YASKAWA

FS100 OPTIONS INSTRUCTIONS

FOR DATA TRANSMISSION FUNCTION

Upon receipt of the product and prior to initial operation, read these instructions thoroughly, and retain for future reference.

MOTOMAN INSTRUCTIONS

MOTOMAN-□□□ INSTRUCTIONS
FS100 INSTRUCTIONS
FS100 OPERATOR'S MANUAL
FS100 MAINTENANCE MANUAL

YASKAWA ELECTRIC CORPORATION

YASKAWA

MANUAL NO. RE-CKI-A459

MANDATORY

- This manual explains the data transmission function of the FS100 system. Read this manual carefully and be sure to understand its contents before handling the FS100.
- General items related to safety are listed in Chapter 1: Safety of the FS100 Instructions. To ensure correct and safe operation, carefully read the FS100 Instructions 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 FS100.

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

To ensure safe and efficient operation at all times, be sure to follow all instructions, even if not designated as "CAUTION" and "WARNING".

WARNING

- Before operating the manipulator, check that servo power is turned off when the emergency stop button on the programing pendant is pressed.
  When the servo power is turned off, the SERVO ON LED on the programing pendant is 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.

Fig. : Emergency Stop Button

- In the case of not using the programming pendant, be sure to supply the emergency stop button on the equipment. Then before operating the manipulator, check to be sure that the servo power is turned OFF by pressing the emergency stop button.
  Connect the external emergency stop button to the 5-6 pin and 16-17 pin of the robot system signal connector (CN2).
- Upon shipment of the FS100, this signal is connected by a jumper cable in the dummy connector. To use the signal, make sure to prepare a new connector, and then input it.

If the signal is input with the jumper cable connected, it does not function, which may result in personal injury or equipment damage.

- 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.

Fig. : Release of EM

- Observe the following precautions when performing teaching operations within the P-point maximum 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.

The emergency stop button is located on the programing pendant.

WARNING

- Confirm that no person is present in the P-point maximum envelope of the manipulator and that you are in a safe location before:
  - Turning on the power for the FS100.
  - Moving the manipulator with the programming pendant.
  - Running the system in the check mode.
  - 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.

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 cabinet of the FS100 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 Warning Labels in the FS100 Instructions before operating the manipulator:

Definition of Terms Used Often in This Manual

The MOTOMAN is the YASKAWA industrial robot product.

The MOTOMAN usually consists of the manipulator, the FS100 controller, manipulator cables, the FS100 programming pendant (optional), and the FS100 programming pendant dummy connector (optional).

In this manual, the equipment is designated as follows:

Equipment	Manual Designation
FS100 controller	FS100
FS100 programming pendant	Programming pendant
Cable between the manipulator and the controller	Manipulator Cable
FS100 programming pendant dummy connector	Programming pendant dummy connector

Descriptions of the programming pendant keys, buttons, displays and keyboard of the PC are shown as follows:

Equipment		Manual Designation
Programming Pendant	Character Keys	The keys which have characters printed on them are denoted with [ ]. ex. [ENTER]
	Symbol Keys	The keys which have a symbol printed on them are not denoted with [ ] but depicted with a small picture. ex. PAGE key  The Cursor is an exception, and a picture is not shown.
	Axis Keys Numeric Keys	“Axis Keys” and “Numeric Keys” are generic names for the keys for axis operation and number input.
	Keys pressed simultaneously	When two keys are to be pressed simultaneously, the keys are shown with a “+” sign between them, ex. SHIFTkey  +COORD key 
	Mode Key	Three kinds of modes that can be selected by the mode key are denoted as follows: REMOTE, PLAY, or TEACH
	Button	Three buttons on the upper side of the programming pendant are denoted as follows: HOLD button START button EMERGENCY STOP button
	Displays	The menu displayed in the programming pendant is denoted with { }. ex. {JOB}
PC Keyboard		The name of the key is denoted ex. Ctrl key on the keyboard

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, or that the item is directly selected by touching the screen.

Registered Trademark

In this manual, names of companies, corporations, or products are trademarks, registered trademarks, or brand names for each company or corporation. The indications of (R) and TM are omitted.

1 Outline

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

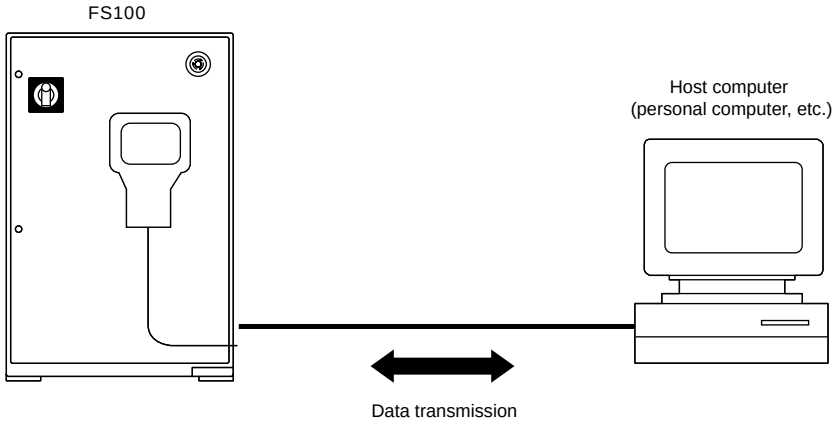
The data transmission function adopts a serial transmission line and standard protocol, making easy connection to a host computer.

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



1.1 DCI 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.

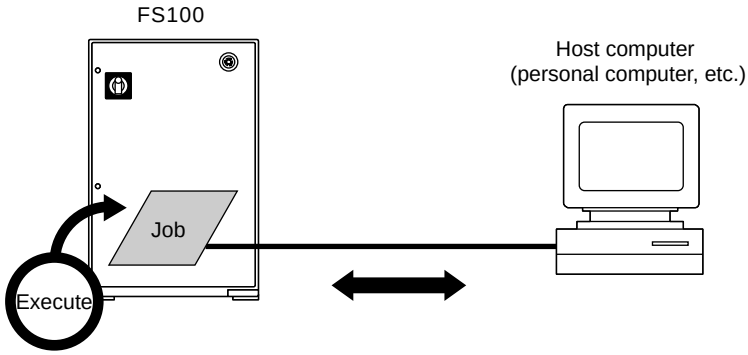


Table 1-1: DCI Function

Job Transmission	**Load**	Job can be transmitted in either mode. • Single job • Related job
	Save	
	Delete	
Variable Transmission	**Load**	• Byte type global variables • Integer type global variables • Double precision type global variables • Real number type global variables • Position type global variables (Robot axes, base axes, station axes)
	Save	

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.

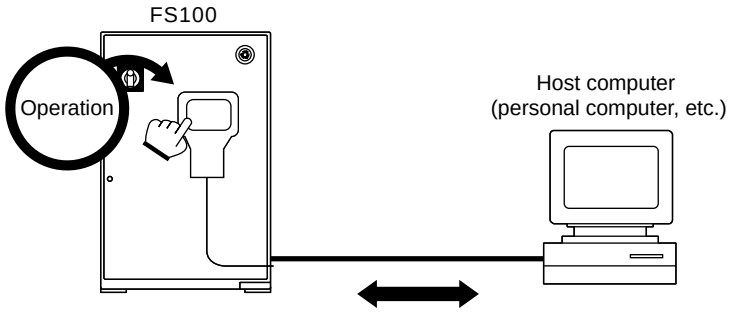


Table 1-2: Stand-alone Function

Job Transmission	**Load**	Job can be transmitted in either mode. • Single job • Related job
	Save	
	Verify	
Condition Data/ General Data Transmission	**Load**	• Tool data • User coordinate data • Variable data
	Save	
	Verify	
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.

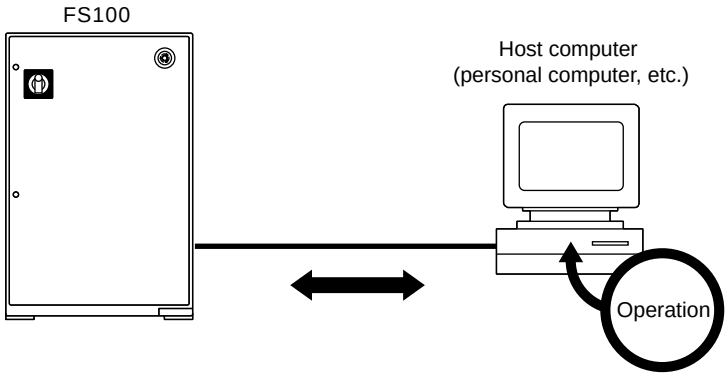


Table 1-3: Host Control Function

File Data Transmission Function	**Job Transmission**	**Load**	Jobs can be transmitted in either mode : • Single job • Related job
		Save	
	Condition Data/ General Data Transmission	**Load**	• Tool data • User coordinate data • Variable data
		Save	
	System Information Transmission	**Save**	• System information • Alarm history
Robot Control Function	**Status Reading**	• Read of error and alarm codes • Read of current position in a joint coordinate system • Read of current position in a specified Cartesian coordinate system • Read of mode, cycle, motion, alarm error and servo status • Read of current job name, line No. and step No. • Read of all job names or related job names • Monitoring completion of manipulator operation • Read of specified user coordinate data • Read of control group and task selected status • Read of variable data	
	System Control	• Start, hold • Reset, cancel • Job deletion • Master job setup • Job, line No. and step No. setup • Mode and cycle selection • Servo power supply ON/OFF • Programming pendant interlock setup/ release • Message display • Joint motion and linear motion to a specified Cartesian coordinate system • Linear motion by increments in a specified coordinate system • Joint motion and linear motion to a specified joint coordinate system • Conversion/reverse conversion of related job of a specified job (Relative job function is necessary) • Write of specified user coordinate data • Change of control group • Change of task to be controlled • Write of variable data	

2 For Using Data Transmission Function

2.1 Remote Mode

The data transmission function can be used with FS100 in remote mode.

2.1.1 Remote Mode

To use the data transmission function, set FS100 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 is used for operating the system.

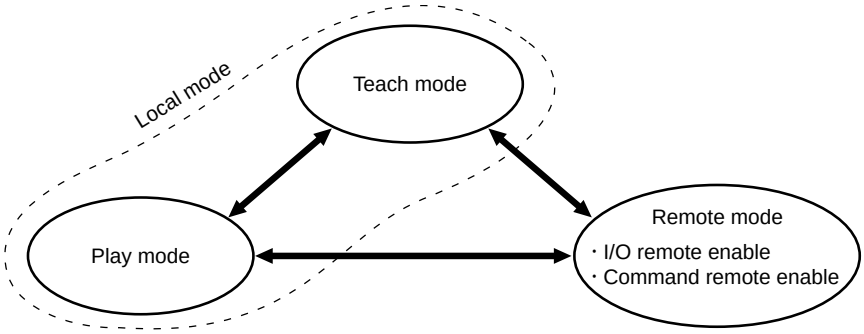
To switch to the remote mode or the local mode, either

1. Set the mode key on the programming pendant to [REMOTE].

– 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.

SUPPLE -MENT For details, refer to *chapter 7 “Remote Function Setting”* .



Operation-site Mode		Operation-site	Condition to Enable the Operation
Local Mode		Programming pendant	The remote lamp is OFF, or “INHIBIT PP/ PANEL” in the pseudo input display is set to invalid.
Remote Mode	I/O remote enable	External I/O control board	The remote lamp is ON, and “INHBIT IO” in the pseudo input display is set invalid.
	Command remote enable	External computer	The remote lamp is ON, and “CMD REMOTE SEL” in the pseudo input display is set valid.

NOTE

- In remote mode, usually operations of the programming pendant is disabled, but they can be also enabled.
- To enable all operations, refer to *chapter 7 “Remote Function Setting”* .
- To selectively enable some of the operations, set the parameter S2C230. For details, refer to *chapter 6.2 “Parameter List” at page 6-2*.

In remote mode, operations on the programming pendant 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 cannot 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 are valid only hold and emergency stop.

Also 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/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 *chapter 5.2.2 "List of Interlock for Commands of Host Control Function" at page 5-7*.

Command Remote Setting	Function Availability
Invalid	DCI function available Stand-alone function available Host control function (only read-only function) available
Valid	Host control function (all commands) available

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.

Parameter	Contents and Set Value	Initial Value
RS005	BSC port function specification when the command remote is invalidated 0 : DCI or stand-alone function 1 : Read-only function in host control	0

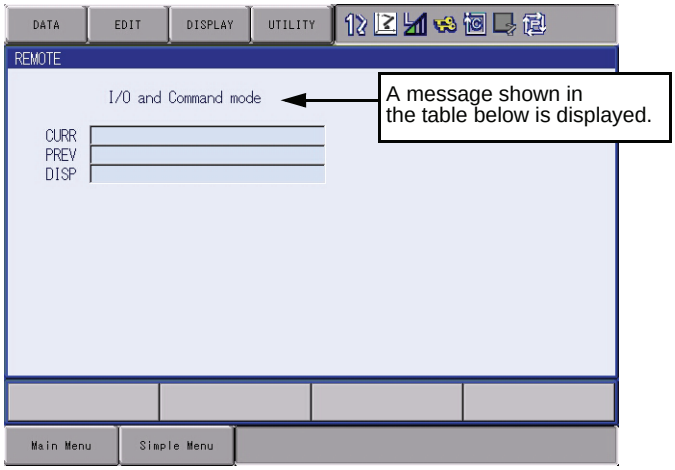
2.1.3 Display in Command Remote Mode

Even in command remote enabled submode, it is not necessary to call the command remote display because operations from FS100 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 7 “Remote Function Setting”* .)



Remote Select Status		Message	Remarks
I/O Remote	**Command Remote**		
×	×	“Remote mode not specified”	Same when the remote lamp is OFF.
○	×	“I/O mode”	Only when the remote lamp is ON.
×	○	“Command mode”	
○	○	“I/O and Command mode”	
Read-only Function Valid		“Remote mode not specified”	“CURR” and “PREV” are displayed.

○ : Valid, × : Invalid

2.2 Serial I/F Port Assignment

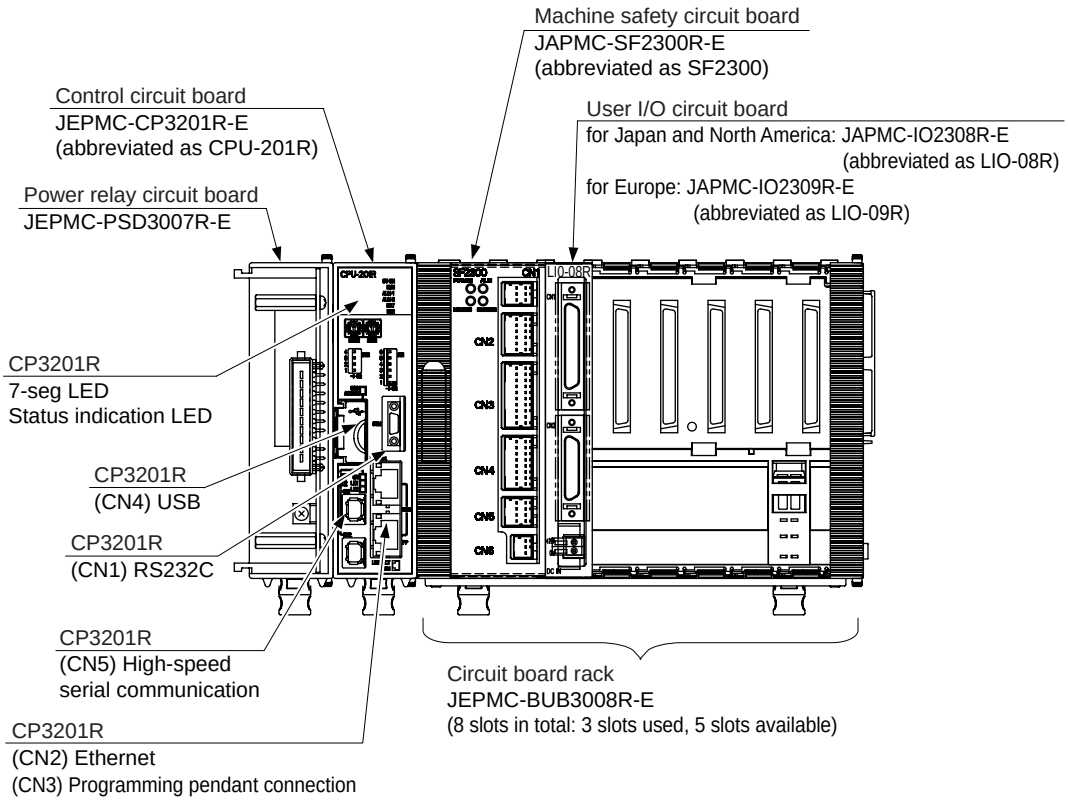
The FS100 has one serial interface port (RS-232CI/F).

