COMPLETE OUR ONLINE SURVEY

Motoman is committed to total customer satisfaction! Please give us your feedback on the technical manuals you received with your Motoman robotic solution.

To participate, go to the following website:

http://www.motoman.com/forms/techpubs.asp
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 INTRODUCTION</td>
<td></td>
</tr>
<tr>
<td>1.1 About this Document</td>
<td>1-1</td>
</tr>
<tr>
<td>1.2 Reference to Other Documentation</td>
<td>1-1</td>
</tr>
<tr>
<td>1.3 Customer Service Information</td>
<td>1-1</td>
</tr>
<tr>
<td>2 SAFETY</td>
<td></td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>2-1</td>
</tr>
<tr>
<td>2.2 Standard Conventions</td>
<td>2-2</td>
</tr>
<tr>
<td>2.3 General Safeguarding Tips</td>
<td>2-4</td>
</tr>
<tr>
<td>2.4 Mechanical Safety Devices</td>
<td>2-4</td>
</tr>
<tr>
<td>2.5 Installation Safety</td>
<td>2-5</td>
</tr>
<tr>
<td>2.6 Programming Safety</td>
<td>2-5</td>
</tr>
<tr>
<td>2.7 Operation Safety</td>
<td>2-6</td>
</tr>
<tr>
<td>2.8 Maintenance Safety</td>
<td>2-7</td>
</tr>
<tr>
<td>3 DATA TRANSMISSION FUNCTION</td>
<td></td>
</tr>
<tr>
<td>1 Outline</td>
<td>1-1</td>
</tr>
<tr>
<td>2 For Using Data Transmission Function</td>
<td>2-1</td>
</tr>
<tr>
<td>3 DCI Function</td>
<td>3-1</td>
</tr>
<tr>
<td>4 Stand-alone Function</td>
<td>4-1</td>
</tr>
<tr>
<td>5 Host Control Function</td>
<td>5-1</td>
</tr>
<tr>
<td>6 Data List</td>
<td>6-1</td>
</tr>
<tr>
<td>7 Comparison of Data Transmission Functions</td>
<td>7-1</td>
</tr>
<tr>
<td>8 Remote Function Setting</td>
<td>8-1</td>
</tr>
</tbody>
</table>
SECTION 1
INTRODUCTION

1.1 About this Document
This manual provides instructions for the Data Transmission Function and contains the following sections:

SECTION 1 – INTRODUCTION
General information about this manual, a list of reference documents, and customer service information.

SECTION 2 – SAFETY
Provides information for the safe use and operation of Motoman products.

SECTION 3 – DATA TRANSMISSION FUNCTION
Provides detailed instructions to utilize the Data Transmission Function.

1.2 Reference to Other Documentation
For additional information refer to the following:

• Concurrent I/O Parameters Manual (P/N 142102-1)
• Operator’s Manual for General Purpose (P/N 142099-1)
• Operator’s Manual for Handling (P/N 142100-1)
• Operator’s Manual for Spot Welding (P/N 142101-1)
• Operator’s Manual for Arc Welding (P/N 142098-1)
• Motoman UP6, XRC Manipulator Manual (P/N 142104-1)
• Motoman SK16X, XRC Manipulator Manual (P/N 142105-1)
• Motoman SK45X, XRC Manipulator Manual (P/N 142106-1)
• Motoman UP130, XRC Manipulator Manual (P/N 142107-1)

1.3 Customer Service Information
If you are in need of technical assistance, contact the Motoman service staff at (937) 847-3200. Please have the following information ready before you call:

• Robot Type (UP6, SK16X, etc.)
• Application Type (welding, handling, etc.)
• Robot Serial Number (located on the back side of the robot arm)
• Robot Sales Order Number (located on back side of XRC controller)
SECTION 2
SAFETY

2.1 Introduction

It is the purchaser's responsibility to ensure that all local, county, state, and national codes, regulations, rules, or laws relating to safety and safe operating conditions for each installation are met and followed.

We suggest that you obtain and review a copy of the ANSI/RIA National Safety Standard for Industrial Robots and Robot Systems. This information can be obtained from the Robotic Industries Association by requesting ANSI/RIA R15.06. The address is as follows:

**Robotic Industries Association**

900 Victors Way  
P.O. Box 3724  
Ann Arbor, Michigan 48106  
TEL: (734) 994-6088  
FAX: (734) 994-3338

Ultimately, the best safeguard is trained personnel. The user is responsible for providing personnel who are adequately trained to operate, program, and maintain the robot cell. **The robot must not be operated by personnel who have not been trained!**

We recommend that all personnel who intend to operate, program, repair, or use the robot system be trained in an approved Motoman training course and become familiar with the proper operation of the system.

This safety section addresses the following:

- Standard Conventions (Section 2.2)
- General Safeguarding Tips (Section 2.3)
- Mechanical Safety Devices (Section 2.4)
- Installation Safety (Section 2.5)
- Programming Safety (Section 2.6)
- Operation Safety (Section 2.7)
- Maintenance Safety (Section 2.8)
2.2 **Standard Conventions**

This manual includes information essential to the safety of personnel and equipment. As you read through this manual, be alert to the four signal words:

- DANGER
- WARNING
- CAUTION
- NOTE

Pay particular attention to the information provided under these headings which are defined below (in descending order of severity).

**DANGER!**

Information appearing under the DANGER caption concerns the protection of personnel from the immediate and imminent hazards that, if not avoided, will result in immediate, serious personal injury or loss of life in addition to equipment damage.

**WARNING!**

Information appearing under the WARNING caption concerns the protection of personnel and equipment from potential hazards that can result in personal injury or loss of life in addition to equipment damage.

**CAUTION!**

Information appearing under the CAUTION caption concerns the protection of personnel and equipment, software, and data from hazards that can result in minor personal injury or equipment damage.

**NOTE:** Information appearing in a NOTE caption provides additional information which is helpful in understanding the item being explained.
2.3 **General Safeguarding Tips**

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. General safeguarding tips are as follows:

- Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation of this robot, the operator's manuals, the system equipment, and options and accessories should be permitted to operate this robot system.
- Do not enter the robot cell while it is in automatic operation. Programmers must have the teach pendant when they enter the robot cell.
- Improper connections can damage the robot. All connections must be made within the standard voltage and current ratings of the robot I/O (Inputs and Outputs).
- The robot must be placed in Emergency Stop (E-STOP) mode whenever it is not in use.
- In accordance with ANSI/RIA R15.06, section 6.13.4 and 6.13.5, use lockout/tagout procedures during equipment maintenance. Refer also to Section 1910.147 (29CFR, Part 1910), Occupational Safety and Health Standards for General Industry (OSHA).

2.4 **Mechanical Safety Devices**

The safe operation of the robot, positioner, auxiliary equipment, and system is ultimately the user's responsibility. The conditions under which the equipment will be operated safely should be reviewed by the user. The user must be aware of the various national codes, ANSI/RIA R15.06 safety standards, and other local codes that may pertain to the installation and use of industrial equipment. Additional safety measures for personnel and equipment may be required depending on system installation, operation, and/or location. The following safety measures are available:

- Safety fences and barriers
- Light curtains
- Door interlocks
- Safety mats
- Floor markings
- Warning lights

Check all safety equipment frequently for proper operation. Repair or replace any non-functioning safety equipment immediately.
2.5 **Installation Safety**

Safe installation is essential for protection of people and equipment. The following suggestions are intended to supplement, but not replace, existing federal, local, and state laws and regulations. Additional safety measures for personnel and equipment may be required depending on system installation, operation, and/or location. Installation tips are as follows:

- Be sure that only qualified personnel familiar with national codes, local codes, and ANSI/RIA R15.06 safety standards are permitted to install the equipment.
- Identify the work envelope of each robot with floor markings, signs, and barriers.
- Position all controllers outside the robot work envelope.
- Whenever possible, install safety fences to protect against unauthorized entry into the work envelope.
- Eliminate areas where personnel might get trapped between a moving robot and other equipment (pinch points).
- Provide sufficient room inside the workcell to permit safe teaching and maintenance procedures.

2.6 **Programming Safety**

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. Programming tips are as follows:

- Any modifications to PART 1 of the MRC controller PLC can cause severe personal injury or death, as well as damage to the robot! Do not make any modifications to PART 1. Making any changes without the written permission of Motoman will **VOID YOUR WARRANTY**!
- Some operations require standard passwords and some require special passwords. Special passwords are for Motoman use only. **YOUR WARRANTY WILL BE VOID** if you use these special passwords.
- Back up all programs and jobs onto a floppy disk whenever program changes are made. To avoid loss of information, programs, or jobs, a backup must always be made before any service procedures are done and before any changes are made to options, accessories, or equipment.
- The concurrent I/O (Input and Output) function allows the customer to modify the internal ladder inputs and outputs for maximum robot performance. Great care must be taken when making these modifications. Double-check all modifications under every mode of robot operation to ensure that you have not created hazards or dangerous situations that may damage the robot or other parts of the system.
- Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation, manuals, electrical design, and equipment interconnections of this robot should be permitted to operate the system.
• Inspect the robot and work envelope to be sure no potentially hazardous conditions exist. Be sure the area is clean and free of water, oil, debris, etc.
• Be sure that all safeguards are in place.
• Check the E-STOP button on the teach pendant for proper operation before programming.
• Carry the teach pendant with you when you enter the workcell.
• Be sure that only the person holding the teach pendant enters the workcell.
• Test any new or modified program at low speed for at least one full cycle.

2.7 Operation Safety

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. Operation tips are as follows:

• Be sure that only trained personnel familiar with the operation of this robot, the operator's manuals, the system equipment, and options and accessories are permitted to operate this robot system.
• Check all safety equipment for proper operation. Repair or replace any non-functioning safety equipment immediately.
• Inspect the robot and work envelope to ensure no potentially hazardous conditions exist. Be sure the area is clean and free of water, oil, debris, etc.
• Ensure that all safeguards are in place.
• Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation, manuals, electrical design, and equipment interconnections of this robot should be permitted to operate the system.
• Do not enter the robot cell while it is in automatic operation. Programmers must have the teach pendant when they enter the cell.
• The robot must be placed in Emergency Stop (E-STOP) mode whenever it is not in use.
• This equipment has multiple sources of electrical supply. Electrical interconnections are made between the controller, external servo box, and other equipment. Disconnect and lockout/tagout all electrical circuits before making any modifications or connections.
• All modifications made to the controller will change the way the robot operates and can cause severe personal injury or death, as well as damage the robot. This includes controller parameters, ladder parts 1 and 2, and I/O (Input and Output) modifications. Check and test all changes at slow speed.
2.8 **Maintenance Safety**

All operators, programmers, plant and tooling engineers, maintenance personnel, supervisors, and anyone working near the robot must become familiar with the operation of this equipment. All personnel involved with the operation of the equipment must understand potential dangers of operation. Maintenance tips are as follows:

- Do not perform any maintenance procedures before reading and understanding the proper procedures in the appropriate manual.
- Check all safety equipment for proper operation. Repair or replace any non-functioning safety equipment immediately.
- Improper operation can result in personal injury and/or damage to the equipment. Only trained personnel familiar with the operation, manuals, electrical design, and equipment interconnections of this robot should be permitted to operate the system.
- Back up all your programs and jobs onto a floppy disk whenever program changes are made. A backup must always be made before any servicing or changes are made to options, accessories, or equipment to avoid loss of information, programs, or jobs.
- Do not enter the robot cell while it is in automatic operation. Programmers must have the teach pendant when they enter the cell.
- The robot must be placed in Emergency Stop (E-STOP) mode whenever it is not in use.
- Be sure all safeguards are in place.
- Use proper replacement parts.
- This equipment has multiple sources of electrical supply. Electrical interconnections are made between the controller, external servo box, and other equipment. Disconnect and lockout/tagout all electrical circuits before making any modifications or connections.
- All modifications made to the controller will change the way the robot operates and can cause severe personal injury or death, as well as damage the robot. This includes controller parameters, ladder parts 1 and 2, and I/O (Input and Output) modifications. Check and test all changes at slow speed.
- Improper connections can damage the robot. All connections must be made within the standard voltage and current ratings of the robot I/O (Inputs and Outputs).
Upon receipt of the product and prior to initial operation, read these instructions thoroughly, and retain for future reference.

The YASNAC XRC operator’s manuals above correspond to specific usage. Be sure to use the appropriate manual.
MANDATORY

• This manual explains the data transmission function of the YASNAC XRC system and general operations. Read this manual carefully and be sure to understand its contents before handling the YASNAC XRC.

• General items related to safety are listed in Section 1: Safety of the Setup Manual. To ensure correct and safe operation, carefully read the Setup Manual before reading this manual.

CAUTION

• Some drawings in this manual are shown with the protective covers or shields removed for clarity. Be sure all covers and shields are replaced before operating this product.

• The drawings and photos in this manual are representative examples and differences may exist between them and the delivered product.

• YASKAWA may modify this model without notice when necessary due to product improvements, modifications, or changes in specifications. If such modification is made, the manual number will also be revised.

• If your copy of the manual is damaged or lost, contact a YASKAWA representative to order a new copy. The representatives are listed on the back cover. Be sure to tell the representative the manual number listed on the front cover.

• YASKAWA is not responsible for incidents arising from unauthorized modification of its products. Unauthorized modification voids your product’s warranty.
NOTES FOR SAFE OPERATION
Read this manual carefully before installation, operation, maintenance, or inspection of the YASNAC XRC.
In this manual, the Notes for Safe Operation are classified as “WARNING”, “CAUTION”, “MANDATORY”, or "PROHIBITED".

- **WARNING**: Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury to personnel.
- **CAUTION**: Indicates a potentially hazardous situation which, if not avoided, could result in minor or moderate injury to personnel and damage to equipment. It may also be used to alert against unsafe practices.
- **MANDATORY**: Always be sure to follow explicitly the items listed under this heading.
- **PROHIBITED**: Must never be performed.

Even items described as “CAUTION” may result in a serious accident in some situations. At any rate, be sure to follow these important items.

**NOTE**: To ensure safe and efficient operation at all times, be sure to follow all instructions, even if not designated as “CAUTION” and “WARNING”.

• Before operating the manipulator, check that servo power is turned off when the emergency stop buttons on the playback panel or programming pendant are pressed. When the servo power is turned off, the SERVO ON READY lamp on the playback panel and the SERVO ON LED on the programming pendant are turned off.

Injury or damage to machinery may result if the emergency stop circuit cannot stop the manipulator during an emergency. The manipulator should not be used if the emergency stop buttons do not function.

Emergency Stop Button

• Once the emergency stop button is released, clear the cell of all items which could interfere with the operation of the manipulator. Then turn the servo power ON

Injury may result from unintentional or unexpected manipulator motion.

Release of Emergency Stop

• Always set the Teach Lock before entering the robot work envelope to teach a job.

Operator injury can occur if the Teach Lock is not set and the manipulator is started from the playback panel.

• Observe the following precautions when performing teaching operations within the working envelope of the manipulator:
  - View the manipulator from the front whenever possible.
  - Always follow the predetermined operating procedure.
  - Ensure that you have a safe place to retreat in case of emergency.

Improper or unintended manipulator operation may result in injury.

• Confirm that no persons are present in the manipulator’s work envelope and that you are in a safe location before:
  - Turning on the YASNAC XRC power
  - Moving the manipulator with the programming pendant
  - Running check operations
  - Performing automatic operations

Injury may result if anyone enters the working envelope of the manipulator during operation. Always press an emergency stop button immediately if there are problems. The emergency stop button is located on the right side of both the YASNAC XRC playback panel and programming pendant.
Definition of Terms Used Often in This Manual

The MOTOMAN manipulator is the YASKAWA industrial robot product.
The manipulator usually consists of the controller, the playback panel, the programming pendant, and supply cables.
The MOTOMAN manipulator is the YASKAWA industrial robot product.
In this manual, the equipment is designated as follows.

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Manual Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>YASNAC XRC Controller</td>
<td>XRC</td>
</tr>
<tr>
<td>YASNAC XRC Playback Panel</td>
<td>Playback Panel</td>
</tr>
<tr>
<td>YASNAC XRC Programming Pendant</td>
<td>Programming Pendant</td>
</tr>
</tbody>
</table>

CAUTION

• Perform the following inspection procedures prior to conducting manipulator teaching. If problems are found, repair them immediately, and be sure that all other necessary processing has been performed.
  - Check for problems in manipulator movement.
  - Check for damage to insulation and sheathing of external wires.

