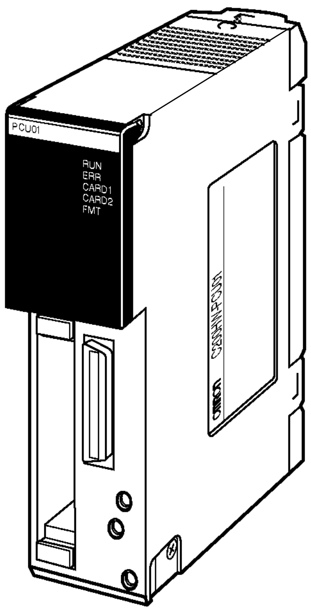
SYSMAC
C200HW-PCU01
C200HW-PCS01-V2
PC Card Unit

Operation Manual

Produced February 2001



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.

MS-DOS is a registered trademark of Microsoft Corporation.

PC-DOS is a registered trademark of International Business Machines Corporation.

PCM Plus is a registered trademark of Phoenix Technologies, Ltd.

In general, system names and product names mentioned in this manual are registered trademarks of the respective developers or manufacturers.

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.

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

TABLE OF CONTENTS

Appendices

About this Manual:

This manual describes the installation and operation of the PC Card Unit and includes the sections described below. A PC Card Unit enables usage of a PC card with a C200HX/HG/HE PC to provide an interface to extra memory or an Ethernet connection.

In this manual Programmable Controller has been abbreviated as PC. Do not confuse this with personal computer, which has not been abbreviated except in the case of the PC Card Unit.

Please read this manual carefully and be sure you understand the information provided before attempting to install and operate a PC Card Units. **Be sure to read the following section before operating the PC Card Unit.**

Section 1 explains the features and system configuration of the PC Card Unit.

Section 2 outlines the individual system components and their functions. Devices that can be purchased separately are also described.

Section 3 outlines the installation of a PC Card Unit and peripheral devices.

Section 4 describes how to install a PC Card Unit and set up the C200HX/HG/HE.

Section 5 describes formatting Memory Cards, the file format, and file transfers between a PC Card Unit and the C200HX/HG/HE.

Section 6 outlines the installation of a PC Card Unit and connection to Ethernet.

Section 7 describes how to install Ethernet Cards and use the setup software.

Section 8 explains how to use SEND and RECV commands to transfer data.

Section 9 provides information on communicating in Ethernet Systems using FINS commands, and explains how to use the CMCR instruction to issue FINS commands.

Section 10 describes sockets (an interface for directly using TCP and UDP functions from the user program) and explains how to use socket services.

Section 11 describes the FINS commands that can be sent to the C200HX/HG/HE CPU and the FINS commands that can be sent to the PC Card Unit.

Section 12 provides explanations of each indicator and the actions to be taken for them, and explains how to use the error log.

The ***Appendices*** provide information on standard models, specifications, connector pin assignments, response codes from the C200HX/HG/HE CPU, FINS response codes from the PC Card Unit, differences with the CV-series Ethernet Unit, an example using a memory card with a personal computer, precautions for setting up the network, contents of version upgrade (PCS01-E to PCS01-EV1), and remote tool connection procedure.

⚠**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 the Programmable Controller (PC) and related devices.

The information contained in this section is important for the safe and reliable application of the PC. You must read this section and understand the information contained before attempting to set up or operate a PC system.

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 operation 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 OMRON PCs. Be sure to read this manual before attempting to use the software 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 abovementioned applications.

3 Safety Precautions

⚠ **WARNING** Never attempt to disassemble any Units while power is being supplied. Doing so may result in serious electrical shock or electrocution.

⚠ **WARNING** Never touch any of the terminals while power is being supplied. Doing so may result in serious electrical shock or electrocution.

4 Operating Environment Precautions

Do not operate the control system in the following places.

- Where the PC is exposed to direct sunlight.
- Where the ambient temperature is below 0°C or over 55°C.
- Where the PC may be affected by condensation due to radical temperature changes.
- Where the ambient humidity is below 10% or over 90%.
- Where there is any corrosive or inflammable gas.
- Where there is excessive dust, saline air, or metal powder.
- Where the PC is affected by vibration or shock.
- Where any water, oil, or chemical may splash on the PC.

- Provide proper shielding when installing in the following locations:
  - Locations subject to static electricity or other sources of noise.
  - Locations subject to strong electromagnetic fields.
  - Locations subject to possible exposure to radiation.
  - Locations near to power supply lines.

⚠ **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.

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

- Always ground the system to 100 Ω or less when installing the system to protect against electrical shock.
- Always turn off the power supply to the PC before attempting any of the following. Performing any of the following with the power supply turned on may lead to electrical shock:
  - Mounting or removing any Units (e.g., I/O Units, CPU Unit, etc.) or memory cassettes.
  - Assembling any devices or racks.
  - Connecting or disconnecting any cables or wiring.

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

- Use the Units only with the power supplies and voltages specified in the operation manuals. Other power supplies and voltages may damage the Units.
- Take measures to stabilize the power supply to conform to the rated supply if it is not stable.
- Provide circuit breakers and other safety measures to provide protection against shorts in external wiring.
- Do not apply voltages exceeding the rated input voltage to Input Units. The Input Units may be destroyed.
- Do not apply voltages exceeding the maximum switching capacity to Output Units. The Output Units may be destroyed.
- Always disconnect the LG terminal when performing withstand voltage tests.
- Install all Units according to instructions in the operation manuals. Improper installation may cause faulty operation.
- Be sure to tighten Backplane screws, terminal screws, and cable connector screws securely.
- Do not attempt to take any Units apart, to repair any Units, or to modify any Units in any way.
- Be sure to have this stickers in place on the Units when wiring. Wiring without the stickers in place may result in wiring cuttings entering the Unit and causing faulty operation.

- Remove the stickers after the completion of wiring to ensure proper heat dissipation. Operating the Unit with the sticker in place may cause heat build-up and possible faulty operation.
- Use crimp terminals for wiring. Wiring bare wires directly to terminals may result in a fire.
- Be sure that terminal blocks and connectors are correct before connecting them. Improper connection may damage the Units.
- Do not use the setup software for the C200HW-PCS01(-EV1) to perform settings for the C200HW-PCS01-V2.

⚠ **Caution** The following precautions are necessary to ensure the general safety of the system. Always heed these precautions.

- Provide double safety mechanisms to handle incorrect signals that can be generated by broken signal lines or momentary power interruptions.
- Provide external interlock circuits, limit circuits, and other safety circuits in addition to any provided within the PC to ensure safety.

⚠ **Caution** When using the PC Card, be sure to observe the following precautions.

- Insert or eject the PC Card for at least 10 seconds after it is inserted or ejected. (That is, don’t eject the PC Card for at least 10 seconds after it is inserted and don’t insert the the PC Card for at least 10 seconds after it is ejected.)
- Don’t eject the PC Card while it is being accessed (while either the CARD1 indicator or CARD2 indicator is lit).

6 EC Directives

PC Card Units that meet EC directives must be installed as follows:

1, 2, 3...

1. PC Card Units are defined for installation inside control panels, so they must all be installed within control panels. In addition, the control panels must be grounded, enclosed metal housings.
2. Used reinforced insulation or double insulation for the DC power supplies used for the communications power supply, internal circuit power supply, and the I/O power supplies.
3. PC Card Units that meet EC directives also meet the common emission standard (EN50081-2). When PC Card Units are built into equipment, however, the measure necessary to ensure that the standard is met will vary with the overall configuration of the control panel, the other devices connected to the control panel, and other conditions. You must therefore confirm that EC directives are met for the overall machine or device.

To reduce noise, wire the control panel with as thick and short electric lines as possible and ground to 100 Ω min.

Part 1
Features and Components

This part of the manual introduces the PC Card Unit's features, describes the components of the PC Card Unit, and explains the system configuration.

SECTION 1
Features and System Configuration

This section provides an introduction to the PC Card Unit’s features and explains the system configuration.

1-1 Features

The PC Card Unit provides various functions for using PC cards with SYSMAC C200HX/HG/HE Programmable Controllers. CIO, DM, and EM data (but not the user program) can be loaded and saved between C200HX/HG/HE Programmable Controllers and memory cards inserted in the PC Card Unit. These operations can be executed from ladder programs.

Inserting a commercially available Ethernet card into the PC Card Unit allows reading from and writing to C200HX/HG/HE memory from a personal computer or workstation, including reading programs. (The program data that is read, however, cannot be edited with the Ladder Support Software.)

The PC Card Unit provides the following features.

Standard Memory Cards

The PC Card Unit provides two PCMCIA 2.1-conforming PC card interface slots (but 3.3V cards are not supported). Either two type-I or -II or one type-III PC card can be installed. SRAM cards, FLASH cards, and ATA interface cards can all be used.

Personal Computer Compatibility

PC card data written by the PC Card Unit can be read and edited by commercially available personal computers, which can also be used to write data to the PC card.

File Operations

C200HX/HG/HE memory contents can be written to memory cards as files, using ladder-diagram instructions. File contents can be compared with C200HX/HG/HE memory and files can be searched. Word data can be separated by commas in the file format, so commercially available spreadsheets can be used.

Ethernet Expansion

With the C200HW-PCS01-V2 PC Card Unit Ethernet Set, commercially available Ethernet cards can be used to exchange data with CV Ethernet Units, personal computers, workstations, etc., by using the FINS protocol via UDP/IP.

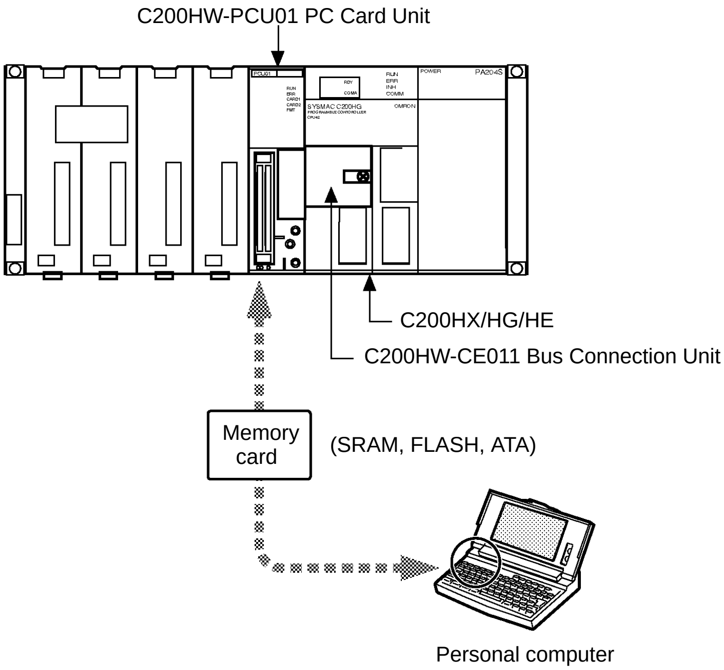
FINS protocol is a communications protocol developed by OMRON for FA. For details, refer to *Section 11 Using FINS Commands and Responses*.

Note The PC Card Unit cannot be used with the C200HE-CPU11-E.

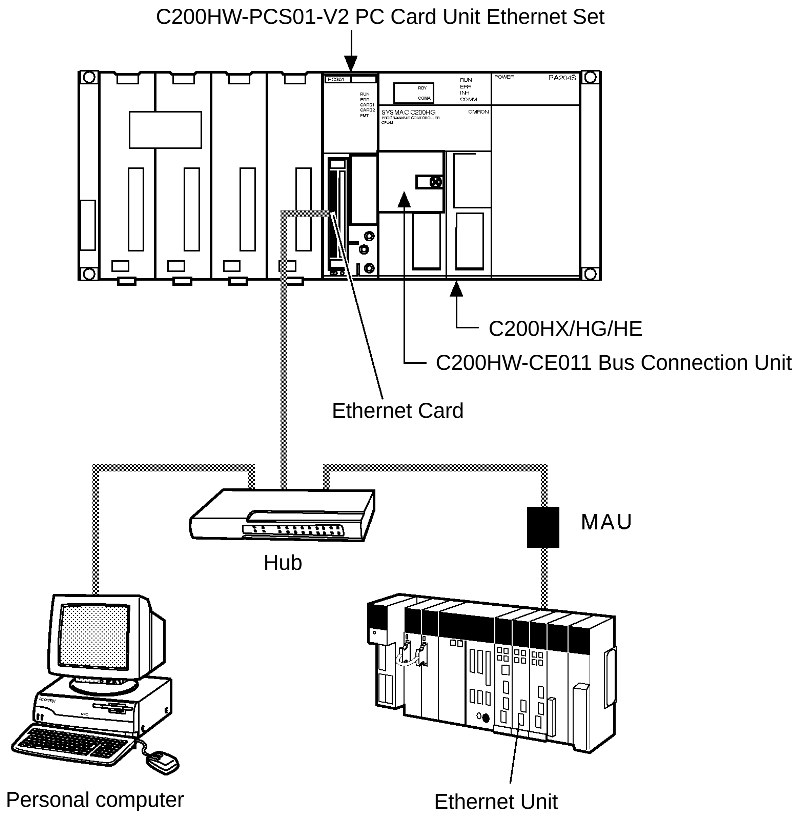
1-2 System Configuration

Systems using the PC Card Unit can be configured in either of two ways. The first is to use the C200HW-PCU01, which is the basic system for using PC cards. The second is to use the C200HW-PCS01-V2 PC Card Unit Ethernet Set, which allows you to add Ethernet functionality to the basic system.

Basic System Configuration (C200HW-PCU01)



Ethernet System Configuration (C200HW-PCS01-V2)


C200HW-PCS01-V2 PC Card Unit Ethernet Set
C200HX/HG/HE
C200HW-CE011 Bus Connection Unit
Ethernet Card
MAU
Hub
Personal computer
Ethernet Unit

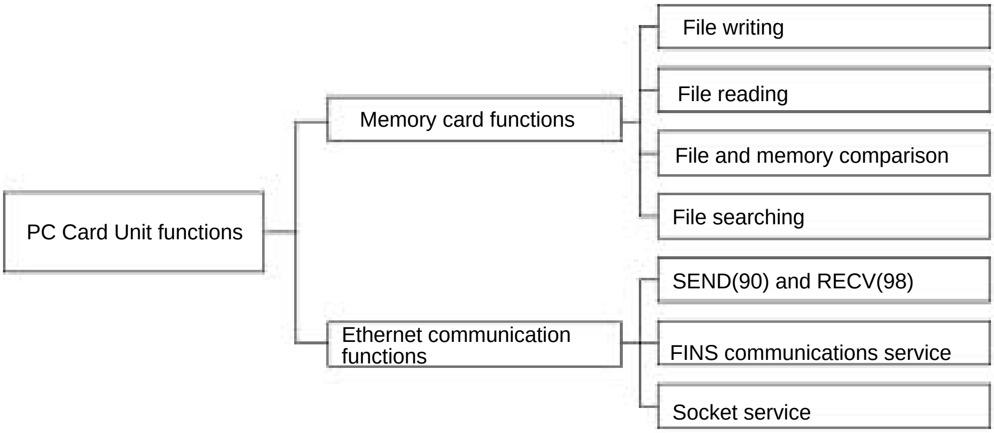
Note A MAU (Media Attachment Unit) is required to connect twisted-pair cable to a CV-series Ethernet Unit.

1-3 PC Card Unit Basics

This section provides a basic introduction to the PC Card Unit.

1-3-1 Functions

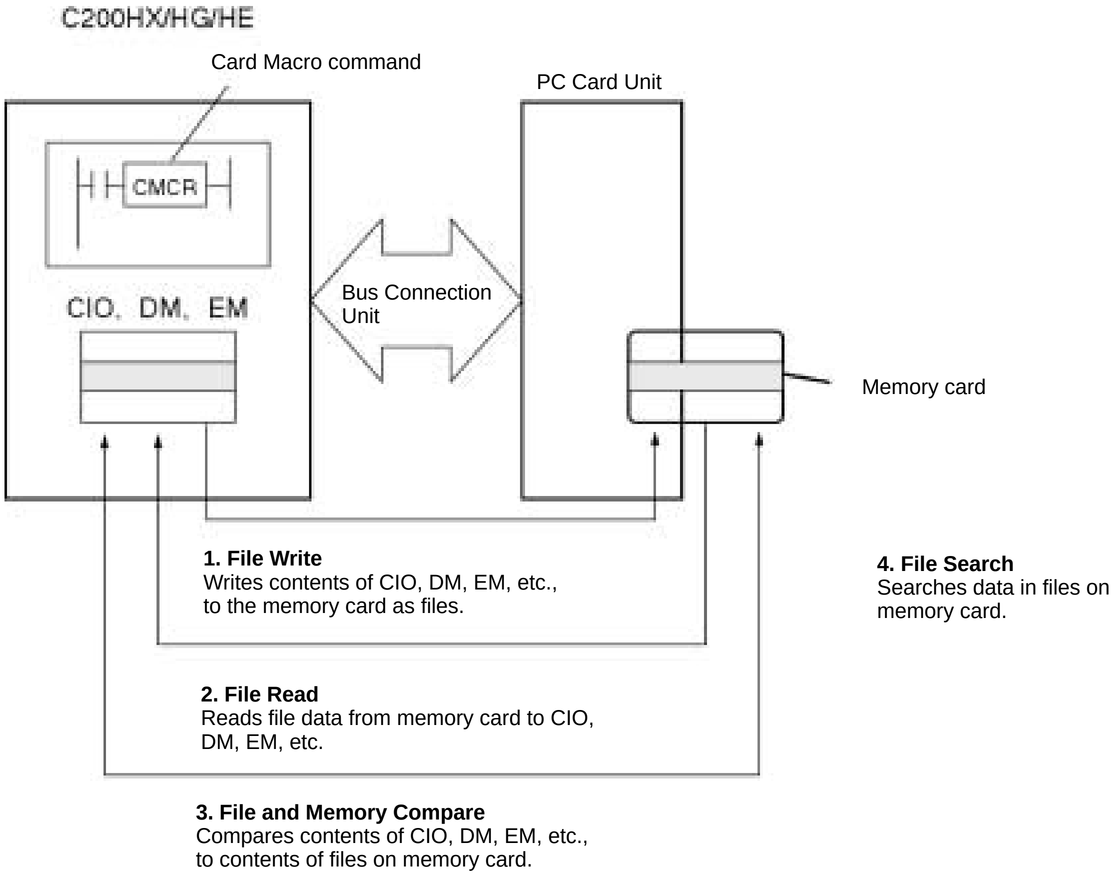
The PC Card Unit has both memory card functions and Ethernet communications functions, as shown in the following diagram.



The various services that can be performed using the memory card and Ethernet communications functions are described on the following pages.

1-3-2 Memory Card Functions

When the memory card functions are used, the following four operations can be executed with respect to memory cards.

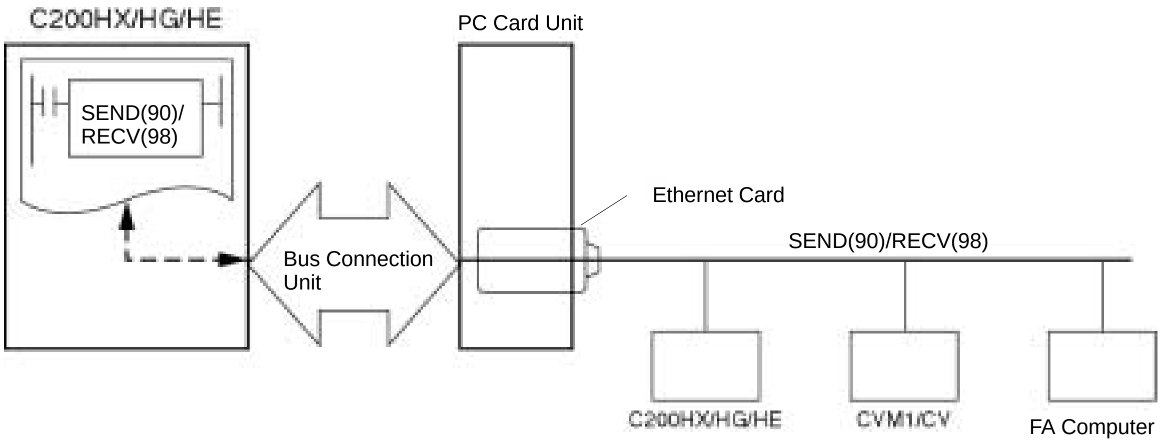

C200HX/HG/HE
Card Macro command
CMCR
CIO, DM, EM
Bus Connection Unit
PC Card Unit
Memory card
1. File Write
Writes contents of CIO, DM, EM, etc., to the memory card as files.
2. File Read
Reads file data from memory card to CIO, DM, EM, etc.
3. File and Memory Compare
Compares contents of CIO, DM, EM, etc., to contents of files on memory card.
4. File Search
Searches data in files on memory card.

Note Read and write operations for the ladder program cannot be executed.

1-3-3 Ethernet Communication Functions

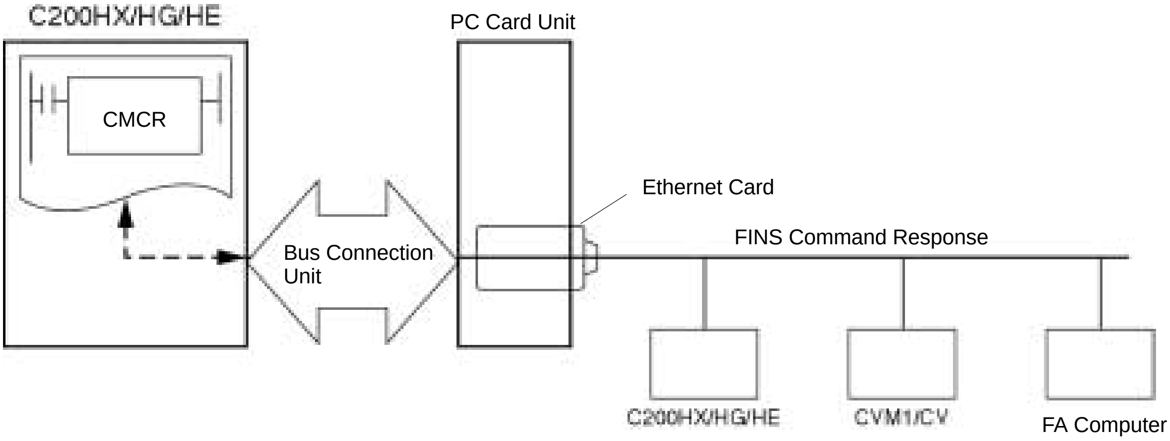
The C200HW-PCS01-V2 PC Card Unit Ethernet Set supports three communications methods. For details regarding Ethernet communications setup and methods, refer to *Section 6 Preparations for Operation* onwards.

Communications Using SEND(90) and RECV(98)



Data communications can be carried out by means of SEND(90) and RECV(98). The data to be sent or received must be set, but the communications processing is automatically performed internally. For details, refer to *Section 8 Using SEND(90) and RECV(98)*.

FINS Communications Service (FINS Command/Response)

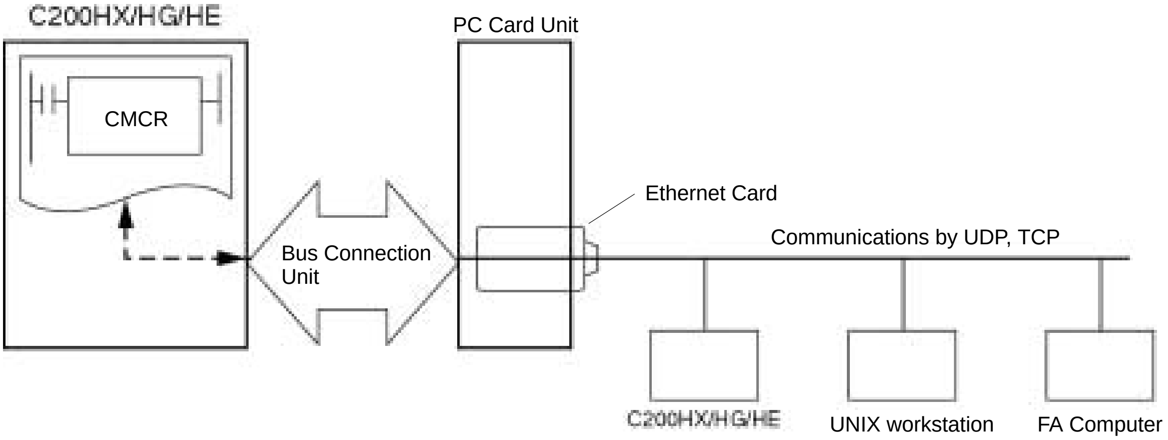


When PC Card Unit Receives a FINS Command
When the PC Card Unit receives a command, it automatically interprets the command and sends a response. All of the processing is performed automatically, so there is no need to create a user program at the Programmable Controller. For details, refer to *Section 9 FINS Commands*.

When a FINS Command is Sent From the PC Card Unit
A FINS command is issued using the CMCR instruction, and a response is returned from the recipient of the command. That response is returned automatically if the recipient of the FINS command is a C200HX/HG/HE or CVM1/CV Programmable Controller. For details, refer to *Section 9 FINS Commands*.

FINS Communications Between a Computer and a PC Card Unit
To use FINS commands from a computer, create the command data in the program at the computer according to the FINS command format. Have the FINS responses that are returned form the PC Card Unit analyzed and processed according to that format. For details, refer to *Section 9 FINS Commands*.

Socket Services



A socket is an interface which allows a user program to directly use TCP (Transmission Control Protocol) and UDP (User Datagram Program). Socket services are employed by using the CMCR instruction. Using socket services allows the PC Card Unit to communicate with UNIX workstations and FA Computers other than OMRON Programmable Controllers, with any protocol. For details, refer to *Section 10 Socket Services*.

SECTION 2
Components

This section describes individual product components and briefly explains the functions of each product. Unpack each product and make sure that it contains all the necessary components.

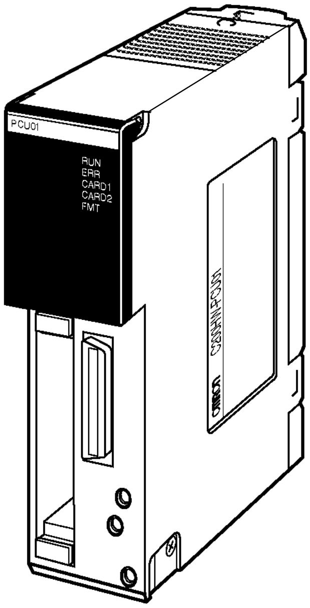
2-4 Other Required Peripheral Devices describes the devices that must be purchased separately by the customer.

2-1 C200HW-PCU01 PC Card Unit

This section describes the components of the C200HW-PCU01 PC Card Unit. Check the actual product against the drawings shown below.

2-1-1 PC Card Unit

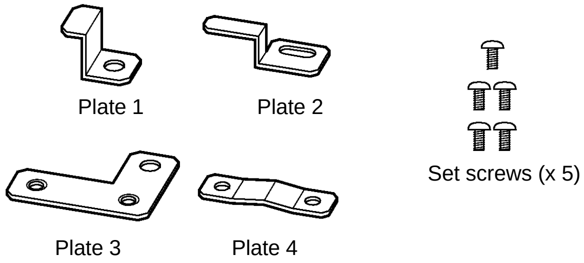
The PC Card Unit is the basic system that allows PC cards to be used in the C200HX/HG/HE.

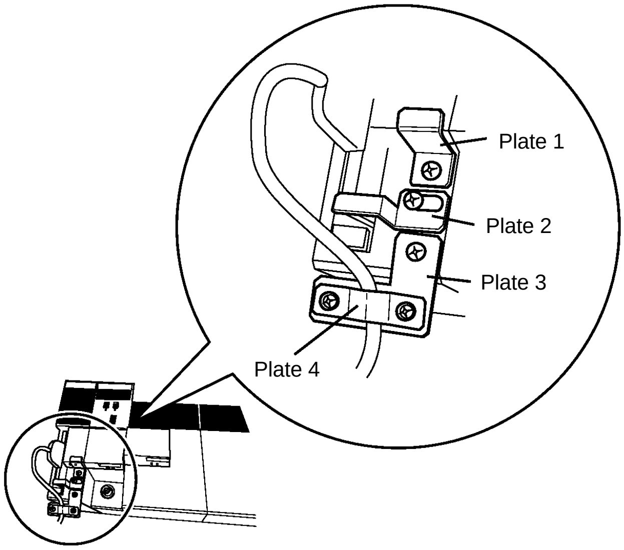


PC Card Unit

2-1-2 Plates and Set Screws

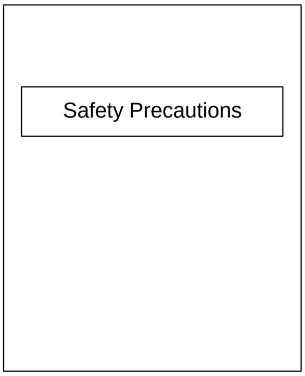
The following plates and set screws are used to secure the PC card, PC card cable, and Bus Connection Unit.





2-1-3 Safety Precautions

This sheet describes precautions to ensure safe use of the PC Card Unit. Be sure to read the sheet before using the PC Card Unit. This sheet is provided with the Unit.



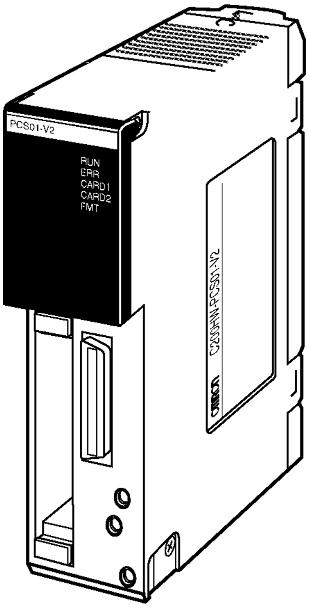
Safety Precautions

2-2 C200HW-PCS01-V2 Ethernet Set

This section describes the components of the C200HW-PCS01-V2 Ethernet Set. Check the actual product against the drawings shown below.

2-2-1 Ethernet Set

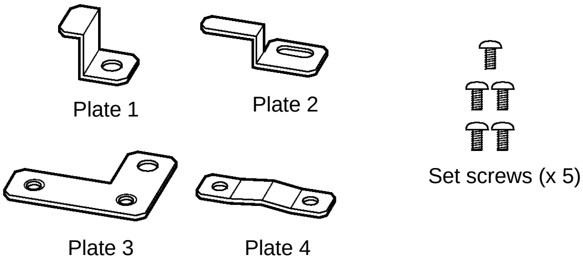
The Ethernet Set is a PC Card Unit (basic system) that has been equipped to support an Ethernet connection.

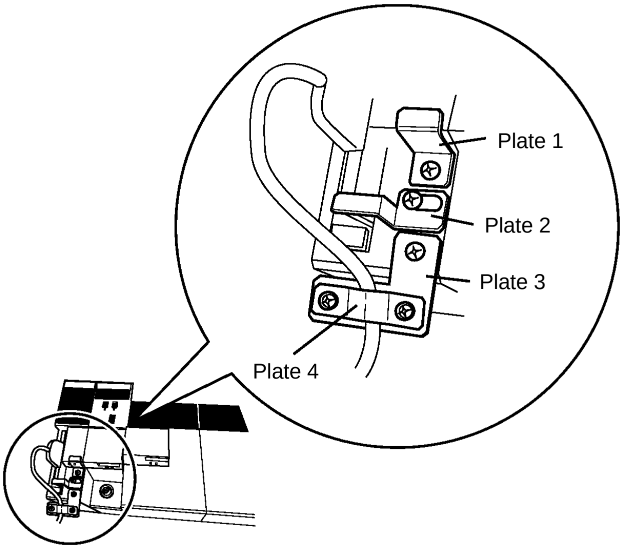


PC Card Unit

2-2-2 Plates and Set Screws

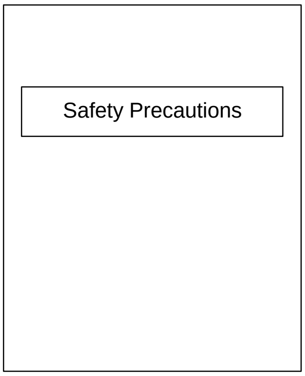
The following plates and set screws are used to secure the PC card, Ethernet cable, and Bus Connection Unit.





2-2-3 Safety Precautions

This sheet describes precautions to ensure safe use of the PC Card Unit. Be sure to read the sheet before using the PC Card Unit. This sheet is provided with the Unit.



2-3 C200HW-CE011/CE012 Bus Connection Unit

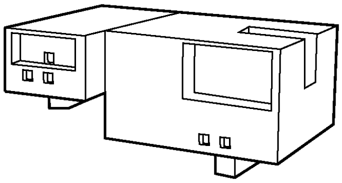
This section describes the components of the C200HW-CE011/CE012 Bus Connection Unit. Check the actual product against the drawings shown below.

Note 1. The previous Bus Connection Unit models, C200HW-CE001 and C200HW-CE002, cannot be used with a PC Card Unit.

2. The C200HW-CE011 and C200HW-CE012 are designed for a PC Card Unit and cannot be used with any other Units, except that the C200HW-CE012 also enables connection of a SYSMAC LINK or SYSMAC NET Link Unit.

2-3-1 C200HW-CE011 Bus Connection Unit

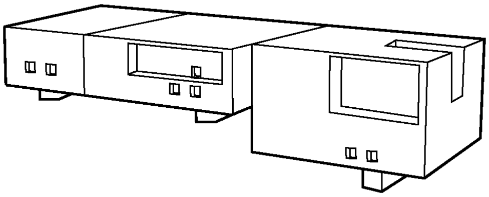
The C200HW-CE011 is used to connect a single PC Card Unit to the C200HX/HG/HE.



C200HW-CE011 Bus Connection Unit

2-3-2 C200HW-CE012 Bus Connection Unit

This Bus Connection Unit is used to connect one SYSMAC LINK Unit or SYSMAC NET Link Unit to the C200HX/HG/HE together with one PC Card Unit.



C200HW-CE012 Bus Connection Unit

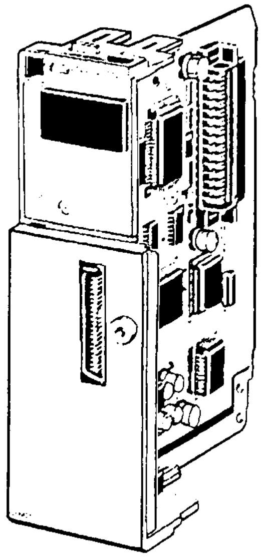
2-4 Other Required Peripheral Devices

This section describes the peripheral devices that must be prepared by customers to use a PC Card Unit. Refer to this section when purchasing peripheral devices.

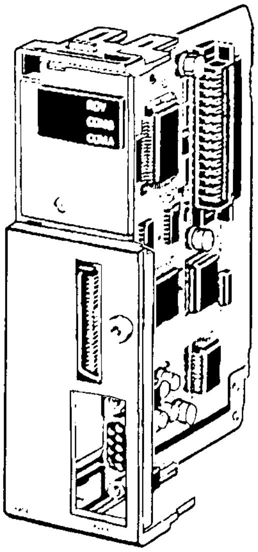
2-4-1 Devices Required for the PC Card Unit

To connect a PC Card Unit to the C200HX/HG/HE, a Communications Board must be mounted in the C200HX/HG/HE.

There are two Communications Boards that can be used. Use the C200HW-COM01 if only the PC Card Unit is to be used. Use the C200HW-COM04-E if a RS-232C port is required in the system in addition to the PC Card Unit.



C200HW-COM01 (with CPU bus interface only)

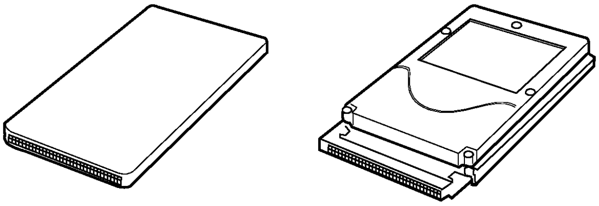


C200HW-COM04-E (with CPU bus interface and RS-232C port)

Note No Communications Board can be mounted in the C200HE-CPU11-E.

2-4-2 Devices Required for Memory Cards

Memory Cards



Applicable Memory Cards ATA-compatible cards can be used.

Note Before purchasing a memory card (SRAM, FLASH, or ATA), always make sure that it can be used with Phoenix PCM Plus 3.2.

2-4-3 Devices Required for Ethernet

The following devices are required to install Ethernet by using the setup software (installed on the Ethernet Set).

Host Link Cable The user must provide the host link cable to connect the computer to the PC Card Unit.

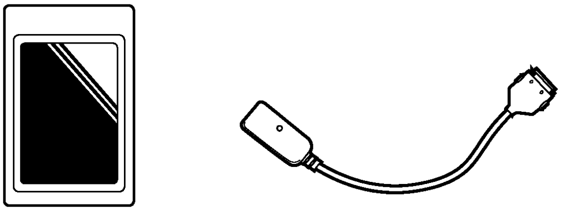
RS-232C connectors are used. Applicable connectors are as follows:

XM2A-0901 (connector)	OMRON
XM2S-0911 (connector cover)	OMRON

Personal Computer Prepare an AT or compatible computer. The computer must have at least one RS-232C port available.

Use an operating system supporting terminal software (e.g., HyperTerminal) that allows the exchange of binary files using the Zmodem protocol.
Example: Windows 98 or Windows NT 4.0.

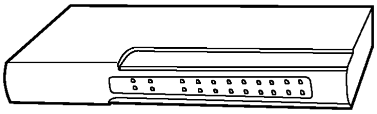
Ethernet Card



Ethernet Card Adaptor

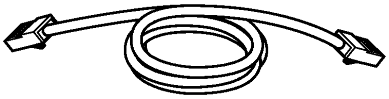
Applicable Ethernet Cards The PC Card Unit uses a DOS ODI driver. Although two DOS ODI drivers are currently in existence (SPEC3 and SPEC4), the PC Card Unit supports the SPEC3 ODI driver only. Therefore, Ethernet cards that have only the SPEC4 ODI driver cannot be used.

Hub



Hub

Cable



Twisted-pair Cable

Part 2
Installation and Operation

This part of the manual describes how to install a PC Card Unit and set up the C200HX/HG/HE. It includes information on memory card formatting, the file format, and file transfer operations between a PC Card Unit and the C200HX/HG/HE.

SECTION 3
Preparations for Operation

This section outlines the steps required to install the PC Card Unit and peripheral devices and prepare for operation. Be sure you have read this section and understood all of the procedures before attempting to actually make the settings or do any programming.

3-1 Outline

The procedures to prepare for operation are outlined below. Be sure to familiarize yourself with this basic procedure.

Procedures are explained in more detail in *3-2 Procedures*.

1, 2, 3...

1. **Communications Board Installation**

   Mount the Communications Board to the C200HX/HG/HE.

2. **C200HX/HG/HE Setup**

   Assign a function code for the CARD MACRO (CMCR) expansion instruction.

3. **PC Card Unit Setup**

   Mount the PC Card Unit and make the required settings.

4. **Programming**

   Create the program.

5. **Program Debugging**

   Debug the program.

6. **Operation.**

3-2 Procedures

This section outlines the steps required to install and set up the PC Card Unit and peripheral devices and prepare for application development and operation. Be sure that you thoroughly understand all of these sets. References for further reading are provided for each procedure.

1. Communications Board

Mount either the C200HW-COM01 or C200HW-COM04-E Communications Board to the C200HX/HG/HE CPU. Refer to the *C200HW-COM01 to 06-E Communications Board Operation Manual* for details.

2. C200HX/HG/HE Setup

Two steps are required to prepare the C200HX/HG/H for PC Card Unit application.

- **Adding CMCR** *4-1*

  Use either the SYSMAC Support Software or the Programming Console to assign a function code to the CARD MACRO (CMCR) instruction.

- **System Switch Setting** *4-1*

  Turn ON DIP switch pin no. 4 to enable setting expansion instructions for the C200HX/HG/HE.

3. PC Card Unit Setup

Two steps are required to set up the PC Card Unit.

- **System Switch Setting** *4-3*

  Set the startup mode, memory card initialization type, the operating level, and other settings. If only the memory card function is to be used, set only the operating level and leave the other pins set to OFF.

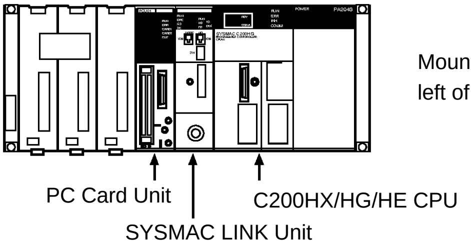
- **Unit and Peripheral Connections** *4-4 to 4-6*

  Mount the PC Card Unit to the Backplane and install the Bus Connection Unit and memory card.

Unit Restrictions and Mounting Locations

When a PC Card Unit is used, only one SYSMAC LINK Unit (or SYSMAC NET Link Unit) can be used on the CPU Backplane.

When a PC Card Unit and SYSMAC LINK Unit (or SYSMAC NET Link Unit) are used together on the CPU Backplane, they must be mounted in the slots shown in the following illustration.



Mount the PC Card Unit to the left of the SYSMAC LINK Unit

The previous Bus Connection Units (C200HW-CE001/002) cannot be used with the PC Card Unit. Likewise, the PC Card Unit’s Bus Connection Units (C200HW-CE011/012) cannot be used with other Units, except that the C200HW-CE012 can be used to also connect a SYSMAC LINK or SYSMAC NET Link Unit.

4. Programming

5-1 to 5-5

Use SYSMAC Support Software to create the ladder program. For details regarding ladder programming, refer to the *SYSMAC Support Software Operation Manuals.*

5. Debugging

5-6

Use a memory card with the program that has been created and correct any bugs that may be found.

6. Operation

Proceed with actual operation.

SECTION 4
Installation and Switch Settings

This section describes how to install a PC Card Unit and set up the C200HX/HG/HE.

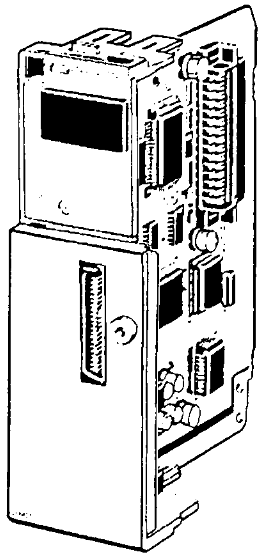
4-1 Before Installing a PC Card Unit

Before installing a PC Card Unit, set up the C200HX/HG/HE.

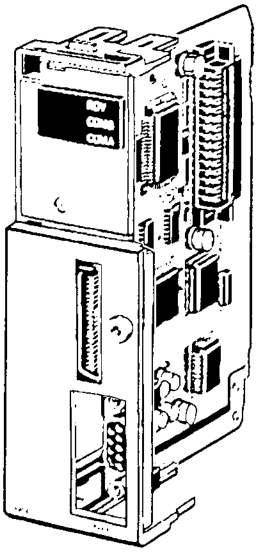
Note Always set up the C200HX/HG/HE. Otherwise, a PC Card Unit cannot be used.

4-1-1 Mounting a Communications Board

A Communications Board must be mounted in the C200HX/HG/HE to connect a PC Card Unit. There are two Communications Boards that can be used, as shown below. Use the Communications Board that matches your system requirements (i.e., the C200HW-COM04-E can be used to provide an RS-232C port in addition to the CPU bus interface for the PC Card Unit.



C200HW-COM01 (with CPU bus interface only)

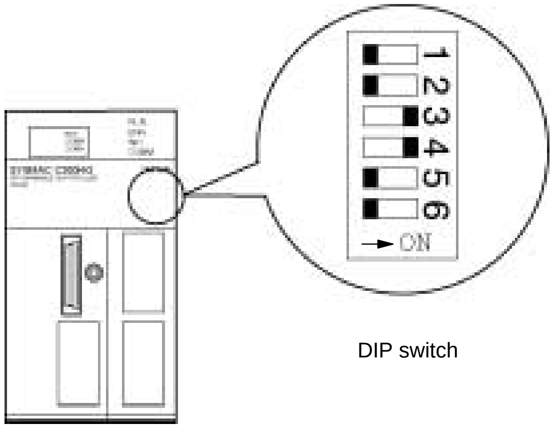


C200HW-COM04-E (with CPU bus interface and RS-232C port)

Refer to the *C200HW-COM01 to 06-E Communications Board Operation Manual* for details on installing a Communications Board.

4-1-2 Setting the C200HX/HG/HE System Switch

The CMCR expansion instruction is used to operate the memory card in the PC Card Unit. Turn ON pin no. 4 on the DIP switch to enable allocating expansion instructions for the C200HX/HG/HE.



4-1-3 Allocating a Function Code for CMCR

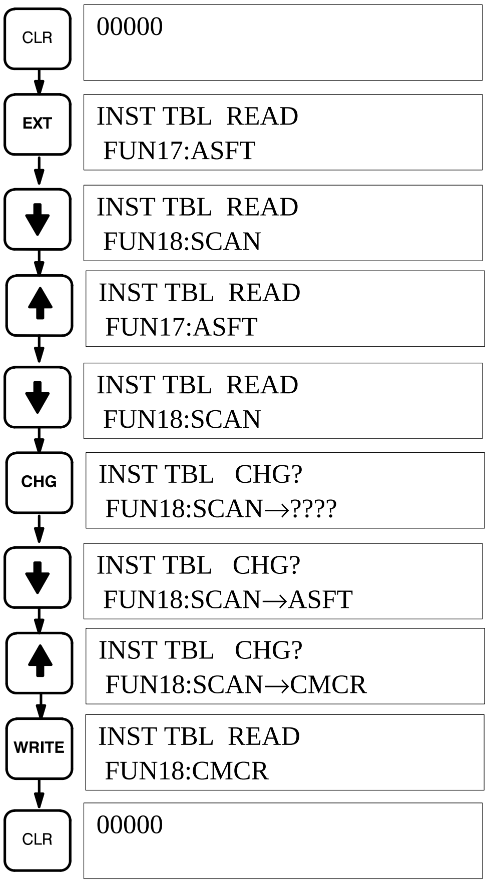
Use the SYSMAC Support Software or Programming Console to allocate a function code to the CMCR instruction (CARD MACRO instruction).

- The following 18 function codes can be allocated to expansion instructions: 17 to 19, 47, 48, 60 to 69, and 87 to 89.
- A default instruction is already allocated to each expansion function code.
- One instruction cannot be allocated to more than one function code.
- Information about the correspondence between expansion instructions and function codes is stored in the system area in the user program.
- This operation can be performed only when DIP switch pin no. 1 is OFF (user program write-enabled) and pin no. 4 is ON (settings other than default setting).
- Switch the PC operating mode to PROGRAM mode.

Programming Console

Execute the function for allocating/reading a function code to an expansion instruction.

Example



CLR	00000	Press CLR to bring up the initial display.
EXT	INST TBL READ FUN17:ASFT	Press EXT to begin displaying function code assignments.
↓	INST TBL READ FUN18:SCAN	Press the Up and Down Arrow keys to scroll through the function code assignments.
↑	INST TBL READ FUN17:ASFT	The Up Arrow key displays function codes in ascending order: 17, 18, ... , 89, 17, 18, ...
↓	INST TBL READ FUN18:SCAN	The Down Arrow key displays function codes in descending order: 17, 89, 88, ... 17, 89, ...
CHG	INST TBL CHG? FUN18:SCAN→????	Press CHG to change the displayed function code assignment.
↓	INST TBL CHG? FUN18:SCAN→ASFT	Press the Up and Down Arrow keys to scroll through the instructions.
↑	INST TBL CHG? FUN18:SCAN→CMCR	
WRITE	INST TBL READ FUN18:CMCR	Press WRITE to enter the change into memory.
CLR	00000	Press CLR to return to the initial display.

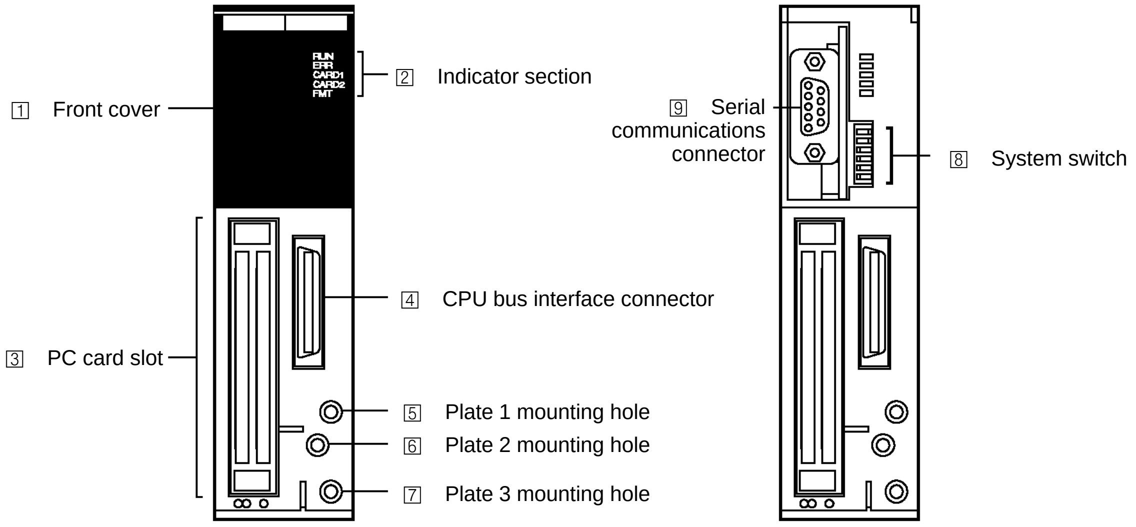
SYSMAC Support Software

Refer to the allocation procedure for function codes in the *SYSMAC Support Software Operation Manual: C-series PCs.*

4-2 Component Names and Functions

This section describes the name and function of each component of a PC Card Unit.

4-2-1 Front View



(When front cover is opened)

Name and Function of Each Component

The page numbers in parentheses indicate where relevant procedures are explained.

1 **Front cover (p.31)**
Open the front cover when setting the system switch or connecting the Ethernet setup cable.

2 **Indicator section (p.30)**
The indicators show the current operation status.

3 **PC card slot (p.42)**
A PC card is mounted in this slot.

4 **CPU bus interface connector (p.38)**
This connector allows data exchange with the Programmable Controller. Use the C200HW-CE011 or C200HW-CE012 Bus Connection Unit to connect the PC Card Unit to the Programmable Controller.

5 **Plate 1 mounting hole (p.41)**
This screw hole is used to fix Plate 1 to secure a Bus Connection Unit in place.

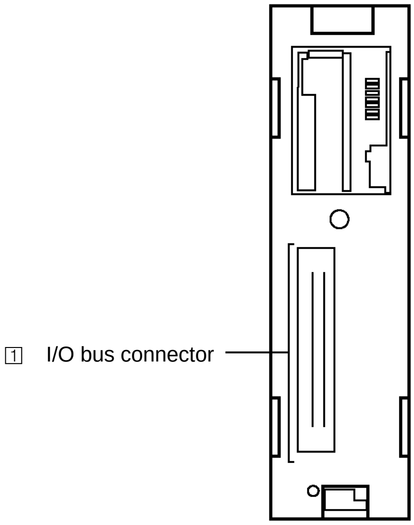
6 **Plate 2 mounting hole (p.43)**
This screw hole is used to fix Plate 2 to secure a PC Card in place.

7 **Plate 3 mounting hole**
This screw hole is used to fix Plate 3 to secure the adapter cable attached to the Ethernet Card.

8 **System switch (p.31)**
This DIP switch is used to set the startup mode, memory card format, format slot, and operating level. It is also used to start formatting a memory card.

9 **Serial communications connector (p.83)**
This connector is used to set up Ethernet. Connect the personal computer used for setup to the serial communications connector. The signal pin layout is the same as that of the Host Link connector.

4-2-2 Rear View



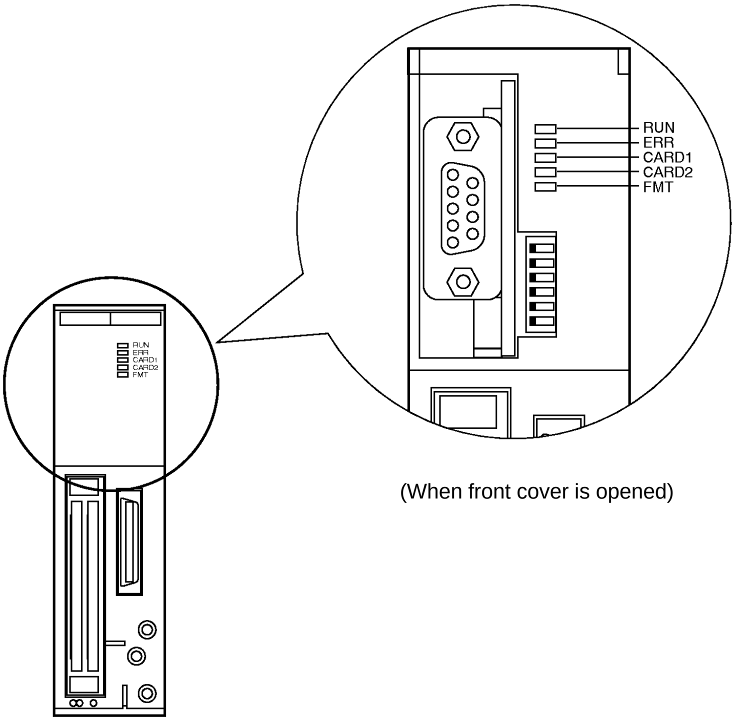
Name and Function of Component

The page number in parentheses indicates where the relevant procedure is explained.

1 I/O bus connector (p.35)
This connector is used to supply electrical power from the C200HX/HG/HE to the PC Card Unit.

4-2-3 Indicator Section

The indicators show the operating status of the PC Card Unit, the PC card slot, etc. Refer to *12-1 Indicators and the Error Log* for troubleshooting procedures.



(When front cover is opened)

Meaning of Indicators

Name	Color	Description	Meaning (when lit)
RUN	Green	Unit running	Lights when the Unit is operating (see note 1) or when files are being transferred.
	Flashing	Data is being read or written	(See note 2)
ERR	Red	Error	Lights when an error has occurred in the Unit (see note 3).
CARD1	Orange	Card 1	Lights when PC card socket 1 is being accessed.
CARD2	Orange	Card 2	Lights when PC card socket 2 is being accessed.
FMT	Orange:	Format mode:	Lights when the Unit is in PC Card formatting mode.
	Flashing	Format standby	
	Lit	Formatting	

Note

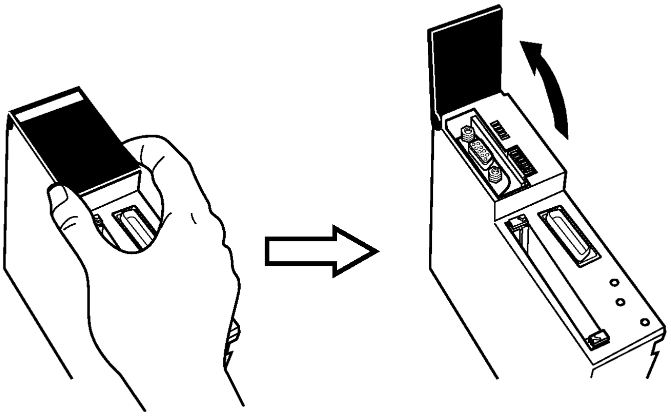
1. The meaning of the status “the Unit is operating” differs depending on the system.
   When only the PC Card Unit is used: Status in which the CMCR instruction can be executed in the ladder program.
   When Ethernet is used: Status in which IP protocol can be used (The RUN indicator may light even when the C200HX/HG/HE is stopped.)
2. The RUN indicator flashes when data is being read or written between the PC Card Unit and the C200HX/HG/HE. When the card is removed or inserted the indicator may go off temporarily.
3. The ERR indicator lights if the Ethernet setup file is incorrect or no Ethernet card is mounted when the Ethernet is being used.

4-3 Setting the System Switch

The system switch is used to set the PC Card Unit. Use this switch to set the startup mode, memory card format, format slot, and operating level. The system switch can also be used to start formatting a memory card.

4-3-1 Opening the Front Cover

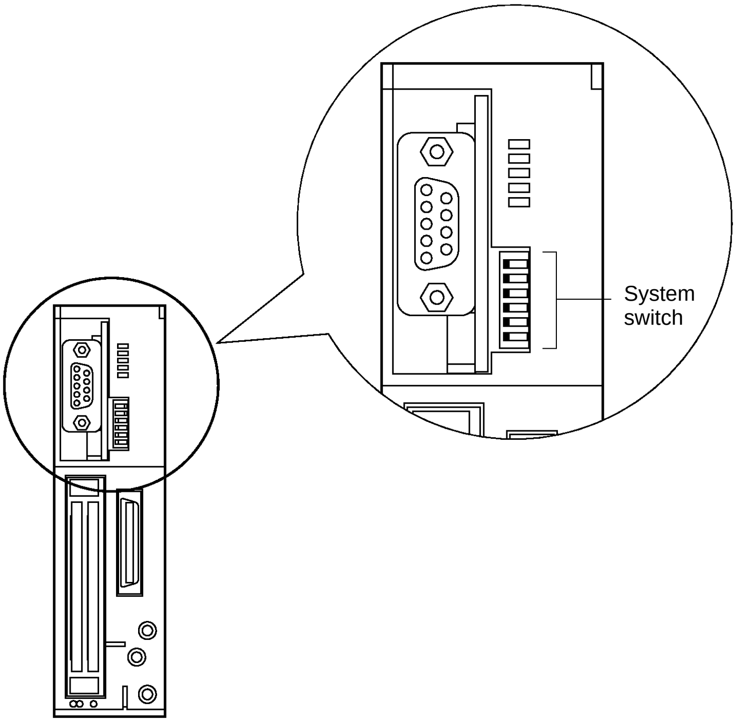
Before setting the system switch, open the front cover as shown below.



4-3-2 System Switch Functions

The system switch functions are described below.

Note Restart the PC Card Unit to use the new system switch settings.



System Switch Functions

Pin No.	Function	OFF	ON
6, 5	Sets the startup mode	See the table below.	
4	Specifies the card format	SRAM/ATA card (FAT format)	FLASH card (MS-FLASH format)
3	Specifies the slot to be formatted	Slot 1	Slot 2
2	Starts formatting (see note)	Formatting is started by turning the pin OFF and then ON, or ON and then OFF.	
1	Sets the operating level	Level #1	Level #0

Note Pin 2 is valid only in memory card formatting mode (when the FMT indicator is flashing).

