

## T2N Ethernet Port

---

### Instruction Manual

---

#### REQUIREMENTS

- Keep this instruction manual where it can be easily referred to by users and those responsible for the equipment.
- Read this instruction manual carefully before using the equipment.
- After this instruction manual has been read, keep it beside the equipment.

[Main Menu](#)  
[Contents](#)

**July, 1998**

**TOSHIBA CORPORATION**

© Copyright 1998 by Toshiba Corporation Tokyo, Japan

Toshiba Corporation reserves the right to make changes and improvements to this publication and/or related products at any time without notice. No obligation shall be incurred, except as noted in this publication.

This publication is copyrighted and contains proprietary material. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means — electrical, mechanical, by photocopying, recording, or otherwise — without obtaining prior written permission from Toshiba.

### **Important Information**

Misuse of this equipment can result in property damage or human injury. Because controlled system applications vary widely, you should satisfy yourself as to the acceptability of this equipment for your intended purpose. In no event will Toshiba Corporation be responsible or liable for either indirect or consequential damage or injury that may result from the use of this equipment.

No patent liability is assumed by Toshiba Corporation with respect to the use of information, illustration, circuits equipment, or application examples in this publication.

# Safety Precautions

Thank you for purchasing the Toshiba PU235N/245N CPU module, which is for use with the T2N programmable controller.

This document describes the usage and handling of the PU235N/245N built-in Ethernet Port and presents important cautionary items.

To use the PU235N/245N Ethernet Port safely, please read this document and all other related documentation thoroughly and acquire thorough knowledge of the equipment and points that require care before installing, operating, maintaining, or inspecting the PU235N/245N.

## [Concerning Important Points]

1. The PU235N/245N is designed and manufactured for use with general industrial equipment, including all types of production line control and manufacturing equipment.

The PU235N/245N is not designed or manufactured for use in equipment or systems on which human lives depend.

Consult with your Toshiba sales representative in advance if you need to use an PU235N/245N in a equipment or systems that are used in transportation equipment (such as trains), medical applications, aerospace applications, nuclear reactor control, or underwater relay equipment or other special contexts.

2. Although the PU235N/245N is manufactured under strict quality control, if an PU235N/245N is to be used in critical equipment in which human lives could be endangered or in which large losses could occur should the PU235N/245N fail, install safety equipment so that such catastrophes cannot occur.
3. The PU235N/245N should only be installed, wired, used, or maintained by persons who have a general knowledge of the handling of control equipment. Electrical shock, fires, failures, and malfunctions can occur if an PU235N/245N is handled incorrectly. Persons whose knowledge of control and electrical equipment is inadequate should avoid installing, wiring, using, or maintaining this equipment, and should instead employ persons with the requisite specialized knowledge.
4. This document and the related separate documentation is written for persons with a general knowledge of programmable controllers and control equipment.

If you have any questions about the contents of this documentation, please feel free to contact you Toshiba representative.

**Safety Precautions (continued)**

**[Warning indications]**

This instruction manual provides important information to allow these products to be used safely and correctly, to prevent in advance dangers to people managing or using this equipment or other persons, and to prevent in advance losses to property. Be sure you thoroughly understand the displays and symbols described below and follow the cautionary items and warnings included in this document carefully.

Store this document carefully so that you can refer to it at any time even after you have read it.

**Description of Safety Signs**

Safety sign	Meaning
 <b>WARNING</b>	Indicates a potentially hazardous situation which could result in death or serious injury if you do not follow the instructions.
 <b>CAUTION</b>	Indicates a potentially hazardous situation which may result in minor or moderate injury and/or property damage if you do not follow the instructions.

**Explanations of Safety Symbols**

Safety symbol	Meaning
	This means "Hazard Alert" or "Beware of Hazard". The exact nature of the hazard is shown in the triangle or described in text near the triangle.
	This is a safety alert symbol indicating a potential hazard. This symbol is used together with a signal word.

**Safety Precautions (continued)**

- **Precautions for operation**

**WARNING**

1. Build emergency stop, interlock, and similar circuits outside the PU235N/245N.

If a breakdown or malfunction occurs in the PU235N/245N, there is the danger that someone could be hurt or fatally injured. Also, other equipment could be damaged or a breakdown could occur.

T2N User's Manual - Hardware

3. Preparing for Operation (Hardware)

**CAUTION**

1. Take all necessary safety precautions before performing operations such as changing a running program, forcing output, or switching to RUN (operate) or HALT (stop) modes.

Equipment can be damaged and breakdowns can be caused by operating this equipment incorrectly.

T2N User's Manual - Hardware

Program Operation Manual

3. Preparing for Operation (Hardware)

### Safety Precautions (continued)

- **Mounting in the base unit**



#### CAUTION

1. The Ethernet Port is built-in the PU235N/245N. Do not remove the Ethernet Port from a PU235N/245N.
2. The PU235N/245N itself should only be mounted or removed when all power is turned off. Similarly, connections to the terminal block should only be made or removed when all power is turned off. Not observing this precaution could result in electrical shocks, malfunctions, or damage to the PU235N/245N or other equipment.
3. Do not allow foreign objects such as wire shreds to get into the PU235N/245N. This could result in fire, breakdown, or malfunction.
4. Verify that connectors, cables, and the mounting of the PU235N/245N itself in the base unit are all secured and that there is no play, missing screws, or disconnections in any of these parts.  
  
If any of these parts is inadequately secured, breakdown or malfunction could occur as the result of vibration.

3.3 Mounting in the Base Unit

- **Connecting to the network**



#### CAUTION

1. Do not connect or disconnect cables when either the T2N or HUB power supply is turned on. This can damage the PU235N/245N or cause it to malfunction.
2. When installing a 10BASE-T network, the cabling must be laid with adequate safety precautions and in accordance with all applicable standards. Installation and wiring should be performed by a qualified professional.  
  
See the **ISO/IEC8802-3** standard for installation environment standards.

3.1 Connecting to the Network

**Safety Precautions (continued)****Concerning:**

- **Preparing for operation (software)**
- **Computer link protocol and the PC link protocol transmission**
- **Socket interface communication**
- **RAS information**

**CAUTION**

1. Chapters 4 - 7 present information related to using the functions provided by the Ethernet Port from a T2N, including the instruction (request) format, important items that require attention, and sample programs. That chapter also presents items considered necessary when using the Ethernet Port.  
  
Make a point of understanding the content of chapter 4 thoroughly before writing programs that use the Ethernet Port. The sample programs present basic examples of Ethernet Port usage, and should be reviewed carefully before use in an actual system.
  4. Preparing for operation (software)
  5. Computer link protocol and the PC Link protocol transmission
  6. Socket interface communication
  7. RAS information

- **Installation environment and base unit mounting**

**CAUTION**

1. Only use this equipment in the environment described in the T2N User's Manual - Hardware.  
Use in any other environment can cause electrical shocks, fires, failures, and malfunctions.
  2. Mount this equipment according to the mounting method specified in the T2N User's Manual - Hardware.  
  
Mounting in any direction other than the specified direction, or defective or inadequate mounting, can result in the equipment falling, fires, failures, and malfunctions.
- 9.1 Installation Environment and Base Unit Mounting

## Safety Precautions (continued)

- Basic points in laying out a network

 CAUTION
<ol style="list-style-type: none"><li>1. When laying network cable, take adequate safety measures and perform all operations in accordance with the relevant standards. See the <b>ISO/IEC8802-3</b> standard for details on laying networks.</li><li>2. We strongly recommend hiring a qualified contractor who has specialized knowledge about safety precautions and standards.</li><li>3. Avoid installing 10BASE5, 10BASE2 or 10BASE-T networks in environments that have high noise levels. If it is necessary to install network cables in an environment subject to noise (electromagnetic interference), use the noise reduction measures described elsewhere in this document.</li></ol>
9.2 Network Wiring

- Maintenance

 CAUTION
<ol style="list-style-type: none"><li>1. Always turn off the power before attaching or removing modules or cables. Electrical shock, malfunctions, and breakdowns are possible if these operations are performed with the power on.</li><li>2. To keep the system operating normally at all times and to prevent trouble in advance, perform daily inspections, periodic inspections, and cleaning.</li><li>3. Refer to the section on error recovery in this document if the Ethernet Port does not operate correctly. Contact your local Toshiba dealer or service representative if a breakdown occurs, and request that the module be repaired or returned to Toshiba. Operation and safety cannot be guaranteed if service is performed by anyone other than an authorized Toshiba service representative.</li><li>4. Do not disassemble or modify the module hardware, and do not modify the module software. Breakdowns or malfunctions may lead to fire, electrical shock, or injury.</li><li>5. Do not continue to use the module if it emits smoke or unusual odors, or if it is in an abnormal state of any type. Using the module in an abnormal state can lead to malfunctions, breakdowns, and electrical shock. In such cases, immediately turn off all power and contact your local Toshiba dealer or service representative. Customers should never attempt repairs or modifications to this equipment themselves: repair and modification operations are extremely dangerous.</li></ol>
Appendix 1 Maintenance and Inspection

# Usage Recommendations

This section collects the knowledge and operations that should be known by all users of this equipment. Read this section carefully in conjunction with the safety precautions, thoroughly master knowledge of the equipment, safety information, and the items that require attention, and use the equipment correctly.

- **Support for 10BASE-T network**

Usage Recommendations
1. Do not exceed the maximum cable length (100 m) in systems using 10BASE-T. 1.3 Basic Functionality

- **Ethernet Port registration**

Usage Recommendations
1. Before performing the settings described in this chapter, register the Ethernet Port as T2N I/O. The I/O type is ETHER (PU235N) or E/S (PU245N). 4.1 Ethernet Port Setup Flowchart (Software)

- **Restrictions on subnet masks**

Usage Recommendations
1. The Ethernet Port does not allow the user to set the subnet mask. The Ethernet Port that the network address number is 1 can use the subnet mask. Therefore the Ethernet Port cannot be used in systems that operate a subnetwork. 4.4 Parameter Setup Request

**Usage Recommendations (continued)**

- **Computer link protocol transmission**

<b>Usage Recommendations</b>	
1.	Since UDP/IP does not provide control functions to guaranteed communication reliability, higher level protocols (such as retry) are required.
2.	No processing is performed if a computer link protocol and a PC link protocol transmission messages are received at any point other than a message transmission UDP socket.
	5.1 Computer Link Protocol Transmission
	5.3 PC Link Protocol Transmission (Data write)
	5.4 PC Link Protocol Transmission (Data read)

- **Limitations on the T2N SEND and RECV instructions**

<b>Usage Recommendations</b>	
When using the SEND and RECV instructions:	
1.	When using fixed period interrupt programs, there are limitations on the number of words that can be specified.
	Appendix 4 Limitations on the T2N SEND and RECV Instructions

# Before Reading This Document

Thank you for purchasing a Toshiba PROSEC T2N multi-purpose programmable controller.

This manual describes the T2N Ethernet Port specifications and usage, and presents sample programs for the Ethernet Port used with the PROSEC T2N.

The PU235N has Ethernet Port and the PU245N has Ethernet and TOSLINE-S20LP Port.

The T2N Ethernet Port is built-in a T2N and the T3H Ethernet module is independent module. It is called “port” with this manual and called “station” with Ethernet Module EN311 for T3H Instruction Manual are same meaning. This manual describes port.

When using this product, use the product correctly based on a thorough reading of this manual. When using a T3H Ethernet module, see “Ethernet Module EN311 for T3H Instruction Manual”. This document is organized as follows.

## **Chapter 1: Ethernet Port Overview**

Presents an overview of the PU235N/245N Ethernet Port, including its functions, specifications, and systems in which it can be used. Refer to this chapter to understand the basic performance of the Ethernet Port.

## **Chapter 2: Ethernet Port Parts and Functions**

Describes the function and names of the components of the Ethernet Port. Since this chapter presents information required to understand the hardware settings described in the next chapter, read this chapter carefully and use the Ethernet Port correctly.

## **Chapter 3: Preparing for Operation (Hardware)**

Describes the hardware preparations and settings required to operate the Ethernet Port normally.

## **Chapter 4: Preparing for Operation (Software)**

Describes the software settings required to operate the Ethernet Port normally, and presents sample programs.

## **Chapter 5: Computer Link Protocol and the PC Link Protocol Transmission**

Describes the instruction format used with computer link protocol and PC link protocol transmission, and presents sample programs.

## **Chapter 6: Socket Interface Communication**

Presents an overview of the socket interface, cautionary items, and the instruction format used on the Ethernet Port with the socket interface, and presents sample programs.

## **Chapter 7: RAS Information**

Describes the instruction format and the format of the Ethernet Port RAS information (including port status and down information), and presents sample programs.

## **Chapter 8: Error Handling**

Describes techniques for determining the causes of errors when Ethernet Port operation is abnormal, and procedures for recovering from those errors.

## Chapter 9: Installation and Wiring

Describes the procedures for installing the Ethernet Port (and T2N), procedures for laying transmission cable, and the nature of the construction required.

## Appendix

Presents request code, completion status value, and error code tables and describes the minimum transmission delay times and execution times for each instruction.

Note that in addition to this manual, Toshiba also provides a T2N User's Manual - Basic Hardware and Function, a T-series Instruction Set, and a T-series Computer Link Operation Manual. All these manuals should be read together.

- T2N User's Manual - Basic Hardware and Function (UM-TS02\*\*\*-E001) Describes the configuration, specifications, installation and wiring techniques, maintenance, and preventive maintenance procedures for the T2N base system hardware and describes the function provided by the T2N CPU and the use of that CPU, and presents information required for creating user programs for the T2N.
- T-series Instruction Set (UM-TS03\*\*\*-E004) Describes the detailed specifications of the instruction words for the ladder and SFC program languages, which are two of the programming languages supported on the T2N.
- T-series Computer Link Operation Manual (UM-TS03\*\*\*-E008) Describes the specifications and use of computer link protocol transmission, which is built into the T2N CPU.

### Registered trademarks:

Ethernet is a registered trademark of Xerox Corporation.

Cheapernet is not a registered trademark, but rather is a common name.

3 Com and Etherlink III are trademarks of 3 Com Corporation.

PC-NFS Pro is trademarks of Sun Microsystems, Inc..

Visual Ctt is trademarks of Microsoft Corporation.

Microsoft is trademarks of Microsoft Corporation.

UNIX is trademarks of UNIX System Laboratories, Inc..

# Contents

<b>Safety Precautions</b> .....	<b>2</b>
<b>Usage Recommendations</b> .....	<b>8</b>
<b>Before Reading This Document</b> .....	<b>10</b>
<b>1. Ethernet Port Overview</b> .....	<b>14</b>
1.1 System Configuration Using the Ethernet Port.....	14
1.2 LAN Standards and Terminology .....	15
1.3 Basic Functionality .....	16
1.4 Ethernet Port Specifications.....	21
1.5 Software Organization .....	22
<b>2. PU235N/245N Parts and Functions</b> .....	<b>24</b>
2.1 Front Panel.....	24
2.2 Component Functions.....	25
<b>3. Preparing for Operation (Hardware)</b> .....	<b>26</b>
3.1 Connecting to the Network.....	27
3.2 Network Wiring Equipment.....	28
<b>4. Preparing for Operation (Software)</b> .....	<b>31</b>
4.1 Ethernet Port Setup Flowchart (Software) .....	31
4.2 Ethernet Port Instruction Specifications.....	32
4.3 Ethernet Port Reset.....	39
4.4 Parameter Setup Request.....	41
4.5 Operating Mode Control Requests.....	46
<b>5. Computer Link Protocol and the PC Link Protocol Transmission</b> .....	<b>50</b>
5.1 Computer Link Protocol Transmission .....	50
5.2 Sample Computer Link Protocol Program .....	51
5.3 PC Link Protocol Transmission (Data write) .....	57
5.4 PC Link Protocol Transmission (Data read).....	61
5.5 Sample PC Link Protocol Transmission Program.....	64

---

<b>6. Socket Interface Communication</b> .....	<b>68</b>
6.1 Overview .....	68
6.2 Ethernet Port Socket Interface Usage Notes.....	72
6.3 Socket Interface Information .....	82
6.4 Using the Ethernet Port Socket Interface .....	84
6.5 Sample Programs .....	112
<b>7. RAS Information</b> .....	<b>122</b>
7.1 T2N Special Relays and Registers.....	123
7.2 Down Information .....	125
7.3 Remote Port Verification Request.....	126
7.4 Remote Port Loopback (for an Ethernet Port) .....	128
7.5 Time Setting .....	130
7.6 RAS Information Readout .....	132
<b>8. Error Handling</b> .....	<b>140</b>
8.1 LED Display .....	140
8.2 Status (error status) .....	143
8.3 T2N Special Relays and Registers .....	146
8.4 Socket Interface Information .....	147
8.5 Troubleshooting .....	148
<b>9. Installation and Wiring</b> .....	<b>163</b>
9.1 Installation Environment and Base Unit Mounting.....	163
9.2 Network Wiring .....	164
<b>Appendix</b>	
Appendix 1 Maintenance and Inspection .....	169
Appendix 2 Request Code Table .....	171
Appendix 3 Status Table .....	172
Appendix 4 Limitations on the T2N SEND and RECV Instructions .....	175

# 1. Ethernet Port Overview

## 1.1 System Configuration Using the Ethernet Port

This section describes the systems that can be configured using the PU235N/245N Ethernet Port for the PROSEC T2N programmable controller. The remainder of this document refers to the PROSEC T2N programmable controller as the T2N, and the Ethernet Port as the PU235N/245N Ethernet Port.

The Ethernet Port is a communication port for connecting a T2N system to a 10BASE-T (Twisted pair Ethernet) LAN (local area network). It is operated as a T2N option module. The T2N connects to the above mentioned LAN through the Ethernet Port and exchanges data with host computers (workstations or personal computers) or other programmable controllers on the LAN.

The remainder of this section presents a system that uses the Ethernet Port. In this example, the T2N is connected to host machines such as workstations and personal computers over a 10BASE-T LAN and the T2N is connected to lower level controllers using a Toshiba proprietary control LAN.

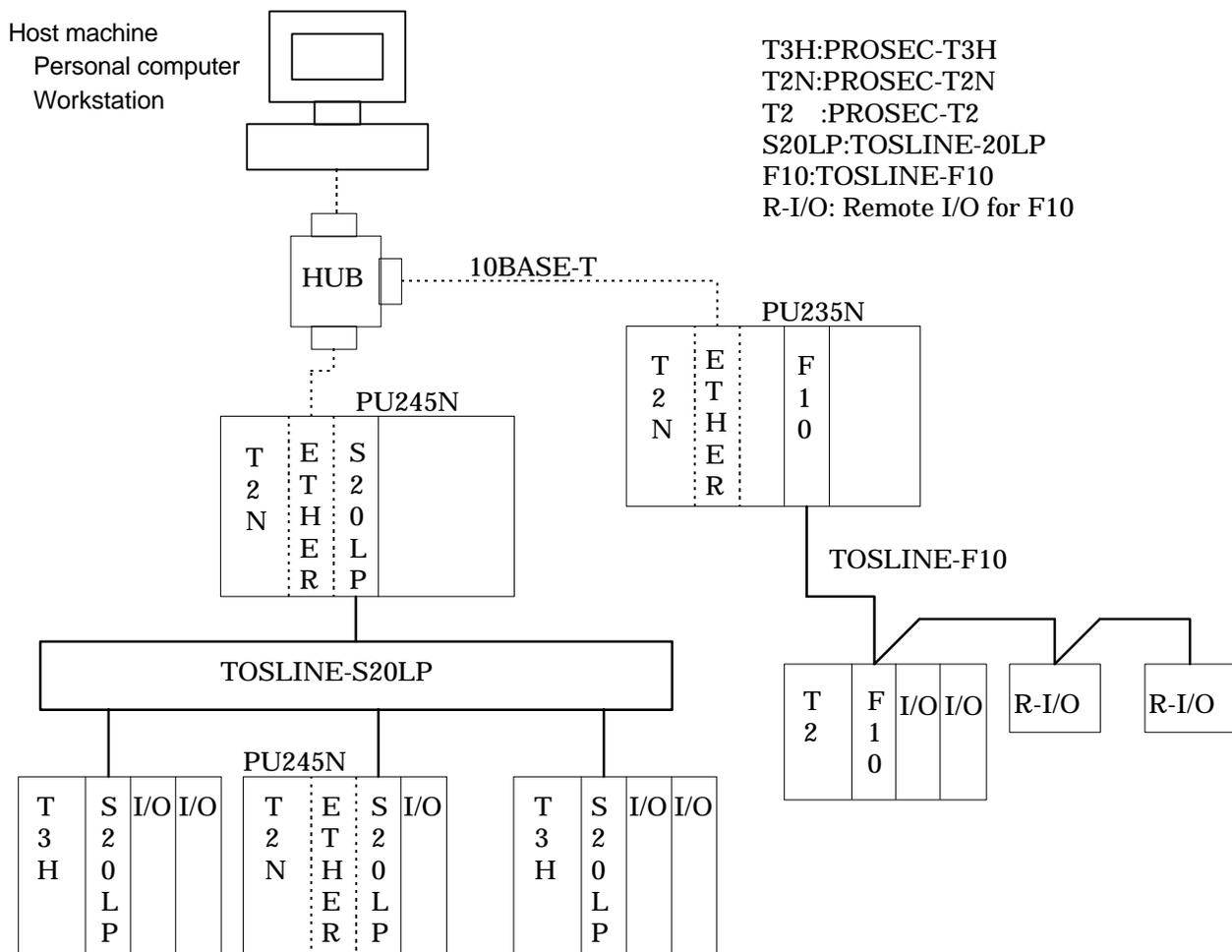


Figure 1.1 System Configuration Examples

## 1.2 LAN Standards and Terminology

The LANs commonly referred to as Ethernet LANs are defined by two standards, the Ethernet version 2.0 standard (DIX) developed by DEC, Intel, and Xerox, and the ISO 8802-3 standard (IEEE 802.3), which is an international version of the DIX standard. Since the ISO 8802-3 includes the DIX standard, network equipment built to those standards will be basically compatible.

Since the Ethernet Port physical layer has a structure that conforms to the ISO 8802-3 standard, users must be sure to **use network equipment that conforms to the ISO 8802-3 standard.**

The networking terminology used in this document is the terminology used in the ISO 8802-3 standard. Table 1.3 presents the correspondence between the DIX standard terminology (common terms) and the ISO 8802-3 standard terminology.

**Table 1.1 Networking Terminology Correspondences**

DIX Standard	ISO 8802-3 Standard
Ethernet	10BASE5
Cheapernet	10BASE2
Coaxial cable	Coaxial cable
Transceiver cable	AUI (attachment unit interface) cable
Transceiver	MAU (medium attachment unit)
Ethernet address	MAC address

Ethernet is a registered trademark of Xerox, Corporation.

Cheapernet is not a registered trademark, but rather is a common name.

### 1.3 Basic Functionality

The Ethernet Port has the following main features.

#### 1. T series computer link protocol transmission (slave port)

The host machine can upload or download user programs, execute control commands, and read or write register data to a T2N on the 10BASE-T LAN using the T series computer link protocol. This function can be used simply by making basic settings on the Ethernet Port from the T2N; there is no need to write special user programs for the T2N.

Computer link protocol uses UDP/IP as the transmission protocol.

UDP/IP: User datagram protocol/Internet protocol.

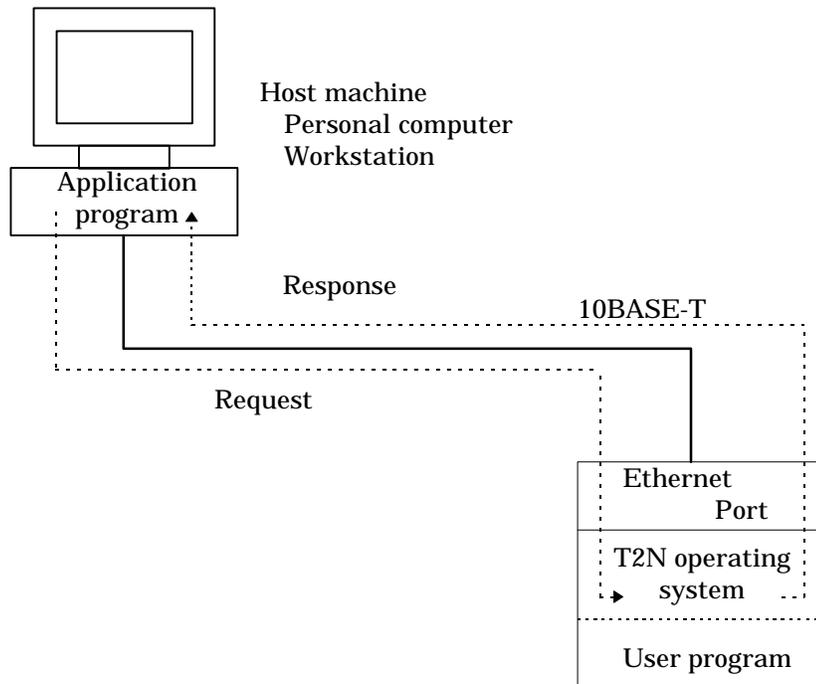


Figure 1.2 T Series Computer Link Protocol Transmission (slave port)

2. T series/PC link protocol transmission

The T2N can use the T series/PC link protocol to read or write register data on another T2N on the 10BASE-T LAN.

While T2N user programs for reading and writing are necessary to use the host functions, only the basic Ethernet Port settings performed from the T2N are required to use the slave functions, and no special user programs are required on the T2N.

PC link protocol transmission uses UDP/IP as the transmission protocol.

UDP/IP: User datagram protocol/Internet protocol .

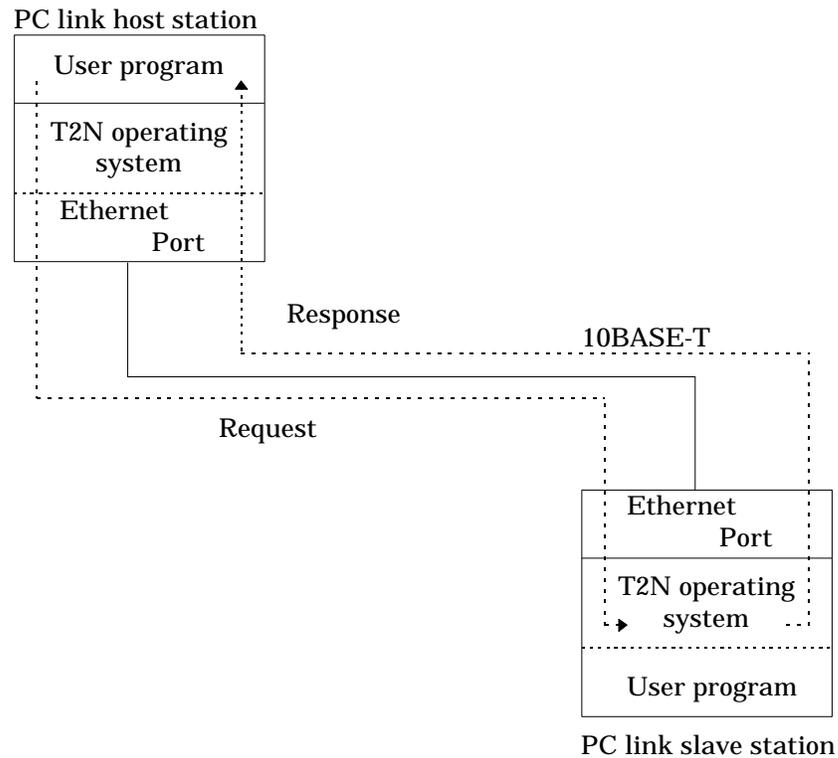


Figure 1.3 T Series/PC Link Protocol Transmission

### 3. Socket interface support

This is a standard communications technique used on engineering workstations and similar computers. Transmission between user programs on the T2N and host machines is possible using the socket interface. There are eight sockets on an Ethernet Port and the transmission protocol (TCP/IP or UDP/IP) can be specified for each socket.

TCP/IP: Transmission control protocol/Internet protocol

UDP/IP: User datagram protocol/Internet protocol

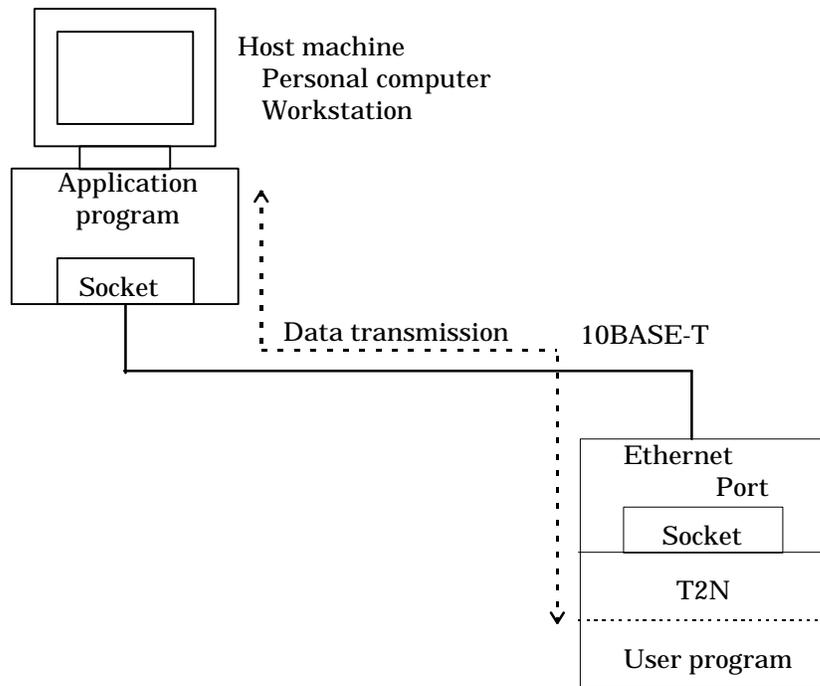


Figure 1.4 Socket Interface

4. PC (programmable controller) gateway function

\* This function is under development, but is currently not supported.

Toshiba plans to add support to the T2N for a gateway function between the Ethernet Port and the TOSLINE-S20LP Port. This function will allow access to stations (PROSEC-T3H/T2N) on the TOSLINE-S20LP from ports on the 10BASE-T LAN using the computer link protocol.

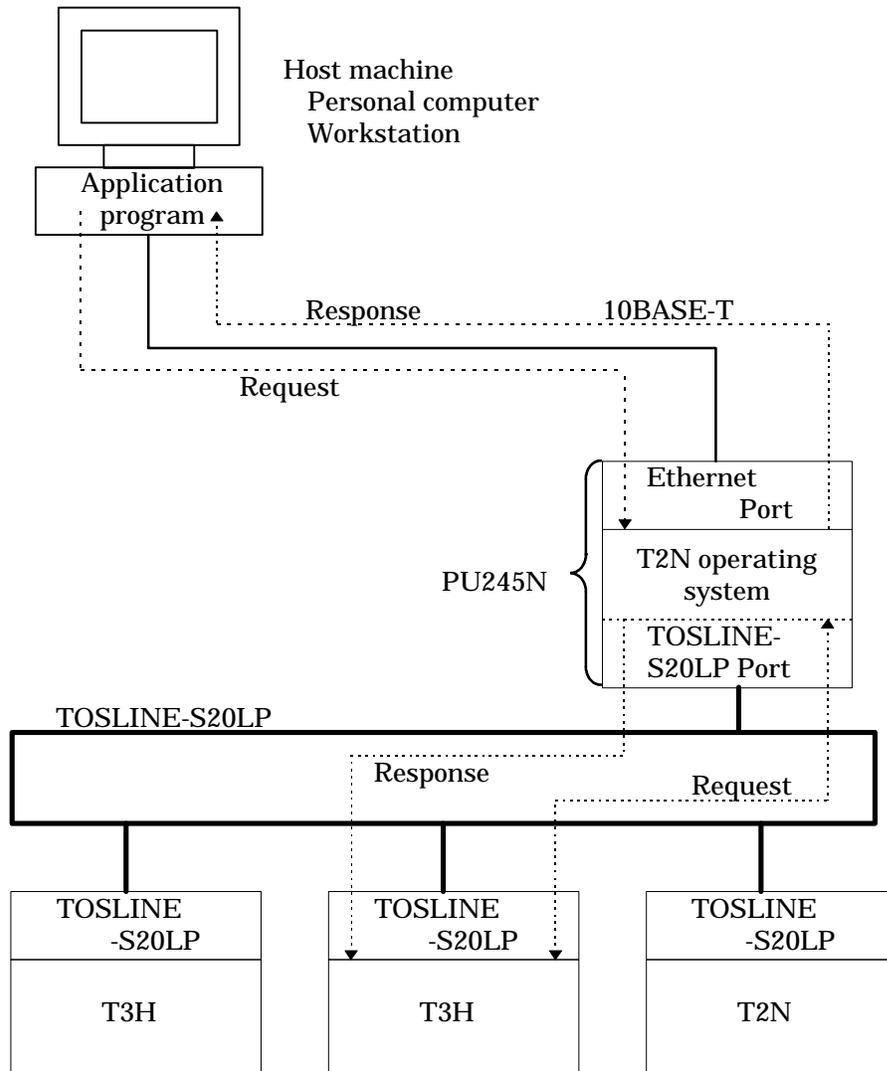


Figure 1.5 Gateway Function

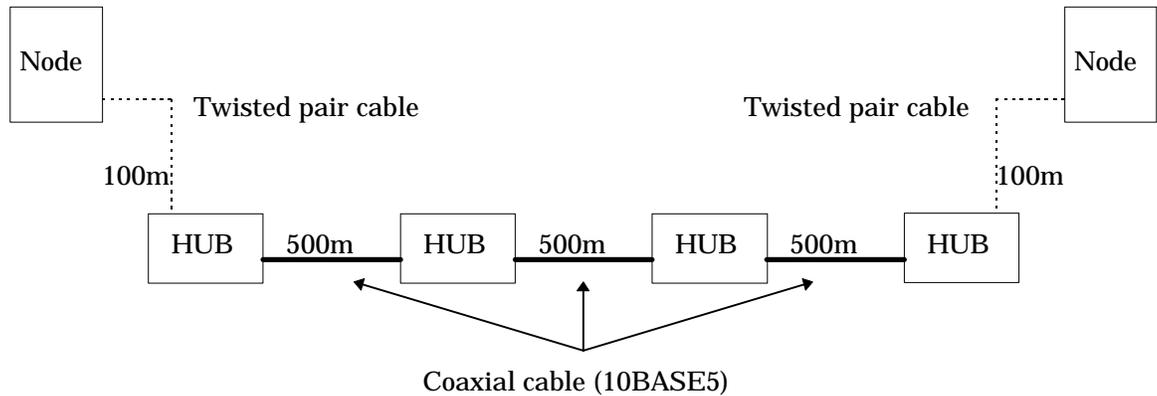
**5. Support for 10BASE-T Network**

The Ethernet Port can connect to a 10BASE-T network

10BASE-T is a LAN technology that uses baseband transmission, provides a data transmission rate of 10 Mbps, and supports single segments of up to 100 meters in length.

This network uses 10BASE-T (Twisted pair cable) as the transmission cable.

The figure below shows the maximum interport separation.



It is regarded that HUB is repeater and a twisted pair cable is single segment (link segment).

**Figure 1.6 10BASE-T**

Usage Recommendations
1. Do not exceed the maximum cable length (100 m) in systems using 10BASE-T.

## 1.4 Ethernet Port Specifications

Table 1.2 lists the Ethernet Port specifications. The general specifications of the Ethernet Port conform to those of the T2N main unit.

**Table 1.2 Functional Specifications**

Item	Specification	
Module type	PU235N/245N (Ethernet Port is built-in)	
Transmission specifications	10BASE-T	
	Media access method	CSMA/CD
	Modulation	Baseband
	Transmission topology	Bus type
	Transmission speed	10 Mbps
	Maximum distance between ports	200 m (Case of single HUB) 1700 m (Case of four HUB)
	Maximum segment length	100 m (Between port and HUB)
	Maximum number of ports	single port per segment
	Minimum port separation	None
	Connector	8-pin modular connector (RJ-45)
Connecting cable	Unshield twisted pair	
Transmission services	<ol style="list-style-type: none"> <li>1. T series computer link protocol (slave port)</li> <li>2. T series/PC link protocol</li> <li>3. Socket interface (8 sockets)</li> <li>4. PC gateway function (currently unsupported)</li> </ol>	
RAS functions	<ol style="list-style-type: none"> <li>1. Self diagnostics at power on ROM, RAM, LAN controller, EEPROM, T2N interface buffer memory, MAC address</li> <li>2. Test functions accessible from user programs <ul style="list-style-type: none"> <li>• Remote port verification request</li> <li>• Remote port loopback test (Between T2N and T2N/T3H systems)</li> </ul> </li> <li>3. Data provided by RAS data readout <ul style="list-style-type: none"> <li>• Event trace</li> <li>• LAN controller (network) information</li> <li>• Protocol state</li> </ul> </li> <li>4. Time/date setup function</li> <li>5. Station status in the T2N special registers (SW)</li> <li>6. Socket information in the T2N interface buffer</li> </ol>	
Consumed current [A]	0.6A (5V DC)	
External dimensions [mm]	99.0 (W) ×122 (H)	
Weight [grams]	200	
Mounting method	Built-in T2N	
Number of modules mountable	1 modules per T2N	
Access method	SEND/RECV instructions (module control and transmission functions)	

\* Segment: A network unit formed from a single transmission medium (twisted pair cable)

### 1.5 Software Organization

The Ethernet Port transmission functions (computer link protocol transmission, PC link protocol transmission and socket interface transmission) are implemented with the software organization shown in the figure.

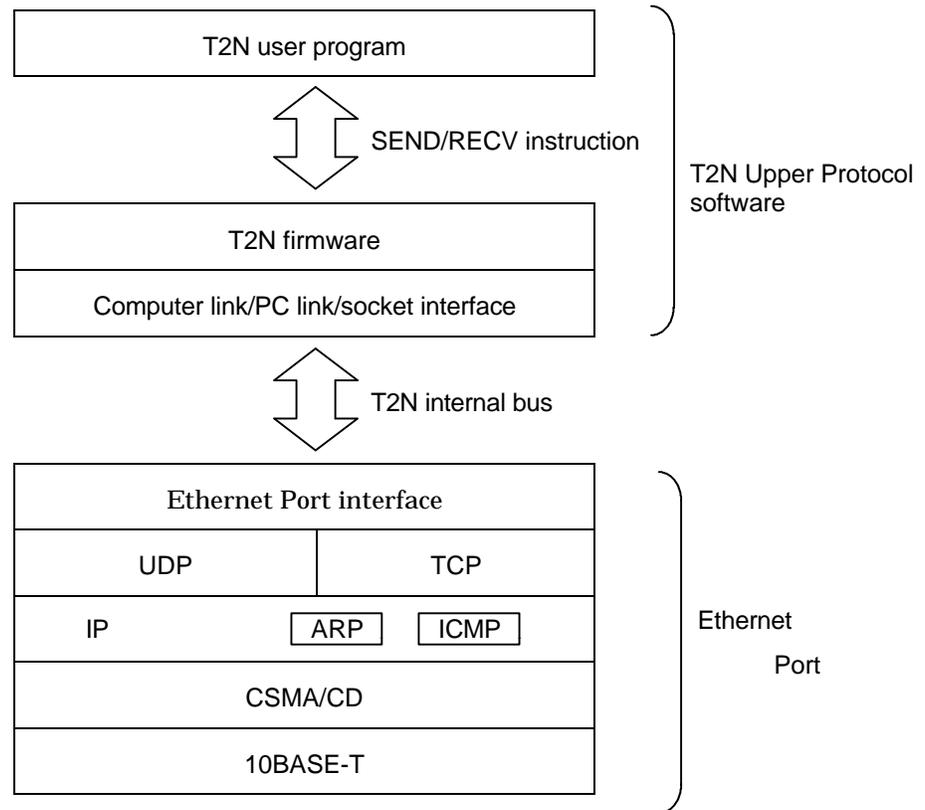


Figure 1.7 Software Organization

#### TCP (transmission control protocol)

Before transferring data with the remote port, a connection (virtual communication circuit) is first established and then data is transferred. Transmission reliability is guaranteed by packet order control, retransmission, flow control, and other techniques.

The Ethernet Port supports the following:

- Response bit (ACK)
- Push bit (PSH)
- Synchronized bit (SYN)
- Fin bit (FIN)
- Retransferring
- Window (flow) control
- Segment division
- Reporting of the maximum receivable segment length

The Ethernet Port does not support the reset bit (RST).

**UDP (User datagram protocol)**

Transfers data with the remote port in datagram units. Provides high-speed transfers since a connection is not used, but does not have control functions that assure reliability, as does TCP. Thus it requires support from a higher level protocol.

The Ethernet Port supports the:

- the UDP checksum

function.

**IP (Internet protocol)**

Sends and receives data in the datagram format. Sends data in datagram units to the remote port according to the IP address.

The Ethernet Port supports:

- Data fragmentation and reassembly,
- Direct routing, and
- Broadcast

functions.

The Ethernet Port does not support the indirect routing and multicast functions.

Although the Ethernet Port physical layer structure conforms to the ISO8802-3 standard, it adopts DIX standard frames as the data link frames (the IP datagram send/receive frames).

**ARP (Address resolution protocol)**

Searches for a port MAC address (Ethernet address) that has a specified IP address.

**ICMP (Internet control message protocol)**

Reports IP packet errors, verifies the state of the IP network, and supports related functions.

The Ethernet Port supports the:

- Echo request/reply,
- Timestamp request/reply, and
- Information request

functions.

The Ethernet Port does not support the:

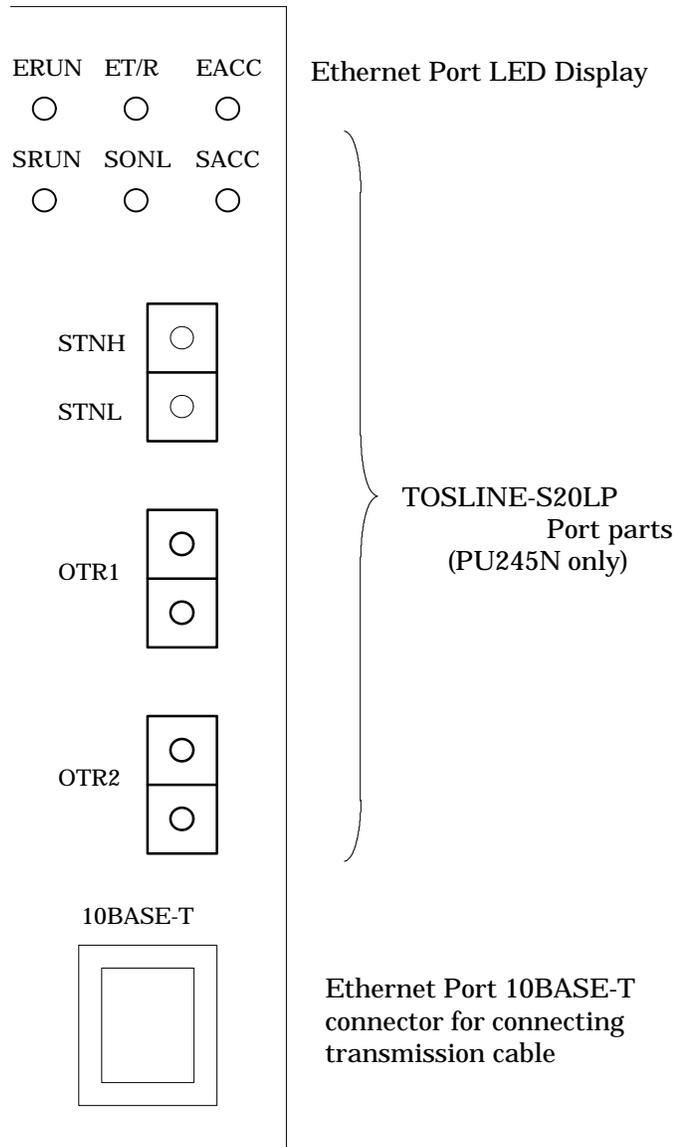
- Destination unreachable,
- Source quench,
- Redirect,
- Time exceeded for a datagram,
- Parameter problem on datagram,
- Information reply,
- Address mask request, and
- Address mask reply

functions.