• Always return the programming pendant to the hook on the XRC cabinet after use.

  The programming pendant can be damaged if it is left in the manipulator’s work area, on the floor, or near fixtures.

• Read and understand the Explanation of the Alarm Display in the setup manual before operating the manipulator.
Descriptions of the programming pendant and playback panel keys, buttons, and displays are shown as follows:

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Manual Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming Pendant</td>
<td></td>
</tr>
<tr>
<td>Character Keys</td>
<td>The keys which have characters printed on them are denoted with [ ].</td>
</tr>
<tr>
<td></td>
<td>ex. [ENTER]</td>
</tr>
<tr>
<td>Symbol Keys</td>
<td>The keys which have a symbol printed on them are not denoted with [ ] but depicted</td>
</tr>
<tr>
<td></td>
<td>with a small picture.</td>
</tr>
<tr>
<td></td>
<td>ex. page key [ ]</td>
</tr>
<tr>
<td></td>
<td>The cursor key is an exception, and a picture is not shown.</td>
</tr>
<tr>
<td>Axis Keys</td>
<td>“Axis Keys” and “Number Keys” are generic names for the keys for axis operation and</td>
</tr>
<tr>
<td>Number Keys</td>
<td>number input.</td>
</tr>
<tr>
<td>Keys pressed simultaneously</td>
<td>When two keys are to be pressed simultaneously, the keys are shown with a “+” sign</td>
</tr>
<tr>
<td></td>
<td>between them.</td>
</tr>
<tr>
<td></td>
<td>ex. [SHIFT]+[COORD]</td>
</tr>
<tr>
<td>Displays</td>
<td>The menu displayed in the programming pendant is denoted with { }.</td>
</tr>
<tr>
<td></td>
<td>ex. {JOB}</td>
</tr>
<tr>
<td>Playback Panel</td>
<td></td>
</tr>
<tr>
<td>Buttons</td>
<td>Playback panel buttons are enclosed in brackets. ex. [TEACH] on the playback panel</td>
</tr>
</tbody>
</table>

**Description of the Operation Procedure**

In the explanation of the operation procedure, the expression "Select • • • " means that the cursor is moved to the object item and the SELECT key is pressed.
1 Outline

1.1 DCI Function .................................................1-2
1.2 Stand-alone Function ....................................1-3
1.3 Host Control Function .................................1-4

2 For Using Data Transmission Function

2.1 Remote Mode ................................................2-1
  2.1.1 Remote Mode ....................................2-1
  2.1.2 Command Remote Valid/Invalid ......................2-3
  2.1.3 Display in Command Remote Mode ..................2-3
2.2 Serial I/F Port Assignment ............................2-4
2.3 Parallel Operation of XRC .............................2-5
  2.3.1 No Multiple-operation of DCI, Stand-alone, and Host Control Functions 2-5
  2.3.2 File Access and Editing for a Single Target ........2-5
2.4 Differences from MRC ....................................2-6
  2.4.1 Multiport Processing ................................2-6
  2.4.2 Group Axes .......................................2-6
  2.4.3 Coordinated Operation and Independent Operation ....2-6
  2.4.4 Condition Data and System Data ....................2-6
2.5 Transmission Specifications .........................2-7
  2.5.1 Basic Specifications ................................2-7
  2.5.2 Transmission Control Characters ..................2-7
  2.5.3 Transmission Format ................................2-8
  2.5.4 Error Control System ................................2-9
  2.5.5 Character Configuration ............................2-9
  2.5.6 Data Link Establishment ............................2-9
  2.5.7 Configuration of Heading and Text .................2-10
  2.5.8 Transmission Parameters .........................2-10
    ■ Transmission Control Monitoring Timer ............2-10
    ■ Transmission Control Resending Sequence .........2-11
  2.5.9 Connection of D-SUB Connector Pins ...............2-12
  2.5.10 Connection ......................................2-12

3 DCI Function

3.1 Outline ..................................................3-1
3.2 Commands for Job Transmission .......................3-1
  3.2.1 LOADJ ............................................3-1
    ■ Function ..........................................3-1
    ■ Configuration .....................................3-2
  3.2.2 SAVEJ ............................................3-3
3.2.3 DELETEJ .................................................. 3-3
  ■ Function ................................................. 3-3
  ■ Configuration ......................................... 3-3
3.2.4 SWAIT .................................................. 3-4
  ■ Function ................................................. 3-4
  ■ Configuration ......................................... 3-4

3.3 Commands for Variable Transmission ............... 3-4
3.3.1 LOADV .................................................. 3-4
  ■ Function ................................................. 3-4
  ■ Configuration ......................................... 3-4
3.3.2 SAVEV .................................................. 3-4
  ■ Function ................................................. 3-4
  ■ Configuration ......................................... 3-4

3.4 Registrating DCI Instruction ......................... 3-5
3.5 Concurrent Tasks from Multiple Jobs ................ 3-7
3.6 DCI Parallel Execution ................................ 3-8
  ■ Parallel Execution Using NWAIT ...................... 3-8
  ■ Parallel Execution Using PSTART (Optional) ....... 3-9

3.7 Transmission Procedure .............................. 3-9
3.7.1 Job Transmission .................................... 3-9
  ■ Saving Procedure ...................................... 3-9
  ■ Loading Procedure .................................... 3-10
3.7.2 Variable Transmission .............................. 3-12
  ■ Saving Procedure ...................................... 3-12
  ■ Loading Procedure .................................... 3-12

3.8 Axis Data Transmission Format ...................... 3-14
3.9 Alarm Codes ............................................ 3-15

4 Stand-alone Function
4.1 Outline .................................................. 4-1
4.2 Operation Flow ........................................ 4-2
4.3 Operation ............................................... 4-3
  4.3.1 Selecting External Memory Unit .................... 4-3
  4.3.2 Save .................................................. 4-4
    ■ Saving Job ........................................... 4-4
    ■ Saving File .......................................... 4-5
  4.3.3 Load .................................................. 4-7
    ■ Loading Job .......................................... 4-7
    ■ Loading File ......................................... 4-8
  4.3.4 Job Selection Mode ................................ 4-9
    ■ Single Selection Mode ............................... 4-9
    ■ Related Selection Mode .............................. 4-9
5 Host Control Function

5.1 File Data Transmission Function
5.1.1 Transmission Procedure
5.1.2 Data Management

5.2 Robot Control Function
5.2.1 Command Transmission
5.2.2 List of Interlock for Commands of Host Control Function
5.2.3 Command that Handle Axis Data
5.2.4 Response to MOV-type Command
5.2.5 Status Read Function
5.2.6 System Control Function
5.2.7 I/O Read/Write Function

5.3 Commands for Multi-control Group and Independent Control Functions
5.3.1 Commands for Multi-control Group
5.3.2 Commands for Independent Control Function

5.4 Alarm Codes
5.5 Interpreter Message List

6 Data List
6.1 Header Number List
6.2 Parameter List
7  Comparison of Data Transmission Functions

8  Remote Function Setting
1 Outline

The data transmission function is for communication with a host computer such as a personal computer in BSC complying protocol.

The data transmission function adopts a serial transmission line and standard protocol, making easy connection to a host computer.

The data transmission function is not only for transmission of job but also for controlling robot system by a host computer using a set of commands. The robot commands in the ASCII code command format are easy to use and helpful for a quick development of necessary software to be run on the host computer.

The data transmission function is divided into the following three functions.

- DCI (Data Communication by Instruction)
- Stand-alone function
- Host control function
The DCI function executes instructions described in a job to perform data transmission with a host computer. This function loads and saves jobs and variables.

<table>
<thead>
<tr>
<th>DCI Function</th>
<th>Load</th>
<th>Job can be transmitted in either mode.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job</td>
<td>Save</td>
<td>• Single job</td>
</tr>
<tr>
<td>Transmission</td>
<td>Delete</td>
<td>• Related job</td>
</tr>
<tr>
<td>Variable</td>
<td>Load</td>
<td>• Byte type global variables</td>
</tr>
<tr>
<td>Transmission</td>
<td>Save</td>
<td>• Integer type global variables</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Double precision type global variables</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Real number type global variables</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Position type global variables</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Robot axes, base axes, station axes)</td>
</tr>
</tbody>
</table>
1.2 Stand-alone Function

The stand-alone function is for data transmission with host computer by operation on the programming pendant. This function loads and saves jobs and condition data.

![Diagram of stand-alone function](image)

<table>
<thead>
<tr>
<th>Job Transmission</th>
<th>Load</th>
<th>Save</th>
<th>Verify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job can be transmitted in either</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mode.</td>
<td>• Single job</td>
<td>• Tool data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Related job</td>
<td>• Weaving data</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• User coordinate data</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Welding data</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Variable data</td>
<td></td>
</tr>
</tbody>
</table>

| Condition Data/ General Data      | Load                                      | Save                                      | Verify          |
| Transmission                      |                                            |                                            |                 |
|                                   | • Tool data                               | • System information                      |                 |
|                                   | • Weaving data                            | • Alarm history                           |                 |
|                                   | • User coordinate data                    |                                            |                 |
|                                   | • Welding data                            |                                            |                 |
|                                   | • Variable data                           |                                            |                 |

| System Information Transmission   | Save                                      |                                            |                 |
|                                   | • System information                      |                                            |                 |
|                                   | • Alarm history                           |                                            |                 |
1.3 Host Control Function

The host control function is for loading and saving jobs, reading robot status, and controlling the system by sending a command from a host computer.

![Diagram of XRC and Host computer](image)

<table>
<thead>
<tr>
<th>File Data Transmission Function</th>
<th>Load</th>
<th>Save</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Transmission</td>
<td>Jobs can be transmitted in either mode:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Single job</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Related job</td>
<td></td>
</tr>
<tr>
<td>Condition Data/General Data Transmission</td>
<td>Load</td>
<td>Save</td>
</tr>
<tr>
<td></td>
<td>• Tool data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Weaving data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• User coordinate data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Welding data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Variable data</td>
<td></td>
</tr>
<tr>
<td>System Information Transmission</td>
<td>Save</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• System information</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Alarm history</td>
<td></td>
</tr>
<tr>
<td>Host Control Function</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Status Reading</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of error and alarm codes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of current position in a joint coordinate system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of current position in a specified Cartesian coordinate system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of mode, cycle, motion, alarm error and servo status</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of current job name, line No. and step No.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of all job names or related job names</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Monitoring completion of manipulator operation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of specified user coordinate data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of control group and task selected status</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Read of variable data</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Robot Control Function</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Start, hold</td>
</tr>
<tr>
<td>• Reset, cancel</td>
</tr>
<tr>
<td>• Job deletion</td>
</tr>
<tr>
<td>• Master job setup</td>
</tr>
<tr>
<td>• Job, line No. and step No. setup</td>
</tr>
<tr>
<td>• Mode and cycle selection</td>
</tr>
<tr>
<td>• Servo power supply ON/OFF</td>
</tr>
<tr>
<td>• Programming pendant and playback panel interlock setup/release</td>
</tr>
<tr>
<td>• Message display</td>
</tr>
<tr>
<td>• Joint motion and linear motion to a specified Cartesian coordinate system</td>
</tr>
<tr>
<td>• Linear motion by increments in a specified coordinate system</td>
</tr>
<tr>
<td>• Joint motion and linear motion to a specified joint coordinate system</td>
</tr>
<tr>
<td>• Conversion/reverse conversion of related job of a specified job (Relative job function is necessary)</td>
</tr>
<tr>
<td>• Write of specified user coordinate data</td>
</tr>
<tr>
<td>• Change of control group</td>
</tr>
<tr>
<td>• Change of task to be controlled</td>
</tr>
<tr>
<td>• Write of variable data</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>System Control</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Start, hold</td>
</tr>
<tr>
<td>• Reset, cancel</td>
</tr>
<tr>
<td>• Job deletion</td>
</tr>
<tr>
<td>• Master job setup</td>
</tr>
<tr>
<td>• Job, line No. and step No. setup</td>
</tr>
<tr>
<td>• Mode and cycle selection</td>
</tr>
<tr>
<td>• Servo power supply ON/OFF</td>
</tr>
<tr>
<td>• Programming pendant and playback panel interlock setup/release</td>
</tr>
<tr>
<td>• Message display</td>
</tr>
<tr>
<td>• Joint motion and linear motion to a specified Cartesian coordinate system</td>
</tr>
<tr>
<td>• Linear motion by increments in a specified coordinate system</td>
</tr>
<tr>
<td>• Joint motion and linear motion to a specified joint coordinate system</td>
</tr>
<tr>
<td>• Conversion/reverse conversion of related job of a specified job (Relative job function is necessary)</td>
</tr>
<tr>
<td>• Write of specified user coordinate data</td>
</tr>
<tr>
<td>• Change of control group</td>
</tr>
<tr>
<td>• Change of task to be controlled</td>
</tr>
<tr>
<td>• Write of variable data</td>
</tr>
</tbody>
</table>
1.3 Host Control Function
2.1 Remote Mode

The data transmission function can be used with XRC in remote mode.

2.1.1 Remote Mode

To use the data transmission function, set XRC to remote mode.

In remote mode, the operation is ordered from a host computer; whereas in local mode, teach mode, and play mode, the programming pendant and playback panel are used for operating the system.

To switch to the remote mode or the local mode, either

- Press [REMOTE] on the playback panel, or
- Turn ON the remote selection signal of the external input signals.

The remote mode has two sub-modes; “I/O remote enable” and “Command remote enable”. Which sub-mode takes effect in remote mode is set in the pseudo input display. Refer to Section 8 “Remote Function Setting”.

![Diagram showing mode transitions]

- Local mode
- Teach mode
- Play mode
- Remote mode
  - I/O remote enable
  - Command remote enable
In remote mode, operations on the programming pendant or the playback panel are valid except the operation-related entries. This holds true in “I/O remote enable” and “Command remote enable” submodes. The concept is based on the conventional I/O control introduced to command control.

Note that the edit-related operations can not be entered from more than one operating device.

In “Command remote enable” submode, to enable command remote controls only, issue the HLOCK command. When the HLOCK command is ON, operations on the programming pendant and playback panel are valid only hold and emergency stop. In this state, remote mode cannot be canceled, and the following I/O operations are disabled: selection between remote mode and local mode, external start, external servo ON, cycle selection, I/O prohibit, P.P/P PANEL prohibit, and master job call. Other I/O operations are valid.
2.1.2 Command Remote Valid/Invalid

Availability of each function of data transmission differs depending on the command remote setting (Enabled / Disabled).

When the command remote is set invalid, the read/monitor system commands (hereinafter called read-only function) in the host control function in addition to the DCI function and stand-alone function can be used. For the details of read/monitor system commands, refer to Section 5.2.2 “List of Interlock for Commands of Host Control Function”.

<table>
<thead>
<tr>
<th>Command Remote Setting</th>
<th>Function Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid</td>
<td>DCI function available</td>
</tr>
<tr>
<td></td>
<td>Stand-alone function available</td>
</tr>
<tr>
<td></td>
<td>Host control function (only read-only function) available</td>
</tr>
<tr>
<td>Valid</td>
<td>Host control function (all commands) available</td>
</tr>
</tbody>
</table>

To validate the read-only function in the above host control function, set the parameter RS005 to “1”. When the command remote is validated by pressing [REMOTE] with the read-only function valid, the command remote status is entered so that all commands can be used. When the command remote is invalidated by pressing [REMOTE] again, the read-only function becomes validated again.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents and Set Value</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS005</td>
<td>BSC port function specification when the command remote is invalidated</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0 : DCI or stand-alone function</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : Read-only function in host control</td>
<td></td>
</tr>
</tbody>
</table>

2.1.3 Display in Command Remote Mode

Even in command remote enabled submode, it is not necessary to call the command remote display because operations from XRC is available. To call the command remote display, select “REMOTE” from “I/O” under the top menu.

This display is used in common with the I/O remote mode display.

The message in the remote display changes according to the remote function selection. (Refer to Chapter 8 “Remote Function Setting”.)
2.2 Serial I/F Port Assignment

The XRC has one serial interface port. The FC1 protocol and the BSC complying protocol (for data transmission function : option) can be assigned to the port to communicate with external devices.

