVE) command data									
	2710H	0001H	8203H	FC00H	1000H	0000H	C424H	2037H	0000H	——

Remote node: Not specified
Host computer IP address: 196.36.32.55 (C4$_{hex}$.24$_{hex}$.20$_{hex}$.37$_{hex}$)
Timeout value: Not set
Local port number: set to 4096 (1000$_{hex}$)
Results storage area: set to D01020 (03FC$_{hex}$)
(Refer to page 127 for details on the results storage area.)
TCP socket number (Ethernet Unit socket number): set to 1
Command code

	0	1	2	3	4	5	6	7	8	9
D01010	TCP OPEN REQUEST (PASSIVE) response									
	2710H	Re-sponse code	——	——	——	——	——	——	——	——

Stores the response after command execution.

	0	1	2	3	4	5	6	7	8	9
D01020	TCP OPEN REQUEST (PASSIVE) results storage area									
	Re-sponse code	Remote IP address		Remote TCP port number	——	——	——	——	——	——

	0	1	2	3	4	5	6	7	8	9
D1030	TCP CLOSE REQUEST command data									
	2714H	0001H	8204H	1A00H	——	——	——	——	——	——

Results storage area: set to D01050 (041A$_{hex}$)
TCP socket number to close: set to 1 (0001$_{hex}$)
Command code

	0	1	2	3	4	5	6	7	8	9
D01040	TCP CLOSE REQUEST response									
	2714H	Re-sponse code			——	——	——	——	——	——
D01050	TCP CLOSE REQUEST results storage area									
	Re-sponse code	——	——	——	——	——	——	——	——	——

	0	1	2	3	4	5	6	7	8	9
D02000	TCP SEND REQUEST command data									
	2713H	0001H	820BH	C200H	0064H	Send data: 100 bytes (0064$_{hex}$)				

No. of send bytes: 100 bytes (0064$_{hex}$)
Results storage area: set to D03010 (0BC2$_{hex}$)
TCP socket number
Command code

	0	1	2	3	4	5	6	7	8	9
D03000	TCP SEND REQUEST response									
	2713H	Response code	—	—	—	—	—	—	—	—
D03010	TCP SEND REQUEST results storage area									
	Response code	No. of bytes sent	—	—	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
D04000	TCP RECEIVE REQUEST command data									
	2712H	0001H	820FH	B400H	0064H	0000H	—	—	—	—

Timeout value: Not set

No. of bytes to receive: 100 bytes (0064_{hex})

Results storage area: Set to D04020 ($0FB4_{hex}$)

TCP socket number

Command code

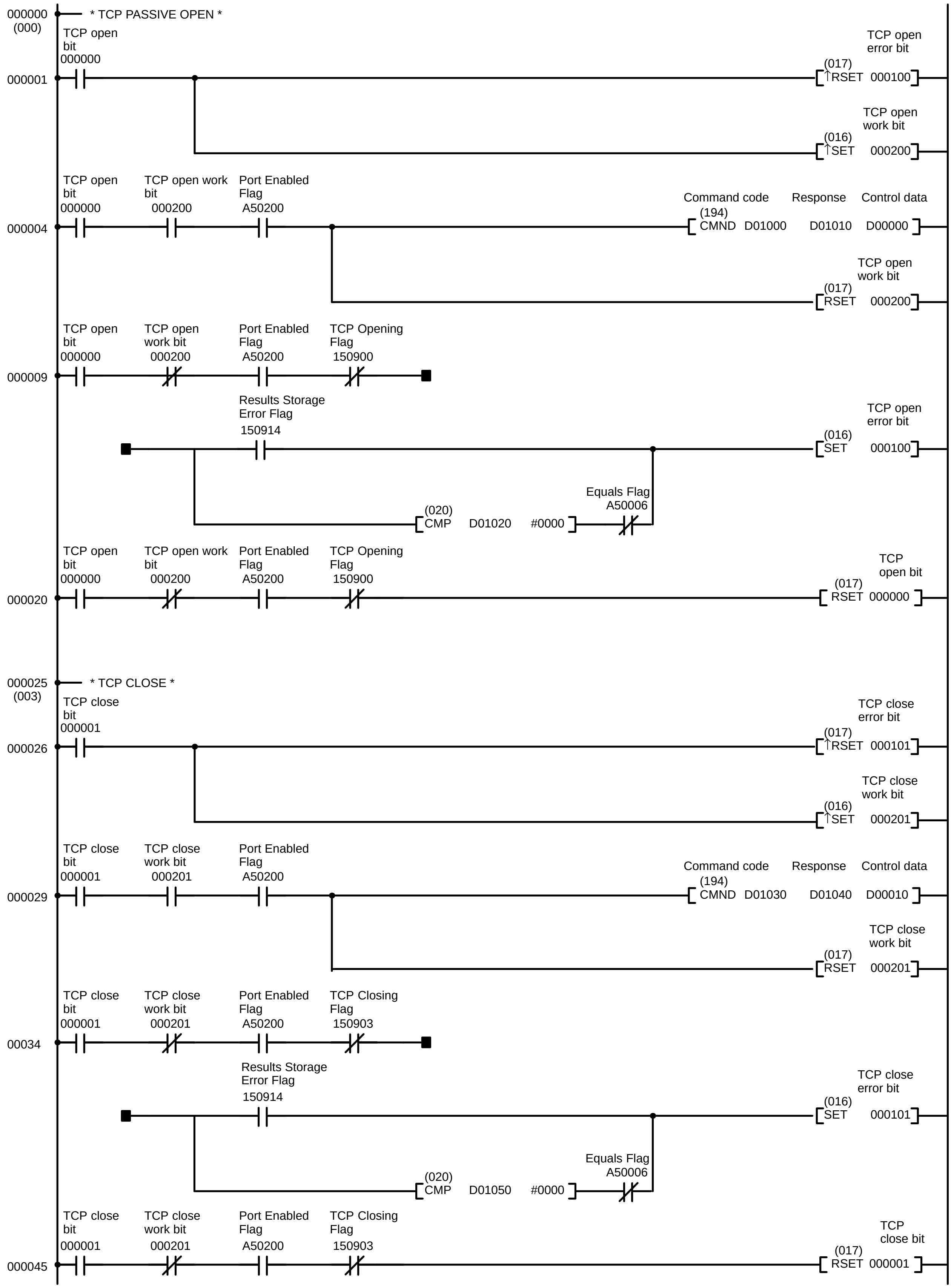
	0	1	2	3	4	5	6	7	8	9
D04010	TCP RECEIVE REQUEST response									
	2712H	Response code	—	—	—	—	—	—	—	—
D04020	TCP RECEIVE REQUEST results storage area									
	Response code	No. of bytes received	Receive data: 100 bytes (0064_{hex})							

CIO Area

The following bits are used to control CMND(194) execution. The bits in CIO 0000 are used to signal execution, the bits in CIO 0001 are used to signal errors, and the work bits in CIO 0002 are used to control the program so that the instruction is executed only once.

	15 to 8		7	6	5	4	3	2	1	0
CIO 0000							TCP receive bit	TCP send bit	TCP close bit	TCP open bit
CIO 0001							TCP receive error bit	TCP send error bit	TCP close error bit	TCP open error bit
CIO 0002							TCP receive work bit	TCP send work bit	TCP close work bit	TCP open work bit

Program Example 1



Explanation of Ladder Program

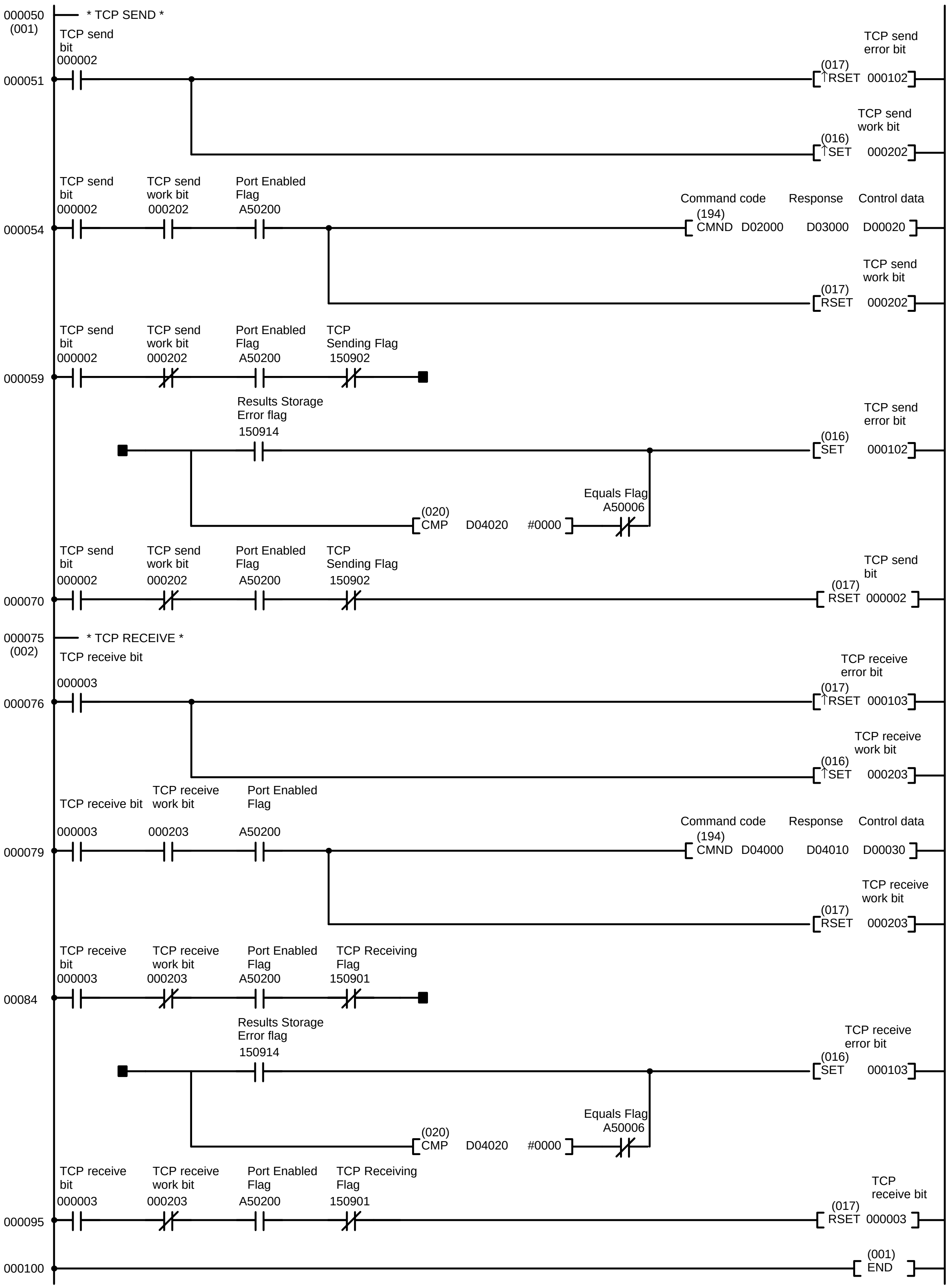
The first section of program uses CMND(94) to execute TCP PASSIVE OPEN. Program execution starts when CIO 000000 turns ON.

Program address	Explanation
000001	Starting TCP OPEN execution. When CIO 000000 turns ON, the TCP open error bit (CIO 000100) turns OFF and the TCP open work bit (CIO 000200) turns ON to request one execution of TCP OPEN.
000004	Execution of the TCP OPEN REQUEST via CMND(194) If the Port Enabled Flag (A50200) is ON while the TCP open work bit (CIO 000200) is ON, a TCP OPEN REQUEST is executed via CMND(194)) and the TCP open work bit (CIO 000200) is turned OFF. Execution pauses until CMND(194) instruction execution has been completed. A50200 is the Port Enabled Bit because the FINS communication port is port #0.
000009	Error evaluation. The Port Enabled Flag (A50200) turns ON and the TCP Opening Flag (CIO 150900) turns OFF to indicate that execution of TCP OPEN via CMND(194) has ended. The Results Storage Error Flag (CIO 150914) and the response code in the results storage area (D01020) are checked and the TCP open error bit (CIO 000100) turns ON if an error has occurred.
000020	End of TCP OPEN execution. CIO 000000 turns OFF to indicate that execution has been completed.

The next section of program uses CMND(94) to execute TCP CLOSE. Execution starts when CIO 000001 turns ON

Program address	Explanation
000026	Starting TCP CLOSE execution. When CIO 000001 turns ON, the TCP close error bit (CIO 000101) turns OFF and the TCP close bit (CIO 000201) turns ON to request one execution of TCP CLOSE.
000029	Execution of the TCP CLOSE via CMND(194) If the Port Enabled Flag (A50200) is ON while the TCP close work bit (CIO 000201) is ON, TCP CLOSE is executed via CMND(194) and the TCP close work bit (CIO 000202) turns OFF. Execution pauses until the CMND(194) instruction has been completed.
000034	Error evaluation. The Port Enabled Flag (A50200) turns ON and the TCP closing bit (CIO 150903) turns OFF to indicate that execution of TCP CLOSE via CMND(194) has ended. The Results Storage Error Flag (CIO 150914) and the response code in the results storage area (D01050) are checked and the TCP close error bit (CIO 000101) turns ON if an error has occurred.
000045	End of TCP CLOSE execution. CIO 000001 turns OFF to indicate that execution has been completed.

Program Example 2



Explanation of Ladder Program

The next section of program uses CMND(94) to execute TCP SEND. Execution starts when CIO 000002 turns ON.

Program address	Explanation
000051	Starting TCP SEND execution. When CIO 000002 turns ON, the TCP send error bit (CIO 000102) turns OFF and the TCP send bit (CIO 000202) turns ON to request one execution of TCP SEND.
000054	Execution of the TCP SEND via CMND(194). If the Port Enabled Flag (A50200) is ON while the TCP send work bit (CIO 000202) is ON, TCP SEND is executed via CMND(194) and the TCP send bit (CIO 000202) turns OFF. Execution pauses until the CMND(194) instruction has been completed.
000059	Error evaluation. The Port Enabled Flag (A50200) turns ON again and the TCP Sending Flag (CIO 150902) turns OFF to indicate that TCP SEND execution via CMND(194) has ended. The Results Storage Error Flag (CIO 150914) and the response code in the results storage area (D03010) are checked and the TCP send error bit (CIO 000102) is turned ON if an error has occurred.
000070	End of TCP SEND execution. CIO 000002 turns OFF to indicate that the instruction execution has been completed.

The next section of program uses CMND(94) to execute TCP RECEIVE. Execution starts when CIO 000003 turns ON.

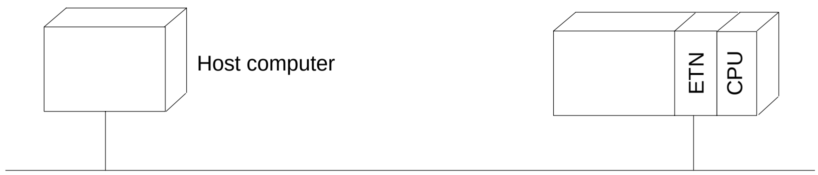
Program address	Explanation
000076	Starting TCP RECEIVE execution. When CIO 000003 turns ON, the TCP receive error bit (CIO 000103) turns OFF and the TCP receive work bit (CIO 000203) turns ON to request one execution of TCP RECEIVE.
000079	Execution of the TCP RECEIVE via CMND(194). If the Port Enabled Flag (A50200) is ON while the TCP receive work bit (CIO 000203) is ON, TCP RECEIVE is executed via CMND(194) and the TCP receive work bit (CIO 000203) turns OFF. Execution pauses until the CMND(194) instruction has been completed.
000084	Error evaluation. The Port Enabled Flag (A50200) turns ON again and the TCP Receiving Flag (CIO 150901) turns OFF to indicate that execution of TCP RECEIVE via CMND(194) has ended. The Results Storage Error Flag (CIO 150914) and the response code in the results storage area (D04020) are checked and the TCP receive error bit (CIO 000103) is turned ON if an error has occurred.
000095	End of TCP RECEIVE execution. CIO 000003 turns OFF to indicate that execution has been completed.

5-2-9 Ladder Programming for UDP/IP Communications

The following program example sends and receives 100 bytes of data between the Ethernet Unit and the host computer using UDP/IP communications. The system and data area applications are described before the program example and details of program operation are described following each section of the example.

System Configuration

The system configuration for the program example and the Ethernet Unit system setup are shown below.



IP address: 196.36.32.55
Port #4096

IP address: 196.36.32.101
Set using the CPU Bus Unit system setup bytes 8 to 11 (local IP address). (Refer to *2-6 Ethernet Unit System Setup* for details.)

Port: #4096
Set using the local port number to open in TCP PASSIVE open (2710 hex) using CMND(194).

FINS Network address: 01 (hex)
You must set the routing tables in advance. (Refer to *3-5 Routing Tables* for details.)

Node number: 01 (hex)
Set using the rotary switches on the front panel of the Unit. (Refer to *2-2 Rotary Switch Settings* for details.)

Unit address: 10 (hex)
Calculate the Unit address as follows: Unit number rotary switch setting as set on the front panel of the Unit (= 0 hex) + 10 hex.

Explanation of Operation

UDP OPEN REQUEST is executed for the Ethernet Unit when CIO 000000 turns ON. UDP CLOSE REQUEST is executed for the Ethernet Unit when CIO 000001 turns ON. UDP SEND REQUEST is executed for the Ethernet Unit when CIO 000002 turns ON and 100 bytes of data are sent starting from D02008. UDP RECEIVE REQUEST is executed for the Ethernet Unit when CIO 000003 turns ON and 100 bytes of data are received and stored starting from D04025.

The appropriate bit in CIO 000001 turns ON when an error occurs. The following areas can be used to access details about errors:

CMND(194) response codes
Response codes in results storage area
Port Enabled Flags (A50208 to A50215)
Completion codes (A503 to A510)

Data Area Application

The data area words and bits used in the communications program are allocated as shown in the following diagrams. These words and bits are used for CMND(194) control data, command data, and results storage.

DM Area

	0	1	2	3	4	5	6	7	8	9
D00000	CMND(194) control data for UDP OPEN REQUEST									
	000AH	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Response monitor time: 5 s
FINS communications port: Port #0
Destination Ethernet Unit designation
Network address: 01_{hex}
Node number: 01_{hex}
Unit address: 10_{hex}
Number bytes to receive: 4 bytes
Number of bytes to send: 10 bytes ($000A_{hex}$)

	0	1	2	3	4	5	6	7	8	9
D00010	CMND(194) control data for UDP CLOSE REQUEST									
	0008H	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Number bytes to receive: 4 bytes
Number of bytes to send: 8 bytes

	0	1	2	3	4	5	6	7	8	9
D00020	CMND(194) control data for UDP SEND REQUEST									
	0074H	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Number bytes to receive: 4 bytes
Number of bytes to send: 114 (0072_{hex}) bytes
Command format = 14 bytes + 100 bytes transmission data

	0	1	2	3	4	5	6	7	8	9
D00030	CMND(194) control data for UDP RECEIVE REQUEST									
	000CH	0004H	0001H	0110H	0000H	0032H	——	——	——	——

Number bytes to receive: 4 bytes
Number of bytes to send: 12 bytes ($000C_{hex}$)
Number of bytes received is specified in command data.

	0	1	2	3	4	5	6	7	8	9
D01000	UDP OPEN REQUEST command data									
	2701H	0001H	8203H	FC00H	1000H	——	——	——	——	——

Local port number: set to 4096 (1000_{hex})
Results storage area: set to D01020 ($03FC_{hex}$)
(Refer to page 127 for details on formats for socket services)
UDP socket number (Ethernet Unit socket number): set to 1
Command code

	0	1	2	3	4	5	6	7	8	9
D01010	UDP OPEN response									
	2701H	Response code	—	—	—	—	—	—	—	—

Stores the response after command execution.

	0	1	2	3	4	5	6	7	8	9
D01020	UDP OPEN REQUEST results storage area									
	Response code	—	—	—	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
D1030	UDP CLOSE REQUEST command data									
	2704H	0001H	8204H	1A00H	—	—	—	—	—	—

Results storage area: set to D01050 ($041A_{hex}$)

UDP socket number closed: set to 1 (0001_{hex})

Command code

	0	1	2	3	4	5	6	7	8	9
D01040	UDP CLOSE REQUEST response									
	2704H	Response code			—	—	—	—	—	—
D01050	UDP CLOSE REQUEST results storage area									
	Response code	—	—	—	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
D02000	UDP SEND REQUEST command data									
	2703H	0001H	820BH	C200H	C424H	2037H	1000H	0064H	—	—

No. of send bytes: 100 bytes (0064_{hex})

Destination port: Port #4096 (1000_{hex})

Destination address: 196.36.32.55
($C4_{hex}.24_{hex}.20_{hex}.37_{hex}$)

Results storage area: Set to D03010 ($0BC2_{hex}$)

UDP socket number

Command code

	0	1	2	3	4	5	6	7	8	9
D03000	UDP SEND REQUEST response									
	2703H	Re-sponse code	——	——	——	——	——	——	——	——
D03010	UDP SEND REQUEST results storage area									
	Re-sponse code	No. of send bytes	——							

	0	1	2	3	4	5	6	7	8	9
D04000	UDP RECEIVE REQUEST command data									
	2702H	0001H	820FH	B400H	0064H	0000H	——	——	——	——

Timeout value: Not set
No. of bytes to receive: 100 bytes (0064_{hex})
Results storage area: set to D04020 ($0FB4_{hex}$)
UDP socket number used
Command code

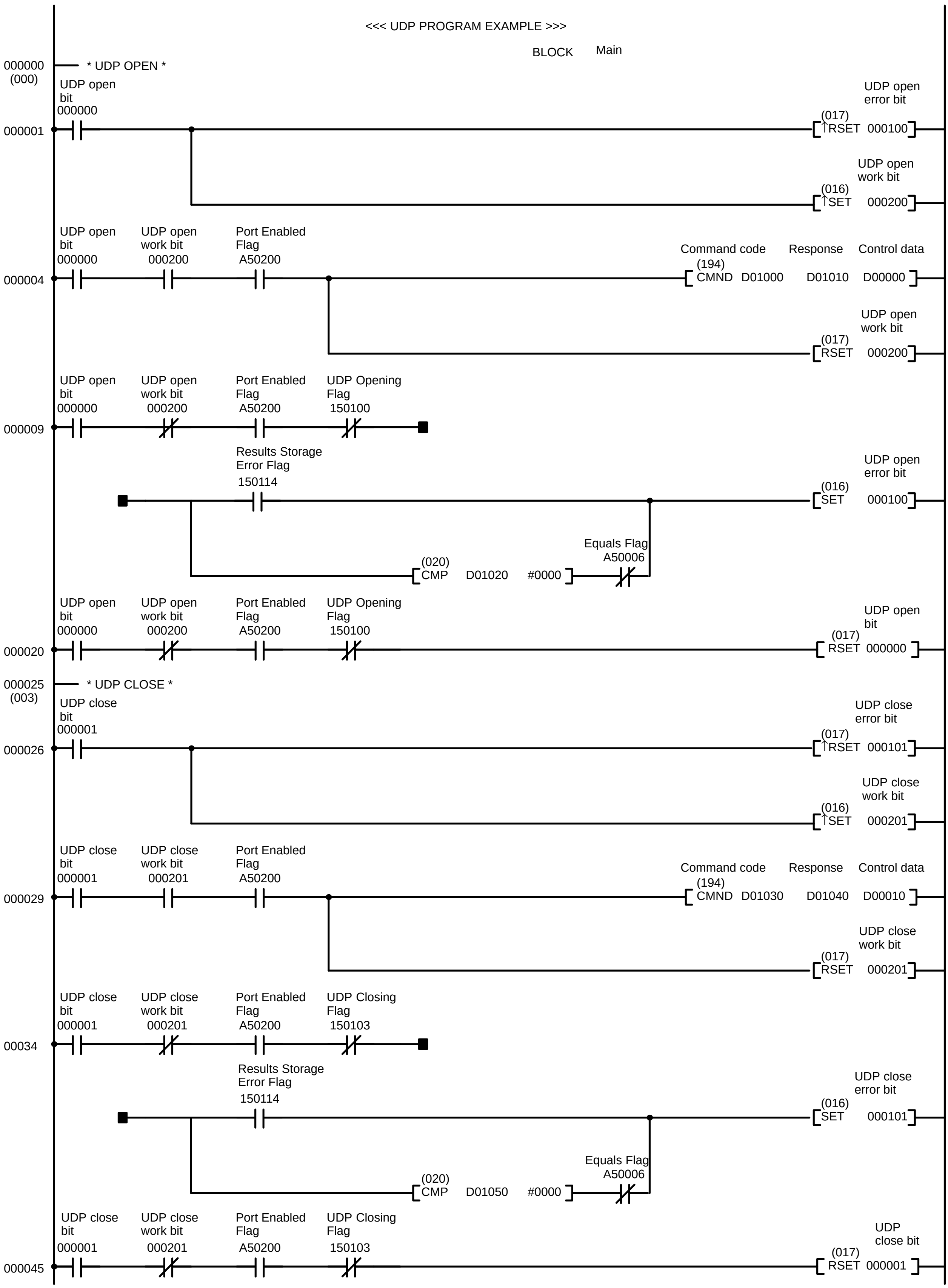
	0	1	2	3	4	5	6	7	8	9
D04010	UDP RECEIVE REQUEST response									
	2702H	Re-sponse code	——	——	——	——	——	——	——	——
D04020	UDP RECEIVE REQUEST results storage area									
	Re-sponse code	Source IP address		Source port number	No. of bytes to receive	Receive data: 100 bytes (0064_{hex})				

CIO Area

The following bits are used to control CMND(194) execution. The bits in CIO 0000 are used to signal execution, the bits in CIO 0001 are used to signal errors, and the work bits in CIO 0002 are used to control the program so that the instruction is executed only once.

	15 to 8		7	6	5	4	3	2	1	0
CIO 0000							UDP receive bit	UDP send bit	UDP close bit	UDP open bit
CIO 0001							UDP receive error bit	UDP send error bit	UDP close error bit	UDP open error bit
CIO 0002							UDP receive work bit	UDP send work bit	UDP close work bit	UDP open work bit

Program Example 1



Explanation of Ladder Program

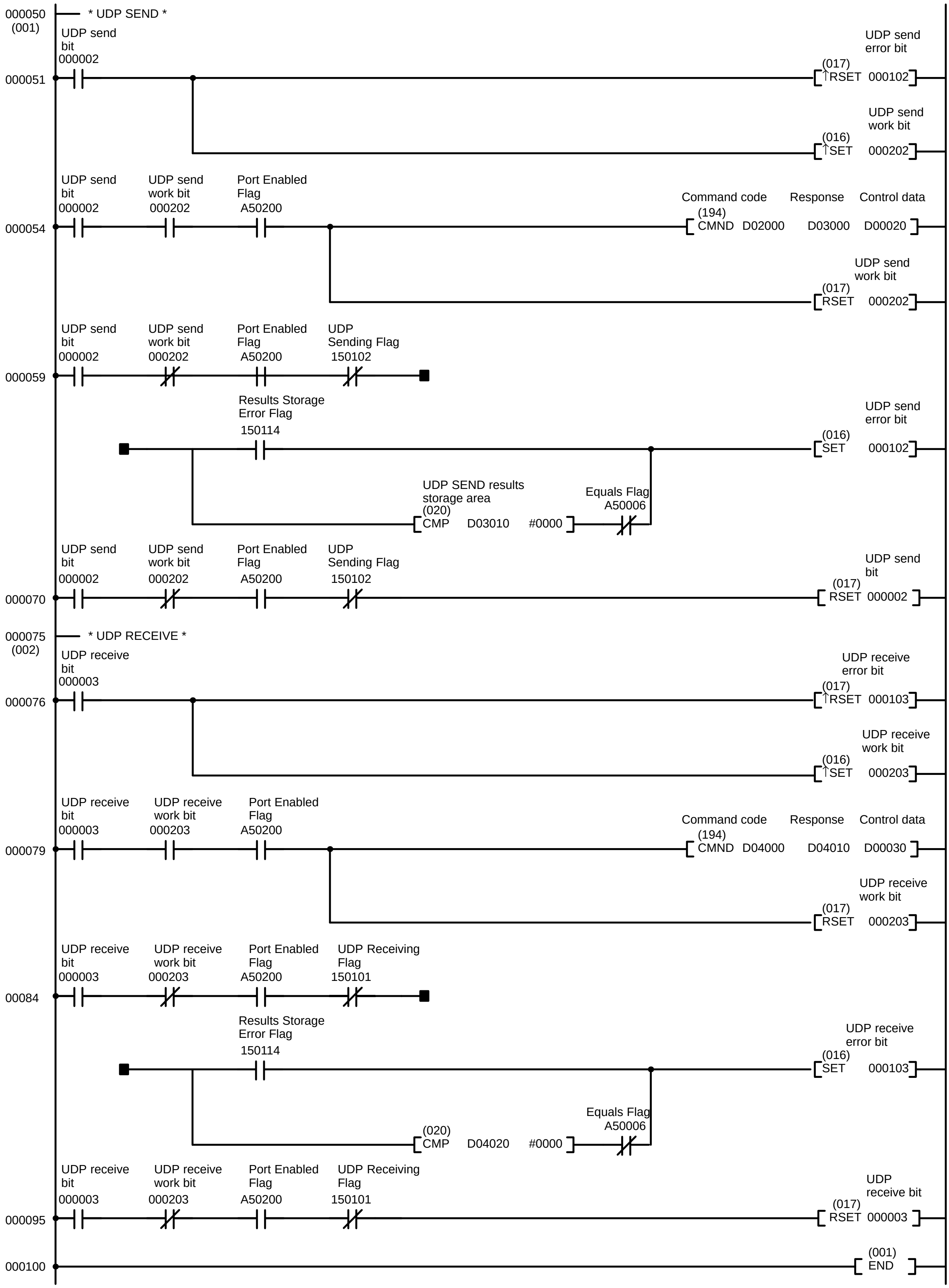
The first section of program uses CMND(94) to execute UDP OPEN. Execution starts when CIO 000000 turns ON.

Program address	Explanation
000001	Starting UDP OPEN execution. If CIO 000000 is ON, the UDP open error bit (CIO 000100) turns OFF and the UDP open work bit (CIO 000200) turns ON to request one execution of UDP OPEN.
000004	Execution of UDP OPEN via CMND(194) If the Port Enabled Flag (CIO A50200) is ON while the UDP open work bit (CIO 000200) is ON, UDP OPEN is executed via CMND(194) and the UDP open work bit (CIO 000200) turns OFF. Execution pauses until CMND(194) instruction has been completed. A50200 is the Port Enabled Bit because the FINS communication port is port #0.
000009	Error evaluation. The Port Enabled Flag (A50200) turns ON and the UDP Opening Flag (CIO 150100) turns OFF to indicate that execution of UDP OPEN via CMND(194) has ended. The Results Storage Error Flag (CIO 150114) and the response code in the results storage area (D01020) are checked and the UDP open error bit (CIO 000100) is turned ON if an error has occurred.
000020	End of UDP OPEN execution. CIO 000000 turns OFF to indicate that execution has been completed.

The next section of program uses CMND(94) to execute UDP CLOSE. Execution starts when CIO 000001 turns ON.

Program address	Explanation
000026	Starting UDP CLOSE execution. If CIO 000001 is ON, UDP close error bit (CIO 000101) turns OFF and the UDP close work bit (CIO 000201) turns ON to request one execution of UDP CLOSE.
000029	Execution of the UDP CLOSE via CMND(194). If the Port Enabled Flag (A50200) is ON while the UDP close work bit (CIO 0000201) is ON, UDP CLOSE is executed via CMND(194) and the UDP close work bit (CIO 000201) turns OFF. Execution pauses until CMND(194) instruction has been completed.
000034	Error evaluation. The Port Enabled Flag (CIO A50200) turns ON again and the UDP Closing Flag (CIO 150103) turns OFF to indicate that execution of UDP CLOSE via CMND(194) has ended. The Results Storage Error Flag (CIO 150114) and the response code in the results storage area (D01050) are checked and the UDP close error bit (CIO 000101) is turned ON if an error has occurred.
000045	End of UDP CLOSE Execution CIO 000001 turns OFF to indicate that execution has been completed.

Program Example 2



Explanation of Ladder Program

The next section of program uses CMND(94) to execute UDP SEND. Execution starts when CIO 000002 turns ON.