## 2. Ethernet Port Parts and Functions

This chapter describes the sections of the Ethernet Port and their functions.

### 2.1 Front Panel

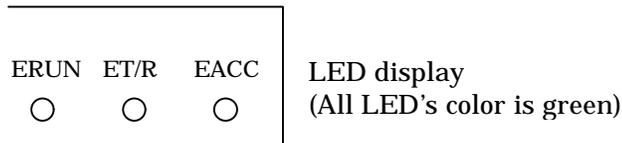


**Figure 2.1 PU235N/245N External Dimensions**

## 2.2 Component Functions

### (1) LED Display

Indicators that display the Ethernet Port operating state.



**Figure 2.2 LED's display and name**

### Content Displayed

**Table 2.1 LED Display Content**

LED	Content
ERUN (Ether Run)	Indicates whether or not the Ethernet Port is operating normally. On.... The Ethernet Port is operating normally. Off.... The Ethernet Port error occurred.
ET/R (Ether Transmission / Receive)	Indicates whether or not the Ethernet Port is transmitting data or receiving data. On.... Data transmission or receipt in progress Off.... Data is not being transmitted or received.
EACC (Ether Access)	Indicates whether or not the T2N is accessing the Ethernet Port. On.... The T2N is accessing the Ethernet Port. Off.... The T2N is not accessing the Ethernet Port.

### (2) 10BASE-T connecting connector

This connector is used for connecting 10BASE-T twisted pair cable and Ethernet Port.

See section 3.1, " Connecting to the Network ", for detailed information on connection.

### 3. Preparing for Operation (Hardware)

This section presents the Ethernet Port setup.

**WARNING**

1. Build emergency stop, interlock, and similar circuits outside the PU235N/245N. If a breakdown or malfunction occurs in the PU235N/245N, there is the danger that someone could be hurt or fatally injured. Also, other equipment could be damaged or a breakdown could occur.

**CAUTION**

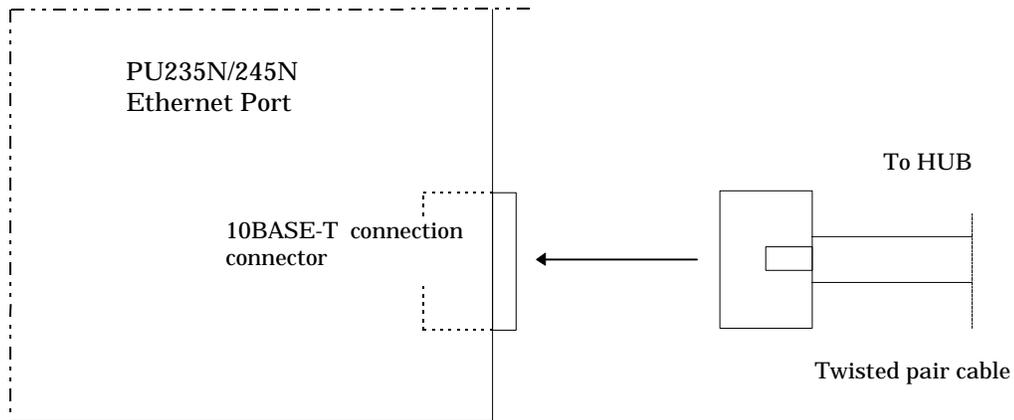
2. Take all necessary safety precautions before performing operations such as changing a running program, forcing output, or switching to RUN (operate) or HALT (stop) modes.  
Equipment can be damaged and breakdowns can be caused by operating this equipment incorrectly.

### 3.1 Connecting to the Network

The Ethernet Port supports 10BASE-T network.

#### (1) Connection to a 10BASE-T Network

This section describes the connection procedure starting from the state where the 10BASE-T twisted pair cable is already supplied and ready to be connected. See section 9.2, "Network Wiring", for details on wiring the 10BASE-T twisted pair cable.



**Figure 3.1 Connection to a twisted pair cable**

- (1) Align the twisted pair cable with the 10BASE-T connector and insert the modular connector.
- (2) Pull the 10BASE-T connector and verify that the connector is locked and cannot be pulled out.

<b>CAUTION</b>
<p>1. When installing a 10BASE5, 10BASE2 or 10BASE-T network, the cabling must be laid with adequate safety precautions and in accordance with all applicable standards. Installation and wiring should be performed by a qualified professional.</p> <p>See the <b>ISO/IEC8802-3</b> standard for installation environment standards.</p>

<b>Usage Recommendations</b>
<ol style="list-style-type: none"> <li>1. Do not exceed the maximum cable length (100 m) in systems using 10BASE-T.</li> <li>2. A HUB and a twisted pair cable are required in systems configured using a 10BASE-T network. (See section 3.2, "Network Wiring Equipment.")</li> </ol> <p style="margin-left: 20px;">These parts must be ordered separately if required.</p>

## 3.2 Network Wiring Equipment

This section describes the wiring and related equipment that makes up the network to which the Ethernet Port is connected. Since the Ethernet Port conforms to the ISO 8802-3 (IEEE 802.3) standard, users should select equipment to be used in the network that also conforms to this standard.

The following section introduces recommended wiring and related equipment. We strongly recommend that these models be used when new items are needed.

Contact the manufacturer directly for detailed information on this equipment.

### (1) 10BASE-T

- Twisted pair cable
 

<b>Catalog no.</b>	ECM-04-□□□M	□□□:Length(1 to 100)
<b>Manufacturer</b>	Showa Electric Wire & Cable Co., Ltd. (Enable cross cable)	
  
- HUB
 

<b>Catalog no.</b>	ETH-807B ( 8 ports)	
	ETH-707A (16 ports)	
<b>Manufacturer</b>	Showa Electric Wire & Cable Co., Ltd. (Enable cross cable)	

(2) 10BASE5

- Coaxial cable

Catalog no.	EEA-9202-□M+TMJ x 2	□: Length
Standard outside diameter	10.3 mm diameter, with terminators at both ends	
Manufacturer	Showa Electric Wire & Cable Co., Ltd.	

- Plugs

Catalog no.	EPL-6213
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- Terminators

Catalog no.	EEJ-6213
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- Adapters

Catalog no.	EAD-6213
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- Grounding clamps

Catalog no.	EEL-6213
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- AUI cable

Catalog no.	AUI15-□M	□: Length
Standard outside diameter	6.5 mm diameter (thin wire), molded connectors	
Length	1, 2, 3, 5, 10, and 15 meters	
Manufacturer	Showa Electric Wire & Cable Co., Ltd.	

Catalog no.	AUI50-□M	□: Length
Standard outside diameter	10.5 mm diameter (thick wire), molded connectors	
Length	20, 30, 40, and 50 meters	
Manufacturer	Showa Electric Wire & Cable Co., Ltd.	

- MAU

Catalog no.	ETR-6071 02
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

## (3) 10BASE2

- Coaxial cable

Catalog no.	RG-58A/U
Standard outside diameter	5.0 mm diameter, with BNC plugs at both ends
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- BNC plugs

Catalog no.	EPL-1067
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- T connectors

Catalog no.	ETS-8191
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- Adapters

Catalog no.	EAD-8191
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

- Terminators

Catalog no.	EEJ-8191
Manufacturer	Showa Electric Wire & Cable Co., Ltd.

**CAUTION**

1. When installing a 10BASE5, 10BASE2 or 10BASE-T network, the cabling must be laid with adequate safety precautions and in accordance with all applicable standards. Installation and wiring should be performed by a qualified professional.

See the **ISO/IEC8802-3** standard for installation environment standards.

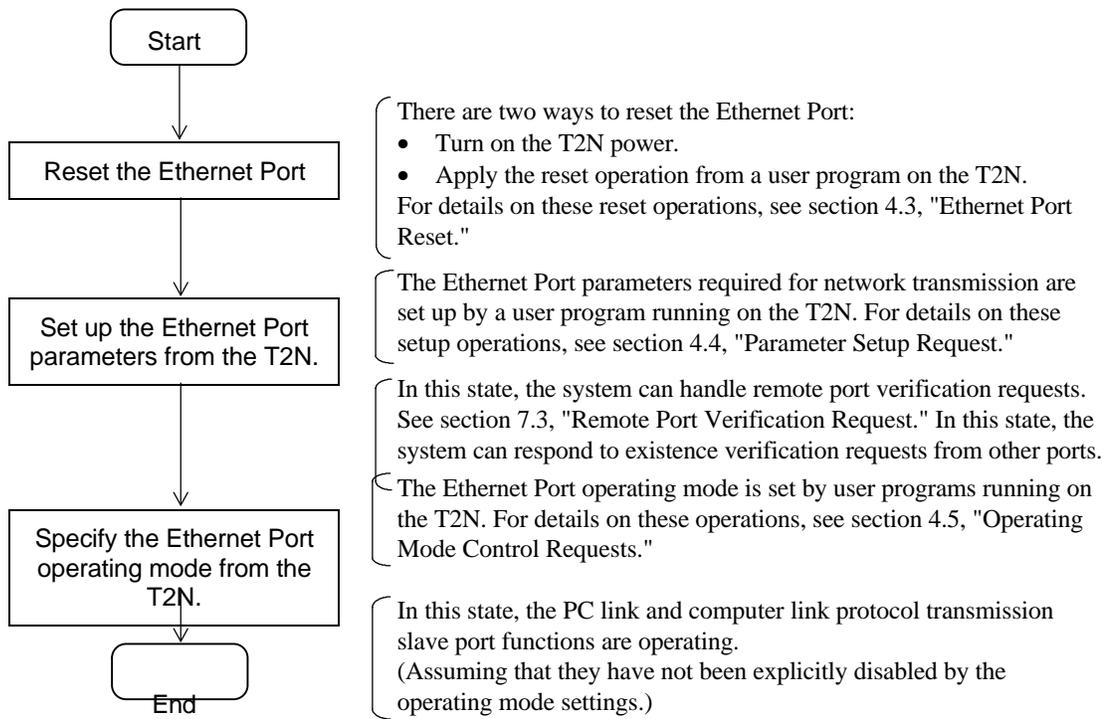
## 4. Preparing for Operation (Software)

This section describes setting up the Ethernet Port software for operation.

 <b>CAUTION</b>
<p>1. Chapter 4 presents information related to using the functions provided by the Ethernet Port from a T2N, including the instruction (request) format, important items that require attention, and sample programs.</p> <p>Make a point of understanding the content of chapter 4 thoroughly before writing programs that use the Ethernet Port. The sample programs present basic examples of Ethernet Port usage, and should be reviewed carefully before use in an actual system.</p>

### 4.1 Ethernet Port Setup Flowchart (Software)

This section presents the flowchart for Ethernet Port setup.



**Figure 4.1 Ethernet Port Setup Flowchart (Software)**

<b>Usage Recommendations</b>
<p>Before performing the settings described in this chapter, register the PU235N/245N as T2N I/O. The I/O type is ETHER or E/S.</p>

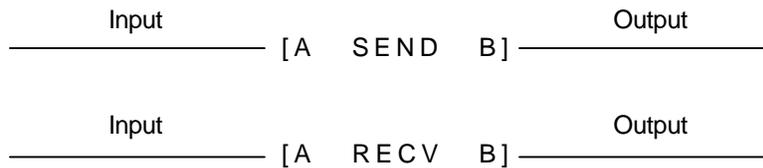
## 4.2 Ethernet Port Instruction Specifications

Before discussing the software side of the Ethernet Port setup procedure, this section presents an overview of the SEND and RECV instructions, which are the instructions that are used to control the Ethernet Port from T2N user programs. Ethernet Port control and transmission control (PC link protocol transmission and socket interface transmission) are performed solely using the SEND and RECV instructions.

See the individual items for details on the actual requests sent to the Ethernet Port.

### (1) Instruction format

Figure 4.2 shows the formats of the SEND and RECV instructions.



#### Operand

	Name	Device						Register										INDEX			Con-stant	Index		
		X	Y	S	L	R	Z	X W	Y W	S W	L W	R W	W	T	C	D	F	I W	O W	I			J	K
A	Transfer parameter							✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						✓
B	Status								✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						✓

Input	Operation		Output	ERF
OFF	No execution		OFF	OFF
ON	Execution	Transmission port instruction busy/instruction execution in progress (response wait)	OFF	OFF
		Response reception complete (normal complete)	ON	OFF
		Instruction error	ON	ON

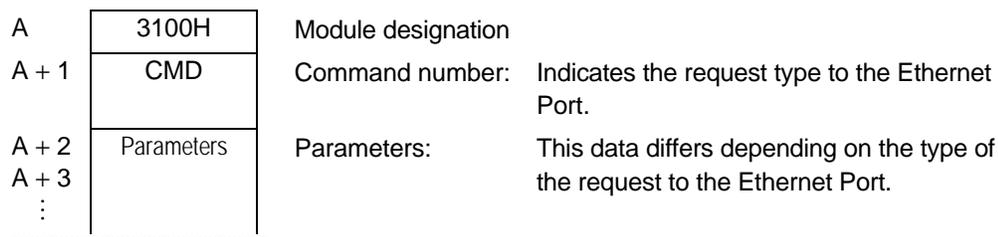
Figure 4.2 SEND/RECV Instruction Formats

#### <Important items>

- a. ERF is the S0051 special relay. See the T-series Instruction Set for information on ERF usage.
- b. For these instructions, **instruction execution starts at the point the input changes state from off to on.**
- c. Since these instructions span multiple scans, applications must **hold the input in the on state until the output goes to the on state.**  
If the input is turned off before the output goes on, the system will not be able to execute the following instruction processing normally.
- d. After instruction execution terminates, it is possible to start execution again by applying a new input.

(2) Transfer parameters (A)

The data required for a request to the Ethernet Port is set up in an area that starts in the register indicated by [A]. Then the application executes a SEND or RECV instruction. Here we present an overview of the structure of the transfer parameters. Tables 4.1 and 4.2 list the requests that can be issued to an Ethernet Port.



Module designation: Designates the Ethernet Port mounted in the T2N.  
 3 (Module ID): Must be set to 3 for an Ethernet Port.  
 1 (channel number): Must be set to 1 for an Ethernet Port.  
 00 (station number): This two-digit field must be set to 00.

<Important items>  
 e. Do not modify the contents of the transfer parameters while an instruction execution is in progress.  
 The SEND or RECV instruction may not complete.

Table 4.1 Instructions (Requests) to the Ethernet Port (Part 1)

Request	Function	Instruction	CMD
Reset request	Resets the Ethernet Port.	SEND	0011H
Parameter setup request	Sets up the Ethernet Port parameters.	SEND	0012H
Operating mode control request	Sets the operating mode.	SEND	0013H
Remote port verification request	Verifies the existence of another port on the network.	SEND	0014H
RAS information readout request	Reads out the Ethernet Port RAS information.	RECV	0015H
Time set request	Sets the Ethernet Port internal clock Time information for event trace (RAS information)	SEND	0018H
Register read request (PC link function)	Reads register data from remote T2N into registers in the local T2N.	RECV	0021H
Register write request (PC link function)	Writes register data from the local T2N to registers in remote T2N.	SEND	0021H
Remote port loopback request	Performs a loopback operation with another Ethernet Port on the network.	SEND	000FH
UDP open request	Opens a socket interface (UDP).	SEND	0031H
UDP send request	Sends data from a socket interface in the UDP open state.	SEND	0032H
UDP receive request	Reads data received by a socket interface in the UDP open state.	RECV	0033H
UDP close request	Closes (terminates) a socket interface in the UDP open state.	SEND	0034H

CMD (command number): Indicates the request type to the Ethernet Port.

**Table 4.2 Instructions (Requests) to the Ethernet Port (Part 2)**

Request	Function	Instruction	CMD
TCP open request	Opens a socket interface (TCP).	SEND	0035H
TCP send request	Sends data from a socket interface in the TCP open state.	SEND	0037H
TCP receive request	Reads data received by a socket interface in the TCP open state.	RECV	0038H
TCP close request	Closes (terminates) a socket interface in the TCP open state.	SEND	0039H

CMD (command number): Indicates the request type to the Ethernet Port.

**<Important items>**

- f. Items ① to ③ are CMD classes. Class ① is Ethernet Port control, ② is PC link protocol transmission, and ③ is socket interface transmission.
- For a single Ethernet Port, it is not possible to request another class ① CMD while a class ① CMD is executing. This is also true for class ② CMDs. If such a request is issued, the status (described in the following status information) will be transmission port busy, and the request will go to the wait state.
  - For a single Ethernet Port, when requesting a class ③ CMD, it is not possible to request another class ③ command for a socket for which an instruction is executing. If such a request is issued, the status (described in the following status) will be transmission port busy, and the request will go to the wait state.
  - There are eight sockets, and each can independently accept and execute a class ③ instruction.
- g. If the T2N switches from the run to the halt state and then back to the run state while an instruction is executing, it is possible that the Ethernet Port could execute the request issued immediately prior to the halt state. This section describes the handling required after returning to the run state.
- The T2N should always set the Ethernet Port to standby mode after the T2N returns to run mode. (See 4.5.) Setting the Ethernet Port to standby mode causes all the requests issued to the Ethernet Port to be discarded, computer link and PC link protocol transmissions to be stopped, and the socket interface to be initialized, i.e. set to the closed state.
  - Before setting the Ethernet Port to standby mode, verify the operating mode (See 4.5.) with the Ethernet Port station status. (See section 7.1.) If the Ethernet Port is in run mode, set it to standby mode. However, if it is in standby mode, there is no need to set it to standby mode again.

(3) Status (B)

The status during the execution of a SEND or RECV instruction and after that instruction terminates is stored in the register shown at B. When TermSTS = 0BH, the detailed information (Ethernet Port error information) will be loaded into the registers indicated at location B + 1. Figure 4.3 shows the structure of the status information. Tables 4.3 and 4.4 list the TermSTS values, and table 4.5 and 4.6 lists the contents of the detailed information.

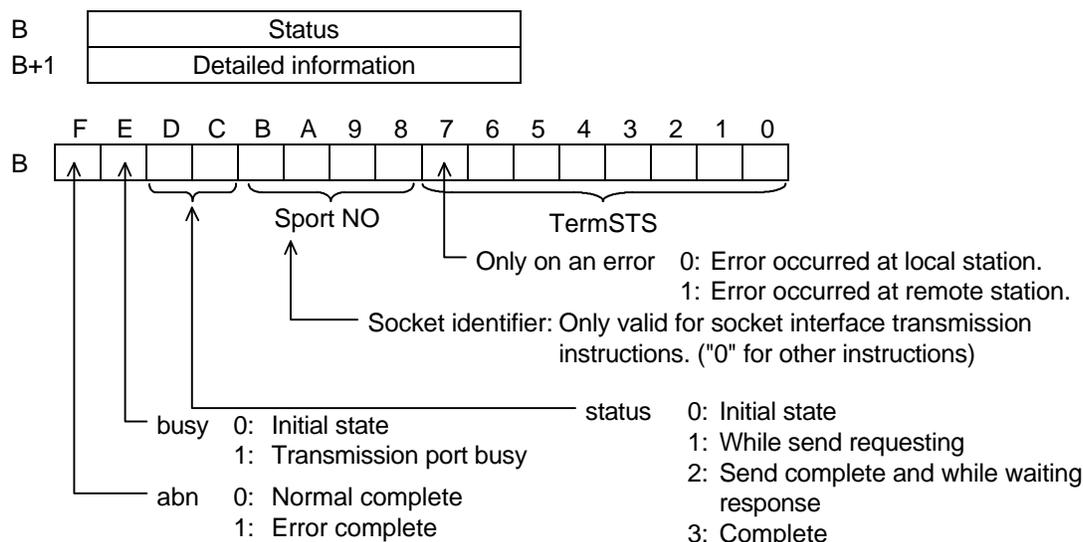


Figure 4.3 Status Structure

<Important items>

h. The **user program must set bits C and D to 0** before first activating a SEND/RECV instruction. Failure to do so will cause the SEND/RECV instruction to not complete. Also, **do not write to the status during instruction execution**. This will cause the SEND/RECV instruction to not complete.

i. When the number of SEND/RECV instructions are used in the user program and their instructions are executed simultaneously, separate the status and detailed information register every instruction.

Table 4.3 TermSTS (Part 1)

TermSTS	Status	Meaning
00H	Normal complete	
01H	Register specification error	<ul style="list-style-type: none"> <li>Occurs if a register other than XW, YW,W, LW, RW, D, F, T, C, or SW is specified as the data storage register in one of the following requests. Requests: RAS information readout, remote port loopback, UDP send/receive, TCP send/receive, register read/write.</li> <li>In a register read or write request, the request specified a read or write from a T register to a register other than a T register. (This also applies to C registers.)</li> </ul>
02H	Response timeout	<ul style="list-style-type: none"> <li>Occurs if a response is not received within the specified time for a remote port loopback or register read/write request.</li> </ul>

**Table 4.4 TermSTS (Part 2)**

TermSTS	Status	Meaning
03H	Parameter error	<ul style="list-style-type: none"> <li>Occurs if a value other than 3 is specified for the module ID.</li> <li>Occurs if a value other than 1 is specified for the channel number.</li> <li>Occurs if a command number other than one of the following is specified for a SEND instruction. 000FH, 0011H - 0014H, 0018H, 0021H, 0031H, 0032H, 0034H, 0035H, 0037H, 0039H</li> <li>Occurs if a command number other than one of the following is specified for a RECV instruction. 0015H, 0021H, 0033H, 0038H</li> <li>Occurs if a value other than 1 to 8 is specified for the socket identifier. (See section 6.4 "Using the Ethernet Port Socket Interface".)</li> <li>Occurs if an incorrect parameter is specified in a RAS information readout request. (See section 7.6 "RAS Information Readout ".)</li> </ul>
04H	Memory write protect	<ul style="list-style-type: none"> <li>Occurs if an attempt to write is made when the remote registers were write protected.</li> <li>Occurs if an attempt to read is made when the local registers were write protected.</li> </ul>
05H	(Reserved)	
06H	Module error Initialization in progress Transmission complete timeout	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port is down.</li> <li>Occurs if the Ethernet Port is initializing.</li> <li>Occurs if a T2N internal timer times out. Requests: UDP open, UDP send, UDP close, TCP send, TCP close</li> </ul>
07H	No send channel (CH)	<ul style="list-style-type: none"> <li>Occurs if the corresponding Ethernet Port is not installed.</li> </ul>
08H	Port address out of range	<ul style="list-style-type: none"> <li>Occurs if the local port IP address was specified for the remote destination IP address. Requests: Remote port loopback, register read/write</li> </ul>
09H	Incorrect transmission word count	<ul style="list-style-type: none"> <li>Occurs if either 0 words or 486 or more words were specified for either a register read/write or a remote port loopback request.</li> <li>Occurs if either 0 words or 1001 or more words were specified for a UDC/TCP send/receive request.</li> </ul>
0AH	Boundary error	<ul style="list-style-type: none"> <li>Occurs if the specified area (start address + register range) does not exist in the T2N data storage registers. Requests: RAS information readout, remote port loopback, UDP send/receive, TCP send/receive, register read/write</li> </ul>
0BH	Transmission error	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port returns an error response. Detailed information (the Ethernet Port response status) is stored at B+1.</li> </ul>
0CH	No I/O response	<ul style="list-style-type: none"> <li>Occurs if the T2N cannot access the Ethernet Port.</li> </ul>
0DH	(Reserved)	
0EH	Send data capacity exceeded	<ul style="list-style-type: none"> <li>Occurs if a request is discarded when a T2N internal resource insufficiency occurs due to increasing amounts of transfer data.</li> <li>Occurs if a T2N retransmission times out.</li> </ul>
0FH	(Reserved)	

Table 4.5 Detailed Information (Ethernet Port Error Responses) PART1

Error	Code	Description
Normal complete	0001H	Request completed normally
Local port fault	0002H	The local port is in down mode
Local port standby	0003H	In standby mode data transmission (socket interface transmission or PC link protocol transmission) was requested.
Timeout	0020H	Timeout occurred during TCP open processing or receive processing, or during UDP receive processing.
Length error	0030H	Incorrect send data length
Port mode error	0040H	One of the following requests was issued in a mode other than standby mode: a. Parameter setup, b. MAC address setup, c. Reset
MAC/IP not set up	0050H	Occurs if the MAC address or IP address is not set up.
Transmission prohibited state	0060H	<ul style="list-style-type: none"> <li>Occurs if a PC link request was issued from the T2N in run mode in the message transmission prohibited state.</li> <li>Occurs if a socket open, close, send, or receive request is issued by the T2N in run mode in the socket interface transmission prohibited state.</li> </ul>
Format error	0070H	Occurs when there is a format error in the request text.
	0071H	Port number error
	0072H	Request code error
	0073H	Time setting: Year
	0074H	Time setting: Month
	0075H	Time setting: Day
	0076H	Time setting: Hour
	0077H	Time setting: Minute
	0078H	Time setting: Second
Opened	0080H	Occurs if an open request is issued for an already open socket.
Unopened	0081H	<ul style="list-style-type: none"> <li>Occurs if send, receive, or close request is issued for an unopened socket.</li> <li>Occurs if send, receive request is issued for TCP socket during receiving CLOSE request (RCLOSE bit is 1).</li> </ul>
Incorrect socket identifier	0082H	Occurs if the socket identifier is out of range (1 to 8).
Incorrect control request specification	0083H	Occurs if a request that was prohibited by an operating mode control request is issued. <ul style="list-style-type: none"> <li>Occurs if a run request is issued in run mode.</li> <li>Occurs if a standby request is issued in standby mode.</li> <li>Occurs if a run request and a standby request are issued at the same time.</li> </ul>
LAN controller driver error	0090H	Occurs if the setting of the local port IP address and port number in the area reserved for transmission fails.
	0091H	<ul style="list-style-type: none"> <li>Error in the TCP or UDP protocol (such as a transmission phase error)</li> <li>Occurs if send request is issued for TCP socket that remote port is down and NOACK bit is 1.</li> </ul>
	0092H	<ul style="list-style-type: none"> <li>Occurs if the setting of the remote port IP address in the remote port information area (the NETDATA table) fails.</li> <li>Occurs if the remote port IP address network address differs from the local port IP address network address.</li> </ul>

**Table 4.6 Detailed Information (Ethernet Port Error Responses) PART2**

Error	Code	Description
Memory pool allocation failure	00A0H	Occurs if the Ethernet Port internal OS fails to allocate memory.
Port 2 task start failure	00A1H	Occurs if computer link or PC link socket identifier acquisition fails.
Broadcast specification error	00A2H	Occurs if the remote IP address is broadcast on a remote port verification request.
MAC address specification error	00B0H	BCC error
	00B1H	Occurs if an error occurs after address saving on a MAC setup request.
Resource shortage error	00C0H	Occurs if send request is issued for the Ethernet Port internal buffer fill.
Receive request from the T2N	00C1H	The specifiable size from user program < The Ethernet Port is receiving data size
	00C2H	The specifiable size of user program beyond the limits from 1 to 1000.

### 4.3 Ethernet Port Reset

This section describes the Ethernet Port reset procedures. There are two ways to reset an Ethernet Port as follows:

- (1) T2N power on
- (2) Reset by a user program running on the T2N

In Ethernet Port reset processing, when one of the above resets occurs, the Ethernet Port first performs the self check described below. If the self check completes normally, the Ethernet Port switches to standby mode (the parameter setting wait state), thus completing the reset sequence. If an error is discovered during the self check, the Ethernet Port goes to the down state.

- Ethernet Port self check items
  - System ROM check
  - System RAM check
  - LAN controller check
  - T2N interface buffer memory check
  - EEPROM check

See section 4.5, "Operating Mode Control Requests", for more information on the Ethernet Port operating modes.

#### (1) T2N power on

When power is applied to the T2N, power is also applied to the Ethernet Port, and the PU235N/245N performs its reset processing sequence.

(2) Reset request from the T2N

It is possible to initiate reset processing from user programs on the T2N. T2N can execute the reset processing instruction on standby mode (See section 4.5). This section describes the format of the reset processing instruction.

**Reset request (using the SEND instruction)**

a. Function

Reset request from a user program to an Ethernet Port.

b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0011H	CMD number

c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

d. Important items

An error will occur if the Ethernet Port is in run mode or down mode.

Run mode - Status: Transmission error (TermSTS = 0BH)  
 Detailed information: Port mode error (0040H)

Down mode - Status: Module error/initialization in progress (TermSTS = 06H)

e. Sample program

This sample ladder program issues a reset request to the channel 1 Ethernet Port.

This program is executed by turning on the R0501 A contact.

Parameters: 12544:3100H, 00017:0011H

```

|R0501                                     R0601 |
1|-| |----- ( )-----|
|          /* Reset request          */
|R0601                                     |
2|-| |--[ 12544 MOV RW000][ 00017 MOV RW001]-----|
|          /* Designates the module and sets the CMD number. */
|          |
|          |[RW000 SEND RW010][ RST R0501]-----|
|          /* Sets the request.      */
    
```

## 4.4 Parameter Setup Request

This section describes the procedure for Ethernet Port parameter setup in the Ethernet Port from a T2N user program. The Ethernet Port parameters consist of the following:

- (1) Local port IP address
- (2) UDP port number for computer link and PC link transmission

Here we first describe the IP address, subnet mask, and port number, which are the parameters required for TCP/IP and UDP/IP transmission.

### (1) IP address

The IP address is the address that specifies a particular port when transmitted data using the IP (internet protocol). Therefore IP addresses must be set and managed so that they are not duplicated. The following presents an overview of IP addresses.

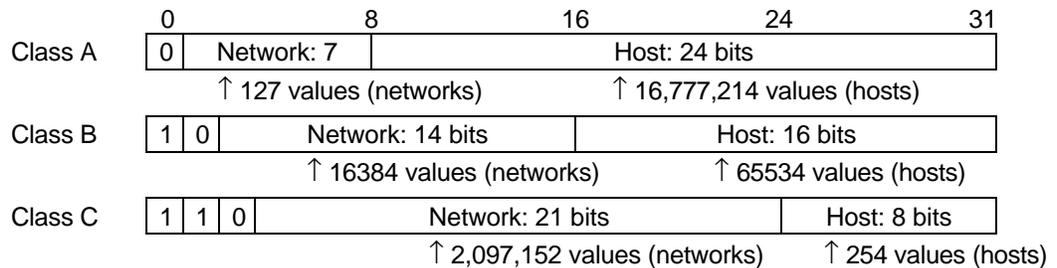
#### a. IP address format

An address consists of a network address and a host address. The network address is an address that specifies a particular network, and the host address is an address that specifies a particular port on that network. The same network address must be assigned to all ports on the same (i.e., any given) network.

An IP address is a 32-bit data item. IP addresses are written as the decimal values of each byte in the 32-bit datum, with the decimal values separated by dots.

Example: 11000000 00000000 00000000 00000001 → 192.0.0.1

IP addresses are divided into five classes, A through E, according to how many bits in the 32-bit value are used as the network address. (The networking standards also define classes D and E, but the Ethernet Port does not support them.) Figure 4.4 shows the classes A through C.



**Figure 4.4 IP Address Class Divisions**

In this system, networks with a large number of connected ports use class A addresses and networks with a small number of connected ports use class C addresses. Class B addresses are used for networks that fall between classes A and C. Expressed in decimal, the first byte of the class A through C networks will be: 0 to 127 for class A, 128 to 191 for class B, and 192 to 223 for class C.

#### b. IP address acquisition

Taking future uses into account, we strongly recommend acquiring an official IP address.

**c. Notes on IP addresses**

The following IP addresses cannot be used with the Ethernet Port, both for the local port and for remote port addresses.

- Addresses starting with 127 (7FH).
- Addresses in which the network address is all zeros or all ones.
- Addresses in which the host address is all zeros or all ones.

**d. Broadcast addresses**

The only broadcast address that can be used with the Ethernet Port is 255.255.255.255 (FF.FF.FF.FF).

The broadcast address 0.0.0.0 used on some UNIX systems (4.2 BSD) cannot be used.

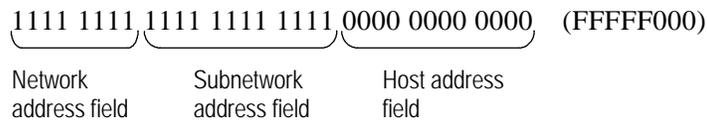
UNIX is an operating system used on engineering workstations (EWS) and other computers. There are several versions of UNIX, including ones developed by AT&T Bell Laboratories Inc., and ones developed by the University of California at Berkeley. In particular, 4.2 BSD is a version developed at Berkeley.

**(2) Subnets and the subnet mask**

One technique for managing a network with a large number of ports is to divide that network into multiple subnets. A subnet operates by taking some number of bits in the host address of the IP address for each class, and using those bits as a subnetwork address. This allows the installation to operate the network internally as multiple subnetworks, while the network is recognized externally as a single network.

The subnet mask is used to determine how many bits of the host address to allocate as the subnet address. In the subnet mask, the network address field and the subnetwork address field are set to 1 and the remaining field (the subnet host address field) is set to 0.

Example: When a subnetwork address field of 12 bits is used in a class A IP address.



**When a subnet mask is used, all ports on the subnetwork must use the same subnet mask.**

In the Ethernet Port, the subnet mask cannot be set by the user. The subnet masks (with the network address field filled with ones) for each class are generated automatically as shown in table 4.7 from the specified IP address.

**Table 4.7 Relationship between the Local IP Address and the Generated Subnet Mask on the Ethernet Port**

Local IP Address	Subnet mask
class A	255.0.0.0 (FF000000)
class B	255.255.0.0 (FFFF0000)
class C	255.255.255.0 (FFFFFF00)

Usage Recommendations
1. The PU235N/245N does not allow the user to set the subnet mask. Therefore the Ethernet Port cannot be used in systems that operate a subnetwork.

(3) Port number

An IP address identifies a particular port on a network. However, since, on any given port, there will be multiple applications using TCP/IP or UDC/IP to communicate with applications on other ports, the question of to which application the data should be passed arises.

UDP ports have the role of acting as the point of contact between UDP and applications that transfer data using UDP. UDP ports are managed using port numbers. In the same manner, TCP uses TCP ports.

**The combination of a port number and an IP address is called a socket.** The Ethernet Port transmits data to or from other ports using these sockets in the computer link protocol, the PC link protocol, and in socket interface transmission.

**The range of port numbers that can be used with the Ethernet Port is from 1024 to 65535, for both TCP and UDP.** This is because UNIX allocates port numbers 1 to 1023 for standard services.

**However, note that the same port number cannot be used for both TCP and UDP on the Ethernet Port.**

The following section describes the format of the parameter setup request.

Parameter setup request (using the SEND instruction)

a. Function

This is a request from a user program that sets up the Ethernet Port parameters in an Ethernet Port in standby mode.

b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0012H	CMD number
A+2	Local port IP address for the Ethernet Port	
A+3		
A+4	UDP port number for message	UDP port number used for computer link or PC link protocol transmission.

- IP address setup procedure: For the address 133.113.90.10

↓  
85H. 71H. 5AH. 0AH

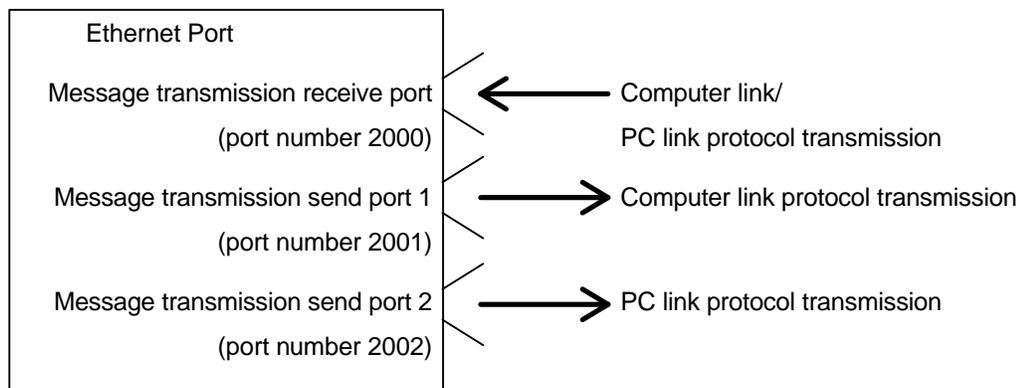
	F	8 7	0
A+2	71H	85H	
A+3	0AH	5AH	

- Message transmission UDP port number

As described in chapter 1, UDP/IP is used for both computer link and PC link protocol transmission, which are collectively referred to as message transmission. These techniques use dedicated UDP sockets that are separate from those used by socket interface transmission.

**Set the port number to a value in the range 1024 to 65535.** (UNIX allocates port numbers 1 to 1023 for standard services.)

The UDP port number specified at A+4 is the UDP port number allocated for message transmission (receive). The two port numbers following that port number are automatically allocated for message transmission (send). If computer link protocol/PC link protocol reception is performed directed at the message transmission UDP port number specified at A+4, the Ethernet Port/T2N will process the received data in the order received and send the result from one of the message transmission send ports. (See figure 4.5.)



**Figure 4.5 Relationship between Message Transmission Send and Receive Port Numbers**

As mentioned in the section on socket interface transmission, UDP port numbers used for message transmission must not duplicate UDP socket port numbers or TCP socket port numbers.

- c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

- d. Important items

- An error will occur if the Ethernet Port is in run mode or down mode.
  - Run mode - Status: Transmission error (TermSTS = 0BH)  
Detailed information: Port mode error (0040H)
  - Down mode - Status: Module error (TermSTS = 06H)
- An error occurs if a UDP port number is set to a value outside the range 1024 to 65535.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Port number error (0071H)

- e. Sample program

This sample ladder program issues a parameter setup request to the channel 1 Ethernet Port.

This program is executed by turning on the R0502 A contact.

Parameters: 12544:3100H, 00018:0012H  
29061:7185H, 02658:0A62H

```

|R0502                                     R0602 |
1|-| |-----( )--|
|           /* Parameter setup request */
|R0602                                     |
2|-| |--[ 12544 MOV RW000][ 00018 MOV RW001]-----|
|           /* Designates the module and sets the CMD number. */
|           |
|           |[ 29061 MOV RW002][ 02658 MOV RW003][ 01025 MOV RW004]-----|
|           /* Sets the IP address, Port number. */
|           |
|           |[RW000 SEND RW010][ RST R0502]-----|
|           /* Sets the request.          */
|

```

## 4.5 Operating Mode Control Requests

This section describes the Ethernet Port operating modes and the procedures for controlling the Ethernet Port operating mode from user programs on the T2N.

The Ethernet Port has the following operating modes:

### (1) Initialization mode

- This mode indicates that the Ethernet Port is executing reset processing.
- If reset processing completes normally, the Ethernet Port switches to standby mode.
- If reset processing exits with an error (e.g., if a self test error occurs) the Ethernet Port switches to down mode.
- During initialization, the Ethernet Port cannot execute requests from T2N user programs.

### (2) Standby mode

- In this mode, Ethernet Port parameter can set up.
- Ethernet Port parameters cannot be set up in any mode other than standby mode.
- After the Ethernet Port parameters have been set up, the Ethernet Port can be switched to run mode by T2N user program instructions.

### (3) Run mode

- In this mode, the Ethernet Port can communicate with other ports.
- The Ethernet Port can be switched to standby mode by T2N user program instructions.
- In run mode, transfers using computer link/PC link protocol and transfers using the socket interface can be independently enabled or disabled.

### (4) Down mode

- This mode indicates that the Ethernet Port is in an unrecoverable error state.
- Once the Ethernet Port enters the down state, it can only be restored by turning the power off (power cycling) and then on again.
- The details of the down mode state can be verified by reading the data in the T2N interface buffer memory.

(See chapter 7, "RAS Information.")

Figure 4.6 shows the Ethernet Port mode transitions.

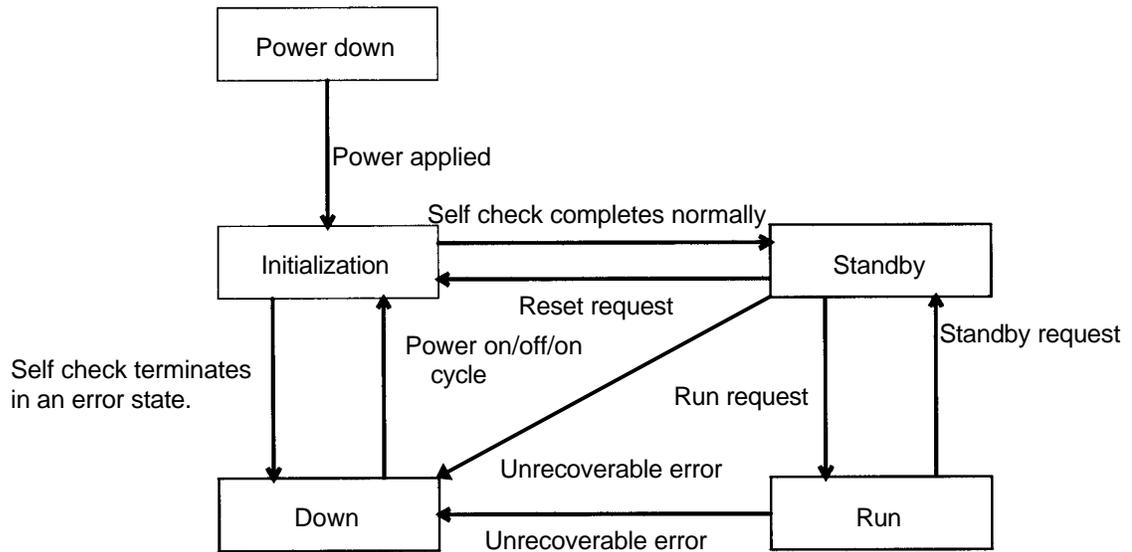


Figure 4.6 Ethernet Port Mode Transitions

This section describes the operating mode control request instruction format.