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.

Parameter	Contents and Set Value	Initial Value
RS000	Standard port protocol specification 0 : NON 1 : System reserved 2 : BSC LIKE (Data transmission function) 3 : FC1	2

Fig. 2-1: CPU Unit Configuration



2.3 Parallel Operation of FS100

The FS100 is capable of parallel processing.

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

Operation	Warning
Stand-alone Programming pendant	Error message for 5 seconds
DCI	Alarm
Host control	Interpreter message (or error message)

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 cannot 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 Transmission Specifications

This section explains the transmission specifications for the data transmission.

2.4.1 Basic Specifications

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

1 Can be changed by transmission parameter setting

2.4.2 Transmission Control Characters

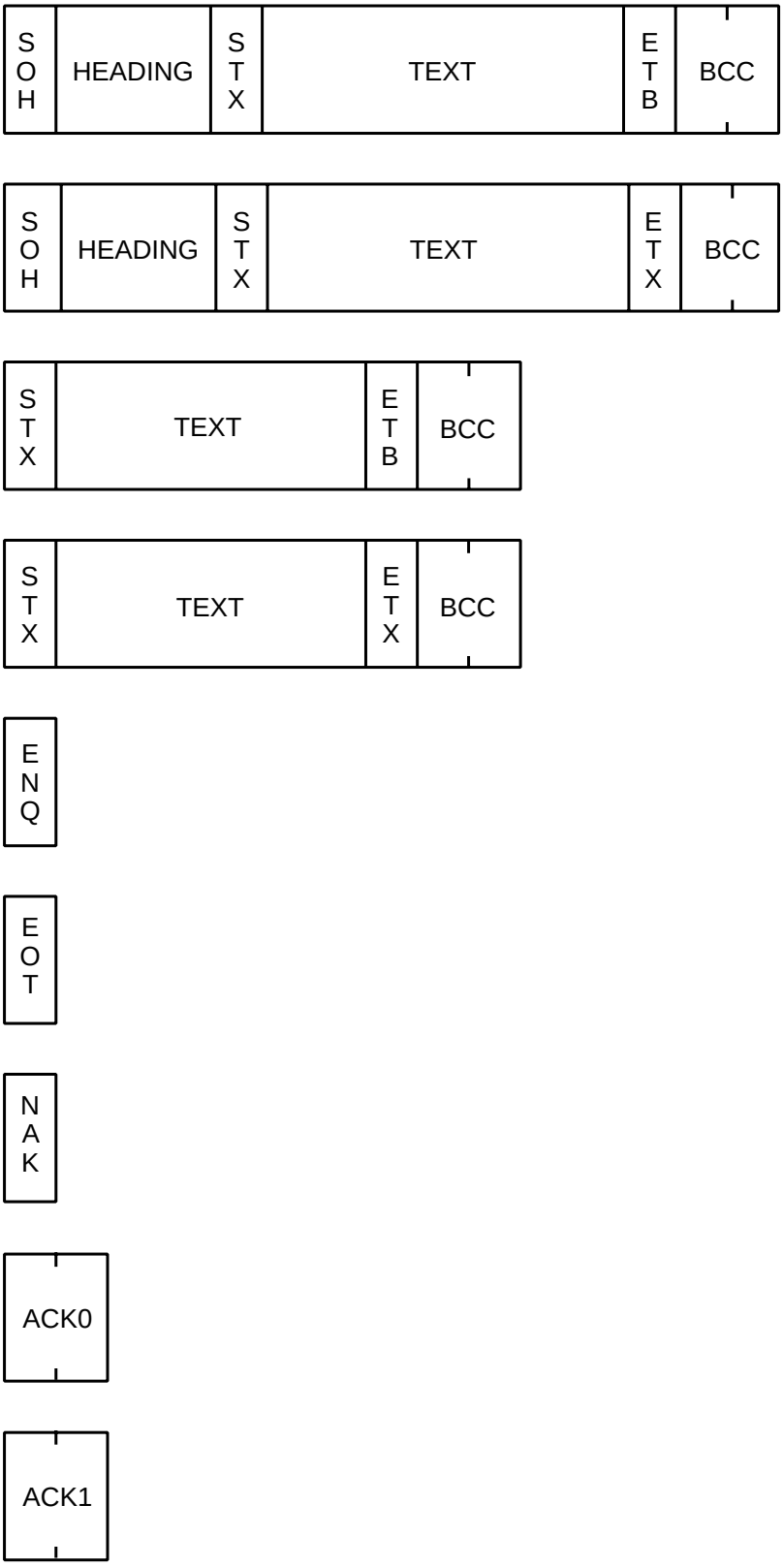
The transmission control characters are shown in the table below.

Table 2-1: Transmission Control Characters and Codes

Control Character	Code (hexadecimal)	Meanings of Control Character
DLE	10	Data Link Escape
SOH	01	Start of Heading
STX	02	Start of Text
ETX	03	End of Text
EOT	04	End of Transmission
ENQ	05	Enquiry
NAK	15	Negative Acknowledgment
ETB	17	End of Text Block
ACK0	10, 30	Even Affirmative Acknowledgment
ACK1	10, 31	Odd Affirmative Acknowledgment

2.4.3 Transmission Format

The transmission format is as follows.

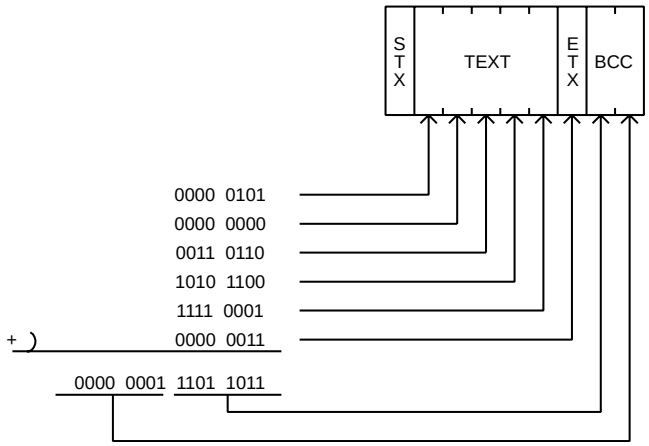


2.4.4 Error Control System

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.

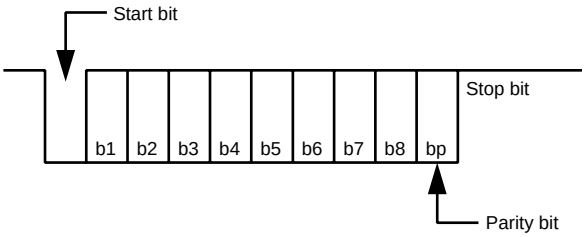
<Example>



- 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.

2.4.5 Character Configuration

The character configuration is as follows.

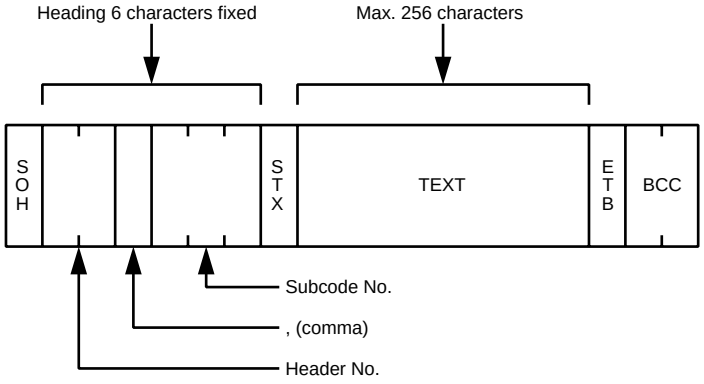


2.4.6 Data Link Establishment

A data link is established by responding ACK0 to ENQ.

2.4.7 Configuration of Heading and Text

The configuration of heading and text is as follows.



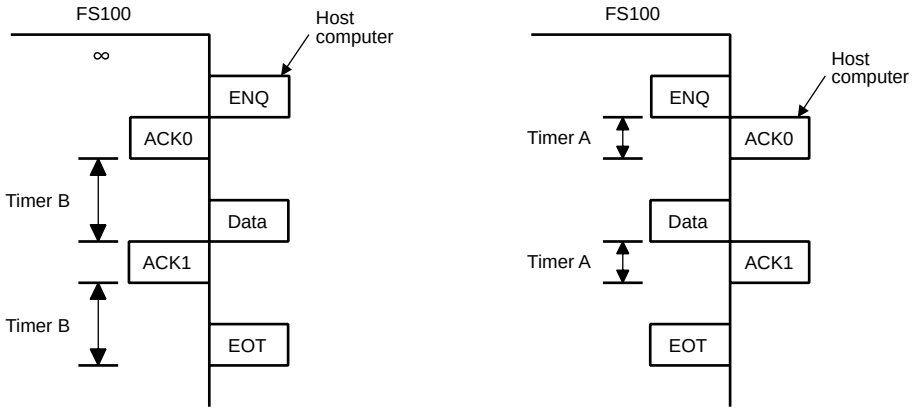
2.4.8 Transmission Parameters

2.4.8.1 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.



2.4.8.2 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.

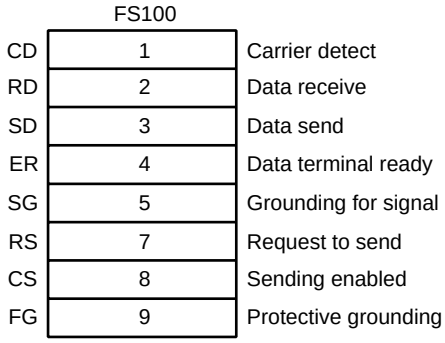
- Retry 1 : Number of resendings of a sequence character at an invalid or no response at all.
  Recommended value is 10 times.
- Retry 2 : Number of resendings of a text at a block check error (reception of NAK).
  Recommended value is 3 times.

Parameter	Contents and Set Value	Initial Value
RS030	Number of data bits 7 : 7 (bit) 8 : 8 (bit)	8
RS031	Number of stop bits 0 : 1 (bit) 1 : 1.5 2 : 2	0
RS032	Parity specification 0 : No specification 1 : Odd parity 2 : Even parity	2
RS033	Transmission speed specification 1 : 150 (baud rate) 2 : 300 3 : 600 4 : 1200 5 : 2400 6 : 4800 7 : 9600 8 : 19200	7
RS034	Timer A Sequence monitoring timer Serves as protection against invalid or no response Unit : 0.1 sec. (Setting range : 0 to 100)	30
RS035	Timer B Text reception monitoring timer Serves as protection against no response of text end character Unit : 0.1 sec. (Setting range : 0 to 255)	200
RS036	Retry 1 Number of resendings of a sequence character at an invalid or no response (Setting range : 0 to 30)	10
RS037	Retry 2 Number of resendings of a text at a block check error (reception of NAK). (Setting range : 0 to 10)	3
RS038	Block check method 0 : Check sum	0

2.4.9 Connection of D-SUB Connector Pins

The connection of D-SUB connector pins is shown below.

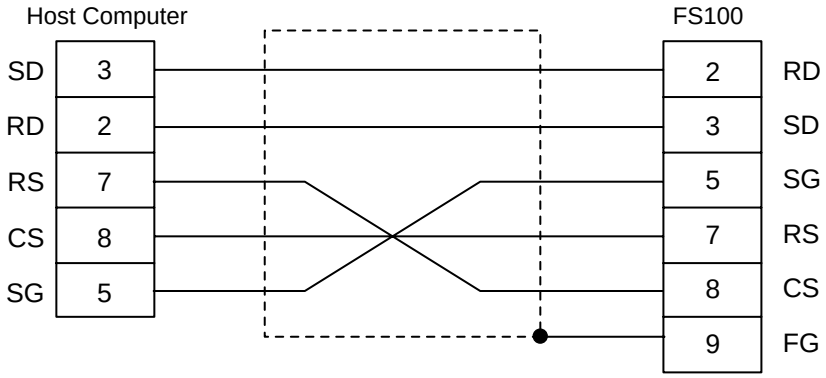
Fig. 2-2: YCP01 board (D-SUB9P)



2.4.10 Connection

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

Fig. 2-3: Board



- Connect “RS” of the FS100 to “CS” of a host computer.
  This prevents data overrun when reception processing speed of the FS100 cannot catch up with data sending from the host computer. In other words, “RS” signal from the FS100 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 FS100 sends data when the “CS” signal is ON.

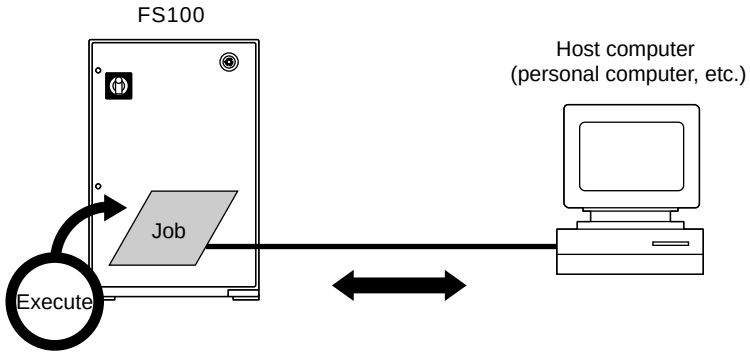
3 DCI Function

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



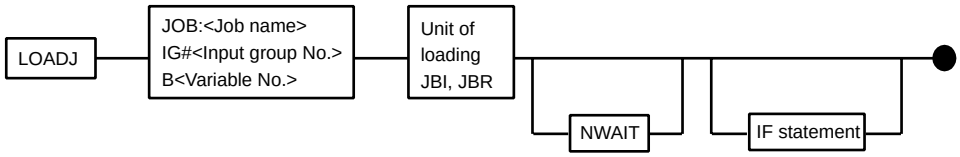
3.2 Commands for Job Transmission

3.2.1 LOADJ

3.2.1.1 Function

Loads specified jobs as single or related jobs, from the external memory unit to the memory of the FS100.

3.2.1.2 Configuration



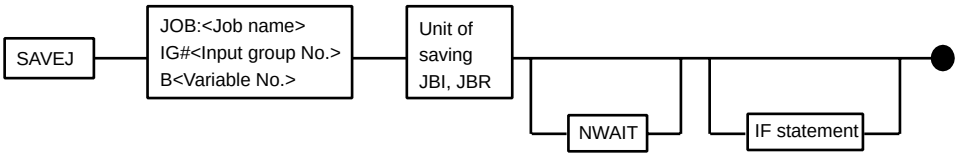
- If the FS100 memory already contains a job having the same name as the job to be loaded, the existing job is deleted and the new job is loaded.
  However, if the job to be loaded is as follows, an alarm occurs.
  - Execution starting job
  - Job under execution/halting
  - Job registered in job call 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.

3.2.2 SAVEJ

3.2.2.1 Function

Saves a specified job as single or related jobs, from the memory of the FS100 to the external memory unit.

3.2.2.2 Configuration



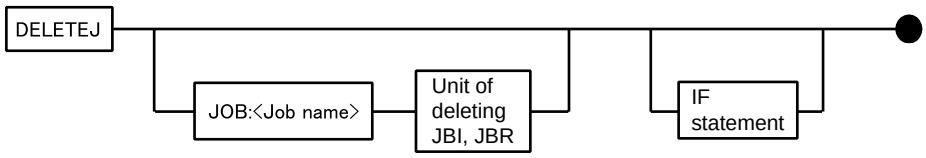
- Specify input group numbers (BCD/BIN, parity specification), and variable numbers in the same way as for the CALL command.
  If the pattern input value is 0, the operation is not executed.
  A variable number 0 is valid.
- Unit of saving : Select either a single job (JBI) or related jobs (JBR).
- When 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

3.2.3.1 Function

Deletes all jobs except its own job or specified jobs as single or related jobs, from the memory of the FS100.

