HITACHI

USER'S MANUAL

OPTION

RS-232C/422

（LQE560/565）

S10mini S10V

SVE-1-121 (F)

USER'S MANUAL

OPTION

RS-232C/422

（LQE560/565）

S10mini S10V

First Edition, February 2003, SVE-1-121(B) (out of print)
Second Edition, January 2007, SVE-1-121(C) (out of print)
Third Edition, October 2008, SVE-1-121(D) (out of print)
Fourth Edition, April 2010, SVE-1-121(E) (out of print)
Fifth Edition, February 2013, SVE-1-121(F)

Printed in Japan.

BI<IC> (FL-MW2007, AI10)

SAFETY PRECAUTIONS

- Before installation, operation, maintenance, and/or inspection of this product, be sure to read through carefully this manual and other related manuals. Do not use this product until you are familiar with all the information concerning this product, safety information, and precautions provided in those manuals.
- Keep this manual in a readily accessible place so that users of this product may easily reach it.
- This manual contains information on potential hazards that is intended as a guide for safe use of this product. The potential hazards listed in the manual are divided into four hazard levels of danger, warning, caution, and notice, according to the level of their severity. The following are definitions of the safety labels containing the corresponding signal words DANGER, WARNING, CAUTION, and NOTICE.

DANGER : This safety label identifies precautions that, if not heeded, will result in death or serious injury.

WARNING : Identifies precautions that, if not heeded, could result in death or serious injury.

CAUTION : Identifies precautions that, if not heeded, could result in minor or moderate injury.

NOTICE : This safety label without a safety alert symbol identifies precautions that, if not heeded, could result in property damage or loss not related to personal injury.

Failure to observe any of the CAUTION and *NOTICE* statements used in this manual could also lead to a serious consequence, depending on the situation in which this product is used. Therefore, be sure to observe all of those statements without fail.

The following are definitions of the phrases “serious injury,” “minor or moderate injury,” and “property damage or loss not related to personal injury” used in the above definitions of the safety labels.

Serious injury: Is an injury that requires hospitalization for medical treatment, has aftereffects, and/or requires long-term follow-up care. Examples of serious injuries are as follows: vision loss, burn (caused by dry heat or extreme cold), electric-shock injury, broken bone, poisoning, etc.

Minor or moderate injury: Is an injury that does not require either hospitalization for medical treatment or long-term follow-up care. Examples of minor or moderate injuries are as follows: burn, electric-shock injury, etc.

Property damage or loss not related to personal injury: Is a damage to or loss of personal property. Examples of property damages or losses not related to personal injury are as follows: damage to this product or other equipment or their breakdown, loss of useful data, etc.

The safety precautions stated in this manual are based on the general rules of safety applicable to this product. These safety precautions are a necessary complement to the various safety measures included in this product. Although they have been planned carefully, the safety precautions posted on this product and in the manual do not cover every possible hazard. Common sense and caution must be used when operating this product. For safe operation and maintenance of this product, establish your own safety rules and regulations according to your unique needs. A variety of industry standards are available to establish such safety rules and regulations.

1. ⚠ General Safety Guidelines

Before installing, operating inspecting or conducting maintenance on this unit, read the following instructions carefully:

- Follow all the operating procedures provided in this manual.
- Pay special attention to and follow all the hazard warnings on the machine and in the manual. Failure to do so can cause injury to yourself or damage to the machine.
- Do not perform any operation or action in any way other than as provided in this manual. When in doubt, call the designated field engineer. Keep in mind that the hazard warnings in this manual or on the machine cannot cover every possible case, as it is impossible to predict and evaluate all circumstances beforehand.
  Be alert and use your common sense.
- Do not install, wire, handle, modify, or use maintenance parts in any manner not described in this manual. Such a practice may result in breakdown of this equipment or peripherals, injury or even death. Hitachi will not be responsible for any accident or failure resulting from such mishandling.

Read the following safety guidelines carefully and follow them when you conduct maintenance of the machine.

Before starting maintenance

- Maintenance of the machine must be done only by trained and qualified field engineers.
- Read and follow the safety guidelines and procedures in this manual and the related manuals.
- In this manual and on the machine, hazard warnings are provided to aid you in preventing or reducing the risk of death, personal injury, or product damage.
  Understand and follow these hazard warnings fully.
- Keep in mind that the hazard warnings in this manual or on the machine cannot cover every possible case, as it is impossible to predict and evaluate all circumstances beforehand.
  Be alert and use your common sense.

During work

- For each procedure, follow the given sequence of steps.
- Use the special tools and instruments, specified for the work in the manual or commercially available tools and instruments which fit the purpose.
- Use measurement instruments and powered tools which are properly calibrated or periodically inspected.
- Keep the maintenance area neat and tidy.
- Always put away parts, materials or tools when not in use.
- Wear an eye protector where anything may fly about.
- When using sharp objects or cutting tools, make sure that no part of your body lies in the path of the blade bit, or point.
- Before finishing your work, make sure that all parts removed during maintenance have been installed back in their original positions in the machine.
  Make sure that no tool or foreign material is left in the machine.

Prevention of electric shocks

- Before starting work, make sure that, unless otherwise specifically instructed, there is no potential electric hazard in the maintenance area such as insufficient grounding or a wet floor.
- Before starting work, note where the emergency power-off switches are located and make sure you know how to operate them.
- Unless otherwise specifically instructed, cut off all power sources to the machine before starting maintenance. Just switching off the machine power supplies is usually not enough.
  When power is fed from a wall or floor outlet, unplug the power supply cord, or turn off the switch on the power distribution panel or board. Attach a notice on the panel or board prohibiting the use of the switch.
  If the energy isolating device such as the switch on the power distribution panel or board accepts a lockout device, turn off the power, lock out the energy isolating device, and bring the key with you. When you take over the work and the key for the lockout device if applicable, do not assume that the power is off. Make sure yourself that the above-mentioned conditions such as switches are satisfied. If necessary, use a measurement tool to ensure that the power is off.
- Do not touch any uninsulated conductor or surface, where so instructed, which remains charged for a limited time after the external power supply to the machine is disconnected.
- When working on a machine which has a grounding terminal, make sure that the terminal is properly connected to the facility's ground.
- When working close to a hazardously energized part, do not work alone; work with another person who can immediately turn off the power in an emergency.
- Do not wear any metallic item such as a wrist watch with a metallic surface, or metallic accessories.
  If you wear eyeglasses with a metallic frame, take care not to let the frame touch an uninsulated surface.
- Make sure that your hands and arms are dry.
- Unless otherwise specifically instructed, use only one hand when it is necessary to work near an exposed live electric circuit.
  This prevents the completion of the circuit through your heart even if you accidentally touch the circuit.
- Do not use a dental mirror near an exposed live electric circuit.
  The mirror surface is conductive and can become hazardous even if it is made of plastic.
- Unless otherwise specifically instructed, do not supply power to any subassembly such as a power supply unit or a motor while it is removed from the machine.

Procedures in an emergency

For electric shock

- Do not panic. Do not become another victim through contact with the injured person.
- First, shut off the electric current passing through the victim.
  Use the emergency power-off switch, if there is one, or, otherwise, a normal power-off switch. If this cannot be done, push the victim away from the source of the electric current by using a nonconductive object such as a dry wooden stick.
- Then, call an ambulance.
- If the victim is unconscious, artificial respiration may be necessary.
  A proper method for performing artificial respiration or resuscitation should be learned beforehand. If the victim's heart is not beating, cardio-pulmonary resuscitation should be performed by a trained and qualified person.

For outbreak of fire

- First, shut off all the power from the machine using the emergency power-off switch, if there is one, or the normal power-off switch.
- If the fire continues burning after the power is shut off, take suitable actions including the use of a fire extinguisher or a call for the fire department.

2. Hazard Warning Statements

The following are the hazard warning statements contained in this manual.

2.1 WARNING Statement

(chapter 3, page 3-5)

WARNING

- Switch off the power supply before making connections to the terminal block. Making connections with the power supply being switched on may incur electrical shock hazards.
- Electric shock hazards exist so that you might suffer burns or become electrocuted. Further, the system might malfunction due to noise interference. Therefore, ground the line ground (LG), frame ground (FG), and shield wire (SHD).

2.2 CAUTION Statement

(chapter 7, page 7-2)

CAUTION

Before replacing the module, switch it off to avoid electrical shock hazards and also to prevent it from being damaged or malfunctioning.

2.3 NOTICE Statements

(chapter 1, page 1-6)

NOTICE

Users of this product must have adequate knowledge of the Windows® environment and user interface. This system conforms to the Windows® standard. This manual is prepared for users who are familiar with the basic Windows® operating procedures.

(chapter 2, page 2-2)

NOTICE

A CMU module is needed to select “Free-running – Task” with an S10V module.

(chapter 2, page 2-3)

NOTICE

- Do not alter the module switch setting while the module is switched on.
- Do not allow the following two different module switch settings to be intermixed on a single unit:
  "Free-running – Computing function" and "Free-running – Task"
  The following sets of module switch settings can be intermixed on a single unit:
  • [H-7338 protocol] and [Free-running – Computing function]
  • [H-7338 protocol] and [Free-running – Task]
- Do not allow channel numbers to be defined in duplicate across the RS-232C/422 modules (LQE060, LQE160, LQE165, LQE560, LQE565) connected a single unit.
- Up to two (four channels) of the RS-232C/422 modules (LQE060, LQE160, LQE165, LQE560, LQE565) can be connected a single unit. Do not mount than three modules.

(chapter 3, page 3-2)

NOTICE

S10mini Series

- Mount the option module in option slots that is located to the immediate right of the CPU module. Be sure that no I/O module is mounted between this option module and CPU module. Also, ensure that there is no unoccupied slot between option modules.
- This module (LQE560/565) can be mounted together with the RS-232C/422 module (LQE060/160/165), which is specially designed for use with the S10mini.

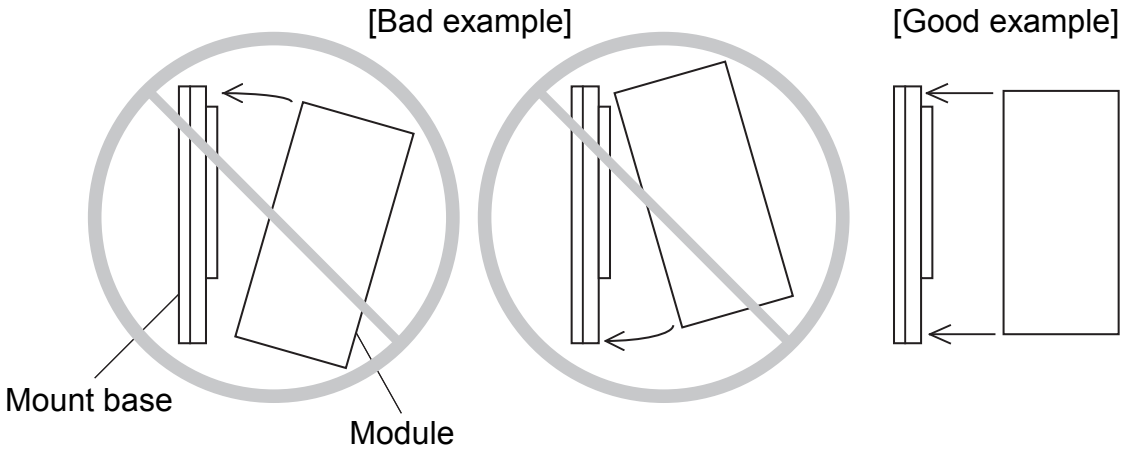
S10V Series

- There are no specific rules about the mounting position or unoccupied slots.
- The RS-232C/422 module (LQE060/160/165) cannot be used as it is specially designed for use with the S10mini.
- An S10V CMU module is prerequisite to carrying out data transmission by tasking (C mode).

(chapter 3, page 3-3)

NOTICE

- Dust or other foreign matter might accumulate on the connector, resulting in poor contact. Immediately after the module is unpacked, perform the mounting and wiring procedures.
- To prevent the module from being damaged, observed the following precautions when you mount or demount the module:
  - Before mounting the module to the mount base connector, check that the connector pins are properly aligned and not bent, broken, or soiled with dirt or the like.
  - Ensure that the module is parallel to the mount base vertical surface as shown below when mounting. If you connect a module to or disconnect it from its connector while it is tilted, the connector pins may become damaged.
  - If the mount base is positioned overhead due to the employed enclosure structure, use a stepladder or the like and mount the module squarely. If you mount the module obliquely, the connector may become damaged.



(chapter 3, page 3-5)

NOTICE

- For the ground wiring of the FG (frame ground), connect the FG terminal of each module with an external terminal to the FG terminal of the mount base. Perform class D grounding from the FG terminal of the mount base.
- Use a ground wire with a wire diameter of 2 mm^2 or more.
- Connect the shielding cable of the communication cable to the SHD terminal of the CPU or LPU module.

(chapter 3, page 3-13)

NOTICE

- Be sure to connect the signal ground (SG) of the RS-232C/422 module and that of the remote station with an interface cable.
- Ground the shielded grounding terminal of the interface cable at both CPU or LPU module and the remote station to provide added noise immunity. If the CPU or LPU module and the remote station differ in their ground potential, ground the interface cable at the RS-232C/422 module only.
- Connect the shielded grounding (SHD) terminal of the interface cable (at the RS-232C/422 module) to the shielded grounding (SHD) terminal of the CPU or LPU module terminal strip.

(chapter 4, page 4-2)

NOTICE

- Select either of the following settings to suit the application usage.
  - [Free-running – Computing function]
  - [Free-running – Task]
    [S10mini] An extension memory module is needed if the LQP000 is used as a CPU.
    [S10V] An option module CMU (LQP520) is needed.
  - H-7338 protocol
- The following software packages are needed to edit the LGB table:
  [S10mini] [External serial link system, RS-232C link system] (Model: S-7890-24)
  [S10V] [External serial link system, RS-232C link system] (Model: S-7895-24)
- With the free-running protocol, it is necessary to edit the LGB table to suit the specifications of the remote station.
  With the H-7338 protocol, there is no need to set LGB (no effect).
- Edits to the LGB table do not take effect until after the module is reset (using the CPU or LPU module reset switch). If the power fails and then recovers after or during a reset after the LGB table has been edited, the table will be reset to its previous settings. In this case, edit the LGB table again and reset the module.

(chapter 4, page 4-16)

NOTICE

With the ASCII specification, the text (binary data) before its ASCII conversion and the ECD are checked.

(chapter 4, page 4-25)

NOTICE

With the S10V series, a CMU module is needed to use task systems (C mode).

(chapter 4, page 4-26)

NOTICE

- Using with S10/2α
  When using an optional adapter (LWZ800) to connect the RS-422 module (LQE565) to an S10/2α unit, use the RS-422 module (LQE565) on other than channel 0 if CPU module RS-422 host interrupt is to be used.
  If the RS-422 module (LQE565) is used on channel 0, Z200 would be duplicated between the CPU module and the RS-422 module as a host interrupt register, causing both host interrupts to turn on.
- Using with the S10mini
  No constraint is placed on the host interrupt from channel 0 of the RS-422 module (LQE565). The S10mini CPU module does not support host interrupt.
- Using with the S10V
  No constraint is placed on the host interrupt from channel 0 of the RS-422 module (LQE565). The S10V CPU module uses Z204 as a host interrupt register.

(chapter 4, page 4-28)

NOTICE

Before installing the S10mini external serial link system, be sure to exit all the currently open Windows® programs. Do not forget to exit anti-virus software and other memory-resident programs. If you install the S10mini external serial link system without exiting such programs, an error may occur during installation. If such an error occurs, first uninstall the external serial link system as directed in “4.5.2 Uninstalling,” exit all the Windows® programs, and then install the S10mini external serial link system again.

(chapter 4, page 4-30)

NOTICE

- The S10V basic system is required for operating the S10V external serial link system. If it is not installed, you cannot install the S10V external serial link system.
- Before installing the S10V external serial link system, be sure to exit all the currently open Windows® programs. Do not forget to exit anti-virus software and other memory-resident programs. If you install the S10V external serial link system without exiting such programs, an error may occur during installation. If such an error occurs, first uninstall the S10V external serial link system as directed in "4.5.2 Uninstalling," exit all the Windows® Programs, and then install the S10V external serial link system again.

(chapter 4, page 4-31)

NOTICE

- If Windows® opens a window during the uninstall process to display the question "Remove Shared File?," click the No button to retain shared files.
- When you want to reinstall the external serial link system, be sure to perform an uninstall and then perform an install.

(chapter 4, page 4-34)

NOTICE

- The S10mini Series does not support GP-IB. When connecting it to a personal computer, select either RS-232C or Ethernet.
- The S10V external serial link window does not have a GPIB button.

(chapter 5, page 5-6)

NOTICE

It is not permitted to define a computing function on channel No. 0 and a subroutine on channel No. 1 or use subroutines for transmission and computing functions for reception.
All send/receive handlers associated with a given CPU or LPU unit must be uniformly defined as computing functions or subroutines.

(chapter 5, page 5-6)

NOTICE
When using computing functions to build a ladder program with the S10mini, be sure to register them in the LGB table as instructed “4.2　Editing the LGB Table” and “4.3　LGB Table Settings” and then select [Build] – [Receive] from the ladder program edit window in ladder chart system to receive the CPU data with the tool. This will allow the computing functions to be assembled into a ladder program.

(chapter 5, page 5-32)

NOTICE
When using any other printer, edit the items marked by an asterisk (*) to meet the specifications of that printer as needed.

(chapter 5, page 5-36)

NOTICE
● The printer used this time prints so slow when compared with the speed of data transfer that the continuous transmission of print data to it would cause the printer to malfunction with a received data buffer overflowing.　The ladder program pauses transmission for 20 seconds after printing 30 lines to prevent this. ● The sample program is designed to promote understanding.　The working program should make error checks on the send handler return code and system registers (S).

(chapter 7, page 7-2)

NOTICE
Static electricity could cause damage to the module.　Before handling the module, discharge static electricity on the human body.

(chapter 7, page 7-4)

NOTICE

- The receive task entry table is not automatically batch-saved. Users should specify its address when saving it.
  The LGB table and the user computing function entry table are automatically batch-saved.
- If a power failure occurs before or during the reset that is carried out after the end of editing from a tool or batch loading, the data written to flash memory in the module would take effect, rather than the edits or settings entered by the batch load. In this case, reset the module again after the end of editing or batch loading.

WARRANTY AND SERVICING

Unless a special warranty contract has been arranged, the following warranty is applicable to this product.

1. Warranty period and scope
   Warranty period
   The warranty period for this product is for one year after the product has been delivered to the specified delivery site.

   Scope
   If a malfunction should occur during the above warranty period while using this product under normal product specification conditions as described in this manual, please deliver the malfunctioning part of the product to the dealer or Hitachi Engineering & Services Co., Ltd. The malfunctioning part will be replaced or repaired free of charge. If the malfunctioning is shipped, however, the shipment charge and packaging expenses must be paid for by the customer.

   This warranty is not applicable if any of the following are true.

   - The malfunction was caused by handling or use of the product in a manner not specified in the product specifications.
   - The malfunction was caused by a unit other than that which was delivered.
   - The malfunction was caused by modifications or repairs made by a vendor other than the vendor that delivered the unit.
   - The malfunction was caused by a relay or other consumable which has passed the end of its service life.
   - The malfunction was caused by a disaster, natural or otherwise, for which the vendor is not responsible.

   The warranty mentioned here means the warranty for the individual product that is delivered. Therefore, we cannot be held responsible for any losses or lost profits that result from the operation of this product or from malfunctions of this product. This warranty is valid only in Japan and is not transferable.

2. Range of services
   The price of the delivered product does not include on-site servicing fees by engineers. Extra fees will be charged for the following:

   - Instruction for installation and adjustments, and witnessing trial operations.
   - Inspections, maintenance and adjustments.
   - Technical instruction, technical training and training schools.
   - Examinations and repairs after the warranty period is concluded.
   - Even if the warranty is valid, examination of malfunctions that are caused by reasons outside the above warranty scope.

This Page Intentionally Left Blank

This manual provides information on the following hardware and program products:

<Hardware product>
RS-232C/422 (LQE560/565)

<Program products>
S-7890-24, EXTERNAL SERIAL LINK, RS-232C LINK SYSTEM, 07-02
S-7895-24, S10V EXTERNAL SERIAL LINK, RS-232C LINK SYSTEM, 01-03

Revision record

Revision No.	Revision record (revision details and reason for revision)	Month, Year	Remarks
B	First Edition	February 2003	
D	Section 7.3, “Replacing or Adding on the Module” is newly added.	October 2008	
E	Subsection 1.2.2, a receive-monitoring time condition is changed in the free-running transmission specifications.	April 2010	
	Section 3.4, a change is made to an example of RS-232C wiring shown.		
	Subsection 7.4.3, an additional type of transmission error is added.		
	Subsection 7.4.4, an additional type of reception error is added.		
F	• All the safety precautions and instructions in this manual have been reviewed and necessary changes are added to them. • Windows® 7 (32-bit) operating system is newly supported.	February 2013	

In addition to the above changes, all the unclear descriptions and typographical errors found are also corrected without prior notice.

PREFACE

Thank you for purchasing the RS-232C/422 module, which is an option for use with the S10mini/S10V.
This manual, named "USER'S MANUAL OPTION RS-232C/422," describes how to use the RS-232C/422 module. For proper use of the RS-232C/422 module, it is requested that you thoroughly read this manual.

The S10mini and S10V products are available in two types: standard model and environmentally resistant model. The environmentally resistant model has thicker platings and coatings than those for the standard model.
The model number of the environmentally resistant model is marked by adding the suffix "-Z" to the model number of the standard model.

(Example) Standard model: LQE560
Environmentally resistant model: LQE560-Z

This manual is applicable to both the standard model and environmentally resistant models. Although the descriptions contained in this manual are based on the standard model, follow the instructions set forth in this manual for proper use of the product even if you use the environmentally resistant model.

<Trademarks>
Microsoft® Windows® operating system, Microsoft® Windows® 95 operating system, Microsoft® Windows® 98 operating system, Microsoft® Windows® 2000 operating system, Microsoft® Windows® XP operating system, Microsoft® Windows® 7 (32-bit) operating system are registered trademarks of Microsoft Corporation in the United States and/or other countries.

<Note for storage capacity calculations>

- Memory capacities and requirements, file sizes and storage requirements, etc. must be calculated according to the formula 2^n. The following examples show the results of such calculations by 2^n (to the right of the equals signs).
  1 KB (kilobyte) = 1,024 bytes
  1 MB (megabyte) = 1,048,576 bytes
  1 GB (gigabyte) = 1,073,741,824 bytes
- As for disk capacities, they must be calculated using the formula 10^n. Listed below are the results of calculating the above example capacities using 10^n in place of 2^n.
  1 KB (kilobyte) = 1,000 bytes
  1 MB (megabyte) = $1{,}000^2$ bytes
  1 GB (gigabyte) = $1{,}000^3$ bytes

CONTENTS

FIGURES

TABLES

This Page Intentionally Left Blank

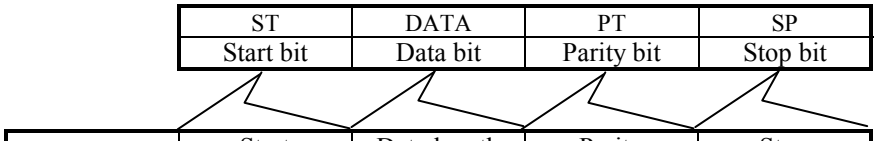
1 SPECIFICATIONS

1.1 Use

The RS-232C module (Model: LQE560) is used in conjunction with an S10V LPU module or S10mini CPU module to carry out data communication under free-running or H-7338 protocols pursuant to EIA RS-232C (hereafter abbreviated “RS-232C”) specifications.
The RS-422 module (Model: LQE565) is used in conjunction with an S10V LPU module or S10mini CPU module to carry out data communication under free-running or H-7338 protocols pursuant to EIA RS-422 (hereafter abbreviated “RS-422”) specifications.
When these modules (LQE560, LQE565) are used with an S10mini CPU unit, they can be intermixed with RS-232C/RS-422 modules (LQE060, LQE160, LQE165) dedicated to the S10mini. (The LQE060, LQE160, and LEQ165 cannot be used with an S10V LPU unit.)
An S10V CMU module is prerequisite to combining these modules (LQE560, LQE565) with an S10V LPU module to carry out data transmission by tasking (C mode).

1.2 Specifications

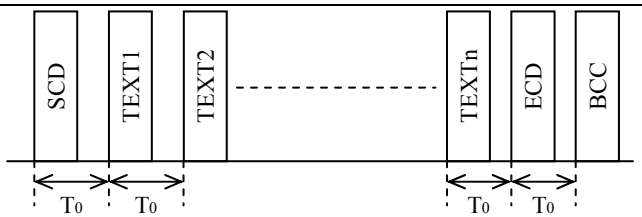
1.2.1 System specifications

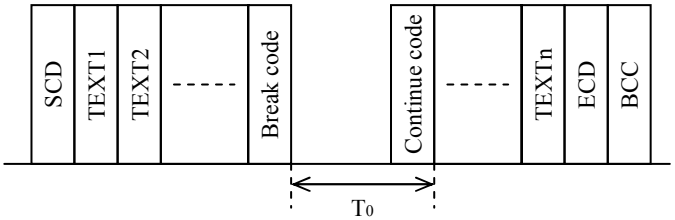
<table>
<tr><th colspan="2">Item</th><th colspan="3">Specification</th></tr>
<tr><td colspan="2">Model</td><td>LQE560</td><td colspan="2">LQE565</td></tr>
<tr><td colspan="2">Protocol</td><td colspan="3">Free-running/H-7338, switched</td></tr>
<tr><td colspan="2">Total number of RS-232C/RS-422 modules mounted</td><td colspan="3">S10mini: Two modules/CPU (mounted left-justified)
S10V: Two modules/LPU (no need to mount left-justified)</td></tr>
<tr><td colspan="2">Module slot width</td><td colspan="3">One-slot width module</td></tr>
<tr><td colspan="2">Mass</td><td>220 g</td><td colspan="2">220 g</td></tr>
<tr><td colspan="2">Transmission mode</td><td colspan="3">Serial transmission (Bit serial transmission)</td></tr>
<tr><td colspan="2">Communication mode</td><td colspan="3">Half-duplex/full-duplex, switched</td></tr>
<tr><td colspan="2">Synchronization method</td><td colspan="3">Start-stop synchronization</td></tr>
<tr><td colspan="2">Interface</td><td>Compliant with EIA RS-232-C</td><td colspan="2">Compliant with EIA RS-422</td></tr>
<tr><td colspan="2">Data frame structure</td><td colspan="3"> 

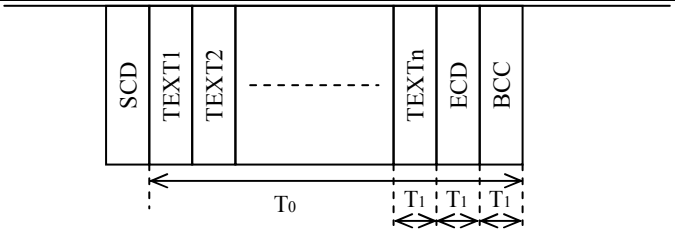
<table>
<tr><th></th><th>Start</th><th>Data length</th><th>Parity</th><th>Stop</th></tr>
<tr><td rowspan="10">Free-running</td><td rowspan="10">1 bit</td><td rowspan="5">7 bits</td><td>Even</td><td rowspan="2">2 bits</td></tr>
<tr><td>Odd</td></tr>
<tr><td>Even</td><td rowspan="2">1 bit</td></tr>
<tr><td>Odd</td></tr>
<tr><td rowspan="2">No</td><td>2 bits</td></tr>
<tr><td>1 bit</td></tr>
<tr><td rowspan="5">8 bits</td><td>Even</td><td rowspan="2">2 bits</td></tr>
<tr><td>Odd</td></tr>
<tr><td>Even</td><td rowspan="2">1 bit</td></tr>
<tr><td>Odd</td></tr>
<tr><td rowspan="2">No</td><td>2 bits</td></tr>
<tr><td>1 bit</td></tr>
<tr><td>H-7338 protocol</td><td>1 bit</td><td>8 bits</td><td>Odd</td><td>1 bit</td></tr>
</table></td></tr>
<tr><td rowspan="2">Baud rate</td><td>Free-running</td><td colspan="3">300, 600, 1200, 2400, 4800, 9600, 19200 (bps)</td></tr>
<tr><td>H-7338 protocol</td><td colspan="3">19200 (bps)</td></tr>
<tr><td rowspan="5">Connecting cable</td><td>Distance</td><td>Up to 15 m</td><td colspan="2">Up to 500 m</td></tr>
<tr><td>Wire type</td><td colspan="3">Shielded twisted pair cable</td></tr>
<tr><td>Wire diameter</td><td>0.08 mm^2 or more</td><td colspan="2">0.3 mm^2 or more</td></tr>
<tr><td>Resistance</td><td>229 Ω/km or less (20°C)</td><td colspan="2">54.4 Ω/km or less (20°C)</td></tr>
<tr><td>Recommended brand</td><td>CO-VV-SB(MA)13P × 28AWG(7/0.127) (*)
(manufactured by Hitachi Cable, Ltd.)</td><td colspan="2">CO-SPEV-SB-5P 0.3 mm^2
(manufactured by Hitachi Cable, Ltd.)</td></tr>
<tr><td rowspan="3">Connector</td><td>Type</td><td colspan="3">D-sub 9-pin connector</td></tr>
<tr><td rowspan="2">Remarks</td><td colspan="3">Cover: HDE-CTH1 (manufactured by HIROSE ELECTRIC CO., LTD.)</td></tr>
<tr><td colspan="3">Connector: HDEB-9S (manufactured by HIROSE ELECTRIC CO., LTD.)</td></tr>
<tr><td colspan="2">Cable grounding condition</td><td colspan="3">Double-ended grounding</td></tr>
</table>

(*) Choose the number of cores from between 5P and 8P to meet the number of signals required.

1.2.2 Free-running transmission specifications

<table>
<tr><th colspan="2">Item</th><th colspan="5">Specification</th></tr>
<tr><td colspan="2">Transmission control procedure</td><td colspan="5">Free-running</td></tr>
<tr><td colspan="2" rowspan="3">Priority level</td><td colspan="5">Local station prioritized (Rejects receive requests while transmitting)</td></tr>
<tr><td colspan="5">Remote station prioritized (Accepts receive requests even while transmitting)</td></tr>
<tr><td colspan="5">No priority level (Full-duplex communication)</td></tr>
<tr><td colspan="2" rowspan="2">Data change mode</td><td colspan="5">Transmits and receives text data as it is.</td></tr>
<tr><td colspan="5">Transmits text data after ASCII conversion and receives text data after binary conversion.</td></tr>
<tr><td rowspan="4">Transmitted block structure</td><td>Start code</td><td colspan="5">No, 1 to 4 characters</td></tr>
<tr><td>Text</td><td colspan="5">No, 1 to 512 bytes</td></tr>
<tr><td>End code</td><td colspan="5">No, 1 to 4 characters</td></tr>
<tr><td>Block check character</td><td colspan="5">No, horizontal even parity, horizontal odd parity</td></tr>
<tr><td colspan="2">Send delay time</td><td colspan="5"> 

T0 = 0 to 32767 ms (in intervals of 1 ms)</td></tr>
<tr><td rowspan="2">Send break and continue codes</td><td>Break code</td><td>No</td><td colspan="2">1 character</td><td colspan="2">2 characters</td></tr>
<tr><td>Continue code</td><td>No</td><td>1 character</td><td>2 characters</td><td>1 character</td><td>2 characters</td></tr>
<tr><td colspan="2">Send break timeout</td><td colspan="5"> 

T0 = 0 to 3276.7 s (in intervals of 100 ms)</td></tr>
<tr><td colspan="2">Receive monitoring time</td><td colspan="5"> 

T0 = 0 to 3276.7 s (in intervals of 100 ms)
T1 = 1 to 32767 ms (if “Text variable length” is specified)</td></tr>
<tr><td colspan="2" rowspan="2">Request to Send (RS) output</td><td colspan="5">RS output (RS pin held ON)</td></tr>
<tr><td colspan="5">No RS output</td></tr>
<tr><td colspan="2" rowspan="2">Equipment Ready (ER) output</td><td colspan="5">Not Ready output</td></tr>
<tr><td colspan="5">Ready output (ER pin held ON)</td></tr>
<tr><td colspan="2" rowspan="2">Data Set Ready (DR) input</td><td colspan="5">No checking</td></tr>
<tr><td colspan="5">Checking enabled</td></tr>
<tr><td colspan="2" rowspan="2">Control signal automatic control</td><td colspan="5">Manual control</td></tr>
<tr><td colspan="5">Automatic control</td></tr>
<tr><td colspan="2">Send buffer size</td><td colspan="5">512 bytes</td></tr>
<tr><td colspan="2">Receive buffer size</td><td colspan="5">512 bytes × 8 buffers</td></tr>
</table>

1.3 System Software Specifications

1.3.1 System overview

When you use the RS-232C/422 module, you must register various items of information in the module. Register the module information using the following system software (tools) and by performing operating procedures similar to those for general Windows® applications.

Table 1-1 Types of System Software (Tools)

Package	Model		Supply style
	For S10mini	For S10V	
External serial link system RS-232C link system	S-7890-24	S-7895-24	Optional

1.3.2 Required hardware and software

The following hardware and software are required for the use of the RS-232C/422 system software:

(1) For S10mini
- Personal computer (main unit) containing a Pentium 133 MHz of faster CPU
- Personal computer (main unit) containing a Pentium 300 MHz of faster CPU (when Windows® 2000 or Windows® XP is used)
- Display having a resolution of 800 × 600 dots (SVGA) or higher
- Microsoft® Windows® 95 operating system, Microsoft® Windows® 98 operating system, Microsoft® Windows® 2000 operating system or Microsoft® Windows® XP operating system
- Microsoft® Internet Explorer 4.01 or later
- At least 32 MB of RAM
- At least 64 MB of RAM (when Windows® 2000 is used)
- At least 128 MB of RAM (when Windows® XP is used)
- At least 10 MB of free hard disk space
- Cable for connecting the personal computer to the CPU unit (RS-232C cross cable with D-sub 9-pin connectors) or cable for connecting the personal computer to the ET.NET module (10BASE-T twisted pair cross cable with RJ-45 modular connectors)

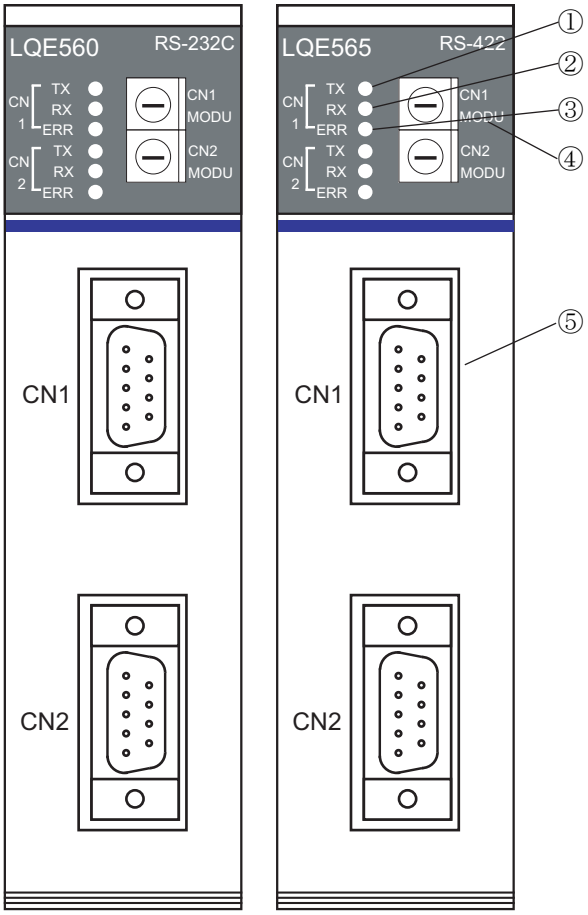
(2) For S10V

- Personal computer (main unit) containing a Pentium 300 MHz of faster CPU, or a 1 GHz or faster CPU (when Windows® 7 (32-bit version) is used)
- Display having a resolution of 800 × 600 dots (SVGA) or higher
- Microsoft® Windows® 2000 operating system, Microsoft® Windows® XP operating system or Microsoft® Windows® 7 (32-bit) operating system
- At least 64 MB of RAM (when Windows® 2000 is used)
- At least 128 MB of RAM (when Windows® XP is used)
- At least 1 GB of RAM (when Windows® 7 (32-bit) is used)
- At least 10 MB of free hard disk space
- Cable for connecting the personal computer to the LPU unit (RS-232C cross cable with D-sub 9-pin connectors) or cable for connecting the personal computer to the CMU or ET.NET module (10BASE-T or 100BASE-T twisted pair cross cable with RJ-45 modular connectors)

NOTICE
Users of this product must have adequate knowledge of the Windows® environment and user interface. This system conforms to the Windows® standard. This manual is prepared for users who are familiar with the basic Windows® operating procedures.

2 NAMES AND FUNCTIONS OF EACH PART

2.1 Names and Functions of Each Part



① TX LED
Glows when transmitting data.

② RX LED
Glows when receiving data.

③ ERR LED
Glows when a hardware error has occurred.

④ Module switch
Use to set a communications protocol and a channel number.
Settings do not take effect until the module restarts.

