FANUC Robot series

R-30*i*B Plus/R-30*i*B Mate Plus/R-30*i*B Compact Plus
CONTROLLER

PLC Motion I/O Interface Function

OPERATOR'S MANUAL

B-84214EN/01

• Original Instructions

Thank you very much for purchasing FANUC Robot.

Before using the Robot, be sure to read the "FANUC Robot SAFETY HANDBOOK (B-80687EN)" and understand the content.

The products in this manual are controlled based on Japan's “Foreign Exchange and Foreign Trade Law". The export from Japan may be subject to an export license by the government of Japan. Further, re-export to another country may be subject to the license of the government of the country from where the product is re-exported. Furthermore, the product may also be controlled by re-export regulations of the United States government. Should you wish to export or re-export these products, please contact FANUC for advice.

In this manual, we endeavor to include all pertinent matters. There are, however, a very large number of operations that must not or cannot be performed, and if the manual contained them all, it would be enormous in volume. It is, therefore, requested to assume that any operations that are not explicitly described as being possible are "not possible".

SAFETY PRECAUTIONS

This chapter must be read before using the robot.

For detailed functions of the robot operation, read the relevant operator's manual to understand fully its specification.

For the safety of the operator and the system, follow all safety precautions when operating a robot and its peripheral equipment installed in a work cell.
For safe use of FANUC robots, you must read and follow the instructions in “FANUC Robot SAFETY HANDBOOK (**B-80687EN)**”.

1 DEFINITION OF USER

The personnel can be classified as follows.

Operator:
- Turns the robot controller power on/off
- Starts the robot program from operator panel

Programmer or Teaching operator:
- Operates the robot
- Teaches the robot inside the safety fence

Maintenance technician:
- Operates the robot
- Teaches the robot inside the safety fence
- Performs maintenance (repair, adjustment, replacement)

- Operator is not allowed to work in the safety fence.
- Programmer/Teaching operator and maintenance technician is allowed to work in the safety fence. Works carried out in the safety fence include transportation, installation, teaching, adjustment, and maintenance.
- To work inside the safety fence, the person must be trained on proper robot operation.

Table 1 (a) lists the work outside the safety fence. In this table, the symbol “○” means the work allowed to be carried out by the worker.

Table 1 (a) List of work outside the fence

	Operator	Programmer or Teaching operator	Maintenance technician
Turn power ON/OFF to Robot controller	○	○	○
Select operating mode (AUTO, T1, T2)		○	○
Select remote/local mode		○	○
Select robot program with teach pendant		○	○
Select robot program with external device		○	○
Start robot program with operator’s panel	○	○	○
Start robot program with teach pendant		○	○
Reset alarm with operator’s panel		○	○
Reset alarm with teach pendant		○	○
Set data on teach pendant		○	○
Teaching with teach pendant		○	○
Emergency stop with operator’s panel	○	○	○
Emergency stop with teach pendant	○	○	○
Operator’s panel maintenance			○
Teach pendant maintenance			○

In the robot operating, programming and maintenance, the operator, programmer/teaching operator and maintenance technician take care of their safety using at least the following safety protectors.

- Use clothes, uniform, overall adequate for the work
- Safety shoes
- Helmet

2 DEFINITION OF SAFETY NOTATIONS

To ensure the safety of users and prevent damage to the machine, this manual indicates each precaution on safety with "**WARNING**" or "**CAUTION**" according to its severity. Supplementary information is indicated by "**NOTE**". Read the contents of each "**WARNING**", "**CAUTION**" and "**NOTE**" before using the robot.

Symbol	Definitions
⚠**WARNING**	Used if hazard resulting in the death or serious injury of the user will be expected to occur if he or she fails to follow the approved procedure.
⚠**CAUTION**	Used if a hazard resulting in the minor or moderate injury of the user, or equipment damage may be expected to occur if he or she fails to follow the approved procedure.
NOTE	Used if a supplementary explanation not related to any of WARNING and CAUTION is to be indicated.

- Check this manual thoroughly, and keep it handy for the future reference.

TABLE OF CONTENTS

APPENDIX

1 OVERVIEW

1.1 OVERVIEW / DESCRIPTION

PLC Motion I/O Interface Function is a function to control FANUC robot by using a Program Logic Controller (PLC). The robot controller creates and executes a Teach Pendant (TP) program by adding Function Block (FB) to the PLC program that sends command data to operate the robot and executing the PLC program. The digital input/output signal (DI/DO) between the PLC and the Robot controller is used for data transmission and reception.