Startup Mode

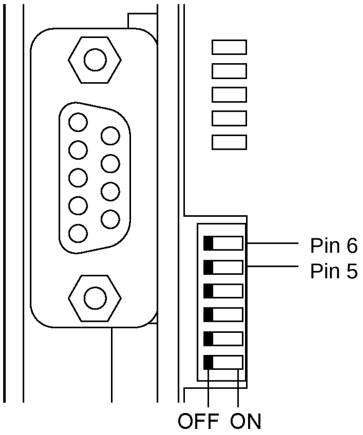
Pin 6	Pin 5	Startup mode
OFF	OFF	Normal mode
OFF	ON	File transfer mode
ON	OFF	Memory card formatting mode

Note
1. In normal mode, always turn OFF pins 2, 3, and 4.
2. By default, all pins are set to OFF. Change the switch settings as necessary.

4-3-3 Setting the Startup Mode

Pins 6 and 5 are used to set the startup mode. The setting procedure and pin settings are described below.

Note Restart the PC Card Unit to use the new system switch settings. By default, all pins are set to OFF. Change the switch settings as necessary.



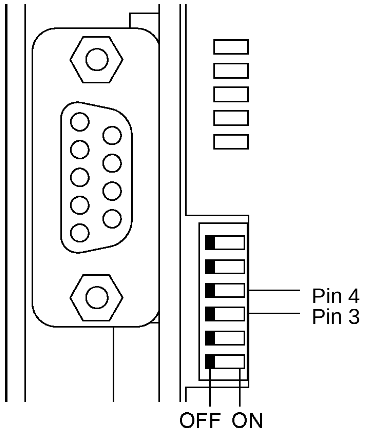
Startup Mode Settings (Pins 6 and 5)

Pin 6 and pin 5 settings	Description
OFF ■□ OFF ■□	Normal mode (pins 2, 3, and 4 must be turned OFF.)
OFF ■□ ON □■	File transfer mode This mode is used to transfer the settings with the Ethernet setup software.
ON □■ OFF ■□	Memory card formatting mode The FMT indicator flashes. This mode is used to format the card inserted in the PC card slot. SRAM, ATA, and FLASH cards can be inserted in the slot.

4-3-4 Setting the Card Format and Slot

Pins 4 and 3 are used to set the card format and the slot to be formatted. The setting procedure and pin settings are described below.

Note Restart the PC Card Unit to use the new system switch settings. By default, all pins are set to OFF. Change the switch settings as necessary.



Card Format and Slot Settings (Pins 4 and 3)

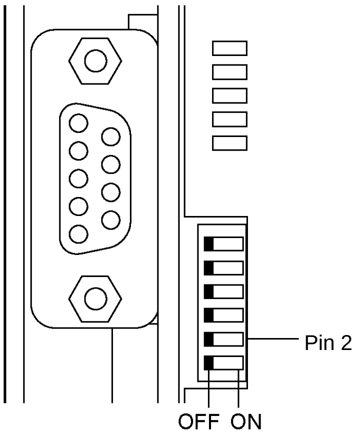
Pin 4 and pin 3 settings	Description	
OFF ■□ OFF ■□	SRAM and ATA cards	Format card in slot 1
OFF ■□ ON □■	SRAM and ATA cards	Format card in slot 2
ON □■ OFF ■□	FLASH card	Format card in slot 1
ON □■ ON □■	FLASH card	Format card in slot 2

Note Memory cards can also be formatted using PCMCIA2.1-compliant equipment such as personal computers. In this case, SRAM and ATA Cards must be formatted in FAT format, and FLASH Cards must be formatted in MS-FLASH format.

4-3-5 Formatting Memory Cards

Note Formatting a memory card erases all data from the card. Before formatting a memory card, always back up all the necessary data from the card.

Turning pin 2 OFF then ON, or ON then OFF starts formatting the memory card. Pin no. 2 is valid, however, only when the startup mode is set to the memory card formatting mode (FMT indicator will be lit).



If formatting starts normally, the FMT indicator will light and the indicator for the PC card slot will flash.

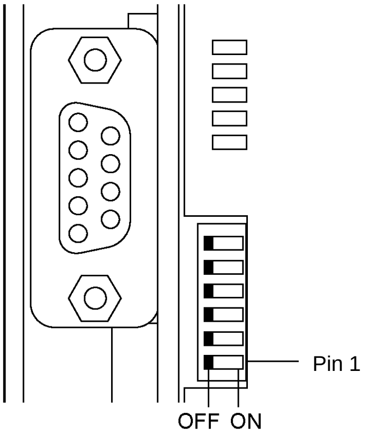
The FMT indicator will go OFF when formatting is complete.

Turn the power OFF, then remove the card.

Note Refer to *5-2 Formatting Memory Cards* for details on formatting memory cards.

4-3-6 Setting the Operating Level

Pin 1 is used to set the PC Card Unit operating level. The procedure and pin settings are described below.



Operating Level (Pin 1)

Pin 1 setting	Description
OFF ■□□	Level #1
ON □□■	Level #0

If the SYSMAC LINK Unit or SYSMAC NET Link Unit is to be used together with a PC Card Unit, select a different level.

Note Do not switch the operating level while the PC Card Unit is operating. Otherwise, the Unit will fail to operate normally.

4-4 Mounting the PC Card Unit

This section describes how to mount the PC Card Unit on the CPU Backplane.

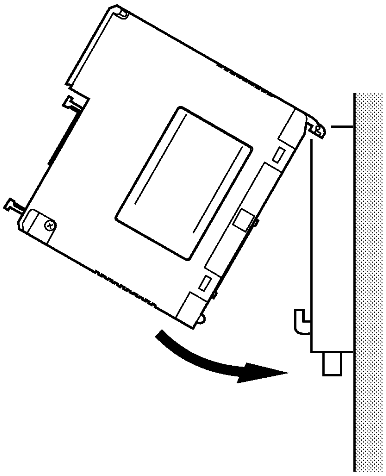
Note Before mounting the PC Card Unit on a CPU Backplane, always mount the C200HW-COM01 or C200HW-COM04-E Communications Board in the C200HX/HG/HE. Refer to the *C200HW-COM01 to 06-E Communications Board Operation Manual* for the mounting procedure.

4-4-1 When a SYSMAC LINK Unit or SYSMAC NET Link Unit is Not Used

Note Always install the PC Card Unit in the slot on the left of the C200HX/HG/HE CPU. If the PC Card Unit is installed in another slot, the Bus Connection Unit cannot be installed.

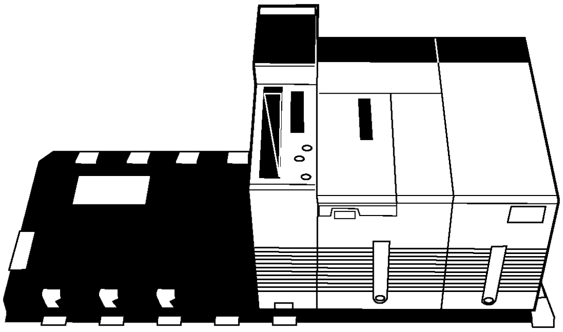
1, 2, 3...

1. Hitch the claw on the upper edge of the bottom of the PC Card Unit to the CPU Backplane as shown in the figure below.

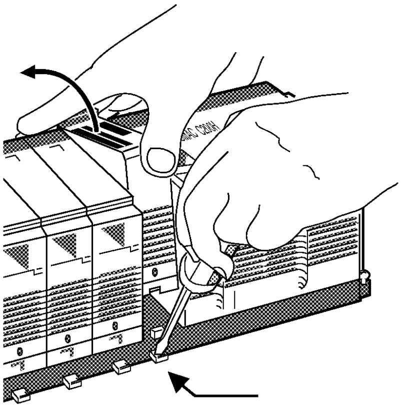


2. Insert the connector on the PC Card Unit all the way into the connector on the CPU Backplane.

   When the PC Card Unit is correctly installed a click will be audible. Insert the connector until the click is heard.



Note To remove the PC Card Unit, lift the Unit while pressing the lock lever with a screwdriver as shown in the figure below.

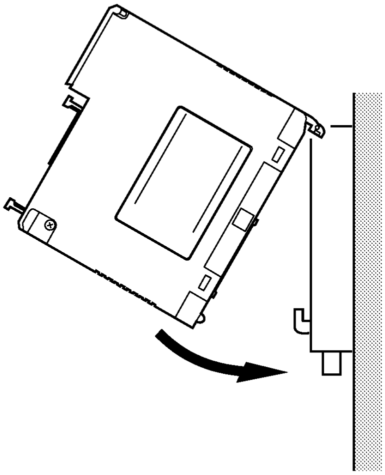


4-4-2 When a SYSMAC LINK Unit or SYSMAC NET Link Unit is Used

Note Always mount the SYSMAC LINK Unit or SYSMAC NET Link Unit in the slot on the left of the C200HX/HG/HE CPU. Then, mount the PC Card Unit in the slot on the left of the SYSMAC LINK Unit or SYSMAC NET Link Unit. If these Units are mounted in other slots, the Bus Connection Unit cannot be mounted.

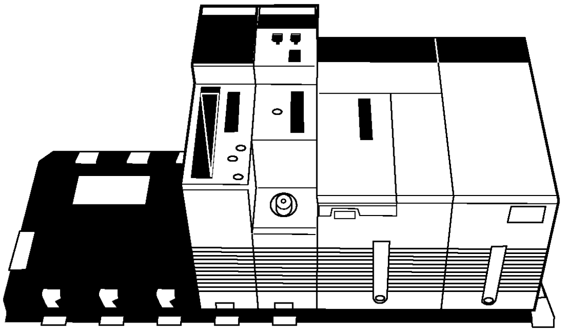
1, 2, 3...

1. Hitch the claw on the upper edge of the bottom of the PC Card Unit to the CPU Backplane as shown in the figure below.



2. Insert the connector on the PC Card Unit all the way into the connector on the CPU Backplane.

   When the PC Card Unit is correctly installed a click will be audible. Insert the connector until the click is heard.



4-5 Installing the Bus Connection Unit

There are two Bus Connection Units that can be used: C200HW-CE011 and C200HW-CE012.

The C200HW-CE011 is used to connect a PC Card Unit to the C200HX/HG/HE. The C200HW-CE012 is used to connect a SYSMAC LINK Unit or SYSMAC NET Link Unit to the C200HX/HG/HE together with a PC Card Unit.

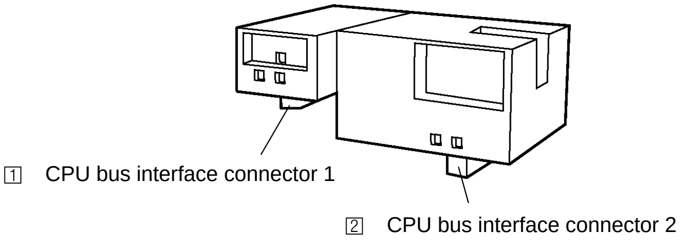
This section describes how to install the C200HW-CE011 and C200HW-CE012.

4-5-1 Installing the C200HW-CE011

Before installing the C200HW-CE011 Bus Connection Unit, mount the C200HX/HG/HE CPU and PC Card Unit in the correct slots in the CPU Backplane.

Name and Function of Each Component

The name and function of each component are described below.


1 CPU bus interface connector 1
2 CPU bus interface connector 2

C200HW-CE011 Bus Connection Unit

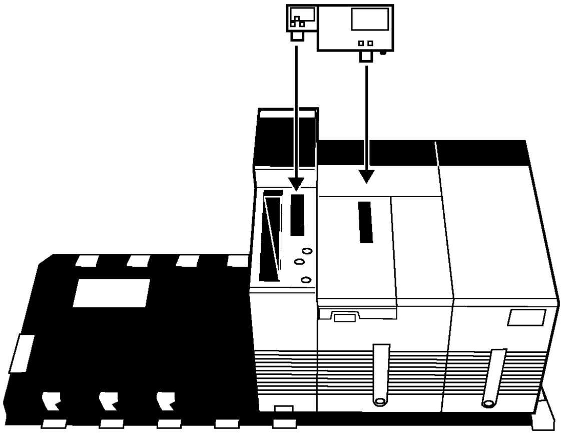
1 **CPU bus interface connector 1**
This connector connects to the CPU Bus Interface Connector on the PC Card Unit.

2 **CPU bus interface connector 2**
This connector connects to the CPU Bus Interface Connector on the C200HX/HG/HE CPU.

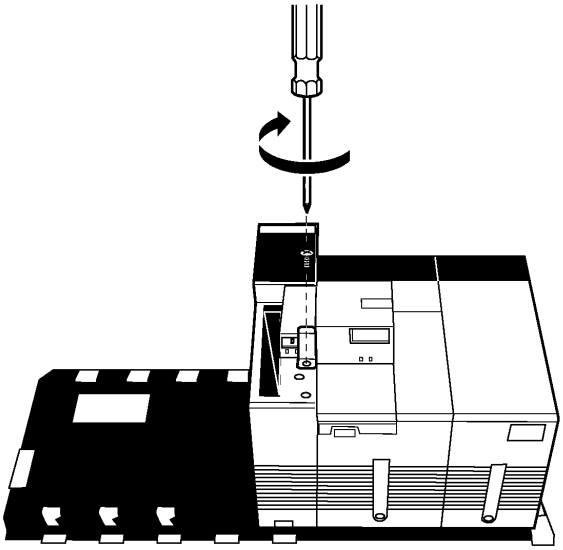
Installation Procedure

1, 2, 3...

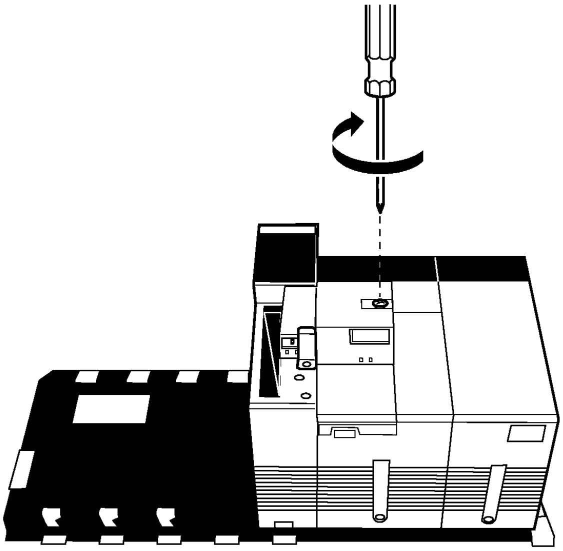
1. Connect the CPU bus interface connectors 1 and 2 to the two connectors in the Bus Connection Unit as shown in the figure below and press them firmly into place.



2. Secure the Bus Connection Unit with plate 1. To do so, secure plate 1 by tightening a screw into the plate-1 mounting hole as shown in the figure below.



3. Secure the Bus Connection Unit to the C200HX/HG/HE CPU with a screw as shown in the figure below.

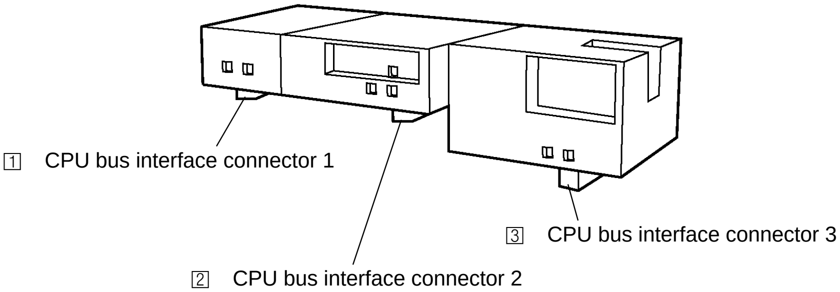


4-5-2 Installing the C200HW-CE012

Before installing the C200HW-CE012 Bus Connection Unit, mount the C200HX/HG/HE CPU, SYSMAC LINK Unit or SYSMAC NET Link Unit, and PC Card Unit in the correct slots in the CPU Backplane.

Name and Function of Each Component

The name and function of each component are described below.


1 CPU bus interface connector 1
3 CPU bus interface connector 3
2 CPU bus interface connector 2

C200HW-CE012 Bus Connection Unit

1 CPU bus interface connector 1
This connector connects to the CPU bus interface connector on the PC Card Unit.

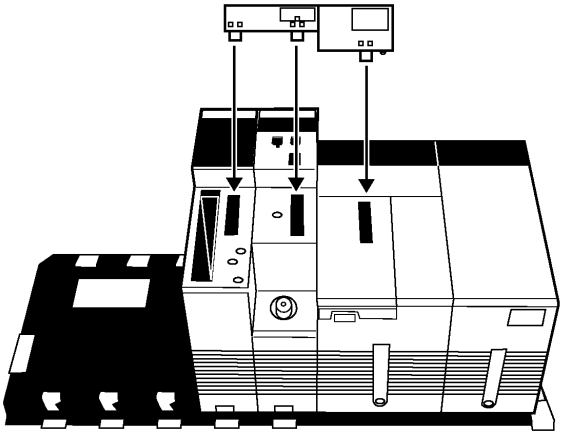
2 CPU bus interface connector 2
This connector connects to the CPU bus interface connector on the SYSMAC LINK Unit or SYSMAC NET Link Unit.

3 CPU bus interface connector 3
This connector connects to the CPU bus interface connector on the C200HX/HG/HE CPU.

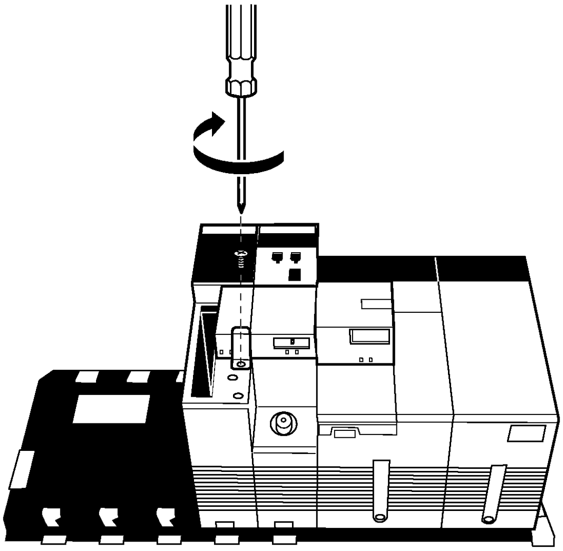
Installation Procedure

1, 2, 3...

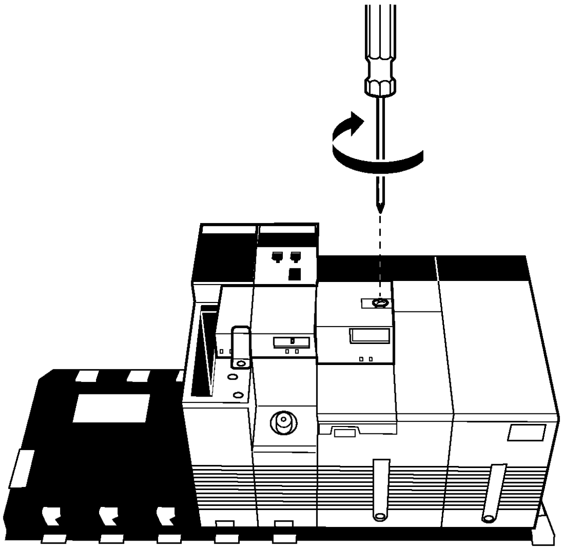
1. Connect the CPU bus interface connectors 1, 2, and 3 all the way into the three corresponding connectors on the Bus Connection Unit as shown in the figure below.



2. Secure the Bus Connection Unit with plate 1. To do so, secure plate 1 by tightening a screw into the plate-1 mounting hole in the front surface of the PC Card Unit as shown in the figure below.



3. Secure the Bus Connection Unit to the C200HX/HG/HE CPU with a screw as shown in the figure below.



4-6 Installing and Removing Memory Cards

This section includes information on the memory cards to be used and explains how to install memory cards in the PC card slots in the front surface of the PC Card Unit.

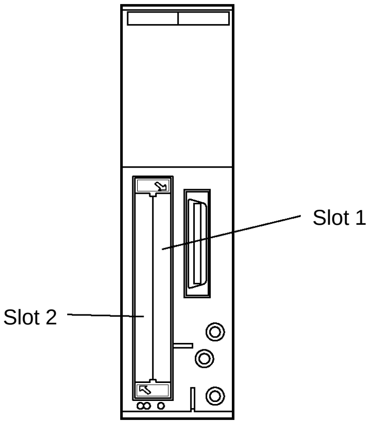
4-6-1 Memory Cards

Use memory cards that fall within the following range of current consumption:

I_{5V} (1 slot) ≤ 0.5 A, I_{12V} (1 slot) ≤ 0.1 A
I_{5V} (2 slots) + 3.4 x I_{12V} (2 slots) ≤ 1.0 A

Before purchasing memory cards (SRAM, FLASH, or ATA), always make sure that they can be used with Phoenix PCM Plus 3.2.

4-6-2 PC Card Slots



Slot 1: A type-I, -II, or -III PC card can be mounted in this slot.

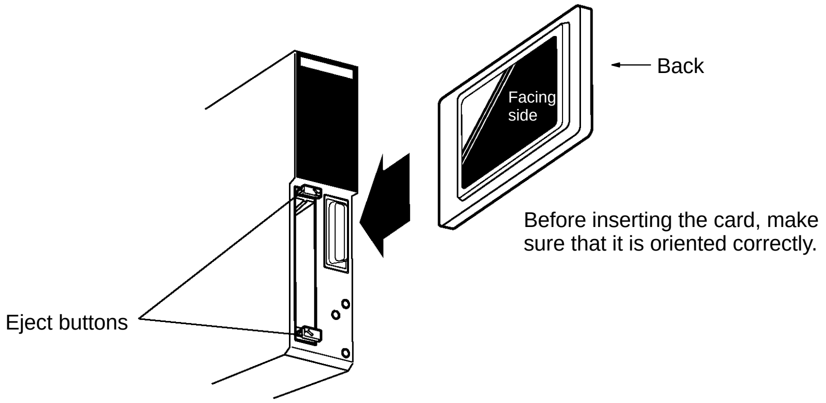
Slot 2: A type-I or -II PC Card can be mounted in this slot. If, however, a type-III PC Card is mounted in slot 1, slot 2 cannot be used.

Note Always use the same type of card in each slot. If a different type of card is mounted in the same slot, an error will occasionally occur depending on the types of the cards.

4-6-3 Installing a Memory Card

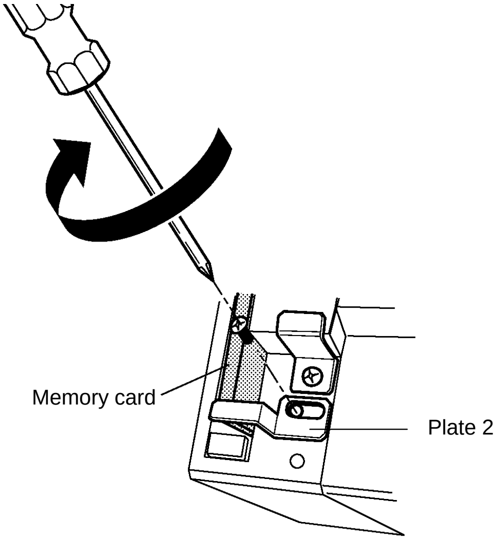
1, 2, 3...

1. Mount a memory card in a PC card slot as shown below. Press the memory card until it is aligned with the eject buttons.


Back
Facing side
Before inserting the card, make sure that it is oriented correctly.
Eject buttons

2. If another memory card is to be installed, repeat Step 1.

3. Secure the memory card with plate 2. To do so, secure plate 2 by tightening a set screw into the plate 2 mounting hole in the front surface of the PC Card Unit as shown below.



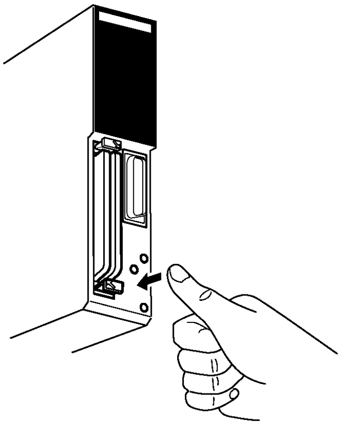
Note Do not press the eject button with Plate 2 mounted.

The Ethernet card cannot be installed or removed when the power is ON. Always turn off the power before installing or removing the Ethernet Card.

Removing the Memory Card

1, 2, 3...

1. Remove the set screw, then remove plate 2.
2. Press the eject button as shown below.



The card inserted into slot 2 can be removed or installed by loosening the set screws on plate 2. When removing or installing a card into slot 1, always remove plate 2.

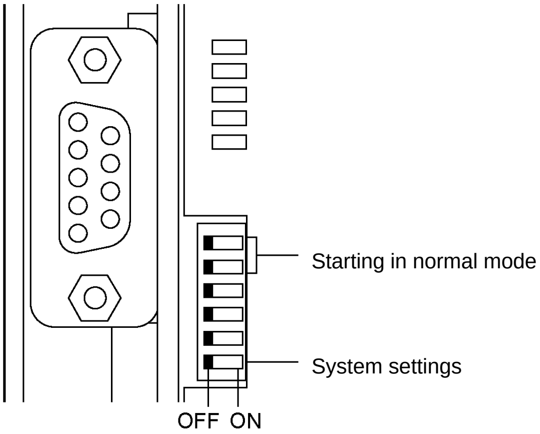
4-7 Starting the PC Card Unit

Turning on the C200HX/HG/HE automatically starts the PC Card Unit.

Before turning on the C200HX/HG/HE to start the PC Card Unit, check the items described in this section. Also, after the PC Card Unit is started, check the indicators and SR bits to confirm that operation is normal.

Checking the System Switch Check the startup mode and system settings.

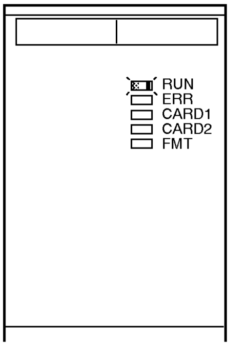
- Make sure that pins 2 to 6 are all set to OFF.
- Make sure that pin 1 is set to the correct operating level for your system.



Checking the Memory Card Make sure that the memory card is correctly mounted in the PC Card Unit.

Turning ON the C200HX/HG/HE Turn on the C200HX/HG/HE.

Checking Operation with Indicators If the PC Card Unit is started normally, the RUN indicator will light, If the ERR indicator lights, eliminate the cause of the error as described in *12-1 Indicators and the Error Log*.



Checking System Operation with SR Bits Check whether the PC Card Unit has been recognized by the C200HX/HG/HE. Use the ladder program to read word SR 252. If bit 01 is ON, the PC Card Unit has been correctly recognized by the C200HX/HG/HE.

Word address	Bit number	Operating level
SR 252	01	#0
	04	#1

Note Operations from the time the power is switched on until the RUN indicator is lit.

- The SEND/RECV Enable Flag (SR 252) turns ON immediately after the power is turned on.
- The instruction will be executed after the RUN indicator lights.

- The Data Link Operating Flag turns ON when the power is turned on and turns OFF when the RUN indicator is lit.
- It takes 25 seconds until the RUN indicator is lit (35 seconds when Ethernet is built-in). During this time the PC Card driver (socket service and card service), the Ethernet ODI driver, and the UDP/IP protocol are loaded into the Unit.

AR Restart

When the relevant bit on word AR01 is turned from ON to OFF, the PC Card Unit is restarted. For operating level 0, turn bit 11 from ON to OFF, for operating level 1 turn bit 10 from ON to OFF.

Clock Specifications

When the PC Card Unit is restarted, either by resetting the power supply or using AR restart, read the C200HX/HG/HE clock to set the clock on the PC Card Unit.

SECTION 5
Using Memory Cards

This section describes formatting Memory Cards, file formats, and file transfers between the PC Card Unit and the C200HX/HG/HE CPU.

5-1 Outline

The PC Card Unit allows data, such as CIO, DM, and EM data, to be transferred between a memory card mounted in the PC Card Unit and the C200HX/HG/HE CPU. File read/write operations are performed using the CMCR instruction executed in the ladder program in the C200HX/HG/HE. Refer to *4-1 Before Installing a PC Card Unit* for details on allocating a function code to CMCR.

⚠ Caution When using the PC Card, be sure to observe the following precautions.

- Insert or eject the PC Card for at least 10 seconds after it is inserted or ejected. (That is, don't eject the PC Card for at least 10 seconds after it is inserted and don't insert the the PC Card for at least 10 seconds after it is ejected.)
- Don't eject the PC Card while it is being accessed (while either the CARD1 indicator or CARD2 indicator is lit).

Applicable PC Cards

The PC Card Unit has two PCMCIA 2.1-compliant PC card interface slots, in which two type-I or type-II PC cards or one type-III PC card can be mounted. Memory cards on the market, such as SRAM, ATA, and FLASH memory cards, can be used.

Before purchasing memory cards (SRAM, FLASH, or ATA), always make sure that they can be used with PHOENIX PCM Plus 3.2.

File Operations

The CMCR instruction (CARD MACRO) is used to transfer data between the PC Card Unit and the C200HX/HG/HE CPU.

If the user specifies the memory address from which data is to be read, the number of words, and a file name, data in C200HX/HG/HE memory can be written to the specified file in the memory card. Similarly, files can be written from the memory card to C200HX/HG/HE memory. C200HX/HG/HE files can also be compared and searched.

The PC Card Unit also supports saving data with words separated by commas (CSV) as the file format, allowing data to be processed with commercially available spreadsheet software.

File Save Format

Data can be saved in any of the following three file format. Select the appropriate format for your purpose.

- Single word comma separated value (CSV) format

  The contents of the specified words are saved in ASCII format. The data for each word is separated by a comma (4-byte ASCII characters).

- Double word CSV format

  The contents of the specified words are saved in ASCII format. The data for each pair of words is separated by a comma (8-byte ASCII characters).

- Binary format

  The contents of the specified words are saved directly in a file.

Specify the save format in the control data when programming the CMCR instruction. Refer to *5-3 File Operations: CMCR Instruction* for details.

Accessing Data from Personal Computers

Data stored in a memory card in a PC Card Unit can be accessed from a personal computer. Conversely, data stored in a personal computer can be written to the PC Card Unit and then accessed by the PC.

5-2 Formatting Memory Cards

This section describes how to format memory cards.

5-2-1 Specifying the Card Format and Slot

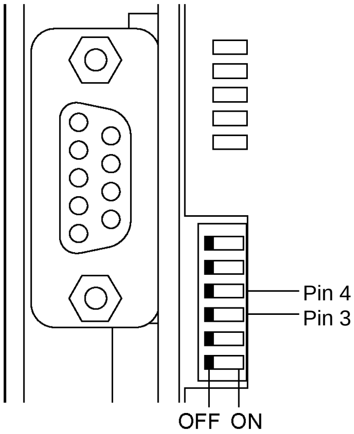
The format differs according to the memory card being used.

- SRAM and ATA Cards: DOS FAT format
- FLASH Cards: MS-FLASH format

Use the system switch on the front panel of a PC Card Unit to specify the format and the slot containing the card to be formatted.

Specifying the Card Format and Slot (Pins 4 and 3)

Note Restart the PC Card Unit to use new system switch settings. By default, all pins are set to OFF. Change the switch settings as necessary.



Pin 4 and pin 3 settings		Description	
OFF OFF	■□ ■□	SRAM/ATA Card	Format card in slot 1
OFF ON	■□ □■	SRAM/ATA Card	Format card in slot 2
ON OFF	□■ ■□	FLASH Card	Format card in slot 1
ON ON	□■ □■	FLASH Card	Format card in slot 2

Note Memory cards can also be formatted using PCMCIA2.1-compliant equipment such as personal computers. In this case, SRAM and ATA Cards must be formatted in FAT format, and FLASH Cards must be formatted in MS-FLASH format.

Newly purchased memory cards may not be formatted on the PC Card Unit. In such a case, format them using a personal computer.

If a personal computer is not PCMCIA2.1-compliant, format the SRAM Card with the following procedure using the PC Card Unit.

1, 2, 3...

1. Connect an ANSI terminal, such as the Hyperterminal for Windows 95, to the PC Card Unit using a host link cable. Set the communications conditions as follows:

   9,600 bps, 8 bits, no parity, 1 stop bit
2. Set only pins 5 and 6 to ON, and then turn on the PC Card Unit.
3. Insert the SRAM Card into slot 2.

4. Input the following underlined characters.

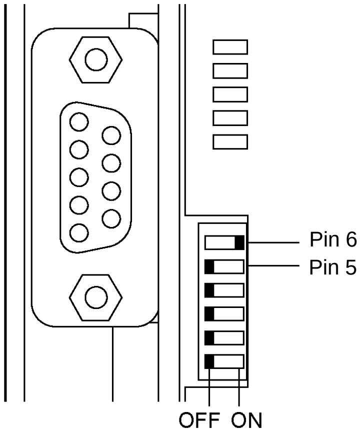
```
F:\>FORMAT H: /U↵
```

5. If an MS-DOS error message appears, select “FAIL.” The format will be forcibly executed.

5-2-2 Setting the Memory Card Formatting Mode

Use pins 6 and 5 to set the memory card formatting mode. The setting procedure and pin settings are described below.

Note Restart the PC Card Unit to use new system switch settings. By default, all pins are set to OFF. Change the switch settings as necessary.



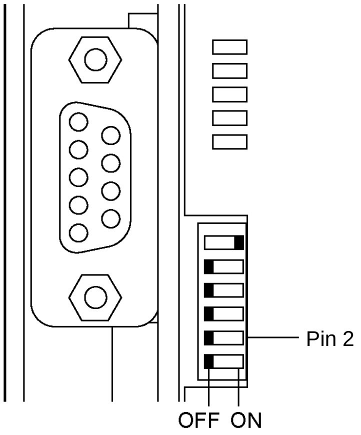
Start Mode Settings (Pins 6 and 5)

Pin 6 and pin 5 settings	Description
ON □■ OFF ■□	Memory card formatting mode. The FMT indicator flashes. This mode is used to format the card inserted in the PC card slot. SRAM, ATA, and FLASH cards can be inserted in the slot.

5-2-3 Starting Memory Card Formatting

Note Formatting a memory card erases all data from the card. Before formatting a memory card, always back up all the necessary data from the card.

Turning pin 2 OFF then ON, or ON then OFF starts formatting the memory card. Pin no. 2, however, is valid only when the startup mode is the memory card formatting mode (FMT indicator will be lit).



Any card is formatted into a single partition and is given volume label “C200HW.”

SRAM and ATA cards are formatted in FAT format, and FLASH cards are formatted in MS-FLASH format.

5-2-4 Memory Card Compatibility

Memory Cards used for the PC Card Unit are not compatible with the Memory Cards used for the CV-series CPUs. Memory Cards used for the CV-series Personal Computer Unit (CV500-VP2□□), or the CV-series ISA Control Unit (CV500-ISP0□) however, are compatible.

5-3 File Operations: CMCR Instruction

This section explains how to use the CARD MACRO expansion instruction (CMCR).

The CMCR instruction can be used when the C200HX/HG/HE SEND/RECV Enable Flag is ON.

5-3-1 Summary of CMCR

Process	Name	Function
Process No. 1	File Write	Writes the contents of C200HX/HG/HE memory to a file on a memory card in the PC Card Unit.
Process No. 2	File Read	Reads to C200HX/HG/HE memory the contents of a file in a memory card in the PC Card Unit.
Process No. 3	File and Memory Compare	Compares to C200HX/HG/HE memory the contents of a file in on a memory card in the PC Card Unit.
Process No. 4	File Search	Searches data in the files on a memory card in the PC Card Unit.

5-3-2 CMCR Format

The CMCR format is as follows:

CMCR	
Control	**C:** Beginning control data word (rightmost)
Source	**S**: Beginning command data word (rightmost)
Destination	**D:** Beginning response word

Control Data

Control data is written in the following format, from the beginning control data word (C).

Word	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C+0	Level (0/1)				*	Port No.			Process No.							
C+1	Control code															
C+2	Control code															
C+3	Control code															
C+4	Control code															
C+5	Control code															
C+6	Control code															
C+7	Control code															

*Control code enable

- **(Operating) Level**
  Setting to “0” specifies level #0; setting to “1” specifies level #1.
- **Control Code Enable**
  When bit 11 of word C+0 is ON (i.e., set to “1”), the control codes in words C+1 through C+7 are transmitted to the PC Card Unit.
- **Port Number**
  The port number varies depending on the process number. For explanations of the various processes, refer to *5-3-6 Using CMCR Processes*.
- **Process Number**
  The process number specifies a CMCR process (from 1 to 4).
- **Control Code**
  The control code specifies the filename for writing, reading, comparison, or searching.

Command Data

In word S+0, specify the command data length +1. The maximum command data length is 1,000 words.

S+0	Command data length (Number of words: BCD) $1 to $1001
S+1	Command data
S+2	Command data
	• • •
S+1000	Command data

Response Data

The response data varies depending on the CMCR process. For explanations of the various processes, refer to *Section 5-3-6 Using CMCR Processes*.

5-3-3 Files Created Using the CMCR Instruction

- **Maximum Number of Files per Card**
  The entire card may be used, but this depends on the FAT size of the card.
- **Maximum Size for One File**
  The entire card may be used, but the offset setting is up to 65,535. A file size of up to 4,294,967,295 bytes (2^{16} – 1) can be read.

5-3-4 Related SR Words and Bits

The following table shows the SR words and bits that are related to CMCR execution.

Word	Bit no.	Function
SR 237	00 to 07	Response code for operating level #0 after CMCR execution.
	08 to 15	Response code for operating level #1 after CMCR execution.
SR 252	00	Turns ON when there is an error after CMCR execution for operating level #0.
	01	ON when CMCR can be executed for operating level #0.
	03	Turns ON when there is an error after CMCR execution for operating level #1.
	04	ON when CMCR can be executed for operating level #1.

5-3-5 Response Codes

Code	Meaning
00	Normal completion
01	Parameter error (offset, file size, number of words read, etc.)
02	Disk full, file I/O error, file type error
03	File does not exist.
04	Comparison, search failure
05 to FE	Undefined
FF	Process number error

Note CMCR Errors
When the C200HX/HG/HE ER Flag is ON.
Either the set parameter range has been exceeded or the instruction was executed when the SEND/RECV Enable Flag was OFF.
When the PC Card Unit displays an error response.
The relevant bit on word 252 is ON. Refer to *5-3-4 Related SR Words and Bits.*

5-3-6 Using CMCR Processes

The following four subsections explain how to use the four CMCR processes. The processes are explained one by one, in order of process number.

5-3-7 File Write (Process No. 1)

The File Write process writes the contents of C200HX/HG/HE memory to a file on a memory card in the PC Card Unit.

Control Data

Word	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C+0		Level (0/1)				1	Port no.			0	0	0	0	0	0	0	1
C+1		Control code															
to		• • •															
C+7		Control code															

- **(Operating) Level**
  Setting to “0” specifies level #0; setting to “1” specifies level #1.
- **Port Number**

Bit 8 0: Overwrites existing file.

1: Creates new file.

When “offset” (explained below) is set to “1,” the new data is added to the end of the existing file.

When “offset” is set to “0,” a new file with the same name as an existing file overwrites the existing file. At this time, if “data length” (explained below) is set to “1” (no transmission data), the existing file is deleted.

Bit 9 0: Makes the created file a comma separated value file. Carriage returns are not inserted. The separation method is determined by bit 10. The file is created as one line with no carriage returns (CR + LF).

1: Makes the created file a binary file, and saves the memory contents just as it is.

Bit 10 0: Puts a comma between the data for each word.

1: Puts a comma between the data for every two words (double-length words).

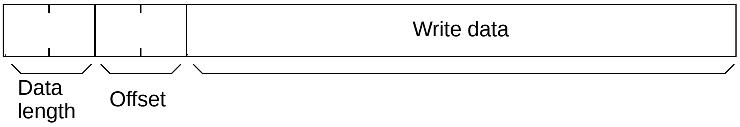
- **Control Code**

  Write the hexadecimal values for the drive and filename for the file to be written. The extension must be written in three characters. Shift JIS 1-byte alphanumeric code is used.

  If there are already directories on the card, a directory can be specified.

  Slot 1 is drive G, and slot 2 is drive H.

Command Data



- **Data Length**

  The data length (including the data length and offset designations) is specified by the number of words. Specify a number from 1 to 1001 in BCD.

- **Offset**

  The transmission data offset is specified by the number of elements (i.e., the number of words for binary and single-word comma separation, or the number of double-length words double-word comma separation). Specify a number from 0 to FFFF in hexadecimal.

  When adding data to a file, all numbers other than 0 have the same meaning (i.e., add to the end of the file).

Single-word comma separation, binary:	Specified number of words.
Double-word comma separation:	Specified number of elements.

- **Write Data**

  The data to be written to the file. The maximum amount of data depends on the type of file created, as follows:

Single-word comma separation, binary:	999 words
Double-word comma separation:	998 words (449 elements)

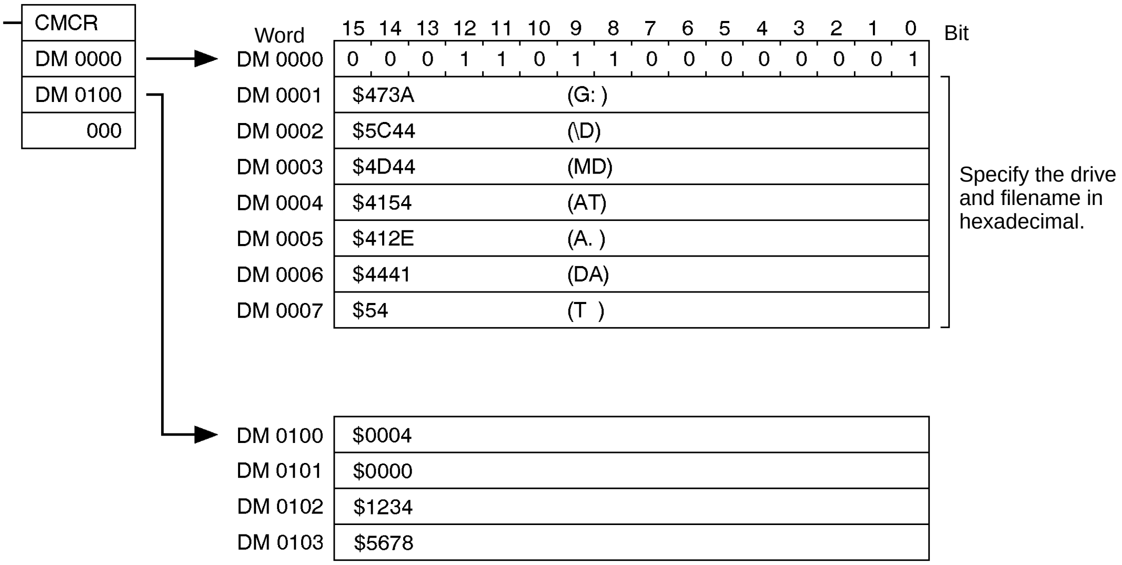
Response Data There is no response data.

Response Codes

00:	Normal completion.
01:	Offset or file size error (e.g., offset + size exceeds file size).
02:	Disk full, file output error, or file type error.
03:	File does not exist.

File Writing Example

In the following example, two words of data, from DM 0102 and DM 0103, are written as a new file to the operating level #1 PC Card Unit's memory card. The drive and filename are G:\DMDATA.DAT.



CMCR
DM 0000
DM 0100
000

Word	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DM 0000	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	1

Word		
DM 0001	$473A	(G:)
DM 0002	$5C44	(\D)
DM 0003	$4D44	(MD)
DM 0004	$4154	(AT)
DM 0005	$412E	(A.)
DM 0006	$4441	(DA)
DM 0007	$54	(T)

Specify the drive and filename in hexadecimal.

Word	
DM 0100	$0004
DM 0101	$0000
DM 0102	$1234
DM 0103	$5678

Settings and Results

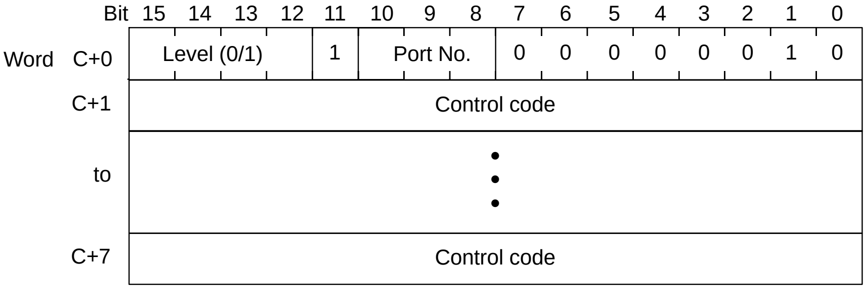
The following table shows examples of settings for bits 8 to 10 and the command data and file writing results.

Function	Bit			Command data	Contents of file
	10	9	8		
New (Binary)	*	1	1	00 04 00 00 12 34 56 78	12 34 56 78 (4 bytes, binary file)
New (CSV word)	0	0	1	00 06 00 00 12 34 56 78 9a bc de f0	1234,5678,9abc,def0 (19 bytes, ASCII text file)
New (CSV double-length word)	1	0	1	00 06 00 00 12 34 56 78 9a bc de f0	12345678,9abcdef0 (17 bytes, ASCII text file)
Delete	*	*	1	00 01	–
Add (CSV word)	0	0	1	00 04 00 01 12 34 56 78	1234,5678 → 1234,5678,1234,5678 (19 bytes, ASCII text file)
Overwrite (CSV word)	0	0	0	00 03 00 01 12 34	1234,5678 → 1234,1234 (9 bytes, ASCII text file)
Overwrite (CSV double-length word)	1	0	0	00 04 00 01 12 34 56 78	12345678,9abcdef0 → 12345678,12345678 (17 bytes, ASCII text file)

5-3-8 File Read (Process No. 2)

The File Read process reads files on a memory card in the PC Card Unit to C200HX/HG/HE memory.

Control Data



Word	Bit 15–12	11	10–8	7	6	5	4	3	2	1	0
C+0	Level (0/1)	1	Port No.	0	0	0	0	0	0	1	0
C+1	Control code										
to	• • •										
C+7	Control code										

- **(Operating) Level**
  Setting to "0" specifies level #0; setting to "1" specifies level #1.

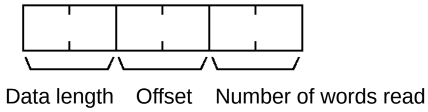
- **Control Code Enable**
  Fixed at "1" (control codes enabled).
- **Port Number**

Bit 8	0:	Reads the specified file.
	1:	Reads the specified number of elements in the file.
Bit 9	0:	Reads data as comma separated values. The separation method is determined by bit 10.
	1:	Reads as binary file.
Bit 10	0:	Uses comma separation in single word units (one word).
	1:	Uses comma separation in double word units (two words).

- **Control Code**
  Write in hexadecimal value the drive and filename for the file to be read. The extension must be written in three characters. Shift JIS 1-byte alphanumeric code is used.

  Slot 1 is drive G, and slot 2 is drive H.

Command Data



- **Data Length**
  Fixed at 0003.
- **Offset**
  The beginning read offset is specified by word or number of elements (double-length words). Specify a number from 0 to FFFF in hexadecimal.

Single-word comma separation, binary:	Specified by word.
Double-word comma separation:	Specified by no. of elements.

- **Number of Words Read**
  The number of words read is specified by word or number of elements.

Single-word comma separation, binary:	Specified by word.
Double-word comma separation:	Specified by no. of elements.

Make the setting from 1 to 3E7 in hexadecimal. The number of words read depends on the type of file, as follows:

Single-word comma separation, binary:	999 words ($3E7)
Double-word comma separation:	449 (998 words) ($1F3)

Note Comma-separated files are read as text files with one-byte separations, such as commas, every 4 or 8 bytes. If 2-byte separations, such as carriage returns (CR + LF), are included, an error will be generated with a response code of 2. Convert to 1-byte characters beforehand.

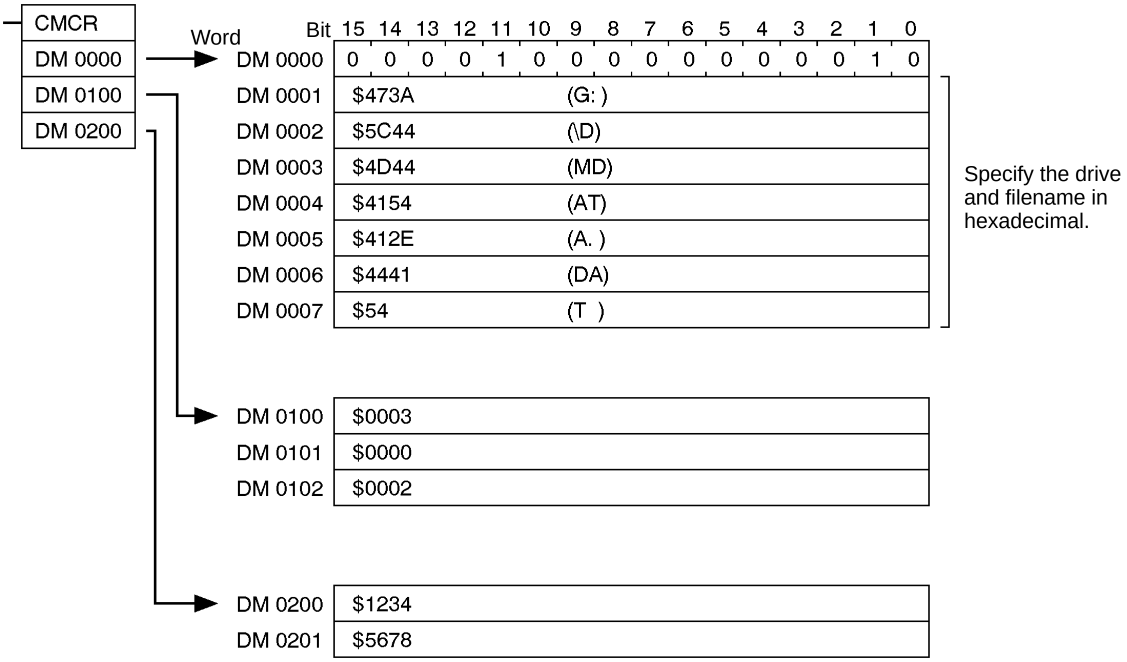
Response Data The data read from the specified file is stored. When reading the number of elements, the number of elements in the file is stored in 2-word hexadecimal.

Response Codes

00:	Normal completion.
01:	Error in offset value or number of words read.
02:	File I/O error or file type error.
03:	File does not exist.

File Reading Example

In the following example, a file called “G:\DMDATA.DAT” in the operating level #0 PC Card Unit is read in comma separated word units. The offset value is 0000 and the number of words read is “2.” The data is read to DM 0200 onwards.



CMCR
DM 0000
DM 0100
DM 0200

Word	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DM 0000	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0

Word		
DM 0001	$473A	(G:)
DM 0002	$5C44	(\D)
DM 0003	$4D44	(MD)
DM 0004	$4154	(AT)
DM 0005	$412E	(A.)
DM 0006	$4441	(DA)
DM 0007	$54	(T)

Specify the drive and filename in hexadecimal.

Word	
DM 0100	$0003
DM 0101	$0000
DM 0102	$0002

Word	
DM 0200	$1234
DM 0201	$5678

Settings and Results

The following table shows examples of settings for bits 8 to 10 and the command data and file reading results.

Function	Bit			Command data	Contents of PC card file	D: Response data storage words
	10	9	8			
Size (Binary)	*	1	1	*	12 34 56 78 9a bc de f0	0000 0004
Size (CSV word)	0	0	1	*	1234,5678,9abc,def0	0000 0004
Size (CSV double-length word)	1	0	1	*	12345678,9abcdef0	0000 0002
Read (Binary)	*	1	0	00 03 00 00 00 02	12 34 56 78 9a bc de f0	12 34 56 78
Read (CSV word)	0	0	0	00 03 00 00 00 02	1234,5678,9abc,def0	12 34 56 78
Read (CSV double-length word)	1	0	0	00 03 00 00 00 02	12345678,9abcdef0	12 34 56 78 9a bc de f0

5-3-9 File and Memory Compare (Process No. 3)

The File and Memory Compare process compares a file on a memory card in the PC Card Unit to C200HX/HG/HE memory. The results of the comparison are indicated by the response code.

If the specified length of data does not perfectly match, or if the file size is smaller than the specified length of data, the response code (04) shows that there is no match.

Control Data

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word C+0	Level (0/1)				1	Port No.			0	0	0	0	0	0	1	1
C+1	Control code															
to	• • •															
C+7	Control code															

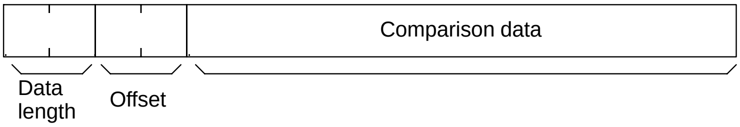
- **(Operating) Level**
  Setting to “0” specifies level #0; setting to “1” specifies level #1.
- **Control Code Enable**
  Fixed at “1” (control codes enabled).
- **Port Number**

Bit 9	0:	Compares data as comma separated values. The separation method is determined by bit 10.
	1:	Compares as binary file.
Bit 10	0:	Puts comma separation in single word units (one word).
	1:	Puts comma separation in double word units (two words).

- **Control Code**
  Input in hexadecimal values the drive and filename for the file to be compared. The extension must be written in three characters. Shift JIS 1-byte alphanumeric code is used.

  Slot 1 is drive G, and slot 2 is drive H.

Command Data



- **Data Length**
  The data length (including the data length and offset designations) is specified by the number of words. Specify a number from 1 to 1001 in BCD.
- **Offset**
  The offset specifies the comparison data offset. For binary files it is specified by the number of words, and for comma separated files it is specified by the number of elements. Specify a number from 0 to FFFF in hexadecimal.
- **Comparison Data**
  The maximum value for comparison data depends on the type of file created, as follows:

Single-word comma separation, binary:	999 words (data length: 1001)
Double-word comma separation:	998 words (data length: 1000)

Note Comma-separated files are read as text files with one-byte separations, such as commas, every 4 or 8 bytes. If 2-byte separations, such as carriage returns (CR + LF), are included, an error will be generated with a response code of 2. Convert to 1-byte characters beforehand.

Response Data

No response data. Comparison results are indicated by response codes.

Response Codes

00: Normal completion.

01: Error in offset value or size.

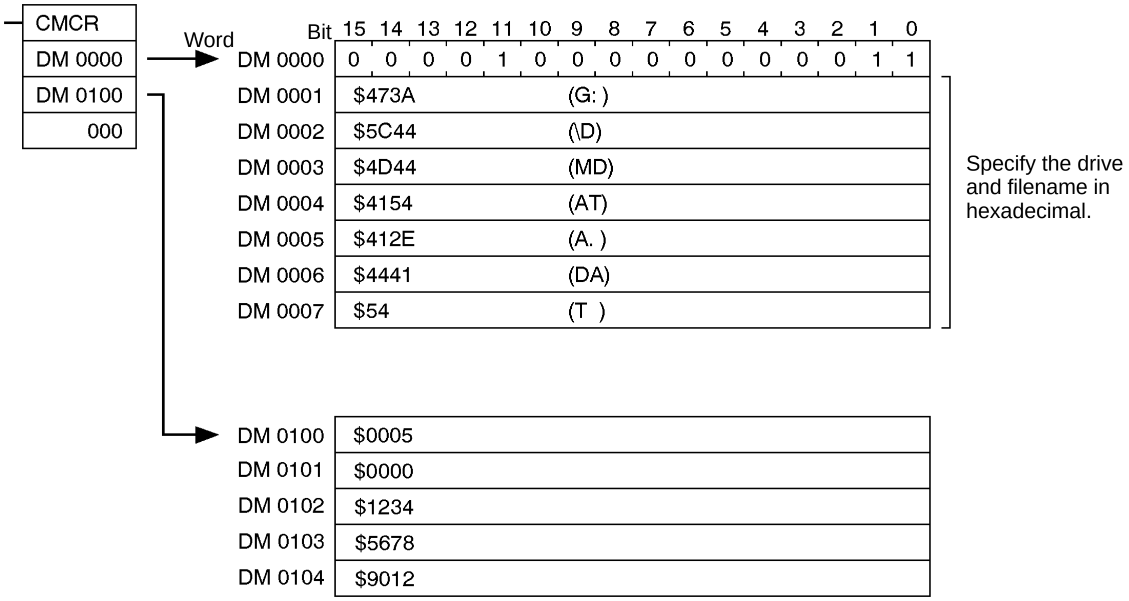
02: File input error or file type error.

03: File does not exist.

04: Data does not match.

File Reading Example

In the following example, a file called “G:\DMDATA.DAT” at the operating level #0 PC Card Unit is compared to C200HX/HG/HE memory as binary data. The comparison data length is three words, and the offset value is 0000 (beginning of file).



CMCR
DM 0000
DM 0100
000

Word	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DM 0000	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1

Word		
DM 0001	$473A	(G:)
DM 0002	$5C44	(\D)
DM 0003	$4D44	(MD)
DM 0004	$4154	(AT)
DM 0005	$412E	(A.)
DM 0006	$4441	(DA)
DM 0007	$54	(T)

Specify the drive and filename in hexadecimal.

Word	
DM 0100	$0005
DM 0101	$0000
DM 0102	$1234
DM 0103	$5678
DM 0104	$9012

Settings and Results

The following table shows examples of settings for bits 9 and 10 and the command data and comparison results.

Function	Bit		Command data	Contents of PC card file	Response code
	10	9			
Binary comparison	*	1	00 04 00 00 12 34 56 78	12 34 56 78	00 (Match)
	*	1	00 03 00 01 56 78	12 34 56 78	00 (Match)
	*	1	00 03 00 00 56 78	12 34 56 78	04 (No match)
CSV word comparison	0	0	00 06 00 00 12 34 56 78 9a bc de f0	1234,5678,9abc,def0	00 (Match)
	0	0	00 04 00 02 9a bc de f0	1234,5678,9abc,def0	00 (Match)
	0	0	00 04 00 00 9a bc de f0	1234,5678,9abc,def0	04 (No match)
CSV double-length word comparison	1	0	00 06 00 00 12 34 56 78 9a bc de f0	12345678,9abcdef0	00 (Match)
	1	0	00 04 00 01 9a bc de f0	12345678,9abcdef0	00 (Match)
	1	0	00 04 00 00 9a bc de f0	12345678,9abcdef0	04 (No match)

5-3-10 File Search (Process No. 4)

The File Search process searches files, by specified data, on a memory card in the PC Card Unit.