Program address	Explanation
000051	Starting UDP SEND execution. When CIO 000002 turns ON, the UDP send error bit (CIO 000102) turns OFF and the UDP send work bit (CIO 000202) turns ON to request one execution of UDP SEND.
000054	Execution of UDP SEND via CMND(194). If the Port Enabled Flag (A50200) is ON while the UDP send work bit (CIO 000202) is ON, UDP SEND is executed via CMND(194) and the UDP send work bit (CIO 000202) turns OFF. Execution pauses until the CMND(194) instruction has been completed.
000059	Error evaluation. The Port Enabled Flag (A50200) turns ON again and the UDP Sending Flag (CIO 150102) turns OFF to indicate that the UDP SEND via CMND(194) execution has ended. The Results Storage Error Flag (CIO 150114) and the response code in the results storage area (D03010) are checked and the UDP send error bit (CIO 000102) is turned ON if an error has occurred.
000070	End of UDP SEND execution. CIO 000002 turns OFF to indicate that execution has been completed.

The next section of program uses CMND(94) to execute UDP RECEIVE. Execution starts when CIO 000003 turns ON

Program address	Explanation
000076	Starting UDP RECEIVE execution. When CIO 000003 turns ON, the UDP receive error bit (CIO 000103) turns OFF and the UDP receive work bit (CIO 000203) turns ON to request one execution of UDP RECEIVE.
000079	Execution of UDP RECEIVE via CMND(194) If the Port Enabled Flag (A50200) is ON while the UDP receive work bit (CIO 000203) is ON, UDP RECEIVE is executed via CMND(194) and the UDP receive work bit (CIO 000203) turns OFF. Execution pauses until the CMND(194) instruction has been completed.
000084	Error evaluation. The Port Enabled Flag (A50200) turns ON again and the UDP Receiving Flag (CIO 150101) turns OFF to indicate that execution of UDP RECEIVE via CMND(194) has ended. The Results Storage Error Flag (CIO 150114) and the response code in the results storage area (D04020) are checked and the UDP receive error bit (CIO 000103) is turned ON if an error has occurred.
000095	End of UDP RECEIVE execution. CIO 000003 turns OFF to indicate that execution has been completed.

5-2-10 UDP Communications Using SFC

This section explains UDP communications programming using an SFC (sequential function chart) program. The allocation of the DM and CIO areas in memory are identical to the allocations described for the ladder programs.

Summary of Operation

Step 10 (ST0010)

Action 10 (AC0010): Initialize UDP commands

Sets the command format and control data for UDP OPEN REQUEST, UDP CLOSE REQUEST, UDP SEND REQUEST, and UDP RECEIVE REQUEST command execution via CMND(194).

Step 20 (ST0020)

Action 20 (AC0020): Dummy action

Execution reverts to this step after each process is executed. The jump destination from this step is determined by the ON/OFF status of the four bits used to start UDP process execution:

CIO 000000 ON:	UDP OPEN REQUEST routine
CIO 000001 ON:	UDP CLOSE REQUEST routine
CIO 000002 ON:	UDP SEND REQUEST routine
CIO 000003 ON:	UDP RECEIVE REQUEST routine

Step 21 (ST0021) to Step 24 (ST0024)

Action 21 (AC0021) to Action 24 (AC0024)

These actions clear the bits used to start execution of UDP requests (CIO 000000 to CIO 000003) and the error flags (CIO 000100 to CIO 000103).

Step 30 (ST0030) to Step 60 (ST0060)

Call the sub-charts for each operation:

Step 30 (ST0030):	Calls UDP OPEN REQUEST
Step 40 (ST0040):	Calls UDP CLOSE REQUEST
Step 50 (ST0050):	Calls UDP SEND REQUEST
Step 60 (ST0060):	Calls UDP RECEIVE REQUEST

Step 100 (ST0100) to Step 130 (ST0130)

The sub-charts called by Step 30 (ST0030) to Step 60 (ST0060).

Step 100 (ST0100):	Executes UDP OPEN via CMND(194)
Step 110 (ST0110):	Executes UDP CLOSE via CMND(194)
Step 120 (ST0120):	Executes UDP SEND via CMND(194)
Step 130 (ST0130):	Executes UDP RECEIVE via CMND(194)

Transition 100 (TN0100) to Transition 130 (TN0130)

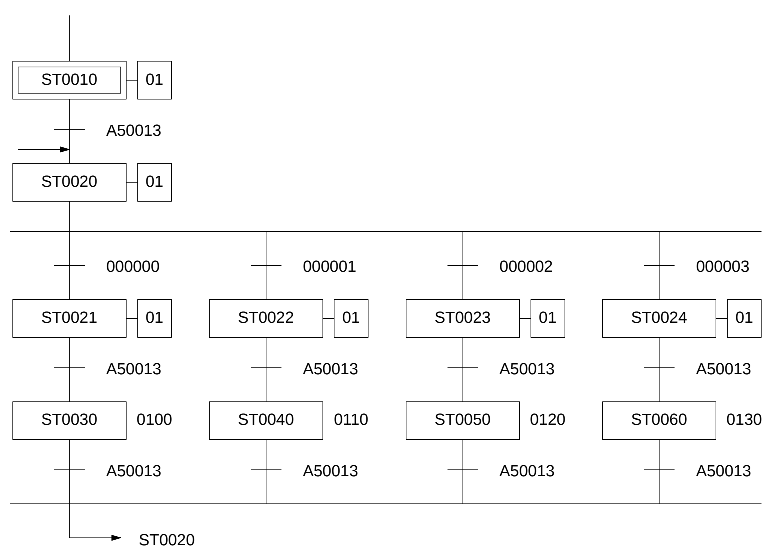
Execution pauses until execution has been completed.

Error evaluation is carried out after each of the following steps is executed: Step 101 (ST0101), Step 111 (ST0111), Step 121 (ST0121), and Step 131 (ST0131). If an error has occurred, one of the following error flags is turned ON:

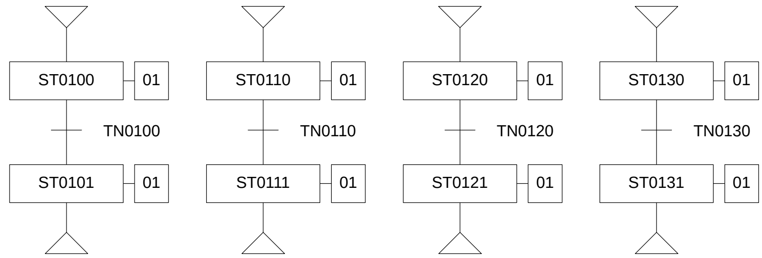
Error during UDP OPEN REQUEST:	CIO 000100 turns ON
Error during UDP CLOSE REQUEST:	CIO 000101 turns ON
Error during UDP SEND REQUEST:	CIO 000102 turns ON
Error during UDP RECEIVE REQUEST:	CIO 000103 turns ON

Program Example

SFC Chart: Main



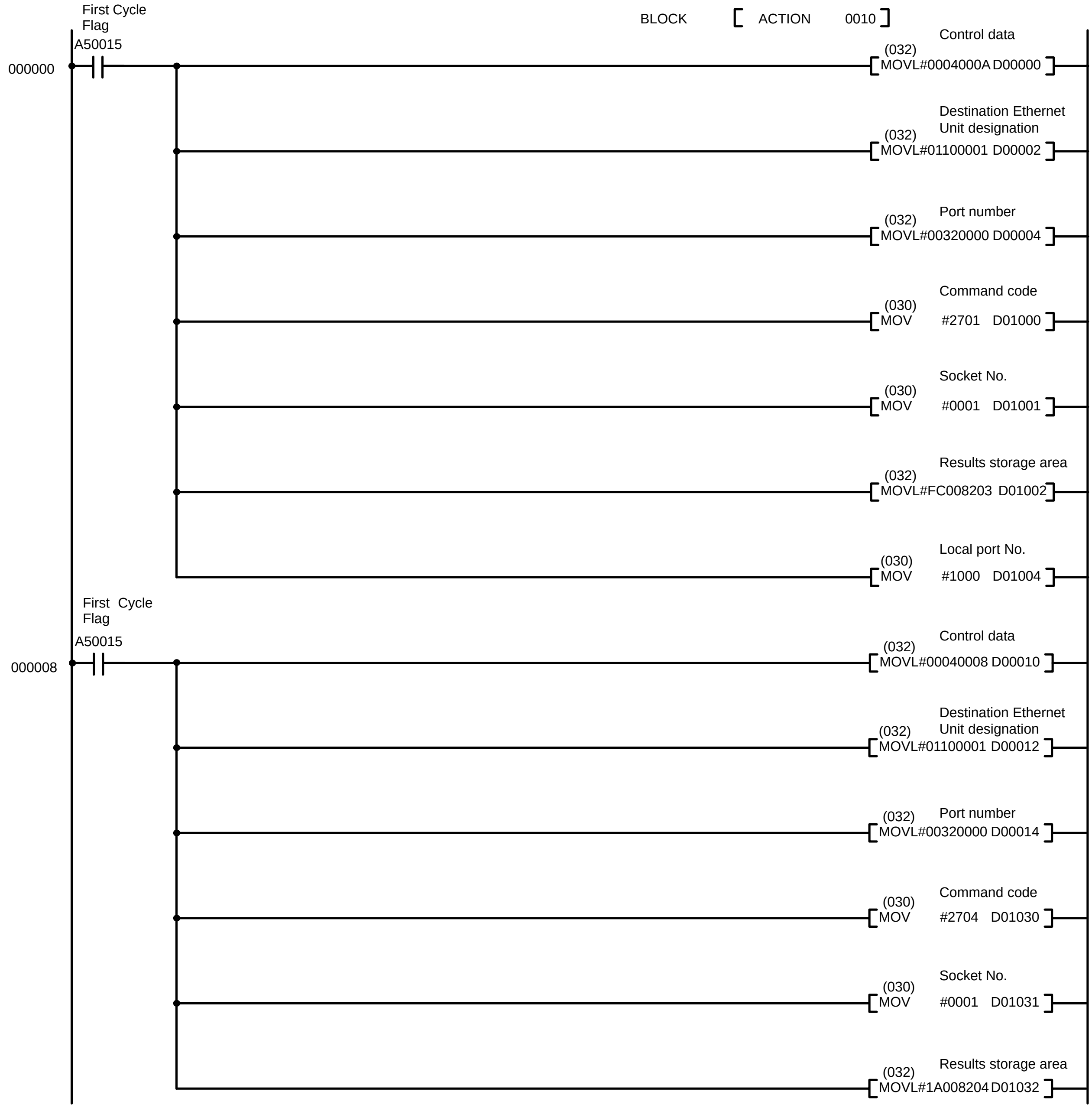
SFC Chart: Subcharts

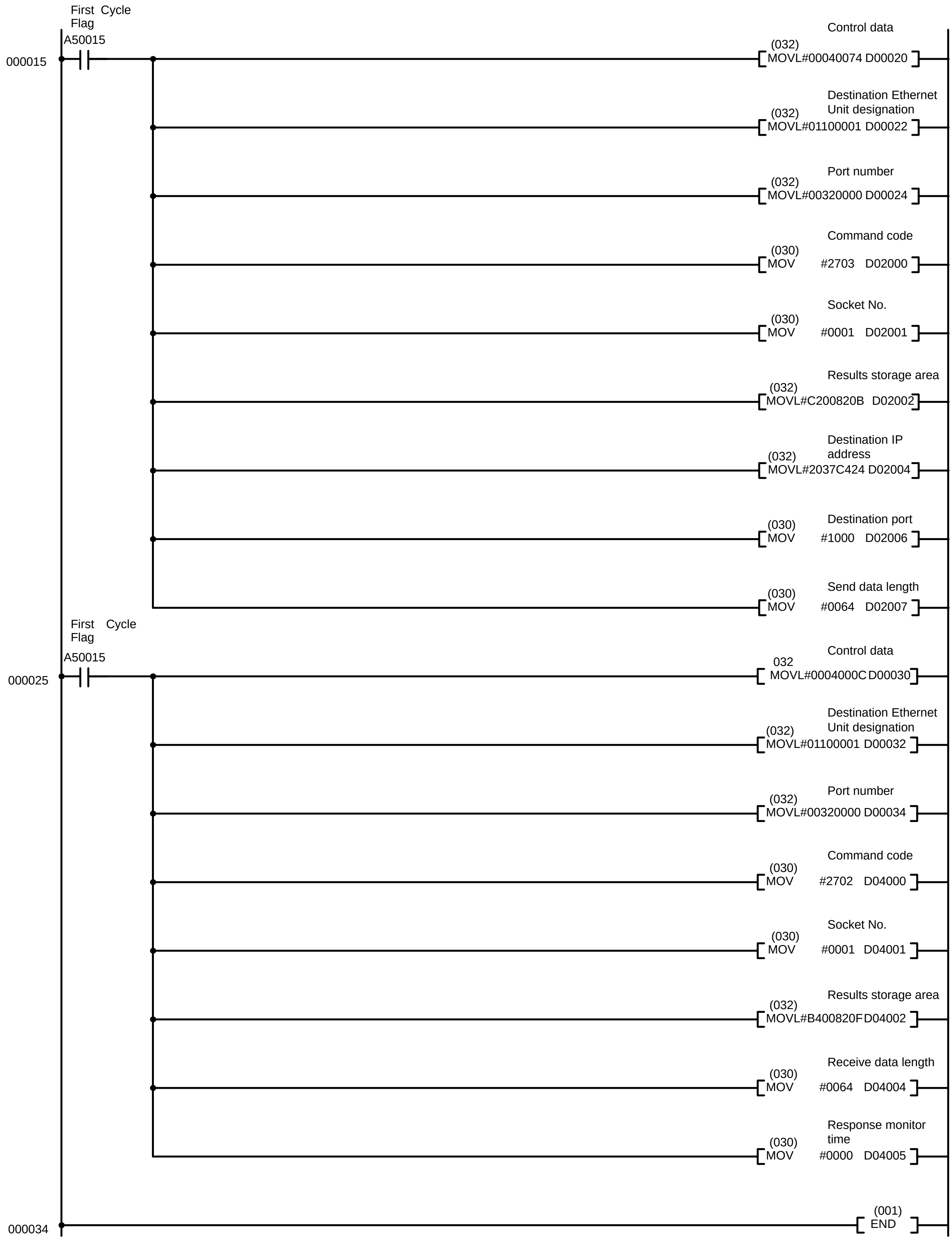


Action Blocks

	AQ	SV	Action	FV
ST0010	N		AC0010	
ST0020	Sub-chart dummy step			
ST0021	N		AC0021	
ST0022	N		AC0022	
ST0023	N		AC0023	
ST0024	N		AC0024	
ST0030	Sub-chart dummy step			
ST0040	Sub-chart dummy step			
ST0050	Sub-chart dummy step			
ST0060	Sub-chart dummy step			
ST0100	N		AC0100	
ST0110	N		AC0110	
ST0120	N		AC0120	
ST0130	N		AC0130	
ST0101	N		AC0101	
ST0111	N		AC0111	
ST0121	N		AC0121	
ST0131	N		AC0131	

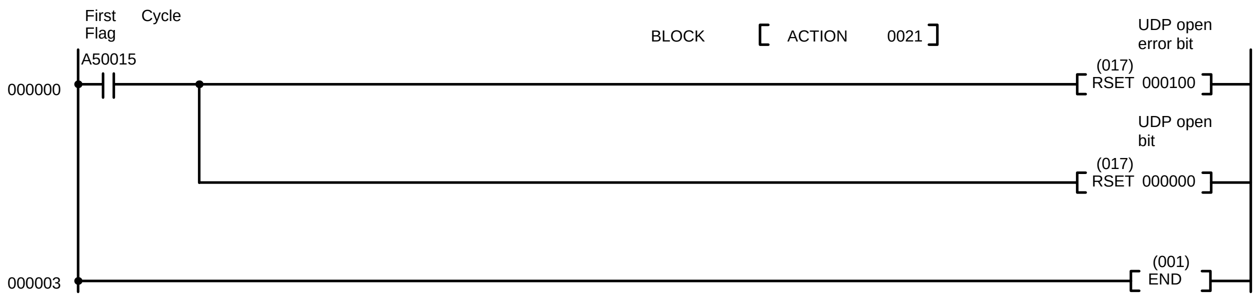
<ST0010> AC0010



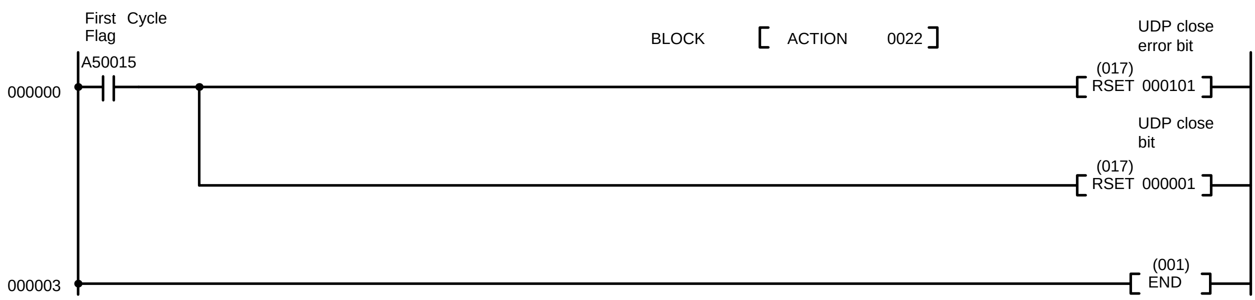

First Cycle Flag
A50015
000015
Control data
(032)
MOVL#00040074 D00020
Destination Ethernet Unit designation
(032)
MOVL#01100001 D00022
Port number
(032)
MOVL#00320000 D00024
Command code
(030)
MOV #2703 D02000
Socket No.
(030)
MOV #0001 D02001
Results storage area
(032)
MOVL#C200820B D02002
Destination IP address
(032)
MOVL#2037C424 D02004
Destination port
(030)
MOV #1000 D02006
Send data length
(030)
MOV #0064 D02007
First Cycle Flag
A50015
000025
Control data
032
MOVL#0004000C D00030
Destination Ethernet Unit designation
(032)
MOVL#01100001 D00032
Port number
(032)
MOVL#00320000 D00034
Command code
(030)
MOV #2702 D04000
Socket No.
(030)
MOV #0001 D04001
Results storage area
(032)
MOVL#B400820F D04002
Receive data length
(030)
MOV #0064 D04004
Response monitor time
(030)
MOV #0000 D04005
000034
(001)
END

<ST0020> AC0020

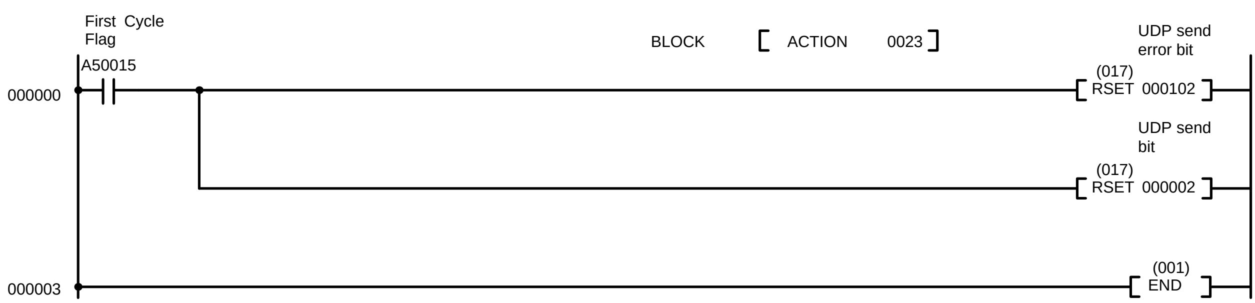
<ST0021> AC0021



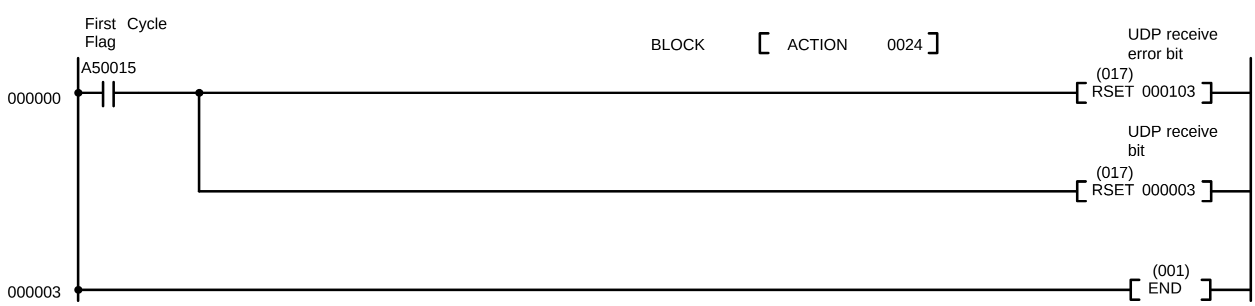
<ST0022> AC0022



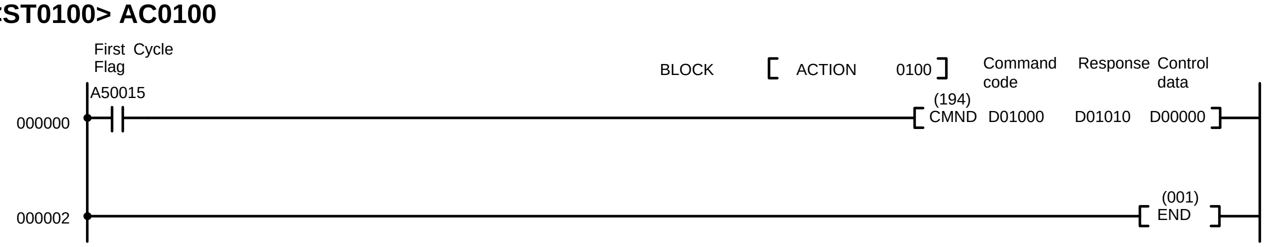
<ST0023> AC0023



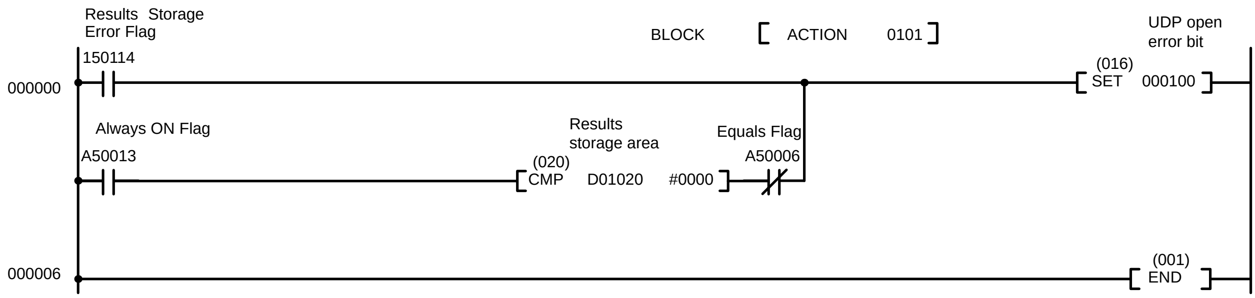
<ST0024> AC0024



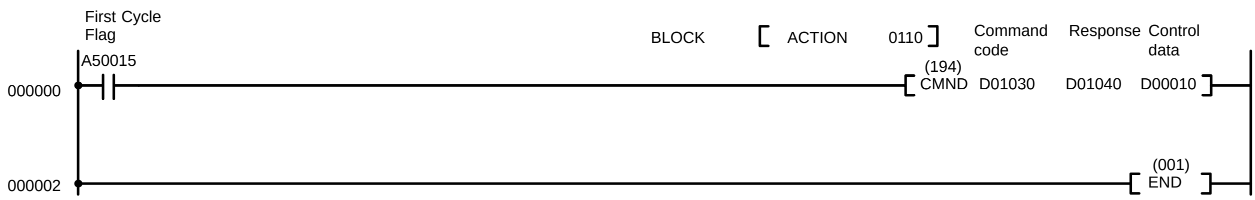
<ST0100> AC0100



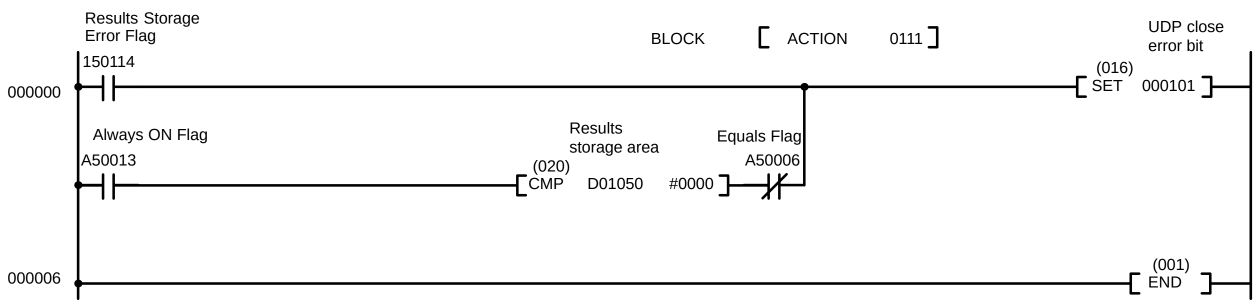
<ST0101> AC0101



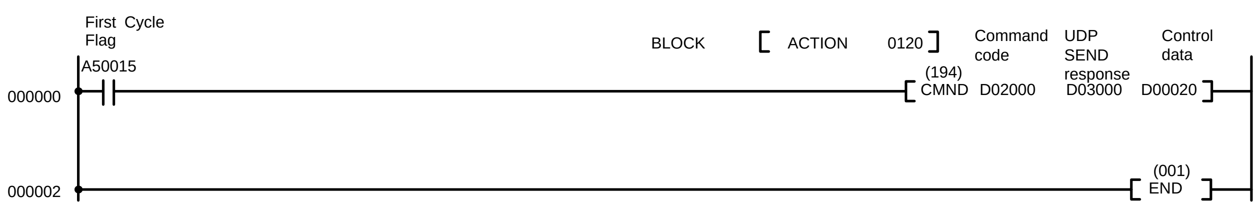
<ST0110> AC0110



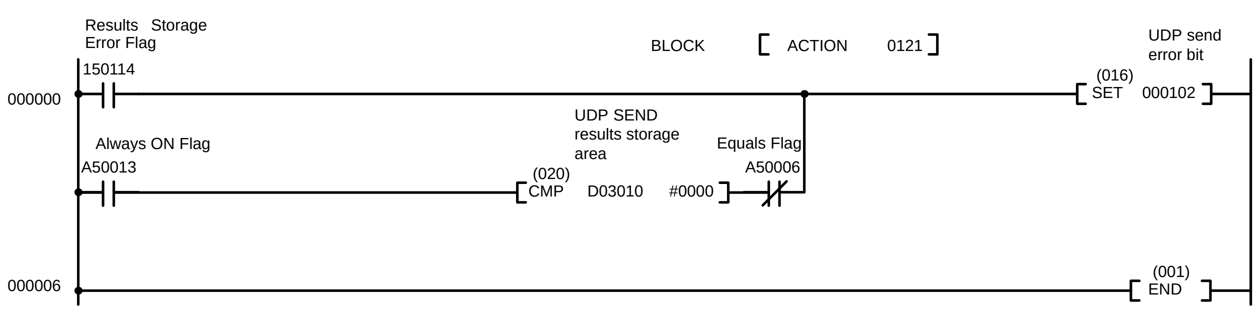
<ST0111> AC0111



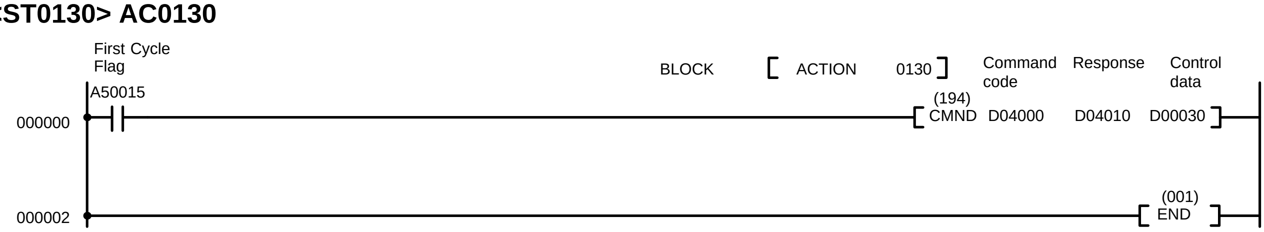
<ST0120> AC0120



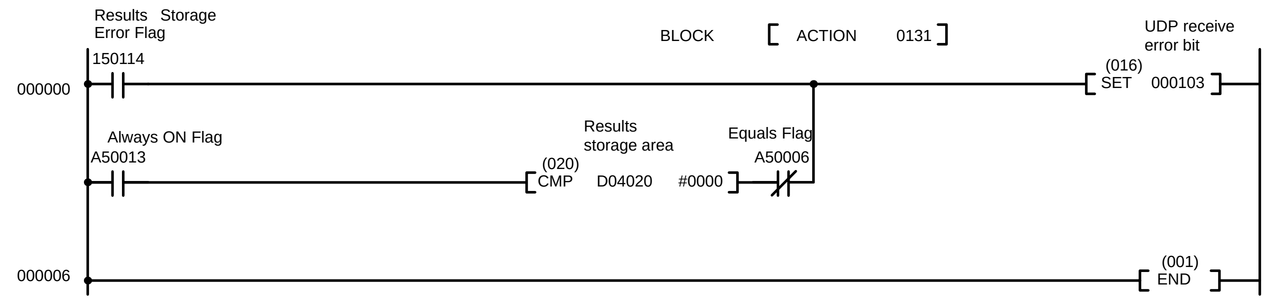
<ST0121> AC0121



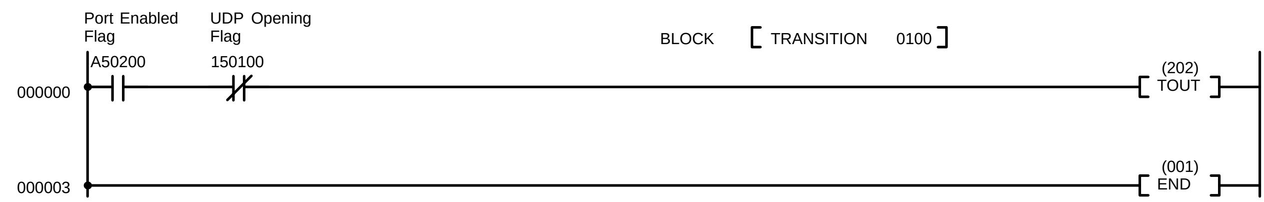
<ST0130> AC0130



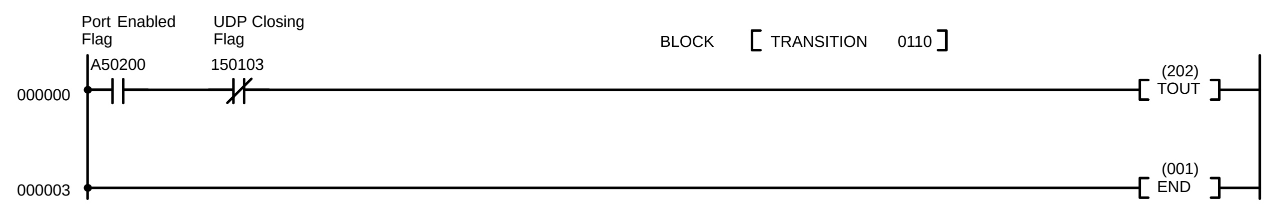
<ST0131> AC0131



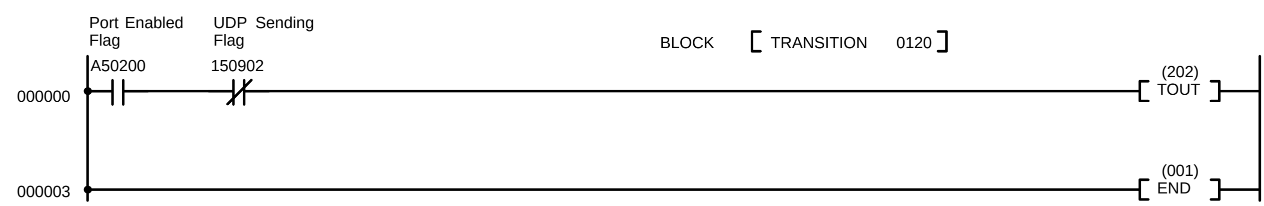
<TN0100>



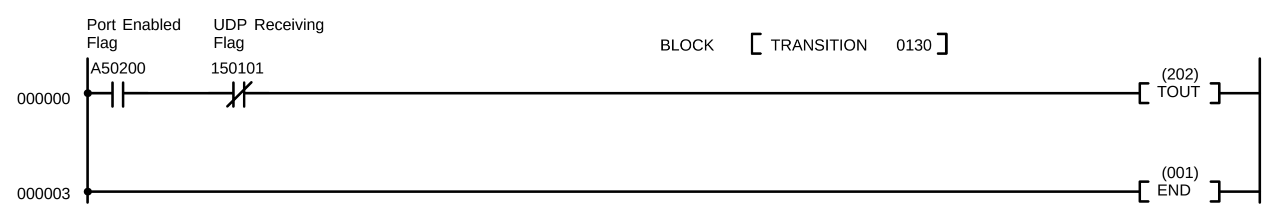
<TN0110>



<TN0120>



<TN0130>



5-2-11 TCP Communications Using SFC

This section explains TCP communications programming using an SFC (sequential function chart) program. The allocation of the DM and CIO areas in memory are identical to the allocations described for the ladder programs.