**Operating mode control request (using the SEND instruction)**

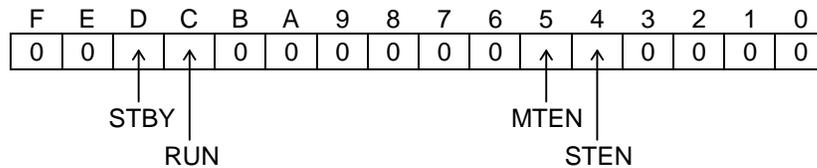
**a. Function**

This is a request from a user program that controls the Ethernet Port operating mode.

**b. Transfer parameters**

A	3100H	Module designation
A+1	CMD = 0013H	CMD number
A+2	Specified information	

Specified information



- Bit D: STBY (standby request)  
Switches the Ethernet Port from run mode to standby mode.
- Bit C: RUN (run request)  
Switches the Ethernet Port from standby mode to run mode.
- Bit 5: MTEN (message transmission enable/prohibit request)  
Computer link/PC link protocol transmission enable/prohibit  
1: Enable, 0: Prohibit
- Bit 4: STEN (socket interface transmission enable/prohibit request)  
Socket interface transmission enable/prohibit  
1: Enable, 0: Prohibit

Examples: Standby request - 2000H

- Run request - 1030H (Allows both message transmission and socket interface transmission)
- 1010H (Allows socket interface transmission only)
- 1020H (Allows message transmission only)

**c. Status (See figure 4.3 for details on bits C, D, E and F.)**

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

## d. Important items

- Values other than 1010H, 1020H, 1030H, and 2000H may not be specified for the specification information.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Incorrect control request specification (0083H)
- Simultaneous run mode and standby mode requests (30x0H) cannot be executed.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Incorrect control request specification (0083H)
- A mode that has already been established cannot be requested. However, requests that change the MTEN/STEN state within the same mode do not cause an error.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Incorrect control request specification (0083H)
- No execution is possible in down mode.
  - Status: Module error/initialization in progress (TermSTS = 06H)
- An error occurs if a control request that switches the Ethernet Port to run mode is issued without first setting the IP address.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: MAC/IP not set up (0050H)

## e. Sample program

This sample ladder program issues a operating mode control request to the channel 1 Ethernet Port.

This program is executed by turning on the R0503 A point.

Parameters: 12544:3100H, 00019:0013H  
04144:1030H

```

|R0503                                     R0603 |
1|-| |----- ( )-----|
|                                     /* Control request */
|R0603
2|-| |-[ 12544 MOV RW000][ 00019 MOV RW001]-----|
|                                     /* Designates the module and sets the CMD number. */
|                                     +[ 04144 MOV RW002]-----|
|                                     /* Setup specified information. */
|                                     +[RW000 SEND RW010][ RST R0503]-----|
|                                     /* Sets the request. */
|

```

## 5. Computer Link Protocol and the PC Link Protocol Transmission

### CAUTION

- Chapter 5 presents information related to using the functions provided by the Ethernet Port from a T2N, including the instruction (request) format, important items that require attention, and sample programs.  
Make a point of understanding the content of chapter 4 thoroughly before writing programs that use the Ethernet Port. The sample programs present basic examples of Ethernet Port usage, and should be reviewed carefully before use in an actual system.

This chapter describes computer link protocol transmission from a host computer and PC link protocol transmission with the T2N.

### 5.1 Computer Link Protocol Transmission

The T series computer link protocol transmission was developed by Toshiba for use in controlling the PROSEC-T Series programmable controllers. A host computer can, with respect to a T2N, upload or download user programs, issue control commands, or read and write register data using the T series computer link protocol.

When a T2N and a host computer communicate over an Ethernet Port using the computer link protocol, they use UDP/IP as the transmission protocol.

When a host computer transmits to a T2N using computer link protocol, the host computer's UDP socket is used. (Socket refers to the combination of an IP address and a port number.) The host computer sends commands and data from its own UDP socket to the Ethernet Port message transmission UDP socket.

When the T2N uses computer link protocol transmission, it simply performs the:

- Parameter setup (IP address and the UDP port number for message transmission) and
- Operating mode control (message transmission permission)

that were described in an earlier chapter. There is no need to create any special user programs on the T2N.

See the T Series Computer Link Operation Manual (UM-TS03\*\*\*-E008) for details on the functions, specifications, and usage of T series computer link protocol transmission.

#### Usage Recommendations

- Since UDP/IP does not provide control functions to guaranteed communication reliability, higher level protocols (such as retry) are required.
- No processing is performed if a computer link protocol transmission message is received at any point other than a message transmission UDP socket.

## 5.2 Sample Computer Link Protocol Program

This section presents a sample program that runs on the host computer (a personal computer or workstation) and that uses computer link protocol transmission.

This program issues a read/write request for 32 words from the start of the T2N D registers.

### Operating environment

- Hardware: Toshiba J-3100PV  
3Com Etherlink III (3C509/B)
- Software: PC-NFS Pro 1.1.1.0  
Microsoft VisualC++ 1.0

### Transfer parameters

- Local IP address: 133. 113. 98. 200
- Local UDP port number: 3001
- Remote IP address: 133. 113. 98. 10
- Remote UDP port number: 3001

```

/*****
/*
/* SAMPLE.H
/*
/*
/* Write and read requests can be issued to other T2N units
/* by modifying this definition file.
/*
/*
/*
/* (C) Copyright TOSHIBA Corporation 1995
/* All Rights Reserved
/*
/*****
/*-----*/
/* Definition declarations */
/*-----*/
#define MY_PORT 3001 /* Local port number */
#define DST_PORT 3001 /* Remote port number */
#define DST_IP "133.113.98.10" /* Remote IP address */
#define LOOP 10 /* Data transmission execution count */
#define DATASIZ 32 /* Data length */
/* Computer link function = word units */
#define RECVTOUT 30 /* RECVFROM timeout time */
#define REG_TYPE "D" /* Register type */
#define REG_ADR "0000" /* Register start address */

```

```
/*
 *
 * Sample program implementing register read and write processing
 * using the computer link protocol.
 *
 * This program is a sample program designed to verify the T2N
 * Ethernet port computer link function.
 * This program issues read and write requests to the specified
 * T2N registers.
 * Also note that registers in other ports can be accessed by
 * modifying the SAMPLE.H definition file.
 *
 *
 *
 * (C) Copyright TOSHIBA Corporation 1995
 * All Rights Reserved
 *
 */
/*-----*/
/* Include declarations */
/*-----*/

#define WIN31 /* This is a Windows 3.1 application. */
#define USECOMM /* The COMM API is required. */

#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <memory.h>
#include <sys%types.h>
#include <fcntl.h>
#include <errno.h>
#include <sys%socket.h>
#include <sys%timeb.h>
#include <time.h>
#include <sys%in.h>
#include <sys%netdb.h>
#include "sample.h"
```

```

/*-----*/
/* Prototype declarations */
/*-----*/
int  cl_send( int );          /* Register read/write processing */

void main()
{
int  ists;                    /* Return status */
int  isdp ;                   /* Socket identifier */
int  i ;                      /* Loop counter */
struct  sockaddr_in  Soc;
WORD  wVersionRequested = (0,1); /* WINSOC version number specification */
WSADATA  WsaDtata ;          /* WINSOC data structure */
/* initialization */
ists = 0 ;                   /* Initial value setting */
/* Windows socket initialization */

ists = WSASStartup( wVersionRequested, &WsaDtata ) ;
if( ists == 0 ){             /* If normal return: */
isdp = socket(AF_INET, SOCK_DGRAM, 0); /* Create socket (UDP) */
if( isdp != INVALID_SOCKET ){ /* If normal return: */
Soc.sin_family = AF_INET;
Soc.sin_addr.s_addr = 0 ; /* Set local IP address. */
Soc.sin_port = htons( MY_PORT ); /* Set local port number. */
/* Execute bind processing. */

ists = bind(isdp, (struct sockaddr *)&Soc, sizeof(Soc) );
if( ists != SOCKET_ERROR ){ /* If bind completed normally: */
for( i = 0; i < LOOP; i++){ /* Loop for the specified number
of times. */
ists = cl_send( isdp ); /* Register read/write processing
*/

if( ists != 0 ) /* If an error occurred in
register read/write processing: */
break ; /* Terminate processing. */
}
}
closesocket( isdp ) ; /* Socket close processing */
}
}
WSACleanup() ; /* Report WINSOC termination. */
}

```

```

/* SLW */
/*****
/*
/*      cl_send - Register read/write processing
/*
/*
/*****
/*
/*      Calling sequence
/*
/*
/*      cl_send();
/*
/*
/* Arguments:  Isdp   : int    I   : socket identifier
/*
/*             ists   : int    O   : return status
/*
/*
/*****
int cl_send( int Isdp )
{
    int  j, k      ;           /* Loop counter           */
    int  ists = 0  ;           /* Return status         */
    int  iret      ;           /* Return status         */
    int  ircvsiz   ;           /* Reception data length */
    int  isndsiz   ;           /* Data length           */
    int  icunt     ;           /* Data setup index      */
    int  idat      ;           /* Temporary             */
    char csnd[ 1000 ] ;        /* Data send buffer      */
    char crcv[ 1000 ] ;        /* Data receive buffer   */
    fd_set ibits;             /* SELECT() bit mask     */
    struct timeval timeout;   /* SELECT() timer value  */
    struct sockaddr_in  Snd, Rcv;

    iret = -1 ;
    memset( &Snd, 0x00, sizeof( Snd ) );
    memset( &Rcv, 0x00, sizeof( Rcv ) );
    memset( csnd, 0x00, sizeof( csnd ) ); /* Clear the send data buffer.
*/
    memset( crcv, 0x00, sizeof( crcv ) ); /* Clear the receive data buffer.
*/
                                           /* Set up the register write send data.
*/
    sprintf( &csnd[ 0 ], "(A01DW" ) ; /* Set up the register write command.
*/
                                           /* Set the starting register number.
*/
    sprintf( &csnd[ strlen(csnd)], "%s", REG_TYPE );
    sprintf( &csnd[ strlen(csnd) ], "%04s", REG_ADR );
    sprintf( &csnd[ strlen( csnd) ], ", " ) ;
    idat = DATASIZ ;           /* Set the data count. */

```

```

sprintf( &csnd[ strlen( csnd ) ], "%02d", idat ) ;
sprintf( &csnd[ strlen( csnd) ], ", " ) ;
icunt = 0 ; /* Comparison data storage index. */
for( j = 0, k = 0; k < DATASIZ; k++ ){ /* Set register write data. */
    sprintf( &csnd[ strlen( csnd) ], "%04X", k ) ;
    j = j + 4 ;
    sprintf( &csnd[ strlen( csnd) ], ", " ) ;
    j = j + 1 ;
    icunt = icunt + 4 ;
}
sprintf( &csnd[ (strlen( csnd )) - 1], "Yr" ) ;
isndsiz = strlen( csnd ); /* Set the send data length.
*/

Snd.sin_family = AF_INET;
Snd.sin_addr.s_addr = inet_addr( DST_IP ); /* Set the remote IP address.
*/
Snd.sin_port = htons( DST_PORT ); /* Set the remote port number.
*/
/* Send a register write request.
*/
ists = sendto( Isdp, csnd, isndsiz, 0,
    (struct sockaddr *)&Snd, sizeof( Snd ) );
if ( ists == isndsiz ){ /* If the send length was OK. */
    /* Set the SELECT argument. */
    memset(&ibits,0x00,sizeof(fd_set)); /* Set the BIT initial value. */
    timeout.tv_sec = RECVTOUT; /* Set the timeout time. */
    FD_SET(Isdp,&ibits); /* BIT setup */
    ists = select( 1, &ibits,0,0,(struct timeval *)&timeout);
    if(ists > 0){ /* If normal return: */
        /* Register write response reception */
        ists = recvfrom( Isdp, crcv, sizeof( crcv ), 0
            ,(struct sockaddr *)&Rcv, &ircvsiz );
        if ( ists > 5 ){
            if( crcv[ 4 ] == 'C' && crcv[ 5 ] == 'E' ||
                crcv[ 4 ] == 'E' && crcv[ 5 ] == 'E' ){
                return( -1 ) ; /* Register write response error */
            }
            /* Register read request creation */
            memset( csnd, 0x00, sizeof( csnd ) ) ;
            memset( crcv, 0x00, sizeof( crcv ) ) ;
            sprintf( &csnd[ 0 ] , "A01DR" );/* Register read command setup
*/
            /* Set up the starting reg. no.
*/
            sprintf( &csnd[ strlen(csnd)], "%s", REG_TYPE );
            sprintf( &csnd[ strlen(csnd) ], "%04s", REG_ADR );
            sprintf( &csnd[ strlen( csnd) ], ", " ) ;

```

```

/* Set the data count. */
idat = DATASIZ ; /* Register read data length. */
sprintf( &csnd[ strlen( csnd ) ], "%02d", idat ) ;
sprintf( &csnd[ strlen( csnd ) ], "%r" ) ;
isndsiz = strlen(csnd) ; /* Set up the data transmission length.
*/
/* Send the register read command.
*/
ists = sendto( Isdp, csnd, isndsiz, 0,
              (struct sockaddr *)&Snd, sizeof( Snd ) );
if( ists == isndsiz ){
    /* Set the BIT initial value. */
    memset(&ibits, 0x00, sizeof(fd_set));
    timeout.tv_sec = RECVTOUT; /* Set the timeout time. */
                                /* BIT setup */
    FD_SET(Isdp, &ibits);
    ists = select( 1, &ibits, 0, 0,
                  (struct timeval *)&timeout);
    if(ists <= 0){ /* If timeout and error: */
        return( -1 ) ; /* recvfrom timed out. */
    }
    /* Register read response reception */
    ists = recvfrom( Isdp, crcv, sizeof( crcv ), 0
                   ,(struct sockaddr *)&Rcv, &ircvsiz) ;
    if ( ists > 5 ){
        if( crcv[ 6 ] == 'C' && crcv[ 7 ] == 'E' ){
            iret = -1 ; /* Register read response error */
        }
        else{ /* Register read response was normal. */
            iret = 0 ;
        }
    }
}
}
}
}
return( iret ) ;
}

```

### 5.3 PC Link Protocol Transmission (Data write)

The T series PC link protocol allows one T2N on a network to write register data to another T2N or T3H.

As is the case with the computer link protocol, the PC link protocol uses UDP/IP as its transmission protocol.

The PC link main port uses a UDP socket for message transmission and transmits to the slave Ethernet Port message transmission UDP socket. The user program that runs on the T2N of the main port is discussed later.

When this function is used on the PC link slave port, the T2N simply performs the:

- Parameter setup and
- Operating mode control (message transmission permission)

that were described in an earlier chapter. There is no need to create any special user programs on the T2N.

<b>Usage Recommendations</b>	
1.	Since UDP/IP does not provide control functions to guarantee communication reliability, higher level protocols (such as retry) are required.
2.	No processing is performed if a PC link protocol transmission message is received at any point other than a message transmission UDP socket.

#### Register write request (using the SEND instruction)

##### a. Function

The register data specified by the local port (T2N) is written to the specified register on the remote port (T2N or T3H).

##### b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0021H	Command number
A+2	WordSize	(1) Transfer register data length: 1 to 485 words
A+3	SRID	(2) Local port register type code
A+4	SRegNO	(3) Local port register number
A+5	DRID	(4) Remote port register type code
A+6	DRegNO	(5) Remote port register number
A+7	Timecnt	(6) Timer count
A+8	D-IPAddress	Remote port IP address (The input format is the same as that for parameter setup requests.)
A+9		
A+10	D-UDP PortNO	Remote port message transmission UDP port number

##### (1) Transferred register data length

Specifies the transmission data length in word units. (1 to 485 words)

For the T/C registers, the data length will be between 1 and 323 words.

(2) Local port register type code

This code indicates the type of register in which the local port transmission source data is stored. See figure 5.1 for the codes that can be specified.

(3) Local port register number

Specifies the starting number of the register specified by the local port register type code.

(4) Remote port register type code

This code specifies the type of the register at the remote port to which the transmitted data will be written. See figure 5.1 for the codes that can be specified.

(5) Remote port register number

Specifies the starting number of the register specified by the remote port register type code.

(6) Timer count

Specifies a time used to limit the response time from the remote port in 0.1 second units.

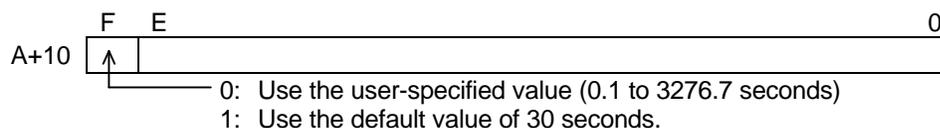
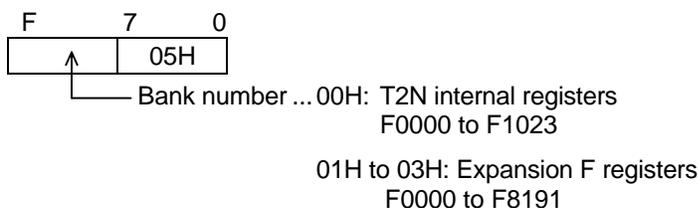


Figure 5.1 indicates the register type code that the local or remote port can specify the code. See “T-series Instruction Set (Ladder, SFC)” (UM-TS03\*\*\*-E004).

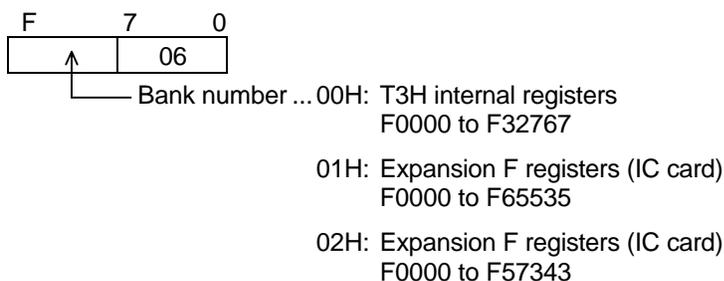
RID	Meaning	RID	Meaning
0000H	XW/YW registers	**05H	F register (0 - 8191) *1
0001H	W register	**06H	F register (0 - 65535) *2
0002H	LW register	0007H	T register *3
0003H	RW register	0008H	C register *3
0004H	D register	0009H	SW register

Figure 5.1 Register Type Code

\*1 Includes the expansion F registers. One bank has 8k words (0 to 8191).



\*2 Includes the expansion F registers(Only T3H). One bank has 64K words (0 to 65535).



\*3 The T and C registers hold data that has flags attached.

**Supplement:**

The transfer data length has a **maximum of 323 words** when data is transferred over the Ethernet.

For the T registers, **it is not possible to specify a range that spans both the basic T registers (0 to 511) and the expansion T registers (512 to 999)**. The T3H has the expansion T register, but the T2N does not have.

c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

d. Important items

- **The write operation is not synchronized with the remote port T2N/T3H scan operation.** If synchronization is necessary, applications must perform synchronization processing, such as transferring the data to a separate registers at the start of the T2N scan.
- The command number (0021H) is the same as the "Data Read".
- A transfer word count error (TermSTS = 09H) occurs if the transferred register data length exceeds the range 1 to 485 words (or 1 to 323 words for the T and C registers).
- A register specification error (TermSTS = 01H) occurs if a value other than a stipulated value is specified for the register type codes (SRID/DRID).
- A register specification error (TermSTS = 01H) occurs if 0007H (the T register) is specified for the local port register type code and a register other than the T register is specified for the remote port register type code. This also holds for the C register.
- A boundary error (TermSTS = 0AH) occurs if a range spanning both the base T registers (0 to 511) and the expansion T registers (512 to 999) is specified for the T register.
- For data storage registers, a boundary error occurs if the specified area (start register + register range) does not exist in either the local or remote ports.  
Local port: TermSTS = 0AH, remote port: TermSTS = 8AH
- A response timeout error (TermSTS = 02H) occurs if the response time from the remote port exceeds the time specified by the timer counter value.



## 5.4 PC Link Protocol Transmission (Data read)

The T series PC link protocol allows a T2N on the network to read register data from another T2N or T3H port's registers into its own registers.

As is the case with the computer link protocol, the PC link protocol uses UDP/IP as its transmission protocol.

The PC link main port uses a UDP socket for message transmission and transmits to the slave Ethernet Port message transmission UDP socket. The user program that runs on the main port is discussed later.

When this function is used on the PC link slave port, the T2N simply performs the:

- Parameter setup and
- Operating mode control (message transmission permission)

that were described in an earlier chapter. There is no need to create any special user programs on the T2N.

Usage Recommendations	
1.	Since UDP/IP does not provide control functions to guarantee communication reliability, higher level protocols (such as retry) are required.
2.	No processing is performed if a PC link protocol transmission message is received at any point other than a message transmission UDP socket.

### Register read request (using the RECV instruction)

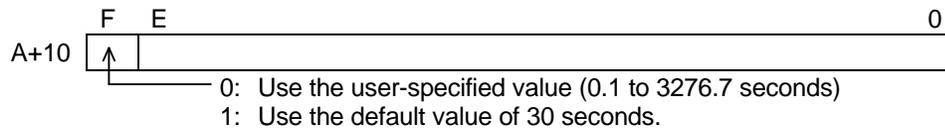
#### a. Function

The register data specified by the remote port (T2N/T3H) is read to the specified register on the local port (T2N).

#### b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0021H	Command number
A+2	WordSize	(1) Transfer register data length: 1 to 485 words
A+3	SRID	(2) Local port register type code
A+4	SRegNO	(3) Local port register number
A+5	DRID	(4) Remote port register type code
A+6	DRegNO	(5) Remote port register number
A+7	Timecnt	(6) Timer count
A+8	D-IPAddress	Remote port IP address (The input format is the same as that for parameter setup requests.)
A+9		
A+10	D-UDP PortNO	Remote port message transmission UDP port number

- (1) Transferred register data length  
 Specifies the transmission data length in word units. (1 to 485 words)  
 For the T/C registers, the data length will be between 1 and 323 words.
- (2) Local port register type code  
 This code specifies the type of the local port register into which the data read out will be stored. See table 5.1 for the codes that can be specified here.
- (3) Local port register number  
 Specifies the starting number of the register specified by the local port register type code.
- (4) Remote port register type code  
 This code specifies the type of the remote port register that holds the transmission source data. See table 5.1 for the codes that can be specified.
- (5) Remote port register number  
 Specifies the starting number of the register specified by the remote port register type code.
- (6) Timer count  
 Specifies a time used to limit the response time from the remote port in 0.1 second units.



c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

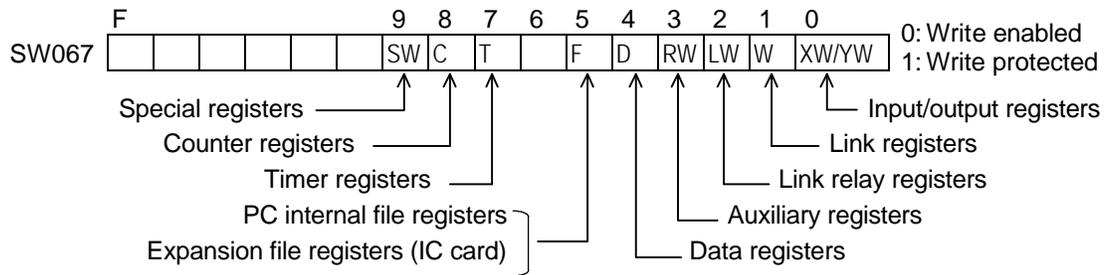
d. Important items

- **The read operation is not synchronized with the remote port T2N/T3H scan operation.**  
 If synchronization is necessary, applications must perform synchronization processing, such as transferring the data to a separate registers at the start of the T2N scan.
- The command number (0021H) is the same as the "Data Write".
- A transfer word count error (TermSTS = 09H) occurs if the transferred register data length exceeds the range 1 to 485 words (or 1 to 323 words for the T and C registers).
- A register specification error (TermSTS = 01H) occurs if a value other than a stipulated value is specified for the register type codes (SRID/DRID).
- A register specification error (TermSTS = 01H) occurs if 0007H (the T register) is specified for the local port register type code and a register other than the T register is specified for the remote port register type code. This also holds for the C register.
- A boundary error (TermSTS = 0AH) occurs if a range spanning both the base T registers (0 to 511) and the expansion T registers (512 to 999) is specified for the T register.

- For data storage registers, a boundary error occurs if the specified area (start register + register range) does not exist in either the local or remote ports.

Local port: TermSTS = 0AH, remote port: TermSTS = 8AH

- A response timeout error (TermSTS = 02H) occurs if the response time from the remote port exceeds the time specified by the timer counter value.
- Read protection can be specified for each register type on the local T2N by using the special coil settings described below. In this case, a memory write protect error (TermSTS = 04H) will occur.



- When PC link transmission or the UDP socket interface is used, if an application transmits to a nonexistent remote port, an transmission error (TermSTS = 0BH, detailed information = 0020H) occurs.**

## 5.5 Sample PC Link Protocol Transmission Program

This section presents a sample program that uses PC link protocol transmission.

This program creates increment data within the local port and writes it to the remote port's registers. Next it reads out the same area and compares it with the data within the local port to check that the function succeeded.

This program assumes that the parameter setup request and the operating mode control request (message transmission permission) have already completed.

Local port IP address: 133. 113. 98. 10 (85H. 71H. 62H 0AH)  
Local port message transmission UDP port number: 2000  
RW090: Execution count specification (1 to 32767)  
RW091: Operation when a comparison error occurs (continue: 0, stop: 1)  
RW092: Transfer register data length (1 to 485 words)  
RW093: Write time transfer source register type  
RW094: Write time transfer source register start number  
RW095: Write time transfer destination register type/read time readout source register type  
RW096: Write time transfer destination register start number/read time readout source register start number  
RW097: Read time storage register type  
RW098: Read time storage register start number  
W0991, W0990: Remote destination IP address  
133. 113. 98. 20 (85H. 71H. 62H. 14H)  
W0992: Remote destination message transmission UDP port number ..... 1025

This ladder program is executed by turn on the R0520 A contact.



```

/* Register write */
R0092
4|-| |--[12544 MOV W0903][00033 MOV W0904]-----|
|   | /* Designates the module and sets the CMD number. */
|   |
|   |[RW092 MOV W0905]-----|
|   | /* Sets the transfer register data length. */
|   |
|   |[RW093 MOV W0906][RW094 MOV W0907]-----|
|   | /* Sets the transmission source register type, start number. */
|   |
|   |[RW095 MOV W0908][RW096 MOV W0909]-----|
|   | /* Sets the transfer destination register type, start number. */
|   |
|   |[00100 MOV W0910]-----|
|   | /* Sets the response time limit (10 seconds). */
|   |
|   |[W0991•W0990 DMOV W0912•W0911][W0992 MOV W0913]-----|
|   | /* Remote destination IP address, message transmission UDP port
|   |number */
|   |
|   |[W0903 SEND W0914][SET R0093][RST R0092]-----|
|   | /* Sets the request, and starts post-completion readout. */
|   |
/* Register readout */
R0093
5|-| |--[12544 MOV W0920][00033 MOV W0921]-----|
|   | /* Designates the module and sets the CMD number. */
|   |
|   |[RW092 MOV W0922]-----|
|   | /* Sets the transfer register data length. */
|   |
|   |[RW097 MOV W0923][RW098 MOV W0924]-----|
|   | /* Sets the storage register type, start number. */
|   |
|   |[RW095 MOV W0925][RW096 MOV W0926]-----|
|   | /* Sets the transfer destination register type, start number. */
|   |
|   |[00100 MOV W0927]-----|
|   | /* Sets the response time limit (10 seconds). */
|   |
|   |[W0991•W0990 DMOV W0929•W0928][W0992 MOV W0930]-----|
|   | /* Remote destination IP address, message transmission UDP port
|   |number */
|   |
|   |[W0920 RECV W0931][SET R0094][RST R0093]-----|
|   | /* Sets the request, and starts post-completion comparison. */

```



## 6. Socket Interface Communication

**⚠ CAUTION**

1. Chapter 6 presents information related to using the functions provided by the Ethernet Port from a T2N, including the instruction (request) format, important items that require attention, and sample programs. That chapter also presents items considered necessary when using the Ethernet Port. Make a point of understanding the content of chapter 4 thoroughly before writing programs that use the Ethernet Port. The sample programs present basic examples of Ethernet Port usage, and should be reviewed carefully before use in an actual system.

This chapter describes the functions and positioning of the socket interface, and the differences between the UDP socket interface and the TCP socket interface. Next, this chapter presents notes on using the Ethernet Port socket interface and describes the procedures for using this functionality.

### 6.1 Overview

This section describes the functions and positioning of the socket interface, and the differences between the UDP socket interface and the TCP socket interface.

#### 1. Positioning and functionality

The socket interface is a programming interface that allows user programs to use the TCP/IP and UDP/IP functions. Socket interface communication is a communication technique that has become standard on engineering workstations (EWS) and other networked computer systems.

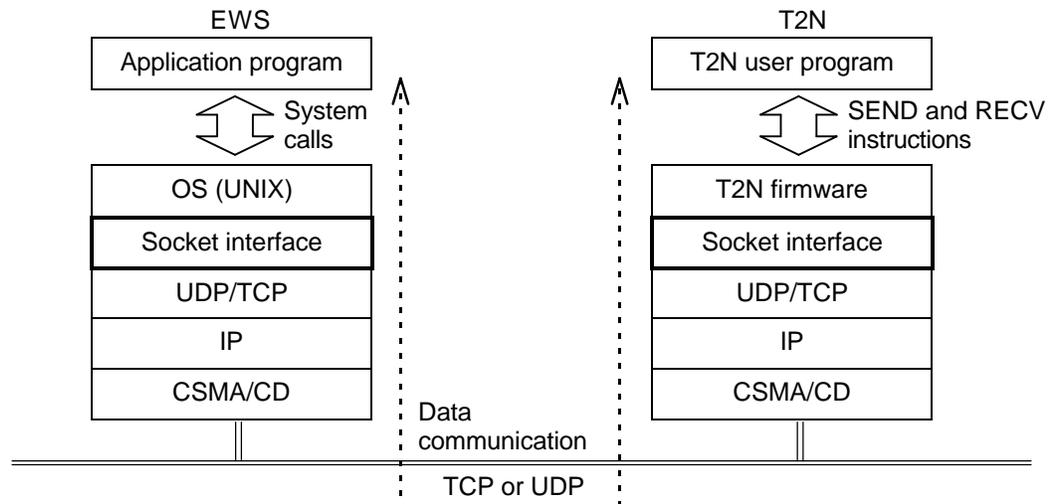
The socket interface can be used for communication between host application programs and user programs running on the T2N.

The Ethernet Port socket interface provides eight sockets, and the communication protocol (TCP/IP or UDP/IP) used with the remote destination can be specified for each socket. User programs on the T2N use the SEND and RECV instructions to use the socket interface.

TCP/IP (Transmission Control Protocol/Internet Protocol)

UDP/IP (User Datagram Protocol/Internet Protocol)

Figure 6.1 shows the positioning of the socket interface.



**Figure 6.1 Positioning of the Socket Interface**

## 2. Differences between the UDP socket interface and the TCP socket interface

### a. UDP socket interface (See figure 6.2.)

- Provides an in/out port (socket) for data communication between a transmission source port and a transmission destination port.
- Data for the transmission source and transmission destination (IP addresses and port numbers) is specified in the send/receive data.
- Since the transmission source and transmission destination are specified in each data item, a UDP socket can communicate with multiple remote UDP sockets.
- The UDP socket interface provides no control functionality such as reception verification using, e.g. ACK (acknowledge) responses, or retransmission processing to assure transmission reliability. Therefore, reliability must be assured by higher level protocols implemented by user programs.

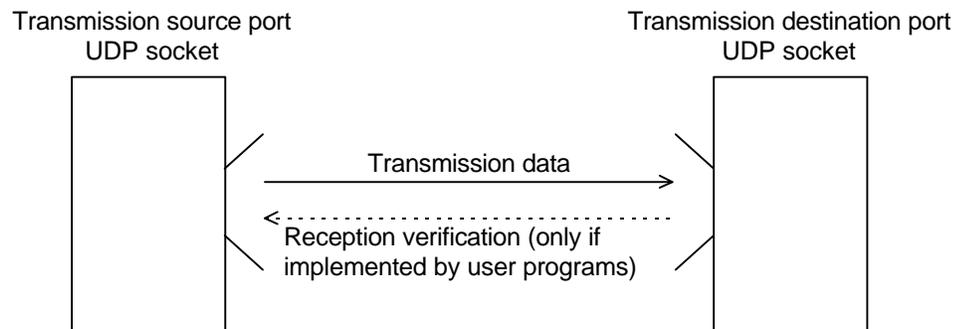


Figure 6.2 UDP Sockets

b. TCP socket interface (See figures 6.3 and 6.4.)

- In TCP communication, a pipe-like virtual communications line (connection) is established between sockets on two ports that are communicating (sending and receiving), and since the protocol handles control of reception verification and retransmission processing, it provides a highly reliable communication.
- There are two methods for establishing a connection: passive open and active open. In passive open, the local socket is put into a state in which it is waiting for a request to establish a connection from another port. In active open, a request is issued for the establishment of a connection with a socket in the passive open state.
- When establishing a connection between two ports, one port must first open a socket in passive open mode, and the other port must perform an active open on a local socket with respect to a remote socket that is in the passive open state.
- Since the passive open socket port "provides" data transmission and other services, it is called the "server."
- Since the active open socket port "requests" services, it is called the "client."
- A socket for which a connection has been established is unable to transfer data with any other sockets unless that connection is first released.
- Since sockets are connected by connections, there is no need to specify the transmission source or transmission destination with every data item transmitted.

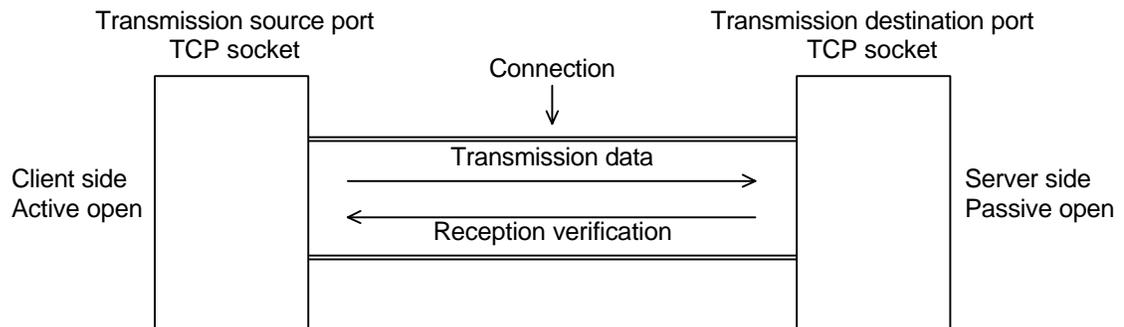


Figure 6.3 TCP Socket (Connection)

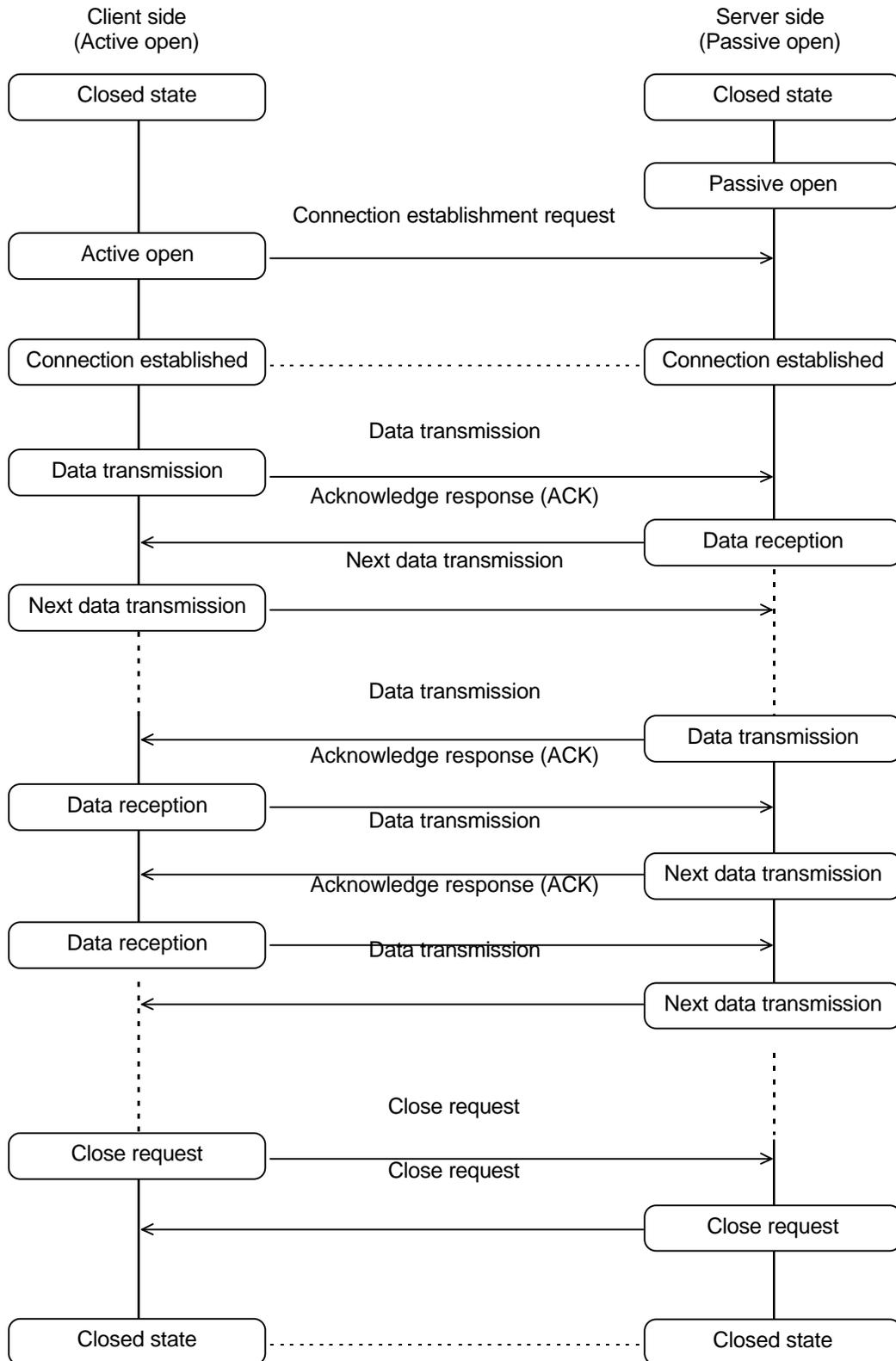


Figure 6.4 TCP Connection Communication Procedure (Overview)

## 6.2 Ethernet Port Socket Interface Usage Notes

This section presents points that require attention when using the socket interface on the Ethernet Port for data transmission.

### 1. Data handling and data segmentation in UDP socket interface transmission and reception

- a. In UDP socket interface transmission, data sent in a single transmission is handled by the receiving Ethernet Port as a single data unit, and that data is transmitted to the T2N in a batch operation by a receive request.
- b. The size of data units that can be handled by the Ethernet Port socket interface in a single operation is limited to a maximum of 2000 bytes. If the sending port sends a data unit larger than 2000 bytes, the Ethernet Port UDP socket interface will not be able to receive that data. Users should determine the maximum size of the data units to be sent or received at the system design stage.
- c. Transmitted data that exceeds 1472 bytes is divided (fragmented) into units of 1472 bytes. This is the limit of the length of packets (data units flowing on the network) transmitted over the network (10BASE-T). (See figure 6.8.)

For example, when transmitting a 2000 byte data set, which is the largest data size that can be handled by the Ethernet Port, the phenomenon shown in figure 6.5 can occur.

- (1) The sending T2N user program requests a transmission of 2000 bytes.
  - (2) The transmitted data is divided into two fragments, A (with 1472 bytes) and B (with 528 bytes) by the send side Ethernet Port.
  - (3) A and B are put onto the network in order.
  - (4) The receive side Ethernet Port connects the received data A and B, recovering the original data.
  - (5) The data is accepted by a receive request from the receive side T2N for 2000 bytes of data and the data is stored in the specified register. (See section 6.4 (3).)
- d. In the UDP socket interface, after the transmitted data is reassembled to the original data it is passed to the user program and stored in the specified register.

(The fragments A and B are reassembled to the original data by UDP/IP processing.)

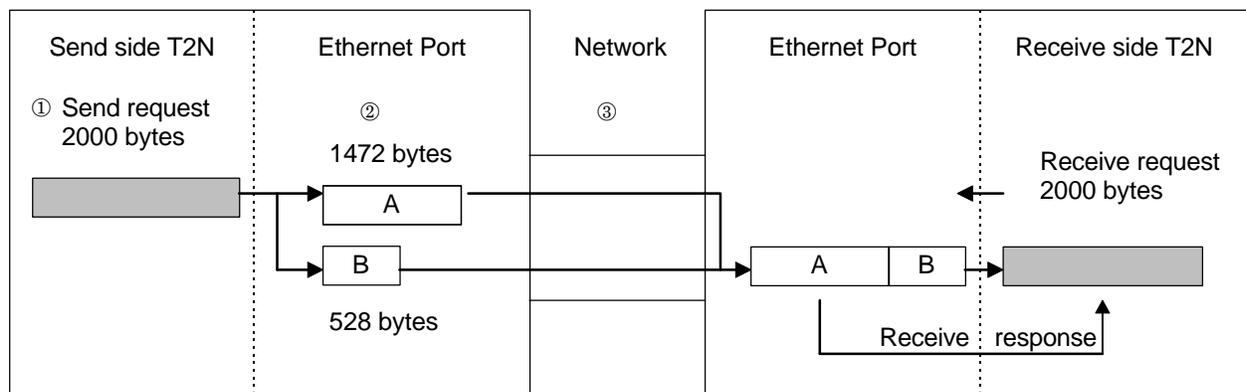


Figure 6.5 Transmission Data Fragmentation in the UDP Socket Interface

- e. The T2N receives the data which specifies the size of reception data in the transfer parameter from the Ethernet Port.
- f. **Allocate an area one word larger than the transmitted data size as the T2N register area that will hold the received data.** As the result of a receive request, information indicating the size of the transmitted data for the single operation plus the data transmitted in the single operation will be stored in that area.
- g. **In receive request programs from the T2N, set the size of reception data to a value equal to or larger than that of the maximum size for the transmission data in a single operation.** If the size of reception data is smaller than the transmitted size of transfer data to socket, the data that exceed the specified size of reception data is canceled.