Control Data

Word	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C+0	Level (0/1)				1	Port No.			0	0	0	0	0	1	0	0
	C+1	Control code															
	to	• • •															
	C+7	Control code															

- **(Operating) Level**
  Setting to “0” specifies level #0; setting to “1” specifies level #1.
- **Control Code Enable**
  Fixed at “1” (control codes enabled).
- **Port Number**

  Bit 9 0: Searches as comma separated values. The separation method is determined by bit 10.

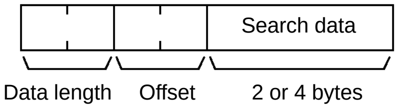
  1: Searches as binary file.

  Bit 10 0: Puts comma separation in single word units (one word).

  1: Puts comma separation in double word units (two words).
- **Control Code**
  Input in hexadecimal values the drive and filename for the file to be searched. The extension must be written in three characters. Shift JIS 1-byte alphanumeric code is used.

  Slot 1 is drive G, and slot 2 is drive H.

Command Data



- **Data Length**
  For word comparison, specify “3” and for double-length word comparison specify “4.” For binary, specify “3.”
- **Offset**
  The offset specifies the search data offset. For binary files it is specified by the number of words, and for comma separated files it is specified by the number of elements. Specify a number from 0 to FFFF in hexadecimal.
- **Search Data**
  The length of the search data depends on the type of file created, as follows:

  Single-word comma separation, binary: One word

  Double-word comma separation: Two words

Note Comma-separated files are read as text files with one-byte separations, such as commas, every 4 or 8 bytes. If 2-byte separations, such as carriage returns (CR + LF), are included, an error will be generated with a response code of 2. Convert to 1-byte characters beforehand.

Response Data

When the search data is found, the location of the matching data is stored. It is returned as data within a range of 0 to FFFF.

The word location of matching data is as indicated as being a certain number of words from the specified offset, i.e., the first word after the offset becomes 0000; the second word, 0001; the third word, 0002, etc.

Response Codes

00: Normal completion.

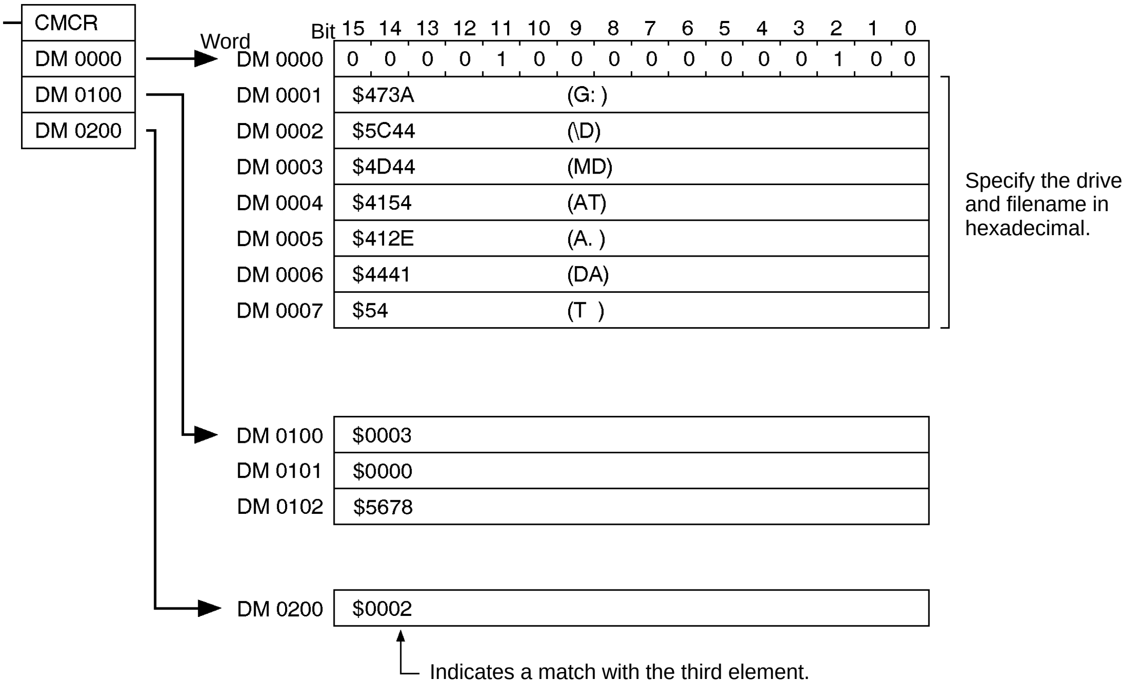
01: Parameter error (for example, if the data length is 3 for double word comparison).

02: File input error or file type error.

03: File does not exist.

File Reading Example

In the following example, the contents (single-word comma separation) of a file called “G:\DMDATA.DAT” at the operating level #0 PC Card Unit is searched for the specified data in C200HX/HG/HE memory. The offset is 0000 and the search data is $5678. The contents of DMDATA.DAT is 0000, 1234, 5678, 9123, 4560, 0000, 1111.



CMCR
DM 0000
DM 0100
DM 0200

Word	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DM 0000	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0

Word		
DM 0001	$473A	(G:)
DM 0002	$5C44	(\D)
DM 0003	$4D44	(MD)
DM 0004	$4154	(AT)
DM 0005	$412E	(A.)
DM 0006	$4441	(DA)
DM 0007	$54	(T)

Specify the drive and filename in hexadecimal.

Word	
DM 0100	$0003
DM 0101	$0000
DM 0102	$5678

Word	
DM 0200	$0002

Indicates a match with the third element.

Settings and Results

The following table shows examples of settings for bits 9 and 10 and the command data and search results.

Function	Bit		Command data	Contents of file	Response data (location of match)
	10	9			
Binary search	*	1	00 03 00 01 9a bc	12 34 56 78 9a bc de f0	00 01
CSV word search	0	0	00 03 00 01 9a bc	1234,5678,9abc,def0	00 01
CSV double-length word search	1	0	00 04 00 01 9a bc de f0	12345678,9abcdef0	00 00

5-3-11 File Precautions

- When comma-separated files are written by CMCR, they are written as one line.
- Comma-separated files created at a personal computer are read with the assumption that they were created in one of the following formats.

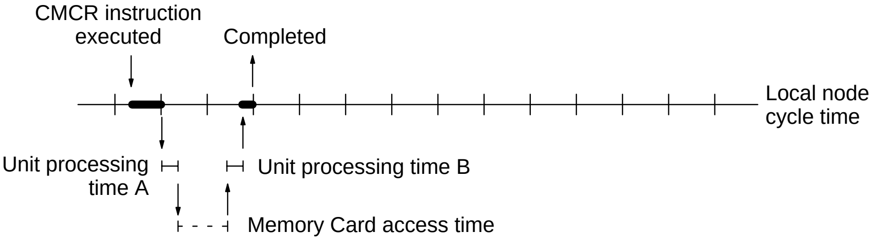
  4 bytes, comma, 4 bytes, comma, 4 bytes, comma....

  8 bytes, comma, 8 bytes, comma, 8 bytes, comma....
- Additional data can only be written to a file if there is space available on the card, and file data can only be overwritten in the range where an offset can be specified.
- Individual fields in comma separated files are not enclosed by quotation marks.

5-4 Memory Card Access Times for CMCR Instructions

The minimum times for executing CMCR instructions are as follows:

Minimum CMCR Instruction Execution Times



The access time will differ depending on the type of Memory Card used, so in this instance it is calculated as 0.

Minimum CMCR Instruction Execution Time
= local node cycle time + Unit processing time A + Memory Card access time
+ Unit processing time B + local node cycle time.

- **Unit Processing Time A**
  Time to write data on the card: no. of write words × 0.013 + 5 ms
  Time to read data from a card: 5 ms
- **Unit Processing Time B**
  Time to write data on the card: 5 ms
  Time to read data from a card: no. of read words × 0.013 + 5 ms

Example:The minimum instruction execution time for writing 999 words

Local node cycle time × 2 + 999 × 0.013 + 5 ms + 5ms
= local node cycle time × 2 + 22.987 ms

Actual Measurement Values

The write time for a 999-word SRAM card:	24 ms
The write time for a 1-word SRAM card:	9.7 ms

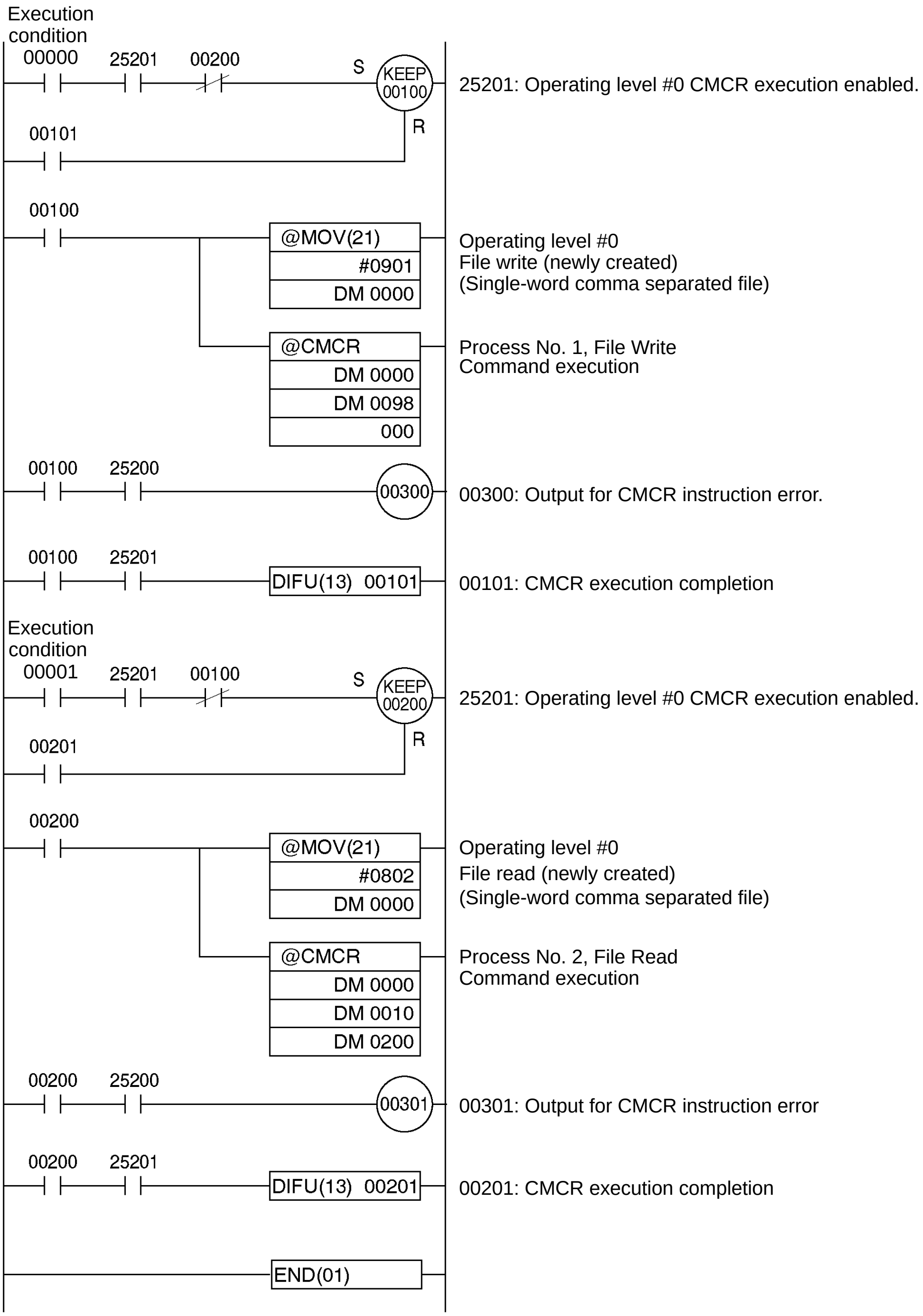
C200HX/HG/HE Cycle Delay Time

When a PC Card Unit is connected to the C200HX/HG/HE CPU, the cycle time will be delayed by a maximum of 6 ms.

5-5 Sample Program

This section provides a sample program for writing and reading files. In this program, when execution condition IR 00000 is turned from OFF to ON and then OFF again, the data in 100 DM words beginning with DM 0100 is written to G:\DMSAVE.DAT. When execution condition IR 00001 is turned from OFF to ON and then OFF again, the 100 words of data is read from DMSAVE.DAT to the DM words from DM 0200 onwards.

Program



25201: Operating level #0 CMCR execution enabled.

Operating level #0
File write (newly created)
(Single-word comma separated file)

Process No. 1, File Write
Command execution

00300: Output for CMCR instruction error.

00101: CMCR execution completion

25201: Operating level #0 CMCR execution enabled.

Operating level #0
File read (newly created)
(Single-word comma separated file)

Process No. 2, File Read
Command execution

00301: Output for CMCR instruction error

00201: CMCR execution completion

DM Area Settings

DM0000	Control data		
DM0001	47	3A	‘G’, ‘:’
DM0002	5C	44	‘\’, ‘D’
DM0003	4D	53	‘M’, ‘S’
DM0004	41	56	‘A’, ‘V’
DM0005	45	2E	‘E’, ‘.’
DM0006	44	41	‘D’, ‘A’
DM0007	54		‘T’

DM0010	0003	Fixed at 0003.
DM0011	0000	Offset: 0
DM0012	0064	Read data length: 100 words

DM0098	0102	Write data length: 102 words
DM0099	0000	Offset: 0
DM0100	Data	
	:	
DM0199	Data	

5-6 Debugging Ladder Programs

This section shows how to debug the ladder program, using the sample program provided in *5-5 Sample Program* as an example.

Procedure

The procedure for debugging a ladder program is outlined below. Each of these steps is then explained in more detail.

1, 2, 3...

1. Start up the Support Software.
2. Allocate a function code to CMCR.
3. Create the ladder program.
4. Go online.
5. Transfer the program to the PC.
6. Prepare the system for debugging.
7. Execute the program.
8. Debug the program.
9. Check the write data.

Support Software Startup

From the drive in which the SYSMAC Support Software is installed, enter “SSS” and then press the Enter key to start up the SYSMAC Support Software. Then, from the System Setup menu, set the PC model, PC communications parameters, and so on.

Expansion Instruction Setup

Select “X:Set instructions” from the Utility Menu. Then use “E:Edit instructions” to set a function code for the CMCR instruction in the instructions table. For details, refer to the *SYSMAC Support Software Operation Manual: C-series PCs*.

The settings can also be made using the Programming Console. For details, refer to *4-1-3 Allocating a Function Code for CMCR*.

Ladder Program Creation

Create the sample program.

Online Connection

Go online by pressing Ctrl+O followed by the F1 key (PC connection).

Program Transfer

Use “Monitoring/P:Transfer program/W:Computer → PC” to transfer the program to the C200HX/HG/HE Programmable Controller.

Debug Preparation

Insert the memory card into slot 1 in the PC Card Unit. Set the control data, data length, offset, and write data to the DM area using the DM Menu.

Program Execution

Press Ctrl+O followed by the F3 key (MONITOR) to switch to MONITOR mode.

Debugging

Use the "Monitor data" operation under the Monitoring Menu to execute the write command by turning execution condition IR 00000 from OFF to ON and back OFF.

Check to see that error output IR 00300 does not turn ON. If it does turn ON, use CMCR response codes 23700 to 23707 to check the error.

Execute the read command by turning execution condition IR 00001 from OFF to ON and back OFF.

Just as for the write command, check to see that error output IR 00301 does not turn ON, and check the response codes if it does.

Use the DM Menu to check to see that the contents of DM 0100 to DM 0199 have been copied to DM 0200 to DM 0299.

Checking the Write Data

Eject the memory card from the PC Card Unit.

Insert the memory card in a personal computer with a 3G8F5-PCM01 PC Card Interface Board installed, or in a computer with a PC card slot.

Check to see that a file called "DMSAVE.DAT" has been created on the memory card. This is a text data file, in single-word comma separated value format.

This completes the ladder program debugging operation.

Part 3
Using Ethernet

This part of the manual describes how to install Ethernet Cards and use the setup software. It includes directions on how to use the SEND and RECV instructions to transfer data and provides details on the FINS commands and responses used for communications between the C200HX/HG/HE CPU and the PC Card Unit.

SECTION 6
Preparations for Operation

This section outlines the steps required to install the PC Card Unit and Ethernet function and prepare for operation. Be sure you have read this section and understood all of the procedures before attempting to actually make the settings or do any programming.

6-1 Outline

The procedures required to prepare for operation are outlined below. Be sure to familiarize yourself with these basic procedures.

These procedures are explained in more detail in *6-2 Procedures.*

1, 2, 3...

1. **Communications Board Installation**

   Mount the Communications Board in the C200HX/HG/HE CPU.
2. **C200HX/HG/HE Setup**

   Assign a function code to the CARD MACRO (CMCR) expansion instruction.
3. **PC Card Unit Setup**

   Mount the PC Card Unit and make the required settings.
4. **Ethernet Setup**

   Install the Ethernet Card and set up Ethernet using the setup software so that the communications software can be used.
5. **Programming**

   Write the program.
6. **Debugging**

   Debug the program.
7. **Operation**

6-2 Procedures

This section outlines the procedures required to install and set up the PC Card Unit and peripheral devices for application development and operation. Be sure that you thoroughly understand all of these procedures. References for further reading are provided for each procedure.

1. Communication Board

Mount either the C200HW-COM01 or C200HW-COM04-E Communications Board in the C200HX/HG/HE CPU. Refer to the *C200HW-COM01 to 06-E Communications Board Operation Manual* for details.

2. C200HX/HG/HE Setup

Two steps are require to prepare the C200HX/HG/H for PC Card Unit application.

- **Addition of CMCR** 4-1

  Use either the SYSMAC Support Software or the Programming Console to assign a function code to the CARD MACRO (CMCR) instruction.
- **System Switch Setting** 4-1

  Turn ON DIP switch pin number 4 to enable setting expansion instructions for the C200HX/HG/HE.

3. PC Card Unit Setup

Two steps are required to set up the PC Card Unit.

- **System Switch Setting** 4-3

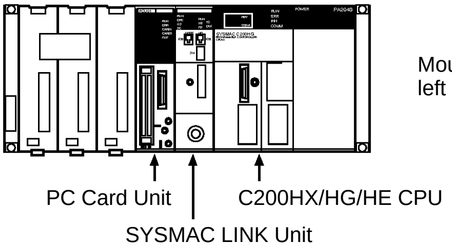
  Set the startup mode, memory card initialization type, the operating level, etc. If only the memory card function is to be used, set only the operating level and leave the other pins set to OFF.
- **Unit and Peripheral Connections** 4-4 to 4-6

  Mount the PC Card Unit on the Backplane and install the Bus Connection Unit and memory card.

Unit Restrictions and Mounting Locations

When a PC Card Unit is used, only one SYSMAC LINK Unit or SYSMAC NET Link Unit can be used on the CPU Backplane.

When a PC Card Unit and SYSMAC LINK Unit or SYSMAC NET Link Unit are used together on the CPU Backplane, they must be mounted in the slots shown in the following illustration.



4. Ethernet Setup

- **Ethernet Card Installation** 7-2

Install the Ethernet Card into the PC Card Unit.

- **Ethernet Connection** 7-3

Connect the PC Card Unit to the Ethernet network.

- **Personal Computer Setup** 7-4

Connect the PC Card Unit to a personal computer and copy the setup software.

- **Setup Software Settings** 7-5 and 7-6

Activate the setup software on the personal computer and carry out all settings for Ethernet.

5. Programming 8 to 11

Use SYSMAC Support Software with a personal computer to create the ladder program. For details regarding ladder programming, refer to the SYSMAC Support Software Operation Manuals.

6. Program Debugging Use a memory card with the program that has been created and correct any bugs that may be found.

7. Operation Proceed with actual operation.

Note

1. Do not use the setup software for the C200HW-PCS01(-EV1) to perform settings for the C200HW-PCS01-V2.
2. The C200HW-PCS01-V2 is factory-set to ODI driver (cntpclat.com). For this reason, if the Unit is started up without installing the C-NET(PC)C-10L Ethernet Card made by CONTEC, the ERR indicator will light. Perform the required settings for the ODI driver and Ethernet communications according to the Ethernet Card used.

6-3 Communications

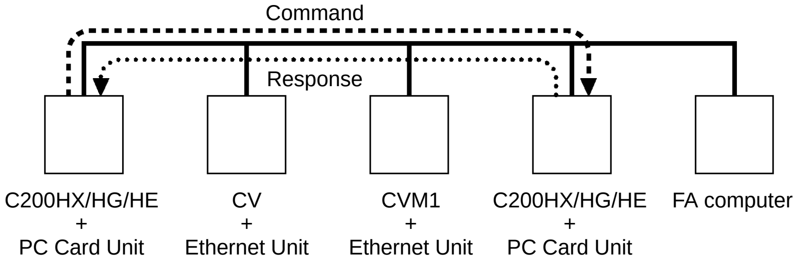
6-3-1 Outline

The PC Card Unit has a built in message service that enables the use of data communications and commands to control PCs and FA computers on nodes on a network.

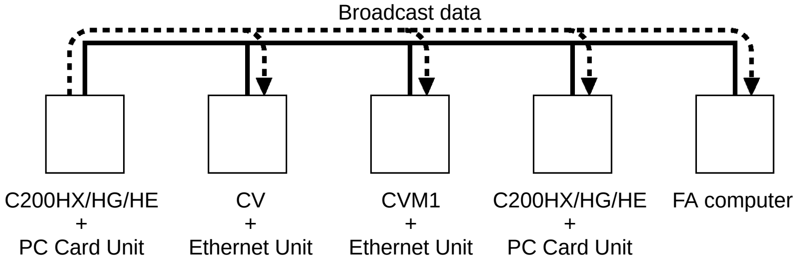
The message service operates by setting the communications destination each time it sends a command or data and then receives the response from the destination node. The message service can be set so as not to send any response if a response is not necessary. A message can be sent to all nodes on the network simultaneously. This is called broadcasting.

The PC Card Unit can participate in communications with C200HX/HG/HE CPUs, CV-series PCs, and computers on the Ethernet network.

Node-to-node Data Transfer



Broadcasting

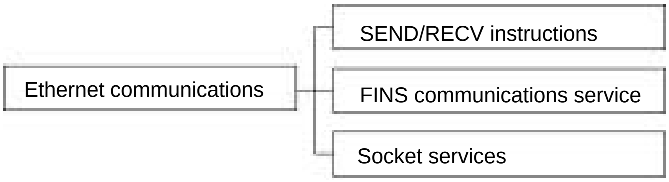


There are three communications systems used by the message service for the PC Card Unit: ladder diagram instructions (SEND and RECV), FINS commands, and socket services. These are detailed next.

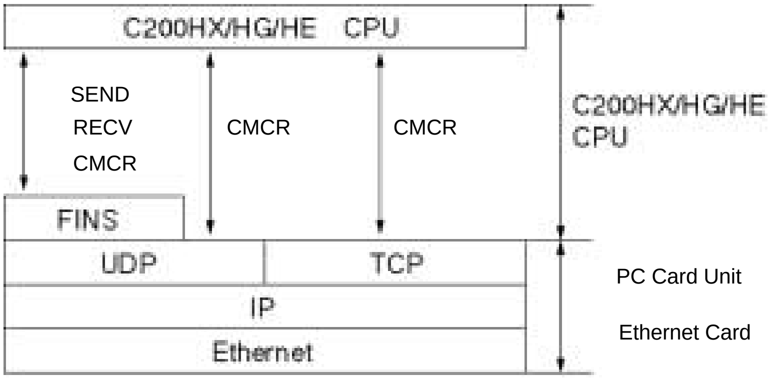
6-3-2 Ethernet Communications Function

The C200HW-PCS01-V2 PC Card Unit Ethernet Set supports the three communications methods shown in the following diagram.

Overview



Software Configuration



Note FINS commands are issued by using CMCR.

The three communication methods are outlined below.

6-3-3 SEND and RECV

Data is transmitted from the PC Card Unit using the SEND and RECV instructions. Although it is necessary to specify the data for transmission, detailed data processing is automatically performed by the Unit.

Refer to *Section 8 Using SEND(90) and RECV(98)* for details on SEND and RECV.

6-3-4 FINS Communications Service (FINS Commands and Responses)

The FINS communication service is explained below in three categories: when the PC Card Unit receives a FINS command; when a FINS command is sent from the PC Card Unit; and when FINS communication is carried out between the computer and the PC Card Unit.

For details on FINS commands and responses, refer to *Section 9 FINS Commands* and *Section 11 Using FINS Commands and Responses*.

When the PC Card Unit Receives a FINS Command

When the PC Card Unit receives a FINS command, it automatically interprets the command and sends a response. All processing is conducted automatically, so there is no need to program the PC to respond to FINS commands.

When a FINS Command is Sent From the PC Card Unit

A FINS command is issued using the CMCR instruction, and a response is returned from the recipient of the command. That response is returned automatically if the recipient of the FINS command is a C200HX/HG/HE or CVM1/CV Programmable Controller.

FINS Communication Between the Computer and the PC Card Unit

To use FINS commands from a computer, create the command data in the program at the computer according to the FINS command format. Have the FINS responses that are returned form the PC Card Unit analyzed and processed according to that format. For details, refer to *Section 9 FINS Commands*.

6-3-5 Socket Services

A socket is an interface which allows a user program to directly use TCP (Transmission Control Protocol) and UDP (User Datagram Program). Socket services are employed by using the CMCR instruction. Using socket services allows the PC Card Unit to communicate with UNIX workstations and FA Computers other than OMRON Programmable Controllers, with any protocol. For details, refer to *Section 10 Socket Services*.

SECTION 7
Setting Up Ethernet

This section describes how to install Ethernet cards and use the setup software. Before setting up Ethernet, always set up the PC Card Unit as described in *Section 4 Installation and Switch Settings*.

7-1 Preparations

Before setting up the Ethernet, always perform the setup procedure described below.

Note

1. Do not use the setup software for the C200HW-PCS01(-EV1) to perform settings for the C200HW-PCS01-V2.
2. The C200HW-PCS01-V2 is factory-set to ODI driver (cntpclat.com). For this reason, if the Unit is started up without installing the C-NET(PC)C-10L Ethernet Card made by CONTEC, the ERR indicator will light. Perform the required settings for the ODI driver and Ethernet communications according to the Ethernet Card used.
3. The service conditions, slots used, and restrictions on PC cards are the same for the PC Card Unit Ethernet Set as for the PC Card Unit. Be sure to observe the restrictions described in *Section 4 Installation and Switch Settings*.

1, 2, 3...

1. **Mounting the Communications Board**

   Mount the C200HW-COM01 or C200HW-COM04-E Communications Board in the C200HX/HG/HE CPU. Refer to the *C200HW-COM01 to 06-E Communications Board Operation Manual* for details.

2. **Setting Up the C200HX/HG/HE**

   - **Adding CMCR** *4-1*

     Use the SYSMAC Support Software or Programming Console to allocate a function code to the CMCR instruction (CARD MACRO instruction).

   - **Setting the System Switch** *4-1*

     Turn ON pin no. 4 on the DIP switch to enable allocating function codes to expansion instructions for the C200HX/HG/HE.

3. **Setting Up the PC Card Unit**

   - **Setting the System Switch** *4-3*

     Set the startup mode, memory card format, operating level, etc. If only a memory card is to be used, set the operating level only. Other pins can be left OFF.

   - **Connecting the PC Card Unit to Peripherals** *4-4 to 4-6*

     Install the PC Card Unit on the CPU Backplane, and mount a Bus Connection Unit and memory card.

7-2 Installing and Removing the Ethernet Card

This section describes restrictions on Ethernet cards and explains how to install/remove an Ethernet card in/from the PC Card Unit.

7-2-1 Ethernet Cards

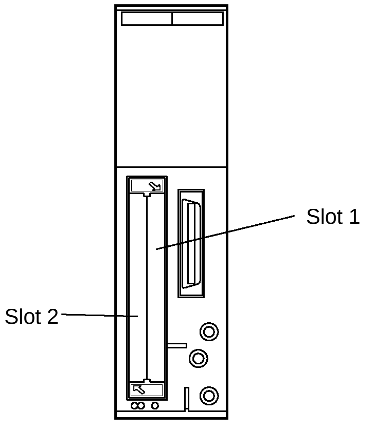
Applicable Ethernet Cards

The PC Card Unit uses a DOS ODI driver. Although two DOS ODI drivers are currently in existence (SPEC3 and SPEC4), the PC Card Unit supports the SPEC3 ODI driver only. Therefore, Ethernet cards that have only the SPEC4 ODI driver cannot be used.

Note

1. Before installing or removing an Ethernet card, always turn the PC Card Unit OFF. An Ethernet card cannot be installed or removed while the PC Card Unit is ON.
2. Only one Ethernet card can be used.

7-2-2 PC Card Slots



Slot 1: A type-I, -II, or -III PC card can be mounted in this slot.

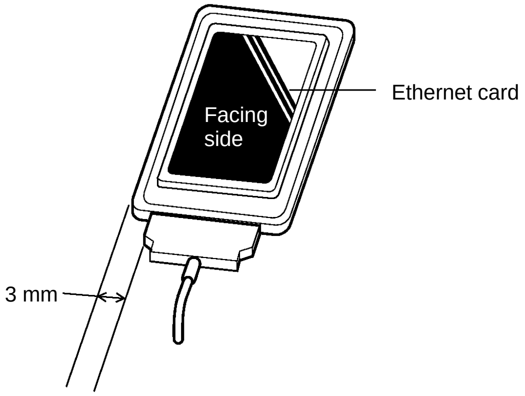
Slot 2: A type-I or -II PC Card can be mounted in this slot. If, however, a type-III PC Card is mounted in slot 1, slot 2 cannot be used.

Note The card inserted into slot 2 can be removed or installed by loosening the set screws on plate 2. When removing or installing a card into slot 1, always remove plate 2.

To avoid accidental removal of the Ethernet Card, it is recommended that the Ethernet Card be installed in slot 1.

7-2-3 Installing and Removing an Ethernet Card

Note Use an Ethernet card that has a connector more than 3 mm away from the edge of the card, as shown in the figure below. If the distance between the edge of the card and the connector is 3 mm or less, the plate for securing the PC card cannot be installed.

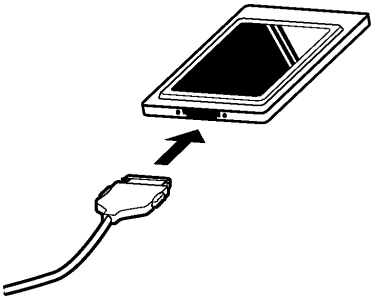


Note There is a restriction on the current consumption of the Ethernet card. Use an Ethernet card that meets the following conditions:
I_{5V} (1 slot) ≤ 0.5 A, I_{12V} (1 slot) ≤ 0.1 A
I_{5V} (2 slots) + 3.4 x I_{12V} (2 slots) ≤ 1.0 A

Installing an Ethernet Card

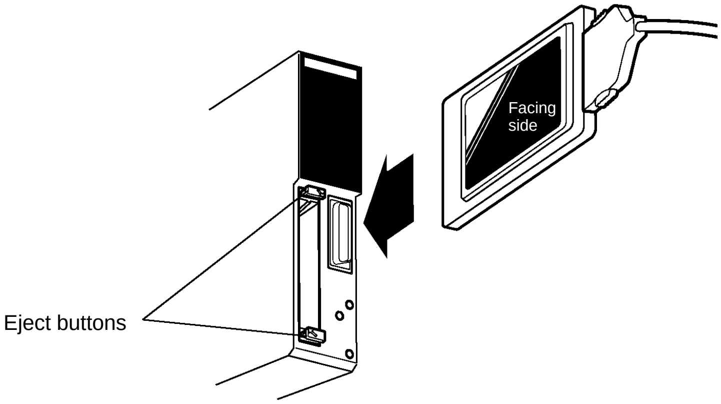
1, 2, 3...

1. Connect the attached adapter to the Ethernet card as shown in the figure below. Carefully check the shape of each connector to ensure correct connection. Be sure to press the connector all the way in.

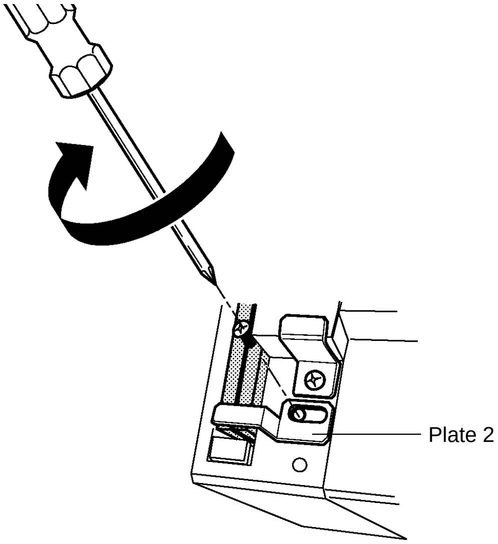


2. Insert the Ethernet card into the PC card slot as shown in the figure below. Press the Ethernet card until its edge is aligned with the eject buttons.

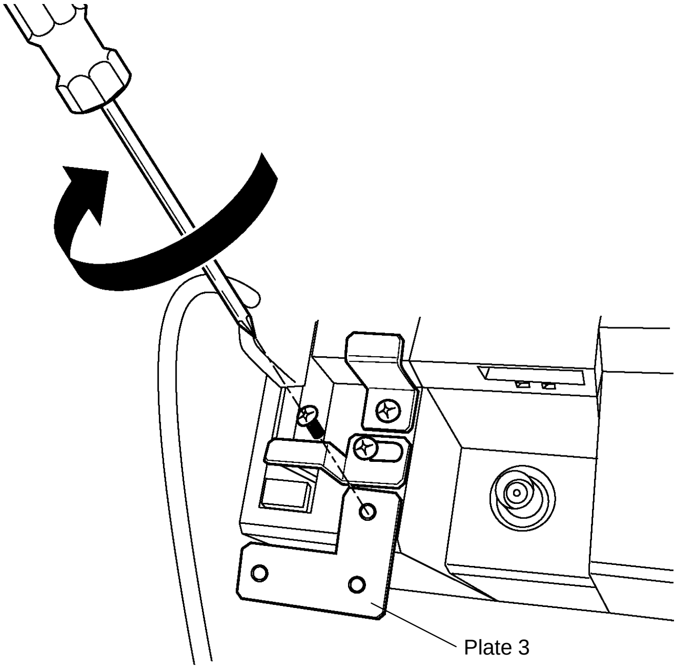
   To avoid accidental removal of the Ethernet Card, it is recommended that the Ethernet Card be installed in slot 1.



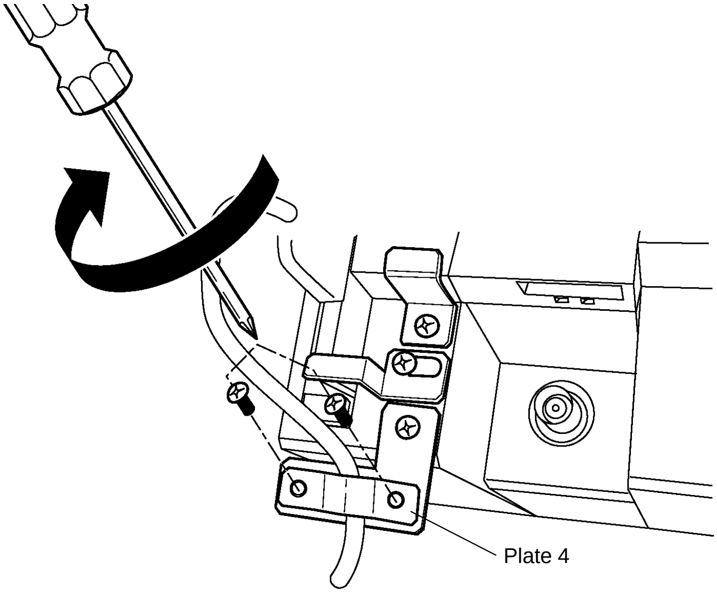
3. If a memory card is to be installed, install it in the same way as step 2.
4. Secure the Ethernet card with plate 2. To do so, secure plate 2 by tightening a set screw into the plate 2 mounting hole in the front panel of the PC Card Unit as shown below.



5. Secure plate 3 by tightening a set screw into the plate 3 mounting hole.

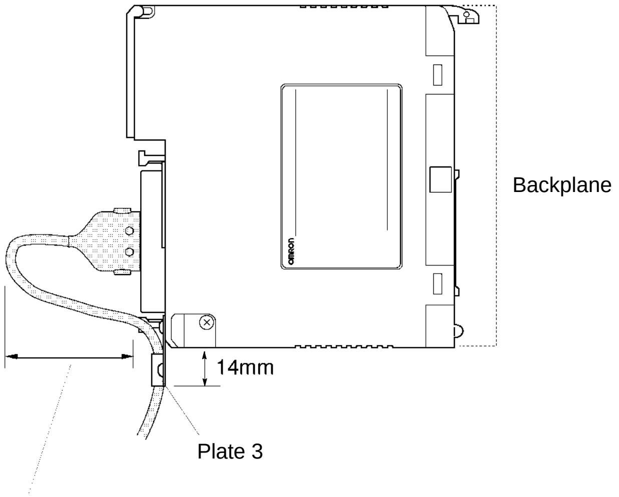


6. Secure the adapter cable with plates 3 and 4 by tightening set screws as shown below.



Installation Precautions

For the PC Card Unit Ethernet Set, consider the lengths of the cable and plate 3 indicated in the figure below to ensure that there is sufficient room to install the Ethernet Set properly.



Consider the length of this cable to ensure that there is sufficient room to install the Ethernet Set.

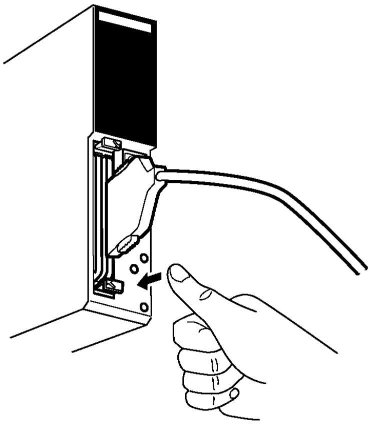
Removing an Ethernet Card

1, 2, 3...

1. Remove the set screw, then remove plate 2.
2. Remove the set screw, then remove plate 3.
3. Press the eject button as shown in the figure below.

   The card inserted into slot 2 can be removed or installed by loosening the set screws on plate 2. When removing or installing a card into slot 1, always remove plate 2.

Note Do not press the eject button with the plates mounted.



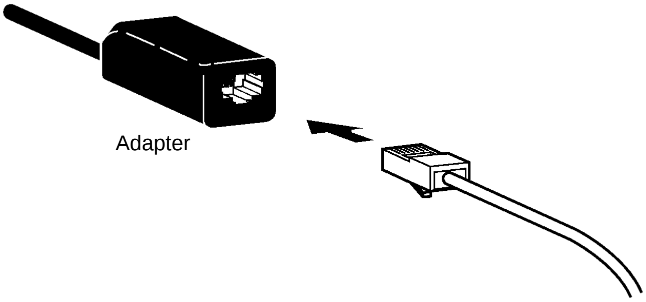
7-3 Connecting to the Ethernet Network

This section explains connection to a 10BASE-T network as an example of Ethernet connection.

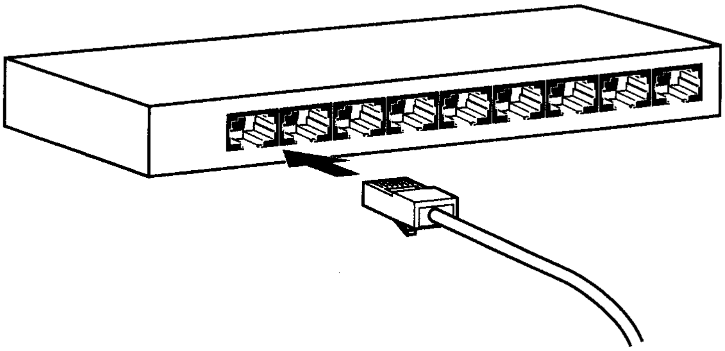
7-3-1 Connecting to the Hub

1, 2, 3...

1. Connect the adapter from the Ethernet card to a twisted-pair cable.

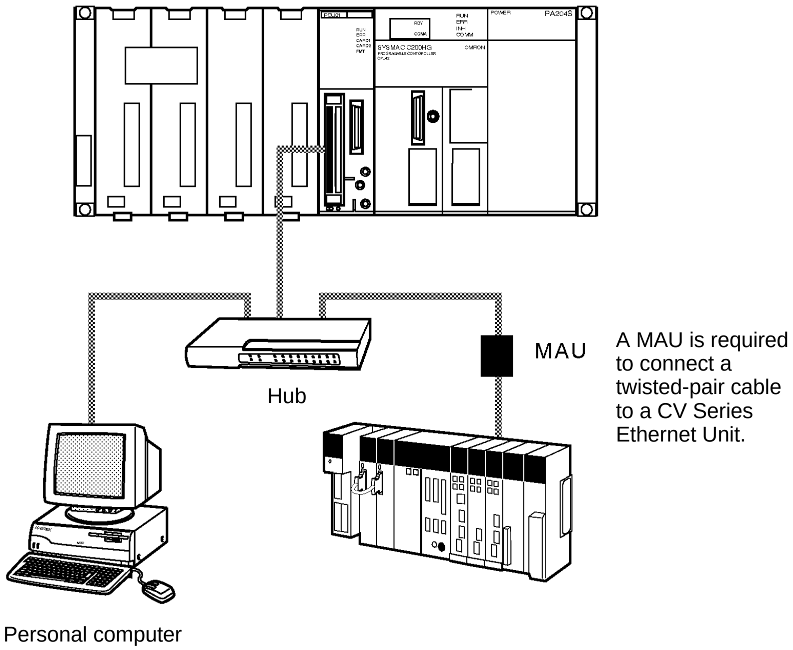


2. Connect the twisted-pair cable to the hub as shown below.



7-3-2 Constructing an Ethernet Network

The following diagram illustrates an example of Ethernet network configuration. For the types of networks and the types and number of devices that can be connected, refer to documentation for the hub.



A MAU is required to connect a twisted-pair cable to a CV Series Ethernet Unit.

7-4 Connecting to the Personal Computer

This section describes how to connect to a personal computer to enable using the setup software to set your Ethernet connection.

7-4-1 Restrictions on Personal Computers

Use a computer that meets the conditions shown below.

Applicable Computers and Operating Systems

An AT or compatible computer can be used. The computer must have at least one RS-232C port available.

Use an operating system supporting terminal software (e.g., HyperTerminal) that allows the exchange of binary files using the Zmodem protocol.
Example: Windows 98 or Windows NT 4.0.

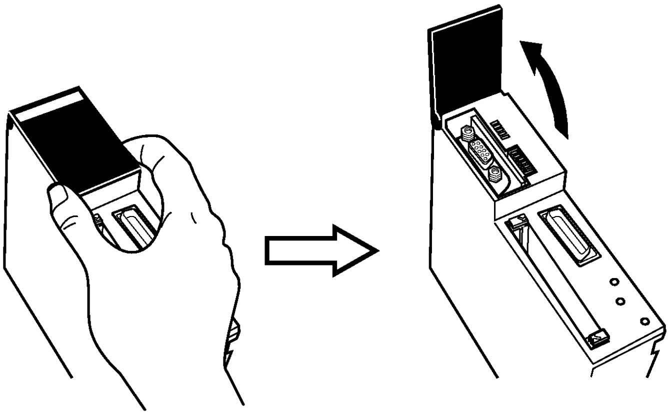
7-4-2 Connecting the Computer to the PC Card Unit

Connect the computer to the PC Card Unit with a Host Link Cable.

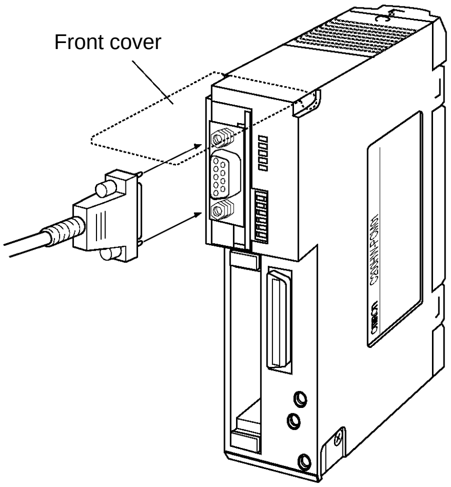
Note

1. Before connecting or disconnecting the cable, always turn both the PC Card Unit and the computer OFF.
2. If the computer does not have a 25-pin D-SUB connector RS-232C connector, wire your own cable. Refer to *Appendix C Connector Pin Assignments* for wiring.

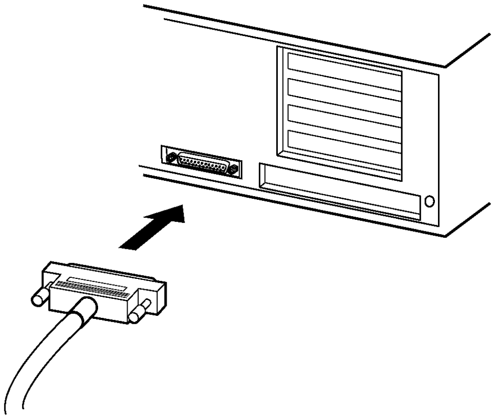
1, 2, 3... 1. Open the front cover as shown below.



2. Connect the 9-pin D-SUB connector of the Host Link Cable to the serial communication connector on the PC Card Unit. Insert the connector all the way in, then tighten the screws.



3. Connect the 25-pin D-SUB connector of the Host Link Cable to the serial communication connector on the personal computer. Insert the connector all the way in, then tighten the screws.



7-5 Setting Up the Ethernet Environment

The Ethernet communications environment must be set up to perform Ethernet communications from a PC Card Unit. This section describes how to set up the Ethernet environment with the setup software.

Refer to *7-6 Setup Software Operation* for details on how to use the setup software.

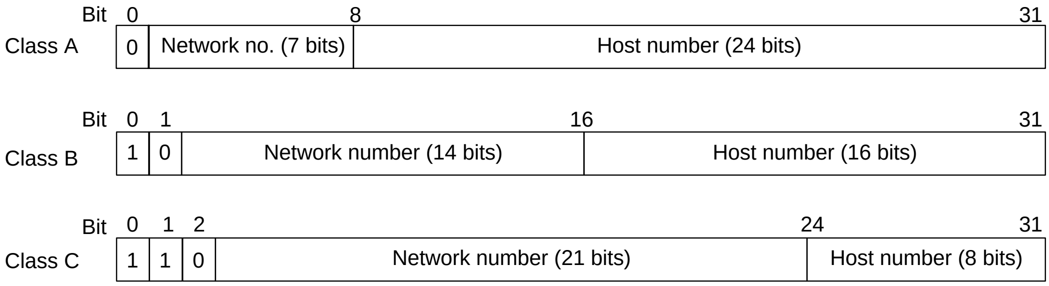
7-5-1 IP Address

Because a PC Card Unit uses an Ethernet expansion function (UDP/IP and TCP/IP), it must have its own IP address. An IP address is used to identify each node (such as computers, CV-series Ethernet Units, and PC Card Units) on an Ethernet network. IP addresses must be set and managed to ensure that each address is unique.

IP Address Structure

An IP address is a 32-bit binary data, consisting of a network number and a host number. The network number is an address used to identify each network. The host number is used to identify each host (i.e., node).

IP addresses are divided into three classes: Class A, Class B, and Class C. Select an applicable class according to the scale of the network being used.



The numbers of networks and hosts that can be identified differ according to the class.

Class	Number of networks	Number of hosts
Class A	Small	More than 65536 (2^{16})
Class B	Medium	256 (2^8) to 65536 (2^{16})
Class C	Large	Less than 256 (2^8)

An IP address consists of 32 bits divided into four segments of 8 bits each. These segments are separated with a dot and represented as four decimal numbers.

Example: 10000010 00111010 00010001 00100000 → 130.58.17.32

Note

1. All nodes within a network must have the same network number.
2. The network number of an IP address is used to identify each Ethernet network (IP network segment). The network number is not the same as the network address used for FINS communication.
3. An IP network segment is a logical network consisting of nodes that have the same network number.

Assigning IP Addresses

Assign a unique IP address to each node within a network (or among multiple networks).

The internet protocol (IP) is a standard international communications protocol. Therefore, obtaining an IP address from a public organization prevents address-related problems when the network is expanded in the future. Obtaining an official IP address is not required for a local Ethernet network that is not connected to outside networks. We do, however, recommend that you obtain an official IP address if there is any chance of connection to outside networks at any time in the future.

Assigning an IP Address to a PC Card Unit

To assign an IP address to the PC Card Unit, select “1. Enter IP Address for Unit” from the main menu of the setup software. Refer to *7-6 Setup Software Operation* for details on how to specify the IP address.

7-5-2 Subnet Mask

Connecting a lot of nodes to a network makes network operation and management difficult. In this case, it is better to construct the system by subdividing a network into multiple subnetworks. These subnetworks operate independently within the network, but they are viewed as a single network from outside the network.

To allow such network operations, the host number of the IP address must be subdivided into a host number and a subnet number.

A subnet mask is the information used to identify which part of the host number is going to be used as the subnet number. All bits in the subnet mask that correspond to the bits in the IP address used either as the network number or the subnet number are set to “1” and the remaining bits, which correspond to the bits in the IP address actually used for the host number, are set to “0.” In the following example, only the last 8 bits are used as the host number.

Example: 11111111 11111111 11111111 00000000 → FFFFFF00

All nodes on a subnetwork must have the same subnet mask value.

If no subnetwork is to be used, a subnet mask need not be specified. In this case, the subnet mask is recognized as the one that has no subnet number (or that consists of only the network number and host number).

Specifying a Subnet Mask for a PC Card Unit

To assign a subnet mask to a PC Card Unit, select “2. Enter Subnet Address for Unit” from the main menu of the setup software. Refer to *7-6 Setup Software Operation* for details on how to specify a subnet address.

7-5-3 ODI Driver

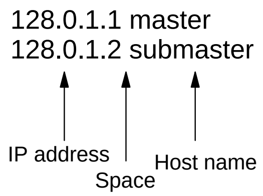
Specify the file name (including extension “.COM”) of the ODI driver provided with the Ethernet card.

To specify an ODI driver path, select “3. Enter ODI Driver Path” from the main menu of the setup software. Refer to *7-6 Setup Software Operation* for details on how to specify an ODI driver path.

7-5-4 Hosts

Specify the IP addresses and host names of all the nodes to be registered in the IP network to be used. For a PC Card Unit, up to 127 nodes can be specified.

Example:



Set the hosts for the IP network to be used. To do so, an existing HOSTS file can be copied, or a new HOSTS file can be created. Insert an space between the IP address and the host name. The maximum number of lines in a HOSTS file is 127.

To set the hosts, select “4. Edit HOSTS File” from the main menu of the setup software. Refer to *7-6 Setup Software Operation* for details on how to edit a HOSTS file.

7-5-5 Conversion Table for IP Addresses and FINS Node Addresses

When a UDP/IP address has been used on the Ethernet an IP address must be used for the specified node address. Communications using FINS commands use a FINS node address. The conversion between the IP address and the FINS node address is made in the PC Card Unit. The conversion table for IP addresses and FINS node addresses shows the correspondence between FINS node addresses and IP addresses.

Specify IP addresses in decimal notation. If 0 is added to the beginning of each field, the address will be interpreted as an octal number. Enter an space between the IP address and node address.

When using only the socket communications, also register the IP-FINS node address conversion table of the local node.

7-5-6 FINS Routing Tables

The FINS routing tables are required to return responses to FINS commands sent from a computer or CV-series PC on another network to the PC Card Unit. They show the route to each network to which response is to be returned.

The FINS routing tables consist of a local network table and a relay network table.

Note Even if there is only one network, always specify the local network table. If the PC Card Unit is to communicate with a node on another network, the relay station must be a CV-series PC.

If routing tables are already set for C200HX/HG/HE Programmable Controllers, the PC Card Unit will use those routing tables. Routing tables established by the setup software will be invalid.

Local Network Table

Specify the unit no. of the PC Card Unit and the network address of the network to which the PC Card Unit is connected.

- Local network address: Address of the network to which the PC Card Unit is connected (1 to 127)
- Unit no.: 0 (fixed)

Relay Network Table

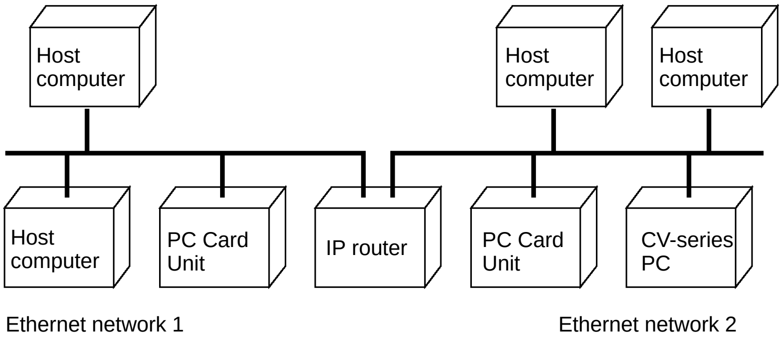
The relay network table shows the route to each network with which the PC Card Unit is to communicate. This table need not be specified if the PC Card Unit is to communicate only with nodes within the local network.

- End network: Address of the network on which the destination node resides (1 to 127)
- Relay network: Address of the network linking the local node to the next relay node (1 to 127)
- Relay node: Address of the relay node between the relay network and the end network (Ethernet Unit: 1 to 127)

To edit the routing tables, select “6. Edit Routing Table for FINS” from the main menu of the setup software. Refer to *7-6 Setup Software Operation* for details on how to edit the routing tables.

7-5-7 Gateway Address

An IP router can be used to connect multiple IP network segments. Networks connected by an IP router can communicate with each other. FINS communications allow nodes on different networks to communicate with one another as if these nodes are on the same network. To establish such communications, gateway addresses must be specified.



Assign an identical value to the FINS network addresses of the Ethernet networks connected by the IP router. (The network number or subnet number of the IP address, however, must be different for each network.) For example, the PC Card Units shown in the figure above have the same FINS network address (same layer).

Broadcasting is not allowed between Ethernet networks connected by an IP router. For example, data broadcast within Ethernet network 1 is not transmitted to nodes on Ethernet network 2.

To specify a gateway address, select “7. Edit Gateway Address for Unit” from the main menu of the setup software. Refer to *7-6 Setup Software Operation* for details on how to specify a gateway address.

7-5-8 Other Ethernet Environment Settings

The following Ethernet environment settings cannot be changed for the PC Card Unit:

- Broadcast address: All bits are fixed at 1.
- Packet size: The size is fixed at 1,500 bytes. UDP data is split into 1,472-byte data before transmission. TCP is 1,024 bytes.
- KEEP ALIVE: Fixed at 60 seconds.
- Send/Receive packet: Fixed at 2,048 bytes.

7-5-9 Setting the Port Number

In the Ethernet Set, FINS communications are processed through the UDP port. The default port number is 9600.

To change the port number, use the following procedure.

1, 2, 3...

1. With reference to the information on making backup copies in *7-7 Backing Up and Restoring Settings*, make backup copies of the setting files.
2. Using the text editor of the personal computer, change the last line of the BASE.BAT file in the same folder as the setting files in the way shown below. Input the port number in decimal.

   base_e → base_e <*port_number*>
3. After saving the BASE.BAT file, with reference to the information on restoring files in *7-7 Backing Up and Restoring Settings*, restore the BASE.BAT file.

   When the port number is changed, only the BASE.BAT file has to be restored. It is not necessary to restore other files.

7-6 Setup Software Operation

7-6-1 Setup Procedure

The procedure required for making Ethernet communications environment settings using the setup software is given below. The setup software is an application that operates on the PC Card Unit and is used from the terminal software installed on the personal computer.

1, 2, 3...

1. Turn OFF the power supply to the C200HX/HG/HE and the personal computer.
2. Connect the personal computer and the PC Card Unit using an RS-232C cable. Refer to *7-4 Connecting to the Personal Computer* for details.
3. Start up the terminal software and set the communications conditions.
4. Change the PC Card Unit's startup mode (set pins 5 and 6 of the system switch to ON).
5. Turn ON the power supply to the C200HX/HG/HE and the personal computer.
6. Start up the setup software (SETUP.EXE) using the terminal software on the personal computer.
7. Make the Ethernet communications environment settings using the setup software.
8. Restart the PC Card Unit in normal mode (i.e., with pins 5 and 6 of the system switch set to OFF). At this point the settings changes are enabled. Refer to *7-6-12 Returning the PC Card Unit to Normal Mode* for details.

Note

1. Do not use the setup software for the C200HW-PCS01(-EV1) to perform settings for the C200HW-PCS01-V2.
2. Backing Up and Restoring Settings
   Before changing settings using the setup software, it is recommended that a backup copy is made of the settings. Backing up and restoring settings is performed not from the setup software, but from HyperTerminal. For details of the procedures, refer to *7-7 Backing Up and Restoring Settings*.

7-6-2 Preset Items

Before starting Ethernet communications through the Ethernet card, use the setup software to set up the Ethernet communications environment.

The items set using the setup software are described on the following pages.

Preset Items

Menu	Description
1. Enter IP address for UNIT	Specify the IP address of the PC Card Unit.
2. Enter Subnet address for UNIT	Specify the subnet mask of the PC Card Unit.
3. Enter ODI Driver Path.	Specify the file name (including the extension ".COM") of the ODI driver provided with the Ethernet card. An ODI driver (cntpclat.com) for the C-NET(PC)C-10L Ethernet card made by CONTEC is installed in the Unit.
4. Edit HOSTS File.	Set the HOSTS file for the IP network. To do so, you can copy the existing HOSTS file or create a new HOSTS file. Insert a space between the IP address and host name. The maximum number of lines in the HOSTS file is 127.
5. Edit Conversion Table for IP address and FINS Node address.	Specify the conversion correspondence between each pair of a FINS node address and IP address. Specify IP addresses in decimal notation. Note that if 0 is added to the beginning of each field, the address will be interpreted as an octal number. Enter a space between the IP address and node address.
6. Edit Routing Table for FINS.	Specify the data required for internetwork communication. The data consists of a local network table and a relay network table. Even if there is only one network, specify just the local network table. If the PC Card Unit is to communicate with a node on another network, the relay station must be a CV-series PC.
7. Edit Gateway address for UNIT	Enter the IP address to be used as a gateway with other Ethernet networks.
8. Display File Details.	Not used.
9. Exit	Exit the setup software to return to the DOS prompt.