Summary of Operation

Step 10 (ST0010)

Action 10 (AC0010): Initialize for TCP commands

Sets the command format and control data for TCP OPEN REQUEST, TCP CLOSE REQUEST, TCP SEND REQUEST, and TCP RECEIVE REQUEST command execution via CMND(194).

Step 20 (ST0020)

Action 20 (AC0020): Dummy action

Execution reverts to this step after each process is executed. The jump destination from this step is determined by the ON/OFF status of the four bits used to start TCP process execution:

CIO 000000 ON:	TCP OPEN REQUEST routine
CIO 000001 ON:	TCP CLOSE REQUEST routine
CIO 000002 ON:	TCP SEND REQUEST routine
CIO 000003 ON:	TCP RECEIVE REQUEST routine

Step 21 (ST0021) to Step 24 (ST0024)

Action 21 (AC0021) to Action 24 (AC0024)

These actions clear the bits used to start execution of TCP requests (CIO 000000 to CIO 000003) and the error flags (CIO 000100 to CIO 000103).

Step 30 (ST0030) to Step 60 (ST0060)

Call the sub-charts for each operation:

Step 30 (ST0030):	Calls TCP OPEN REQUEST
Step 40 (ST0040):	Calls TCP CLOSE REQUEST
Step 50 (ST0050):	Calls TCP SEND REQUEST
Step 60 (ST0060):	Calls TCP RECEIVE REQUEST

Step 100 (ST0100) to Step 130 (ST0130)

The sub-charts called by Step 30 (ST0030) to Step 60 (ST0060).

Step 100 (ST0100):	Executes TCP OPEN REQUEST via CMND(194)
Step 110 (ST0110):	Executes TCP CLOSE REQUEST via CMND(194)
Step 120 (ST0120):	Executes TCP SEND REQUEST via CMND(194)
Step 130 (ST0130):	Executes TCP RECEIVE REQUEST via CMND(194)

Transition 100 (TN0100) to Transition 130 (TN0130)

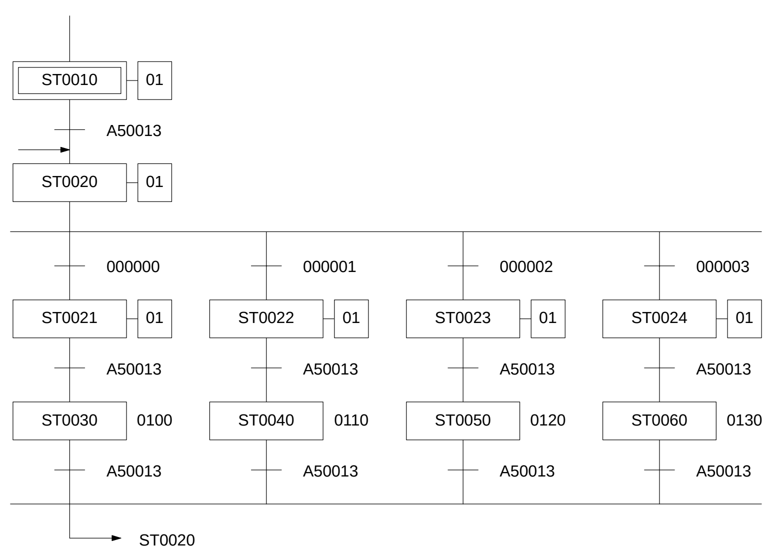
Execution pauses until instruction execution has been completed.

Error evaluation is carried out after each of the following steps is executed: Step 101 (ST0101), Step 111 (ST0111), Step 121 (ST0121), and Step 131 (ST0131). If an error has occurred, one of the following error flags is turned ON:

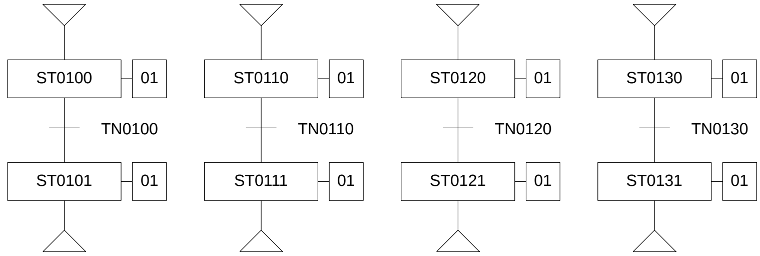
Error during TCP OPEN REQUEST:	CIO 000100 turns ON
Error during TCP CLOSE REQUEST:	CIO 000101 turns ON
Error during TCP SEND REQUEST:	CIO 000102 turns ON
Error during TCP RECEIVE REQUEST:	CIO 000103 turns ON

Program Example

SFC Chart: Main



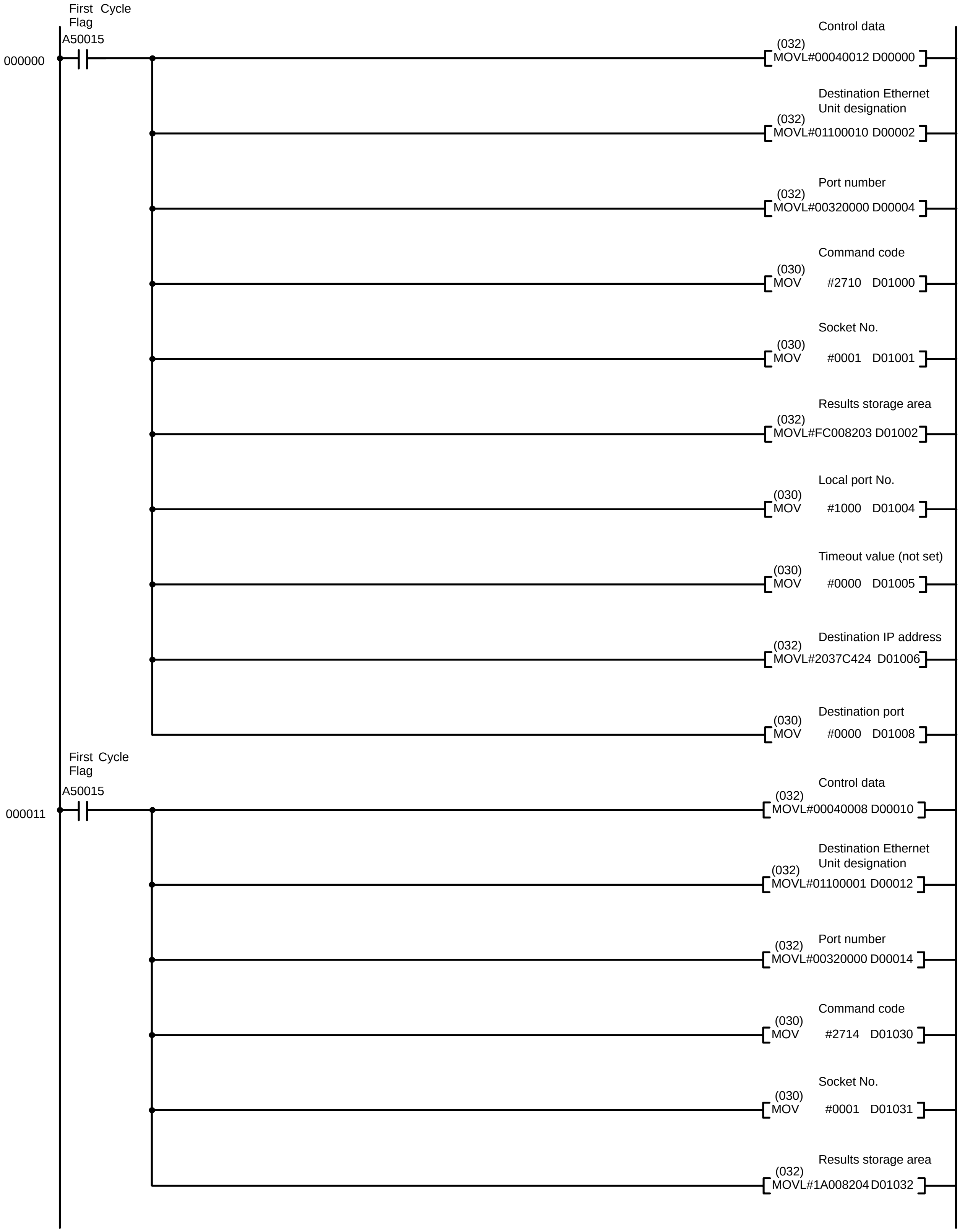
SFC Chart: Subcharts

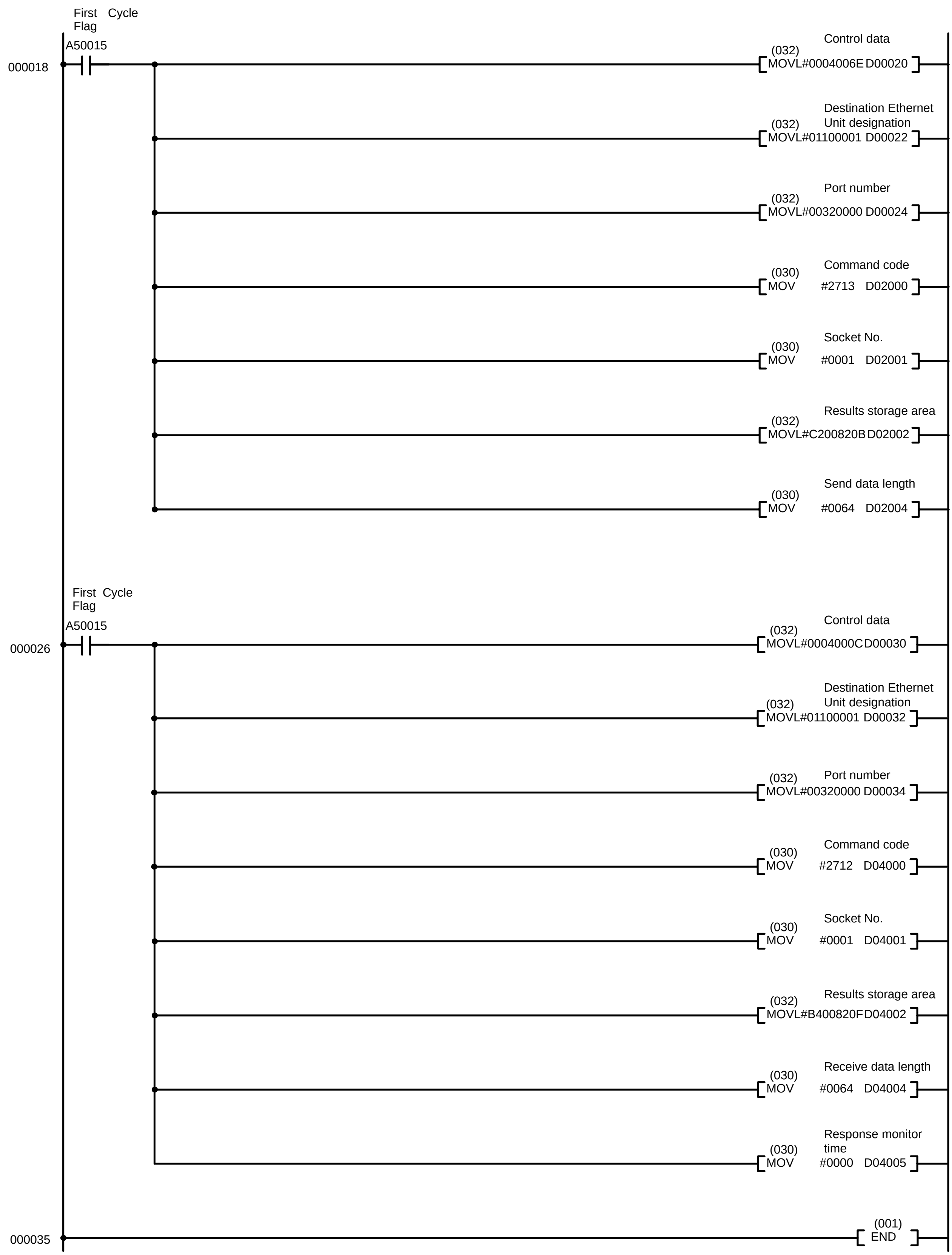


Action Blocks

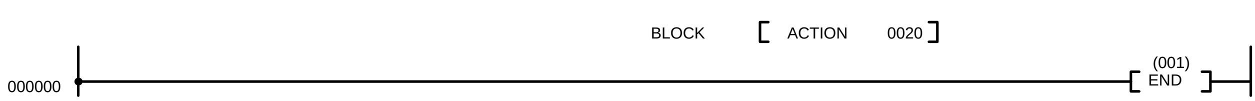
Step	AQ	SV	Action	FV
ST0010	N		AC0010	
ST0020	Sub-chart dummy step			
ST0021	N		AC0021	
ST0022	N		AC0022	
ST0023	N		AC0023	
ST0024	N		AC0024	
ST0030	Sub-chart dummy step			
ST0040	Sub-chart dummy step			
ST0050	Sub-chart dummy step			
ST0060	Sub-chart dummy step			
ST0100	N		AC0100	
ST0110	N		AC0110	
ST0120	N		AC0120	
ST0130	N		AC0130	
ST0101	N		AC0101	
ST0111	N		AC0111	
ST0121	N		AC0121	
ST0131	N		AC0131	

<ST0010> AC0010

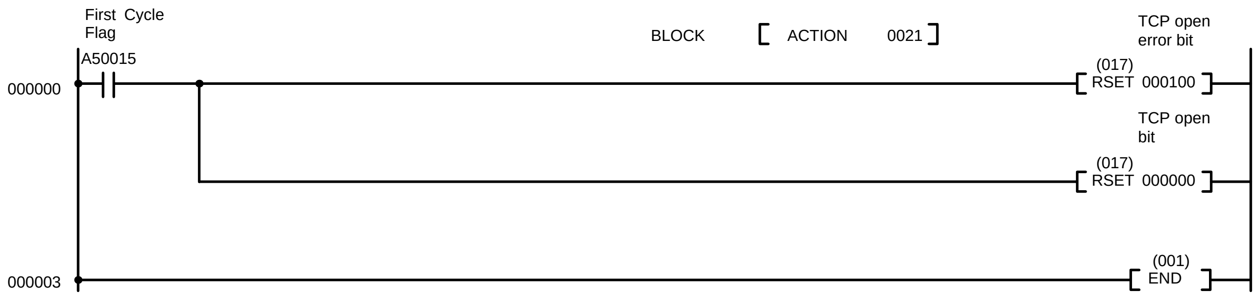




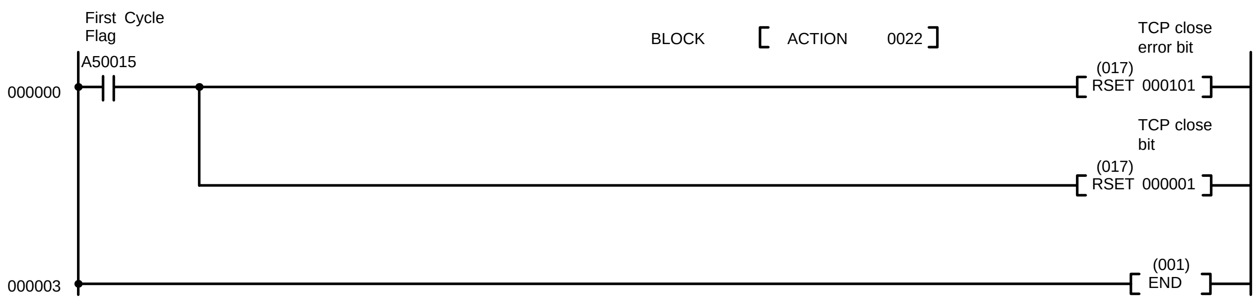
<ST0020> AC0020



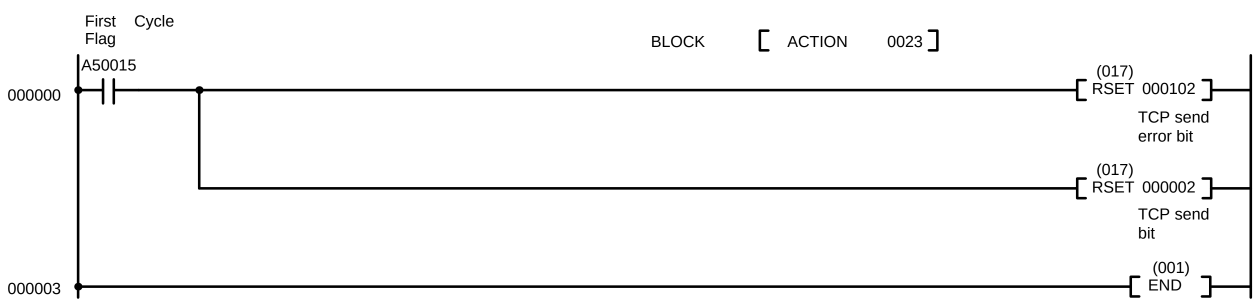
<ST0021> AC0021



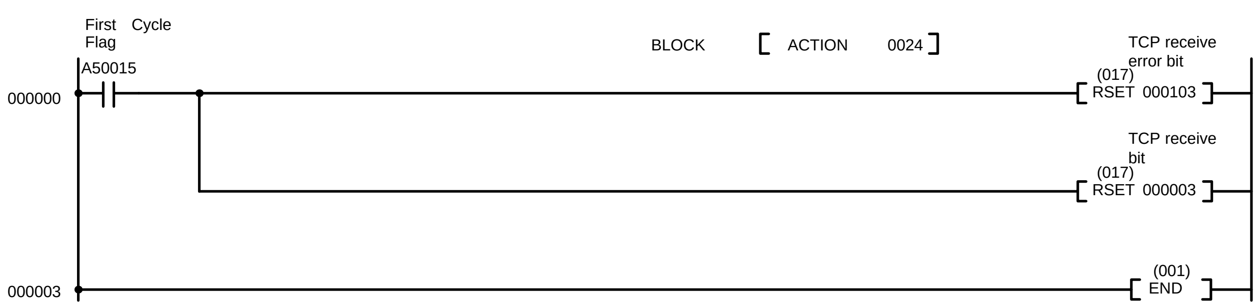
<ST0022> AC0022



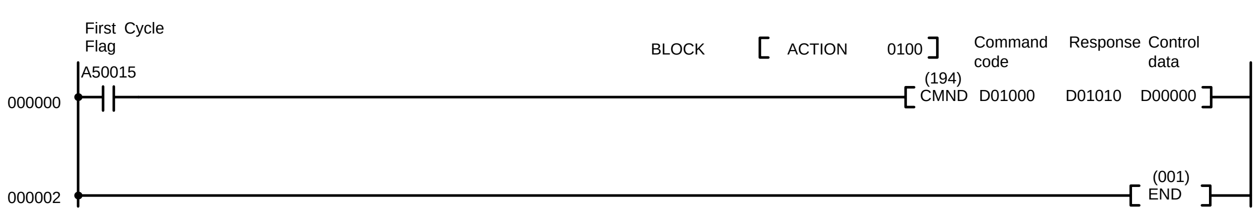
<ST0023> AC0023



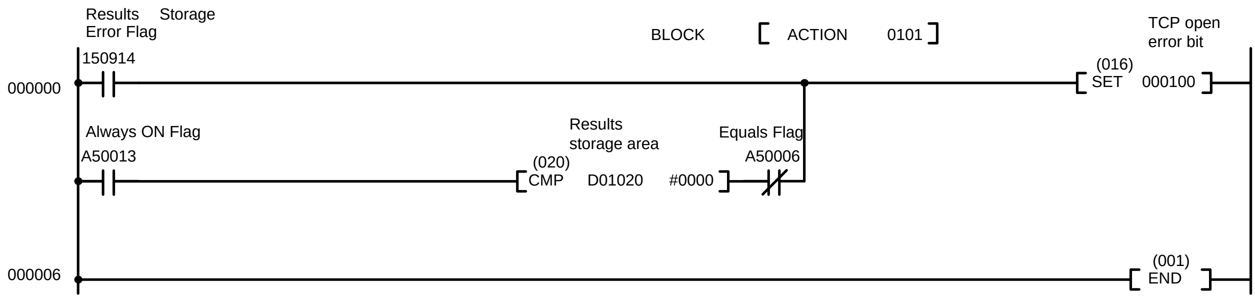
<ST0024> AC0024



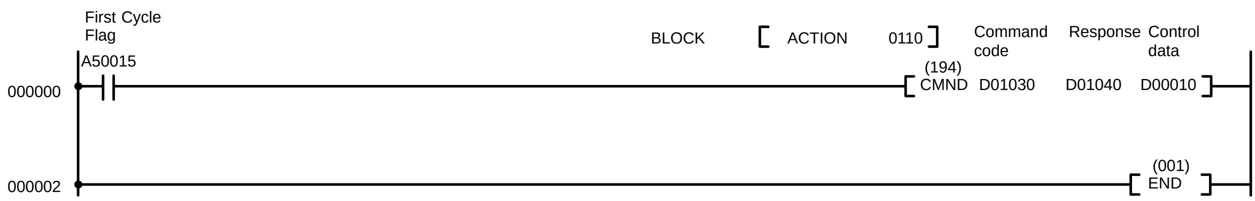
<ST0100> AC0100



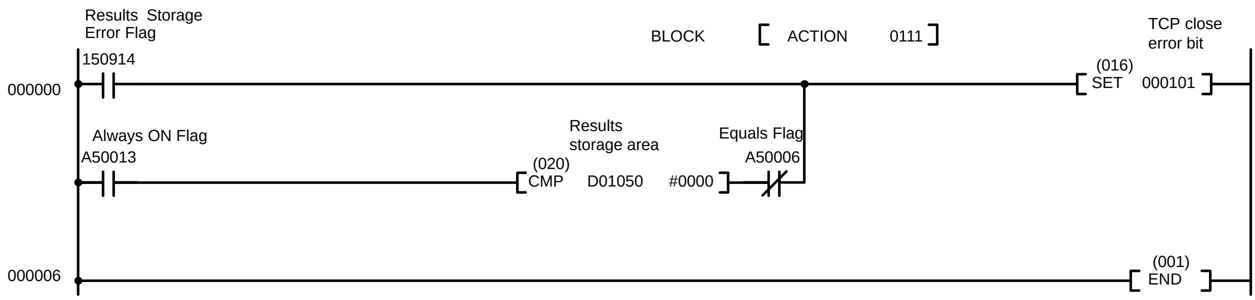
<ST0101> AC0101



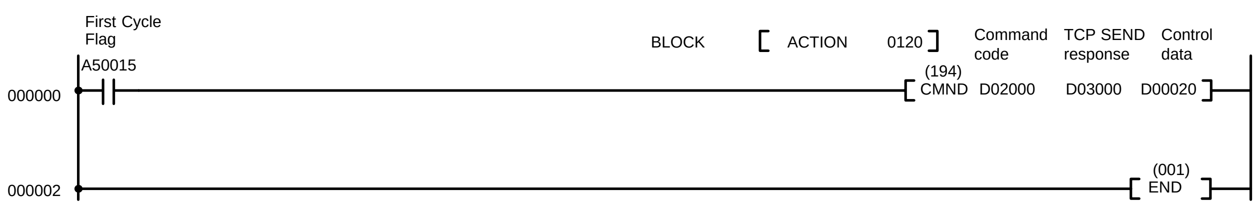
<ST0110> AC0110



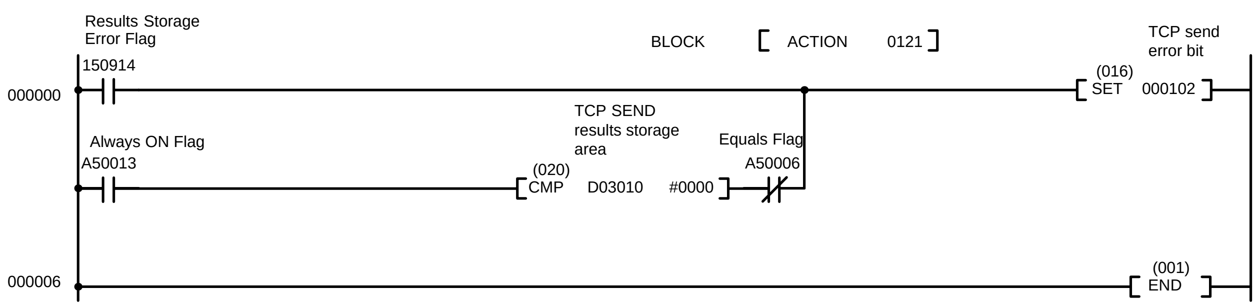
<ST0111> AC0111



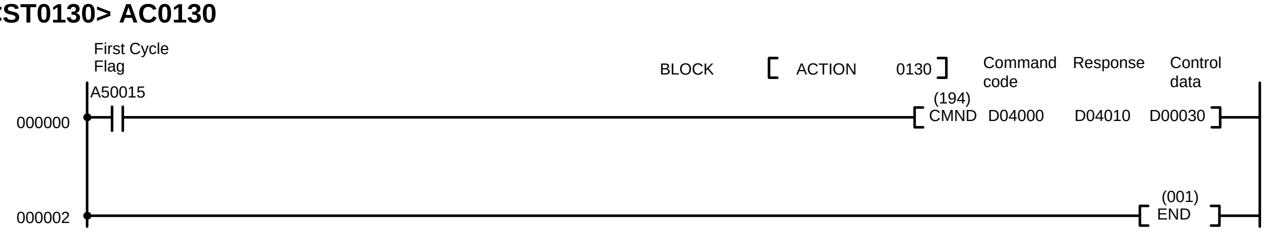
<ST0120> AC0120



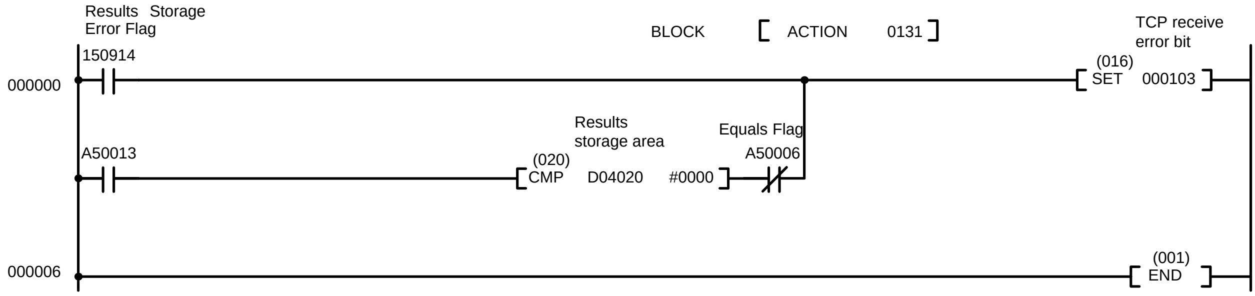
<ST0121> AC0121



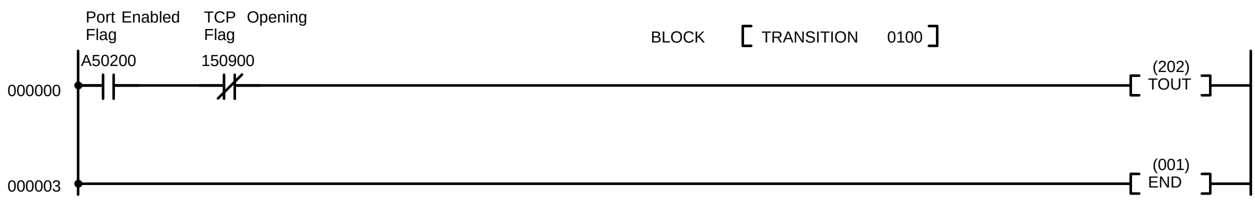
<ST0130> AC0130



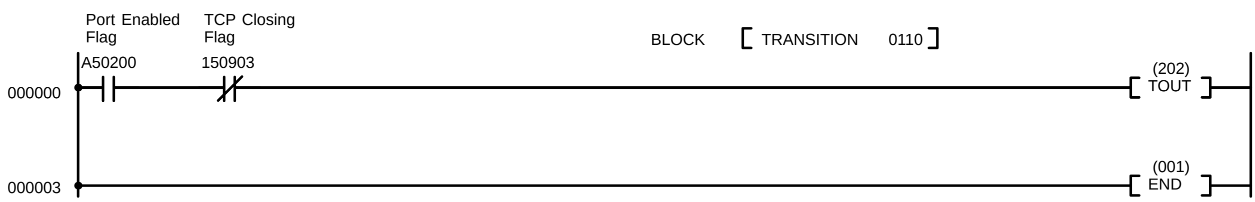
<ST0131> AC0131



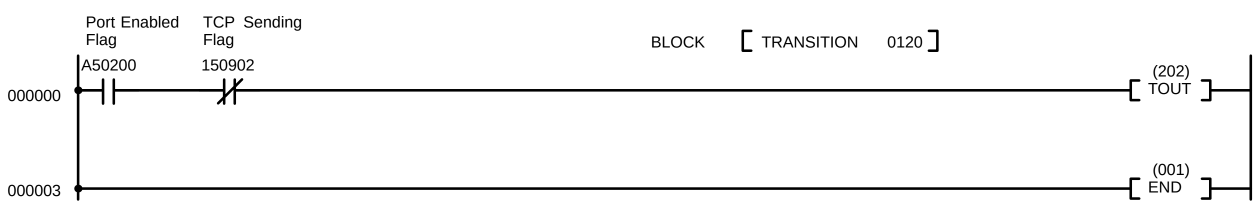
<TN0100>



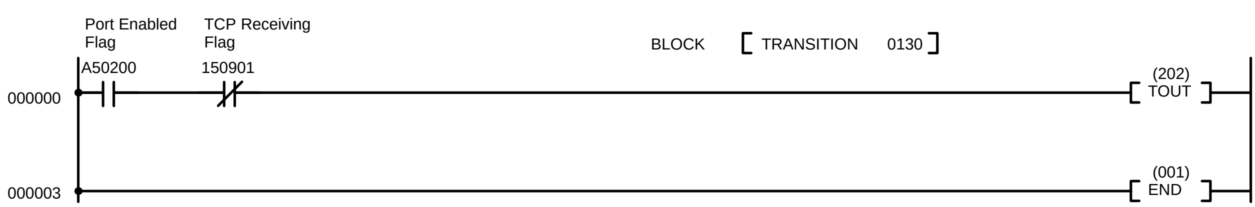
<TN0110>



<TN0120>



<TN0130>



5-2-12 Minimum Transfer Delays

The transfer delays for socket service is calculated as the sum of the communications processing times for both nodes.

Transfer delay = Destination node send processing time + Local node receive processing time + Local node send processing time + Destination node receive processing time

Calculate the minimum Ethernet Unit transfer delays for sending and receiving using the following formulas.

Note Actual delays may be longer than the calculated times due to the operating conditions.

TCP receive processing time = (CPU Bus Unit service interval x 5) + 19 + (0.030 x number of bytes received) (ms)

TCP send processing time = (CPU Bus Unit service interval x 5) + 19 + (0.020 x number of bytes sent) (ms)

UDP receive processing time = (CPU Bus Unit service interval x 5) + 19 + (0.029 x number of bytes received) (ms)

UDP send processing time = (CPU Bus Unit service interval x 5) + 19 + (0.013 x number of bytes sent) (ms)

CPU Bus Unit Service Interval

The CPU Bus Unit service interval depends on the setting in the PS Setup for CPU execution. This service interval is as follows for the two possible settings:

Synchronous: PC cycle time
Asynchronous: Peripheral device processing time

Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for details on the cycle time and instruction execution times.

Synchronous PC Execution

The actual processing time may be larger than the value calculated with the formulas above if synchronous CPU execution is set in the PC Setup. When the PC is set to for synchronous execution, replace the values underlined in the above formulas with the first integer multiple of the cycle time which exceeds the value.

Note
1. The formulas above calculate guideline figures for the transfer delays when the Ethernet Unit uses a single socket only. The user program execution time is not included in these calculations.
2. The communications time for the destination node varies according to the equipment used. If the destination node is not an Ethernet Unit, refer to the documentation for the actual equipment to calculate the communications time.
3. The transfer times may exceed the calculated values due to the operating environment. Factors affecting the transfer times are the network load ratio (amount of network traffic), the window size of each node, the load on the Ethernet Unit (including operating multiple socket services, FTP server, etc.), and the system configuration.

Example

The following table shows calculation of the approximate minimum transfer delay for sending 512 bytes of data between two PCs using a TCP socket.