A change in assignment can be made only in local mode.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents and Set Value</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS000</td>
<td>Standard port protocol specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 : NON</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : System reserved</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : BSC LIKE (Data transmission function)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : FC1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
2.3 Parallel Operation of XRC

The XRC is capable of parallel processing. For instance, it can check signals with programming pendant while saving files to YASNAC FC2, or can edit files with the programming pendant while monitoring operation status by the host control function.

The parallel operation has the following restrictions. When an operation against these restrictions is made, a warning message is displayed.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Warning</th>
</tr>
</thead>
<tbody>
<tr>
<td>YASNAC FC2</td>
<td>Error message for 3 seconds</td>
</tr>
<tr>
<td>Stand-alone</td>
<td></td>
</tr>
<tr>
<td>Programming pendant</td>
<td></td>
</tr>
<tr>
<td>Playback panel</td>
<td></td>
</tr>
<tr>
<td>DCI</td>
<td>Alarm</td>
</tr>
<tr>
<td>Host control</td>
<td>Interpreter message (or error message)</td>
</tr>
</tbody>
</table>

2.3.1 No Multiple-operation of DCI, Stand-alone, and Host Control Functions

All DCI, stand-alone, and host control use BSC LIKE protocol and the same port, therefore these functions can not be performed by parallel processing.

Warning message : Serial port not defined
Warning message : Serial port being used
Warning message : Protocol being used

2.3.2 File Access and Editing for a Single Target

Access to a single target file is available. Parallel processing of reads from two or more sources is impossible.

During access to a file for other function, the HLOCK command of the host control function cannot be issued.

Key operations are ignored while the HLOCK command is ON.

Warning message : Data accessed with other functions
2.4 Differences from MRC

The data transmission function in XRC is intended to reuse basically the user applications with succession to the data transmission of MRC. There are some differences resulted from functional differences between the MRC and the XRC.

2.4.1 Multiport Processing

The XRC is not applicable for multiport processing.

2.4.2 Group Axes

The control group information used for the CGROUP and RGROUP commands in the host control function differs depending on the number of manipulators.

2.4.3 Coordinated Operation and Independent Operation

Up to 6 tasks can be changed by the CTASK command. No command related to coordinated operation is available.

2.4.4 Condition Data and System Data

Condition data and system data have different file name in option function, accordingly their communication header differ.
2.5 Transmission Specifications

This section explains the transmission specifications for the data transmission.

2.5.1 Basic Specifications

<table>
<thead>
<tr>
<th>Interface</th>
<th>Complies to RS-232C (RS/CS method)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission Speed</td>
<td>9600 bps</td>
</tr>
<tr>
<td>Transmission Mode</td>
<td>Half-duplex transmission system (point-to-point)</td>
</tr>
<tr>
<td>Synchronization system</td>
<td>Asynchronous (stop bit 1 *1)</td>
</tr>
<tr>
<td>Protocol</td>
<td>BSC LIKE</td>
</tr>
<tr>
<td>Transmission Code</td>
<td>ASCII, shift JIS</td>
</tr>
<tr>
<td></td>
<td>8-bit data length *1</td>
</tr>
<tr>
<td></td>
<td>Even parity *1</td>
</tr>
<tr>
<td></td>
<td>Nontransparent</td>
</tr>
<tr>
<td>Error Check</td>
<td>BCC</td>
</tr>
<tr>
<td>Response Method</td>
<td>ACK alternating response</td>
</tr>
</tbody>
</table>

*1 Can be changed by transmission parameter setting

2.5.2 Transmission Control Characters

The transmission control characters are shown in the table below.

<table>
<thead>
<tr>
<th>Control Character</th>
<th>Code (hexadecimal)</th>
<th>Meanings of Control Character</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLE</td>
<td>10</td>
<td>Data Link Escape</td>
</tr>
<tr>
<td>SOH</td>
<td>01</td>
<td>Start of Heading</td>
</tr>
<tr>
<td>STX</td>
<td>02</td>
<td>Start of Text</td>
</tr>
<tr>
<td>ETX</td>
<td>03</td>
<td>End of Text</td>
</tr>
<tr>
<td>EOT</td>
<td>04</td>
<td>End of Transmission</td>
</tr>
<tr>
<td>ENQ</td>
<td>05</td>
<td>Enquiry</td>
</tr>
<tr>
<td>NAK</td>
<td>15</td>
<td>Negative Acknowledgment</td>
</tr>
<tr>
<td>ESB</td>
<td>17</td>
<td>End of Text Block</td>
</tr>
<tr>
<td>ACK0</td>
<td>10, 30</td>
<td>Even Affirmative Acknowledgment</td>
</tr>
<tr>
<td>ACK1</td>
<td>10, 31</td>
<td>Odd Affirmative Acknowledgment</td>
</tr>
</tbody>
</table>
2.5.3 Transmission Format

The transmission format is as follows.
### 2.5.4 Error Control System

The error control is performed by a check sum of all the characters from SOH or STX to ETB or ETX. The check sum is calculated as shown below.

**<Example>**

- **Start of calculation**: Calculation is started when SOH or STX used as the block start sequence appears. These block start sequence are not included in the sum. As for a STX led by a SOH, STX is included in the sum.

- **End of calculation**: Calculation is ended when ETB or ETX used as the block end sequence appears, with the ETB or ETX included in the sum.

### 2.5.5 Character Configuration

The character configuration is as follows.

### 2.5.6 Data Link Establishment

A data link is established by responding ACK0 to ENQ.
2.5 Transmission Specifications

2.5.7 Configuration of Heading and Text

The configuration of heading and text is as follows.

![Diagram of heading and text configuration]

2.5.8 Transmission Parameters

- **Transmission Control Monitoring Timer**

  Two timers are provided for transmission control monitoring. Both are transmission parameters so that their settings can be changed for each system.

  - **Timer A**: Sequence monitoring timer. Serves as protection against invalid or no response. Recommended value is 3 sec.
  
  - **Timer B**: Text reception monitoring timer. Serves as protection against no response of text end character. Recommended value is 20 sec.
Transmission Control Resending Sequence

Two constants below are related to the transmission control resending sequence. Both are transmission parameters like the transmission control monitoring timers, whose settings can be changed for each system.

- **Retry 1:** Number of resendings of a sequence character at an invalid or no response at all. Recommended value is 10 times.
- **Retry 2:** Number of resendings of a text at a block check error (reception of NAK). Recommended value is 3 times.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents and Set Value</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS030</td>
<td>Number of data bits</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>7 : 7 (bit)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>8 : 8 (bit)</td>
<td></td>
</tr>
<tr>
<td>RS031</td>
<td>Number of stop bits</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0 : 1 (bit)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : 1.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : 2</td>
<td></td>
</tr>
<tr>
<td>RS032</td>
<td>Parity specification</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0 : No specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : Odd parity</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : Even parity</td>
<td></td>
</tr>
<tr>
<td>RS033</td>
<td>Transmission speed specification</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>1 : 150 (baud rate)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : 300</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : 600</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 : 1200</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 : 2400</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6 : 4800</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7 : 9600</td>
<td></td>
</tr>
<tr>
<td>RS034</td>
<td>Timer A</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Sequence monitoring timer</td>
<td>Servs as protection against invalid or no response Unit : 0.1 sec. (Setting range : 0 to 100)</td>
</tr>
<tr>
<td>RS035</td>
<td>Timer B</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Text reception monitoring timer</td>
<td>Servs as protection against no response of text end character Unit : 0.1 sec. (Setting range : 0 to 255)</td>
</tr>
<tr>
<td>RS036</td>
<td>Retry 1</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Number of resendings of a sequence character at an invalid or no response (Setting range : 0 to 30)</td>
<td></td>
</tr>
<tr>
<td>RS037</td>
<td>Retry 2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Number of resendings of a text at a block check error (reception of NAK). (Setting range : 0 to 10)</td>
<td></td>
</tr>
<tr>
<td>RS038</td>
<td>Block check method</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0 : Check sum</td>
<td></td>
</tr>
</tbody>
</table>
2.5.9 Connection of D-SUB Connector Pins

The connection of D-SUB connector pins is shown below.
XCP01 board (D-SUB9P)

2.5.10 Connection

Since the system is “null-modem”, connect the pins as shown below.
XCP01 board

- Connect “RS” of the XRC to “CS” of a host computer. This prevents data overrun when reception processing speed of the XRC cannot catch up with data sending from the host computer. In other words, “RS” signal from the XRC controls start-hold of data transmission from the host computer. The sending interface controller must be capable of coping with CS input displacement in units of a single byte.

- The XRC sends data when the “CS” signal is ON.

- The “ER” signal of the XRC is always ON when transmission is ready. Use this signal on a host computer side whenever necessary.

- The “CD” signal of the XRC is connected to the “ER” signal of a host computer. However, the XRC does not use this signal.
3 DCI Function

3.1 Outline

The data communication by instruction (DCI) function loads, saves jobs and variables according to an instruction that executes data transmission with a host computer. The DCI function is classified as follows.

- Job load, save and delete functions
- Variable load and save functions

3.2 Commands for Job Transmission

3.2.1 LOADJ

- Function
  Loads specified jobs as single or related jobs, from the external memory unit to the memory of the XRC.
3.2 Commands for Job Transmission

**Configuration**

- If the XRC memory already contains a job having the same name as the job to be loaded, the existing job is deleted and the new job is loaded. However, if the job to be loaded is as follows, an alarm occurs.
  - Execution starting job
  - Job under execution/halting
  - Job registered in job call stack

- Specify input group numbers (BCD/BIN, parity specification), and variable numbers in the same way as for the CALL command. If the pattern input value is 0, the operation is not executed. A variable number 0 is valid.

- Unit of loading: Select either a single job (JBI) or related jobs (JBR)

- When the NWAIT is specified, the next instruction is executed without waiting completion of job loading.

- While a job is being loaded by the LOADJ command for which NWAIT is specified, if an access is attempted to a job called by the CALL command or JUMP command, an alarm occurs. If a LOADJ or SAVEJ command has already been executed, a job is loaded after completion of the execution.
3.2 Commands for Job Transmission

3.2.2 SAVEJ

■ Function

Saves a specified job as single or related jobs, from the memory of the XRC to the external memory unit.

■ Configuration

- Specify input group numbers (BCD/BIN, parity specification), and variable numbers in the same way as for the CALL command. If the pattern input value is 0, the operation is not executed. A variable number 0 is valid.

- Unit of saving: Select either a single job (JBI) or related jobs (JBR).

- When the NWAIT is specified, the next command is executed without waiting completion of job saving. When a LOADJ or SAVEJ command has already been executed, a job is saved after completion of the execution.

3.2.3 DELETEJ

■ Function

Deletes all jobs except its own job or specified jobs as single or related jobs, from the memory of the XRC.

■ Configuration

- Unit of deleting: Select either a single job (JBI) or related jobs (JBR).

- The following jobs can not be deleted.
  - Execution starting job
  - Job under execution/halting
  - Job registered in job call stack
**3.3 Commands for Variable Transmission**

### 3.2.4 SWAIT

**Function**
Waits for completion of loading or saving jobs or variables. Use this command to recognize a completion of LOADJ, SAVEJ, LOADV, and SAVEV commands when a NWAIT is specified for these instructions.

**Configuration**

```
SWAIT
```

### 3.3 Commands for Variable Transmission

#### 3.3.1 LOADV

**Function**
Loads the specified global variables from an external memory unit to the XRC memory.

**Configuration**

```
LOADV
```

#### 3.3.2 SAVEV

**Function**
Saves the specified global variables from the XRC memory to an external memory unit.

**Configuration**

```
SAVEV
```
3.4 Registrating DCI Instruction

**Operation**

Move the cursor to the address area ➔ Move the cursor to the line where an instruction is to be registered in the job content display ➔ Press [INFORM LIST] ➔ Select an instruction to be registered ➔ Change the additional items and variable data ➔ Press [INSERT] and [ENTER]

**Explanation**

*1 In the job content display in teach mode, move the cursor to the line just above the place where an instruction is to be registered.

*2 The instruction list dialog is displayed. The cursor moves to the instruction list dialog while the cursor in the address area changes to an underline.

*3 The instruction where the cursor is positioned is displayed with the previously registered additional items in the input buffer line.
3.4 Registrating DCI Instruction

*4 <To register items as displayed in the input buffer>
Perform operation *5.

>To edit any additional items>
With the cursor on the instruction to be registered, press [SELECT]. The cursor moves to the input buffer line.

• Changing a numerical value data of additional items.
Move the cursor to the additional item whose numerical value is to be changed. Pressing simultaneously [SHIFT] and the cursor key increments or decrements the value.

To enter a value by pressing the number keys, press [SELECT] to display the input line.

Enter a value, then press [ENTER]. The value displayed in the input line is changed.

• Adding, changing, or deleting the additional items
To add, change or delete the additional items, select an instruction in the input buffer line to display the detail edit display.

To add an additional item, select “NOT USED” of an additional item selection status, then display the selection dialog to select an additional item to be added.
To delete an additional item, move the cursor to an additional item to be deleted, then select “NOT USED” to delete.
• Changing the data type
  To change the data type of additional item, move the cursor to the □ of the additional
  item and press [SELECT] to select a data type.

After having added, changed or deleted the additional items, press [ENTER]. The
detail edit display is ended and the job content display appears.

*5 The instruction displayed in the input buffer line is registered.
To register an instruction just before an END instruction, it is not necessary to press
[INSERT].

3.5 Concurrent Tasks from Multiple Jobs

As an option, commands related to DCI function can be executed from more than one job
simultaneously. The operations are explained below.

• The DCI related commands can be executed in any job regardless of distinction among
  the ordinary job, concurrent job (option), or job activated in series (option).

• Multiplexing of DCI transmission function is not supported. Therefore, it is impossible to
  manipulate files on two or more external memory units (such as personal computer) con-
  nected to the XRC.

• If two or more commands related to DCI function are issued concurrently, the execution
  starts after completion of processing of the currently executing command. Therefore, if a
  module issues a command request while another module is executing DCI function, the
  request has to wait until the ongoing processing completes.
3.6 DCI Parallel Execution

By using the function described below, the DCI instruction can be executed in parallel with general instructions such as a move instruction and operating instruction. When this function is used, the robot can be moved or the calculation is executed during data transmission; this function is effective for reduction of tact time, etc.

- Parallel Execution Using NWAIT

```plaintext
NOP
MOVJ VJ=50.00
MOVJ VJ=50.00
LOADJ JOB:ABC JBI NWAIT
MOVJ VJ=50.00
MOVJ VJ=50.00
SWAIT
CALL JOB:ABC

END
```

In the above job, when the command ① is executed, loading of the host computer and the job are executed. Normally, when NWAIT is not specified, the commands of ② and after are not executed until the job loading is completed. However, when NWAIT is specified, the commands ② and ③ are executed sequentially during the job loading; at execution of SWAIT command ④, the execution of command ⑤ is waited for the job “ABC” loading is completed. At the time of completion of job “ABC” loading, the command ⑤ is executed to execute the job “ABC”.

At this time, if SWAIT command is not specified before the command ⑤, the command ⑥ is executed during the loading of job “ABC”, and an alarm occurs. Therefore, be sure to verify that loading is completed before executing a job to be loaded, by using SWAIT command. To load/save variables, be sure to input a SWAIT command before using variables to be loaded/saved as shown below.

```plaintext
(Correct)                        (Wrong)
NOP                               NOP
. . .                             . . .
LOADV B000 NWAIT                 LOADV B000 NWAIT
SWAIT                             SET B001 B000
SET B001 B000                    SET B001 B000
```
3.7 Transmission Procedure

Parallel Execution Using PSTART (Optional)

By using an independent control command (optional), DCI commands can be executed in parallel with general commands. For example, to execute the job “R1” for robot 1 is to be executed in parallel with the job “S1” for station 1 during job loading, the following procedure is taken:

Job “R1” : Job for robot 1
Job “S1” : Job for station 1

```
[JOB:R1]                  [JOB:S1]
  NOP                     NOP
  MOVJ VJ=50.00           MOVJ VJ=50.00
  MOVJ VJ=50.00           MOVJ VJ=50.00
  PSTART JOB:S1 SUB1......① END
  LOAD JOB:ABC...........②
  PWAIT...................③
  CALL JOB:ABC...........④
  END
```

When PSTART command ① is executed, the job “S1” starts execution in parallel with the job “R1”. The job “ABC” is loaded by the command ② during execution of the job “S1”; when loading is completed, the XRC waits for the job “S1” to be completed by the command ③. When the execution of job “S1” is completed, the job “ABC” is executed by the command ④.