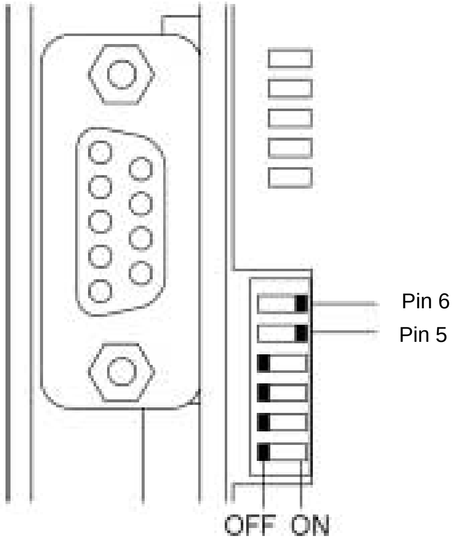
7-6-3 Connecting to the PC Card Unit

The procedure for connecting to the PC Card Unit is given below. The connection method is explained here using HyperTerminal, which is provided with Windows operating systems, as an example.

Before making the settings described below, connect the personal computer and the PC Card Unit using a Host Link Cable with reference to *7-4 Connecting to the Personal Computer*.

Changing the Startup Mode (System Switch)

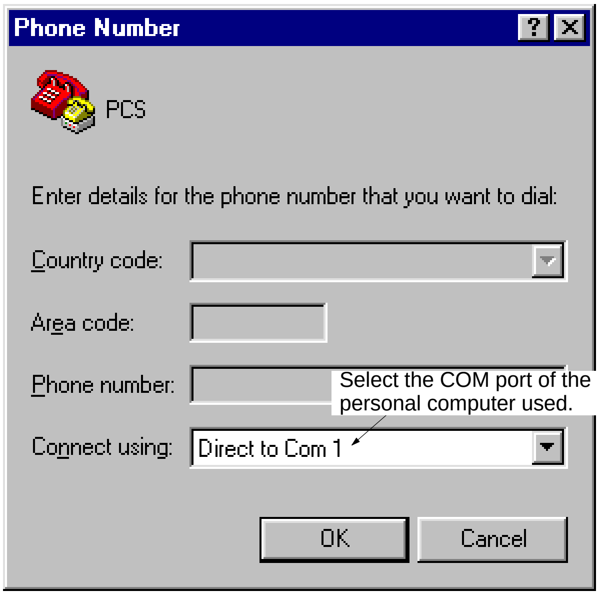
Change the PC Card Unit's startup mode (system switch) by setting pins 5 and 6 to ON.



HyperTerminal Startup and Initial Settings

1, 2, 3...

1. Start up HyperTerminal at the personal computer. The *Connection Description* dialog box will be displayed.
2. Enter a name. In this example, the name "PCS" is used. Next, the dialog box shown below will be displayed.


Phone Number

PCS

Enter details for the phone number that you want to dial:

Country code:

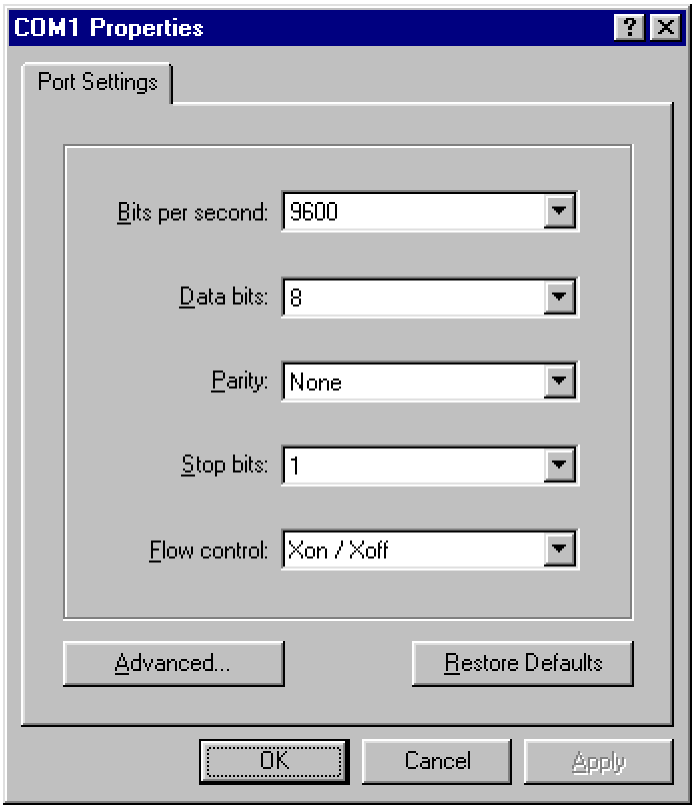
Area code:

Phone number:

Connect using: Direct to Com 1

Select the COM port of the personal computer used.

3. Click the “OK” Button, and make the following communications settings in the dialog box displayed next.



COM1 Properties

Port Settings

Bits per second: 9600

Data bits: 8

Parity: None

Stop bits: 1

Flow control: Xon / Xoff

4. Click the “OK” Button.
5. Turn ON the power supply to the PC Card Unit. (If the PC Card Unit has already been started up, press ENTER.) The PC Card Unit will start up. If connections have been made successfully, the following prompt will be displayed.

```
:
:
F: ¥>█
```

7-6-4 Starting and Exiting the Software

This section describes how to start and exit the setup software.

Note Backing Up and Restoring Settings
Before changing settings using the setup software, it is recommended that a backup copy is made of the settings. Backing up and restoring settings is performed not from the setup software, but from HyperTerminal. For details of the procedures, refer to *7-7 Backing Up and Restoring Settings*.

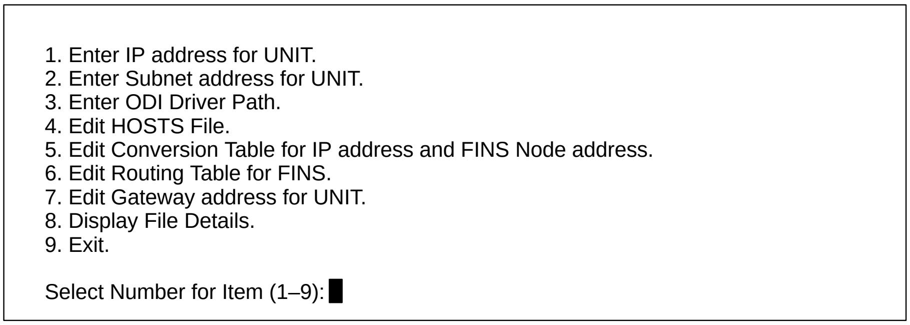
Starting the Setup Software

1, 2, 3... 1. Enter “SETUP” from the terminal software.

Note For Japanese display, be sure to specify the “J” option. If this option is not specified, messages will be displayed in English.

```
F: \>SETUP
```

The setup software is started, and the following main menu appears on the display:



```
1. Enter IP address for UNIT.
2. Enter Subnet address for UNIT.
3. Enter ODI Driver Path.
4. Edit HOSTS File.
5. Edit Conversion Table for IP address and FINS Node address.
6. Edit Routing Table for FINS.
7. Edit Gateway address for UNIT.
8. Display File Details.
9. Exit.

Select Number for Item (1–9): █
```

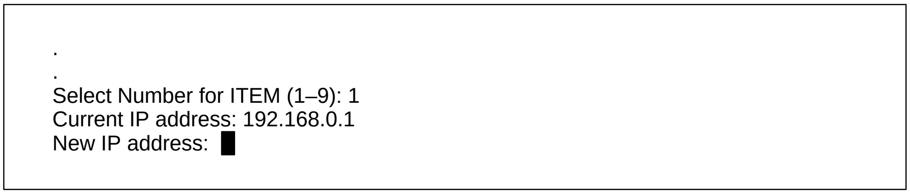
2. Select an item by inputting a number between 1 and 9 and pressing ENTER. If a number between 1 and 7 is selected, the settings for that item can be read or changed.

Exiting the Setup Software Select “9. Exit” from the main menu.
The setup software terminates and a prompt is displayed in the terminal software.

7-6-5 IP Address

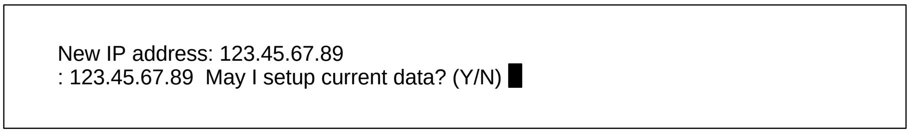
Specify the IP address of the PC Card Unit as described below.

1, 2, 3... 1. Select “1. Enter IP address for UNIT” from the main menu.



```
.
.
Select Number for ITEM (1–9): 1
Current IP address: 192.168.0.1
New IP address:  █
```

2. Specify the IP address of the PC Card Unit. Input a decimal number.



```
New IP address: 123.45.67.89
: 123.45.67.89  May I setup current data? (Y/N) █
```

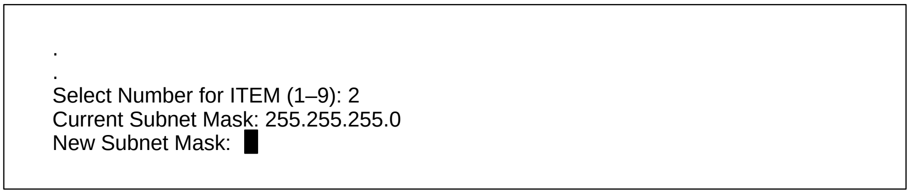
3. Make sure that the IP address is correct. Then, enter Y and press ENTER. The screen will return to the main menu. To cancel the setting, enter N and press ENTER to return to the main menu.

Note When the IP address of the PC Card Unit has been changed, always select “4. Edit HOSTS File.” and set the HOSTS file.

7-6-6 Subnet Mask

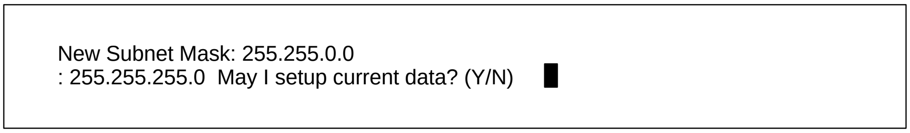
Specify the subnet mask of the PC Card Unit as described below.

1, 2, 3... 1. Select “2. Enter Subnet address for UNIT” from the main menu.



```
.
.
Select Number for ITEM (1–9): 2
Current Subnet Mask: 255.255.255.0
New Subnet Mask:  █
```

2. Specify the subnet mask of the PC Card Unit. Input a decimal number.



```
New Subnet Mask: 255.255.0.0
: 255.255.255.0  May I setup current data? (Y/N)
```

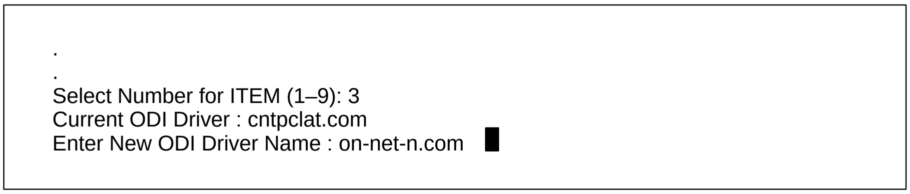
3. Make sure that the subnet mask is correct. Then, enter Y and press ENTER. The screen will return to the main menu. To cancel the setting, enter N and press ENTER to return to the main menu.

7-6-7 ODI Driver

Set the ODI driver to be used with the Ethernet card.

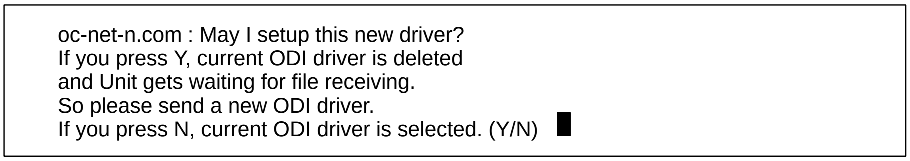
The default setting is “cntpclat.com.”

1, 2, 3... 1. Select “3. Enter ODI Driver Path.” from the main menu.



```
.
.
Select Number for ITEM (1–9): 3
Current ODI Driver : cntpclat.com
Enter New ODI Driver Name : on-net-n.com
```

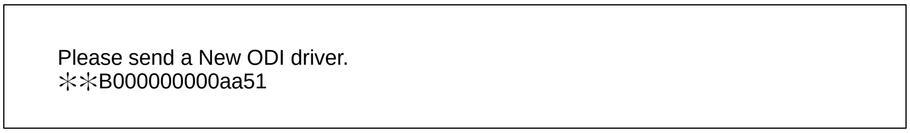
2. Enter an ODI driver name (including the extension “.COM”).
   The drive and the directory cannot be specified. Only the filename must be specified.



```
oc-net-n.com : May I setup this new driver?
If you press Y, current ODI driver is deleted
and Unit gets waiting for file receiving.
So please send a new ODI driver.
If you press N, current ODI driver is selected. (Y/N)
```

Note After confirming that the current ODI driver is no longer required, enter Y and press ENTER. (The current ODI driver will be erased.)

3. Make sure that the ODI driver name is correct. To cancel the setting, enter N and press ENTER to return to the main menu. If Y is entered and ENTER is pressed, the PC Card Unit will go on standby to receive 1 file via the Zmodem protocol, and the following message will be displayed.

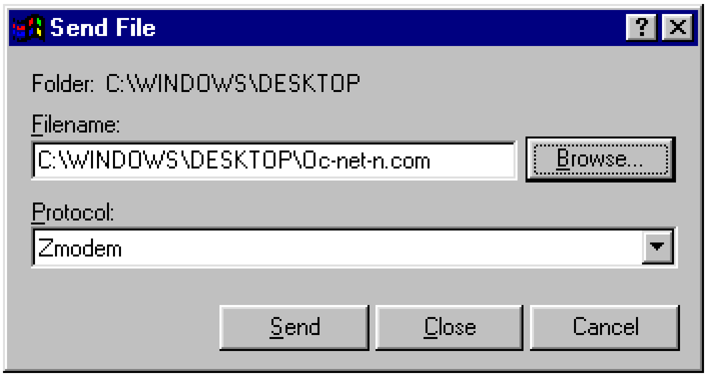


```
Please send a New ODI driver.
**B000000000aa51
```

4. After confirming that the PC Card Unit is on standby by checking that the message above is displayed, start up the terminal software, and send the new ODI driver to the PC Card Unit.

Example: Procedure for HyperTerminal

- Select “Transfer” and then “Send File.” The “Send File” dialog box shown below will be displayed.
- Specify the path and file name of the ODI driver specified in step 2 and click the “Send” Button.



Send File

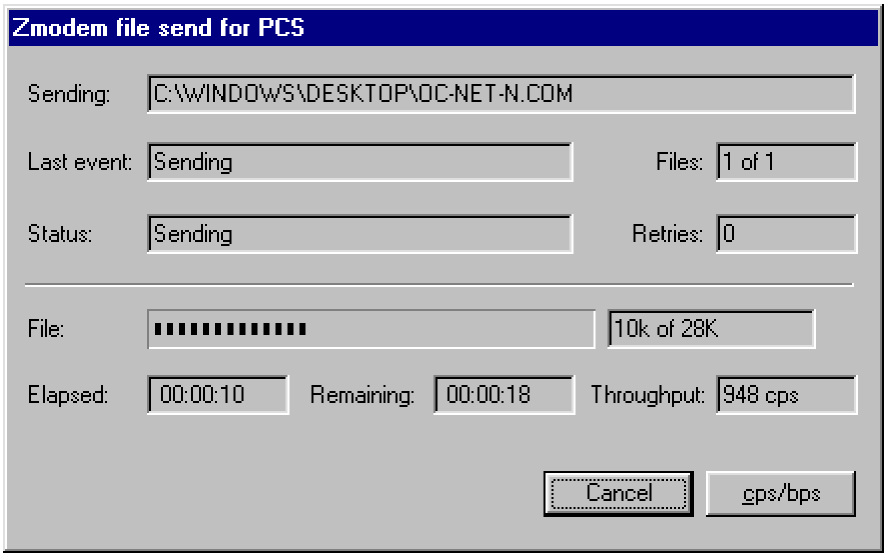
Folder: C:\WINDOWS\DESKTOP

Filename: C:\WINDOWS\DESKTOP\0c-net-n.com

Protocol: Zmodem

Note Be sure to select “Zmodem” as the protocol.

- The transfer will start and the following dialog box will be displayed.



Zmodem file send for PCS

Sending: C:\WINDOWS\DESKTOP\0C-NET-N.COM

Last event: Sending Files: 1 of 1

Status: Sending Retries: 0

File: 10k of 28K

Elapsed: 00:00:10 Remaining: 00:00:18 Throughput: 948 cps

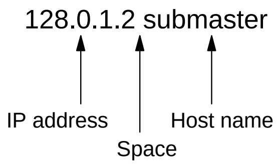
- When transfer is completed, the screen will return to the setup software’s main screen.

7-6-8 HOSTS File

Edit the HOSTS file as described below.

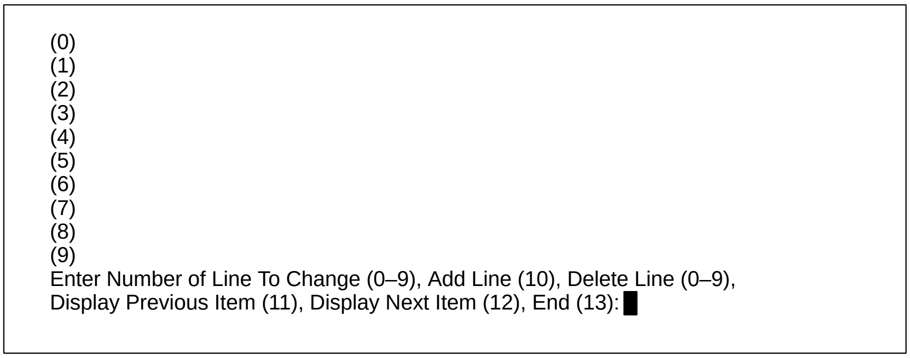
If a HOSTS file already exists in your personal computer, copy the file beforehand. In each record of the HOSTS file, enter the IP address, a space, and the host name in this order. Up to 127 nodes can be specified. An existing HOSTS file can be used, but the maximum number of lines is still 127.

Example:



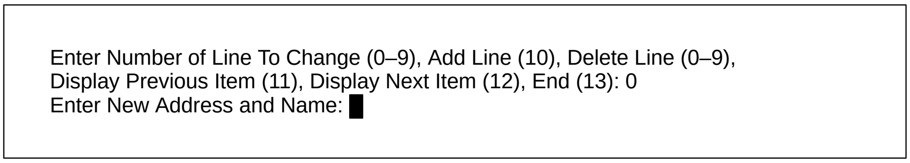
128.0.1.2 submaster

1, 2, 3... 1. Select “4. Edit HOSTS File.” from the main menu.



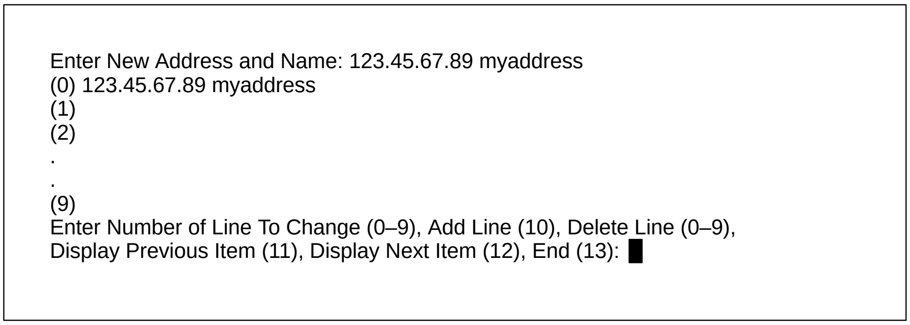
```
(0)
(1)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13):█
```

2. Select 0.



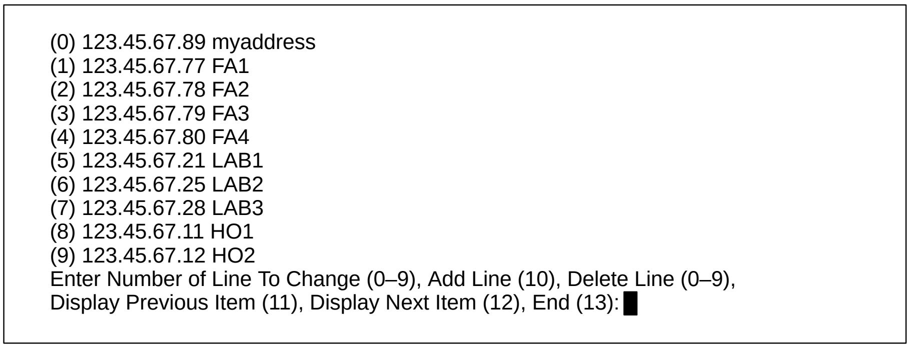
```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): 0
Enter New Address and Name: █
```

3. Specify the IP address and host name of the remote node. To do so, enter the IP address, a space, and the host name in this order.



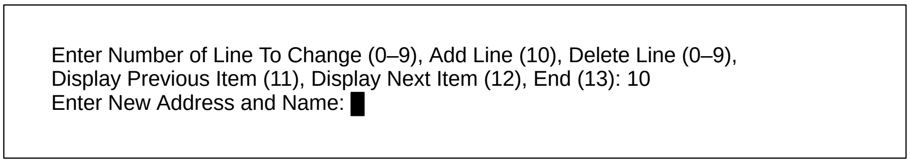
```
Enter New Address and Name: 123.45.67.89 myaddress
(0) 123.45.67.89 myaddress
(1)
(2)
.
.
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): █
```

4. Select 1, and specify the IP address and host name in the same way as for step 3. Then, continue with this setting procedure for the rest of the HOSTS file ((2) and onwards).



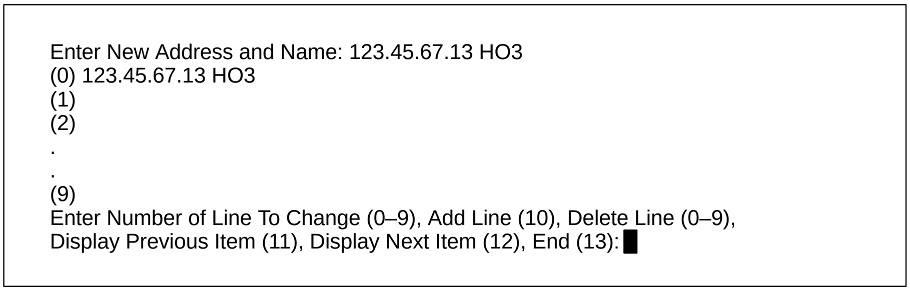
```
(0) 123.45.67.89 myaddress
(1) 123.45.67.77 FA1
(2) 123.45.67.78 FA2
(3) 123.45.67.79 FA3
(4) 123.45.67.80 FA4
(5) 123.45.67.21 LAB1
(6) 123.45.67.25 LAB2
(7) 123.45.67.28 LAB3
(8) 123.45.67.11 HO1
(9) 123.45.67.12 HO2
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13):█
```

- The screen only displays (0) to (9), as shown in the figure above.
- To specify the line next to (9) (eleventh line), select “Add Line (10).”



```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): 10
Enter New Address and Name: █
```

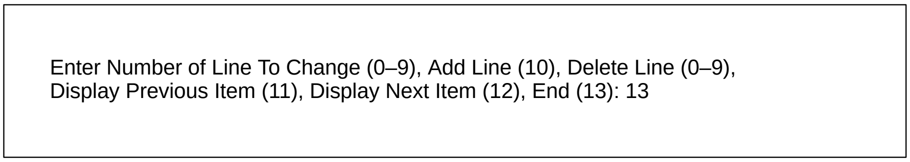
- If an IP address and host name are entered, the display will change as shown below.



```
Enter New Address and Name: 123.45.67.13 HO3
(0) 123.45.67.13 HO3
(1)
(2)
.
.
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): █
```

- Then, select the line on which an IP address and host name are to be entered (or select “Add Line (10)”), and enter the IP address and host name. Continue with this setting procedure as required.
- To delete data from a line, select “Delete Line (0-9)”, enter the line number (0 to 9), and press ENTER. When “Enter New Address and Name:” is displayed, press ENTER. The data is deleted from the line.
- Selecting “Display Previous Item (11)” scrolls up the screen to display the previous 10 lines.
- Selecting “Display Next Item (12)” scrolls down the screen to display the next 10 lines.

5. After editing the HOSTS file, select “End (13).”



```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): 13
```

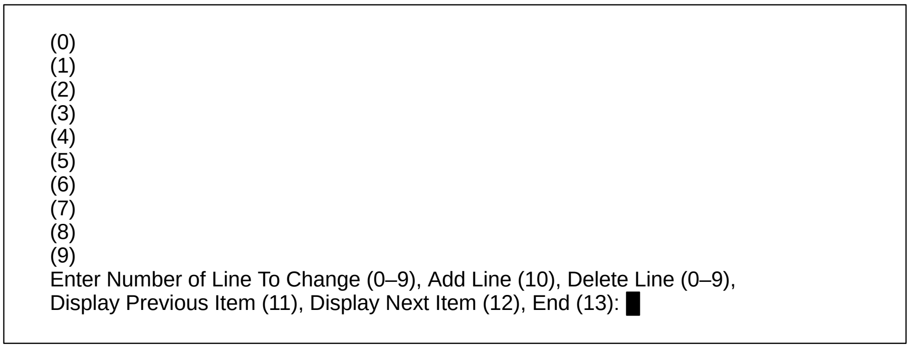
The screen will return to the main menu.

7-6-9 Address Conversion Table

Edit the conversion table file for IP addresses and FINS node addresses as described below. Up to 127 nodes can be specified.

1, 2, 3...

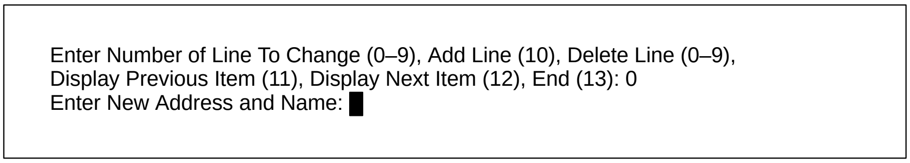
1. Select “5. Edit Conversion Table for IP address and FINS Node address” from the main menu.



```
(0)
(1)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): █
```

As shown above, no addresses are registered as default settings.

2. Select 0.



```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): 0
Enter New Address and Name: █
```

3. Specify the IP address of the remote node and the corresponding FINS node address. To do so, enter the IP address, a space, and the FINS node address (0 to 127) in this order.

```
Enter New Address and Name: 123.45.67.89 14
(0) 123.45.67.89 14
(1)
(2)
.
.
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): █
```

4. Select 1, and specify the IP address and the corresponding FINS node address in the same way as Step 3.

   Then, continue with this setting procedure for the rest of the conversion table ((2) and onwards).

   Example:

```
(0) 123.45.67.89 14
(1) 123.45.67.77 15
(2) 123.45.67.78 16
(3) 123.45.67.79 21
(4) 123.45.67.80 22
(5) 123.45.67.21 31
(6) 123.45.67.25 32
(7) 123.45.67.28 33
(8) 123.45.67.11 40
(9) 123.45.67.12 41
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): █
```

- The screen only displays (0) to (9) as shown in the figure above.
- To specify the line next to (9) (eleventh line), select “Add Line (10).”

```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): 10
Enter New Address and Name: █
```

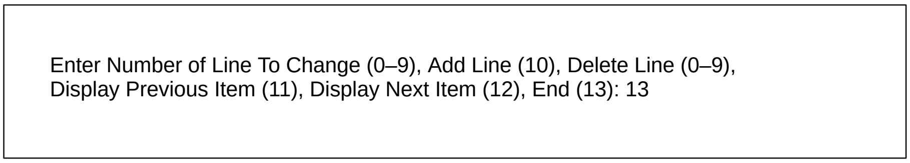
- If an IP address and FINS node address are entered, the display is changed as shown below.

```
Enter New Address and Name: 123.45.67.13 42
(0) 123.45.67.13 42
(1)
(2)
.
.
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): █
```

- Then, select the line on which an IP address and host name are to be entered (or select “Add Line (10)”), and enter the IP address and host name. Continue with this setting procedure as required.
- To delete data from a line, select “Delete Line (0-9)”, enter the line number (0 to 9), and press ENTER. When “Enter New Address and Name:” is displayed, press ENTER. The data is deleted from the line.

- Selecting “Display Previous Item (11)” scrolls up the screen to display the previous 10 lines.
- Selecting “Display Next Item (12)” scrolls down the screen to display the next 10 lines.

5. After editing the Conversion Table for IP Address and FINS Node Address, select “End (13).”



```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13): 13
```

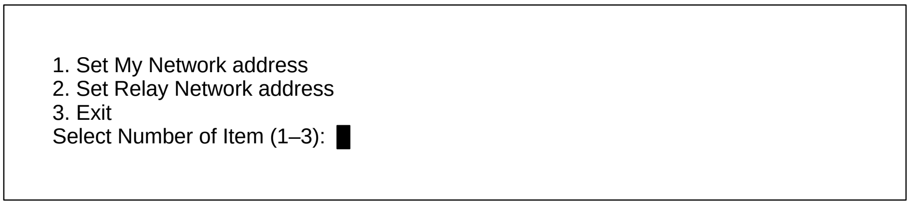
The screen will return to the main menu.

7-6-10 FINS Routing Tables

Editing the routing tables for FINS is described next. Set the local network and relay network.

Note Always specify both the local network and relay network during the editing procedure. Even if only one of the networks needs to be modified, always perform the setting procedures for both of them.

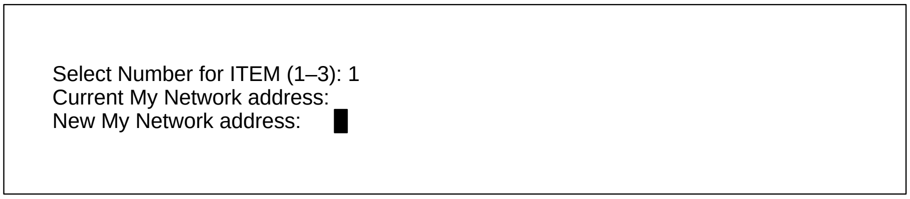
1, 2, 3... 1. Select “6. Edit a Routing Table for FINS.” from the main menu.



```
1. Set My Network address
2. Set Relay Network address
3. Exit
Select Number of Item (1–3): █
```

Local Network

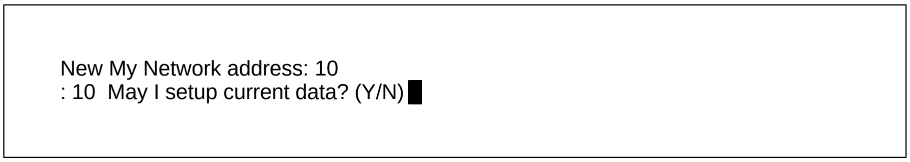
2. Enter 1 and press ENTER to select “1. Set My Network address.”



```
Select Number for ITEM (1–3): 1
Current My Network address:
New My Network address:     █
```

3. Specify the local network number. To do so, enter a local network no., a space, and 0 (fixed) in this order. “0” is the unit no. of a PC Card Unit.

   If a node number is to be specified from another node via FINS communication, specify 16 ($10). The node no. of the C200HX/HG/HE CPU is 0.

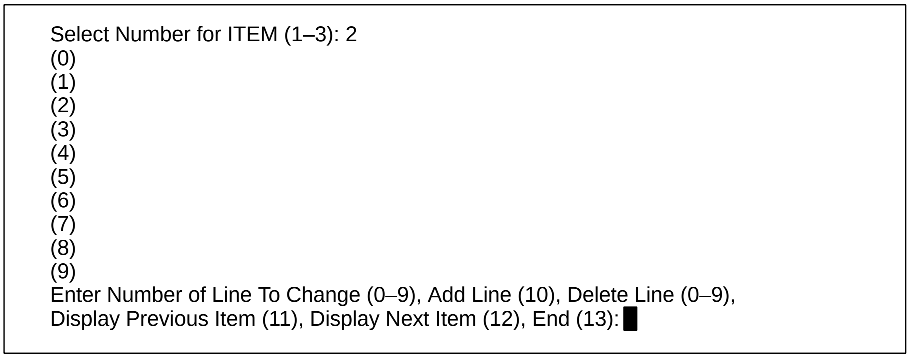


```
New My Network address: 10
: 10  May I setup current data? (Y/N)█
```

4. Make sure that the local network number is correct. Then, enter Y and press ENTER. The screen will return to the screen displayed in Step 1. To cancel the setting, enter N and press ENTER to return to the screen displayed in Step 1.

Relay Networks

5. Enter 2 and press ENTER to select “2. Set Relay Network address.”



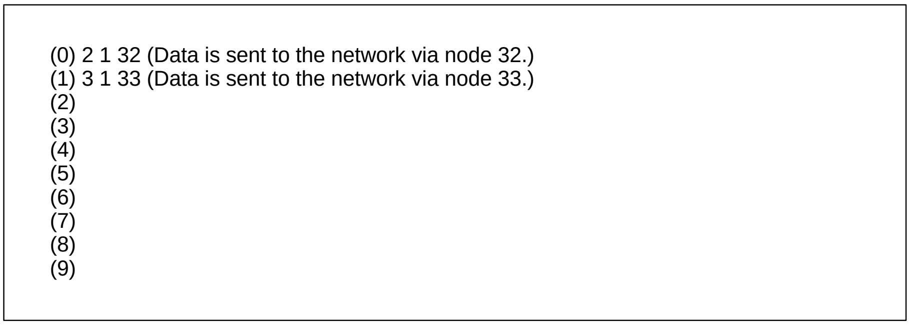
```
Select Number for ITEM (1–3): 2
(0)
(1)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13):
```

By default, nothing is yet registered, as shown above.

6. Specify the relay network. To do so, enter the number of the network containing the end target node (decimal number), a space, the local network no. (decimal number), a space, and the relay node no. in this order. (The relay node no. is the number of the exit node within the local network that is used to transfer data to the end target network.)

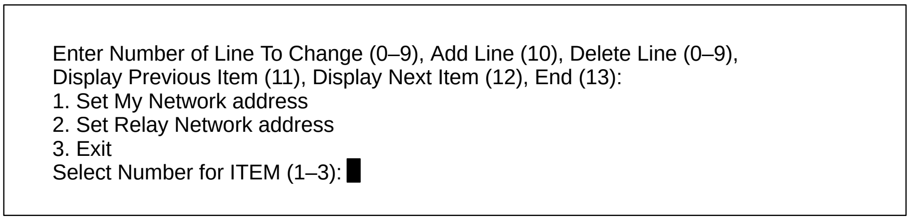
   Up to 127 networks can be specified.

   An example of specifying relay networks is shown below. The input procedure is the same as that described in *7-6-8 HOSTS File*.



```
(0) 2 1 32 (Data is sent to the network via node 32.)
(1) 3 1 33 (Data is sent to the network via node 33.)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
```

7. After editing the relay network table, select “End (13).”



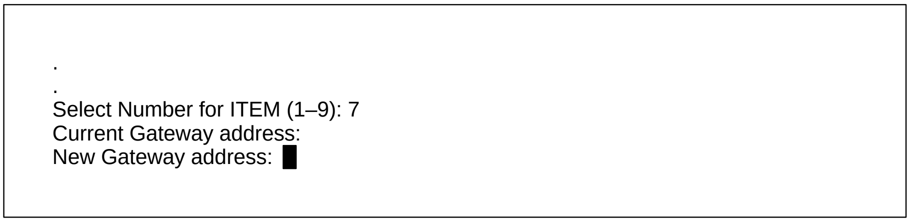
```
Enter Number of Line To Change (0–9), Add Line (10), Delete Line (0–9),
Display Previous Item (11), Display Next Item (12), End (13):
1. Set My Network address
2. Set Relay Network address
3. Exit
Select Number for ITEM (1–3):
```

8. Enter 3 and press ENTER to select “3. Exit.”
   The screen will return to the main menu.

7-6-11 Gateway Address

1, 2, 3...

1. Select “7. Edit Gateway address for UNIT” from the main menu.



```
.
.
Select Number for ITEM (1–9): 7
Current Gateway address:
New Gateway address:
```

2. Enter the IP address of the node to be used as a gateway.

```
New Gateway address: 123.45.0.0
: 123.45.0.0  May I setup current data? (Y/N) █
```

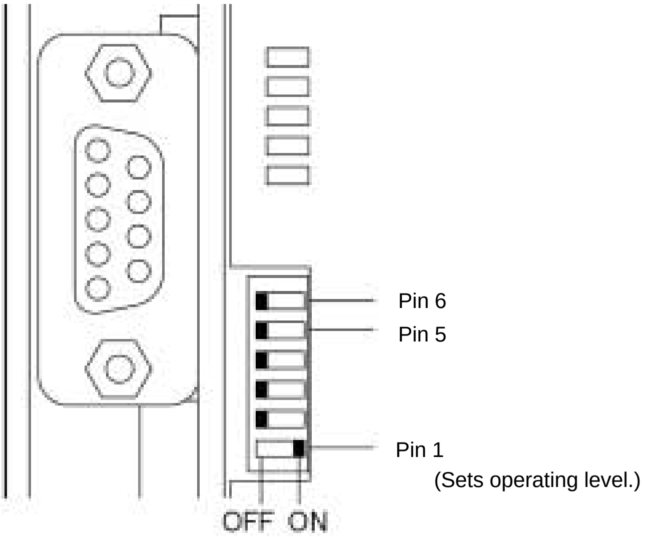
3. Make sure that the IP address is correct. Then, enter Y and press ENTER. The screen will return to the main menu. To cancel the setting, enter N and press ENTER to return to the main menu.

7-6-12 Returning the PC Card Unit to Normal Mode

Use the following procedure to return the PC Card Unit to normal mode.

1, 2, 3...

1. Turn OFF the power to the C200HX/HG/HE and the personal computer.
2. Disconnect the RS-232C cable.
3. Change the PC Card Unit’s startup mode (system switch) by setting pins 5 and 6 to OFF.



4. Turn ON the power to the C200HX/HG/HE and the personal computer. The PC Card Unit will start up in normal mode.

7-7 Backing Up and Restoring Settings

Before changing the PC Card Unit’s Ethernet communications environment settings settings, it is recommended that a backup copy is made of the settings.

The methods for making a backup copy of the Ethernet communications environment settings stored in the PC Card Unit, and for restoring these settings to the PC Card Unit are explained below.

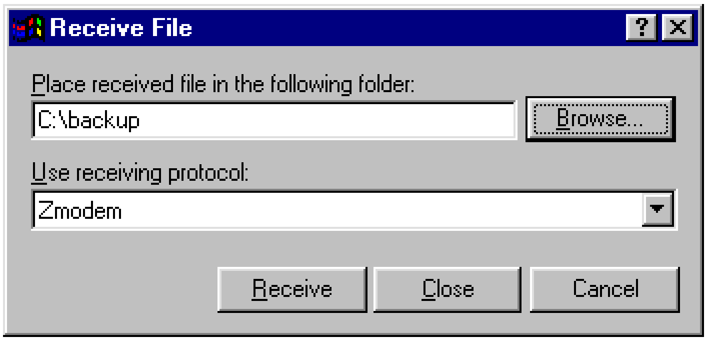
7-7-1 Backing Up Files

Use the following procedure to make a backup copy of the Ethernet communications environment settings saved in the PC Card Unit.

1, 2, 3...

1. Turn OFF the power supplies to the personal computer and the PC Card Unit.
2. Connect the personal computer and the PC Card Unit using a Host Link Cable. (Refer to 7-4 *Connecting to the Personal Computer.*)

3. Set pins 5 and 6 of the PC Card Unit's system switch to ON, and start up the PC Card Unit.
4. Start up HyperTerminal at the personal computer and connect to the PC Card Unit. (Refer to *7-6 Setup Software Operation*.)
5. At the HyperTerminal, select "Transfer" and "Receive File." The "Receive File" dialog box shown below will be displayed.
6. Specify the directory in which the backup file is to be saved, and click the "Close" Button.



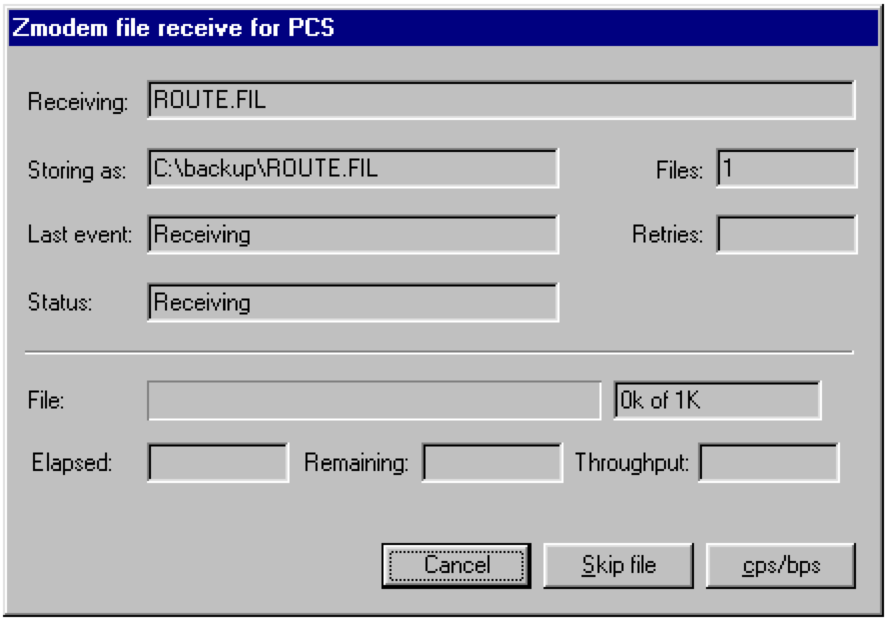
Place received file in the following folder: C:\backup

Use receiving protocol: Zmodem

Note 1. When making a backup copy, be sure to specify "Zmodem" under "Use receiving protocol:."

2. If "Zmodem with Crash Recovery" is selected, and backup is made several times to the same directory, files with the same name will have a number added to the filename. When making backup copies more than once, use a different directory.

7. After the prompt, enter "backup" and press ENTER. Backup processing (i.e., transfer of the file from the PC Card Unit to the personal computer) is started, and the following dialog box is displayed.



Receiving: ROUTE.FIL

Storing as: C:\backup\ROUTE.FIL

Files: 1

Last event: Receiving

Retries:

Status: Receiving

File: 0k of 1K

Elapsed:

Remaining:

Throughput:

When backup processing is completed, the screen will return to the prompt screen.

Note For details on the method for returning the PC Card Unit to normal mode, refer to *7-6-12 Returning the PC Card Unit to Normal Mode*.

Backed Up Files

If backup is completed successfully, the following 8 files are copied to the specified directory in the personal computer.

alias
base.bat
net.cfg
nowodi
ip_fins.fil
route.fil
setup.fil
hosts

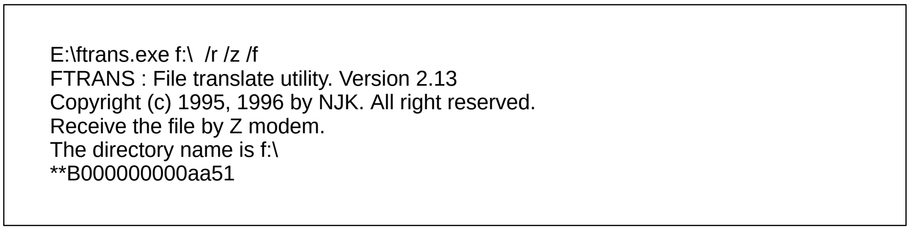
Note When restoring files, be sure to transfer all of the above files to the PC Card Unit.

7-7-2 Restoring Files

Use the following procedure to restore the Ethernet communications environment settings (which were backed up to the personal computer) to the PC Card Unit.

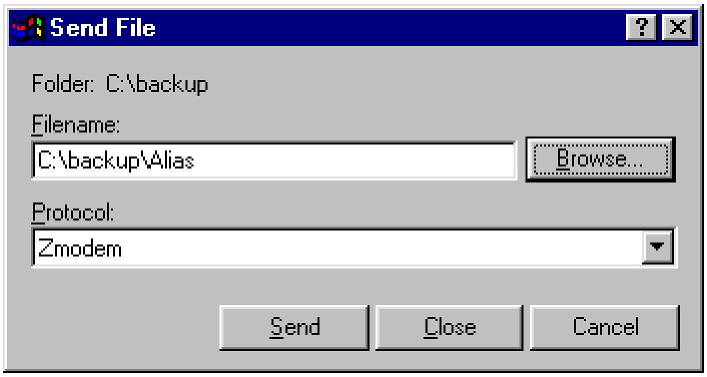
Note
1. Be sure to transfer all 8 backed up files to the PC Card Unit. For details on backing up files, refer to *7-7-1 Backing Up Files*.
2. With HyperTerminal, only one file can be sent for each transfer operation. Therefore, when restoring backed up files, the “Send File” operation must be performed 8 times.

1, 2, 3...
1. Turn OFF the power supplies to the personal computer and the PC Card Unit.
2. Connect the personal computer and the PC Card Unit using a Host Link Cable. (Refer to 7-4 *Connecting to the Personal Computer.*)
3. Start up HyperTerminal at the personal computer. (Refer to *7-6 Setup Software Operation*.)
4. Set pin 5 of the system switch to ON, set pin 6 to OFF, and then start up the PC Card Unit.



```
E:\ftrans.exe f:\  /r /z /f
FTRANS : File translate utility. Version 2.13
Copyright (c) 1995, 1996 by NJK. All right reserved.
Receive the file by Z modem.
The directory name is f:\
**B000000000aa51
```

5. At the HyperTerminal, select “Transfer” and “Send File.” The “Send File” dialog box shown below will be displayed.
6. Specify the backup file to be restored.



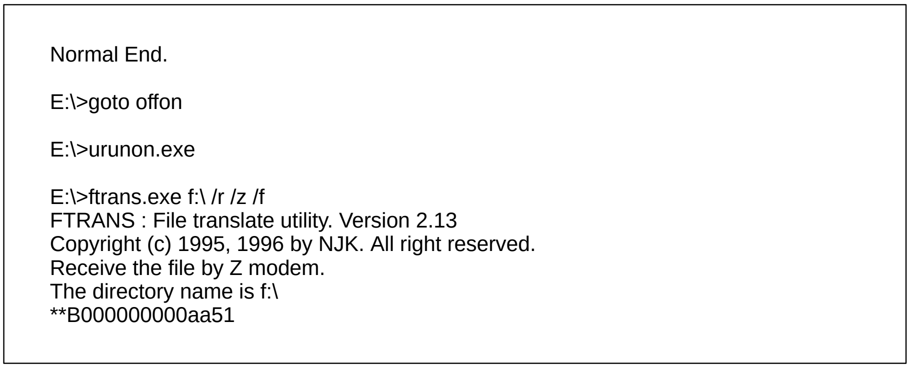
Send File

Folder: C:\backup

Filename: C:\backup\Alias

Protocol: Zmodem

7. Click the “Send” Button.



```
Normal End.

E:\>goto offon

E:\>urunon.exe

E:\>ftrans.exe f:\ /r /z /f
FTRANS : File translate utility. Version 2.13
Copyright (c) 1995, 1996 by NJK. All right reserved.
Receive the file by Z modem.
The directory name is f:\
**B000000000aa51
```

8. Repeat steps 5 to 7 until all the backed up files have been transferred to the PC Card Unit. Refer to *Backed Up Files* on page 101 for the names of the files.
9. After the last file has been sent, confirm that the PC Card Unit is on standby to receive files.
10. Set pins 5 and 6 of the PC Card Unit’s system switch to OFF, and then restart the PC Card Unit. This will enable the restored Ethernet communications environment settings.

SECTION 8
Using SEND(90) and RECV(98)

This section describes how to use the SEND(90) and RECV(98) instructions to transfer data.

8-1 Outline

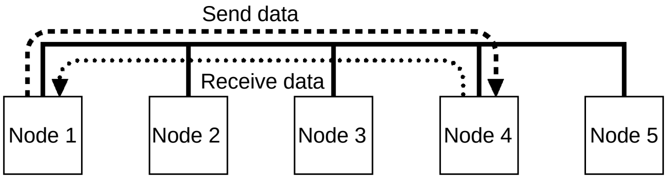
The SEND(90) and RECV(98) instructions are used in the PC user program to send data from a PC to other nodes, such as PCs or computers.

8-1-1 Specifying Destination Nodes

Either of the following two communications modes is used depending on how destination nodes are specified:

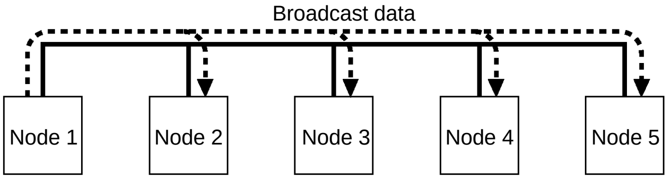
Node-to-node Data Transfer

Data can be transferred between the local node and the destination node by specifying the node address in the control data of the SEND(90) and RECV(98) instructions.



Broadcasting

The same data can be sent to all nodes on the network by specifying 00 as the node address of the destination node. In this case, no response is received from the nodes. (Broadcasting is not applicable to RECV(98).)



8-1-2 Communications Specifications

PC communications are based on the following specifications:

Item	Specifications
Direction of transmission	1:1 SEND(90) and RECV(98) instructions 1:N SEND(90) instruction (broadcast mode, no response, N = 127 or less)
Data length	SEND(90): 1,000 words max. (2,000 bytes) (727 words when data is broadcast or sent to local node) RECV(98) : 1,000 words max. (2,000 bytes)
Data contents	The following data is sent or received when each instruction is executed: SEND(90): Data transmission request and response data RECV(98): Data reception request and response data
Response monitoring time	00: Default (2.2 seconds) 01 to FF: User setting (approx. 0.1 to 25.5 seconds in 0.1 s increments)
No. of retries	0 to F: 0 to 15 times

UDP Port No. 9600 is used for SEND/RECV.

8-1-3 Send/Receive Data Areas

The data areas that can be transferred using the SEND(90) and RECV(98) instructions differs according to the PC model being used. Check the operation manual for your PC for details.

Send/Receive Data Areas

When remote node is C-series PC			When remote node is CV-series PC		
Area	**Area type**	**Word addresses**	**Area**	**Area type**	**Word addresses**
IR and SR areas	00	0 to 511	CIO Area	00	0 to 2555
LR area (LR)	06	0 to 63	CPU Bus Link Area (G)	01	0 to 255
HR area (HR)	07	0 to 99	Auxiliary Area (A)	02	0 to 511
AR area (AR)	08	0 to 27	Timer Area (TIM)	03	0 to 1023
Timer/Counter Area (TIM/CNT)	03	0 to 511	Counter Area (CNT)	04	0 to 1023
DM Area (DM)	05	0 to 9999	DM Area (DM)	05	0 to 24575
Expansion DM Banks 0 to 7: Current bank:	 10 to 17 18	0 to 6143	Expansion DM Banks 0 to 7: Current bank:	 10 to 17 18	0 to 32765

Indirect Addressing: Operand 2 (D) Data When Bit 12 of Word (C+1) = 1

Word	Bits 15 to 08	Bits 07 to 04	Bits 03 to 00
D	Area code	0	Word address
(D+1)	Word address	Word address	Word address

Specified in binary coded decimal (BCD)

Set the send/receive area within the above ranges.

A data length of 1,000 words is valid only when the remote node supports it. If the remote node can only return 256 words, any data greater than 257 words is not guaranteed.

The PC Card Unit supports a data length of 1 to 997 words for the SEND(90) instruction, and a data length of 1 to 1,000 words for the RECV(98) instruction.

8-1-4 Instruction Status

Status information for executed SEND(90), RECV(98), and CMCR instructions is reflected in the SR Area of the PC as instruction execution flags and instruction response codes.

Structure of Instruction Execution Flags

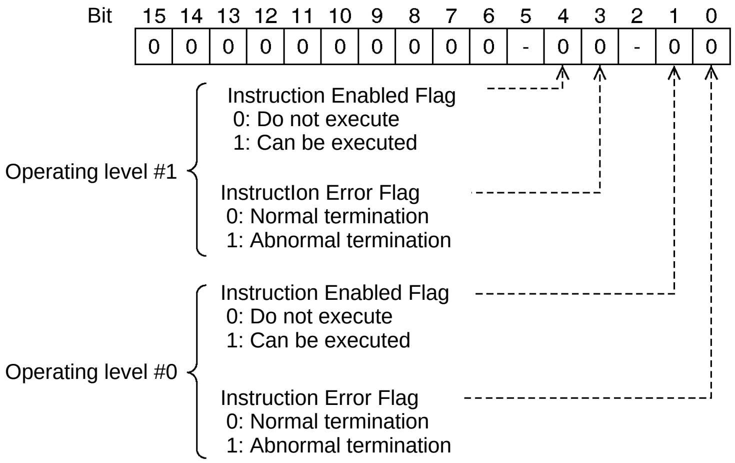
The following two instruction execution flags are used:

Instruction Enabled Flag:
ON (“1”) when the SEND(90), RECV(98), or CMCR instruction can be executed.

Instruction Error Flag:
ON (“1”) when the SEND(90), RECV(98), or CMCR instruction has terminated abnormally.

If multiple instructions are to be used, always make sure that the Instruction Enabled Flag is ON before executing the next instruction (i.e., perform exclusive control).

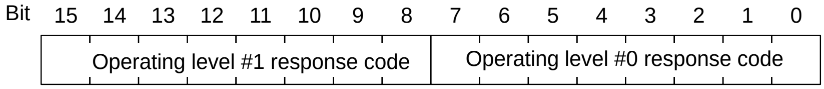
C200HX/HG/HE Word SR 252



Note Only one instruction can be executed for each operating level at a time. (Two instructions can be simultaneously executed on two operating levels.) If more than one instruction is to be used on the same operating level, perform exclusive control by using the Instruction Enabled Flag, i.e., do not execute SEND(90) or RECV(98) while the Instruction Enabled Flag for that level is ON.

Instruction Response Codes When an instruction terminates, the status is recorded as a response code. The code is retained until the next instruction is executed. It is set to 00 during execution. The instruction response code is one byte and differs from the command/ response end code (2 bytes).

C200HX/HG/HE Word SR 237


Bit
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Operating level #1 response code
Operating level #0 response code

Response Codes

Response code	Item	Explanation
00	Normal completion	Processing terminated normally.
01	Parameter error (cannot send)	SEND(90)/RECV(98) instruction parameter specification error Local node address specification error
02	Cannot send	The system was reset during instruction processing. The local node has not been registered in the network.
05	Response time-out	No response was received within response monitoring time.
06	Response error	Refer to *Response Error* below.

Response Error The SEND(90) and RECV(98) instructions are converted to FINS commands and are sent to the destination node from the PC Card Unit. The SEND(90) instruction is converted to MEMORY AREA WRITE and the RECV(98) instruction is converted to MEMORY AREA READ. When these SEND(90) and RECV(98) instructions are executed, if the instructions are not completed normally (i.e., the MRES is not 0), a response error is generated. (Refer to 124.)

Possible causes for a response error include the following.

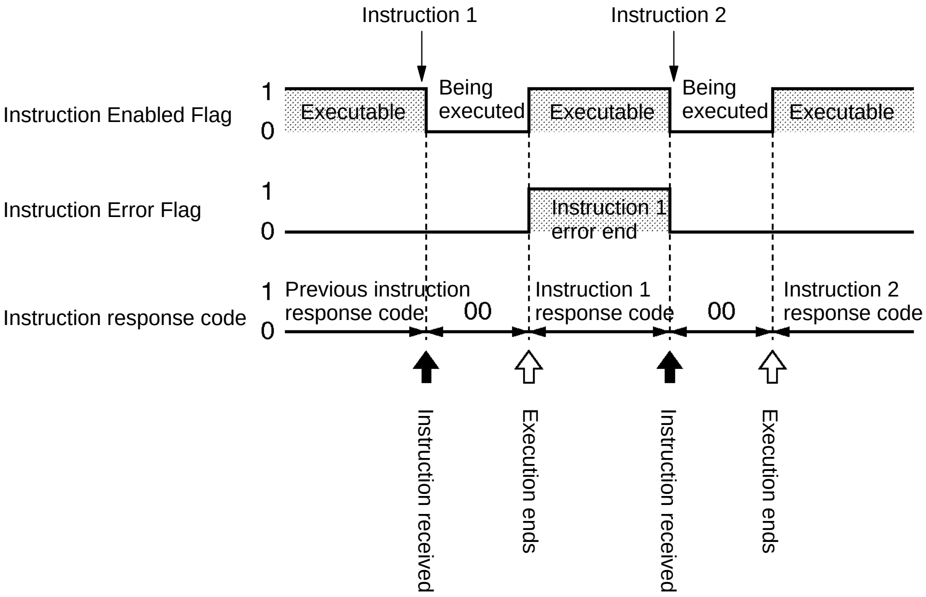
- The destination node is not participating in the network.
- The destination node is busy.
- Processing is being performed at the destination node and so commands cannot be received.

- Communications controller error
- Node address setting error
- PC error

Flag ON/OFF Timing

The Instruction Enabled Flag, Instruction Error Flag, and instruction response code are refreshed as shown below.

Example: When two instructions are executed consecutively and the first instruction causes an error



8-1-5 SEND(90)/RECV(98)/CMCR Data Processing Timings

The table below shows when data for the SEND(90)/RECV(98)/CMCR instruction is sent.

Send data	Send/Receive end processing
All data that has been processed by the time the END instruction is executed is sent.	Processing is performed when the END instruction is executed.

8-2 SEND(90)

The SEND(90) instruction sends data from a memory area of the local node to a memory area of the destination node.

Format

The SEND(90) instruction format is shown below.