Module switch	Communications protocol	Channel number
0	Free-running – Computing function	#0
1	Free-running – Computing function	#1
2	Free-running – Computing function	#2
3	Free-running – Computing function	#3
4	Free-running – Task	#0
5	Free-running – Task	#1
6	Free-running – Task	#2
7	Free-running – Task	#3
8	H-7338 protocol	#0
9	H-7338 protocol	#1
A	H-7338 protocol	#2
B	H-7338 protocol	#3
C	Reserved for maintenance use	
D		
E		
F		

⑤ RS-232C connector (LQE560), RS-422 connector (LQE565)
Used to connect the module to a Remote device.

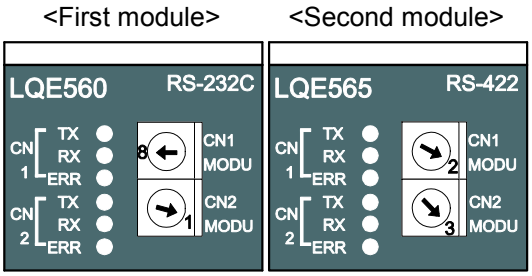
NOTICE

A CMU module is needed to select “Free-running – Task” with an S10V module.

NOTICE

- Do not alter the module switch setting while the module is switched on.
- Do not allow the following two different module switch settings to be intermixed on a single unit:
  "Free-running – Computing function" and "Free-running – Task"
  The following sets of module switch settings can be intermixed on a single unit:
  • [H-7338 protocol] and [Free-running – Computing function]
  • [H-7338 protocol] and [Free-running – Task]
- Do not allow channel numbers to be defined in duplicate across the RS-232C/422 modules (LQE060, LQE160, LQE165, LQE560, LQE565) connected a single unit.
- Up to two (four channels) of the RS-232C/422 modules (LQE060, LQE160, LQE165, LQE560, LQE565) can be connected a single unit. Do not mount than three modules.

Proper setup example



Module switch setting

	Module switch	Communications protocol	Channel number
First module-CN1	8	H-7338 protocol	#0
First module-CN2	1	Free-running – Computing function	#1
Second module-CN1	2	Free-running – Computing function	#2
Second module-CN2	3	Free-running – Computing function	#3

This Page Intentionally Left Blank

3 MOUNTING AND WIRING

3.1 Mount Base

This module can be mounted in the mount bases shown in Table 3-1.

Table 3-1 Mount Bases Applicable to the RS-232C/422 Module

Series	Name	Model	Specifications
S10mini	2-slot mount base	HSC-1020	Power supply + CPU + 2 slots (option for I/O)
	4-slot mount base	HSC-1040	Power supply + CPU + 4 slots (option for I/O)
	8-slot mount base	HSC-1080	Power supply + CPU + 8 slots (option for I/O)
S10V	4-slot mount base	HSC-1540	Power supply + LPU + 4 slots (option for I/O)
	8-slot mount base	HSC-1580	Power supply + LPU + 8 slots (option for I/O)

3.2 Mounting the Module

Mount the option module in option slots (slot number 0 through 7) on the mount base as shown below:

NOTICE

S10mini Series

- Mount the option module in option slots that is located to the immediate right of the CPU module. Be sure that no I/O module is mounted between this option module and CPU module. Also, ensure that there is no unoccupied slot between option modules.
- This module (LQE560/565) can be mounted together with the RS-232C/422 module (LQE060/160/165), which is specially designed for use with the S10mini.

S10V Series

- There are no specific rules about the mounting position or unoccupied slots.
- The RS-232C/422 module (LQE060/160/165) cannot be used as it is specially designed for use with the S10mini.
- An S10V CMU module is prerequisite to carrying out data transmission by tasking (C mode).

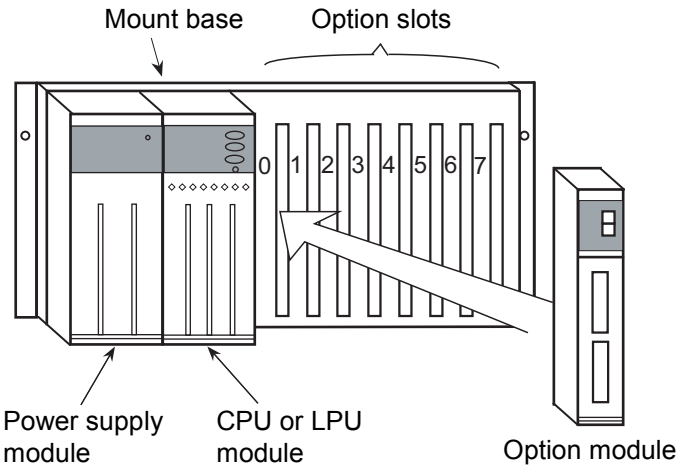
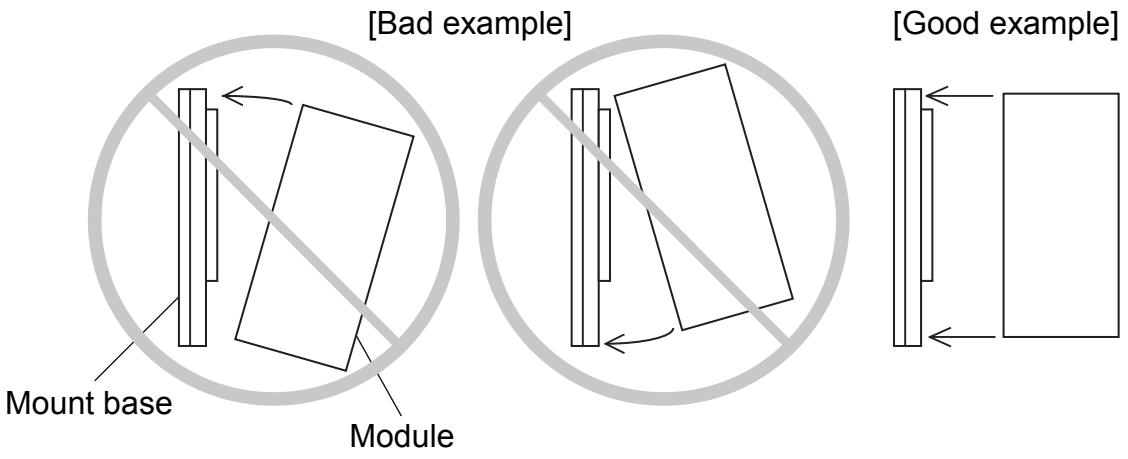


Figure 3-1 Mounting the Option Module

NOTICE

- Dust or other foreign matter might accumulate on the connector, resulting in poor contact. Immediately after the module is unpacked, perform the mounting and wiring procedures.
- To prevent the module from being damaged, observed the following precautions when you mount or demount the module:
  - Before mounting the module to the mount base connector, check that the connector pins are properly aligned and not bent, broken, or soiled with dirt or the like.
  - Ensure that the module is parallel to the mount base vertical surface as shown below when mounting. If you connect a module to or disconnect it from its connector while it is tilted, the connector pins may become damaged.
  - If the mount base is positioned overhead due to the employed enclosure structure, use a stepladder or the like and mount the module squarely. If you mount the module obliquely, the connector may become damaged.



3.3 Ground Wiring

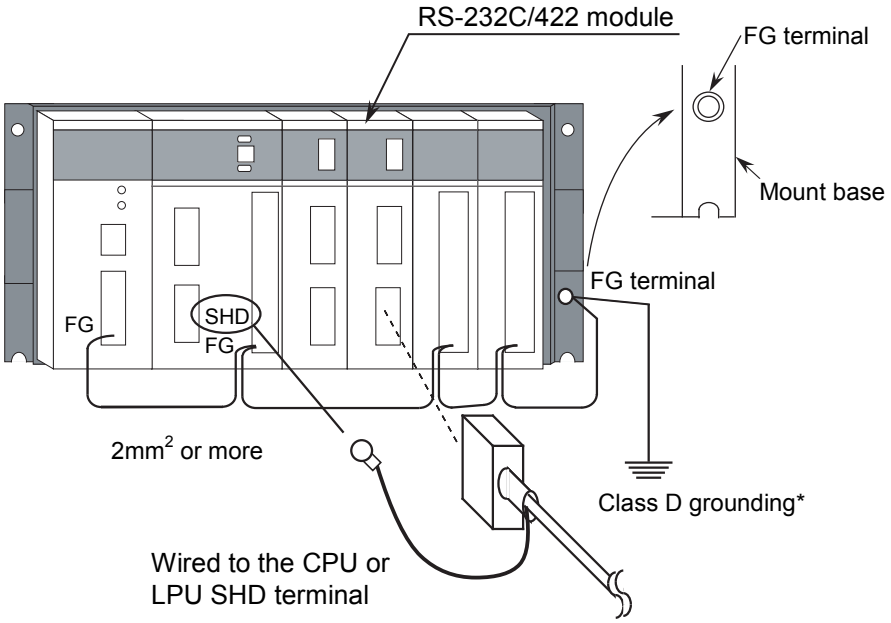


Figure 3-2 Ground Wiring

* Class D grounding is defined in the Technical Standard for Electrical Facilities of Japan. This standard states that the grounding resistance must be 100 ohms or less for equipment operating on 300 VAC or less, and 500 ohms or less for devices that shut down automatically within 0.5 seconds when shorting occurs in low tension lines.

WARNING

- Switch off the power supply before making connections to the terminal block. Making connections with the power supply being switched on may incur electrical shock hazards.
- Electric shock hazards exist so that you might suffer burns or become electrocuted. Further, the system might malfunction due to noise interference. Therefore, ground the line ground (LG), frame ground (FG), and shield wire (SHD).

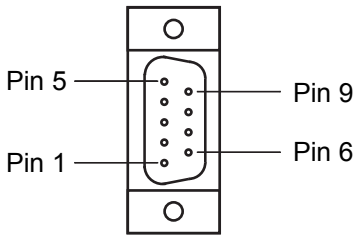
NOTICE

- For the ground wiring of the FG (frame ground), connect the FG terminal of each module with an external terminal to the FG terminal of the mount base. Perform class D grounding from the FG terminal of the mount base.
- Use a ground wire with a wire diameter of 2 mm^2 or more.
- Connect the shielding cable of the communication cable to the SHD terminal of the CPU or LPU module.

3.4 RS-232C Wiring

3.4.1 Pin configuration and signal definitions

(1) Pin configuration



RS-232C connector
(D-sub 9-pin)

Pin No.	Signal name	I/O	Explanation
1	CD (Data Carrier Detect)	Input	On: Receiver carrier detected Off: Receive carrier not detected
2	RD (Receive Data)	Input	On: Received data space Off: Received data mark
3	SD (Send Data)	Output	On: Received data space Off: Received data mark
4	ER (Equipment Ready)	Output	On: Module ready to transmit and receive Off: Module not ready to transmit and receive
5	SG (Signal Ground)	–	Signal ground
6	DR (Data Set Ready)	Input	On: Remote station ready to operate Off: Remote station not ready to operate
7	RS (Request to Send)	Output	On: Module having a request to send Off: Module having no request to send
8	CS (Clear to Send)	Input	On: Remote station ready to transmit Off: Remote station not ready to transmit
9	Not used	–	

Voltage levels

	Mark	Space
Interpretation	1/off	0/on
Output condition	-5 V to -15 V	+5 V to + 15 V
Input condition	≤ -3 V	+3 V ≤

(2) Signal definitions

CD ..... A control signal line reporting that the remote station is receiving a valid signal from the communication line if it is a modem. RD is enabled when CD is on. When CD turns on, the RS-232C module enters data from the remote station. RD is kept in the mark (off) state while CD is off.

RD ..... Received data line from the remote station to the RS-232C module. Enabled when CD is on. RD is kept in the mark (off) state while the remote station is not transmitting data (that is, CD is off).

SD..... Transmitted data line from the RS-232C module to the remote station. Enabled when the four control lines of RS, CS, DR, and ER are on.

ER..... A control signal line reporting that the RS-232C module is ready to transmit data to and from the remote station. If the remote station is a modem, it is connected to the line when ER turns on, and is disconnected when ER turns off.

SG..... Signal ground. It provides a reference voltage (0 V) for all the signals.

DR ..... A control signal line reporting that the remote station is ready to operate. The remote station is connected to the line if it is a modem, allowing control signals to be transmitted to and from the RS-232C module.

RS ..... A control signal line reporting that data is available from the RS-232C module for transmission to the remote station. The remote station remains ready to receive data from the RS-232C module while RS is on. Once RS has turned off, it cannot turn on again until CS turns off.

CS ..... A control signal line reporting that the remote station is ready to transmit over the communication line if it is a modem. The remote station is ready to receive transmitted data from the RS-232C module the CS is on.

3.4.2 Wiring methods

Table 3-2 RS-232C Wiring Methods

Connection	Name	Wiring method (Logical connection)			Remote station example
		RS-232C module	Cable	Remote station	
Direct	Full modem support (Standard type) (Transmits and receives data while implementing CD-based received management and DR-based transmit management.)	RS CS CD SD RD ER DR SG		RS CS CD SD RD ER DR SG	Personal computer
	CD-based received management (Readies the RS-232C module to receive on Request to Send (RS) from the remote station.)	RS CS CD SD RD ER DR SG	(P)(P)	RS CS CD SD RD ER DR SG	
	DR-based received management (Transmits data from the RS-232C module on Equipment Ready (ER) from the remote station.)	RS CS CD SD RD ER DR SG	(P)(P)	RS CS CD SD RD ER DR SG	Character display
	Data only (Transmits and receives data without checking the remote station and RS-232C module status.)	RS CS CD SD RD ER DR SG	(P)(P) (P)(P)	RS CS CD SD RD ER DR SG	Printer
Modem	Modem connection	RS CS CD SD RD ER DR SG		RS CS CD SD RD ER DR SG	

Note: In the table, (P) denotes the availability of a Request to Send (RS), or the act of keeping ER in the ready state.

3.4.3 Wiring example

An example of standard wiring is shown below.

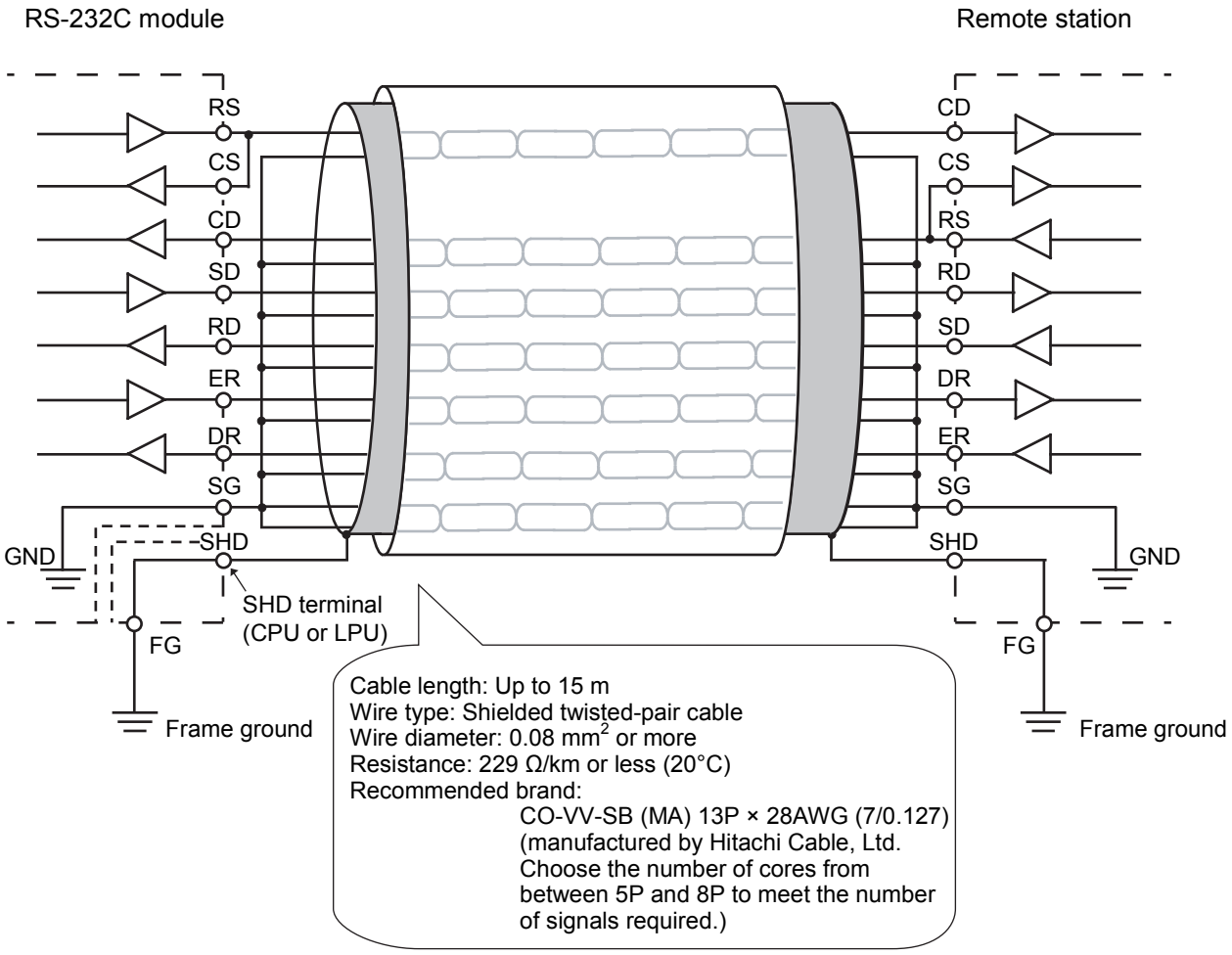
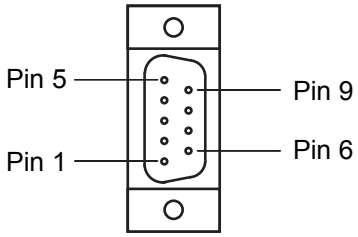


Figure 3-3 Example of RS-232C Wiring

3.5 RS-422 Wiring

3.5.1 Pin configuration and signal definitions

(1) Pin configuration



RS-422 connector
(D-sub 9-pin)

Voltage levels

	Mark	Space
Interpretation	1/off	0/on
Output condition	-3 V to -6 V	+3 V to + 6 V
Input condition	≤ -0.2 V	+0.2 V ≤

Pin No.	Signal name	I/O	Explanation
1	RD-L (Receive Data Low)	Input	Received data standard
2	RD-H (Receive Data High)	Input	On: Received data space Off: Received data mark
3	SD-H (Send Data High)	Output	On: Transmitted data space Off: Transmitted data mark
4	SD-L (Send Data Low)	Output	Transmitted data standard
5	SG (Signal Ground)	–	Signal ground
6	Not used	–	
7	ATT-H (Attention-int High)	Output	On: Interrupt request existing Off: No interrupt request
8	Not used	–	Interrupt signal standard
9	ATT-L (Attention-int Low)	Output	

(2) Signal definitions

RD ..... Receive data line from the remote station to the RS-232C module.

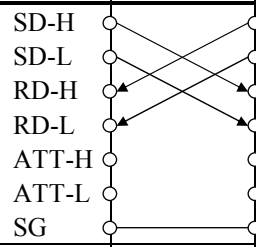
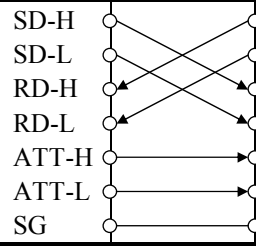
SD..... Transmitted data line from the RS-232C module to the remote station.

SG..... Signal ground between stations.

ATT ..... An interrupt signal line from the RS-422 module to the remote station when the remote station is a LINK-PCS.

3.5.2 Wiring methods

Table 3-3 RS-422 Wiring Methods

Connection	Name	Wiring method (Logical connection)			Remote station example
		RS-422 module	Cable	Remote station	
Direct	Data only				Indicator
	Data and attention interrupt				LINK-PCS

3.5.3 Wiring example

An example of standard wiring is shown below.

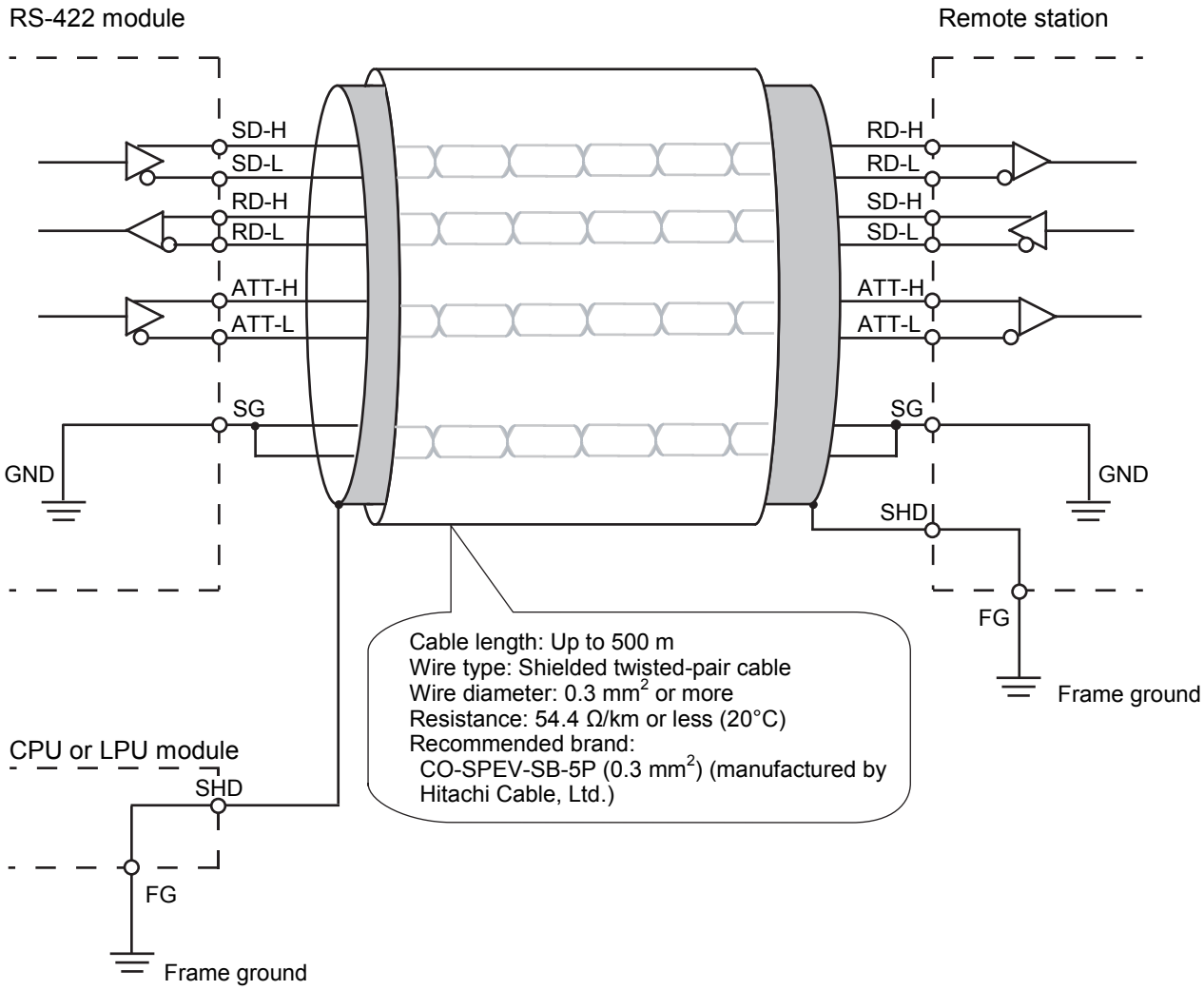


Figure 3-4 Example of RS-422 Wiring

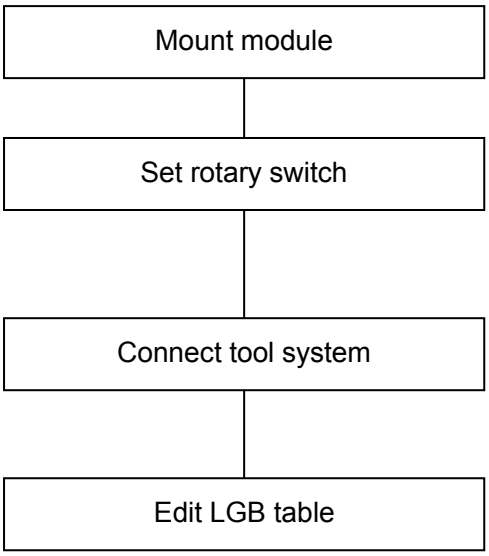
NOTICE

- Be sure to connect the signal ground (SG) of the RS-232C/422 module and that of the remote station with an interface cable.
- Ground the shielded grounding terminal of the interface cable at both CPU or LPU module and the remote station to provide added noise immunity. If the CPU or LPU module and the remote station differ in their ground potential, ground the interface cable at the RS-232C/422 module only.
- Connect the shielded grounding (SHD) terminal of the interface cable (at the RS-232C/422 module) to the shielded grounding (SHD) terminal of the CPU or LPU module terminal strip.

This Page Intentionally Left Blank

4 OPERATION

4.1 Starting the System



(1) Switch off the CPU or LPU and mount an RS-232C or RS-422 module.

(2) Set the module switch (communications protocol, channel number) to suit specific applications per connector. For how to use H-7338 protocol host interrupts (ATT signal), see "4.4 Host Interrupts."

(3) Connect the tool system to the CPU or LPU, and start the system. For how to install and start the tool system, see "4.5 Installing and Starting Up the System."

(4) Edit the LGB table to meet the specifications of the remote station connected. See "4.2 Editing the LGB Table" and "4.3 LGB Table Settings."

NOTICE

- Select either of the following settings to suit the application usage.
  - [Free-running – Computing function]
  - [Free-running – Task]
    [S10mini] An extension memory module is needed if the LQP000 is used as a CPU.
    [S10V] An option module CMU (LQP520) is needed.
  - H-7338 protocol
- The following software packages are needed to edit the LGB table:
  [S10mini] [External serial link system, RS-232C link system] (Model: S-7890-24)
  [S10V] [External serial link system, RS-232C link system] (Model: S-7895-24)
- With the free-running protocol, it is necessary to edit the LGB table to suit the specifications of the remote station.
  With the H-7338 protocol, there is no need to set LGB (no effect).
- Edits to the LGB table do not take effect until after the module is reset (using the CPU or LPU module reset switch). If the power fails and then recovers after or during a reset after the LGB table has been edited, the table will be reset to its previous settings. In this case, edit the LGB table again and reset the module.

4.2 Editing the LGB Table

Term LGB (Line Group Block) refers to a set of information that a communications control program transmits and receives over a line. Users decide this information by using the remote station link system that has been started from the tool system.
The way the LGB table is set is so important that it could inhibit successful hardware wiring to the remote station or disable correct transmission/reception due to inconsistent transmission procedures.
Edit the LGB table to meet the specifications of the remote station connected to each channel number.

Table 4-1 LGB Table Settings

Item	LQE560 (RS-232C)	LQE565 (RS-422)
Data frame	√	√
Baud rate	√	√
Priority level	√	√
Data change mode	√	√
Text size	√	√
Start code	√	√
End code	√	√
Block check character	√	√
Send delay time	√	√
Send break and continue codes	√	√
Send break timeout	√	√
Receive timeout	√	√
RS-422 gate control	–	–
Request to Send (RS)	√	–
Equipment Ready (ER)	√	–
Data Set Ready (DR)	√	–
Control signal automatic control	√	–
System selection	√	√

√: Setting enabled
–: Setting disable

4.3 LGB Table Settings

The individual items of the LGB table that users can edit are described on the pages that follow.

■ Data frame

Decide the structure of one byte of data transmitting over the line.

Table 4-2 Data Frame

Tool selection item		Data frame content	Default
No.	Display		
1	ST+7DT+EP+2SP	ST 2^0 —— 2^6 EP SP SP	
2	ST+7DT+OP+2SP	ST 2^0 —— 2^6 OP SP SP	
3	ST+7DT+EP+1SP	ST 2^0 —— 2^6 EP SP	
4	ST+7DT+OP+1SP	ST 2^0 —— 2^6 OP SP	
5	ST+8DT+2SP	ST 2^0 —— 2^7 SP SP	
6	ST+8DT+1SP	ST 2^0 —— 2^7 SP	
7	ST+8DT+EP+1SP	ST 2^0 —— 2^7 EP SP	
8	ST+8DT+OP+1SP	ST 2^0 —— 2^7 OP SP	√
9	ST+7DT+2SP	ST 2^0 —— 2^6 SP SP	
10	ST+7DT+1SP	ST 2^0 —— 2^6 SP	
11	ST+8DT+EP+2SP	ST 2^0 —— 2^7 EP SP SP	
12	ST+8DT+OP+2SP	ST 2^0 —— 2^7 OP SP SP	

ST: Start bit OP: Odd parity bit
DT: Data bit SP: Stop bit
EP: Even parity bit

■ Baud rate

Set the baud rate (bps) of the line (between 300 and 19,200 bps).

The four channels can be set to 19,200 bps at the same time (but not for the LQE060).

Table 4-3 Baud Rates

Tool selection item		Baud rate content	Default
No.	Display		
1	300 [BPS]	300 [bps]	
2	600 [BPS]	600 [bps]	
3	1200 [BPS]	1200 [bps]	
4	2400 [BPS]	2400 [bps]	
5	4800 [BPS]	4800 [bps]	√
6	9600 [BPS]	9600 [bps]	
7	19200 [BPS]	19200 [bps]	

BPS, bps: bits per second

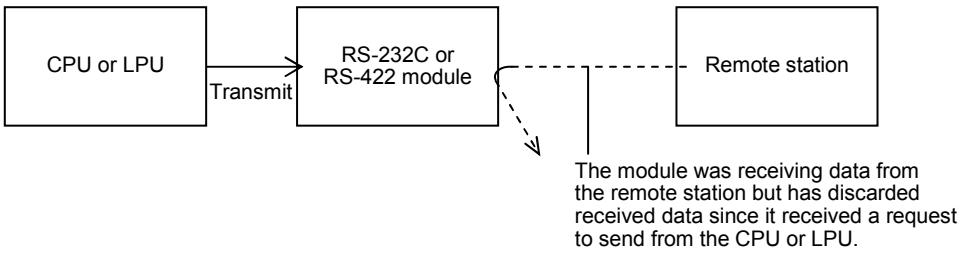
■ Priority level

Specify the priority level of the local station (RS-232C or RS-422 module) or the remote station (mating device). The priority level indicates that which the RS-232C or RS-422 module will give priority to, the CPU or LPU module or the remote station, when action is initiated from them concurrently.

Table 4-4 Priority Levels

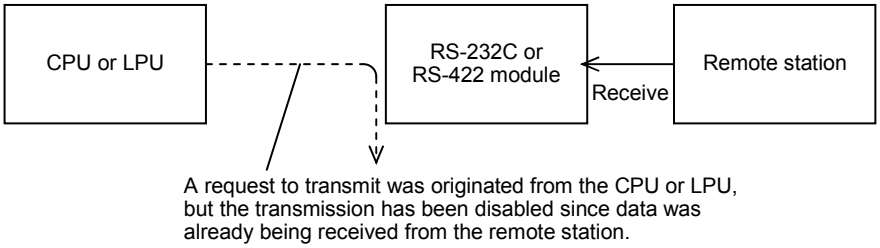
Tool selection item		Priority level content	Default
No.	Display		
1	Local station prioritized	Local station prioritized (Half-duplex communication)	√
2	Remote station prioritized	Remote station prioritized (Half-duplex communication)	
3	No priority level (Full-duplex communication)	No priority level (Full-duplex communication)	

- Local station prioritized



In the example shown above, the application program running on the CPU or LPU learns from the S-register that data reception from the remote station has made way for transmission.
The remote station, however, is unable to recognize that data received from the remote station has been discarded. It is the CPU or LPU's responsibility to notify the remote station.

- Remote station prioritized



In the example shown above, the application program running on the CPU or LPU learns from the S-register that data transmission has been disabled.

■ Data change mode

Specify whether to handle text data in circulation as ASCII data or as binary data.

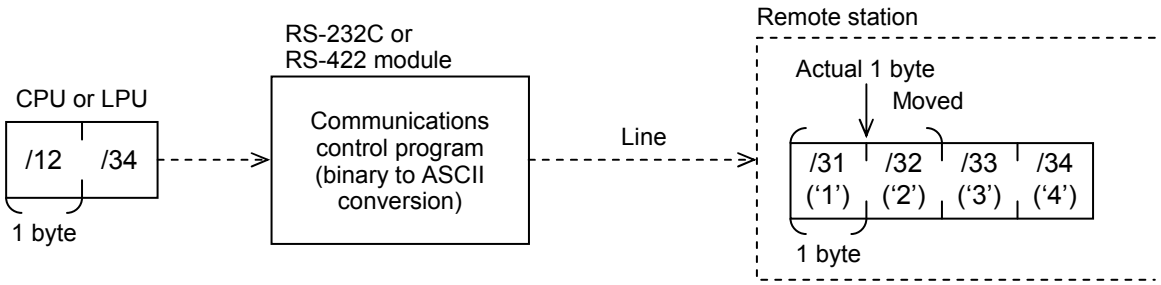
Table 4-5 Data Change Mode

Tool selection item		Data change mode content	Default
No.	Display		
1	ASCII	Handle text data as ASCII	
2	Binary	Handle text data as binary	√

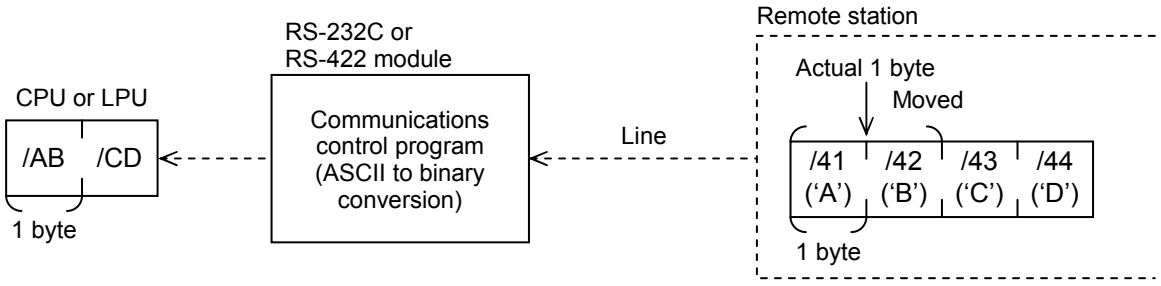
- ASCII specification

  With the ASCII specification, a remote station program performs a data conversion between ASCII and binary, so that data traffic over the line doubles.

  Transmitting any text data other than '0' to '9' and 'A' to 'F' would invoke an error.



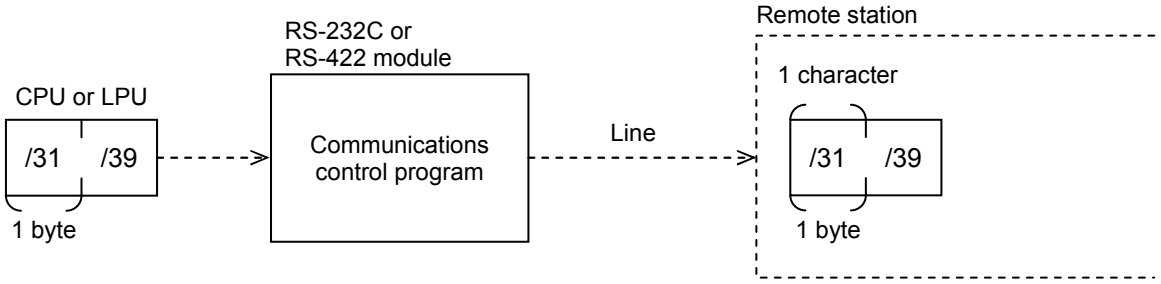
In the example shown above, if data /12 and /34 is transmitted from the CPU or LPU, the communications control program will perform a binary to ASCII conversion on the data to transmit data /31 ('1'), /32 ('2'), /33 ('3'), and /34 ('4'). A remote station program must be created to meet this condition.



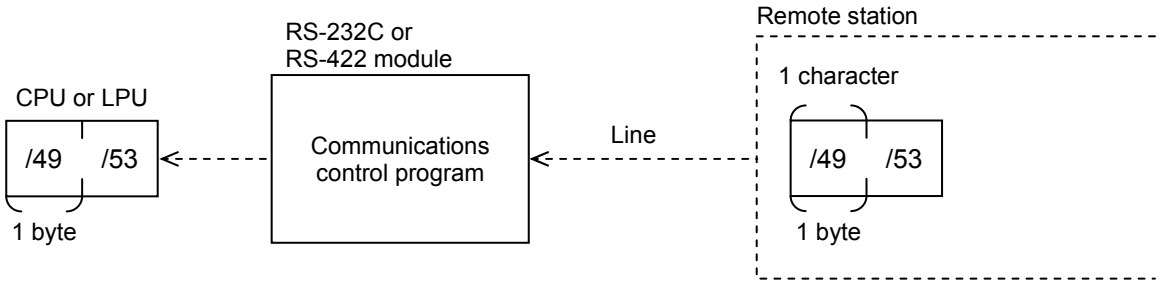
In the example shown above, if data /41 ('A'), /42 ('B'), /43 ('C'), and /44 ('D') is received from the remote station, the communications control program will perform an ASCII to binary conversion on the data to pass data /AB and /CD to the CPU or LPU. A remote station program must be created to meet this rule.

- Binary specification

  With the binary specification, there is no need for the remote station program to perform a data conversion between ASCII and binary.



In the example shown above, if data /31 and /39 is transmitted from the CPU or LPU, the communications control program will pass the data on to the remote station as it is.



In the example shown above, if data /49 and /53 is received from the remote station, the communications control program will pass the data on to the CPU or LPU as it is.