3.7 Transmission Procedure

3.7.1 Job Transmission

Saving Procedure

The transmission from the XRC to a host computer proceeds as follows.

XRC → Host computer
1. The ENQ code is sent out to establish a data link.
2. After the data link is established, data are sent out to the host computer.
3. After the transmission completes, the XRC waits for a response from the host computer to verify the completion of transmission. Therefore, the host computer should return a response.
4. The transmission is terminated upon receipt of the response from the host computer.
3.7 Transmission Procedure

The data type is distinguished by the header number and the subcode number. Refer to the header number list.

*1 File name: CR (File name does not include extension.)
*2 ACK0 or ACK1
*3 Normal completion: 0000CR (ASCII code)
   Abnormal completion: Integer except 0000 CR (ASCII code)

Loading Procedure

The transmission from a host computer to the XRC proceeds as follows.

Host computer → XRC
1. The ENQ code is sent out to establish a data link.
2. After the data link is established, a request to send is sent out to the host computer.
3. When the request to send is accepted, the XRC enters receiving status, waiting for the ENQ code from the host computer. Therefore, the host computer should send data after the data link is established.
4. The transmission is terminated at completion of data reception from the host computer.

A request to send consists of a header number and a subcode number. Refer to the header number list.
At transmission, memory capacity is checked and if received data can not be stored, an alarm occurs. If the transmission itself is normal, reception is continued and an alarm is displayed after the transmission is terminated. If an error occurs during reception, the job data will not be stored.

*1 File name: CR (File name does not include extension.)

*2 ACK0 or ACK1
3.7 Transmission Procedure

3.7.2 Variable Transmission

The variable transmission is performed in the same way as for the data as shown below.

- Saving Procedure

- Loading Procedure

For headers, refer to the header number list.
### 3.7 Transmission Procedure

#### 1

<table>
<thead>
<tr>
<th>Type</th>
<th>Format</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte type global variable</td>
<td>□□□□</td>
<td>(0 to 255)</td>
</tr>
<tr>
<td>Integer type global variable</td>
<td>±□□□</td>
<td>(-32768 to +32767)</td>
</tr>
<tr>
<td>Double precision type global variable</td>
<td>±□□□□□□□□□□□□□□□</td>
<td>(-2147483648 to 2137383647)</td>
</tr>
<tr>
<td>Real number type global variable</td>
<td>7 significant digits</td>
<td>(-1.70141E+38 to +1.70141E+38)</td>
</tr>
</tbody>
</table>

#### 2

Position type (robot axis) global variable:
- Pulse type
  - S, L, U, R, B, T (Unit: pulse)
    - (-9999999999 to 9999999999)
  - XYZ type
    - X, Y, Z, TX, TY, TZ, TYPE
      - d0 = 0: Flip
      - d0 = 1: No flip
      - d2 = 0: Up
      - d2 = 1: Back
      - Unit: degree (°), significant 2 decimal points
        - -9999.99 to 9999.99
      - Unit: mm, significant 3 decimal points
        - -999999.999 to 999999.999

Position type (base axis) global variable:
- Pulse type
  - 1, 2, 3 (Unit: pulse)
    - (-9999999999 to 9999999999)
  - XYZ type
    - X, Y, Z (Unit: mm, significant 3 decimal points)
      - (-999999.999 to 999999.999)

Position type (station axis) global variable:
- Pulse type
  - 1, 2, 3, 4, 5, 6 (Unit: pulse)
    - (-9999999999 to 9999999999)

#### 2

0000 or error code

The response is as follows when an error occurs in response.

```
SOH 90,000 STX DATA CR ETX BCC
```

If a stop operation (hold and emergency stop) is done during data transmission (while jobs or variables are loaded or saved), the robot stops but the data transmission continues. In this case, the start lamp goes OFF. The restart will not be accepted until completion of the data transmission.
3.8 Axis Data Transmission Format

The XRC data transmission function has the following restrictions on transmission of the XRC internal data.

The robot axes are fixed to a 6-axis set. A base axis and a station axis are recognized as an external axis. Up to three base axes are available. With station axis data added after base axis data, up to six axes can be handled. For example, SAVEV BP005 is read as SAVEV BP005 + EX005.

If the system lacks one of the variables, only the existing one is used. If the system has both variables but not registered, an error occurs.

The definition of the robot, base, and station axes is used as it is, free of the predetermined axis data R1, B1, and S1.

<Example>
Transmission data of SAVEV in different system configurations are shown below.

- In a system having two base axes (X and Z) and no station axis
  If BP005 is pulse type and 1st axis is 100 and 2nd axis is 200, then
  SAVEV BP005 → 03, 007 100, 200, 0, 0, 0, 0
  If BP005 is XYZ type and X-axis is 123.456 and Z-axis is 234.567, then
  SAVEV BP005 → 03, 008 123.456, 234.567, 0, 0, 0, 0

- In a system having no base axis and three station axes
  If EX005 is pulse type and 1st axis is 500, 2nd axis is 600, and 3rd axis is 700
  SAVES EX005 → 03, 007 500, 600, 700, 0, 0, 0

- In a system having two base axes (X and Z) and three station axes
  If BP005 is pulse type, 1st axis is 100 and 2nd axis is 200, and
  EX005 is pulse type, 1st axis is 500, 2nd axis is 600, and 3rd axis is 700, then
  SAVEV BP005 → 03, 007 100, 200, 500, 600, 700, 0
  (Same as for SAVEV EX005)
  If BP005 is XYZ type, X axis is 123.456, and Z axis is 234.567, and
  EX005 is pulse type, 1st axis is 500, 2nd axis is 600, and 3rd axis is 700, then
  SAVEV BP005 → 03, 008 123.456, 234.567, 500, 600, 700, 0
  (Same as for SAVEV EX005)
### 3.9 Alarm Codes

#### Code Message Data

<table>
<thead>
<tr>
<th>Code</th>
<th>Message Description</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>4104</td>
<td>WRONG EXECUTION OF LOAD INST</td>
<td>Refer to the table below.</td>
</tr>
<tr>
<td>4105</td>
<td>WRONG EXECUTION OF SAVE INST</td>
<td></td>
</tr>
<tr>
<td>4106</td>
<td>WRONG EXECUTION OF DELETE INST</td>
<td></td>
</tr>
</tbody>
</table>

#### Data Contents

<table>
<thead>
<tr>
<th>Data</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>Insufficient memory capacity</td>
</tr>
<tr>
<td>002</td>
<td>Job editing prohibited</td>
</tr>
<tr>
<td>003</td>
<td>Attempted to load or delete a job being executed.</td>
</tr>
<tr>
<td>004</td>
<td>No specified job</td>
</tr>
<tr>
<td>012</td>
<td>Position data destroyed</td>
</tr>
<tr>
<td>013</td>
<td>Position variable not registered</td>
</tr>
<tr>
<td>017</td>
<td>Instruction destroyed</td>
</tr>
<tr>
<td>019</td>
<td>Invalid character in job name</td>
</tr>
<tr>
<td>020</td>
<td>Invalid character in label</td>
</tr>
<tr>
<td>023</td>
<td>Invalid character in this system</td>
</tr>
<tr>
<td>024</td>
<td>Syntax error</td>
</tr>
<tr>
<td>104</td>
<td>Error response from host computer</td>
</tr>
<tr>
<td>111</td>
<td>Syntax error</td>
</tr>
<tr>
<td>112</td>
<td>Error in position data</td>
</tr>
<tr>
<td>113</td>
<td>No NOP or END instruction</td>
</tr>
<tr>
<td>117</td>
<td>Format error</td>
</tr>
<tr>
<td>118</td>
<td>Invalid number of data</td>
</tr>
</tbody>
</table>
### 3.9 Alarm Codes

<table>
<thead>
<tr>
<th>Data</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>120</td>
<td>Data range exceeded</td>
</tr>
<tr>
<td>122</td>
<td>Destroyed file exists</td>
</tr>
<tr>
<td>125</td>
<td>No serial port setting</td>
</tr>
<tr>
<td>126</td>
<td>This serial port already used.</td>
</tr>
<tr>
<td>127</td>
<td>This protocol already used.</td>
</tr>
<tr>
<td>128</td>
<td>File accessing in other function</td>
</tr>
<tr>
<td>211</td>
<td>System block error (Receiving EOT while waiting ACK)</td>
</tr>
<tr>
<td>212</td>
<td>System block error (Receiving EOT at starting receiving)</td>
</tr>
<tr>
<td>213</td>
<td>System block error (Receiving EOT before receiving the last block)</td>
</tr>
<tr>
<td>214</td>
<td>System block error (Receiving codes other than EOT before receiving the last block)</td>
</tr>
<tr>
<td>221</td>
<td>Sending error (Retry for NAK exceeded)</td>
</tr>
<tr>
<td>222</td>
<td>Sending error (Timeup for timer A retry)</td>
</tr>
<tr>
<td>223</td>
<td>Sending error (ACK0/ACK1 order error after retry)</td>
</tr>
<tr>
<td>231</td>
<td>Receiving error (Timeup for timer A while waiting ACK after ENQ, timeup for timer A while waiting ENQ response)</td>
</tr>
<tr>
<td>232</td>
<td>Receiving error (Timeup for timer B while receiving a text)</td>
</tr>
<tr>
<td>233</td>
<td>Receiving error (Heading length is shorter than 6 characters)</td>
</tr>
<tr>
<td>234</td>
<td>Receiving error (Heading length is longer than 6 characters)</td>
</tr>
<tr>
<td>235</td>
<td>Receiving error (Header number error)</td>
</tr>
<tr>
<td>236</td>
<td>Receiving error (Text length exceeds 256 bytes)</td>
</tr>
<tr>
<td>240</td>
<td>Software error</td>
</tr>
<tr>
<td>241</td>
<td>Hardware error (Overrun)</td>
</tr>
<tr>
<td>242</td>
<td>Hardware error (Parity error)</td>
</tr>
<tr>
<td>243</td>
<td>Hardware error (Framing error)</td>
</tr>
<tr>
<td>244</td>
<td>Hardware error (Sending timeup (timer A))</td>
</tr>
<tr>
<td>245</td>
<td>Hardware error (Sending timeup (timer B))</td>
</tr>
</tbody>
</table>
4 Stand-alone Function

4.1 Outline

In stand-alone mode, the file data transmission function is available. By the operations on the XRC programming pendant, file data can be sent from the XRC to a host computer such as personal computer to be saved, and from a host computer to the XRC memory to be loaded.

Load: Transmits file data from a host computer to the XRC.
Save: Transmits file data from the XRC to a host computer.
Verify: Verifies data between the XRC and the host computer and informs if some parts are not matched.

The following data can be transmitted between the XRC and a host computer. System information can be saved but not loaded.

- Job data
- Condition data/General data
- System information
Transmission of file data is performed in the following manner.

1. **Top menu**
   - [EXT MEM]

2. **Sub menu**
   - [LOAD]
   - [SAVE]
   - [VERIFY]
   - [DEVICE]

3. **Select a device**

4. **Select a data group**

5. **Select a data**

6. **Press "EXECUTE"**

7. **End**
4.3 Operation

4.3.1 Selecting External Memory Unit

**Operation**

Select {FD/PC CARD} under the top menu ➔ Select {DEVICE} ➔ Select “DEVICE” ➔ Select the device to be changed

**Explanation**

*1 The device selection display is shown.

*2 The selection dialog is shown.

*3 The device is changed.
4.3 Operation

4.3.2 Save

The operation to transmit data from the XRC to the external memory unit is explained in the following.

### Saving Job

**Operation**

Select {FD/ PC CARD} under the top menu ➔ Select {SAVE} ➔ Select “JOB” ➔
Select the job to be saved ➔ Press [ENTER] ➔ Select “YES”

**Explanation**

*1 The external memory menu display is shown.

<table>
<thead>
<tr>
<th>DATA</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOPPY DISK/PC CARD</td>
<td>R1</td>
<td>3:</td>
<td></td>
</tr>
<tr>
<td>PC(SAVE)</td>
<td>UN-UNUSED MEM</td>
<td>123.4 KB</td>
<td></td>
</tr>
<tr>
<td>JOB</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILE/GENERAL DATA</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALL CMOS AREA</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*2 The external memory job list display is shown.

<table>
<thead>
<tr>
<th>DATA</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOPPY DISK/PC CARD</td>
<td>R1</td>
<td>3:</td>
<td></td>
</tr>
<tr>
<td>PC(SAVE)</td>
<td>SINGLE</td>
<td>NO.:7</td>
<td></td>
</tr>
<tr>
<td>*TEST0001</td>
<td>TEST0002</td>
<td>TEST0003</td>
<td></td>
</tr>
<tr>
<td>TEST0004</td>
<td>TEST0005</td>
<td>TEST0006</td>
<td></td>
</tr>
<tr>
<td>TEST0007</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*3 The selected job is marked with “★”.

<table>
<thead>
<tr>
<th>DATA</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOPPY DISK/PC CARD</td>
<td>R1</td>
<td>3:</td>
<td></td>
</tr>
<tr>
<td>PC(SAVE)</td>
<td>SINGLE</td>
<td>NO.:7</td>
<td></td>
</tr>
<tr>
<td>★TEST0001</td>
<td>TEST0002</td>
<td>TEST0003</td>
<td></td>
</tr>
<tr>
<td>TEST0004</td>
<td>TEST0005</td>
<td>TEST0006</td>
<td></td>
</tr>
<tr>
<td>TEST0007</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1
4.3 Operation

*4 The confirmation dialog is shown.

*5 The job starts to be saved, and the transmission display is shown.

To interrupt the saving, press [SELECT].
When the saving is completed or interrupted, the job content display appears.

Saving File

**Operation**

Select {FD/PC CARD} under the top menu ➔ Select {SAVE} *4 ➔ Select the file group to be saved *2 ➔ Select the file to be saved *3 ➔ Press [ENTER] *4 ➔ Select “YES” *5

**Explanation**

*1 The external memory menu display is shown.
4.3 Operation

*2 The file selection display is shown.

*3 The selected file is marked with “★”.

*4 The confirmation dialog is shown.

*5 The file starts to be saved, and the transmission display is shown.

To interrupt the saving, press [SELECT].
When the saving is completed or interrupted, the file selection display reappears.
4.3.3 Load

The operation to transmit data from the external memory unit to the XRCi is explained in the following.

### Loading Job

**Operation**

Select {FD/ PC CARD} under the top menu ➤ Select {LOAD} ➤ Select “JOB” ➤ Enter the job to be loaded ➤ Select “EXEC”

**Explanation**

*1 The external memory menu display is shown.

```
<table>
<thead>
<tr>
<th>DATA</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOPPY DISK/PC CARD</td>
<td>R1</td>
<td>C</td>
<td>S</td>
</tr>
<tr>
<td>PC(LOAD)</td>
<td>UN-USED MEM :123.4 KB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOB</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILE/GENERAL DATA</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALL CMOS AREA</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

*2 The display to input the job name to be loaded is shown.

```
<table>
<thead>
<tr>
<th>DATA</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOPPY DISK/PC CARD</td>
<td>R1</td>
<td>C</td>
<td>S</td>
</tr>
<tr>
<td>PC(LOAD)</td>
<td>JOB MODE :SINGLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOB NAME</td>
<td>EXEC</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

4-7
### Loading File

**Operation**

Select (FD/ PC CARD) under the top menu ➔ Select (LOAD)*1 ➔ Select the file group to be loaded*2 ➔ Select the file to be loaded*3 ➔ Press [ENTER]*4 ➔ Select “YES”*5

**Explanation**

*1 The external memory menu display is shown.

![External Memory Menu Display](image)

*2 The file selection display is shown.

![File Selection Display](image)

*3 The selected file is marked with “*”.

![Selected File Display](image)
*4 The confirmation dialog is shown.

The file starts to be loaded, and the transmission display is shown.

To interrupt the loading, press [SELECT].
When the loading is completed or interrupted, the file selection display reappears.