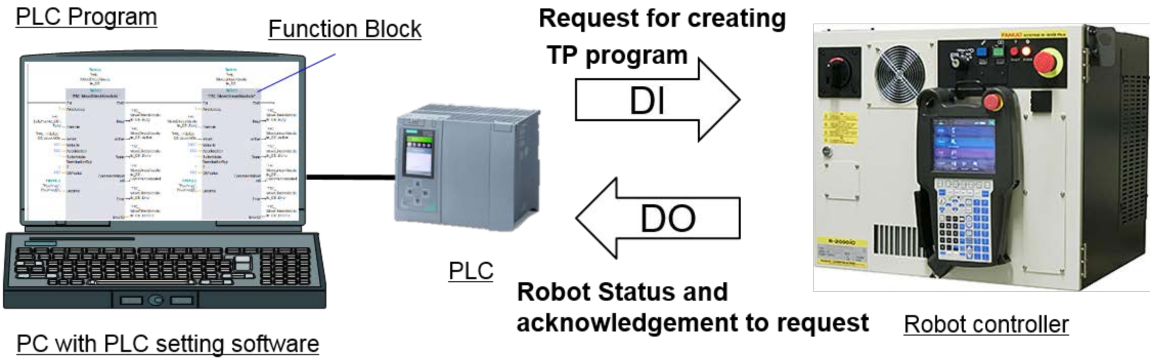

PLC Program
Function Block
Request for creating TP program
DI
DO
PLC
Robot Status and acknowledgement to request
Robot controller
PC with PLC setting software

Fig. 1.1(a) System Overview

A library for the Siemens PLC project is provided as a sample by FANUC representatives. You can control the FANUC robot in addition to Siemens PLC by creating FB based on the sample library on the PLC setting software to be used. The sample FBs are designed to be PLCopen compliant, and therefore follow the standard defined by IEC 61131-3.

NOTE
The target market for this software is material handling with simple pick and place operations. The software is not designed for complicated processes that use real time sensor feedback to dynamically adjust the robot path, such as arc welding applications.

1.2 COMMUNICATION

The communication uses cyclic communication by using DI/DO. This function uses 64 bytes (512 points) for each continuous DI/DO area.
The basic processing flow is as follows.

1. PLC write FB's data to robot controller's DI area as a request (REQ).
2. Robot controller reads the REQ from PLC as DI and processes the application.
3. After processing REQ, robot controller outputs the data to the DO area as acknowledgement (ACK).

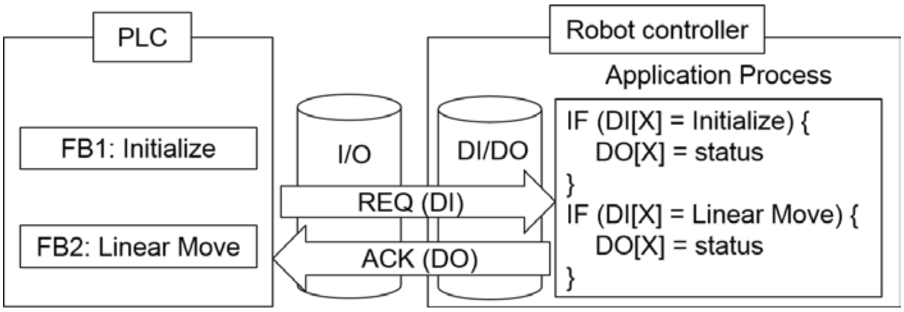


Fig. 1.2(a) Schematic diagram of communication

1.3 SOFTWARE REQUIREMENTS

1.3.1 Robot Controller Software

An R30*i*B Plus or an R30*i*B Mate Plus or R30*i*B Compact Plus controller with HandlingTool and the following software options are necessary:

- S525: PLC Motion I/O Interface Function

1.3.2 Compatibility

PLC Motion I/O Interface Function cannot be loaded into robot controller at the same time as PLC Motion Interface (R892) or Remote Motion Interface (R912).

1.4 LIMITATIONS

1.4.1 Single Motion Group

Only one group is supported for the first release. However, extended axes are supported.