2. Data handling and data segmentation in TCP socket interface transmission and reception

- a. When using the TCP socket interface for data transmission, the data transmitted in a single operation is handled by the receive side Ethernet Port as one section of a continuous data stream that comes from the sender. (See figure 6.6.)  
  
In this technique, the data is not handled as a single unit, as it is in UDP socket interface transmission.
- b. The amount of data transmitted to the T2N from the TCP socket interface due to a T2N receive request will vary depending on both the timing with which the T2N issues the receive request, and the timing with which the data arrives at the TCP socket interface.
- c. In the Ethernet Port TCP socket interface, up to 4380 bytes of received data can be stored per socket. Also note that up to 2000 bytes of data can be transferred to the T2N for a single receive request from the T2N. For example, if the remote port sends 800 bytes at a time, then the amount of received data per socket may be either 800, 1600, 2400, 3200, or 4000 bytes. If the amount of received data is 800 or 1600 bytes, those 800 or 1600 bytes can be acquired by the T2N by a receive request from the T2N. In cases where the amount of received data is 2400, 3200, or 4000 bytes, 2000 bytes will be transferred by the first receive request from the T2N, and the remaining 400, 1200, or 2000 bytes can be transferred by issuing another receive request from the T2N.
- d. Therefore, to handle one unit of transmitted data as a single unit of data on the receiving side as well, at the system design stage users must analyze both the data format (head/tail delimiting codes, data sequencing) and the amount of data to be transmitted, and provide routines to decode the transmitted data in the receiving T2N user software.
- e. **Allocate an area one word larger than the transmitted data size as the T2N register area that will hold the received data.** As the result of a receive request, information indicating the size of the transmitted data for the single operation plus the data transmitted in the single operation will be stored in that area.

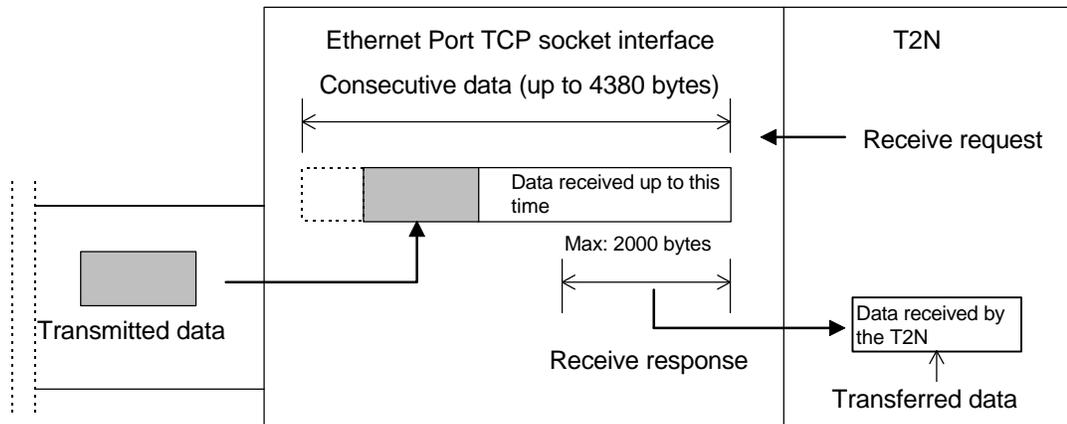


Figure 6.6 Handling of Transmitted Data in the TCP Socket Interface

- f. When the TCP socket interface is used, transmitted data is divided (fragmented) into units of a size determined when the connection between the two sockets was established. The upper limit of the fragment size is 1460 bytes. This is the limit of the length of packets (data units flowing on the network) transmitted over the network. (See figure 6.8.)

For example, when transmitting a 2000 byte data set, which is the largest data size that can be handled by the Ethernet Port, the phenomenon shown in figure 6.7 can occur. The largest segment size that can be received in this case is 1460 bytes.

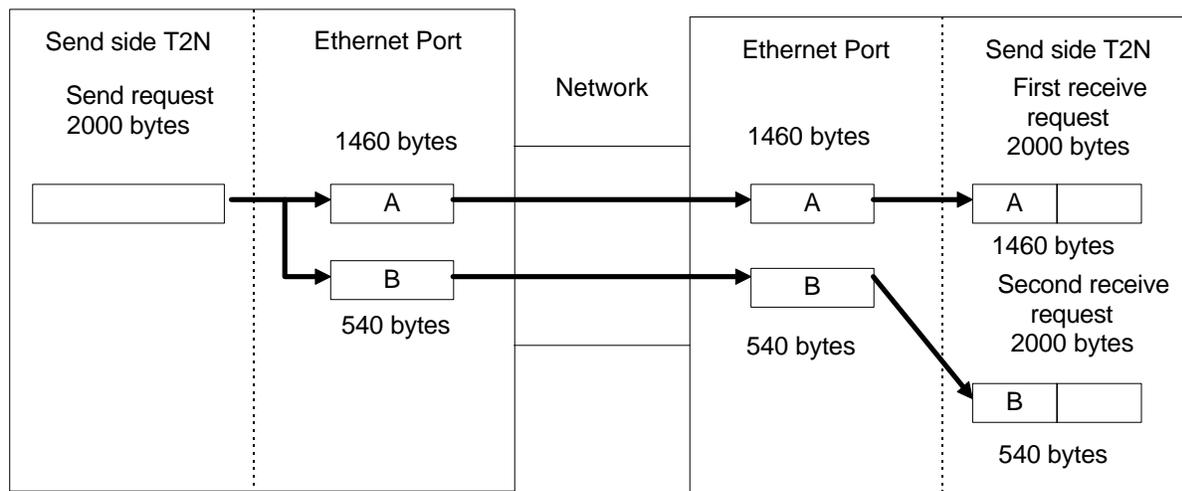
- ① The sending T2N user program requests a transmission of 2000 bytes.
- ② The transmitted data is divided into two units called **segments**, A (with 1460 bytes) and B (with 540 bytes) by the send side Ethernet Port.
- ③ A and B are put onto the network in order.

The data that arrived at the receive side Ethernet Port is acquired by a receive request for 2000 bytes from the T2N, and stored in the user specified register area. (See section 6.3 (7).)

The amount of data that can be acquired by a TCP receive request differs depending on the timing with which the T2N issues the receive request and the timing with which the data arrives at the Ethernet Port.

- If data has not yet arrived at the receive side Ethernet Port when the receive request is issued:
  - When segment A arrives at the Ethernet Port, only segment A will be passed to the requester.
- If only segment A has arrived at the receive side Ethernet Port when the receive request is issued:
  - Only segment A will be passed to the requester.
- If segments A and B have arrived at the receive side Ethernet Port when the receive request is issued:
  - A data item consisting of A and B will be passed to the requester.

If B was not received, another receive request can be issued from the T2N to acquire B.



**Figure 6.7 Transmission Data Fragmentation in the TCP Socket Interface**

- g. When the transmission data is fragmented as discussed above, the amount of data transferred to the T2N in a TCP socket interface receive request differs depending both on the timing with which the T2N issues the receive request and the timing with which the data arrives at the Ethernet Port.
- h. To handle single units of transmission data on the receiving side as single units as well, the user software on the receiving side must recognize the end of the transferred data (either by using counts in the transmitted data or by including an end marker in the transmitted data) and iterate receive requests until all the transmitted data has been received.
- i. **User programs must store all incoming data so that the register area used to transfer data to the T2N is not overwritten when using iterated receive requests.**

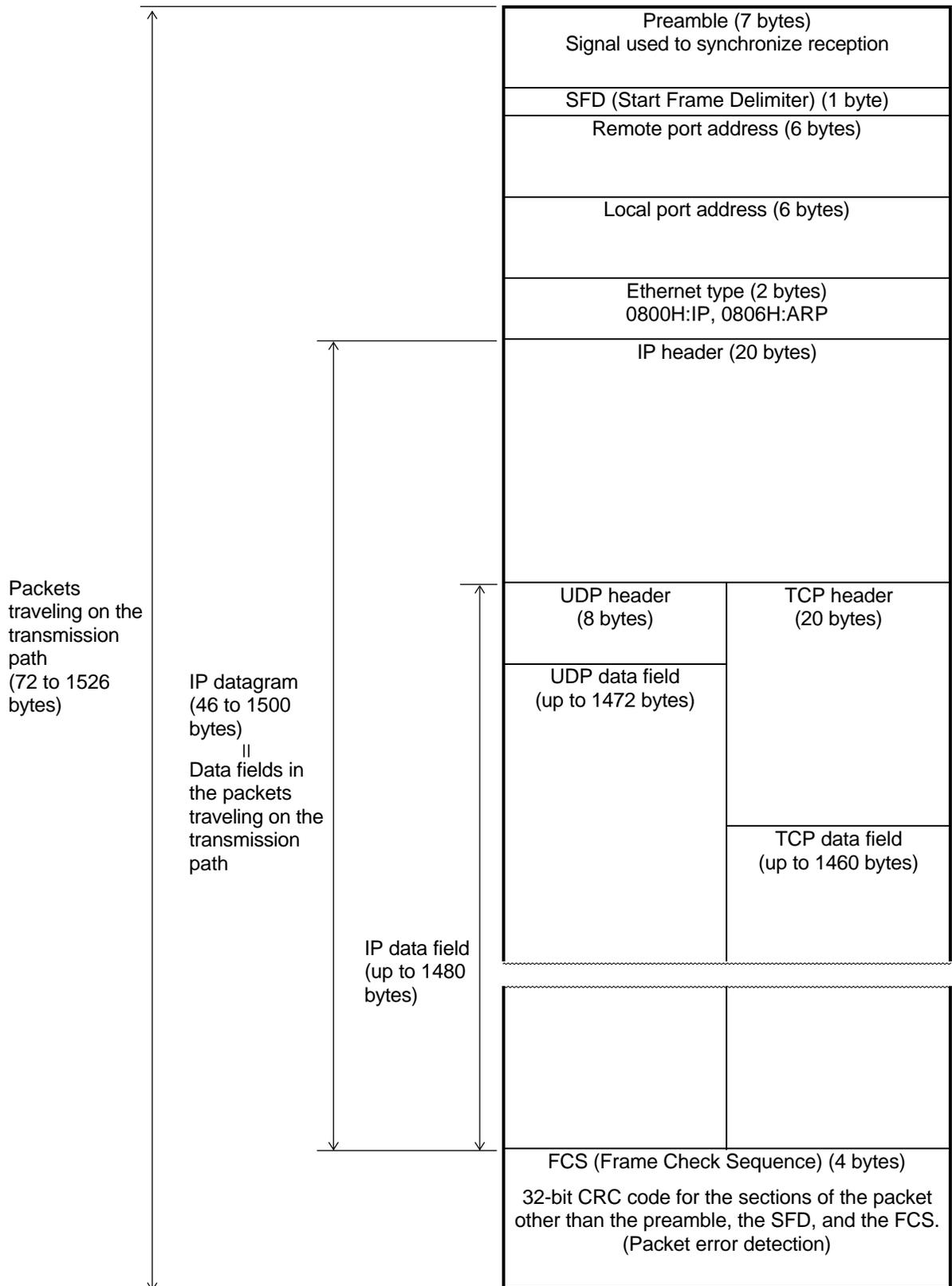


Figure 6.8 Packet Format/IP Datagram

### 3. Socket interface port numbers

- a. **Set the port numbers for local TCP and UDP sockets to values in the range 1024 to 65535.** The Ethernet Port will issue an error if any other values are used. This is because the port numbers from 1 to 1023 are allocated for UNIX workstation standard services.
- b. There are cases where there are reserved ports with numbers outside the 1 to 1023 range on specific workstations. Verify this on the workstations actually being used, and be sure that these reserved port numbers are not duplicated on the Ethernet Port.
- c. The same port number cannot be assigned to multiple sockets for TCP and UDP sockets. Be especially careful not to duplicate UDP port numbers used for message transmission (computer link protocol and PC link protocol transmission).
- d. When TCP sockets and UDP sockets are used together in the same Ethernet Port, the same port number cannot be used for both TCP and UDP sockets.

### 4. Notes that apply to both the UDP and the TCP socket interface

- a. **Careful attention to changes in bits in the socket status is required when using socket interface send/receive requests.**
- b. When using multiple Ethernet Port functions at the same time, communication processing may become slower, depending on the details of the user programs executing.
- c. Communication efficiency may degrade if the network to which the Ethernet Port is connected is heavily loaded.
- d. Since processing a close request requires processing the send and receive requests issued to that socket before the close request was issued, a significant amount of time may be required to process some close requests.
- e. The Ethernet Port includes a common transfer/reception buffer (of about 60k bytes) to hold data. This common transfer/reception buffer is shared by all open sockets. In the Ethernet Port TCP socket interface, up to 4380 bytes of received data can be stored per socket. In the case of using all TCP sockets, this buffer of about 60k bytes do not become full. In the case of UDP socket, the amount of reception data is no limit. So, if the T2N does not receive the reception data in the socket, the common transfer/reception buffer may become full. In this case, new incoming data will be discarded and the T2N can not transmit data, either (the send request is insufficient resources error). To avoid this state, user programs should either continuously issue receive requests and readout any data in the reception buffer, or else monitor the socket status (See section 6.3) for the presence of reception data and perform reception processing as required. If the insufficient resources error to not receive data occurs, it is possible to cancel that the socket, the reception data becomes full, is closed.
- f. Since Ethernet Port receive requests (both TCP and UDP) put the object socket in a wait state waiting for data from the remote port, transmission over that socket is not possible.  
  
If full-duplex communication with the remote port is required, set up two sockets, one for transmission and one for reception.
- g. There are a total of eight requests used for socket interface transmission. Of those, the T2N waits internally for completion for the timeout time for five requests, namely,
  - UDP open, send, and close requests
  - TCP send and close requests

If the timeout time expires, the "transmission complete timeout (no instruction response)" error is issued. Since this error has the same code as module failure errors, first check whether or not the Ethernet Port has failed before attempting to recover from the error.

- h. User programs specify the time limit until completion for the following three requests used in socket interface transmission:
- UDP receive request
  - TCP open request
  - TCP receive request

If this time limit is exceeded, a "timeout" error is issued. In particular, the Ethernet Port reports the timeout error to the T2N.

- i. The Ethernet Port uses a priority ordering in processing sockets 1 through 8. Therefore, systems constructed so that socket 1 is activated frequently may not be able to process socket 8, resulting in "Transmission completion timeout" errors occurring. When constructing a system, take the socket utilization conditions into account when allocating sockets.

Since the send/receive processing for each socket requires about 50 ms, applications that issue send or receive requests to a given socket should leave an interval of at least 50 ms times the number of sockets used between each request.

$$(\text{Interval between requests to the same socket}) \geq (\text{Number of sockets used} \times 50 \text{ ms})$$

Similarly, the above intervals averaging 50 ms should be left between transmissions from remote ports to the local port.

$$(\text{Interval between remote port transmissions}) \geq (\text{Number of remote ports} \times 50 \text{ ms})$$

## 5. UDP socket interface notes

- a. In transmission using UDP sockets, the user program must implement any processing required to assure transmission reliability, such as verifying the reception of transmitted data and retransmission.
- b. It is possible to broadcast to all ports on the network using UDP sockets.
- Send side: Set the transmission destination address to "255.255.255.255". This is "FF.FF.FF.FF" in hexadecimal. The "0.0.0.0" used by some UNIX versions (in particular, 4.2 BSD) cannot be used.  
Set the UDP port number for the socket to which you want to issue a broadcast as the transmission destination object socket UDP port number. It is convenient to prepare a dedicated socket in advance if broadcast transmission is to be used.
- Receive side: Specify "0.0.0.0" as the transmission source IP address.  
Specify the UDP port number for the broadcast transmitting socket as the transmission source UDP port number.
- c. Data transmission between sockets on the same port is not supported by UDP.
- d. **When PC link transmission or the UDP socket interface is used, if an application transmits to a nonexistent remote port, an transmission error (TermSTS = 0BH, detailed information = 0020H) occurs.**

## 6. TCP socket interface notes

- a. When a passive open is executed, the specified socket is put in the active open wait state. If the remote IP address (DIPAddress) and the remote port object socket TCP port number (DTCP\_PORTNO) values are set to 0, the socket will be able to handle active opens from any remote port.
- b. A connection is established by an active open from another socket applied to a socket in the passive open state. A connection cannot be established by a passive open from another socket applied to a socket in the passive open state. Similarly, a connection cannot be established by an active open from another socket applied to a socket in the active open state. That is, the roles of two sockets for which a connection is to be established must be determined in advance.
- c. For a socket in the passive open state, it is not possible to open multiple connections from other ports by issuing multiple active opens.
- d. Connections cannot be established between sockets on the same port.
- e. If transmission is impossible due to network congestion or other reason when a send request is executed, the send request is stored internally in the Ethernet Port so that it can be sent later. The number of items that can be stored is three per socket. If the Ethernet Port cannot store the request, it returns an error.
- f. If a close request is issued first for a connection that is being established, then it will be possible to execute the next open request for the object sockets at the point when the close request processing completes normally. On engineering workstation and personal computer systems, TCP sockets on the port that performed the close processing for the open connection cannot be opened again for a fixed period, usually about 1 to 2 minutes, although this time varies between systems.
- g. When the Ethernet Port issues a close request first, the Ethernet Port waits 10 seconds. If the T2N using the same port number reopen request during waiting for 10 seconds, the T2N becomes “LAN controller driver error”.
- h. If a close request comes from the remote port TCP socket while a connection is established, the local port must also close the socket. As a technique for checking for the reception of close requests from the remote TCP socket, user programs should monitor the RCLOSE and CONN bits in the socket status. (See section 6.3.) RCLOSE is set to 1 and CONN is set to 0 when a close request is received from the remote TCP socket. The Ethernet Port will return an error if send or receive requests are issued in this state. A receive request in the receive wait state also returns an error.
- i. The Ethernet Port in the current state does not support a keep alive function for TCP connections. Verification from the Ethernet Port side to other ports is not performed. However, the Ethernet Port does respond to verifications from other ports. (Responding to verifications from other ports is performed independently of user programs.) Therefore, the disappearance of remote ports cannot be detected at the TCP level. We recommend using the detection techniques described in items j and k below to detect this condition.

Keep alive function: A function that checks, at the TCP protocol level, whether or not the connection is operating normally if there has been no activity over the connection for a certain fixed period.

- j. For a TCP receive request, the Ethernet Port simply waits for data from the remote port. In situations where data is sent periodically from the remote port, reception wait timeouts, i.e. no response from the remote port states, can be detected by setting a reception wait time limit. The following techniques, among others, can be used to detect no response states when data is sent with no fixed period.

- Periodic execution of existence verification requests (See section 7.3.)
- The techniques described in the next section, which use a separate TCP connection

One of the following problems may have occurred when there is no response from the remote port:

- The remote port may have gone down.
- Power may have been lost.
- The remote ports and/or local port may have become disconnected from the network.

- k. With a connection established, if the no response state from the remote port continues when data is sent and the Ethernet Port internal resend processing times out (the timeout time is about 1 minute), the NOACK bit in the socket status (See section 6.3.) will be set to 1. User programs should close such sockets.

Also note that the Ethernet Port will return an error if send or receive requests are issued in this state.

User program send requests are seen as completing as soon as they are received by the Ethernet Port. This means that **even if the remote port goes to the no response state, a number of send requests equal to the Ethernet Port internal storage capacity (i.e., three requests) will complete normally**. When the number of send requests exceeds the Ethernet Port internal storage capacity, errors will be returned for all further send requests.

- l. Segments with the reset bit (RST) set to 1 (reset segments) are not supported by the Ethernet Port. If a reset segment is received from the remote port, the NOACK bit in the socket status (see section 6.3) will be set to 1 about 32 seconds later. Such sockets should be closed.

Reset segment: A transmitted segment used to forcibly close a connection from one of the ports. Connections are forcibly closed in this manner if an error that cannot be recovered from by normal means such as retransmission has occurred, or if a port has recovered after having gone down.

- m. **In the Ethernet Port TCP socket interface, when reception data in the Ethernet Port is read out by a receive request from the T2N, the Ethernet Port sends an ACK (acknowledge response) to the port that sent the data.** Since an ACK will not be sent to the sending port if the received data remains in the Ethernet Port if the T2N does not read out that received data, the sending port will resend the data. If this state continues for a predetermined period, the sending port may decide that the TCP connection is abnormal and close the connection. To prevent this from occurring, the T2N software should monitor the RCV (receive data present) bit and issue receive requests when necessary.

### 6.3 Socket Interface Information

The information for the eight socket interface sockets held by the Ethernet Port can be read out with the T2N READ instruction. The Ethernet Port holds five words of socket status information for each socket. User programs should access this information as necessary when using socket interface transmission.

- Remote port TCP/UDP port number
- Remote port IP address
- TCP/UDP port number for this socket
- Socket status

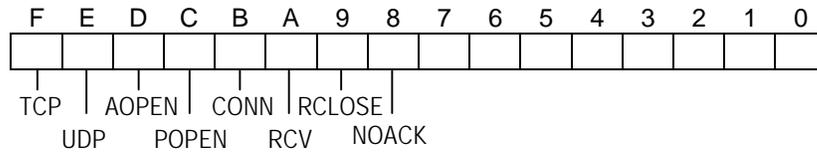
Table 6.1 lists the structure of the socket interface information and the extended memory area addresses.

**Table 6.1 Socket Interface Information**

	Socket identifier							
	1	2	3	4	5	6	7	8
Remote port number (TCP only)	0F4BH	0F53H	0F5BH	0F63H	0F6BH	0F73H	0F7BH	0F83H
Remote port IP address (TCP only)	0F4CH 0F4DH	0F54H 0F55H	0F5CH 0F5DH	0F64H 0F65H	0F6CH 0F6DH	0F74H 0F75H	0F7CH 0F7DH	0F84H 0F85H
Local port TCP/UDP port number	0F4EH	0F56H	0F5EH	0F66H	0F6EH	0F76H	0F7EH	0F86H
Socket status	0F4FH	0F57H	0F5FH	0F67H	0F6FH	0F77H	0F7FH	0F87H

For UDP sockets, the remote port number and IP address data are invalid.

The socket status has the following structure.



Bit no.	Bit	Condition under which the bit is set to 1	Condition under which the bit is reset to 0
F	TCP	TCP used by the socket	The socket closed in response to a close request
E	UDP	UDP used by the socket	The socket closed in response to a close request
D	AOPEN	TCP socket in active open state	The socket closed in response to a close request
C	POPEN	TCP socket in passive open state or UDP socket in open state	The socket closed in response to a close request
B	CONN	TCP connection established	Disconnected by the local port with a close request Disconnected by the remote port
A	RCV	Received text present	No received text
9	RCLOSE	An established TCP connection disconnected by the remote port	The socket closed in response to a close request
8	NOACK	No response received from the remote port for a local port send operation in a state where a TCP connection was established, and the TCP resend function resulted in a timeout (because the remote port was down, etc.).	The socket closed in response to a close request
		If the wait time exceeds the set value during a TCP active open request. This bit is not set to 1 for a TCP passive open request timeout.	When a TCP active open request is reissued.

A socket status in which neither AOPEN nor POPEN is set to 1 indicates that the socket is in the closed state.

**Important items**

This area is a read-only area. Do not attempt to write this data with the WRITE instruction. It will no longer be possible to acquire correct data after such an operation.

**Sample:** Socket interface information readout program

```

|
|R0000
1|-| |-[03907 MOV RW010][00005 MOV RW011][H00FE_READ RW010 -> D1000]--|
|
READ instruction description
H00FE: Module designation. The Ethernet Port must specify H00FE.
RW010: Specifies the start address of the socket status information.
        If 03907 (0F4BH) is specified, the socket status information for
        socket 1 will be read out.
RW011: Specifies the number of words (00005) to be read out.
D1000: Specifies the starting address of the area to hold the read out socket
        status information.
In this case, the read out downloaded information is stored at locations
D1000 to D1004.
⊙ Setting R0000 to 1 will store the socket status information for socket
  1 at locations D1000 to D1004.

```

## 6.4 Using the Ethernet Port Socket Interface

This section describes the requests used by T2N user programs to use UDP and TCP sockets. There are eight types of request as listed below.

- UDP socket (open request, send request, receive request, close request)
- TCP socket (open request, send request, receive request, close request)

### 1. UDP open request (using the SEND instruction)

#### a. Function

This request opens any of the eight sockets.

Transmission protocol: UDP/IP

#### b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0031H	Command number
A+2	SportNO	Socket identifier (1 to 8)
A+3	UDP_PORTNO	Local port specified socket UDP port number (1024 to 65535)

#### c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

SportNO: Socket identifier (1 to 8)

TermSTS: See tables 4.3 and 4.4.

Detail information: See table 4.5.

#### d. Important items

- After this request completes normally, UDP send, receive or close requests can be sent to the socket.
- An error is returned if an open request is issued for an already open socket.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Already open (0080H)
- An error is returned if a value other than 1 to 8 is specified as the socket identifier.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Illegal socket identifier (0082H)
- An error is returned if a value other than a local port specified socket UDP port number is specified.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Port number error (0071H)

- Not only must the specified socket UDP port number on the local port not overlap with UDP port numbers used for message transmission and previously used UDP port numbers, but it also must not overlap with any TCP port number. An error will occur if the port number overlaps.

Status: Transmission error (TermSTS = 0BH)

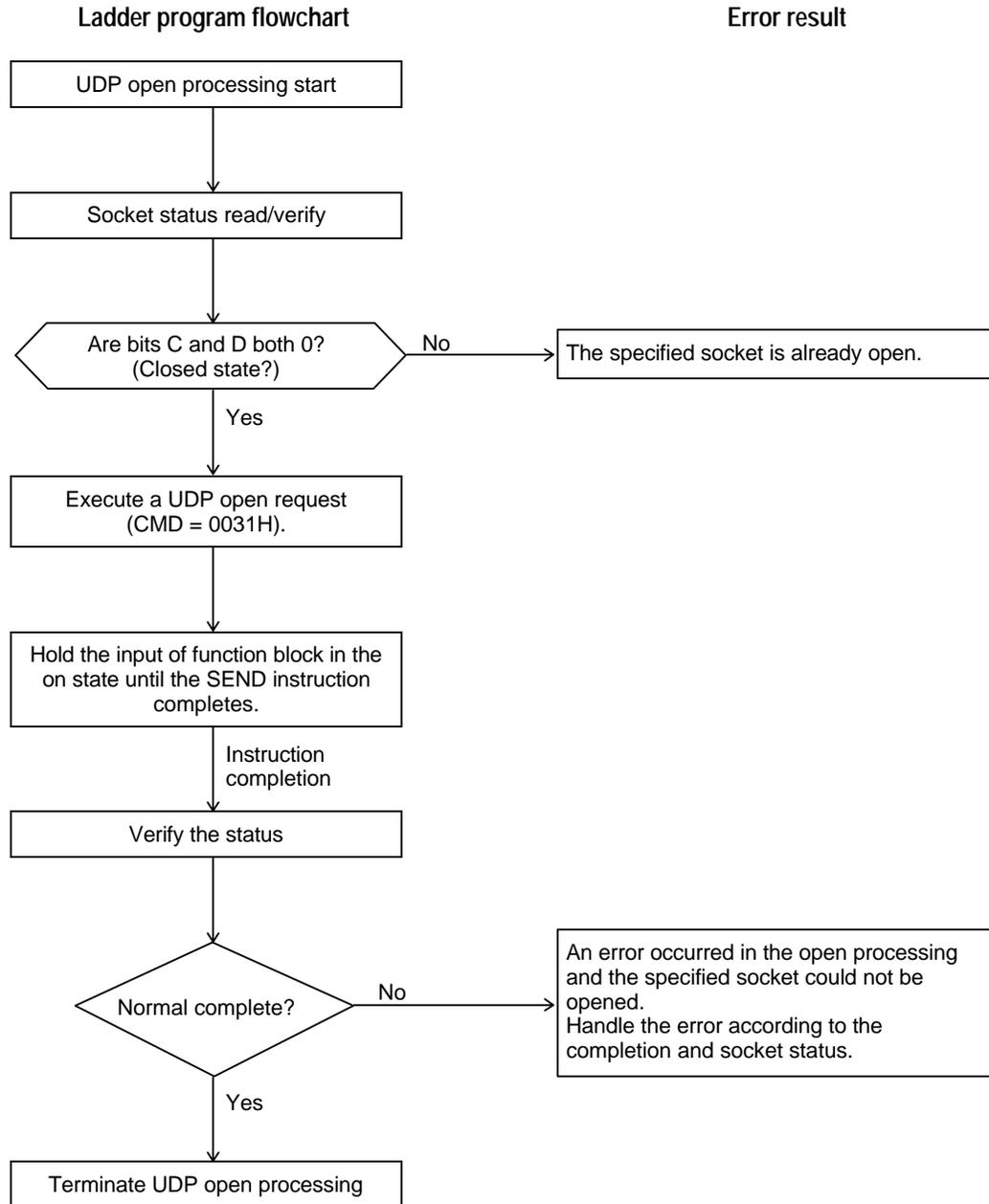
Detailed information: Already open (0080H)

- If the T2N internal completion wait time limit (2 seconds) is exceeded for this request, the error listed below is returned.

Status: Send completion timeout (TermSTS = 06H)

In this case, first verify that the Ethernet Port is not in the down state by checking the station status, and then verify the open/closed state of the socket in the socket status.

e. UDP open processing example



2. UDP send request/broadcast send request (using the SEND instruction)

a. Function

This request sends local port register data using an open UDP socket.

Amount of data sent: 1 to 1000 words

b. Transfer parameters

A	3100H	Module designation, 1: channel number
A+1	CMD = 0032H	Command number
A+2	SportNO	Socket identifier (1 to 8)
A+3	DIPAddress	Transmission destination IP address
A+4		(The input format is the same as that for the parameter setup request.)
A+5	D_UDP_PORTNO	Transmission destination object socket UDP port number
A+6	WordSize	Transmission data length: 1 to 1000 words
A+7	SRID	Transmission data storage register type code
A+8	StreqNOD	Transmission data storage register number

Transmission data storage register type code: Type code for the register that holds the transmission data. (See Figure 5.1)

Transmission data storage register number: Starting number of the registers that hold the transmission data.

c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

SportNO: Socket identifier (1 to 8)  
 TermSTS: See tables 4.3 and 4.4.  
 Detail information: See table 4.5.

d. Important items

- The T/C register flag data is not transmitted if the T/C registers are not specified as storage registers for transmitted data.

- An error is returned if a send request is issued for a socket that is not yet opened.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Unopened (0081H)

- An error is returned if a value other than 1 to 8 is specified as the socket identifier.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Illegal socket identifier (0082H)

- If the IP address of the local port is specified as the transmission destination IP address, it will be possible to send data, but it will not be possible to receive that data with a receive request.

- Values in the range 1024 to 65535 can be specified as the object socket UDP port number for the transmission destination. An error will be returned if a value outside that range is specified.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Port number error (0071H)

- An error will be returned if a value of 0 words or 1001 or more words is specified as the transmission data size.

Status: Transmission word count error (TermSTS = 09H)

- The T2N performs an area check for the register area based on the transmission data storage register type and number, and if an error is found it returns an error.

Status: Boundary error (TermSTS = 0AH)

- If the T2N internal completion wait time limit (2 seconds) is exceeded for this request, the error listed below is returned.

Status: Send completion timeout (TermSTS = 06H)

In this case, first verify that the Ethernet Port is not in the down state by checking the station status, and then perform the required error handling (the send request etc.).

- The Ethernet Port uses a priority ordering in processing sockets 1 through 8. Therefore, systems constructed so that socket 1 is activated frequently may not be able to process socket 8, resulting in the following error response:

Status: Send completion timeout (TermSTS = 06H)

When constructing a system, take the socket utilization conditions into account when allocating sockets. Since the send/receive processing for each socket requires about 50 ms, applications that issue send or receive requests to a given socket should leave an interval of at least 50 ms times the number of sockets used between each request.

**(Interval between requests to the same socket)  $\geq$  (Number of sockets used  $\times$  50 ms)**

Similarly, the above intervals averaging 50 ms should be left between transmissions from remote ports to the local port.

**(Interval between remote port transmissions)  $\geq$  (Number of remote ports  $\times$  50 ms)**

- **When PC link transmission or the UDP socket interface is used, if an application transmits to a nonexistent remote port**, an transmission error (TermSTS = 0BH, detailed information = 0020H) occurs.

#### e. Broadcast transmission

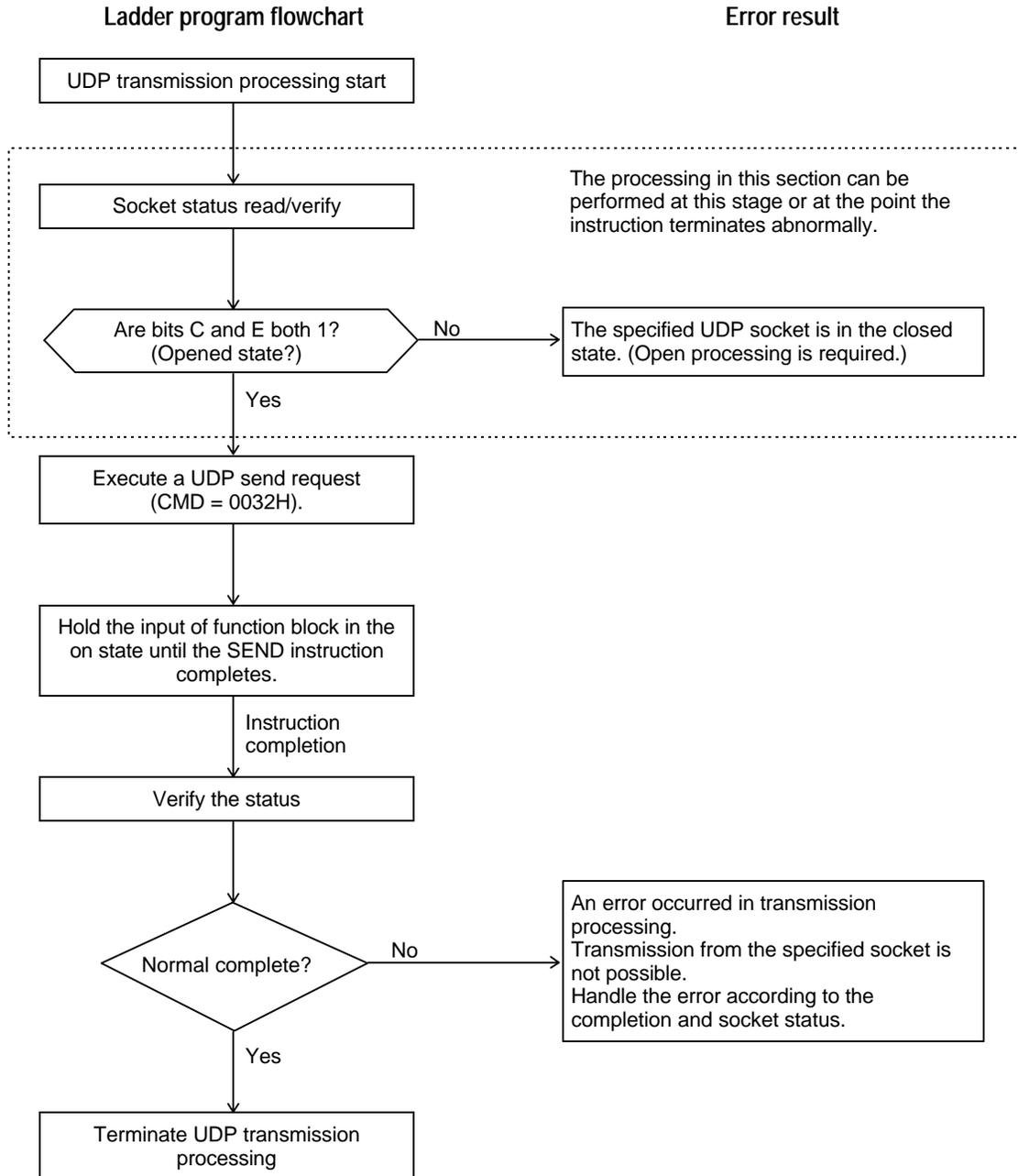
It is possible to broadcast to all ports on the network using UDP sockets.

Send side: Set the transmission destination address to "255.255.255.255", ("FF.FF.FF.FF" in hexadecimal).

The "0.0.0.0" used by some UNIX versions (in particular, 4.2 BSD) cannot be used.

Specify the UDP port number for the socket to which you want to issue a broadcast as the transmission destination object socket UDP port number.

f. UDP transmission processing example



3. UDP receive request/broadcast receive request (using the RECV instruction)

a. Function

If an opened UDP socket receives data, read the received data into registers on the local port.

If no data has been received, wait until data arrives (the wait time can be set) and then, after reception, read the received data into the local port registers.

Reception data size: 1 to 1000 words.

b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0033H	Command number
A+2	SportNO	Socket identifier (1 to 8)
A+3	DIPAddress	Transmission destination IP address
A+4		(The input format is the same as that for the parameter setup request.)
A+5	D_UDP_PORTNO	Transmission source object socket UDP port number
A+6	WordSize	Reception data length: 1 to 1000 words
A+7	DRID	Reception data storage register type code
A+8	DregNO	Reception data storage register number
A+9	TimeCNT	Reception wait timeout time

Reception data length: Set this parameter to a value greater than or equal to the largest size for transmission data that could be sent to the object socket in a single operation.

Reception data storage register type code: Type code for the register that stores the received data. (See Figure 5.1)

Reception data storage register number: Starting number of the registers that store the received data.

Reception wait timeout time: Specified in 0.1 second units. (1 to 65535)  
If zero is specified, the system is set to an unlimited (infinite) wait state.

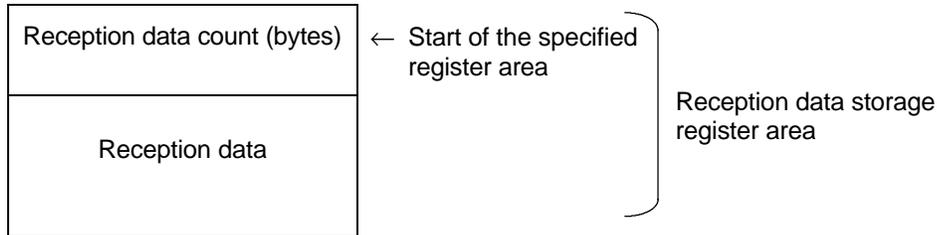
c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

SportNO: Socket identifier (1 to 8)  
TermSTS: See tables 4.3 and 4.4.  
Detail information: See table 4.5.

#### d. Methods for storing reception data

- Allocate a reception data storage register area with a size equal to the size of the reception data plus one word. The reception data count (the size of the transmission data for a single operation) and the transmission data for a single operation will be stored in this area as shown in the figure.



- The T2N checks for register area allocation of the reception data size plus one word and returns an error if the register area does not exist.

Status: Boundary error (TermSTS = 0AH)

In this case, data is not transferred to the register area, and the data is discarded. The data is not retained in the Ethernet Port either.

#### e. Important items

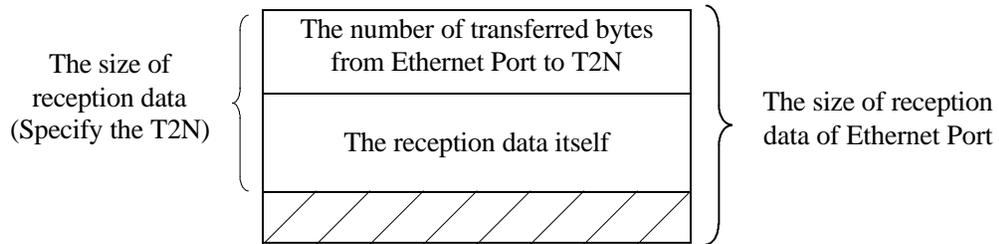
- An error is returned if a receive request is issued for a socket that is not yet opened.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Unopened (0081H)
- An error is returned if a value other than 1 to 8 is specified as the socket identifier.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Illegal socket identifier (0082H)
- It is not possible to receive data sent from the local port, even if the local port IP address is specified at the transmission source IP address.
- Values in the range 1024 to 65535 can be specified as the object socket UDP port number for the transmission source. An error will be returned if a value outside that range is specified.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Port number error (0071H)
- An error will be returned if a value of 0 words or 1001 or more words is specified as the reception data size.
  - Status: Transmission word count error (TermSTS = 09H)
- An error will be returned if the reception timeout time is exceeded.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Timeout(0020H)
- Allocate a reception data storage register area with a size equal to the size of the transmission data plus one word.** The size of the transmission data for the single operation and the size of reception data will be stored in this area.

- If the transmission data size for a single operation is larger than the reception data size, the section of the transmitted data that exceeds the specified reception size is discarded, resulting in the following error response.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Insufficient data size(00C1H)

If this error is occurred, a reception data storage register of the T2N enters the data following below. The data of slash bar is not received to T2N and discarded.



- The Ethernet Port uses a priority ordering in processing sockets 1 through 8. Therefore, systems constructed so that socket 1 is activated frequently may not be able to process socket 8, resulting in the following error response:

Status: Send completion timeout (TermSTS = 06H)

When constructing a system, take the socket utilization conditions into account when allocating sockets. Since the send/receive processing for each socket requires about 50 ms, applications that issue send or receive requests to a given socket should leave an interval of at least 50 ms times the number of sockets used between each request.

**(Interval between requests to the same socket) ≥ (Number of sockets used × 50 ms)**

Similarly, the above intervals averaging 50 ms should be left between transmissions from remote ports to the local port.