4.3.4 Job Selection Mode

To select a job to save, load, or verify, the following selection modes are available.

- Single Selection Mode
  Only the selected job is loaded, saved, or verified.

- Related Selection Mode
  The selected job and the related jobs and data files are loaded, saved, or verified.
4.3 Operation

Switching Selection Mode

Operation

Press the page key in the external memory job list display.*1

Explanation

*1 Each time the page key \( \rightarrow \) is pressed, the displays in “single selection mode” and in “related selection mode” appears alternately.

For single selection mode

<table>
<thead>
<tr>
<th>JOB</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>J:TEST</td>
<td>S:000</td>
<td>R1</td>
<td>TOOL:*</td>
</tr>
<tr>
<td>0000 NOP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0001 MOVJ VJ=50.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0002 MOVL V=276</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0003 WVON WEV#(1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0004 ARCON ASF#(1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0005 MOVL V=138</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0006 MOVL V=138</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>=&gt;MOVJ VJ=100.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Only the selected job is loaded, saved, and collated.

For related selection mode

<table>
<thead>
<tr>
<th>JOB</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>J:JOB-1</td>
<td>S:000</td>
<td>R1</td>
<td>TOOL:*</td>
</tr>
<tr>
<td>0000 CALL JOB:JOB-11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0006 CALL JOB:JOB-1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0007 MOVJ VJ=50.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0008 MOVJ VJ=50.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0009 MOVJ VJ=50.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0010 MOVL V=276</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0011 MOVL V=276</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0012 WVON WEV#(1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0013 TIMER T=0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The selected job and the data file and related job are loaded, saved and collated.

* For single selection mode

* For related selection mode

![Diagram showing job content and job edit display utility]
4.3.5 Selecting Job and Data File

There are two ways to select a job and various data files to be loaded, saved, verified, or deleted.

- EACH Selection
  Selects job and data file one by one.

- BATCH Selection
  Selects all the jobs and data files at once.

For BATCH selection, proceeds the following operation.

**Operation**

Select (EDIT) of the menu in the external memory job list display or the file selection display*1 ➔ Select (SELECT ALL)

**Explanation**

*1 The pulldown menu is displayed.

---

4.4 Transmission Procedure

The transmission procedure is the same as for DCI function. Refer to Section 3.7 “Transmission Procedure”.

---

[Image of external memory job list display]
4.4 Transmission Procedure
5 Host Control Function

The XRC supports the host control function which carries out the following file data transfer or robot control according to the commands given by the host computer.

- File data transfer function
- Robot control function

To use the host control function, the following settings should be made.

- The “COMMAND REMOTE” described in Chapter 8 “Remote Function Setting”, should be set valid (marked with “•”).
- The parameter RS000 should be set to “2”.
- The host control function should be validated. Whether the host control function is validated, can be verified in the “remote display” described in Section 2.1.3 “Display in Command Remote Mode”.

5.1 File Data Transmission Function

According to commands from a host computer, the host control function sends the stored data of user memory of the XRC to the host computer or receives data from the host computer. The following data can be transmitted between the XRC and a host computer. The system information can be sent only to a host computer.

- Job data
- Condition file/General data
- System information

5.1.1 Transmission Procedure

Load

The transmission from a host computer to the XRC proceeds as follows.

Host computer → XRC
1. The ENQ code is sent from the host computer to establish a data link.
2. After the data link is established, the data is sent from the host computer.
3. After the transmission is completed, the host computer should get ready to receive.
4. After the data link is established, a response to the data sent from the host computer is returned from the XRC to terminate the transmission.
5.1 File Data Transmission Function

The data type is distinguished by the header number and the subcode number. Refer to the header number list.

Loading File Data (Host Control Function)

*1 ACK0 or ACK1
*2 Normal completion: 0000CR (ASCII code)
   Abnormal completion: "Integer except 0000"CR (ASCII code)
*3 File name: CR (File name does not include extension)

Save

The transmission from the XRC to a host computer proceeds as follows.

XRC → Host computer
1. The ENQ is sent from the host computer to establish a data link.
2. After the data link is established, a request to send is sent from the host computer.
3. The request to send consists of a header number and a subcode number. Refer to the header number list.
4. After the request to send is accepted, the host computer should get ready to receive data. The XRC sends the ENQ code to establish a data link.
5. After the data link is established, receive the data sent from the XRC. The transmission terminates at completion of reception. If the data requested to send are not found, or the header of the request to send has an error, the XRC sends the following response message instead of data. Check the header and take an appropriate action. 

SOH 90,000 STX DATA CR ETX BCC
5.1 File Data Transmission Function

5.1.2 Data Management

The jobs for the XRC may refer to another job or condition data according to instructions. When saving a single job or condition data to the host computer, the correspondence between job and files should be controlled.

To reduce this labor, the related jobs and condition data can be transmitted in a batch as the related job data. When specification of “related job data” is made, the master job, the related job, and the related condition data are transmitted sequentially. The header number and the subcode number indicate that the related job data are added.

Refer to the header number list.

*1 ACK0 or ACK1
*2 File name: CR (File name does not include extension)
5.2 Robot Control Function

To control manipulators by a host computer, the host control function can executes the commands listed in the outline.

5.2.1 Command Transmission

The command transmission proceeds as follows.

1. The ENQ code is sent from the host computer to establish a data link.
2. After the data link is established, commands are sent. Commands and file data are distinguished by the header number. Refer to the header number list. The transmission of a command should be completed in a single block. The XRC can not receive divided single command, nor receive to execute more than one command in a single block.
3. After the sending is completed, the host computer should get ready to receive. The XRC sends the ENQ code to establish a data link.
4. After the data link is established, the XRC sends the response for the command and terminates the transmission. The command format and the response format are explained in the following.

For the command that requires returning data as a response, the response format at normal completion of transmission is as shown in (2).

**Command Format**

```
SOH 01,000 STX COMMAND Data1, Data2, Data3 CR ETX BCC
```

**Response Format**

1. `SOH 90,000 STX (0000 or Error code) CR ETX BCC`
   - 0000: Normal completion
   - Error code: Number with 4 digits other than 0000. In case of smaller than 1000, 0 is added before the number.

2. `SOH 90,001 STX Data1, Data2, … DataN CR ETX BCC`

If the XRC can not execute the sent command, the XRC returns an interpreter message. An example of DELETE command (delete a job) is shown.
5.2 Robot Control Function

5.2.2 List of Interlock for Commands of Host Control Function

The executability of each command differs depending on the status of the XRC as shown in the following table.

<table>
<thead>
<tr>
<th>Command Name</th>
<th>Read/Write Enabled</th>
<th>Only Read Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Non-alarm/Non-error</td>
<td>Non-alarm/Non-error</td>
</tr>
<tr>
<td></td>
<td>Teach Mode</td>
<td>Alarm/Error</td>
</tr>
<tr>
<td></td>
<td>Stop</td>
<td>Operating</td>
</tr>
<tr>
<td>Read or Monitor</td>
<td>RALARM</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>RPOSC</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>RPOSJ</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>RSTATS</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>RJSEQ</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>JWAIT</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>RGROUP</td>
<td>○</td>
</tr>
<tr>
<td>Read or Data Access</td>
<td>RJDIR</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>RUFRAME</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>UPLOAD</td>
<td>○</td>
</tr>
<tr>
<td></td>
<td>SAVEV</td>
<td>○</td>
</tr>
</tbody>
</table>
### 5.2 Robot Control Function

<table>
<thead>
<tr>
<th>Command Name</th>
<th>Read/Write Enabled</th>
<th>Only Read Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Non-alarm/Non-error</td>
<td>Non-alarm/Non-error</td>
</tr>
<tr>
<td></td>
<td>Teach Mode</td>
<td>Play Mode</td>
</tr>
<tr>
<td></td>
<td>Stop</td>
<td>Operating</td>
</tr>
<tr>
<td><strong>Operation</strong></td>
<td>HOLD</td>
<td>RESET</td>
</tr>
<tr>
<td></td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td><strong>Activation</strong></td>
<td>START</td>
<td>MOVJ</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td><strong>Editing</strong></td>
<td>DELETE</td>
<td>CVTRJ</td>
</tr>
<tr>
<td></td>
<td>○</td>
<td>MOVE</td>
</tr>
<tr>
<td><strong>Job selection</strong></td>
<td>SETMJ</td>
<td>JSEQ</td>
</tr>
<tr>
<td></td>
<td>○</td>
<td>MOVE</td>
</tr>
</tbody>
</table>
5.2 Robot Control Function

5.2.3 Command that Handle Axis Data

The data transmission function of the XRC has restrictions on handling control axis data. Since the manipulator axes are fixed to a six-axis set, any manipulator having more than seven axes cannot use the following commands.

A base axis and a station axis are recognized as an external axis. Up to three base axes can be used. With station axis data added to the base axis data, up to six axes can be handled.

This applies to the following commands.
RPOSJ, RPOSC, RUFRAME
MOVJ, MOVL, IMOV, PMOVJ, PMOVL, WUFRAME

5.2.4 Response to MOV-type Command

The responses to MOV-type command are as follows.

- If the manipulator is moving by operations other than commands, the interpreter message 2010 (manipulator moving) is returned and the manipulator does not move.
- If the manipulator is in stop status, it turns ON the start lamp and moves according to the command, and returns a response immediately.
- If the manipulator is moving according to the previous commands, only a single command is accepted and the response is held up. After completing execution of the preceding commands, when starting execution of the suspended command, the manipulator returns a response.

This applies to the following commands.
MOVJ, MOVL, IMOV, PMOVJ, PMOVL

<Interpreter message>

Ø : Possible to execute
A : Alarm/error occurring 2060
M : Incorrect mode 2080
H : Hold 2020 to 2050
MOVE : Manipulator moving 2010
C : No command remote setting 2100

*1 “Ø” if not being held ; “H” if being held
*2 “MOVE” if the manipulator is moving by operation other than command ; “Ø” if the manipulator is moving by command since a single command can be accepted.
*3 “Ø” during an alarm ; “A” during error
5.2.5 Status Read Function

The details of each command are described.

Read/Monitor Command

RALARM

Reads the error alarm code. Although the XRC has the subcode to error code, it can not read by RALARM because the command has no argument of the subcode.

Command format: RALARM

Response format: Data-1, Data-2, ..., Data-9 or Error code

Data-1 = Error code (0 to 9999)
Data-2 = Alarm code (0 to 9999)
Data-3 = Alarm data (0 to 256)
Data-4 = Alarm code (0 to 9999)
Data-5 = Alarm data (0 to 256)
Data-6 = Alarm code (0 to 9999)
Data-7 = Alarm data (0 to 256)
Data-8 = Alarm code (0 to 9999)
Data-9 = Alarm data (0 to 256)

Example:
Command: RALARM
Response: 0, 1234, 12, 0, 0, 0, 0, 0, 0

ROPOSJ

Reads the current position in joint coordinate system.

Command format: RPOSJ

Response format: Data-1, Data-2, ..., Data-12 or Error code

Data-1 = Number of S-axis pulses
Data-2 = Number of L-axis pulses
Data-3 = Number of U-axis pulses
Data-4 = Number of R-axis pulses
Data-5 = Number of B-axis pulses
Data-6 = Number of T-axis pulses
Data-7 = Number of 7th axis pulses
Data-8 = Number of 8th axis pulses
Data-9 = Number of 9th axis pulses
Data-10 = Number of 10th axis pulses
Data-11 = Number of 11th axis pulses
Data-12 = Number of 12th axis pulses
5.2 Robot Control Function

<Example>
Command    RPOSJ
Response     500, 2600, 1250, 10789, 624, 36, 0, 0, 0, 0, 0, 0

**RPOSC**
Reads the current position in a specified coordinate system. Whether there is an external axis or not can be specified.

Command format: RPOSC Data-1, Data-2

Data-1 = Specification of coordinate system
0 : Base coordinate
1 : Robot coordinate
2 : User coordinate 1
25 : User coordinate 24

Data-2 = With or Without external axis
0 : Without external axis
1 : With external axis

Response format: Data-1, Data-2, ..., Data-14

Data-1 = X coordinate value (unit: mm, significant 3 decimal points)
Data-2 = Y coordinate value (unit: mm, significant 3 decimal points)
Data-3 = Z coordinate value (unit: mm, significant 3 decimal points)
Data-4 = Wrist angle TX (unit: degree (°), significant 2 decimal points)
Data-5 = Wrist angle TY (unit: degree (°), significant 2 decimal points)
Data-6 = Wrist angle TZ (unit: degree (°), significant 2 decimal points)
Data-7 = Type
Data-8 = Tool number (0 to 23)
Data-9 = Number of 7th axis pulses (for travel axis, mm)
Data-10 = Number of 8th axis pulses (for travel axis, mm)
Data-11 = Number of 9th axis pulses (for travel axis, mm)
Data-12 = Number of 10th axis pulses
Data-13 = Number of 11th axis pulses
Data-14 = Number of 12th axis pulses

- Data-9 to Data-14 are added only when “With external axis” is specified.
- If the specified user coordinate system is undefined, an error occurs.
- The data of type is represented by the following bit data coded into a decimal number.
5.2 Robot Control Function

<Example>
Command   RPOSC 2, 0
Response   100.0, 50, 34, 12.34, 180.0, 0, 0, 0, 0, 0

RSTATS
Reads the status of mode, cycle, operation, alarm error, and servo.

Command format : RSTATS

Response format : Data-1, Data-2 or Error code

<Example>
Command  RSTATS
Response   1, 0
RJSEQ
Reads the current job name, line No. and step No.

Command format : RJSEQ

Response format : Data-1, Data-2, Data-3 or Error code

Data-1 = Read job name (8 characters which can be processed in MS-DOS)
Data-2 = Read line No. (0 to 9999)
Data-3 = Read step No. (0 to 999)

<Example>
Command  RJSEQ
Response   WORK-A, 10, 5

JWAIT
JWAIT is for checking operations (job) of the manipulator. If a response is returned immediately after the job is started, in such a case with START command, completion of the job cannot be known.
Specify a waiting time as an operand for JWAIT command.
No response is sent out until the operation of manipulator is completed or the specified waiting time has elapsed. JWAIT returns as a response, the information whether the operation has completed or not.

Command format : JWAIT Time

Time = Waiting time (-1.0 to 32767 sec.)
-1.0 indicates infinite time.

Response format : Data or Error code

Data = Operation status (0 : completed, -1 : not completed)

Waits for stop of job execution. The response varies depending on the following status.

<table>
<thead>
<tr>
<th>&lt;Status&gt;</th>
<th>&lt;Response&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>END or PAUSE execution during waiting time</td>
<td>Completed</td>
</tr>
<tr>
<td>Step execution during waiting time</td>
<td>Completed</td>
</tr>
<tr>
<td>Stopped by hold, alarm, emergency stop, servo OFF during waiting time</td>
<td>Not completed</td>
</tr>
<tr>
<td>Stopped by changing mode during waiting time</td>
<td>Not completed</td>
</tr>
<tr>
<td>Test run is interrupted during waiting time</td>
<td>Not completed</td>
</tr>
</tbody>
</table>
5.2 Robot Control Function

<Example>
Command JWAIT 10
Response 0000

RGROUP

Reads the current control group set by CGROUP command or CTASK command, and the task selection status.
For details, refer to Section 5.3 “Commands for Multi-control Group and Independent Control Functions”.

Command format : RGROUP

Response format : Data-1, Data-2 or Error code

Data-1 = Control group information. The control group information differs depending on the number of manipulators in the system.

Data-2 : Task information
0 : Master task
1 : Sub 1 task
2 : Sub 2 task
3 : Sub 3 task
4 : Sub 4 task
5 : Sub 5 task

In a system where independent control is not allowed, “0” is returned.
5.2 Robot Control Function

<Example>
Command  RGROUP
Response  3, 0

In a system with one manipulator, the above example shows that the current control group is robot 1 and station 1, and the task selection status is master task.

Read/Data Access System Commands

RJDIR
Reads all job names, or the names of jobs related to the parent job.

Command format : RJDIR Job-Name

Job-Name = All the job names currently registered
          = Parent job name (8 characters which can be processed in MS-DOS)

If a parent job name is specified, RJDIR reads the name of related jobs excluding the parent job. If there is no related child job, the command returns the null list. If the parent job has related child jobs but they are not registered in the system, an error occurs.