Executed each Cycle	Differentiated
SEND(90)	@SEND(90)
S	S
D	D
C	C

S: Beginning source word (rightmost) (local node)
D: Beginning destination word (rightmost) (destination node)
(Beginning indirect reception word)
C: First control data word (rightmost) (local node)

Control Data

Control data is written beginning with the first control data word in the following format:

	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word	C+0	Number of words transferred															
	C+1	*1	*2	*3	*4	No. of retries				Response monitoring time							
	C+2	0	0	0	Unit address of destination node					Destination node address							

*1: Fixed at 1
*2: Operating level (0: level #1, 1: level #0)
*3: Response request bit (0: Response required, 1: Response not required)
*4: Indirect addressing bit for beginning destination word
(0: Direct addressing, 1: Indirect addressing)

Number of words transferred:
Specify the total number of words of data to be stored in the destination node in hexadecimal (0 to 3E8).

Operating Level:
Specify the operating level of the PC Card Unit.

Response request bit:
If a response is not required, set this bit to 1 (ON: response not required). For broadcasting, the bit is automatically set to 1.

Indirect addressing bit for beginning destination word:
Because CV-series PCs support larger areas than C-series PCs, the user may not be able to specify the beginning destination word of the destination node for the SEND(90) instruction operand. The beginning destination word of the destination node may also need to be changed in some situations. In this case, set this bit to 1 (ON: indirect addressing). Then, the second operand (D) will be an indirect beginning reception word, allowing the destination beginning word (rightmost) of the destination node to be specified with the word specified for D in BCD format as shown below.

	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word	D+0	Area type								0	0	0	0	Word addr (5th digit)			
	D+1	Word addr (4th digit)				Word addr (3rd digit)				Word addr (2nd digit)				Word addr (1st digit)			

Specify the area as follows:

When remote node is a C-series PC		When remote node is a CV-series PC	
Area	Area type	Area	Area type
IR/SR area	00	CIO Area	00
LR area	06	CPU Bus Link Area (G)	01
HR area	07	Auxiliary Area	02
AR area	08	Timer Area	03
Timer/Counter Area	03	Counter Area	04
DM area	05	DM Area	05
Expansion DM Banks 0 to 7: Current bank:	10 to 17 18	Expansion DM Banks 0 to 7: Current bank:	10 to 17 18

No. of retries:
Specify the maximum number of retries to be executed when no response is returned after the SEND(90) instruction is executed. Specify the number in hexadecimal (0 to F).

Response monitoring time:
Specify the response wait time when the response request bit is set to 0 (response required). Specify the time in hexadecimal (0 to FF). This setting is not valid for broadcasting because no response is returned.

Unit address of destination node:
The address is fixed at $00.

Destination node address:
Specify the node address of the destination node in hexadecimal. If $00 is specified, data is broadcasted to all nodes on the network.

Control Data Settings

The table below shows the range of values that can be set for each item.

Item	Setting
Number of words transferred	$0000 to $03E8 (0 to 1,000 words)
Operating level	0 (OFF): level #1 1 (ON): level #0
Response request bit	0 (OFF): Response required 1 (ON): Response not required
Indirect addressing bit for beginning destination word	0 (OFF): Direct addressing 1 (ON): Indirect addressing
No. of retries	$0 to $F (0 to 15)
Response monitoring time	$00: Default (2.2 seconds) $01 to $FF (approx. 0.1 to 25.5 seconds in 0.1 s increments)
Unit address of destination node	$00: PC's CPU
Destination node address	$01 to $7E: Destination node address (1 to 127) $00: Broadcast (Local node address cannot be specified.)

Note No response is returned for broadcasting even if the response request bit is set to 0 (OFF: response required).

8-3 RECV(98)

The RECV(98) instruction writes data from the memory area of the source (remote) node to the memory area of the local node.

Format

The RECV(98) instruction format is shown below.

Executed each Cycle	Differentiated
RECV(98)	@RECV(98)
S	S
D	D
C	C

S: Beginning source word (rightmost) (remote node)
(Beginning indirect reception word)
D: Beginning destination word (rightmost) (local node)
C: First control data word (rightmost) (local node)

Control Data

Control data is written beginning with the first control data word in the following format:

Word	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C+0	Number of words transferred															
	C+1	*1	*2	*3	*4	No. of retries				Response monitoring time							
	C+2	0	0	0	Unit address of source node					Source node address							

*1: Fixed at 1
*2: Operating level (0: level #1, 1: level #0)
*3: Response request bit (0: Response required, 1: Response not required)
*4: Indirect addressing bit for beginning source word
(0: Direct addressing, 1: Indirect addressing)

Number of words transferred:
Specify the total number of words to be written from the source node to the local node in hexadecimal (0 to 3E8).

Operating Level:
Specify the operating level of the PC Card Unit

Response request bit:
Set this bit to 0 (OFF: response required) because the RECV(98) instruction must return response data.

Indirect addressing bit for beginning source word:
Because CV-series PCs support larger areas than C-series PCs, the user may not be able to specify the beginning source word of the source node in the RECV(98) instruction operand. The beginning source word of the source node may also need to be changed in some situations. In this case, set this bit to 1 (ON: indirect addressing). Then, the first operand (S) becomes an beginning indirect source word, allowing the beginning source word (rightmost) at the source node to be specified with the word specified for S in BCD format as shown below.

Word	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S+0	Area type								0	0	0	0	Word addr (5th digit)			
	S+1	Word addr (4th digit)				Word addr (3rd digit)				Word addr (2nd digit)				Word addr (1st digit)			

When remote node is C-series PC		When remote node is CV-series PC	
Area	**Area type**	**Area**	**Area type**
IR/SR area	00	CIO Area	00
LR area (LR)	06	CPU Bus Link Area (G)	01
HR area (HR)	07	Auxiliary Area (A)	02
AR area (AR)	08	Timer Area (TIM)	03
Timer/Counter Area (TIM/CNT)	03	Counter Area (CNT)	04
DM area (DM)	05	DM Area (DM)	05
Expansion DM Banks 0 to 7: Current bank:	10 to 17 18	Expansion DM Banks 0 to 7: Current bank:	10 to 17 18

No. of retries:
Specify the maximum number of retries to be executed when no response is returned after the RECV(98) instruction is issued. Specify the number in hexadecimal (0 to F).

Response monitoring time:
Specify the response wait time when the response request bit is set to 0 (response required). Specify the time in hexadecimal (0 to FF).

Unit address of source node:
The address is fixed at $00

Source node address:
Specify the node address of the source node in hexadecimal (1 to 7E).

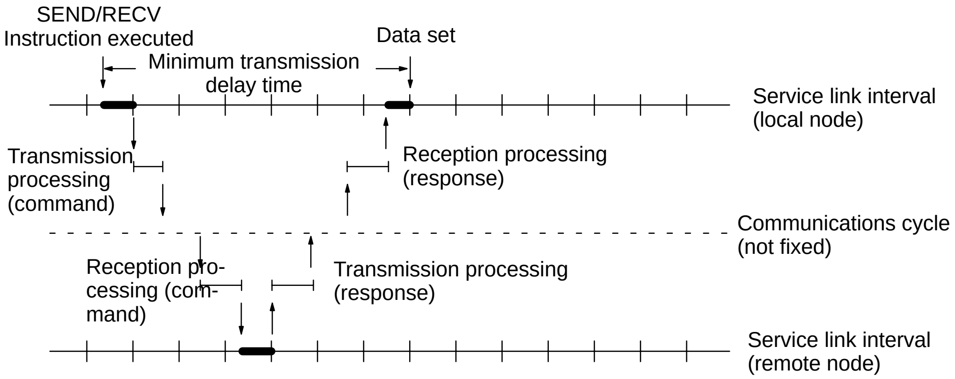
Control Data Settings

The table below shows the range of values that can be set for each item.

Item	Setting
Number of words transferred	$0000 to $03E8 (0 to 1000word)
Operating level	0 (OFF): level #1 1 (ON): level #0
Response request bit	0 (OFF): Response required
Indirect addressing bit for beginning source word	0 (OFF): Direct addressing 1 (ON): Indirect addressing
No. of retries	$0 to $F (0 to 15)
Response monitoring time	$00: Default (2.2 seconds) $01 to $FF (approx. 0.1 to 25.5 seconds in 100 ms increments)
Unit address of source node	$00: PC
Source node address	$01 to $7E: Source node address (1 to 127)

8-4 Minimum Transmission Delay Time for SEND/RECV Instructions

The minimum transmission delay time for SEND/RECV instructions is as illustrated below.


SEND/RECV
Instruction executed
Minimum transmission delay time
Data set
Service link interval (local node)
Transmission processing (command)
Reception processing (response)
Communications cycle (not fixed)
Reception processing (command)
Transmission processing (response)
Service link interval (remote node)

The Ethernet communications cycle time differs according to system configuration and load, so in this instance the communications cycle time is calculated as 0.

Minimum transmission delay time
= service link interval (local node) + transmission processing (command)
+ communications cycle + reception processing (command)
+ service link interval (remote node) + transmission processing (response)
+ communications cycle + reception processing (response)
+ service link interval (local node).

- **Service Link Interval**
  One cycle time for the C200HX/HG/HE CPU

- **Transmission/Reception Processing (Command)**
  For SEND instructions: No. of transmission words × 0.013 + 5 ms
  For RECV instructions: 5 ms
- **Transmission/Reception Processing (Response)**
  For SEND instructions: 5 ms
  For RECV instructions: No. of transmission words × 0.013 + 5 ms

Example: When the RECV instruction is executed for word 256.

Local node cycle time × 2 + remote node cycle time + 5 ms + 5 ms
+ (256 × 0.013 + 5) ms + (256 × 0.013 + 5) ms
= local node cycle time × 2 + remote node cycle time + 26.656 ms.

Actual Measurement Values

For communication with a C200HX, cycle time of 1.5 ms and during the SYSMAC LINK data link

Word 256 SEND/RECV instruction execution times for a C200HX/HG/HE.

SEND Instruction	34.6 ms
RECV Instruction	37.0 ms

Word 1 SEND/RECV instruction execution times for a C200HX/HG/HE.

SEND Instruction	28.4 ms
RECV Instruction	27.4 ms

SECTION 9
FINS Commands

This section provides information on communicating in Ethernet Systems using FINS commands, and explains how to use the CMCR instruction to issue FINS commands.

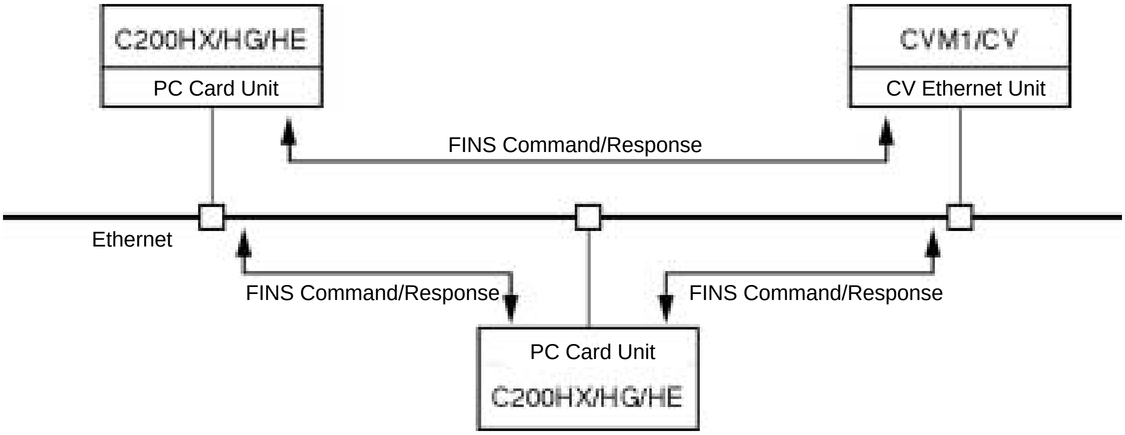
9-1 FINS Communications Service

The FINS communications service is a communications protocol that was developed by OMRON for its factory automation (FA) networks. FINS communications allow C200HX/HG/HE Programmable Controllers 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.

For details regarding FINS commands, refer to *Section 11 Using FINS Commands and Responses* in this manual and also to the *FINS Commands Reference Manual (W227)*.

Communication Units that provide the FINS communications service utilize the particular functions of the network type that is being used. The PC Card Unit executes the FINS communications service using UDP/IP. When FINS commands are used for communications between Programmable Controllers as shown in the following illustration, however, communications are executed without the user having to pay attention to the UDP/IP protocol.

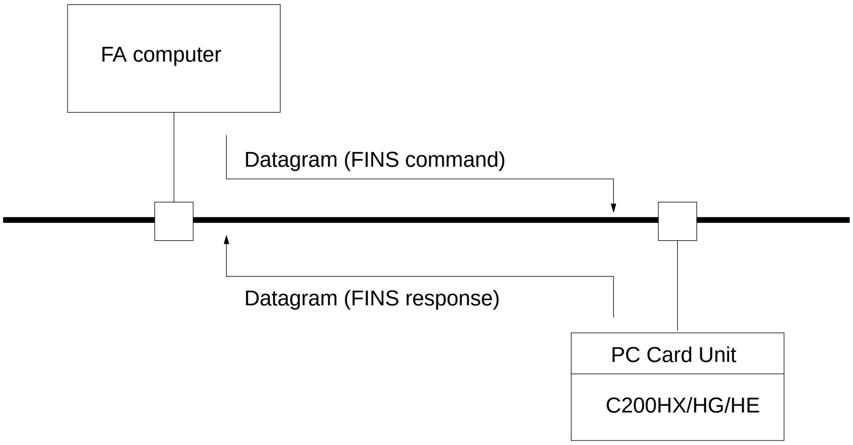


Note The PC Card Unit uses the CMCR instruction for sending and receiving FINS commands.

When communications are transmitted from an FA computer to a C200HX/HG/HE as shown in the following illustration, the C200HX/HG/HE's memory can be read, written, and controlled by simply sending datagrams expressing FINS commands to the PC Card Unit's UDP port for FINS communications. There is thus no need to prepare a user program at the C200HX/HG/HE.

Note
1. The PC Card Unit uses the CMCR instruction for sending and receiving FINS commands.
2. Port 9600 is normally used for sending and receiving FINS commands. To change this, refer to *7-5-9 Setting the Port Number*.

3. FINS commands to a PC Card Unit at the same Programmable Controller are not transmitted to the Ethernet.



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.

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 such as arrival confirmation processing to ensure reliability must be programmed into the user application.

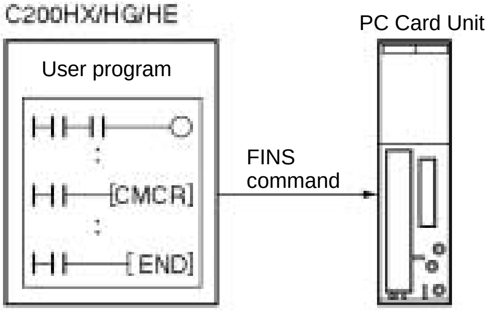
9-2 Using FINS Communications

This subsection outlines the methods for using FINS communications.

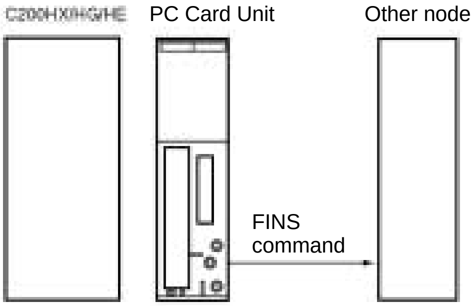
9-2-1 FINS Communications by Ladder Program

The processing flow for executing FINS communications from a ladder program is explained below.

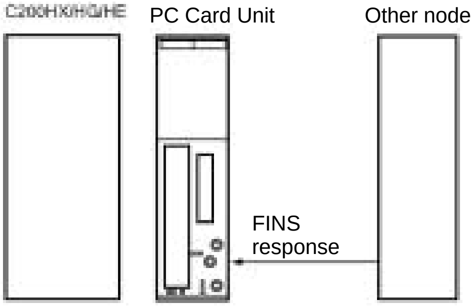
1, 2, 3... 1. The CMCR instruction is used to issue a FINS command.



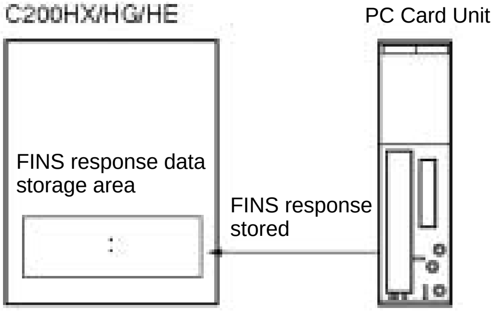
2. The PC Card Unit receives the FINS command and sends it to the specified node.



3. The PC Card Unit receives a FINS response from the other node.



4. The FINS response data that was received by the PC Card Unit is stored in the FINS response data storage area that was specified by the CMCR instruction.



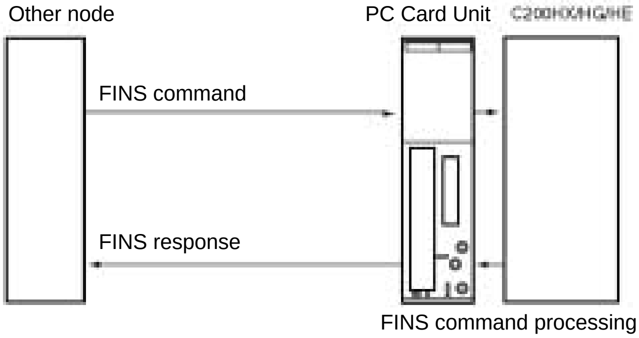
For details on how to use FINS commands and responses, refer to *9-4 Using FINS Commands and Responses*. For an explanation of using socket services with FINS commands, refer to *Section 10 Socket Services*.

Service Time for FINS Communications From a Ladder Program

The service time for FINS communications from a ladder program is the same as for SEND(90) and RECV(98). For details, refer to *8-4 Minimum Transmission Delay Time for SEND/RECV Instructions*. The instruction execution time is the same as for SEND(90).

9-2-2 FINS Communications From Another Node

When the PC Card Unit receives a FINS command from another node, it automatically returns a response as shown in the following illustration.



A ladder program related to FINS command/response processing is not required for the C200HX/HG/HE.

For details regarding the handling of FINS command/response data at other nodes, refer to *9-6 FINS Communications From Computers*.

9-3 Using the CMCR Instruction

With the PC Card Unit, the CMCR instruction is used to issue FINS commands. This subsection explains how to use the CMCR instruction for that purpose. For an explanation of how to use the CMCR for memory card file operations, or for details regarding the basic CMCR format, refer to *5-3 File Operations: CMCR Instruction*.

9-3-1 CMCR Format for FINS Commands

The CMCR instruction's format for issuing FINS commands is as follows:

CMCR	
Control	**C:** Beginning control data storage word
Source	**S**: Beginning command data storage word (see note 1)
Destination	**D:** Beginning response word (see note 2)

Note 1. The data length and FINS command are stored here.
2. The FINS response is stored here.

9-3-2 Control Data

Control data is written in the following format, from the beginning control data word (C).

Word	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit
C+0	Operating level #0 = $0810, operating level #1 = $1810																
C+1	Number of bytes to send																
C+2	Number of bytes to receive																
C+3	0	0	0	0	0	0	0	0	Destination network address								
C+4	Destination node number								Destination unit address								
C+5	*	0	0	0	0	0	0	0	0	0	0	0	Number of retries				
C+6	Response monitor time																

*Response flag (0: Response required; 1: Response not required)

Operating Level Setting to “$0810” specifies operating level #0; setting to “$1810” specifies operating level #1.

Number of Bytes to Send Specifies the number of bytes of data to be transmitted from the FINS command code onwards, within a range of 2 to 2,000 ($2 to 7D0).

Number of Bytes to Receive Specifies the number of bytes of response data (from the FINS command code onwards), within a range of 4 to 2,000 ($4 to 7D0), to be stored beginning with the the response data storage word.

Destination Network Address Specifies the *destination network address* within a range of 0 to 127 ($0 to 7F). Set to 00 when transmitting to a Unit in the local network.

Destination Node Number Specifies, within a range of 0 to 127 ($0 to 7F), the destination node number in the network set by the *destination network address*. To broadcast within that network, set the *destination node number* to FF. Set to 00 when transmitting to a local node in the local network.

Destination Unit Address Sets the device or Unit where the command is to be sent. To send the command to the Programmable Controller, set the *destination unit address* to 00. To send to a Special I/O Unit, set a value ($10 to 1F) with $10 added to the unit number ($0 to F). The unit number of the PC Card Unit is 0 (with a unit address of $10). If the *destination unit address* is set to FE, the command will be sent to the Communications Unit corresponding to the *destination node number*.

Set the network to 0, node to 0, and Unit to $10 when sending to the PC Card Unit in the local node.

Response Request Bit This bit is normally set to OFF (0: Request response). If no response is required, set the bit to ON (1: Don’t request response).

Number of Retries Specifies the number of retries to be performed if a response is not returned following execution of the CMCR instruction. Any number from 0 to 15 ($0 to F) can be set.

Response Monitor Time The response monitor time (i.e., the timeout time) can be set from 0 to 65,535 ($0 to FFFF), in units of 110 ms. If the default setting of “0” is specified, the response monitoring time will be 2.2 s.

The *response monitor time* setting is only valid when the *response request bit* is set to OFF (0: Request response).

Note If more than the *number of bytes to receive* is received, the bytes over the set number will be discarded. If fewer bytes are received, the data will be stored in the specified area and the data in the remainder of that area will be unchanged.

9-3-3 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 07D0 (2 to 2,000 bytes) (after FINS command code)
Number of bytes to receive	$0004 to 07D0 (4 to 2,000 bytes)
Destination network address	$00: To the local network $01 to 7F (1 to 127): To a designated network
Destination node number	$00: To the local node $01 to 7F (1 to 127): To a designated node
Destination unit address	$00: PC $10 to 1F: CPU Bus Unit (unit number: 0 to F, respectively)
Response request bit (see note 2.)	$0 (OFF): Request response $1 (ON): Don’t request response
Number of retries	$0 to F (0 to 15)
Response monitor time	2.2 s (default) $0001 to FFFF: Units of 110 ms

Note
1. A maximum of 1,462 bytes can be sent for broadcasts.
2. In the case of broadcasting, set the Response Request Bit to ON (1: Don't request response).

9-3-4 Command Data

In word S+0, specify the FINS command data length +1. The FINS command is stored in words S+1 onwards. The maximum command data length is 1,000 words.

S+0	Command data length (Number of words: BCD) $1 to $1001
S+1	FINS Command data
S+2	FINS Command data
	• • •
S+1000	FINS Command data

Note If there is an odd number of bytes of data, set the remaining data to 0.

9-3-5 Response Data

The response returned form the transmission destination node is stored. The contents of the FINS response depend on the FINS command. For details regarding the contents of response data, refer to *Section 11 Using FINS Commands and Responses*.

9-3-6 Instruction Status

Status information for executed SEND(90), RECV(98), and CMCR instructions is reflected in the SR Area of the PC as instruction execution flags and instruction response codes.

Structure of Instruction Execution Flags

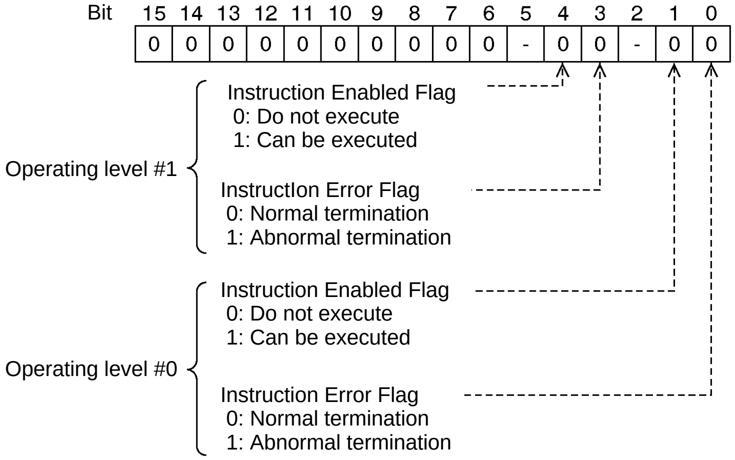
The following two instruction execution flags are used:

Instruction Enabled Flag:
ON ("1") when the SEND(90), RECV(98), or CMCR instruction can be executed.

Instruction Error Flag:
ON ("1") when the SEND(90), RECV(98), or CMCR instruction has terminated abnormally. This status is retained until the next instruction is executed.

If multiple instructions are to be used, always make sure that the Instruction Enabled Flag is ON before executing the next instruction (i.e., perform exclusive control).

C200HX/HG/HE Word SR 252

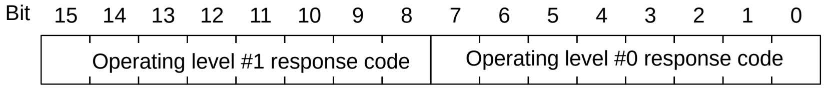


Note Only one instruction can be executed for each operating level at a time. (When a PC Card Unit and a SYSMAC Link Unit are connected, instructions can be executed in two operating levels simultaneously.) If more than one instruction is to be used in the same operating level, perform exclusive control by using the Instruction Enabled Flag, i.e., do not execute SEND(90) or RECV(98) while the Instruction Enabled Flag for that operating level is ON.

Instruction Response Codes

When an instruction terminates, the status is recorded as a response code. The code is retained until the next instruction is executed. It is set to 00 during execution. The instruction response code is one byte and differs from the command/ response end code (2 bytes).

C200HX/HG/HE Word SR 237


Bit
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Operating level #1 response code
Operating level #0 response code

Response Codes

Response code	Item	Explanation
00	Normal completion	Processing terminated normally.
01	Parameter error (cannot send)	Outside specified parameter range. Local node address specification error
02	Cannot send	The system was reset during instruction processing. The local node has not been registered in the network.
05	Response time-out	No response was received within response monitoring time.
06	Response error	Refer to *Response Error* below.

Response Error

The SEND(90) and RECV(98) instructions are converted to FINS commands and are sent to the destination node from the PC Card Unit. The SEND(90) instruction is converted to MEMORY AREA WRITE and the RECV(98) instruction is converted to MEMORY AREA READ. When these SEND(90) and RECV(98) instructions are executed, if the instructions are not completed normally (i.e., the MRES is not 0), a response error is generated. (Refer to 124.)

Possible causes for a response error include the following.

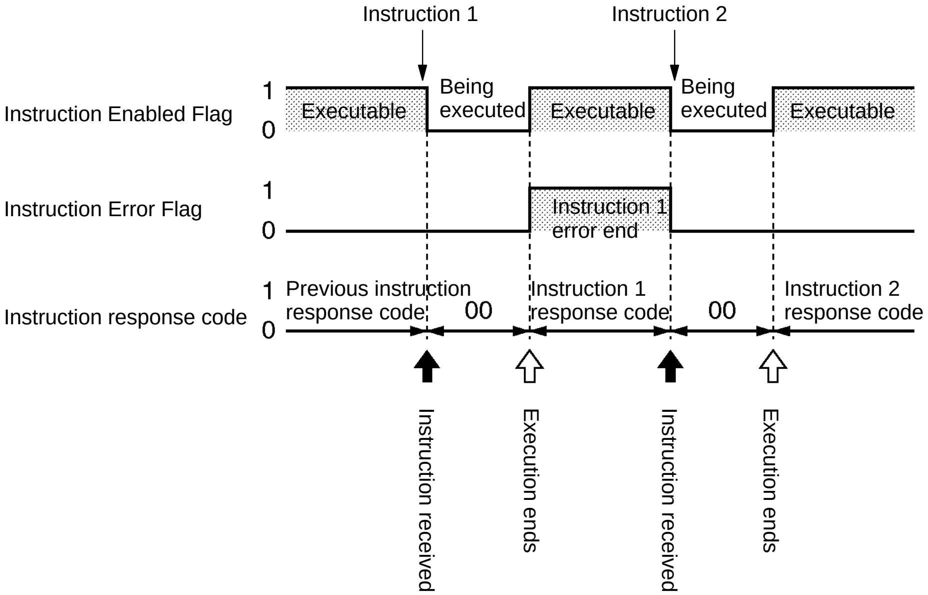
- The destination node is not participating in the network.
- The destination node is busy.

- Processing is being performed at the destination node and so commands cannot be received.
- Communications controller error
- Node address setting error
- PC error

Flag ON/OFF Timing

The Instruction Enabled Flag, Instruction Error Flag, and instruction response code are refreshed as shown below.

Example: When two instructions are executed consecutively and the first instruction causes an error



9-3-7 SEND(90)/RECV(98)/CMCR Data Processing Timing

The table below shows when data for the SEND(90), RECV(98), or CMCR instruction is sent.

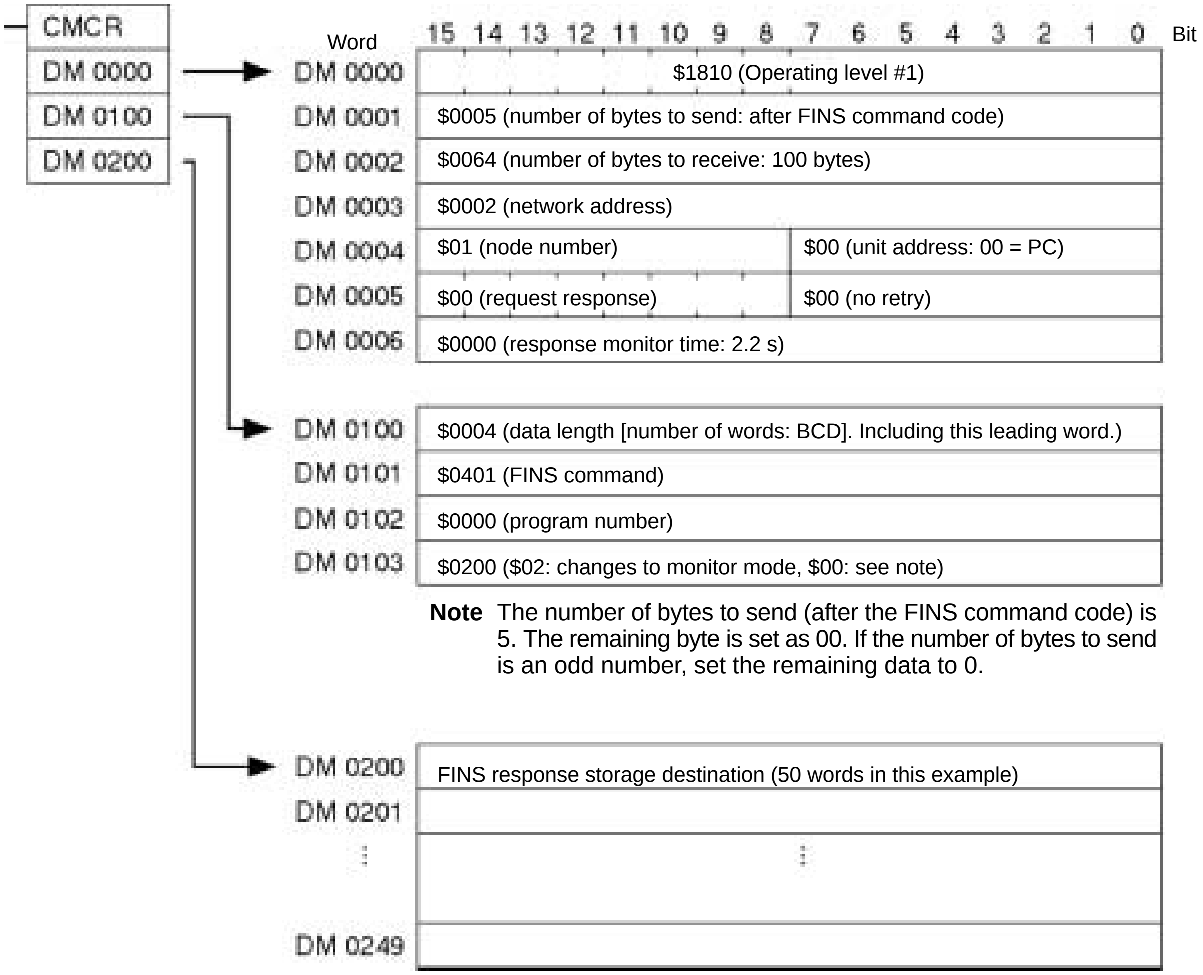
Send data	Send/Receive end processing
All data that has been processed by the time the END instruction is executed is sent.	Processing is performed when the END instruction is executed.

9-3-8 FINS Command Example

The example below shows how to use the CMCR instruction to issue a FINS command. In this example, a RUN (0401) command is issued to a Programmable Controller on an Ethernet network, at the following destination.

• Operating level:	1
• Number of bytes to send:	5
• Number of bytes to receive:	100 (Allow sufficient space.)
• Network address:	2
• Node number:	1
• Unit address:	00 (Programmable Controller)
• Response:	Required
• Number of retries:	0
• Response monitor time:	2.2 s (default: 00)

When the RUN (0401) command is issued to this destination with these conditions, the following data is set in the DM area.



CMCR
DM 0000
DM 0100
DM 0200

Word	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0 Bit
DM 0000	$1810 (Operating level #1)	
DM 0001	$0005 (number of bytes to send: after FINS command code)	
DM 0002	$0064 (number of bytes to receive: 100 bytes)	
DM 0003	$0002 (network address)	
DM 0004	$01 (node number)	$00 (unit address: 00 = PC)
DM 0005	$00 (request response)	$00 (no retry)
DM 0006	$0000 (response monitor time: 2.2 s)	

Word	Contents
DM 0100	$0004 (data length [number of words: BCD]. Including this leading word.)
DM 0101	$0401 (FINS command)
DM 0102	$0000 (program number)
DM 0103	$0200 ($02: changes to monitor mode, $00: see note)

Note The number of bytes to send (after the FINS command code) is 5. The remaining byte is set as 00. If the number of bytes to send is an odd number, set the remaining data to 0.

Word	Contents
DM 0200	FINS response storage destination (50 words in this example)
DM 0201	
⋮	⋮
DM 0249	

9-4 Using FINS Commands and Responses

With the C200HX/HG/HE, the CMCR instruction is used for issuing FINS commands. This subsection explains communications data formats, FINS commands to the C200HX/HG/HE and to the PC Card Unit, and how to specify memory areas.

9-4-1 Command/Response Parameters

All parameters are specified in hexadecimal unless otherwise specified.

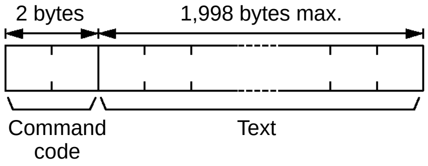
9-4-2 Communications Data Formats

Headers

When FINS commands are issued from a device such as an FA computer, a header must be affixed before the command code. For details regarding headers, refer to *9-6 FINS Communications From Computers*.

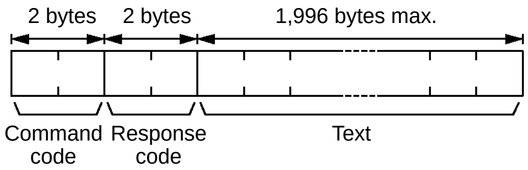
Commands

Commands have the following format.



Responses

Responses have the following format.



Note The maximum size of a command or response depends upon the type of network through which the transmission is relayed. The above limits are for FINS commands and responses on Ethernet networks.

9-4-3 Commands and Responses for C200HX/HG/HE CPUs

Command code		Name	PC mode			Page
			RUN	MONITOR	PROGRAM	
01	01	MEMORY AREA READ	Valid	Valid	Valid	*165*
	02	MEMORY AREA WRITE	Valid	Valid	Valid	*166*
	04	MULTIPLE MEMORY AREA READ	Valid	Valid	Valid	*167*
03	06	PROGRAM AREA READ	Valid	Valid	Valid	*168*
	07	PROGRAM AREA WRITE	Not valid	Not valid	Valid	*169*
04	01	RUN	Valid	Valid	Valid	*169*
	02	STOP	Valid	Valid	Valid	*170*
05	01	CONTROLLER DATA READ	Valid	Valid	Valid	*170*
06	01	CONTROLLER STATUS READ	Valid	Valid	Valid	*171*
07	01	CLOCK READ	Valid	Valid	Valid	*172*
	02	CLOCK WRITE	Not Valid	Valid	Valid	*173*
21	01	ERROR CLEAR	Valid	Valid	Valid	*173*
23	01	FORCED SET/RESET	Not valid	Valid	Valid	*174*
	02	FORCED SET/RESET CANCEL	Not valid	Valid	Valid	*175*
	0A	MULTIPLE FORCED STATUS READ	Valid	Valid	Valid	*175*

9-4-4 PC Card Unit FINS Commands List

The following is a list of the FINS commands supported by the PC Card Unit.

Command Code		Name	Page
05	01	CONTROLLER DATA READ	*177*
08	01	INTERNODE ECHO TEST	*177*
	02	BROADCAST TEST RESULTS READ	*178*
	03	BROADCAST TEST DATA SEND	*178*
21	02	ERROR LOG READ	*179*
	03	ERROR LOG CLEAR	*180*
22	02	SINGLE FILE READ	*180*
	03	SINGLE FILE WRITE	*181*
	05	FILE DELETE	*182*
	07	FILE COPY	*182*
	08	FILE NAME CHANGE	*183*
	12	SINGLE FILE READ WITH COMMAS	*183*
	13	SINGLE FILE WRITE WITH COMMAS	*184*

9-4-5 Response Codes

Response codes are 2-byte codes that indicate the results of command execution. The first byte of the response code indicates the general results of execution and is called the main response code, or MRES. The second byte provides further details on the results of execution and is called the sub-response code, or SRES.

The following tables shows the relationship between the MRES and the execution results. Refer to *Appendix D: Response Codes from the C200HX/HG/HE CPU* for details on all response codes including the SRES and actions for each.

Main Response Codes

MRES	Execution completed	MRES	Execution completed
00	Normal completion	20	Read not possible
01	Local node error	21	Write not possible
02	Destination node error	22	Not executable in current mode
03	Controller error	23	No unit
04	Not executable	24	Start/Stop not possible
05	Routing error	25	Unit error
10	Command format error	26	Command error
11	Parameter error	30	Access right error

9-4-6 Memory Area Designations

The following table gives the addresses to use when reading or writing PC data.

Memory area	Data	Data area address	Address used in communications		Memory area code	No. of bytes
			1st and 2nd bytes	3rd byte		
CIO area	Bit status	00000 to 51115	0000 to 01FF	00 to 0F	00	1
	Word contents	000 to 511		00 to 00	80	2
LR area	Bit status	LR 0000 to LR 6315	03E8 to 0427	00 to 0F	00	1
	Word contents	LR 00 to LR 63		00 to 00	80	2
HR area	Bit status	HR 0000 to HR 9915	0428 to 048B	00 to 0F	00	1
	Word contents	HR 00 to HR 99		00 to 00	80	2
AR area	Bit status	AR 0000 to AR 2715	048C to 04A7	00 to 0F	00	1
	Word contents	AR 00 to AR 27		00 to 00	80	2
Timer/Counter Area	Completion Flag status	TIM 000 to TIM 511 CNT 000 to CNT 511	0000 to 01FF	00 to 00	01	1
	PV	TIM 000 to TIM 511 CNT 000 to CNT 511		00 to 00	81	2
DM Area	Word contents	DM 0000 to DM 9999	0000 to 270F	00 to 00	82	2
Expansion DM Area	Word contents	EM 0000 to EM 6143	0000 to 17FF	00 to 00	90 to 98	2

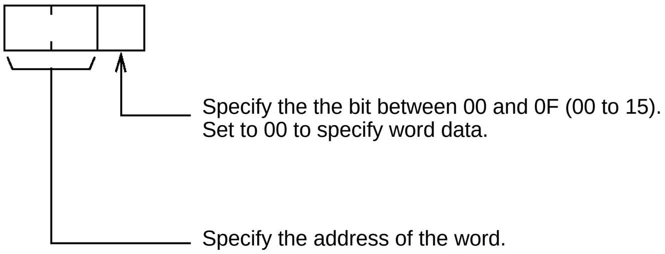
Note 1. The size of the memory area varies depending on the PC. Refer to the PC's operation manual for details on the size limits of the memory areas.

2. The meanings of the memory area codes for the Expansion DM area are shown below.

Memory area code	Meaning
98	Current bank
90 to 97	Banks 0 to 7
A8 to AF	Banks 8 to 15

Word/Bit Addresses

Each word/bit address specifies a specific bit or word. The rightmost two digits of the address specify bit 00 to 15 (or 00 if not required), and leftmost four digits specify the word address.


Specify the the bit between 00 and 0F (00 to 15).
Set to 00 to specify word data.
Specify the address of the word.

To obtain the corresponding address of the desired word or bit, add the data area word address (hexadecimal) to the first address of the range of addresses used in communications (see table on previous page). For example, the address for word AR 13 is computed as follows:

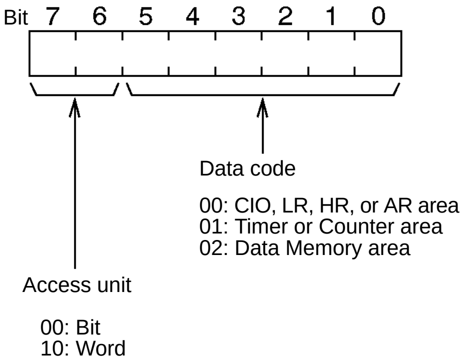
First address for AR area:	048C
048C + 0D (13 in decimal):	0499

The word address for AR 13 would thus be 049900 (the memory area code would specify this as a word) and the address of bit 12 in AR 13 would be 04990C.

The unit of access (bit or word) and the data code are specified as shown in the following illustration.

Memory Area Codes

Memory area codes are the following configuration of access sizes and area codes.



Number of Bytes

A specific number of bytes is required for each element specified for either MEMORY AREA READ or MEMORY AREA WRITE. This will vary depending on the access unit of the memory area code, as indicated below.

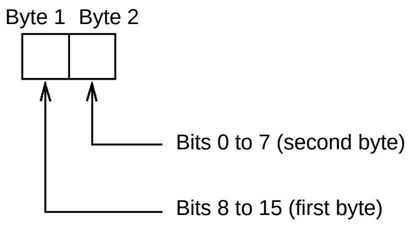
Bit Access;	1 byte per element
Word Access;	2 bytes per element

Data Configuration

The configuration of the various types of data that can be read or written is shown below. The number of bytes required for each type of data is also given.

- **Flag or Bit Status (One Byte)** **00:** Bit is OFF (0)
  01: Bit is ON (1)

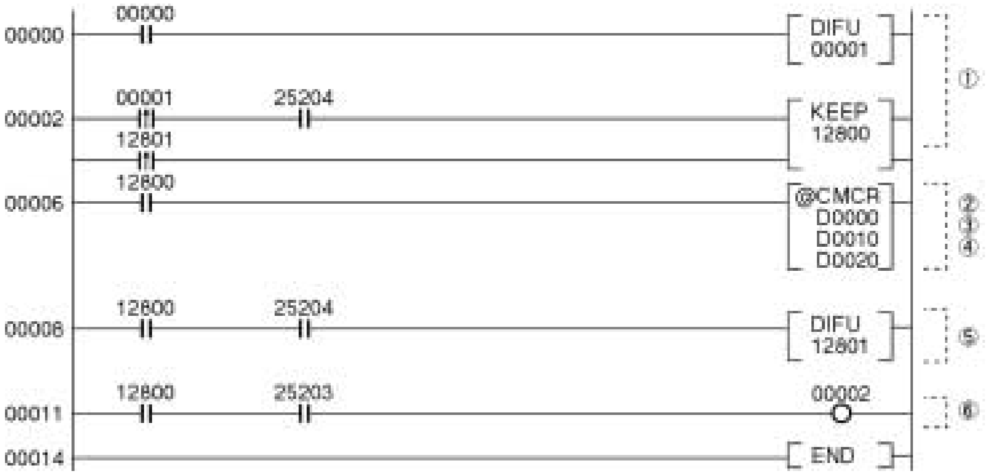
- **Word Contents or PV (Two Bytes)**



9-5 Sample Program

This subsection provides a sample program for executing FINS commands and responses.

9-5-1 Program Example



1, 2, 3...

1. The CMCR instruction is executed when the execution condition 00000 turns ON while the Instruction Enabled Flag (bit 25204) is ON. Bit 12800 stays ON from the time CMCR execution is started until it is completed.
2. The CMCR instruction's control data and command data are set in advance in the C200HX/HG/HE's memory area.

Control Data

Word	Contents	Meaning
D0000	$1810	Operating level #1
D0001	$0002	Number of bytes to send: 2 bytes
D0002	$0064	Number of bytes to receive: 100
D0003	$0001	Destination network address: $01
D0004	$0100	Destination node number: $01 Destination unit address: $00 (PC)
D0005	$0000	Request response Number of retries: 0 (No retries)
D0006	$0000	Response monitor time: 2.2 s ($0000 is default)

Command Data

Word	Contents	Meaning
D0010	$0002	Data length: 2 words (BCD)
D0011	$0701	FINS command: CLOCK READ

3. Specifies the FINS response storage destination. In this example, the FINS response data is stored from DM 0020 onwards.

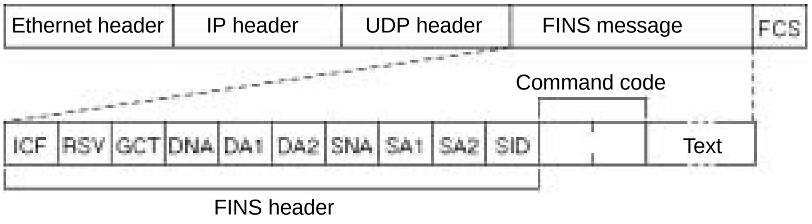
4. Executes the CMCR instruction.
5. CMCR instruction execution is completed when bit 12801 is turned ON while the Instruction Enabled Flag (bit 25204) is ON.
6. Turns ON if an error occurs during communications execution.

9-6 FINS Communications From Computers

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

9-6-1 Frame Format

With the FINS communications service, datagrams in the Ethernet frame format shown below are sent and received.



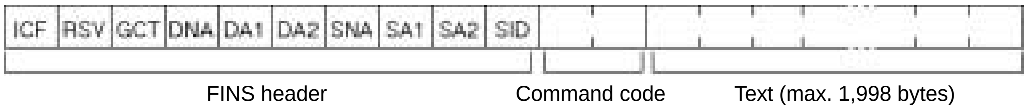
When a FINS command from a computer is used, the destination's IP address and UDP port number must be set.

9-6-2 FINS Command and Response Formats

This subsection explains the data formats for FINS commands and responses.

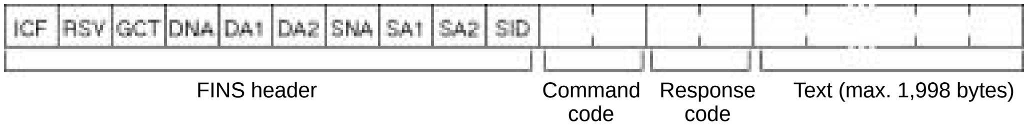
FINS Commands

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



FINS 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.

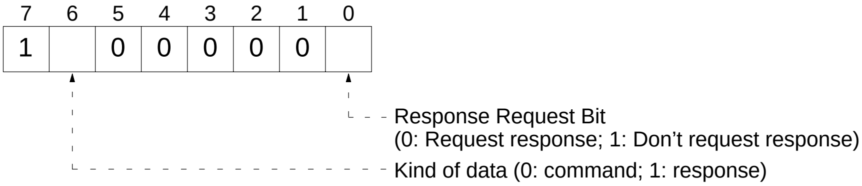


An error response will be returned if the IP-FINS node address conversion table is not correctly set.

9-6-3 FINS Header Information

This subsection explains the FINS header contents and settings used for FINS commands and responses.

ICF (Information Control Field)
Information such as the data classification, response request, communications between networks, and so on, is set in the ICF. The configuration is as follows:


7 6 5 4 3 2 1 0
1 0 0 0 0 0
Response Request Bit
(0: Request response; 1: Don't request response)
Kind of data (0: command; 1: response)

RSV Reserved (system reserved area)

Set to $00.

GCNT (Permissible Number of Gateways)

Set to $02.

DNA (Destination Network Address)
Specifies the network address of the destination node.

$01 to 7F: (1 to 127)

DA1 (Destination Node Number)
Specifies the destination node number for sending the command. It is used for FINS, and is different from the IP address used for Ethernet.

$00:	Local PC
$01 to 7F (1 to 127):	Node in Ethernet

DA2 (Destination Unit Address)
Specifies as an absolute address a unit number at the destination node. (Example: CPU Special I/O Unit #0 is $10.)

$00:	PC (CPU)
$10 to $1F:	CPU Special I/O Unit (PC Card Unit is $10.)
$FE:	Communications Unit

SNA (Source Network Address)

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

SA1 (Source Node Number)

$00:	Local PC
$01 to 7F (1 to 127):	Node in Ethernet

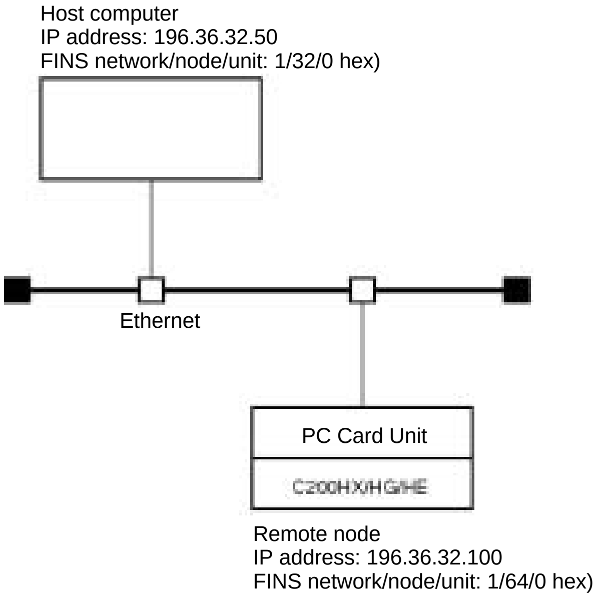
SA2 (Source Node Address)

$00:	PC (CPU)
$10 to $1F:	CPU Special I/O Unit (PC Card Unit is $10.)
$FE:	Communications 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.

9-6-4 Designating Remote Addresses

UDP sockets are used when sending FINS commands from a host computer to the C200/HX/HG/HE. This subsection provides an example of addressing remote PCs from the host computer for communications.


Host computer
IP address: 196.36.32.50
FINS network/node/unit: 1/32/0 hex)
Ethernet
PC Card Unit
C200HX/HG/HE
Remote node
IP address: 196.36.32.100
FINS network/node/unit: 1/64/0 hex)

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

Item	Setting	Remarks
Destination IP Address	196.36.32.100	PC Card Unit address
UDP port number	9600 (FINS UDP default port number)	PC Card Unit’s UDP port number
FINS addresses	DNA, DA1, DA2 = 1, 64, 0 (hex)	C200HX/HG/HE
	SNA, SA1, SA2 = 1, 32, 0 (hex)	Host computer

SECTION 10
Socket Services

This section describes sockets (an interface for directly using TCP and UDP functions from the user program) and explains how to use socket services.

10-1 About Socket Services

This subsection provides an outline of PC Card Unit socket services.

10-1-1 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, thereby enabling communications in any protocol with FA computer other than OMRON PCs, with UNIX workstations, and so on.

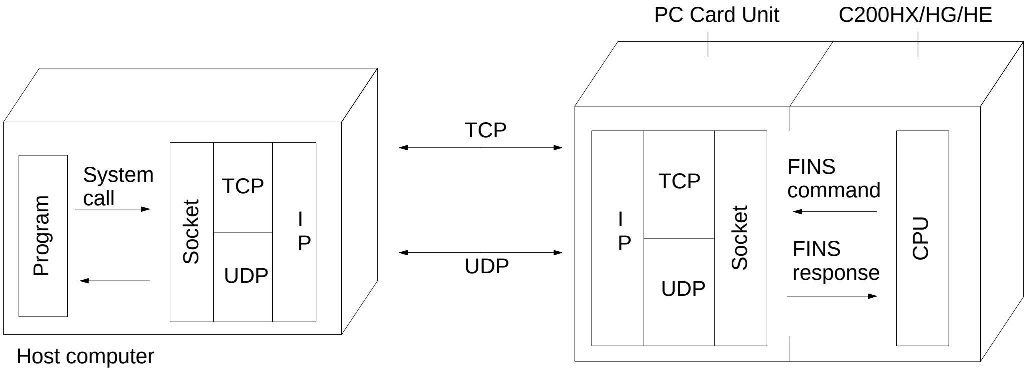
Socket services are supplied for most host computers (or some personal computers) as a C-language interface library. This interface library allows user programs to communicate using TCP and UDP.

The socket interface is supported for UNIX workstations in the form of system calls.

Socket services are achieved for C200HX/HG/HE PCs by sending FINS commands from the user program to a PC Card Unit. Refer to *Section 9 FINS Commands*, *10-2 Using Socket Services*, *10-3 Sample Programs for TCP and UDP Communications*, and *Section 11 FINS Commands and Responses*.

10-1-2 Socket Operation

A socket operates as shown in the following diagram when communicating between an FA computer and a PC Card Unit. Data from the computer or C200HX/HG/HE PC passes through the socket and is communicated using the TCP or UDP protocol.



10-1-3 Differences between TCP and UDP

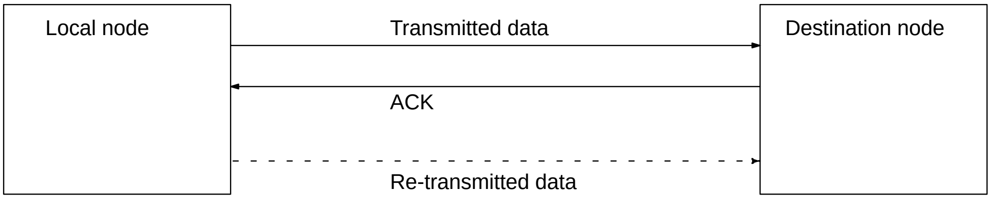
There are differences in the socket services between TCP and UDP.

TCP Communications

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

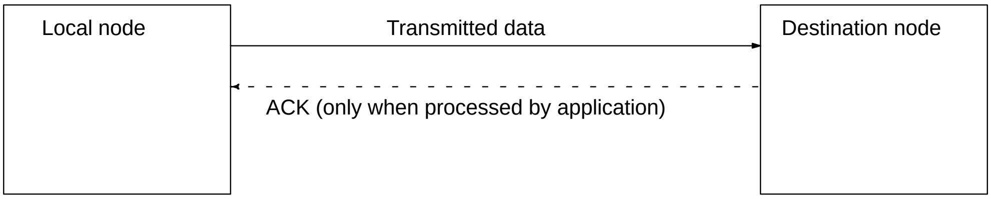
1, 2, 3...

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.

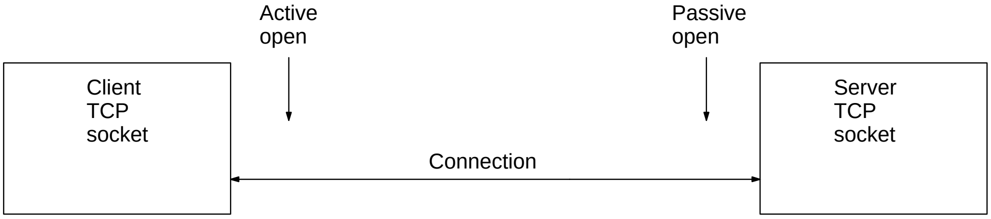


10-1-4 Opening TCP Sockets

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.



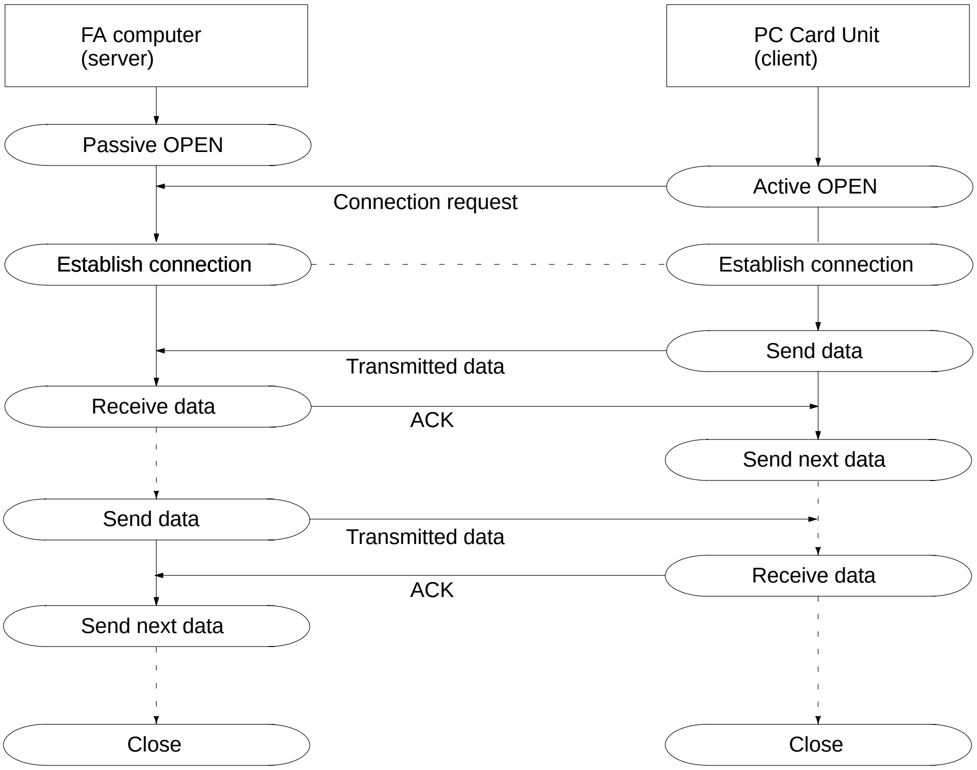
Client and Server

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. For the PC Card Unit, four sockets are provided for UDP and TCP together. Communications with multiple UDP sockets is possible, because connections are not set for UDP sockets.

TCP Communications Procedure

The communications procedure is shown below for communications between an FA computer and PC Card Unit using a TCP socket.



10-1-5 Socket Services 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 from 1024 to 65535 should be used for socket services. The PC Card Unit does not support port #0.

For FINS communications, the 9600 port is used as the default. Also, some port numbers over 1024 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 in /etc/services.

10-1-6 Fragmentation of Transmitted Data

The PC Card 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.