Local CPU execution:	Synchronous
Local CPU cycle time:	40 ms
Destination CPU execution:	Asynchronous
Destination CPU cycle time:	Variable
Destination node peripheral device service interval:	5 ms

Item	Formula	Comments
Local receive processing time	40 x 5 + $\underline{19 + 0.030 \times 512}$ = 200 + $\underline{34.36}$ → 200 + $\underline{40}$ = 240 ms	Adjustments made for synchronous execution.
Local send processing time	40 x 5 + $\underline{19 + 0.020 \times 512}$ = 200 + $\underline{29.24}$ → 200 + $\underline{40}$ = 240 ms	Adjustments made for synchronous execution.
Destination send processing time	5 x 5 + 19 + 0.030 x 512 = 25 + 34.36 ≅ 60 ms	---
Destination receive processing time	5 x 5 + 19 + 0.020 x 512 = 25 + 29.24 ≅ 55 ms	---
Minimum transfer delay	240 + 240 + 60 + 55 = **595 ms**	---

SECTION 6
FINS Commands to Ethernet Units

This section explains FINS commands that can be sent to Ethernet Units. For information on FINS commands that can be sent to PCs or to other Units that support FINS commands, refer to the *FINS Command Reference Manual*.

6-1 Command Codes and Response Codes

6-1-1 Command Code List

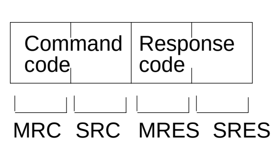
The command codes listed in the following table can be sent to an Ethernet Unit.

Command code		Name	Page
MRC	SRC		
04	03	RESET	129
05	01	CONTROLLER DATA READ	129
06	01	CONTROLLER STATUS READ	130
08	01	INTERNODE LOOPBACK TEST	132
	02	BROADCAST TEST RESULTS READ	133
	03	BROADCAST DATA SEND	133
21	02	ERROR LOG READ	133
	03	ERROR LOG CLEAR	134
27	01	UDP OPEN REQUEST	135
	02	UDP RECEIVE REQUEST	136
	03	UDP SEND REQUEST	137
	04	UDP CLOSE REQUEST	138
	10	TCP OPEN REQUEST (PASSIVE)	139
	11	TCP OPEN REQUEST (ACTIVE)	140
	12	TCP RECEIVE REQUEST	142
	13	TCP SEND REQUEST	143
	14	TCP CLOSE REQUEST	145
	20	PING	146
	60	IP ADDRESS TABLE READ	146
	61	IP ROUTER TABLE READ	147
	62	PROTOCOL STATUS READ	148
	63	MEMORY STATUS READ	153
	64	SOCKET STATUS READ	154

6-1-2 Response Code List

Response codes are 2-byte codes which indicate the results of command execution. They are returned in the response following the command code.

The first byte of a response code is the MRES (main response code), which categorizes the results of command execution. The second byte is the SRES (sub-response code) which specifies the results.



MRC: Main request code
SRC: Sub-request code
MRES: Main response code
SRES: Sub-response code

The MRES codes are shown in the following table along with the results they indicate. Refer to *Section 9 Troubleshooting* for details on response codes including the SRES.

MRES	Execution results
00	Normal completion
01	Local node error
02	Remote node error
03	Unit error (controller error)
04	Service not supported
05	Routing error
10	Command format error
11	Parameter error
22	Status error
23	Operating environment error
25	Unit error

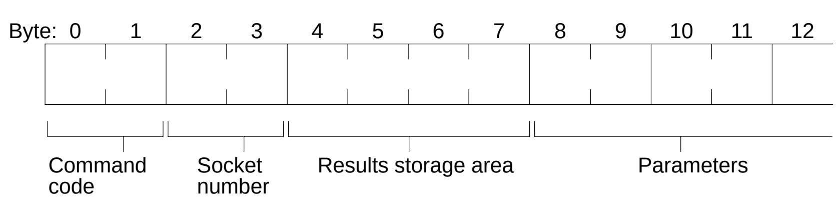
6-2 Socket Applications

The format of the following FINS commands partially differs when the sockets are used.

Command code		Name	Page
MRC	SRC		
27	01	UDP OPEN REQUEST	135
	02	UDP RECEIVE REQUEST	136
	03	UDP SEND REQUEST	137
	04	UDP CLOSE REQUEST	138
	10	TCP OPEN REQUEST (PASSIVE)	139
	11	TCP OPEN REQUEST (ACTIVE)	140
	12	TCP RECEIVE REQUEST	142
	13	TCP SEND REQUEST	143
	14	TCP CLOSE REQUEST	145

6-2-1 Format

The basic format of these commands is shown in the diagram below.



Command Code Specifies the requested process.

Socket Number Specifies the socket number for which the process is requested, from 1 to 8.

Results Storage Area Specifies the area to store the results of the requested process.

Parameters Specifies the parameters for the command code. Parameters depend on the command being executed; for details, refer to the following pages.

6-2-2 PC Memory Areas

The memory areas of the PC that can be specified for results storage when executing commands from the PC are listed in the table below. The *Variable type* is set in the first byte of the results storage area. The remaining three bytes contain the address for communications.

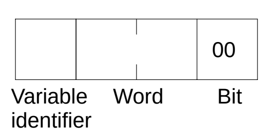
Addresses in the *Addresses for communications* column are not the same as the actual memory addresses.

Memory area	Data type		Word addresses	Addresses for communications	Variable type	Bytes
Data areas	Current value of word	CIO	CIO 0000 to CIO 2555	000000 to 09FB00	80	2
		G	G000 to G255	0A0000 to 0AFF00		
		A	A000 to A511	0B0000 to 0CFF00		
DM area		DM	D00000 to D24575	000000 to 5FFF00	82	2
Expansion DM area		Bank 0	E00000 to E32765	000000 to 7FFD00	90	2
		Bank 1	E00000 to E32765	000000 to 7FFD00	91	
		Bank 2	E00000 to E32765	000000 to 7FFD00	92	
		Bank 3	E00000 to E32765	000000 to 7FFD00	93	
		Bank 4	E00000 to E32765	000000 to 7FFD00	94	
		Bank 5	E00000 to E32765	000000 to 7FFD00	95	
		Bank 6	E00000 to E32765	000000 to 7FFD00	96	
		Bank 7	E00000 to E32765	000000 to 7FFD00	97	
		Current bank	E00000 to E32765	000000 to 7FFD00	98	2

Note The addresses in the above table are for the CV1000 and CVM1-CPU11. Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for address ranges for other PCs.

Word and Bit Addresses

Three bytes of data are used to express data memory addresses of PCs. The most significant two bytes give the word address and the least significant byte gives the bit number between 00 and 15. The word address combined with the bit number expresses the bit address. The bit number is always 00 because Ethernet Units can handle only word data, i.e., individual bits cannot be addressed.



Word addresses for specific memory area words can be calculated by converting the normal decimal word address to hexadecimal and adding it to the first word in the *Addresses for communications* column in the above table. For example, the address for communications for D00200 would be 0000 (from above table) plus C8 (decimal 200 converted to hexadecimal), or 00C8.

6-3 Command/Response Reference

This section describes the FINS commands that can be sent to Ethernet Units and the responses to each command.

The command, response, and (where applicable) the results storage blocks are given with the commands in graphic form as shown in the following diagram. If the data is fixed, it is included in the blocks. If the data is variable, it is described following the blocks. Each box represents 1 byte; ever two boxes; 1 word. The following diagram shows 2 bytes, or 1 word.

Two bytes

Response codes applicable to the command are described at the end of the command description. If any Unix error codes are generated, these are also described. Refer to your Unix error symbol definition file /usr/include/sys/errno.h for details. Unix errors are returned in the results storage area.

6-3-1 RESET

Reset the Ethernet Unit.

Command Block

04	03
Command code	

Response Block

04	03		
Command code		Response code	

Precautions

No response will be returned if the command ends normally. A response will be returned only if an error occurs.

In some cases, send requests (SEND(192)/RECV(193) instructions) made from the PC to the Ethernet Unit just before execution of the RESET command may not be executed.

All sockets (for FINS communications or FTP socket services) are closed immediately before resetting.

Response Codes

Response code	Description
1001	Command too large

6-3-2 CONTROLLER DATA READ

Reads the following data from the Ethernet model, version, IP address, subnet mask, FINS UDP port number, mode settings, Ethernet address.

Command Block

05	01
Command code	

Response Block

05	01			20 bytes	20 bytes	4 bytes	4 bytes					6 bytes
Command code		Response code		Model	Version	IP address	Subnet mask	FINS UDP port number		Mode setting		Ethernet address

Parameters

Model, Version (Response)

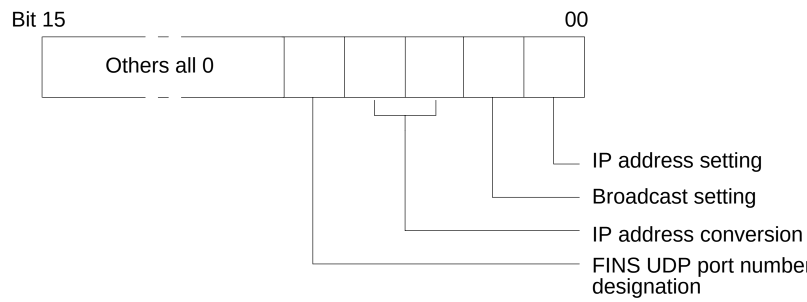
The Ethernet Unit mode and version are returned as ASCII characters occupying 20 bytes each (i.e., 20 characters each).

Example Model: CV500-ETN01
Version: V1.00

IP Address, Subnet Mask (Response) The Ethernet Unit's IP address and subnet mask are returned as 4 bytes each.

FINS UDP Port Number (Response) The Ethernet Unit's UDP port number for FINS is returned as 2 bytes.

Mode Setting (Response) The mode setting in the system setup is returned.



IP Address Setting
0: Node number rotary switch setting used
1: Node number rotary switch setting not used

Broadcast Setting
0: All 1s for host number (4.3BSD specification)
1: All 0s for host number (4.2BSD specification)

IP Address Conversion
00, 01: Automatic generation
10: IP address table
11: IP address table + automatic generation

FINS UDP Port Number Designation
0: Default value (9600)
1: System setup value

Ethernet Address (Response) The Ethernet address of the Ethernet Unit is returned. The Ethernet address is the address marked on the label on the side of the Ethernet Unit.

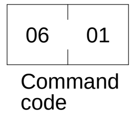
Response Codes

Response code	Description
0000	Normal
1001	Command too large

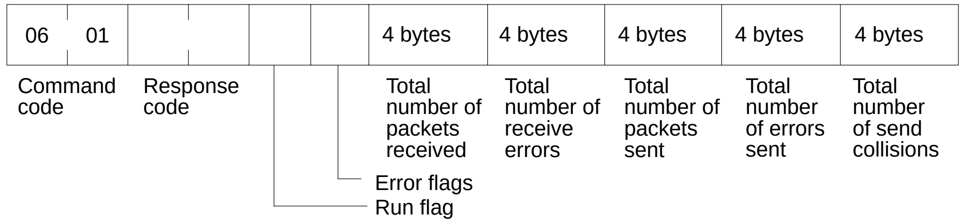
6-3-3 CONTROLLER STATUS READ

Reads the controller status.

Command Block



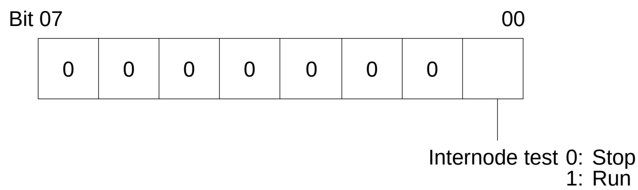
Response Block



Parameters

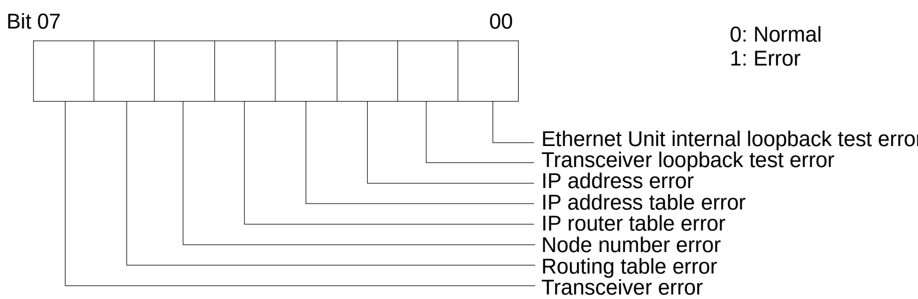
Run Flag (Response)

Indicates the Ethernet Unit’s operating status. Returned in the following format.



Error Flags (Response)

Indicates which errors occurred when the Ethernet Unit is booted.



The cause of each type of error is listed below.

Ethernet Unit Internal Loopback Test Error
An error occurred during self diagnostic testing after the Unit was turned on or reset. Refer *7-4 Other RAS Functions* for details.

Transceiver Loopback Test Error
An error occurred during the self diagnostic testing after the Unit was turned on or reset. Refer to *7-4 Other RAS Functions* for details.

IP Address Error
All bits for the network number or host number are set to 0 or 1.

IP Address Table Error
More than 32 records exist in the IP address table.

IP Router Table Error
More than 8 records exist in the IP router table.

Node Number Error
Rotary switches are not set between 1 and 126.

Routing Table Error
Local network table contains 0 or more than 16 records.
Relay network table contains more than 20 records.

Transceiver Error
The transceiver cable is disconnected or the transceiver is defective.

Total Number of Packets Received (Response)

The total number of packets received by the Ethernet Unit is returned.

Total Number of Receive Errors (Response)

The total number of errors detected while the Ethernet Unit was receiving is returned. The types of error detected are short packet errors, alignment errors, CRC errors, and 151–byte minimum frame receiving and sending controller overflow errors.

Total Number of Packets Sent (Response)

The total number of packets sent by the Ethernet Unit is returned.

Total Number of Errors Sent (Response)

The total number of packet errors detected while the Ethernet Unit was sending is returned.

Total Number of Send Collisions

The number of packets damaged by 16 collisions with data from other nodes during Ethernet Unit transmissions.

Precautions

Counting of the total number of packets received, total number of receive errors, total number of packets sent, total number of errors sent, and total number of send collisions is discontinued when the counted value reaches the maximum value.

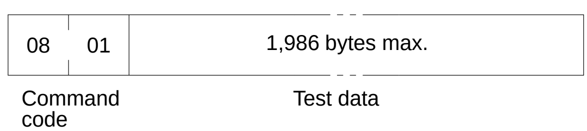
Response Codes

Response code	Description
0000	Normal
1001	Command too large

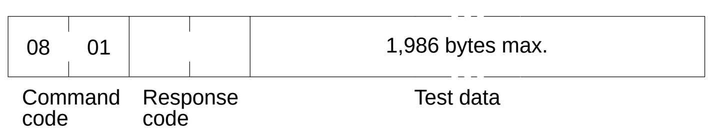
6-3-4 INTERNODE LOOPBACK TEST

Runs the loopback test between specified nodes.

Command Block



Response Block



Parameters

Test Data (Command, Response)

This command specifies the data to be sent to the specified nodes. Up to 1,986 bytes can be specified. The response sends back data identical to the data specified in the command. An abnormality is assumed if the data returned in the response differs from the test data sent.

Precautions

The test destination node is the destination node specified in the CMND(194) instruction operands.

Always specify the unit address of the Ethernet Unit in the CMND(194) instruction.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small (No test data)

6-3-5 BROADCAST TEST RESULTS READ

Reads the results (number of times data received) of the broadcast test.

Command Block

08 02	
Command code	

Response Block

08 02		
Command code	Response code	Times received

Parameters

Times Received (Response)

The number of times the data has been received normally during the broadcast send test is returned. The number of times received is cleared each time the result is read.

Response Codes

Response code	Description
0000	Normal
1001	Command too large

6-3-6 BROADCAST DATA SEND

Sends test data simultaneously to all nodes on the network.

Command Block

08 03	1,460 bytes max.
Command code	Test data

Parameters

Test Data (Command)

This command specifies the data to be sent to the specified nodes. Up to 1,460 bytes can be specified. The broadcast send test can be run without specifying the test data.

Precautions

No response is made to this command.

When using this command, set the FINS header parameters (or the control data for the CMND(194) instruction) as follows:

Destination node number:	FF (broadcast data)
Destination unit address:	FE (Ethernet Unit)
Response/no response flag:	1 (no response)

6-3-7 ERROR LOG READ

Reads the error log.

Command Block

21 02		
Command code	Beginning record number	Number of records

Response Block

21 02					10 bytes	10 bytes each
Command code	Response code	Maximum number of stored records	Number of stored records	Number of records	Error log data	Error log records

Parameters

Beginning Record Number (Command)

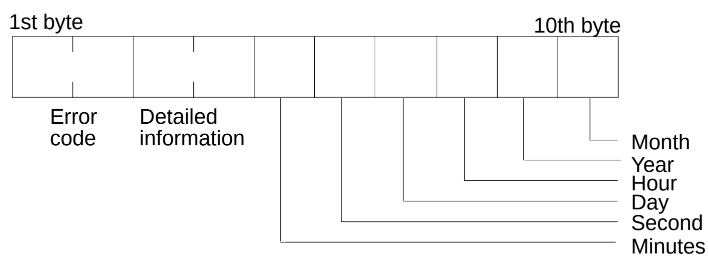
The first record to be read. The first record number can be specified in the range between 0000 and 00C6 (0 to 198 decimal) where 0000 is the oldest record.

Number of Records (Command, Response)
The number of records to read is specified between 0000 and 00C7 (0 to 199 decimal) in the command. The response returns the actual number of records read.

Maximum Number of Stored Records (Response)
The maximum number of records that can be stored in the error log. Care is required as the error log differs according to the type of PC or CPU Bus Unit. In an Ethernet Unit, the maximum number of stored records is fixed at C7 (199 decimal).

Number of Stored Records (Response)
The number of records stored at the time the command is executed is returned.

Error Log Data (Response)
The specified number of error log records from the beginning record number is returned sequentially. The total number of bytes in the error log is calculated as the number of records x 10 bytes/record. Each error log record thus comprises 10 bytes, configured as follows:



Error Code, Detailed Information
Details of the error stored in the record. Refer to *Section 9 Troubleshooting* for details.

Minute, Second, Day, Hour, Year, Month
Indicate the time at which the error stored in the record occurred.

Precautions

If the error log contains fewer records than the number specified in the number of records parameter, all records stored in the error log at the time the command is executed will be returned and the command executed will end normally.

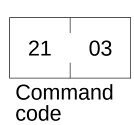
Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1103	Beginning record number is out of range
3005	No record exists at beginning record number

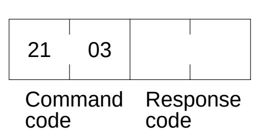
6-3-8 ERROR LOG CLEAR

Clears the number of records stored in the Ethernet Unit error log.

Command Block



Response Block



Response Codes

Response code	Description
0000	Normal
1001	Command too large

6-3-9 UDP OPEN REQUEST

Requests processing to open a socket.

Command Block

27	01								
Command code		UDP socket number		Results storage area				Local UDP port number	

Response Block

27	01		
Command code		Response code	

Results Storage Format

Results storage response code	

Parameters

UDP Socket Number (Command)

The UDP socket number to be opened specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Local UDP Port Number (Command)

The UDP port number for communications with the socket is specified as 2 bytes (0 cannot be specified). Packets received at this port are distributed to the socket specified in the UDP socket number, and send packets are distributed from the UDP socket to this port. The port number specified as the FINS UDP port number (default value 9600) cannot be used.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	UDP socket number is out of range. Local UDP port number is zero.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
220F	Specified socket is already open.

Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Internal buffer cannot be reserved due to high receive load (ENOBUFS).
0049	Duplicate UDP port number (EADDRINUSE).

6-3-10 UDP RECEIVE REQUEST

Requests that data be sent from a UDP socket.

Command Block

27 02				
Command code	UDP socket number	Results storage area	Number of reception bytes	Timeout value

Response Block

27 02	
Command code	Response code

Results Storage Format

Results storage response code	Source IP address	Source UDP port number	Number of reception bytes	Received data

Parameters

UDP Socket Number (Command)

The UDP socket number to receive data specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Number of Reception Bytes (Command, Results Storage Area)

The maximum number of bytes of data to be received is given in the command. The number of bytes of data received will be stored in the results storage area. Up to 1,982 bytes can be specified.

Timeout Value (Command)

The maximum control time between receiving the receive request and storing the result. If this set time limit is exceeded, the code for a timeout error will be set as the results storage response code. The value is set in units of 0.1 s. The timeout time will be unlimited if the value is set to 0.

Source IP Address (Results Storage Area)

The IP address of the node sending data.

Source UDP Port Number (Results Storage Area)

The port number of the node sending data.

Received Data (Results Storage Area)

The data sent from the remote node.

Precautions

If a packet is received which contains more bytes than the number specified in *Number of reception bytes* specified in the command, the specified number of bytes will be stored and the remainder of the bytes will be discarded.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	UDP socket number or number of reception bytes is out of range.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
220F	The specified socket is currently receiving data.
2210	The specified socket is not open.

Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Internal buffer cannot be reserved due to high reception load (ENOBUFS).
0080	A receive request timeout error occurred.
0081	The specified socket was closed while receiving data.

6-3-11 UDP SEND REQUEST

Requests that data be received by a UDP socket.

Command Block

27 03						1974 bytes max.
Command code	UDP socket number	Results storage area	Destination IP address	Destination UDP port number	Number of bytes sent	Send data

Response Block

27 03	
Command code	Response code

Results Storage Format

Response code	Number of bytes sent

Parameters

UDP Socket Number (Command)
The UDP socket number to send the data specified as 2 bytes between 1 and 8.

Results Storage Area (Command)
The area in which the result of the command execution is stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Destination IP Address (Command)
The IP address of the node to which data is being sent.

Destination UDP Port Number (Command)
The UDP port number of the node to which data is being sent.

Number of Bytes Sent (Command, Results Storage Area)
The number of bytes in the data sent by this command. Up to 1,974 bytes can be specified, or up to 1,472 bytes can be specified if the broadcast address is specified as the send destination. The results storage area stores the actual number of bytes sent.

Send Data (Command)
Specifies the data sent to the remote node.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1003	The number of bytes sent does not match the sent data length.
1100	UDP socket number or number of bytes sent is out of range. The destination IP address is 0.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
2210	The specified socket is not open.

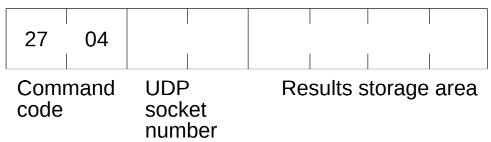
Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Internal buffer cannot be reserved due to high reception load (ENOBUFS).
0042	The send destination IP address is a broadcast address and the number of bytes sent exceeds 1,472. (EMSGSIZE)
004C	Incorrect network number. Incorrect destination IP address (EADDRNOTAVAIL).
004E	No network number in IP router table. Router incorrectly specified. Incorrect destination IP address (ENETUNREACH).
0051	Router incorrectly specified. Incorrect destination IP address (EHOSTUNREACH).

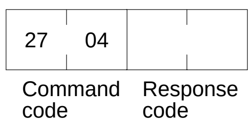
6-3-12 UDP CLOSE REQUEST

Requests processing to close a socket.

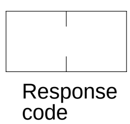
Command Block



Response Block



Results Storage Format



Parameters

UDP Socket Number (Command)

The UDP socket number to be closed specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	UDP socket number is out of range.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
2210	Specified socket is not open.

Results Storage Area Response Codes

Response code	Description
0000	Normal

6-3-13 TCP OPEN REQUEST (PASSIVE)

Requests processing to open a TCP socket. The socket will wait to be connected to another node.

Command Block

27	10																
Command code		TCP socket number		Results storage area				Local TCP port number		Timeout value		Remote IP address				Remote TCP port number	

Response Block

27	10		
Command code		Response code	

Results Storage Format

Response code		Remote IP address				Remote port number	

Parameters

TCP Socket Number (Command)

The TCP socket number to be opened specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Local TCP Port Number (Command, Results Storage Area)

The TCP port number for communications with the socket is specified as 2 bytes (0 cannot be specified). Do not specify the port number of the FTP server (port #21).

Timeout Value (Command)

The maximum control time between receiving the open request and storing the result. If this set time limit is exceeded, the code for a timeout error will be set as the results storage response code. The value is set in units of 0.1 s. The timeout time is unlimited if the value is set to 0.

Remote IP Address (Command, Results Storage Area)

Specify the remote node's IP address. If all zeroes are set, no remote node is specified and connection is awaited from any node. If any other value is set, connection is awaited from the specified remote node. The IP address of the connected remote node will be stored in the results storage area.

Remote Port Number (Command, Results Storage Area)

Specify the remote TCP port number with this command. If all zeroes are set, no remote TCP port number is specified. If any other value is set, it specifies the TCP port number of the remote node. The TCP port number of the connected remote node will be stored in the results storage area.

Precautions

Processing varies as shown in the table below according to the specified combination of remote IP address and remote TCP port number.

Remote IP address	Remote TCP port	Description
0	0	All connection requests received
0	Not 0	Received only when port number matches.
Not 0	0	Received only when IP address matches.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	TCP socket number is out of range. Local TCP port number is 0.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
220F	The specified socket (connection) is already open or is currently being opened.

Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Internal buffer cannot be reserved due to high reception load.
0042 (see note)	An error occurred (EMSGSIZE).
0045	A communication error occurred with the remote node (ECONNABORTED).
0049	Duplicated port numbers (EADDRINUSE).
004A (see note)	An error occurred (ECONNREFUSED).
004B (see note)	A communication error occurred with the remote node (ECONNRESET).
004E (see note)	A parameter error occurred at the remote IP address (ENETUNREACH).
0051 (see note)	A parameter error occurred at the remote IP address (EHOSTUNREACH).
0053	A communication error occurred with the remote node (ETIMEDOUT). No remote exists.
0080	An open request timeout error occurred.
0081	Socket was closed during opening procedure.
0082	Connection could not be established with the specified remote.

Note These errors occur only in large multilayered networks.

6-3-14 TCP OPEN REQUEST (ACTIVE)

Requests processing to open a TCP socket. The socket will be connected to another node.

Command Block

27 11					
Command code	TCP socket number	Results storage area	Local TCP port number	Remote IP address	Remote TCP port number

Response Block

27 11	
Command code	Response code

Results Storage Format

Response code	Local port number

Parameters

TCP Socket Number (Command)

The TCP socket number to be opened specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Local TCP Port Number (Command, Results Storage Area)

The TCP port number for communications with the socket is specified as 2 bytes (0 cannot be specified). Do not specify the port number of the FTP server (port #21). An available TCP port number is automatically assigned if 0 is specified.

Remote IP Address (Command)

Specify the remote node's IP address (must be non-zero).

Remote Port Number (Command)

Specify the remote TCP port number (must be non-zero).

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	TCP socket number is out of range or remote IP address is 0.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
220F	The specified socket (connection) is already open or is being opened.

Results Storage Area Response Codes

Response code	Description
0000	Normal
000D	A parameter error occurred at the remote IP address (EACCES).
003E	Internal buffer cannot be reserved due to high receive load (ENOBUFS).
0042 (see note)	An error occurred (EMSGSIZE).
0044	Received ICMP data (ENOPROTOOPT).
0045	Local socket closed (ECONNABORTED).
0049	Duplicated port numbers (EADDRINUSE).
004A	An error occurred (ECONNREFUSED). Passive remote is not available. An attempt was made to actively open local TCP port.
004B (see note)	A communication error occurred with the remote node (ECONNRESET).
004C	A parameter error occurred at the remote IP address (EADDRNOTAVAIL). Specified incorrectly.
004E	A parameter error occurred at the remote IP address (ENETUNREACH). No network number in IP router table or incorrect router setting.
0051	A parameter error occurred at the remote IP address (EHOSTUNREACH). Incorrect router setting.
0053	A communication error occurred with the remote node (ETIMEDOUT). No remote exists.
0081	Socket was closed during opening procedure.

Note These errors occur only in large multilayered networks.

6-3-15 TCP RECEIVE REQUEST

Requests that data be sent from a TCP socket.

Command Block

Command code	TCP socket number	Results storage area	Number of reception bytes	Timeout value
27 12				

Response Block

Command code	Response code
27 12	

Results Storage Format

Response code	Number of reception bytes	Received data
		Received bytes

Parameters

TCP Socket Number (Command)

The TCP socket number to receive data specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Number of Reception Bytes (Command, Results Storage Area)

The maximum number of bytes of data to be received is given in the command. The number of bytes of data received will be stored in the results storage area. Up to 1,982 bytes can be specified.

Timeout Value (Command)

The maximum control time between receiving the receive request and storing the result. If this set time limit is exceeded, the code for a timeout error will be set as the results storage response code. The value is set in units of 0.1 s. The timeout time is unlimited if the value is set to 0.

Received Data (Results Storage Area)

Stores the received data.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	TCP socket number or number of reception bytes is out of range.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
2210	No connection could be established to the specified socket.
220F	The specified socket is receiving data.

Results Storage Area Response Codes

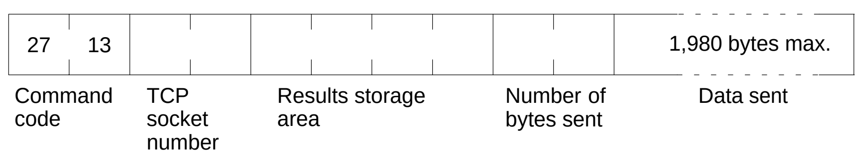
Response code	Description
0000	Normal
003E	Internal buffer cannot be reserved due to high receive load (ENOBUFS).
0042 (see note)	Received ICMP data (EMSGSIZE).
0044 (see note)	Received ICMP data (ENOPROTOOPT).
0045 (see note)	A communication error occurred with the remote node (ECONNABORTED).
004B (see note)	A communication error occurred with the remote node (ECONNRESET).
004E (see note)	Received ICMP data (ENETUNREACH).
004F (see note)	Received ICMP data (EHOSTDOWN).
0051 (see note)	Received ICMP data (EHOSTUNREACH).
0053 (see note)	A communications error occurred with the remote node (ETIMEDOUT).
0080	A receive request timeout error occurred.
0081	Socket was closed while receiving.

Note These errors occur only in large multilayered networks.

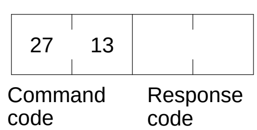
6-3-16 TCP SEND REQUEST

Requests that data be received at a TCP socket.

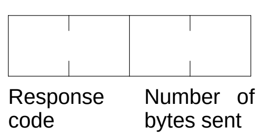
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command)

The TCP socket number to send the data specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Number of Bytes Sent (Command, Results Storage Area)

The number of bytes in the data sent specified between 1 and 1,980. The results storage area stores the actual number of bytes sent.

Data Sent (Command)

Specifies the data to be sent.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1003	The number of bytes sent does not match the amount of data.
1100	The TCP socket number or number of bytes sent is out of range.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
2210	No connection could be established to the specified socket.
220F	The specified socket is sending data.

Results Storage Area Response Codes

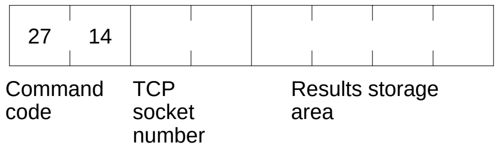
Response code	Description
0000	Normal
0020	Connection to the remote socket was broken during transmission (EPIPE).
003E	Internal buffer cannot be reserved due to high receive load (ENOBUFS).
0042 (see note)	An error occurred (EMSGSIZE).
0044 (see note)	Received ICMP data (ENOPROTOOPT).
0045 (see note)	A communication error occurred with the remote node (ECONNABORTED).
004B (see note)	A communication error occurred with the remote node (ECONNRESET).
004E (see note)	A parameter error occurred at the remote IP address (ENETUNREACH).
004F (see note)	Received ICMP data (EHOSTDOWN).
0051 (see note)	A parameter error occurred at the remote IP address (EHOSTUNREACH).
0053 (see note)	A communication error occurred with the remote node (ETIMEDOUT).
0081	The specified socket was closed during transmission.

Note These errors occur only in large multilayered networks.

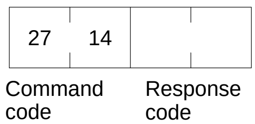
6-3-17 TCP CLOSE REQUEST

Requests processing to close a TCP socket. Other processing being carried out is forcibly ended and a code is recorded in the results storage area.

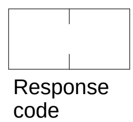
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command)

The TCP socket number to be closed specified as 2 bytes between 1 and 8.

Results Storage Area (Command)

The area in which the results of the command execution are stored. The first byte specifies the memory area and data type (variable type). The 2nd to 4th bytes specify the beginning address of the results storage area. Refer to page 128 for details about the variable types and addresses that can be specified.