1.4.2 Hot Start

Hot start is not allowed for a controller running this function. If hot start is set to TRUE, this function sends an error requesting a restart when it receives the FRC_Initialize command. Hot Start will be disabled and it will work after power off/on.

1.4.3 Motion With Short Distance

If the motion with very short distance (for example, motion of the robot by 10 mm) is executed by this function, the Robot controller will automatically decelerate to process this motion.

1.4.4 Original Path Resume

The original path resume feature is disabled when this function is installed. Therefore, if the robot is jogged away from its path between motion commands, the robot will go directly from its current jogged position to the next PLC command position. It does not go back to the original programmed path.

1.4.5 Controller Backup/TP Programs Backup

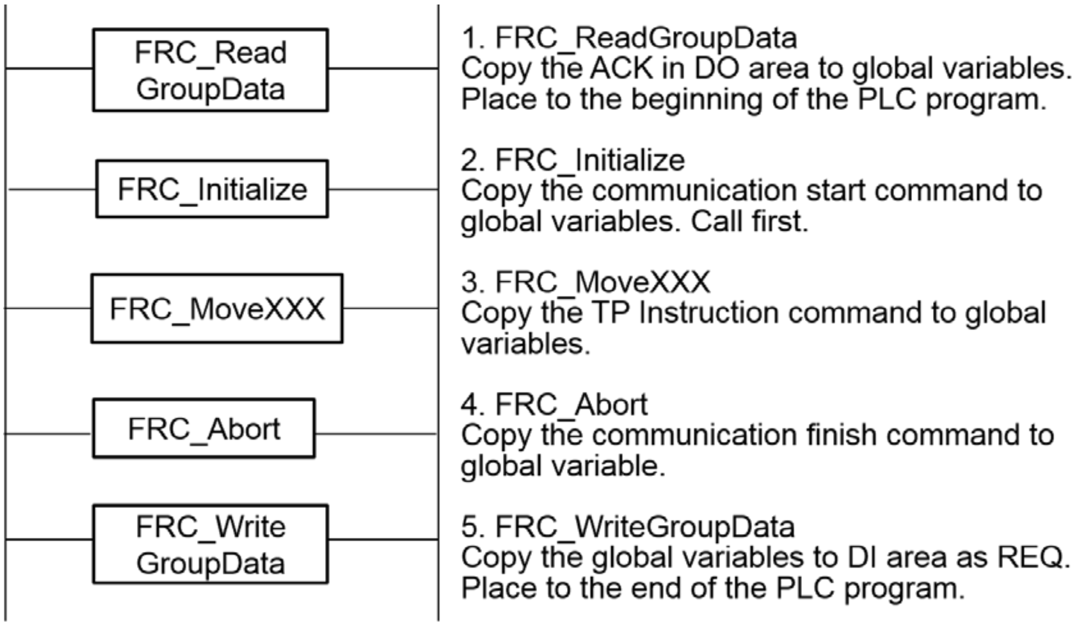
When this function is running, it locks up a TP program to run the commands. If you do a file backup or system backup at this time, the backup will fail, since the locked TP program cannot be backed up. You can issue the FRC_Abort command to stop this function, thus allowing file backup to proceed.

2 SPECIFICATION OF INTERFACE

This section shows PLC program and FBs corresponding to the PLC Motion I/O Interface Function.

2.1 PROCESS FLOW OF PLC PROGRAM

PLC program that use FBs is set to be executed cyclically. FBs are placed as follows.
Global variable are values defined in the PLC setting software and can be referenced in the PLC project. For more information about global variable, see section 2.3.



1. FRC_ReadGroupData
Copy the ACK in DO area to global variables.
Place to the beginning of the PLC program.

2. FRC_Initialize
Copy the communication start command to global variables. Call first.

3. FRC_MoveXXX
Copy the TP Instruction command to global variables.

4. FRC_Abort
Copy the communication finish command to global variable.

5. FRC_WriteGroupData
Copy the global variables to DI area as REQ.
Place to the end of the PLC program.

Fig. 2.1(a) Placement of FB

The process flow of FBs in the PLC program is as follows.