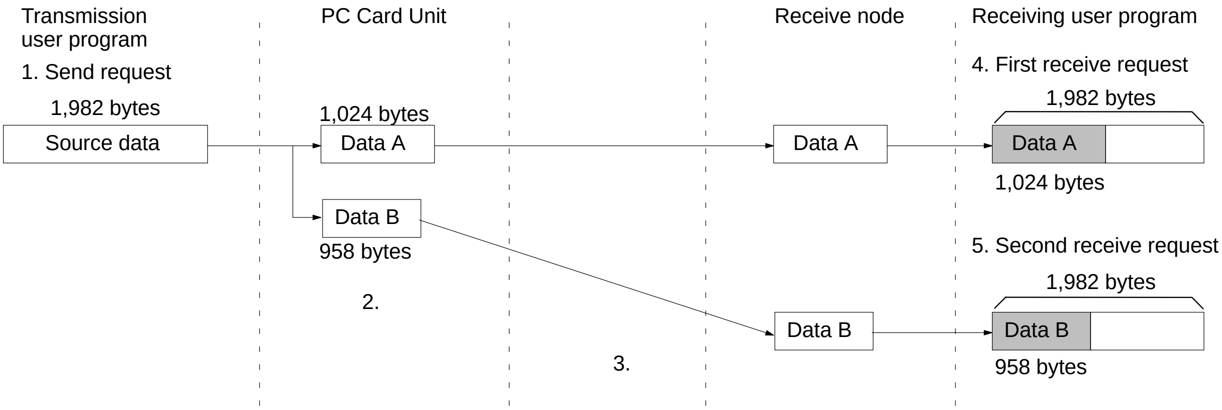
Cautions when Using TCP

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 PC Card 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.

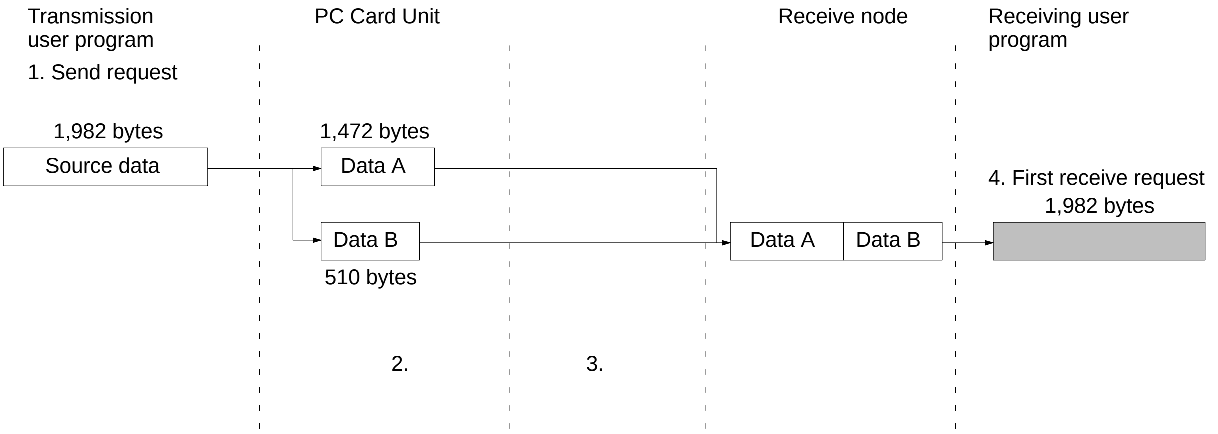
When making the receive request, it is not necessary to specify the same data length as the transmitted data length. For example, if the length setting is shorter than the actual length of the data, all the data can be received by repeating the receive requests.

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 PC Card 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.

Note If the data length in the receive data request is erroneously set smaller than the actual length of the data, all received data exceeding the specified data length will be discarded.

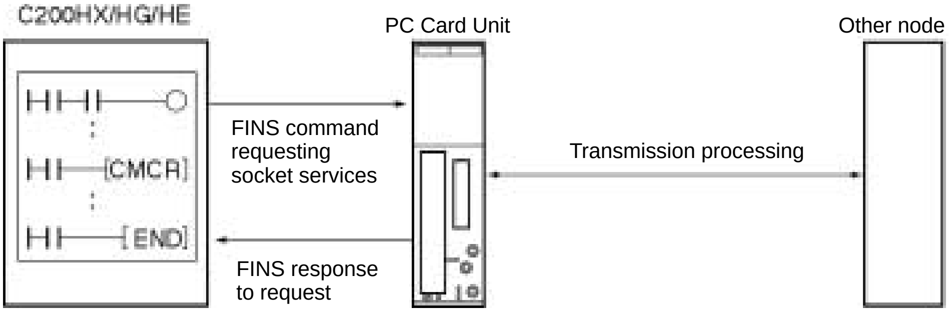
10-2 Using Socket Services

With the C200HX/HG/HE, socket services are controlled by FINS commands to the PC Card Unit. For details regarding FINS commands and responses, refer to *Section 9 FINS Commands*.

10-2-1 FINS Commands Used With Socket Services

Socket services are executed by having the CMCR instruction executed in the C200HX/HG/HE program to issue a FINS command to the PC Card Unit requesting socket services.

Set the FINS node address as follows: network to 0, node to 0, and Unit to $10.



10-2-2 Procedure for Using Socket Services

The PC Card Unit has four sockets that can be used for TCP or UDP. Open, close, send, and receive processes are available for communications using 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.

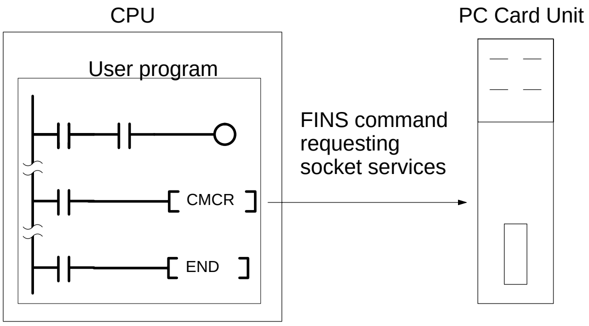
Send
Sends data from a specified open socket.

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

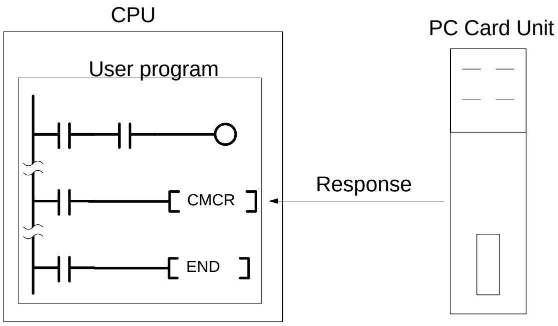
These processes are carried out by using the CMCR instruction to send FINS commands to the PC Card Unit. The process from sending a request for processing to completion is shown in the following illustrations.

1, 2, 3...

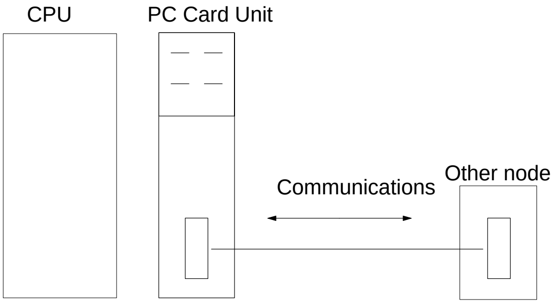
1. Use CMCR to issue a socket service request command (MRC: 27) to the PC Card Unit.



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

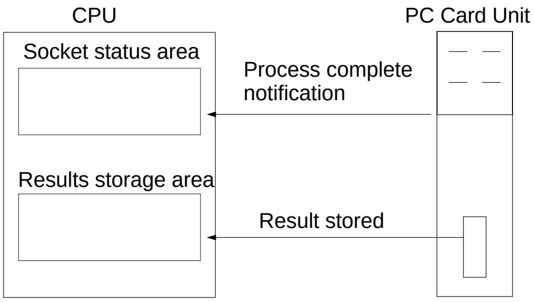


3. The PC Card Unit starts the process requested by the parameters in the socket service request command.



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.

   The socket status area is in AR words 08 to 15. The result storage area is specified by the FINS command.

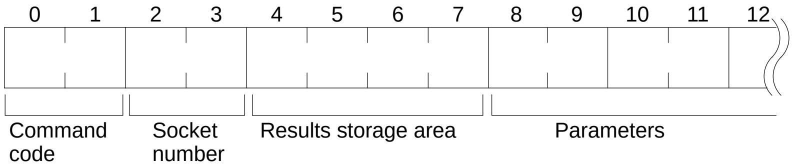


Socket Service Request Commands

Command code		Name	Description	Page
MRC	SRC			
27	01	UDP OPEN REQUEST	Opens the UDP socket.	*187*
	02	UDP RECEIVE REQUEST	Receives data at the UDP socket.	*188*
	03	UDP SEND REQUEST	Sends data from the UDP socket.	*189*
	04	UDP CLOSE REQUEST	Closes the UDP socket and ends communications.	*190*
	10	TCP OPEN REQUEST (PASSIVE)	Opens a TCP socket and waits for connection to another node.	*191*
	11	TCP OPEN REQUEST (ACTIVE)	Opens a TCP socket and connects to another node.	*192*
	12	TCP RECEIVE REQUEST	Receives data at the TCP socket.	*193*
	13	TCP SEND REQUEST	Sends data from the TCP socket.	*194*
	14	TCP CLOSE REQUEST	Closes the TCP socket and ends communications.	*195*

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 4.

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

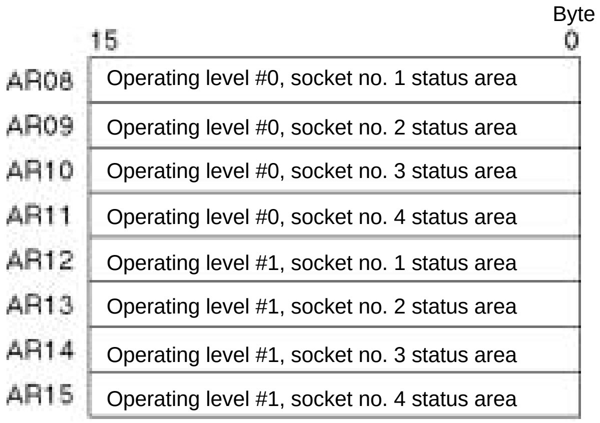
Parameters
Specifies the parameters defined for the command code.

Refer to *Section 11 FINS Commands and Responses* for details about commands.

Socket Status Area

Each PC Card Unit has four sockets that can be used for TCP or UDP. The status of each of these sockets can be confirmed from the by AR bits.

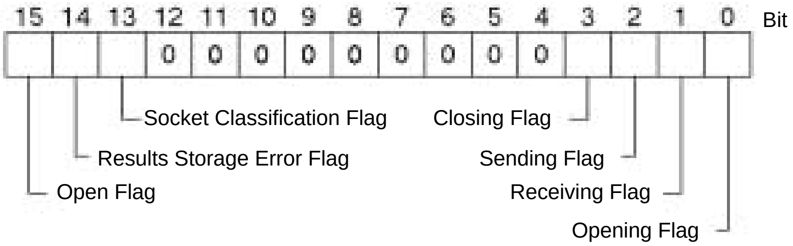
AR Words 08 to 15



AR08	Operating level #0, socket no. 1 status area
AR09	Operating level #0, socket no. 2 status area
AR10	Operating level #0, socket no. 3 status area
AR11	Operating level #0, socket no. 4 status area
AR12	Operating level #1, socket no. 1 status area
AR13	Operating level #1, socket no. 2 status area
AR14	Operating level #1, socket no. 3 status area
AR15	Operating level #1, socket no. 4 status area

Bit Configuration

The meaning of individual bits in each status word is shown in the following diagram.

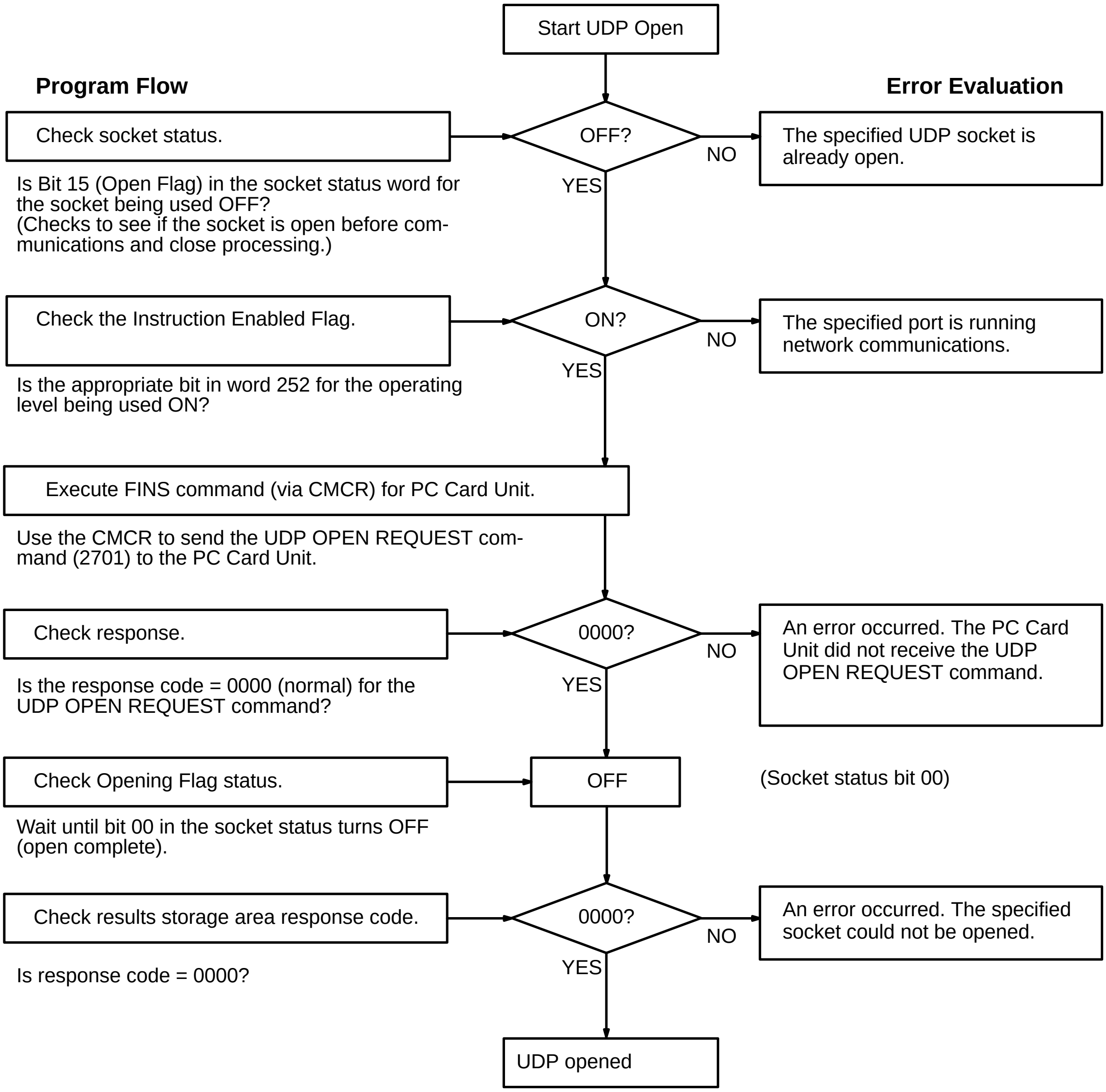

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Bit
0 0 0 0 0 0 0 0 0
Socket Classification Flag
Results Storage Error Flag
Open Flag
Closing Flag
Sending Flag
Receiving Flag
Opening Flag

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 12	Not used			
Bit 13	Socket Classification Flag	1	TCP socket	Indicates that the open socket is a TCP socket. (This has no meaning if the Open Flag is “0.”)
		0	UDP socket	Indicates that the open socket is a UDP socket. (This has no meaning if the Open Flag is “0.”)
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 a PC Card 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.

10-2-3 Socket Services and Socket Status

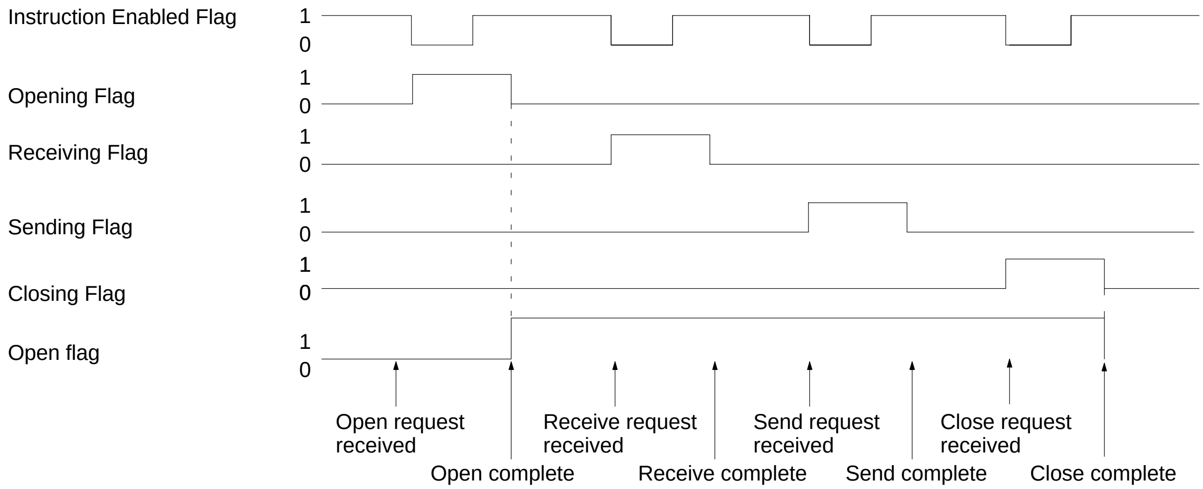
When using socket services, it is important to consider the timing of the status changes in the socket status area. The diagram below shows a program flowchart for opening UDP.

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.


Start UDP Open
Program Flow
Error Evaluation
Check socket status.
OFF?
NO
YES
The specified UDP socket is already open.
Is Bit 15 (Open Flag) in the socket status word for the socket being used OFF?
(Checks to see if the socket is open before communications and close processing.)
Check the Instruction Enabled Flag.
ON?
NO
YES
The specified port is running network communications.
Is the appropriate bit in word 252 for the operating level being used ON?
Execute FINS command (via CMCR) for PC Card Unit.
Use the CMCR to send the UDP OPEN REQUEST command (2701) to the PC Card Unit.
Check response.
0000?
NO
YES
An error occurred. The PC Card Unit did not receive the UDP OPEN REQUEST command.
Is the response code = 0000 (normal) for the UDP OPEN REQUEST command?
Check Opening Flag status.
OFF
(Socket status bit 00)
Wait until bit 00 in the socket status turns OFF (open complete).
Check results storage area response code.
0000?
NO
YES
An error occurred. The specified socket could not be opened.
Is response code = 0000?
UDP opened

10-2-4 Communications Timing Chart

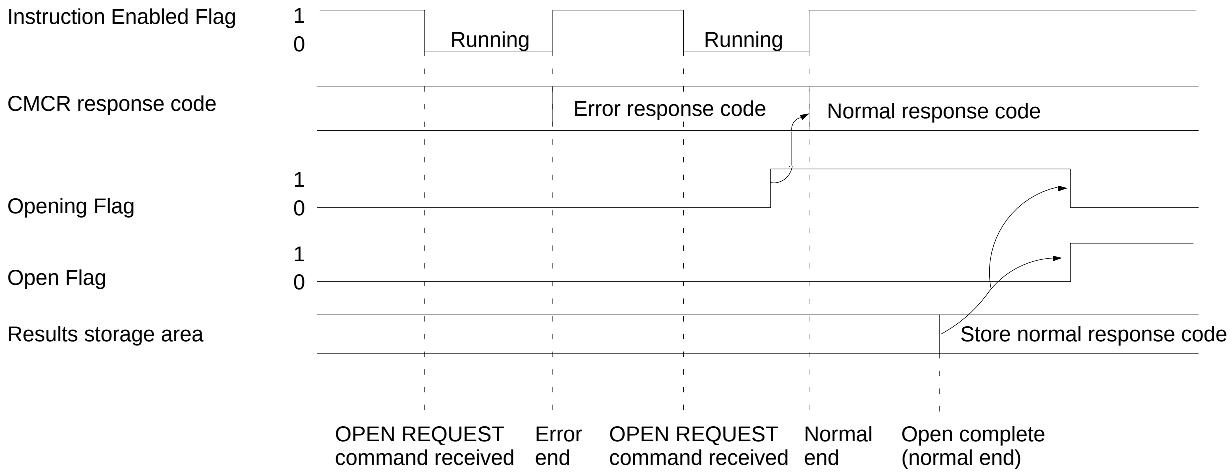
The timing of the status changes of the bits in the socket status area and the Instruction Enabled Flag is shown in the following diagram.



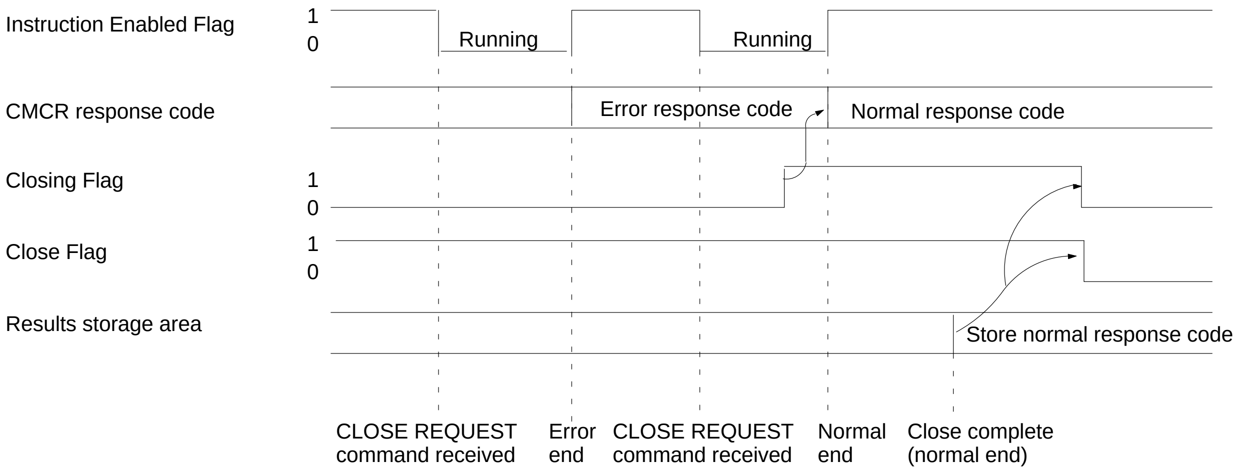
10-2-5 Socket Service Timing Chart

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

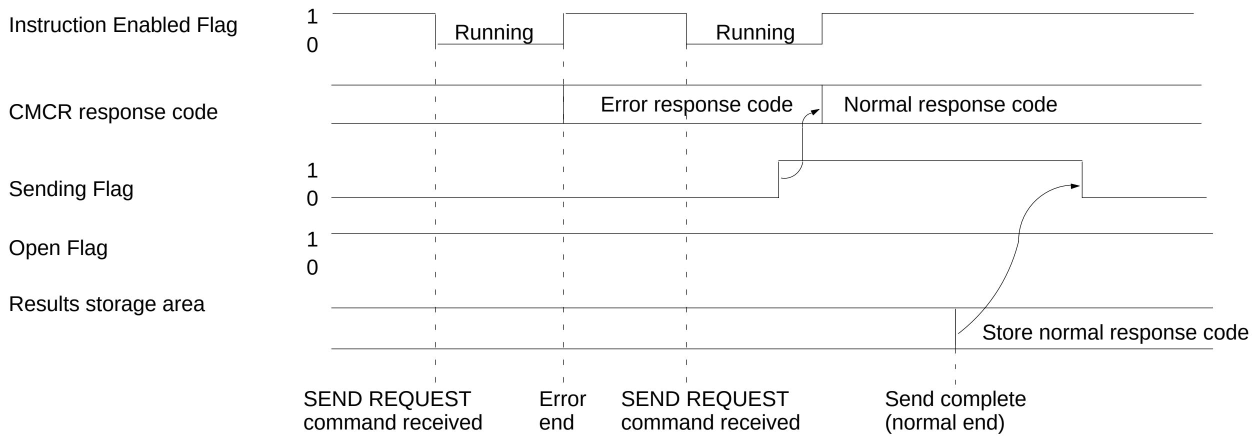
OPEN REQUEST



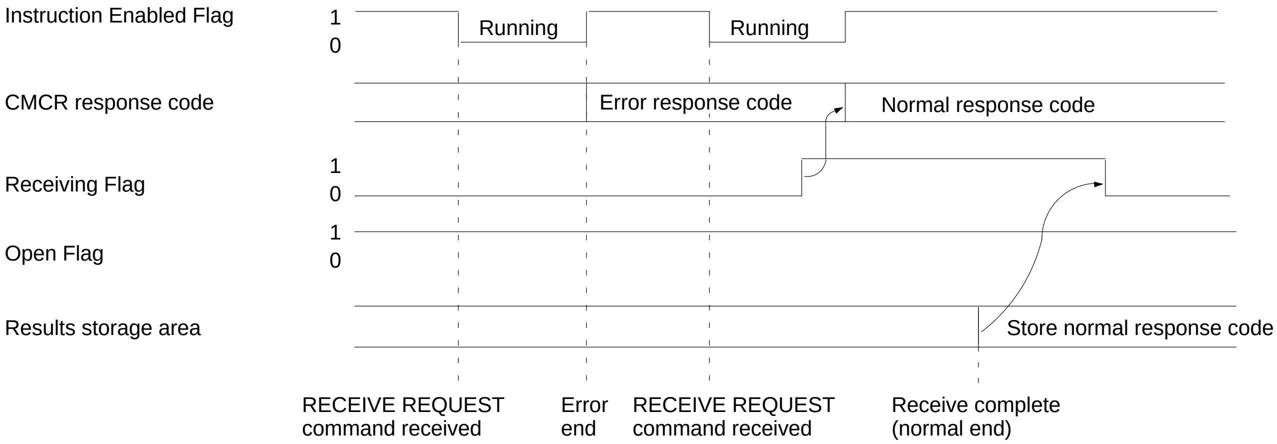
CLOSE REQUEST



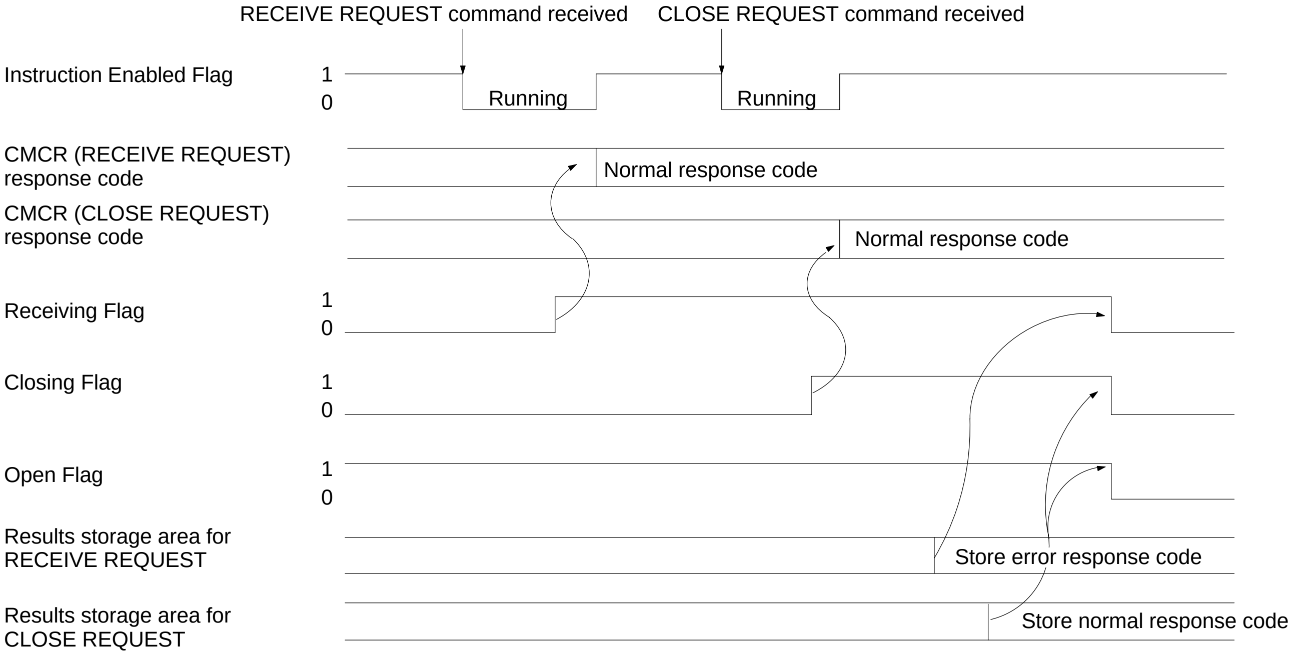
SEND REQUEST



RECEIVE REQUEST



CLOSE REQUEST during RECEIVE REQUEST


RECEIVE REQUEST command received
CLOSE REQUEST command received
Instruction Enabled Flag
1
0
Running
Running
CMCR (RECEIVE REQUEST) response code
Normal response code
CMCR (CLOSE REQUEST) response code
Normal response code
Receiving Flag
1
0
Closing Flag
1
0
Open Flag
1
0
Results storage area for RECEIVE REQUEST
Store error response code
Results storage area for CLOSE REQUEST
Store normal response code

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.

10-2-6 Precautions in Using Socket Services

UDP and TCP Socket Services

- If a short response monitor time is specified in the CMCR control data and the PC Card 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 CMCR.
- If the Results Storage Error Flag turns ON in socket status, it indicates that the specified results storage area does not exist in the C200HX/HG/HE. Correct the user program.
- Communications processing may be slowed down by multiple simultaneous usage of PC Card Unit functions, or by 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 PC Card Unit internally provides 64 bytes of receive buffers 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

- With the UDP socket, data can be broadcasted to all nodes of the network simultaneously by setting a broadcast address for the destination node address. 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

- 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 areas 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 = 0081 (specified socket closed during transmission)
- After closing the port of a connected TCP/IP socket, the port cannot be used within 60 seconds of 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. A connection will not be established for an actively opened socket even if the active open command is executed from another socket. Moreover, you cannot actively open multiple connections to a socket passively opened at the PC Card Unit.

10-3 Sample Programs for TCP and UDP Communications

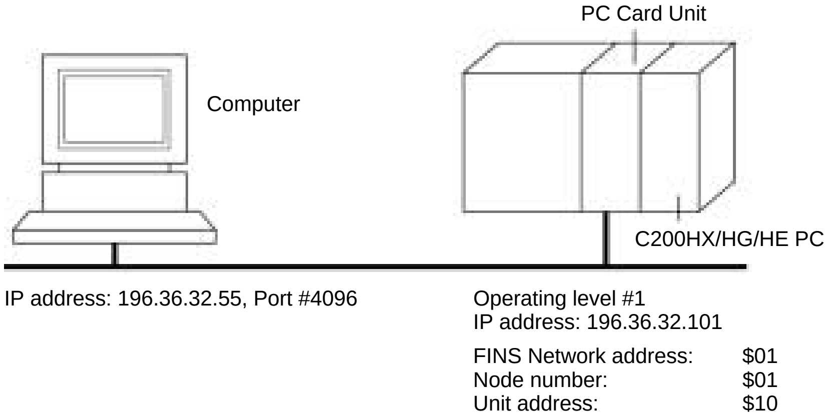
This subsection provides sample programs for TCP/IP communications and UDP/IP communications.

10-3-1 Ladder Programming for TCP/IP Communications

The following program example sends and receives 100 bytes of data between the PC Card Unit and the 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.

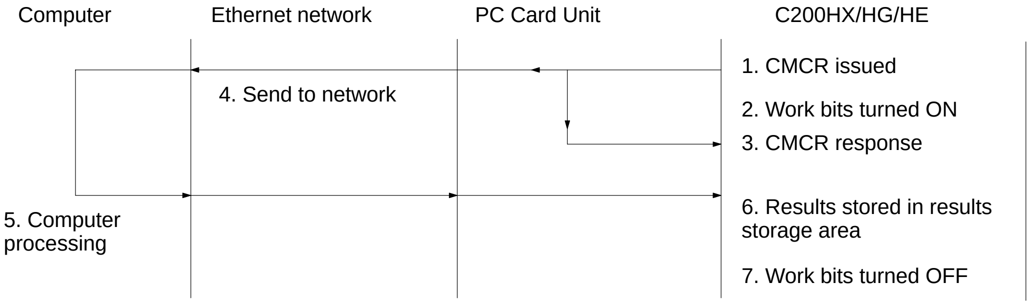
System Configuration

The system configuration for the program example and the PC Card Unit system setup are shown below. To establish a TCP connection in this example, the PC Card Unit is passively opened and the computer is actively opened.


PC Card Unit
Computer
C200HX/HG/HE PC
IP address: 196.36.32.55, Port #4096
Operating level #1
IP address: 196.36.32.101
FINS Network address: $01
Node number: $01
Unit address: $10

CMCR Data Flow to the PC Card Unit

The data flow for sending CMCR instructions to the PC Card Unit is shown in the following diagram.



Note These work bits (CIO 00000 to CIO 00003) are set by the program to control CMCR execution and are not system flags, such as the Instruction Enabled Flag (25204).

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 CMCR control data, command data, and results storage.

Example

The following diagrams indicate that DM 0000 is the first word used in CMCR for a TCP OPEN REQUEST and it contains 1810_{hex}, that DM 00001 contains 0012_{hex}, that CIO 00000 is used to control TCP open processing, and that CIO 0001 is used to control TCP close processing.

How to Read Tables

DM Area

	0	1	2	3	4	5	6	7	8	9
DM 0000	CMCR control data for TCP OPEN REQUEST (PASSIVE)									
	$1810	$0012								

CIO Area

	15 to 8		7	6	5	4	3	2	1	0
CIO 000									TCP close bit	TCP open bit

Memory Map for Sample Programs

DM Area

	0	1	2	3	4	5	6	7	8	9
DM 0000	CMCR control data for TCP OPEN REQUEST (PASSIVE)									
	$1810	$0012	$0004	$0001	$0110	$0000	$0032	—	—	—

CMCR timeout value

Number of retries

PC Card Unit address (Match environment of PC being used.)
Network address: $01
Node number: $01
Unit address: $10

Number of response bytes to receive: 4 bytes

Number of bytes to send: 18

Operating level #1 (or $0810 of operating level #0)

	0	1	2	3	4	5	6	7	8	9
DM 0010	CMCR control data for TCP CLOSE REQUEST									
	$1810	$0008	$0004	$0001	$0110	$0000	$0032	—	—	—

Number of response bytes to receive: 4 bytes

Number of bytes to send: 8 bytes

	0	1	2	3	4	5	6	7	8	9
DM 0020	CMCR control data for TCP SEND REQUEST									
	$1810	$006E	$0004	$0001	$0110	$0000	$0032	—	—	—

Number of response bytes to receive: 4 bytes

Number of bytes to send: 110 bytes
Command format = 10 bytes + 100 bytes transmission data

	0	1	2	3	4	5	6	7	8	9
DM 0030	CMCR control data for TCP RECEIVE REQUEST									
	$1810	$000C	$0004	$0001	$0110	$0000	$0032	—	—	—

Number of response bytes to receive: 4 bytes

Number of bytes to send: 12 bytes
Number of bytes received specified in command data.

	0	1	2	3	4	5	6	7	8	9
DM 1000	TCP OPEN REQUEST (PASSIVE) command data									
	$0010	$2710	$0001	$8203	$FC00	$1000	$0000	$C424	$2037	$0000

Remote port number: Not specified
Computer IP address: 196.36.32.55
($C4, $24, $20, $37)
Timeout value: Not set
Local port number: set to 4096 ($1000)
Results storage area: set to DM 1020 ($03FC)
TCP socket number (PC Card Unit socket number): set to 1
Command code
Transmission data length: 10 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 1010	TCP OPEN REQUEST (PASSIVE) response									
	$2710	Response code	—	—	—	—	—	—	—	—

Stores the response after CMCR command execution.

	0	1	2	3	4	5	6	7	8	9
DM 1020	TCP OPEN REQUEST (PASSIVE) results storage area									
	Response code	Remote IP address		Remote TCP port number	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
DM 1030	TCP CLOSE REQUEST command data									
	$0005	$2714	$0001	$8204	$1A00	—	—	—	—	—

Results storage area: set to DM 1050 ($041A)
TCP socket number to close: set to 1
Command code
Transmission data length: 5 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 1040	TCP CLOSE REQUEST response									
	$2714	Response code	—	—	—	—	—	—	—	—
DM 1050	TCP CLOSE REQUEST results storage area									
	Response code	—	—	—	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
DM 2000	TCP SEND REQUEST command data									
	$0056	$2713	$0001	$820B	$C200	$0064	Send data: 100 bytes			

Number of bytes to send: set to 100 bytes ($0064)
Results storage area: set to DM 3010 ($0BC2)
Socket number to use: set to 1
Command code
Transmission data length: 56 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 3000	TCP SEND REQUEST response									
	$2713	Response code	——	——	——	——	——	——	——	——
DM3010	TCP SEND REQUEST results storage area									
	Response code	No. of bytes sent	——	——	——	——	——	——	——	——

	0	1	2	3	4	5	6	7	8	9
DM 4000	TCP RECEIVE REQUEST command data									
	$0007	$2712	$0001	$820F	$B400	$0064	$0000	——	——	——

Timeout value: Not set

No. of bytes to receive: 100 bytes ($0064)

Results storage area: Set to DM 4020 ($0FB4)

Socket number to use: set to 1

Command code

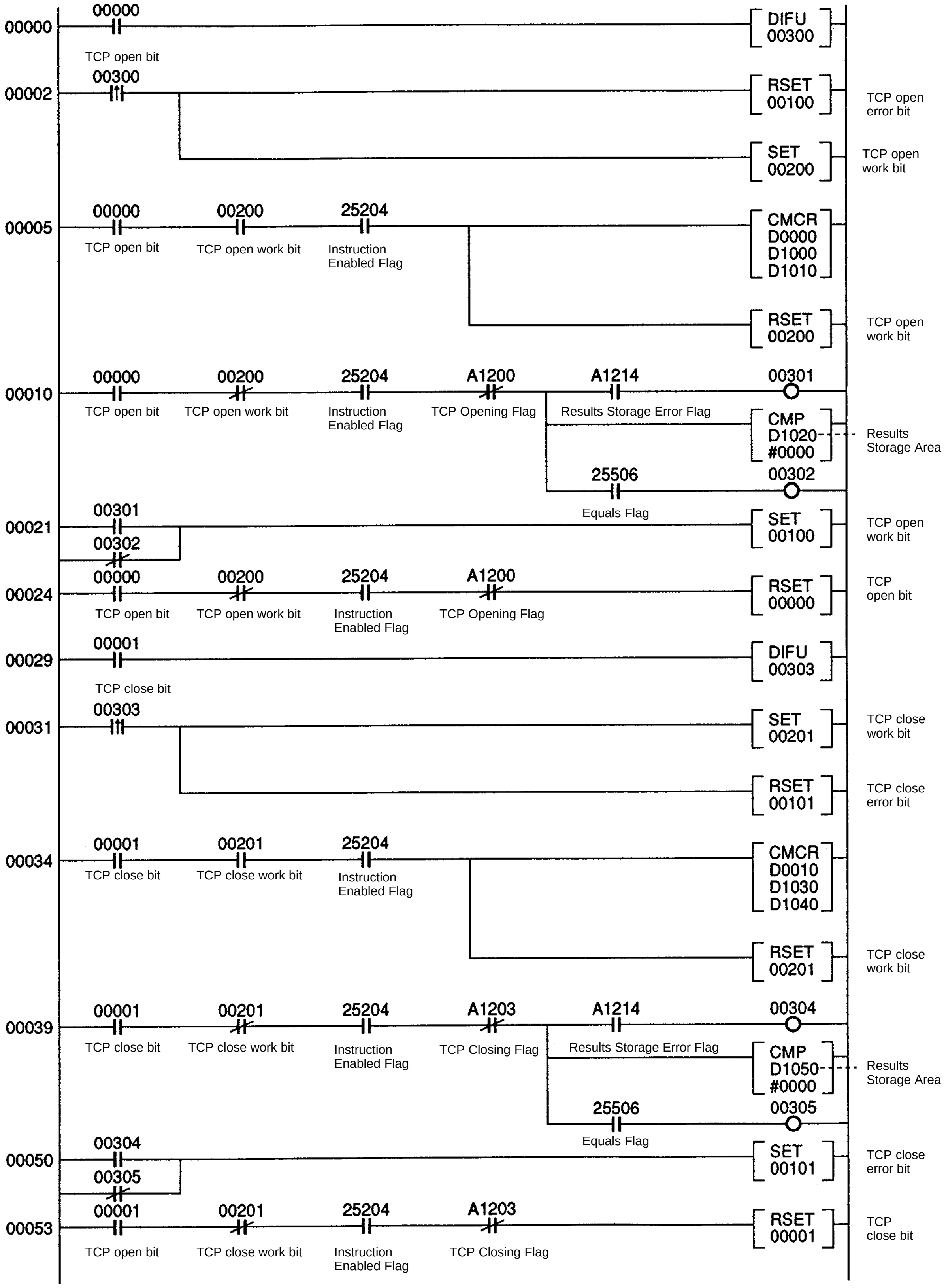
Transmission data length: 7 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 4010	TCP RECEIVE REQUEST response									
	$2712	Response code	——	——	——	——	——	——	——	——
DM 4020	TCP RECEIVE REQUEST results storage area									
	Response code	No. of bytes received	Send data: 100 bytes							

CIO Area

	15 to 8		7	6	5	4	3	2	1	0
CIO 000							TCP receive bit	TCP send bit	TCP close bit	TCP open bit
CIO 001							TCP receive error bit	TCP send error bit	TCP close error bit	TCP open error bit
CIO 002							TCP receive work bit	TCP send work bit	TCP close work bit	TCP open work bit

Program Example 1



Explanation of Ladder Program

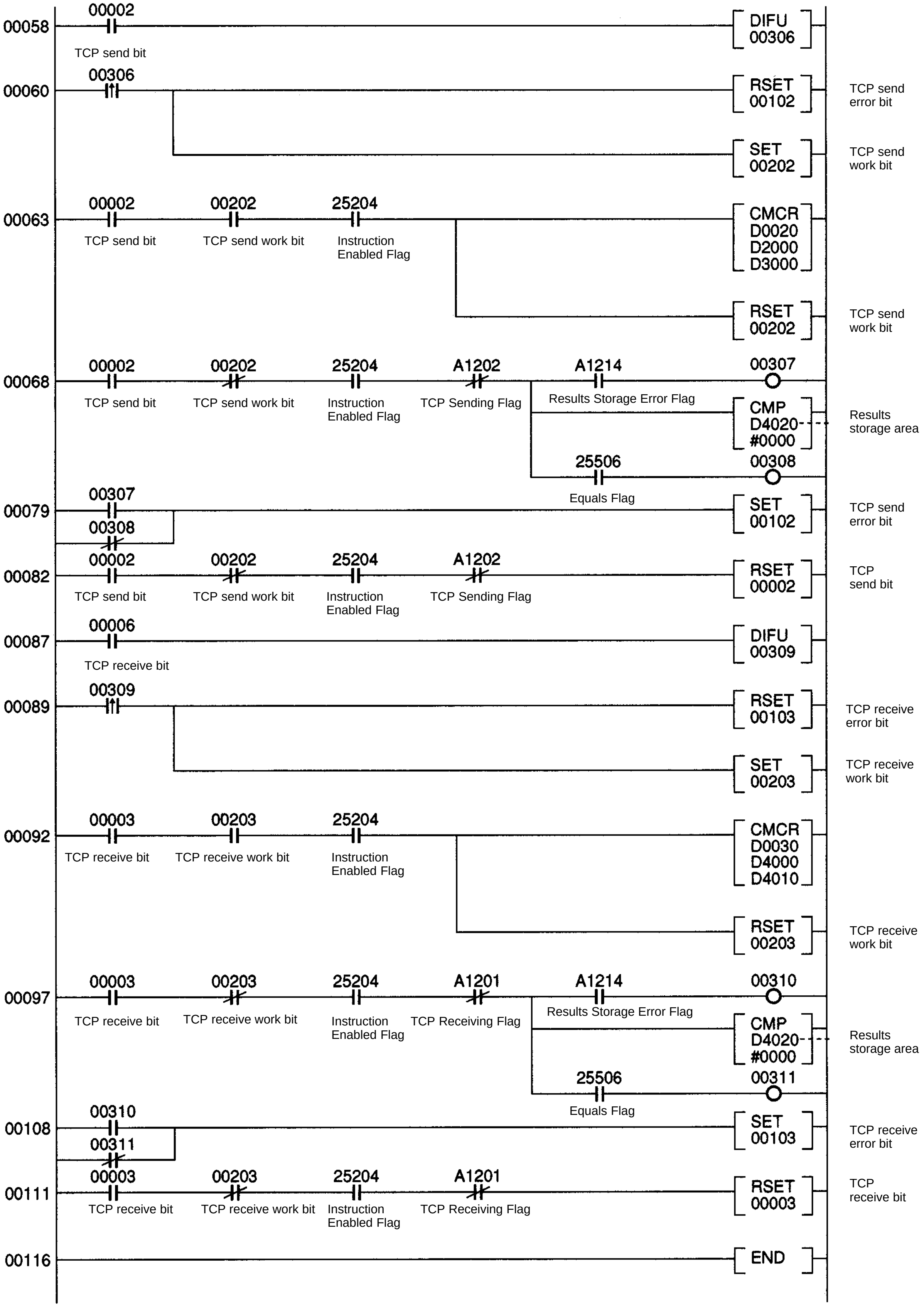
The first section of the program uses CMCR to execute TCP PASSIVE OPEN. Program execution starts when CIO 00000 turns ON.

Program address	Explanation
00000	Starting TCP OPEN execution. When CIO 00000 turns ON, the TCP open error bit (CIO 00100) turns OFF and the TCP open work bit (CIO 00200) turns ON to request one execution of TCP OPEN.
00005	Execution of the TCP OPEN REQUEST via CMCR If the Instruction Enabled Flag (25204) is ON while the TCP open work bit (CIO 00200) is ON, a TCP OPEN REQUEST is executed via CMCR and the TCP open work bit (CIO 00200) is turned OFF. Execution pauses until CMCR instruction execution has been completed.
00010	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the TCP Opening Flag (CIO A1200) turns OFF to indicate that execution of TCP OPEN via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (DM 1020) are checked and the TCP open error bit (CIO 00100) turns ON if an error has occurred.
00024	End of TCP OPEN execution. CIO 00000 turns OFF to indicate that execution has been completed.

The next section of program uses CMCR to execute TCP CLOSE. Execution starts when CIO 00001 turns ON

Program address	Explanation
00029	Starting TCP CLOSE execution. When CIO 00001 turns ON, the TCP close error bit (CIO 00100) turns OFF and the TCP close bit (CIO 00201) turns ON to request one execution of TCP CLOSE.
00034	Execution of the TCP CLOSE via CMCR. If the Instruction Enabled Flag (25204) is ON while the TCP close work bit (CIO 00201) is ON, TCP CLOSE is executed via CMCR and the TCP close work bit (CIO 00201) turns OFF. Execution pauses until the CMCR instruction has been completed.
00039	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the TCP closing bit (CIO A1203) turns OFF to indicate that execution of TCP CLOSE via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (DM 1050) are checked and the TCP close error bit (CIO 00101) turns ON if an error has occurred.
00053	End of TCP CLOSE execution. CIO 00001 turns OFF to indicate that execution has been completed.

Program Example 2



Explanation of Ladder Program

The next section of program uses CMCR to execute TCP SEND. Execution starts when CIO 00002 turns ON.

Program address	Explanation
00053	Starting TCP SEND execution. When CIO 00002 turns ON, the TCP send error bit (CIO 00102) turns OFF and the TCP send work bit (CIO 00202) turns ON to request one execution of TCP SEND.
00063	Execution of the TCP SEND via CMCR. If the Instruction Enabled Flag (25204) is ON while the TCP send work bit (CIO 00202) is ON, TCP SEND is executed via CMCR and the TCP send work bit (CIO 00202) turns OFF. Execution pauses until the CMCR instruction has been completed.
00068	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the TCP Sending Flag (CIO A1202) turns OFF to indicate that TCP SEND execution via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (DM 4020) are checked and the TCP send error bit (CIO 00102) is turned ON if an error has occurred.
00082	End of TCP SEND execution. CIO 00002 turns OFF to indicate that the instruction execution has been completed.

The next section of program uses CMCR to execute TCP RECEIVE. Execution starts when CIO 00003 turns ON.

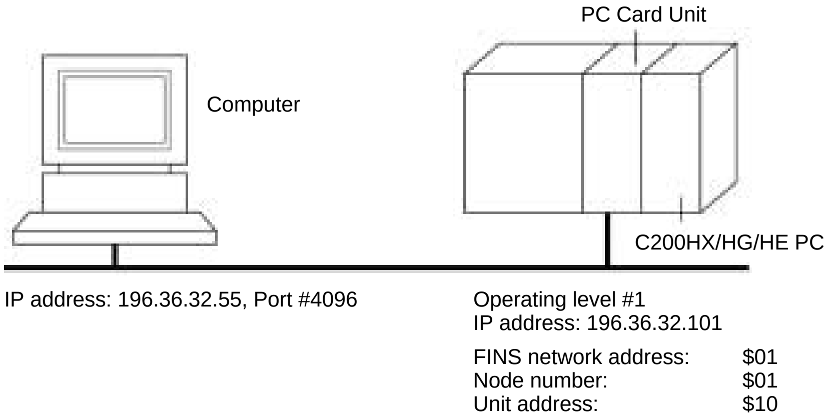
Program address	Explanation
00087	Starting TCP RECEIVE execution. When CIO 00003 turns ON, the TCP receive error bit (CIO 00103) turns OFF and the TCP receive work bit (CIO 00203) turns ON to request one execution of TCP RECEIVE.
00092	Execution of the TCP RECEIVE via CMCR. If the Instruction Enabled Flag (25204) is ON while the TCP receive work bit (CIO 00203) is ON, TCP RECEIVE is executed via CMCR and the TCP receive work bit (CIO 00203) turns OFF. Execution pauses until the CMCR instruction has been completed.
00097	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the TCP Receiving Flag (CIO A1201) turns OFF to indicate that execution of TCP RECEIVE via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (DM 4020) are checked and the TCP receive error bit (CIO 00103) is turned ON if an error has occurred.
00111	End of TCP RECEIVE execution. CIO 00003 turns OFF to indicate that execution has been completed.

10-3-2 Ladder Programming for UDP/IP Communications

The following program example sends and receives 100 bytes of data between the PC Card 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 PC Card Unit system setup are shown below. For purposes of this example, the network settings for the PC Card Unit and the computer are as shown in the diagram.


PC Card Unit
Computer
C200HX/HG/HE PC
IP address: 196.36.32.55, Port #4096
Operating level #1
IP address: 196.36.32.101
FINS network address: $01
Node number: $01
Unit address: $10

Explanation of Operation

UDP OPEN REQUEST is executed for the PC Card Unit when CIO 00000 turns ON. UDP CLOSE REQUEST is executed for the PC Card Unit when CIO 00001 turns ON. UDP SEND REQUEST is executed for the PC Card Unit when CIO 00002 turns ON and 100 bytes of data are sent beginning with DM 2009. UDP RECEIVE REQUEST is executed for the PC Card Unit when CIO 00003 turns ON and 100 bytes of data are received and stored beginning with DM 4025.

The appropriate bit in CIO 0001 turns ON when an error occurs. The following areas can be used to access details about errors:

CMCR response codes
Response codes in results storage area

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 CMCR control data, command data, and results storage.

Example

The following diagrams indicate that DM 0000 is the first word used in CMCR for a UDP OPEN REQUEST and it contains 1810_{hex}, that DM 0001 contains $000A_{hex}$, that CIO 00000 is used to control UDP open processing, and that CIO 00001 is used to control UDP close processing.

How to Read Tables

DM Area

	0	1	2	3	4	5	6	7	8	9
DM 0000	CMCR control data for UDP OPEN REQUEST (PASSIVE)									
	$1810	$000A								

CIO Area

	15 to 8		7	6	5	4	3	2	1	0
CIO 000									UDP close bit	UDP open bit

Memory Map for Sample Programs

DM Area

	0	1	2	3	4	5	6	7	8	9
DM 0000	CMCR control data for UDP OPEN REQUEST (PASSIVE)									
	$1810	$000A	$0004	$0001	$0110	$0000	$0032	—	—	—

CMCR timeout value: 5 s

Number of retries

Destination PC Card Unit designation
Network address: $01
Node number: $01
Unit address: $10

Number bytes to receive: 4 bytes

Number of bytes to send: 10

Operating level #1 (or $0810 for operating level #0)

	0	1	2	3	4	5	6	7	8	9
DM 0010	CMCR control data for UDP CLOSE REQUEST									
	$1810	$0008	$0004	$0001	$0110	$0000	$0032	—	—	—

Number bytes to receive: 4 bytes

Number of bytes to send: 8 bytes

	0	1	2	3	4	5	6	7	8	9
DM 0020	CMCR control data for UDP SEND REQUEST									
	$1810	$0074	$0004	$0001	$0110	$0000	$0032	—	—	—

Number bytes to receive: 4 bytes

Number of bytes to send: 116 bytes
Command format = 16 bytes + 100 bytes transmission data

	0	1	2	3	4	5	6	7	8	9
DM 0030	CMCR control data for UDP RECEIVE REQUEST									
	$1810	$000C	$0004	$0001	$0110	$0000	$0032	—	—	—

Number bytes to receive: 4 bytes

Number of bytes to send: 12 bytes
Number of bytes received specified in command data.

	0	1	2	3	4	5	6	7	8	9
DM 1000	UDP OPEN REQUEST (PASSIVE) command data									
	$0006	$2701	$0001	$8203	$FC00	$1000	—	—	—	—

Local port number: set to 4096 ($1000)

Results storage area: set to DM 1020 ($03FC)

UDP socket number (PC Card Unit socket number): set to 1

Command code

Transmission data length: 6 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 1010	UDP OPEN REQUEST (PASSIVE) response									
	$2701	Response code	—	—	—	—	—	—	—	—

Stores the response after CMCR command execution.