Response format : Name-1, Name-2, ..., Name-N or Error code

Name-1 = Job name-1 (8 characters)
Name-2 = Job name-2 (8 characters)
Name-N = Job name-N (8 characters)

<Example>
Command  RJDIR MASTER-1
Response  WORK-A, WORK-B, SAMPLE-1

RUFRAME
Reads a specified user coordinate data.

Command format : RUFRAME Data-1

Data-1 = User coordinate No.
        0 : Reserved
        1 : Reserved
        2 : User coordinate 1
        25 : User coordinate 24

Response format : Data-1, Data-2, ..., Data-28

Data-1 = ORG X coordinate value (unit : mm, significant 3 decimal points)
5.2 Robot Control Function

Data-2  =  ORG Y coordinate value (unit : mm, significant 3 decimal points)
Data-3  =  ORG Z coordinate value (unit : mm, significant 3 decimal points)
Data-4  =  ORG wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-5  =  ORG wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-6  =  ORG wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-7  =  ORG type
Data-8  =  XX X coordinate value (unit : mm, significant 3 decimal points)
Data-9  =  XX Y coordinate value (unit : mm, significant 3 decimal points)
Data-10 =  XX Z coordinate value (unit : mm, significant 3 decimal points)
Data-11 =  XX wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-12 =  XX wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-13 =  XX wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-14 =  XX type
Data-15 =  XY X coordinate value (unit : mm, significant 3 decimal points)
Data-16 =  XY Y coordinate value (unit : mm, significant 3 decimal points)
Data-17 =  XY Z coordinate value (unit : mm, significant 3 decimal points)
Data-18 =  XY wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-19 =  XY wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-20 =  XY wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-21 =  XY type
Data-22 =  Tool No. (0 to 23)
Data-23 =  Number of 7th axis pulses (for travel axis, mm)
Data-24 =  Number of 8th axis pulses (for travel axis, mm)
Data-25 =  Number of 9th axis pulses (for travel axis, mm)
Data-26 =  Number of 10th axis pulses
Data-27 =  Number of 11th axis pulses
Data-28 =  Number of 12th axis pulses

- ORG, XX, XY coordinates are read in the base coordinate system.
- In a system having no external axis, Data-23 to Data-28 are “0”.
- If the specified user coordinate system is not registered, an error occurs.
- If the group axis of the specified user coordinate system is not R1, an error occurs.
- If ORG, XX, and XY have different base axis data, an error occurs.

<Example>
Command  RUFRAME 2
Response  600.0, 12.34, 500.0, 180.0, 0.0, 0.0, 0, ..., 0

SAVEV
Sends variable data to a host computer.

Command format : SAVEV Data-1, Data-2

Data-1  :  Type of variables
          0 : Byte type variables
          1 : Integer type variables
          2 : Double precision type variables
3: Real number type variables
4: Robot axis position type variables
5: Base axis position type variables
6: Station axis position type variables (only pulse type)

Data-2 : Variable No.

Response format: Data-1, Data-2, Data-3, ..., Data-10 or Error code

Data-1 = Byte value / Integer value / Double precision integer value / Real number value / Position data type
Position data type = 0: Pulse type
1: Cartesian type

(When the position data type is “0”)
Data-2 = Number of S-axis pulses / Number of base 1st axis pulses / Number of station 1st axis pulses
Data-3 = Number of L-axis pulses / Number of base 2nd axis pulses / Number of station 2nd axis pulses
Data-4 = Number of U-axis pulses / Number of base 3rd axis pulses / Number of station 3rd axis pulses
Data-5 = Number of R-axis pulses / Number of base 4th axis pulses / Number of station 4th axis pulses
Data-6 = Number of B-axis pulses / Number of base 5th axis pulses / Number of station 5th axis pulses
Data-7 = Number of T-axis pulses / Number of base 6th axis pulses / Number of station 6th axis pulses
Data-8 = Tool No.

(When the position data type is “1”)
Data-2 = Coordinate data
0: Base coordinate
1: Robot coordinate
2: User coordinate 1
3: User coordinate 2
... 25: User coordinate 24
26: Tool coordinate
27: Master tool coordinate

Data-3 = X coordinate value / Base 1st Cartesian value (unit: mm, significant 3 decimal points)
Data-4 = Y coordinate value / Base 2nd Cartesian value (unit: mm, significant 3 decimal points)
Data-5 = Z coordinate value / Base 3rd Cartesian value (unit: mm, significant 3 decimal points)
Data-6 = Wrist angle RX coordinate value (unit: degree (°), significant 2 decimal points)
5.2 Robot Control Function

Data-7 = Wrist angle RY coordinate value (unit : degree (°), significant 2 decimal points)
Data-8 = Wrist angle RZ coordinate value (unit : degree (°), significant 2 decimal points)
Data-9 = Form

Data of the form is a value obtained by converting the following bit data to decimal notation.

Data-10 = Tool No.

<Example>
Command  SAVEV 0, 0
Response   123

In the above example, 123, the value of byte type variable B000, is sent to the host computer.

5.2.6 System Control Function

■ Operation System Commands

**HOLD**

Turns HOLD ON/OFF

Command format : HOLD [Data]

Data = Specification of HOLD ON/OFF status
(0 : OFF, 1 : ON)

Response format : 0000 or Error code

<Example>
Command  HOLD 1
Response   0000

**RESET**

Resets an alarm of manipulator.
The transmission alarms can be reset only by the programming pendant.

Command format : RESET
5.2 Robot Control Function

Response format : 0000 or Error code

<Example>
Command  RESET
Response  0000

CANCEL
Cancels an error.

Command format : CANCEL

Response format : 0000 or Error code

<Example>
Command  CANCEL
Response  0000

MODE
Selects a mode.

Command format : MODE Mode-No

Mode-No.=1 or 2
1 : Teach mode
2 : Play mode

Response format : 0000 or Error code

<Example>
Command  MODE 2
Response  0000

CYCLE
Selects cycle.

Command format : CYCLE Cycle-No

Cycle-No  = Cycle specification (1 to 3)
1 : Step
2 : 1 cycle
3 : Auto

Response format : 0000 or Error code

<Example>
Command  CYCLE 2
Response  0000
SVON
Turns servo power supply ON/OFF.
To turn the servo ON/OFF by this command, connect the external servo ON (EXSVON) signal CN05-9 of the XIO board, to the CN05-10.

Command format : SVON Data
Data = Specification of servo power supply ON/OFF status
      (0 : OFF, 1 : ON)
Response format : 0000 or Error code

<Example>
Command  SVON 1
Response   0000

HLOCK
Sets an interlock between the programming pendant or playback panel and I/O operation signals. While the interlock is ON, all operations except the followings are prohibited.
• Emergency stop from the programming pendant
• Hold and emergency stop from the playback panel
• Input signals except I/O mode change, external start, external servo ON, cycle change, I/O prohibited, P.P/P.PANEL prohibited, and master call

HLOCK is invalid while the programming pendant is in edit mode or accessing to a file for other function.

Command format : HLOCK Data
Data = Interlock status setting (0 : OFF, 1 : ON)
Response format : 0000 or Error code

<Example>
Command  HLOCK 1
Response   0000

MDSP
Receives message data and displays the message in the remote display of the programming pendant. If the currently shown display is not the remote display, it is changed forcibly to the remote display to display the MDSP command message.

Command format : MDSP Data
Data = Message to be displayed (Max. 30 characters)
Response format : 0000 or Error code
<Example>
Command MDSP auto running
Response 0000

CGROUP
Changes an objective control group of various commands used in the host control function. The XRC can support multiple number of manipulators and stations. In this case, CGROUP is used when any control group for commands such as RPOSC is to be changed. For details, refer to Section 5.3 “Commands for Multiple-control Group and Independent Control Functions”.

When the power supply is started up, robot 1, base 1, and station 1 (when a base and a stations exist) are specified.

Command format : CGROUP Data-1

Data-1 = Control group specification. A control group can be specified according to the following data. The specifications of control group differs depending on the number of manipulators in a system. The following settings can not be made.

- Selection of control axis which does not exist
- Simultaneous specification of R1 and R2
- Specification of multiple number of stations

In a system with a base axis (such as travel axis), when the manipulator with this base axis is specified, this base axis is automatically specified.

Response format : 0000 or Error code
5.2 Robot Control Function

<Example>
Command  CGROUP 6
Response  0000

In the above example with two manipulators, robot 2 and station 1 are validated. By issuing RPOS after this command is issued, the current positions of robot 2 and station 1 can be read.

CTASK (Optional)

Changes the task for control in the host control function.

When the power supply is started up or in a system where an independent control is not allowed, this command is to be used as follows.

For details, refer to Section 5.3 “Commands for Multi-control Group and Independent Control Functions”.

- When the power supply is started up, a master task is selected as the task to be controlled.
- CTASK cannot be used in a system where an independent control is not allowed.

Command format : CTASK Data-1

Data-1 = Specified task
0 : Master task
1 : Sub 1 task
2 : Sub 2 task
3 : Sub 3 task
4 : Sub 4 task
5 : Sub 5 task

Response format : 0000 or Error code

<Example>
Command  CTASK 1
Response  0000

■ Start-up System Commands

START

Starts a job.

If a job name is specified for an operand, the relation between the job and the master job is checked and the execution is started from the beginning of the job. If no job name is specified, the execution is started from the current line number of the set execution job.

Command format : START [Job-Name]

Job-Name = Starting job name (8 characters)
Can be omitted.
5.2 Robot Control Function

Response format: 0000 or Error code

Example:
Command: START WORK-A
Response: 0000

MOVJ
Moves a manipulator to a specified coordinate position in joint motion.

Command format: MOVJ Data-1, Data-2, ..., Data-16

- Data-1 = Motion speed (0.01 to 100.0%)
- Data-2 = Coordinate specification
  0: Base coordinate
  1: Robot coordinate
  2: User coordinate 1
  25: User coordinate 24
- Data-3 = X coordinate value (unit: mm, significant 3 decimal points)
- Data-4 = Y coordinate value (unit: mm, significant 3 decimal points)
- Data-5 = Z coordinate value (unit: mm, significant 3 decimal points)
- Data-6 = Wrist angle TX (unit: degree (°), significant 2 decimal points)
- Data-7 = Wrist angle TY (unit: degree (°), significant 2 decimal points)
- Data-8 = Wrist angle TZ (unit: degree (°), significant 2 decimal points)
- Data-9 = Type
- Data-10 = Tool No. (0 to 23)
- Data-11 = Number of 7th axis pulses (for travel axis, mm)
- Data-12 = Number of 8th axis pulses (for travel axis, mm)
- Data-13 = Number of 9th axis pulses (for travel axis, mm)
- Data-14 = Number of 10th axis pulses
- Data-15 = Number of 11th axis pulses
- Data-16 = Number of 12th axis pulses

- In a system without external axis, Data-11 to Data-16 should be set to "0".
- If a specified user coordinate is not defined, an error occurs.

Response format: 0000 or Error code

Example:
Command: MOVJ 50.0, 2, 123.1, 50.34, 10.8, 180.0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Response: 0000

MOVL
Moves a manipulator to a specified coordinate position in linear motion.

Command format: MOVL Data-1, Data-2, ..., Data-17
5.2 Robot Control Function

Data-1 = Motion speed selection (0 : V (speed), 1 : VR (posture speed))
Data-2 = Motion speed (0.1 to 0.00 mm/s, 0.1 to 0.0°/s)
Data-3 = Coordinate specification
  0 : Base coordinate
  1 : Robot coordinate
  2 : User coordinate 1
  25 : User coordinate 24

Data-4 = X coordinate value (unit : mm, significant 3 decimal points)
Data-5 = Y coordinate value (unit : mm, significant 3 decimal points)
Data-6 = Z coordinate value (unit : mm, significant 3 decimal points)
Data-7 = Wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-8 = Wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-9 = Wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-10 = Type
Data-11 = Tool No. (0 to 23)
Data-12 = Number of 7th axis pulses (for travel axis, mm)
Data-13 = Number of 8th axis pulses (for travel axis, mm)
Data-14 = Number of 9th axis pulses (for travel axis, mm)
Data-15 = Number of 10th axis pulses
Data-16 = Number of 11th axis pulses
Data-17 = Number of 12th axis pulses

  • In a system without external axis, Data-12 to Data-17 should be set to “0”.
  • If a specified user coordinate is not defined, an error occurs.

Response format : 0000 or Error code

<Example>
Command  MOVL 0, 500.0, 2, 123.1, 50.34, 10.8, 180.0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Response  0000

IMOV
Moves a manipulator from the current position for a specified coordinate incremental value in linear motion.

Command format : IMOV Data-1, Data-2, ..., Data-17

Data-1 = Motion speed selection (0 : V (speed), 1 : VR (posture speed))
Data-2 = Motion speed (0.1 to 0.00 mm/s, 0.1 to 0.0°/s)
Data-3 = Coordinate specification
  0 : Base coordinate
  1 : Robot coordinate
Data-4 = X coordinate incremental value (unit: mm, significant 3 decimal points)
Data-5 = Y coordinate incremental value (unit: mm, significant 3 decimal points)
Data-6 = Z coordinate incremental value (unit: mm, significant 3 decimal points)
Data-7 = Wrist angle TX incremental value (unit: degree (°), significant 2 decimal points)
Data-8 = Wrist angle TY incremental value (unit: degree (°), significant 2 decimal points)
Data-9 = Wrist angle TZ incremental value (unit: degree (°), significant 2 decimal points)
Data-10 = Reserved
Data-11 = Tool No. (0 to 23)
Data-12 = Number of 7th axis pulses (for travel axis, mm)
Data-13 = Number of 8th axis pulses (for travel axis, mm)
Data-14 = Number of 9th axis pulses (for travel axis, mm)
Data-15 = Number of 10th axis pulses
Data-16 = Number of 11th axis pulses
Data-17 = Number of 12th axis pulses

- In a system without external axis, Data-12 to Data-17 should be set to “0”.
- If a specified user coordinate is not defined, an error occurs.

Response format: 0000 or Error code

<Example>
Command: IMOV 0, 100.0, 2, 10.0, 10.0, 10.0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Response: 0000

**PMOVJ**
Moves a manipulator to a specified pulse position in joint motion.

Command format: PMOVJ Data-1, Data-2, ⋅⋅⋅, Data-14

Data-1 = Motion speed (0.01 to 100.0 %)
Data-2 = Number of S-axis pulses
Data-3 = Number of L-axis pulses
Data-4 = Number of U-axis pulses
Data-5 = Number of R-axis pulses
Data-6 = Number of B-axis pulses
Data-7 = Number of T-axis pulses
Data-8 = Tool No. (0 to 23)
Data-9 = Number of 7th axis pulses
Data-10 = Number of 8th axis pulses
Data-11 = Number of 9th axis pulses
Data-12 = Number of 10th axis pulses
Data-13 = Number of 11th axis pulses
Data-14 = Number of 12th axis pulses
  • In a system without external axis, Data-9 to Data-14 should be set to “0”.

Response format: 0000 or Error code

<Example>
Command: PMOVJ 20.0, 100, 200, 300, 400, 500, 0, 0, 0, 0, 0, 0, 0, 0
Response: 0000

**PMOVL**
Moves a manipulator to a specified pulse position in linear motion.

Command format: PMOVL Data-1, Data-2, ..., Data-15

Data-1 = Motion speed selection (0: V(speed), 1: VR(posture speed))
Data-2 = Motion speed (0.1 to 999.9 mm/s, 0.1 to 999.9°/s)
Data-3 = Number of S-axis pulses
Data-4 = Number of L-axis pulses
Data-5 = Number of U-axis pulses
Data-6 = Number of R-axis pulses
Data-7 = Number of B-axis pulses
Data-8 = Number of T-axis pulses
Data-9 = Tool No. (0 to 23)
Data-10 = Number of 7th axis pulses
Data-11 = Number of 8th axis pulses
Data-12 = Number of 9th axis pulses
Data-13 = Number of 10th axis pulses
Data-14 = Number of 11th axis pulses
Data-15 = Number of 12th axis pulses
  • In a system without external axis, Data-10 to Data-15 should be set to “0”.