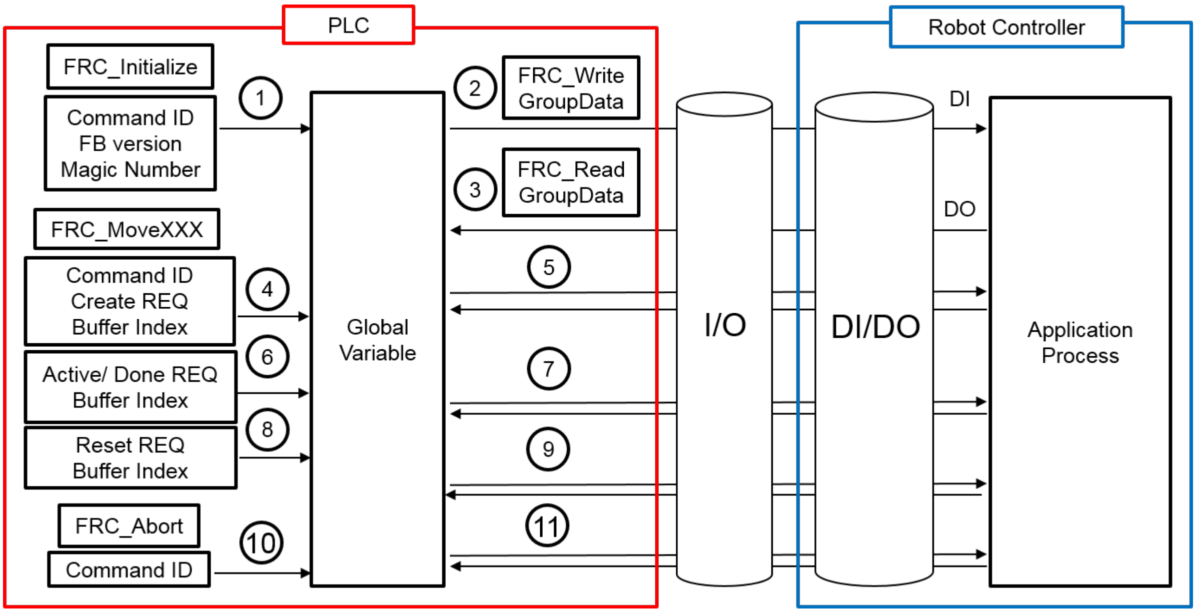


Fig. 2.1(b) Process flow of FBs

1. If the FRC_Initialize is executed, copy the Command data (Command ID, FB version, and Magic Number (like CRC)) to global variable in the PLC.
2. The FRC_WriteGroupData copy the global variable to DI area.
3. The Robot controller reads the DI data and process it. If the Magic Number and FB version are collect, the Robot controller creates TP program. And the Robot controller outputs the ACK to DO area. The ACK in DO are is inputted in global variable by FRC_ReadGroupData.
4. If the FRC_MoveXXX is executed, the PLC sends the Command data for create request of TP instruction (Create REQ move instruction Command ID, input parameters, Buffer Index) to the Robot controller via the FRC_WriteGroupData.
5. The move instruction correspond to the command ID is added in the TP program. After the move instruction is created, the Robot controller returns Create ACK and Buffer Index to the PLC. The execution status of move instruction is held in the Robot controller buffer. (The PLC refers to the buffer in the Robot controller by sending REQ. The buffer index is also sent at the time of REQ to know which buffer is correspond. For more information about buffer, see section 2.7.1.)
6. The PLC sends Active/Done REQ to check the buffer status.
7. If the move instruction is running/done, the Robot controller returns Active/Done ACK.
8. If PLC received Done ACK, the PLC sends Reset REQ to reset the buffer status.
9. The Robot Controller reset the buffer status and returns Reset ACK.
10. If the FRC_Abort is executed, the PLC sends the Abort command.
11. The Robot controller finish the communication session and returns Command ID ACK.

2.2 DATA TYPE

This section describes the data types used in FB.

2.2.1 Basic Data Type

This data types is defined in this article

Data Type	Length [Bits]	Value range
BOOL	1	True, False
SINT	8	-128 ~ +127
INT	16	-32,768 ~ +32,767
DINT	32	-2,147,483,648 ~ +2,147,483,647
USINT	8	0 ~ 255
UINT	16	0 ~ 65,535
UDINT	32	0 ~ 4,294,967,295
REAL	32	-3,402823e+38 ~ +3,402823e+38

NOTE

The DI/DO is big endian in the robot controller. If the PLC adopts little endian, byte swap is required in the FRC_WriteGroupData and FRC_ReadGroupData when using 2 bytes or 4 bytes data.