	0	1	2	3	4	5	6	7	8	9
DM 1020	UDP OPEN REQUEST (PASSIVE) results storage area									
	Response code	—	—	—	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
DM 1030	UDP CLOSE REQUEST command data									
	$0005	$2704	$0001	$8204	$1A00	—	—	—	—	—

Results storage area: set to DM 1050 ($041A)

UDP socket number closed: set to 1

Command code

Transmission data length: 5 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 1040	UDP CLOSE REQUEST response									
	$2704	Response code	—	—	—	—	—	—	—	—
DM 1050	UDP CLOSE REQUEST results storage area									
	Response code	—	—	—	—	—	—	—	—	—

	0	1	2	3	4	5	6	7	8	9
DM 2000	UDP SEND REQUEST command data									
	$0059	$2703	$0001	$820B	$C200	$C424	$2037	$1000	$0064	Send data: 100 bytes

Number of bytes to send: 100 bytes ($0064)

Destination port: Port #4096 ($1000)

Destination address: 196.36.32.55 ($C4.$24.$20.$37)

Results storage area: Set to DM 3010 ($0BC2)

Number of socket to be used: set to 1

Command code

Transmission data length: 59 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 3000	UDP SEND REQUEST response									
	$2703	Response code	——	——	——	——	——	——	——	——
DM 3010	UDP SEND REQUEST results storage area									
	Response code	No. of bytes sent	——	——	——	——	——	——	——	——

	0	1	2	3	4	5	6	7	8	9
DM 4000	UDP RECEIVE REQUEST command data									
	$0007	$2702	$0001	$820F	$B400	$0064	$0000	——	——	——

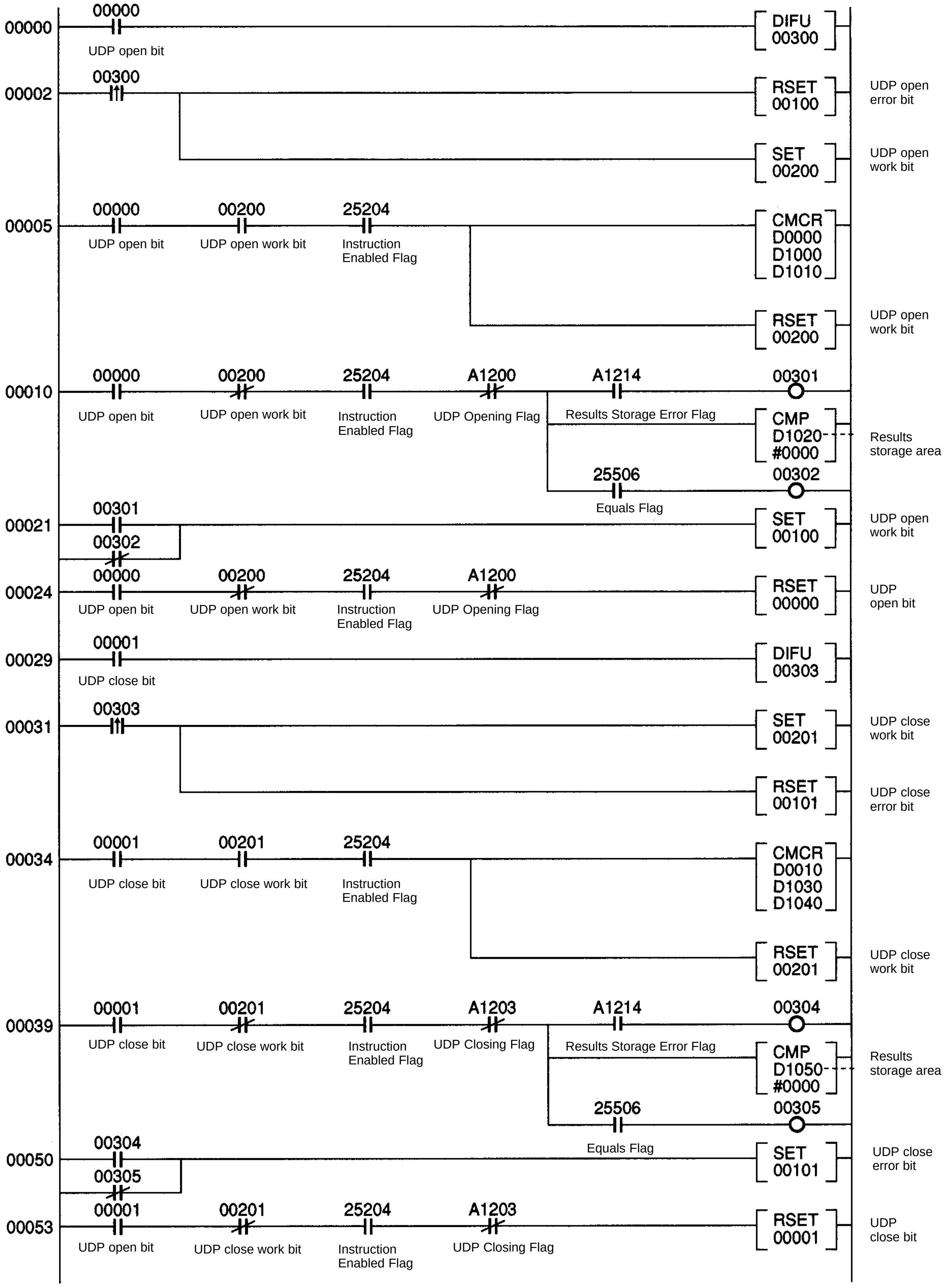
Timeout value: Not set
No. of bytes to receive: 100 bytes ($0064)
Results storage area: set to DM 4020 ($0FB4)
Number of socket to be used: set to 1
Command code
Transmission data length: 7 words (BCD)

	0	1	2	3	4	5	6	7	8	9
DM 4010	UDP RECEIVE REQUEST response									
	$2702	Response code	——	——	——	——	——	——	——	——
DM 4020	UDP RECEIVE REQUEST results storage area									
	Response code	Source IP address		Source port number	No. of bytes to receive	Send data: 100 bytes				

CIO Area

	15 to 8		7	6	5	4	3	2	1	0
CIO 000							UDP receive bit	UDP send bit	UDP close bit	UDP open bit
CIO 001							UDP receive error bit	UDP send error bit	UDP close error bit	UDP open error bit
CIO 002							UDP receive work bit	UDP send work bit	UDP close work bit	UDP open work bit

Program Example 3



Explanation of Ladder Program

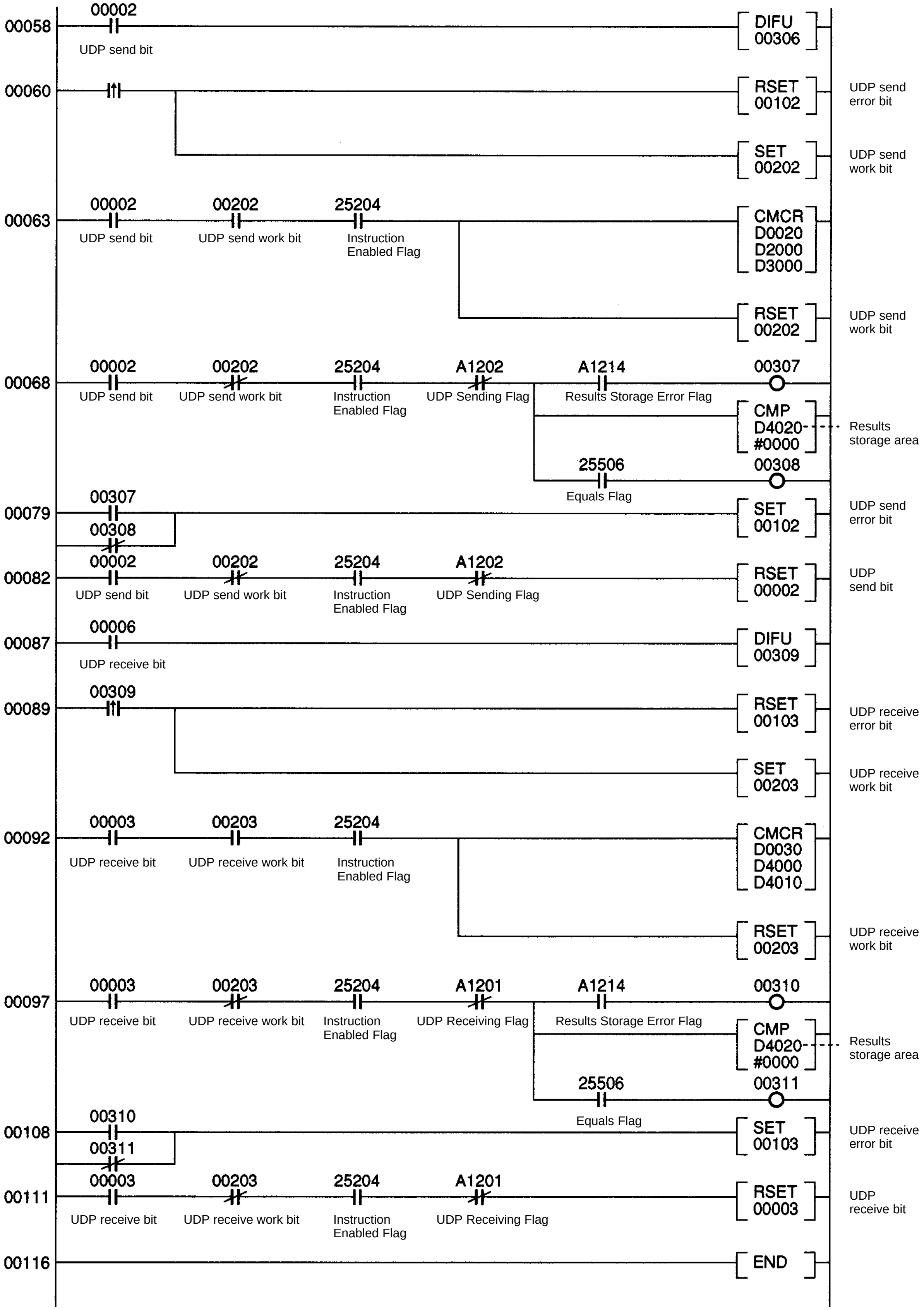
The first section of program uses CMCR to execute UDP PASSIVE OPEN. Program execution starts when CIO 00000 turns ON.

Program address	Explanation
00000	Starting UDP OPEN execution. When CIO 00000 turns ON, the UDP open error bit (CIO 00100) turns OFF and the UDP open work bit (CIO 00200) turns ON to request one execution of UDP OPEN.
00005	Execution of the UDP OPEN REQUEST via CMCR If the Instruction Enabled Flag (25204) is ON while the UDP open work bit (CIO 00200) is ON, a UDP OPEN REQUEST is executed via CMCR and the UDP open work bit (CIO 00200) is turned OFF. Execution pauses until CMCR instruction execution has been completed.
00010	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the UDP Opening Flag (CIO A1200) turns OFF to indicate that execution of UDP OPEN via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (D1020) are checked and the UDP open error bit (CIO 00100) turns ON if an error has occurred.
00024	End of UDP OPEN execution. CIO 00000 turns OFF to indicate that execution has been completed.

The next section of program uses CMCR to execute UDP CLOSE. Execution starts when CIO 00001 turns ON.

Program address	Explanation
00029	Starting UDP CLOSE execution. When CIO 00001 turns ON, the UDP close error bit (CIO 00100) turns OFF and the UDP close bit (CIO 00201) turns ON to request one execution of UDP CLOSE.
00034	Execution of the UDP CLOSE via CMCR. If the Instruction Enabled Flag (25204) is ON while the UDP close work bit (CIO 00201) is ON, UDP CLOSE is executed via CMCR and the UDP close work bit (CIO 00201) turns OFF. Execution pauses until the CMCR instruction has been completed.
00039	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the UDP closing bit (CIO A1203) turns OFF to indicate that execution of UDP CLOSE via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (D1050) are checked and the UDP close error bit (CIO 00101) turns ON if an error has occurred.
00053	End of UDP CLOSE execution. CIO 00001 turns OFF to indicate that execution has been completed.

Program Example 4



Explanation of Ladder Program

The next section of program uses CMCR to execute UDP SEND. Execution starts when CIO 00002 turns ON.

Program address	Explanation
00053	Starting UDP SEND execution. When CIO 00002 turns ON, the UDP send error bit (CIO 00102) turns OFF and the UDP send work bit (CIO 00202) turns ON to request one execution of UDP SEND.
00063	Execution of the UDP SEND via CMCR. If the Instruction Enabled Flag (25204) is ON while the UDP send work bit (CIO 00202) is ON, UDP SEND is executed via CMCR and the UDP send work bit (CIO 00202) turns OFF. Execution pauses until the CMCR instruction has been completed.
00068	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the UDP Sending Flag (CIO A1202) turns OFF to indicate that UDP SEND execution via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (D4020) are checked and the UDP send error bit (CIO 00102) is turned ON if an error has occurred.
00082	End of UDP SEND execution. CIO 00002 turns OFF to indicate that the instruction execution has been completed.

The next section of program uses CMCR to execute UDP RECEIVE. Execution starts when CIO 00003 turns ON.

Program address	Explanation
00087	Starting UDP RECEIVE execution. When CIO 00003 turns ON, the UDP receive error bit (CIO 00103) turns OFF and the UDP receive work bit (CIO 00203) turns ON to request one execution of UDP RECEIVE.
00092	Execution of the UDP RECEIVE via CMCR. If the Instruction Enabled Flag (25204) is ON while the UDP receive work bit (CIO 00203) is ON, UDP RECEIVE is executed via CMCR and the UDP receive work bit (CIO 00203) turns OFF. Execution pauses until the CMCR instruction has been completed.
00097	Error evaluation. The Instruction Enabled Flag (25204) turns ON again and the UDP Receiving Flag (CIO A1201) turns OFF to indicate that execution of UDP RECEIVE via CMCR has ended. The Results Storage Error Flag (CIO A1214) and the response code in the results storage area (D4020) are checked and the UDP receive error bit (CIO 00103) is turned ON if an error has occurred.
00111	End of UDP RECEIVE execution. CIO 00003 turns OFF to indicate that execution has been completed.

SECTION 11
Using FINS Commands and Responses

This section describes the FINS commands that can be sent to the C200HX/HG/HE CPU and the FINS commands that can be sent to the PC Card Unit.

11-1 Commands and Responses for C200HX/HG/HE CPUs

This subsection explains how to use commands and responses for C200HX/HG/HE CPUs.

With the C200HX/HG/HE, the CMCR instruction is used for issuing FINS commands. This subsection explains communications data formats, FINS commands to the C200HX/HG/HE and to the PC Card Unit, and how to specify memory areas.

11-1-1 Command/Response Parameters

All parameters are specified in hexadecimal unless otherwise specified.

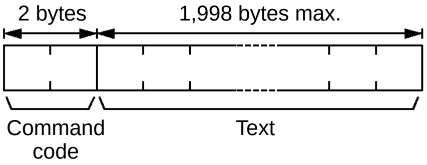
11-1-2 Communications Data Formats

Headers

When FINS commands are issued from a device such as an FA computer, a header must be affixed before the command code. For details regarding headers, refer to *9-6 FINS Communications From Computers*.

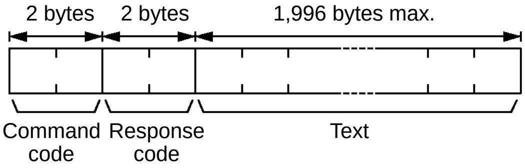
Commands

Commands have the following format.



Responses

Responses have the following format.



Note The maximum size of a command or response depends upon the type of network through which the transmission is relayed. The above limits are for FINS commands and responses on Ethernet networks.

11-1-3 Commands and Responses

Command code		Name	PC mode			Page
			RUN	MONITOR	PROGRAM	
01	01	MEMORY AREA READ	Valid	Valid	Valid	*165*
	02	MEMORY AREA WRITE	Valid	Valid	Valid	*166*
	04	MULTIPLE MEMORY AREA READ	Valid	Valid	Valid	*167*
03	06	PROGRAM AREA READ	Valid	Valid	Valid	*168*
	07	PROGRAM AREA WRITE	Not valid	Not valid	Valid	*169*
04	01	RUN	Valid	Valid	Valid	*169*
	02	STOP	Valid	Valid	Valid	*170*
05	01	CONTROLLER DATA READ	Valid	Valid	Valid	*170*
06	01	CONTROLLER STATUS READ	Valid	Valid	Valid	*171*
07	01	CLOCK READ	Valid	Valid	Valid	*172*
	02	CLOCK WRITE	Not Valid	Valid	Valid	*173*
21	01	ERROR CLEAR	Valid	Valid	Valid	*173*
23	01	FORCED SET/RESET	Not valid	Valid	Valid	*174*
	02	FORCED SET/RESET CANCEL	Not valid	Valid	Valid	*175*
	0A	MULTIPLE FORCED STATUS READ	Valid	Valid	Valid	*175*

11-1-4 Memory Area Designations

The following table gives the addresses to use when reading or writing PC data.

Memory area	Data	Data area address	Address used in communications		Memory area code	No. of bytes
			1st and 2nd bytes	3rd byte		
CIO area	Bit status	00000 to 51115	0000 to 01FF	00 to 0F	00	1
	Word contents	000 to 511		00 to 00	80	2
LR area	Bit status	LR 0000 to LR 6315	03E8 to 0427	00 to 0F	00	1
	Word contents	LR 00 to LR 63		00 to 00	80	2
HR area	Bit status	HR 0000 to HR 9915	0428 to 048B	00 to 0F	00	1
	Word contents	HR 00 to HR 99		00 to 00	80	2
AR area	Bit status	AR 0000 to AR 2715	048C to 04A7	00 to 0F	00	1
	Word contents	AR 00 to AR 27		00 to 00	80	2
Timer/Counter Area	Completion Flag status	TIM 000 to TIM 511 CNT 000 to CNT 511	0000 to 01FF	00 to 00	01	1
	PV	TIM 000 to TIM 511 CNT 000 to CNT 511		00 to 00	81	2
DM Area	Word contents	DM 0000 to DM 9999	0000 to 270F	00 to 00	82	2
Expansion DM Area	Word contents	EM 0000 to EM 6143	0000 to 17FF	00 to 00	90 to 98	2

Note 1. The size of the memory area varies depending on the PC. Refer to the PCs operation manual for details on the size limits of the memory areas.

2. The meanings of the memory area codes for the Expansion DM area are shown below.

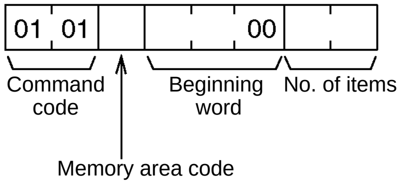
Memory area code	Meaning
98	Current bank
90 to 97	Banks 0 to 7
A8 to AF	Banks 8 to 15

11-1-5 MEMORY AREA READ

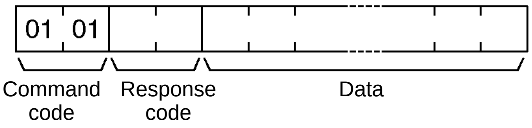
Reads the contents of the specified number of consecutive memory area words starting from the specified word. All words must be in the same memory area (here, all memory areas with the same memory area code are considered as one area).

Note Even when the no. of items is 00, this command will end normally.

Command Block



Response Block



Parameters

Memory area code (command): The data area to read.

Beginning word(command): The address of the first word/bit/flag to read from memory. Specify 00 for the 3rd byte.

No. of items (command): The number of items to be read. Specify 0000 to 03F1 (0 to 1009 decimal). The command can complete normally even if zero items are specified.

Data (response): The data from the specified words is returned in sequence starting from the beginning address. PVs for timers and counters are returned as BCD. The required number of bytes in total is calculated as follows:

No. of bytes required by each item x No. of items

Memory Areas

The data in the following table can be read. (Refer to *11-1-4 Memory Area Designations* for PC word/bit address designations.)

Memory area	Data	Memory area code	No. of bytes
CIO, LR, HR, or AR area	Word contents	80	2
Timer/Counter	Completion Flag status	01	1
	PV	81	2
DM	Word contents	82	2
Expansion DM	Word contents	90 to 98, A8 to AF	2

Response code

Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU and Appendix E FINS Response Codes from the PC Card Unit*.

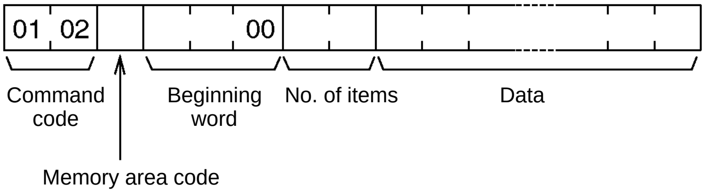
11-1-6 MEMORY AREA WRITE

Writes data to the specified number of consecutive words starting from the specified word.

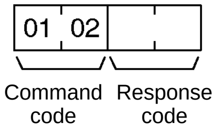
Note a) Even when the no. of items is 00 this command will end normally.

b) When data is written in the Timer/Counter PV Area, the Completion Flags will be turned OFF (0).

Command Block



Response Block



Parameters

Memory area code (command): The data area to write.

Beginning word(command): The first word/value to write. Specify 00 for the 3rd byte.

No. of items (command): The number of items to be written. Specify 0000 to 03E5 (0 to 997 decimal). The command can complete normally even if zero items are specified.

Data (command): The data to be written. PVs for timers and counters are written as BCD. The required number of bytes in total is calculated as follows:

The required number of bytes in total is calculated as follows:

2 bytes x No. of items

The following data can be written (refer to *11-1-4 Memory Area Designations* for the word/bit address designations).

Memory area	Data	Memory area code	No. of bytes
CIO, LR, HR, or AR area	Word contents	80	2
Timer/Counter	PV	81	2
DM	Word contents	82	2
Expansion DM	Word contents	90 to 98, A8 to AF	2

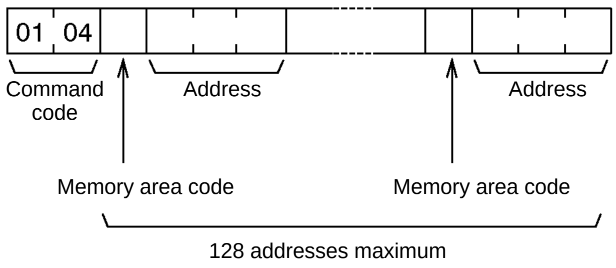
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-1-7 MULTIPLE MEMORY AREA READ

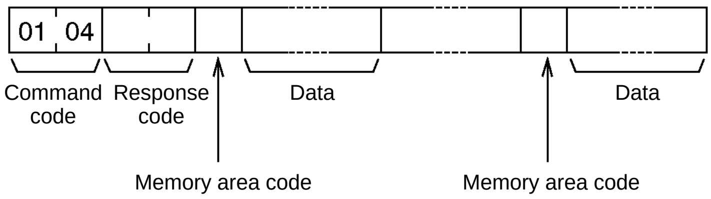
Reads the contents of the specified number of non-consecutive memory area words, starting from the specified word.

Note If there is an error in the command code or a beginning address, no data will be read.

Command Block



Response Block



Parameters

Memory area code (command): The data area to read.

Address (command): The word/bit/flag to read. The content of up to 128 addresses can be read. (If part of the data to be read is in the Expansion DM area, the content of up to 100 addresses only can be read.)

Data (response): The data in the specified memory area(s) will be returned in sequence starting from the beginning address.

Memory Areas

The following data can be read (refer to *11-1-4 Memory Area Designations* for memory area designations):

Memory area	Data	Memory area code	No. of bytes
CIO, LR, HR, or AR area	Bit status	00	1
	Word contents	80	2
Timer/Counter	Completion Flag status	01	1
	PV	81	2
DM	Word contents	82	2
Expansion DM	Word contents	90 to 98, A8 to AF	2

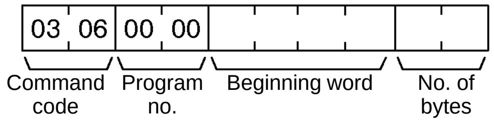
Response code

Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

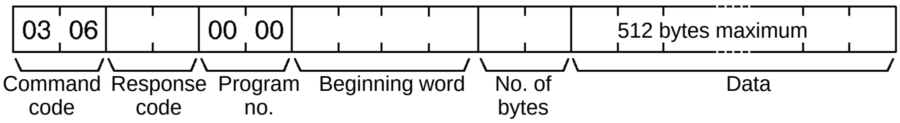
11-1-8 PROGRAM AREA READ

Reads the contents of the specified number of consecutive program area words starting from the specified word. The program is read in machine language (object code). A maximum of 512 bytes can be read with each command.

Command Block



Response Block

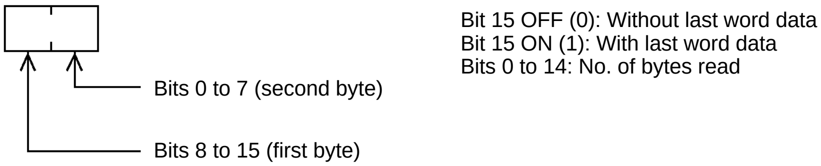


Parameters

Program no. (command and response): Set to 0000.

Beginning word(command and response): Set a relative byte address with 00000000 as the starting address. The beginning word must be an even number. The address set in the command will be returned in the response.

No. of bytes (command and response): The number of bytes in an even number 0200 (512 in decimal) or smaller. The number of bytes actually read will be returned in the response. Bit 15 will be ON (1) in the response block when the last word data of the program area is returned.


Bit 15 OFF (0): Without last word data
Bit 15 ON (1): With last word data
Bits 0 to 14: No. of bytes read
Bits 0 to 7 (second byte)
Bits 8 to 15 (first byte)

Data (response): The data in the specified program area will be returned in sequence starting from the beginning word.

Note If the designated number of bytes is larger than the program area, the program will be read through the final address and a response code indicating an address range error or response length error will be returned.

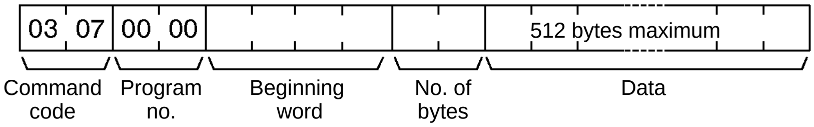
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-1-9 PROGRAM AREA WRITE

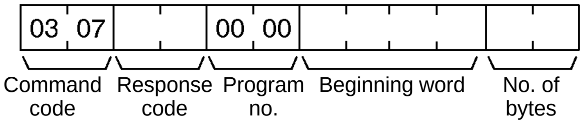
Writes data to the specified number of consecutive program area words starting from the specified word. A maximum of 512 bytes can be written with each command. To write larger amounts of data, use multiple commands and specify the beginning word and number of words for each.

Note
1. When bit 15 is ON (1) the index create of the PC is activated.
2. When the no. of bytes is set to 8000, only an index marker is created.

Command Block



Response Block

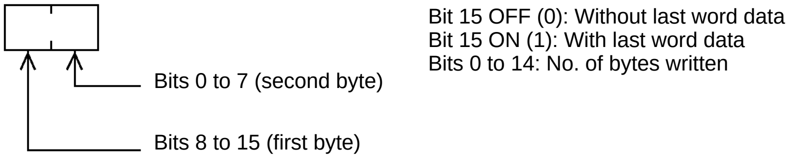


Parameters

Program no. (command and response): Set to 0000.

Beginning word (command and response): Set a relative byte address with 00000000 as the starting address. The beginning word must be an even number. The address set in the command will be returned in the response.

No. of bytes (command and response): The number of bytes in an even number (512 or smaller). The number of bytes actually written will be returned in the response. Bit 15 must be turned ON (1) when data for the last write to the program area so that the PC can generate an index. To write only an index marker, specify 8000 for the number of bytes.


Bits 0 to 7 (second byte)
Bits 8 to 15 (first byte)
Bit 15 OFF (0): Without last word data
Bit 15 ON (1): With last word data
Bits 0 to 14: No. of bytes written

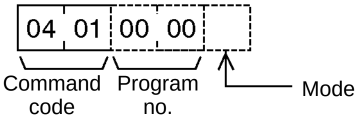
Data (command): The data to be written.

Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

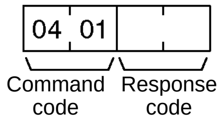
11-1-10 RUN

Changes the PC to MONITOR or RUN mode, enabling the PC to execute its program.

Command Block



Response Block



Parameters

Program no. (command): Set to 0000.

Mode (command): As follows:

02:	MONITOR mode
04:	RUN mode

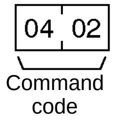
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

Note If the mode is not specified, the PC will go to MONITOR mode.

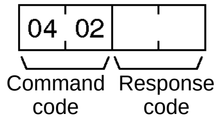
11-1-11 STOP

Changes the PC to PROGRAM mode, stopping program execution.

Command Block



Response Block



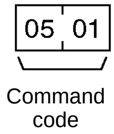
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-1-12 CONTROLLER DATA READ

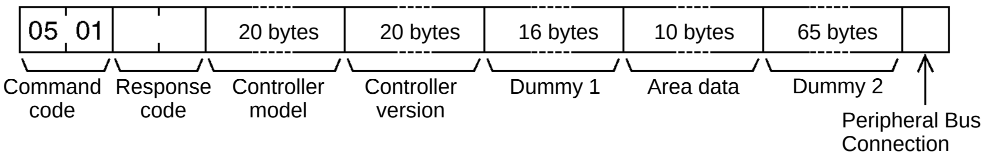
Reads the following data:

- Controller model and version
- Area data
- PC status

Command Block



Response Block

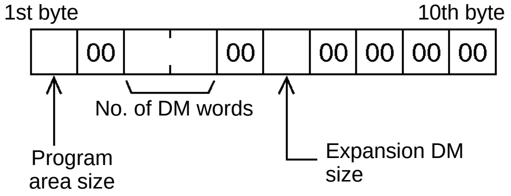


Parameters

Controller model and Controller version (response): Both are returned in ASCII (20 bytes (i.e., 20 ASCII characters) max. each). If the model or version information does not require 20 bytes, the remainder of the 20 bytes will be filled with spaces (ASCII 20).

Dummy 1 and Dummy 2 (response): All zeros will be returned.

Area data (response): As follows:



Item	Meaning	Unit
Program area size	The size of PC Setup and program area	K words (1K words = 1,024 words; 1 word = 2 bytes))
No. of DM words	Total words in the DM area 6656 has no Expansion DM 8000 Expansion DM has 1000 words 9000 Expansion DM has 2000 words 10000 Expansion DM has 3000 words	Words (1 word = 2 bytes)
Expansion DM size	Banks in the Expansion DM area	Banks (1 bank = 32,766 words)

Peripheral Bus Connection (response):
The Peripheral Bus connection is indicated as follows:

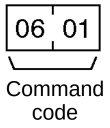
00: No peripheral connection recognized
80: Peripheral connected

Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

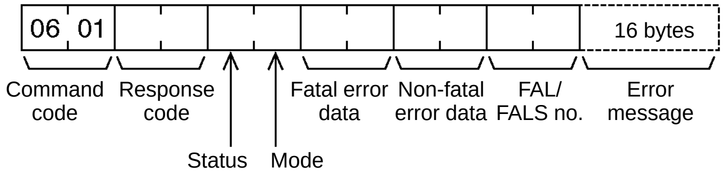
11-1-13 CONTROLLER STATUS READ

Reads the status of the Controller.

Command Block



Response Block



Parameters

Status (response): The operating status of the PC as follows:

00: STOP (program not being executed)

01: RUN (program being executed)

80: CPU on standby (the start switch is OFF or the CPU is waiting for a signal from a device such as a Remote I/O Slave Unit).

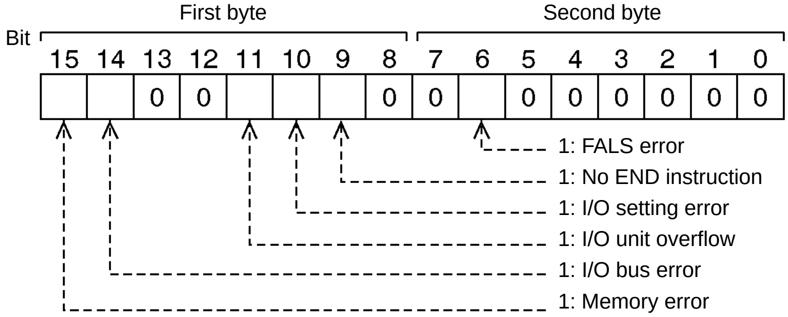
Mode (response): One of the following PC modes:

00: PROGRAM

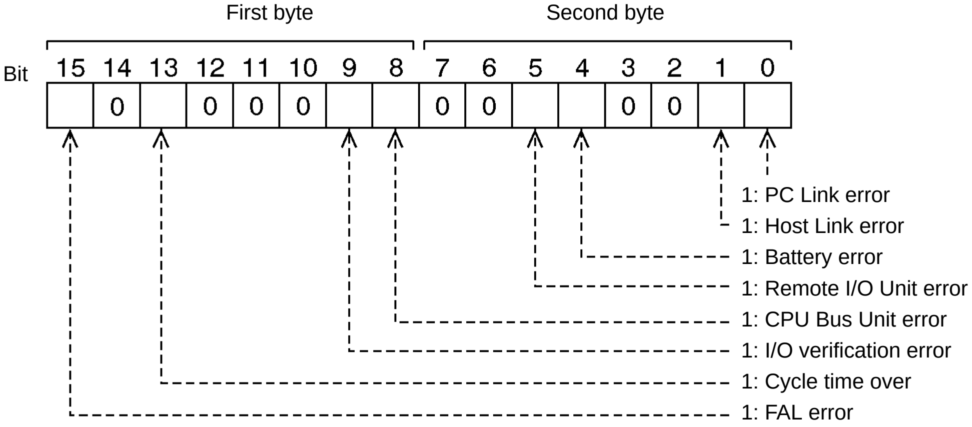
02: MONITOR

04: RUN

Fatal error data (response): The contents of fatal error information from the PC (for details refer to your PC’s operation manual).



Non-fatal error data (response): The contents of non-fatal error information from the PC (for details refer to your PC’s operation manual).



FAL/FALS no. (response): The number of the highest priority FAL/FALS error is returned as BCD between 00 and 99 (decimal) to the second byte. The first byte is always 00. If no error has occurred, 0000 is returned.

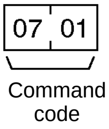
Error message (response): The error message of the present FAL/FALS number is returned as 16 ASCII characters (16 bytes). If there is no error message, nothing will be returned.

Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

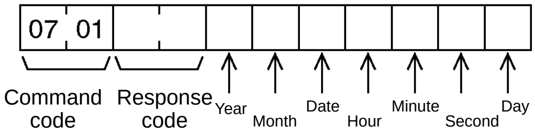
11-1-14 CLOCK READ

Reads the clock.

Command Block



Response Block



Parameters

Year, month, date, hour, minute, second, day (response): Each value is expressed in BCD.

Year: The rightmost two digits of the year.

Hour: 00 to 23.

Day: As follows:

Value	00	01	02	03	04	05	06
Day	Sun	Mon	Tues	Wed	Thur	Fri	Sat

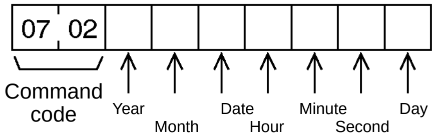
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-1-15 CLOCK WRITE

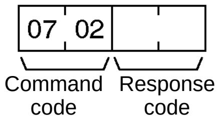
Sets the clock.

Note Specify all data.

Command Block



Response Block



Parameters

Year, month, date, hour, minute, second, day (command): Each specified value is expressed in BCD.

Year: The rightmost two digits of the year.

Hour: Specify 00 to 23.

Day: As follows:

Value	00	01	02	03	04	05	06
Day	Sun	Mon	Tues	Wed	Thur	Fri	Sat

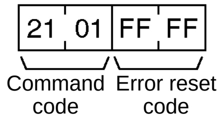
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-1-16 ERROR CLEAR

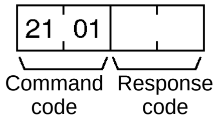
Clears errors from the PC. A normal response will be returned even if an error has not occurred.

Note The cause of the error must be removed before executing the ERROR CLEAR command, or the same error will occur again after the ERROR CLEAR command is executed.

Command Block



Response Block



Parameters

Error reset code (command): Set to FFFF.

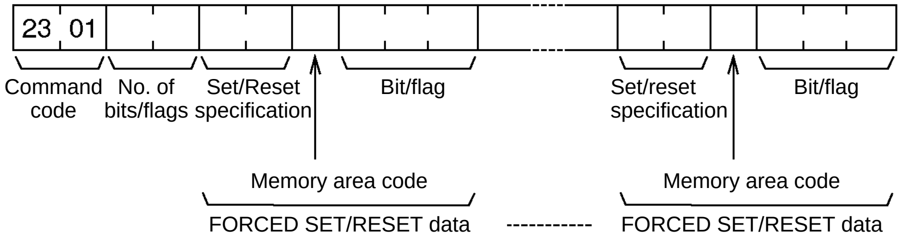
Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-1-17 FORCED SET/RESET

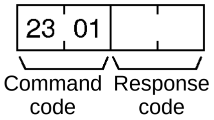
Force-sets (ON) or force-resets (OFF) bits/flags or releases force-set status. Bits/flags that are forced ON or OFF will remain ON or OFF and cannot be written to until the forced status is released.

Note
1. This command cannot be used to release the status of Completion Flags for timers or counters. If forced status is released and the Completion Flag is turned ON, it will be force-set; if forced status is released and the Completion Flag is turned OFF, it will be force-reset.
2. FORCED SET/RESET can be used for more than one bit.
3. Forced status will be maintained until it is released using FORCED SET/RESET CANCEL.

Command Block



Response Block



Parameters

No. of bits/flags (command): The number of bits/flags to be processed.

Set/Reset specification (command): The action to be taken for each bit/flag:

Value	Name
0000	Force-reset (OFF) (0)
0001	Force-set (ON) (1)
8000	Forced status released and bit turned OFF (0)
8001	Forced status released and bit turned ON (1)
FFFF	Forced status released

Memory area code (command): The memory area of the bit or flag to be controlled. Only the specified areas will be controlled. When using the PC Card Unit only bits on the same word and memory area can be specified.

Bit/Flag (command): The bit or flag to be controlled.

Memory Areas
The bits (flags) in the following memory areas can be FORCED SET/RESET or released. (Refer to *11-1-4 Memory Area Designations* for memory area designations.)

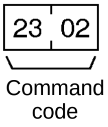
Memory area	Data	Memory area code
CIO, LR, HR, and AR areas	Bits status	00
Timer/Counter	Completion Flag status	01

Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

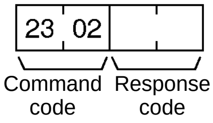
11-1-18 FORCED SET/RESET CANCEL

Cancels all bits (flags) that have been forced ON or forced OFF.

Command Block



Response Block



Note The bits (flags) in the following memory areas can be forced set or forced reset.

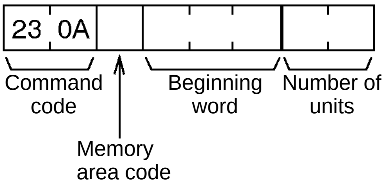
Memory area	Data
CIO, LR, HR, and AR areas	Bits status
Timer/Counter	Completion Flag status

Response code
Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

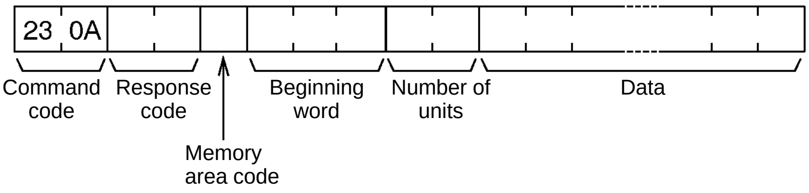
11-1-19 MULTIPLE FORCED STATUS READ

Reads the forced status of the specified range of words or timers/counters.

Command Block



Response Block



Parameters

Memory area code, Beginning word, Number of units (command, response): Specify the memory area code, the beginning word in that area, and the number of words or timers/counters to read. The number of units can be between 0001 and 0040 (1 to 64 in decimal).

The actual area, beginning word, and number of unit to be read will be returned in the response.

Memory Areas

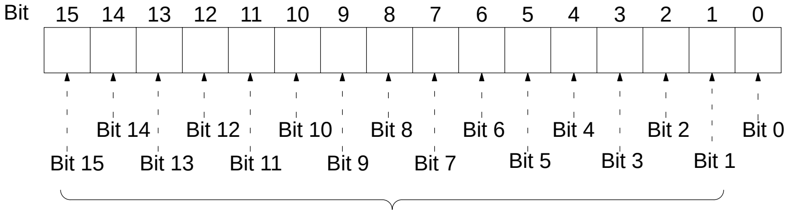
Forced status can be read in the following areas. Refer to *11-1-4 Memory Area Designations* for memory area designations.

Area	Data type	Memory area code	Number of bytes
CIO, LR, HR, and AR areas	Current value of word	80	2
Timer/counter area	Completion Flag status	01	1

Note Forced status is read by words for the CIO, LR, HR, or AR area and by bits (flags) for the timer/counter area.

Data (response): Forced status is returned beginning from the specified word or timer/counter. The number of bytes returned will be (the number of units) x (the number of bytes/unit).

CIO, LR, HR, and AR Areas:



The status for each bit is as follows:
OFF (0): No forced status in effect
ON (1): Forced ON or forced OFF

Timers/Counters: Status of the Completion Flag will be returned as follows:

00: No forced status in effect
01: Forced ON or forced OFF

Response code

Refer to *Appendix D Response Codes from the C200HX/HG/HE CPU*.

11-2 PC Card Unit Commands and Responses

When a PC Card Unit receives a FINS command, it processes a response and returns it to the node that sent the command.

11-2-1 PC Card Unit FINS Commands List

The following is a list of the FINS commands supported by the PC Card Unit.

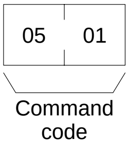
Command Code		Name	Page
05	01	CONTROLLER DATA READ	*177*
08	01	INTERNODE ECHO TEST	*177*
	02	BROADCAST TEST RESULTS READ	*178*
	03	BROADCAST TEST DATA SEND	*178*
21	02	ERROR LOG READ	*179*
	03	ERROR LOG CLEAR	*180*
22	02	SINGLE FILE READ	*180*
	03	SINGLE FILE WRITE	*181*
	05	FILE DELETE	*182*
	07	FILE COPY	*182*
	08	FILE NAME CHANGE	*183*
	12	SINGLE FILE READ WITH COMMAS	*183*
	13	SINGLE FILE WRITE WITH COMMAS	*184*

Note For details regarding FINS commands requesting socket services (27□□), refer to *11-3 FINS Commands Requesting Socket Services*.

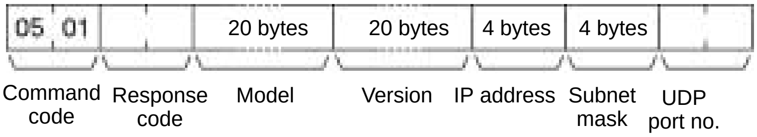
11-2-2 CONTROLLER DATA READ

Reads the PC Card Unit's model and version.

Command Block



Response Block



Parameters

Model (response): The model is returned in 20 bytes of ASCII (i.e., 20 ASCII characters). If the model information does not require 20 bytes, the remainder of the 20 bytes will be filled with spaces ($20).

Version (response): The version is returned in 20 bytes of ASCII (i.e., 20 ASCII characters). If the version information does not require 20 bytes, the remainder of the 20 bytes will be filled with spaces ($20).

IP address, subnet mask (response):
There are four bytes returned respectively for the PC Card Unit's IP address and subnet mask.

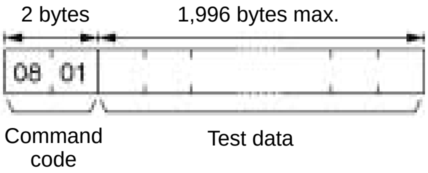
UDP port number for FINS (response):
There are two bytes returned for the PC Card Unit's FINS UDP port.

Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

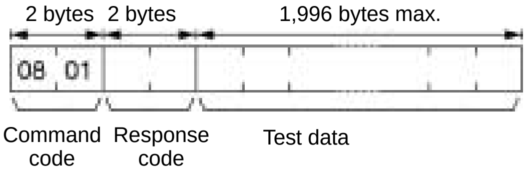
11-2-3 INTERNODE ECHO TEST

Performs an internode echo test with specified nodes.

Command Block



Response Block



Parameters

Test data (command, response): Up to 1,996 bytes of test data can be included in the command. This data is transmitted to the indicated node and returned unchanged if communications are normal. If the data returned in the response differs from that transmitted in the command, it means that an error occurred in the test.

Note 1. The test destination is set with the CMCR instruction's control data.

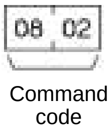
2. For the unit address, the PC Card Unit must be specified.

Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

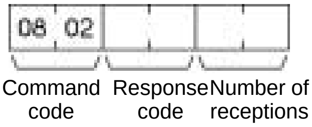
11-2-4 BROADCAST TEST RESULTS READ

Reads the results (number of receptions) of the broadcast tests carried out for all the nodes in the network, using the BROADCAST TEST DATA SEND command. Refer to *11-2-5 BROADCAST TEST DATA SEND* for details on that command.

Command Block



Response Block



Parameters

Number of receptions (response): The number of times that the BROADCAST TEST DATA SEND command has been executed since the last BROADCAST TEST RESULTS READ command was executed is returned.

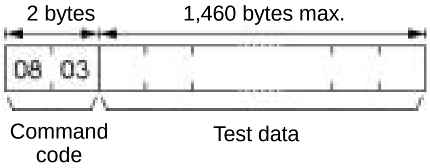
This parameter is cleared each time BROADCAST TEST RESULTS READ is executed. If the number of receptions does not equal the number of times that the BROADCAST TEST DATA SEND command has been executed since the last BROADCAST TEST RESULTS READ command was executed, it means that an error has occurred.

Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-5 BROADCAST TEST DATA SEND

Broadcasts the test data in the command to all nodes in the specified network. No response is returned when this command is executed, but reception of the test data can be verified by executing the BROADCAST TEST RESULTS READ command. For details regarding that command, refer to *11-2-4 BROADCAST TEST RESULTS READ*.

Command Block



Parameters

Test data (command): Sets the data to be broadcast, up to a maximum of 1,460 bytes. The broadcast test can be executed even if no test data is set.

Note When using this command, set the FINS header parameters (or the control data for CMCR) as shown below.

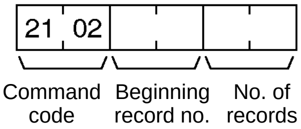
Destination node number:	$FF (broadcast transmission)
Destination unit address:	$FE (PC Card Unit)
Response bit:	1 (Response not returned)

Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-6 ERROR LOG READ

Reads the error data from the PC Card Unit. Error data is erased after it has been read.

Command Block

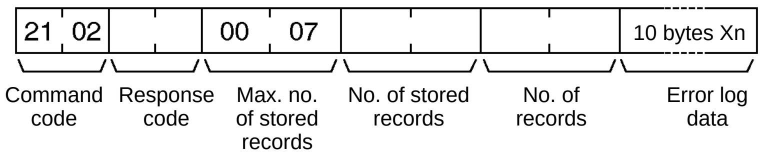


Parameters

Beginning record no. (command): Designates the beginning record number in a range of 0000 to 0006 (0 to 6 in decimal). The first record is 0000. When error data is read from the middle, all preceding data is erased from the system. One record consists of 10 bytes and stores the data for one error record.

No. of records (command): Designates the number of records to be read in a range of 0001 to 0007 (1 to 7 in decimal).

Response Block



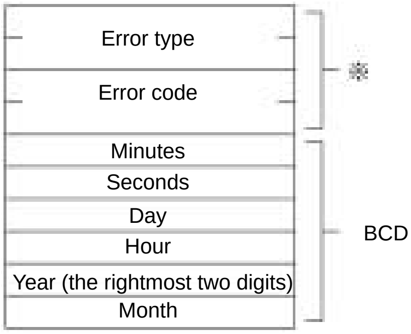
Parameters

Max. no. of stored records (response): Designates the maximum number of records stored. For the PC Card Unit, this is always 7.

No. of stored records (response): Returns the number of records remaining after records have been read. When no records have been read, returns the current number of records.

No. of records read (response): Returns the actual number of records read.

Error log data (response): Error log data is stored record by record in the following block.



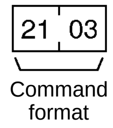
Refer to *12-1-2 Error Log and Indicators* for details on error codes.

Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

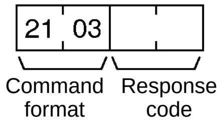
11-2-7 ERROR LOG CLEAR

Resets the PC Card Unit's Error Log.

Command Block



Response Block

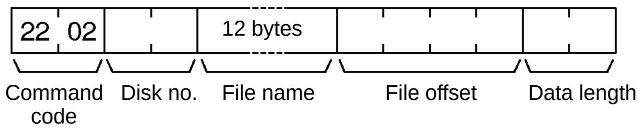


Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-8 SINGLE FILE READ

Reads the contents of a file stored in the file device connected to the PC Card Unit.

Command Block



Parameters

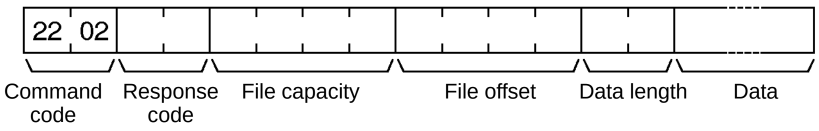
Disk no. (command): Specify the file device from which to read as follows:
G drive (slot 1): $0006 H drive (slot 2): $0007

File name (command): The name of the file to be read (in capital letters). Set as: File name of up to eight letters + period + 3-letter extension. One-byte alphanumeric is used. When there are not enough letters in the name to fill 12 bytes, fill the remaining bytes with spaces ($20).

File offset (command): Designates the offset from the beginning of the file from which to start reading.

Data length (command): Designates the number of bytes to be read. When set to 0000, checks whether there is any data in the file at the specified offset.

Response Block



Parameters

File capacity (response): File size in bytes. When a SINGLE FILE READ is executed for a file with a capacity of 0 bytes (no data), the data length will be 0000 and no data will be returned.

File offset (response): Returns the offset from the beginning of the file.

Data length (response): Returns the actual number of bytes of data read.

Data: Data is returned in sequence from the beginning word (offset) through the specified data length.

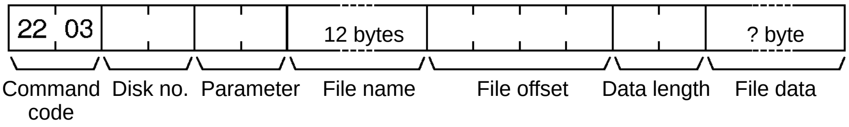
Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-9 SINGLE FILE WRITE

Writes a new file to the file device connected to the PC Card Unit or appends/overwrites an existing file stored in the file device. Designation can be made to protect existing files if an attempt is made to create a new file of the same name as an existing file.

Note Do not create any files on the memory card that begin with “TMP...”. The system uses this prefix.

Command Block



Parameters

Disk no. (command): Specify the file device to which to write (in the PC Card Unit) as follows:
G drive (slot 1): $0006   H drive (slot 2): $0007

Parameter (command): As follows:

0000: Writes a new file. If a file with the same name already exists, the new file will not be created.

0001: Writes a new file. If a file with the same name already exists, it will be overwritten.

0002: Appends data to an existing file. (Appended to the end of the file.)

0003: Overwrites an existing file.

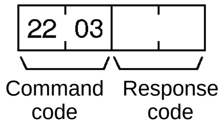
File name (command): The name of the file to be written (in capital letters). Set as: File name of up to eight letters + period + 3-letter extension. One-byte alphanumeric code is used. When there are not enough letters to fill all 12 bytes, fill the remaining bytes with spaces ($20).

File offset (command): The number of bytes from the start of the file from which to start writing (files start at 00000000). To create a new file or append data to an existing file, specify 00000000 as the file offset.

Data length (command): The number of bytes to be written.

File data (command): The data to be written to the file. The data is written in sequence from the file offset through the specified data length.

Response Block



Note

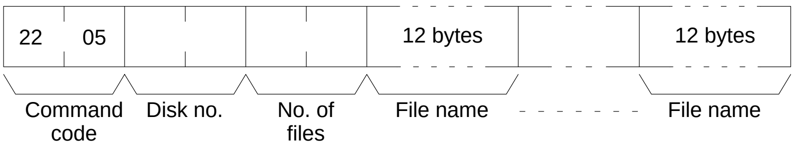
1. If the data length is set to 0000 when creating a new file, a file with capacity of 0 (no data) will be created.
2. If the data length is set to 0000 when overwriting an existing file, the command will check whether there is any data at the file offset.
3. When the data exceeds the file device’s remaining capacity SINGLE FILE WRITE cannot be executed. In this case the contents of the existing file will be left unchanged.
4. When a SINGLE FILE WRITE is executed, the PC Card Unit’s time data will be used to record the time and date of the file.

Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-10 FILE DELETE

Deletes files stored in a file device connected to the PC Card Unit.

Command Block



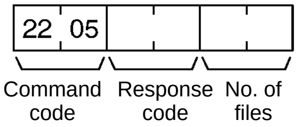
Parameters

Disk no. (command): Specify the file device (memory card) as follows:
G drive (slot 1): $0006 H drive (slot 2): $0007

No. of files (command): The number of files to be deleted

File name (command): The names of the files to be deleted (in capital letters). Set as: File name of up to eight letters + period + 3-letter extension. One-byte alphanumeric code is used. When there are not enough letters to fill all 12 bytes, fill the remaining bytes with spaces ($20).

Response Block



No. of files (response): The number of files that have been deleted.

Note
1. The specified files will be deleted in sequence. If nonexisting file names have been specified, the PC Card Unit will ignore them and the operation will continue.
2. If the specified number of files and the number of file names do not coincide no files will be deleted.

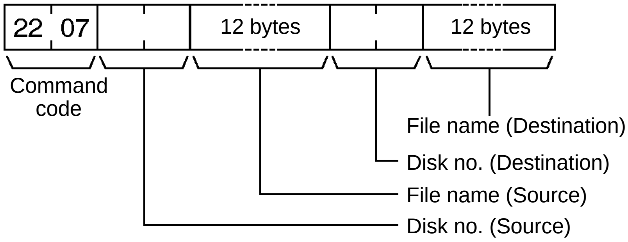
Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-11 FILE COPY

Copies a file onto the memory card of the PC Card Unit.

Note Do not create any files on the memory card that begin with “TMP...”. The system uses this prefix.

Command Block

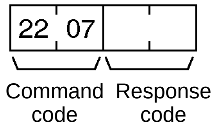


Parameters

Disk no. (command): Set the file device (memory cards) as follows:
G drive: $0006 H drive: $0007

File name (command): The file to be copied and a new name for the copied file. Set as: File name of up to eight letters + period + 3-letter extension. One-byte Shift JIS alphanumeric code is used. When there are not enough letters to fill all 12 bytes, fill the remaining bytes with spaces ($20).

Response Block



Note

1. The copied file name cannot be omitted, even if the destination and source file names are the same.
2. The file will not be copied if an existing file name is given.
3. When a FILE COPY is executed, the PC Card Unit's time data will be used to record the time and date of the copied file.

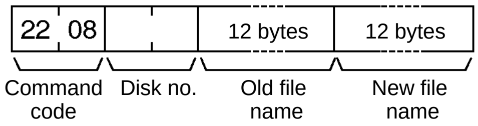
Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-12 FILE NAME CHANGE

Changes a file name.

Note Do not create any files on the memory card that begin with "TMP...". The system uses this prefix.

Command Block



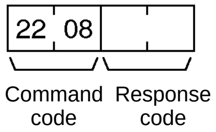
Disk no. (command): Set the file device (memory cards) as follows:
G drive (slot 1): $0006 H drive (slot 2): $0007

Parameters

Old file name (command): The original file name for the file.

New file name (command): The new name for the file. The file will not be copied if an existing file name is given. Set as: File name of up to eight letters + period + 3-letter extension. One-byte alphanumeric code is used. When there are not enough letters to fill all 12 bytes, fill the remaining bytes with spaces ($20).

Response Block

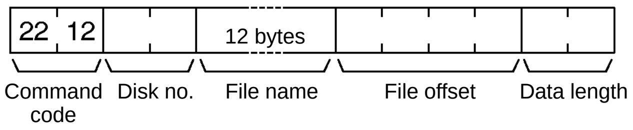


Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-13 SINGLE FILE READ WITH COMMAS

Reads file data contained in the file device connected to the PC Card Unit. Reads specified files as comma-separated value files (CSV files), and then removes the commas before returning the data.

Command Block



Parameters

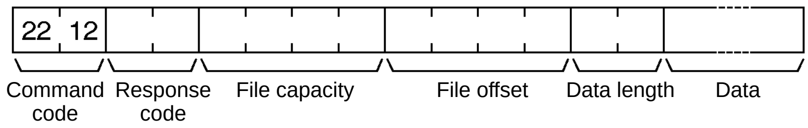
Disk no. (command): Set the file device (on the PC Card Unit) as follows:
G drive (slot 1): $0006 H drive (slot 2): $0007

Beginning file name (command): The name of the file to be read.
Set as: File name of up to eight letters + period + 3-letter extension. One-byte alphanumeric code is used. When there are not enough letters to fill all 12 bytes, fill the remaining bytes with spaces ($20).

File offset (command): Designates the offset from the beginning of the file from which to start reading.

Data length (command): The number of bytes to be read. The file is comma-separated and the data will thus be in word units. Therefore, set the number of bytes to an even number. When the length is set to 0000, the command will check whether there is any data at the file offset.

Response Block



Parameters

File capacity (response): The capacity (bytes) of the file that was read. When the file capacity is 0 bytes and the SINGLE FILE READ is executed, the data length will be 0000 and no data will be read.

File offset (response): File offset from the first byte of the file is returned.

Data length (response): The number of bytes read.

Data: The data is read in sequence from the file offset through the specified data length.

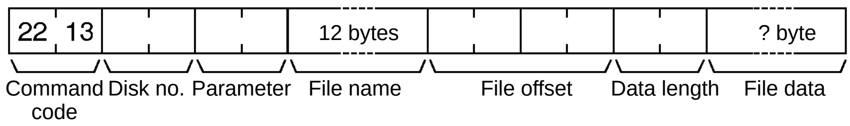
Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

11-2-14 SINGLE FILE WRITE WITH COMMAS

Writes a new file to the file device connected to the PC Card Unit or appends to/overwrites an existing file stored in the file device. Writes specified files as comma-separated value files (CSV files), i.e., adds commas to the data before writing the files. Designation can be made to protect existing files if an attempt is made to create a new file of the same name as an existing file.

Note Do not create any files on the Memory Card that begin with “TMP...”. The system uses this prefix.

Command Block



Parameters

Disk no. (command): File offset codes for data on a PC Card Unit are designated as follows:
G drive (slot 1): $0006 H drive (slot 2): $0007

Parameter (command): As follows:

0000: Writes a new file. If a file with the same name already exists, the new file will not be created.

0001: Writes a new file. If a file with the same name already exists, it will be overwritten.

0002: Appends data to an existing file. (Appended to the end of the file.)

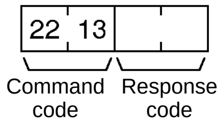
0003: Overwrites an existing file.

File name (command): The name of the file to be written.
Set as: File name of up to eight letters + period + 3-letter extension. One-byte alphanumeric code is used. When there are not enough letters to fill all 12 bytes, fill the remaining bytes with spaces ($20).

File offset (command): The number of bytes from the start of the file from which to start writing (files start at 00000000). To create a new file or add data to an existing file, specify 00000000 as the file offset.

Data length (command): The number of bytes to be written.

File data (command) : The data to be written to the file. The data is written in sequence from the file offset through the specified data length.



Response code
Refer to *Appendix E FINS Response Codes from the PC Card Unit*.

Note

1. If the data length is set to 0000 when creating a new file, a file with capacity of 0 (no data) will be created.
2. If the data length is set to 0000 when overwriting an existing file, the command will check whether there is any data at the file offset.
3. When the data exceeds the file device's remaining capacity SINGLE FILE WRITE cannot be executed. In this case the contents of the existing file will be left unchanged.
4. When a SINGLE FILE WRITE is executed, the PC Card Unit's time data will be used to record the time and date of the file.
5. The PC Card Unit's time will revert to that of the PC to which it is connected when either the power supply is turned on or it is reset.

11-3 FINS Commands Requesting Socket Services

This subsection explains the FINS commands that request socket services for the PC Card Unit. These FINS commands are used for executing communications by TCP/IP or UDP/IP.

11-3-1 Summary of Commands

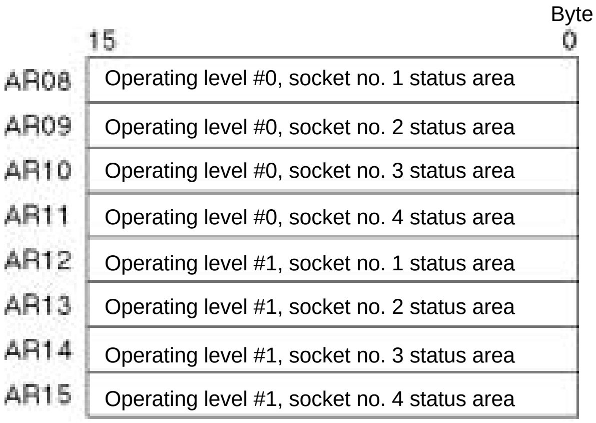
The following table summarizes the FINS commands requesting socket services for PC Card Unit response processing.

Command code		Name	Page
MRC	SRC		
27	01	UDP OPEN REQUEST	187
	02	UDP RECEIVE REQUEST	188
	03	UDP SEND REQUEST	189
	04	UDP CLOSE REQUEST	190
	10	TCP OPEN REQUEST (PASSIVE)	191
	11	TCP OPEN REQUEST (ACTIVE)	192
	12	TCP RECEIVE REQUEST	193
	13	TCP SEND REQUEST	194
	14	TCP CLOSE REQUEST	195

11-3-2 Socket Status Area

Each PC Card Unit has four sockets that can be used for TCP or UDP. The status of each of these sockets can be confirmed from the respective AR bits.