Response format: 0000 or Error code

<Example>
Command: PMOVL 0, 123.0, 10, 200, 300, 400, 500, 0, 0, 0, 0, 0, 0, 0, 0
Response: 0000

---

**Editing System Commands**

**DELETE**
Deletes a specified job.

Command format: DELETE Job-Name

Job-Name = Job name to be deleted (8 characters which can be processed in MS-DOS)
  = * : Delete all jobs
Response format: 0000 or Error code

**Example**
Command: DELETE WORK-B
Response: 0000

**CVTRJ (Optional)**
Converts a specified job to a relative job of a specified coordinate.

Command format: CVTRJ Data-1, Data-2

Data-1 = Name of job to be converted
Data-2 = Conversion coordinate system specification
  0: Base coordinate
  1: Robot coordinate
  2: User coordinate 1
  25: User coordinate 24
  26: Reserved

• If the specified user coordinate is not defined, an error occurs.

Response format: 0000 or Error code

**Example**
Command: CVTRJ TESTJOB, 2
Response: 0000

**NOTE**
This function requires the relative job function of the XRC.

**CVTSJ (Optional)**
Converts a specified job to a standard job (pulse job) in a specified converting method.

Command format: CVTSJ Data-1, Data-2, Data-3

Data-1 = Name of job to be converted
Data-2 = Converting method specification
  0: Previous step regarded (B-axis sign same)
  1: Form regarded
  2: Previous step regarded (R-axis travel amount minimum)

Data-3 = Reference position variable. Position variable No. indicating the first step conversion reference position when the previous step is regarded.

Response format: 0000 or Error code
5.2 Robot Control Function

<Example>
Command  CVTSJ SAMPLE01, 1, 0
Response   0000

In the above example, P000 is to be the reference point and the job “SAMPLE01” is converted to a standard job with the form regarded.

NOTE This function required the relative job function of the XRC.

WUFRAME
Writes a user coordinate data to a specified user coordinate system.

Command format : WUFRAME Data-1, Data-2, ..., Data-29

Data-1  =  User coordinate No.
         0 : Reserved
         1 : Reserved
         2 : User coordinate 1
             ...
         25 : User coordinate 24

Data-2  =  ORG X coordinate value (unit : mm, significant 3 decimal points)
Data-3  =  ORG Y coordinate value (unit : mm, significant 3 decimal points)
Data-4  =  ORG Z coordinate value (unit : mm, significant 3 decimal points)
Data-5  =  ORG wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-6  =  ORG wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-7  =  ORG wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-8  =  ORG type
Data-9  =  XX X coordinate value (unit : mm, significant 3 decimal points)
Data-10 =  XX Y coordinate value (unit : mm, significant 3 decimal points)
Data-11 =  XX Z coordinate value (unit : mm, significant 3 decimal points)
Data-12 =  XX wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-13 =  XX wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-14 =  XX wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-15 =  XX type
Data-16 =  XY X coordinate value (unit : mm, significant 3 decimal points)
Data-17 =  XY Y coordinate value (unit : mm, significant 3 decimal points)
Data-18 =  XY Z coordinate value (unit : mm, significant 3 decimal points)
Data-19 =  XY wrist angle TX (unit : degree (°), significant 2 decimal points)
Data-20 =  XY wrist angle TY (unit : degree (°), significant 2 decimal points)
Data-21 =  XY wrist angle TZ (unit : degree (°), significant 2 decimal points)
Data-22 =  XY type
Data-23 =  Tool No. (0 to 23)
Data-24 =  Number of 7th axis pulses (for travel axis, mm)
5.2 Robot Control Function

Data-25 = Number of 8th axis pulses (for travel axis, mm)
Data-26 = Number of 9th axis pulses (for travel axis, mm)
Data-27 = Number of 10th axis pulses
Data-28 = Number of 11th axis pulses
Data-29 = Number of 12th axis pulses

Response format : 0000 or Error code

- ORG, XX, and XY coordinate are written in the base coordinate system.
- In a system without external axis, Data-24 to Data-29 should be set to “0”.
- If the group axis of the specified user coordinate system is not R1, an error occurs.
- For base axis data of ORG, XX, and XY, the same data should be used.

Example:
Command  WUFRAME 2, 600.0, 12.34, 500.0, 180.0, 0.0, 0.0, 0, ..., 0
Response  0000

LOADV

Receives variable data from a host computer and write it in a specified variable.

Command format : LOADV Data-1, Data-2, ..., Data-12

Data-1 = Type of variables
    0 : Byte type variables
    1 : Integer type variables
    2 : Double precision type variables
    3 : Real number type variables
    4 : Robot axis position type variables
    5 : Base axis position type variables
    6 : Station axis position type variables (only pulse type)

Data-2 = Variable No.
Data-3 = Byte value / Integer value / Double precision type integer value / Real number value
/ Position data type
    Position data type = 0 : Pulse type
                        1 : Cartesian type

(When the position data type is 0)
Data-4 = Number of S-axis pulses / Number of base 1st axis pulses / Number of station 1st axis pulses
Data-5 = Number of L-axis pulses / Number of base 2nd axis pulses / Number of station 2nd axis pulses
Data-6 = Number of U-axis pulses / Number of base 3rd axis pulses / Number of station 3rd axis pulses
Data-7 = Number of R-axis pulses / Number of base 4th axis pulses / Number of station 4th axis pulses
5.2 Robot Control Function

Data-8  = Number of B-axis pulses / Number of base 5th axis pulses / Number of station 5th axis pulses
Data-9  = Number of T axis pulses / Number of base 6th axis pulses / Number of station 6th axis pulses
Data-10 = Tool No.

(When the position data type is 1)
Data-4  = Coordinate data
Coordinate data = 0 : Base coordinate
                 1 : Robot coordinate
                 2 : User coordinate 1
                 3 : User coordinate 2
                 ...
                 25 : User coordinate 24
                 26 : Tool coordinate
                 27 : Master tool coordinate

Data-5  = X coordinate value / Base 1st axis Cartesian value (unit : mm, significant 3 decimal points)
Data-6  = Y coordinate value / Base 2nd axis Cartesian value (unit : mm, significant 3 decimal points)
Data-7  = Z coordinate value / Base 3rd axis Cartesian value (unit : mm, significant 3 decimal points)
Data-8  = Wrist angle RX coordinate value (unit : degree (°), significant 2 decimal points)
Data-9  = Wrist angle RY coordinate value (unit : degree (°), significant 2 decimal points)
Data-10 = Wrist angle RZ coordinate value (unit : degree (°), significant 2 decimal points)
Data-11 = Form
Data of the form is a value obtained by converting the following bit data to decimal notation.

Response format : 0000 or Error code

<Example>
Command  LOADV 0, 0, 123
Response   0000

Data-12  = Tool No.

Data of the form is a value obtained by converting the following bit data to decimal notation.

Response format : 0000 or Error code

<Example>
Command  LOADV 0, 0, 123
Response   0000
In the above example, 123 is stored in the XRC byte type variable B000.

## Job Selection System Commands

### SETMJ
Sets a specified job as a master job.
At the same time, the specified job is set as a execution job.

Command format: SETMJ Job-Name

Job-Name = Job name to be set (8 characters which can be processed in MS-DOS)

Response format: 0000 or Error code

<Example>
Command: SETMJ WORK-C
Response: 0000

### JSEQ
Sets a job name and a line No.

Command format: JSEQ Data-1, Data-2

Data-1 = Job name to be set (8 characters which can be processed in MS-DOS)
Data-2 = Line No. to be set (0 to 9999)

Response format: 0000 or Error code

<Example>
Command: JSEQ WORK-A, 10
Response: 0000
5.2 Robot Control Function

5.2.7 I/O Read/Write Function

The host control function can read out or write in (change) I/O signal status using the host computer.

The following table shows the number of signals and the types of signals to be sent or received by the host control function.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Signal Range (Qty)</th>
<th>Classification</th>
<th>Read-out</th>
<th>Write-in</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xxx</td>
<td>0010 to 0247 (192)</td>
<td>General input signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>1xxx</td>
<td>1010 to 1247 (192)</td>
<td>General output signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>2xxx</td>
<td>2010 to 2327 (256)</td>
<td>External input signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>3xxx</td>
<td>3010 to 3327 (256)</td>
<td>External output signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>4xxx</td>
<td>4010 to 4287 (224)</td>
<td>Specific input signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>5xxx</td>
<td>5010 to 5387 (304)</td>
<td>Specific output signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>6xxx</td>
<td>6000 to 6787 (488)</td>
<td>Auxiliary relay</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>7xxx</td>
<td>7010 to 7887 (704)</td>
<td>Control status signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>8xxx</td>
<td>8010 to 8107 (80)</td>
<td>Control status signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>82xx</td>
<td>8210 to 8247 (32)</td>
<td>Pseudo input signal</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>9xxx</td>
<td>9010 to 9327 (256)</td>
<td>Network input</td>
<td>●</td>
<td>●</td>
</tr>
</tbody>
</table>

○ : Possible to execute

Transmission Procedure

The transmission from the host computer proceeds as follows.

1. The ENQ code is sent from the host computer to establish a data link.
2. After the data link is established, the data is sent from the host computer. The data transmission should be completed in a single block.
3. After the request to send is accepted, the host computer should be ready to receive. The XRC sends the ENQ code to establish the data link.
4. After the data link is established, the data sent from the XRC is received to terminate the transmission at completion of receipt.

The read/write function can be distinguished by the header number. Refer to the header number list.
### Read-out of I/O Signal Status

The I/O signal status can be read out when the read/write function is enabled and an alarm or an error is not occurring.

---

**DATA-1**

Command format: Data-1, Data-2

- **Data-1** = Start No.
- **Data-2** = Number of data points

---

**DATA-2**

Response format (at normal completion): Data-1, Data-2, ..., Data-256

- **Data-1** = First 8 points of data
- **Data-2** = Second 8 points of data
- **Data-256** = Last (up to 256th) 8 points of data

Response format (at abnormal completion): [SOH] 90,000 [STX] Error code [CR][ETX] [BCC]

Error code: Number with 4 digits other than 0000

Number smaller than 1000, 0 is added before the number.

---

<Example>

When 3 points are read out from 7010

**Command** 7010, 3

**Response** 2, 0, 5
Write-in of I/O Signal Status

The write-in of I/O signal status is enabled when the manipulator is stopped in teach mode and when no alarm or error is occurring.

DATA-3
Command format : Data-1, Data-2, Data-3, Data-4, ..., Data-258
Data-1 = Start No.
Data-2 = Number of data points
Data-3 = First 8 points of data
Data-4 = Second 8 points of data

Data-258 = Last (up to 256th) 8 points of data

DATA-4
Response format (at normal completion) : 0000
Response format (at abnormal completion) : Number with 4 digits other than 0000
Number smaller than 1000, 0 is added before the number.

<Example>
When status of 3 points is changed from 9010
Command 9010, 3, 4, 3, 12
Response 0000
5.3 Commands for Multi-control Group and Independent Control Functions

5.3.1 Commands for Multi-control Group

The XRC can control more than one manipulator or station simultaneously. The following commands are available for this multi-control function.

- CGROUP : Changing the control group
- RGROUP : Reading the control group and task selected status

The following table shows the combination which can be set by using the above commands.

<table>
<thead>
<tr>
<th>R1 (robot 1)</th>
<th>R2 (robot 2)</th>
<th>R3 (robot 3)</th>
<th>S□*1</th>
</tr>
</thead>
<tbody>
<tr>
<td>×</td>
<td>×</td>
<td>×</td>
<td>●</td>
</tr>
<tr>
<td>●*2</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>●</td>
<td>×</td>
<td>×</td>
<td>●</td>
</tr>
<tr>
<td>×</td>
<td>●</td>
<td>×</td>
<td>●</td>
</tr>
<tr>
<td>×</td>
<td>●</td>
<td>×</td>
<td>●</td>
</tr>
<tr>
<td>×</td>
<td>×</td>
<td>●</td>
<td>×</td>
</tr>
<tr>
<td>×</td>
<td>×</td>
<td>●</td>
<td>●</td>
</tr>
</tbody>
</table>

*1 Either one station among S1 to S6 can be selected in a system having several stations.
*2 Base axes is included in robot axes.

The following commands have influence when the above commands are used. The operations of these commands are applicable to the set control group.

<table>
<thead>
<tr>
<th>Read System Commands</th>
<th>Startup System Commands</th>
<th>Editing System Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPOSJ</td>
<td>MOVJ</td>
<td>WUFRAME</td>
</tr>
<tr>
<td>RPOSC</td>
<td>MOVL</td>
<td></td>
</tr>
<tr>
<td>RUFRAME</td>
<td>IMOV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PMOVJ</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PMOVL</td>
<td></td>
</tr>
</tbody>
</table>
5.3.2 Commands for Independent Control Function

The XRC supports the independent control function which can execute more than one job simultaneously. For this independent function, the following commands are available.

- CTASK : Changing the tasks
- RGROUP : Reading the control group or task selected status

By using the above commands, a task to be controlled can be changed.

The following commands have influence when the independent control function is used.

1. **Job startup (START)**
   - Starts up a job. When a job name specification is provided for operand, execution of that job is started from the head of job as a task that is currently selected. When a job name is not specified, all tasks that are currently set are executed from the current line No.

2. **Waiting for completion of startup (J WAIT)**
   - As a response, returns the information whether the currently selected task operation has been completed.

3. **Master job registration (SETMJ)**
   - Sets a specified job as a master job, to the currently selected task.

4. **Job selection (J SEQ)**
   - Sets a job name, a line No. to the currently selected task.

5. **Read of selected job (RJ SEQ)**
   - Reads the job name, line No., and step No. of the currently selected task.

6. **Read of status (RSTATS)**
   - Returns the system status disregarding the selected task status. However, the “running” status differs from the conventional status; the “running” is entered even if only one task was operating.
### 5.4 Alarm Codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Contents</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>4112</td>
<td>Data sending error</td>
<td>The EOT code is sent out and the data link is canceled.</td>
</tr>
<tr>
<td></td>
<td>1 : NAK retry over</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : Timer A timeup retry over</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : Alternating response error retry over</td>
<td></td>
</tr>
<tr>
<td>4113</td>
<td>Data receiving error</td>
<td>For 3 to 6, the EOT code is sent out and the data link is canceled.</td>
</tr>
<tr>
<td></td>
<td>1 : Receiving timeup (Timer A)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : Receiving timeup (Timer B)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : Short heading length</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 : Long heading length</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 : Illegal header No.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6 : Text longer than 256 characters</td>
<td></td>
</tr>
<tr>
<td>4114</td>
<td>Transmission hardware error</td>
<td>The EOT code is not sent.</td>
</tr>
<tr>
<td></td>
<td>1 : Overrun error</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : Parity error</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : Framing error</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 : Sending timeup (Timer A)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 : Sending timeup (Timer B)</td>
<td></td>
</tr>
<tr>
<td>4115</td>
<td>Transmission system block</td>
<td>For 4, the EOT code is sent out and the data link is canceled.</td>
</tr>
<tr>
<td></td>
<td>This alarm notifies that the transmission procedure is correct but the received contents makes inconsistency in the system. Usually this alarm is resulted from violation of rules on the other party or illegal notification.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : EOT was received while waiting for ACK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : EOT was received while waiting for ENQ</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : EOT was received before receiving the last block</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 : Code other than EOT was received after receiving the last block</td>
<td></td>
</tr>
<tr>
<td>4206</td>
<td>Transmission system error</td>
<td>The EOT code is not sent.</td>
</tr>
<tr>
<td></td>
<td>This alarm notifies an error on processing of transmission system. This alarm occurs in the following cases.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>100 Error in transmission task</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• A job containing position type variable of which the value is not set, was to be saved.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• A job which does not exist on the memory, was to be saved.</td>
<td></td>
</tr>
</tbody>
</table>
5.5 Interpreter Message List

The interpreter messages are classified into the following categories.
- 1xxx : Command text general error
- 2xxx : Command execution mode error
- 3xxx : Command execution error
- 4xxx : Job registration error
- 5xxx : File contents error