■ Text size

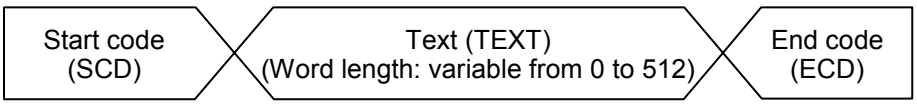
Specify the word length of text data between 0 and 512.

Table 4-6 Text Sizes

Tool selection item		Text size content	Default
Setting	Display		
0	No text	No text	
1 to 512	001 to 512 [BYTE]	1 to 512 (bytes)	256
–	Text variable length	Text variable length (up to 512 bytes)	

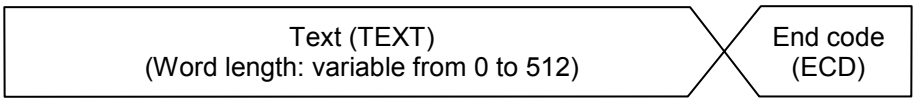
Text data starts with the data next to an SCD received if available and ends when an ECD is encountered or data has been received in a specified word length.
Hence, a text word length, and an SCD and an ECD can be specified to allow various forms of blocks to be transmitted and received.
With the ASCII specification, the communications control program converts transmitted from binary to ASCII and converts received data from ASCII to binary.
With the text variable length specification, a receive timeout period specification is mandatory.

- If an SCD and an ECD are available



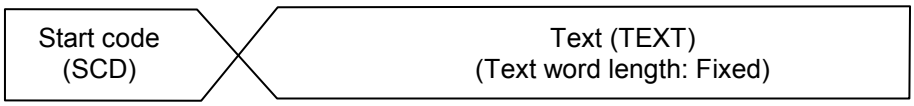
In the case above, if users set an ECD in the text even though the text length is 512, the communications control program will stop transmitting or receiving the text as soon as the ECD is encountered. When an ECD does not exist, the communications control program assume a text length of 512, and an SCD before the text and an ECD after.

- If an ECD is available



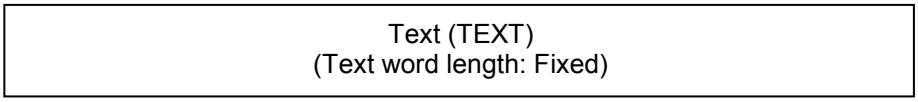
In this case as well, users may set an ECD in the text to let the communications control program to handle the text as being variable-length.

- If an SCD is available



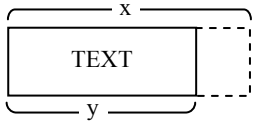
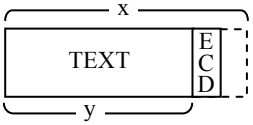
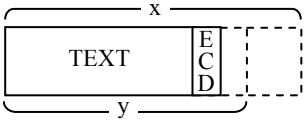
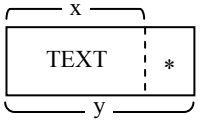
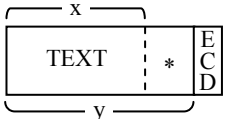
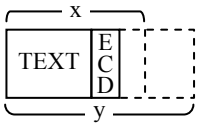
In this case, the text is fixed in the text word length specified.

- If only text is available



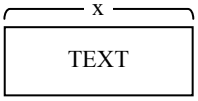
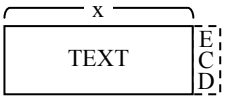
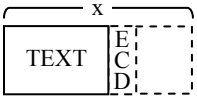
The table gives the relationship between the text word length specified by LGB and the word length transmitted by the send handler. The word length transmitted by the send handler is assumed x bytes, and the text word length specified by LGB is assumed y bytes.

Table 4-7 Data Transmitted Over the Line (1)

Size relation	LGB-specified end code	End code embedded in the text	Data transmitted over the line	
x ≥ y	No	–		Text is transmitted in the LGB-specified word length.
	Yes	No		LGB-specified text + ECD (end code) are transmitted.
		Yes		Text is transmitted, from the beginning to the embedded ECD (end code).
x < y	No	–	 * Previous data in the send buffer	Text + previous data in the send buffer are transmitted.
	Yes	No	 * Previous data in the send buffer	Text + previous data in the send buffer + ECD (end code) are transmitted.
		Yes		Text is transmitted, from the beginning to the embedded ECD (end code).

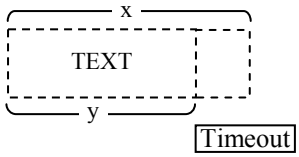
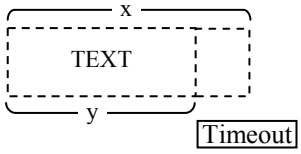
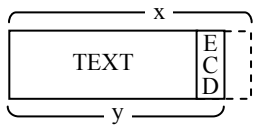
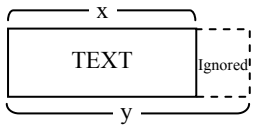
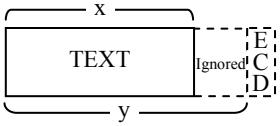
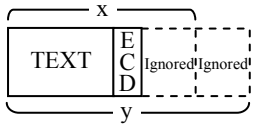
With the text variable length specification, the following kinds of data are transmitted over the line depending on the word length transmitted by the send handler.

Table 4-8 Data Transmitted Over the Line (2)

LGB-specified end code	End code embedded in the text	Data transmitted over the line	
No	–		Text is transmitted in the transmitted word length specified by the send handler.
Yes	No		Text is transmitted in the transmitted word length specified by the send handler, + ECD (end code).
	Yes		Text is transmitted, from the beginning to the embedded ECD (end code).

The table gives the relationship between the text word length specified by LGB and the word length incorporated by the receive handler. The text word length specified by LGB is assumed x bytes, and the data word length from the line is assumed y bytes.

Table 4-9 Data Stored in the Received Data Buffer

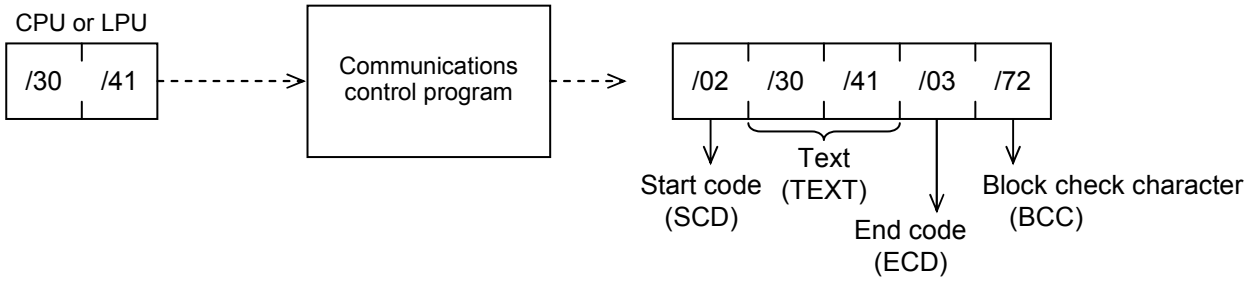
Size relation	LGB-specified end code	End code embedded in the text	Data stored in the received data buffer	
x > y	No	–		A text timeout occurs while waiting for received data until the text word length specified by LGB is reached. When a text variable length is set, however, the text data that has been received prior to a timeout is stored in the received data buffer.
	Yes	No		
		Yes		Text is received, from the beginning to the embedded ECD (end code).
x ≤ y	No	–		Text is received only in the word length specified by LGB, with extra data being ignored.
	Yes	No		
		Yes		Text is received, from the beginning to the embedded ECD (end code).

The table below gives the relationship between the receive buffer for the receive handler and received data.
The word length incorporated into the receive buffer is assumed x bytes, and the word length stored in the receive buffer is assumed y bytes.

Table 4-10 Data Incorporated into a User-Specified Area

Size relation	Data incorporated into a user-specified area
x ≥ y	(Receive buffer) y Received data Received data area 0 — 0 x Cleared to 0
x < y	(Receive buffer) y Received data Received data area x Uncollected data is ignored for the computing function handler. For the ask handler, see "5.3.2 Subroutines."

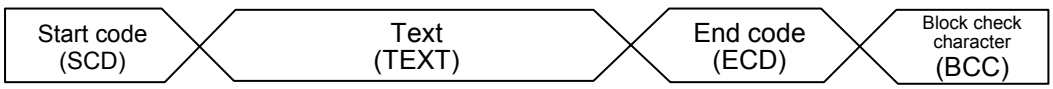
The communications control program attaches a start code (SCD), an end code (ECD), and a block check character (BCC) code to the data to be transmitted according to LGB table specifications.



The example shown above assumes:

Start code: One character specified/02 (STX: Start of Text)
End code: One character specified/03 (ETX: End of Text)
Block check character: Horizontal even parity

It is necessary, therefore, to create a remote station program to meet these LGB table specifications.
A typical block structure is shown below.



■ Start code (SCD)

Signifies the beginning of text. "Available" or "Unavailable," and, if "Available" is specified, a code length (1 to 4 characters) and code data can be set.

Table 4-11 Start Code

Tool selection item			Start code content	Default	
No.	Menu display	Start code display			
1	No start code	No start code	No start code	√	
2	One start code	CD1	One start code		CD=/02(STX)
3	Two start codes	CD1+CD2	Two start codes		
4	Three start codes	CD1+CD2+CD3	Three start codes		
5	Four start codes	CD1+CD2+CD3+CD4	Four start codes		

CD1 to 4: Hexadecimal values that denote start codes /00 to /FF

- If an SCD is available, the communications control program recognizes an incoming SCD as a sign of starting reception from the remote station, ignoring all data that had had been received earlier.
  When the communications control program transmits text to the remote station, it prefixes the text with an SCD code.
  The SCD is not converted to ASCII even when ASCII is specified.

■ End code (ECD)

Signifies the end of text. Available or unavailable, and, if available is specified, a code length (1 to 4 characters) and code data can be set.

Table 4-12 End Code

Tool selection item			End code content	Default	
No.	Menu display	End code display			
1	No end code	No start code	No end code		
2	One end code	CD1	One end code	√	CD=/03(STX)
3	Two end codes	CD1+CD2	Two end codes		
4	Three end codes	CD1+CD2+CD3	Three end codes		
5	Four end codes	CD1+CD2+CD3+CD4	Four end codes		

CD1 to 4: Hexadecimal values that denote end codes /00 to /FF

- If an ECD is available, the communications control program recognizes an incoming ECD as a sign of ending reception from the remote station.
  When the communications control program transmits text to the remote station, it postfixes the text with the specified ECD code.
  The ECD is not converted to ASCII even when ASCII is specified.

■ Block check character (BCC)

Used to check a transmitted or received frame for validity. If an ECD is available, a BCC exists next to the ECD; if an ECD is Unavailable, it exists next to the text.

Table 4-13 Block Check Character

Tool selection item		BCC content	Default
No.	Menu display		
1	No BCC	No BCC	√
2	Even parity check	Horizontal even parity check	
3	Odd parity check	Horizontal odd parity check	

As for the BCC check, “Available” or “Unavailable,” and, if “Available” is specified, horizontal even or horizontal odd parity can be specified.

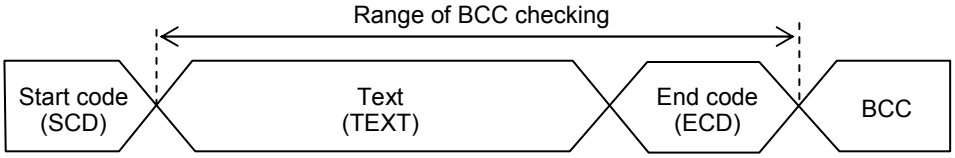
Horizontal even parity··· $(BCC)_E = (0x00)\ EOR\ (\sum_{i=0}^{n} EOR\ D_i)$

Horizontal odd parity··· $(BCC)_{07} = (0x7F)\ EOR\ (\sum_{i=0}^{n} EOR\ D_i)$ (7 data bits)

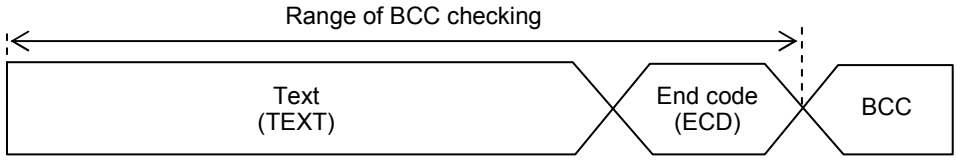
$(BCC)_{08} = (0xFF)\ EOR\ (\sum_{i=0}^{n} EOR\ D_i)$ (8 data bits)

The possible ranges of BCC checking are shown below.

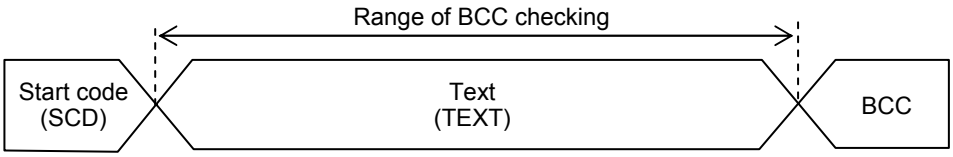
- If an SCD and an ECD are available



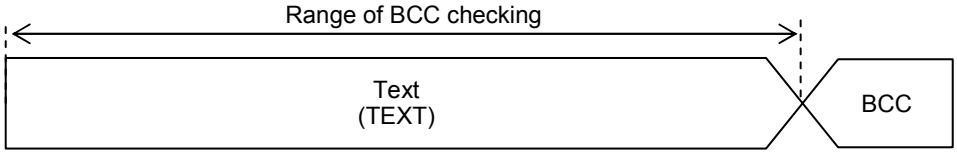
- If an ECD is available



- If an SCD is available



- If only text is available



If an ECD is Unavailable, the text is checked by assuming that it has a fixed length of data as specified.

Create a remote station program to meet these rules if a BCC checking is available.

NOTICE
With the ASCII specification, the text (binary data) before its ASCII conversion and the ECD are checked.

■ Send delay time

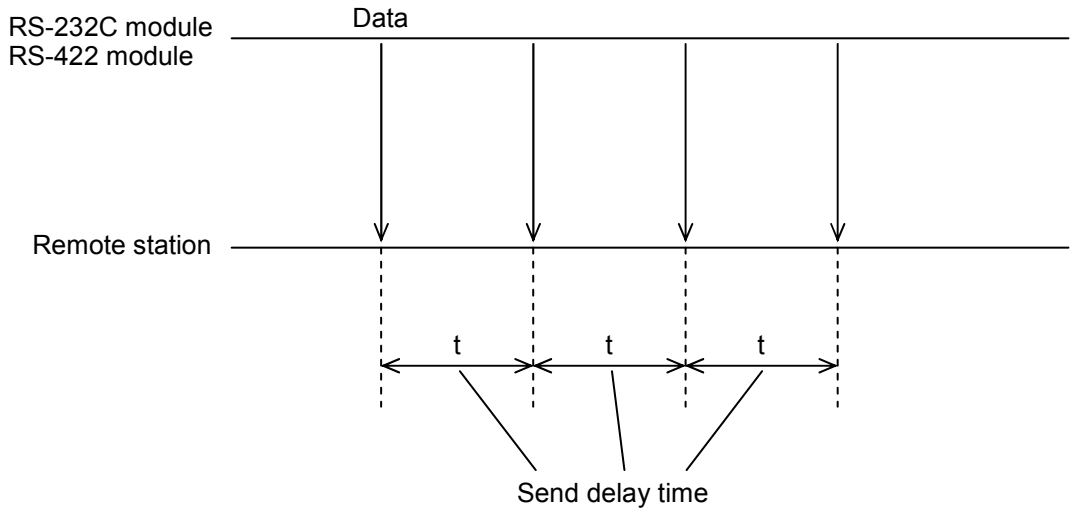
When the communications control program transmits data to the remote station, it specifies the time interval between two successive instances of data transmission.

Table 4-14 Send Delay Times

Tool setting		Send delay time content	Default
Setting	Display		
0	No data send delay time	No data send delay time	√
1 to 32767	00001 to 32767 [ms]	1 to 32767 [ms]	

Table 4-15 Send Delay Time Setting Ranges

Baud rate	Send delay time setting range
300 [BPS]	64 to 32767 [ms]
600 [BPS]	32 to 32767 [ms]
1200 [BPS]	16 to 32767 [ms]
2400 [BPS]	8 to 32767 [ms]
4800 [BPS]	4 to 32767 [ms]
9600 [BPS]	2 to 32767 [ms]
19200 [BPS]	1 to 32767 [ms]



■ Send break and continue codes

Use the send break and continue codes to allow the remote station to request the ongoing transmission of text from the communications control program to break or continue (for reasons such as inability to handle the incoming data).

Table 4-16 Send Break and Continue Codes

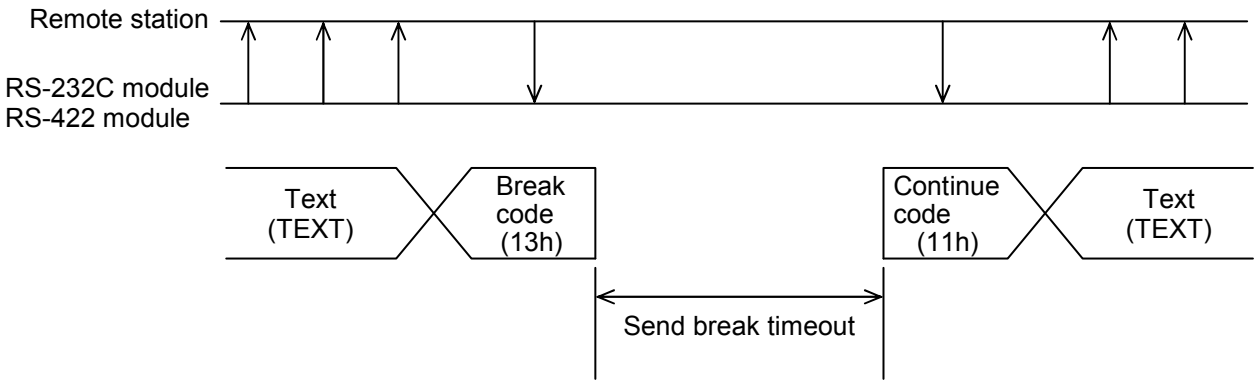
Tool selection item			Break and continue code content	Default
No.	Menu display	Break and continue code display		
1	No break and continue code	No break and continue code	No break and continue code	√
2	One break code, one continue code	BR: CD1 CN: CD2	One break code, one continue code	
3	One break code, two continue codes	BR: CD1 CN: CD2+CD3	One break code, two continue codes	
4	Two break codes, one continue codes	BR: CD1+CD2 CN: CD3	Two break codes, one continue codes	
5	Two break codes, two continue codes	BR: CD1+CD2 CN: CD3+CD4	Two break codes, two continue codes	

BR: Break code CN: Continue code
CD1 to 4: Hexadecimal values that denote send break and continue codes /00 to /FF

"Available" or "Unavailable" for send break and continue handling and, if "Available" is specified, break codes (1 to 2 characters) and continue codes (1 to 2 characters) can be specified.

Once the communications control program receives a break code, it receives only a continue code and ignores any other incoming code. Both break and continue codes are used without conversion even if they are in ASCII.

A send break timeout refers to the period of time that that expires before the communications control program receives a continue code after receiving a break code. An error would occur if this period is exceeded.



The example shown above assumes:

Break code: One character specified/13 (CD3: Device control 3 [X-OFF])

Continue code: One character specified/11 (DC1: Device code 3 [X-ON])

Because the communications control program monitors independently the send break timeout and the receive timeout defined later, an error would still occur if the receive monitoring time is exceeded even while transmission has been interrupted.

■ Send break timeout

Table 4-17 Send Break Timeout

Tool setting		Send break timeout content	Default
Setting	Display		
0	No text send break timeout monitoring	No text send break timeout monitoring	32767
1 to 32767	00001 to 32767 [100 ms]	0.1 to 3276.7 [s]	

■ Receive timeout

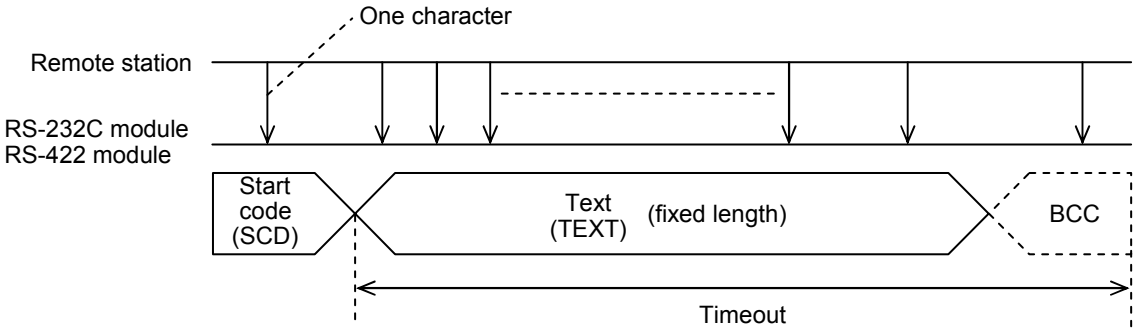
Define the period of time for the communications control program to monitor while receiving text.

Table 4-18 Receive Timeout

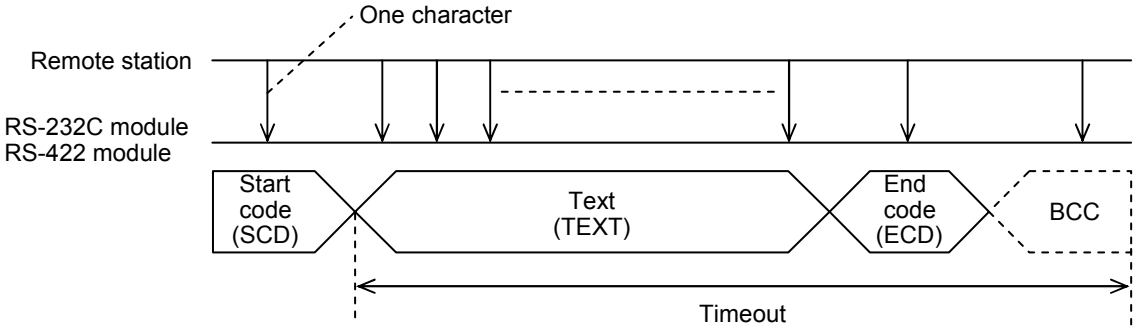
Tool setting		Receive timeout setting	Default
Setting	Display		
0	No text receive timeout monitoring	No text receive timeout monitoring	32767
1 to 32767	00001 to 32767	0.1 to 3276.7 [s]	

- If a text fixed length (0 to 512) has been set

  The period of time that expires from the moment at which text reception starts to the moment at which a specified size of text is received or an end code (if a BCC is specified, the BCC) is encountered is set as a receive timeout.

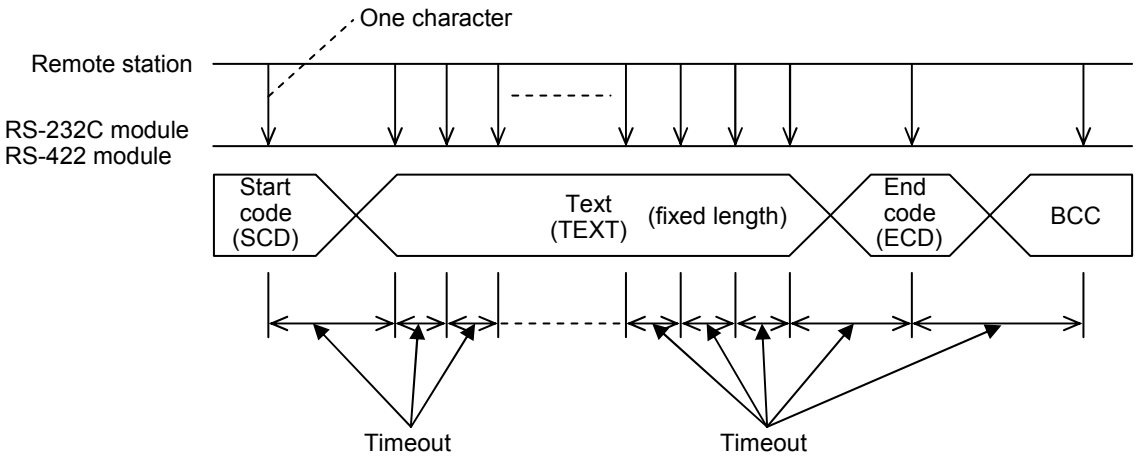


  Encountering an end code before receiving a specified size of text



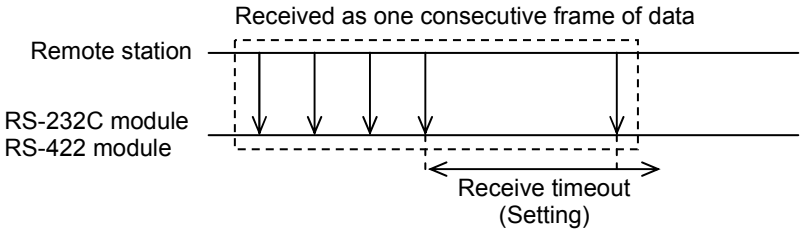
- If a text variable length has been set

  The period of time that expires between two characters is set as a receive timeout.
  If the next batch of data is not received during this period, the communications control program assumes the end of data and turns on the receive-complete flag.



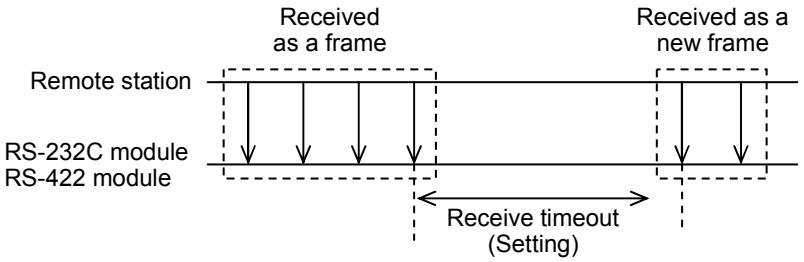
• Example of receiving text with a text variable length (1)

Text received within the receive timeout is treated as one consecutive frame of data.



• Example of receiving text with a text variable length (2)

Text received past the receive timeout is treated as a new frame.



■ RS-422 gate control

RS-422 gate control is not used. Its setting would be ignored.

■ Request to Send (RS)

Specify Request to Send (RS pin status) available/unavailable output to the remote station.
The RS-232C module can transmit send data only if "RS available" is specified.
This setting is invalid on the RS-422 module.

Table 4-19 Request to Send (RS)

Tool selection item		RS content	Default
No.	Display		
1	RS available output	RS available output	√
2	RS unavailable output	RS unavailable output	

- RS available
  The RS-232C module continues transmitting constant RS available output to the remote station while becoming ready to transmit.
- RS unavailable
  The RS-232C module continues transmitting constant RS unavailable output to the remote station while becoming not ready to transmit.

  If send data is transmitted when "RS available" is specified, the transmitted data would be kept from being transmitted and the ready-to-send flag in the CPU or LPU system registers (see "5.2.1 Transmitted information") would remain to indicate "Now transmitting."

○ Set "RS available" when transmitting data to the remote station.
○ Set "RS unavailable" when not transmitting data to the remote station.
  If the remote station supports a ready-to-receive/not-ready-to-receive switching function, connect the RS pin of the RS-232C module and the ready-to-receive/not-ready-to-receive detection pin of the remote station (typically, the CD pin) to prevent false reception of invalid data (such as noise) other than transmitted data.

■ Equipment Ready (ER)

Specify RS-232C module ready or not ready output to the remote station.
Ready and not ready definitions are determined by the protocol between the RS-232C module and the remote station. Generally, the state of the RS-232C module being ready to receive is defined as being "ready."
This setting is inblide on the RS-422 module.

Table 4-20 Equipment Ready (ER)

Tool selection item		ER content	Default
No.	Display		
1	Not Ready output	Not Ready output	
2	Ready output	Ready output	√

- If ready
  The RS-232C module continues transmitting the Ready state to the remote station via the Equipment Ready pin.
- If not ready
  The RS-232C module continues transmitting the Not Ready state to the remote station via the Equipment Ready pin.

○ Set “Ready” when receiving data from the remote station.
○ Set “Not Ready” when not receiving data from the remote station.
If the remote station supports a ready-to-send/not-ready-to-send switching function, connect the ER pin of the RS-232C module and the ready-to-send/not-ready-to-send detection pint of the remote station (typically, the DR or CS pin) to control the remote station into the ready or not ready to transmit state.

■ Data Set Ready (DR)

Specify whether to check the remote station for its ready state (DR pin status).
This setting is invalid on the RS-422 module.

Table 4-21 Data Set Ready

Tool selection item		Data set ready content	Default
No.	Display		
1	Checking unavailable	Checking unavailable	√
2	Checking available	Checking available	

- If checking is available
  The communications control program checks the remote station for its ready state (DR pin status) and transmits data only when it is ready. Transmitting data while the remote station is not ready would invoke an error.

- If checking unavailable
  The communications control program transmits data to the remote station without checking it for its ready state (DR pin status).

○ If the remote station supports a ready to receive output function
  Connect the DR pin of the RS-232C module and the ready-to-receive output pin of the remote station (typically, the ER pin) to enable checking.
○ If the remote station does not support a ready to receive output function
  Set to disable checking.

■ Control signal automatic control

Specify whether to carry out control signal input checking and output control automatically or manually.
This setting is invalid on the RS-422 module.

Table 4-22 Control Signal Automatic Control

Tool selection item		Setting content	Default
No.	Display		
1	Manual setting	Control signal manual setting	√
2	Automatic control	Control signal automatic control	

- Manual setting
  The communications control program behaves as directed by the settings of Request to Send (RS), Equipment Ready (ER), and Data Set Ready (DR).
- Automatic control
  "RS available" is transmitted only at transmitted data transmission.
  Equipment Ready (ER) turns to ready after startup and turns to not ready when a hardware error occurs. Equipment Ready (ER), Carrier Detect (CD), and Clear to Send (CS) are all set to enable input checking.

■ System selection

Use the module switch to select a computing function system or a task system.
Register a computing function or set a task.

- Computing function

  With the S10mini, register a computing function to transmit data to and from the remote station.

  With the S10V, computing functions are preset and there are no additional functions to set.

Table 4-23 Registering Computing Functions

Name	Function
SD0	Channel No. 0 send computing function
SD1	Channel No. 1 send computing function
SD2	Channel No. 2 send computing function
SD3	Channel No. 3 send computing function
RV0	Channel No. 0 receive computing function
RV1	Channel No. 1 receive computing function
RV2	Channel No. 2 receive computing function
RV3	Channel No. 3 receive computing function

- Task system

  Specify a user task to be started from the OS on the CPU or LPU when the RS-232C/422 module has received data.

  Task system does not start when specifying 0 to the started task number.

Table 4-24 Task Systems

Item	Setting range		Remarks
	S10mini	S10V	
Started task number	0 to 127	0 to 255 (*)	Integer input
Start cause	0 to 16	0 to 32	Integer input

(*) In S10V, task system does not start even if specifying the started task number because task number 230 to 255 have been reserved.

NOTICE
With the S10V series, a CMU module is needed to use task systems (C mode).

4.4 Host Interrupts

If the H-7338 protocol is set in the RS-422 module (LQE565), interrupts can be raised in the host computer.

4.4.1 Host interrupt register

Whenever the start of a Z-coil is detected, a 150 ms interrupt is raised in the host computer. Though this routine is carried out at the same interval as a sequence cycle, it is not synchronized with a sequence cycle. Table 4-25 lists the Z-coils associated with channel numbers.

Table 4-25 Host Interrupt Register

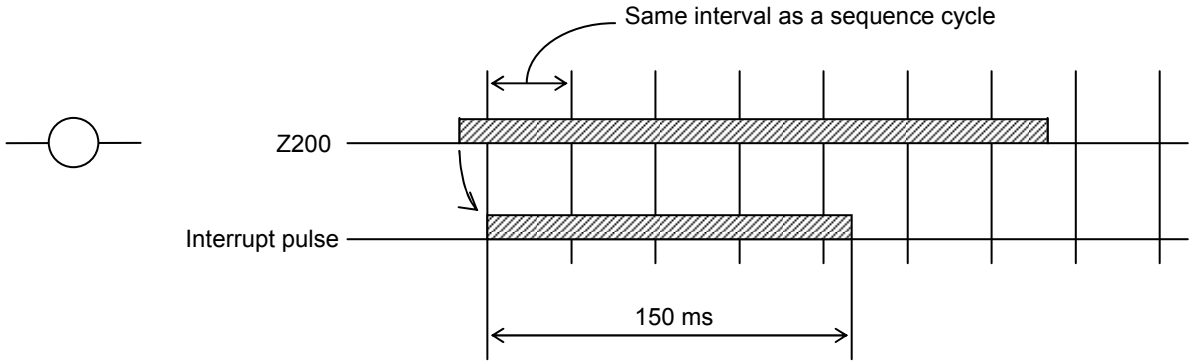
Channel number	Host interrupt register
Channel 0	Z200
Channel 1	Z201
Channel 2	Z202
Channel 3	Z203

NOTICE

- Using with S10/2α
  When using an optional adapter (LWZ800) to connect the RS-422 module (LQE565) to an S10/2α unit, use the RS-422 module (LQE565) on other than channel 0 if CPU module RS-422 host interrupt is to be used.
  If the RS-422 module (LQE565) is used on channel 0, Z200 would be duplicated between the CPU module and the RS-422 module as a host interrupt register, causing both host interrupts to turn on.
- Using with the S10mini
  No constraint is placed on the host interrupt from channel 0 of the RS-422 module (LQE565). The S10mini CPU module does not support host interrupt.
- Using with the S10V
  No constraint is placed on the host interrupt from channel 0 of the RS-422 module (LQE565). The S10V CPU module uses Z204 as a host interrupt register.

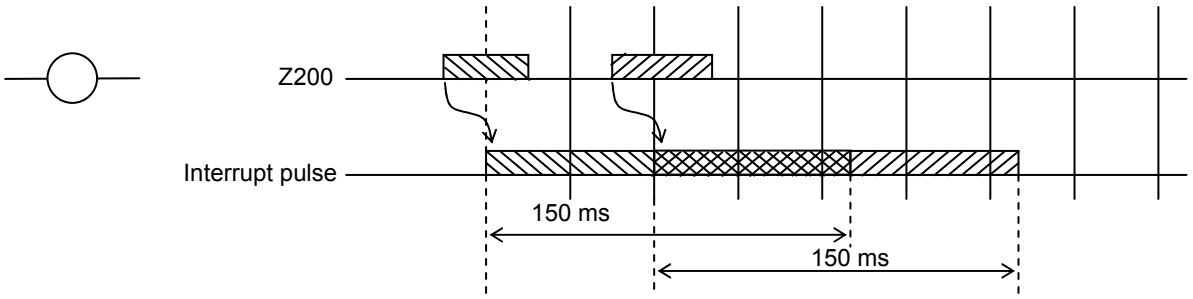
An example of raising an interrupt in the host computer where the RS-422 module (LQE565) is used on channel 0 is shown below.
Settling pulse width: Minimum one sequence cycle

Timing chart



[Stretched interrupt pulse width]
If Z200 makes two times or more on to off transitions within a period of 150 ms, double interrupts would arise, resulting a stretched interrupt pulse width.

Timing chart



4.5 Installing and Starting Up the System

4.5.1 Installing

(1) Installing the S10mini external serial link system

To install the S10mini external serial link system, you must execute the setup program that is stored in the S10mini external serial link system DISK1 folder on the CD.

Double-click "setup.exe" that is stored in the DISK1 folder on the S10mini external serial link system CD. Since no window opens upon completion of installation, attach a shortcut to the desktop as needed.

Click the Start button and choose [(All) Programs] – [Hitachi S10] – [EXTERNAL SERIAL LINK SYSTEM] – [EXTERNAL SERIAL LINK SYSTEM] from the [Start] menu on the Windows® screen. Click and hold the right mouse button on the [EXTERNAL SERIAL LINK SYSTEM] and move the pointer to the desktop. Then, choose [Copy Here] from the pop-up menu.

NOTICE

Before installing the S10mini external serial link system, be sure to exit all the currently open Windows® programs. Do not forget to exit anti-virus software and other memory-resident programs. If you install the S10mini external serial link system without exiting such programs, an error may occur during installation. If such an error occurs, first uninstall the external serial link system as directed in "4.5.2 Uninstalling," exit all the Windows® programs, and then install the S10mini external serial link system again.

(2) Installing the S10V external serial link system

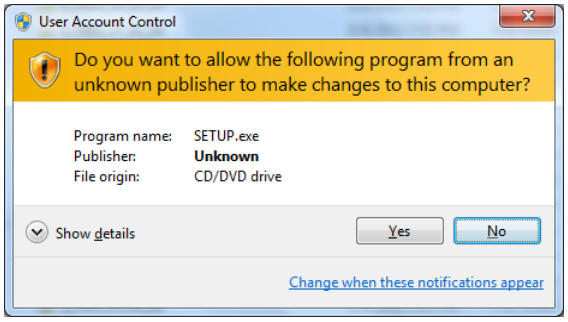
To install the S10V external serial link system, you must execute the setup program that is stored in the S10V external serial link system DISK1 folder on the CD.

Double-click "setup.exe" that is stored in the DISK1 folder on the S10V external serial link system CD. Since no window opens upon completion of installation, attach a shortcut to the desktop as needed.

Click the Start button and choose [(All) Programs] – [Hitachi S10V] – [S10V EXTERNAL SERIAL LINK SYSTEM] – [S10V EXTERNAL SERIAL LINK SYSTEM] from the [Start] menu on the Windows® screen. Click and hold the right mouse button on the [S10V EXTERNAL SERIAL LINK SYSTEM] and move the pointer to the desktop. Then, choose [Copy Here] from the pop-up menu.