Precautions

Any other processing, such as sending or receiving data, being carried out when this close command is executed will be forcibly ended and a code will be stored in the results storage area to indicate that this processing was forcibly ended.

Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
1100	The TCP socket number is out of range.
1101	The variable type for the results storage area is out of range.
1103	Non-zero bit address specified for the results storage area.
2210	No connection could be established to the specified socket.

Results Storage Area Response Codes

Response code	Description
0000	Normal

6-3-18 PING

Equivalent processing to the Unix computer `ping` command (see below).

Command Block

27	20						
Command code		Destination IP address				Timeout value	

Response Block

27	20		
Command code		Response code	

Parameters

Destination IP Address (Command)

The IP address of the destination node for the Ping command echo request packet.

Timeout Value (Command)

The wait time for the echo reply packet. The value is set in seconds. The timeout time is set at 20 seconds if the value is specified as 0. If the echo reply packet is not received within the set time limit, the code for a timeout error will be set as the results storage response code.

Remarks

PING Command

The PING command runs the echoback test using the ICMP protocol. When the PING command is executed, an echo request packet is sent to the remote node ICMP. Correct communications are confirmed when the returned response packet is received normally. The echo reply packet is automatically returned by the remote node ICMP. Refer to *7-1 Echoback Test using PING.*

Response Codes

Response code	Description
0000	Normal (echo reply received from the remote node)
0205	Timeout error
1001	Command too large
1002	Command too small
1100	Zero destination address
220F	PING command currently being executed

6-3-19 IP ADDRESS TABLE READ

Reads the IP address table.

Command Block

27	60		
Command code		Number of records	

Response Block

27	60									6 bytes	6 bytes
Command code		Response code		Maximum number of stored records		Number of stored records		Number of records		IP address table records	

Parameters

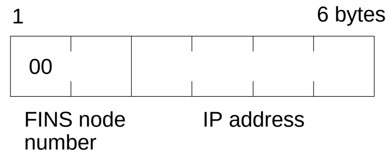
Number of Records (Command, Response)

The number of records to read is specified between 0000 and 0020 (0 to 32 decimal) in the command. If this value is set to 0, the number of stored records is returned but the IP address table records are not returned. The response returns the actual number of records read.

Maximum Number of Stored Records (Response)
The maximum number of records that can be stored in the IP address table is returned. The maximum number of stored records is fixed at 32.

Number of Stored Records (Response)
The number of IP address table records stored at the time the command is executed is returned.

IP Address Table Records (Response)
The number of IP address table records specified in the number of records parameter is returned. The total number of bytes in the IP address table records is calculated as the number of records x 6 bytes/record. The configuration of the 6 bytes of data in each record is as shown in the following diagram.



FINS Node Number
Node number for communications via the FINS command.

IP Address
Node number used by TCP/IP protocol.

Precautions

If the IP address table contains fewer records than the number specified in the *number of records* parameter, all the records contained in the IP address table when the command is executed will be returned and the command execution will end normally.

An error response will be returned if the IP address conversion method in the system mode settings is set to the automatic generation method.

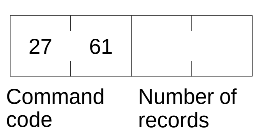
Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small
2307	IP address conversion method is set to the automatic generation method.

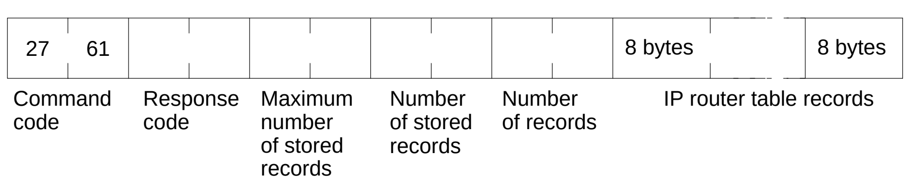
6-3-20 IP ROUTER TABLE READ

Reads the IP router table.

Command Block



Response Block



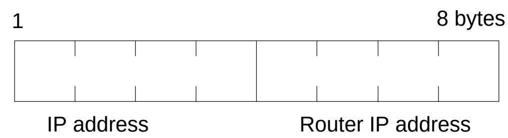
Parameters

Number of Records (Command, Response)
The number of records to read is specified between 0000 and 0008 (0 to 8 decimal) in the command. If this value is set to 0, the number of stored records will be returned but the IP router table records will not be returned. The response returns the actual number of records read.

Maximum Number of Stored Records (Response)
The maximum number of records that can be stored in the IP router table is returned. The maximum number of stored records is fixed at 8.

Number of Stored Records (Response)
The number of IP router table records stored at the time the command is executed is returned.

IP Router Table Records (Response)
The number of IP router table records specified in the *number of records* parameter are returned. The total number of bytes in the IP router table records is calculated as the number of records x 8 bytes/record. The configuration of the 8 bytes of data in each record is shown below.



IP Address
The network number from the IP address. The subnet mask set for the Ethernet Unit is used here, i.e., both the network and subnet numbers are returned.

Router IP Address
The IP address of a router connected to a network specified with IP addresses.

Precautions

If the IP router table contains fewer records than the number specified in the *number of records* parameter, all the records contained in the IP router table when the command is executed will be returned and the command execution will end normally.

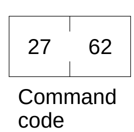
Response Codes

Response code	Description
0000	Normal
1001	Command too large
1002	Command too small

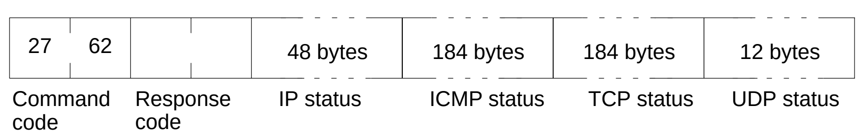
6-3-21 PROTOCOL STATUS READ

Reads the Ethernet Unit protocol status.

Command Block



Response Block



Parameters

IP Status (Response)
Twelve types of IP status information occupying 4 bytes each are returned in the following sequence:

1, 2, 3...

1. Total number of IP packets received.
2. The number of IP packets discarded due to an error with the checksum specified in the packet header.
3. The number of IP packets discarded because the received packet was larger than the overall packet length value specified in the packet header.
4. The number of IP packets discarded because the minimum size of the IP header data could not be stored in the first short buffer when an attempt was

made to store the packet. Refer to *6-3-22 MEMORY STATUS READ* and *Appendix F Buffer Configuration*.

5. The number of packets discarded for one of the following reasons:
    - The IP header length value specified in the IP header was smaller than the smallest size of the IP header.
    - The size of the first short buffer was smaller than the IP header length value specified in the IP header when storing the packet.
6. The number of IP packets discarded because the IP header length was larger than the overall packet length value specified in the packet header.
7. The number of fragmented packets received.
8. The number of received fragmented IP packets discarded because a queue for reassembly could not be secured.
9. The number of fragmented IP packets discarded because they could not be reassembled within 12 seconds after being received.
10. Always 0.
11. Always 0.
12. Always 0.

ICMP Status (Response)

Ten types (46 items) of ICMP status information occupying 4 bytes each are returned in the following sequence:

1, 2, 3...

1. The number of times the ICMP error routine was called. The ICMP error routine uses ICMP packets to inform the source about errors. The routine is called when an illegal packet is received (error in IP option processing or error in relay processing) or if the object port does not exist when using UDP.
2. Always 0.
3. Always 0.
4. Total number of outputs of each packet type during ICMP output. The 19 statistical values are returned in the order shown below. Contents are defined for 12 types only; all other types contain 0.

#0	#1	#2	#3	#4	#5	to	#15	#16	#17	#18

4 bytes

Type number	Description
#0	Echo reply
#1, #2	Undefined, always 0
#3	Destination unreachable
#4	Source quench
#5	Routing redirect
#6, #7	Undefined, always 0
#8	Echo
#9, #10	Undefined, always 0
#11	Time exceeded
#12	Parameter problem
#13	Time stamp
#14	Time stamp reply
#15	Information request
#16	Information request reply
#17	Address mask request
#18	Address mask reply

5. The number of received ICMP packets discarded because the type-indication code was out of range.
6. The number of received ICMP packets discarded because the overall packet length value specified in the packet header was smaller than the minimum ICMP packet length.
7. The number of received ICMP packets discarded because of an incorrect checksum value specified in the packet header.
8. The number of received ICMP packets discarded because the ICMP header length value specified in the packet header did not match the lengths of individual header types.
9. The number of responses returned to received ICMP packets requiring a response.
10. Total number of inputs of each packet type during ICMP input. The 19 statistical values are returned in the order shown below. Contents are defined for 12 types only; all other types contain 0.

#0	#1	#2	#3	#4	#5	to	#15	#16	#17	#18

4 bytes

Type number	Description
#0	Echo reply
#1, #2	Undefined, always 0
#3	Destination unreachable
#4	Source quench
#5	Routing redirect
#6, #7	Undefined, always 0
#8	Echo
#9, #10	Undefined, always 0
#11	Time exceeded
#12	Parameter problem
#13	Time stamp
#14	Time stamp reply
#15	Information request
#16	Information request reply
#17	Address mask request
#18	Address mask reply

TCP Status (Response)

Three types (46 items) of TCP status information occupying 4 bytes each are returned in the following sequence:

1) Connection Information
Fifteen items are returned in the following sequence:

1, 2, 3...

1. The number of times active connections were correctly established.
2. The number of times a SYN packet was received while waiting to establish a passive connection.
3. The number of times active or passive connections were correctly established.
4. The number of times an established connection was cut off.
5. The number of times the connection wait status was cut off.
6. The number of times protocol control blocks or other actively allocated structures were released.

7. The number of segments for the round-trip time (time from segment transmission to ACK).
8. The number of times the round-trip time was changed.
9. The number of times a delayed acknowledgement (ACK) was sent. If the order of the received segments is reversed, ACK is sent with a packet of data separate from ACK (response to input data, etc.) or is immediately sent with the ACK for other data.
10. The number of times the connection was cut off because no ACK was returned after several resend attempts.
11. The number of times no ACK was returned within the resend timer set time. (The resend timer sets the maximum time limit between the data being output and ACK being returned.)
12. The number of times no window advertisement is received within the time set on the duration timer. (The duration timer sets the maximum time limit for a window advertisement to be received if the transmission window is smaller than necessary and the resend timer is not set. If no window advertisement is received within the time limit, the number of segments permitted by the transmission window are sent. If the transmission window is set to 0, a window probe (1 octet of data) is sent before the timer restarts.)
13. The number of times no segment was sent or received within the time set on the hold timer.

    If the connection is open when the hold timer set time elapses and the SO_KEEPALIVE option is set for the socket, the next routine is executed via a timer routine.

    A hold packet is sent. This packet is designed so that the remote TCP sends an ACK or RST. The disconnection is completed when RST is received from the remote node. The packet is resent a set number of times if no response is made to the hold packet. If still no response is made, the connection is cut off.
14. The number of times the hold packet is resent.
15. The number of times the hold packet is sent without response before the connection is cut off.

2) Send Information

Ten information items are returned in the following sequence:

1, 2, 3...

1. The total number of packets sent.
2. The number of data packets sent.
3. The number of data bytes sent.
4. The number of data packets resent.
5. The number of data bytes resent.
6. The number of ACK packets sent.
7. The number of window probes (1 octet of data) sent.
8. The number of emergency data packets sent.
9. The number of window advertisement packets sent.
10. The number of control packets (SYN, FIN, RST) sent.

3) Receive Information

Twenty-one information items are returned in the following sequence:

1, 2, 3...

1. The total number of packets received.
2. The number of packets received continuously.
3. The number of bytes received continuously.
4. The number of received packets discarded due to an incorrect checksum.

5. The number of packets discarded because the TCP header was smaller than the minimum size for a TCP header or was larger than the IP packet.
6. The number of packets discarded because the TCP header and IP header could not be stored in the first short buffer.
7. The number of resent packets received.
8. The number of bytes in the resend packets.
9. The number of duplicated resend packets received.
10. The number of bytes in the duplicated resend packets received.
11. The number of out-of-range data packets received.
12. The number of bytes in the out-of-range data packets received.
13. The number of packets where the data was larger than the window.
14. The number of bytes in the packets where the data was larger than the window.
15. The number of packets received after closing.
16. The number of window probe packets received.
17. The number of resent ACK packets received.
18. The number of ACK packets received with no data set.
19. The number of ACK packets received.
20. The number of ACK packets received for received transmission acknowledgements (ACK).
21. The number of window advertisement packets received.

UDP Status (Response)

Three items of UDP information occupying 4 bytes each are returned in the following sequence:

1, 2, 3...

1. The number of packets discarded because the size of the first short buffer was smaller than the minimum size (28) of the IP header and UDP header when the packet was stored.
2. The number of packets discarded due to an incorrect checksum specified in the UDP header.
3. The number of packets discarded because the IP overall length specified in the IP header was shorter than the UDP overall length in the UDP header.

Precautions

All the above values are set to 0 if network operation stops due to incorrect settings in the system setup.

Response Codes

Response code	Description
0000	Normal
1001	Command too large

6-3-22 MEMORY STATUS READ

Reads the status of the Ethernet Unit's network memory. The network memory contains 108K bytes that are used as required as for communications buffers for communications servicing. The network memory consists of 352 short buffers (128 bytes each) and 64 long buffers (1,024 bytes each). Refer to *Appendix F Buffer Configuration*.

Command Block

27	63
Command code	

Response Block

27	63			92 bytes
Command code		Response code		Memory status

Parameters

Memory Status (Response)

A total of 23 data items in six areas are returned in the following order. Each item consists of 4 bytes.

1, 2, 3...

1. Short Buffer Application: Two items are returned.
   a) The number of short buffers currently being used.
   b) The number of short buffers in the system (fixed at 352 decimal).
2. Short Buffer Application by Type: Thirteen items are returned.
   a) The number of short buffers used for storing communications data.
   b) The number of short buffers used for protocol headers (TCP, UDP, IP, ICMP, ARP)
   c) The number of short buffers used in socket structures
   d) The number of short buffers used as protocol control blocks
   e) The number of short buffers used for routing tables
   f) Not used (always 0)
   g) The number of short buffers used for ARP tables
   h) The number of short buffers used for IP fragment re-assembly queue headers
   i) The number of short buffers used for storing socket addresses
   j) Not used (always 0)
   k) The number of short buffers used for storing socket options
   l) The number of short buffers used for storing access rights
   m) The number of short buffers used for storing interface addresses
3. Long Buffer Application: Two items are returned.
   a) The number of long buffers currently being used.
   b) The number of long buffers in the system (fixed at 64 decimal).
4. Not Used: Always 0.
5. Network Memory Application: Two items are returned.
   a) The number of bytes used (in K bytes)
   b) The percentage used.
6. Memory Exhaustion Log

   Counts for the following values indicate a high load on the Ethernet Unit. These high loads may be caused by problems in communications, particularly FINS communications and UDP sockets. If these values are consistently high, check your applications.

The following values are reset when the Ethernet Unit is started or reset. The values will be counted to the maximum value and then stop, i.e., the maximum value will be maintained once reached until the Unit is restarted or reset.

a) The number of times an attempt was made to secure a short buffer without WAIT when there were no short buffers available.

b) The number of times an attempt was made to secure a short buffer with WAIT when there were no short buffers available.

c) The number of times an attempt was made to release and secure a short buffer already being used by another socket when there were no short buffers available.

Precautions

All the above values are set to 0 if Ethernet communications functions are stopped due to improper settings in the system setup.

Response Codes

Response code	Description
0000	Normal
1001	Command too large

6-3-23 SOCKET STATUS READ

Reads the Ethernet Unit network socket status.

Command Block

27	64

Command code

Response Block

27	64			32 bytes

Command code / Response code / Socket status

Parameters

Socket Status (Response)

Returns eight types of information in records of 32 bytes each. A maximum of 64 records can be returned. The format of each record is shown below.

4 bytes	4 bytes	4 bytes	4 bytes	4 bytes	4 bytes	4 bytes	4 bytes
Protocol	Receive queue	Send queue	Local IP address	Local port number	Remote IP address	Remote port number	TCP transition

Protocol
The protocol used for the socket is returned as a number.
00 00 00 01: ICMP; 00 00 00 06: TCP; 00 00 00 11: UDP

Receive Queue
The number of bytes in the reception queue.

Send Queue
The number of bytes in the send queue.

Local IP Address
The local IP address allocated to the socket.

Local Port Number
The local port number allocated to the socket.

Remote IP Address
The remote IP address allocated to the socket.

Remote Port Number
The remote port number allocated to the socket.

TCP Transitions
The TCP connection status is returned as one of the numbers shown in the following table.

Number	Stage	Status
00 00 00 00	CLOSED	Closed.
00 00 00 01	LISTEN	Waiting for connection.
00 00 00 02	SYN SENT	SYN sent in active status.
00 00 00 03	SYN RECEIVED	SYN sent and received.
00 00 00 04	ESTABLISHED	Already established.
00 00 00 05	CLOSE WAIT	Received FIN, waiting to close.
00 00 00 06	FIN WAIT 1	Completed and FIN sent.
00 00 00 07	CLOSING	Completed and exchanged FIN. Awaiting ACK.
00 00 00 08	LAST ACK	FIN received and completed. Awaiting ACK.
00 00 00 09	FIN WAIT 2	Close completed and ACK received. Awaiting FIN.
00 00 00 0A	TIME WAIT	After closing, pauses twice the maximum segment life (2MSL).

Precautions

All the above values are set to 0 if Ethernet communications functions are stopped due to improper settings in the system setup.

Response Codes

Response code	Description
0000	Normal
1001	Command too large

SECTION 7
RAS Features

This section describes the RAS features of the Ethernet Unit. These features are designed to increase the Reliability, Availability, Serviceability of Ethernet Units.

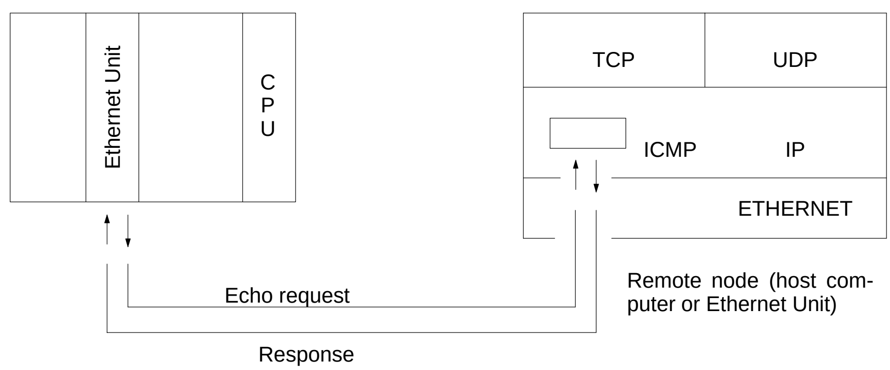
7-1 Loopback Test Using PING

The Ethernet Unit incorporates the PING command supported as a standard feature by many host computers. The PING command is an loopback test using ICMP (Internet Control Message Protocol).

7-1-1 PING Command

The PING command sends an echo request packet to a remote node and receives an echo response packet to confirm that the remote node is communicating correctly. The PING command uses the ICMP echo request and responses. The echo response packet is automatically returned by the ICMP.

The PING command is normally used to check the connections of remote nodes when configuring a network. The Ethernet Unit supports both the ICMP echo request and reply functions.



7-1-2 Ethernet Unit

The Ethernet Unit automatically returns the echo response packet in response to an echo request packet sent by another node (host computer or other Ethernet Unit). An echo request packet can be sent to another node by issuing the FINS command to execute the PING command from the PC. Refer to page 146 for details on the PING command.

7-1-3 Host Computer

The PING command can be executed from the host computer to send an echo request packet to an Ethernet Unit. The method for using the PING command from a Unix computer is given next.

Method

Input the following at the host computer prompt ($):

`$ ping` *IP_address* (*host_name*)

The destination is specified by its IP address or host name. If the host name is used, the host name must be defined in file /etc/hosts.

Note The PING command is not supported by some host computers.

Application Examples

These examples show sending the PING command to the node at IP address 130.25.36.8. The “$” on the example screen represents the host computer prompt. User inputs are underlined. Comments are placed after semicolons.

Normal Execution

```
$ ping 130.25.36.8                        ; Executes PING command
PING 130.25.36.8: 56 data bytes
64 bytes from 130.25.36.8: icmp_seq=0. time=0.ms
64 bytes from 130.25.36.8: icmp_seq=0. time=0.ms
       .             .             .
       .             .             .
       .             .             .
64 bytes from 130.25.36.8: icmp_seq=0. time=0.ms
←Enter DEL key to cancel.                 ;User presses DEL key.
----130.25.36.8 PING Statistics----
9 packets transmitted, 9 packets received, 0% packets loss
round–trip (ms) min/avg/max = 0/1/16
$
```

Abnormal Execution

```
$ ping 130.25.36.8                        ; Executes PING command
PING 130.25.36.8: 56 data bytes
←Enter DEL key to cancel.                 ;User presses DEL key.
----130.25.36.8 PING Statistics----
9 packets transmitted, 0 packets received, 100% packets loss
$
```

Refer to operating system documentation for your host computer for details about using the host computer’s PING command.

7-2 Internode Test

The internode test sends data to and from specified nodes and uses the responses to check the network. The Ethernet Unit has a built-in function to run the internode test.

7-2-1 Running the Test

The test parameters are set and the test is started and stopped from the PC used to start the internode test. These setting are made using the CVSS.

Test Procedure

1, 2, 3...

1. Set the PC used to start the internode test to PROGRAM mode.
2. Write the test parameters into PC memory from the CVSS. The test parameters are described following this procedure.
3. Turn on the Internode Test Start Bit in the CPU Bus Unit Area allocated to the Ethernet Unit. The Internode Test Start Bit is described following the test parameters. This will begin the internode test.
4. If desired, the test parameters can be changed while the internode test is actually running. Use the same procedure as used in step 2.
5. Turn off the Internode Test Start Bit in the CPU Bus Unit Area allocated to the Ethernet Unit to stop the test.

Note

1. The PC must be in PROGRAM mode to run the internode test. Set the PC in PROGRAM mode before running the test. The remote node can be in any mode.
2. The test parameters become effective immediately when they are set or changed. It is not necessary to reboot or restart. If the test parameters are changed during the test, the test will continue with the new parameters.

Setting the Test Parameters

Before starting the test, set the following parameters in the software switches in the words in the DM area allocated to the Ethernet Unit. The test parameters become effective immediately after they are set. It is not necessary to reboot or restart the PC.

Configuration of the Software Switches

The software switches are stored at offsets from the first word calculated with the formula first word = D02000 + (100 x unit number).

Offset	Contents	
+0	Bits 8 to 15: Remote network address	Bits 0 to 7: Remote node number
+1	Number of send bytes	
+2	Response monitor time	

Setting Range

The parameter setting ranges are given in the following table.

Parameter	Range
Remote network address	01 to 7F (1 to 127 decimal)
Remote node number	01 to 7E (0 to 126 decimal)
Number of send bytes	0001 to 07CC (1 to 1996 decimal) 0000 specifies the max. length of 1,996 bytes.
Response monitor time (Unit: 10 ms)	0001 to FFFF (1 to 65535 decimal) 0000 specifies 2 s.

Note

1. The following limits apply to the number of bytes sent when running the test through a SYSMAC NET or SYSMAC LINK network:

   SYSMAC NET: 1,986 bytes
   SYSMAC LINK: 512 bytes

2. Broadcast transmissions (target node address = FF) cannot be used for the internode test.
3. A timeout error will occur if no response is received within the set response monitor time.

Starting and Stopping the Internode Test

Bit 01 of the first word in the words allocated to the Ethernet Unit in the CPU Bus Unit area is used as the Internode Test Start Bit. Turn ON (1) bit 01 to start the internode test and turn bit 01 OFF (0) to stop the internode test. The address of the word containing the Internode Test Start Bit is calculated by the following formula:

Word = 1500 + (25 x unit number)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0

Internode Test Start Bit
0: Stop internode test
1: Start internode test

Internode test run status is shown on the TS indicator on the Unit's front panel.

TS indicator	Run status
Lit	Internode test running
Not lit	Internode test stopped

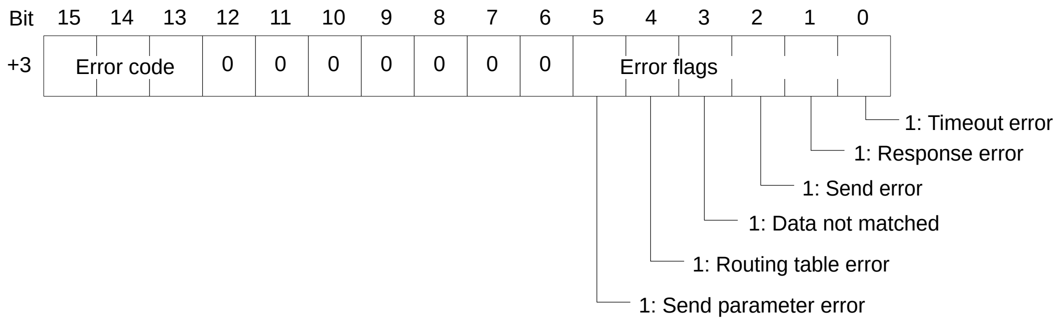
7-2-2 Checking Results

The results of the internode test are stored in the fourth through ninth words of the portion of the CPU Bus Unit area allocated to the Ethernet Unit. The status information area stores the test status and numbers of test runs and errors.

Test Status

The result of the test run and descriptions of errors are stored as the test status. The test status is stored at the word whose address is calculated by the following formula:

Word = D02000 + (100 x unit number) + 3



Error Code

If the test is run repeatedly, the code for the latest test results is stored.

Bit			Description
15	**14**	**13**	
0	0	0	Normal
0	0	1	Timeout error
0	1	0	Response (response code) error
0	1	1	Send error
1	0	0	Data not matched
1	0	1	Routing table error (see note)
1	1	0	Send parameter error (see note)

Note The internode test does not count errors when a routing table error or send parameter error has occurred.

Error Flags

The bits corresponding to errors are turned ON (1) if errors occur. The flag status is maintained until the internode test is run again.

Number of Test Runs and Errors

The number of test runs and total errors from the time the test is started until it is stopped are stored. The configuration of this area is shown in the following table. Each result is stored at a word relative to the first word the address of which can be calculated using the following formula:

Word = D02000 + (100 x unit number) + 4 to +8

Offset	Contents
+4	Number of internode test runs
+5	Number of timeout errors
+6	Number of response errors
+7	Number of send errors
+8	Number of times data did not match

Note

1. The contents of the test status area and test runs/errors area are maintained until the internode test is run again.
2. When the number of tests counts to the maximum value (FFFF), subsequent internode test runs are counted from 0. However, the maximum value is maintained and further errors are not counted when the number of errors reaches the maximum value.

7-3 Error Log

This section describes the error log which records all errors occurring during Ethernet Unit operation.

7-3-1 Error Log

The error log records errors which occur during Ethernet Unit operation. The error log is read and cleared from Programming Device (such as the CVSS) or by sending FINS commands from other nodes, such as host computers. See pages 133 and 134 for details on FINS commands.

Errors Recorded

The following errors are recorded in the error log:

Errors related to network operation.
Errors related to sending or receiving.
Errors related to the PC.

Error Log Table

Each error occupies one record of the error log table. The specifications of the error log table are given in the following table.

Item	Specification
Record length	10 bytes/record
No. of records	199 records max.
Data format	Binary
Record configuration	Error code (2 bytes) Detailed information (2 bytes) Time stamp (6 bytes)

Error Log Overflows

If an error occurs when the error log table already contains the maximum 199 records, the oldest record is discarded and the latest error is added to the end of the table.

Note The error log is not cleared when the PC is turned on or reset.

Error Codes Table

Error codes that appear in the error log are described in the following table.

Error code	Description	Detailed information		Appropriate response
		First byte (bits 08 to 15)	Second byte (bits 00 to 07)	
0001	PC watchdog timer error	00	00	Replace the PC’s CPU.
0002	PC service monitor error	Monitor time (s)		Check the operating environment.
0003	PC common RAM error	01: cyclic 02: event	00	Check the operating environment.
0005	Unit number error	Unit number setting	Unit number recorded in PC	Check the I/O table.
0009	System setup read error	FINS response code		Check the system setup for the Ethernet Unit.
000C	Routing table read error	FINS response code		Check the routing tables.

Error code	Description	Detailed information		Appropriate response
		First byte (bits 08 to 15)	Second byte (bits 00 to 07)	
0105	No transmission due to incorrect node number setting	Commands: Bits 00 to 07: Source node number Bits 08 to 14: Source network address Bit 15: OFF Responses: Bits 00 to 07: Destination node number Bits 08 to 14: Destination network address Bit 15: ON		Check the rotary switch settings on the front of the Unit. Be sure that the settings are within range and that the same number has not been assigned to more than one Unit.
0108	No transmission possible; unit does not exist			Check the unit number setting on the front of the Unit.
010B	No transmission possible; PC error			Refer to the *CV-series PC Operation Manual: Ladder Diagrams*.
010C	No transmission possible; unit number setting error			Check the rotary switch settings on the front of the Unit. Be sure that the settings are within range and that the same number has not been assigned to more than one Unit.
010D	No transmission possible; remote address setting error			Be sure that the remote node is properly set in the routing tables.
010E	No transmission possible; routing tables not set			Be sure that routing tables are properly set at the local node, target node, and any relay nodes.
010F	No transmission possible; routing table error			Check the routing tables.
0110	No transmission possible; too many relays			Reconfigure the network or change the routing tables. You can only communicate on three network levels.
0111	Received command size error			Check the command format and be sure that the correct command data is set. All zeros (0000) will be stored as the detailed information if the received packet size does not match the FINS header.
0113	No transmission possible; I/O setting error			Either correct the system configuration or the I/O table to agree with each other.
0114	No transmission possible; CPU bus error			Check Unit and cable connections and then refer to the *CV-series PC Operation Manual: Ladder Diagrams*.
0116	No transmission possible; CPU Bus Unit error			Check Unit and cable connections and then refer to the *CV-series PC Operation Manual: Ladder Diagrams*.
0117	Internal reception buffer full			Adjust the system to reduce traffic density.
0118	Illegal packet discarded			Check to be sure that there are no nodes sending illegal packets.

7-3-2 Time Stamp

The time stamp information is recorded sequentially from the first word, as shown in the following table.

Word	Bit	Data	Range
+0	0 to 7	Seconds	00 to 59 (BCD)
	8 to 15	Minutes	00 to 59 (BCD)
+1	0 to 7	Hours	00 to 23 (BCD): 24-hour clock
	8 to 15	Days	01 to 31 (BCD)
+2	0 to 7	Months	01 to 12 (BCD)
	8 to 15	Years	00 to 99 (BCD): Last 2 digits of year AD

7-4 Other RAS Features

The RAS features listed in this section are supported in addition to the features described in previous sections.

7-4-1 Self-diagnostic Features

Self-diagnostic testing is run automatically when the Ethernet Unit is turned on or restarted to check that the hardware is operating normally. Details of the tests and the actions when an error occurs are shown in the following table. The results of the self-diagnostic test are not stored in the error log.

Test details	Indicators when error occurs		Unit operation
	RUN	ERC	
Internal memory test	Not lit	Lit	Operation stops
Communications controller loopback test	Lit	Lit	Ethernet communications stop.
Loopback test to transceiver			

Note If the power supply is turned ON while the transceiver is not connected, the loopback test to transceiver will not be performed. Consequently, the ERC indicator will not light, and Ethernet communications functions will not stop.

7-4-2 Reading Network Information

The Ethernet Unit provides FINS commands which offer equivalent data reading functions to the NETSTAT commands which are supported as standard by a host computer equipped with TCP(UDP)/IP.

The FINS commands shown in the following table can be sent to the Ethernet Unit to read network information. Refer to the pages given in the table for more information.

Command code		Name	Function	Page
MRC	SRC			
27	62	PROTOCOL STATUS READ	Reads the status of each Ethernet protocol: IP, ICMP, TCP, UDP.	148
	63	MEMORY STATUS READ	Reads the memory status of the Ethernet Unit network memory.	153
	64	SOCKET STATUS READ	Reads the status of the Ethernet Unit network sockets: TCP, UDP.	154

SECTION 8
Maintenance and Inspections

This section describes the procedures and information required to maintain the Ethernet Unit in an Ethernet system, including periodic maintenance and inspections.

8-1 Maintenance

The Ethernet Unit makes up part of a network. Repair a defective Ethernet Unit as soon as possible as it can have a negative effect on the entire network. We recommend that customers keep one or more spare Ethernet Units to allow immediate recovery of the network.

8-1-1 Replacing an Ethernet Unit

Observe the following precautions when replacing the Ethernet Unit.