3.2.3.2 Configuration



- Unit of deleting : Select either a single job (JBI) or related jobs (JBR).
- The following jobs cannot be deleted.
  - Execution starting job
  - Job under execution/halting
  - Job registered in job call stack

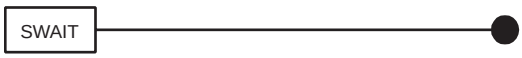
3.2.4 SWAIT

3.2.4.1 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.

3.2.4.2 Configuration



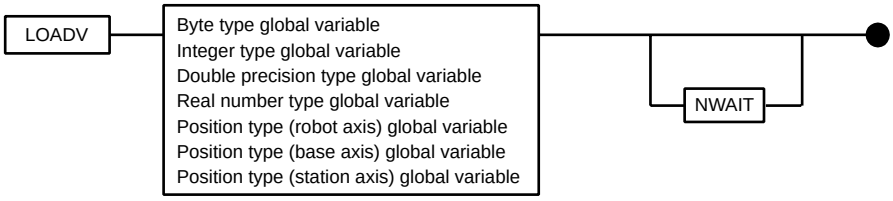
3.3 Commands for Variable Transmission

3.3.1 LOADV

3.3.1.1 Function

Loads the specified global variables from an external memory unit to the FS100 memory.

3.3.1.2 Configuration

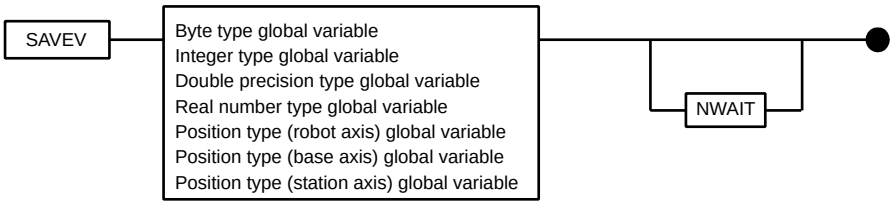


3.3.2 SAVEV

3.3.2.1 Function

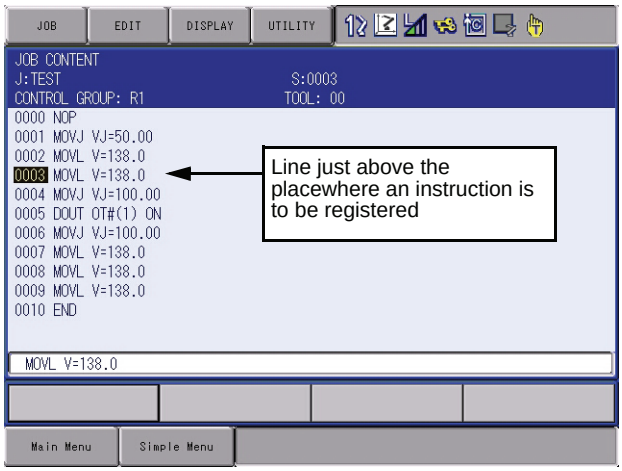
Saves the specified global variables from the FS100 memory to a external memory unit.

3.3.2.2 Configuration



3.4 Registering DCI Instruction

1. Move the cursor to the address area.
2. Move the cursor to the line where an instruction is to be registered in the job content display.
   – 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.



3. Press [INFORM LIST].
4. Select an instruction to be registered.
   – The instruction list dialog is displayed.

– The cursor moves to the instruction list dialog while the cursor in the address area changed to an underline.
The instruction where the cursor is positioned is displayed with the previously registered additional items in the input buffer line.

5. Change the additional items and variable data.

– <To register items as displayed in the input buffer>

(1) Perform operation described in the step 6 below.

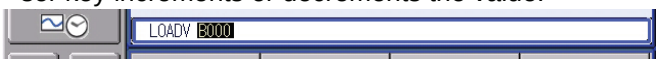
– <To edit any additional items>

(1) 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

I) 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.

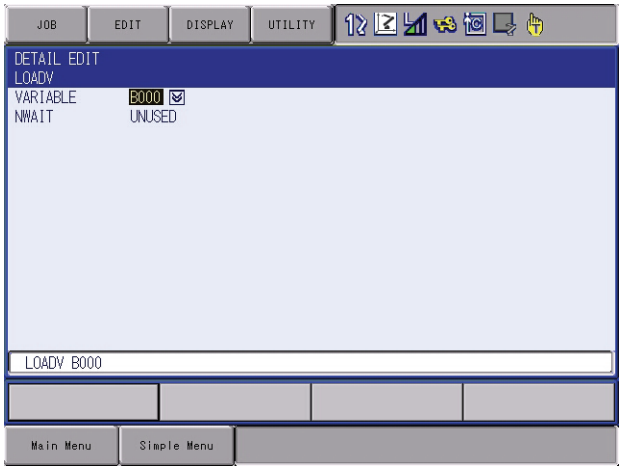


II) To enter a value by pressing the number key, press [SELECT] to display the input line.

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.



• Adding the additional item

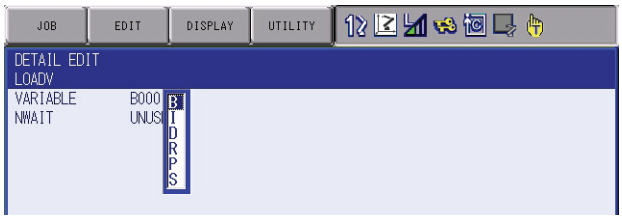
I) Select “NOT USED” of an additional item selection status, then display the selection dialog.

II) 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

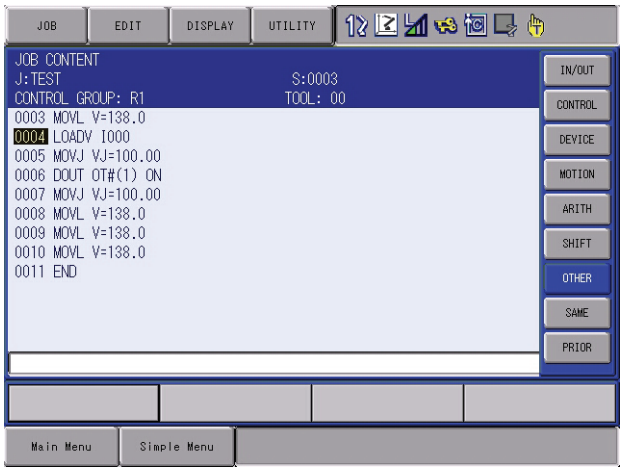
(1) To change the data type of additional item, move the cursor to the ☒ of the additional item and press [SELECT] to select a data type.



(2) After having added, changed or deleted the addtional items, press [ENTER].

– The detail edit display is ended and the job content display appears.

6. Press [INSERT] and [ENTER].
   – 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) connected to the FS100.
- 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.

3.6.1 Parallel Execution Using NWAIT

```
NOP
MOVJ  VJ=50.00
MOVJ  VJ=50.00
LOADJ  JOB:ABC  JBI  NWAIT · · · ①
MOVJ  VJ=50.00 · · · · · · · · · · · · · ②
MOVJ  VJ=50.00 · · · · · · · · · · · · · ③
SWAIT · · · · · · · · · · · · · · · · · · ④
CALL  JOB:ABC · · · · · · · · · · · · · · ⑤
· · ·
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 ⑤ 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.

(Correct)

```
NOP
· · ·
LOADV  B000  NWAIT
· · ·
SWAIT
SET  B001  B000
```

(Wrong)