2.2.2 FRC_COORDSYS_T

This data type is used in the Robot position data.

Member Name	Data Type	Description
UTOOLNUM	USINT	Tool Frame Number
UFRAMENUM	USINT	User Frame Number
HEADER	Array[0~7] of SINT	Array of Robot configuration

Member Name	Data Type	Description
HEADER[0]	SINT	Front/Back (Not 0/ 0)
HEADER[1]	SINT	Up/Down (Not 0/ 0)
HEADER[2]	SINT	Left/Right (Not 0/ 0)
HEADER[3]	SINT	Flip/Non Flip (Not 0/ 0)
HEADER[4]	SINT	J4 Turn Number (1: 180º~539º, 0: -179º~179º, -1: -539º~-180º)
HEADER[5]	SINT	J5 Turn Number (1: 180º~539º, 0: -179º~179º, -1: -539º~-180º)
HEADER[6]	SINT	J6 Turn Number (1: 180º~539º, 0: -179º~179º, -1: -539º~-180º)
HEADER[7]	SINT	Not used

2.2.3 FRC_POS_T

This data type is used for the Robot position data.

Member Name	Data Type	Description
COORDSYS	FRC_COORDSYS_T	Array of the Robot position configuration data
POS	Array[0~8] of REAL	Array of the Robot position
POS[0]	REAL	X/J1(Cartesian/Joint Position)
POS[1]	REAL	Y/J2(Cartesian/Joint Position)
POS[2]	REAL	Z/J3(Cartesian/Joint Position)
POS[3]	REAL	W/J4(Cartesian/Joint Position)
POS[4]	REAL	P/J5(Cartesian/Joint Position)
POS[5]	REAL	R/J6(Cartesian/Joint Position)
POS[6]	REAL	E1
POS[7]	REAL	E2
POS[8]	REAL	E3

2.3 I/O MAPPING

This is the DI/DO area where the PLC and the Robot controller exchange. It is used as global variable in the PLC logic. The meaning of each data and the used FB are explained. For the input value and output value, see section 2.5, 2.6 and 2.7.

2.3.1 REQ DATA

This is transmitted data from the PLC to the Robot controller. When writing in FRC_WriteGroupData, this data is written in the memory of DI area of the Robot controller.

Data Name	Data Type	Description
COUNT_HEAD	USINT	Meaning: REQ counter at the beginning of data Used FB: FRC_WriteGroupData
CMD_ID	USINT	Meaning: Command ID Used FB: Other than FRC_WriteGroupData and FRC_ReadGroupData
BUF_INDEX	USINT	Meaning: Buffer index Used FB: TP instruction FB
VAL1	USINT	Meaning: Acceleration value Used FB: TP move instruction FB
VAL2	UINT	Meaning: The content changes depending on the command Used FB: TP move instruction FB, FRC_TriggerAlongPath

Data Name	Data Type	Description
VAL3	UINT	Meaning: Index value of DI/DO Used FB: FRC_TriggerAlongPath, FRC_WaitForDIN
VAL4	UINT	Meaning: The content changes depending on the command Used FB: FRC_WaitForDIN, FRC_WaitTime, FRC_TriggerAlongPath, FRC_AddMotionOption
VAL5	UINT	Not used in the current version
VAL6	USINT	Meaning: The content changes depending on the command Used FB: TP move instruction FB, FRC_TriggerAlongPath, FRC_GroupSetOverride, FRC_SetUFrame, FRC_SetUtool, FRC_SetPayload
VAL7	USINT	Meaning: CNT value for CNT Termination Type Used FB: TP move instruction FB
POS_UTOOLNUM	USINT	Meaning: The content changes depending on the command Used FB: TP move instruction FB, FRC_TriggerAlongPath, FRC_AddMotionOption, FRC_Initialize
POS_UFRAMENUM	USINT	Meaning: The content changes depending on the command Used FB: TP move instruction FB, FRC_WriteUFrameData, FRC_WriteUtoolData, FRC_TriggerAlongPath, FRC_AddMotionOption, FRC_Initialize
POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of Robot Used FB: TP move instruction FB
POS	Array[0~8] of REAL	Meaning: The content changes depending on the command Used FB: TP move instruction FB, FRC_WriteUFrameData, FRC_WriteUtoolData, FRC_Initialize
RESERVED	USINT	Not used in the current version
RESERVED	USINT	Not used in the current version
CREATE_REQ	BOOL	Meaning: Request to confirm that TP line is created in the Robot controller. Used FB: TP instruction FB
ACTIVE_REQ	BOOL	Meaning: Request to confirm that TP line is executing in the Robot controller. Used FB: TP instruction FB
DONE_REQ	BOOL	Meaning: Request to confirm that TP line is completed in the Robot controller. Used FB: TP instruction FB
RESET_REQ	BOOL	Meaning: Request to empty the Robot controller’s buffer contents. Used FB: TP instruction FB
NOT_REQFIN	BOOL	Meaning: Request incomplete bit Used FB: FRC_MoveCircularAbsolute/Relative, FRC_TriggerAlongPath, FRC_AddMotionOption
RESERVED	BOOL	Not used in the current version
RESERVED	BOOL	Not used in the current version
CTRL_PLC	BOOL	Meaning: Handshake bit from PLC Used FB: FRC_WriteGroupData, FRC_ReadGroupData
COUNT_TAIL	USINT	Meaning: REQ counter at the end of data Used FB: FRC_WriteGroupData