- Always turn off the power supply before replacing the Ethernet Unit.
- Check that the spare Ethernet Unit is operating normally before replacing a defective Unit with it.
- When returning a defective Unit for repairs, provide as much written information as possible on the symptoms of the problem.
- If a problem occurs with poor contacts, wipe the contacts with a clean cloth soaked with industrial alcohol. Carefully remove any lint remaining on the contacts before replacing the Unit.

8-1-2 Settings After Replacing a CPU

The EEPROM in the PC's CPU holds the information listed below. This information must be stored in any new CPU used to replace a defective one.

- Routing tables
- System Setup for the Ethernet Unit

8-2 Inspections

Carry out regular inspections to ensure the Ethernet Unit is functioning perfectly.

8-2-1 Items

Most of the parts that make up an Ethernet Unit are semiconductor components. None of the parts in the Unit will wear out after a specific lifetime, but some parts may deteriorate due to extreme operating condition. Therefore, it is important to inspect the Unit regularly.

Inspection Interval

Normally inspect once or twice per year. Choose the inspection period according to the severity of the operating conditions. New installation should be inspected more frequently until they are judged to be stable.

Inspection Items

Correct any of the items in the table below not conforming to the specified standard.

Item	Details	Standard
Environment	Temperature around Unit	0°C to 55°C
	Humidity around Unit	10% to 90% (with no condensation)
	Accumulated dust	No accumulated dust
Mounting	Ethernet Unit firmly attached	No looseness
	Transceiver cable connector fully pushed in	No looseness
	Condition of transceiver cable	No visible abnormality

8-2-2 Tools Required for Inspection

The following tools are needed to inspect the Ethernet Unit:

Standard Tools

Flat-blade and Phillips screwdrivers
Industrial alcohol and a clean cloth
Tester or digital voltmeter

Tools Required Under Special Circumstances

Synchroscope
Thermometer and hygrometer
Pen oscilloscope

SECTION 9
Troubleshooting

This section describes errors that can occur when using the Ethernet Unit and what to do about them. This section does not deal with errors that can occur in general PC operation. Refer to the *CV-series PC Operation Manual: Ladder Diagrams* for general troubleshooting information.

9-1 Unit Indicators

The indicators on the Ethernet Unit can indicate what type of error has occurred. Refer to the following table.

Indicator	Probable cause	Remedy
RUN indicator not lit	Power not supplied to PC or voltage is too low.	Supply power or increase the voltage.
	Defective Ethernet Unit.	Replace the Ethernet Unit.
	Loose Ethernet Unit mounting screws.	Fully tighten the Ethernet Unit mounting screws.
	Ethernet Unit mounted in an incorrect slot.	Mount the Ethernet Unit correctly. Refer to page 12.
ERH indicator lit	Incorrect unit number setting on the rotary switches.	Correct the unit number setting on the rotary switches. Refer to 10.
	I/O table not correctly set up.	Correctly set up the I/O table.
	A watchdog timer (WDT) error occurred in the PC.	Determine and remedy the PC error.
	Defective EEPROM in the PC.	Store the routing tables and network parameters again or replace the PC.
ERH indicator flashing	Incorrect IP address in the System Setup for the Ethernet Unit.	Correct the IP address. Refer to page 24.
ERC indicator lit	Incorrect node number setting on the rotary switches.	Correct the node number setting on the rotary switches. Refer to page 10.
	Internal Ethernet Unit error (RUN indicator not lit).	Replace the Ethernet Unit.
	An error occurred in the transceiver loopback test during the self-diagnostic test. Refer to page 164.	Check the transceiver and transceiver cable.

9-2 Troubleshooting Procedures

The following procedures can be used to troubleshoot various problems in system operation.

Most of the initial steps in these procedures are in question form. Continue in sequence until a “Yes” answer tells you to jump to a specified step or until you are told to perform a specific action that corrects the problem. If performing the action does not correct the problem, return to the beginning of the procedure and start over.

9-2-1 Startup Problems

1, 2, 3...

1. RUN indicator lit?
   True ➧ Step 9.
2. ERH indicator lit?
   True ➧ Step 12.
3. ERC indicator lit?
   True ➧ Step 10.
4. Power not supplied to CPU?
   True ➧ Make sure that sufficient power is supplied to the CPU.
5. Ethernet Unit loose on Rack?
   True ➧ Make sure that Unit is firmly mounted and tighten the mounting screws firmly.

6. If the Ethernet Unit mounted in an incorrect slot on the Rack?

   True ➧ Mount the Unit to a slot that supports CPU Bus Units.

7. Do the RUN and ERH indicators not light if the Ethernet Unit is mounted to another CPU Rack?

   True ➧ Replace the Ethernet Unit.

8. Replace the CPU on the Rack where the indicators didn't light.
9. ERC indicator not lit?

   True ➧ Step 15.

10. Is the node number out of range (should be set between 1 and 126)?

    True ➧ Set the node number to a value between 1 and 126 not used by another node on the network.

11. Replace the following components one at a time in the order given and discard any faulty ones: transceiver cable, transceiver, Ethernet Unit.
12. Is the unit number out of range (should be set between 00 and 15)?

    True ➧ Set the unit number to a value between 00 and 15 not used by another unit on the PC.

13. Is the same unit number being used by another unit on the PC?

    True ➧ Set the unit number to a number not used by any other unit on the same PC.

14. Create an I/O table in the PC.
15. ERH indicator not flashing?

    True ➧ Step 18.

16. Is the System Setup for the Ethernet Unit not complete?

    True ➧ Complete setting the parameters in the System Setup.

17. Is the IP address incorrect?

    True ➧ Correct the IP address in the System Setup, and read out the controller status using the READ CONTROLLER STATUS command in FINS communications and correct any problems indicated.

18. ERH indicator not lit?

    True ➧ Go to *9-2-6 Network Connection Problems* on page 174.

19. Are the routing tables not complete?

    True ➧ Set appropriate routing tables, and read out the controller status using the READ CONTROLLER STATUS command in FINS communications and correct any problems indicated.

20. Has a CPU error occurred in the PC's CPU?

    True ➧ Restart the CPU. If the error persists, replace the CPU.

9-2-2 FINS Communications Problems (SEND(192)/RECV(193)/CMND(194))

1, 2, 3...

1. Refer to *9-3 Troubleshooting via Response Codes* on page 175 and remove any causes of the problem discovered there before proceeding with this procedure.

2. Is the control data for the instruction set incorrectly?

   True ➧ Make sure that the FINS network address is not set to 0 for the Ethernet Unit and check the network address, node number and unit address.

3. Is a CMND(194) instruction being addressed to the local node, e.g., for socket services?

   True ➧ Go to *9-2-6 Network Connection Problems* on page 174.

4. Are different UDP port numbers set for FINS communications for both the local node and the remote node?

   True ➧ Set the same port number for both nodes.

5. Are the local and remote nodes both on the same network, i.e., do they have the same network number in their IP addresses?

   True ➧ Step 10.

6. Is the IP address of the remote node missing from the IP address table?

   True ➧ Set the IP address of the remote node in the local IP address table. (Note: Use of the IP address table must be set in the mode settings for the IP address table to be effective.)

7. Is the IP address of the remote node missing from the IP router table?

   True ➧ Set the network number of the remote node in the IP router table.

8. Are the FINS network addresses of the local and remote nodes different?

   True ➧ Set both nodes to the same FINS network address.

9. Are you attempting a broadcast transmission?

   True ➧ Do not attempt to broadcast to a node with a different network number. You can broadcast only to nodes on the same IP network.

10. Set the IP address of the remote node in the IP address table or use automatic address conversion.
11. Are you attempting to use a gateway PC to communicate with a PC on another network?

    True ➧ Go to *9-2-6 Network Connection Problems* on page 174.

12. Are the routing tables set improperly?

    True ➧ Set routing tables at the local node, target node, and any relay nodes.

9-2-3 UDP Socket Problems

General Problems

1, 2, 3...

1. Go through the procedure in *9-2-2 FINS Communications Problems (SEND(192)/RECV(193)/CMND(194))* before starting this procedure.
2. Is the response code a value other than 0000?

   True ➧ Go to *9-3 Troubleshooting via Response Codes* on page 175.

3. Is the response code in the results storage area a value other than 0000?

   True ➧ Go to *9-4 Results Storage Area Response Codes* on page 177.

4. Go to *9-2-6 Network Connection Problems* on page 174.

Opening and Closing Problems

Refer to *General Problems* on page 170.

Reception Problems

1, 2, 3...

1. Is reception processing finishing?

   True ➧ Step 10.

2. Is the remote node not processing to send data?

   True ➧ Adjust the remote node to process send data.

3. Read controller status using the FINS command READ CONTROLLER STATUS.
4. Have any reception errors occurred?

   True ➧ There may be noise affecting the network. Increase the number of retries or take measures against noise as described in *Section 2***.**

5. Read memory status using the FINS command READ MEMORY STATUS.
6. Is the network memory being used more than 80%?

   True ➧ There may be too much load on the Ethernet Unit. If the memory exhaustion records show counts, UDP data may be corrupted. Check your applications.

7. Read protocol status using the FINS command READ PROTOCOL STATUS.
8. Are any of the first three items in the status being counted?

   True ➧ The UDP protocol of the remote node may not be compatible with that of the Ethernet Unit. Use another communications service.

9. Check the remote node for transmission problems using the next procedure. If nothing is uncovered, go to *9-2-6 Network Connection Problems* on page 174.
10. Return to the procedure for general problems.

Transmission Problems

1, 2, 3...

1. Is send processing not finishing?

   True ➧ Return to the procedure for general problems.

2. Read protocol status using the FINS command READ PROTOCOL STATUS.
3. Have any failures to reach the destination been counted in the ICMP status?

   True ➧ The UDP socket specified by the destination UDP port number in the send parameters is not opening at the remote node.

4. Read controller status using the FINS command READ CONTROLLER STATUS.
5. Have any send errors occurred?

   True ➧ There may be too much traffic on the network and UDP packets may be getting corrupted. Use TCP sockets or adjust the network to reduce excessive traffic.

6. Check the reception status on the remote node.
7. Return to the procedure for general problems.

9-2-4 TCP Socket Problems

General Problems

1, 2, 3...

1. Go through the procedure in *9-2-2 FINS Communications Problems (SEND(192)/RECV(193)/CMND(194))* before starting this procedure.
2. Is the response code a value other than 0000?

   True ➧ Go to *9-3 Troubleshooting via Response Codes* on page 175.
3. Is the response code in the results storage area a value other than 0000?

   True ➧ Go to *9-4 Results Storage Area Response Codes* on page 177.
4. Read controller status using the FINS command READ CONTROLLER STATUS.
5. Have any send collisions occurred?

   True ➧ There may be too much traffic going on the network. Reduce network traffic.
6. Go to *9-2-6 Network Connection Problems* on page 174.

Opening Problems

1, 2, 3...

1. Are you attempting an active open?

   True ➧ Step 3.
2. Is the passive open not finishing?

   True ➧ Use an active open at the remote node.
3. Is the response code in the results storage area a value other than 0049?

   True ➧ Step 6.
4. Read socket status using the FINS command READ SOCKET STATUS.
5. Does the port number being used exist at the local node?

   True ➧ Make sure that the same port number is not being used simultaneously by more than one process, including FTP services.
   Sockets can remain in ESTABLISHED or some later status even if closed; be sure to close ports from both sides of the connection.
   A socket can remain open in TIME WAIT status for up to one minute on the side that closes the socket first; we recommend that the active-side port number be set to 0 and that you close the socket from the active side.
6. Return to *General Problems* on page 172.

Closing Problems

Refer to *General Problems* on page 172.

Reception Problems

1, 2, 3...

1. Is reception processing not finishing?

   True ➧ Step 6.
2. Is the remote node not processing to send data?

   True ➧ Adjust the remote node to process send data.
3. Read memory status using the FINS command READ MEMORY STATUS.
4. Is the network memory being used more than 80%?

   True ➧ There may be too much load on the Ethernet Unit. If the memory exhaustion records show counts, processing may be slow. Check your applications.
5. Return to the procedure for general problems.

6. Is the response code in the results storage area a value other than 0000?
   True ➧ Return to the procedure for general problems.
7. Is the number of bytes received in the results storage area 0?
   True ➧ The remote node's TCP socket has closed. Close the local socket.

Transmission Problems

1, 2, 3...

1. Is send processing finishing?
   True ➧ Return to the procedure for general problems.
2. Read socket status using the FINS command READ SOCKET STATUS.
3. Is there the maximum number of bytes in the send queue (4,096 bytes)?
   True ➧ Make sure that reception processing is being performed at the remote note.

9-2-5 FTP Service Problems

Connection Problems

1, 2, 3...

1. Are you unable to connect to the Ethernet Unit's FTP server?
   True ➧ Step 5.
2. If the FTP indicator lit?
   True ➧ The FTP server is connected to another client. Wait until the client has finished.
3. Are the parameter settings in the host computer incorrect?
   True ➧ Correct the computer settings. Refer to the documentation for your computer.
4. Go to *9-2-6 Network Connection Problems* on page 174.
5. Are you unable to log in?
   True ➧ Check the login name and password settings.
6. Can you display the MEMCARD directory using `ls` from the host computer.
   True ➧ End.
7. Is a Memory Card not inserted in the PC?
   True ➧ Insert a Memory Card.
8. Is power not supplied to the Memory Card?
   True ➧ Turn on the power supply switch for the Memory Card.
9. Has the Memory Card not been initialized?
   True ➧ Initialize the Memory Card using the CVSS.
10. Is the Memory Card faulty?
    True ➧ Check the backup battery for RAM Cards and replace it if necessary. Replace the Memory Card if necessary.

File Transfer Problems

1, 2, 3...

1. Are you not connected to the FTP server?
   True ➧ Return to the previous procedure.
2. Is the current directory not MEMCARD?
   True ➧ Make MEMCARD the current directory (execute `cd /MEMCARD`).
3. Can you execute `get`?
   True ➧ Step 6.

4. Is the file you are attempting to get not actually on the Memory Card?
   True ➧ Designate a file actually on the Card.
5. Return to *9-2-1 Startup Problems* on page 168.
6. Can you execute `put`?
   True ➧ Step 11.
7. Is an EPROM Memory Card mounted?
   True ➧ You cannot write to EPROM Memory Cards mounted in the PC. Use a Memory Card Writer.
8. Is an EEPROM Memory Card mounted?
   True ➧ Make sure the PC is in PROGRAM mode and that the version of the CPU is V1 or later.
9. Is there insufficient free space available on the Memory Card?
   True ➧ Delete unneeded files or use a different Memory Card.
10. Return to *9-2-1 Startup Problems* on page 168.
11. Is the data in the file transferred with `put` or `get` normal (i.e., not corrupted)?
    True ➧ End
12. Use the `type` command and change the file data type to binary.

9-2-6 Network Connection Problems

1, 2, 3...

1. Go through the procedure in *9-2-1 Startup Problems* before starting this procedure.
2. Is the power supply indicator lit on the transceiver?
   True ➧ Step 5.
3. Is the transceiver cable loose?
   True ➧ Connect the cable firmly.
4. Replace the following components one at a time in the order given and discard any faulty ones: transceiver cable, transceiver, Ethernet Unit.
5. Execute `ping` at each node for all nodes on the network to see if communications are possible.
6. Was not even one response received at every node in the network?
   True ➧ Check terminators, the coaxial cable, and transceiver cables.
7. Were responses not received only between certain nodes?
   True ➧ Make sure the distance between transceivers on the coaxial cable is a multiple of 2.5 m.
   Check IP address settings.
   Make sure that the remote node supports ICMP.
8. Is an FTP client not mounted on the remote node?
   True ➧ Mount an FTP client.
9. Are you using FINS communications (e.g., SEND(192), RECV(193), or CMND(194))?
   True ➧ Do an internode test.
10. Read protocol status using the FINS command READ PROTOCOL STATUS and check the following parameters in the returned status data. If any

of these items have actually been counted, the remote node may not support the relevant service.

IP (all communications): Items 2 through 6
ICMP (PING): Items 5 through 8
TCP (FTP and TCP sockets): Items 4 through 6 under reception status
UDP (FINS and UDP sockets): Items 1 through 3

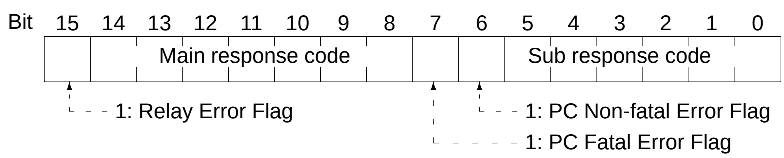
If item 4, 8, or 9 under the IP status is being counted, there may be too much traffic through the Ethernet Unit. Check your user applications.

9-3 Troubleshooting via Response Codes

You can troubleshoot the errors in FINS communications from the response codes when the SEND(192), RECV(193), or CMND(194) instructions have been used. For the storage areas of the response codes refer to information beginning on page 29.

The table below lists response codes (main and sub-codes) returned after execution of FINS commands, the probable cause of errors, and recommended remedies. Refer to the *FINS Command Reference Manual* for further information on response codes.

The 6th, 7th, and 15th bits of the response codes have specific functions. The 6th bit will be ON when a non-fatal error has occurred in the PC at the remote node; the 7th bit will be ON when a fatal error has occurred in the PC at the remote node; and the 15th bit will be ON when a network relay error has occurred. Refer to information given after the following table for more information on relay errors.



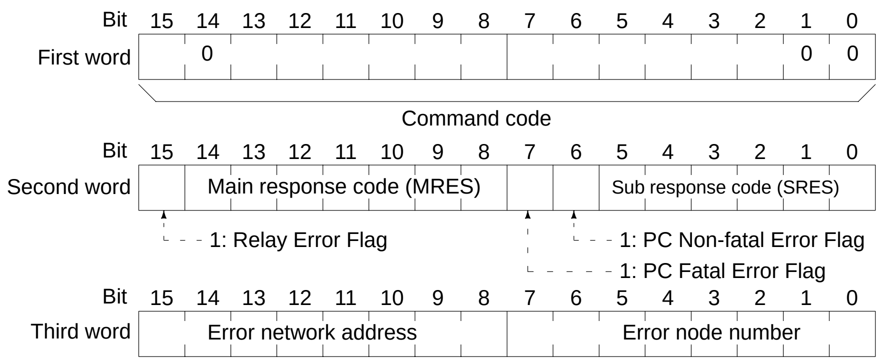
Main code	Sub-code	Probable cause	Remedy
00: Normal completion	00	---	---
01: Local node error	03	Send error in local node was caused by lack of available space in internal buffers.	Reduce the load (traffic) on the Ethernet Unit. Check your user applications.
02: Remote node error	01	IP address of remote node not set correctly in IP address table or IP router table.	Set IP address of remote node into IP address table and, if internetwork transmission is required, into the IP router table.
	02	No node with the specified unit address.	Check the remote node’s unit address and make sure the correct one is being used in the control data.
	05	Message packet was corrupted by transmission error.	Check the protocol and controller status by reading them with FINS commands. Increase the number of transmit retry attempts.
		Response time-out, response watchdog timer interval too short.	Increase the value for the response monitor time in the control data.
		The transmission frame may be corrupted or the internal reception buffer full.	Read out the error log and correct as required.
03: Communications controller error	01	Error occurred in the communications controller, ERC indicator is lit.	Take corrective action, referring to troubleshooting procedures in this section.
	02	CPU error occurred in the PC at the remote node.	Check CPU indicators at the remote node and clear the error in the CPU (refer to the PC’s operation manuals)
	04	Unit number setting error	Make sure the unit number is specified within range and that the same unit number is not used twice in the same network.

Main code	Sub-code	Probable cause	Remedy
04: Not executable (service not supported)	01	An undefined command has been used.	Check the command code and be sure the command is supported by the Unit to which you are sending it.
		A short frame (4 bytes) is being used for the FINS header frame.	Check the FINS header frame length. The Ethernet Unit does not support short headers.
05: Routing error	01	Remote node is not set in the routing tables.	Set the remote node in the routing tables. Refer to *Section 3 FINS Communications Service*.
	02	Routing tables aren't registered completely.	Set routing tables at the local node, remote node, and any relay nodes.
	03	Routing table error	Set the routing tables correctly.
	04	The maximum number of relay nodes (2) was exceeded in the command.	Redesign the network or reconsider the routing table to reduce the number of relay nodes in the command. Communications are possible on three network levels, including the local network.
10: Command format error	01	The command is longer than the max. permissible length.	Check the command format of the command and set it correctly. Be sure broadcast transmissions don't exceed 1,473 bytes.
	02	The command is shorter than min. permissible length.	Check the command format of the command and set it correctly.
	03	The designated number of data items differs from the actual number in the command data.	Check the number of items and the data, and make sure that they agree.
	05	Data for another node on the same network was received from the network.	Check the header parameters in the command data and be sure the correct command format is being used.
		An attempt was made to send response data for a broadcast address.	
11: Parameter error	00	The parameters in the command data were incorrect or the UDP/TCP socket number was not within the proper range.	Check the parameters and be sure the socket number is between 1 and 8.
	01	A correct memory area code has not been used or Expansion Data Memory is not available.	Check the command's memory area code and set the appropriate code.
	03	The first word is in an inaccessible area or the bit number is not 00.	Set a first word that is in an accessible area. The bit number must be 00 for Ethernet Units.
22: Operation mode mismatch (Status error)	0F	The same socket service is already in progress at the specified socket number.	Use the socket status flag in PC memory to be sure that socket service has finished before starting services again.
	10	The specified socket is not open.	Open the socket. For TCP sockets, be sure to wait until connection is made.
23: No such Unit (Environment error)	05	IP address conversion failed.	Check the IP address and subnet mask in the System Setup and be sure that settings are correct.
	07	IP address conversion is set for automatic conversion only.	Check the mode settings in the System Setup. This error will be generated for the READ IP ADDRESS command only.
25: Unit error	03	I/O setting error (The I/O table differs from the actual Unit configuration.)	Either change the actual configuration to match the registered one, or generate the I/O table again.
	05	CPU bus error (An error occurred during data transfer between the CPU and a CPU Bus Unit.)	Check the Units and cable connections and issue the ERROR CLEAR command.
	0A	An error occurred during CPU Bus Unit data transfer.	Check the Units and cable connections and issue the ERROR CLEAR command.

Network Relay Errors

For network relay errors using SEND(192) or RECV(193), check the path of the command using the routing tables and the nature of the error using the response code to eliminate the cause of the error.

For network relay errors using CMND(194), the location of the relay error is recorded in the second through third words of the response, as shown below.



9-4 Results Storage Area Response Codes

The response codes stored in the results storage area can be used to troubleshoot socket service problems. Refer to page 127 for the format used for socket applications.

The UNIX socket service error messages corresponding to the response codes are give in the following table. Refer to the documentation for the devices involved when communicating between an Ethernet Unit and other devices.

Response code	UNIX error message	Description	Probable remedy
0003	ESRCH	No such process	Close the local socket and try reopening it.
0006	ENXIO	No such device or address	
0009	EBADF	Bad file number (incorrect socket specification)	
000D	EACCES	Permission denied (Broadcast address specified for remote IP address for active TCP open)	Check the IP address of the remote node and try to reconnect.
000E	EFAULT	Bad address (copy failed between kernel and user area)	Close the local socket and try reopening it.
0011	EEXIST	File exists	
0016	EINVAL	Invalid argument (socket library argument error)	
0018	EMFILE	Too many open files (More than 32 sockets)	
0020	EPIPE	Broken pipe (remote node closed socket)	Close the local socket.
003C	EPROTONOSUPPORT	Protocol not supported (protocol other than UDP, TCP, or RAW specified)	Close the local socket and try reopening it.
003D	EPROTOTYPE	Protocol wrong type for socket	
003E	ENOBUFS	No buffer space available	There is too much load (traffic) on the Ethernet Unit. Check your user applications.

Response code	UNIX error message	Description	Probable remedy
003F	EISCONN	Socket is already connected (connection attempted to open socket)	Close the local socket and try reopening it.
0040	ENOTCONN	Socket is not connected (send attempted to closed socket)	
0041	EALREADY	Operation already in progress (connection attempted to existing non-block connection)	
0042	EMSGSIZE	Message too long	Check the length of send data. UDP or TCP: 1 to 1,982 bytes UDP broadcasts: 1 to 1,472 bytes
0043	EDESTADDRREQ	Destination address required (destination address not specified)	Close the local socket and try reopening it.
0044	ENOPROTOOPT	Protocol not available (unsupported option specified)	
0045	ECONNABORTED	Software caused connection abort (another task closed socket)	
0046	EINPROGRESS	Operation now in progress (non-block connection ended during processing)	
0047	ENOTSOCK	Socket operation on non-socket	
0048	EOPNOTSUPP	Operation not supported on socket	
0049	EADDRINUSE	Address already in use (UDP or TCP open request sent for port already in use)	Check the port number. TCP ports can remain unusable for 1 min after closing.
004A	ECONNREFUSED	Connection refused (TCP socket (active open) processing refused by remote node)	Passively open a remote TCP socket, checking the remote IP address and remote TCP port number.
004B	ECONNRESET	Connection reset by peer (TCP socket closed by remote node)	Close the local socket and try reconnecting.
004C	EADDRNOTAVAIL	Can't assign requested address (mistake in remote IP address)	Check the setting of the remote IP address and try reconnecting.
004D	EAFNOSUPPORT	Address family not supported by protocol family	Close the local socket and try reopening it.
004E	ENETUNREACH	Network is unreachable	Set the path to the remote node in the IP router table.
004F	EHOSTDOWN	Host is down	Check the remote host and communications path.
0050	EWOULDBLOCK	Operation would block	Close the local socket and try reopening it.
0051	EHOSTUNREACH	No route to host	The specified node does not exist on the designated IP network segment. Check the communications path.
0053	ETIMEDOUT	Connection timed out (TCP timed out)	Check the remote host and communications path.
0063	ESELABORT	Used for internal Ethernet Unit processing	Close the local socket and try reopening it.
0080	(None)	Timed out for passive TCP open request	Either the remote node is not executing an active TCP open or there is a block on the network.
0081	(None)	Closed by close command during socket servicing	No action is necessarily called for.
0082	(None)	Connection with remote node not achieved for passive TCP open request	The remote IP address and TCP port number settings differ from those of the remote socket (active side).

Appendix A
Standard Models

Name	Specifications			Model
Ethernet Unit	See *Appendix B specifications.*			CV500-ETN01
CV-series CPUs	V1 CPUs	CV500	Ladder or SFC + Ladder	CV500-CPU01-EV1
		CV1000	Ladder or SFC + Ladder	CV1000-CPU01-EV1
		CV2000	Ladder or SFC + Ladder	CV2000-CPU01-EV1
		CVM1	Ladder only	CVM1-CPU11-EV1
				CVM1-CPU01-EV1
Memory Cards	RAM		64K bytes	HMC-ES641
			128K bytes	HMC-ES151
			256K bytes	HMC-ES251
			512K bytes	HMC-ES551
	EEPROM		64K bytes	HMC-EE641
	(Requires CV500-MCW01 to write except in PROGRAM mode on V1 CPUs.)		128K bytes	HMC-EE151
	EPROM		512 bytes	HMC-EP551
	(Requires CV500-MCW01 to write.)		1M bytes	HMC-EP161
Transceivers Transceiver cables Coaxial cables Terminators	All Ethernet system components must comply with IEEE802.3 standards.			

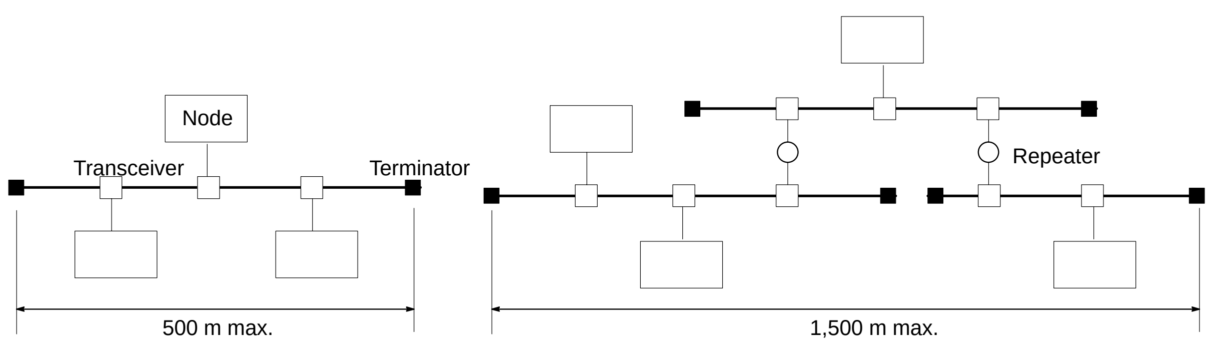
Appendix B
Specifications

Specifications

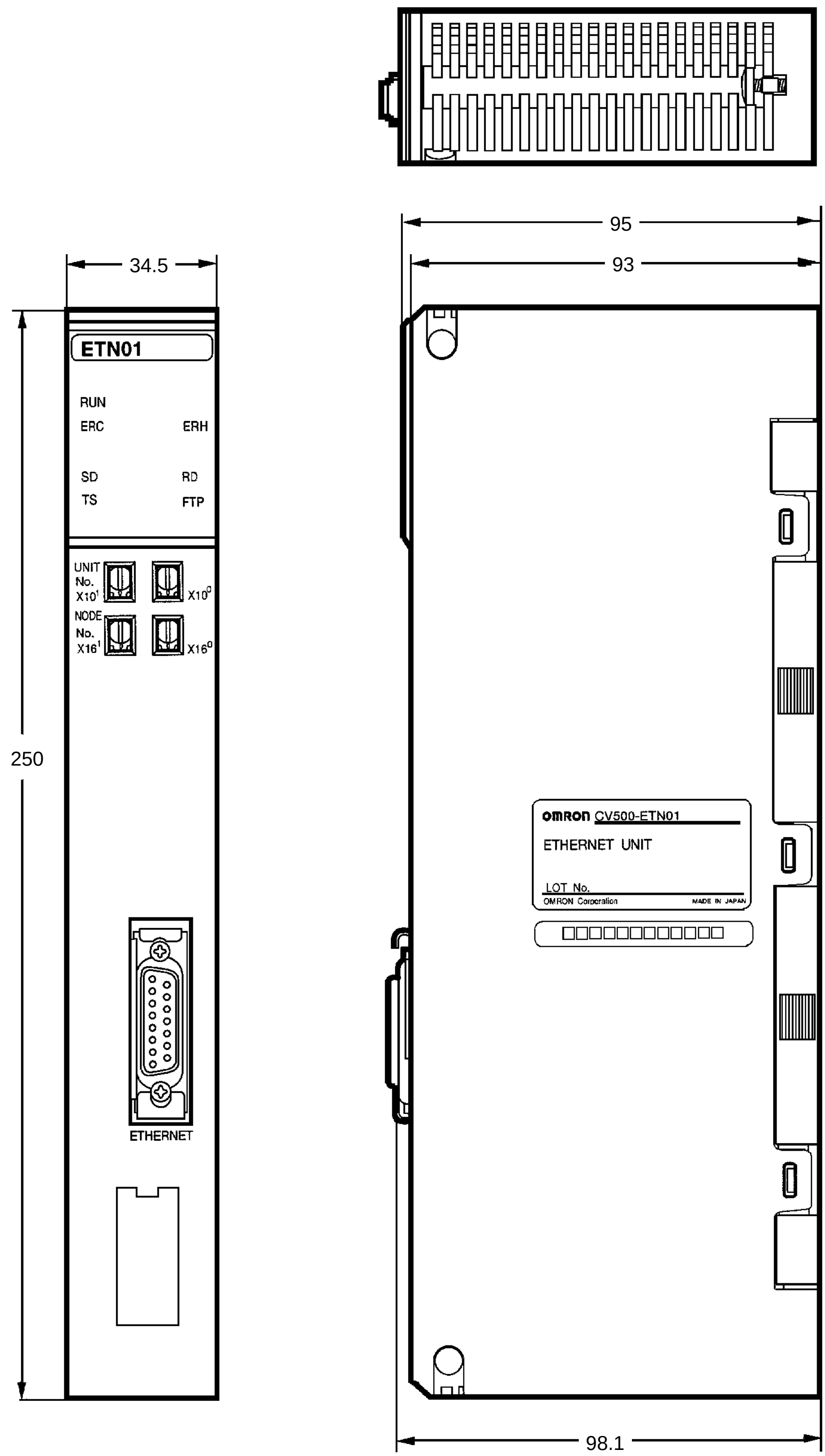
The specifications of the Ethernet Unit are given in the following table. Other specifications are the same as those for other CV-series Units.