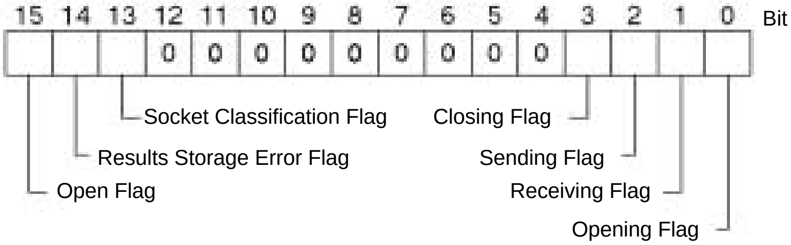
AR Words 08 to 15



AR08	Operating level #0, socket no. 1 status area
AR09	Operating level #0, socket no. 2 status area
AR10	Operating level #0, socket no. 3 status area
AR11	Operating level #0, socket no. 4 status area
AR12	Operating level #1, socket no. 1 status area
AR13	Operating level #1, socket no. 2 status area
AR14	Operating level #1, socket no. 3 status area
AR15	Operating level #1, socket no. 4 status area

Bit Configuration

The meaning of individual bits in each status word is shown in the following diagram.



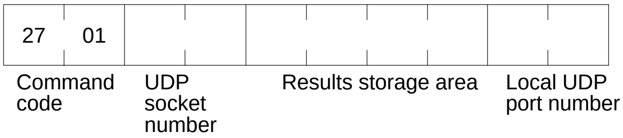
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Bit
0 0 0 0 0 0 0 0 0
Socket Classification Flag
Results Storage Error Flag
Open Flag
Closing Flag
Sending Flag
Receiving Flag
Opening Flag

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 12	Not used			
Bit 13	Socket Classification Flag	1	TCP socket	Indicates that the open socket is a TCP socket. (This has no meaning if the Open Flag is “0.”)
		0	UDP socket	Indicates that the open socket is a UDP socket. (This has no meaning if the Open Flag is “0.”)
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 a PC Card 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.

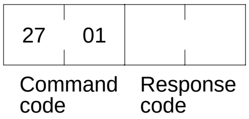
11-3-3 UDP OPEN REQUEST

Requests processing to open a socket.

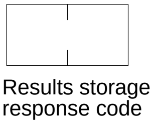
Command Block



Response Block



Results Storage Format



Parameters

UDP Socket Number (Command): Specify the UDP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Local UDP Port Number (Command): The UDP port number for communications with the socket is specified as two 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.

Note Do not designate the port that is being used as the UDP port number for FINS (default: 9600).

Response Codes

Response code	Description
0000	Normal
1100	UDP socket number is out of range. Local UDP port number is zero.
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
220F	Specified socket is already open.

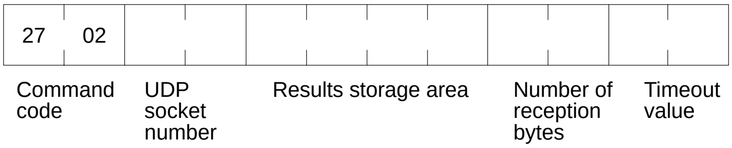
Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Cannot secure the internal buffer. (ENOBUFS)
0049	Duplicate UDP port number. (EADDRINUSE)

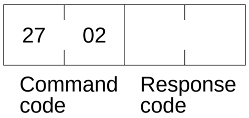
11-3-4 UDP RECEIVE REQUEST

Requests that data be received from a UDP socket.

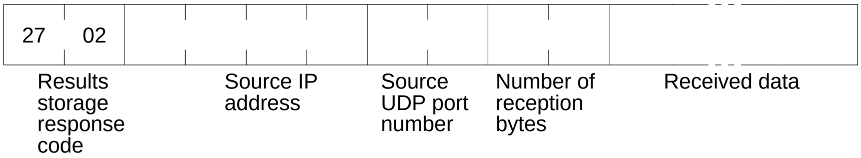
Command Block



Response Block



Results Storage Format



Parameters

UDP Socket Number (Command): Specify the UDP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

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.

Note If more than the specified number of bytes is received, only the specified amount will be stored and the remainder will be discarded.

Timeout Value (Command): Specifies 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 (0080) 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.

Response Codes

Response code	Description
0000	Normal
1100	UDP socket number or number of reception bytes is out of range.
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
2201	The specified socket is opened as a TCP socket.
220F	The specified socket is currently receiving data.
2210	The specified socket is not open.

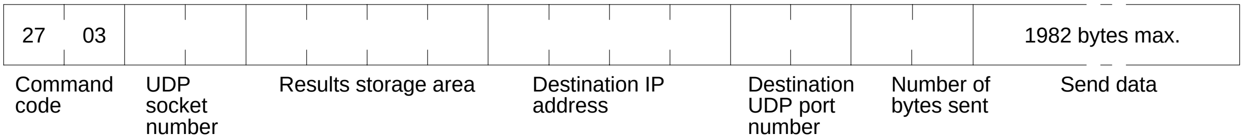
Results Storage Area Response Codes

Response code	Description
0000	Normal
0042	Message size was too large to be stored in the buffer. (EMSGSIZE)
0080	A timeout error occurred.
0081	The specified socket was closed while receiving data.

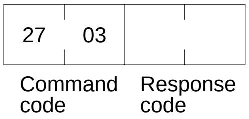
11-3-5 UDP SEND REQUEST

Requests that data be sent by a UDP socket.

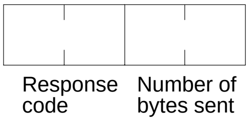
Command Block



Response Block



Results Storage Format



Parameters

UDP Socket Number (Command): Specify the UDP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Destination IP Address (Command): Specifies in hexadecimal the IP address of the node to which data is being sent. For broadcasting, set everything to “1” for the host address.

Destination UDP Port Number (Command): Specifies in hexadecimal 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 1,982 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
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 memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
2201	The specified socket is opened as a TCP socket.
220F	The specified socket is currently receiving data.
2210	The specified socket is not open.

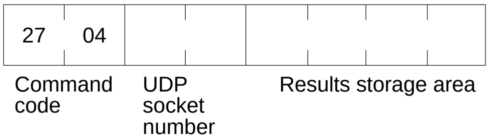
Results Storage Area Response Codes

Response code	Description
0000	Normal
0042	The send destination IP address is a broadcast address and the number of bytes sent exceeds 1,472. (EMSGSIZE)
004E	Could not reach the remote network. (ENETUNREACH)

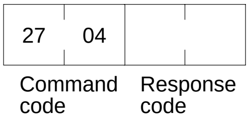
11-3-6 UDP CLOSE REQUEST

Requests processing to close a socket.

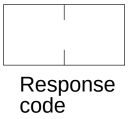
Command Block



Response Block



Results Storage Format



Parameters

UDP Socket Number (Command): Specify the UDP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Response Codes

Response code	Description
0000	Normal
1100	UDP socket number is out of range.
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
2201	The specified socket is opened as a TCP socket.
2210	Specified socket is not open.

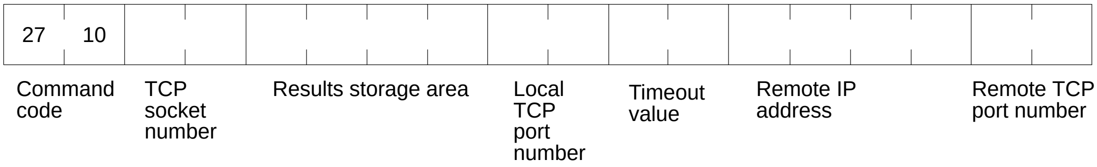
Results Storage Area Response Codes

Response code	Description
0000	Normal

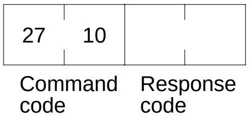
11-3-7 TCP OPEN REQUEST (PASSIVE)

Requests processing to open a TCP socket. The socket will wait to be connected to another node.

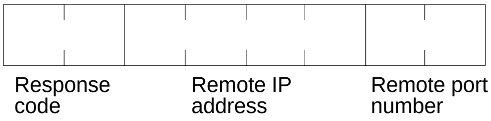
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command): Specify the TCP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Local TCP Port Number (Command): The TCP port number for communications with the socket is specified as two bytes.

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 (0080) 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.

Response Codes

Response code	Description
0000	Normal
1100	TCP socket number is out of range. Remote IP address is 0.
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
220F	The specified socket (connection) is already open or is currently being opened.

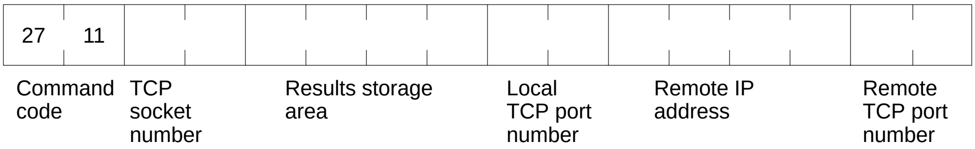
Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Cannot secure the internal buffer. (ENOBUFS)
0080	An open request timeout error occurred.

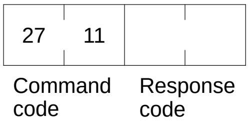
11-3-8 TCP OPEN REQUEST (ACTIVE)

Requests processing to open a TCP socket. The socket will be connected to another node.

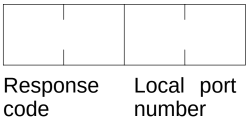
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command): Specify the TCP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Local TCP Port Number (Command, Results Storage Area): The local TCP port number is specified as two bytes. An available TCP port number is automatically assigned if 0 is specified. The TCP port number allocated to the open socket is stored in the results storage area.

Note Do not make other port specifications if you specify the local TCP port number. Making other specifications may cause transmissions to collide on the port.

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
1100	TCP socket number is out of range or remote IP address is 0.
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
220F	The specified socket is already open or is being opened.

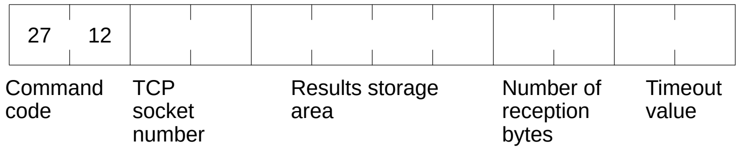
Results Storage Area Response Codes

Response code	Description
0000	Normal
003E	Cannot secure the internal buffer. (ENOBUFS)
0049	Duplicated TCP port numbers. (EADDRINUSE)
004A	A communication error occurred with the remote node. (ENCONNREFUSED) Passive remote is not available. An attempt was made to actively open local TCP port.
004C	The designated IP address is not correct. (EADRNOTAVAIL)
0053	A communication error occurred with the remote node. No remote exists. (ETIMEDOUT)

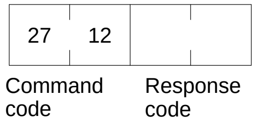
11-3-9 TCP RECEIVE REQUEST

Requests that data be received from a TCP socket.

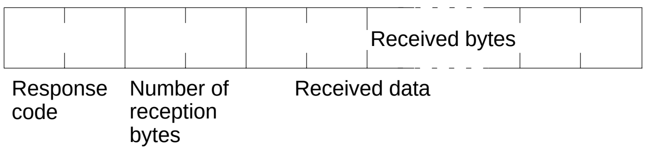
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command): Specify the TCP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

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.

Reception of 0 bytes indicates that the communications were forcibly cut by the remote node. Close the socket.

Note If more than the specified number of bytes is received, only the specified amount will be stored and the remainder will be read when the next data set is received.

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 (0080) 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
1100	TCP socket number is out of range. Number of reception bytes is out of range (i.e., over 1982).
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
2201	The specified socket is opened as a UDP socket.
220F	The specified socket is receiving data.
2210	The specified socket is not open.

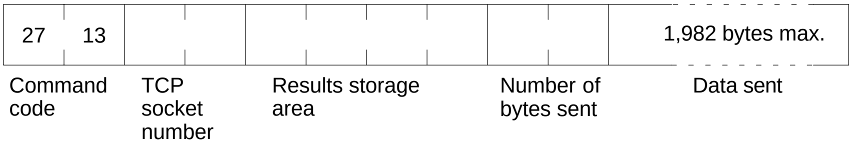
Results Storage Area Response Codes

Response code	Description
0000	Normal
0016	Connection has not been established due to cable trouble, etc., and the receive request became an error. (EINVAL)
0040	Received by a socket not connected. (ENOTCONN)
0042	Message size was too large to be stored in the buffer. (EMSGSIZE)
004B	An error occurred in communications with the remote node. (ECONNRESET)
0080	A timeout error occurred.
0081	Socket was closed while receiving.

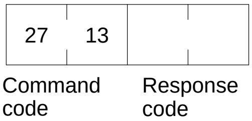
11-3-10 TCP SEND REQUEST

Requests that data be sent by a TCP socket.

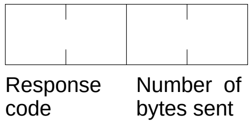
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command): Specify the TCP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Number of Bytes Sent (Command, Results Storage Area): The number of bytes in the data sent is specified between 1 and 1,982. 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
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 memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
2201	The specified socket is opened as a UDP socket.
220F	The specified socket is sending data.
2210	The specified socket is not open.

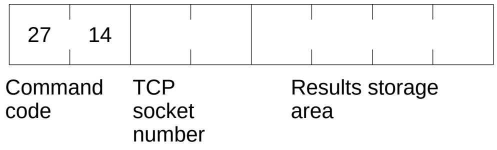
Results Storage Area Response Codes

Response code	Description
0000	Normal
0016	Connection has not been established due to cable trouble, etc., and the send request became an error. (EINVAL)
0020	Socket connection broken during transmission.
003E	Cannot secure the internal buffer. (ENOBUFS)
0045	An error occurred in communications with the remote node. (ECONNABORTED)
004B	An error occurred in communications with the remote node. (ECONNRESET)
004E	Could not reach the remote network. (ENETUNREACH)
0053	An error occurred in communications with the remote node. (ETIMEDOUT)
0081	The specified socket was closed during transmission.

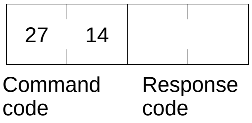
11-3-11 TCP CLOSE REQUEST

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.

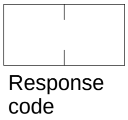
Command Block



Response Block



Results Storage Format



Parameters

TCP Socket Number (Command): Specify the TCP socket number to be opened as two bytes between 1 and 4.

Results Storage Area (Command): The words 0 to 255 and 256 to 511 of the memory area, words 0 to 6143 of the data memory, and words 0 to 6143 (each bank) of the expansion data memory can be used. However, do not designate the special auxiliary area. Refer to *11-1-4 Memory Area Designations*. When a bank that does not have expansion data memory is designated, the Results Storage Area Error Flag will turn ON.

Note 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. Wait for the close processing on the connected node side.

Response Codes

Response code	Description
0000	Normal
1100	The TCP socket number is out of range.
1101	The memory area code for the results storage area is out of range.
1103	Non-zero bit address is specified for the results storage area.
2201	The specified socket is opened as a UDP socket.
2210	No connection could be established to the specified socket.

Results Storage Area Response Codes

Response code	Description
0000	Normal

Part 4
Troubleshooting

This part of the manual includes information on troubleshooting errors and using the error log.

SECTION 12
Error Processing

This section describes the meaning of indicators and the actions to be taken when the indicators light. It also explains how to use the error log.

12-1 Indicators and the Error Log

When an error occurs in the CPU bus, Ethernet network, or other parts of the system, the PC Card Unit indicates the error with indicators and records it in the error log to enable the user to identify the error. This section explains how to retrieve the records in the error log and describes the error log and indicators.

Note Turning the PC Card Unit OFF erases the error log. To prevent this, always retrieve the error log before turning the PC Card Unit OFF.

12-1-1 Reading the Error Log

To read the error log, execute the FINS command ERROR LOG READ (command code 21 02) from another computer or CV-series CPU, addressing it to the PC Card Unit.

Refer to ERROR LOG READ in *11-2 PC Card Unit Commands and Responses* for details on command usage and response details.

12-1-2 Error Log

The following table describes the error types and codes recorded by the PC Card Unit.

Error type	Code	Description	Action
2	1	The PC is not the C200HX/HG/HE.	---
	3	CPU watchdog timer error	Eliminate the cause of the problem in the main PC CPU.
3	1	Ethernet initialization failure. Socket open error. (See note.)	Check the settings and correct where necessary.
	2	Other Ethernet errors. IP cannot be used. (See note.)	Same as above.

Note Type-3 errors don’t occur with the C200HW-PCU01.

12-1-3 Troubleshooting Using Indicators

Non-fatal Errors

ERR indicator	RUN indicator	Cause and action
Lit	Lit	A message was received from a node that is not registered on the conversion table for IP addresses and FINS node addresses. (Does not occur with C200HW-PCS01-V2.) In this case, the received message is processed, but the response is discarded. Unit operation continues. Register the node if necessary.

Fatal Errors

ERR indicator	RUN indicator	Cause and action
Lit	---	A watchdog timer error occurred at the main PC CPU. Eliminate the cause of the problem at the PC.
		A Unit memory parity error (hardware error) occurred. Replace the Unit if necessary.
Lit	Not lit	The default ODI driver is set at the time of delivery. (Occurs with C200HW-PCS01-V2 only.) Change the setting to that of the ODI driver for the Ethernet card used.
		There is an error in the settings (e.g. the ODI driver setting). (Occurs with C200HW-PCS01-V2 only.) Check the settings and correct where necessary.
		The access right for data servicing between the PC Card Unit and the PC has not been released from the PC. Check the connections between devices, and reset the power supply.

12-2 Troubleshooting

Formatting Does Not Finish:

If card access indicator is flashing:
Formatting is in progress. Wait until formatting is complete**.**

If card access indicator is not flashing:
Check whether the card type and the slot match the switch settings. Also, contact the card manufacturer to check whether the card can be used with PHOENIX PCM Plus 3.2.
Try formatting once with a personal computer.

CMCR Terminates With an Error:

Check the parameters.
Check whether the card is write-protected.
Check whether the file name extension is three characters.
Check whether the correct drive and directory are specified.
Check whether a directory has been created beforehand.

SEND or RECV Times Out:

Check whether the command parameters are correctly specified.
Check whether the cable is correctly connected.
Issue a PING command from the personal computer. If PING response is not returned, repeat the setup procedure.

RUN Indicator Does Not Light in File Transfer Mode, and then the Unit Does Not Run in Any Mode:

If the power is interrupted during a file transfer, the built-in flash file system will be damaged and it may not be possible to start up in file transfer mode. If that occurs, it will be necessary to have the Unit repaired.

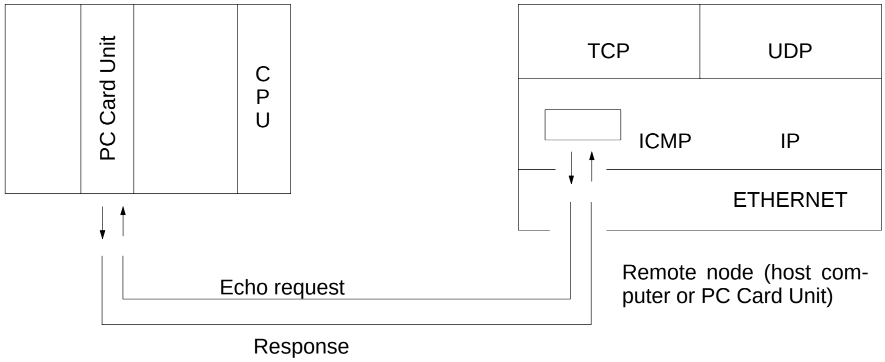
12-3 Echo Test With PING Command

The PC Card Unit incorporates the PING command supported as a standard feature by many host computers. The PING command is an echo test using ICMP (Internet Control Message Protocol).

12-3-1 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 PC Card Unit supports both the ICMP echo request and reply functions.



12-3-2 Using the Host Computer

The PING command can be executed from the host computer to send an echo request packet to a PC Card Unit. The method for using the PING command from a Unix computer is described below.

12-3-3 Using PING from a Unix Computer

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.

Appendix A
Standard Models

Name	Specifications	Model
PC Card Unit	Does not support Ethernet.	C200HW-PCU01
PC Card Unit Ethernet Set	Supports Ethernet	C200HW-PCS01-EV1
Bus Connection Unit	Connects one PC Card Unit	C200HW-CE011
	Connects one PC Card Unit and one SYSMAC LINK or SYSMAC NET Link Unit.	C200HW-CE012
Communications Board	CPU bus interface only	C200HW-COM01
	CPU bus interface and RS-232C port	C200HW-COM04-E
Ethernet Setup Software	Provided with C200HW-PCS01.	——

Appendix B
Specifications

PC Card Unit Specifications

The specifications for the PC Card Unit are shown below.

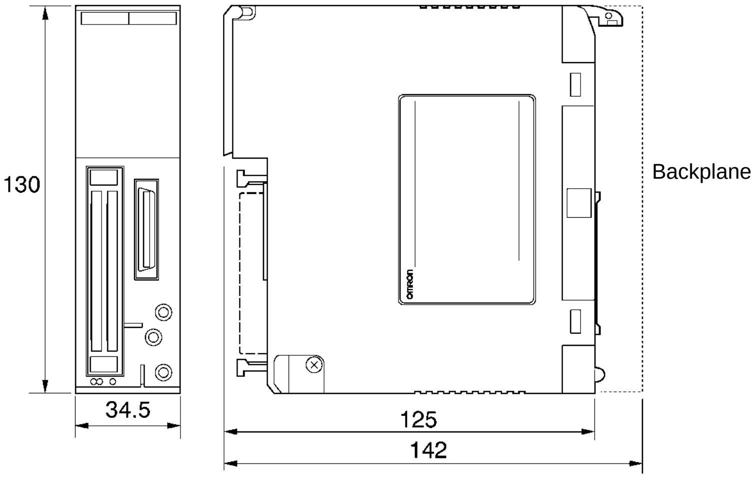
General Specifications

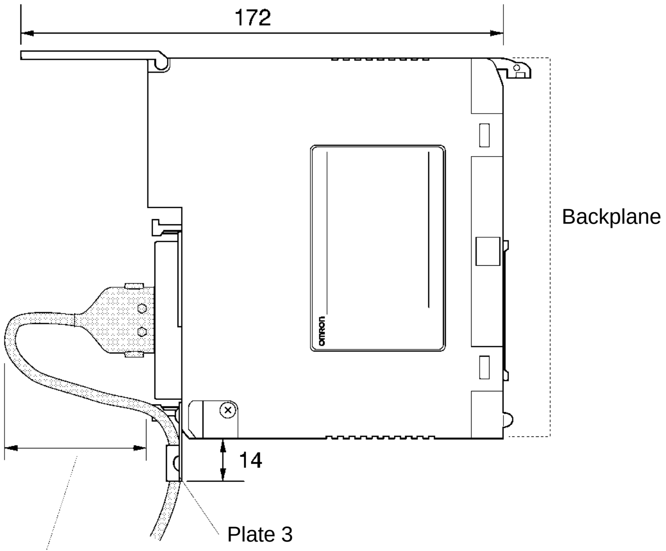
General specifications are the same as those for the SYSMAC Series.

Item	Specifications
CPU	80386SX 25 MHz
Memory	1 MB DRAM
ROM	512 KB (BIOS, DOS, System file storage)
FLASH ROM	1 MB (System file storage)
RTC	No battery backup
Serial port	RS-232C × 1 channel (terminal connection setup)
PC card interface	PCMCIA 2.1 (3.3-V low-voltage specification not supported) Type II × 2 slots or Type III × 1 slot
PC interface	SYSMAC CPU bus interface
LED indicators	RUN, ERR, PC card access, PC card formatting
Setting switch	DIP switch × 6
Power supply	+5 VDC
Current consumption	+5 VDC, 0.7 A max. (for each Unit) + PC card output current (Icard) I_{5V} (1 slot) ≦ 0.5 A, I_{12V} (1 slot) ≦ 0.1 A However, Icard = I_{5V} (2 slots) + 3.4 × I_{12V} (2 slots) ≦ 1.0 A
Dimensions	34.5 × 130 × 125 (WxHxD)
Weight	400 g max.

Dimensions

(Unit: mm)





When using an Ethernet card, be sure to allow sufficient room to install the cable.

Appendix C
Connector Pin Assignments

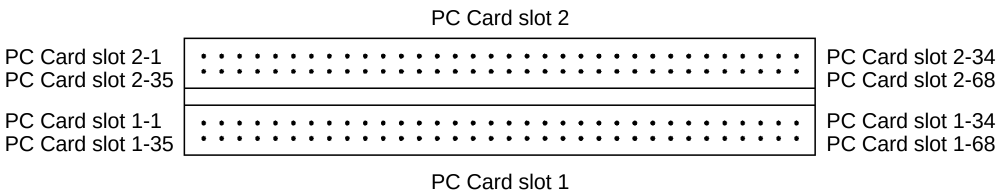
Connector Pin Assignments

The assignments of the connector pins for the PC Card Unit are shown below.

PC Card Interface

(*denotes negative logic)

Pin number	Signal names	Pin number	Signal names
1	GND	35	GND
2	D3	36	CD1*
3	D4	37	D11
4	D5	38	D12
5	D6	39	D13
6	D7	40	D14
7	CE1*	41	D15
8	A10	42	CE2*
9	E*	43	RFSH
10	A11	44	IORD*
11	A9	45	IOWR*
12	A8	46	A17
13	A13	47	A18
14	A14	48	A19
15	WE*/PGM	49	A20
16	RDY/BSY* /REQ* [I/O Card]	50	A21
17	VCC	51	VCC
18	VPP1	52	VPP2
19	A16	53	A22
20	A15	54	A23
21	A12	55	A24
22	A7	56	A25
23	A6	57	Reserved (N.C.)
24	A5	58	RESET
25	A4	59	WAIT*
26	A3	60	NPACK* [I/O Card]
27	A2	61	REG*
28	A1	62	BVD2/SPKR* [I/O Card]
29	A0	63	BVD1/STSCHG* [I/O Card]
30	D0	64	D8
31	D1	65	D9
32	D2	66	D10
33	WP/IOCS16* [I/O Card]	67	CD2*
34	GND	68	GND

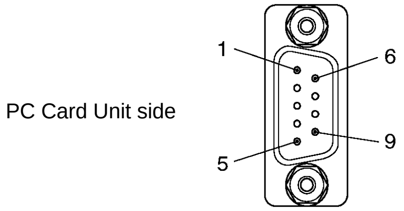


A 2-slot, 68-pin connector for a PC card (with ejector)

Serial Communications Interface

General Serial Connector Signals

Pin number	Signal names	Pin number	Signal names
1	FG	6	NC
2	SD	7	DR
3	RD	8	ER
4	RS	9	SG
5	CS	Connector base	FG



(D sub 9-pin female connector)

Cable Wiring

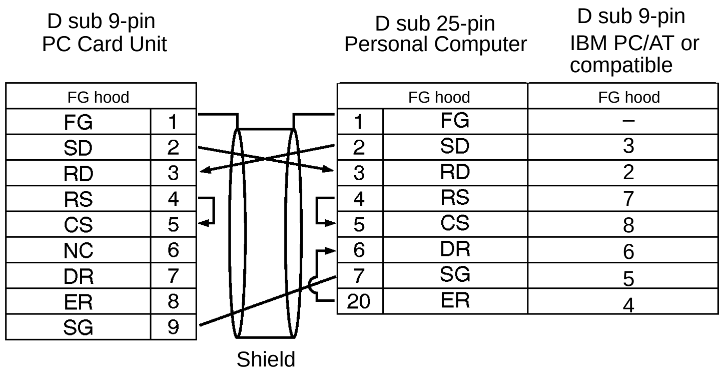
Wire the cable similarly to the following connections.

The following Connector and Connector Cover are available if you want to produce your own cable. Wire the cable with care.

Applicable Connector

Connector	XM2A-0901	OMRON
Connector Cover	XM2S-0911	OMRON

Connection

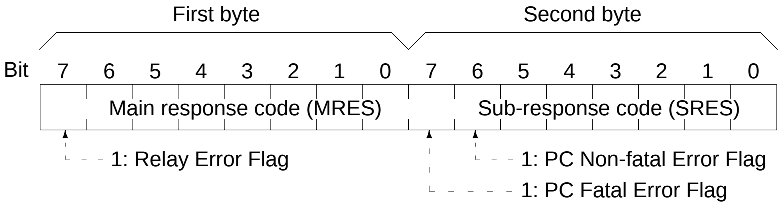


Appendix D
Response Codes from the C200HX/HG/HE CPU

The following table lists the response codes (main and sub-codes) returned after execution of FINS commands, the probable cause of the errors, and recommended remedies.

Upon receipt of some commands, the destination node will issue a request to another node; the other node is referred to as the third node.

Response codes for FINS commands consist of two bytes that indicate the result of executing a command. The structure of the response codes is shown in the following diagram.


First byte
Second byte
Bit 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0
Main response code (MRES)
Sub-response code (SRES)
1: Relay Error Flag
1: PC Non-fatal Error Flag
1: PC Fatal Error Flag

The main response code (MRES) in the first byte classifies the response and the sub-response code (SRES) in the second byte indicates details under the MRES classification. If bit 7 of the first byte is ON (1), a network relay error has occurred. Refer to *Network Relay Errors* in the *FINS Commands Reference Manual* for details on troubleshooting the error. If bit 6 or 7 of the second byte is ON, an error has occurred in the PC or computer returning the response. Refer to the operation manual for the device returning the response for details when troubleshooting the error.

Main code	Sub-code	Probable cause	Remedy
00: Normal completion	00	---	---
01: Local node error	01	Service was interrupted	Check the contents of the destination transmission area of third node.
			Check the data link status.
	01	Local node not part of network	Add to network.
	02	Token time-out, node number too large	Set the local node's node number below the maximum node number.
	03	Number of transmit retries exceeded	Check communications with internode echo test. If the test fails, check network.
	04	Maximum number of frames exceeded	Either check the execution of events in the network and reduce the number of events occurring in one cycle, or increase the maximum number of frames.
	05	Node number setting error (range)	Make sure the node number is within specified range and that there are no duplicate node numbers.
	06	Node number duplication error	Make sure that there are no duplicate node numbers.

Main code	Sub-code	Probable cause	Remedy
02: Destination node error	01	Destination node not part of network	Add to network.
	02	No node with the specified node number	Check the destination node’s node number.
	03	Third node not part of network	Check the third node’s node number.
		Broadcasting was specified.	Check the control data and specify only one node as the third node.
	04	Busy error, destination node busy	Increase the number of transmit retry attempts or re-evaluate the system so that the destination node is not so busy receiving data.
	05	Response time-out, message packet was corrupted by noise	Increase the number of transmit retry attempts.
		Response time-out, response watch-dog timer interval too short	Increase the value for the response watch-dog timer interval in the control data.
		Frame lost in transmission	Check the error log and correct the process.
03: Communications controller error	01	Error occurred in the communications controller, ERC indicator is lit	Take corrective action, referring to communications controller errors and remedies table at the end of this section
	02	CPU error occurred in the PC at the destination node	Clear the error in the CPU (refer to the PC’s operation manuals)
	03	A controller error has prevented a normal response from being returned.	Check network communications status and reset the controller board. If the error still exists, replace the controller board.
	04	Node number setting error	Make sure the node number is within specified range and that there are no duplicate node numbers.
04: Not executable	01	An undefined command has been used.	Check the command code and be sure that the Unit supports it.
	02	Cannot process command because the specified unit model or version is wrong.	Check the unit model and version.
05: Routing error	01	Destination node number is not set in the routing table.	Set the destination node number in the routing table.
	02	Routing table isn’t registered.	Set the source nodes, destination nodes, and relay nodes in the routing table.
	03	Routing table error	Set the routing table 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.
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.
	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.	Check the number of items and the data, and make sure that they agree.
	04	An incorrect command format has been used.	Check the command format of the command and set it correctly.
	05	An incorrect header has been used. (The local node’s relay table or relay node’s local network table is wrong.)	Set the routing table correctly.

Main code	Sub-code	Probable cause	Remedy
11: Parameter error	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.
	02	The access size specified in the command is wrong, or the first address is an odd number.	Set the correct access size for the command.
	03	The first address is in an inaccessible area.	Set a first address that is in an accessible area.
	04	The end of specified word range exceeds the acceptable range.	Check the acceptable limits of the data area and set the word range within the limits.
			Check the data link tables to be sure the limit to link words has not been exceeded.
	06	A non-existent program no. has been specified.	Check the program number and be sure that it is set correctly.
	09	The sizes of data items in the command block are wrong.	Check the command data and be sure that the sixes of the data items are correct.
			Check the data link tables to be sure all nodes in the refresh parameters are in the common link parameters.
	0A	The IOM break function cannot be executed because it is already being executed.	Either abort the current IOM break function processing, or wait until it is completed and execute the command.
			Check the data link tables for duplicate node numbers.
	0B	The response block is longer than the max. permissible length.	Check the command format and set the number of items correctly.
	0C	An incorrect parameter code has been specified.	Check the command data and reenter it correctly.
			Check the data link table file for corruption.
20: Read not possible	02	The data is protected.	Execute the instruction again after issuing the PROGRAM AREA PROTECT CLEAR command.
		An attempt was made to download a file that is being uploaded.	Check the file name and either interrupt servicing or wait for servicing to complete before re-executing the command.
	03	The registered table does not exist or is incorrect.	Set or reset the registered table.
		Too many files open.	Close open files and re-execute the command.
	04	The corresponding search data does not exist.	---
	05	A non-existing program no. has been specified.	Check the program number and be sure that it is set correctly.
	06	A non-existing file has been specified.	Check whether the correct file name was used.
	07	A verification error has occurred.	Check whether the memory contents are correct and replace if incorrect.
			Check the contents of the file. A read error may have occurred.

Main code	Sub-code	Probable cause	Remedy
21: Write not possible	01	The specified area is read-only or is write-protected.	If the specified area is read-only, the write cannot be performed. If it is write-protected, turn off the write-protect switch and execute the instruction again.
	02	The data is protected.	Execute the instruction again after issuing the PROGRAM AREA PROTECT CLEAR command.
		An attempt was made to simultaneously download and upload a file.	Check the file name and either interrupt servicing or wait for servicing to complete before re-executing the command.
		The data link table cannot be written manually because it is set for automatic generation.	Change the system settings to manual data link table generation.
	03	The number of files exceeds the maximum permissible.	Write the file(s) again after erasing unneeded files, or use different disk or memory card that has free space.
		Too many files open.	Close open files and re-execute the command.
	05	A non-existing program no. has been specified.	Check the program number and be sure that it is set correctly.
	06	A non-existent file has been specified.	---
	07	The specified file already exists.	Change the name of the file and execute the instruction again.
	08	Data cannot be changed.	Check the contents of the memory area being written to.
22: Not executable in current mode	01	The mode is wrong (executing).	Check the operating mode.
		Data links are active.	Check the data link status before execution.
	02	The mode is wrong (stopped).	Check the operating mode.
		Data links are active.	Check the data link status before execution.
	03	The PC is in the PROGRAM mode.	Check the PC’s mode.
	04	The PC is in the DEBUG mode.	Check the PC’s mode.
	05	The PC is in the MONITOR mode.	Check the PC’s mode.
	06	The PC is in the RUN mode.	Check the PC’s mode.
	07	The specified node is not the control node.	Check which node is the control node.
	08	The mode is wrong and the step cannot be executed.	Check whether the step has active status or not.
23: No Unit	01	A file device does not exist where specified.	Mount the memory card or disk
	02	The specified memory does not exist.	Check the specifications of the installed file memory.
	03	No clock exists.	Check the model number.
	05	Ethernet setting error. The IP address of the destination node has not been set.	Run SETUP and add the address.
24: Start/stop not possible	01	The data link table either hasn’t been created or is incorrect.	Set the data link table correctly.

Main code	Sub-code	Probable cause	Remedy
25: Unit error	02	Parity/checksum error occurred because of incorrect data.	Transfer correct data into memory.
	03	I/O setting error (The registered I/O configuration differs from the actual.)	Either change the actual configuration to match the registered one, or generate the I/O table again.
	04	Too many I/O points	Redesign the system to remain within permissible limits.
	05	CPU bus error (An error occurred during data transfer between the CPU and a CPU Bus Unit.)	Check the Unit, Service Boards, and cable connections and issue the ERROR CLEAR command.
	06	I/O duplication error (A rack number, unit number, or I/O word allocation has been duplicated.)	Check the system's settings and eliminate any duplication.
	07	I/O bus error (An error occurred during data transfer between the CPU and an I/O Unit.)	Check the Unit, Service Boards, and cable connections and issue the ERROR CLEAR command.
	09	SYSMAC BUS/2 error (An error occurred during SYSMAC BUS/2 data transfer.)	Check the Unit, Service Boards, and cable connections and issue the ERROR CLEAR command.
	0A	Special I/O Unit error (An error occurred during CPU Bus Unit data transfer.)	Check the Unit, Service Boards, and cable connections and issue the ERROR CLEAR command.
	0D	Duplication in SYSMAC BUS word allocation.	Check and regenerate the I/O table.
	0F	A memory error has occurred in internal memory, in the memory card, or in Expansion DM during the error check.	If the error occurred in internal memory or the EM Unit, correct the data in the command an execute it again. If the error occurred in a memory card or EM used for file memory, the file data has been corrupted. Execute the MEMORY CARD FORMAT command. If the above remedies do not eliminate the error, replace the faulty memory.
	10	Terminator not connected in SYSMAC BUS System.	Connect the terminator correctly.

Main code	Sub-code	Probable cause	Remedy
26: Command error	01	The specified area is not protected. This response code will be returned if an attempt is made to clear protection on an area that is not protected.	The program area is not protected, so it isn’t necessary to clear protection.
	02	An incorrect password has been specified.	Specify a password that is registered.
	04	The specified area is protected.	Execute the command again after the PROGRAM AREA PROTECT CLEAR command.
		Too many commands at destination.	The destination has received more than 5 commands. Either interrupt servicing or wait for servicing to complete before re-executing the command.
	05	The service is being executed.	Execute the command again after the service has been completed or aborted.
	06	The service is not being executed.	Execute the service if necessary.
	07	Service cannot be executed from local node because the local node is not part of the data link.	Execute the service from a node that is part of the data link.
		A buffer error has prevented returning a normal response.	Reset the board. If the error persists, replace the board.
	08	Service cannot be executed because necessary settings haven’t been made.	Make the necessary settings.
	09	Service cannot be executed because necessary settings haven’t been made in the command data.	Check the command format and make the necessary settings.
	0A	The specified action or transition number has already been registered.	Execute the command again using an action or transition number that hasn’t been registered.
	0B	Cannot clear error because the cause of the error still exists.	Eliminate the cause of the error and execute the ERROR CLEAR command.
30: Access right error	01	The access right is held by another device.	Execute the command again after the access right has been released. (The command can be executed after the ACCESS RIGHT FORCED ACQUIRE or ACCESS RIGHT RELEASE command is completed. Releasing the access right might affect processes in progress at the node that held the access right.)
40: Abort	01	Command was aborted with ABORT command.	---

Appendix E
FINS Response Codes from the PC Card Unit

When an error occurs in a FINS command sent to a PC Card Unit, an error response code is returned. Although there are other response codes that are given only for certain commands, the following is a list of response codes common to all commands.

Response code	Cause
0103	Transmission buffer full; transmission not possible.
0201	Destination IP address not recognized; transmission not possible.
0202	Specified Unit nonexistent; transmission not possible.
0301	Network error; transmission not possible.
0401	Command frame not supported (MRC, SRC).
0401	Unsupported FINS frame (Protocol) received.
0501	Routing table setting error; transmission not possible.
0502	Routing table nonexistent; transmission not possible.
0503	Routing table error; transmission not possible.
1005	FINS frame with an incorrect destination received.
1001	Transmission size exceeds 1472 bytes.
2305	IP address cannot be converted; transmission not possible.

Appendix F
Differences with the CV-series or CS1-series Ethernet Units

The differences between the PC Card Unit (C200HW-PCS01-V2) and the CV-series or CS1-series Ethernet Units are listed below.

Functions Not Supported by the PC Card Unit

- There is no FTP Server function.
- CMND instructions cannot be used.
- 10BASE-5 is not supported.

Differences in Standard Protocol, Socket Service Functions

- The PC Card Unit has a total of four TCP/IP and UDP/IP sockets, whereas the CV-series and CS1-series Ethernet Units have eight UDP/IP sockets.
- The PC Card Unit uses the CMCR instruction for FINS communications, whereas the CV-series and CS1-series Ethernet Units use the CMND instruction.

Differences in FINS Command/Response Communications

- When commands are transmitted from the Programmable Controller to the host, the PC Card Unit uses the CMCR instruction, whereas the CV-series and CS1-series Ethernet Units use the CMND instruction.
- When commands from the host are received at the Programmable Controller, the PC Card Unit cannot receive multiple FINS commands simultaneously from multiple nodes. After receiving a command, it processes the command and sends a response to the host. If it receives another command before that process has been completed, it returns a “node busy” error message.

Differences in Ethernet Connectors

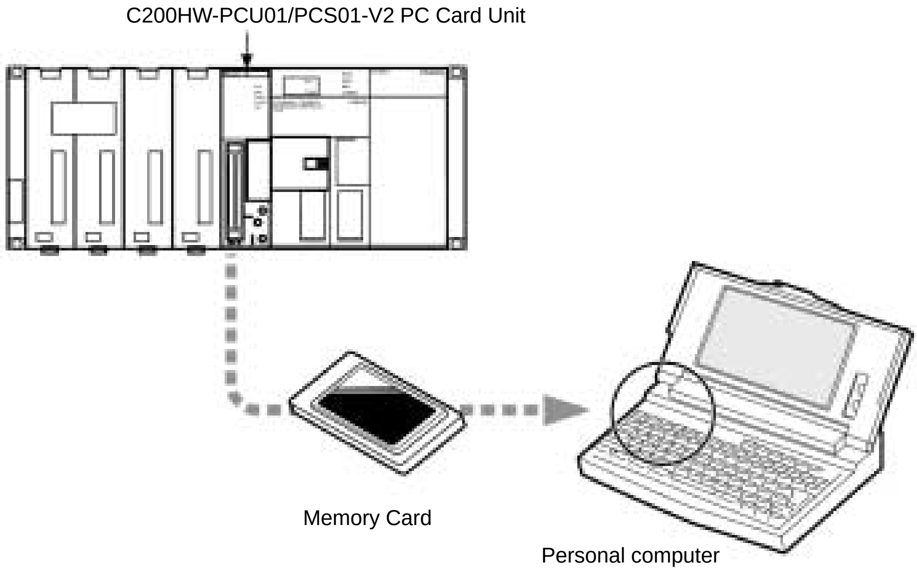
- The PC Card Unit supports 10Base-T and 10Base-2 (using commercially available PC Cards).
- The CV-series and CS1-series Ethernet Units support 10Base-5.

Differences in Mail Support

- The PC Card Unit and CV-series Ethernet Unit do not support sending mail.
- The CS1-series Ethernet Unit does support sending mail.

Appendix G
Example Using a Memory Card with a Personal Computer

The procedure for reading a file saved on the PC Card Unit in comma-separated format, on the personal computer in EXCEL is described below.



The personal computer operating environment used in this example is as follows:

- MS-Windows 95
- EXCEL
- Built-in PC Card driver

1, 2, 3...

1. Insert the Memory Card created on the PC Card Unit into the PC Card slot on the personal computer.
2. In EXCEL select “Open” from the “File” menu.
3. Set the file type to “Text file.”
4. Select “Comma or space separated file.”The file will be read as a comma-separated file.

Note Only numerical values in decimals can be read.

Appendix H
Precautions when Setting Up the Network

- Be sure to follow the safety precautions and specifications when setting up the Ethernet network. Refer to *ISO/IEC 8802-3* for laying the Ethernet.
- It is recommended that network construction be carried out by qualified personnel with an extensive knowledge of safety precautions and specifications.
- Do not install Ethernet network equipment in the vicinity of noisy equipment. When installing in a noisy environment, ensure that each individual network device is in a metal case and that the noise procedures for all optical cables is followed.
- When executing transmission using the UDP/IP, whose communications protocol doesn't perform re-transmission processing (FINS communications), be sure to execute re-transmission using the user program.

Appendix I
Contents of Version Upgrade

This appendix outlines the functions that have been added with upgrades.

Upgrading from C200HW-PCS01 to C200HW-PCS01-EV1

Socket Interface Support

UDP and TCP socket interfaces are supported by FINS commands. Refer to *Section 9 FINS Commands* and *Section 10 Socket Services*.

Remote Tool Connection

Remote tool connection to the C200HX/HG/HE via the PC Card Unit interface is made possible by peripheral bus connection to the C200HX/HG/HE and host link connection from SYSMAC Support Software (SSS).

With remote tool connection via PC Card Unit Ethernet, it is not possible to use SYSMAC Link-related functions such as starting and stopping SYSMAC Link and making data link settings.

Refer to *Appendix J Remote Tool Connection Procedure*.

FINS Command Expansion

- 0501: Added data such as local IP address, FINS UDP port number, etc., to response.
- 0801: Added loopback command.
- Added commands for socket interface.

Refer to *Section 11 Using FINS Commands and Responses*.

Setup Software Expansion

- Added function for reading Ethernet settings.
- Added function for clearing Ethernet settings.

Refer to *7-7 Using SETUP2.EXE*.

FINS Address Checks

When a message is received from a node that is not registered on the conversion table for IP addresses and FINS node addresses, the received message is processed but the response is discarded. Unit operation continues.

Upgrading from C200HW-PCS01-EV1 to C200HW-PCS01-V2

Installation of Setting Files

Files required for Ethernet communications functions have been installed in the Unit.

Refer to *7-6 Setup Software Operation*.

Note Do not use the setup software for the C200HW-PCS01(-EV1) to perform settings for the C200HW-PCS01-V2.

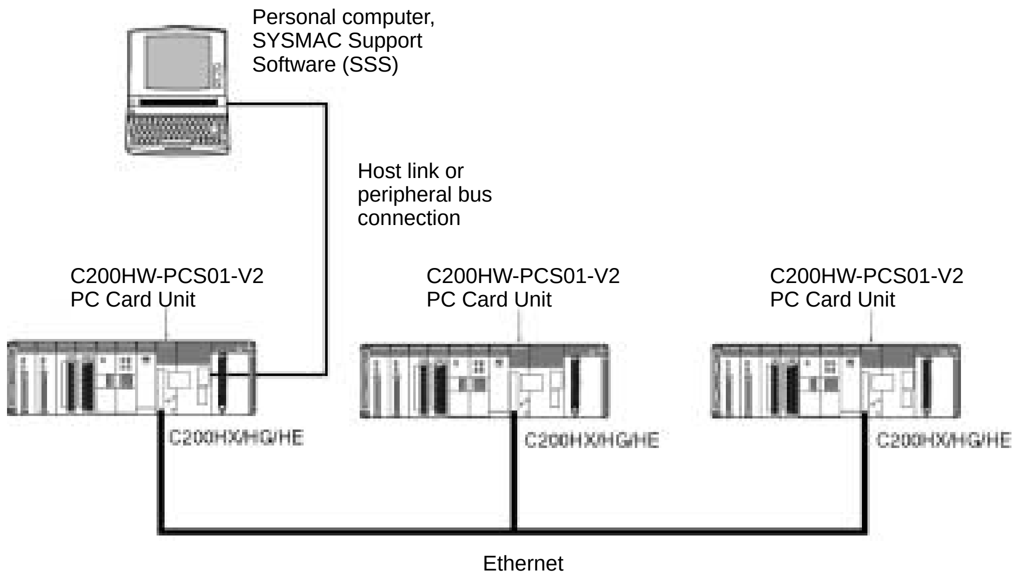
Appendix J
Remote Tool Connection Procedure

This appendix explains the procedure for remote tool connection to the C200HX/HG/HE via the PC Card Unit interface by peripheral bus connection to the C200HX/HG/HE and host link connection from SYSMAC Support Software (SSS).

Outline

When the system is configured as shown in the following diagram, remote tool connection can be executed for a C200HX/HG/HE on the Ethernet.

System Configuration Example



- Remote tool connection can only be executed between C200HX/HG/HE Programmable Controllers with PC Card Units mounted. It cannot be executed for other C-series or CV-series PCs.
- Remote tool connection cannot be executed unless the Ethernet environmental settings are completed for the PC Card Units at both the local and remote nodes.
- Personal computers with SYSMAC Support Software installed can be connected to C200HX/HG/HE Programmable Controllers by host link or by peripheral bus. If a personal computer is directly connected to a PC Card Unit, peripheral bus connection cannot be used.

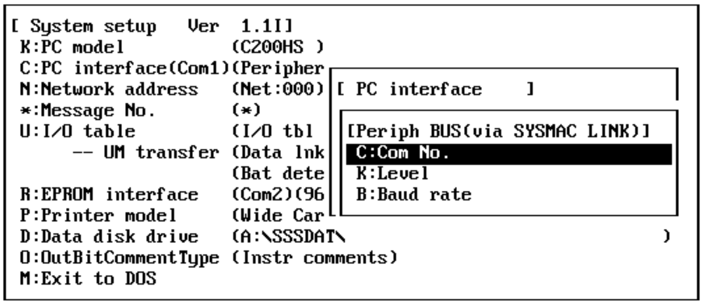
Restrictions

With remote tool connection via PCU Card Unit Ethernet, it is not possible to use SYSMAC Link-related functions such as starting and stopping SYSMAC Link and making data link settings.

Settings for Peripheral Bus

Use the SYSMAC Support Software to make the settings as shown in the following procedure.

1, 2, 3...
1. Select “C:PC interface” from the System Setup Menu.
2. Select “L:Peripheral Bus (via SYSMAC LINK).”



```
[ System setup   Ver  1.1I]
 K:PC model          (C200HS )
 C:PC interface(Com1)(Peripher
 N:Network address   (Net:000) [ PC interface    ]
 *:Message No.       (*)
 U:I/O table         (I/O tbl   [Periph BUS(via SYSMAC LINK)]
      -- UM transfer (Data lnk   C:Com No.
                     (Bat dete   K:Level
 R:EPROM interface   (Com2)(96   B:Baud rate
 P:Printer model     (Wide Car
 D:Data disk drive   (A:\SSSDAT\                          )
 O:OutBitCommentType (Instr comments)
 M:Exit to DOS
```

3. Select “K:Level” or “B:Baud rate.”

```
[ Level ]
0: Level #0
1: Level #1
```

```
[ Baud rate ]
A: 19,200
B: 9,600
C: 4,800
```

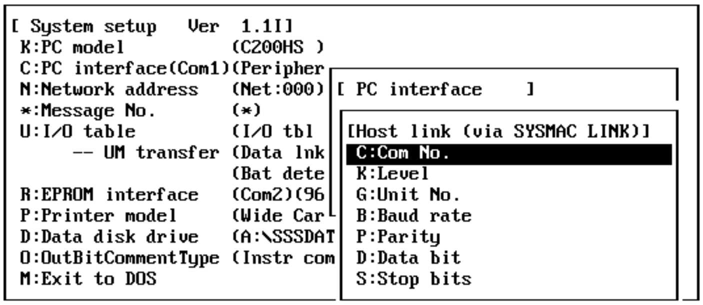
“Level” is the system of the PC Card Unit connected to the C200HX/HG/HE.

Settings for Host Link

Use the SYSMAC Support Software to make the settings as shown in the following procedure.

Note When communicating from a C200HX/HG/HE Communications Board, make sure that the SYSMAC LINK Peripheral Device Initialization Bit (AR 2403) is turned ON when the SYSMAC Support Software connection port is changed.

1, 2, 3...
1. Select “C:PC interface” from the System Setup Menu.
2. Select “C:Host Link (via SYSMAC LINK).”



```
[ System setup   Ver  1.1I]
 K:PC model          (C200HS )
 C:PC interface(Com1)(Peripher
 N:Network address   (Net:000) [ PC interface    ]
 *:Message No.       (*)
 U:I/O table         (I/O tbl   [Host link (via SYSMAC LINK)]
      -- UM transfer (Data lnk   C:Com No.
                     (Bat dete   K:Level
 R:EPROM interface   (Com2)(96   G:Unit No.
 P:Printer model     (Wide Car   B:Baud rate
 D:Data disk drive   (A:\SSSDAT  P:Parity
 O:OutBitCommentType (Instr com  D:Data bit
 M:Exit to DOS                   S:Stop bits
```

3. Set the level, baud rate, number of data bits, the host link unit number, parity, and number of stop bits as required.

```
[ Level ]
0: Level #0
1: Level #1
```

```
[ Baud rate ]
A: 19,200
B: 9,600
C: 4,800
D: 2,400
```

```
[ Data bit ]
7: 7 bits
8: 8 bits
```

```
[ Unit No. ]
00 unit
          (00 to 31)
```

```
[ Parity ]
E: Even parity
O: Odd parity
N: No parity
```

```
[ Stop bits ]
1: 1 bit
2: 2 bits
```

Note This procedure cannot be used with the base-mounted C200H-LK201-(V1) Host Link Unit.

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.

Bus Connection Unit A Unit used to connect a single PC Card Unit to the C200HX/HG/HE PC or to one SYSMAC LINK Unit or SYSMAC NET Link Unit to the C200HX/HG/HE PC together with one PC Card Unit.

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.

clear The process of turning a bit or signal OFF.

client A process or node requesting processing from a server.

Communications Board A board that is installed into a C200HX/HG/HE PC enabling communications with peripheral devices or Special I/O Units via RS-232C, RS-422, or RS-485.

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.

CSV File format which allows data to be processed with commercially available spreadsheet software. Supported by the PC Card Unit.

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.

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.

echo 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.

Ethernet card A card which provides Ethernet capability to the Unit or hardware in which it is installed.

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.

memory card A data storage media similar to a floppy disk.

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 card Memory card used with the PC Card Unit. PCMCIA 2.1-compliant PC cards and/or memory cards on the market, such as SRAM, ATA, and FLASH memory cards, can be used.

PC Card Unit This Unit provides various functions for using PC cards with SYSMAC C200HX/HG/HE Programmable Controllers. CIO, DM, and EM data (but not the user program) can be loaded and saved between C200HX/HG/HE Programmable Controllers and Memory Cards inserted in the PC Card Unit.

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.

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 that can be used for data calculation or other manipulation in programming, i.e., a ‘work space’ in memory. Also see *work word*.

work word A word that can be used for data calculation or other manipulation in programming, i.e., a ‘work space’ in memory. A large portion of the IR area is always reserved for work words. Parts of other areas not required for special purposes may also be used as work words.

write-protect A state in which the contents of a storage device can be read but cannot be altered.

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.

Index

Revision History

A manual revision code appears as a suffix to the catalog number on the front cover of the manual.

Cat. No. V217-E1-1

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	February 2001	Original production

END USER LICENSE AGREEMENT

IMPORTANT – READ CAREFULLY BEFORE USING THE EMBEDDED SYSTEM WHICH CONTAINS MICROSOFT SOFTWARE. By using the embedded system containing software, you indicate your acceptance of the following Software License Agreement.

———————————————————————————

SOFTWARE LICENSE AGREEMENT

(Embedded Products)

This software license agreement, including the Warranty and Special Provisions set forth in the appendix or separate booklet included in this package, is a legal agreement between you (either an individual or an entity, hereinafter "End User") and the manufacturer ("Embedded System Manufacturer") of the embedded system containing software product. By using the embedded system on which software program(s) have been preinstalled ("SOFTWARE"), you are agreeing to be bound by the terms of the agreement.

1. GRANT OF LICENSE. This License Agreement permits you to use the Microsoft SOFTWARE as preinstalled on the embedded system.

2. INTELLECTUAL PROPERTY. ISA CONTROL UNIT / PC CARD UNIT contains intellectual property, i.e. software programs, that is licensed for the end user customer's use (hereinafter "End User"). This is not a sale of such intellectual property. The End User shall not copy, disassemble, reverse engineer, or decompile the software program.

3. COPYRIGHT. The SOFTWARE is owned by Microsoft Corporation or its suppliers and is protected by United States copyright laws and international treaty provisions and all other applicable national laws. Therefore, you must treat the SOFTWARE like any other copyrighted material (e.g., a book or musical recording).

4. U.S. GOVERNMENT RESTRICTED RIGHTS. The SOFTWARE and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the United States Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of The Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software -- Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Microsoft Corporation/One Microsoft Way/Redmond, WA 98052-6399.

Please see the Warranty and Special Provisions for information concerning governing law.

Product support for the SOFTWARE is not provided by Microsoft Corporation or its subsidiaries. For product support, please refer to Embedded System Manufacturer's support number provided in the documentation for the embedded system. Should you have any questions concerning this Agreement, or if you desire to contact Embedded System Manufacturer for any other reason, please refer to the address provided in the documentation for your embedded system.

FOR THE LIMITED WARRANTY AND SPECIAL PROVISIONS PERTAINING TO YOUR COUNTRY, PLEASE REFER TO EMBEDDED SYSTEM DOCUMENTATION OR THE WARRANTY AND SPECIAL PROVISIONS BOOKLET INCLUDED IN THIS PACKAGE.

APPENDIX

WARRANTY AND SPECIAL PROVISIONS

LIMITED WARRANTY

LIMITED WARRANTY. Embedded System Manufacturer warrants that (a) the SOFTWARE will perform substantially in accordance with the accompanying written materials for a period of ninety (90) days from the date of receipt. Any implied warranties on the SOFTWARE are limited to ninety (90) days. Some states/jurisdictions do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you.

CUSTOMER REMEDIES. Embedded System Manufacturer's and its suppliers' entire liability and your exclusive remedy shall be, at Embedded System Manufacturer's option, either (a) return of the price paid, or (b) repair or replacement of the SOFTWARE that does not meet the above Limited Warranty and which is returned to Embedded System Manufacturer with a copy of your receipt. This Limited Warranty is void if failure of the SOFTWARE has resulted from accident, abuse, or misapplication. Any replacement SOFTWARE will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

NO OTHER WARRANTIES. THE MICROSOFT SOFTWARE PROGRAMS ARE PROVIDED TO THE END USER "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK OF THE QUALITY AND PERFORMANCE OF THE SOFTWARE PROGRAM IS WITH YOU.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES. EMBEDDED MANUFACTURER'S SUPPLIERS SHALL NOT BE HELD TO ANY LIABILITY FOR ANY DAMAGES SUFFERED OF INCURRED BY THE END USER (INCLUDING, BUT NOT LIMITED TO, GENERAL, SPECIAL, CONSEQUENTIAL OR INCIDENTAL DAMAGES INCLUDING DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION AND THE LIKE), ARISING FROM OR IN CONNECTION WITH THE DELIVERY, USE OR PERFORMANCE OF THE SOFTWARE PROGRAM.

SPECIAL PROVISIONS

This Software License Agreement and Warranty are governed by the laws of the State of Washington, U.S.A.