```
NOP
· · ·
LOADV  B000  NWAIT
· · ·
SET  B001  B000
```

3.6.2 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
LOADJ 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 FS100 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

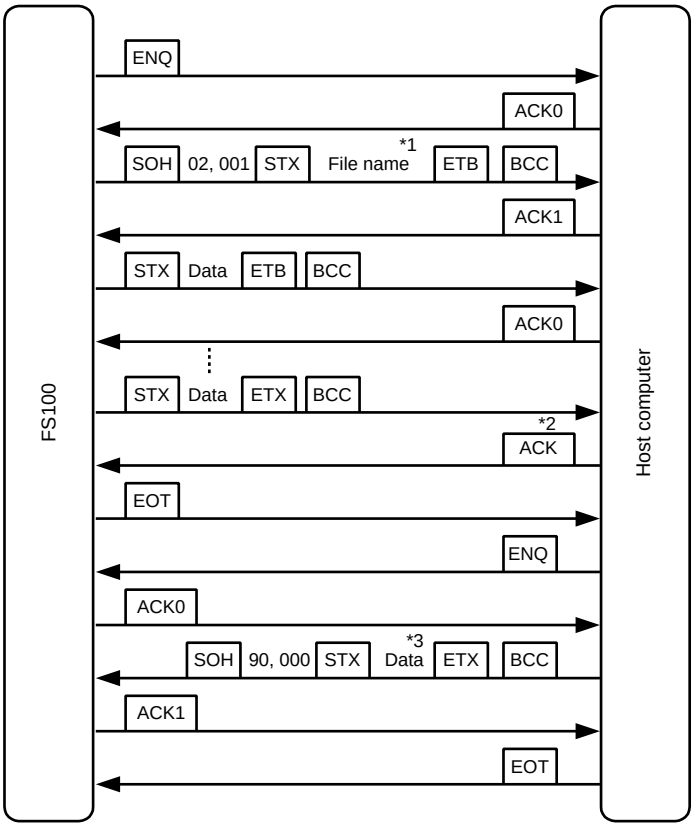
3.7.1.1 Saving Procedure

The transmission from the FS100 to a host computer proceeds as follows.

FS100 → 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 FS100 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.

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)

3.7.1.2 Loading Procedure

The transmission from a host computer to the FS100 proceeds as follows.

Host computer → FS100

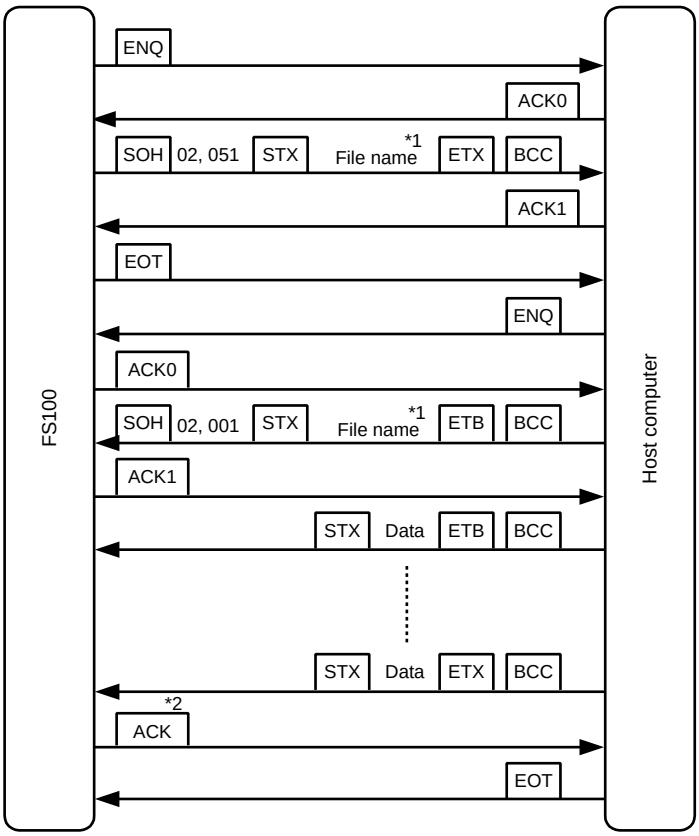
1. The ENQ code is sent out to establish a data link.
2. After the data link is established, a request to send is sent out to the host computer.
3. When the request to send is accepted, the FS100 enters receiving status, waiting for the ENQ code from the host computer. Therefore, the host computer should 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 cannot 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.

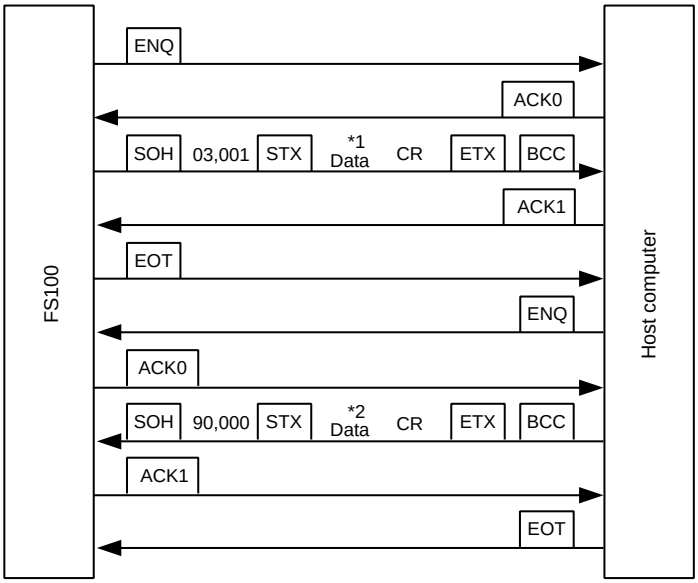


1 File name : CR (File name does not include extension.)
2 ACK0 or ACK1

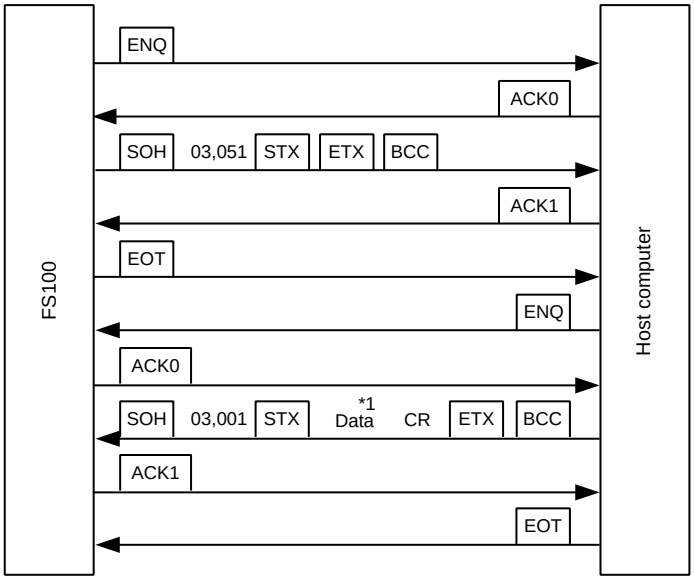
3.7.2 Variable Transmission

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

3.7.2.1 Saving Procedure



3.7.2.2 Loading Procedure



For headers, refer to the header number list.

1

Byte type global variable :	□□□	(0 to 255)
Integer type global variable :	± □□□□□	(-32768 to +32767)
Double precision type global variable :	± □□□□□□□□□□	(-2147483648 to 2137383647)
Real number type global variable :	7 significant digits	(-1.70141E+38 to +1.70141E+38)
Position type (robot axis) global variable :	• Pulse type or XYZ type depending on the setting status • The order varies depending on the number of robot's axes.	

Pulse type
① 6-axis robot
S, L, U, R, B, T (Unit : pulse)
(-999999999 to 999999999)
② 7-axis robot
S, L, U, R, B, T, E (Unit : pulse)
(-999999999 to 999999999)

XYZ type
① 6-axis robot
X, Y, Z, Rx, Ry, Rz, TYPE

d0 = 0 : Flip d0 = 1 : No flip
d1 = 0 : Up d1 = 1 : Back

Unit : degree (°), significant 4 decimal points
-9999.9999 to 9999.9999

Unit : mm, significant 3 decimal points
-999999.999 to 999999.999

② 7-axis robot
X, Y, Z, Rx, Ry, Rz, Re, TYPE

d0 = 0 : Flip d0 = 1 : No flip
d1 = 0 : Up d1 = 1 : Back

Unit : degree (°), significant 4 decimal points
-9999.9999 to 9999.9999

Unit : mm, significant 3 decimal points
-999999.999 to 999999.999

Position type (base axis) global variable :	Pulse type or XYZ type depending on the internal setting status

Pulse type
1, 2, 3 (Unit : pulse)
(-999999999 to 999999999)
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) (-999999999 to 999999999)

String type global variable : String (16 halfwidth characters)

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 FS100 data transmission function has the following restrictions on transmission of the FS100 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
4104	WRONG EXECUTION OF LOAD INST	Refer to the table below
4105	WRONG EXECUTION OF SAVE INST	
4106	WRONG EXECUTION OF DELETE INST	

Data	Contents
001	Insufficient memory capacity
002	Job editing prohibited
003	Attempted to load or delete a job being executed.
004	No specified job
012	Position data destroyed
013	Position variable not registered
017	Instruction destroyed
019	Invalid character in job name
020	Invalid character in label
023	Invalid character in this system
024	Syntax error
090	Control command sending/receiving error (Ethernet)
104	Error response from host computer
111	Syntax error
112	Error in position data
113	No NOP or END instruction
117	Format error
118	Invalid number of data
120	Data range exceeded
122	Destroyed file exists
125	No serial port setting
126	This serial port already used.
127	This protocol already used.
128	File accessing in other function
211	System block error (Receiving EOT while waiting ACK)
212	System block error (Receiving EOT at starting receiving)
213	System block error (Receiving EOT before receiving the last block)
214	System block error (Receiving codes other than EOT before receiving the last block)
221	Sending error (Retry for NAK exceeded)
222	Sending error (Timeup for timer A retry)
223	Sending error (ACK0/ACK1 order error after retry)
231	Receiving error (Timeup for timer A while waiting ACK after ENQ, timeup for timer A while waiting ENQ response)
232	Receiving error (Timeup for timer B while receiving a text)
233	Receiving error (Heading length is shorter than 6 characters)
234	Receiving error (Heading length is longer than 6 characters)

Data	Contents
235	Receiving error (Header number error)
236	Receiving error (Text length exceeds 256 bytes)
237	Receiving error (Receiving other than ENQ while waiting ENQ, receiving other than control code while waiting control code, receiving other than STX, SOH, ENQ, EOT while waiting text)
240	Software error
241	Hardware error (Overrun)
242	Hardware error (Parity error)
243	Hardware error (Framing error)
244	Hardware error (Sending timeup (timer A))
245	Hardware error (Sending timeup (timer B))

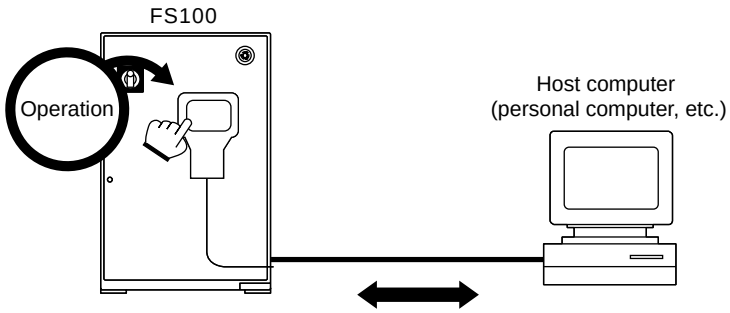
4 Stand-alone Function

4.1 Outline

In stand-alone mode, the file data transmission function is available.

By the operations on the FS100 programming pendant, file data can be sent from the FS100 to a host computer such as personal computer to be saved, and from a host computer to the FS100 memory to be loaded.

Load : Transmits file data from a host computer to the FS100.
Save : Transmits file data from the FS100 to a host computer.
Verify : Verifies data between the FS100 and the host computer and informs if some parts are not matched.



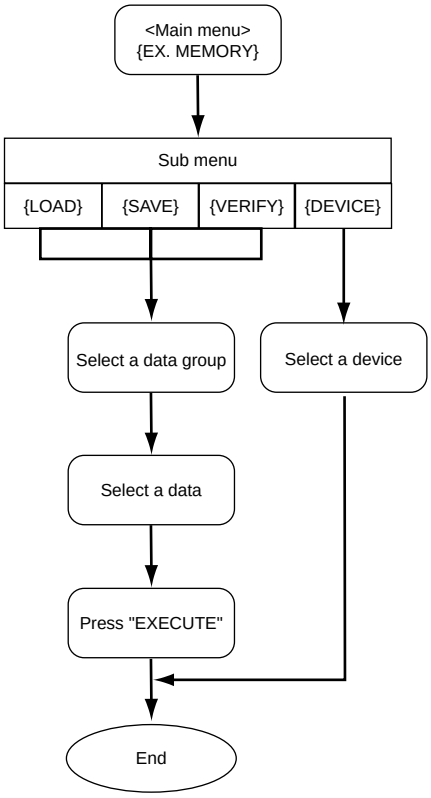
The following data can be transmitted between the FS100 and a host computer.

System information can be saved but not loaded.

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

4.2 Operation Flow

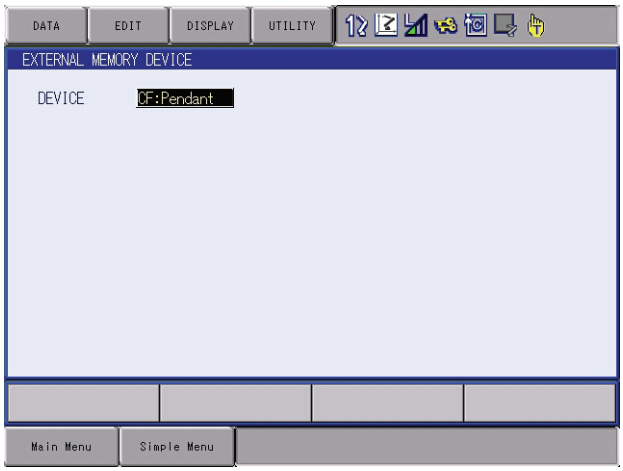
Transmission of file data is performed in the following manner.



4.3 Operation

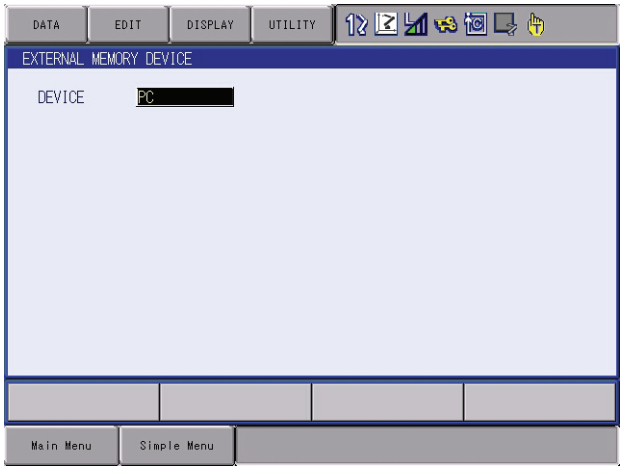
4.3.1 Selecting External Memory Unit

1. Select {EX. MEMORY} under the main menu.
2. Select {DEVICE}.
   – The device selection display is shown.



3. Select “DEVICE”.
   – The selection dialog is shown.

4. Select the device to be changed.
   – The device is changed.

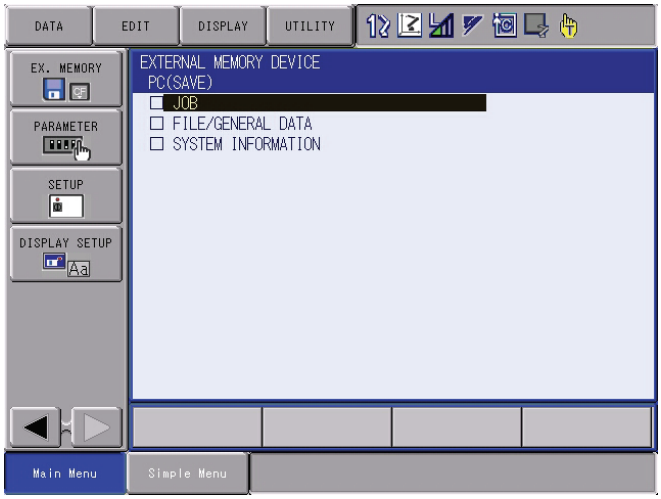


4.3.2 Save

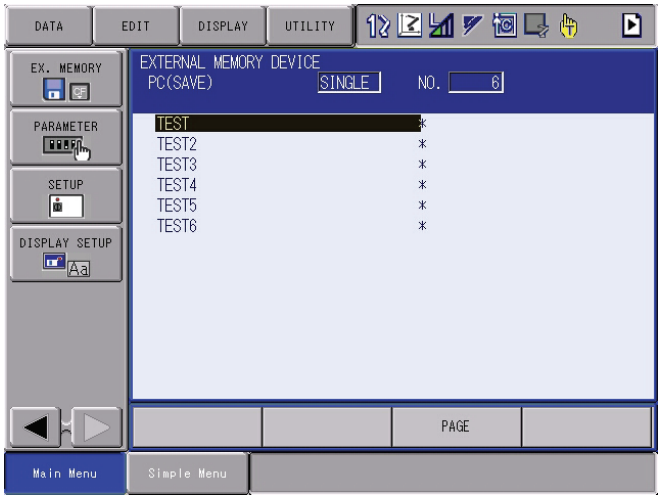
The operation to transmit data from the FS100 to the external memory unit is explained in the following.

4.3.2.1 Saving Job

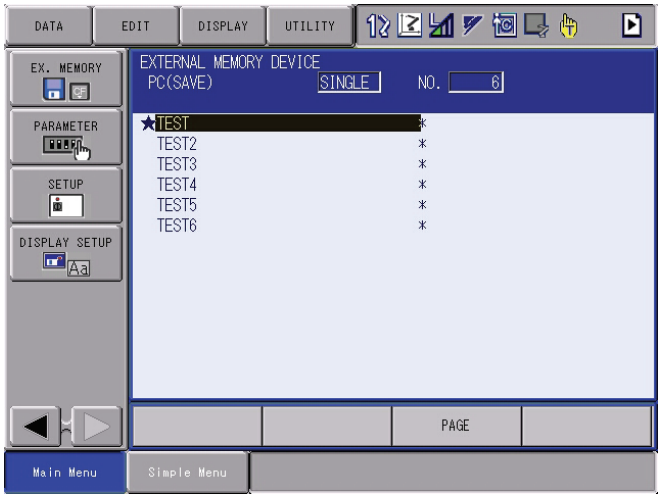
1. Select {EX. MEMORY} under the main menu.
2. Select {SAVE}.
   – The external memory menu display is shown.



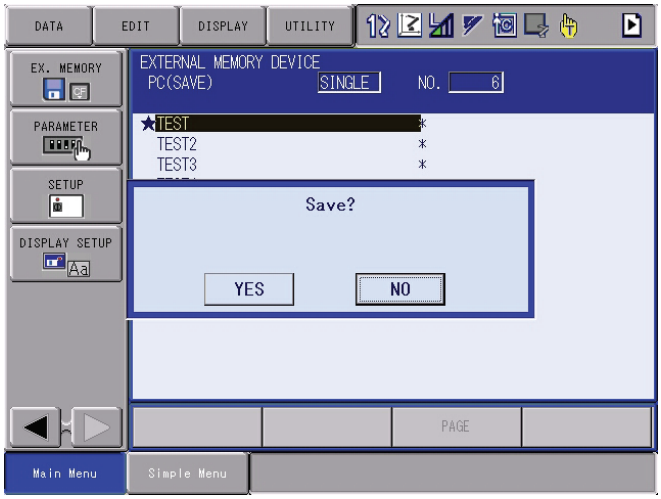
3. Select “JOB”.
   – The external memory job list display is shown.



4. Select the job to be saved.
   – The select job is marked with "★".



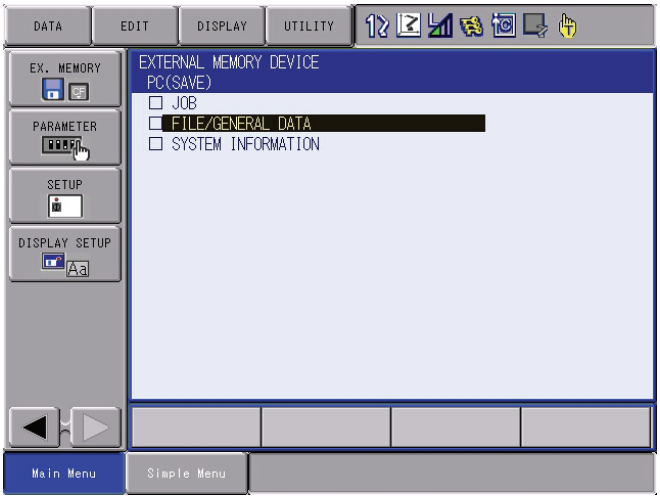
5. Press [ENTER].
   – The confirmation dialog is shown.



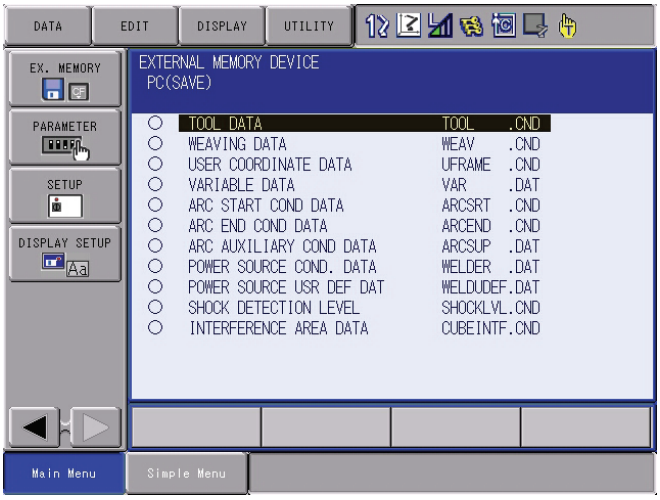
6. Select "YES".
   – The job starts to be saved, and the transmission display is shown.
   – To interrupt the saving, press [SELECT].
   When the saving is completed or interrupted, the job content display appears.

4.3.2.2 Saving File

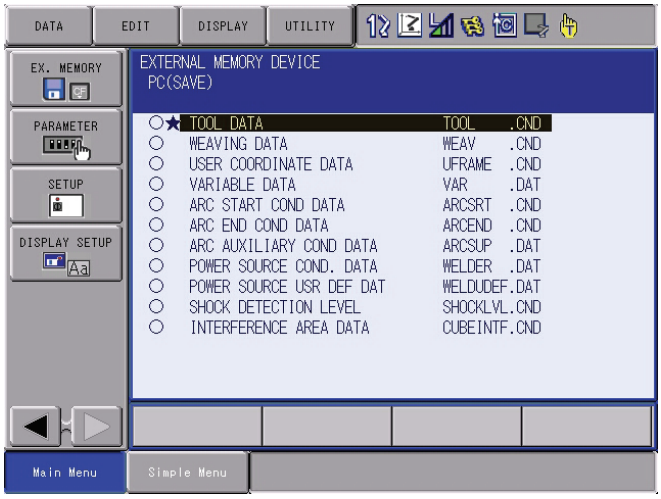
1. Select {EX. MEMORY} under the main menu.
2. Select {SAVE}.
   – The external memory menu display is shown.



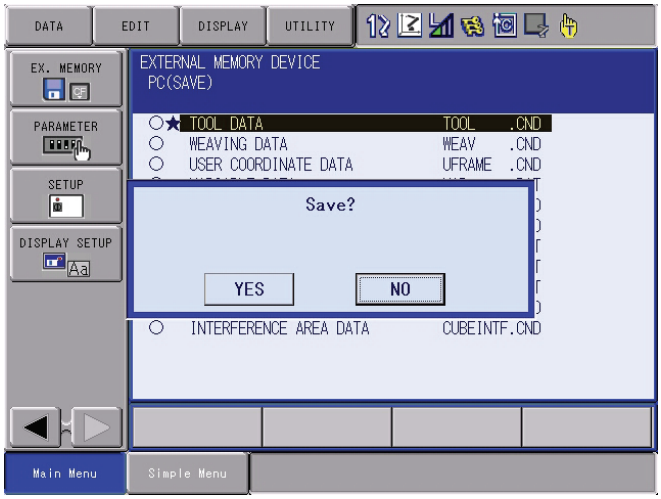
3. Select the file group to be saved.
   – The file selection display is shown.



4. Select the file to be saved.
   – The select file is marked with “★”.



5. Press [ENTER].
   – The confirmation dialog is shown.



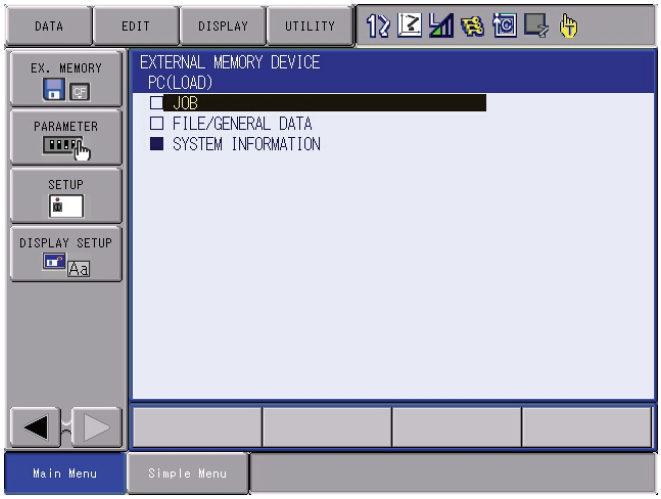
6. Select “YES”.
   – The file starts to be saved, and the transmission display is shown.
   – To interrupt the saving, press [SELECT].
   When the saving is completed or interrupted, the file selection display reappears.

4.3.3 Load

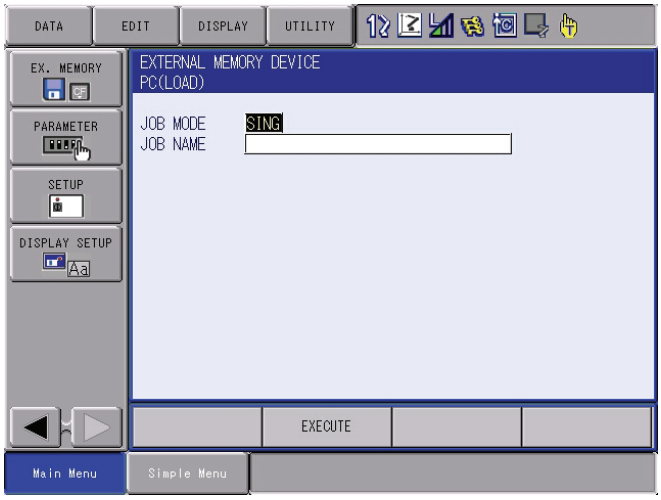
The operation to transmit data from the external memory unit to the FS100 is explained in the following.

4.3.3.1 Loading Job

1. Select {EX. MEMORY} under the main menu.
2. Select {LOAD}.
   – The external memory menu display is shown.



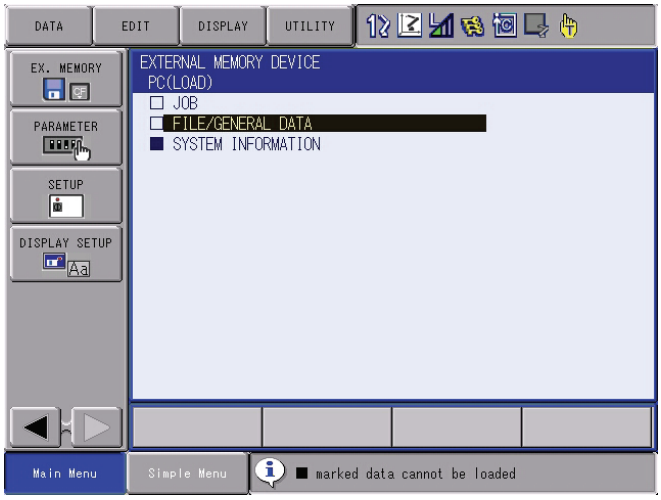
3. Select “JOB”.
   – The display to input the job name to be loaded is shown.



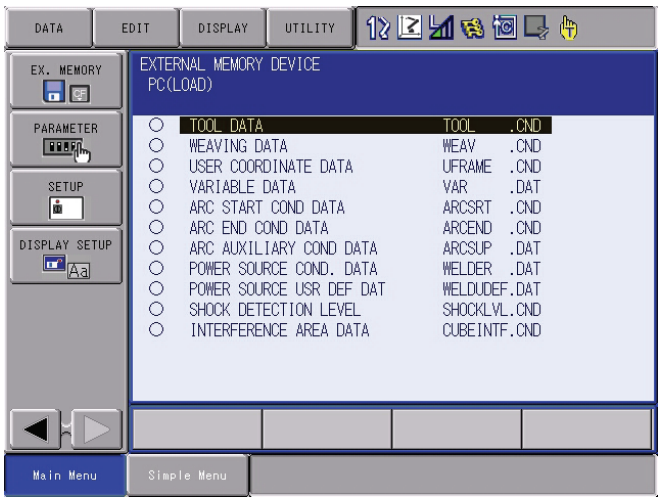
4. Enter the job to be loaded.
5. Select “EXEC”.

4.3.3.2 Loading File

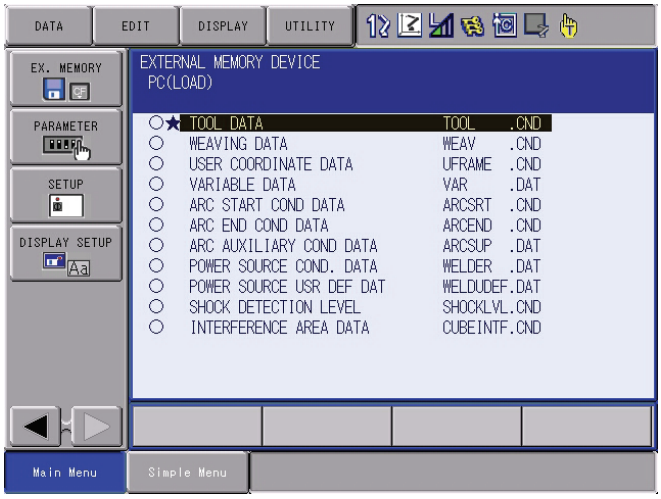
1. Select {EX. MEMORY} under the main menu.
2. Select {LOAD}.
   – The external memory menu display is shown.



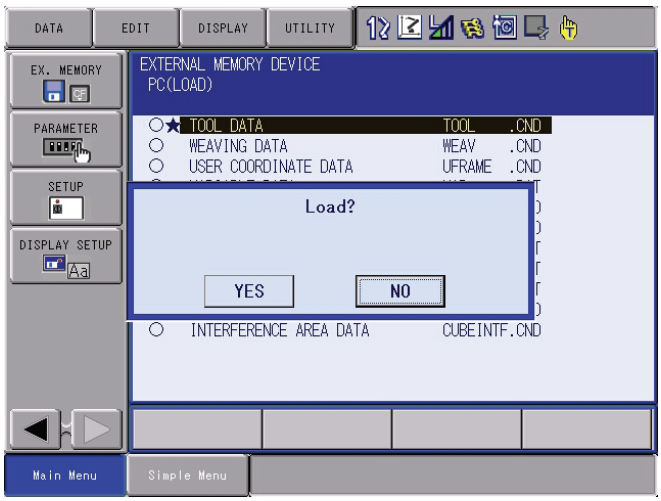
3. Select the file group to be loaded.
   – The file selection display is shown.



4. Select the file to be loaded.
   – The selected file is marked with "★".



5. Press [ENTER].
   – The confirmation dialog is shown.



6. Select "YES".
   – The file is 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

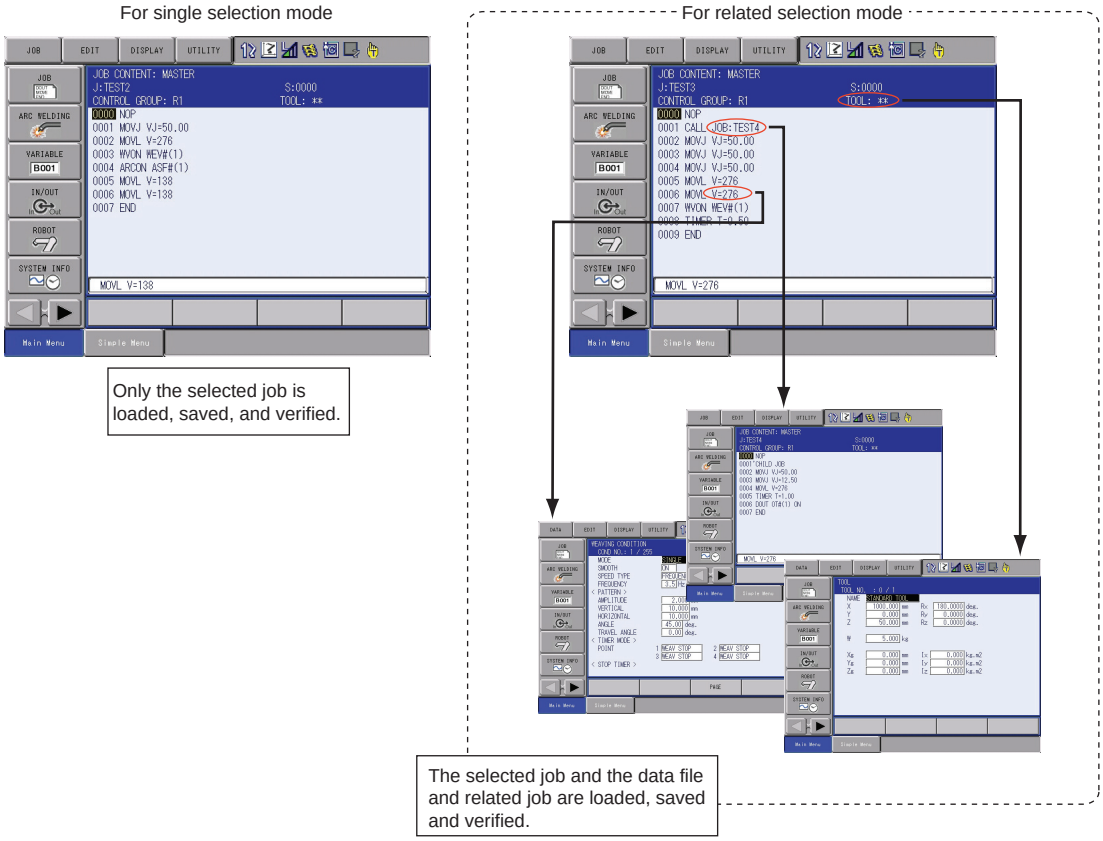
To select a job to save, load, or verify, the following selection modes are available.

4.3.4.1 Single Selection Mode

Only the selected job is loaded, saved, or verified.

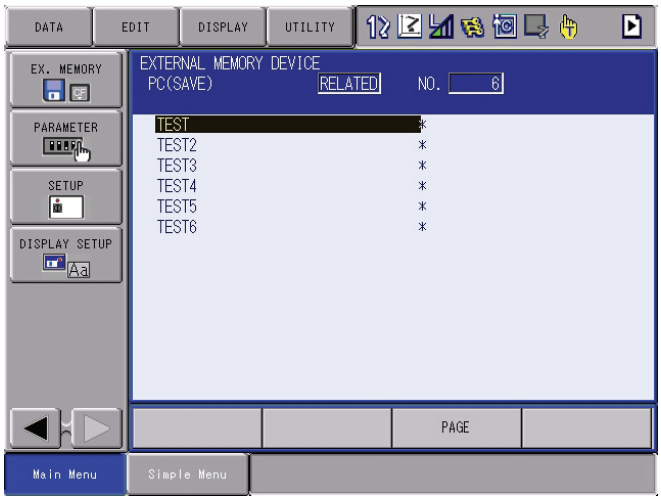
4.3.4.2 Related Selection Mode

The selected job and the related jobs and data files are loaded, saved, or verified.



4.3.4.3 Switching Selection Mode

1. Press the page key in the external memory job list display.
   – Each time the page key  is pressed, the displays in “single selection mode” and in “related selection mode” appears alternately.



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.

4.3.5.1 EACH Selection

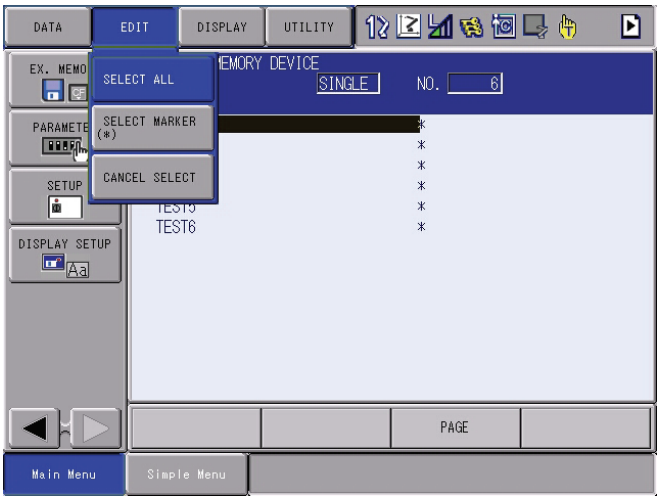
Selects job and data file one by one.

4.3.5.2 BATCH Selection

Selects all the jobs and data files at once.

For BATCH selection, proceeds the following operation.