2.3.2 ACK DATA

This is transmitted data from the Robot controller to the PLC. When reading ACK in FRC_ReadGroupData, this data is written from the memory corresponding to the DO area of the Robot controller.

Data Name	Data Type	Description
COUNT_HEAD	USINT	Meaning: ACK counter at the beginning of data Used FB: FRC_ReadGroupData
MOSTRM	BOOL	PLC Motion I/O Interface Status (TRUE: OK). After FRC_Initialize is success, the Robot controller returns this value cyclically.
SRVRDY	BOOL	Servo ready (TRUE: OK). After FRC_Initialize is success, the Robot controller returns this value cyclically.
TPENB	BOOL	TP operation mode (TRUE: enabled. FALSE: disabled).After FRC_Initialize is success, the Robot controller returns this value cyclically.
REMOTE	BOOL	Remote configuration (TRUE: remote). After FRC_Initialize is success, the Robot controller returns this value cyclically.
SINGLESTEP	BOOL	Single step mode (TRUE: enabled). After FRC_Initialize is success, the Robot controller returns this value cyclically.
RESERVED	BOOL	Not used in the current version
RESERVED	BOOL	Not used in the current version
RESERVED	BOOL	Not used in the current version
TPMODE	USINT	TP program status (0: Running, 1: Aborted, 2: Paused). After FRC_Initialize is success, the Robot controller returns this value cyclically.
RESERVED	USINT	Not used in the current version
TCP_SPEED	REAL	TCP speed. After FRC_Initialize is success, the Robot controller returns this value cyclically.
RESERVED	USINT	Not used in the current version
POS_IDX	USINT	Meaning: Touchup Position Index Used FB: FRC_TouchUp
ACK_POS	FRC_POS_T	Meaning: Robot Position data Used FB: FRC_GroupReadActualPosition, FRC_GroupReadJointPosition, FRC_ReadUFrameData, FRC_GroupReadUtoolData, FRC_TouchUp
ERR_ID	UDINT	Meaning: Error ID in the Robot controller Used FB: Other than FRC_WriteGroupData and FRC_ReadGroupData
CMDID_ACK	USINT	Meaning: ACK for CMD_ID Used FB: Other than FRC_WriteGroupData and FRC_ReadGroupData
BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Used FB: TP instruction FB
CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Used FB: TP instruction FB
ACTIVE_ACK	BOOL	Meaning: ACK for ACTIVE_REQ Used FB: TP instruction FB
DONE_ACK	BOOL	Meaning: ACK for DONE_REQ Used FB: TP instruction FB
RESET_ACK	BOOL	Meaning: ACK for RESET_REQ Used FB: TP instruction FB
REQFIN_ACK	BOOL	Meaning: ACK for NOT_REQFIN Used FB: Other than FRC_WriteGroupData and FRC_ReadGroupData
RESERVED	BOOL	Not used in the current version
RESERVED	BOOL	Not used in the current version
CTRL_ROB	BOOL	Meaning: Handshake bit from Robot controller Used FB: FRC_WriteGroupData, FRC_ReadGroupData
COUNT_TAIL	USINT	Meaning: ACK counter at the end of data Used FB: FRC_ReadGroupData

2.4 CLASSIFICATION OF FB

The FBs of this function can be divided into three categories:

1. Communication FB: These FBs communicate the data between the Robot controller and the PLC.
2. Administrative FB: These FBs command the Robot controller to perform an immediate action.
3. TP Instruction FB: These FBs append a TP instruction to the TP program.