Item	Specification
Model number	CV500-ETN01
Media access method	CSMA/CD
Modulation	Baseband
Transmission paths	Bus type
Baud rate	10 Mbps
Transmission media	Coaxial cable
Maximum transmission distance	500 m/network (segment); 2.5 km/system
Maximum number of nodes	100/network (segment)
Distance between nodes	Integral multiples of 2.5 m
Maximum transceiver cable length	50 m max.
Transceiver supply voltage capacity	0.35 A at 12 V
Communications services	FINS FTP server Socket
Communications buffers	See *Appendix F Buffer Configuration.*
RAS features	PING command/response (loopback test via ICMP) Internode test Error log Self diagnostics (hardware) Network status readout
Maximum current consumption	1.7 A max.
Dimensions (see following diagram)	34.5 x 250 x 98.1 mm (W x H x D)
Weight	500 g

Note Ethernet networks are divided into units called segments, in which each segment is one length of the transmission media (coaxial cable). The transmission distances for a network can be illustrated using the segment configuration diagrams, as shown below.



Dimensions (Unit: mm)



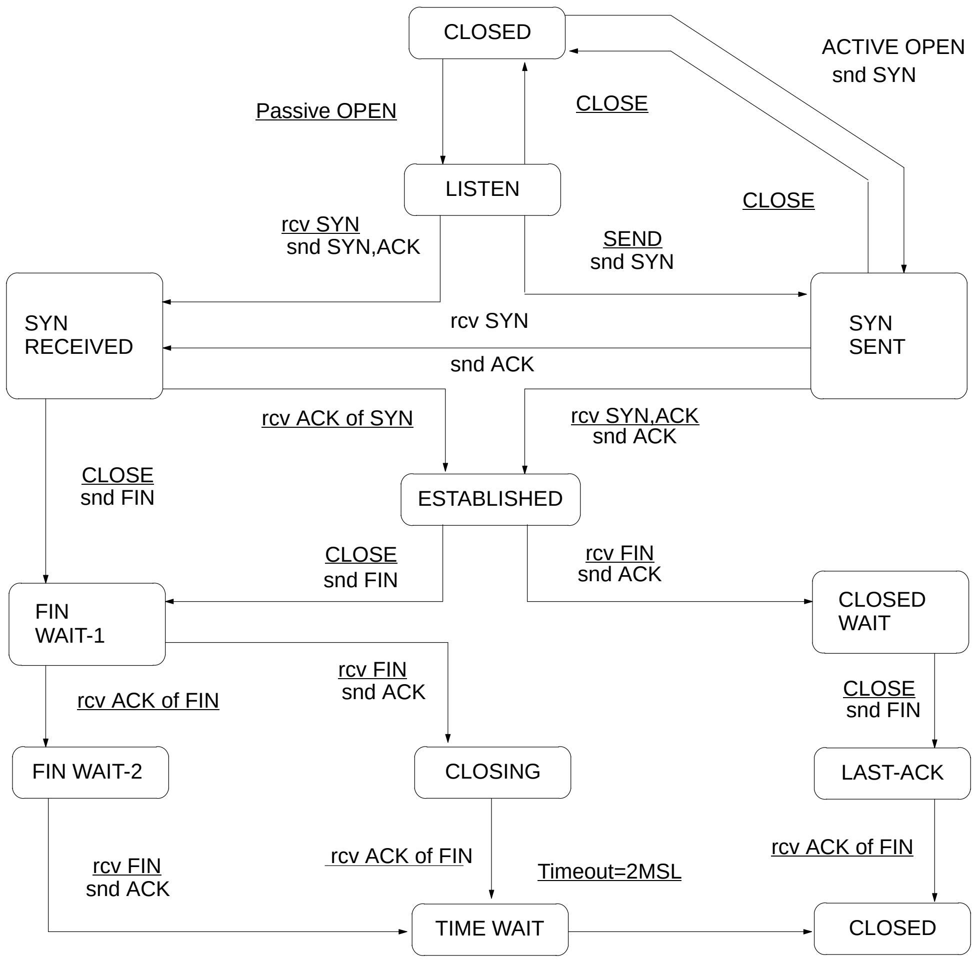
Weight: 500 g

Appendix C
Ethernet Network Parameters

Parameter	Value	Description
TCP send buffer	4,096 bytes	Maximum capacity of the TCP send buffer
TCP receive buffer	4,096 bytes	Maximum capacity of the TCP receive buffer
UDP send buffer	9,000 bytes	Maximum capacity of the UDP send buffer
UDP receive buffer	9,016 bytes	Maximum capacity of the UDP receive buffer
RAW send buffer	2,048 bytes	Maximum capacity of the RAW send buffer
RAW receive buffer	2,048 bytes	Maximum capacity of the RAW receive buffer
FINS receive buffer	16,383 bytes	Maximum capacity of the FINS receive buffer
Hold timer	75 s (12 min max.)	The hold timer is used for processing confirmation of arrival of TCP sockets or the KEEP ALIVE option. Arrival confirmation is processed when connecting TCP sockets or transmitting data. When the KEEP ALIVE option is set and data transmission does not take place, confirmation of the connection is periodically performed. If no response is received for confirmation, confirmation processing is executed every 75 s. The connection is broken if this status continues for 12 min (and ETIMEDOUT is stored as the response in the results storage area). Although TCP socket services do not support the KEEP ALIVE option, they will response to KEEP ALIVE options sent from other nodes.
Resend timer	Initial value: 1 s Maximum value: 64 s	The resend timer is used to monitor completion of reception of arrival confirmations when transferring data via socket services, including FTP server TCP sockets. If the timer setting is exceeded before arrival confirmation is received, data is resent and timer operation is restarted with a longer time setting. This procedure is repeated up to 12 times (final timer value: 64 s). The hold timer is started after the 12th timeout.
Continue timer	Initial value: 5 s Maximum value: 60 s	The continue timer starts if preparations have been completed to send data but the send window is too small (either 0 or too small) to send the data and the remote node has not requested that communications be restarted. Confirmation of the window size is requested from the remote node when the continue timer times out. The initial value of the timer is 5 s and confirmation processing will continue consecutively with increasingly longer times until the maximum time of 60 s is reached, at which point the hold timer will be started.
2MSL timer	60 s	The 2MSL timer starts at the TCP socket that first closes the socket and will run for 60 s in the TIME_WAIT status.
IP reassemble timer	12 s	A fragmented IP packet is discarded if it cannot be reassembled within 12 seconds.
ARP timer	20 min/3 min	If a complete ARP table entry (with an Ethernet address) is not referred to for 20 minutes, it is removed from the table. An incomplete ARP table entry (no response yet returned to the ARP request) is removed from the table after 3 minutes.
Window size	4,096 bytes	The initial value of the maximum capacity used to control the convergence of TCP sockets. Actually, the node negotiates with the remote node and uses the smaller of the values for the two nodes. The window size will fluctuate with the available space in the TCP reception buffers of the remote node when processing communications.
Fragment size	1,500 bytes	UDP data is separated into 1,472-byte fragments. The remaining 28 bytes are for the IP header.
Segment size	1,024 bytes	TCP data is separated into 1,024-byte segments.
TTL (Time to Live)	30	Decremented each time an IP router is passed.

Appendix D
TCP Status Transitions

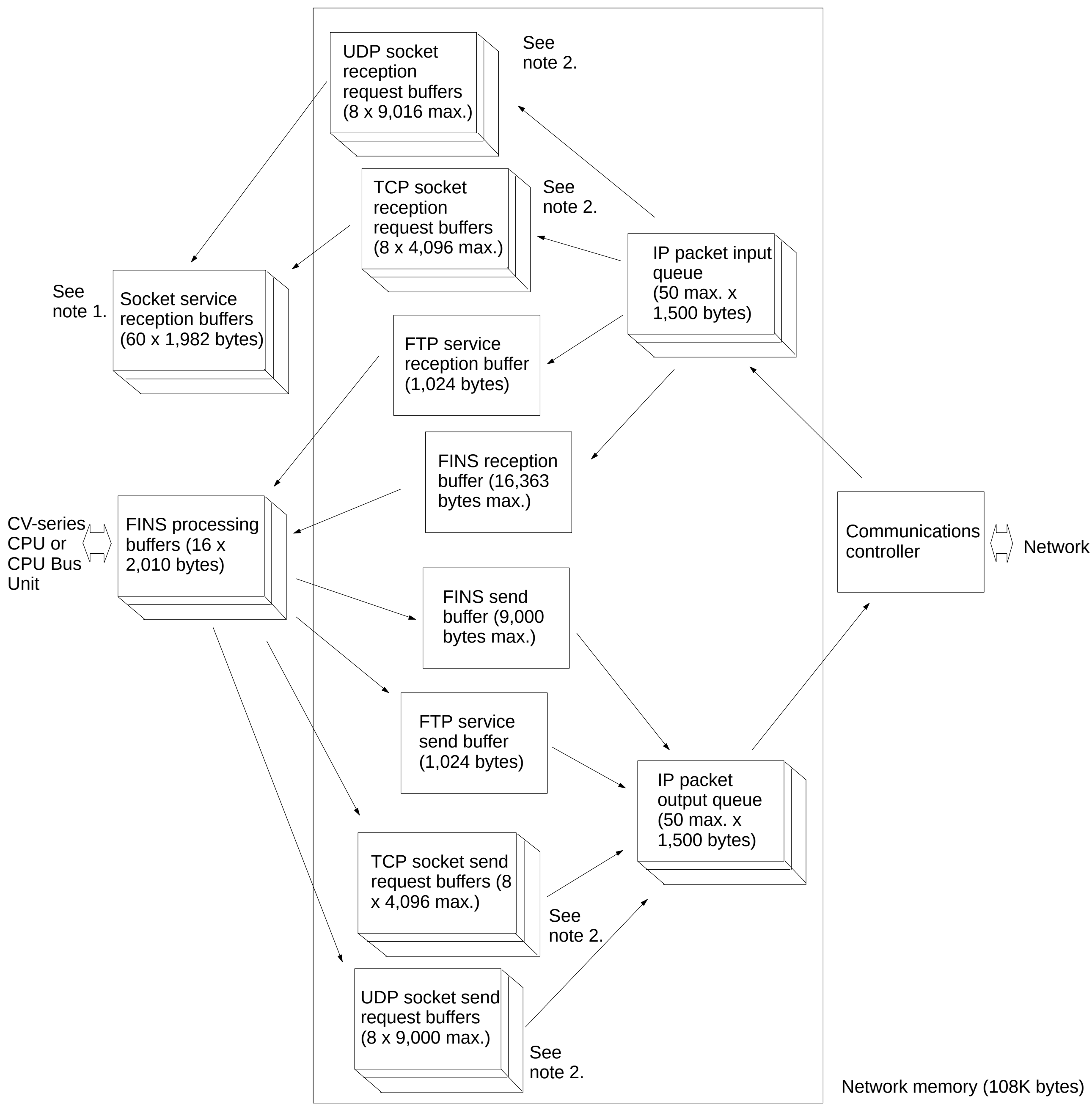
The following diagram shows transitions that take place in the TCP socket status according to the socket status data returned for the FINS command SOCKET STATUS READ (2764).



Appendix E
ASCII Characters

Bits 1 to 4		Bits 5 to 7							
Binary		**0000**	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**
	Hex	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0000	**0**	NUL	DLE	Space	0	@	P		p
0001	**1**	SOH	DC$_1$	!	1	A	Q	a	q
0010	**2**	STX	DC$_2$	”	2	B	R	b	r
0011	**3**	ETX	DC$_3$	#	3	C	S	c	s
0100	**4**	EOT	DC$_4$	$	4	D	T	d	t
0101	**5**	ENQ	NAK	%	5	E	U	e	u
0110	**6**	ACK	SYN	&	6	F	V	f	v
0111	**7**	BEL	ETB	’	7	G	W	g	w
1000	**8**	BS	CAN	(	8	H	X	h	x
1001	**9**	HT	EM	)	9	I	Y	i	y
1010	**A**	LF	SUB	*	:	J	Z	j	z
1011	**B**	VT	ESC	+	;	K	[	k	{
1100	**C**	FF	FS	,	<	L	\	l	|
1101	**D**	CR	GS	-	=	M	]	m	}
1110	**E**	SO	RS	.	>	N	^	n	~
1111	**F**	SI	US	/	?	O	_	o	DEL

Appendix F
Buffer Configuration



Network Memory

Most of the buffers used for communications servicing by the Ethernet Unit are administered in a buffer configuration called network memory. Network memory consists of 108K bytes of memory divided into short and long buffers. The use of short and long buffers is determined by the status of the various services when the Ethernet Unit is running. The capacity of all buffers cannot be used due to limits in the mounted memory capacity. The status of the short and long buffers can be accessed by execution the FINS command MEMORY STATUS READ (2763).

Note 1. Reception buffers for socket services and buffers for FINS processing are always secured regardless of the current operating status.

2. The status of UDP and TCP socket reception and send buffers can be accessed by executing the FINS command SOCKET STATUS READ (2764).

Appendix G
Auxiliary Area Data

The following table and descriptions cover the words and bits in the Auxiliary Area of PC memory that are related to the Ethernet Unit.

Word(s)	Bit(s)	Contents	Function
A001	00 to 15	CPU Bus Unit Restart Bits	Bits A00100 through A00115 can be turned ON to reset CPU Bus Units number #0 through #15, respectively. The Restart Bits are turned OFF automatically when restarting is completed. Do not turn these bits ON and OFF in the program; manipulate them from the CVSS.
A015	00 to 15	CPU Bus Service Disable Bits	Bits A01500 through A01515 can be turned ON to stop service to CPU Bus Units numbered #0 through #15, respectively. Turn the appropriate bit OFF again to resume service to the CPU Bus Unit.
A302	00 to 15	CPU Bus Unit Initializing Flags	Bits A30200 through A30215 turn ON while the corresponding CPU Bus Units (Units #0 through #15, respectively) are initializing.
A401	12	CPU Bus Error Flag	Bit A40112 is turned ON when an error occurs during the transmission of data between the CPU and CPU Bus Units, or a WDT (watchdog timer) error occurs in a CPU Bus Unit. The unit number of the CPU Bus Unit involved is contained in word A405.
	13	Duplicate Number Flag	Bit A40113 is turned ON when two Racks are assigned the same rack number, two CPU Bus Units are assigned the same unit number, or the same words are allocated to more than one Rack or Unit in the PC Setup.
A402	03	CPU Bus Unit Setting Error Flag	Bit A40203 is turned ON when the CPU Bus Units actually installed differ from the Units registered in the I/O table. The unit number of the CPU Bus Unit involved is stored in word A427.
	07	CPU Bus Unit Error Flag	Bit A40207 is turned ON when a parity error occurs during the transmission of data between the CPU and CPU Bus Units. The unit number of the CPU Bus Unit involved is stored in word A422.
A405	00 to 15	CPU Bus Unit Error Unit Number	Bits A40500 through A40515 correspond to CPU Bus Units #0 through #15, respectively. When a CPU Bus Error occurs, the bit corresponding to the unit number of the CPU Bus Unit involved turns ON.
A410	00 to 15	CPU Bus Unit Duplicate Number	The duplicate CPU Bus Unit number is written to word A410 when A40113 is turned ON.
A422	00 to 15	CPU Bus Unit Error Unit Number	The unit number of the CPU Bus Unit involved is written to word A422 when A40207 turns ON.
A427	00 to 15	CPU Bus Unit Setting Error Unit Number	The unit number of the CPU Bus Unit involved is written to word A427 when A40203 turns ON.
A502	00 to 07	Port #0 to #7 Enabled Flags	Bits A50200 through A50207 are turned ON to indicate that ports #0 through #7, respectively, are enabled for the SEND(192), RECV(193), and CMND(194) in an Ethernet, SYSMAC NET Link, or SYSMAC LINK System.
	08 to 15	Port #0 to #7 Execute Error Flags	Bits A50208 through A50215 are turned ON to indicate that an error has occurred in ports #0 through #7, respectively, during data communications using SEND(192), RECV(193), or CMND(194).
A503 to A510	00 to 15	Port #0 to #7 Completion Codes	A503 through A510 contain the completion codes for ports #0 through #7, respectively, following data communications using SEND(192), RECV(193), or CMND(194).

Appendix H
PC Memory Area Information and System Setup

PC Memory Area Information

CPU Bus Unit Area

Unit number	Allocated words	Unit number	Allocated words
0	CIO 1500 to CIO 1524	8	CIO 1700 to CIO 1724
1	CIO 1525 to CIO 1549	9	CIO 1725 to CIO 1749
2	CIO 1550 to CIO 1574	10	CIO 1750 to CIO 1774
3	CIO 1575 to CIO 1599	11	CIO 1775 to CIO 1799
4	CIO 1600 to CIO 1624	12	CIO 1800 to CIO 1824
5	CIO 1625 to CIO 1649	13	CIO 1825 to CIO 1849
6	CIO 1650 to CIO 1674	14	CIO 1850 to CIO 1874
7	CIO 1675 to CIO 1699	15	CIO 1875 to CIO 1899

The status information listed below is stored in the CPU Bus Unit Area words allocated to each Unit. The first word is calculated using the following formula: First word = 1500 + (25 x unit number). All other words are calculated by adding the specified offset to the address of the first word.

Offset	Content
+0 (first word)	Internode Test Start Bit
+1	Status of UDP socket #1
+2	Status of UDP socket #2
+3	Status of UDP socket #3
+4	Status of UDP socket #4
+5	Status of UDP socket #5
+6	Status of UDP socket #6
+7	Status of UDP socket #7
+8	Status of UDP socket #8
+9	Status of TCP socket #1
+10	Status of TCP socket #2
+11	Status of TCP socket #3
+12	Status of TCP socket #4
+13	Status of TCP socket #5
+14	Status of TCP socket #6
+15	Status of TCP socket #7
+16	Status of TCP socket #8
+17	FTP server status

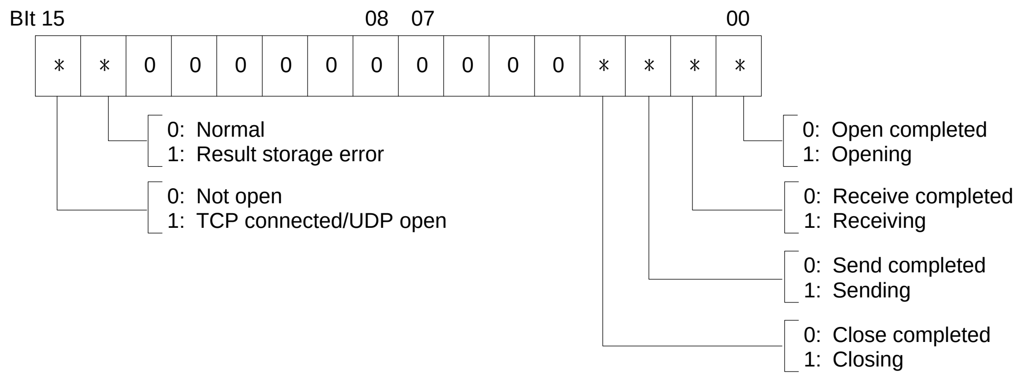
The bit configurations of each word are described next.

Internode Test Start Bit

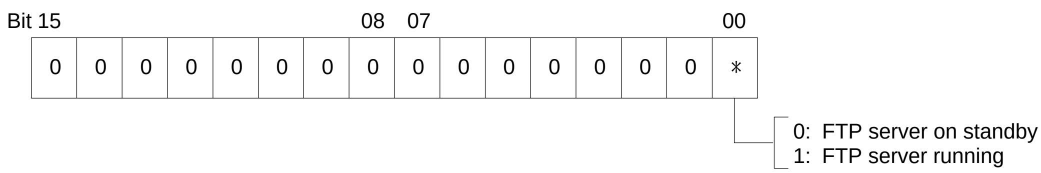
	Bit 15							08	07							00
+0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0

0: Stops internode test
1: Starts internode test

UDP/TCP Socket Status



FTP Server Status



DM Area (Software Switches)

Unit number	Allocated words	Unit number	Allocated words
0	D2000 to D2099	8	D2800 to D2899
1	D2100 to D2199	9	D2900 to D2999
2	D2200 to D2299	10	D3000 to D3099
3	D2300 to D2399	11	D3100 to D3199
4	D2400 to D2499	12	D3200 to D3299
5	D2500 to D2599	13	D3300 to D3399
6	D2600 to D2699	14	D3400 to D3499
7	D2700 to D2799	15	D3500 to D3599

The information listed below is stored in the DM Area words allocated to each Unit starting at the first word. The first word is calculated using the following formula: First word = D02000 + (100 x unit number). All other words are calculated by adding the specified offset to the address of the first word.

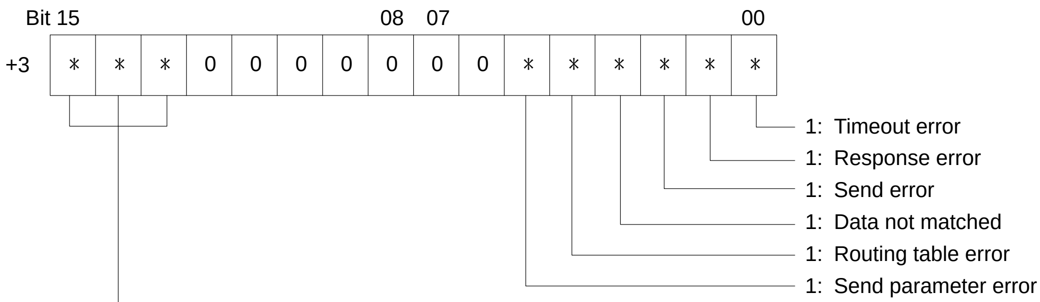
Offset	Contents	
+0	Bits 8 to 15: Target network address	Bits 0 to 7: Target node number
+1	Number of send bytes	
+2	Response monitor time	
+3	Test status	
+4	Number of internode test runs	
+5	Number of timeout errors	
+6	Number of response errors	
+7	Number of send errors	
+8	Number of times data did not match	

Words +0 through +2 must be set when running an internode test. The set parameters and their setting ranges are shown in the following table.

Parameter	Range
Target network address	01 to 7F (1 to 127 decimal)
Target node number	01 to 7E (0 to 126 decimal)
Number of send bytes	0001 to 07CC (1 to 1996 decimal) 0000 specifies the max. length of 1,998 bytes.
Response monitor time (Unit: 0.1 s)	0001 to FFFF (1 to 65535 decimal) 0000 specifies 2 s

Note The network address used here is the address used by the FINS communications service. It is NOT the IP address network number.

The following diagram shows the configuration of the test status, which indicates the results of the internode test.



Bits			Meaning
15	14	13	
0	0	0	Normal
0	0	1	Timeout error
0	1	0	Response (response code) error
0	1	1	Send error
1	0	0	Data not matched
1	0	1	Routing table error
1	1	0	Send parameter error

System Setup

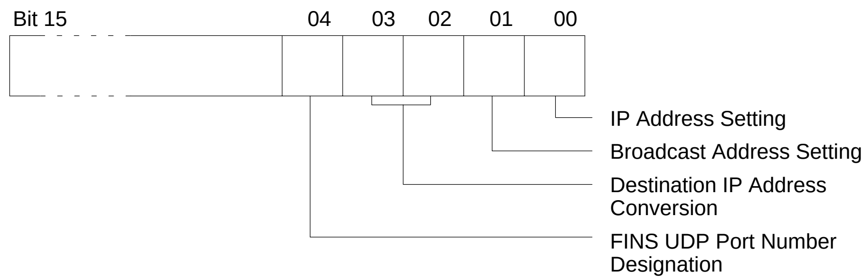
The system setup is stored beginning from the first byte in the system setup area as shown in the following table.

Position relative to first byte	Setting	Size
+0	Mode setting	2 bytes
+2	Not used.	6 bytes
+8	Local IP address	4 bytes
+12	Subnet mask	4 bytes
+16	FINS UDP port #	2 bytes
+18	FTP login name	12 bytes
+30	FTP password	8 bytes
+38	IP address table	194 bytes
+232	IP router table	66 bytes

Individual Settings

The following items can be set for the Ethernet Unit.

Mode Settings
Set bits #0 to #4 as described below.



Bit 00, IP Address Setting
Specifies whether the node number set on the rotary switches on the front panel of the Unit is to be used as the node number in the IP address. Set this bit to 0 if automatic generation is specified as the IP address conversion method.

Bit	Setting
0	Node number setting rotary switch used
1	Node number setting rotary switch not used

Bit 01, Broadcast Address Setting
Specifies when broadcasts are run.

Bit	Setting
0	Broadcast when the host number is all ones (4.3BSD specification)
1	Broadcast when the host number is all zeros (4.2BSD specification)

Note The broadcast address is the address to which the host number is set to send a transmission simultaneously to all nodes on a network. Either all ones or all zeros can be used for the broadcast address. This setting must be the same for all Ethernet Units and other nodes that are to be part of the broadcast. Refer to your host computer's documentation to specify the broadcast address for it.

Bits 02 and 03, Destination IP Address Conversion
Specifies the method for converting the FINS partner node number to an IP address.

Bits		Setting
03	02	
0	0	Automatic conversion: IP addresses will be generated automatically from the FINS node numbers. The IP address table will not be used.
0	1	
1	0	IP address table used.
1	1	IP address table and automatic generation used.

Note Refer to *Section 3 FINS Communications Service* for details about IP address conversion.

Bit 04, FINS UDP Port Number Designation
Specifies the method by which the UDP port number is specified for the FINS communications service.

Bit 04	Setting
0	Default value (9600)
1	Use value set in system setup

Local IP Address

Set the IP address for the Ethernet Unit. The Ethernet Unit cannot communicate unless this address is set. The ERH indicator will flash if no IP address is set or if an illegal IP address is set.

The following IP addresses cannot be set:

- Addresses with all bits of the host number set to 0 or 1
- Addresses with all bits of the network number set to 0 or 1
- Addresses with all bits of the subnet number set to 1
- IP addresses starting with 127 (7F hexadecimal), for example 127.35.21.16

If bit 0 of the mode setting is set to 1 (ON) in the first system setup word, the node number set on the rotary switches on the front panel of the Unit will be automatically input as the host number in the fourth byte (octet) of the local IP address. If the node number set on the Unit is used as the host number, set only the most significant 3 bytes of the address in the local IP address setting; the IP address will automatically be updated each time the node number is changed on the Unit's rotary switches, eliminating the need to change the system setup.

Subnet Masks

Set all bits in the subnet mask that correspond to the bits in the IP address used for either the network number or subnet number to "1" and set all bits in the subnet mask that correspond to the bits in the IP address used for the host number to "0".

The subnet mask must be set only when configuring a system which runs using subnetworks. If no subnet mask is set, the subnet mask will be automatically set to one of the following values, depending on the local IP address setting:

- Class-A IP Address: FF000000
- Class-B IP Address: FFFF0000
- Class-C IP Address: FFFFFF00

FINS UDP Port Number

Set the UDP port number used for the FINS communications service. This port number is valid if the FINS UDP port number designation bit (bit #4) in the mode settings is set to 1 in the first system setup word. The FINS UDP port number cannot be set to 0. If a value of 0 is set, the default value (9600) will be used as the FINS UDP port number. We recommend that the FINS UDP port number be set to a value greater than 1023.

FTP Login Name

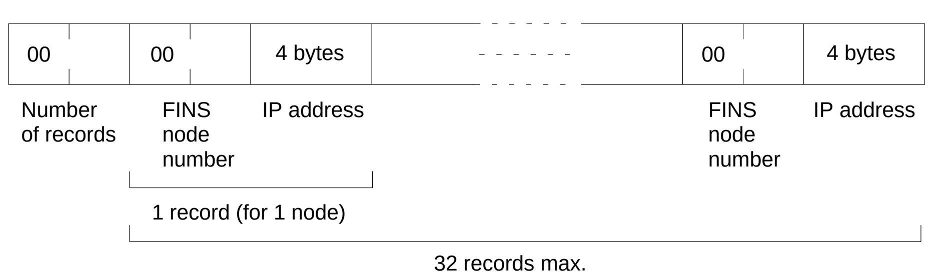
Set the login name of the Unit's FTP server. Refer to *4-2 Setting Login Names and Passwords* for details. If no login name is specified, the default login name of "CONFIDENTIAL" will be used and no FTP password will be required. Any password that is set for the default login name will be ignored.

FTP Password

Set the password for the Unit's FTP server. Refer to 4-2 *Setting Login Names and Passwords* for details.

IP Address Table

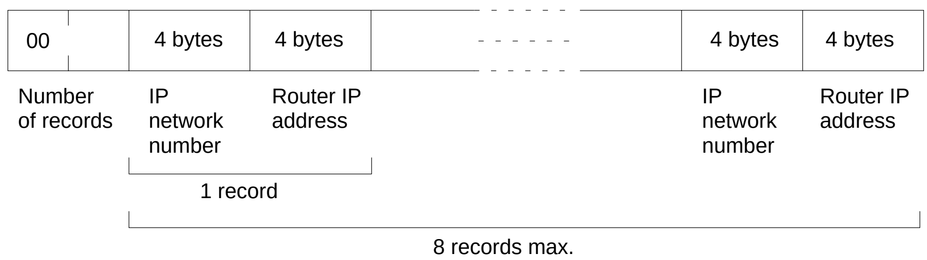
This table contains the conversion data used to generate IP addresses from FINS node numbers when using FINS communications. This table is ignored if the IP address conversion method is set to automatic generation only. The configuration of the IP address table is shown in the following diagram.



The first two bytes specify how many records (nodes) are stored in the table. This is followed by a record for each node. Each record consists of an FINS node number (2 bytes with the most significant byte always 00) and the corresponding IP address (4 bytes). The records are repeated for the specified number of records, up to the maximum of 32 records (nodes).

IP Router Table

The IP router table sets how the Ethernet Unit communicates via the IP router with nodes on other IP network segments. The configuration of the IP router table is shown in the following diagram.



The first two bytes specify how many records (up to the maximum of 8) are stored in the table. This is followed a record for each network segment with which communications are to be conducted. Each record consists of the IP network number for another network (4 bytes) and the IP address of the router that communications must first pass through to get to the network segment (4 bytes). The records are repeated for the specified number of records.

The length of the network numbers (in bytes) of the other networks to be communicated with depend on the class of the IP addresses. Although a total of four bytes are allowed for the IP network number, set only the required number of bytes for the network number from the start of the allocated space and fill any remaining bytes with zeros.

Appendix I
IP Network Address Request Form

The request form used to apply for an IP address is provided below. This form can be received electronically by requesting the NETINFO/INTERNET-NUMBER-TEMPLATE.TXT by electronic mail from HOSTMASTER@INTERNIC.NET. (Note: This template is about to be modified to accommodate the automatic registration process.) There is also a list of recommended reading at the end of this appendix.

Request Form

[netinfo/internet–number–template.txt] [04/93]

This form must be completed as part of the application process for obtaining an Internet Protocol (IP) Network Number. To obtain an Internet number, please provide the following information online, via electronic mail, to HOSTMASTER@INTERNIC.NET. If electronic mail is not available to you, please mail hardcopy to:

Network Solutions
InterNIC Registration Services
505 Huntmar Park Drive
Herndon, VA 22070

Once Registration Services receives your completed application we will send you an acknowledgement, via electronic or postal mail.

NOTE: This application is solely for obtaining a legitimate IP network number assignment. If you're interested in officially registering a domain please complete the domain application found in netinfo/domain–template.txt. If FTP is not available to you, please contact HOSTMASTER@INTERNIC.NET or phone the NIC at (800) 444–4345 (703) 742–4777 for further assistance.

NOTE: European network applications should use the European template (netinfo/european–ip–template.txt). Please follow their instructions for submission.

YOUR APPLICATION MUST BE TYPED.

1) If the network will be connected to the Internet, you must provide the name of the governmental sponsoring organization, and the name, title, mailing address, phone number, net mailbox, and NIC Handle (if any) of the contact person (POC) at that organization who has authorized the network connection. This person will serve as the POC for administrative and policy questions about authorization to be a part of the Internet. Examples of such sponsoring organizations are: DISA DNSO, the National Science Foundation (NSF), or similar military or government sponsors.

NOTE: If the network will NOT be connected to the Internet, then you do not need to provide this information.

1a. Sponsoring Organization:
1b. Contact name (Lastname, Firstname):
1c. Contact title:
1d. Mail Address :
1e. Phone :
1f. Net mailbox :
1g. NIC handle (if known):

2) Provide the name, title, mailing address, phone number, and organization of the technical POC. The online mailbox and NIC Handle (if any) of the technical POC should also be included. This is the POC for resolving technical problems associated with the network and for updating information about the network. The technical POC may also be responsible for hosts attached to this network.

2a. NIC handle (if known):
2b. Technical POC name (Lastname, Firstname):
2c. Technical POC title:
2d. Mail address :
2e. Phone :
2f. Net Mailbox :

3) Supply the short mnemonic name for the network (up to 12 characters). This is the name that will be used as an identifier in internet name and address tables.

3. Network name:

4) Identify the network geographic location and the responsible organization establishing the network.

4a. Postal address for main/headquarters network site:
4b. Name of Organization:

5) Question #5 is for MILITARY or DOD requests, ONLY.

If you require that this connected network be announced to the NSFNET please answer questions 5a, 5b, and 5c.