1. Select {EDIT} of the menu in the external memory job list display or the file selection display.
   – The pull down menu is displayed.



2. Select {SELECT ALL}.

4.4 Transmission Procedure

The transmission procedure is the same as for DCI function.

Refer to *chapter 3.7 “Transmission Procedure” at page 3-13*.

5 Host Control Function of FS100

The FS100 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 7 “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 *chapter 2.1.3 “Display in Command Remote Mode” at page 2-4*.

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 FS100 to the host computer or receives data from the host computer.

The following data can be transmitted between the FS100 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

5.1.1.1 Load

The transmission from a host computer to the FS100 proceeds as follows.

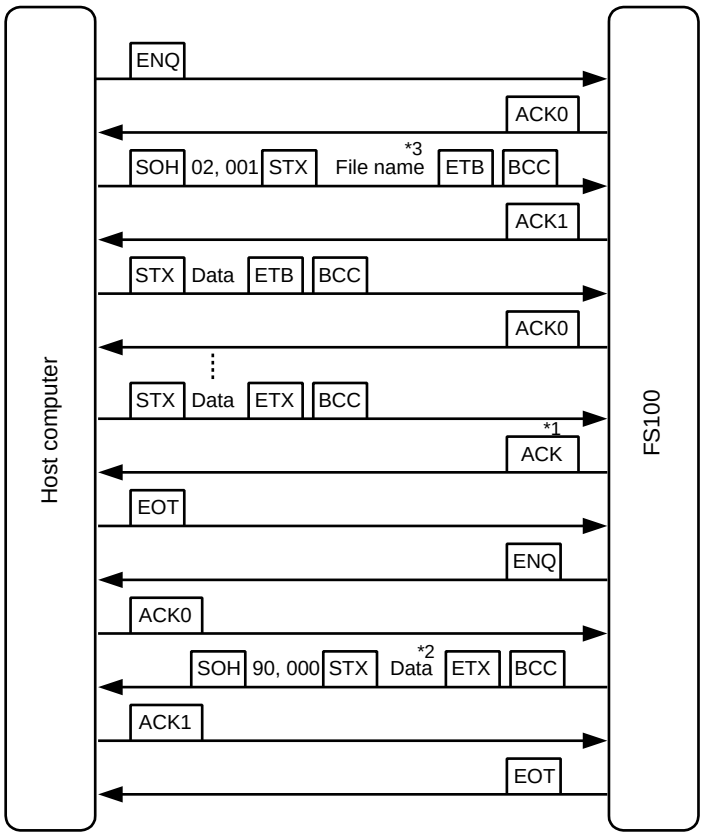
Host computer → FS100

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 FS100 to terminate the transmission.

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

Refer to the header number list.

Fig. 5-1: 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)

5.1.1.2 Save

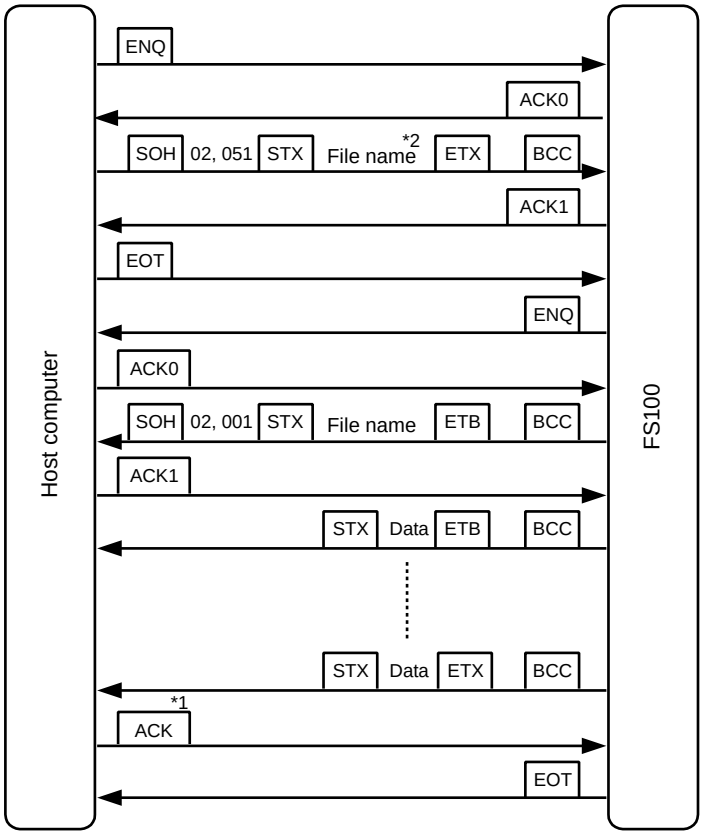
The transmission from the FS100 to a host computer proceeds as follows.

FS100 → 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 FS100 sends the ENQ code to establish a data link.
5. After the data link is established, receive the data sent from the FS100.
   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 FS100 sends the following response message instead of data.
   Check the header and take an appropriate action.

   SOH 90,000 STX DATA CR ETX BCC

Fig. 5-2: Saving File Data (Host Control Function)



1 ACK0 or ACK1
2 File name : CR (File name does not include extension)

5.1.2 Data Management

The jobs for the FS100 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.

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 FS100 cannot 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 FS100 sends the ENQ code to establish a data link.
4. After the data link is established, the FS100 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**

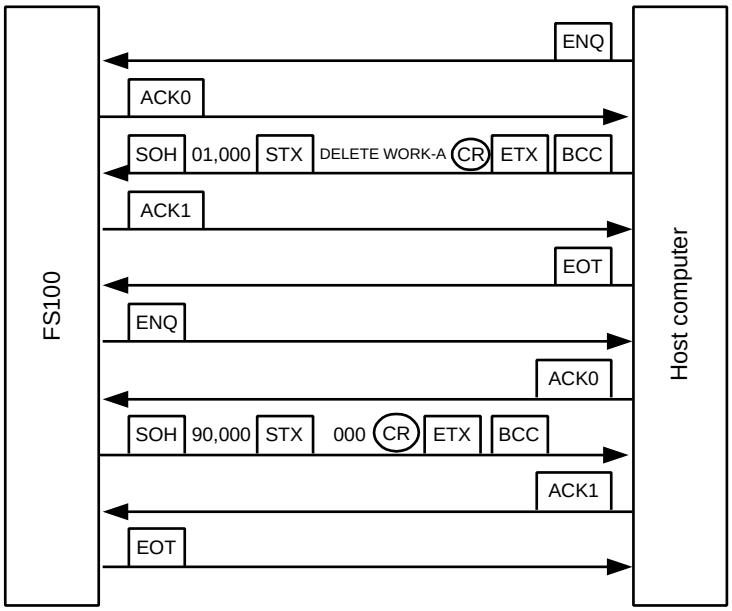
(1) [SOH] 90,000 [STX] {0000 or Error code} CR (ETX) [BCC]
0000 : Normal completion
Error code : Number with 4 digits other than 0000. In case of smaller than 1000, 0 is added before the number.

(2) [SOH] 90,001 [STX] Data1, Data2, ・・・・ DataN (CR) [ETX] [BCC]

If the FS100 cannot execute the sent command, the FS100 returns an interpreter message.

An example of DELETE command (delete a job) is shown.

Fig. 5-3: Sending Command from Host Computer



5.2.2 List of Interlock for Commands of Host Control Function

The executability of each command differs depending on the status of the FS100 as shown in the following table.

Command Name		Read/Write Enabled					Only Read Enabled	
		Non-alarm/Non-error				Alarm/ Error	Non-alarm/ Non-error	Alarm/ Error
		Teach Mode		Play Mode				
		Stop	Operating	Stop	Operating			
Read or Monitor	RALARM	○	○	○	○	○	○	○
	RPOSC	○	○	○	○	○	○	○
	RPOSJ	○	○	○	○	○	○	○
	RSTATS	○	○	○	○	○	○	○
	RJSEQ	○	○	○	○	○	○	○
	JWAIT	○	○	○	○	A	○	A
	RGROUP	○	○	○	○	○	○	○
Read or Data Access	RJDIR	○	○	○	○	○	C	C
	RUFRAME	○	○	○	○	○	C	C
	UPLOAD	○	○	○	○	○	C	C
	SAVEV	○	○	○	○	○	C	C
Operation	HOLD	○	○	○	○	○	C	C
	RESET	○	○	○	○	○	C	C
	CANCEL	○	○	○	○	○	C	C
	MODE	○	○	○	○	○/A *3	C	C
	CYCLE	○	○	○	○	○/A *3	C	C
	SVON 0 (OFF)	○	○	○	○	○	C	C
	SVON 1 (ON)	○	○	○	○	A	C	C
	HLOCK	○	○	○	○	○	C	C
	MDSP	○	○	○	○	○	C	C
	CGROUP	○	○	○	○	○	C	C
	CTASK	○	○	○	○	○	C	C
Activation	START	M	M	○/H *1	MOVE/○*2	A	C	C
	MOVJ	M	M	○/H *1	MOVE/○*2	A	C	C
	MOVL	M	M	○/H *1	MOVE/○*2	A	C	C
	IMOV	M	M	○/H *1	MOVE/○*2	A	C	C
	PMOVJ	M	M	○/H *1	MOVE/○*2	A	C	C
	PMOVL	M	M	○/H *1	MOVE/○*2	A	C	C
Editing	DELETE	○	MOVE	M	M	A	C	C
	CVTRJ	○	MOVE	M	M	A	C	C
	CVTSJ	○	MOVE	M	M	A	C	C
	WUFRAME	○	MOVE	M	M	A	C	C
	DOWNLOAD	○	○/MOVE*4	○	○/MOVE*4	A	C	C
	LOADV	○	○	○	○	A	C	C
Job selection	SETMJ	○	MOVE	○	MOVE	A	C	C
	JSEQ	○	MOVE	○	MOVE	A	C	C

	<Interpreter message>
○ : 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 "○" if not being held ; "H" if being held
2 "MOVE" if the manipulator is moving by operation other than command ; "○" if the manipulator is moving by command since a single command can be accepted.
3 "○" during an alarm ; "A" during error
4 Only a single job can be executed.

5.2.3 Command that Handle Axis Data

The data transmission function of the FS100 has restrictions on handling control axis data.

① For the following commands, the order of response data varies whether the number of robot's axes is 6 or 7.

Object commands: RPOSJ, RPOSC, MOVJ, MOVL, IMOV, PMOVJ, PMOVL

② Since the manipulator axes are fixed to a six-axis set, any manipulator having more than seven axes cannot use the following commands.

Object commands: RUFRAME, 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 applied to the following commands.

MOVJ, MOVL, IMOV, PMOVJ, PMOVL

5.2.5 Status Read Function

The details of each command are described.

5.2.5.1 Read/Monitor Command

■ **RALARM**

Reads the error alarm code.

Although the FS100 has the subcode to error code, it cannot read by RALARM because the command has no argument of the subcode.

Command format : RALARM

Response format : Data-1, Data-2, ····, Data-10 or Error code
Data-1 = Error code (0 to 9999)
Data-2 = Error data (0 to 256)
Data-3 = Alarm code (0 to 9999)
Data-4 = Alarm data (0 to 256)
Data-5 = Alarm code (0 to 9999)
Data-6 = Alarm data (0 to 256)
Data-7 = Alarm code (0 to 9999)
Data-8 = Alarm data (0 to 256)
Data-9 = Alarm code (0 to 9999)
Data-10 = Alarm data (0 to 256)

<Example>

Command RALARM
Response 0, 1234, 12, 0, 0, 0, 0, 0, 0

■ **RPOSJ**

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 E-axis (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

<Example>

Command RPOSJ
Response 500, 2600, 1250, 10789, 624, 36, 0, 0, 0, 0, 0, 0

■ **RPOSC**

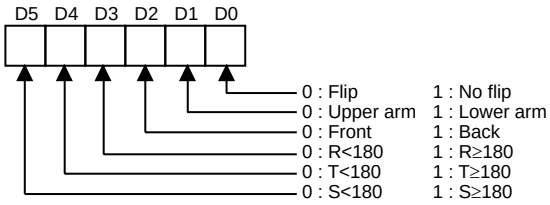
Reads the current position in a specified coordinate system. Whether there is an external axis or not can be specified.

Command format : RPOSC Data-1, Data-2
Data-1 = Specification of coordinate system
0 : Base coordinate
1 : Robot coordinate
2 : User coordinate 1
:
:
17 : User coordinate 16
Data-2 = With or Without external axis
0 : Without external axis
1 : With external axis

Response format : Data-1, Data-2, ····, Data-14
* The order of response data varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-1	X coordinate value (unit : mm, significant 3 decimal points)	X coordinate value (unit : mm, significant 3 decimal points)
Data-2	Y coordinate value (unit : mm, significant 3 decimal points)	Y coordinate value (unit : mm, significant 3 decimal points)
Data-3	Z coordinate value (unit : mm, significant 3 decimal points)	Z coordinate value (unit : mm, significant 3 decimal points)
Data-4	Wrist angle Rx (unit : degree (°), significant 4 decimal points)	Wrist angle Rx (unit : degree (°), significant 4 decimal points)
Data-5	Wrist angle Ry (unit : degree (°), significant 4 decimal points)	Wrist angle Ry (unit : degree (°), significant 4 decimal points)
Data-6	Wrist angle Rz (unit : degree (°), significant 4 decimal points)	Wrist angle Rz (unit : degree (°), significant 4 decimal points)
Data-7	Type	Elbow angle Re (unit : degree (°), significant 4 decimal points)
Data-8	Tool number (0 to 15)	Tool number (0 to 15)
Data-9	Number of 7th axis pulses (for travel axis, mm)	Number of 7th axis pulses (for travel axis, mm)
Data-10	Number of 8th axis pulses (for travel axis, mm)	Number of 8th axis pulses (for travel axis, mm)
Data-11	Number of 9th axis pulses (for travel axis, mm)	Number of 9th axis pulses (for travel axis, mm)
Data-12	Number of 10th axis pulses	Number of 10th axis pulses
Data-13	Number of 11th axis pulses	Number of 11th axis pulses
Data-14	Number of 12th axis pulses	Number of 12th axis pulses
Data-15	-	

- “Number of 7th axis pulses” and after 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.



<Example>

Command RPOSC 2, 0
Response 100.0, 50, 34, 12.34, 180.0, 0, 0, 0, 0, 0, 0

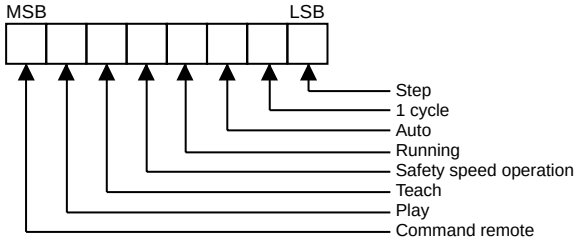
■ **RSTATS**

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

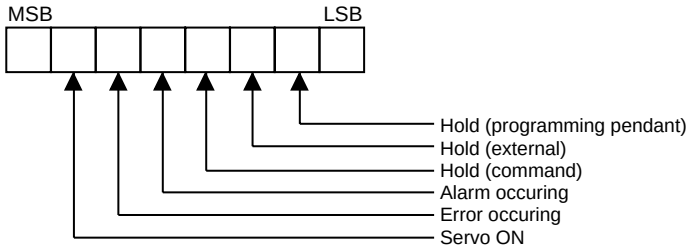
Command format : RSTATS

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

Data-1



Data-2



<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
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.

<Status>	**<Response>**
END or PAUSE execution during waiting time	Completed
Step execution during waiting time	Completed
Stopped by hold, alarm, emergency stop, servo OFF during waiting time	Not completed
Stopped by changing mode during waiting time	Not completed
Test run is interrupted during waiting time	Not completed
Waiting timeup	Not completed
Stopped (including when the control power ON)	Completed
Stopped (hold)	Interpreter message 2020 to 2050
Stopped (Alarm/error occurring)	Interpreter message 2060
Stopped (servo OFF)	Interpreter message 2070

<Example>

Command JWAIT 10
Response 0000

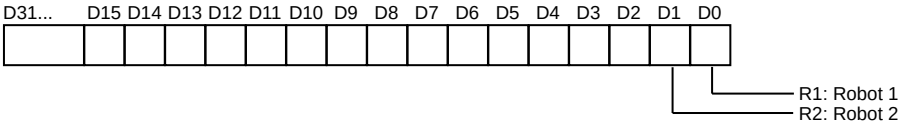
■ **RGROUP**

Reads the current control group set by CGROUP command or CTASK command, and the task selection status.

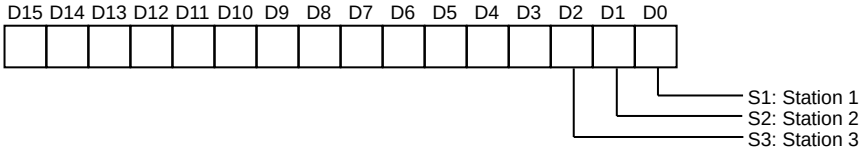
Command format : RGOUP

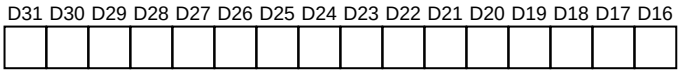
Response format : Data-1, Data-2, Data-3 or Error code

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



Data-2 = Station control group information.
The control group information differs depending on the number of manipulators in the system.





Data-3 = 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.

<Example>

Command RGROUP
Response 2, 1, 0

The above example shows that the current control group is robot1, robot 2 and station 1, and the task selection status is master task.

5.2.5.2 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
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 (32 characters)
Name-2 = Job name-2 (32 characters)
: :
: :
Name-N = Job name-N (32 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