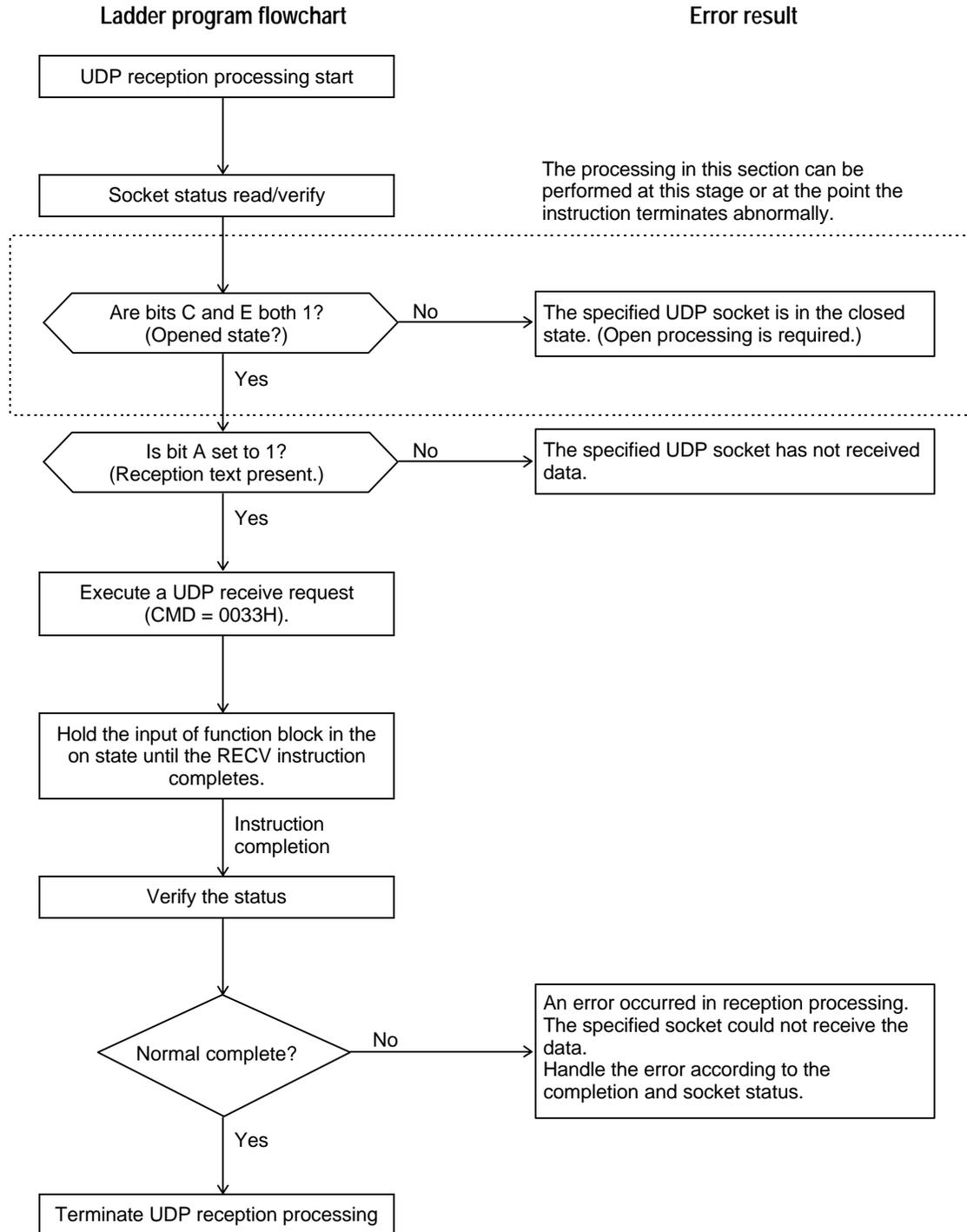
**(Interval between remote port transmissions) ≥ (Number of remote ports × 50 ms)**

**f. Broadcast reception**

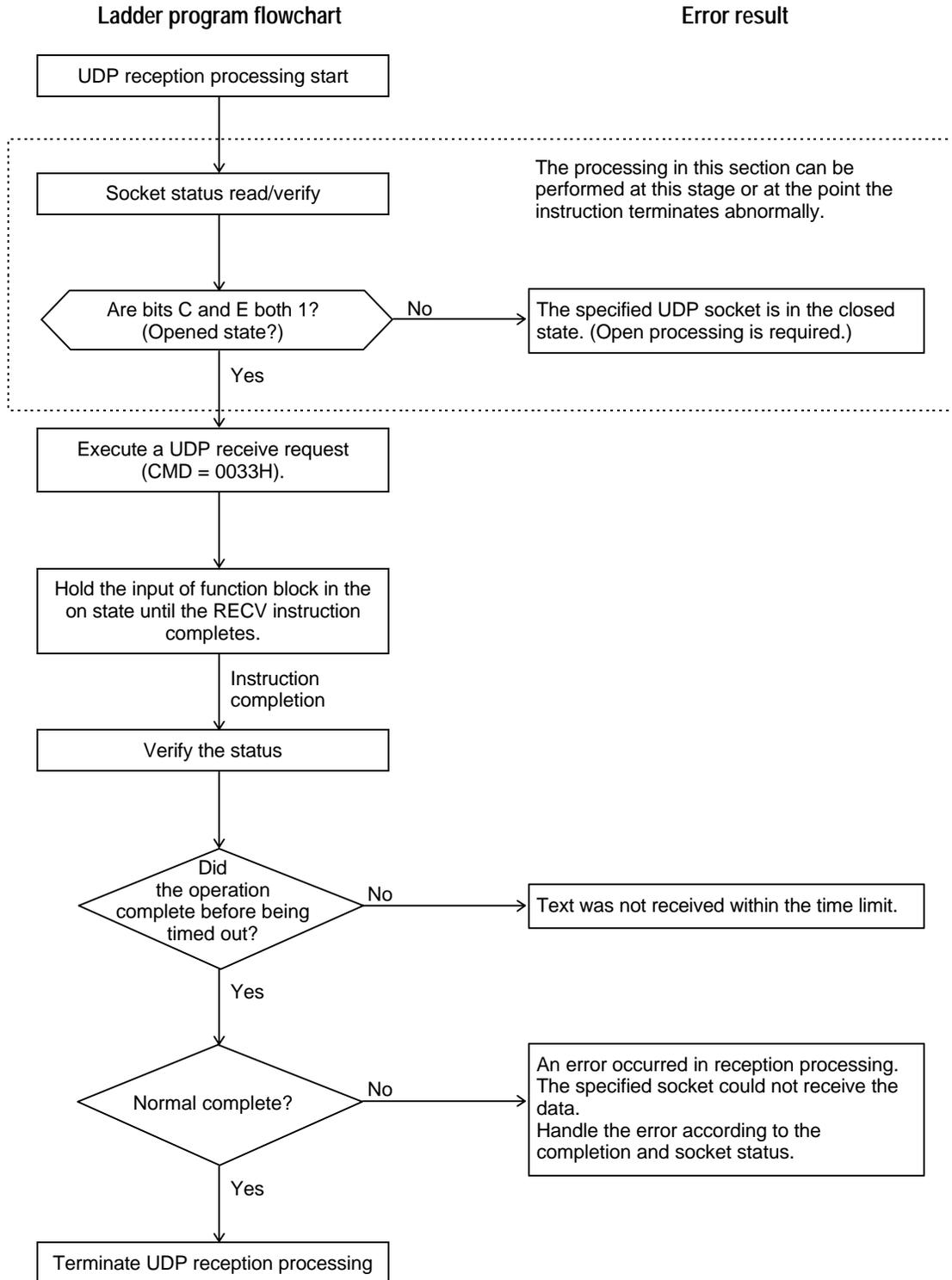
It is possible to broadcast to all ports on the network using UDP sockets.

Receive side: Specify the transmission source IP address to be "0.0.0.0".  
Specify the UDP port number for the broadcast transmitting socket as the transmission source UDP port number.

g. UDP reception processing example (Socket status monitoring type)



h. UDP reception processing example (Receive request issued first type)



**4. UDP close request (using the SEND instruction)**

**a. Function**

Close the open UDP socket.

Release the (unlimited wait state) receive request and terminate the UDP socket.

**b. Transfer parameters**

A	3100H	Module designation
A+1	CMD = 0034H	Command number
A+2	SportNO	Socket identifier (1 to 8)

**c. Status (See figure 4.3 for details on bits C, D, E and F.)**

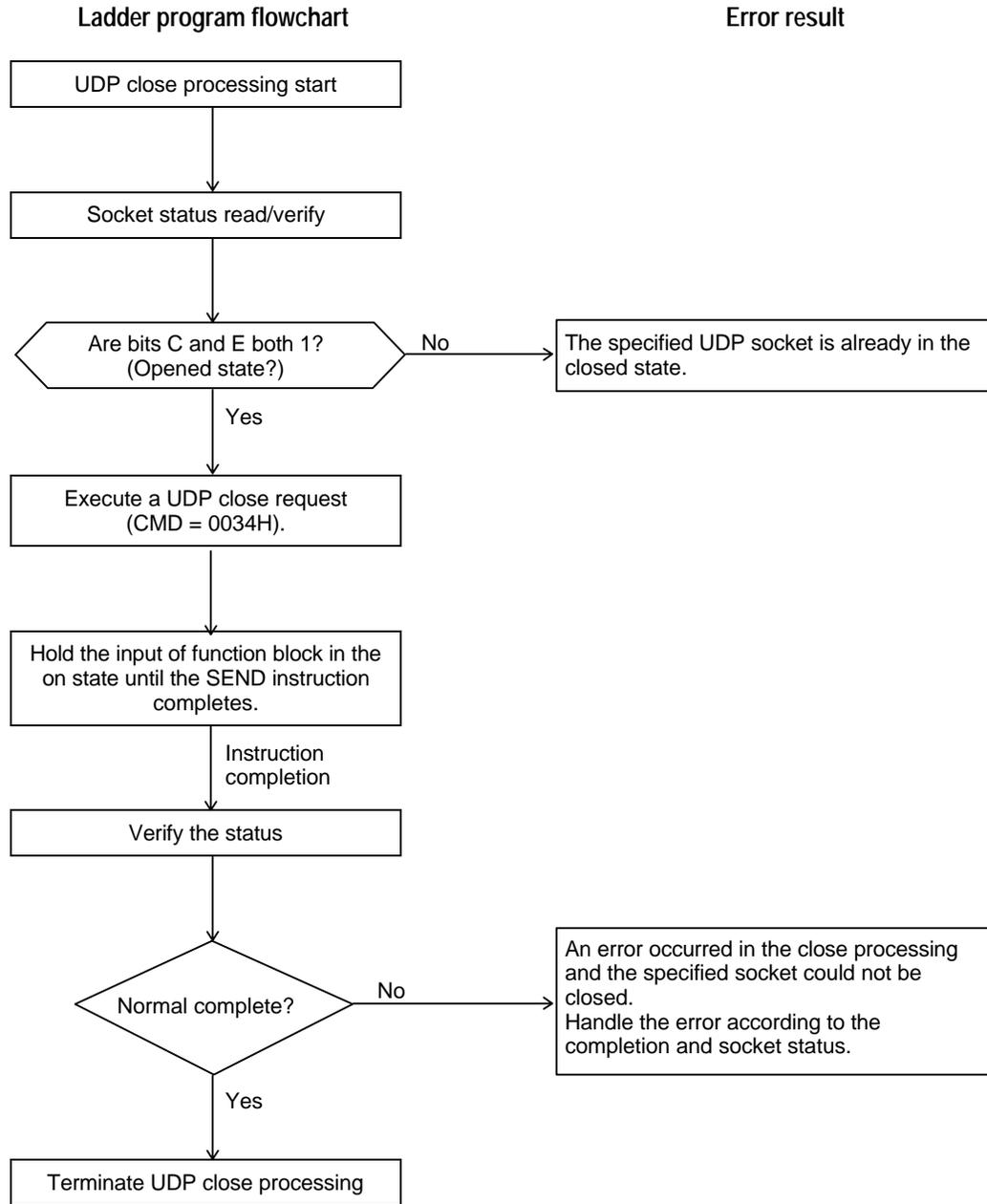
	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

SportNO: Socket identifier (1 to 8)  
 TermSTS: See tables 4.3 and 4.4.  
 Detail information: See table 4.5.

**d. Important items**

- UDP close requests are processed with the highest priority and other executing requests are discarded, even if the object socket is executing a UDP open, send, or receive request. Other requests during the execution of this request will be discarded.
- The next open request for the object socket can be executed as soon as the close request completes normally.
- An error is returned if a close request is issued for a socket that is not yet opened.  
 Status: Transmission error (TermSTS = 0BH)  
 Detailed information: Unopened (0081H)
- An error is returned if a value other than 1 to 8 is specified as the socket identifier.  
 Status: Transmission error (TermSTS = 0BH)  
 Detailed information: Illegal socket identifier (0082H)
- A UDP receive request will result in an error being returned if an unlimited wait state UDP receive request is terminated with this request.  
 Status: Transmission completion timeout (TermSTS = 06H)  
**Supplement:**  
 The T2N sees its requests (module control and socket interface transmission) to the Ethernet Port as being "transmission" requests. As a result, when a request is forcibly terminated and a response is not received from the Ethernet Port, the result is a send request timeout even if the instruction was a RECV instruction.
- If the T2N internal completion wait time limit (2 seconds) is exceeded for this request, the error listed below is returned.  
 Status: Send completion timeout (TermSTS = 06H)  
 In this case, first verify that the Ethernet Port is not in the down state by checking the station status, and then verify the open/closed state of the socket in the socket status.

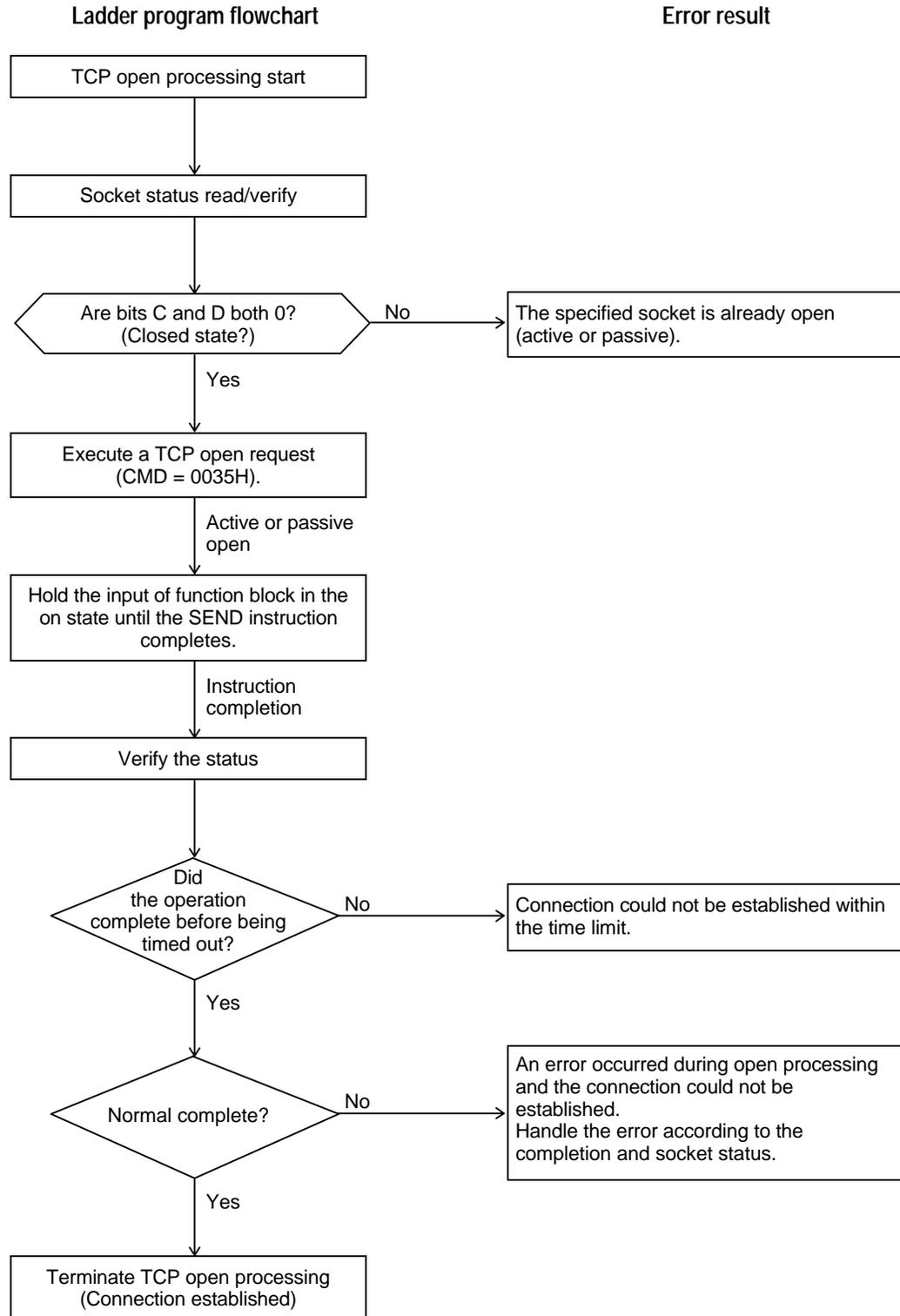
e. UDP close processing example





- Not only must the specified socket TCP port number on the local port not overlap with any TCP port number already in use in a local port socket, but it also must not overlap with any **UDP port numbers or UDP port numbers used for message transmission**. An error will occur if the port number overlaps.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Already open (0080H)
- A connection to the local port itself cannot be established, even if the local port IP address is specified as the remote port IP address.
- Values in the range 1024 to 65535 can be specified as the object socket TCP port number for the remote port. An error will be returned if a value outside that range is specified.
  - Status: Transmission error (TermSTS = 0BH)
  - Detailed information: Port number error (0071H)
- An error will be returned if the open timeout time is exceeded.
  - ①The passive open timeout
    - Status: Transmission error (TermSTS = 0BH)
    - Detailed information: Timeout(0020H)
  - ②The active open timeout
    - Status: Transmission complete timeout (TermSTS = 06H)
- While the open wait timeout limit can be set to any value in the range 0.1 to 6553.5 seconds or unlimited (infinite), **we recommend that the timeout limit be set in the range 0.1 to 32 seconds for active opens**. This is because the Ethernet Port processing continuation time for active open processing is 32 seconds. Even if the timeout limit is set to a value over 32 seconds, once 32 seconds has elapsed, the open processing will never actually be performed. The system will be in an idle state until the specified time has elapsed.
- When a passive open operation is executed, the specified socket goes to the active open wait state. If the DIPAddress and DTCP\_PORTNO values are set to 0, the socket is set to the state in which a particular remote port is not specified. (In this state the socket can handle an active open from any other (i.e., any remote) port on the network.)
- A connection is established by an active open from another socket applied to a socket in the passive open state. A connection cannot be established by a passive open from another socket applied to a socket in the passive open state. Similarly, a connection cannot be established by an active open from another socket applied to a socket in the active open state.
- For a socket in the passive open state, it is not possible to open multiple connections from other ports by issuing multiple active opens.

e. TCP open processing example



**6. TCP send request (using the SEND instruction)**

**a. Function**

This request sends local port register data using an open TCP socket.

Amount of data sent: 1 to 1000 words

**b. Transfer parameters**

A	3100H	Module designation
A+1	CMD = 0037H	Command number
A+2	SportNO	Socket identifier (1 to 8)
A+3	WordSize	Transmission data length: 1 to 1000 words
A+4	SRID	Transmission data storage register type code
A+5	StreqNOD	Transmission data storage register number

Transmission data storage register type code: Type code for the register that holds the transmission data. (See Figure 5.1)

Transmission data storage register number: Starting number of the registers that hold the transmission data.

**c. Status (See figure 4.3 for details on bits C, D, E and F.)**

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

SportNO: Socket identifier (1 to 8)  
 TermSTS: See tables 4.3 and 4.4.  
 Detail information: See table 4.5.

**d. Important items**

- Send requests complete at the point they are stored in the Ethernet Port. This is to allow the send to be performed later even if that send request cannot be executed immediately due to network congestion or other problems. The Ethernet Port can store up to three requests per socket, and the Ethernet Port returns an error for send requests that it cannot store.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Timeout(0020H)

- The T/C register flag data is not transmitted if the T/C registers are not specified as storage registers for transmitted data.
- An error is returned if a send request is issued for a socket that is not yet opened.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Unopened (0081H)

- An error is returned if a value other than 1 to 8 is specified as the socket identifier.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Illegal socket identifier (0082H)

- An error will be returned if a value of 0 words or 1001 or more words is specified as the transmission data size.

Status: Transmission word count error (TermSTS = 09H)

- The T2N performs an area check for the register area based on the transmission data storage register type and number, and if an error is found it returns an error.

Status: Boundary error (TermSTS = 0AH)

- If a close request from the remote port TCP socket has arrived while a connection is established, and the user program issues a send request, the Ethernet Port returns the following error.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Unopened (0081H)

Verify the socket status (see section 6.3) RCLOSE and CONN bits, and close the socket.

- With a connection established, if a no response state from the remote port continues when data is sent and the Ethernet Port internal resend processing times out (the timeout time is about 1 minute), the NOACK bit in the socket status will be set to 1. User programs should close such sockets. Errors will be returned for send/receive requests issued in this state.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Protocol error (0091H)

However, note that the Ethernet Port will return normal complete responses for TCP send requests issued prior to the timeout, since the send requests from the T2N are stored internally by the Ethernet Port.

When the number of send requests that can be stored is exceeded, errors will be returned for that and all following send requests.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Timeout (0020H)

- One of the following problems may have occurred when there is no response from the remote port:
  - The remote port may have gone down.
  - Power may have been lost.
  - The remote ports and/or local port may have become disconnected from the network.
- If the T2N internal completion wait time limit (2 seconds) is exceeded for this request, the error listed below is returned.

Status: Send completion timeout (TermSTS = 06H)

In this case, first verify that the Ethernet Port is not in the down state by checking the station status, and then perform the required error handling.

- The Ethernet Port uses a priority ordering in processing sockets 1 through 8. Therefore, systems constructed so that socket 1 is activated frequently may not be able to process socket 8, resulting in the following error response:

Status: Send completion timeout (TermSTS = 06H)

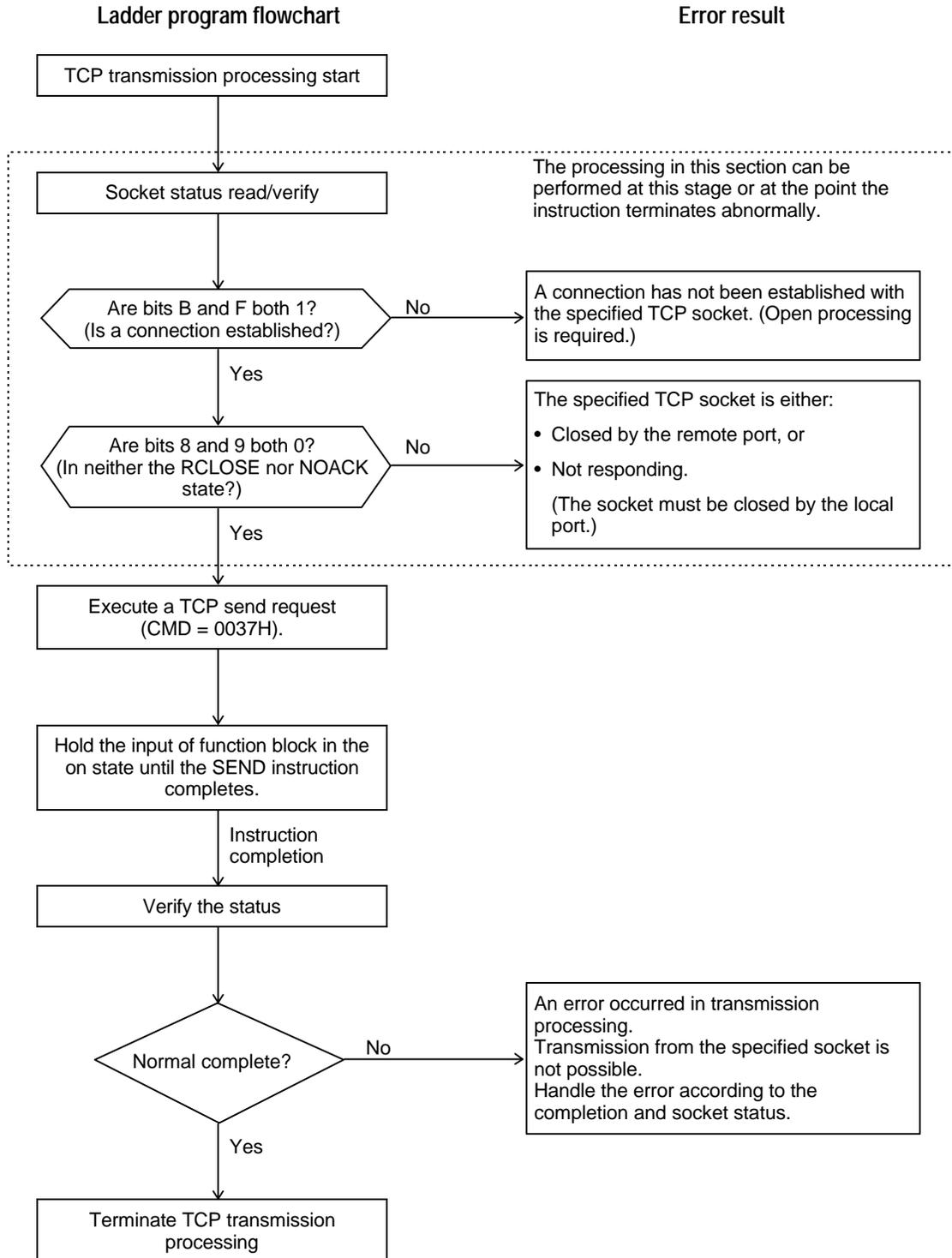
When constructing a system, take the socket utilization conditions into account when allocating sockets. Since the send/receive processing for each socket requires about 50 ms, applications that issue send or receive requests to a given socket should leave an interval of at least 50 ms times the number of sockets used between each request.

**(Interval between requests to the same socket)  $\geq$  (Number of sockets used  $\times$  50 ms)**

Similarly, the above intervals averaging 50 ms should be left between transmissions from remote ports to the local port.

**(Interval between remote port transmissions)  $\geq$  (Number of remote ports  $\times$  50 ms)**

e. TCP transmission processing example



7. TCP receive request (using the RECV instruction)

a. Function

If an opened TCP socket receives data, read the received data into registers on the local T2N.

If no data has been received, wait until data arrives (the wait time can be set) and then, after reception, read the received data into the local T2N registers.

Reception data size: 1000 words (fixed)

b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0038H	Command number
A+2	SportNO	Socket identifier (1 to 8)
A+3	WordSize	Reception data length: 1000 words
A+4	DRID	Reception data storage register type code
A+5	DregNO	Reception data storage register number
A+6	TimeCNT	Reception wait timeout time

Reception data length: This parameter must be set to 1000 words.

Reception data storage register type code: Type code for the register that holds the received data. (See Figure 5.1)

Reception data storage register number: Starting number of the registers that hold the received data.

Reception wait timeout time: specified in 0.1 second units. (1 to 65535)  
If zero is specified, the system is set to an unlimited (infinite) wait state.

c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

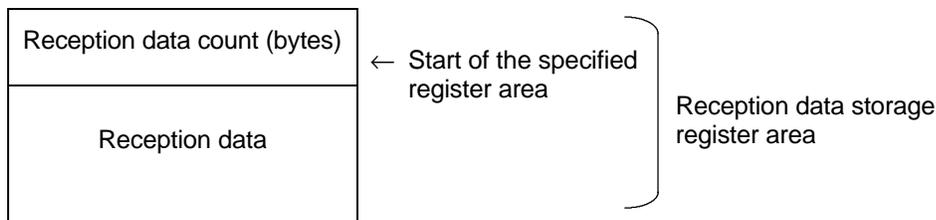
SportNO: Socket identifier (1 to 8)

TermSTS: See tables 4.3 and 4.4.

Detail information: See table 4.5.

d. Methods for storing reception data

- Applications must allocate a reception data storage area of 1001 words. The reception data count and the reception data (the data received by the object socket at the point the receive request was issued) will be stored in this area as shown below.



- The T2N checks for register area allocation of the reception data size plus one word and returns an error if the register area does not exist.

Status: Boundary error (TermSTS = 0AH)

In this case, data is not transferred to the register area, and the data is discarded. The data is not retained in the Ethernet Port either.

e. **Important items**

- An error is returned if a receive request is issued for a socket that is not yet opened.  
Status: Transmission error (TermSTS = 0BH)  
Detailed information: Unopened (0081H)
- An error is returned if a value other than 1 to 8 is specified as the socket identifier.  
Status: Transmission error (TermSTS = 0BH)  
Detailed information: Illegal socket identifier (0082H)
- An error will be returned if a value of 0 words or 1001 or more words is specified as the reception data size.  
Status: Transmission word count error (TermSTS = 09H)
- An error will be returned if the reception timeout time is exceeded.  
Status: Transmission error (TermSTS = 0BH)  
Detailed information: Timeout(0020H)
- If a close request from the remote port TCP socket has arrived while a connection is established, and the user program issues a receive request, the Ethernet Port returns the following error. Issuing a receive request in the receive wait state will also result in the following error being returned.  
Status: Transmission error (TermSTS = 0BH)  
Detailed information: Unopened (0081H)  
Verify the socket status (see section 6.3) RCLOSE and CONN bits, and close the socket.
- **Allocate a reception data storage register area per a socket with a size equal to the size of the transmission data plus one word.**
- If the reception data size is larger than the reception data size of the receiving Ethernet Port, the size of transmission data for the single operation and the size of reception data at that time is transmitted to the T2N's reception data storage register area.
- Since a TCP receive request merely places the socket in the receive wait state, it cannot detect remote ports that are not responding. Use one of the following methods to prevent this state.
  - Monitor for a no-response state by sending data periodically over a separate (different) TCP connection.
  - Execute a remote port presence verification request for the remote port periodically. (See section 7.3.)
  - Monitor the interval between data receptions when data is being received periodically. (Receive request issued first type: specify a limited timeout limit.) (Socket status monitoring type: monitor the interval between points when the received text present bit is set to the on state.)

- One of the following problems may have occurred when there is no response from the remote port:
  - The remote port may have gone down.
  - Power may have been lost.
  - The remote ports and/or local port may have become disconnected from the network.
- **In the Ethernet Port TCP socket interface, when reception data in the Ethernet Port is read out by a receive request from the T2N, the Ethernet Port sends an ACK (acknowledge response) to the port that sent the data.** Since an ACK will not be sent to the sending port if the received data remains in the Ethernet Port if the T2N does not read out that received data, the sending port will resend the data. If this state continues for a predetermined period, the sending port may will decide that the TCP connection is abnormal and close the connection. To prevent this from occurring, the T2N software should monitor the RCV (receive data present) bit and issue receive requests when necessary.
- With TCP receive requests, the amount of data transferred to the reception data storage register area will differ with both the timing with which the T2N issued the receive request and the timing with which the data arrives at the Ethernet Port. To handle single units of transmission data on the receiving side as single units as well, the user software on the receiving side must recognize the end of the transferred data (either by using counts in the transmitted data or by including an end marker in the transmitted data) and iterate receive requests until all the transmitted data has been received.
- User programs must save all incoming data so that the register area used to transfer data to the T2N is not overwritten when using iterated receive requests.
- The Ethernet Port uses a priority ordering in processing sockets 1 through 8. Therefore, systems constructed so that socket 1 is activated frequently may not be able to process socket 8, resulting in the following error response:

Status: Send completion timeout (TermSTS = 06H)

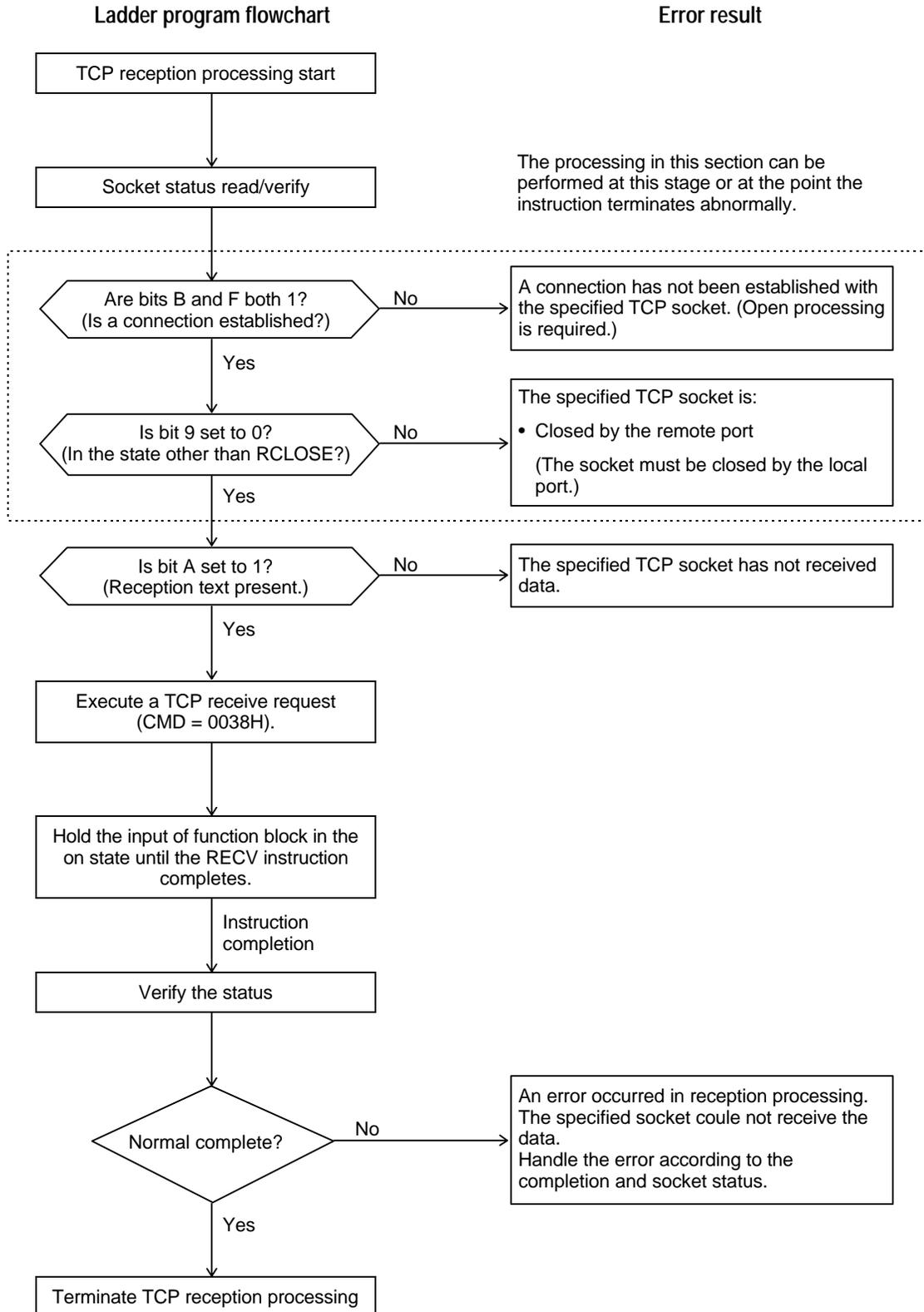
When constructing a system, take the socket utilization conditions into account when allocating sockets. Since the send/receive processing for each socket requires about 50 ms, applications that issue send or receive requests to a given socket should leave an interval of at least 50 ms times the number of sockets used between each request.

**(Interval between requests to the same socket)  $\geq$  (Number of sockets used  $\times$  50 ms)**

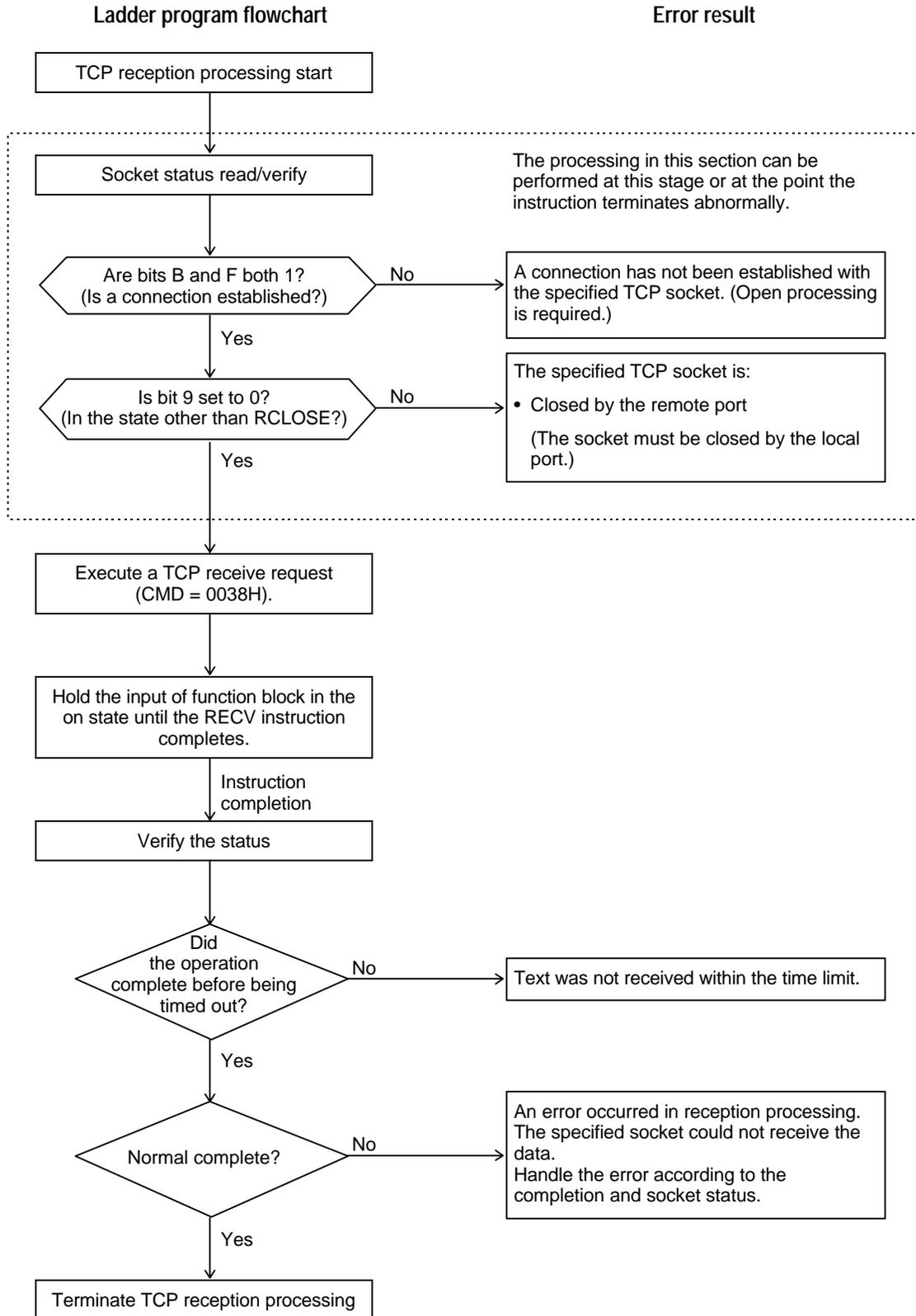
Similarly, the above intervals averaging 50 ms should be left between transmissions from remote ports to the local port.

**(Interval between remote port transmissions)  $\geq$  (Number of remote ports  $\times$  50 ms)**

f. TCP reception processing example (Socket status monitoring type)



g. TCP reception processing example (Receive request issued first type)



8. TCP close request (using the SEND instruction)

a. Function

Close the open TCP socket.

Release the (unlimited wait state) receive request and terminate the TCP socket.

b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 0039H	Command number
A+2	SportNO	Socket identifier (1 to 8)

c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					SportNO				TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

SportNO: Socket identifier (1 to 8)  
 TermSTS: See tables 4.3 and 4.4.  
 Detail information: See table 4.5.

d. Important items

- TCP close requests are processed with the highest priority and other executing requests are discarded, even if the object socket is executing a TCP open, send, or receive request. Other requests during the execution of this request will be discarded.

- An error is returned if a close request is issued for a socket that is not yet opened.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Unopened (0081H)

- An error is returned if a value other than 1 to 8 is specified as the socket identifier.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Illegal socket identifier (0082H)

- **A TCP receive requests will result in an error being returned** if an unlimited wait state TCP open request/receive request is terminated with this request.

Status: Transmission completion timeout (TermSTS = 06H)

Supplement:

The T2N sees its requests (module control and socket interface transmission) to the Ethernet Port as being "transmission" requests. As a result, when a request is forcibly terminated and a response is not received from the Ethernet Port, the result is a send request timeout even if the instruction was a RECV instruction.

- If a close request is comes from the remote port TCP socket while a connection is established, the local port must also close the socket. As a technique for checking for the reception of close requests from the remote TCP socket, user programs should monitor the RCLOSE and CONN bits in the socket status. (See section 6.3.)

- If a close request is sent from the Ethernet Port TCP socket while a connection is established, The T2N cannot reopen request using the same port number of the closed port number for 10 seconds or more since the Ethernet Port is set the close request.

**Supplement:**

On engineering workstation and personal computer systems, it may not be possible to reopen a TCP socket on the port that performed the close processing for a fixed period.

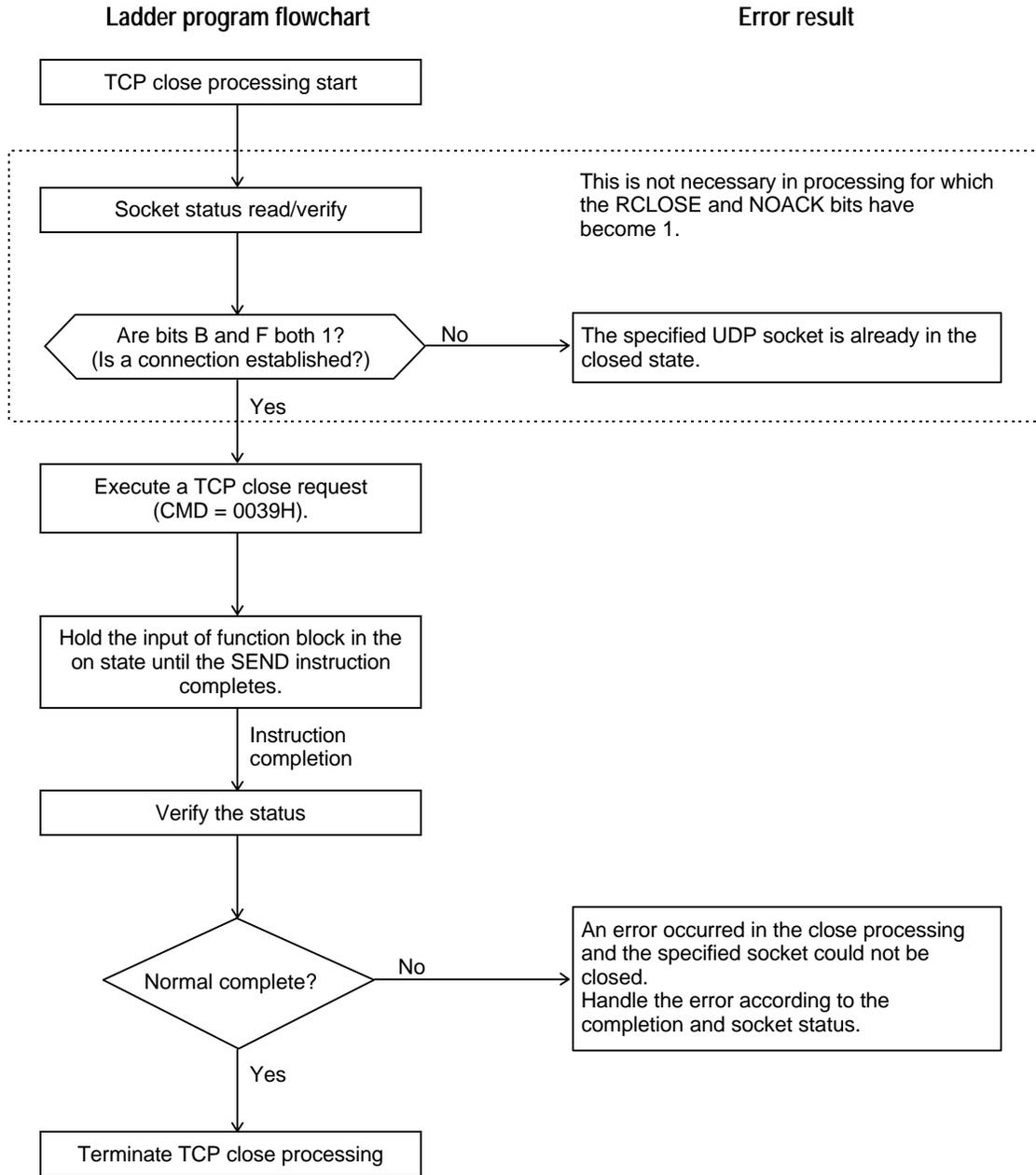
- The Ethernet Port which received a close request from the T2N replies to be received the T2N. Therefore, a close request of the T2N completes the no waiting time. The Ethernet Port send a close request to a remote port and is waiting for a close request from a remote port.
- If the T2N executes a open request in the same port number while the Ethernet Port is waiting for a close request from a remote port, a open request will result in an error being returned.