<Notes on installing in Windows® 7 (32-bit)>

Installing the S10V external serial link system in Windows® 7 (32-bit) operating system requires prior logging onto the operating system with an appropriate Administrator account, which is the Administrator account first created in the initial condition of your personal computer. When you have so logged on, you can then double-click "setup.exe" that is stored in the DISK 1 folder on the S10V external serial link System CD. When "setup.exe" is started, the dialog box as shown below will appear. Click the Yes button to continue the execution of the setup program.



The S10V external serial link system cannot be installed on a per-user basis. To install the S10V external serial link system successfully, the user must first log onto the operating system with an appropriate Administrator account, which is the Administrator account first created in the initial condition of your personal computer.
The S10V external serial link system may not be installed properly in any of the following cases: 1) administrator permission is acquired by using User Account Control(*) with a standard user account and 2) logon is made with an Administrator account that has been created using User Account Control with a standard user account.
If you make a logon with a user account that is different from the one you have used for the installation of the S10V external serial link system, the installed program may be missing from the program menu displayed. In this case, you should perform the following series of steps: 1) make a logon again with the Administrator account first created in the initial condition of your personal computer; 2) uninstall the installed program; and 3) install the program again. When you want to create a new account, be sure to make a logon with an Administrator account. Do not use User Account Control at that time.

(*) User Account Control is a Microsoft Windows feature that temporarily grants administrative rights to standard user accounts.

A message reporting a read-only file detected may be displayed during the reinstallation of the S10V external serial link system. In this case, click the Yes button to set off overwriting.

NOTICE

- The S10V basic system is required for operating the S10V external serial link system. If it is not installed, you cannot install the S10V external serial link system.
- Before installing the S10V external serial link system, be sure to exit all the currently open Windows® programs. Do not forget to exit anti-virus software and other memory-resident programs. If you install the S10V external serial link system without exiting such programs, an error may occur during installation. If such an error occurs, first uninstall the S10V external serial link system as directed in "4.5.2 Uninstalling," exit all the Windows® Programs, and then install the S10V external serial link system again.

4.5.2 Uninstalling

The existing S10V external serial link system needs to be uninstalled when, for instance, you want to upgrade it. The procedure required for uninstalling it is as follows:

(1) Uninstalling from Windows® 2000
Click on [Start] button on your Windows desktop and choose [Settings] – [Control Panel]. When the Control Panel opens, double-click on [Add/Remove Programs]. Then, choose "EXTERNAL SERIAL LINK SYSTEM" (for S10mini controllers) or "S10V EXTERNAL SERIAL LINK SYSTEM" (for S10V controllers) in the [Change or Remove Programs] tab and click the [Change/Remove] button. When the [Confirm File Deletion] dialog box appears, click the [Yes] button.

(2) Uninstalling from Windows® XP
Click on [Start] button on your Windows desktop and choose ([Settings] –)[Control Panel]. When the Control Panel opens, double-click on [Add/Remove Programs]. Then, choose "EXTERNAL SERIAL LINK SYSTEM" (for S10mini controllers) or "S10V EXTERNAL SERIAL LINK SYSTEM" (for S10V controllers) in the [Change or Remove Programs] tab and click the [Change/Remove] button. When the [Confirm File Deletion] dialog box appears, click the [Yes] button.

(3) Uninstalling from Windows® 7 (32-bit) -- for S10V controllers only
Click on [Start] button on your Windows desktop and choose [Control Panel]. When the Control Panel opens, click [Programs and features]. Then, select "S10V EXTERNAL SERIAL LINK SYSTEM" and click [Uninstall/Change] button. When the [Confirm File Deletion] dialog box appears, click the [Yes] button.

NOTICE

- If Windows® opens a window during the uninstall process to display the question “Remove Shared File?,” click the [No] button to retain shared files.
- When you want to reinstall the external serial link system, be sure to perform an uninstall and then perform an install.

4.5.3 Starting up the system

To start up the external serial link system, perform the following procedure:

- S10mini external serial link system startup procedure
  (1) If you want to start up the S10mini external serial link system from the Windows® desktop, double-click the [EXTERNAL SERIAL LINK SYSTEM] icon. Alternatively, if you want to start it up from the [Start] button, choose [(All) Programs] – [Hitachi S10] – [EXTERNAL SERIAL LINK SYSTEM] from the Start menu.
  (2) The [EXTERNAL SERIAL LINK SYSTEM] window appears. (See Figure 4-1.) Then, click a desired command button.

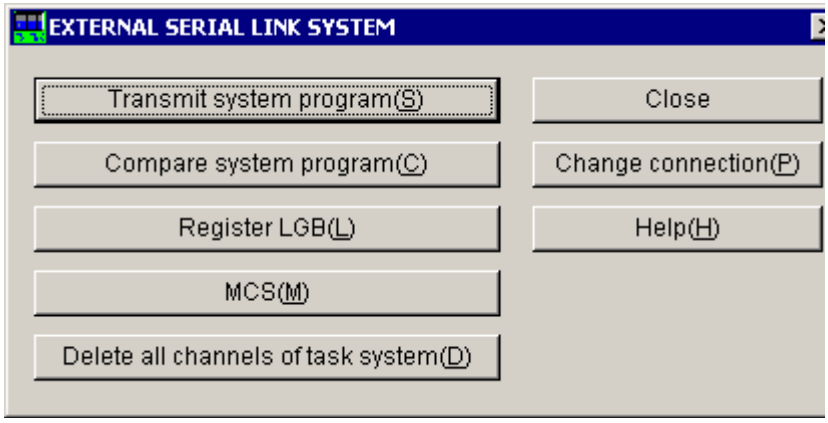


Figure 4-1 [EXTERNAL SERIAL LINK SYSTEM] Window

- S10V external serial link system startup procedure to start it up in online mode
  (1) If you want to start up the S10V external serial link system from the Windows® desktop, double-click the [S10V EXTERNAL SERIAL LINK SYSTEM] icon. Alternatively, if you want to start it up from the [Start] button, choose [(All) Programs] – [Hitachi S10V] – [S10V EXTERNAL SERIAL LINK SYSTEM] – [S10V EXTERNAL SERIAL LINK SYSTEM] from the Start menu.
  The [[S10V] EXTERNAL SERIAL LINK SYSTEM] window will then appear. At this stage of the procedure, the external serial link system is not connected with the PCs yet.

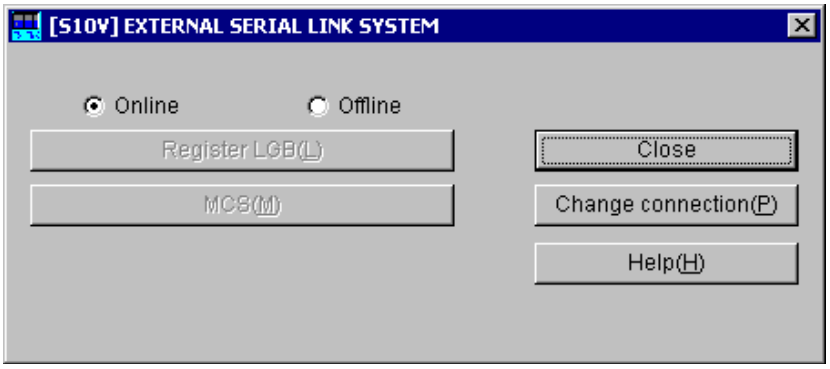


Figure 4-2 [[S10V] EXTERNAL SERIAL LINK SYSTEM] Window

(2) By clicking the Change connection button in the window, display the [Communication type] window on-screen. When the [Communication type] window appears, specify the desired destination of connection and click the OK button. (See "4.5.4 PCs connection change" for details on the communication type.) If you need not change the current connection destination setting, click the Cancel button instead.

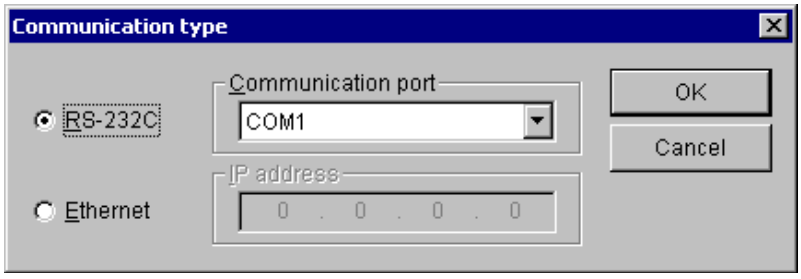


Figure 4-3 [Communication type] Window

(3) The [[S10V] EXTERNAL SERIAL LINK SYSTEM] window appears. Then, click a desired command button.

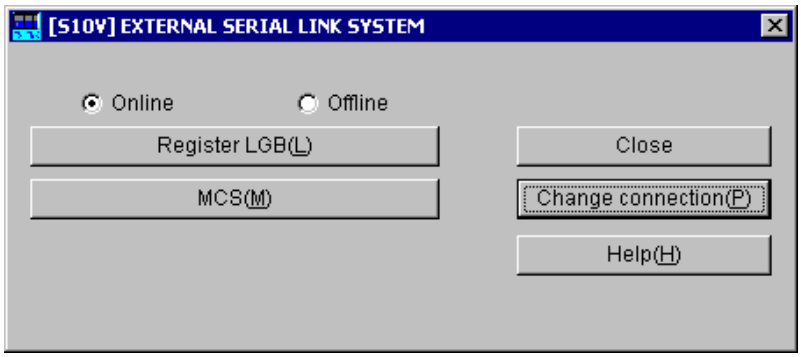


Figure 4-4 [[S10V] EXTERNAL SERIAL LINK SYSTEM] Window

- S10V external serial link system startup procedure to start it up in offline mode
  The procedure described below enables you to create a setup information file for the external serial link and edit it, all in offline mode, even if the actual target machine is not present in your user system. To send the setup information file prepared this way to the target machine, first put the external serial link system into online mode, then read the file in by clicking the Load button in the [Enter LGB] window, and finally send it by clicking the Write button. (For details on LGB registration, see "4.5.5 LGB registration.")

  (1) Take the same action as specified in Step (1) under "● S10V external serial link system startup procedure to start it up in online mode." The external serial link system's main menu will then appear on screen.

  (2) Choose the [Offline] radio button. The Change connection button is then replaced by the Edition file select button.

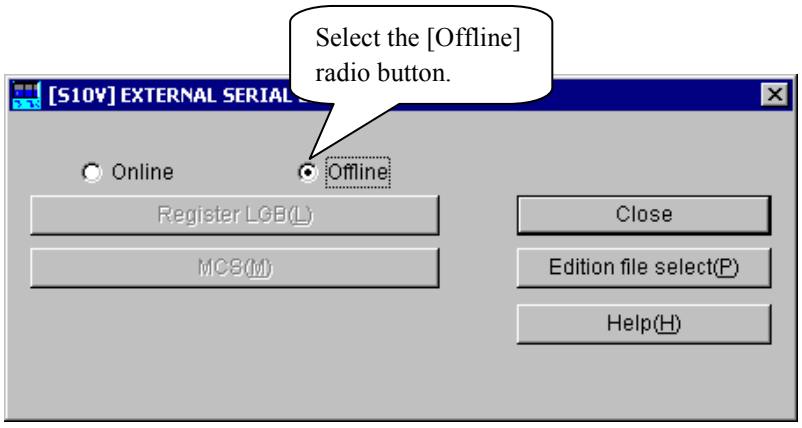


Figure 4-5 Selecting the [Offline] Radio Button

(3) Click the Edition file select button and choose the desired external serial link setup information file you want to edit in offline mode.

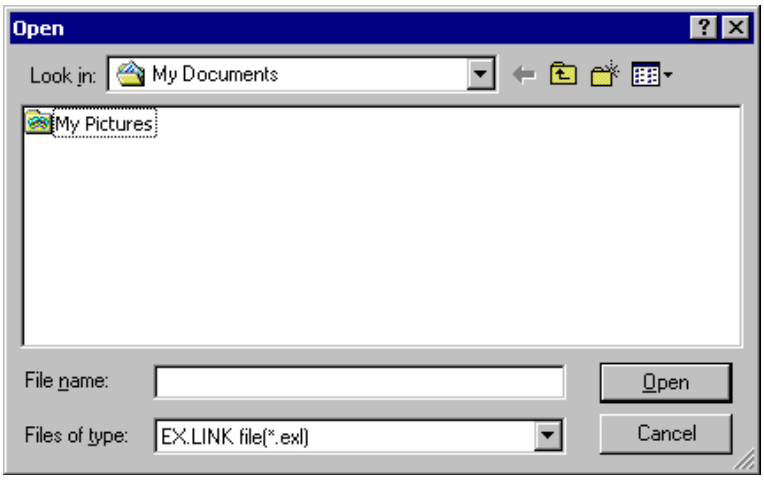
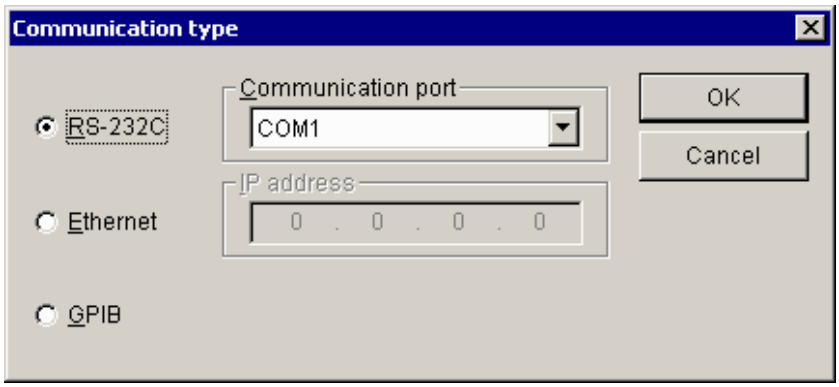


Figure 4-6 Selecting the Edition File You Want to Edit

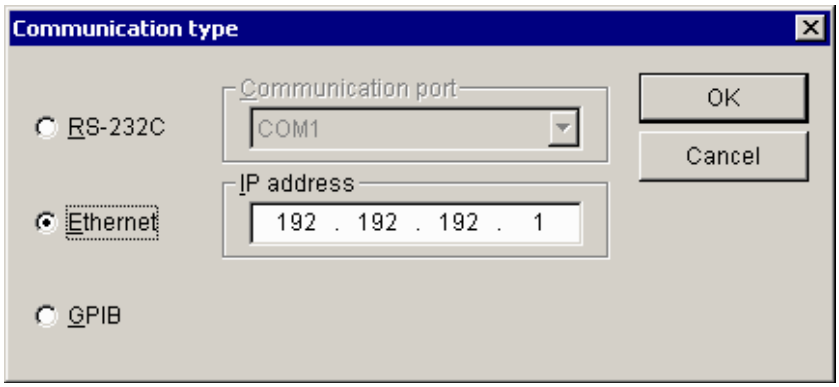
4.5.4 PCs connection change

Function: Sets the PCs-to-personal computer communication type.
Procedure: The procedure is described below.

(1) In the [EXTERNAL SERIAL LINK SYSTEM] or [[S10V] EXTERNAL SERIAL LINK SYSTEM] window, click the Change connection button.
(2) The [Communication type] window then opens the [S10V] EXTERNAL SERIAL LINK SYSTEM does not have a GPIB button.
(3) For RS-232C communication, click "RS-232C" and then select a "Communication port."



(4) For Ethernet communication, click "Ethernet" and then enter the connection destination "IP address."



NOTICE

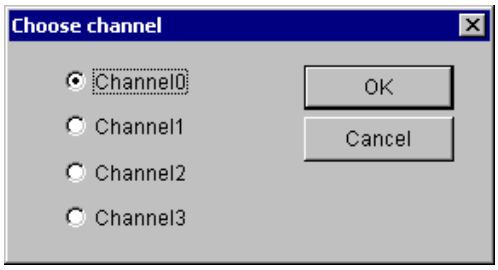
- The S10mini Series does not support GP-IB. When connecting it to a personal computer, select either RS-232C or Ethernet.
- The S10V external serial link window does not have a GPIB button.

(5) After completion of setup, click the OK button. To abort the setup process, click the Cancel button.

4.5.5 LGB registration

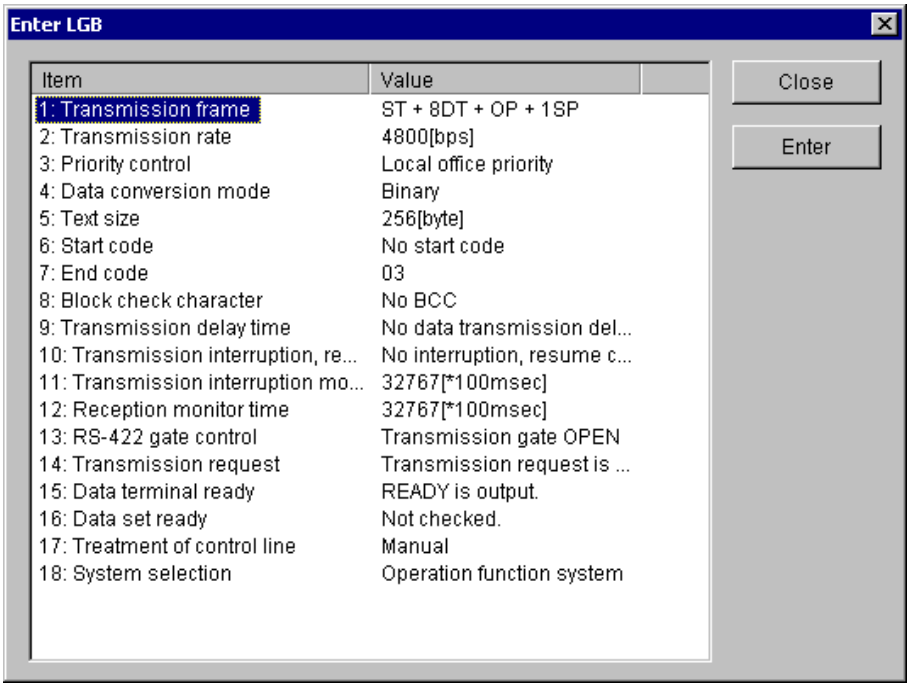
Function: The function of LGB registration is to enable you to edit the LGB table. For details on the LGB table, see "4.2 Editing the LGB Table" and "4.3 LGB Table Settings."
Operation: The operating procedure used is described below.

(1) Click the Register LGB button in either the [EXTERNAL SERIAL LINK SYSTEM] window or the [[S10V] EXTERNAL SERIAL LINK SYSTEM] window.
(2) The [Choose channel] window will appear as long as the S10mini or S10V controller is running in online mode. Choose the desired channel for which you want to edit the LGB table. Then, click the OK button. If you need not edit the LGB table, click the Cancel button to quit this procedure.

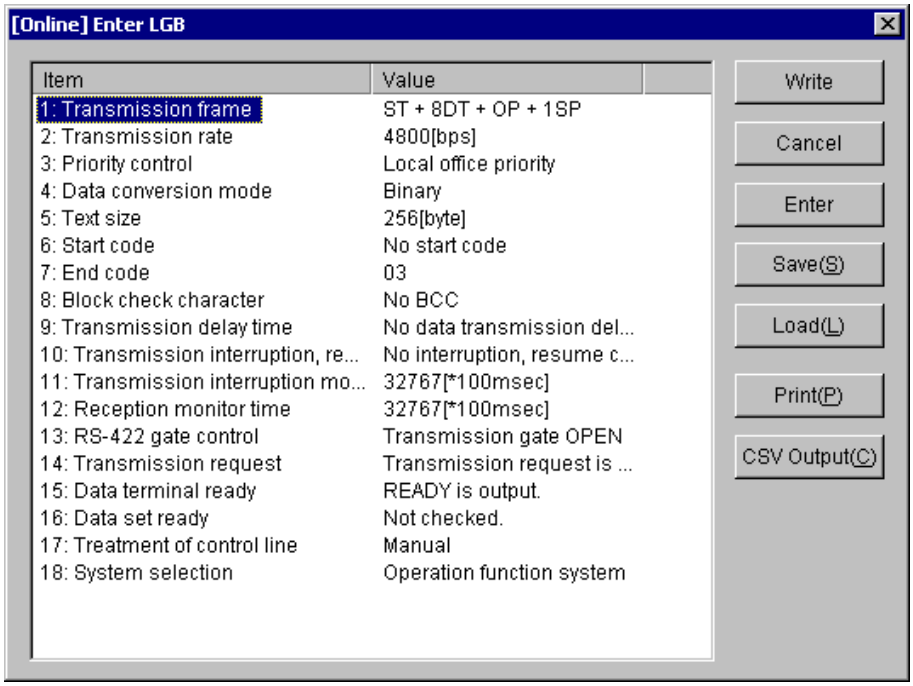


(3) The [Enter LGB] window as shown below will appear. In this window, either double-click the desired item or choose it and click the Enter button. The change-of-settings window for the selected item will then appear to allow you to change its setting in the LGB table. Change the setting according to the information provided in "4.3 LGB Table Settings."

- S10mini version of the [Enter LGB] window

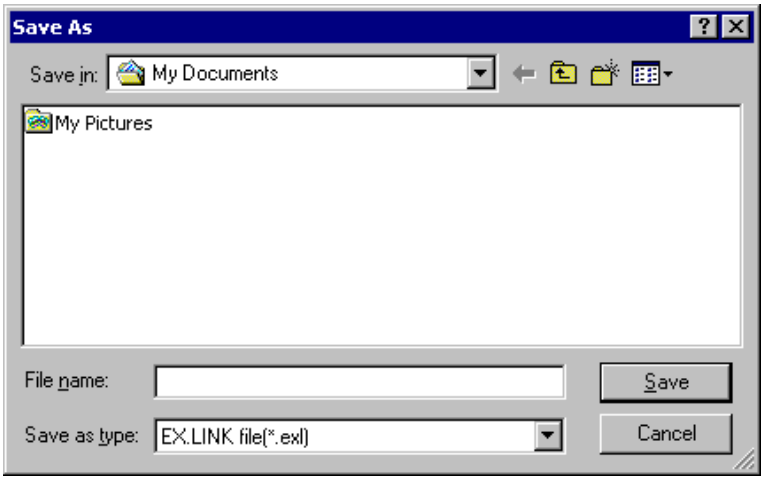


- S10V version of the [Enter LGB] window

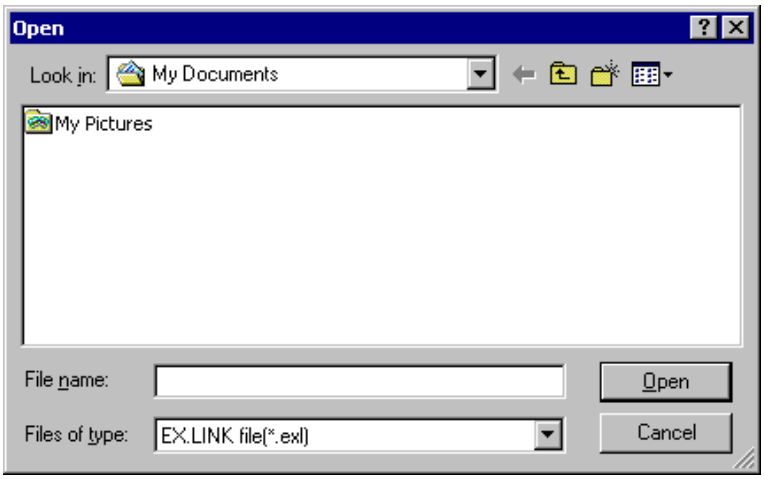


(4) When the change of the existing setting(s) is complete, click the Write or Close button. This reflects the added change in the RS-232C or RS-422 module. If the added change need not be reflected in them, click the Cancel button or the ✕ button in the upper right corner of the window instead.

(5) The updated settings can be saved in a file (only when the S10V controller is running in online mode). To accomplish this, click the Save button. The [Save As] window will then appear. In this window, specify the desired file name and click the Save button. If you need not save them, click the Cancel button instead.



(6) The saved settings in the file can be read in (only when the S10V controller is running in online mode). To accomplish this, click the Load button. The [Open] window will then appear. In this window, specify its file name and click the Open button to read the file in. If you need not read it in, click the Cancel button instead.



(7) The updated settings can be printed or saved in a file in CSV format (only when the S10V controller is used). For details, see “4.5.7 Printing” and “4.5.8 CSV output.”

4.5.6 Selecting an edition file

Function: The function of this action is to choose a file you want to edit in offline mode. The files that you can choose for editing are those setup info files which have been saved in online mode or have been prepared through offline editing. You can also create a new file by specifying a non-existing file name. This function is supported only in S10V controller systems.

Operation: The procedure used is shown below.

(1) In the [[S10V] EXTERNAL SERIAL LINK] window displayed, choose the [Offline] radio button. If it is already selected, skip this step.

(2) If you have not selected an edition file yet or want to change the currently selected edition file, click the [Edition File Select] button. The [Open] window as shown below will then appear.

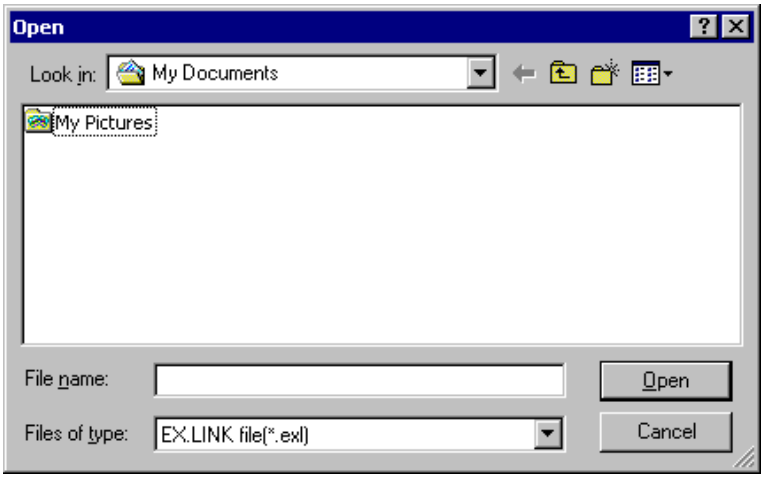


Figure 4-7 [Open] Window -- an Example

(3) Choose one of the setup info files with external serial link information in them displayed in the [Open] window and click the [Open] button. If a non-existent file name is specified instead of choosing an existing file and the [Open] button is clicked, the [Creation confirmation] dialog box as shown below will appear. In this case, choose between task system and operation function system in the dialog box, and click the [OK] button.

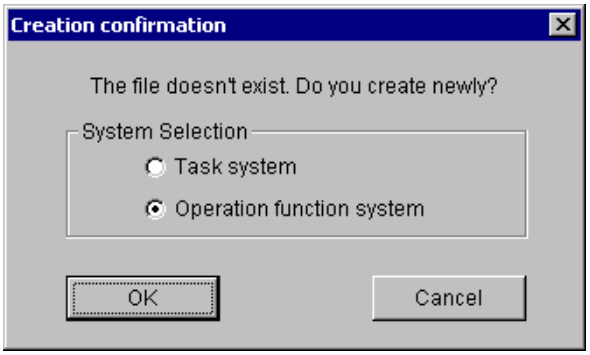


Figure 4-8 [Creation confirmation] Dialog Box

If the selected file is a non-external serial link setup info file or invalid file, the error message dialog box shown below will appear.

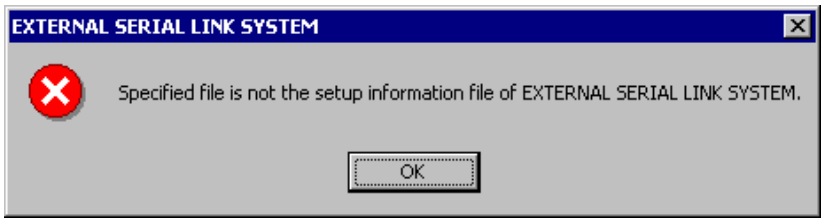


Figure 4-9 Error Message Dialog Box Reporting on an Invalid External Serial Link Setup Info File Specified

If the selected file is a valid external serial link setup info file, you can now edit the setup information in that file as you do while the external serial link system is running in online mode.

4.5.7 Printing

Function: The function of commands used for this purpose is to print the setup information under editing on the printer. This function is supported only in S10V controller systems.
Operation: The procedure used is shown below.

(1) If the external serial link system is running in online mode, establish a connection between the external serial link system and the PCs. (See "4.5.4 PCs connection change.") If it is running in offline mode, choose the desired edition file. (See "4.5.6 Selecting an edition file.")
(2) Click the Print button in the [[Online] Enter LGB] window.
(3) The [Print] dialog box appears. In this dialog box, specify the desired printer and its properties, and then click the OK button.

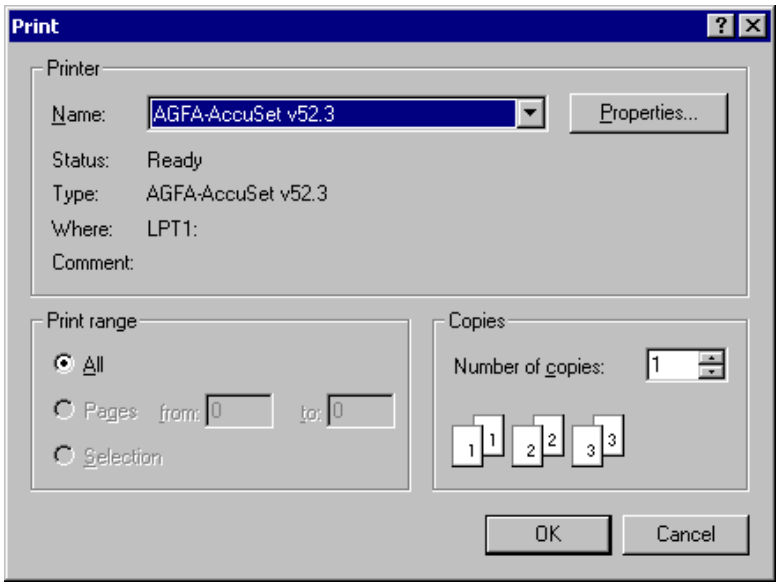


Figure 4-10 [Print] Dialog Box

<Sample printout>

```
ExLink    Module Info     2006/10/31     17:37:33
File name=C:\Documents and Settings\Administrator\Desktop\Nai\aaa\Settings\ex_task.exl

Channel                                     0
1: Transmission frame                       ST + 8DT + OP + 2SP
2: Transmission rate                        19200[bps]
3: Priority control                         No priority (full-duplex operation)
4: Data conversion mode                     Binary
5: Text size                                Flexible text size
6: Start code                               01+02+03+04
7: End code                                 11+12+13+14
8: Block check character                    Odd parity check
9: Transmission delay time                  32767[msec]
10: Transmission interruption, resume code  BR:21+22 CN:23+24
11: Transmission interruption monitor time  32767[*100msec]
12: Reception monitor time                  32767[msec]
13: RS-422 gate control                     Transmission gate CONTROL
14: Transmission request                    Transmission request is not output.
15: Data terminal ready                     READY is output.
16: Data set ready                          Checked.
17: Treatment of control line               Auto
18: System selection                        Task system
Initiation task number                      1
Initiation factor                           0
```

4.5.8 CSV output

Function: The function of this command is to output the setup information under editing to a file in CSV format. This function is supported only in S10V controller systems.

Operation: The procedure used is shown below.

(1) If the external serial link system is running in online mode, establish a connection between the external serial link system and the PCs. (See "4.5.4 PCs connection change.") If it is running in offline mode, choose the desired edition file. (See "4.5.6 Selecting an edition file.")

(2) Click the CSV Output button in the [[Online] Enter LGB] window.

(3) The [Save As] dialog box appears. In this dialog box, specify the desired folder and file to which you want to output the setup information, and then click the Save button.

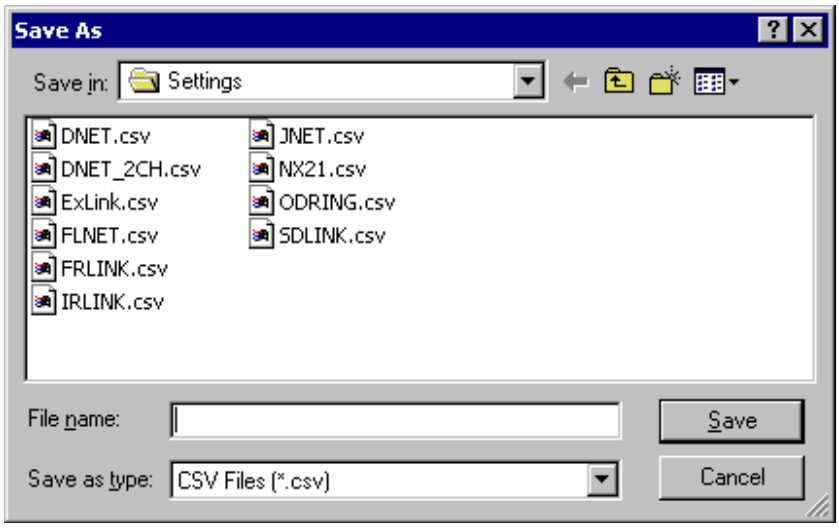


Figure 4-11 [Save As] Dialog Box

<Sample CSV file output>

ExLink Module Info 2006/10/31 17:38:16
File name=C:\Documents and Settings\Administrator\Desktop\Nai\aaa\Settings\ex_task.exl

Channel,0
1: Transmission frame, ST + 8DT + OP + 2SP
2: Transmission rate, 19200[bps]
3: Priority control, No priority (full-duplex operation)
4: Data conversion mode, Binary
5: Text size, Flexible text size
6: Start code, 01+02+03+04
7: End code, 11+12+13+14
8: Block check character, Odd parity check
9: Transmission delay time, 32767[msec]
10: Transmission interruption, resume code, BR:21+22 CN:23+24
11: Transmission interruption monitor time, 32767[*100msec]
12: Reception monitor time, 32767[msec]
13: RS-422 gate control, Transmission gate CONTROL
14: Transmission request, Transmission request is not output.
15: Data terminal ready, READY is output.
16: Data set ready, Checked.
17: Treatment of control line, Auto
18: System selection, Task system
Initiation task number, 1
Initiation factor, 0

4.5.9 Shutting down the system

In the [EXTERNAL SERIAL LINK SYSTEM] or [[S10V] EXTERNAL SERIAL LINK SYSTEM] window, click the [×] button or [Close] button.

5 PROGRAMMING

5.1 Software Configuration

Communication with an external station is carried out in the following way:

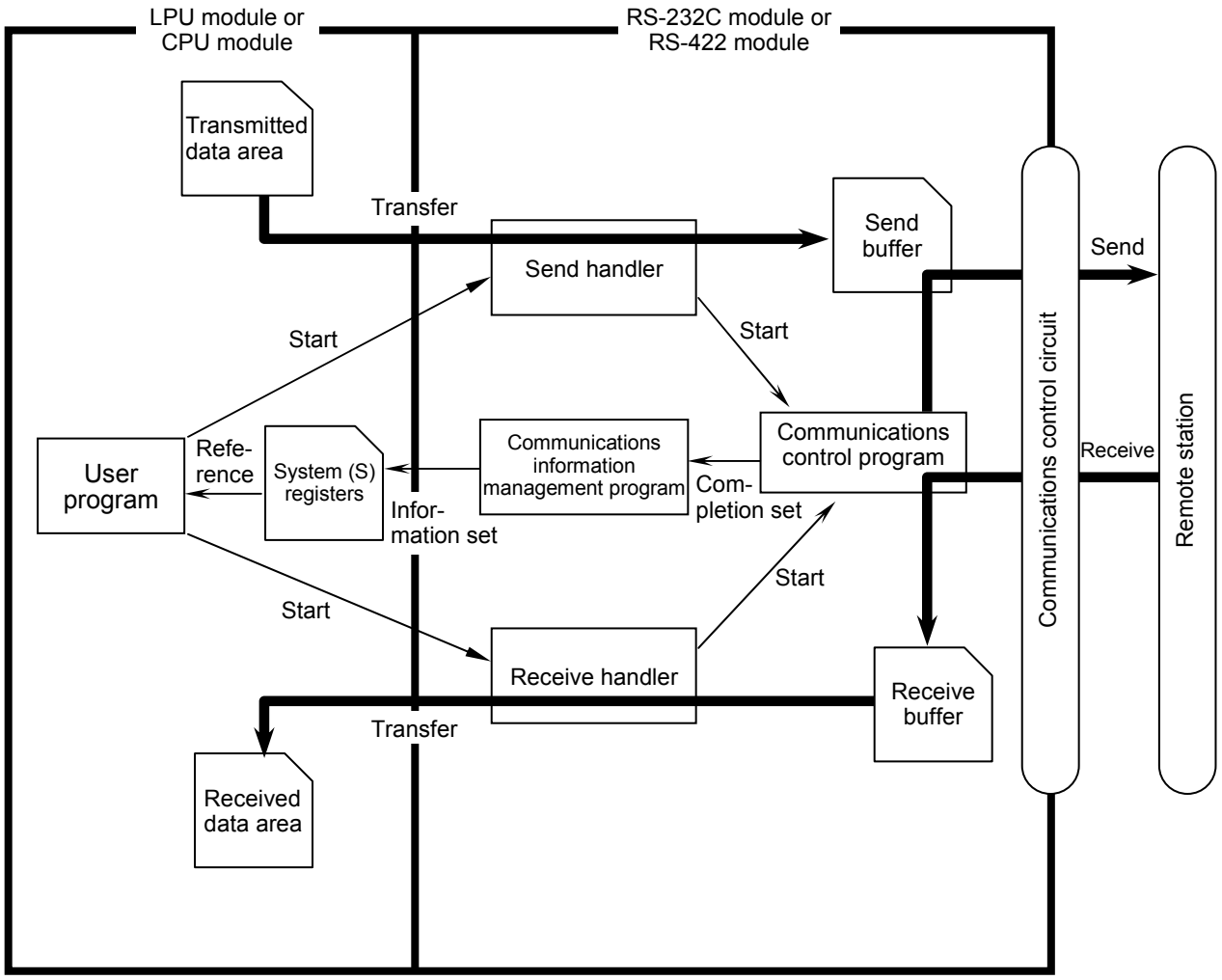


Figure 5-1 Software Configuration

- Communications control program

  On receiving a transmission request from the send handler, the communications control program transmits send data over the line. When it has finished receiving all data from the line, it notifies the communications information control program of the completion of reception.