:
:
17 : User coordinate 16

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

Data-1 = ORG X coordinate value (unit : mm, significant 3 decimal points)
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 4 decimal points)
Data-5 = ORG wrist angle TY (unit : degree (°), significant 4 decimal points)
Data-6 = ORG wrist angle TZ (unit : degree (°), significant 4 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 4 decimal points)
Data-12 = XX wrist angle TY (unit : degree (°), significant 4 decimal points)
Data-13 = XX wrist angle TZ (unit : degree (°), significant 4 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 4 decimal points)
Data-19 = XY wrist angle TY (unit : degree (°), significant 4 decimal points)
Data-20 = XY wrist angle TZ (unit : degree (°), significant 4 decimal points)
Data-21 = XY type
Data-22 = Tool No. (0 to 15)
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.
- For 7-axis robots, this command cannot be used.

<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)
7 : String variables
Data-2 : Variable No.

Response format : Data-1, Data-2, Data-3, ····, Data-11 or Error code
Data-1 = Byte value / Integer value / Double precision integer value / Real number value / Position data type / String

Position data type = 0 : Pulse type
1 : Cartesian type

(When the position data type is "0")

* The order of response data varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-2	Number of S-axis pulses / Number of base 1st axis pulses / Number of station 1st axis pulses	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	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	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	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	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	Number of T-axis pulses / Number of base 6th axis pulses / Number of station 6th axis pulses
Data-8	Tool No. (0 to 15)	Number of E-axis pulses / Number of base 7th axis pulses / Number of station 7th axis pulses
Data-9	-	Tool No. (0 to 15)
Data-10	-	-
Data-11	-	-

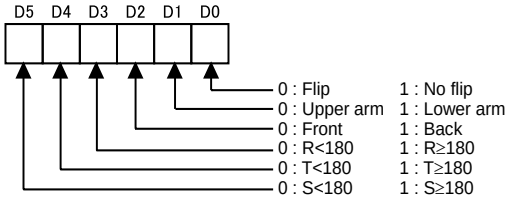
(When the position data type is “1”)

Data-2 = Coordinate data
- 0 : Base coordinate
- 1 : Robot coordinate
- 2 : User coordinate 1
- 3 : User coordinate 2
- :
- :
- 17 : User coordinate 16
- 18 : Tool coordinate
- 19 : Master tool coordinate

* The order of response data varies depending on the number of robot's axes.
When the system contains multiple robots, the order is that of the robot with the maximum number of axes.

	6-axis robot	**7-axis robot**
Data-3	X coordinate value / Base 1st Cartesian value (unit : mm, significant 3 decimal points)	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)	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)	Z coordinate value / Base 3rd Cartesian value (unit : mm, significant 3 decimal points)
Data-6	Wrist angle Rx coordinate value (unit : degree (°), significant 4 decimal points)	Wrist angle Rx coordinate value (unit : degree (°), significant 4 decimal points)
Data-7	Wrist angle Ry coordinate value (unit : degree (°), significant 4 decimal points)	Wrist angle Ry coordinate value (unit : degree (°), significant 4 decimal points)
Data-8	Wrist angle Rz coordinate value (unit : degree (°), significant 4 decimal points)	Wrist angle Rz coordinate value (unit : degree (°), significant 4 decimal points)
Data-9	Form	Elbow angle Re coordinate value (unit : degree (°), significant 4 decimal points)
Data-10	Tool No. (0 to 15)	Form
Data-11	-	Tool No. (0 to 15)

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



<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

5.2.6.1 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

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

NOTE This function can be used when the external mode switch is permitted on the OPERATING CONDITION window.

■ **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 29 of the input terminal block for the manipulator, to 30.

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 and I/O operation signals.

While the interlock is ON, all operations except the followings are prohibited.

- Emergency stop from the programming pendant
- Input signals except I/O mode change, external start, external servo ON, cycle change, I/O prohibited, 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.

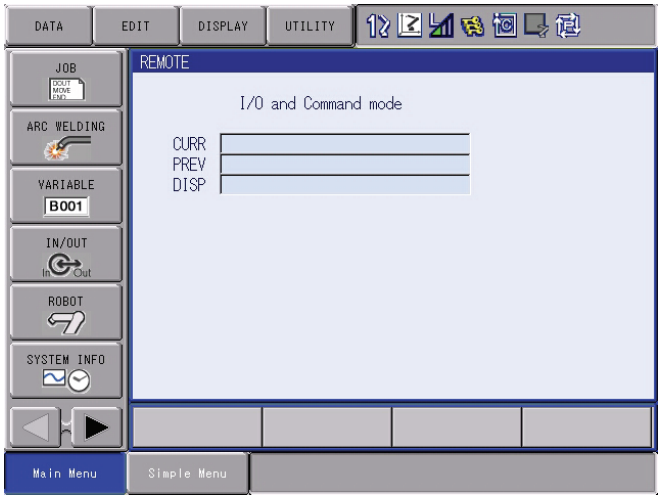
If the currently shown display is not the remote display, it is changed forcibly to the remote display to display the MDSP command message.

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 FS100 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.

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-2

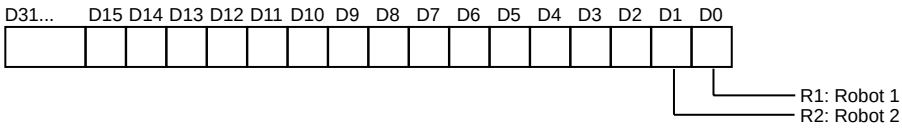
Data-1 = Robot control group specification.

A control group can be specified according to the following data.

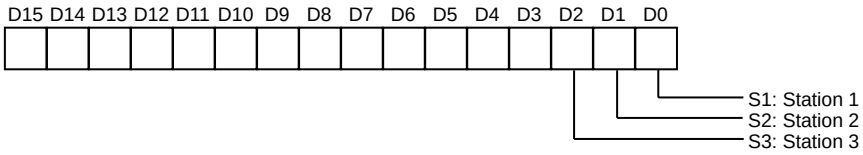
However, the following settings cannot be made.

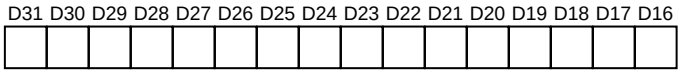
- Selection of control axis which does not exist
- Specification of multiple number of manipulators

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.



Data-2 = Station control group specification. A control group can be specified according to the following data.
However, the following settings cannot be made.
- Selection of control axis which does not exist
- Specification of multiple number of stations





Response format : 0000 or Error code

<Example>

Command CGROUP 2, 1
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 *chapter 5.3 "Commands for Multi-control Group and Independent Control Functions" at page 5-50*.

- When the power supply is started up, a master task is selected as an 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

5.2.6.2 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 (32 characters)
Can be omitted.

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-17
Data-1 = Motion speed (0.01 to 100.0%)
Data-2 = Coordinate specification
0 : Base coordinate
1 : Robot coordinate
2 : User coordinate 1
:
:
17 : User coordinate 16
* The order varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-3	X coordinate value (unit : mm, significant 3 decimal points)	X coordinate value (unit : mm, significant 3 decimal points)
Data-4	Y coordinate value (unit : mm, significant 3 decimal points)	Y coordinate value (unit : mm, significant 3 decimal points)
Data-5	Z coordinate value (unit : mm, significant 3 decimal points)	Z coordinate value (unit : mm, significant 3 decimal points)
Data-6	Wrist angle Rx (unit : degree (°), significant 4 decimal points)	Wrist angle Rx (unit : degree (°), significant 4 decimal points)
Data-7	Wrist angle Ry (unit : degree (°), significant 4 decimal points)	Wrist angle Ry (unit : degree (°), significant 4 decimal points)
Data-8	Wrist angle Rz (unit : degree (°), significant 4 decimal points)	Wrist angle Rz (unit : degree (°), significant 4 decimal points)
Data-9	Type	Elbow angle Re (unit : degree (°), significant 4 decimal points)
Data-10	Tool No. (0 to 15)	Type
Data-11	Number of 7th axis pulses (for travel axis, mm)	Tool No. (0 to 15)
Data-12	Number of 8th axis pulses (for travel axis, mm)	Number of 7th axis pulses (for travel axis, mm)
Data-13	Number of 9th axis pulses (for travel axis, mm)	Number of 8th axis pulses (for travel axis, mm)
Data-14	Number of 10th axis pulses	Number of 9th axis pulses (for travel axis, mm)
Data-15	Number of 11th axis pulses	Number of 10th axis pulses
Data-16	Number of 12th axis pulses	Number of 11th axis pulses
Data-17	-	Number of 12th axis pulses

- In a system without external axis, Data-11 to Data-16 (for 7-axis robots, 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 MOVJ 50.0, 2, 123.1, 50.34, 10.8, 180.0, 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-18

Data-1 = Motion speed selection (0 : V (speed), 1 : VR (posture speed))

Data-2 = Motion speed (0.1 to □□□.□□ mm/s, 0.1 to □□□.□° /s)

Data-3 = Coordinate specification

0 : Base coordinate

1 : Robot coordinate

2 : User coordinate 1

:

:

17 : User coordinate 16

* The order varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-4	X coordinate value (unit : mm, significant 3 decimal points)	X coordinate value (unit : mm, significant 3 decimal points)
Data-5	Y coordinate value (unit : mm, significant 3 decimal points)	Y coordinate value (unit : mm, significant 3 decimal points)
Data-6	Z coordinate value (unit : mm, significant 3 decimal points)	Z coordinate value (unit : mm, significant 3 decimal points)
Data-7	Wrist angle Rx (unit : degree (°), significant 4 decimal points)	Wrist angle Rx (unit : degree (°), significant 4 decimal points)
Data-8	Wrist angle Ry (unit : degree (°), significant 4 decimal points)	Wrist angle Ry (unit : degree (°), significant 4 decimal points)
Data-9	Wrist angle Rz (unit : degree (°), significant 4 decimal points)	Wrist angle Rz (unit : degree (°), significant 4 decimal points)
Data-10	Type	Elbow angle Re (unit : degree (°), significant 4 decimal points)
Data-11	Tool No. (0 to 15)	Type
Data-12	Number of 7th axis pulses (for travel axis, mm)	Tool No. (0 to 15)
Data-13	Number of 8th axis pulses (for travel axis, mm)	Number of 7th axis pulses (for travel axis, mm)
Data-14	Number of 9th axis pulses (for travel axis, mm)	Number of 8th axis pulses (for travel axis, mm)
Data-15	Number of 10th axis pulses	Number of 9th axis pulses (for travel axis, mm)
Data-16	Number of 11th axis pulses	Number of 10th axis pulses
Data-17	Number of 12th axis pulses	Number of 11th axis pulses
Data-18	-	Number of 12th axis pulses

- In a system without external axis, Data-12 to Data-17 (for 7-axis robots, Data-13 to Data-18) 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, 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-18
Data-1 = Motion speed selection (0 : V (speed), 1 : VR (posture speed))
Data-2 = Motion speed (0.1 to □□□.□□ mm/s, 0.1 to □□□.□° /s)
Data-3 = Coordinate specification
0 : Base coordinate
1 : Robot coordinate
2 : User coordinate 1
:
:
17 : User coordinate 16
18 : Tool coordinate
* The order varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-4	X coordinate incremental value (unit : mm, significant 3 decimal points)	X coordinate incremental value (unit : mm, significant 3 decimal points)
Data-5	Y coordinate incremental value (unit : mm, significant 3 decimal points)	Y coordinate incremental value (unit : mm, significant 3 decimal points)
Data-6	Z coordinate incremental value (unit : mm, significant 3 decimal points)	Z coordinate incremental value (unit : mm, significant 3 decimal points)
Data-7	Wrist angle Rx incremental value (unit : degree (°), significant 4 decimal points)	Wrist angle Rx incremental value (unit : degree (°), significant 4 decimal points)
Data-8	Wrist angle Ry incremental value (unit : degree (°), significant 4 decimal points)	Wrist angle Ry incremental value (unit : degree (°), significant 4 decimal points)
Data-9	Wrist angle Rz incremental value (unit : degree (°), significant 4 decimal points)	Wrist angle Rz incremental value (unit : degree (°), significant 4 decimal points)
Data-10	Reserved	Elbow angle Re incremental value (unit : degree (°), significant 4 decimal points)
Data-11	Tool No. (0 to 15)	Reserved
Data-12	Number of 7th axis pulses (for travel axis, mm)	Tool No. (0 to 15)
Data-13	Number of 8th axis pulses (for travel axis, mm)	Number of 7th axis pulses (for travel axis, mm)
Data-14	Number of 9th axis pulses (for travel axis, mm)	Number of 8th axis pulses (for travel axis, mm)
Data-15	Number of 10th axis pulses	Number of 9th axis pulses (for travel axis, mm)
Data-16	Number of 11th axis pulses	Number of 10th axis pulses
Data-17	Number of 12th axis pulses	Number of 11th axis pulses
Data-18	-	Number of 12th axis pulses

• In a system without external axis, Data-12 to Data-17 (for 7-axis robots, Data-13 to Data-18) 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-15
* The order varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-1	Motion speed (0.01 to 100.0 %)	Motion speed (0.01 to 100.0 %)
Data-2	Number of S-axis pulses	Number of S-axis pulses
Data-3	Number of L-axis pulses	Number of L-axis pulses
Data-4	Number of U-axis pulses	Number of U-axis pulses
Data-5	Number of R-axis pulses	Number of R-axis pulses
Data-6	Number of B-axis pulses	Number of B-axis pulses
Data-7	Number of T-axis pulses	Number of T-axis pulses
Data-8	Tool No. (0 to 15)	Number of E-axis pulses
Data-9	Number of 7th axis pulses	Tool No. (0 to 15)
Data-10	Number of 8th axis pulses	Number of 7th axis pulses
Data-11	Number of 9th axis pulses	Number of 8th axis pulses
Data-12	Number of 10th axis pulses	Number of 9th axis pulses
Data-13	Number of 11th axis pulses	Number of 10th axis pulses
Data-14	Number of 12th axis pulses	Number of 11th axis pulses
Data-15	-	Number of 12th axis pulses

- In a system without external axis, Data-9 to Data-14 (for 7-axis robots, Data-10 to Data-15) 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-16
Data-1 = Motion speed selection (0 : V (speed), 1 : VR (posture speed))
Data-2 = Motion speed (0.1 to □□□.□□ mm/s, 0.1 to □□□.□° /s)
* The order varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-3	Number of S-axis pulses	Number of S-axis pulses
Data-4	Number of L-axis pulses	Number of L-axis pulses
Data-5	Number of U-axis pulses	Number of U-axis pulses
Data-6	Number of R-axis pulses	Number of R-axis pulses
Data-7	Number of B-axis pulses	Number of B-axis pulses
Data-8	Number of T-axis pulses	Number of T-axis pulses
Data-9	Tool No. (0 to 15)	Number of E-axis pulses
Data-10	Number of 7th axis pulses	Tool No. (0 to 15)
Data-11	Number of 8th axis pulses	Number of 7th axis pulses
Data-12	Number of 9th axis pulses	Number of 8th axis pulses
Data-13	Number of 10th axis pulses	Number of 9th axis pulses
Data-14	Number of 11th axis pulses	Number of 10th axis pulses
Data-15	Number of 12th axis pulses	Number of 11th axis pulses
Data-16	-	Number of 12th axis pulses

- In a system without external axis, Data-10 to Data-15 (for 7-axis robots, Data-11 to Data-16) 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

5.2.6.3 Editing System Commands

■ **DELETE**

Deletes a specified job.

Command format : DELETE Job-Name
Job-Name = Job name to be deleted
= * : 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
:
:
17 : User coordinate 16
18 : 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

This function requires the relative job function of the FS100.

■ **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

<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.

This function required the relative job function of the FS100.

■ **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
:
:
17 : User coordinate 16

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 4 decimal points)

Data-6 = ORG wrist angle TY (unit : degree (°), significant 4 decimal points)

Data-7 = ORG wrist angle TZ (uunit : degree (°), significant 4 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 4 decimal points)

Data-13 = XX wrist angle TY (unit : degree (°), significant 4 decimal points)

Data-14 = XX wrist angle TZ (unit : degree (°), significant 4 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 4 decimal points)

Data-20 = XY wrist angle TY (unit : degree (°), significant 4 decimal points)

Data-21 = XY wrist angle TZ (unit : degree (°), significant 4 decimal points)

Data-22 = XY type

Data-23 = Tool No. (0 to 15)

Data-24 = Number of 7th axis pulses (for travel axis, mm)

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.
- For 7-axis robots, this command cannot 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-13

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)
- 7 : String variables

Data-2 = Variable No.

Data-3 = Byte value / Integer value / Double precision type integer value / Real number value / Position data type / String

Position data type = 0 : Pulse type
1 : Cartesian type

(When the position data type is 0)

* The order varies depending on the number of robot's axes.

	6-axis robot	**7-axis robot**
Data-4	Number of S-axis pulses / Number of base 1st axis pulses / Number of station 1st axis pulses	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	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	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	Number of R-axis pulses / Number of base 4th axis pulses / Number of station 4th axis pulses
Data-8	Number of B-axis pulses / Number of base 5th axis pulses / Number of station 5th axis pulses	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	Number of T axis pulses / Number of base 6th axis pulses / Number of station 6th axis pulses
Data-10	Tool No. (0 to 15)	Number of E axis pulses / Number of base 7th axis pulses / Number of station 7th axis pulses
Data-11	-	Tool No. (0 to 15)
Data-12	-	-
Data-13	-	-

(When the position data type is 1)

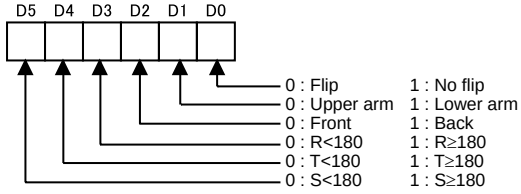
Data-4 = Coordinate data

Coordinate data = 0 : Base coordinate
1 : Robot coordinate
2 : User coordinate 1
3 : User coordinate 2
:
:
17 : User coordinate 16
18 : Tool coordinate
19 : Master tool coordinate

* The order of response data varies depending on the number of robot's axes.
When the system contains multiple robots, the order is that of the robot with the maximum number of axes.

	6-axis robot	**7-axis robot**
Data-5	X coordinate value / Base 1st axis Cartesian value (unit : mm, significant 3 decimal points)	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)	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)	Z coordinate value / Base 3rd axis Cartesian value (unit : mm, significant 3 decimal points)
Data-8	Wrist angle Rx coordinate value (unit : degree (°), significant 4 decimal points)	Wrist angle Rx coordinate value (unit : degree (°), significant 4 decimal points)
Data-9	Wrist angle Ry coordinate value (unit : degree (°), significant 4 decimal points)	Wrist angle Ry coordinate value (unit : degree (°), significant 4 decimal points)
Data-10	Wrist angle Rz coordinate value (unit : degree (°), significant 4 decimal points)	Wrist angle Rz coordinate value (unit : degree (°), significant 4 decimal points)
Data-11	Form	Elbow angle Re coordinate value (unit : degree (°), significant 4 decimal points)
Data-12	Tool No. (0 to 15)	Form
Data-13	-	Tool No. (0 to 15)

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 FS100 byte type variable B000.

5.2.6.4 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

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
Data-2 = Line No. to be set (0 to 9999)

Response format : 0000 or Error code

<Example>

Command JSEQ WORK-A, 10
Response 0000

5.2.7 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.

○ : Possible to execute

Signal	Signal Range (Qty)	Classification	Read-out	Write-in
0xxxx	00010 to 01287 (1024)	General input signal	○	
1xxxx	10010 to 11287 (1024)	General output signal	○	
2xxxx	20010 to 21287 (1024)	External input signal	○	
3xxxx	30010 to 31287 (1024)	External output signal	○	
4xxxx	40010 to 41607 (1280)	Specific input signal	○	
5xxxx	50010 to 52007 (1600)	Specific output signal	○	
7xxxx	70010 to 79997 (7992)	Auxiliary relay	○	
8xxxx	80010 to 80647 (512)	Control status signal	○	
82xxx	82010 to 82207 (160)	Pseudo input signal	○	
25xxx	25010 to 26287 (1024)	Network input	○	○

5.2.7.1 Transmission Procedure

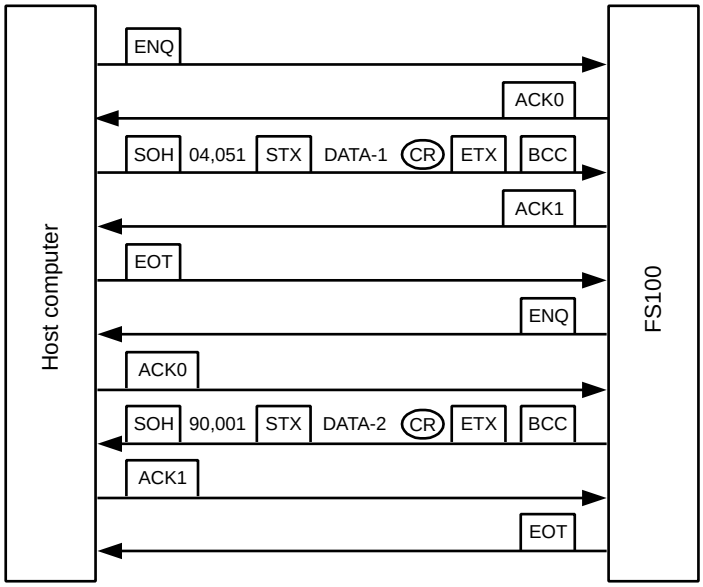
The transmission from the host computer proceeds as follows.

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 FS100 sends the ENQ code to establish the data link.
4. After the data link is established, the data sent from the FS100 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.

5.2.7.2 Read-out of I/O Signal Status



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) :

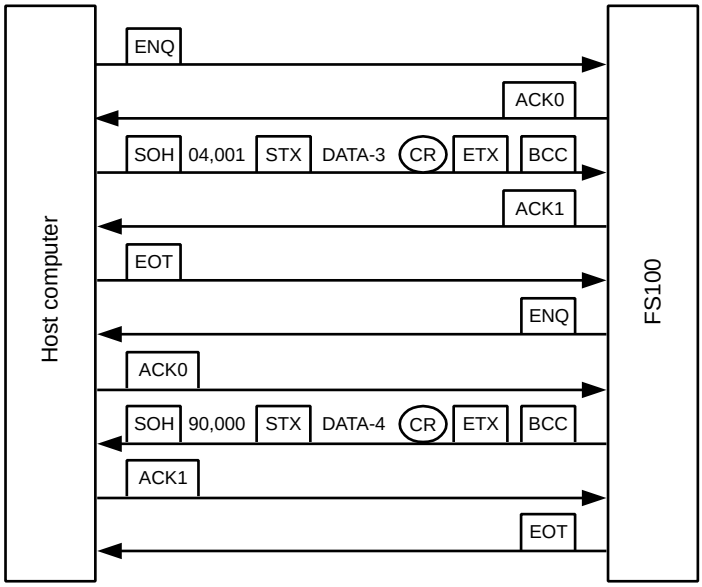
SOH 90,000 STX Error code CR ETX BCC

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 70010

Command 70010, 3
Response 2, 0, 5

5.2.7.3 Write-in of I/O Signal Status



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 22010