Status: Transmission error (TermSTS = 0BH)

Detailed information: Opened (0080H) or

LAN controller driver error (0090H/0092H)

e. TCP close processing example



## 6.5 Sample Programs

This section presents sample programs using UDP and TCP sockets.

### 1. UDP sockets

This is a sample program that issues the requests used with UDP sockets. It assumes that the parameter set up and operating mode control requests (run mode/socket interface transmission enable) have already completed.

#### a. UDP open request

This program issues a UDP open request for socket number 2 on the Ethernet Port.

This ladder program is executed by turning on the A contact on the R0620.

Parameters: 12544: 3100H, 00049: 0031H

UDP port number: 4000

```

| /* UDP open request: executed when R0620 is set on. */
|
|R0620
1|-| |--[12544 MOV RW000][00049 MOV RW001]-----|
|      /* Module designation, CMD number setup */
|      |
|      |[00002 MOV RW002][04000 MOV RW003]-----|
|      /* Socket identifier, UDP port number setup */
|      |
|      |[RW000 SEND RW010][RST R0620]-----|
|      /* Turn R0620 off after request setup and completion. */
|

```













## d. TCP send request

This program issues a TCP send request for socket number 8 on the Ethernet Port.

This ladder program is executed by turning on the A contact on the R0632.

Parameters: 12544: 3100H, 00055: 0037H

Transmission data word count: 1000

Transmission data storage registers: D1000 and above

```

| /* TCP send request: executed when R0632 is set on. */           |
|                                                                     |
|R0632                                                                |
1|-| |-[12544 MOV RW000][00055 MOV RW001]-----|
|   | /* Module designation, CMD number setup */                   |
|   |                                                                     |
|   |[00008 MOV RW002]-----|
|   | /* Socket identifier setup */                                   |
|   |                                                                     |
|   |[01000 MOV RW003]-----|
|   | /* Transmission data word count setup */                       |
|   |                                                                     |
|   |[00004 MOV RW004][00000 MOV RW005]-----|
|   | /* Transmission data storage register type, storage register |
|number setup */                                                    |
|   |                                                                     |
|   |[RW000 SEND RW010][RST R0632]-----|
|   | /* Turn R0632 off after request setup and completion. */     |
|                                                                     |

```

## e. TCP receive request

This program issues a TCP receive request for socket number 8 on the Ethernet Port.

This ladder program is executed by turning on the A contact on the R0631.

Parameters: 12544: 3100H, 00056: 0038H

Reception data word count: RW103

Reception data storage registers: D4000 and above

Reception data transfer destination registers: D1001 and above

Reception timeout time: 1800 seconds

```

| /* TCP receive request: executed when R0631 is set on. */      |
|                                                                    |
|R0631                                                            |
1|-| |--[12544 MOV RW000][00056 MOV RW001]-----|
|      /* Module designation, CMD number setup      */          |
|                                                                    |
|+[00008 MOV RW002][RW103 MOV RW003]-----|
|      /* Socket identifier, reception data word count setup */  |
|                                                                    |
|+[00004 MOV RW004][04000 MOV RW005][18000 MOV RW006]-----|
|      /* Reception data storage register type, register number, |
|timeout time setup */                                          |
|                                                                    |
|+|^|--[00004 MOV RW021][04001 MOV RW022]-----|
|      /* Data transfer source register type, number setup */    |
|                                                                    |
|+[00001 MOV RW025]-----|
|      /* Data transfer count setup      */                      |
|                                                                    |
|+[00004 MOV RW023][01001 MOV RW024]-----|
|      /* Data transfer destination register type, number setup */|
|                                                                    |
|+[RW103 + 01000 -> RW008]-----|
|      /* Transmission event count setup */                      |
|                                                                    |
|+[SET R0635][RST R0631]-----|
|                                                                    |

```

```

| /* TCP fragmented data reception processing */
|
|R0635 R063D
2|-| |+-|/|--[RW000 RECV RW010][SET R063D]-----|
| | /* Request setup */
| |R063D
| +-| |+-|^|--[RTR1D4000]-----|
| | /* Reception data count conversion (bytes to
|words) */
| |R063E
| +-|/|+[RW021 XFER RW025 -> RW023]-----|
| | /* Reception data transfer (D4001 to D1001) */
| |
| | [+ +1 RW009][ +1 RW022][ +1 RW024]-----|
| | /* Increment address and pointer. */
| |
| |R063E
| +[RW009 >= D4000]----- ( )--|
| |
| | /* Reception data size check */
| |R063E
| +-| |+[RW008 > RW024][ RST R063D]-----+[ 00000 MOV RW009]|
| | |
| | | [+ 04001 MOV RW022]|
| | | /* Processing for data less than the reception
|data size */
| |
| | +[RW008 <= RW024]-----+[ RST R063D]-----| |
| | |
| | | [+ 00000 MOV RW009][RST R0635]|
| | /* Processing for data greater than the
|reception data size */
|

```

## 7. RAS Information

This chapter describes the following RAS functions provided by the Ethernet Port.



### CAUTION

- Chapter 7 presents information related to using the functions provided by the Ethernet Port from a T2N, including the instruction (request) format, important items that require attention, and sample programs.  
Make a point of understanding the content of chapter 4 thoroughly before writing programs that use the Ethernet Port. The sample programs present basic examples of Ethernet Port usage, and should be reviewed carefully before use in an actual system.

- Ethernet Port status information
  - Station status (T2N special registers)
  - Down information (T2N interface buffer)
- Test functions from user programs
  - Remote port verification request (corresponds to the UNIX ping command)
  - Inter-port loopback test (between T2N and T2N/T3H units)
- Time setting function
- Information provided by RAS information readout
  - LAN controller (network circuit) information
  - Protocol state
  - State of the T2N-Ethernet Port interface
  - Event trace

## 7.1 T2N Special Relays and Registers

The Ethernet Port status information is reflected in the T2N special relays and special registers.

### a. Special relays (the renewing cycle of 100 ms)

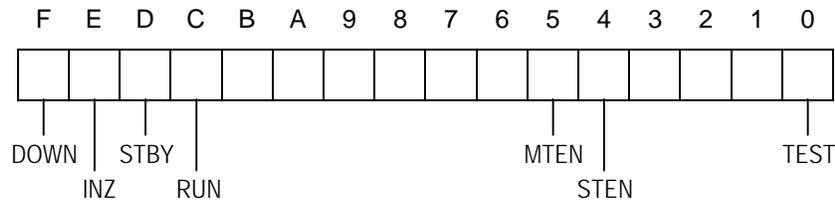
Special relay	Name	Function
S000B	Ethernet Port error (warning)	1: The installed Ethernet Port unit is down. (The T2N continues to operate.)  0: Reset the system from a user program after resolving the error in the down Ethernet Port.

### b. Special registers

T2N read out the internal station status of Ethernet Port and copy the SW register (the renewing cycle of 100 ms).

Special registers	Name
SW063	CH1 station status

### c. Station status format



- Bit F: DOWN (down)..... 1: Down mode, 0: Other than down mode
- Bit E: INZ (initialization)..... 1: Initialization in progress  
0: Initialization completed
- Bit D: STBY (standby)..... 1: Standby mode  
0: Other than standby mode
- Bit C: RUN (run)..... 1: Run mode, 0: Other than run mode
- Bit 5: MTEN (Message transmission)..... 1: Enabled, 0: Prohibited
- Bit 4: STEN (socket interface transmission)..... 1: Enabled, 0: Prohibited
- Bit 0: TEST (test)..... 1: Test function execution in progress

## d. Operation mode and station status of Ethernet Port

Each bit of station status has the following mean. **If you want to judge the mode (initialize mode/ run mode/ standby mode), look at the value of the register.**

The mode of Ethernet Port	The station status
During initialize (Power on / Reset request)	4000H
Standby mode (initialize processing normally complete)	2000H
Run mode : Allows the message transmission	1020H
Run mode : Allows the socket interface	1010H
Run mode : Allows the message transmission and socket interface	1030H

## 7.2 Down Information

When an Ethernet Port goes to down mode, the factor that caused that transition is stored in the T2N interface buffer as one word of data. This data can be read by the T2N with the READ instruction.

Table 7.1 lists the down information error codes and the factors that caused the transition to down mode.

**Table 7.1 Down Information**

Error code (H)	Interpretation
0010	Watchdog timeout
0020	Memory bus stall
0030	A TRAP occurred
0040	Jabbering timeout
0050	LAN controller check error
0160	ROM BCC check data match error
0260	System RAM area check error
0360	T2N interface buffer check error
0500, 0501	Semaphore acquisition error
0502, 0503	Semaphore acquisition error
0504	MBX receive error
0505	MBX send error
0506	TCP resend queue full
0507	Acquired buffer in use
0508	Error in acquiring initial buffer

### Important items

This is a read-only area. Do not write to this area with the WRITE instruction. Subsequent reads will not return correct down information.

**Sample:** Down information readout

```
|R0000
1|-| |--[04081 MOV RW010][00001 MOV RW011][H00FE READ RW010 -> D1000]--|
|
```

#### READ instruction description

H00FE: Module designation ... Ethernet Port must specify H00FE.

RW010: Specifies the 04081 (0FF1H) down information area.  
0FF1H is the down information area for the Ethernet Port.

RW011: Specifies the number of words (00001) to be read out.

D1000: Specifies the register in which to store the read out down information.  
In this case, the down information that is read out is stored in D1000.

The down information is stored in D1000 by turning R0000 on.

### 7.3 Remote Port Verification Request

This request verifies the existence of the specified remote port, and corresponds to the UNIX ping command. This request can be issued when the Ethernet Port operating mode is either run or standby mode. The Ethernet Port will respond to this request if its operating mode is either run or standby mode.

#### Remote port verification request (using the SEND instruction)

**a. Function**

This is a request that verifies the existence of the specified remote port.

**b. Transfer parameters**

A	3100H	Module designation
A+1	CMD = 0014H	CMD number
A+2	D- IP address	IP address of the remote station whose existence is to be verified. (The input format is the same as that for the parameter setup request.)
A+3		

**c. Status (See figure 4.3 for details on bits C, D, E and F.)**

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

**d. Important items**

- This request is possible in run and standby modes. (The IP address of the local port must be set up.)
- The Ethernet Port returns a response to this request in run and standby modes. (The IP address of the local port must be set up.)
- If the remote port exists:                      Normal return
- If the remote port does not exist (within 4 seconds):
  - Status:    Transmission error (TermSTS = 0BH)
  - Detailed information:                      Timeout occurred (0020H)

e. Sample program

This sample ladder program issues a remote port verification request to the Ethernet Port.

This program is executed by turning on the R0504 A point.

Parameters: 12544:3100H, 00020:0014H, 341995909;14627185H

↓  
85H. 71H. 62H. 14H  
↓  
133. 113. 98. 20

```

|R0504                                     R0604 |
1|-| |----- ( )--|
|          /* Remote port verification request */
|R0604                                     |
2|-| |--[12544 MOV RW000][00020 MOV RW001]-----|
|          /* Designates the module and sets the CMD number. */
|          |
|          |[341995909 DMOV RW003•RW002]-----|
|          /* Sets remote port IP address */
|          |
|          |[RW000 SEND RW010][RST R0504]-----|
|          /* Sets the request. */
|          |

```

## 7.4 Remote Port Loopback (for an Ethernet Port)

This request sends test data to the specified remote Ethernet Port (T2N/T3H) and verifies that transmission is being performed correctly by receiving that data looped back from the remote port. A user program must compare the transmitted and received data for equivalence.

### Remote port loopback request (using the SEND instruction)

#### a. Function

This request sends test data to the specified remote Ethernet Port (T2N/T3H) and verifies that transmission is being performed correctly by receiving that data looped back from the remote port. The T2N OS creates the test data.

#### b. Transfer parameters

A	3100H	Module designation
A+1	CMD = 000FH	CMD number
A+2	WordSize	Loopback data length: 1 to 485 words
A+3	SRID	Send data storage register type code
A+4	SRegNO	Send data storage register number
A+5	DRID	Loopback data storage register type code
A+6	DRegNO	Loopback data storage register number
A+7	Timecnt	Timer count
A+8	D-IPAddress	Loopback destination IP address
A+9		(The input procedure is the same as that for the parameter setup request.)
A+10	D-UDP PortNO	Loopback destination message transmission UDP port number

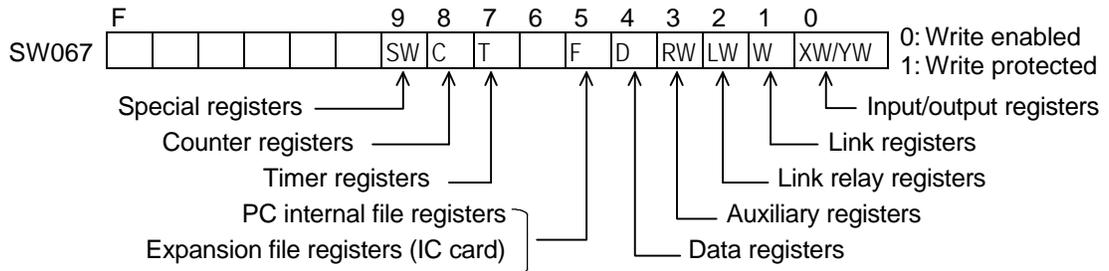
#### c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

#### d. Important items

- This request cannot be executed unless message transmission is enabled with an operating mode control request.
- In response to a request from a user program, the T2N OS performs the processing using the Ethernet Port PC link protocol.
- User programs on the loopback destination do not take part in this operation.
- A user program must compare the transmitted and loopback data for equivalence.
- A transfer word count error (TermSTS = 09H) occurs if the loopback data length exceeds the range 1 to 485 words (or 1 to 323 words for the T and C registers).
- A register specification error (TermSTS = 01H) occurs if a value other than a stipulated value is specified for the register type codes (SRID/DRID).
- A register specification error (TermSTS = 01H) occurs if 0007H (the T register) is specified for the send data storage register type code and a register other than the T register is specified for the loopback data storage register type code. This also holds for the C register.
- A boundary error (TermSTS = 0AH) occurs if a range spanning both the base T registers (0 to 511) and the expansion T registers (512 to 999) is specified for the T register.

- For data storage registers, a boundary error (TermSTS = 0AH) occurs if the specified area (start register + register range) does not exist in the local port.
- A response timeout error (TermSTS = 02H) occurs if the loopback response time exceeds the time specified by the timer counter value.
- If the local port registers are write protected by setting the special coil, the memory write protected status (TermSTS = 04H) is returned.



e. Sample program

This sample ladder program issues a remote port loopback request to the Ethernet Port.

This program is executed by turning on the R0092 A point.

A user program must compare the transmitted and loopback data for equivalence.

Parameters: 12544:3100H, 00015:000FH

```

|R0092
1|-| |--[12544 MOV W0903][00015 MOV W0904]-----|
|      /* Designates the module and sets the CMD number. */
|
|      +[00485 MOV W0905]-----|
|      /* Sets the loopback data length. */
|
|      +[00004 MOV W0906][00000 MOV W0907]-----|
|      /* Sets the send source register type and starting register
|number. */
|
|      +[00004 MOV W0908][00500 MOV W0909]-----|
|      /* Sets the loopback destination register type and starting
|register number. */
|
|      +[00100 MOV W0910]-----|
|      /* Sets the response time limit. (10 seconds) */
|
|      +[341995909 DMOV W0912•W0911][01025 MOV W0913]-----|
|      /* Loopback destination IP address, message transmission UDP
|port number */
|
|      +[W0903 SEND W0914][RST R0092]-----|
|      /* Sets the request. */

```

7.5 Time Setting

The Ethernet Port internal time is set from the T2N. This information is used as the time information in the Ethernet Port event trace.

**Time set request (using the SEND instruction)**

**a. Function**

This request sets the Ethernet Port internal time from a user program.

**b. Transfer parameters**

A	3100H		Module designation
A+1	CMD = 0018H		CMD number
A+2	Month	Year	
A+3	Hour	Day	
A+4	Seconds	Minute	

- **Enter the year, month, day, hour, minute, and seconds as BCD codes.**

The date July 20, 1996 14:30:00 would be entered as:

A	3100H	12544
A+1	0018H	00024
A+2	0796H	01942
A+3	1404H	05124
A+4	0030H	00048

**c. Status (See figure 4.3 for details on bits C, D, E and F.)**

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

**d. Important items**

- This request can be executed in run or standby mode.
- The time set here is updated by the Ethernet Port internal timer, and thus may get out of synchronization with the T2N time.

We recommend setting the Ethernet Port time once a day.

## e. Sample program

This sample ladder program issues a time set request to the Ethernet Port.

This program is executed by turning on the R0508 A point.

Parameters: 12544:3100H, 00024:0018H, 01942:0796H, 05124:1404H, 00048:0030H

```

|R0508                                                    R0608 |
1|-| |------( )--|
|           /* Time set request */
|R0608                                                    |
2|-| |--[12544 MOV RW000][00024 MOV RW001]-----|
|           /* Designates the module and sets the CMD number. */
|           |
|           |[01942 MOV RW002][05124 MOV RW003][00048 MOV RW004]-----|
|           /* Sets the year, month, day, hour, minute, and seconds. */
|           |
|           |[RW000 SEND RW010][RST R0508]-----|
|           /* Sets the request. */
|
|

```

## 7.6 RAS Information Readout

This request reads out the Ethernet Port RAS information (internal error information and phenomenon history) from a user program.

### RAS information readout request (using the RECV instruction)

**a. Function**

This request reads out the Ethernet Port RAS information from a user program.

**b. Transfer parameters**

A	3100H	Module designation
A+1	CMD = 0015H	CMD number
A+2	DRID	RAS information storage register type
A+3	DRegNO	RAS information storage register start address
A+4	Kind	RAS request type
A+5	Start	Start position: Valid when Kind is 2 or 4.
A+6	ReadCnt	Number of items read out: Valid when Kind is 4.

(1) RAS information storage register type code

This code specifies the type of the register used to store the read out RAS information.  
See figure 5.1 for the type codes that can be specified here.

(2) RAS information storage register start address

Specifies the start address for the registers specified by the local port register type code.

(3) RAS request type: specifies the type of RAS information read out

- 1: RAS counter .....LAN controller (line) information (See table 7.2.)
- 2: MIB information .....LAN controller interface, IP, TCP, and UDP information  
(See tables 7.3 to 7.6.)
- 3: Maintenance data (Cannot be used.)
- 4: Event trace information (See table 7.7.)
- 5: RAS information clear

(4) Start position: Only valid when the readout request type is 2 (MIB information) or 4 (trace information).

- MIB information (2) ..... 0 = MIB\_IF
- 1 = MIB\_IP
- 2 = MIB\_TCP
- 3 = MIB\_UDP

Trace information (4)..... 0 to 160; 0 corresponds to the most recent information.

(5) Number of items read out: Only valid when the readout request type is 4 (trace information).

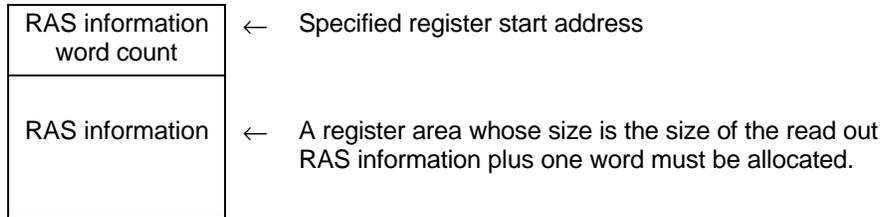
1 to 30 items.

c. Status (See figure 4.3 for details on bits C, D, E and F.)

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
B					0	0	0	0	TermSTS							
B+1	Detailed information (Only valid when TermSTS = 0BH.)															

d. Important items

- This request can be used in run and standby modes.
- The read out RAS information is stored in the specified register area as shown below.



- A boundary error (TermSTS = 0AH) occurs if it is not possible to allocate an area that is the size of the read out RAS information plus one word for the RAS information storage area.
- A parameter error (TermSTS = 03H) occurs on any of the following transfer parameter settings.
  - If the RAS request type (Kind) is out of range (a value other than 1 to 5).
  - If the start position (Start) is out of range (a value of 4 or larger) for MIB information.
  - If the start position (Start) is out of range (a value of 161 or larger) for event trace information.
  - If the read count (ReadCnt) is out of range (a value of 31 or larger) for event trace information.

## e. Sample program

This sample ladder program issues a RAS information readout to the Ethernet Port.

This program is executed by turning on the R0505 A contact.

Parameters: 12544:3100H, 00021:0015H

RAS storage registers: Starting at D0000.

RAS request type: Event trace information, reads out 30 items starting with the most recent data.

```

|R0505                                                    R0605 |
1|-| |------( )--|
|          /* RAS information readout          */
|R0605 |
2|-| |-[12544 MOV RW000][00021 MOV RW001]-----|
|      /* Designates the module and sets the CMD number.          */
|      |
|      |[00004 MOV RW002][00000 MOV RW003]-----|
|      /* Sets the RAS storage register type and start number      */
|      |
|      |[00004 MOV RW004]-----|
|      /* Sets the RAS request type.          */
|      |
|      |[00000 MOV RW005][00030 MOV RW006]-----|
|      /* Sets the information readout start position and count.    */
|      |
|      |[RW000 RECV RW010][RST R0505]-----|
|      /* Sets the request.          */
|

```

f. RAS information details

- (1) RAS counters.....LAN controller (line) information  
 Reads out 128 words of data using RAS information readout.

**Table 7.2 RAS Counters**

Symbol	Description
RAS_CNT [0]	Normal reception count
RAS_CNT [1]	Bus read error (reception buffer read failure) count
RAS_CNT [2]	Remote reset packet reception count (Packets whose Ethernet header type was set to 0900H.)
RAS_CNT [3]	Short packet (packet length under 60 bytes) reception count
RAS_CNT [4]	Alignment error (The number of bits in the received data was not divisible by 8.) count
RAS_CNT [5]	CRC error (received packet CRC check error) count
RAS_CNT [6]	Overflow (Incoming packets discarded due to full receive buffer.) count
RAS_CNT [7]	Remaining registers unused
RAS_CNT [8]	•
•	•
•	•
•	•
RAS_CNT [127]	•

- (2) MIB information: MIB\_IF.....LAN controller interface information  
 Reads out 13 words of data using RAS information readout.

**Table 7.3 MIB\_IF**

Symbol	Description
MIB_IF [0]	It is possible to use the interface number in this system
MIB_IF [1]	Reserved
MIB_IF [2]	Lower level layer of IP with interface type of protocol
MIB_IF [3]	Progress of time with changing states by the interface
MIB_IF [4]	Total number of octets (bytes) received by the interface
MIB_IF [5]	Number of non-broadcast/non-multicast packets transferred upstream
MIB_IF [6]	Number of broadcast/multicast packets transferred upstream
MIB_IF [7]	Number of packets discarded due to reception resource limitation
MIB_IF [8]	Number of packets discarded due to format errors
MIB_IF [9]	Number of packets sent to an undefined protocol
MIB_IF [10]	Total number of octets sent by the interface
MIB_IF [11]	Number of packets from upstream that were not broadcast or multicast packets
MIB_IF [12]	Number of packets from upstream that were broadcast or multicast packets

Packet: A unit of data flowing on the transmission path. (See figure 7.1.)

- (3) MIB information: MIB\_IP.....IP protocol information  
Reads out 17 words of data using RAS information readout.

**Table 7.4 MIB\_IP**

Symbol	Description
MIB_IP [0]	Indicate IP gateway in action or not action
MIB_IP [1]	Default TTL (Time to Live) of IP packet
MIB_IP [2]	Total number of IP datagrams received from the interface
MIB_IP [3]	Number of IP datagrams discarded due to format errors
MIB_IP [4]	Number of IP datagrams discarded due to incorrect delivery.
MIB_IP [5]	Number of IP datagrams sent out.
MIB_IP [6]	Number of IP datagrams sent to an undefined protocol
MIB_IP [7]	Number of IP datagrams transferred to upstream
MIB_IP [8]	Number of IP datagrams transferred from upstream
MIB_IP [9]	Number of IP datagrams discarded due to not exist route
MIB_IP [10]	The maximum number of seconds that maintained reception fragment and waiting reassembly
MIB_IP [11]	Number of received IP fragments that required reassembly
MIB_IP [12]	Number of IP datagrams that were reassembled successfully
MIB_IP [13]	Number of IP datagrams for which reassembly failed
MIB_IP [14]	Number of IP datagrams that were fragmented successfully
MIB_IP [15]	Number of IP datagrams for which fragmentation failed an which were discarded
MIB_IP [16]	Number of created IP fragments

IP datagram: The IP header and IP data sections of a packet (See figure 7.1.)

IP fragment: Items that are divided into multiple IP datagrams when sending over 1500 bytes of data.

Reassembly: Restoring data to its original form from fragmented IP datagrams

- (4) MIB information: MIB\_TCP .... TCP protocol information  
Reads out 10 words of data using RAS information readout.

**Table 7.5 MIB\_TCP**

Symbol	Description
MIB_TCP [0]	The maximum of resent timeout (per ms)
MIB_TCP [1]	Number of active open connections
MIB_TCP [2]	Number of passive open connections
MIB_TCP [3]	Number of times the connection open operation failed
MIB_TCP [4]	Number of currently open connections
MIB_TCP [5]	Number of segments received
MIB_TCP [6]	Number of segments sent
MIB_TCP [7]	Number of segments resent
MIB_TCP [8]	Number of segments discarded due to format errors
MIB_TCP [9]	Number of generated resets

Segment: The TCP header and TCP data block in an IP datagram (See figure 7.1.)

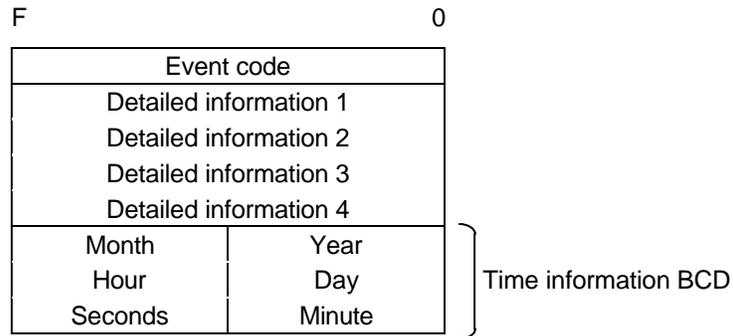
- (5) MIB information: MIB\_UDP ... UDP protocol information  
Reads out 4 words of data using RAS information readout.

**Table 7.6 MIB\_UDP**

Symbol	Description
MIB_UDP [0]	Number of UDP datagrams transferred upstream
MIB_UDP [1]	Number of datagrams addressed to unused ports
MIB_UDP [2]	Number of datagrams discarded due to format errors
MIB_UDP [3]	Number of datagrams transferred from upstream

UDP datagrams: The UDP header and UDP data block in an IP datagram (See figure 7.1.)

- (6) Event trace information
  - Record size: 16 bytes
  - Number of records: 160
  - Operation on overflow: Old information is updated. This means that it is always possible to verify the 160 most recent event trace records.
  - Format: Since this format is for Ethernet Port internal data, it is binary codes. However, the time is expressed in BCD.



- Event trace item: See table 7.7

**Table 7.7 Event Trace Items**

Event code	Detail information 1 (H)	Detail information 2 (H)	Detail information 3 (H)	Detail information 4 (H)	Content
0001H	ROM error (0160) RAM error (0260) DPRAM error (0360)				Initialization error
0002H	Program address	Error code	NMI (0000) TRAP (0001) Watchdog timer check (0002) Watchdog timer check (0003)	Read port contents None Watchdog timer flag Watchdog timer flag	NMI occurrence factor
0004H	Power on (0001)				Start type
	Reset switch (0002)				
	Software reset (0003)	Request code	Station status	Write port contents	
0005H	Initialization (0000)	Mode prior to change	Mode after change		Mode change (station status)
	Control request (0001)	Mode prior to change	Mode after change		
0007H	Task ID	Memory pool number	Error code		Instruction storage buffer allocation error
0008H	Task ID	Station status			Transmission prohibit
0009H	Receive response code				T2N driver reception or response code error
0100H	CMD number (0011) CMD number (0012) CMD number (0013)	Error code	Station status	Write port contents	T2N-Ethernet Port interface completion error
			UDP port (0001)	UDP port number	
			Station status (0002)	Station status	
			Mode prior to change	Mode after change	
			Task start number	Task completion status	
			MAC setting flag	IP setting flag	
0200H	Task ID	Error code	socket (0001)		Message transmission completion error
			bind (0002)		
			sendto (0003)		
			length (0004)	Length	
			recvfrom (0005)		
0300H	Task ID	Request code	Error code		Socket transmission completion error

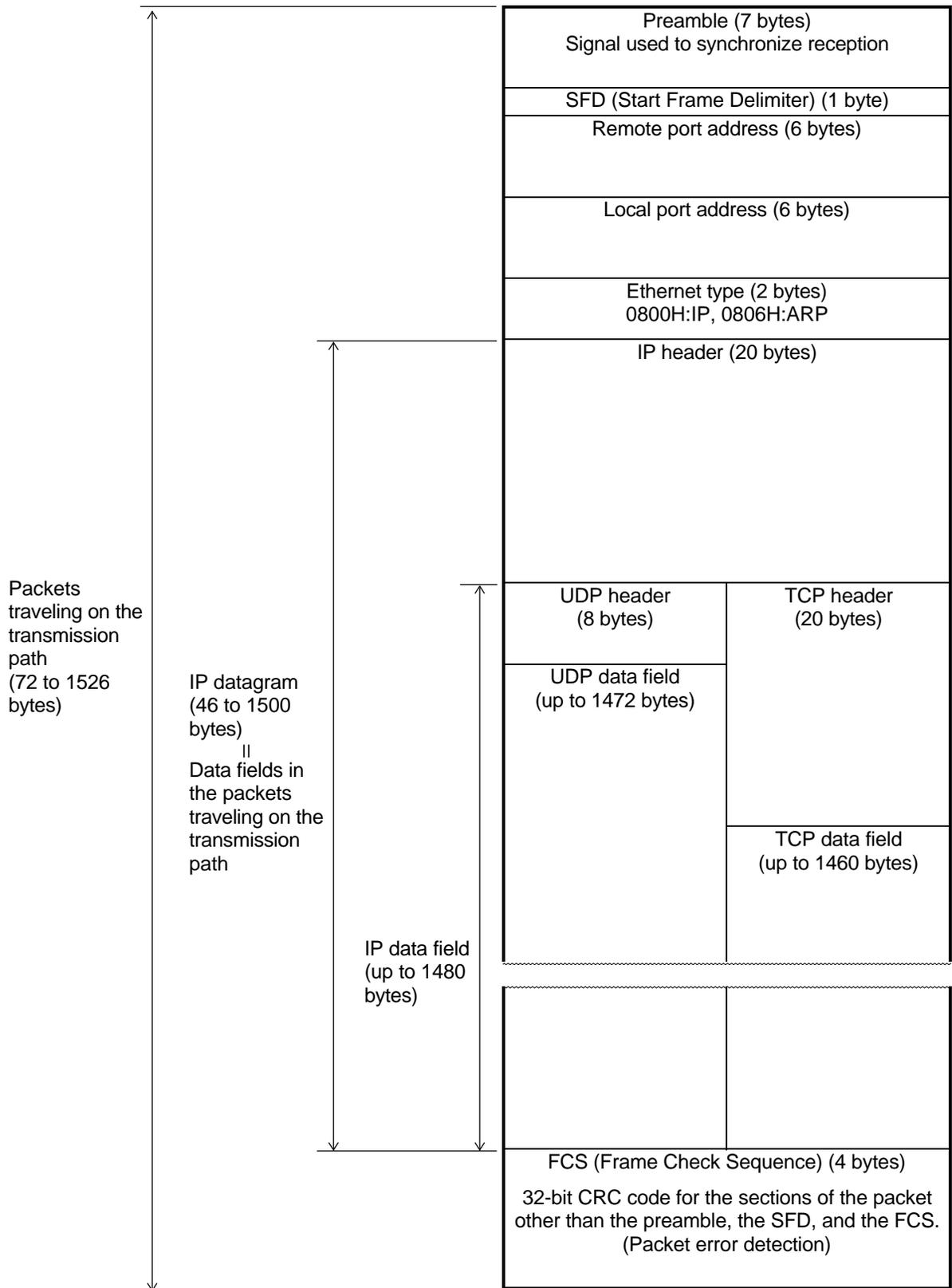


Figure 7.1 Packet Format/IP Datagram

## 8. Error Handling

This chapter describes the error analysis and recovery procedures for errors that occur while using the Ethernet Port. Also refer to the T2N user's manual when recovering from Ethernet Port-related errors.

### 8.1 LED Display

The Ethernet Port indicates its Ethernet Port status in the ERUN, ET/R and EACC LEDs.

Table 8.1 lists the state indicated by the ERUN LED. Note that the Ethernet Port operating mode can be determined by the combination of the ERUN and ET/R LED display states as listed in table 8.2. User programs should read out and verify the down information error codes listed in table 8.2 using the READ instruction issued for the down information in the T2N interface buffer. (See section 7.2, Down Information.)

**Ethernet Port that fail (go to the down state) should be replaced.**

**Table 8.1 States Indicated by the ERUN LED**

LED display	State
ERUN (Run)	Indicates the Ethernet Port normal/error (down mode) status. Lit ..... Ethernet Port normal Off ..... Ethernet Port error (down)

**Table 8.2 Ethernet Port Down States Indicated by ERUN, ET/R LED Combinations**

LED combination displayed	State	
ERUN lit	Ethernet Port operating normally in standby mode (parameter setup wait state) The Ethernet Port goes to this state after power is first applied and after a reset.	
	Ethernet Port operating normally in run mode (transmission possible) The Ethernet Port switches to this mode in response to an operating mode control request. (Parameter setup is required to switch the Ethernet Port to run mode.)	
ERUN off, ET/R (Ethernet transmission/ receive) blinking	An error occurred during operation and the Ethernet Port is down.	
	Possible causes of the Ethernet Port going down	Down information error code
	Watchdog timeout occurred	0010
	Memory bus stall occurred	0020
	Trap occurred	0030
	Jabbering timeout occurred	0040
	Semaphore acquisition error	0500, 0501
	Semaphore acquisition error	0502, 0503
	MBX reception error	0504
	MBX transmission error	0505
	No empty TCP resend queue	0506
	Acquired buffer was in use	0507
	Initial buffer acquisition error	0508
	Down mode due to the occurrence of an error during self diagnostics at power on.	
	Possible causes of the Ethernet Port going down	Down information error code
	LAN controller check error	0050
	ROM BCC check found a discrepancy	0160
System RAM area check error	0260	
T2N interface buffer check error	0360	

The EACC LED indicates when the T2N is accessing the Ethernet Port.

**Table 8.3 States Indicated by the EACC LED**

LED display	State
EACC (Access)	<p>Indicates whether or not the T2N is accessing the Ethernet Port.</p> <ul style="list-style-type: none"> <li>• Lit ..... The T2N is accessing the Ethernet Port.</li> <li>• Off ..... The T2N is not accessing the Ethernet Port.</li> <li>• Lights after Ethernet Port initialization completes when power is first applied. (Access starts.)</li> </ul> <p>If this LED does not light after power is turned on:            Verify that the T2N is correctly connected to the base unit.</p> <p>If this LED does not light at above, then the T2N itself is defective. Replace the T2N.</p> <p><b>Supplement:</b>            We recommend keeping spares on hand to minimize the time required to determine the location of the fault using and to minimize the time required for recovery.</p>

## 8.2 Status (error status)

The status indicates the status during SEND/RECV instruction execution or after execution completes. If a SEND/RECV instruction does not complete normally, refer to the detailed information (Ethernet Port error response) and the status and review the instruction word format at the T2N and Ethernet Port states (operating modes).

The detailed information is stored in the register following the completion status when the completion status TermSTS value indicates a transmission error (OBH).

Figure 8.1 shows the structure of the status information. Tables 8.4 and 8.5 list the TermSTS values, and table 8.6 lists the contents of the detailed information.

The status (error status) has the following structure.

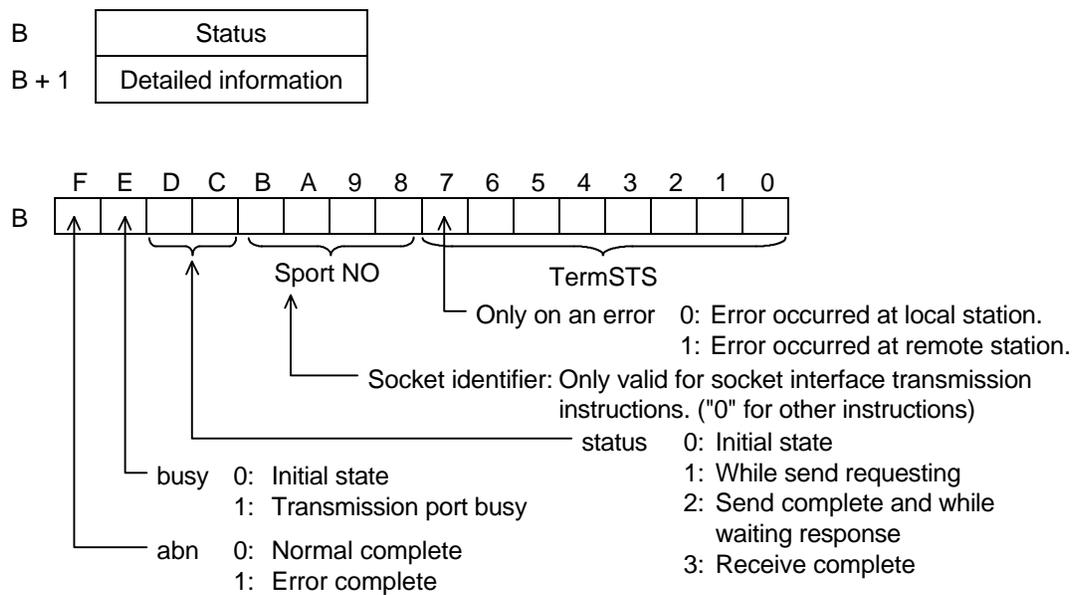


Figure 8.1 Status Structure

Table 8.4 TermSTS (Part 1)

TermSTS	Status	Meaning
00H	Normal complete	
01H	Register specification error	<ul style="list-style-type: none"> <li>Occurs if a register other than XW, YW, W, LW, RW, D, F, T, C, or SW is specified as the data storage register in one of the following requests. Requests: RAS information readout, remote port loopback, UDP send/receive, TCP send/receive, register read/write.</li> <li>In a register read or write request, the request specified a read or write from a T register to a register other than a T register. (This also applies to C registers.)</li> </ul>
02H	Response timeout	<ul style="list-style-type: none"> <li>Occurs if a response is not received within the specified time for a remote port loopback or register read/write request.</li> </ul>

Table 8.5 TermSTS (Part 2)

TermSTS	Status	Meaning
03H	Parameter error	<ul style="list-style-type: none"> <li>Occurs if a value other than 3 is specified for the module ID.</li> <li>Occurs if a value other than 1 is specified for the channel number.</li> <li>Occurs if a command number other than one of the following is specified for a SEND instruction. 000FH, 0011H - 0014H, 0018H, 0021H, 0031H, 0032H, 0034H, 0035H, 0037H, 0039H</li> <li>Occurs if a command number other than one of the following is specified for a RECV instruction. 0015H, 0021H, 0033H, 0038H</li> <li>Occurs if a value other than 1 to 8 is specified for the socket identifier. (See section 6.4 "Using the Ethernet Port Socket Interface".)</li> <li>Occurs if an incorrect parameter is specified in a RAS information readout request. (See section 7.6 "RAS Information Readout".)</li> </ul>
04H	Memory write protect	<ul style="list-style-type: none"> <li>Occurs if an attempt to write is made when the remote registers were write protected.</li> <li>Occurs if an attempt to read is made when the local registers were write protected.</li> </ul>
05H	(Reserved)	
06H	Module error	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port is down.</li> </ul>
	Initialization in progress	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port is initializing.</li> </ul>
	Transmission complete timeout	<ul style="list-style-type: none"> <li>Occurs if a T2N internal timer times out. Requests: UDP open, UDP send, UDP close, TCP send, TCP close</li> </ul>
07H	No send channel (CH)	<ul style="list-style-type: none"> <li>Occurs if the corresponding Ethernet Port is not installed.</li> </ul>
08H	Port address out of range	<ul style="list-style-type: none"> <li>Occurs if the local port IP address was specified for the remote destination IP address. Requests: Remote port loopback, register read/write</li> </ul>
09H	Incorrect transmission word count	<ul style="list-style-type: none"> <li>Occurs if either 0 words or 486 or more words were specified for either a register read/write or a remote port loopback request.</li> <li>Occurs if either 0 words or 1001 or more words were specified for a UDP/TCP send/receive request.</li> </ul>
0AH	Boundary error	<ul style="list-style-type: none"> <li>Occurs if the specified area (start address + register range) does not exist in the T2N data storage registers. Requests: RAS information readout, remote port loopback, UDP send/receive, TCP send/receive, register read/write</li> </ul>
0BH	Transmission error	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port returns an error response. Detailed information (the Ethernet Port response status) is stored.</li> </ul>
0CH	No I/O response	<ul style="list-style-type: none"> <li>Occurs if the T2N cannot access the Ethernet Port.</li> </ul>
0DH	(Reserved)	
0EH	Send data capacity exceeded	<ul style="list-style-type: none"> <li>Occurs if a request is discarded when a T2N internal resource insufficiency occurs due to increasing amounts of transfer data.</li> <li>Occurs if a T2N retransmission times out.</li> </ul>
0FH	(Reserved)	

When a transmission error (0BH) occurs, details of the Ethernet Port error response are stored in the register following the status. (See table 8.6.)