- Communications information management program

  The communications information management program loads information on transmission and reception by the communications control program into the CPU or LPU system (S) registers to notify the user program. (See "5.2.1 Transmitted information," and Subsection "5.2.2 Received information.")

- Send handler

  The send handler provides the following functions:

  • Transfer transmitted data in a specified word length from the transmitted data area specified by the user program to the send buffer in the RS-232C module.
  • Issue a transmissions start request to the communications control program.
  • Make various error checks on launching transmission

  Use the user program to launch the send handler at the desired timing of transmission.

- Receive handler

  The receive handler provides the following functions:

  • If yet-to-be-incorporated received data is available in the receive buffer, transfer data in a specified word length to the received data area specified by the user program.
  • Notify the communications control program that the receive buffer has now been emptied.
  • Make various error checks on launching reception.

    The user program must recognize the end of reception before launching the receive handler to incorporate the data. (For information on recognizing the end of reception, see "5.2.2 Received information.")

- System registers

  Contain information, such as whether transmission enabled, reception complete, and errors encountered. Create a user program by referencing this information.

5.2 System Registers

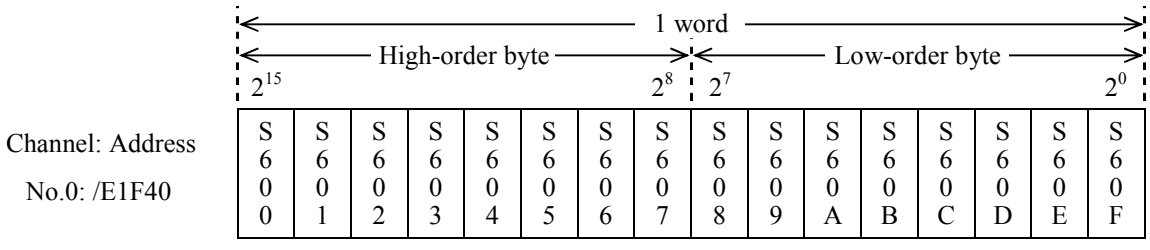
5.2.1 Transmitted information

While the send handler executes transmission from the CPU or LPU to a remote station, transmission information is loaded into the CPU or LPU system (S) registers.
The user program references the S-registers to determine whether transmission is enabled or disabled, and identify send errors encountered.

Table 5-1 Transmitted Information

Channel				Explanation	Bit definition	
No.0	No.1	No.2	No.3		0	1
S600	S620	S5C0	S5E0	Send-enabled flag	Transmission enabled	Transmitting now
S601	S621	S5C1	S5E1	Handler error flag	No error	Error encountered
S602	S622	S5C2	S5E2	Communications control program error	No error	Error encountered
S603	S623	S5C3	S5E3	Reception canceled transmission	No error	Error encountered
S604	S624	S5C4	S5E4	Not used	Not used	
S605	S625	S5C5	S5E5			
S606	S626	S5C6	S5E6			
S607	S627	S5C7	S5E7			
S608	S628	S5C8	S5E8	Error detail code	Coded representations of the handler and communications control errors See the low-order byte of each error code appearing in “7.4.3 Transmit errors.”	
S609	S629	S5C9	S5E9			
S60A	S62A	S5CA	S5EA			
S60B	S62B	S5CB	S5EB			
S60C	S62C	S5CC	S5EC			
S60D	S62D	S5CD	S5ED			
S60E	S62E	S5CE	S5EE			
S60F	S62F	S5CF	S5EF			

The S-registers are initialized to 0 on a CPU or LPU reset. The S-registers can also be read from the CPU or LPU as word data.



No.1: /E1F44
No.2: /E1F38
No.3: /E1F3C

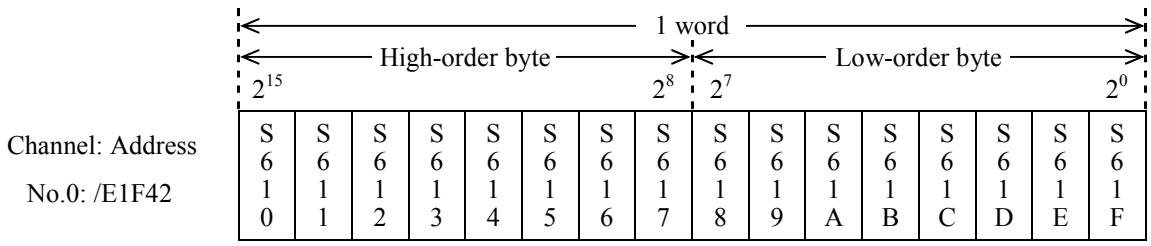
5.2.2 Received information

While the communications control program executes reception by a remote station from the CPU or LPU, successful or unsuccessful reception information is loaded into the CPU or LPU system (S) registers.
The user program references the S-registers to determine whether received data is available and identify receive errors encountered.

Table 5-2 Received Information

Channel				Explanation	Bit definition	
No.0	No.1	No.2	No.3		0	1
S610	S630	S5D0	S5F0	Receive-complete flag	Received data unavailable	Received data available
S611	S631	S5D1	S5F1	Handler error flag	No error	Error encountered
S612	S632	S5D2	S5F2	Communications control program error	No error	Error encountered
S613	S633	S5D3	S5F3	System error	No error	Error encountered
S614	S634	S5D4	S5F4	Handler error code	Handler error description See a high-order bit of the low-order byte of each error code appearing in “7.4.4 Receive errors.”	
S615	S635	S5D5	S5F5			
S616	S636	S5D6	S5F6			
S617	S637	S5D7	S5F7			
S618	S638	S5D8	S5F8	Error detail code	Coded representations of the handler and communications control errors See a low-order bit of each error code appearing in “7.4.4 Receive errors.”	
S619	S639	S5D9	S5F9			
S61A	S63A	S5DA	S5FA			
S61B	S63B	S5DB	S5FB			
S61C	S63C	S5DC	S5FC			
S61D	S63D	S5DD	S5FD			
S61E	S63E	S5DE	S5FE			
S61F	S63F	S5DF	S5FF			

The S-registers are initialized to 0 on a CPU or LPU reset. The S-registers can also be read from the CPU or LPU as word data.



No.1: /E1F46
No.2: /E1F3A
No.3: /E1F3E

5.3 Send/Receive Handlers

Computing functions are used as send/receive handlers when the user program to be invoked is a ladder, or subroutines are used as such when it is a C-mode task.

NOTICE
It is not permitted to define a computing function on channel No. 0 and a subroutine on channel No. 1 or use subroutines for transmission and computing functions for reception. All send/receive handlers associated with a given CPU or LPU unit must be uniformly defined as computing functions or subroutines.

5.3.1 Computing functions

Eight transmit/receive computing functions are available as listed below.

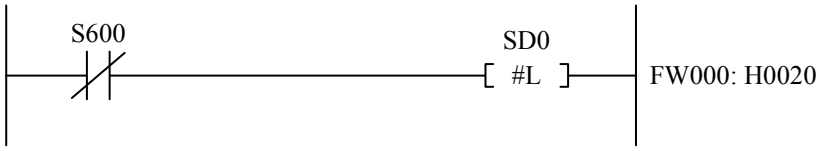
Table 5-3 Send/Receive Computing Functions

Name	Function
SD0	Channel No. 0 send computing function
SD1	Channel No. 1 send computing function
SD2	Channel No. 2 send computing function
SD3	Channel No. 3 send computing function
RV0	Channel No. 0 receive computing function
RV1	Channel No. 1 receive computing function
RV2	Channel No. 2 receive computing function
RV3	Channel No. 3 receive computing function

NOTICE
When using computing functions to build a ladder program with the S10mini, be sure to register them in the LGB table as instructed “4.2 Editing the LGB Table” and “4.3 LGB Table Settings” and then select [Build] – [Receive] from the ladder program edit window in ladder chart system to receive the CPU data with the tool. This will allow the computing functions to be assembled into a ladder program.

SD0, SD1, SD2, SD3 Send computing functions

Function	Transmit data in a specified word length from a parameter-specified area to the remote station.
Parameters	Transfer address: Mnemonic, such as XW000 and FW000 Transfer word length: 1 to 512
Return code	A return code is loaded in a system register. (See "5.2.1 Transmitted information.")
Sample program	Transmits 32 bytes (hexadecimal: /20) from FW000 to the remote station attached to channel No. 0 when input condition S600 is off.



RV0, RV1, RV2, RV3 Receive computing functions

Function

Transfer data in a specified word length to a parameter-specified area. This function performs no action when no received data is available.

Parameters

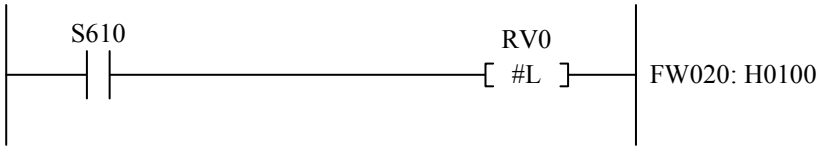
Transfer address: Mnemonic, such as XW000 and FW000
Transfer word length: 1 to 512

Return code

A return code is loaded in a system register.
(See "5.2.2 Received information.")

Sample program

Transfers 256 bytes (hexadecimal: /100) from FW020 to the remote station attached to channel No. 0 when input condition S610 is on.



Remarks

- A receive handler incorporates the oldest received data. It keeps system register S610 from turning off as long as yet-to-be-incorporated received data is available. It turns off system register S610 when yet-to-be-incorporated received data is no longer available.
- A receive computing function can read one block of received data (as stored in one receive buffer) only in a batch (and not in segments).

(Example)

Receive data	Incorporated word length setting on receive handler launch	Incorporated data
'ABCDEF0123' 10 characters received	7	'ABCDEF0'

The receive handler, when relaunched after incorporating seven characters, ignores yet-to-be-incorporated "123" and proceeds to incorporate the next block of received data (data in the next receive buffer).

- If the incorporated word length is set larger than the actual block, the incorporated data is padded with an end code. If space still remains unfilled, /00 is written to the remaining space.

(Example)

Receive data	End code setting	Incorporated word length setting on receive handler launch	Incorporated data
'ABCD' 4 characters received	/03	8	'ABCD' /03000000

5.3.2 Subroutines

Eight send/receive handlers for user-created tasks (application tasks) are available as listed below.

Table 5-4 Send/Receive Handler Names

Name	Address		Function
	S10mini	S10V	
STASK0	/107000	/F46100	Channel No. 0 send subroutine
STASK1	/107006	/F56140	Channel No. 1 send subroutine
STASK2	/107018	/F66180	Channel No. 2 send subroutine
STASK3	/10701E	/F761C0	Channel No. 3 send subroutine
RTASK0	/10700C	/F46120	Channel No. 0 receive subroutine
RTASK1	/107012	/F56160	Channel No. 1 receive subroutine
RTASK2	/107024	/F661A0	Channel No. 2 receive subroutine
RTASK3	/10702A	/F761E0	Channel No. 3 receive subroutine

User-created tasks (application tasks) are created in the C language or in an assembler language (68000 for the S10mini, SH-4 for the S10V).
Because send/receive handlers for application tasks are called by addressing, the application tasks cannot be created (linked) to include send/receive handlers.

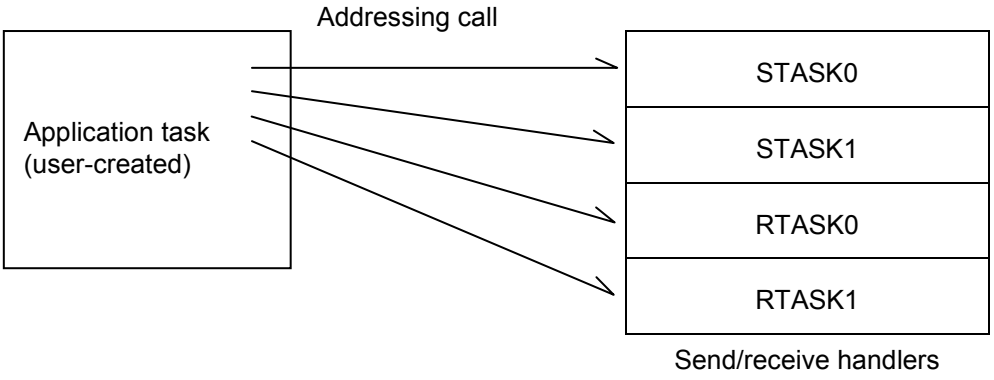


Figure 5-2 Relationship between Application Tasks and Send/Receive Handlers

[S10mini]

STASK0, STASK1, STASK2, STASK3 Send subroutines

Function

Transmit data in a specified word length from a parameter-specified area to the remote station.

Link procedure

(Example)
Channel No. 0

C language [S10mini]	Assembler language [S10mini]
|	|
long (*stask0)( ) ;	|
long rtn, sadr, sbyte ;	move.l #sbyte, -(A7)
|	move.l #sadr, -(A7)
|	lea $107000, A0
stask0 = (long(*)( )) 0x107000 ;	jsr (A0)
|	addq.l #8, A7
|	|
rtn = (*stask0)(sadr, sbyte) ;	|
|	
|	

With an assembler language, the contents of registers other than D0 (return code storage) are guaranteed. (The C language allows users to be unconscious of registers.)

Parameters

sadr: Transmitted data storage address
sbyte: Transmitted byte length
rtn: Return code

With an assembler language, the return code is loaded in the D0 register.

Return code

=0: Normal end
=/FFFFFFFF: Send handler launch error
Error information is loaded in a system register. (See “5.2.1 Transmitted information.”)

Remarks

long (f) ( ); Declares function f that is returned as a function of a pointer to a double-precision integer.

Calling a send subroutine while the module is not available would invoke an error.

[S10mini] Send subroutine examples

C language [S10mini]

Check the most significant bit (send-enabled flag) of transmit system register S600 (address: /E1F40) and, if transmission is enabled, transmit 32 bytes (hexadecimal: /20) from the transmitted data area at address /140000.

```
            :
register    long    (*stask0) (    )    ;
register    long    rtn ;
            :
if ( (*(short*)0xE1F40 & 0x8000) == 0 )
    {
     stask0 = (long(*) ( )) 0x107000 ;
     rtn = (*stask0) (0x140000, 0x20) ;
     if ( rtn != 0 )
        goto errb ;
    }
else
    {
            :
    }
```

Assembler language [S10mini]

Check the most significant bit (send-enabled flag) of transmit system register S600 (address: /E1F40) and, if transmission is enabled, transmit 256 bytes (hexadecimal: /100) from the transmitted data area at address /150000.

```
:
btst        #7, $E1F40
bne         LB1 ──────────→ Go to LB1 if transmission is disabled
move.l      #$100, -(A7)
                  └──────→ 256 bytes of data transferred
move.l      #$150000, -(A7)
                  └──────→ Transmitted data area address /150000
lea         $107000, A0
jsr         (A0)
addq.l      #8, A7
tst.l       D0
bne         ERRB ──────────→ Go to ERRB if an error occurs
:
```

[S10V]

STASK0, STASK1, STASK2, STASK3 Send subroutines

Function

Transmit data in a specified word length from a parameter-specified area to the remote station.

Link procedure

(Example)
Channel No. 0

C language [S10V]	Assembler language [S10V]
|	|
long (*stask0)( ) ;	MOV.L sadr, R4
long rtn, sadr, sbyte ;	MOV.L sbyte, R5
|	MOV.L STASK0_A, R0
|	JSR @R0
stask0 = (long(*)( )) 0xF46100 ;	NOP
|	|
|	sadr: .DATA.L #H'414200
rtn = (*stask0)(sadr, sbyte) ;	(Send address example)
|	sbyte: .DATA.L #H'100
|	(Send size example)
	STASK0_A: .DATA.L #H'F46100
	(Function entry address)

With an assembler language, the contents of registers R8 to R15 are guaranteed. (The C language allows users to be unconscious of registers.)

Parameters

sadr: Transmitted data storage address
sbyte: Transmitted byte length
rtn: Return code

With an assembler language, the return code is loaded in the R0 register.

Return code

=0: Normal end
=/FFFFFFFF: Send handler launch error
Error information is loaded in a system register. (See "5.2.1 Transmitted information.")

Remarks

long (f) ( ); Declares function f that is returned as a function of a pointer to a double-precision integer.

Calling a send subroutine while the module is not available would invoke an error.

[S10V] Send subroutine examples

C language [S10V]

Check the most significant bit (send-enabled flag) of transmit system register S600 (address: /E1F40) and, if transmission is enabled, transmit 32 bytes (hexadecimal: /20) from the transmitted data area at address /414200.

```
            :
register    long    (*stask0) (    )    ;
register    long    rtn ;
            :
if ( (*(short*)0xE1F40 & 0x8000) == 0 )
    {
     stask0 = (long(*) ( )) 0xF46100 ;
     rtn = (*stask0) (0x414200, 0x20) ;
     if ( rtn != 0 )
        goto errb ;
    }
else
    {
            :
    }
```

Assembler language [S10V]

Check the most significant bit (send-enabled flag) of transmit system register S600 (address: /E1F40) and, if transmission is enabled, transmit 256 bytes (hexadecimal: /100) from the transmitted data area at address /414200.

```
                :
                MOV.L       s600,R0
                MOV.W       @R0,R0
                EXTU.W      R0,R0
                MOV.L       sndokflg,R1
                TST         R0,R1
                BF          LB1         → Go to LB1 if transmission is disabled
                MOV.L       sadr,R4
                MOV.L       sbyte,R5
                MOV.L       STASK0_A,R0
                JSR         @R0
                NOP
                TST         R0,R0
                BF          ERRB        → Go to ERRB if an error occurs
                :
s600:           .DATA.L     #H'E1F40
sadr:           .DATA.L     #H'414200   → Transmitted data area address
sbyte:          .DATA.L     #H'100      → Transferred word length
STASK0_A:       .DATA.L     #H'F46100
Sndokflg:       .DATA.L     #H'00008000
                :
```

[S10mini]

RTASK0, RTASK1, RTASK2, RTASK3 Receive subroutines

Function

Transfer data in a specified word length from a parameter-specified area to the remote station. These subroutines perform no action when no received data is available. Receiver handlers (subroutines) incorporate the oldest received data.

Link procedure

(Example)
Channel No. 0

C language [S10mini]	Assembler language [S10mini]
|	|
long (*rtask0)( ) ;	|
long rtn, radr, rbyte ;	move.l #rbyte, -(A7)
|	move.l #radr, -(A7)
|	lea $10700C, A0
rtask0 = (long(*) ( )) 0x10700C ;	jsr (A0)
|	addq.l #8, A7
|	|
rtn = (*rtask0)(radr, rbyte) ;	|
|	
|	

With an assembler language, the contents of registers other than D0 (return code storage) are guaranteed. (The C language allows users to be unconscious of registers.)

Parameters

radr: Transmitted data storage address
rbyte: Transmitted byte length
rtn: Return code

With an assembler language, the return code is loaded in the D0 register.

Return code

=0: Normal end
If there still is yet-to-be-incorporated data, the received data available bit in the system remains set to indicate received data available.
=1: No received data is available in the receive buffer.
=/001A0000: The last data of text is found in the buffer while incorporating received data from it, an end code has been encountered, or data has been incorporated in the text word length that has been specified by LGB.
=/001A00xx: Data with a receive error is found in the buffer while incorporating received data from it. The received data area is cleared to 0 from the data in error to the received byte length. xx in the return code denotes the error code in the low-order byte of the receive error code. (See "7.4.4 Receive errors.") Error information is loaded in a system register. (See "5.2.2 Received information.")
=/FFFFFFFF: Receive handler start error
Error information is loaded in a system register. (See "5.2.2 Received information.")

Calling a send subroutine while the module is not available would invoke an error.

[S10mini] Receive subroutine examples

C language
[S10mini]

Check the most significant bit (receive-complete flag) of receive system register S610 (address: /E1F42) and, if received data is available, transmit 20 bytes (hexadecimal: /14) from the received data buffer to the received data storage area at address /140000.

```
            :
register    long    (*rtask0) ( ) ;
register    long    rtn ;
            :
if ( (*(short*)0xE1F42 & 0x8000) != 0 )
    {
     rtask0 = (long(*) ( )) 0x10700C ;
     rtn = (*rtask0) (0x140000, 0x14) ;
     if ( rtn != 0 )
        goto errb ;
    }
else
    {
            :
    }
```

- According to an application program, a receive C mode subroutine can read one block of received data (as stored in one receive buffer) in segments. (A receive computing function, on the other hand, can only read one block of received data in a batch.)
  An example of the following data received from a remote station is shown below.

  "1234567890" 10 characters received

Incorporated word length setting on receive handler launch	Return code	Incorporated data
3	0 (normal)	"123"
4	0 (normal)	"4567"
4	/001A0000 (normal, block end)	"890" 0

An EOF code (/001A) in the high-order word of the return code (long length) signifies the end of the block. If the low-order word is equal to 0, that block has been normally received; if not, it has been abnormally received. (Data before that had been normally received, though.) The code that designates abnormal reception is the same as in the low-order byte of the receive error code.

If the incorporated word length is set larger than the received word length, the incorporated data is padded with an end code. If space still remains unfilled, 0 is written to the remaining space.
An example of receiving the following data from a remote station is shown below.

"12345" 5 characters received, end code /030001

Incorporated word length setting on receive handler launch	Return code	Incorporated data
7	/001A0000	"12345" /0300
8	/001A0000	"12345" /030001
9	/001A0000	"12345" /03000100

Assembler language [S10mini]

Check the most significant bit (receive-complete flag) of receive system register S610 (address: /E1F42) and, if received data is available, transmit 256 bytes (hexadecimal: /100) from the received data buffer to received data storage area at address /150000.

```
        :
btst    #7, $E1F42
beq     LB1 ──────────→ Go to LB1 if received data is available
move.l  #$100, -(A7)
                  └──→ 256 bytes of data transferred
move.l  #$150000, -(A7)
                  └──→ From data area address /150000
lea     $10700C, A0
jsr     (A0)
addq.l  #8, A7
tst.l   D0
bne     ERRB ──────────→ Go to ERRB if an error occurs
        :
```

[S10V]

RTASK0, RTASK1, RTASK2, RTASK3 Receive subroutines

Function

Transfer data in a specified word length from a parameter-specified area to the remote station. These subroutines perform no action when no received data is available. Receiver handlers (subroutines) incorporate the oldest received data.

Link procedure

(Example)
Channel No. 0

C language [S10V]	Assembler language [S10V]
|	|
long (*rtask0)( ) ;	MOV.L radr, R4
long rtn, radr, rbyte ;	MOV.L rbyte, R5
|	MOV.L RTASK0_A, R0
|	JSR @R0
rtask0 = (long(*) ( )) 0xF46120 ;	NOP
|	|
|	radr: .DATA.L #H'414200
rtn = (*rtask0)(radr, rbyte) ;	(Receive address example)
|	rbyte: .DATA.L #H'08
|	(Receive size example)
	RTASK0_A: .DATA.L #H'F46120
	(Function entry address)

With an assembler language, the contents of registers R8 to R15 are guaranteed. (The C language allows users to be unconscious of registers.)

Parameters

radr: Transmitted data storage address
rbyte: Transmitted byte length
rtn: Return code

With an assembler language, the return code is loaded in the R0 register.

Return code

=0: Normal end
If there still is yet-to-be-incorporated data, the received data available bit in the system remains set to indicate received data available.
=1: No received data is available in the receive buffer.
=/001A0000: The last data of text is found in the buffer while incorporating received data from it, an end code has been encountered, or data has been incorporated in the text word length that has been specified by LGB.
=/001A00xx: Data with a receive error is found in the buffer while incorporating received data from it. The received data area is cleared to 0 from the data in error to the received byte length. xx in the return code denotes the error code in the low-order byte of the receive error code. (See "7.4.4 Receive errors.") Error information is loaded in a system register. (See "5.2.2 Received information.")
=/FFFFFFFF: Receive handler start error
Error information is loaded in a system register. (See "5.2.2 Received information.")

Calling a send subroutine while the module is not available would invoke an error.

[S10V] Receive subroutine examples

C language [S10V]

Check the most significant bit (receive-complete flag) of receive system register S610 (address: /E1F42) and, if received data is available, transmit 20 bytes (hexadecimal: /14) from the received data buffer to the received data storage area at address /414200.

```
            :
register    long    (*rtask0) ( ) ;
register    long    rtn ;
            :
if ( (*(short*)0xE1F42 & 0x8000) != 0 )
     {
      rtask0 = (long(*) ( )) 0xF46120 ;
      rtn = (*rtask0) (0x414200, 0x14) ;
      if ( rtn != 0 )
         goto errb ;
     }
else
     {
            :
     }
```

- According to an application program, a receive C mode subroutine can read one block of received data (as stored in one receive buffer) in segments. (A receive computing function, on the other hand, can only read one block of received data in a batch.)
  An example of the following data received from a remote station is shown below.

  "1234567890" 10 characters received

Incorporated word length setting on receive handler launch	Return code	Incorporated data
3	0 (normal)	"123"
4	0 (normal)	"4567"
4	/001A0000 (normal, block end)	"890" 0

An EOF code (/001A) in the high-order word of the return code (long length) signifies the end of the block. If the low-order word is equal to 0, that block has been normally received; if not, it has been abnormally received. (Data before that had been normally received, though.) The code that designates abnormal reception is the same as in the low-order byte of the receive error code.

If the incorporated word length is set larger than the received word length, the incorporated data is padded with an end code. If space still remains unfilled, 0 is written to the remaining space.
An example of the following data received from a remote station is shown below.

"12345" 5 characters received, end code /030001

Incorporated word length setting on receive handler launch	Return code	Incorporated data
7	/001A0000	"12345" /0300
8	/001A0000	"12345" /030001
9	/001A0000	"12345" /03000100

Assembler language [S10V]

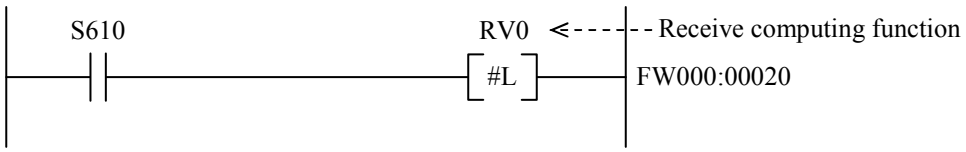
Check the most significant bit (receive-complete flag) of receive system register S610 (address: /E1F42) and, if received data is available, transmit 256 bytes (hexadecimal: /100) from the received data buffer to data storage area at address /414200.

```
                :
                MOV.L     s610,R0
                MOV.W     @R0,R0
                EXTU.W    R0,R0
                MOV.L     rcvokflg,R1
                TST       R0,R1
                BT        LB1      → Go to LB1 if received data is available
                MOV.L     radr,R4
                MOV.L     rbyte,R5
                MOV.L     RTASK0_A,R0
                JSR       @R0
                NOP
                TST       R0,R0
                BF        ERRB     → Go to ERRB if an error occurs
                :
s610:           .DATA.L   #H'E1F42
radr:           .DATA.L   #H'414200  → Received data area address
rbyte:          .DATA.L   #H'100     → Transferred word length
RTASK0_A:       .DATA.L   #H'F46120
rcvokflg:       .DATA.L   #H'00008000
```

5.4 Incorporating Received Data

When the communications control program finished receiving data, relevant information is set in the system (S) registers. Create a user program by referencing this information.

- Creating as a ladder program
  When a receive handler (computing function) is launched to meet the rules of the S-registers, the received data incorporate delay would be confined to within the sequence cycle (standard 30 ms).



- Application task (C mode)
  Create and register the user task that the communications control program launches at the completion of reception.
  This removes the need for the user program to monitor the completion of reception.
  Received data can be incorporated by simply making a subroutine call to a receive handler from a task invoked from the communications control program.

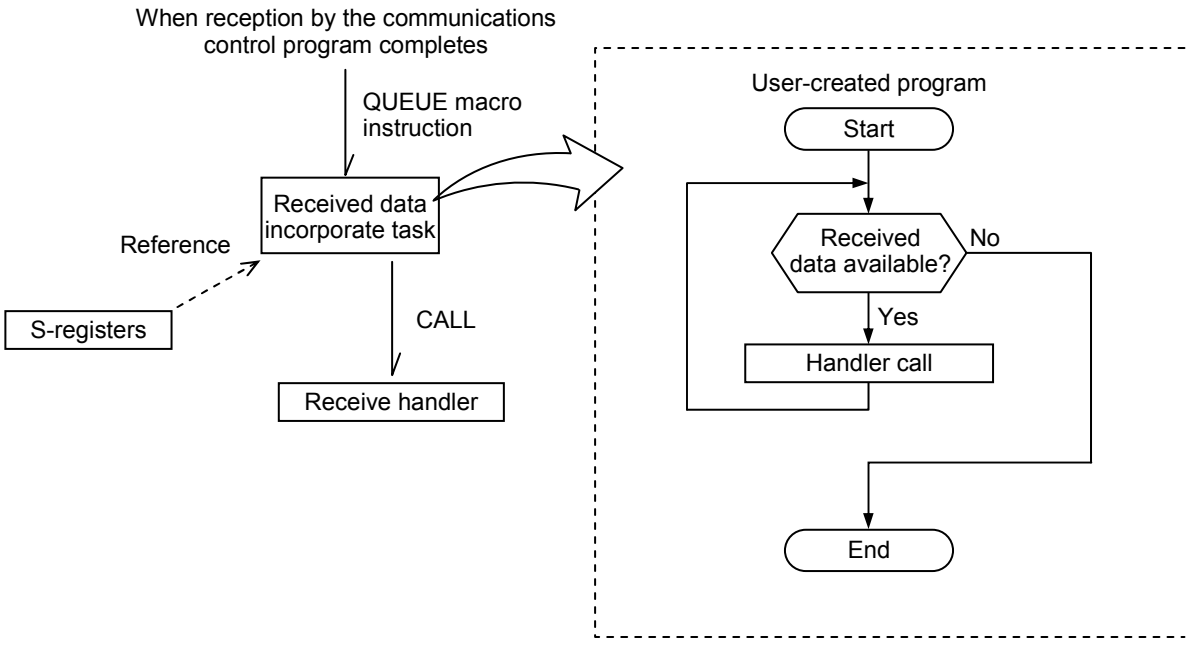


Figure 5-3 Incorporating Received Data with an Application Task (C Mode)

5.5 Hardware Controlled by Software Implementation

The following parameters can be set as send handler parameters to control the hardware of the RS-232C and RS-422 modules:

- Using computing functions
  Send handler name: SD? (? denotes a channel number)
- Using subroutines
  Send handler name: STASK? (? denotes a channel number)

Table 5-5 Hardware Controlled by Software Implementation (1)

Transfer address	Transferred word length	Content	Return information
• Computing function Data register DWFFF • C-mode subroutine Address /62FFE	/8080	Software reset As the module in question is reset, other channels on the same module are reset as well (similar to a reset of the RS-232C/RS-422 module implemented by using the CPU or LPU reset switch)	[S10V] Can be verified by using the tool. [S10mini] A normal ending is evidenced by the display of a successful operation message on the CPU LED. (See "7.4.1 CPU module indicator display messages.")
	/0000 or /0001	Latest hardware status incorporate request	• Channel No. 0 High-order byte of DWFFF (/62FFE) • Channel No. 1 Low-order byte of DWFFF (/62FFF) • Channel No. 2 High-order byte of DWFFF (/62FFC) • Channel No. 3 Low-order byte of DWFFF (/62FFD) 2^7 ... 2^0 ① ② ③ ④ ⑤ ① RS output state: ② CS input state ③ CD input stage → 0: ON(*) 1: OFF(*) ④ ER output state: ⑤ DR input state: → 0: OFF(*) 1: ON(*)

(*) ON: Designates a high on the line.
OFF: Designates a low on the line.

Table 5-6 Hardware Controlled by Software Implementation (2)

Transfer address	Transferred word length	Content	Return information
• Computing function Data register DWFFF • C-mode subroutine Address /62FFE	/0200	DR OFF request (*)	• Channel No. 0 High-order byte of DWFFF (/62FFE) • Channel No. 1 Low-order byte of DWFFF (/62FFF) • Channel No. 2 High-order byte of DWFFF (/62FFC) • Channel No. 3 Low-order byte of DWFFF (/62FFD) /00: OFF report /01: ON report
	/0201	DR ON request (*)	
	/0300	RS OFF request (*)	• Channel No. 0 High-order byte of DWFFF (/62FFE) • Channel No. 1 Low-order byte of DWFFF (/62FFF) • Channel No. 2 High-order byte of DWFFF (/62FFC) • Channel No. 3 Low-order byte of DWFFF (/62FFD) /00: ON report /01: OFF report
	/0301	RS ON request (*)	

(*) ON: Designates a high on the line.
OFF: Designates a low on the line.

With a send parameter address setting of DWFFF, /62FFE, functions or subroutines would return by directing return information to /FF if the parameter word length is other than those listed in Tables 5-5 and 5-6.

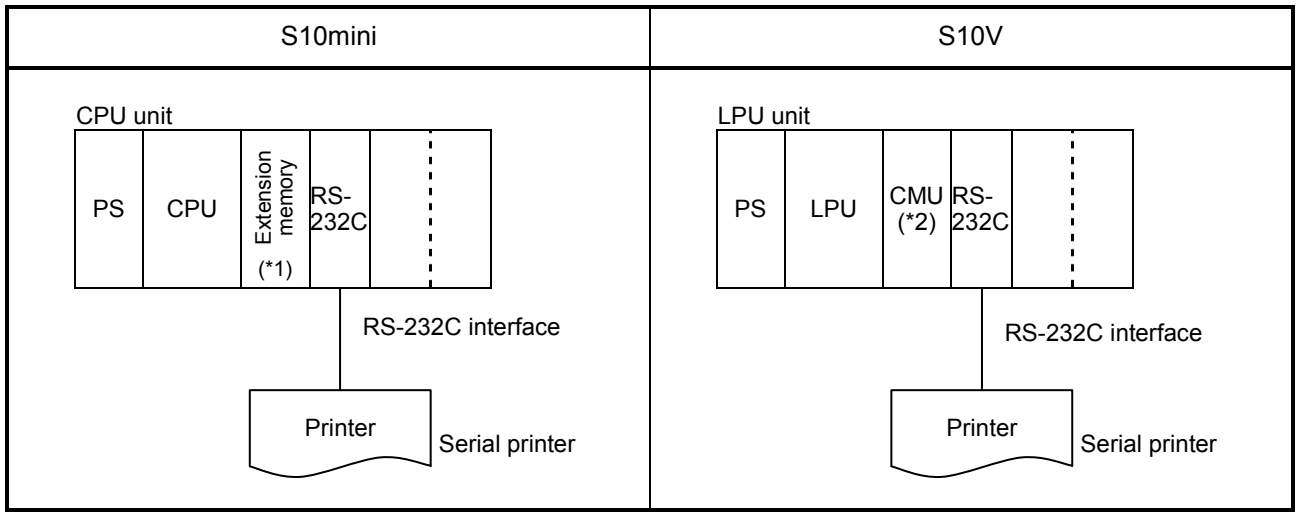
5.6 Sample Programs (Sample RS-232C Wiring with a Printer)

5.6.1 Overview

An RS-232C interface connecting the CPU or LPU unit to a serial printer allows memory data to be printed out in a specified word length, beginning with a specified address.

5.6.2 System configuration

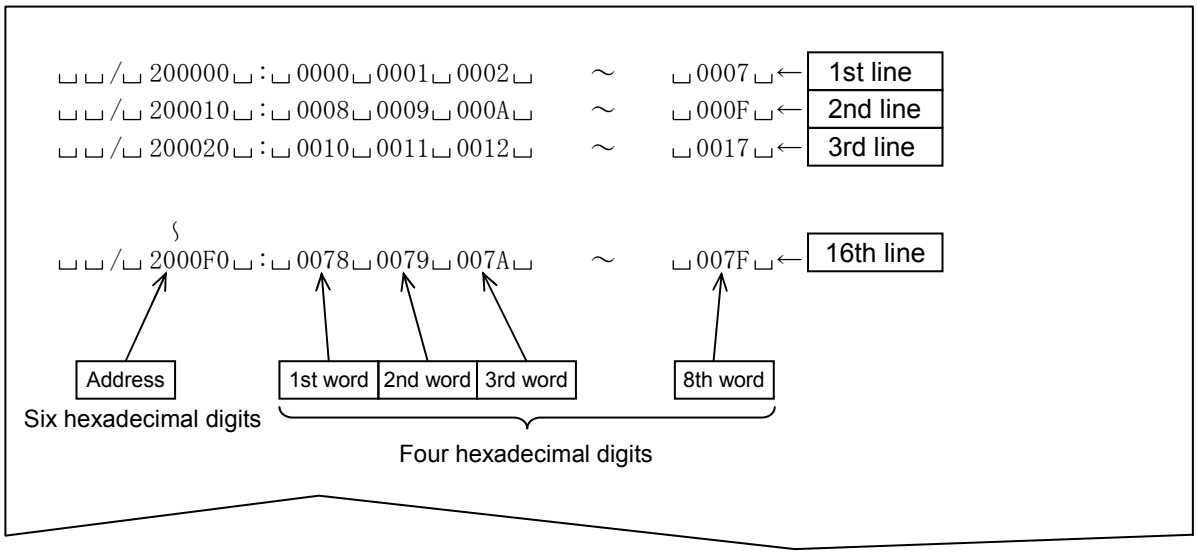
Table 5-7 System Configuration



(*1) If the LQP000 is used as a CPU, an extension memory module is needed when an application task is used.
(*2) An option module CMU is needed when an application task is used.