5a. Do you want MILNET to announce your network to the NSFNET? (Y/N):

5b. Do you have an alternate connection, other than MILNET, to the NSFNET? (please state alternate connection if answer is yes):

5c. If you've answered yes to 5b, please state if you would like the MILNET connection to act as a backup path to the NSFNET? (Y/N):

6) Estimate the number of hosts that will be on the network:

6a. Initially:
6b. Within one year:
6c. Within two years:
6d. Within five years:

7) Unless a strong and convincing reason is presented, the network (if it qualifies at all) will be assigned a class C network number. If a class C network number is not acceptable for your purposes state why. (Note: If there are plans for more than a few local networks, and more than 100 hosts, you are strongly urged to consider subnetting. [See RFC 950])

7. Reason:

8) Networks are characterized as being either Research, Defense, Government – Non Defense, or Commercial, and the network address space is shared between these four areas. Which type is this network?

8. Type of network:

9) What is the purpose of the network?
9. Purpose:

For further information contact InterNIC Registration Services:

Via electronic mail: HOSTMASTER@INTERNIC.NET
Via telephone: (800) 444–4345
Via postal mail: Network Solutions
InterNIC Registration Service
505 Huntmar Park Drive
Herndon, VA 22070

RECOMMENDED READING (available from the NIC)

Bjork, S.; Marine, A., eds. Network Protocol Implementations and Vendors Guide. Menlo Park, CA: SRI International, DDN Network Information Center; 1990 August; NIC 50002 (August 1990). 242 p. (NIC.DDN.MIL NETINFO:VENDORS–GUIDE.DOC).

Braden, R.T.; Postel, J.B. Requirements for Internet Gateways. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1987 June; RFC 1009. 55 p. (RS.INTERNET.NET POLICY RFC1009.TXT).

Defense Advanced Research Projects Agency, Internet Activities Board. IAB Official Protocol Standards. 1991 April; RFC 1200. 31 p. (RS.INTERNIC.NET POLICY RFC1200.TXT).

Feinler, E.J.; Jacobsen, O.J.; Stahl, M.K.; Ward, C.A., eds. DDN Protocol Handbook: Menlo Park, CA: SRI International, DDN Network Information Center; 1985 December; NIC 50004 and NIC 50005 and NIC 50006. 2749 p.

Garcia–Luna–Aceves, J.J.; Stahl, M.K.; Ward, C.A., eds. Internet Protocol Handbook: The Domain Name System (DNS) Handbook. Menlo Park, CA: SRI International, Network Information Systems Center; 1989 August; NIC 50007. 219 p. AD A214 698.

Kirkpatrick, S.; Stahl, M.K.; Recker, M. Internet Numbers. Menlo Park, CA: SRI International, DDN Network Information Center; 1990 July; RFC 1166. 182 p. (RS.INTERNIC.NET POLICY RFC1166.TXT).

Mogul, J.; Postel, J.B. Internet Standard Subnetting Procedure. Stanford, CA: Stanford University; 1985 August; RFC 950. 18 p. (RS.INTERNIC.NET POLICY RFC950.TXT).

Postel, J.B. Internet Control Message Protocol. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1981 September; RFC 792. 21 p. (RS.INTERNIC.NET POLICY RFC792.TXT).

Postel, J.B. Transmission Control Protocol. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1981 September; RFC 793. 85 p. (RS.INTERNIC.NET POLICY RFC793.TXT).

Postel, J.B. Address Mappings. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1981 September; RFC 796. 7 p. (RS.INTERNIC.NET POLICY RFC796.TXT). Obsoletes: IEN 115 (NACC 0968–79)

Postel, J.B. User Datagram Protocol. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1980 August 28; RFC 768. 3 p. (RS.INTERNIC.NET POLICY RFC768.TXT).

Postel, J.B. Internet Protocol. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1981 September; RFC 791. 45 p. (RS.INTERNIC.NET POLICY RFC791.TXT).

Reynolds, J.K.; Postel, J.B. Assigned Numbers. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1990 March; RFC 1060. 86 p. (RS.INTERNIC.NET POLICY RFC1060.TXT).

Reynolds, J.K.; Postel, J.B. Official Internet Protocols. Marina del Rey, CA: University of Southern California, Information Sciences Inst.; 1987 May; RFC 1011. 52 p. (RS.INTERNIC.NET POLICY RFC1011.TXT).

>

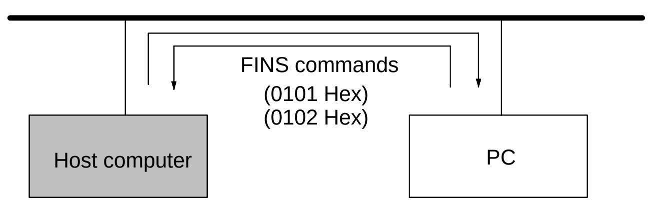
Appendix J
Simple Startup Tests

For smooth startups when performing FINS communications over UDP using an Ethernet Unit (CV500-ETN01), it is recommended that you first perform a startup test (connection verification test) using simple conditions according to the following procedure. Customers starting their system for the first time should perform settings and operations as shown below to check that the system is operating normally. Specific operating conditions and setting methods are given below.

Operating Conditions

- Connect the host computer (workstation or personal computer) to the PC (Programmable Controller) using a 1:1 connection (i.e., without communications between networks beyond the local network).
- Send only read/write commands (0101 Hex, 0102 Hex commands) for the PC variable (I/O memory) area in FINS commands addressed to the CPU Unit to the PC from the Host computer via the Ethernet.
- Set the IP address conversion method to IP address autogeneration, and set the local IP address setting to enable rotary switches for node address settings.

Ethernet (10Base5)



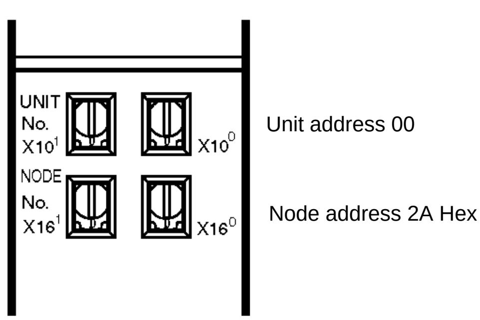
Setting Methods

1) Determine the Ethernet Unit IP Address

Set the Ethernet Unit IP address so that it does not overlap the host computer IP address. For example, if the host computer (workstation or personal computer) IP address is 133.113.102.40 (or 85.71.66.28 hexadecimal), set the Ethernet Unit IP address to 133.113.102.42 (or 85.71.66.2A hexadecimal).

2) Ethernet Unit Settings

Rotary switch settings (See note 1.)



Routing table settings (See note 2.)

Local network table

No.	Local network	Special I/O Unit No.
1	001	00
2		
3		

Set to the same as the unit number setting of the rotary switches on the Unit front panel.

CPU Bus Unit System Setup (See note 3.)

CPU Bus Unit system setup

Unit No. 00 ET

BYTE	b7	b0	HEX	BYTE	b7	b0	HEX
+0	[0000	0000]	00	+10	[0110	0110]	66
+1	[0000	0000]	00	+11	[0010	1010]	2A
+2	[0000	0000]	00	+12	[1111	1111]	FF
+3	[0000	0000]	00	+13	[1111	1111]	FF
+4	[0000	0000]	00	+14	[1111	1111]	FF
+5	[0000	0000]	00	+15	[0000	0000]	00
+6	[0000	0000]	00	+16	[0000	0000]	00
+7	[0000	0000]	00	+17	[0000	0000]	00
+8	[1000	0101]	85	+18	[0000	0000]	00
+9	[0111	0001]	71	+19	[0000	0000]	00

Enable (i.e., set to 0) rotary switches for node address settings.

To use autogeneration, set the subnet mask
(FF.FF.FF.00) to other than the host ID

IP address (Hex)

Note 1. Refer to *2-2 Rotary Switch Settings* for details.
2. Refer to *3-5-1 Setting Routing Tables* for details.
3. Refer to *2-6-1 System Setup Configuration* for details.

When you have completed making the above settings, turn OFF the power supply, and then turn it ON again. If the ERH indicator on the front panel of the Ethernet Unit is not lit, and only the RUN indicator is lit, then the Unit is operating normally.

3) Check the Connections

Send a PING command from the host computer to the Ethernet Unit to check if the Ethernet Unit is connected (i.e., if communications are possible) using an IP protocol label. If the response to the PING command “PING 133.113.102.42” is not normal, there is a problem with the cable or IP address (refer to *7-1 Loopback Test using PING* for details). In this condition, the system will always end in an error no matter how often FINS commands are sent from the host computer.

4) Send FINS Commands

Send the FINS command “Read variable area 0101 Hex” to the PC from UDP port 9600 on the Host computer. Use the same port number as FINS UDP port 9600 (default value). If a normal response is received, the startup test is finished. For details of FINS commands, refer to sending and receiving FINS command and response data and reading the variable area (0101 Hex) in the *FINS Commands Reference Manual*.

Command

ICF	RSV	GCT	DNA	DA1	DA2	SNA	SA1	SA2	SID	Command	
80	00	02	01	2A	00	01	28	00	01	01	01
										Read	

Text

80	00	64	00	00	96

Words: DM 0100 to DM 0150

Response

ICF	RSV	GCT	DNA	DA1	DA2	SNA	SA1	SA2	SID	Command	
C0	00	02	01	28	00	01	2A	00	01	01	01
										Read	

End code		Text				
00	00	XX	XX	…	XX	XX

DM 0100 data to DM 0249 data

Next, change the command code and text, and write the data (Write variable area: 0102 Hex)

Glossary

address	A number used to identify the location of data or programming instructions in memory or to identify the location of a network or a unit in a network.
advertisement	The process of sending out information to make resources available to other devices, e.g., sending information to other nodes in a network to make windows available for communications.
allocation	The process by which the PC assigns certain bits or words in memory for various functions. This includes pairing I/O bits to I/O points on Units.
area	See *data area* and *memory area*.
ARP	Address Resolution Protocol: Determines the Ethernet address (i.e., physical address) by broadcasting based on the target IP address.
ASCII	Short for American Standard Code for Information Interchange. ASCII is used to code characters for output to printers and other external devices.
asynchronous execution	Execution of programs and servicing operations in which program execution and servicing are not synchronized with each other.
Auxiliary Area	A PC data area allocated to flags and control bits.
back-up	A copy made of existing data to ensure that the data will not be lost even if the original data is corrupted or erased.
baud rate	The data transmission speed between two devices in a system measured in bits per second.
BCD	Short for binary-coded decimal.
binary	A number system where all numbers are expressed in base 2, i.e., numbers are written using only 0's and 1's. Each group of four binary bits is equivalent to one hexadecimal digit. Binary data in memory is thus often expressed in hexadecimal for convenience.
binary-coded decimal	A system used to represent numbers so that every four binary bits is numerically equivalent to one decimal digit.
bit	The smallest piece of information that can be represented on a computer. A bit has the value of either zero or one, corresponding to the electrical signals ON and OFF. A bit represents one binary digit. Some bits at particular addresses are allocated to special purposes, such as holding the status of input from external devices, while other bits are available for general use in programming.
bit address	The location in memory where a bit of data is stored. A bit address specifies the data area and word that is being addressed as well as the number of the bit within the word.
broadcast	The process of sending data simultaneously to all nodes on a single network and used to test network communications.
buffer	A temporary storage space for data in a computerized device.

byte A unit of data equivalent to 8 bits, i.e., half a word.

central processing unit A device that is capable of storing programs and data, and executing the instructions contained in the programs. In a PC System, the central processing unit executes the program, processes I/O signals, communicates with external devices, etc.

channel See *word*.

character code A numeric (usually binary) code used to represent an alphanumeric character.

checksum A sum transmitted with a data pack in communications. The checksum can be recalculated from the received data to confirm that the data in the transmission has not been corrupted.

CIO Area A memory area used to control I/O and to store and manipulate data. CIO Area addresses do not require prefixes.

client A process or node requesting processing from a server.

completion code A code stored in the PC to indicate the results (i.e., normal or error) of PC communications.

control bit A bit in a memory area that is set either through the program or via a Programming Device to achieve a specific purpose, e.g., a Restart Bit is turned ON and OFF to restart a Unit.

control data An operand that specifies how an instruction is to be executed. The control data may specify the part of a word is to be used as the operand, it may specify the destination for a data transfer instructions, it may specify the size of a data table used in an instruction, etc.

control signal A signal sent from the PC to effect the operation of the controlled system.

CPU The name of the Unit in a PC that contains the main CPU and other main PC components. See also *central processing unit*.

CPU Bus Unit A special Unit used with CV-series PCs that mounts to the CPU bus. This connection to the CPU bus enables special data links, data transfers, and processing.

CPU Bus Unit Area A part of the CIO Area allocated to CPU Bus Units. The use of the words and bits in this area is determined by the Unit to which they are allocated.

CPU Rack The main Rack in a building-block PC, the CPU Rack contains the CPU, a Power Supply, and other Units. The CPU Rack, along with the Expansion CPU Rack, provides both an I/O bus and a CPU bus.

CTS An acronym for clear-to-send, a signal used in communications between electronic devices to indicate that the receiver is ready to accept incoming data.

CV Support Software A programming package run on an IBM PC/AT or compatible to serve as a Programming Device for CV-series PCs.

CV-series PC Any of the following PCs: CV500, CV1000, CV2000, or CVM1

CVSS See *CV Support Software*.

cycle One unit of processing performed by the CPU, including SFC/ladder program execution, peripheral servicing, I/O refreshing, etc. The cycle is called the scan with C-series PCs.

cycle time The time required to complete one cycle of CPU processing.

data area An area in the PC's memory that is designed to hold a specific type of data.

data length In communications, the number of bits that is to be treated as one unit in data transmissions.

data transfer Moving data from one memory location to another, either within the same device or between different devices connected via a communications line or network.

datagram A unit of data used in network communications.

debug A process by which a draft program is corrected until it operates as intended. Debugging includes both the removal of syntax errors, as well as the fine-tuning of timing and coordination of control operations.

decimal A number system where numbers are expressed to the base 10. In a PC all data is ultimately stored in binary form, four binary bits are often used to represent one decimal digit, via a system called binary-coded decimal.

default A value automatically set by the PC when the user does not specifically set another value. Many devices will assume such default conditions upon the application of power.

delimiter A code sent during communications between devices to indicate the end of the current transmission, but not the end of the entire transmission. See *terminator*.

destination The location where an instruction places the data on which it is operating, as opposed to the location from which data is taken for use in the instruction. The location from which data is taken is called the source.

DIP switch Dual in-line package switch, an array of pins in a signal package that is mounted to a circuit board and is used to set operating parameters.

DM Area A data area used to hold only word data. Words in the DM area cannot be accessed bit by bit.

DM word A word in the DM Area.

loopback test A test executed by sending an FINS command between two nodes on communications networks and used to determine if communications are normal.

EEPROM Electrically erasable programmable read-only memory; a type of ROM in which stored data can be erased and reprogrammed. This is accomplished using a special control lead connected to the EEPROM chip and can be done without having to remove the EEPROM chip from the device in which it is mounted.

electrical noise Random variations of one or more electrical characteristics such as voltage, current, and data, which might interfere with the normal operation of a device.

EPROM Erasable programmable read-only memory; a type of ROM in which stored data can be erased, by ultraviolet light or other means, and reprogrammed.

error code A numeric code generated to indicate that an error exists, and something about the nature of the error. Some error codes are generated by the system; others are defined in the program by the operator.

Ethernet	A hardware local area networking system used for communications.
Ethernet address	A physical address assigned to Ethernet hardware.
even parity	A communication setting that adjusts the number of ON bits so that it is always even. See *parity*.
event processing	Processing that is performed in response to an event, e.g., an interrupt signal.
FA	Factory automation.
fatal error	An error that stops PC operation and requires correction before operation can continue.
FCS	See *frame checksum*.
FINS	Factory Interface Network Service: A protocol that transfers messages between PCs on any of various OMRON FA networks. Also see *CV-mode*.
flag	A dedicated bit in memory that is set by the system to indicate some type of operating status. Some flags, such as the carry flag, can also be set by the operator or via the program.
frame checksum	The results of exclusive ORing all data within a specified calculation range. The frame checksum can be calculated on both the sending and receiving end of a data transfer to confirm that data was transmitted correctly.
FTP	File Transfer Protocol: Transfers data in file units to and from Memory Cards.
function code	A number assigned to a ladder-diagram instruction to input and execute it.
header	The first portion of a command or response in a communications packet. The header specifies basic information that determines the purpose of the packet.
header code	A code in an instruction that specifies what the instruction is to do.
hexadecimal	A number system where all numbers are expressed to the base 16. In a PC all data is ultimately stored in binary form, however, displays and inputs on Programming Devices are often expressed in hexadecimal to simplify operation. Each group of four binary bits is numerically equivalent to one hexadecimal digit.
host interface	An interface that allows communications with a host computer.
host number	The portion of the IP address used to differentiate nodes on an Ethernet network.
I/O allocation	The process by which the PC assigns certain bits in memory for various functions. This includes pairing I/O bits to I/O points on Units.
I/O delay	The delay in time from when a signal is sent to an output to when the status of the output is actually in effect or the delay in time from when the status of an input changes until the signal indicating the change in the status is received.
I/O refreshing	The process of updating output status sent to external devices so that it agrees with the status of output bits held in memory and of updating input bits in memory so that they agree with the status of inputs from external devices.
I/O response time	The time required for an output signal to be sent from the PC in response to an input signal received from an external device.

I/O verification error A error generated by a disagreement between the Units registered in the I/O table and the Units actually mounted to the PC.

I/O word A word in the CIO area that is allocated to a Unit in the PC System and is used to hold I/O status for that Unit.

IBM PC/AT or compatible A computer that has similar architecture to, that is logically compatible with, and that can run software designed for an IBM PC/AT computer.

ICMP Internet Control Message Protocol: Supports IP communications by signalling errors in data transfers.

initialize Part of the startup process whereby some memory areas are cleared, system setup is checked, and default values are set.

input The signal coming from an external device into the PC. The term input is often used abstractly or collectively to refer to incoming signals.

input bit A bit in the CIO area that is allocated to hold the status of an input.

input device An external device that sends signals into the PC System.

instruction A direction given in the program that tells the PC of the action to be carried out, and the data to be used in carrying out the action. Instructions can be used to simply turn a bit ON or OFF, or they can perform much more complex actions, such as converting and/or transferring large blocks of data.

interface An interface is the conceptual boundary between systems or devices and usually involves changes in the way the communicated data is represented. Interface devices such as NSBs perform operations like changing the coding, format, or speed of the data.

internode test A test executed via data area settings between two nodes on communications networks and used to determine if communications are normal.

interrupt (signal) A signal that stops normal program execution and causes a subroutine to be run or other processing to take place.

IOM (Area) A collective memory area containing all of the memory areas that can be accessed by bit, including timer and counter Completion Flags. The IOM Area includes all memory area memory addresses between 0000 and 0FFF.

IP Internet Protocol: Transfers datagrams to target nodes using IP addresses.

IP address An address assigned to the Ethernet Unit as a node in an Ethernet network. The IP address consists of a network number, possibly a subnet number, and a host number.

LAN An acronym for local area network.

LED Acronym for light-emitting diode; a device used as for indicators or displays.

leftmost (bit/word) The highest numbered bits of a group of bits, generally of an entire word, or the highest numbered words of a group of words. These bits/words are often called most-significant bits/words.

link A hardware or software connection formed between two Units. “Link” can refer either to a part of the physical connection between two Units or a software connection created to data existing at another location (i.e., data links).

load The processes of copying data either from an external device or from a storage area to an active portion of the system such as a display buffer. Also, an output device connected to the PC is called a load.

local In network communications, the node or device from which communications are being viewed. See *remote*.

local area network A network consisting of nodes or positions in a loop arrangement. Each node can be any one of a number of devices. This kind of network usually operates over a small area such as a group of offices or a factory floor.

local network table A table that specifies all of the networks that a PC belongs to and the unit numbers of the Units connecting the PC to each of these networks.

local node In network communications, the node from which communications are being viewed. See *remote node*.

master In a SYSMAC NET Link System, a Unit specified to manage network communications.

master number A number assigned to a master in a SYSMAC NET Link System. This number is different from the unit number.

megabyte A unit of storage equal to one million bytes.

memory area Any of the areas in the PC used to hold data or programs.

network address An address set in routing tables and used to differentiate OMRON networks for FINS communications.

network number The portion of the IP address used to differentiate networks.

Network Service Board A device with an interface to connect devices other than PCs to a SYSMAC NET Link System.

Network Service Unit A Unit that provides two interfaces to connect peripheral devices to a SYSMAC NET Link System.

network support table Tables of settings used to establish operating parameters for SYSMAC LINK and SYSMAC NET Link Systems.

node One of the positions in a LAN. Each node incorporates a device that can communicate with the devices at all of the other nodes. The device at a node is identified by the node number.

node number An address used to differentiate nodes (including Ethernet Units) on OMRON networks for FINS protocol. The node number of a CV-series PC is called the “unit number” in the PC Setup.

noise interference Disturbances in signals caused by electrical noise.

nonfatal error A hardware or software error that produces a warning but does not stop the PC from operating.

NSB An acronym for Network Service Board.

NSU An acronym for Network Service Unit.

octal A number system where all numbers are expressed in base 8, i.e., numbers are written using only numerals 0 through 7.

odd parity A communications setting that adjusts the number of ON bits so that it is always odd. See *parity*.

OFF The status of an input or output when a signal is said not to be present. The OFF state is generally represented by a low voltage or by non-conductivity, but can be defined as the opposite of either.

OFF delay The delay between the time when a signal is switched OFF (e.g., by an input device or PC) and the time when the signal reaches a state readable as an OFF signal (i.e., as no signal) by a receiving party (e.g., output device or PC).

offset A positive or negative value added to a base value such as an address to specify a desired value.

ON The status of an input or output when a signal is said to be present. The ON state is generally represented by a high voltage or by conductivity, but can be defined as the opposite of either.

ON delay The delay between the time when an ON signal is initiated (e.g., by an input device or PC) and the time when the signal reaches a state readable as an ON signal by a receiving party (e.g., output device or PC).

operand The values designated as the data to be used for an instruction. An operand can be input as a constant expressing the actual numeric value to be used or as an address to express the location in memory of the data to be used.

operating error An error that occurs during actual PC operation as opposed to an initialization error, which occurs before actual operations can begin.

optical communications A communications method in which signals are sent over optical fiber cable to prevent noise interference and increase transmission distance.

output The signal sent from the PC to an external device. The term output is often used abstractly or collectively to refer to outgoing signals.

output signal A signal being sent to an external device. Generally an output signal is said to exist when, for example, a connection point goes from low to high voltage or from a nonconductive to a conductive state.

overflow The state where the capacity of a data storage location has been exceeded.

overwrite Changing the content of a memory location so that the previous content is lost.

parity Adjustment of the number of ON bits in a word or other unit of data so that the total is always an even number or always an odd number. Parity is generally used to check the accuracy of data after being transmitted by confirming that the number of ON bits is still even or still odd.

parity check Checking parity to ensure that transmitted data has not been corrupted.

PC An acronym for Programmable Controller.

PC Setup A group of operating parameters set in the PC from a Programming Device to control PC operation.

Peripheral Device Devices connected to a PC System to aid in system operation. Peripheral devices include printers, programming devices, external storage media, etc.

peripheral servicing Processing signals to and from peripheral devices, including refreshing, communications processing, interrupts, etc.

present value The current value registered in a device at any instant during its operation. Present value is abbreviated as PV. The use of this term is generally restricted to timers and counters.

Programmable Controller A computerized device that can accept inputs from external devices and generate outputs to external devices according to a program held in memory. Programmable Controllers are used to automate control of external devices. Although single-unit Programmable Controllers are available, building-block Programmable Controllers are constructed from separate components. Such Programmable Controllers are formed only when enough of these separate components are assembled to form a functional assembly, i.e., there is no one individual Unit called a PC.

Programming Device A Peripheral Device used to input a program into a PC or to alter or monitor a program already held in the PC. There are dedicated programming devices, such as Programming Consoles, and there are non-dedicated devices, such as a host computer.

PROM Programmable read-only memory; a type of ROM into which the program or data may be written after manufacture, by a customer, but which is fixed from that time on.

PROM Writer A peripheral device used to write programs and other data into a ROM for permanent storage and application.

prompt A message or symbol that appears on a display to request input from the operator.

protocol The parameters and procedures that are standardized to enable two devices to communicate or to enable a programmer or operator to communicate with a device.

PV See *present value*.

Rack An assembly that forms a functional unit in a Rack PC System. A Rack consists of a Backplane and the Units mounted to it. These Units include the Power Supply, CPU, and I/O Units. Racks include CPU Racks, Expansion I/O Racks, and I/O Racks. The CPU Rack is the Rack with the CPU mounted to it. An Expansion I/O Rack is an additional Rack that holds extra I/O Units. An I/O Rack is used in the C2000H Duplex System, because there is no room for any I/O Units on the CPU Rack in this System.

RAM Random access memory; a data storage media. RAM will not retain data when power is disconnected.

RAS An acronym for reliability, assurance, safety.

refresh The process of updating output status sent to external devices so that it agrees with the status of output bits held in memory and of updating input bits in memory so that they agree with the status of inputs from external devices.

relay A point in a network through which communications pass to reach another network.

remote In network communications, the node or device with which communications are taking place. See *local*.

remote node In network communications, the node with which communications are taking place. See *local node*.

reset The process of turning a bit or signal OFF or of changing the present value of a timer or counter to its set value or to zero.

response code A code sent with the response to a data communications command that specifies how the transmitted data was processed.

response format A format specifying the data required in a response to a data transmission.

Restart Bit A bit used to restart a Unit mounted to a PC.

retrieve The processes of copying data either from an external device or from a storage area to an active portion of the system such as a display buffer. Also, an output device connected to the PC is called a load.

retry The process whereby a device will re-transmit data which has resulted in an error message from the receiving device.

ROM Read only memory; a type of digital storage that cannot be written to. A ROM chip is manufactured with its program or data already stored in it and can never be changed. However, the program or data can be read as many times as desired.

router A device used to connect two networks (i.e., two coaxial cables) in an Ethernet System.

routing table Tables of setting that specify what networks a device is a member of and what nodes must be passed through to reach other specific networks. See *local network table* and *relay network table*.

segment The portion of an Ethernet System that defines one network, i.e., a single coaxial cable and all nodes connected to it.

self diagnosis A process whereby the system checks its own operation and generates a warning or error if an abnormality is discovered.

server A process or node that provides processing to a client.

servicing The process whereby the PC provides data to or receives data from external devices or remote I/O Units, or otherwise handles data transactions for Link Systems.

set The process of turning a bit or signal ON.

set value The value from which a decrementing counter starts counting down or to which an incrementing counter counts up (i.e., the maximum count), or the time from which or for which a timer starts timing. Set value is abbreviated SV.

socket A file structure that serves as an end point for a virtual circuit created for communications.

software switch See *memory switch*.

Special I/O Unit A Unit that is designed for a specific purpose. Special I/O Units include Position Control Units, High-speed Counter Units, Analog I/O Units, etc.

subnet number The portion of the IP address used differentiate subnetworks in an Ethernet network. This address exists only if the user sets a network mask for the IP address to allocate part of the host number as the subnet number.

SV Abbreviation for set value.

synchronous execution Execution of programs and servicing operations in which program execution and servicing are synchronized so that all servicing operations are executed each time the programs are executed.

syntax The form of a program statement (as opposed to its meaning). For example, the two statements, `LET A=B+B` and `LET A=B*2` use different syntaxes, but have the same meaning.

syntax error An error in the way in which a program is written. Syntax errors can include ‘spelling’ mistakes (i.e., a function code that does not exist), mistakes in specifying operands within acceptable parameters (e.g., specifying read-only bits as a destination), and mistakes in actual application of instructions (e.g., a call to a subroutine that does not exist).

SYSMAC LINK System A communications system used to create data links and enable network communications between PCs.

SYSMAC NET Link System An optical LAN formed from PCs connected through SYSMAC NET Link Units. A SYSMAC NET Link System also normally contains nodes interfacing computers and other peripheral devices. PCs in the SYSMAC NET Link System can pass data back and forth, receive commands from any interfaced computer, and share any interfaced peripheral device.

SYSMAC NET Link Unit The Unit used to connect PCs to a SYSMAC NET Link System.

system configuration The arrangement in which Units in a System are connected. This term refers to the conceptual arrangement and wiring together of all the devices needed to comprise the System. In OMRON terminology, system configuration is used to describe the arrangement and connection of the Units comprising a Control System that includes one or more PCs.

system error An error generated by the system, as opposed to one resulting from execution of an instruction designed to generate an error.

system error message An error message generated by the system, as opposed to one resulting from execution of an instruction designed to generate a message.

system setup Parameters set to control the operation of the CVSS, CPU Bus Units, etc.

target node See *remote node*.

TCP Transmission Control Protocol: Performs communications after establishing a connection (i.e., a virtual circuit) with the target node to provide a highly reliable communications method.

terminator 1) The code comprising an asterisk and a carriage return (* CR) which indicates the end of a block of data in communications between devices. Frames within a multi-frame block are separated by delimiters. 2) Unit in a Link System designated as the last Unit on the communications line. 3) A device attached to the end of a network communications line to specify the end of the network.

timer A location in memory accessed through a TC bit and used to time down from the timer's set value. Timers are turned ON and reset according to their execution conditions.

transceiver A physical interface to a network that converts signals.

transfer The process of moving data from one location to another within the PC, or between the PC and external devices. When data is transferred, generally a copy of the data is sent to the destination, i.e., the content of the source of the transfer is not changed.

transmission distance The distance that a signal can be transmitted.

UDP User Datagram Protocol: Performs datagram communications. Data resends, priority control, flow control, and other measures to ensure communications reliability are not performed for UDP communications, i.e., there is no way of guaranteeing normal communications without programming special measures to do so into the user's application program.

UM area The memory area used to hold the active program, i.e., the program that is being currently executed.

Unit In OMRON PC terminology, the word Unit is capitalized to indicate any product sold for a PC System. Though most of the names of these products end with the word Unit, not all do, e.g., a Remote Terminal is referred to in a collective sense as a Unit. Context generally makes any limitations of this word clear.

unit address A number used to control network communications in FINS protocol. Unit addresses are computed for Units in various ways, e.g., 10 hex is added to the unit number to determine the unit address for a CPU Bus Unit.

unit number A number assigned to some Link Units, Special I/O Units, and CPU Bus Units to facilitate identification when assigning words or other operating parameters.

watchdog timer A timer within the system that ensures that the scan time stays within specified limits. When limits are reached, either warnings are given or PC operation is stopped depending on the particular limit that is reached.

WDT See *watchdog timer*.

word A unit of data storage in memory that consists of 16 bits. All data areas consists of words. Some data areas can be accessed only by words; others, by either words or bits.

word address The location in memory where a word of data is stored. A word address must specify (sometimes by default) the data area and the number of the word that is being addressed.

word allocation The process of assigning I/O words and bits in memory to I/O Units and terminals in a PC System to create an I/O Table.

work bit A bit in a work word.

write protect switch A switch used to write-protect the contents of a storage device, e.g., a floppy disk. If the hole on the upper left of a floppy disk is open, the information on this floppy disk cannot be altered.

write-protect A state in which the contents of a storage device can be read but cannot be altered.

Index

U–W

Revision History

A manual revision code appears as a suffix to the catalog number on the front cover of the manual.

Cat. No. W242-E1-2

Revision code

The following table outlines the changes made to the manual during each revision. Page numbers refer to the previous version.

Revision code	Date	Revised content
1	March 1994	Original production
1A	April 1996	**Page 91:** The port information for TCP sockets corrected in the top paragraph. **Pages 95, 97, 103, 105:** The locations of the Results Storage Error Flags corrected in the program.
2	July 2000	**Page v:** Updated Notice contents. **Page xiii:** Safety precautions updated. **Page 10:** Added note to settings explanation. **Page 23:** Updated network table. **Page 24:** Updated Ethernet Unit system setup explanation. **Page 26:** Corrected “1 (ON)” to “0 (OFF)” in middle of page. **Page 32:** Added note to initial startup procedures. **Page 40, 53, 57, 82, 83, 130, 135, and 141:** Corrected numbers of bytes. **Pages 45, 48 to 52, :** Corrected numbers of words in table. **Page 58:** Added note to destination Unit address. **Page 66:** Added note to FTP explanation. **Page 72:** Changed order of procedure. **Page 76:** Updated example of using commands. **Pages 82, 90, 91, and100:** Updated socket services explanation. **Pages 95, 97, 103, 105, 109, 110, 116, and 117:** Corrected diagrams. **Page 126:** Updated graphics text. **Page 129:** Updated references explanation. **Page 162:** Added note to self-diagnostic features. **Page 174:** Updated troubleshooting table. **Page 181:** Corrected times in table. **Page 194:** Updated graphics text and IP address conversion explanation. **Pages 200:** Added new appendix.