**Table 8.6 Detailed Information (Ethernet Port Error Responses)**

Error	Code	Description
Normal complete	0001H	Request completed normally
Local port fault	0002H	The local port is in down mode
Local port standby	0003H	Data transmission in standby mode (socket interface transmission or PC link protocol transmission) was requested.
Timeout	0020H	Timeout occurred during TCP open processing or receive processing, or during UDP receive processing.
Length error	0030H	Incorrect send data length
Port mode error	0040H	One of the following requests was issued in a mode other than standby mode: a. Parameter setup, b. MAC address setup, c. Reset
MAC/IP not set up	0050H	Occurs if the MAC address or IP address is not set up.
Transmission prohibited state	0060H	<ul style="list-style-type: none"> <li>Occurs if a PC link request was issued from the T2N in run mode in the message transmission prohibited state.</li> <li>Occurs if a socket open, close, send, or receive request is issued by the T2N in run mode in the socket interface transmission prohibited state.</li> </ul>
Format error	0070H	Occurs when there is a format error in the request text.
	0071H	Port number error
	0072H	Request code error
	0073H	Time setting: Year
	0074H	Time setting: Month
	0075H	Time setting: Day
	0076H	Time setting: Hour
	0077H	Time setting: Minute
0078H	Time setting: Second	
Opened	0080H	Occurs if an open request is issued for an already open socket.
Unopened	0081H	Occurs if send, receive, or close request is issued for an unopened socket.
Incorrect socket identifier	0082H	Occurs if the socket identifier is out of range (1 to 8).
Incorrect control request specification	0083H	<p>Occurs if a request that was prohibited by an operating mode control request is issued.</p> <ul style="list-style-type: none"> <li>Occurs if a run request is issued in run mode.</li> <li>Occurs if a standby request is issued in standby mode.</li> <li>Occurs if a run request and a standby request are issued at the same time.</li> </ul>
LAN controller driver error	0090H	Occurs if the setting of the local port IP address and port number in the area reserved for transmission fails.
	0091H	Error in the TCP or UDP protocol (such as a transmission phase error)
	0092H	<ul style="list-style-type: none"> <li>Occurs if the setting of the remote port IP address in the remote port information area (the NETDATA table) fails.</li> <li>Occurs if the remote port IP address network address differs from the local port IP address network address.</li> </ul>
Memory pool allocation failure	00A0H	Occurs if the Ethernet Port internal OS fails to allocate memory.
Port 2 task start failure	00A1H	Occurs if computer link or PC link socket identifier acquisition fails.
Broadcast specification error	00A2H	Occurs if the remote IP address is broadcast on a remote port verification request.
MAC address specification error	00B0H	BCC error
	00B1H	Occurs if an error occurs after address saving on a MAC setup request.
Insufficient resources error	00C0H	Occurs if there are insufficient resources for UDP send or PC link transmission.

### 8.3 T2N Special Relays and Registers

The Ethernet Port status information is reflected in the T2N special relays and special registers.

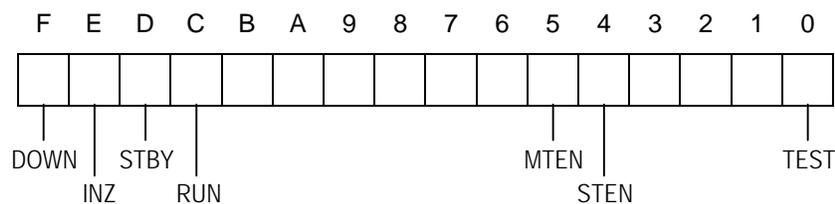
1. Special relays

Special relay	Name	Function
S000B	Ethernet Port error (warning)	1: The installed Ethernet Port unit is down. (The T2N continues to operate.) 0: Reset the system from a user program after resolving the error in the down Ethernet Port.

2. Special registers

Special registers	Name
SW063	CH1 station status

3. Station status format



- Bit F: DOWN (down)..... 1: Down mode, 0: Other than down mode
- Bit E: INZ (initialization)..... 1: Initialization in progress  
0: Initialization completed
- Bit D: STBY (standby)..... 1: Standby mode  
0: Other than standby mode
- Bit C: RUN (run)..... 1: Run mode, 0: Other than run mode
- Bit 5: MTEN (Message transmission)..... 1: Enabled, 0: Prohibited
- Bit 4: STEN (socket interface transmission)..... 1: Enabled, 0: Prohibited
- Bit 0: TEST (test)..... 1: Test function execution in progress

### 8.4 Socket Interface Information

The information for the eight socket interface sockets held by the Ethernet Port can be read out with the T2N READ instruction. (See Section 6.3 Socket Interface Information.) The Ethernet Port holds five words of socket status information for each socket.

- Remote port TCP/UDP port number
- Remote port IP address
- TCP/UDP port number for this socket
- Socket status

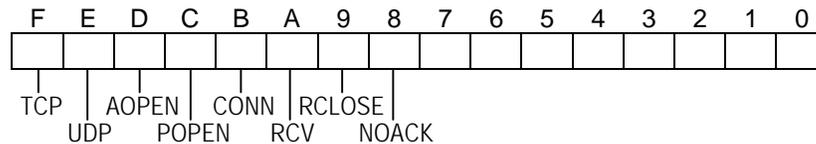
Table 8.7 lists the structure of the socket interface information and the extended memory area addresses.

**Table 8.7 Socket Interface Information**

	Socket identifier							
	1	2	3	4	5	6	7	8
Remote port number (TCP only)	0F4BH	0F53H	0F5BH	0F63H	0F6BH	0F73H	0F7BH	0F83H
Remote port IP address (TCP only)	0F4CH 0F4DH	0F54H 0F55H	0F5CH 0F5DH	0F64H 0F65H	0F6CH 0F6DH	0F74H 0F75H	0F7CH 0F7DH	0F84H 0F85H
Local port TCP/UDP port number	0F4EH	0F56H	0F5EH	0F66H	0F6EH	0F76H	0F7EH	0F86H
Socket status	0F4FH	0F57H	0F5FH	0F67H	0F6FH	0F77H	0F7FH	0F87H

For UDP sockets, the remote port number and IP address data are invalid.

The socket status has the following structure.



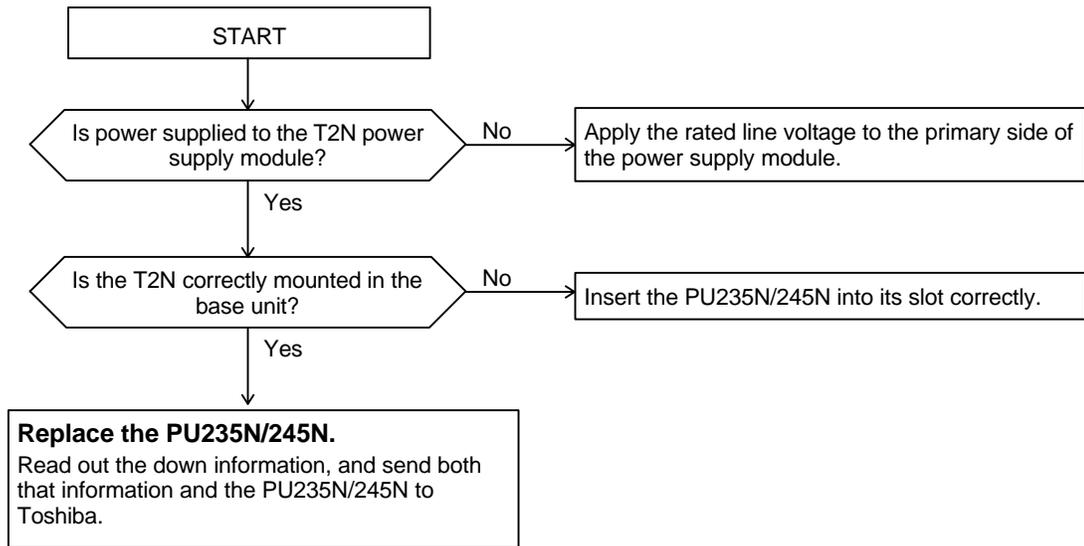
Bit no.	Bit	Condition under which the bit is set to 1	Condition under which the bit is reset to 0
F	TCP	TCP used by the socket	The socket closed in response to a close request
E	UDP	UDP used by the socket	The socket closed in response to a close request
D	AOPEN	TCP socket in active open state	The socket closed in response to a close request
C	POPEN	TCP socket in passive open state or UDP socket in open state	The socket closed in response to a close request
B	CONN	TCP connection established	Disconnected by the local port with a close request Disconnected by the remote port
A	RCV	Received text present	No received text
9	RCLOSE	An established TCP connection disconnected by the remote port	The socket closed in response to a close request
8	NOACK	No response received from the remote port for a local port send operation in a state where a TCP connection was established, and the TCP resend function resulted in a timeout (because the remote port was down, etc.).	The socket closed in response to a close request
		If the wait time exceeds the set value during a TCP active open request. This state does not occur for TCP passive open requests.	When a TCP active open request is reissued.

A socket status in which neither AOPEN nor POPEN is set to 1 indicates that the socket is in the closed state.

## 8.5 Troubleshooting

### 1. If the Ethernet Port fails to come up normally (i.e., does not reach standby state.)

If the Ethernet Port is functional, then the Ethernet Port can be initialized and brought to the standby state (with the ERUN LED lit) by applying power or by issuing a reset request.



### 2. If the Ethernet Port fails during operation

Recovery procedure	<p><b>Replace the PU235N/245N.</b></p> <p>Before replacing the PU235N/245N, read out the down information, and then send both that information and the PU235N/245N to Toshiba.</p>
--------------------	--

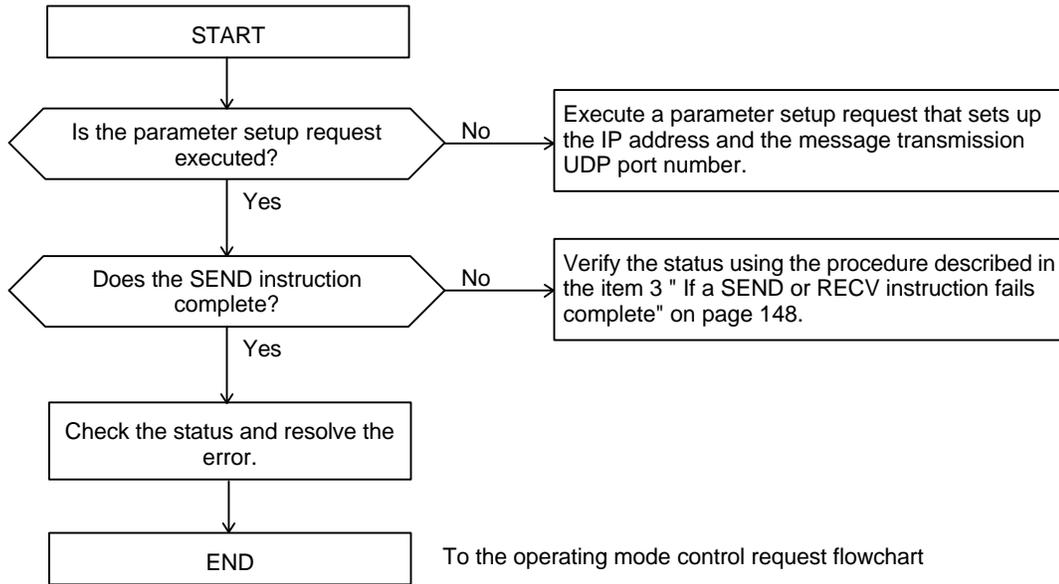
### 3. If a SEND or RECV instruction fails complete

Recovery procedure	<p>(1) Verify that the inputs of function block are held in the on state until the instruction completes.</p> <p>(2) Determine whether the user program is writing to the status register while the instruction is executing.</p>
--------------------	---

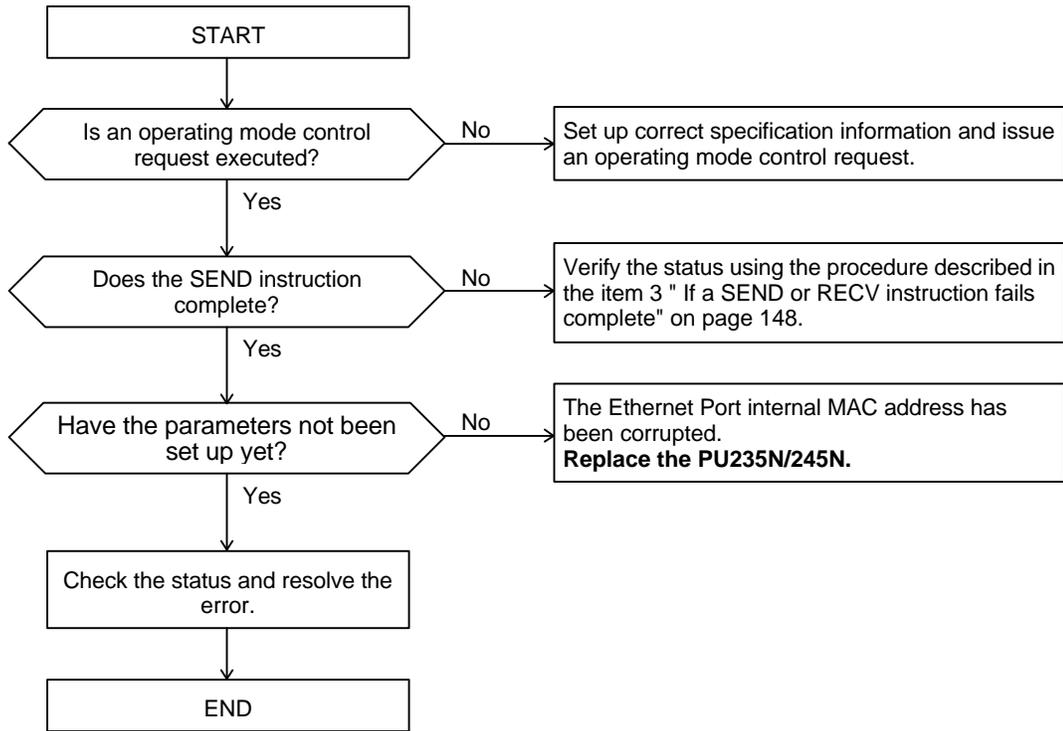
4. If the Ethernet Port fails to enter run mode

This section assumes that the Ethernet Port has come up normally up to this point. Note that the parameters must be set up with a parameter setup request before issuing the operating mode control request used to switch the Ethernet Port from standby mode to run mode.

a. Parameter setup request

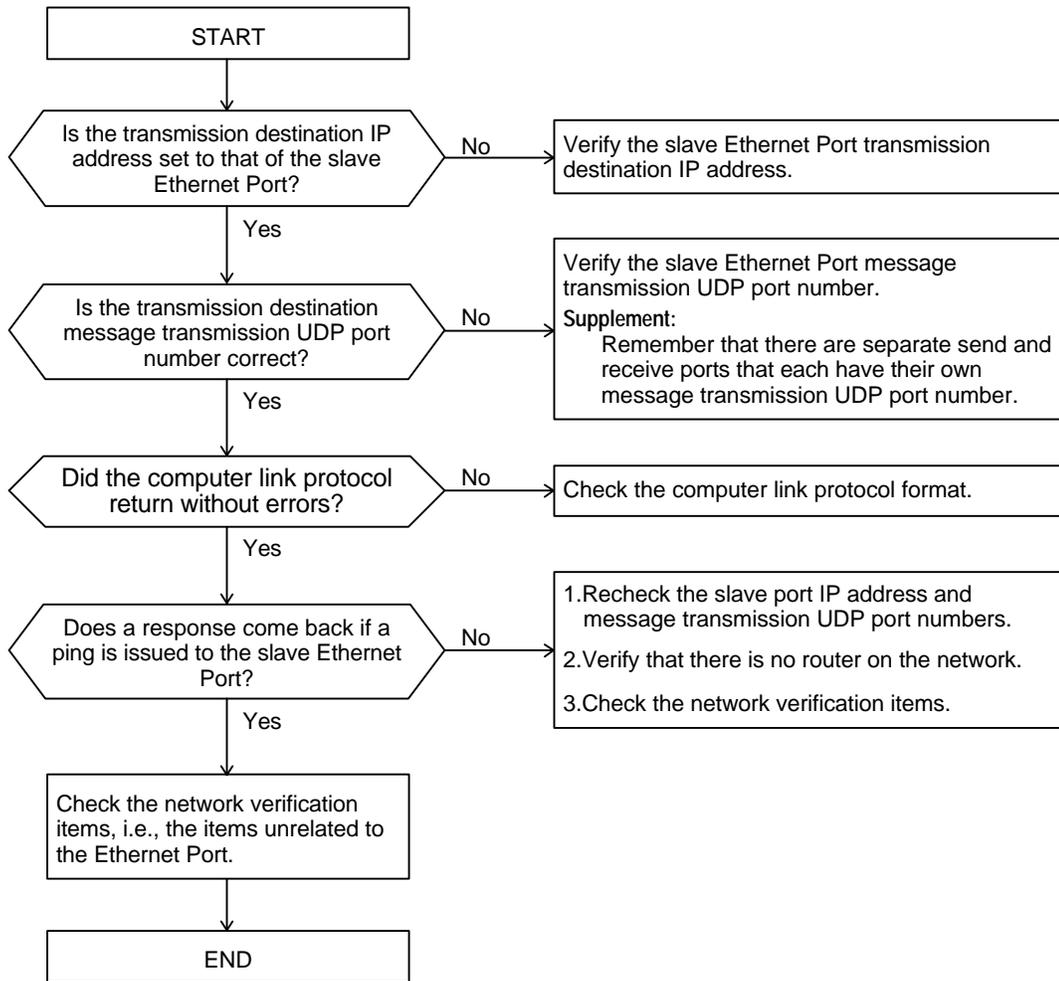


b. Operating mode control request

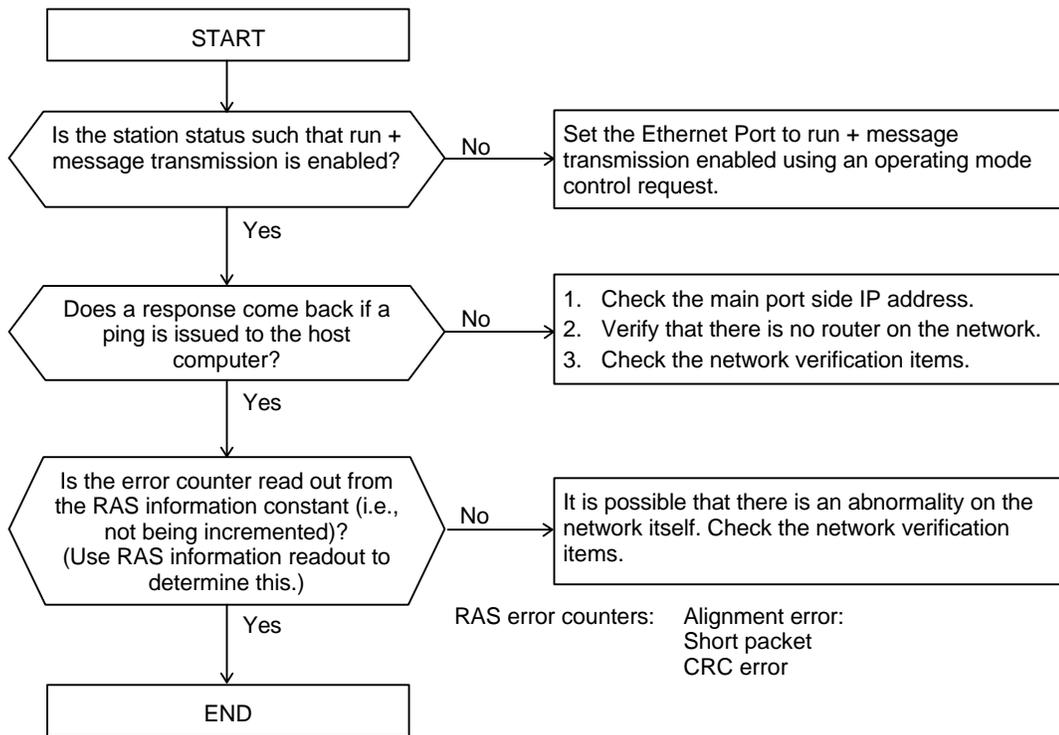


5. If computer link protocol transmission fails

a. Host computer side (main port)

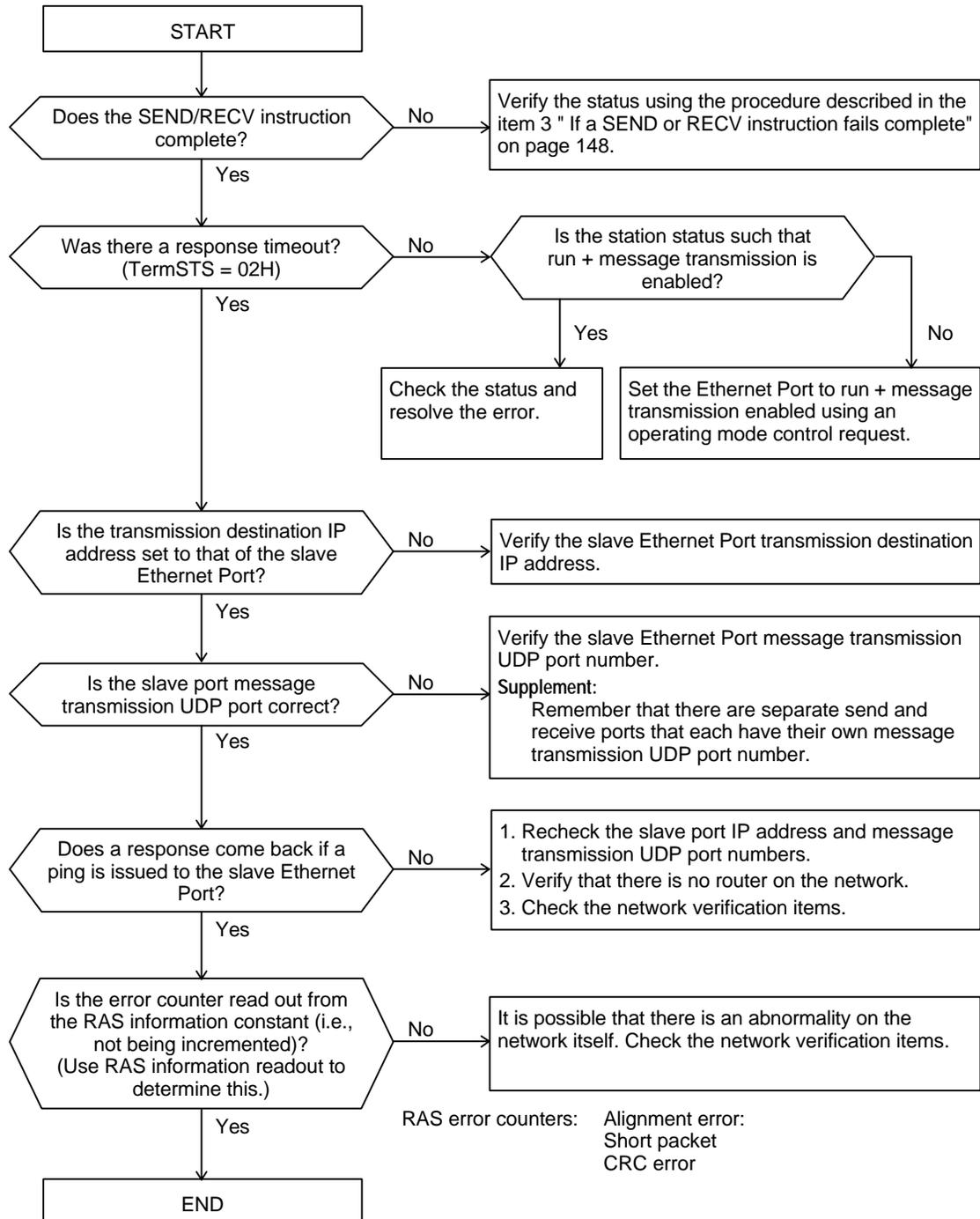


b. Ethernet Port side (slave port)

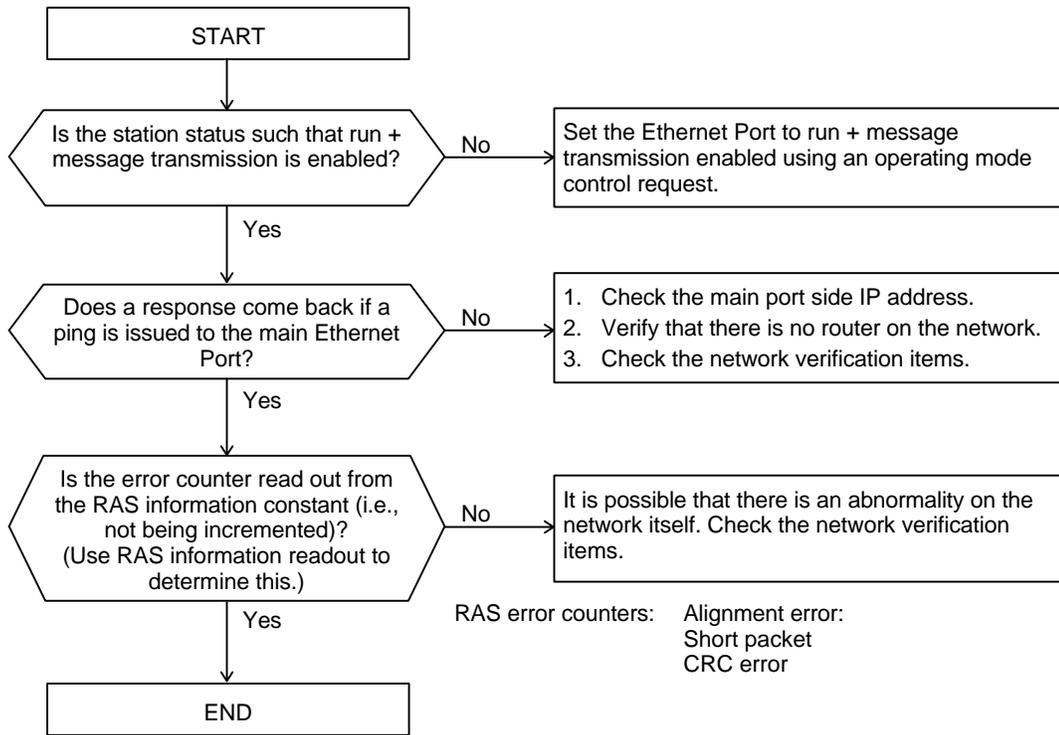


6. If PC link protocol transmission fails

a. Main port side Ethernet Port

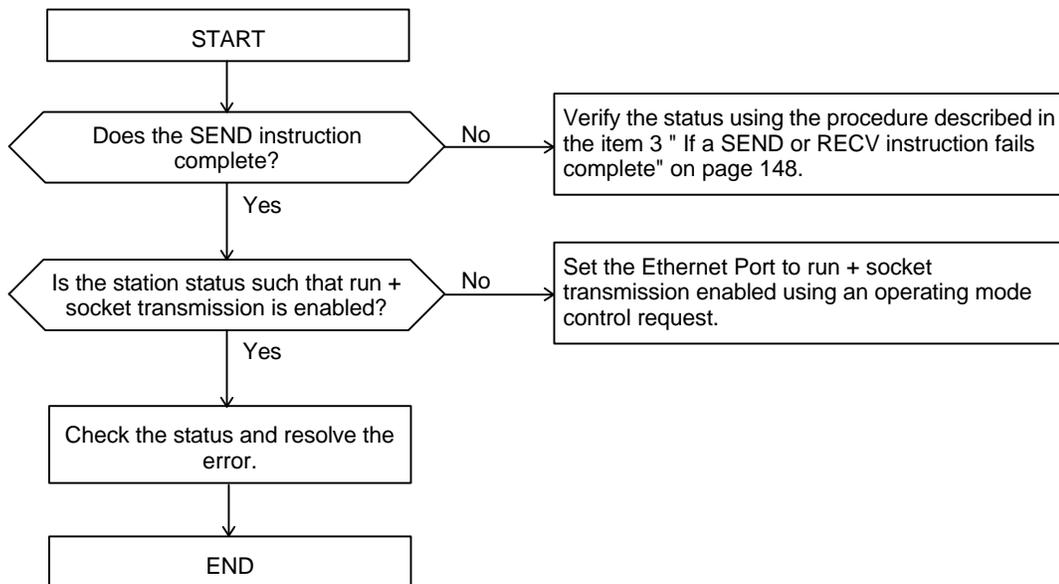


b. Ethernet Port side (slave port)

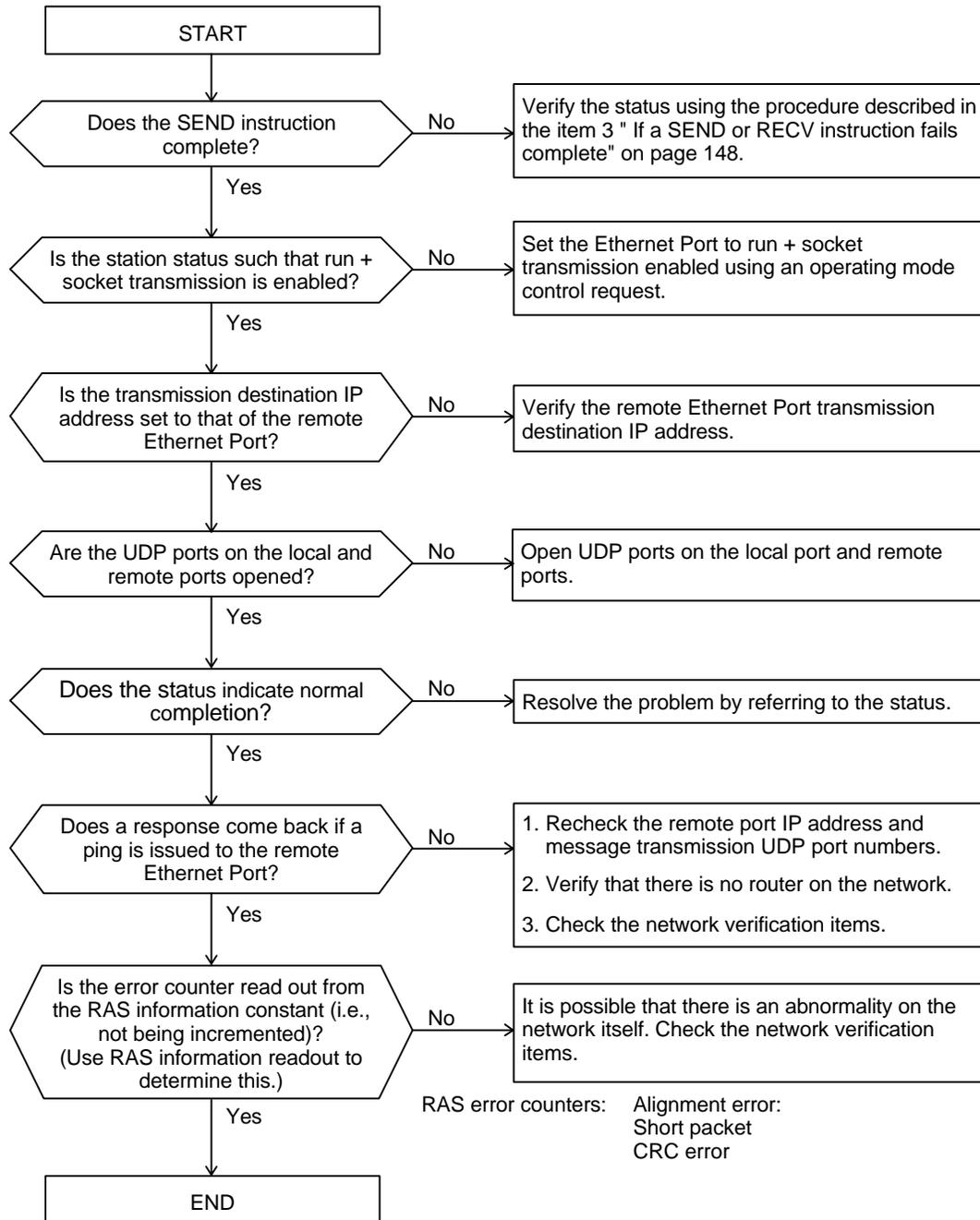


7. If transmission using UDP sockets fails

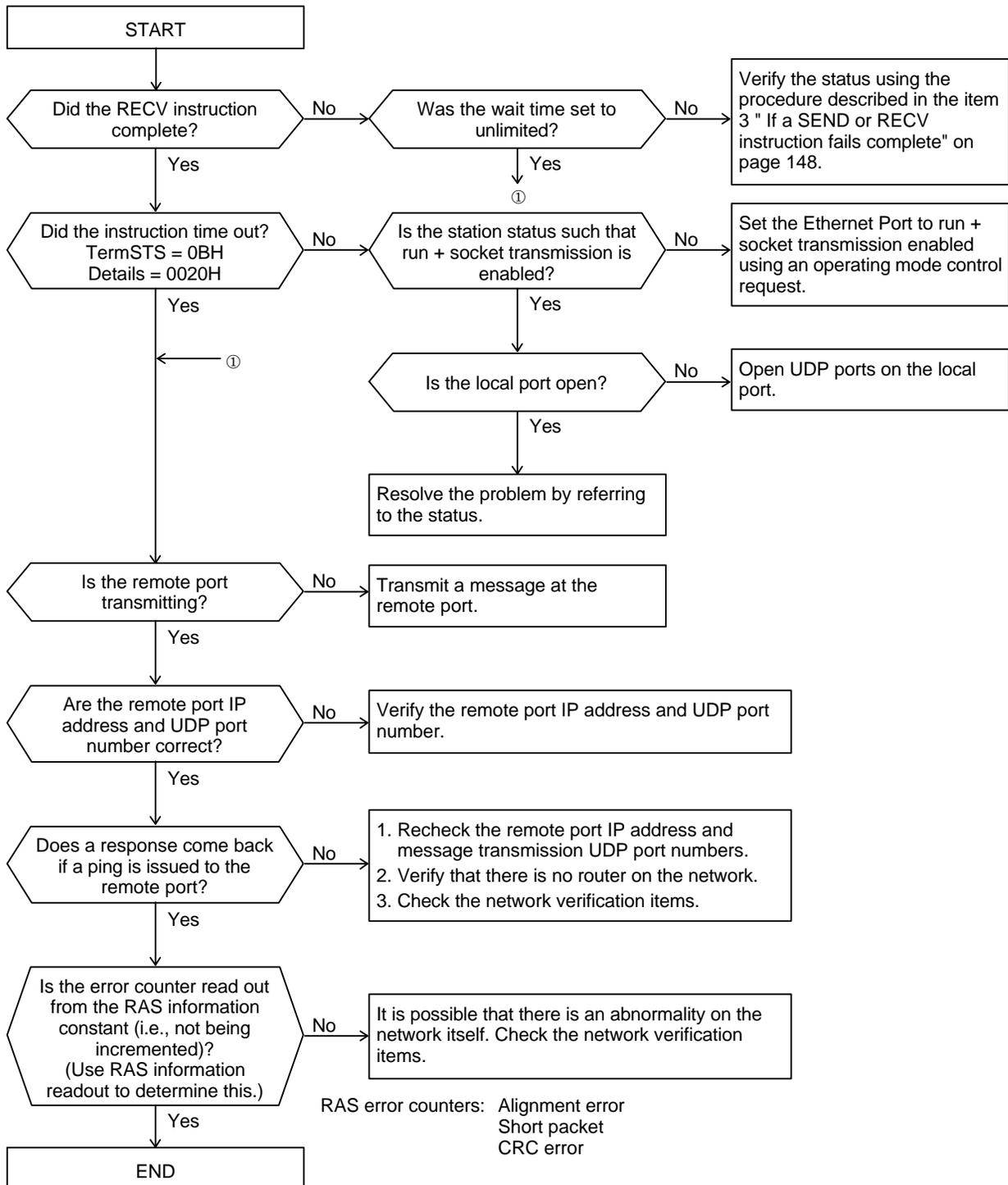
a. Open and close requests



b. Send request



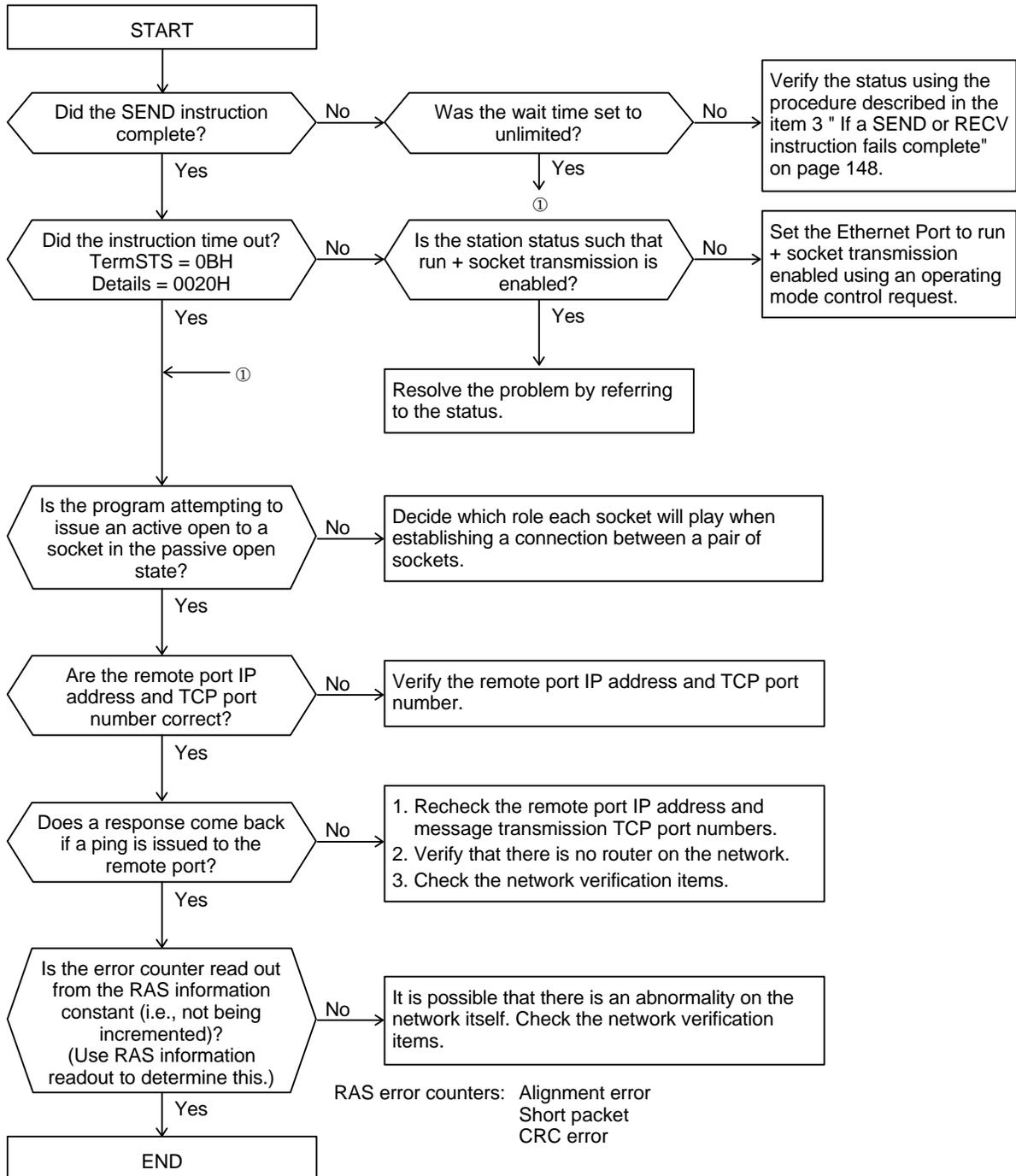
c. Receive request



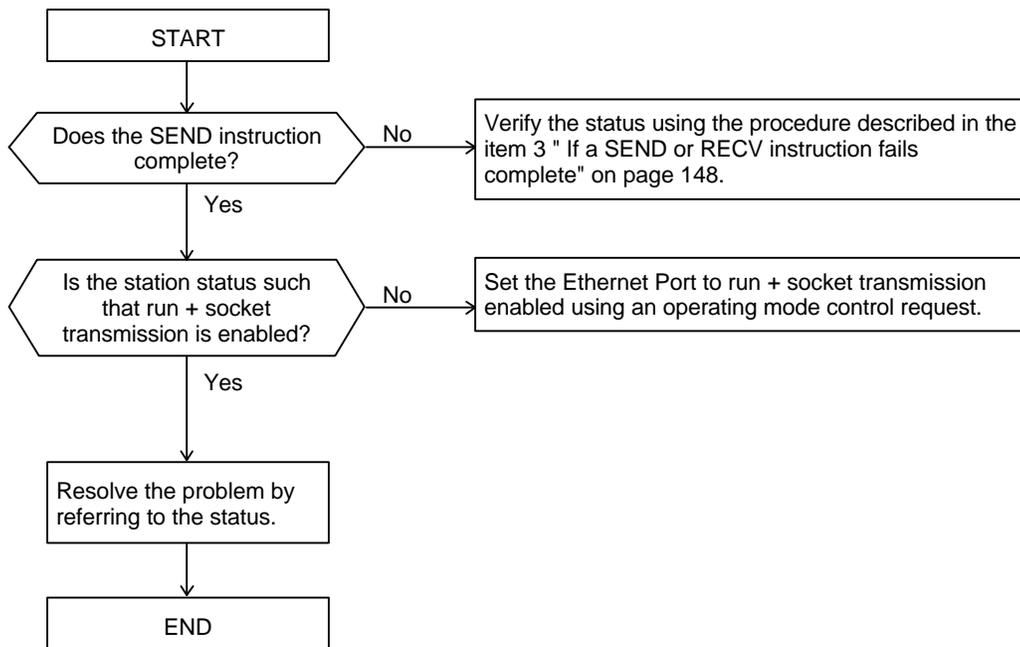
8. If Transmission Using TCP Sockets Fails

When troubleshooting TCP sockets, also refer to the section "TCP socket interface notes" in section 6.2, Ethernet Port Socket Interface Usage Notes.