5.6.3 Print format

The print format is shown below.



␣	Space (blank)	[/20]	
/	Slash	[/2F]	Address mark
:	Colon	[/3A]	Data delimiter

5.6.4 Program configuration

The printer output control program runs as an application task (C mode) created in the C language.

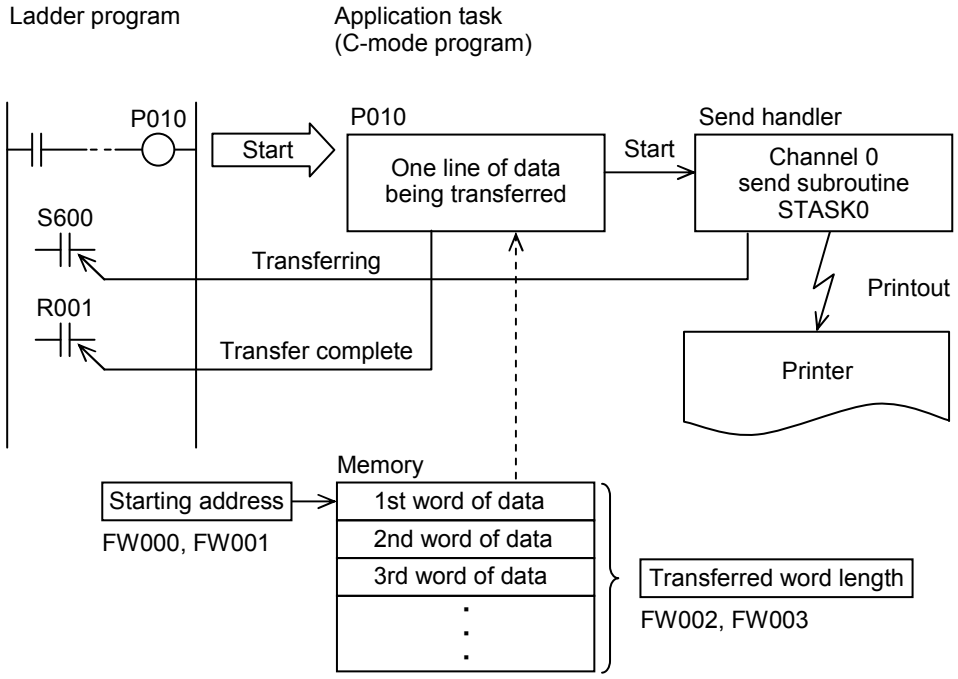


Figure 5-4 Program Configuration

The application task is programmed as a subroutine that prints memory contents line by line and is assigned to P010.
The application task starts when coil P010 is turned on in the ladder program.
Before starting the application task, set the starting address (FW000 to FW001) of the control information table and the transferred word length (FW002 to FW003).

5.6.5 Ladder program linkage table configuration

(1) Printout control information table

Table 5-8 Control Information Table

Symbol	Address	Data	Remarks
FW000	/0E2000	Starting address	32 bits long (binary data)
FW001	/0E2002		
FW002	/0E2004	Transferred word length	32 bits long (binary data)
FW003	/0E2006		

Setting example:

Starting address = Address /120000

Transferred word length = 16 (/10) words

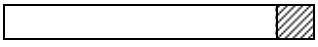
To meet these conditions, set the control information table as described in Table 5-9.

Table 5-9 Control Information Table Setting Example

Symbol	Address	Data
FW000	/0E2000	/0012
FW001	/0E2002	/0000
FW002	/0E2004	/0000
FW003	/0E2006	/0010

(2) Print-complete flag

The print-complete flag turns on when printout of a specified word length of data is completed, and turns off when printout of one line of data is initiated for the first time.

Symbol	Address	Data
R001	/0AC002	2^{15} ... 2^1 2^0 

Because only the LSB (bit 2^0: the least significant bit) of this memory area is valid, the on/off data is set as follows:
When on = /0001
When off = /0000

(3) Transfer-in-progress flag

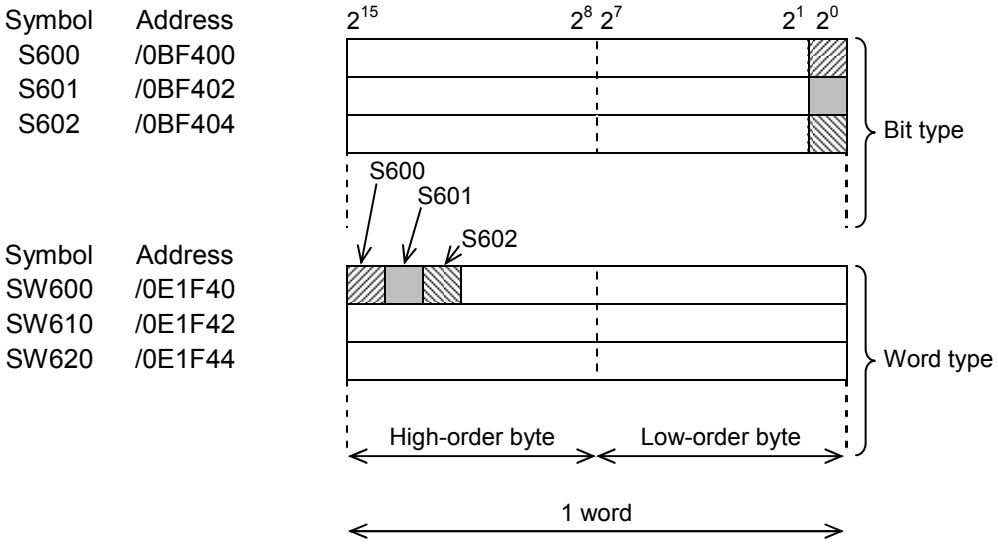
System register S600 of the transfer-in-progress flag of the send handler (STASK0) that is launched on one-line data printout processing is used.

Here, access is made to a bit type area to ease software processing.

Symbol	Address	Data
S600	/0BF400	2^{15} ... 2^1 2^0

Because only the LSB (bit 2^0: the least significant bit) of this memory area is valid, the on/off data is set as follows:
When on = /0001
When off = /0000

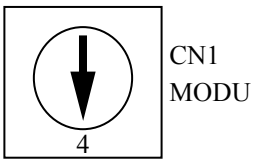
The correspondence between the S600 word type area and the bit type area is shown below.



5.6.6 RS-232C module

(1) Module switch setting

To use CN1 for "Free-running – Task," set the module switch to 4.



(2) RS-232C signal lines

Only data signal lines are used as RS-232C signal connections. Other control lines are out of use. The RS-232C signal lines are wired as shown below.

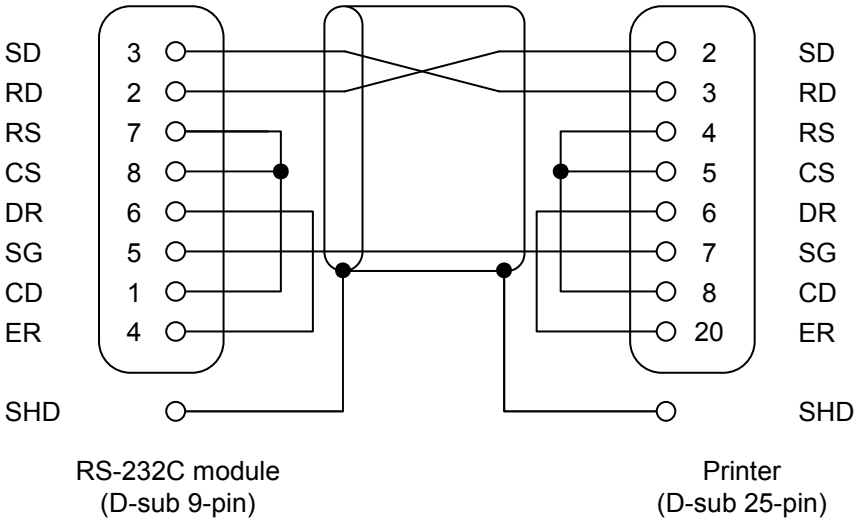


Figure 5-5 RS-232C Signal Connections

5.6.7 Setting the LGB table

Tale 5-10 summarizes the specifications of the printer serial interface.

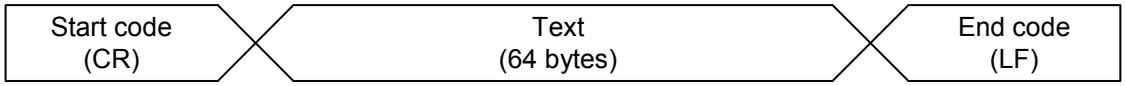
Table 5-10 Serial Interface Specifications

Item	Description
Data frame	Start bit: 1 bit Data bits: 8 bits Parity bit: Yes, even parity Stop bit: 1 bit
Baud rate	4800 bps
Print control	A received data buffer (1 KB) is available. The printer prints the contents of the received data buffer when it receives a line feed code (LF: /0A), executing a line feed automatically.

<Data block structure>

The data block is assumed to have a text length of 64 bytes.

The start code (SCD) and the end code (ECD) of the data block are used in the following format:



CR: Carriage return (/0D)

LF: Line feed (/0A)

No block check character (BCC)

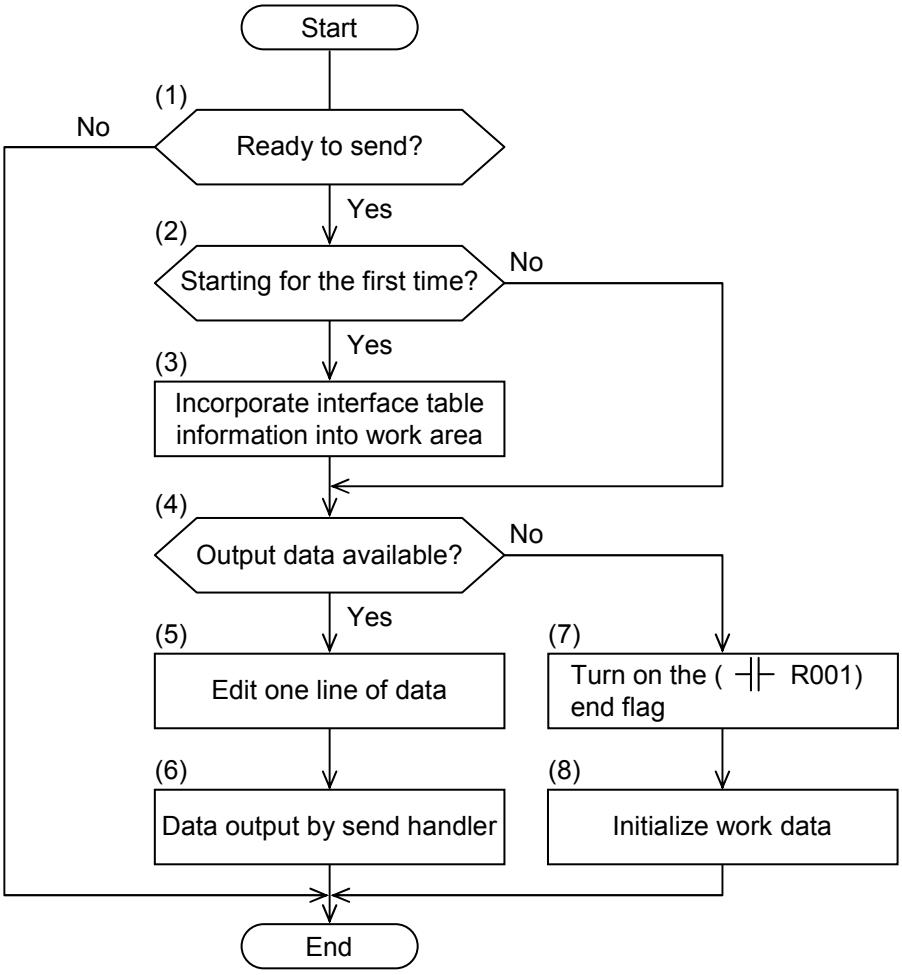
Table 5-11 LGB Table Setting Items

Item	Description
Data frame	ST+8DT+EP+1SP (*)
Baud rate	4800[bps] (*)
Priority level	Local station prioritized
Data change mode	Binary
Text size	64 bytes
Start code	/0D (*)
End code	/0A (*)
Block check character	No BCC
Send delay time	No send delay
Send break and continue codes	No break and continue codes
Send break timeout	3276.7[s]
Receive timeout	3276.7[s]
RS-422 gate control	–
Request to Send (RS)	RS available
Equipment Ready (ER)	Ready
Data Set Ready (DR)	No check
Control signal automatic control	Manual control
System selection	Task system

NOTICE

When using any other printer, edit the items marked by an asterisk (*) to meet the specifications of that printer as needed.

5.6.8 C language program flowchart



(1): Reference the status of system register S600 to check for readiness to transmit.

(2), (3): Check to see if this is the first instance of startup and, if so, incorporate interface table information into a task work area.

(4): Check the next task work area for the output word length remaining.

(5) to (6): If output data is available, edit one line of data and direct the print data to the printer with the send handler.

(7) to (8): When all the output has completed, turn on the end flag (⊣ ⊢ R001) and turn off the initial start flag.

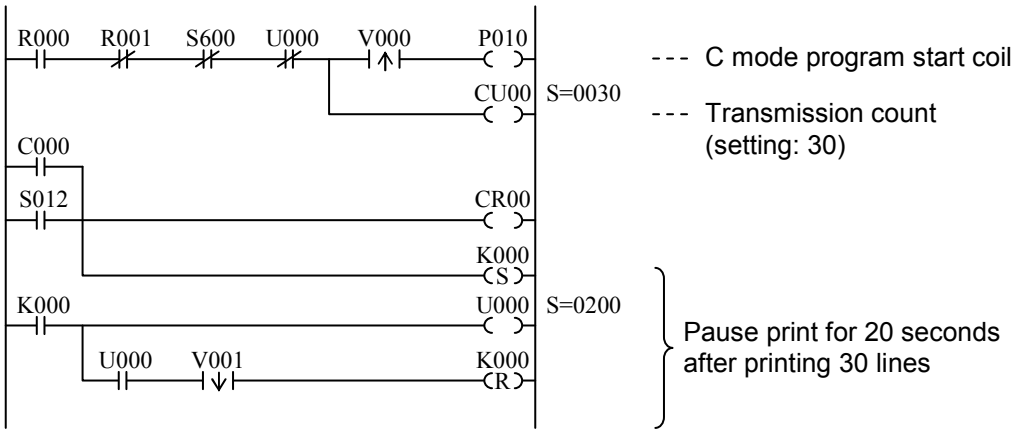
5.6.9 C language sample program

```
1:  /*****************************************************************************/
2:  /*        Sample No.1 :: Memory dump task                                    */
3:  /*****************************************************************************/
4:
5:  #define TXSUB0          0x107000   (←S10mini)        Send handler address
6:  #define IFTB            0xE2000                      F000 control information table address
7:  #define R001            0xAC002                      R001 print-complete flag address
8:  #define S600            0xBF400                      S600 transfer-in-progress flag address
9:  #define MASK            0x0001                       Mask data '1'
10:
11: static struct WORK {    short           flag ;       Processing-in-progress flag
12:                         long            addr ;       Transfer address in process
13:                         long            word ;       Transferred word length in process
14:                         } work ;
15:
16: static char linebf[64] ;                             One-line print buffer
17:
18:
19: p010()
20: {
21: register long (*txsub)() ;
22: register long *lpt ;
23: register char *cpt ;
24: register short wk ;
25: register short ct ;
26: register long retncd ;
27:
28: if( ( *(short *)S600 & MASK ) == 0 )                 Check for readiness to transmit
29:    {
30:    if( work.flag == 0 )                              Incorporate control information table
31:       {
32:       lpt = (long *)IFTB ;
33:       work.addr = *lpt++ ;
34:       work.word = *lpt ;
35:       work.flag = 1 ;
36:       }
37:    if( word.word > 0 )                               Printout processing
38:       {                                              Initialize the line buffer
39:       ct = 64 ;
40:       cpt = &linebf[0] ;
41:       while( --ct >=0 )
42:             *cpt++ = ' ' ;
43:
44:       (long)cpt = &(work.addr) ;                     Set address data
45:       btoas( &linebf[3] , cpt[1] ) ;
46:       btoas( &linebf[5] , cpt[2] ) ;
47:       btoas( &linebf[7] , cpt[3] ) ;
48:
49:       (long)cpt = word.addr ;                        Set memory data
50:       ct = 12 ;
51:       while( ( work.word > 0 ) && ( ct < 50 ) )
```

```
52:                 {
53:                 btoas( &linebf[ct] , *cpt++ ) ;                 High-order byte data
54:                 btoas( &linebf[ct+2] , *cpt++ ) ;               Low-order byte data
55:                 ct += 5 ;                                       Set SP (space)
56:                 work.word -= 1 ;
57:                 }
58:           work.addr += 0x000010 ;
59:
60:           linebf[2] = '/' ;                                     Address mark "/"
61:           linebf[10] = ':' ;                                    Data delimiter ":"
62:
63:           txsub = (long(*)())TXSUB0 ;                           Data transfer
64:           retncd = (*txsub)( &linebf[0] , 64 ) ;
65:           }
66:       else{
67:           work.flag = 0 ;
68:           *(short *)R001 = 1 ;                                  Set the print-complete flag
69:           }
70:       }
71:   return ;
72:   }
73:
74:   /*************************************************************************************/
75:   /*          BINARY → ASCII function ( byte size )                                    */
76:   /*************************************************************************************/
77:   btoas( stp , data )
78:   register char *stp ;                                          Character set pointer
79:   register char data ;                                          Binary data
80:   {
81:   register char wk ;                                            Work register
82:
83:   wk = data ;                                                   Set the high-order digit
84:   wk >>= 4 ;
85:   wk &= (char)0x0F ;
86:   if ( wk <= (char)0x09 )
87:       wk += (char)0x30 ;
88:   else wk += (char)0x37 ;
89:   *stp++ = wk ;
90:
91:   data &= (char)0x0F ;                                          Set the low-order digit
92:   if( data <= (char)0x09 )
93:       data += (char)0x30 ;
94:   else data += (char)0x37 ;
95:   *stp = data ;
96:
97:   return ;
98:   }
99:   /*************************************************************************************/
```

5.6.10 Ladder program

A ladder program that starts the application task (C mode) assigned to P010 is needed to carry out printout.
A sample ladder program (S10mini) is shown below.



R000 .....Print request
R001 .....Print complete
S600.....Transferring a remote station link
C000 .....30-line print counter
U000.....Timer to pause print after printing 30 lines
S012.....STOP→RUN signal
K000.....Save 30-line print counter on power failure

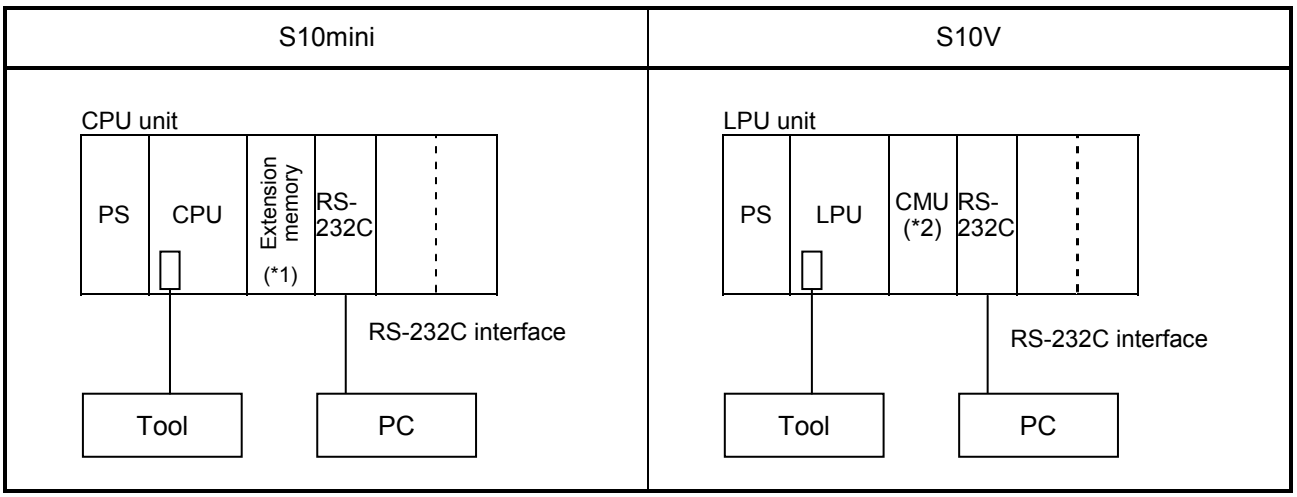
NOTICE
● The printer used this time prints so slow when compared with the speed of data transfer that the continuous transmission of print data to it would cause the printer to malfunction with a received data buffer overflowing. The ladder program pauses transmission for 20 seconds after printing 30 lines to prevent this. ● The sample program is designed to promote understanding. The working program should make error checks on the send handler return code and system registers (S).

5.7 Sample Program (PC-based Program Loading)

5.7.1 System configuration

Connect a PC to the CPU or LPU unit by way of an RS-232C interface to load programs written in the C or other languages into CPU memory directly.

Table 5-12 System Configuration



(*1) If the LQP000 is used as a CPU, an extension memory module is needed when an application task is used.
(*2) An option module CMU is needed when an application task is used.

5.7.2 Program configuration

The S-formatted data receive task starts on receiving S-formatted data from the PC, setting the incoming data at a memory address.

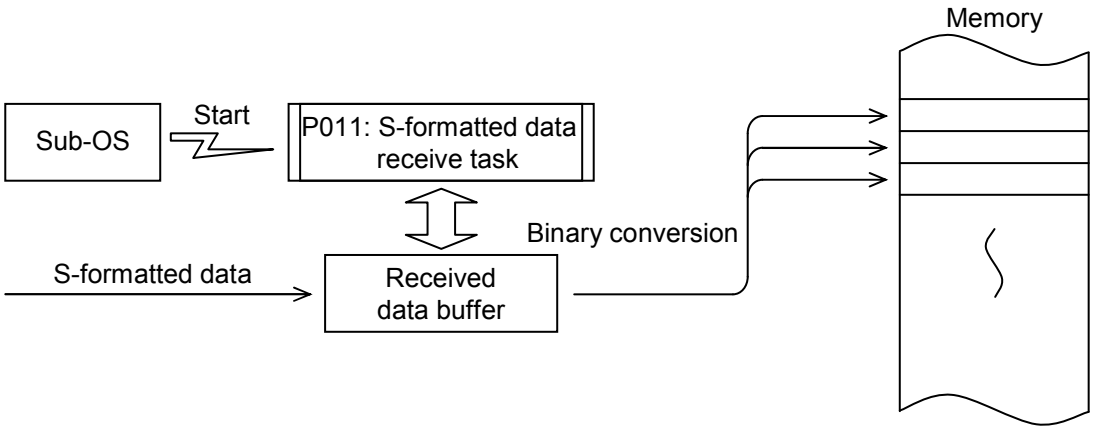


Figure 5-6 Program Configuration

5.7.3 Motorola ‘S’ format (16-bit)

This data format is used to transmit and receive objects, such as 68000, over a line.

Data format

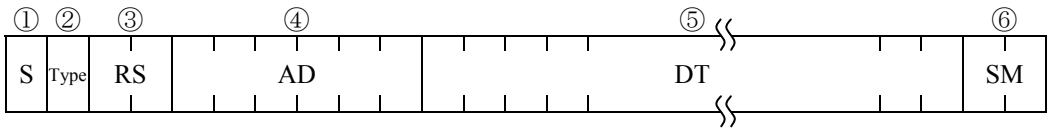
S22411000048E7FFFE207C00100000227C00110000247C00120000267000130000287C001424

S22411002000002A7C001200002C7C00160000203CD0D0D0D0223CD1D1D1D1243CD2D2D2D24B

S224110040263CD3D3D3D3286CD4D4D4D42A3CD5D5D5D52C3CD6D6D6D62E3CD7D7D7D74CDFBD

S2241100607FFF4E750000000000000000000000000000000000000000000000000000000029

S9030000FC



① Signifies the beginning of a record. It is always ‘S’ (/53).

② Designates the record type.

‘2’ (/32): Data record

‘9’ (/39): Last record

③ Designates the byte size of the record in two characters (byte length of ④ to ⑥)

④ Designates the associated starting address in six characters.

⑤ Designates one byte of memory data in two characters.

⑥ Designates the one’s complement of the sum total of data ③ to ⑤ in two characters.

(Supplement)

A CR (Carriage Return: /0D) and an LF (Line Feed: /0A) follow the checksum field of each S-formatted record transferred to the PC. Using these codes, the start and end codes have been decided as follows (see “5.7.4 LGB table settings”):

Start code (SCD) = ‘S’ (=/53) ... One character

End code (ECD) = ‘CR (/0D) + LF (/0A) ... Two characters

5.7.4 LGB table settings

An example of settings in the LGB table is shown below.

Table 5-13 LGB Table Setting Items

Item	Description
Data frame	ST+8DT+OP+1SP
Baud rate	1200[bps]
Priority level	Local station prioritized
Data change mode	Binary
Text size	256 bytes
Start code	/53
End code	/0D+0A
Block check character	No BCC
Send delay time	No send delay
Send break and continue codes	No break and continue codes
Send break timeout	3276.7[s]
Receive timeout	3276.7[s]
RS-422 gate control	–
Request to Send (RS)	RS available
Equipment Ready (ER)	Ready
Data Set Ready (DR)	No check
Control signal automatic control	Manual control
System selection	Task system

Baud rate: Too high a baud rate could result in an increased CPU load on the S10mini CPU unit, resulting in occasional inability to receive data. To avoid this, set the baud rate rather lower.

Data change mode: The receive task converts text data to binary.

Text size: Used 256 bytes as a standard size.

Start code: An S-formatted record begins with ‘S’ so that ‘S’ is used as a standard code.

End code: A CR and an LF follow the checksum field of the S-formatted record. These codes are used as an end code.

Others: Set to meet the specifications of the PC.

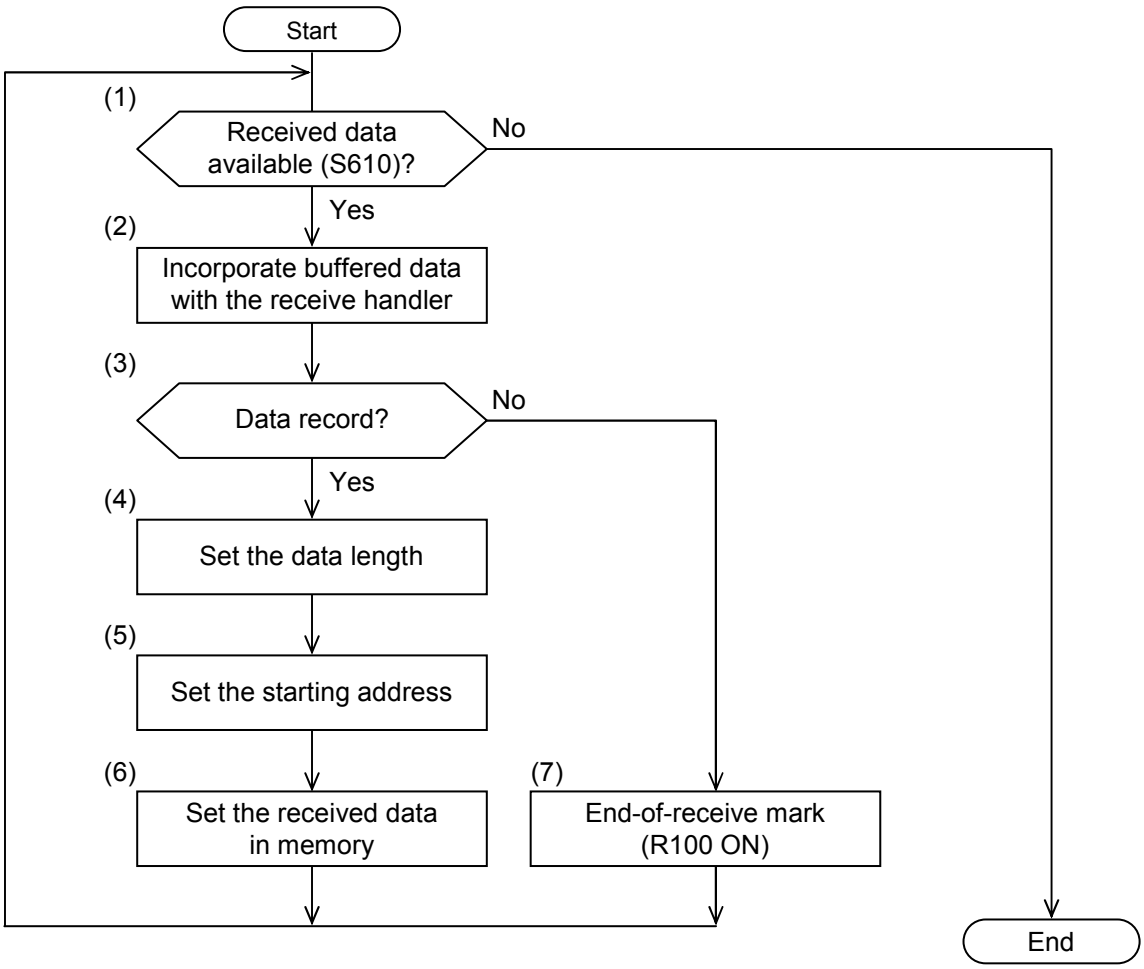
5.7.5 Registering a receive task

Register a PC mode program as a receive task and assign it to P011 (task number 17). The task is started by start cause 01.

Example of receive task registration

Item	Description
Receive task registration	Task number: 17
	Start cause: 01

5.7.6 Start task



(1): Check that the receive flag (S610) is on.

(2): Incorporate the received data with the receive handler.

(3): Check that the record type is '2' (/32).

(4) to (6): If the record is a data record, incorporate its data length and starting address and set the received data in memory as specified by such information.

(7): If the record is not a data record, it is assumed the last record and the receive-complete flag (┤├ R100) is set on. This data read sequence is iterated until the flag turns off.

5.7.7 C language sample program

```
1:  /*****************************************************************************/
2:  /*          Sample No.2 :: Program Loading task                              */
3:  /*****************************************************************************/
4:
5:  #define RXSUB0      0x10700C    (←S10mini)      Receive handler address
6:  #define S610        0xBF420                     S610 transfer-in-progress flag
7:  #define R100        0xAC200                     Receive-complete flag address
8:  #define MASK        0x0001                      Mask data '1'
9:
10: static char buff[512] ;                         Received data buffer (512 bytes)
11:
12: p011()
13: {
14: extern char atob() ;
15: register long (*sub)() ;                        Receive handler
16: register char *addr ;                           Address pointer
17: register short *dpt ;                           Data pointer
18:
19: register long retncd ;                          Return code
20: register short ct ;                             Loop counter
21:
22: union { long lad ;
23:         char cad[4] ;
24:         } adwk ;
25:
26: sub = (long(*)())RXSUB0 ;                       Incorporate received data
27: while( ( *(short *)S610 & MASK ) != 0 )
28:       {
29:       retncd = (*sub)( &buff[0], 80 ) ;
30:       if( buff[0] == '2' )
31:          {                                      Set the data number
32:          ct = (short)atob( &buff[1] ) ;
33:          ct &= 0x003F ;                         Set the address number
34:          adwk.cad[1] = atob( &buff[3] ) ;
35:          adwk.cad[2] = atob( &buff[5] ) ;
36:          adwk.cad[3] = atob( &buff[7] ) ;
37:          adwk.cad[0] = (char)0 ;
38:          (long)addr = adwk.lad ;                Write the data to memory
39:          (char *)dpt = &buff[9] ;
40:          ct -= 3 ;
41:          while( --ct > 0 )
42:                *addr++ = atob( dpt++ ) ;
43:          }
44:       else *(short *)R100 = 1 ;                 Set the receive-complete flag
45:       }
46: return ;
47: }
48: /*****************************************************************************/
49: /*       ASCII → BINARY function                                             */
50: /*****************************************************************************/
51: char atob( pt )
52: register char *pt ;
53: {
54: register char wkh, wkl ;
```

```
55:
56:   wkh = *pt++ ;
57:   wkh -= '0' ;
58:   if( wkh > (char)9 )
59:       wkh -= 7 ;
60:   wkh <<= 4 ;
61:   wkh &= (char)0xF0 ;
62:
63:   wkl = *pt ;
64:   wkl -= (char)0x30 ;
65:   if( wkl > (char)9 )
66:       wkl -= 7 ;
67:   wkl &= (char)0x0F ;
68:
69:   wkh |= wkl ;
70:   return( wkh );
71:   }
72:   /*****************************************************************************/
```

5.7.8 Loading programs

After loading the receive task (P011) into CPU or CMU memory, set the LGB table and register the receive task.
Next, set the PC's baud rate, data frame and so forth and then connect the RS-232C cable.
This will allow S-formatted data to be serially transmitted from the PC to the CPU or CMU.

(Supplement)

- A baud rate of 1200 bps has been set (to allow for the CPU load on the S10mini CPU unit).
- The sample program does not make error checks at receive. The working program should make error checks to launch error handling as needed.

6 GUIDE TO USE

6.1 Seven-Bit Code Table (JIS X 0201)

b7	b6	b5	b4	b3	b2	b1		0	0	0	0	1	1	1	1
							b6	0	0	1	1	0	0	1	1
							b5	0	1	0	1	0	1	0	1
								0	1	2	3	4	5	6	7
0	0	0	0	0				NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1				SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2				STX	DC2	“①	2	B	R	b	r
0	0	1	1	3				ETX	DC3	#	3	C	S	c	s
0	1	0	0	4				EOT	DC4	$	4	D	T	d	t
0	1	0	1	5				ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6				ACK	SYN	&	6	F	V	f	v
0	1	1	1	7				BEL	ETB	’ ②	7	G	W	g	w
1	0	0	0	8				BS	CAN	(	8	H	X	h	x
1	0	0	1	9				HT	EM	)	9	I	Y	i	y
1	0	1	0	10				LF	SUB	*	: ⑥	J	Z	j	z
1	0	1	1	11				VT	ESC	+	; ⑦	K	[	k	{
1	1	0	0	12				FF	IS4	, ③	<	L	¥	l	|
1	1	0	1	13				CR	IS3	−④	=	M	]	m	}
1	1	1	0	14				SO	IS2	. ⑤	>	N	^	n	―
1	1	1	1	15				SI	IS1	/	?	O	_⑧	o	DEL

① Quote
② Apostrophe
③ Comma
④ Minus
⑤ Period
⑥ Colon
⑦ Semicolon
⑧ Underline

6.2 Eight-Bit Code Table (JIS X 0201)

b8	b7	b6	b5	b4	b3	b2	b1		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
								b8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
								b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
								b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
								b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
									0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
				0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p	Free space		⑮	ー⑫	タ	ミ	Undefined	
				0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q			。⑨	ア	チ	ム		
				0	0	1	0	2	STX	DC2	“①	2	B	R	b	r			「	イ	ツ	メ		
				0	0	1	1	3	ETX	DC3	#	3	C	S	c	s			」	ウ	テ	モ		
				0	1	0	0	4	EOT	DC4	$	4	D	T	d	t			、⑩	エ	ト	ヤ		
				0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u			・⑪	オ	ナ	ユ		
				0	1	1	0	6	ACK	SYN	&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
				0	1	1	1	7	BEL	ETB	’ ②	7	G	W	g	w			ァ	キ	ヌ	ラ		
				1	0	0	0	8	BS	CAN	(	8	H	X	h	x			ィ	ク	ネ	リ		
				1	0	0	1	9	HT	EM	)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
				1	0	1	0	10	LF	SUB	*	: ⑥	J	Z	j	z			ェ	コ	ハ	レ		
				1	0	1	1	11	VT	ESC	+	; ⑦	K	[	k	{			ォ	サ	ヒ	ロ		
				1	1	0	0	12	FF	IS4	, ③	<	L	¥	l	|			ャ	シ	フ	ワ		
				1	1	0	1	13	CR	IS3	−④	=	M	]	m	}			ュ	ス	ヘ	ン		
				1	1	1	0	14	SO	IS2	. ⑤	>	N	^	n	―			ョ	セ	ホ	゛⑬		
				1	1	1	1	15	SI	IS1	／	?	O	_⑧	o	DEL			ッ	ソ	マ	゜⑭		⑮

① Quote
② Apostrophe
③ Comma
④ Minus
⑤ Period
⑥ Colon
⑦ Semicolon
⑧ Underline
⑨ Japanese period
⑩ Japanese comma
⑪ Middle point
⑫ Prolonged sound
⑬ Voiced consonant
⑭ P-sound sign attached to kana
⑮ Undefined

6.3 Control Code Definitions

Control code	Code	Control code name	Definition
NUL	/00	Null	Blank
SOH	/01	Start of Heading	Start of heading
STX	/02	Start of Text	Start of text
ETX	/03	End of Text	End of text
EOT	/04	End of Transmission	End of transmission
ENQ	/05	Enquiry	Enquiry
ACK	/06	Acknowledge	Acknowledge
BEL	/07	Bell	Bell
BS	/08	Backspace	Backspace
HT	/09	Horizontal Tabulation	Horizontal tabulation
LF	/0A	Line Feed	Line feed
VT	/0B	Vertical Tabulation	Vertical tabulation
FF	/0C	Form Feed	Form feed
CR	/0D	Carriage Return	Carriage return
SO	/0E	Shift Out	Shift out
SI	/0F	Shift In	Shift in
DLE	/10	Data Link Escape	Data link escape
DC1	/11	Device Control 1 (X-ON)	Device control 1 (used before initiating transmission)
DC2	/12	Device Control 2	Device control 2
DC3	/13	Device Control 3 (X-OFF)	Device control 3 (used before terminating transmission)
DC4	/14	Device Control 4	Device control 4
NAC	/15	Negative Acknowledge	Negative acknowledge
SYN	/16	Synchronous Idle	Synchronous idle
ETB	/17	End of Transmission Block	End of data block
CAN	/18	Cancel	Cancel
EM	/19	End of Medium	End of medium
SUB	/1A	Substitute Character	Substitute character
ESC	/1B	Escape	Escape (used for escaping control codes, as for displays and graphics)
FS	/1C	File Separator	File separator
GS	/1D	Group Separator	Group separator
RS	/1E	Record Separator	Record separator
US	/1F	Unit Separator	Unit separator
SP	/20	Space	Space
DEL	/7F	Delete	Delete

6.4 Abbreviations

Abbreviation	Explanation
ACIA	Asynchronous Communications Interface Adapter
ASCII	American Standard Code for Information Interchange
BCC	Block Check Character
BPS	Bits Per Second
CD	data Carrier Detect
CPMS	Compact Process Monitor System
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CS	Clear to Send
DR	Data set Ready
ECD	End Code
EIA	Electronic Industries Association
EOR	Exclusive OR
ER	Equipment Ready
FE	Framing Error
FG	Frame Ground
GR	General Reset
IRQ	Interrupt Request
LED	Light Emitting Diode
LGB	Line Group Block
MCS	Man-machine Communication System
OVRN	Overrun error
PCs	Programmable Controllers
PE	Parity Error
RD	Receive Data
RS	Request to Send
SCD	Start Code
SD	Send Data
SG	Signal Ground
SHD	Shield
TERM	Terminating Resistor
UFET	User Function Edition Table
WDT	Watchdog Timer

This Page Intentionally Left Blank

7 MAINTENANCE

7.1 Maintenance and Check

To keep the module running in optimal condition, it requires checks. Make checks daily or periodically (twice a year or more often).