Command 22010, 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 FS100 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.

R1 (robot 1)	**R2 (robot 2)**	**S□[1)]**
	×	●
●[2)]	×	×
●	×	●
×	●	×
×	●	●

1 Either one station among S1 to S3 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.

Read System Commands	**Startup System Commands**	**Editing System Command**
RPOSJ RPOSC RUFRAME	MOVJ MOVL IMOV PMOVJ PMOVL	WUFRAME

5.3.2 Commands for Independent Control Function

The FS100 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 (JWAIT)**
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 (JSEQ)**
Sets a job name, a line No. to the currently selected task.

⑤ **Read of selected job (RJSEQ)**
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

Code	Contents	Remarks
4112	Data sending error 1 : NAK retry over 2 : Timer A timeup retry over 3 : Alternating response error retry over	The EOT code is sent out and the data link is canceled.
4113	Data receiving error 1 : Receiving timeup (Timer A) 2 : Receiving timeup (Timer B) 3 : Short heading length 4 : Long heading length 5 : Illegal header No. 6 : Text longer than 256 characters 7 : Receiving other than expected control code	For 3 to 7, the EOT code is sent out and the data link is canceled.
4114	Transmission hardware error 1 : Overrun error 2 : Parity error 3 : Framing error 4 : Sending timeup (Timer A) 5 : Sending timeup (Timer B)	The EOT code is not sent.
4115	Transmission system block 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. 1 : EOT was received while waiting for ACK 2 : EOT was received while waiting for ENQ 3 : EOT was received before receiving the last block 4 : Code other than EOT was received after receiving the last block	For 4, the EOT code is sent out and the data link is canceled.
4206	Transmission system error This alarm notifies an error on processing of transmission system. This alarm occurs in the following cases. 100 Error in transmission task • A job containing position type variable of which the value is not set, was to be saved. • A job which does not exist on the memory, was to be saved.	The EOT code is not sent.

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 5-1: Interpreter Message List (Sheet 1 of 2)

Code	Content
1010	Command error
1011	Error in number of command operands
1012	Command operand value range over
1013	Command operand length error
1020	Disk full of files
2010	Manipulator operating
2020	Hold by programming pendant
2030	Hold by playback panel
2040	External hold
2050	Command hold
2060	Error/alarm occurring
2070	Servo OFF
2080	Incorrect mode
2090	File accessing by other function
2100	Command remote not set
2110	This data cannot be accessed.
2120	This data cannot be loaded.
2130	Editing
3010	Turn ON the servo power.
3040	Perform home positioning.
3050	Confirm positions.
3070	Current value not made
3220	Panel lock ; mode/cycle prohibit signal is ON.
3230	Panel lock ; start prohibit signal is ON.
3350	User coordinate not taught
3360	User file destroyed
3370	Incorrect control group
3380	Incorrect base axis data
3390	Relative job conversion prohibit (at CVTRJ)
3400	Master call prohibit (parameter)
3410	Master call prohibit (lamp On during operation)
3420	Master call prohibit (teach lock)
3430	Robot calibration data not defined
3450	Robot calibration data not defined
3460	Coordinate system cannot be set.
4010	Insufficient memory capacity (job registered memory)

Table 5-1: Interpreter Message List (Sheet 2 of 2)

Code	Content
4012	Insufficient memory capacity (position data registered memory)
4020	Job editing prohibit
4030	Same job name exists
4040	No specified job
4060	Set a execution job.
4120	Position data destroyed
4130	Position data not exist
4140	Incorrect position variable type
4150	END instruction for job which is not master job
4170	Instruction data destroyed
4190	Invalid character in job name
4200	Invalid character in label name
4230	Invalid instruction in this system
4420	No step in job to be converted
4430	Already converted
4480	Teach user coordinate.
4490	Relative job/Independent control function not permitted
5110	Syntax error (syntax of instruction)
5120	Position data error
5130	No NOP or END instruction
5170	Format error (incorrect format)
5180	Incorrect number of data
5200	Data range over
5310	Syntax error (except instruction)
5340	Error in pseudo instruction specification
5370	Error in condition data record
5390	Error in job data record
5430	System not matched
5480	Incorrect welding function type

6 Data List

6.1 Header Number List

Contents			File Name
01,	**000**	Command from a external computer	
02,	**001**	Single job data	xxxxxxxx. JBI
	002	Related job data	xxxxxxxx. JBR
02,	**051**	Request for single job data	
	052	Request for related job data	
02,	**200**	Tool data	TOOL. CND
	202	User coordinate data	UFRAME. CND
	232	Variable data	VAR. DAT
	244	Shock detection level	SHOCKLVL.CND
	240	System information	SYSTEM. SYS
	241	Alarm history data	ALMHIST. DAT
02,	**300**	Request for tool data	TOOL.CND
	302	Request for user coordinate data	UFRAME.CND
	332	Request for variable data	VAR.DAT
	344	Request for shock detection level	SHOCKLVL.CND
	340	Request for system information	SYSTEM.SYS
	341	Request for alarm history data	ALMHIST.DAT
03,	**001**	Byte type variable	
	002	Integer type variable	
	003	Double precision type variable	
	004	Real number type variable	
	005	Robot axis position type variable (pulse type)	
	006	Robot axis position type variable (XYZ type)	
	007	External axis position type variable (pulse type)	
	008	External axis position type variable (XYZ type)	
03,	**051**	Request for byte type variable	
	052	Request for integer type variable	
	053	Request for double precision type variable	
	054	Request for real number type variable	
	055	Request for robot axis position type variable (pulse type)	
	056	Request for robot axis position type variable (XYZ type)	
	057	Request for external axis position type variable (pulse type)	
	058	Request for external axis position type variable (XYZ type)	
04,	**001**	Request for write-in of I/O signals	
	051	Request for read-out of I/O signals	
90,	**000**	Command or data response (normal/error)	
	001	Command or data response (data)	

6.2 Parameter List

Table 6-1: Parameter for Transmission

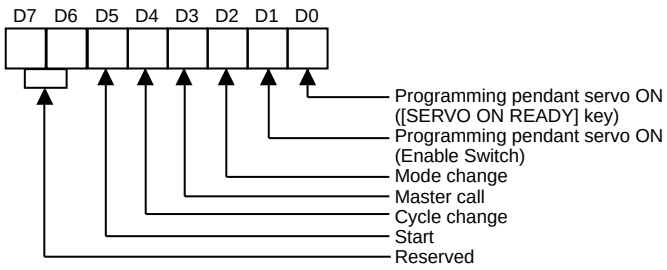
Parameter	Contents and Set Value	Initial Value
S2C230	Programming pendant operation (in remote) specification 0 : Invalid 1 : Valid 	0
RS000	Standard port protocol specification 0 : NON 1 : System reserved 2 : BSC LIKE 3 : FC1	2

Table 6-2: Parameter for Transmission (for BSC protocol)

Parameter	Contents and Set Value	Initial Value
RS030	Number of data bits 7 : 7 (bit) 8 : 8	8
RS031	Number of stop bits 0 : 1 (bit) 1 : 1.5 2 : 2	0
RS032	Parity specification 0 : No specification 1 : Odd parity 2 : Even parity	2
RS033	Transmission speed specification 1 : 150 (baud rate) 2 : 300 3 : 600 4 : 1200 5 : 2400 6 : 4800 7 : 9600 8 : 19200	7
RS034	Timer A : Sequence monitoring timer Serves as protection against invalid response or no response Unit : 0.1 sec. (Setting range : 0 to 100)	30
RS035	Timer B : Text reception monitoring timer Serves as protection against no response of text end character Unit : 0.1 sec. (Setting range : 0 to 255)	200

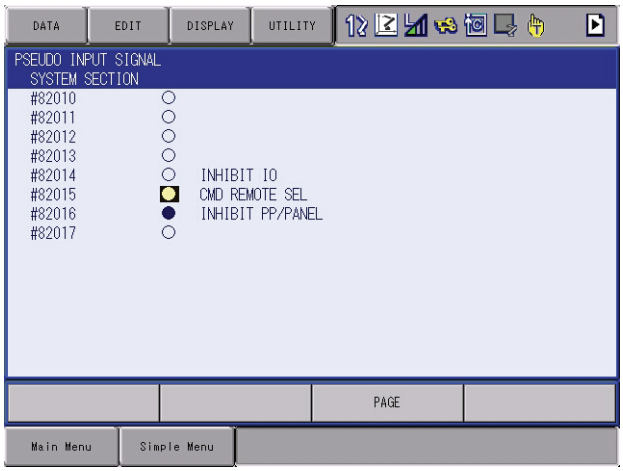
Table 6-2: Parameter for Transmission (for BSC protocol)

Parameter	Contents and Set Value	Initial Value
RS036	Retry 1 : Number of resendings of a control character for invalid response or no response Setting range : 0 to 30	10
RS037	Retry 2 : Number of resendings of a text for a block check error (reception of NAK) Setting range : 0 to 10	3
RS038	Block check method 0 : Check sum	0

7 Remote Function Setting

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

1. Select {IN/OUT} under the main menu.
2. Select {PSEUDO INPUT SIG}.
3. Select an item.
   – Select “INHIBIT IO” or “CMD REMOTE SEL”.
   The item enabled is marked with “●“ while the item disabled is marked with “○“.



   – 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 programming pendant mode key set to [REMOTE].
   – 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 programming pendant mode key set to [REMOTE].
   – When CMD REMOTE SEL is marked with ○ (disabled), the host control function is disabled.
   – When INHIBIT P.P/PANEL is marked with ○ (disabled), the operation from P.P/PANEL is enabled even in remote mode. When INHIBIT P.P/PANEL is marked with ● (enabled), the operation from P.P/PANEL is disabled, except for the operations of emergency stop, hold, and remote key.

FS100 OPTIONS
INSTRUCTIONS
FOR DATA TRANSMISSION FUNCTION

HEAD OFFICE
2-1 Kurosakishiroishi, Yahatanishi-ku, Kitakyushu 806-0004 Japan
Phone +81-93-645-7745 Fax +81-93-645-7746

YASKAWA America Inc. MOTOMAN Robotics Division
805 Liberty Lane, West Carrollton, OH 45449, U.S.A.
Phone +1-937-847-6200 Fax +1-937-847-6277

YASKAWA Nordic AB
Verkstadsgatan 2, PO Box 504,SE-385 25 Torsås, Sweden
Phone +46-480-417-800 Fax +46-486-414-10

YASKAWA Europe GmbH Robotics Div.
Kammerfeldstr. 1, 80591 Allershausen, Germany
Phone +49-8166-90-0 Fax +49-8166-90-103

YASKAWA Electric Korea Co., Ltd
9F, KYOBO Securities Bldg., 26-4, Yeoido-Dong Yeoungeungpo-ku,Seoul, KOREA
Phone +82-2-784-7844 Fax +82-2-784-8495

YASKAWA Electric (Singapore) PTE Ltd.
151 Lorong Chuan, #04-02A, New Tech Park, Singapore 556741
Phone +65-6282-3003 Fax +65-6289-3003

YASKAWA Electric (Thailand) Co., Ltd.
252/246, 4th Floor. Muang Thai-Phatra Office Tower II Rachadaphisek Road, Huaykwang Bangkok, 10320 Thailand
Phone +66-2-693-2200 Fax +66-2-693-4200

Shougang MOTOMAN Robot Co. Ltd.
No.7, Yongchang-North Road, Beijing E&T Development Area, China 100176
Phone +86-10-6788-0548 Fax +86-10-6788-0548-813

YASKAWA ELECTRIC (SHANGHAI) Co., Ltd.
No.18Xizang Zhong Road, 17F, Harbour Ring Plaza, Shanghai 200001, CHINA
Phone +86-21-5385-0655 Fax +86-21-5385-2770

YASKAWA Robotics India Ltd.
#426, Udyog Vihar, Phase- IV, Gurgaon, Haryana, India
Phone +91-124-475-8500 Fax +91-124-414-8016

YASKAWA ELECTRIC CORPORATION

MANUAL NO. RE-CKI-A459