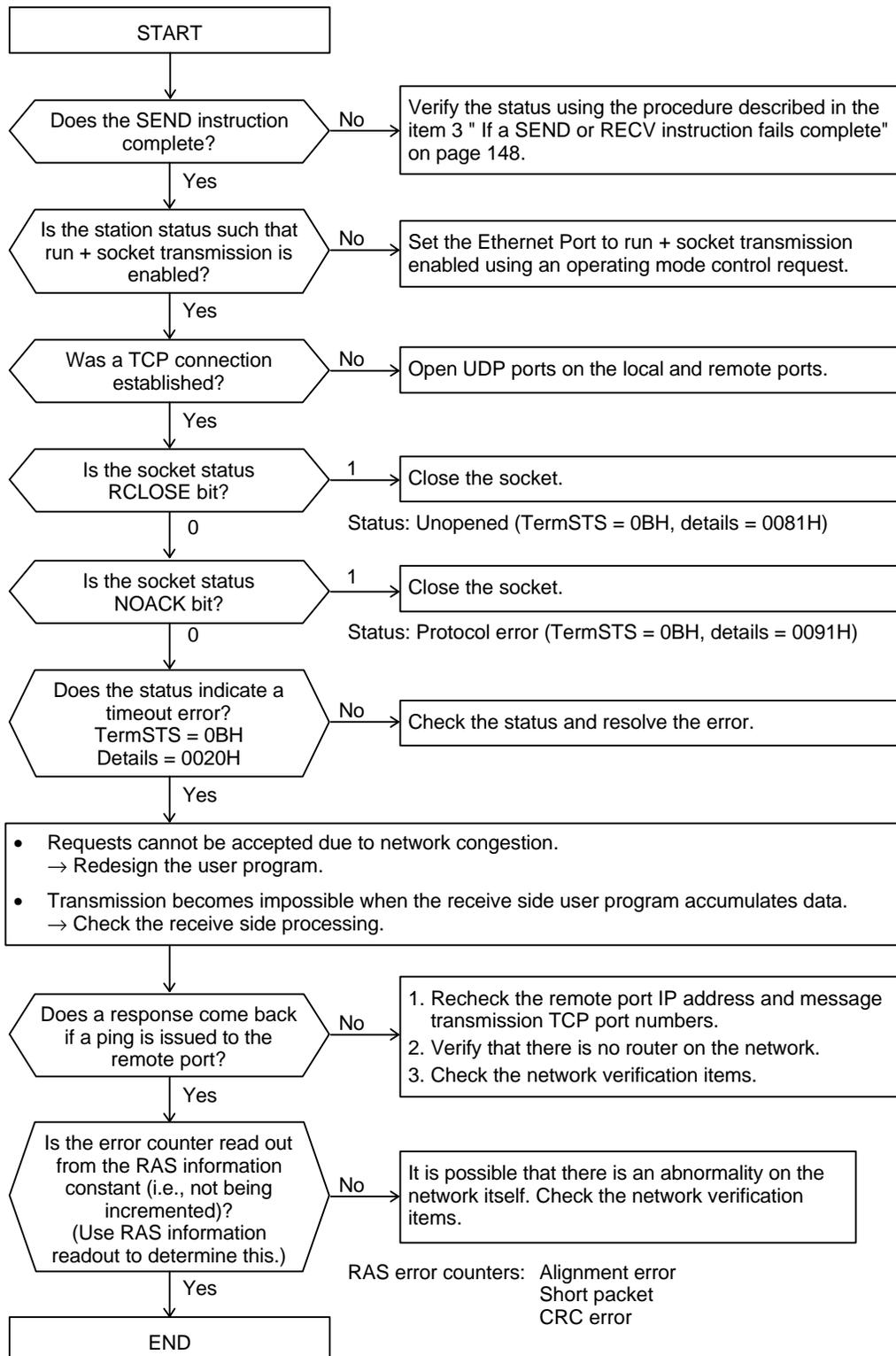
a. Open request



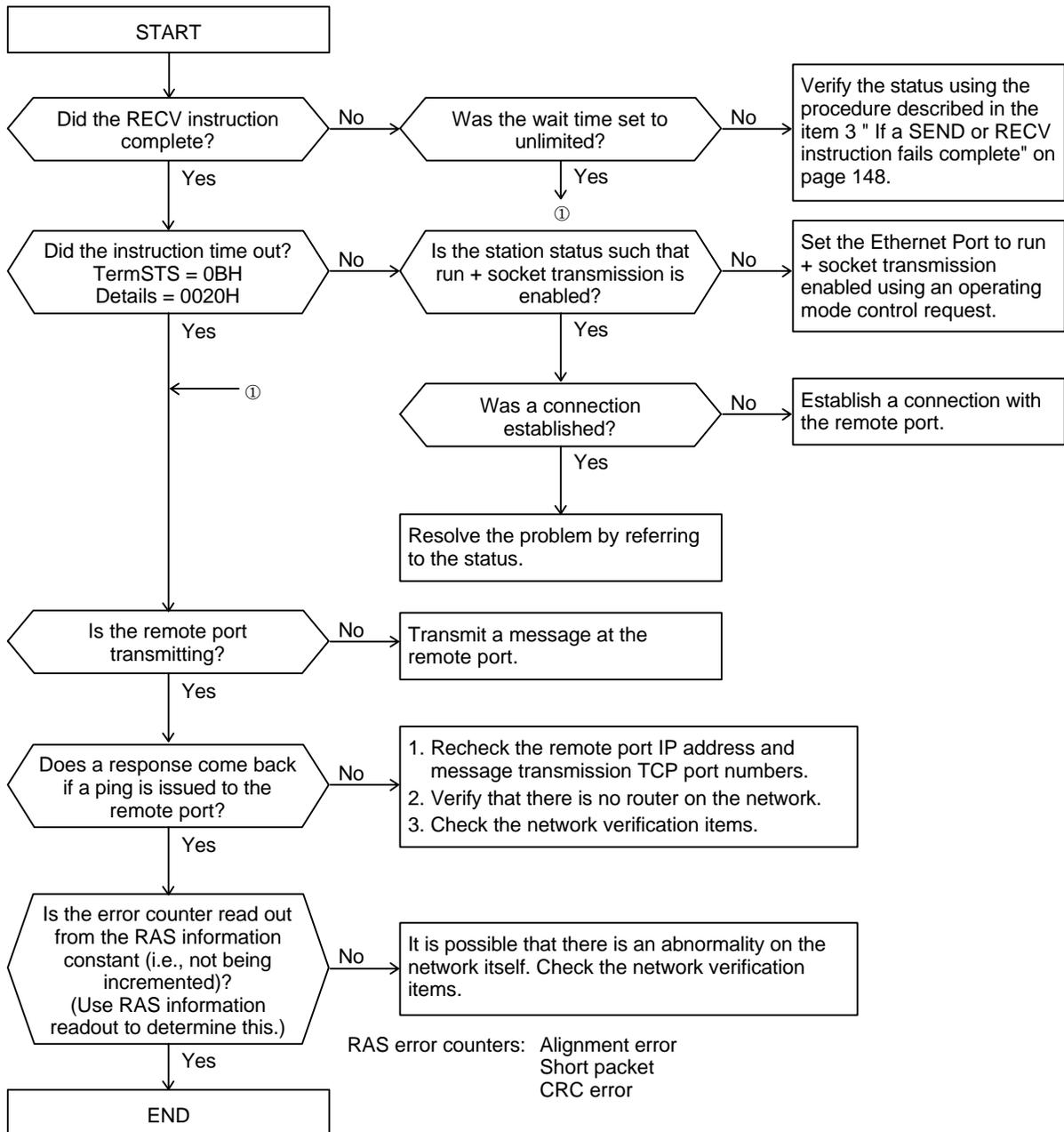
b. Close request



c. Send request



d. Receive request



## 9. Network check items (Ethernet Port)

	Check items	Countermeasure
Transmission medium installation 10BASE-T	1. Is the twisted pair cable connected between the Ethernet Port and HUB?	1. Verify the twisted pair cable connections between the Ethernet Port and HUB.
	2. How long (in meters) are the twisted pair cable (i.e. segment lengths)?	2. The maximum segment length for 10BASE-T cabling is 100 meters.
Network cable check	1. Do all ports on the network respond to ping correctly?	1. Inspect the twisted pair cables for missing termination resistors and damage.
	2. Is the collision frequently happen? Can you confirm the table indicating the collision?	2. Since network loading is heavy: <ul style="list-style-type: none"> <li>• Transmission efficiency can be degraded (TCP), and</li> <li>• There is an increased possibility of packet loss (UDP).</li> </ul> Therefore, the circuit usage conditions should be reviewed.
	3. Are the error counters read out from the RAS information increasing as time passes? Error counters: Alignment error, Short packet, CRC error	3. Check whether there are any defective (or substandard) connections on the cables and HUB. Check for the presence of noise sources in the vicinity of the network cables.

## 9. Installation and Wiring

### 9.1 Installation Environment and Base Unit Mounting

The PU235N/245N must be used in the installation environment described in section 4.1, "Installation Environment," of the T2N User's Manual - Hardware. Also, the PU235N/245N panel must be installed in conformance with the notes and cautionary advice presented in section 4.1, "Installation Environment."



#### CAUTION

1. Only use this equipment in the environment described in the T2N User's Manual - Hardware. Use in any other environment can cause electrical shocks, fires, failures, and malfunctions.
2. Mount this equipment according to the mounting method specified in the T2N User's Manual - Hardware.  
Mounting in any direction other than the specified direction, or defective or inadequate mounting, can result in the equipment falling, fires, failures, and malfunctions.

## 9.2 Network Wiring

See section 3.2, "Network Wiring Equipment," for details on the equipment that makes up the network. This section presents an outline of the wiring laid inside the control panel and outside the control panel. This section also presents individual notes on the twisted pair cables used, the coaxial cables, the AUI cables, and the MAUs.

Basic points in laying out a network

 <b>CAUTION</b>	
1.	When laying network cable, take adequate safety measures and perform all operations in accordance with the relevant standards. See the <b>ISO/IEC8802-3</b> standard for details on laying networks.
2.	We strongly recommend hiring a qualified contractor who has specialized knowledge about safety precautions and standards.
3.	Avoid installing 10BASE5, 10BASE2 or 10BASE-T networks in environments that have high noise levels. If this is unavoidable, adopt the noise prevention measure described below.

### 1. Overview of wiring outside the control panel

#### a. Layout environment and required work

Lay out cables in a manner appropriate for the environment. The steps listed in the following table are generally required.

**Table 9.1 Wiring Environment and Work Operations**

Environment in which cable is laid		Work operations
Major categories	Minor categories	
Locations where cables will not be damaged		Installation without protective pipes
		Installation with protective pipes
Location where cables could be damaged	Places where there is normally motion of people and objects	Installation with protective pipes
	Places where cables may be influenced by humidity, chemicals, oils, or heat	Installation with protective metal pipes
	Places where cables may be damaged by rats, mice, or other animals.	Installation with protective pipes
	Places where cables may receive strong mechanical shocks, or where heavy physical loads may be placed on the cables.	Installation with protective metal pipes
Locations where there is a likelihood of electromagnetic interference		Protective installation and separation of cables.
		Installation with protective metal pipes

The term "installation with protective pipes" means that either metal or PVC pipes may be used.

**b. Overview of wiring without protective pipes**

- (1) To protect the cables, cables should be passed through flooring ducts or wire protecting cable covers should be used.

Note that these cables should not be laid adjacent to cables carrying high currents.

- (2) Lay cables in gaps in the walls so that they will not be damaged, and run cables under the flooring under equipment so that electromagnetic interference will not occur.
- (3) Observe the applicable cable separation standards when running cables in parallel with, when crossing, or when approaching low-voltage indoor power lines.
- (4) Fasten cables every 3 meters so that the cables are not damaged when attaching cables to walls to position the away from the floor, or when running cables vertically.
- (5) When bending cables, be sure that the radius of curvature of the bend does not exceed the bending limit for the cable used.

**c. Notes on piped cabling**

- (1) Do not run coaxial cable and lines carrying high currents in the same pipe.
- (2) When bending pipes, do not bend pipes to an angle sharper than 90 degrees.
- (3) When bending pipes, the radius of curvature must be at least 6 times the internal diameter of the pipe, and must be no smaller than the minimum radius of curvature for bending of the enclosed cables.
- (4) Ground metal pipes.

**d. Separation from other wiring**

As a general principle, coaxial cable should be separated by at least 2 meters from power lines and equipment that generates electric or magnetic fields. If it is difficult to achieve a separation of 2 meters, look up the actual voltage and current of the induction source in the table below to determine the separation distance. However, since transmitted signals are weak signals, restrict induction sources to no more than 440 volts and 100 amperes.

**Table 9.2 Recommended Minimum Separation Distance**

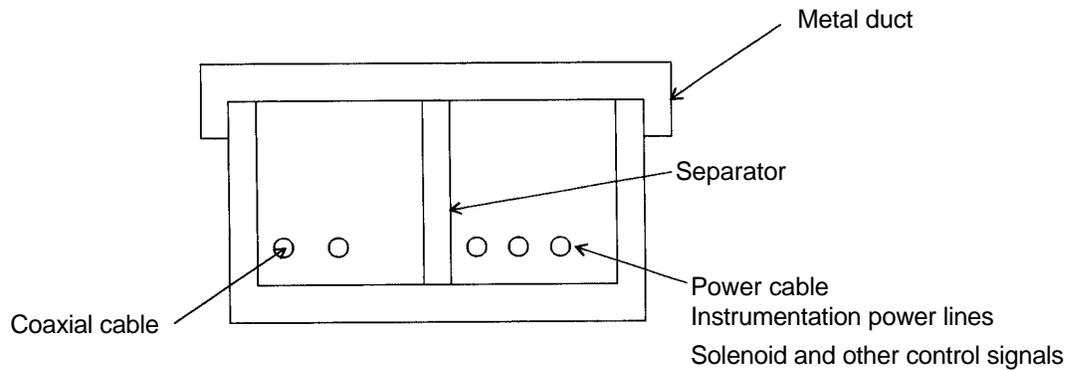
Induction source Voltage, current	Minimum separation for parallel lines (mm)			
	Over 100 A	Up to 100 A	Up to 50 A	Up to 10 A
Over 440 V	2000	2000	2000	2000
Up to 440 V	2000	600	600	600
Up to 220 V	2000	600	600	500
Up to 110 V	2000	600	500	300
Up to 60 V	2000	500	300	150

Considering resistance to noise, we recommend using either covered metal cable ducts or protective steel pipe. Table 9.3 lists the recommended minimum separations for parallel lines in this case.

**Table 9.3 Recommended Minimum Separation Distance (Covered metal ducts and metal pipe) (Units: mm)**

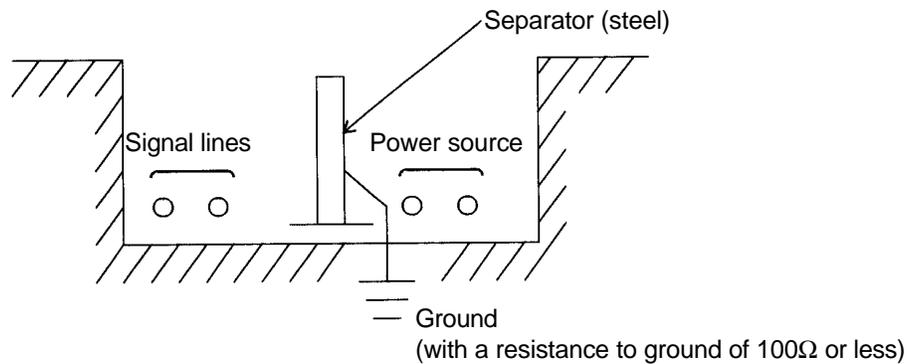
Cable installation method		Covered metal ducts or metal pipe					
Parallel separation		Up to 10 meters	Up to 25 meters	Up to 100 meters	Up to 200 meters	Up to 500 meters	501 meters or over
Power cable	Up to 125 V, up to 10 A	10	10	50	100	200	250
	Up to 250 V, up to 50 A	10	50	150	200	250	250
	Up to 400 V, up to 100 A	50	100	200	250	250	250
	Up to 500 V, up to 200 A	100	200	250	250	250	250
Levels exceeding the above		At least 500					

When using cable ducts, install steel separators to separate power cables from coaxial cables as shown in the figure below.



**Figure 9.1 Duct Internal Layout Example**

When laying cables in a pit, use separators in the same manner as when ducts are used.



**Figure 9.2 Pit Internal Layout Example**

e. **Wiring routes**

The desirable routes for laying the transmission cable can be ordered as follows.

- (1) Using a dedicated route
- (2) Using a route dedicated to computer system related lines
- (3) Using a route dedicated to general measurement equipment
- (4) Using a route dedicated to plant control related lines

2. **Overview of panel internal wiring**

When 10BASE-T is used with the Ethernet Port, twisted pair cable is connected to the Ethernet Port. Keep the following points in mind when wiring inside the control panel.

a. **Wiring path**

Use the weak-signal route within the panel for wiring the coaxial cable.

In particular, keep the coaxial cable at least 50 cm away from power supply cables.

b. **Securing coaxial cable**

Secure the coaxial cable with clamp materials so that the weight of the cable is not applied to the cable's connectors.

Remember that the cable not must be bent beyond its minimum bending radius.

3. **Notes on laying and wiring twisted pair cable (10BASE-T)**

- a. Only use this twisted pair cable in boards. Consult with a specialist contractor if twisted pair cable must be routed out of boards.

4. **Notes on laying and wiring 10BASE2 and 10BASE5 coaxial cable**

- a. Only use this coaxial cable indoors. Consult with a specialist contractor if coaxial cable must be routed outdoors.
- b. Connect terminators to both ends of the coaxial cable.
- c. Insulate all exposed metal parts other than the coaxial cable installation point from ground and other metal parts.

10BASE5: Plugs, adapters, terminators

10BASE2: BNC plugs, adapters, terminators, T connectors.

- d. For safety, ground, at a single point, the coaxial cable's external conductor to a point with a resistance to ground of 100  $\Omega$  or less.

Also, prepare a dedicated network system ground that is not shared with other equipment.

- e. Contact the manufacturer of the cable with regard to the physical characteristics value of the coaxial cable allowable minimum bending radius.
- f. Avoid locations with high temperature, high humidity, dust, or oil mist when installing coaxial cable.

**5. Notes on laying and wiring AUI cable**

- a. When connecting the AUI cable to the MAU connectors, use the slide latches to secure the connectors firmly.
- b. Contact the manufacturer of the cable with regard to the physical characteristics value of the AUI cable allowable minimum bending radius.
- c. Avoid locations with high temperature, high humidity, dust, or oil mist when installing AUI cable.

**6. Notes on MAU (10BASE5) installation**

- a. Attach MAUs at the 2.5-meter separated marks on the coaxial cable.
- b. Since the procedure for attaching MAUs to coaxial cable differs between manufacturer, refer to the MAU user's manual.
- c. Since the coaxial cable is grounded, the MAU main unit should be mounted insulated from ground.
- d. Do not install MAUs in the vicinity of equipment that generates electrical noise. If a MAU must be installed in such a location, provide a MAU enclosure box, and mount the MAU inside the box insulated from that box. Be sure to ground the MAU enclosure box.
- e. Avoid locations with high temperature, high humidity, dust, or oil mist when installing MAU cable.

**Appendix 1 Maintenance and Inspection**

**Periodic Inspection**

Check the following items periodically, once every six months. Also, check these points if any surrounding conditions or any aspect of the environmental changes.

**Table A.1 Periodic Inspection Items**

Inspection item	Inspection	Criterion	Resolution procedure
Power supply systems (T2N power supply/MAU power supply)	Measure the power-supply voltage at the power-supply terminals.	Voltages must be within the stipulated ranges.	Adjust the input voltages to be within the stipulated ranges.
	Check for loose screws at the power-supply terminals.	There must be no looseness.	Turn off (remove power from) the power supply and tighten down the screws with a screwdriver.
	Check for damage to the power-supply cables.	There must be no damage.	Turn off (remove power from) the power supply and replace the damaged cables.
Installation state	Is the module firmly locked in place?	There must be no looseness or play.	First press the module.
Transmission cable connections	10BASE5: Are the AUI cables firmly locked in place?	There must be no looseness or play. The 10BASE5 connector slide latches must be in their locked positions.	First press the cable firmly into the connector and then slide the slide latch to the locked position.
	10BASE2: Are the coaxial cables firmly locked in place?	There must be no looseness or play. The T-connector locks must be engaged.	First press the T-connector firmly into position and then engage the T-connector lock.
	10BASE-T: Are the twisted pair cables firmly locked in place?	There must be no looseness or play. The modular connector locks must be engaged.	First press the twisted pair cable into modular connector.
Transmission cable wiring (See section 9.4, "Network Wiring".)	10BASE5/2-T: Check for damage to the cables.	There must be no damage.	Stop the system, turn off (remove power from) the power supply and replace the damaged cables.
	Are the cables installed near any power or other cables?	There must be no power or other similar cables in the vicinity.	Separate the cables from the power lines. Shield the cables.
Ambient conditions	Verify that temperature, humidity, vibration, and dust levels are all within the stipulated ranges.	The values must be within the ranges from the general specifications for the module.	Improve the environment so that all aspects are within the ranges stipulated in the specifications.

**CAUTION**

1. Always turn off the power before attaching or removing modules, the terminal block, or cables.  
Electrical shock, malfunctions, and breakdowns are possible if these operations are performed with the power on.
2. To keep the system operating normally at all times and to prevent trouble in advance, perform daily inspections, periodic inspections, and cleaning.
3. Refer to the section on error recovery in this document if the Ethernet Port does not operate correctly.  
  
Contact your local Toshiba dealer or service representative if a breakdown occurs, and request that the module be repaired or returned to Toshiba.  
  
Operation and safety cannot be guaranteed if service is performed by anyone other than an authorized Toshiba service representative.
4. Do not disassemble or modify the module hardware, and do not modify the module software.  
Breakdowns or malfunctions may lead to fire, electrical shock, or injury.
5. When measuring the voltages at the module terminal block during inspections, use adequate caution.  
  
There is a danger of electrical shock during this operation.
6. Only exchange modules with the power completely off.  
Exchanging modules with the power on can lead to malfunctions, breakdowns, and electrical shock.
7. Do not continue to use the module if it emits smoke or unusual odors, or if it is in an abnormal state of any type.  
  
Using the module in an abnormal state can lead to malfunctions, breakdowns, and electrical shock.  
  
In such cases, immediately turn off all power and contact your local Toshiba dealer or service representative.  
  
Customers should never attempt repairs or modifications to this equipment themselves: repair and modification operations are extremely dangerous.

**Appendix 2 Request Code Table**

This table lists the request codes (CMD) used with SEND/RECV instructions sent to the Ethernet Port. Refer to the section/item indicated in the reference column in the table for detailed explanations of the request codes.

**Table A.2 Instructions (Requests) to the Ethernet Port**

Request	Function	Instruction	CMD/Class	Reference
Reset request	Resets the Ethernet Port.	SEND	0011H/(1)	Section 4.3
Parameter setup request	Sets up the Ethernet Port parameters.	SEND	0012H/(1)	Section 4.4
Operating mode control request	Sets the operating mode.	SEND	0013H/(1)	Section 4.5
Remote port verification request	Verifies the existence of another port on the network.	SEND	0014H/(1)	Section 7.3
RAS information readout request	Reads out the Ethernet Port RAS information.	RECV	0015H/(1)	Section 7.6
Time set request	Sets the Ethernet Port internal clock Time information for event trace	SEND	0018H/(1)	Section 7.5
Register read request (PC link function)	Reads register data from remote T2N into registers in the local T2N.	RECV	0021H/(2)	Section 5.4
Register write request (PC link function)	Writes register data from the local T2N to registers in remote T2N.	SEND	0021H/(2)	Section 5.3
Remote port loopback request	Performs a loopback test with another Ethernet Port on the network.	SEND	000FH/(2)	Section 7.4
UDP open request	Opens a UDP socket.	SEND	0031H/(3)	Section 6.4 (1)
UDP send request	Transmits data from an open UDP socket.	SEND	0032H/(3)	Section 6.4 (2)
UDP receive request	Reads out data received by an open UDP socket.	RECV	0033H/(3)	Section 6.4 (3)
UDP close request	Closes (terminates) an open UDP socket.	SEND	0034H/(3)	Section 6.4 (4)
TCP open request	Opens a TCP socket.	SEND	0035H/(3)	Section 6.4 (5)
TCP send request	Transmits data from an open TCP socket.	SEND	0037H/(3)	Section 6.4 (6)
TCP receive request	Reads out data received by an open TCP socket.	RECV	0038H/(3)	Section 6.4 (7)
TCP close request	Closes (terminates) an open TCP socket.	SEND	0039H/(3)	Section 6.4 (8)

- The request code (CMD) types (1) to (3) have the following meanings.  
Class (1) is module control, (2) is PC link protocol transmission, and (3) is socket interface transmission.
- For a single Ethernet Port, it is not possible to request another class (1) CMD while a class (1) CMD is executing. This is also true for class (2) CMDs. If such a request is issued, the status will be transmission port busy, and the request will go to the wait state.
- For a single Ethernet Port, when requesting a class (3) CMD, it is not possible to request another class (3) command for a socket for which an instruction is executing. If such a request is issued, the status will be transmission port busy, and the request will go to the wait state.
- There are eight sockets, and each can independently accept and execute a class (3) instruction.

Appendix 3 Status Table

Figure a.1 lists the completion status and detailed information stored in the specified registers during SEND/RCV instruction execution or at instruction completion. Tables A.3 and A.4 list the contents of the TermSTS completion status, and table A.5 lists the contents of the detailed information (the Ethernet Port error response) loaded into the next register after the register used for the completion status.

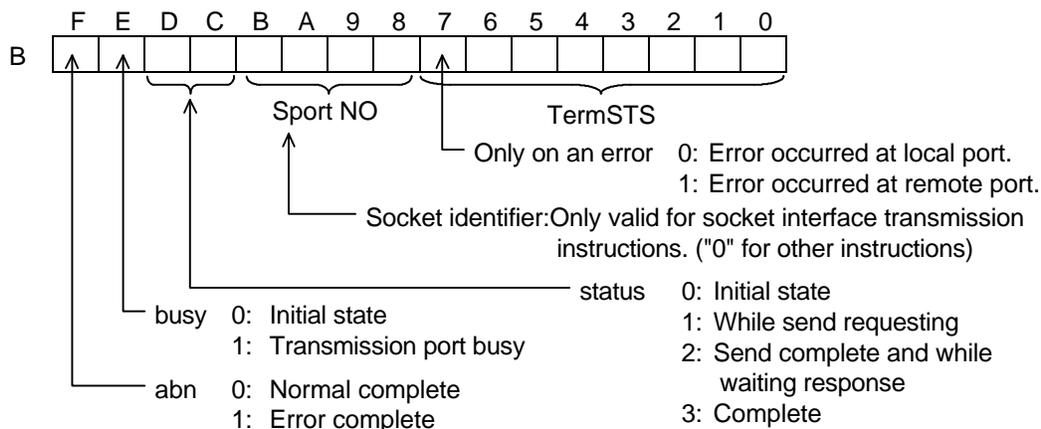


Figure A.1 Status Structure

Table A.3 TermSTS (Part 1)

TermSTS	Status	Meaning
00H	Normal complete	
01H	Register specification error	<ul style="list-style-type: none"> <li>Occurs if a register other than XW, YW, W, LW, RW, D, F, T, C, or SW is specified as the data storage register in one of the following requests. Requests: RAS information readout, remote port loopback, UDP send/receive, TCP send/receive, register read/write.</li> <li>In a register read or write request, the request specified a read or write from a T register to a register other than a T register. (This also applies to C registers.)</li> </ul>
02H	Response timeout	<ul style="list-style-type: none"> <li>Occurs if a response is not received within the specified time for a remote port loopback or register read/write request.</li> </ul>
03H	Parameter error	<ul style="list-style-type: none"> <li>Occurs if a value other than 3 is specified for the module ID.</li> <li>Occurs if a value other than 1, 2, 3, or 4 is specified for the channel number.</li> <li>Occurs if a command number other than one of the following is specified for a SEND instruction. 000FH, 0011H - 0014H, 0018H, 0021H, 0031H, 0032H, 0034H, 0035H, 0037H, 0039H</li> <li>Occurs if a command number other than one of the following is specified for a RCV instruction. 0015H, 0021H, 0033H, 0038H</li> <li>Occurs if a value other than 1 to 8 is specified for the socket identifier. (See section 6.4 "Using the Ethernet Port Socket Interface".)</li> <li>Occurs if an incorrect parameter is specified in a RAS information readout request. (See section 7.6 "RAS Information Readout".)</li> </ul>

Table A.4 TermSTS (Part 2)

TermSTS	Status	Meaning
04H	Memory write protect	<ul style="list-style-type: none"> <li>Occurs if an attempt to write is made when the remote registers were write protected.</li> <li>Occurs if an attempt to read is made when the local registers were write protected.</li> </ul>
05H	(Reserved)	
06H	Module error	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port is down.</li> </ul>
	Initialization in progress	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port is initializing.</li> </ul>
	Transmission complete timeout	<ul style="list-style-type: none"> <li>Occurs if a T2N internal timer times out.</li> </ul> Requests: UDP open, UDP send, UDP close, TCP send, TCP close
07H	No send channel (CH)	<ul style="list-style-type: none"> <li>Occurs if the corresponding Ethernet Port is not installed.</li> </ul>
08H	Port address out of range	<ul style="list-style-type: none"> <li>Occurs if the local port IP address was specified for the remote destination IP address.</li> </ul> Requests: Remote port loopback, register read/write
09H	Incorrect transmission word count	<ul style="list-style-type: none"> <li>Occurs if either 0 words or 486 or more words were specified for either a register read/write or a remote port loopback request.</li> <li>Occurs if either 0 words or 1001 or more words were specified for a UDC/TCP send/receive request.</li> </ul>
0AH	Boundary error	<ul style="list-style-type: none"> <li>Occurs if the specified area (start address + register range) does not exist in the T2N data storage registers.</li> </ul> Requests: RAS information readout, remote port loopback, UDP send/receive, TCP send/receive, register read/write
0BH	Transmission error	<ul style="list-style-type: none"> <li>Occurs if the Ethernet Port returns an error response.</li> </ul> Detailed information (the Ethernet Port response status) is stored at B+1.
0CH	No I/O response	<ul style="list-style-type: none"> <li>Occurs if the T2N cannot access the Ethernet Port.</li> </ul>
0DH	(Reserved)	
0EH	Send data capacity exceeded	<ul style="list-style-type: none"> <li>Occurs if a request is discarded when a T2N internal resource insufficiency occurs due to increasing amounts of transfer data.</li> <li>Occurs if a T2N retransmission times out.</li> </ul>
0FH	(Reserved)	

Details on the Ethernet Port error response when a transmission error (0Bh) occurs are stored in the register following the status. (See table A5.)

**Table A.5 Detailed Information (Ethernet Port Error Responses)**

Error	Code	Description
Normal complete	0001H	Request completed normally
Local port fault	0002H	The local port is in down mode
Local port standby	0003H	Data transmission in standby mode (socket interface transmission or PC link protocol transmission) was requested.
Timeout	0020H	Timeout occurred during TCP open processing or receive processing, or during UDP receive processing.
Length error	0030H	Incorrect send data length
Port mode error	0040H	One of the following requests was issued in a mode other than standby mode: a. Parameter setup, b. MAC address setup, c. Reset
MAC/IP not set up	0050H	Occurs if the MAC address or IP address is not set up.
Transmission prohibited state	0060H	<ul style="list-style-type: none"> <li>Occurs if a PC link request was issued from the T2N in run mode in the message transmission prohibited state.</li> <li>Occurs if a socket open, close, send, or receive request is issued by the T2N in run mode in the socket interface transmission prohibited state.</li> </ul>
Format error	0070H	Occurs when there is a format error in the request text.
	0071H	Port number error
	0072H	Request code error
	0073H	Time setting: Year
	0074H	Time setting: Month
	0075H	Time setting: Day
	0076H	Time setting: Hour
	0077H	Time setting: Minute
	0078H	Time setting: Second
Opened	0080H	Occurs if an open request is issued for an already open socket.
Unopened	0081H	Occurs if send, receive, or close request is issued for an unopened socket.
Incorrect socket identifier	0082H	Occurs if the socket identifier is out of range (1 to 8).
Incorrect control request specification	0083H	Occurs if a request that was prohibited by an operating mode control request is issued. <ul style="list-style-type: none"> <li>Occurs if a run request is issued in run mode.</li> <li>Occurs if a standby request is issued in standby mode.</li> <li>Occurs if a run request and a standby request are issued at the same time.</li> </ul>
LAN controller driver error	0090H	Occurs if the setting of the local port IP address and port number in the area reserved for transmission fails.
	0091H	Error in the TCP or UDP protocol (such as a transmission phase error)
	0092H	<ul style="list-style-type: none"> <li>Occurs if the setting of the remote port IP address in the remote port information area (the NETDATA table) fails.</li> <li>Occurs if the remote port IP address network address differs from the local port IP address network address.</li> </ul>
Memory pool allocation failure	00A0H	Occurs if the Ethernet Port internal OS fails to allocate memory.
Port 2 task start failure	00A1H	Occurs if computer link or PC link socket identifier acquisition fails.
Broadcast specification error	00A2H	If the remote destination was specified to be broadcast in a remote port verification request.
MAC address specification error	00B0H	BCC error
	00B1H	Occurs if an error occurs after address saving on a MAC setup request.
Insufficient resources error	00C0H	Occurs if there are insufficient resources for UDP send or PC link transmission.

## Appendix 4 Limitations on the T2N SEND and RECV Instructions

When transmissions using the Ethernet Port occur with high frequency, the T2N internal processing load increases and it is possible for this to interfere with the T2N periodic interrupts and other internal operations. For example, the T2N processing load increase greatly when requests from other ports and requests from the local port occur at the same time in a single Ethernet Port. Therefore, the Ethernet Port must be used with the following limitations enforced.

Usage Recommendations
When using the SEND and RECV instructions, there are limitations on the interrupt periods that can be set the fixed period interrupt programs.

### (1) When using the PC link protocol with the Ethernet Port

The number of words that can be set for the SEND and RECV instructions is stipulated to be either:

- The **number of words that the local port transmitted** with the SEND instruction.
- or:
- The **number of words that the local T2N transmitted** in response to a data request issued with the RECV instruction from a remote T2N.

When some number of words are transmitted by the local T2N with the SEND instruction and some other number of words are transmitted due to a RECV instruction request from a remote, we focus on the larger of the two word counts when transmission is through the same transmission Ethernet Port.

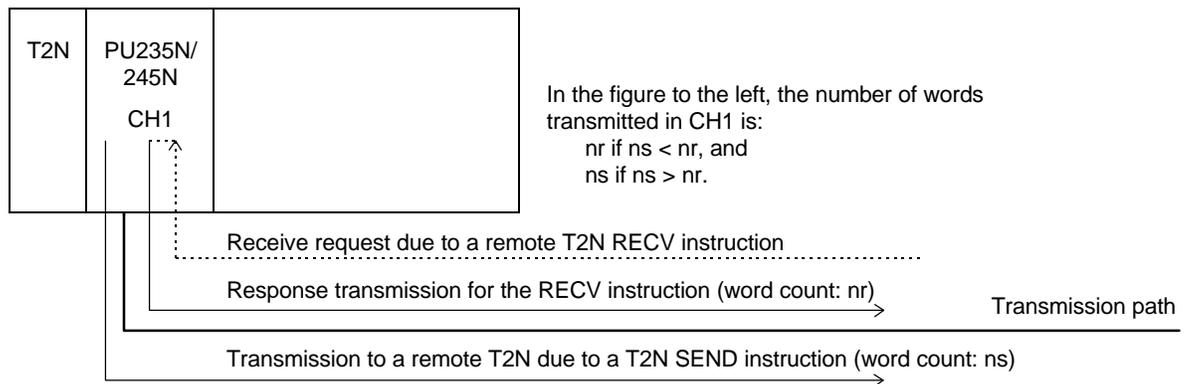


Figure A.2 Word Counts Considered with the PC Link Protocol

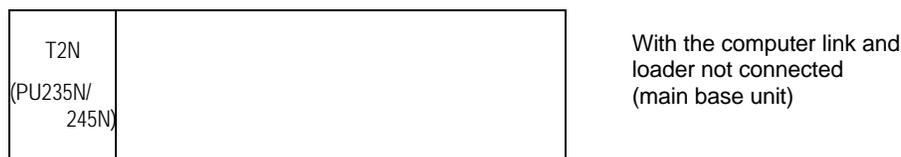
**Table A.6 Number of Words that can be Transmitted Using PC Link Protocol with the Ethernet Port**

[When TCP/UDP socket transmission is not used]

Module classification	Processing time for function instructions	Maximum specifiable number of transmission words for local T2N SEND instructions or remote T2N RECV instructions
PU235N	4 ms (maximum)	485 words (maximum)
	5 ms (maximum)	295 words (maximum)
PU245N	3 ms (maximum)	485 words (maximum)
	4 ms (maximum)	346 words (maximum)
	5 ms (maximum)	131 words (maximum)

Conditions for the above table:

- a. For the T2N configuration, this table assumes the configuration of figure A.3, in which T2N (PU235N/245N) is mounted in the main base unit.
- b. Refer to the T2N User's Manual - Hardware for the processing times for the instructions.
- c. This table assumes that for PU245N, 128 words is specified for S20LP PC link protocol transmission (SEND instruction on local port or RECV instruction from remote ports). However, if S20LP PC link protocol transmission is not used, use the PU235N row in the above table, even if there are PU245N unit installed.
- d. If less than 128 words are used with the S20LP PC link protocol transmission, the number of words that can be specified during PC link transmission on the Ethernet Port can be increased over the values listed in table a.6. If the number of words for S20LP PC link protocol transmission on the PU245N decreased on one word, the number of words for PC link protocol transmission on the Ethernet Port can increase in one word. However, the maximum number of words that can be specified during PC link transmission on the Ethernet Port is 485 words.



**Figure A.3 T2N Configuration when Using the PC Link Protocol**

[Example] In case of the transmission with 1024 words in the TMOV instruction is executed.

$$\begin{aligned} \text{Execution time required} &= 206 + 3.5 * n \text{ (the number of transmission words : 1024)} \\ &= 3790 \end{aligned}$$

In this case, the execution time required is under 4ms. Therefore, the maximum specifiable number of transmission words are 485 words for PU235N, 346 words for PU245N.

[Attention] In case of using the instruction below, be careful to use the maximum specifiable number of transmission. In the case of using another instruction, be careful, too.

TMOV, TNOT, TCMP, XFER, READ, WRITE

(2) When using socket transmission (TCP or UDP) with the Ethernet Port

**Table A.7 Number of Words that can be Transmitted Using TCP/UDP Socket Transmission with the Ethernet Port [When the PC link protocol is not used]**

Module classification	Processing time for function instructions	Maximum number of words that can be specified in socket transmission.
PU235N	1 ms (maximum)	1000 words (maximum)
	2 ms (maximum)	927 words (maximum)
	3 ms (maximum)	712 words (maximum)
	4 ms (maximum)	497 words (maximum)
	5 ms (maximum)	282 words (maximum)
PU245N	1 ms (maximum)	978 words (maximum)
	2 ms (maximum)	763 words (maximum)
	3 ms (maximum)	548 words (maximum)
	4 ms (maximum)	333 words (maximum)
	5 ms (maximum)	118 words (maximum)

**Conditions for the above table:**

- a. For the T2N configuration, this table assumes the configuration of figure A.4, in which T2N (PU235N/245N) is mounted in the main base unit.
- b. Refer to the T2N User's Manual - Hardware for the processing times for the instructions.
- c. This table assumes that for PU245N, 128 words is specified for S20LP PC link protocol transmission (SEND instruction on local port or RECV instruction from remote ports). However, if S20LP PC link protocol transmission is not used, use the PU235N row in the above table, even if there are PU245N unit installed.
- d. If less than 128 words are used with the S20LP PC link protocol transmission, the number of words that can be specified during socket transmission on the Ethernet Port can be increased over the values listed in table A.7. If the number of words for S20LP PC link protocol transmission on the PU245N decreased on one word, the number of words for Socket transmission on the Ethernet Port can increase in one word. However, the maximum number of words that can be specified during PC link transmission on the Ethernet Port is 1000 words.



With the computer link and loader not connected (main base unit)

**Figure A.4 T2N Configuration when Using the Socket Interface**

- (3) When using the PC link protocol and socket transmission (TCP or UDP) with the Ethernet Port.

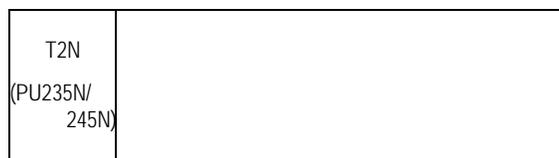
**Table A.8 Number of Words that can be Transmitted Using PC Link Protocol and TCP/UDP Socket Transmission with the Ethernet Port with Ethernet Port.**

**[When PC link transmission and TCP/UDP socket transmission is used either]**

Module classification	Processing time for function instructions	Maximum specifiable number of transmission words for local T2N SEND instructions, remote T2N RECV instructions or socket transmission
PU235N	1 ms (maximum)	1073 words (maximum)
	2 ms (maximum)	858 words (maximum)
	3 ms (maximum)	643 words (maximum)
	4 ms (maximum)	428 words (maximum)
	5 ms (maximum)	213 words (maximum)
PU245N	1 ms (maximum)	1038 words (maximum)
	2 ms (maximum)	823 words (maximum)
	3 ms (maximum)	607 words (maximum)
	4 ms (maximum)	392 words (maximum)
	5 ms (maximum)	177 words (maximum)

Conditions for the above table:

- a. For the T2N configuration, this table assumes the configuration of figure A.5, in which T2N (PU235N/245N) is mounted in the main base unit.
- b. Refer to the T2N User's Manual - Hardware for the processing times for the instructions.
- c. This table assumes that for PU245N, 128 words is specified for S20LP PC link protocol transmission (SEND instruction on local port or RECV instruction from remote ports). However, if S20LP PC link protocol transmission is not used, use the PU235N row in the above table, even if there are PU245N unit installed.
- d. The number of words is the maximum number words that can be specified during PC link transmission and socket transmission. The maximum number of words that can be specified during PC link transmission on Ethernet Port is 485 words, and during socket transmission on Ethernet Port is 1000 words.
- e. If less than 128 words are used with the S20LP PC link protocol transmission, the number of words that can be specified during PC link transmission on the Ethernet Port can be increased over the values listed in table A.8. If the number of words for S20LP PC link protocol transmission on the PU245N decreased on one word, the number of words for PC link protocol transmission or Socket transmission on the Ethernet Port can increase in one word. However, the maximum number of words that can be specified during PC link transmission on the Ethernet Port is 485 words, and during socket transmission on the Ethernet Port is 1000 words.
- f. If the PU245N is used only Ethernet or S20LP PC link transmission, refer to the value listed in table A.6 and if the PU245N is used only socket transmission on the Ethernet Port, refer to the value listed table A.7.



With the computer link and loader not connected (main base unit)

**Figure A.5 T2N Configuration when Using the PC Link Protocol**