Table 7-1 Maintenance and Inspection Items

Item	Point to check
Module appearance	Check the module case for cracks, flaws and other defects. Such defects may be signs of breakage in the internal circuitry, leading to system malfunctions.
LED	Check to see if the RS-232C or RS-422 module ERR LED has not glowed.
Loose mounting screws	Check the mounting screws for tightness. Retighten them if found loose. Loose screws could lead to system malfunctions and eventually result in a burnout under heat.
Cable covering conditions	Check the cable coverings for defects. Coverings out of position could lead to system malfunctioning, electrical shock hazards, and a burnout after shorts.
Dust	Check the module to see if it has not caught dust. Remove dust with a vacuum cleaner if found. Dust could cause internal circuitry to short, resulting in a burnout.
Module replacement	Hot replacement could lead to hardware and software failures. Be sure to switch off the module before replacing it.
Connector status	Dust or foreign matter on the connector contacts could degrade connector characteristics. Be sure to attach the dust cap supplied to connectors when they are out of use.

CAUTION

Before replacing the module, switch it off to avoid electrical shock hazards and also to prevent it from being damaged or malfunctioning.

NOTICE

Static electricity could cause damage to the module. Before handling the module, discharge static electricity on the human body.

7.2 Backing Up User Setup Items

7.2.1 LGB table, receive task entry table, and user computing function entry table

- Backup to flash memory in the module
  Edits made to the LGB table, receive task entry table, and user computing function entry table using are saved to flash memory in the module when CPU or LPU module is reset after the editing. When power recovers, the LGB table, receive task entry table, and user computing function entry table stored in flash memory in the module are loaded to start operation.
- Backup in a batch save
  Table 7-2 lists the areas of the LGB table, receive task entry table, and user computing function entry table that can be batch-saved by the batch save/load system.

Table 7-2 Tables That Are Batch-Saved

Name	Channel No.	Address		Batch save
		S10mini	S10V	
LGB table	1	/F48100 to /F481FE		Batch-saved
	2	/F58100 to /F581FE		Batch-saved
	3	/F68100 to /F681FE		Batch-saved
	4	/F78100 to /F781FE		Batch-saved
Receive task entry table	1	/1070CA to /1070D0	/F481C0 to /F481D0	Not batch-saved (user-specified)*
	2	/10714A to /107150	/F581C0 to /F581D0	Not batch-saved (user-specified)*
	3	/1071CA to /1071D0	/F681C0 to /F681D0	Not batch-saved (user-specified)*
	4	/10724A to /107250	/F781C0 to /F781D0	Not batch-saved (user-specified)*
User computing function entry table	1 to 4	/FAB40 to /FAD3E	No	Batch-saved

* (Note pertaining to the S10mini Series only)
When executing a batch save, back up expansion memory as well to cover the areas of interest. With Batch Save/Load System (Model: S-7890-09) Version 08-00 and higher, whole expansion memory is backed up by default. With earlier versions, be sure to expressly specify expansion memory addresses.

NOTICE
● The receive task entry table is not automatically batch-saved. Users should specify its address when saving it. The LGB table and the user computing function entry table are automatically batch-saved. ● If a power failure occurs before or during the reset that is carried out after the end of editing from a tool or batch loading, the data written to flash memory in the module would take effect, rather than the edits or settings entered by the batch load. In this case, reset the module again after the end of editing or batch loading.

7.2.2 Replacing modules

- When the RS-232C or RS-422 module has been replaced

  When the RS-232C or RS-422 module has been replaced after failures or other adverse conditions, there are two ways to set the LGB table again as follows:
  - Set the LGB table and the receive task entry table from the tool and activate the CPU module reset switch.
  - If the tables have been batch-saved, carry out a reset after loading from the tool, and the tables would be written back to flash memory in the module and take effect from that moment on.

- When the CPU or LPU module has been replaced

  When the CPU or LPU module has been replaced after failures or other adverse conditions, there is no need to set the LGB table and the receive task entry table again (since they have been saved to flash memory in the module). Perform a reset after a batch load for the tables to take effect from that moment on.

7.2.3 Precautions for setting and changing S10mini RS-232C module parameters

In a user system in which two S10mini RS-232C modules are installed, if an attempt is made to set or change parameters for three or more channels at a time by using one of the methods ① and ② described below, the S10mini CPU may go down due to a WDT error, resulting in no reflection of the up-to-date parameter values in those RS-232C modules. To avoid this, the user is advised to reset the CPU each time the parameter values for one or two channels are set in the module, as described below.

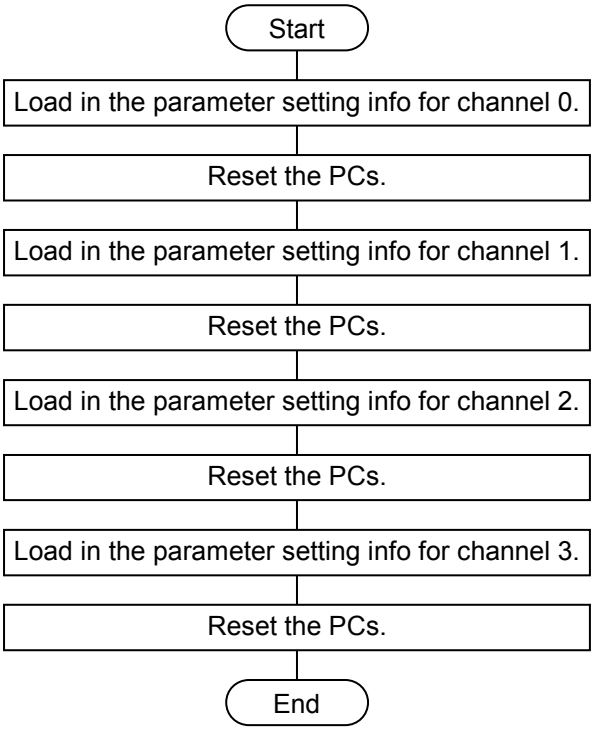
① By loading in the parameter setting information for three or more channels from the parameter setting info files with the ladder chart system's FD function

② By loading in the parameter setting information for three or more channels from the parameter setting info files via the batch saving/loading system

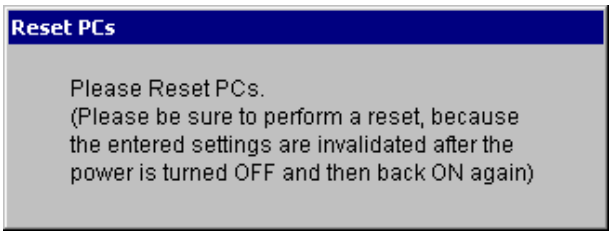
(1) General flowcharts for parameter setting and changing

- Procedure where the ladder chart system's FD function is used

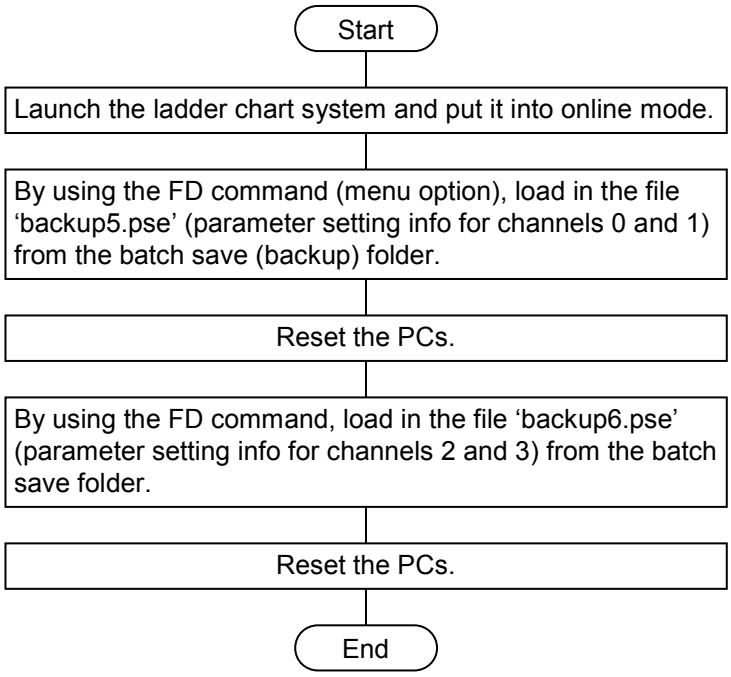
  When saving parameter setting information in file, save the parameter setting information for each channel in a separate file. When loading in the saved parameter setting information, load in the saved parameter setting information for each channel separately, as flowcharted below, and reset the PCs each time the saved parameter setting information for one channel is loaded in.



- Procedure where the backup restore system is used
  The following message appears on screen when all the given parameter setting information has been loaded in at a time by batch-loading. In this case, do not reset the PCs, but instead power the PCs down and then up again.



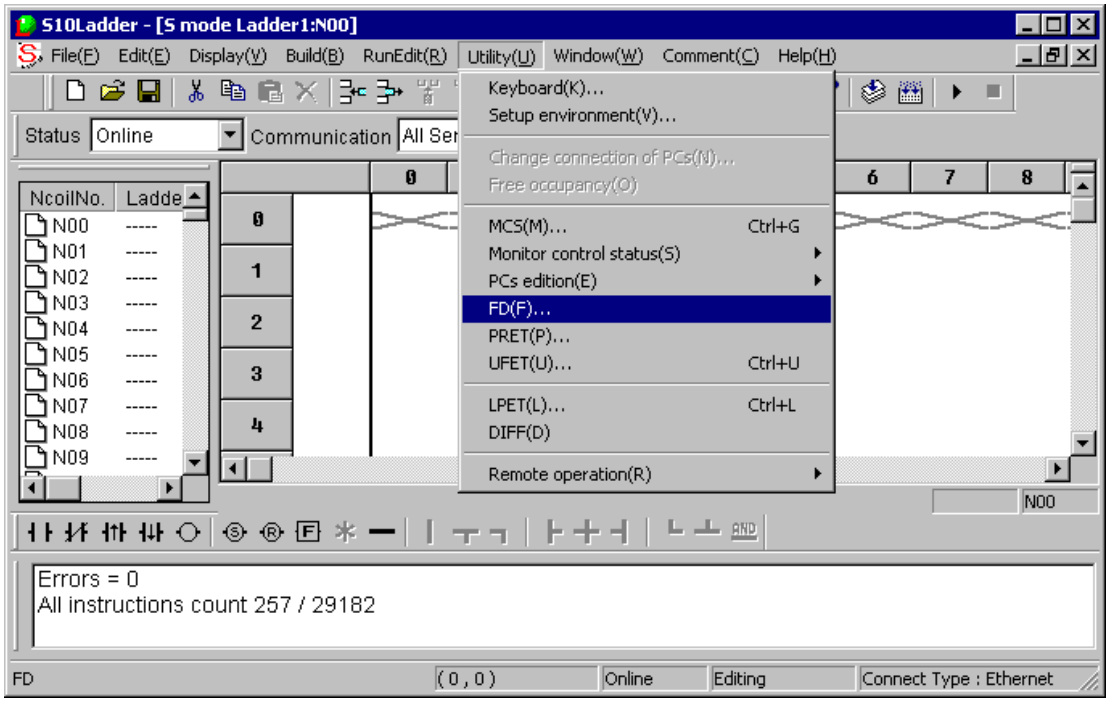
Then, set all necessary parameter values in the RS-232C modules according to the following flowchart:



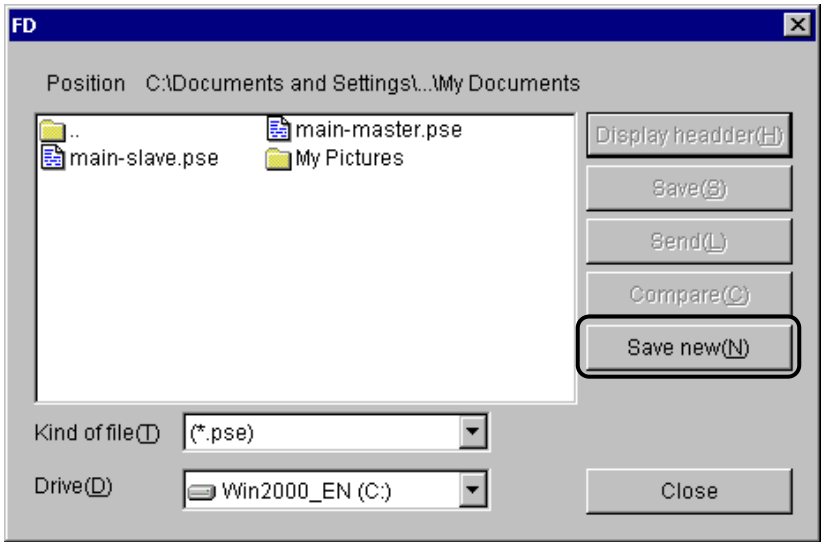
(2) Details of the parameter setting/changing procedures

- Procedure where the ladder chart system's FD function is used: saving parameter setting information

① Launch the ladder chart system and put it into online mode. Then, choose [FD] from the [Utility] pulldown menu, as shown below.

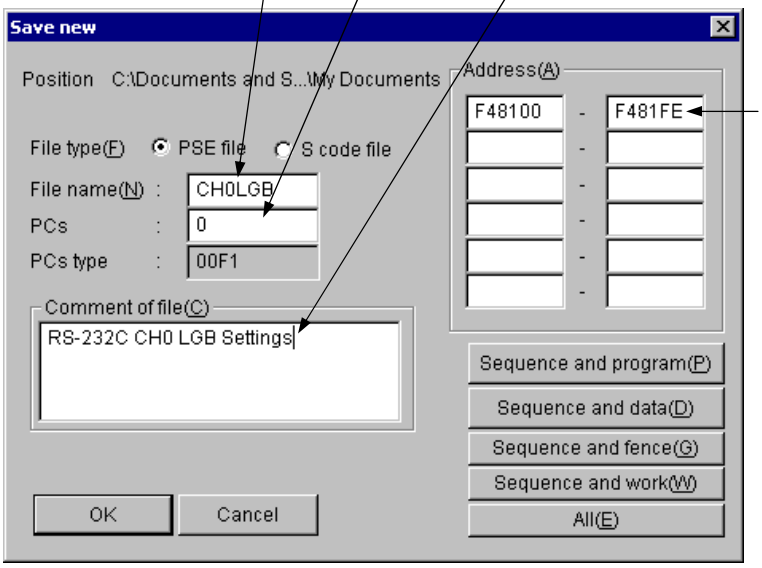


② In the [FD] window displayed, choose the desired folder in which you want to save a parameter setting info file, and then click the Save new button, as shown below.

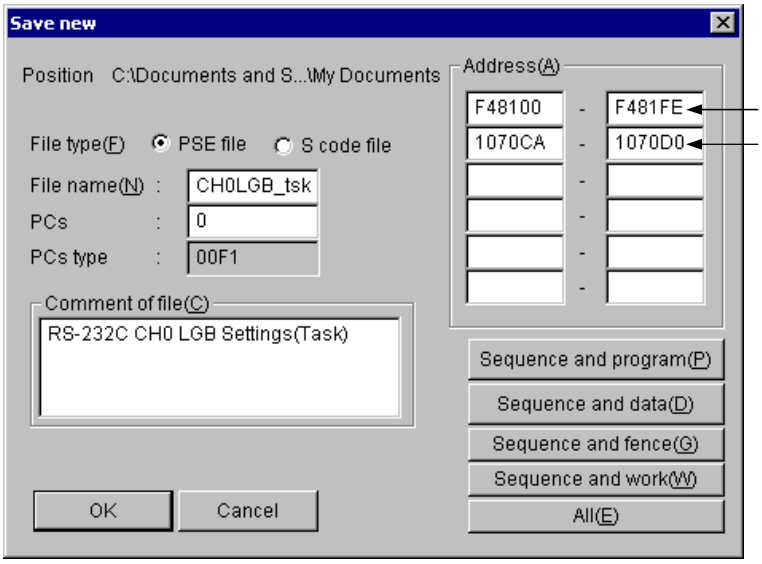


③ Enter the desired File name, PCs, Comment of file, and addresses as header information, as shown below, and click the OK button.

Enter the desired File name, PCs, and Comment of file.



Enter the starting and ending addresses of the desired LGB table listed in the table below into the corresponding Address boxes. (In the snapshot shown left, the starting and ending addresses of the LGB table for channel 0 are entered.)



Enter the starting and ending addresses of the desired LGB table listed in the table below into the corresponding Address boxes. In addition, if a receiving task is registered in the user system, also enter its registration addresses in the Address boxes. (In the snapshot shown left, the following addresses are entered: the starting and ending addresses of the LGB table for channel 0, and the registration addresses of the receiving task.)

The table below shows the addresses to which parameter setting information can be saved.

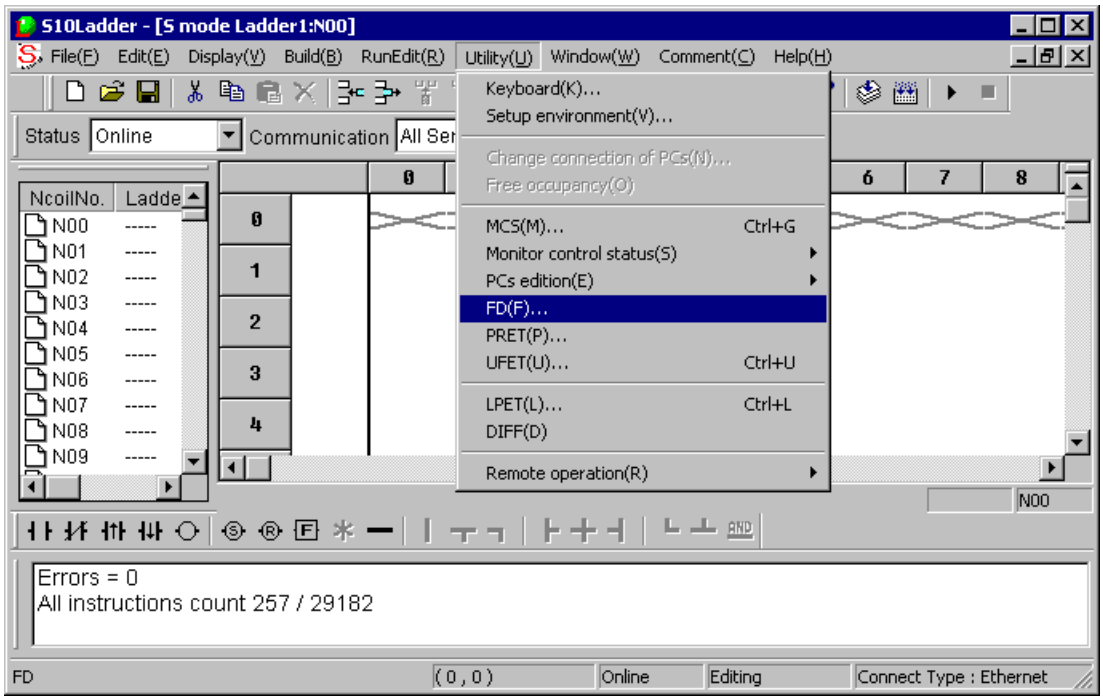
	Channel 0	Channel 1	Channel 2	Channel 3	Remarks
Addresses of LGB table	0xF48100 to 0xF481FE	0xF58100 to 0xF581FE	0xF68100 to 0xF681FE	0xF78100 to 0xF781FE	
Addresses of receiving task registered	0x1070CA to 0x1070D0	0x10714A to 0x107150	0x1071CA to 0x1071D0	0x10724A to 0x107250	Applicable only when a task system is used.

④ When the message shown below appears, the saving process is complete. Click the OK button.

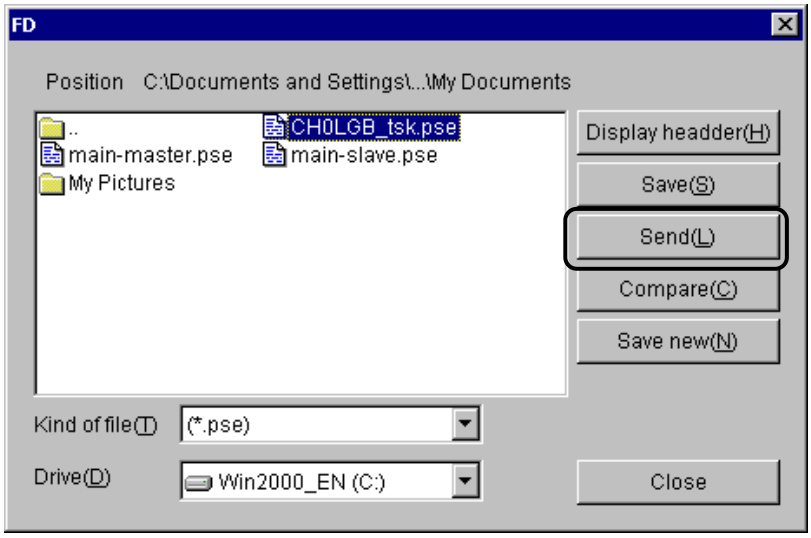
Repeat Steps ② through ④ for each of the other channels.

- Procedure where the ladder chart system's FD function is used: loading in parameter setting information

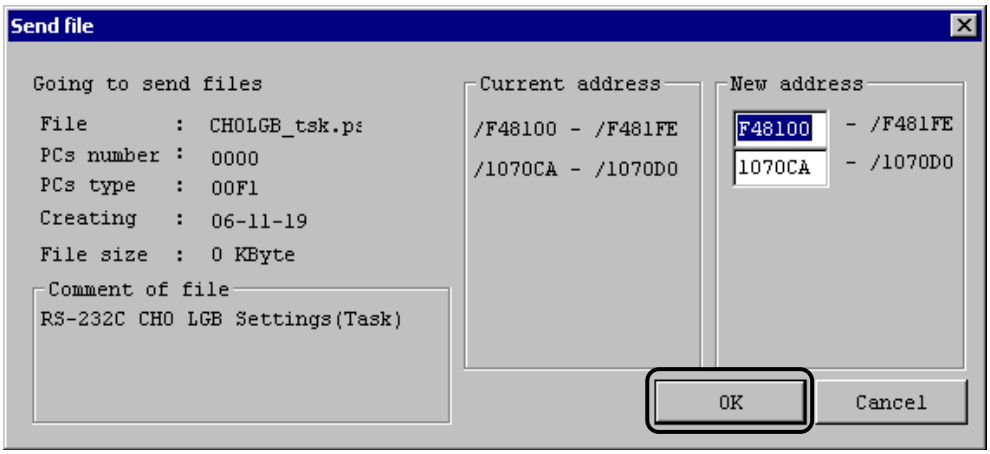
① Launch the ladder chart system and put it into online mode. Then, choose [FD] from the [Utility] pulldown menu, as shown below.



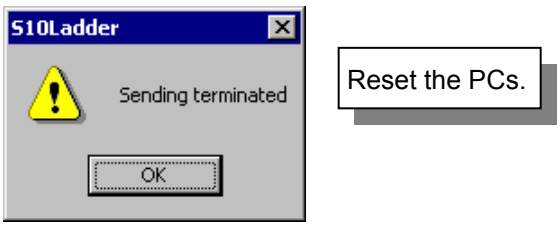
② Choose the desired file that you want to load in, and then click the Send button.



③ The file header as shown below will be displayed. When it is displayed, click the OK button. Loading of the file starts.



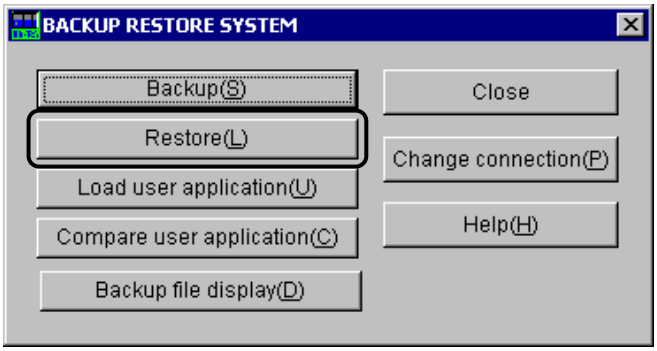
④ When the message shown below appears, the loading process is complete. **Reset the PCs.**



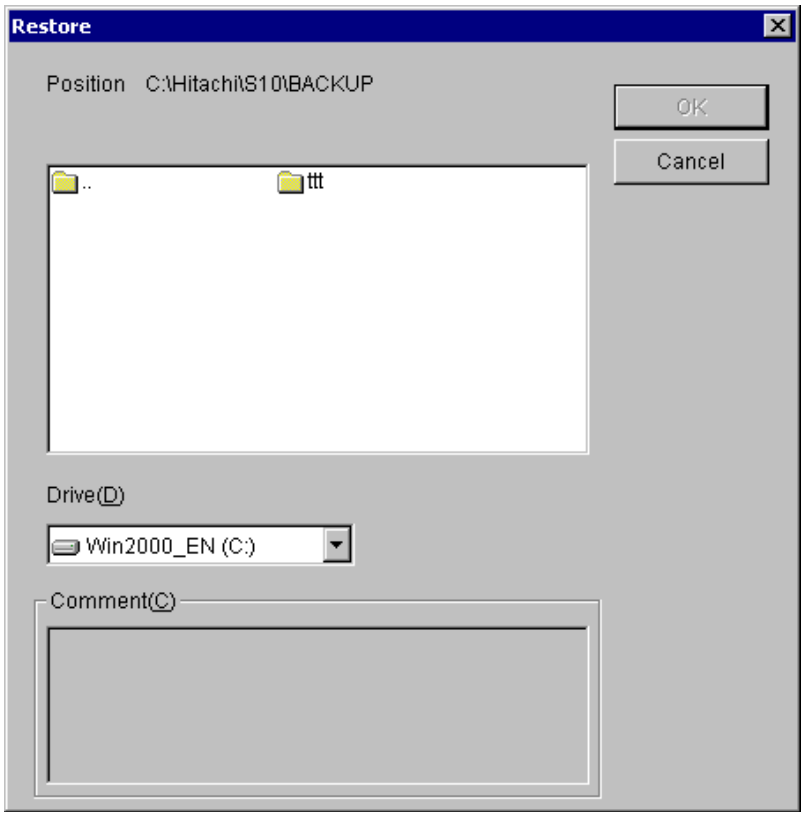
Then, repeat Steps ② through ④ for each of the other channels.

- Procedure where the backup restore system is used

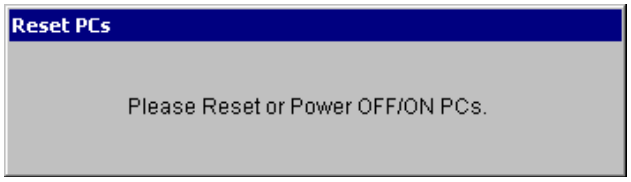
① Launch the batch backup restore system and click the Restore button as shown below.



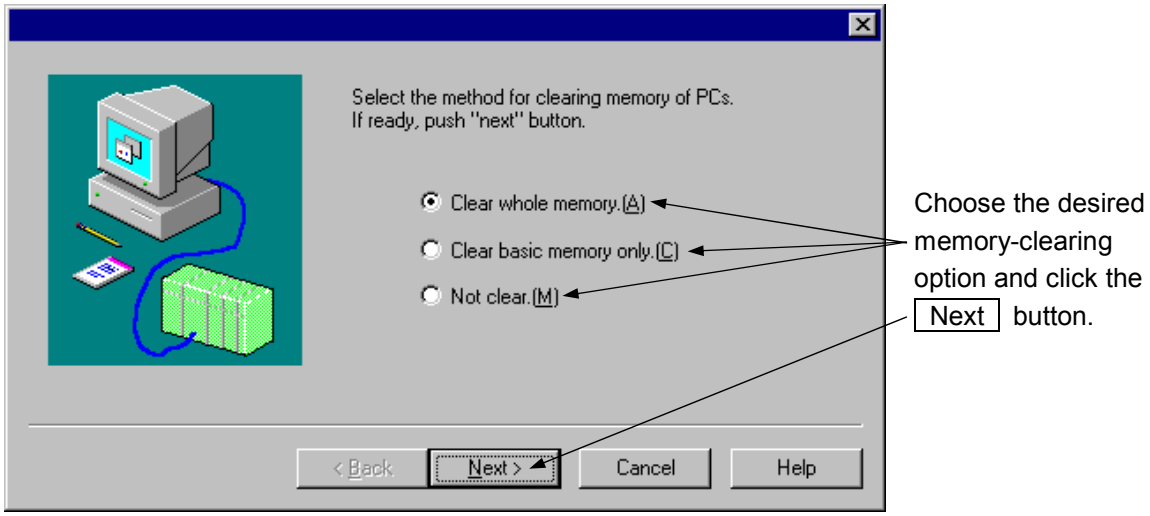
② Choose the desired batch save folder and click the OK button.



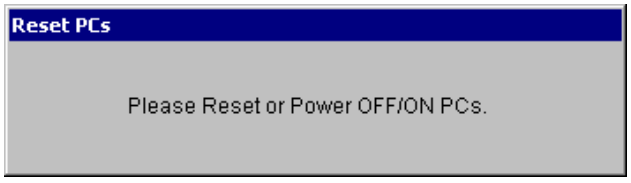
③ When the message shown below appears, either power the PCs down and then up again, or reset it.



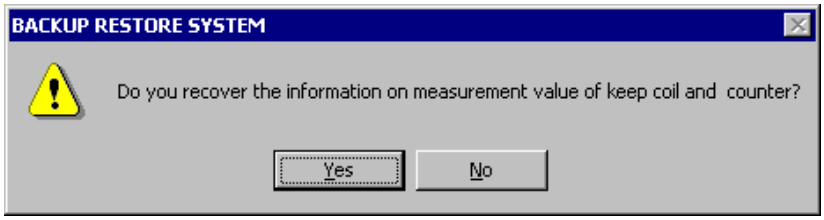
④ Choose the desired memory-clearing option and click the Next button. If you have selected the [Not clear] option, proceed to Step ⑥.



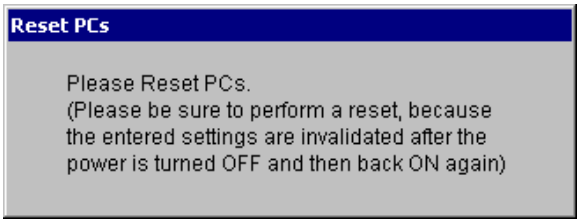
⑤ When the message shown below appears, either power the PCs down and then up again, or reset it.



⑥ If you want to restore the measurement values of keep coils and counters, click the Yes button; otherwise, click the No button. Then, loading of the files starts.

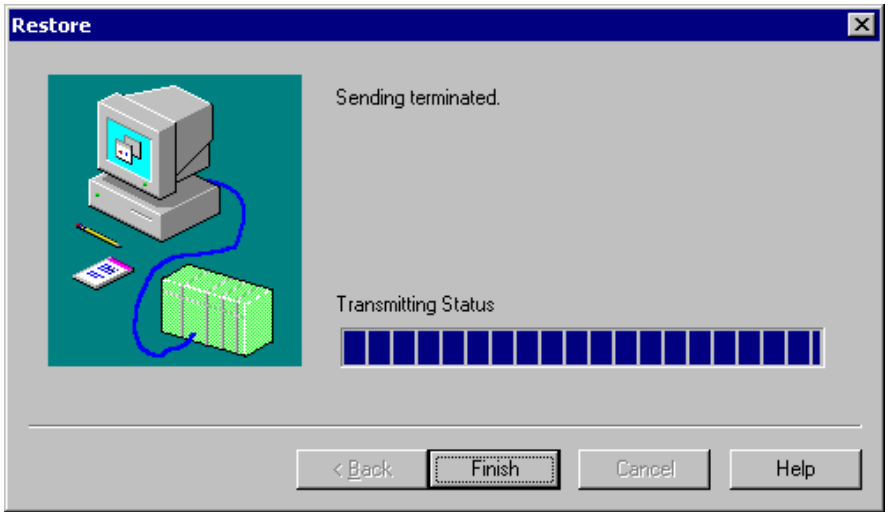


⑦ When the loading is completed, the message shown below appears. Ignoring this message, **power the PCs down and then up again**.

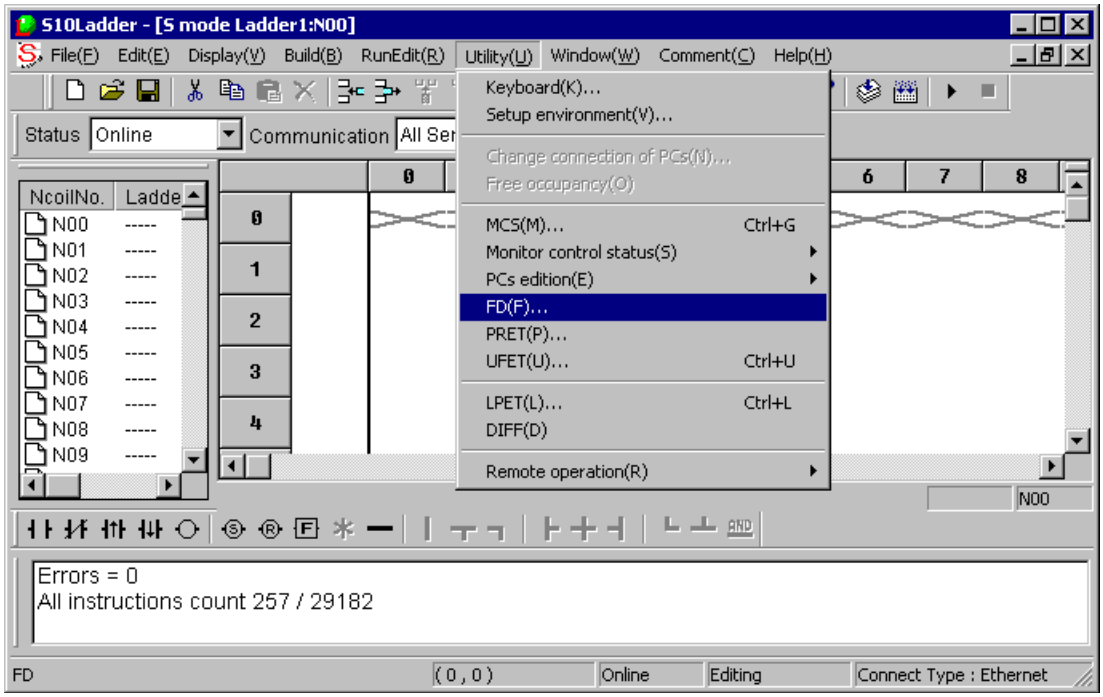


Power the PCs down and then up again.