<table>
<thead>
<tr>
<th>Code</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1010</td>
<td>Command error</td>
</tr>
<tr>
<td>1011</td>
<td>Error in number of command operands</td>
</tr>
<tr>
<td>1012</td>
<td>Command operand value range over</td>
</tr>
<tr>
<td>1013</td>
<td>Command operand length error</td>
</tr>
<tr>
<td>1020</td>
<td>Disk full of files</td>
</tr>
<tr>
<td>2010</td>
<td>Manipulator operating</td>
</tr>
<tr>
<td>2020</td>
<td>Hold by programming pendant</td>
</tr>
<tr>
<td>2030</td>
<td>Hold by playback panel</td>
</tr>
<tr>
<td>2040</td>
<td>External hold</td>
</tr>
<tr>
<td>2050</td>
<td>Command hold</td>
</tr>
<tr>
<td>2060</td>
<td>Error/alarm occurring</td>
</tr>
<tr>
<td>2070</td>
<td>Servo OFF</td>
</tr>
<tr>
<td>2080</td>
<td>Incorrect mode</td>
</tr>
<tr>
<td>2090</td>
<td>File accessing by other function</td>
</tr>
<tr>
<td>2100</td>
<td>Command remote not set</td>
</tr>
<tr>
<td>2110</td>
<td>This data can not be accessed.</td>
</tr>
<tr>
<td>2120</td>
<td>This data can not be loaded.</td>
</tr>
<tr>
<td>2130</td>
<td>Editing</td>
</tr>
<tr>
<td>3010</td>
<td>Turn ON the servo power.</td>
</tr>
<tr>
<td>Code</td>
<td>Contents</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>3040</td>
<td>Perform home positioning.</td>
</tr>
<tr>
<td>3050</td>
<td>Confirm positions.</td>
</tr>
<tr>
<td>3070</td>
<td>Current value not made</td>
</tr>
<tr>
<td>3220</td>
<td>Panel lock ; mode/cycle prohibit signal is ON.</td>
</tr>
<tr>
<td>3230</td>
<td>Panel lock ; start prohibit signal is ON.</td>
</tr>
<tr>
<td>3350</td>
<td>User coordinate not taught</td>
</tr>
<tr>
<td>3360</td>
<td>User file destroyed</td>
</tr>
<tr>
<td>3370</td>
<td>Incorrect control group</td>
</tr>
<tr>
<td>3380</td>
<td>Incorrect base axis data</td>
</tr>
<tr>
<td>3390</td>
<td>Relative job conversion prohibit (at CVTRJ)</td>
</tr>
<tr>
<td>3400</td>
<td>Master call prohibit (parameter)</td>
</tr>
<tr>
<td>3410</td>
<td>Master call prohibit (lamp On during operation)</td>
</tr>
<tr>
<td>3420</td>
<td>Master call prohibit (teach lock)</td>
</tr>
<tr>
<td>3430</td>
<td>Robot calibration data not defined</td>
</tr>
<tr>
<td>4010</td>
<td>Insufficient memory capacity (job registered memory)</td>
</tr>
<tr>
<td>4012</td>
<td>Insufficient memory capacity (position data registered memory)</td>
</tr>
<tr>
<td>4020</td>
<td>Job editing prohibit</td>
</tr>
<tr>
<td>4030</td>
<td>Same job name exists</td>
</tr>
<tr>
<td>4040</td>
<td>No specified job</td>
</tr>
<tr>
<td>4060</td>
<td>Set a execution job.</td>
</tr>
<tr>
<td>4120</td>
<td>Position data destroyed</td>
</tr>
<tr>
<td>4130</td>
<td>Position data not exist</td>
</tr>
<tr>
<td>4140</td>
<td>Incorrect position variable type</td>
</tr>
<tr>
<td>4150</td>
<td>END instruction for job which is not master job</td>
</tr>
<tr>
<td>4170</td>
<td>Instruction data destroyed</td>
</tr>
<tr>
<td>Code</td>
<td>Contents</td>
</tr>
<tr>
<td>-------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>4190</td>
<td>Invalid character in job name</td>
</tr>
<tr>
<td>4200</td>
<td>Invalid character in label name</td>
</tr>
<tr>
<td>4230</td>
<td>Invalid instruction in this system</td>
</tr>
<tr>
<td>4420</td>
<td>No step in job to be converted</td>
</tr>
<tr>
<td>4430</td>
<td>Already converted</td>
</tr>
<tr>
<td>4480</td>
<td>Teach user coordinate.</td>
</tr>
<tr>
<td>4490</td>
<td>Relative job/Independent control function not permitted</td>
</tr>
<tr>
<td>5110</td>
<td>Syntax error (syntax of instruction)</td>
</tr>
<tr>
<td>5120</td>
<td>Position data error</td>
</tr>
<tr>
<td>5130</td>
<td>No NOP or END instruction</td>
</tr>
<tr>
<td>5170</td>
<td>Format error (incorrect format)</td>
</tr>
<tr>
<td>5180</td>
<td>Incorrect number of data</td>
</tr>
<tr>
<td>5200</td>
<td>Data range over</td>
</tr>
<tr>
<td>5310</td>
<td>Syntax error (except instruction)</td>
</tr>
<tr>
<td>5340</td>
<td>Error in pseudo instruction specification</td>
</tr>
<tr>
<td>5370</td>
<td>Error in condition data record</td>
</tr>
<tr>
<td>5390</td>
<td>Error in job data record</td>
</tr>
<tr>
<td>5430</td>
<td>System not matched</td>
</tr>
<tr>
<td>5480</td>
<td>Incorrect welding function type</td>
</tr>
</tbody>
</table>
# 6 Data List

## 6.1 Header Number List

<table>
<thead>
<tr>
<th>Header Number</th>
<th>Contents</th>
<th>File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>01, 000</td>
<td>Command from a external computer</td>
<td></td>
</tr>
<tr>
<td>02, 001</td>
<td>Single job data</td>
<td>xxxxxxxxx. JBI</td>
</tr>
<tr>
<td>002</td>
<td>Related job data</td>
<td>xxxxxxxxx. JBR</td>
</tr>
<tr>
<td>02, 051</td>
<td>Request for single job data</td>
<td></td>
</tr>
<tr>
<td>052</td>
<td>Request for related job data</td>
<td></td>
</tr>
<tr>
<td>02, 200</td>
<td>Tool data</td>
<td>TOOL. CND</td>
</tr>
<tr>
<td>201</td>
<td>Weaving condition data</td>
<td>WEAV. CND</td>
</tr>
<tr>
<td>202</td>
<td>User coordinate data</td>
<td>UFRAME. CND</td>
</tr>
<tr>
<td>203</td>
<td>Welding start condition data</td>
<td>ARCSRT. CND</td>
</tr>
<tr>
<td>204</td>
<td>Welding end condition data</td>
<td>ARCEND. CND</td>
</tr>
<tr>
<td>232</td>
<td>Variable data</td>
<td>VAR. DAT</td>
</tr>
<tr>
<td>240</td>
<td>System information</td>
<td>SYSTEM. SYS</td>
</tr>
<tr>
<td>241</td>
<td>Alarm history data</td>
<td>ALMHIST. DAT</td>
</tr>
<tr>
<td>02, 300</td>
<td>Request for tool data</td>
<td></td>
</tr>
<tr>
<td>301</td>
<td>Request for weaving condition data</td>
<td></td>
</tr>
<tr>
<td>302</td>
<td>Request for user coordinate data</td>
<td></td>
</tr>
<tr>
<td>303</td>
<td>Request for welding start condition data</td>
<td></td>
</tr>
<tr>
<td>304</td>
<td>Request for welding end condition data</td>
<td></td>
</tr>
<tr>
<td>332</td>
<td>Request for variable data</td>
<td></td>
</tr>
<tr>
<td>340</td>
<td>Request for system information</td>
<td></td>
</tr>
<tr>
<td>341</td>
<td>Request for alarm history data</td>
<td></td>
</tr>
<tr>
<td>Contents</td>
<td>File Name</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>03, 001</td>
<td>Byte type variable</td>
<td></td>
</tr>
<tr>
<td>002</td>
<td>Integer type variable</td>
<td></td>
</tr>
<tr>
<td>003</td>
<td>Double precision type variable</td>
<td></td>
</tr>
<tr>
<td>004</td>
<td>Real number type variable</td>
<td></td>
</tr>
<tr>
<td>005</td>
<td>Robot axis position type variable (pulse type)</td>
<td></td>
</tr>
<tr>
<td>006</td>
<td>Robot axis position type variable (XYZ type)</td>
<td></td>
</tr>
<tr>
<td>007</td>
<td>External axis position type variable (pulse type)</td>
<td></td>
</tr>
<tr>
<td>008</td>
<td>External axis position type variable (XYZ type)</td>
<td></td>
</tr>
<tr>
<td>03, 051</td>
<td>Request for byte type variable</td>
<td></td>
</tr>
<tr>
<td>052</td>
<td>Request for integer type variable</td>
<td></td>
</tr>
<tr>
<td>053</td>
<td>Request for double precision type variable</td>
<td></td>
</tr>
<tr>
<td>054</td>
<td>Request for real number type variable</td>
<td></td>
</tr>
<tr>
<td>055</td>
<td>Request for robot axis position type variable (pulse type)</td>
<td></td>
</tr>
<tr>
<td>056</td>
<td>Request for robot axis position type variable (XYZ type)</td>
<td></td>
</tr>
<tr>
<td>057</td>
<td>Request for external axis position type variable (pulse type)</td>
<td></td>
</tr>
<tr>
<td>058</td>
<td>Request for external axis position type variable (XYZ type)</td>
<td></td>
</tr>
<tr>
<td>04, 001</td>
<td>Request for write-in of I/O signals</td>
<td></td>
</tr>
<tr>
<td>051</td>
<td>Request for read-out of I/O signals</td>
<td></td>
</tr>
<tr>
<td>90, 000</td>
<td>Command or data response (normal/error)</td>
<td></td>
</tr>
<tr>
<td>001</td>
<td>Command or data response (data)</td>
<td></td>
</tr>
</tbody>
</table>
# 6.2 Parameter List

## Parameter for Transmission

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents and Set Value</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2C110</td>
<td>Playback panel operation (in remote) specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 : Invalid</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 : Valid</td>
<td></td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Diagram" /></td>
<td></td>
</tr>
</tbody>
</table>

## Parameter for Transmission (for BSC protocol)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Contents and Set Value</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS030</td>
<td>Number of data bits</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>7 : 7 (bit)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>8 : 8</td>
<td></td>
</tr>
<tr>
<td>RS031</td>
<td>Number of stop bits</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0 : 1 (bit)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : 1.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : 2</td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Contents and Set Value</td>
<td>Initial Value</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>RS032</td>
<td>Parity specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 : No specification</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>1 : Odd parity</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 : Even parity</td>
<td></td>
</tr>
<tr>
<td>RS033</td>
<td>Transmission speed specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 : 150 (baud rate)</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>2 : 300</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 : 600</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 : 1200</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 : 2400</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6 : 4800</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7 : 9600</td>
<td></td>
</tr>
<tr>
<td>RS034</td>
<td>Timer A : Sequence monitoring timer</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Serves as protection against invalid response or no response</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unit : 0.1 sec. (Setting range : 0 to 100)</td>
<td></td>
</tr>
<tr>
<td>RS035</td>
<td>Timer B : Text reception monitoring timer</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Serves as protection against no response of text end character</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unit : 0.1 sec. (Setting range : 0 to 255)</td>
<td></td>
</tr>
<tr>
<td>RS036</td>
<td>Retry 1 : Number of resendings of a control character for invalid response or no response</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Setting range : 0 to 30</td>
<td></td>
</tr>
<tr>
<td>RS037</td>
<td>Retry 2 : Number of resendings of a text for a block check error (reception of NAK)</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Setting range : 0 to 10</td>
<td></td>
</tr>
<tr>
<td>RS038</td>
<td>Block check method</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0 : Check sum</td>
<td></td>
</tr>
</tbody>
</table>
# 7 Comparison of Data Transmission Functions

## Comparison of Functions Related to Controller Basic Functionality

<table>
<thead>
<tr>
<th>Function</th>
<th>XRC</th>
<th>MRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiport Function Setup</td>
<td>Not supported</td>
<td>Supported</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Two ports are available for FC1 and data transmission</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Serial processing is possible but parallel processing impossible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Not supported with MRC II.</td>
</tr>
</tbody>
</table>

## Comparison of Functions Related to Data Transmission (DCI)

<table>
<thead>
<tr>
<th>Function</th>
<th>XRC</th>
<th>MRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOADJ, SAVEJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>LOADV, SAVEV</td>
<td>Supported</td>
<td>Supported</td>
</tr>
</tbody>
</table>

## Comparison of Functions Related to Data Transmission (Stand-alone)

<table>
<thead>
<tr>
<th>Function</th>
<th>XRC</th>
<th>MRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job (Single/Related)</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>Save, Load, Verify</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Condition data Save, Load, Verify</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>File names of option function are incompatible.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System data Save</td>
<td>Supported</td>
<td>Supported</td>
</tr>
</tbody>
</table>

## Comparison of Functions Related to Data Transmission (Host Control)

<table>
<thead>
<tr>
<th>Function</th>
<th>XRC</th>
<th>MRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPLOAD, DOWNLOAD</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>File names of condition data option function are incompatible.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RALARM</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RPOS</td>
<td>Not supported</td>
<td>Supported</td>
</tr>
<tr>
<td>Function</td>
<td>XRC</td>
<td>MRC</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>RPOSJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RSTATS</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RJSEQ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RJDIR</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RPOSC</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>JWAIT</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RUFRAME</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>START</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>HOLD</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>RESET</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>CANCEL</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>DELETE</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>SETMJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>JSEQ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>MODE</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>CYCLE</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>SVON</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>HLOCK</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>MDSP</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>MOVJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>MOVL</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>IMOV</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>PMOVJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>PMOVL</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>CVTRJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>WUFRAME</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>CGROUP</td>
<td>Supported</td>
<td>Supported</td>
</tr>
</tbody>
</table>

Control group information differs depending on the number of manipulators in a system.

| RGROUP     | Supported | Supported |
| CTASK      | Supported | Supported |
Comparison of Functions Related to Data Transmission (Host Control)

<table>
<thead>
<tr>
<th>Function</th>
<th>XRC</th>
<th>MRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CVTSJ</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>SAVEV</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>LOADV</td>
<td>Supported</td>
<td>Supported</td>
</tr>
</tbody>
</table>
8 Remote Function Setting

Whether I/O remote control or command remote control should be enabled in the remote mode can be set in the pseudo input display.

**Operation**

Select {IN/OUT} under the top menu ➔ Select {PSEUDO INPUT SIG} ➔ Select an item*1

**Explanation**

*1 Select “INHIBIT IO” or “CMD REMOTE SEL”.
The item enabled is marked with “●” while the item disabled is marked with “○”.

<table>
<thead>
<tr>
<th>DATA</th>
<th>EDIT</th>
<th>DISPLAY</th>
<th>UTILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM SECTION</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#8210</td>
<td>○</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#8211</td>
<td>○</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#8212</td>
<td>○</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#8213</td>
<td>○</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#8214</td>
<td>○</td>
<td>INHIBIT IO</td>
<td></td>
</tr>
<tr>
<td>#8215</td>
<td>○</td>
<td>CMD REMOTE SEL</td>
<td></td>
</tr>
<tr>
<td>#8216</td>
<td>○</td>
<td>INHIBIT PP/PANEL</td>
<td></td>
</tr>
<tr>
<td>#8217</td>
<td>○</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

When INHIBIT IO is marked with ○ (disabled), the I/O remote function is enabled.
When CMD REMOTE SEL is marked with ● (enabled), the command remote function is enabled.

When INHIBIT IO is marked with ○ (disabled), the I/O remote function is enabled so that the operation from external I/O is enabled with the P.PANEL remote key pressed. When INHIBIT IO is marked with ● (enabled), the operation from external I/O is disabled.

When CMD REMOTE SEL is marked with ● (enabled), the host control function is enabled with the P.PANEL remote key pressed. When CMD REMOTE SEL is marked with ○ (disabled), the host control function is disabled.

When INHIBIT PP/PANEL is marked with ○ (disabled), the operation from P.P/P.PANEL is enabled even in remote mode. When INHIBIT PP/PANEL is marked with ● (enabled), the operation from P.P/P.PANEL is disabled, except for the operations of emergency stop, hold, and remote key.
YASNAC XRC OPTIONS
INSTRUCTIONS
FOR DATA TRANSMISSION FUNCTION