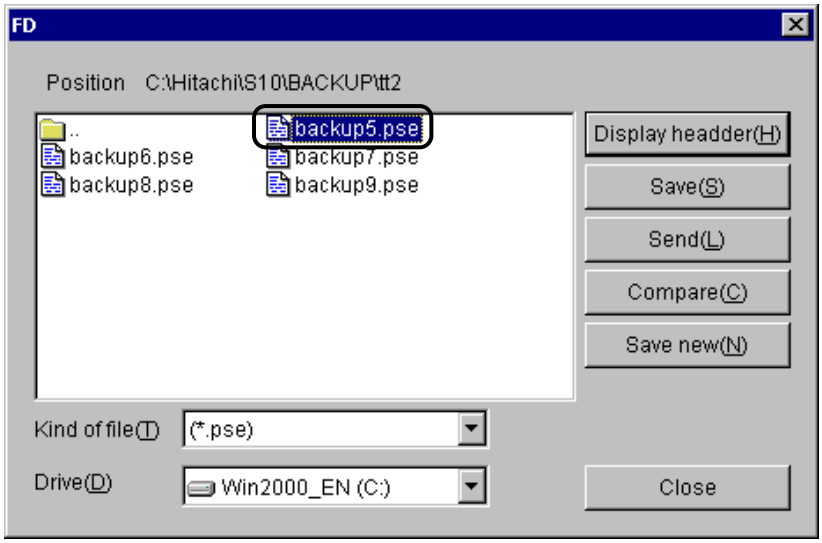
⑧ When the progress-of-loading indication as shown below indicates “100% complete”, the loading of the parameter setting information is completed for all existing modules except the RS-232C modules. Click the Finish button.



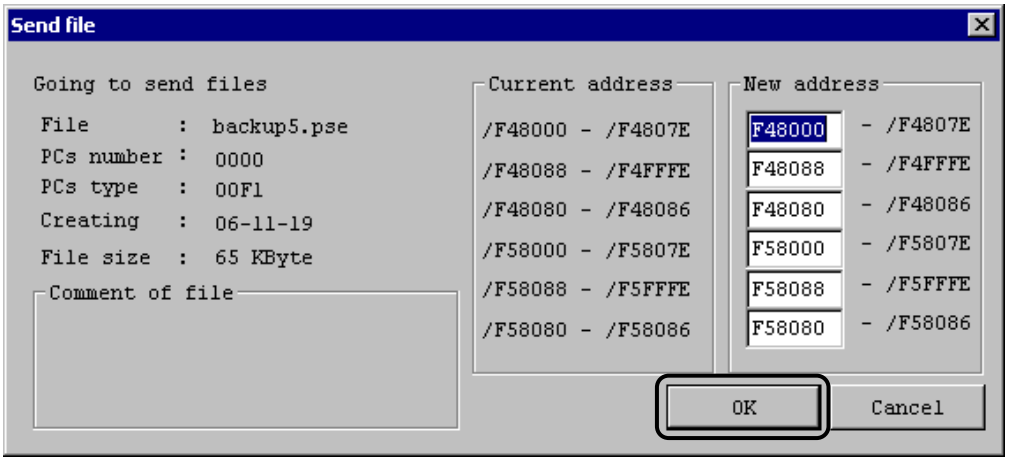
⑨ Launch the ladder chart system and put it into online mode. Then, choose [FD] from the [Utility] pulldown menu.



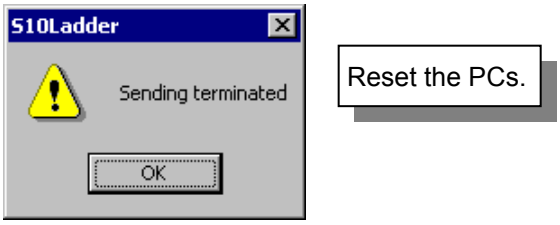
⑩ Open the batch save folder (defaulted to 'C:\hitachi\S10\BACKUP\folder name') and choose 'backup5.pse' (the parameter save file for channels 0 and 1). Then, click the Send button.



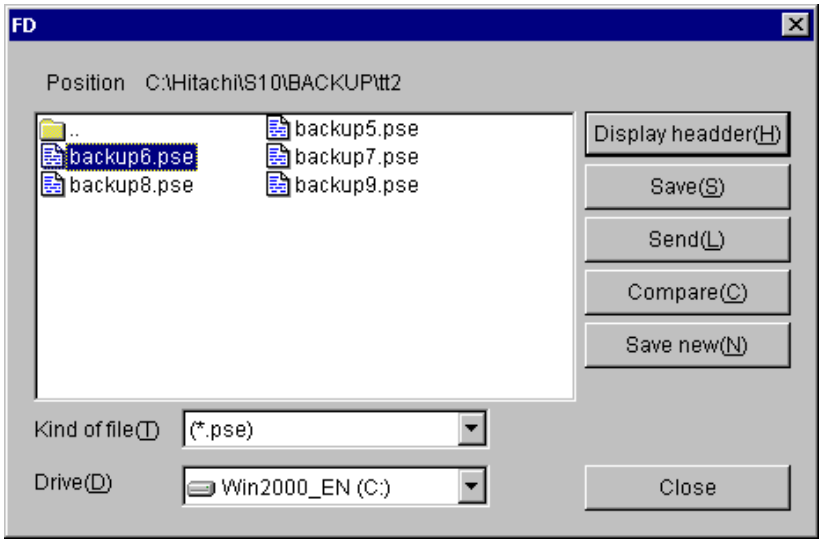
⑪ The file header as shown below will be displayed. When it is displayed, click the OK button. Loading of the file starts.



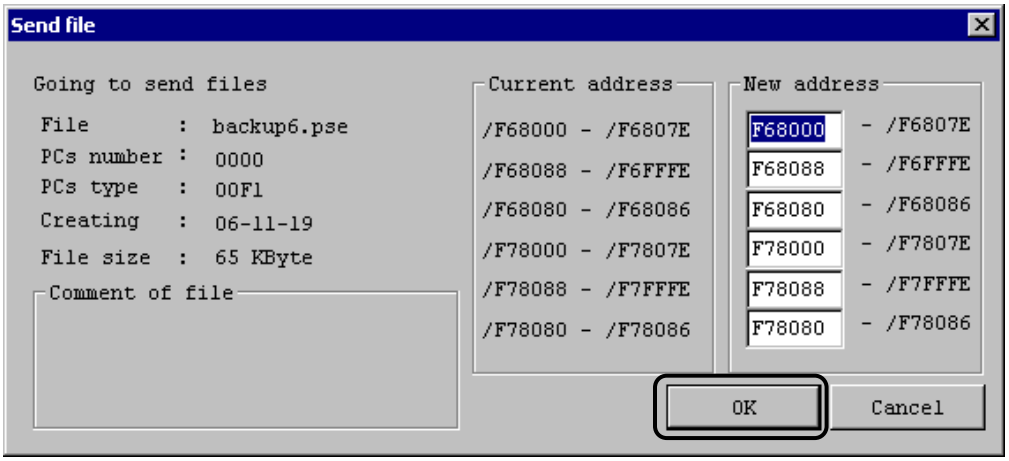
⑫ When the message shown below appears, the loading process is complete. **Reset the PCs.**



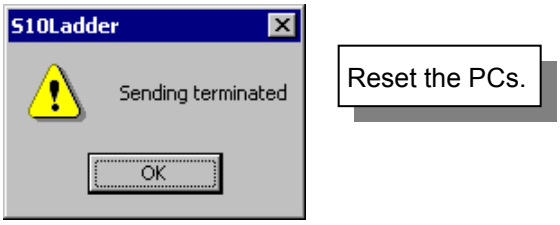
⑬ As you did in Step ⑩, choose 'backup6.pse' (the parameter save file for channels 2 and 3) and click the Send button.



⑭ The file header as shown below will be displayed. When it is displayed, click the OK button. Loading of the second file starts.



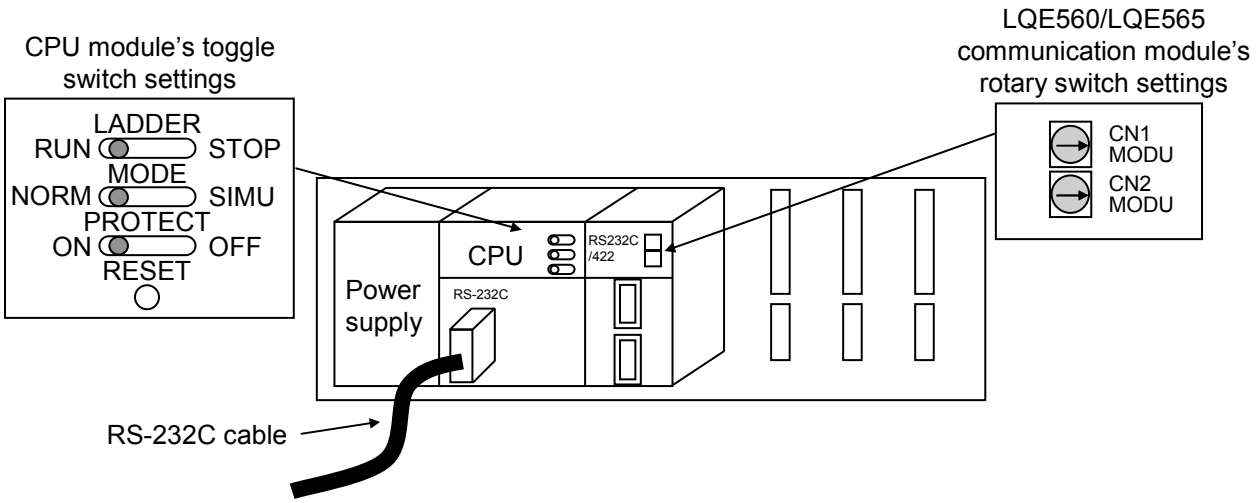
⑮ When the message shown below appears, the loading process is complete. **Reset the PCs.**



7.3 Replacing or Adding on the Module

<Instructions for S10mini users>

- What you should get in preparation
  - ① Personal computer (with Hitachi's S10 External Serial Link System installed in it)
  - ② RS-232C cable (or 10BASE-T cable if the communication module used is an ET.NET module)
  - ③ New or add-on RS-232C or RS-422 module (either the model LQE560 or LQE565)
  - ④ Copies of the parameter values for the module to be replaced. (These copies are prepared for use in cases where the parameters are not accessible for some reason.)
  - ⑤ The above-mentioned ET.NET module is an optional module and, if it is mounted in place, may be selected as the type of communication module to be used. For more information, refer to Section 2.1, "Names and Functions of Each Part," and Section 3.2, "Mounting the Module," in the USER'S MANUAL OPTION ET.NET (LQE520) (manual number SVE-1-103).
- Replacement procedure
  - ① Write down, on a piece of paper, the current settings of the rotary switches (CN1 MODU and CN2 MODU) that are, as shown below, accessible at the front side of the existing RS-232C/RS-422 module to be replaced.
  - ② Write down also the current settings of the toggle switches (LADDER, MODE, and PROTECT) that are, as shown below, accessible at the front side of the CPU module.
  - ③ Connect the personal computer and the CPU module together with the RS-232C cable.



④ Start the S10 External Serial Link System and make a record of the set values of all the existing parameters for each channel:
- Display the "Enter LGB" window for each channel on screen and make a hand-written copy of the set values of the existing parameters for those channels.

  Note: The S10 External Serial Link System does not allow you to use the following functions: "Transmit System Program", "Compare System Program", and "Delete all channels of task system", all of which are provided for use only on S10/2α systems.
- If the existing parameters are not accessible for some reason, use the copies of their set values [item ④] that were obtained in preparation.

⑤ Set the CPU module's LADDER switch in STOP position and turn off the power supply of the controller unit.

⑥ Remove the connecting cable from the RS-232C/RS-422 module.

⑦ Replace the existing RS-232C/RS-422 module with the new one and set the new module's rotary switches in the same way as you wrote down in Step ①.

⑧ Turn on the power supply of the controller unit. Then, from the S10 External Serial Link System, set the parameters for the new RS-232C/RS-422 module in the same way as you wrote down in Step ④.

⑨ Check that all the set parameter values are identical to those that were recorded in Step ④.

⑩ Reset the CPU module by pressing the RESET switch at its front.

⑪ Turn off the power supply of the controller unit.

⑫ Remove the RS-232C cable from both the personal computer and CPU module, which were connected together in Step ③.

⑬ Connect the connecting cable back to the RS-232C/RS-422 module, the cable that was removed from it in Step ⑥.

⑭ Set the CPU module's toggle switches in the same way as you wrote down in Step ②.

⑮ Turn on the power supply of the controller unit and check that the RS-232C/RS-422 module is running normally.

- Add-on procedure

① Write down, on a piece of paper, the current settings of the toggle switches (LADDER, MODE, and PROTECT) that are accessible at the front side of the CPU module.

② Ensure that your application system has been shut down. Then, set the CPU module's LADDER switch in STOP position and turn off the power supply of the controller unit.

③ Mount the add-on RS-232C/RS-422 module in place according to the instructions given under "3.2 Mounting the Module."

④ Set the add-on RS-232C/RS-422 module's CN1 MODU and CN2 MODU No. setting rotary switches according to the instructions given under "2.1 Names and Functions of Each Part."

⑤ Connect the personal computer and the CPU module together with the RS-232C cable. Then, turn on the power supply of the controller unit and set the parameters for the add-on RS-232C/RS-422 module from the S10 External Serial Link System.
Note: The S10 External Serial Link System does not allow you to use the following functions: “Transmit System Program”, “Compare System Program”, and “Delete all channels of task system”, all of which are provided for use only on S10/2α systems.

⑥ Reset the CPU module by pressing the RESET switch at its front.

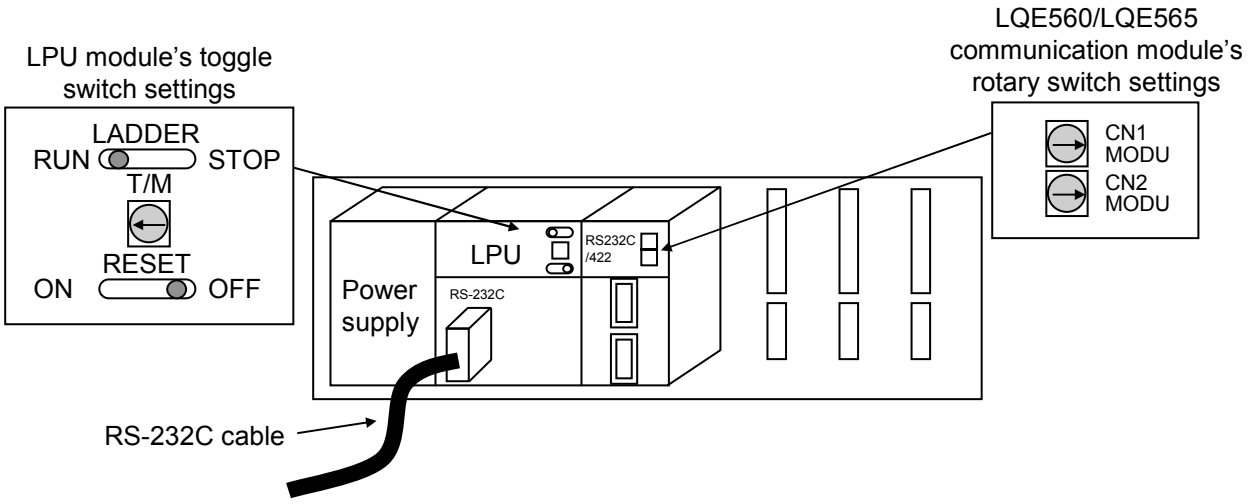
⑦ Turn off the power supply of the controller unit and connect the connecting cable to the add-on RS-232C/RS-422 module.

⑧ Set the CPU module’s toggle switches in the same way as you wrote down in Step ①.

⑨ Turn on the power supply of the controller unit and check that the RS-232C/RS-422 module is running normally.

<Instructions for S10V users>

- What you should get in preparation
  ① Personal computer (with Hitachi’s S10V External Serial Link System installed in it)
  ② RS-232C cable (or 10BASE-T cable if the communication module used is an ET.NET module)
  ③ New or add-on RS-232C or RS-422 module (either the model LQE560 or LQE565)
  ④ Copies of the parameter values for the module to be replaced. (These copies are prepared for use in cases where the parameters are not accessible for some reason.)
  ⑤ The above-mentioned ET.NET module is an optional module and, if it is mounted in place, may be selected as the type of communication module to be used. For more information, refer to Section 2.1, “Names and Functions of Each Part,” and Section 3.2, “Mounting the Module,” in either the USER’S MANUAL OPTION ET.NET (LQE520) (manual number SVE-1-103) or the USER’S MANUAL OPTION ET.NET (LQE720) (manual number SVE-1-128).
- Replacement procedure
  ① Write down, on a piece of paper, the current settings of the rotary switches (CN1 MODU and CN2 MODU) that are, as shown below, accessible at the front side of the existing RS-232C/RS-422 module to be replaced.
  ② Write down also the current settings of the toggle switches (LADDER and T/M) that are, as shown below, accessible at the front side of the LPU module.
  ③ Connect the personal computer and the LPU module together with the RS-232C cable.



④ Start the S10V External Serial Link System and save the set value of the LGB parameter for each channel. To accomplish this saving, see the instructions given in item (5) of Section 4.5.5, "LGB registration," and use the "Save" function in the "Enter LGB" window. (If the LGB parameter is not accessible for some reason, use the copy of its set value [item ④] that was obtained in preparation.)

⑤ Set the LPU module's LADDER switch in STOP position and turn off the power supply of the controller unit.

⑥ Remove the connecting cable from the RS-232C/RS-422 module.

⑦ Replace the existing RS-232C/RS-422 module with the new one and set the new module's rotary switches in the same way as you wrote down in Step ①.

⑧ Turn on the power supply of the controller unit and read in the LGB parameter values that you saved in Step ④, by using the S10V External Serial Link System. To accomplish this reading, see the instructions given in item (6) of Section 4.5.5, "LGB registration," and use the "Load" function in the "Enter LGB" window.

⑨ Write the parameter value read in to the new RS-232C/422 module from the S10V External Serial Link System. To accomplish this writing, see the instructions given in item (4) of Section 4.5.5, "LGB registration," and use the "Write" function in the "Enter LGB" window.

⑩ Check that all the set parameter values are identical to those that were recorded in Step ④.

⑪ Reset the LPU module by setting the RESET switch in ON position and then in OFF position at its front.

⑫ Turn off the power supply of the controller unit.

⑬ Remove the RS-232C cable from both the personal computer and LPU module, which were connected together in Step ③.

⑭ Connect the connecting cable back to the RS-232C/RS-422 module, the cable that was removed from it in Step ⑥.

⑮ Set the LPU module's toggle switches in the same way as you wrote down in Step ②.

⑯ Turn on the power supply of the controller unit and check that the RS-232C/RS-422 module is running normally.

- Add-on procedure

① Write down, on a piece of paper, the current settings of the toggle switches (LADDER and T/M) that are accessible at the front side of the LPU module.

② Ensure that your application system has been shut down. Then, set the LPU module's LADDER switch in STOP position and turn off the power supply of the controller unit.

③ Mount the add-on RS-232C/RS-422 module in place according to the instructions given under "3.2 Mounting the Module."

④ Set the add-on RS-232C/RS-422 module's CN1 MODU and CN2 MODU No. setting rotary switches according to the instructions given under "2.1 Names and Functions of Each Part."

⑤ Connect the personal computer and the LPU module together with the RS-232C cable. Then, turn on the power supply of the controller unit and set the parameters for the add-on RS-232C/RS-422 module from the S10V External Serial Link System.

⑥ Reset the LPU module by setting the RESET switch in ON position and then in OFF position at its front.

⑦ Turn off the power supply of the controller unit and connect the connecting cable to the add-on RS-232C/RS-422 module.

⑧ Set the LPU module's toggle switches in the same way as you wrote down in Step ①.

⑨ Turn on the power supply of the controller unit and check that the RS-232C/RS-422 module is running normally.

7.4 Troubleshooting

7.4.1 CPU module indicator display messages

With the S10mini, one of the messages listed below is displayed in the CPU module indicator when a certain event or error occurs in the RS-232C or RS-422 module.
The S10V collects error information, but does not display errors on the LPU module. Error information collected can be referred from an [[S10V] S10BASE] window if you click Error log button. For more information, refer to "S10V USER'S MANUAL BASIC MODULE (manual number SVE-1-100)."

Table 7-3 S10mini CPU Module Indicator Display Messages

Message	Explanation	User action
R2△ □. □	The module has started up successfully.	
R2△●＊＊＊＊	An error has occurred in the RS-232C or RS-422 module.	Take action as instructed in Subsection 7.4.2 to Subsection 7.4.4.
EXA● PTY	A parity error occurred when the CPU read from the RS-232C or RS-422 module's internal memory.	Reset the CPU using the reset switch. If the display persists, replace the module.

- △ shows "M" for channel number 0 or 1, or "S" for channel number 2 or 3.
- □.□ denotes the version and revision of the module.
- ● denotes a channel number.
- ＊＊＊＊ denotes an error message. For more information, see Subsection 7.4.2 to Subsection 7.4.4.

7.4.2 Hardware errors

When the RS-232C or RS-422 module detects a hardware error, the S10mini displays one of the CPU displays listed in Table 7-4 in the CPU module indicator, S10V displays error log information by clicking an [Error log] button from [[S10V] S10BASE] window.
The ERR LED on the RS-232C or RS-422 module glows and error freeze information is collected at the same time. The RS-232C or RS-422 module shuts down its operation.

Table 7-4 Hardware Errors

Error code	Message	Explanation	User action
/0010	R2△ BUS	Bus error	Reset the CPU. If the same error message recurs, the RS-232C or RS-422 module may have failed. Replace the module.
/0011	R2△ ADDR	Address error	
/0012	R2△ ILLG	Illegal instruction	
/0013	R2△ ZERO	Division by zero	
/0014	R2△ PRIV	Privilege violation	
/0015	R2△ WDT	WDT error	
/0018	R2△ EXCP	Unused exception	
/0019	R2△ PTY	RAM parity error	
/0102	R2△ ROM	ROM checksum error	
/0103	R2△ RAM	RAM check error	
/0100	R2△ MDSW	Invalid module switch setting	Review the module switch setting.
/0112	R2△● LGB	LGB setup error	Reset the LGB table again.

- △ shows “M” for channel number 0 or 1, or “S” for channel number 2 or 3.
- ● denotes a channel number.

7.4.3 Transmit errors

The table below lists the error messages that relate to transmission, and the error codes that are loaded in the system register (S-register).
The S10mini displays error messages in the CPU module indicator, S10V displays error log information by clicking an [Error log] button from [[S10V] S10BASE] window.
Handler errors are not displayed in the CPU indicator.

Table 7-5 Transmit Errors

Error code		Message (S10mini)	Explanation	User action
Error log	S-register			
/4002		Handler error not shown	The send handler was launched while transmission had been broken.	Review the application program.
/4003			Transmission launched while transmission was already in progress.	
/4004			Send handler parameter error	
/4005			The channel to be launched for transmission was under initialization or soft-resetting.	Correct the application so that it will start transmission only when the "ready to send or not" flag indicates "ready to send" (= 0).
			The channel to be launched for transmission shut down.	Reset the CPU. If the same error message recurs, replace the module.
/1080	/2080	R2△● SPRV	Unable to transmit, because data is being received with the remote station prioritized specification.	Transmission starts after the reception completes.
/1081	/2081	R2△● NOCS	Unable to transmit because CS (Clear to Send) input is set to CS Unavailable.	• Verify the correct settings of the control signals of the remote station. • Verify the correct cable connections.
/1082	/2082	R2△● NODR	Unable to transmit because DR (Data Set Ready) input with the DR check specified is set to Not Ready.	
/1083	/2083	R2△● BRTO	Transmission in progress was broken by a break code but no continue code was issued to allow the transmission to resume within the send timeout out period.	• Review the remote station settings and the communications program. • Review the LGB settings.
/1084	/2084	R2△● CSTO	Transmission in progress was broken as CS (Clear to Send) input was set to CS Unavailable. But CS would not set to CS Available during the send timeout period, keeping the transmission from resuming.	• Verify the remote station settings. • Check the cable for breakage.
/1085	/2085	R2△● DRTO	Transmission in progress was broken as DR (Data Set Ready) input was set to Not Ready. But DR would not set to Ready during the send timeout period, keeping the transmission from resuming.	
/1000		R2△● SRBR	Transmission was launched with the local station prioritized specification while reception was in progress. Therefore, the data reception was canceled and the transmission resumed.	Set the LGB priority level to No Priority Level (full-duplex communication).

• △ shows "M" for channel number 0 or 1, or "S" for channel number 2 or 3.
• ● denotes a channel number.

7.4.4 Receive errors

The table below lists the error messages that relate to reception, and the error codes that are loaded in the system register (S-register).
The S10mini displays error messages in the CPU module indicator, S10V displays error log information by clicking an [Error log] button from [[S10V] S10BASE] window.
Handler errors are not displayed in the CPU indicator.

Table 7-6 Receive Errors

Error code	Message (S10mini)	Explanation	User action
/4200	Handler error not shown	Receive handler parameter error	Review the application program.
/4400		An attempt was made to start the receive handler when the module was under initialization or soft-resetting.	Correct the application so that it will start the receive handler only when the receive-completion flag indicates "data received" (= 1).
		The receive handler was launched while the channel shut down.	Reset the CPU. If the same error message recurs, replace the module.
/2080	R2△● RPTY	A parity error occurred in the receive data.	• Verify the LGB settings to meet the communications settings of the remote station. • Check the cable route for sources of noise interference.
/2081	R2△● ROVR	An overrun error occurred in the receive data.	
/2082	R2△● PFRM	A framing error occurred in the receive data.	
/2083	R2△● RVTO	Not all data could be received during the specified period of time.	Review the receive timeout period setting in the LGB table.
/2084	R2△● NOAS	Data other than "0" to "9" and "A" to "T" was received during ASCII conversion.	Review the application running on the remote station.
/2085	R2△● NOEC	Data other than "0" to "9" and "A" to "T" or data other than an end code was received during ASCII conversion.	
/2086	R2△● BCCE	The received BCC did not match during BCC checking.	
/2087	R2△● CDTO	Reception in progress was broken as CD (Carrier Detect) input was set to Off. But CD would not set to On during the receive timeout period, keeping the reception from resuming.	• Verify the remote station settings. • Check the cable for breakage.
/2088	R2△● RVOV	Received data was discarded because all the eight cases of receive buffers were full.	Launch the receive handler to incorporate the received data.
/2089	R2△● RVNZ	Noise was detected in the received data.	Check the cable route for sources of interference.
/2002	R2△● RRBR	Some buffer has been filled with data halfway since transmission was launched while receiving data with the local station prioritized option.	Set the priority level in the LGB table to no priority level (full-duplex communication).

• △ shows "M" for channel number 0 or 1, or "S" for channel number 2 or 3.
• ● denotes a channel number.

7.4.5 Error freeze

When the RS-232C or RS-422 module detects a hardware error, the ERR LED glows and error freeze information is collected at the same time. The RS-232C or RS-422 module shuts down its operation.
The Table 7-7 lists the formats in which error information is presented. For the error codes and stack frames in these formats, see Table 7-8 and Table 7-9.

Table 7-7 Error Freeze Information

Channel				Format	
0	1	2	3	2^{31} ⋯ 2^{16}	2^{15} ⋯ 2^{0}
/F48800	/F58800	/F68800	/F78800	Error code	–
/F48804	/F58804	/F68804	/F78804	Time from reset clear (ms)	
/F48808	/F58808	/F68808	/F78808	Not used	
/F4880C	/F5880C	/F6880C	/F7880C		
/F48810	/F58810	/F68810	/F78810	D0	
/F48814	/F58814	/F68814	/F78814	D1	
/F48818	/F58818	/F68818	/F78818	D2	
/F4881C	/F5881C	/F6881C	/F7881C	D3	
/F48820	/F58820	/F68820	/F78820	D4	
/F48824	/F58824	/F68824	/F78824	D5	
/F48828	/F58828	/F68828	/F78828	D6	
/F4882C	/F5882C	/F6882C	/F7882C	D7	
/F48830	/F58830	/F68830	/F78830	A0	
/F48834	/F58834	/F68834	/F78834	A1	
/F48838	/F58838	/F68838	/F78838	A2	
/F4883C	/F5883C	/F6883C	/F7883C	A3	
/F48840	/F58840	/F68840	/F78840	A4	
/F48844	/F58844	/F68844	/F78844	A5	
/F48848	/F58848	/F68848	/F78848	A6	
/F4884C	/F5884C	/F6884C	/F7884C	A7	
/F48850 | /F488FC	/F58850 | /F588FC	/F68850 | /F688FC	/F78850 | /F788FC	Stack frame	

Table 7-8 Error Code

Code	Explanation	User action
/0010	Bus error	The RS-232C or RS-422 module may have failed. Replace the module.
/0011	Address error	
/0012	Illegal instruction	
/0013	Division by zero	
/0014	Privilege violation	
/0015	WDT error	
/0018	Unused exception	
/0019	RAM parity error	
/0102	ROM checksum error	
/0103	RAM check error	
/0100	Invalid module switch setting	Review the module switch setting.
/0112	LGB setup error	Set the LGB table again.

Table 7-9 Stack Frame Formats

Address	Stack frames other than bus errors/address errors		Stack frames for bus errors/ address errors			
	2^{15} … 2^{0}		2^{15}…2^{5}	2^{4}	2^{3}	2^{2}…2^{0}
/50	Status register			R/W	I/N	FC
/52	Program counter		Access address			
/54						
/56			Instruction register			
/58			Status register			
/5A			Program counter			
/5C						

R/W (read/write): Write = 0, read = 1
I/N (instruction/non-instruction): Instruction = 0, non-instruction = 1
FC: Function code

7.4.6 Communications trace information

The RS-232C or RS-422 module can trace communications information and events. Using this function, trace data can be created to aid in problem determination and corrective action.

Table 7-10 Trace Buffer Structure (Communications Trace Information)

Channel				Format
0	1	2	3	2^{15} ··· 2^{0}
/F4E000	/F5E000	/F6E000	/F7E000	Pointer
/F4E002	/F5E002	/F6E002	/F7E002	Run/Stop
/F4E004	/F5E004	/F6E004	/F7E004	Stop condition type
/F4E006	/F5E006	/F6E006	/F7E006	Trace mode
/F4E008 to /F4E01E	/F5E008 to /F5E01E	/F6E008 to /F6E01E	/F7E008 to /F7E01E	Not used
/F4E020 to /F4E03E	/F5E020 to /F5E03E	/F6E020 to /F6E03E	/F7E020 to /F7E03E	Trace data #00
/F4E040 to /F4E05E	/F5E040 to /F5E05E	/F6E040 to /F6E05E	/F7E040 to /F7E05E	Trace data #01
|	|	|	|	|
/F4FFE0 to /F4FFFE	/F5FFE0 to /F5FFFE	/F6FFE0 to /F6FFFE	/F7FFE0 to /F7FFFE	Trace data #255

- Pointer
  Points to the buffer case in which the next batch of trace data is stored. The pointer is initialized to /20. Valid from /20 to /1FE0.
- Run/Stop
  Run or stop tracing (= 0: stop, ≠ 0: run (default = 1)).
- Stop condition type
  Specify the type of the starting word of the trace data, so that tracing will stop on tracing data of the same type.
- Trace mode
  Specify the trace mode in which tracing executes.
  = 0: Trace stopped
  = 1: Infinite tracing
  = 2: Stop on error (default)
  = 3: Stop on handler trace shutdown
- Trace data
  Trace buffers, which trace data is cyclically stored from #255 back to #0. (See Table 7-11 for more details.)

Table 7-11 Trace Data Details (Communications Trace Information)

Address	Format
/00	Type
/02	Control signal state
/04 | /1A	Transmitted/received data (24 bytes)
/1C	Time from reset clear
/1E	

- Type

  Denotes the distinction between transmission and reception, and an error.

  /1000: Normal transmission

  /2000: Normal reception

  /30**: Transmission error

  /40**: Reception error

  ** denotes the low-order byte of the error code.

- Control signal state

  Stores the control signal I/O state.

  Similar to the information that is incorporated by a latest hardware state incorporate request as described in “5.5 Hardware Controlled by Software Implementation.”

7.4.7 Handler trace information

The RS-232C or RS-422 module can trace handler startups from applications and responses.

Table 7-12 Trace Buffer Structure (Handler Trace Information)

Channel				Format
0	1	2	3	2^{15} ⋯ 2^{0}
/F4D000	/F5D000	/F6D000	/F7D000	Pointer
/F4D002	/F5D002	/F6D002	/F7D002	Run/Stop
/F4D004	/F5D004	/F6D004	/F7D004	Stop condition type
/F4D006	/F5D006	/F6D006	/F7D006	Trace mode
/F4D008	/F5D008	/F6D008	/F7D008	Not used
/F4D010 to /F4D01E	/F5D010 to /F5D01E	/F6D010 to /F6D01E	/F7D010 to /F7D01E	Trace data #00
/F4D020 to /F4D02E	/F5D020 to /F5D02E	/F6D020 to /F6D02E	/F7D020 to /F7D02E	Trace data #01
⏐	⏐	⏐	⏐	⏐
/F4DFF0 to /F4DFFE	/F5DFF0 to /F5DFFE	/F6DFF0 to /F6DFFE	/F7DFF0 to /F7DFFE	Trace data #255

- Pointer
  Points to the buffer case in which the next batch of trace data is stored. The pointer is initialized to /10. Valid from /10 to /FF0.
- Run/Stop
  Run or stop tracing (= 0: stop, ≠ 0: run (default = 1)).
- Stop condition type
  Specify the type of the starting word of the trace data, so that tracing will stop on tracing data of the same type.
- Trace mode
  Specify the trace mode in which tracing executes.
  = 0: Trace stopped
  = 1: Infinite tracing
  = 2: Stop on error (default)
  = 3: Stopped on communications trace shutdown
- Trace data
  Trace buffers, which trace data is cyclically stored from #254 back to #0. (See Table 7-11 for more details.)

Table 7-13 Trace Data Details (Handler Trace Information)

Address	Format
/00	Type
/02	Error code
/04	Parameter 1
/06	
/08	Parameter 2
/0A	
/0C	Time from reset clear
/0E	

- Type
  Denotes the distinction between transmission and reception, and an error.
  /8000: Normal send handler startup
  /9000: Normal receive handler startup
  /8800: Send handler error
  /9800: Receive handler error
- Error code
  Stores a handler error code. For more information, see “7.4.3 Transmit errors,” and “7.4.4 Receive errors.”
- Parameters 1 and 2
  Parameters that are passed on to the handler from the application.

7.4.8 H-7338 error trace information

The RS-232C or RS-422 module can trace errors as they occur during sessions of H-7338 communication, along with the communications data.

Table 7-14 Trace Buffer Structure

Channel				Format
0	1	2	3	2^{31} … 2^{0}
/F48920	/F58920	/F68920	/F78920	Error trace case number
/F48924	/F58924	/F68924	/F78924	Not used
/F48930 to /F4894E	/F58930 to /F5894E	/F68930 to /F6894E	/F78930 to /F7894E	Trace data #0
/F48950 to /F4896E	/F58950 to /F5896E	/F68950 to /F6896E	/F78950 to /F7896E	Trace data #1
|	|	|	|	|
/F48AD0 to /F48AEE	/F58AD0 to /F58AEE	/F68AD0 to /F68AEE	/F78AD0 to /F78AEE	Trace data #13

- Error trace case number
  Points to the buffer case in which the next batch of trace data is stored. The pointer is initialized to /0. Valid from /0 to /0D.
- Trace data
  Trace buffers, which trace data is cyclically stored from #13 back to #0. (See Table 7-15 for more details.)

Table 7-15 Trace Data Details

Address	Format
/00	Error code
/04	Command code
/08	Parameter 1
/0C	Parameter 2
/10	Parameter 3
/14	Parameter 4
/18	Time from reset clear
/1C	Not used

- Error code

  Stores a command and line error code.

Table 7-16 H-7338 Error Trace Error Codes

Error code	Explanation	Corrective action
/00000001	No space was found between parameters.	Verify the remote station settings. Check the cable route for sources of noise interference.
/00000002	A valid parameter range was exceeded.	
/00000101	Receive parity error	Verify the remote station settings. Check also the cable route for breakage and sources of noise interference.
/00000102	Receive overrun error	
/00000103	Receive framing error	
/00000104	Receive noise error	

- Command code

  Stores an H-7338 communications command.
- Parameters 1 to 4

  Store H-7338 communications parameters.

7.4.9 Error Counters

The RS-232C or RS-422 module has error counters to count communications errors as they occur. The error counters are initialized on a reset.

Table 7-17 Error Counters

Channel				Format
0	1	2	3	2^{15} … 2^{0}
/F48900	/F58900	/F68900	/F78900	Normal transmission
/F48902	/F58902	/F68902	/F78902	CS lost while transmitting
/F48904	/F58904	/F68904	/F78904	Send timeout
/F48906	/F58906	/F68906	/F78906	Normal reception
/F48908	/F58908	/F68908	/F78908	Receive overrun error
/F4890A	/F5890A	/F6890A	/F7890A	CD lost while receiving
/F4890C	/F5890C	/F6890C	/F7890C	Receive framing error
/F4890E	/F5890E	/F6890E	/F7890E	Receive parity error
/F48910	/F58910	/F68910	/F78910	Receive noise error
/F48912	/F58912	/F68912	/F78912	Break sequence received
/F48914	/F58914	/F68914	/F78914	Receive timeout
/F48916	/F58916	/F68916	/F78916	Received data discarded counter
/F48918 to /F4891E	/F58918 to /F5891E	/F68918 to /F6891E	/F78918 to /F7891E	Not used

7.5 Trouble Report

Fill out this form and submit it to local source.

Your company name			Person in charge	
Data and time of occurrence	(year / month / day / hour / minute)			
Where to make contact	Address			
	Telephone			
	FAX			
	E-mail			
Model of defective module			CPU/LPU model	
OS Ver. Rev.	Program name: Ver. Rev.			
Support program	Program name: Ver. Rev.			
Symptom of defect				
Connection load	Type			
	Model			
	Wiring state			
System configuration and switch setting				
Space for correspondence				

This Page Intentionally Left Blank