2.5 COMMUNICATION FB

These FBs communicate the data between the Robot controller and the PLC. These FBs have no internal status.

NOTE

The DI/DO is big endian in the robot controller. If the PLC adopts little endian, byte swap is required in the FRC_WriteGroupData and FRC_ReadGroupData when using 2 bytes or 4 bytes data.

2.5.1 FRC_WriteGroupData

This FB writes the send data in the PLC as DI and sends to the Robot controller. This FB should be placed at the end of the PLC program.

The handshake is performed so that the DI/DO area is not rewritten simultaneously with the PLC and the Robot controller. This handshake process is supposed to be managed by FRC_WriteGroupData.
Prepare the DI/DO handshake bit on PLC and Robot side. The state of the handshake bit in the DI/DO area is monitored to determine whether it is the PLC or the Robot controller that can access the DI/DO area. (The PLC monitors the DO bit on the Robot controller side. The Robot controller monitors the DI bit on the PLC side.)
For more information about handshake bit, see section 2.3.

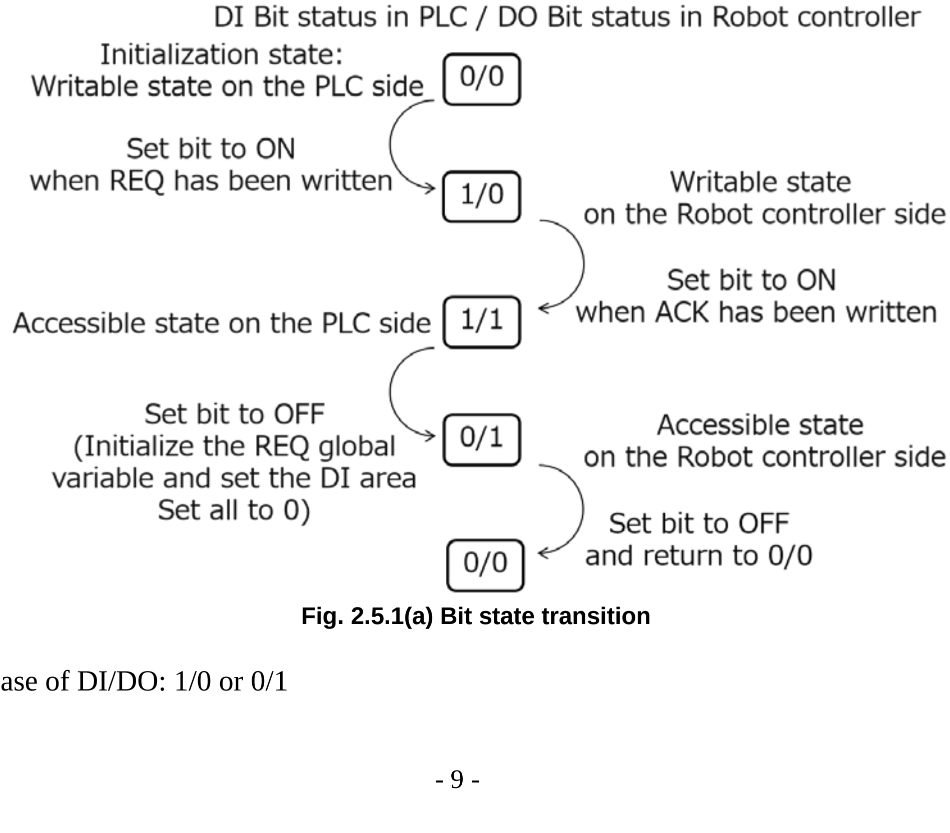

DI Bit status in PLC / DO Bit status in Robot controller
Initialization state:
Writable state on the PLC side
0/0
Set bit to ON
when REQ has been written
1/0
Writable state
on the Robot controller side
Set bit to ON
when ACK has been written
Accessible state on the PLC side
1/1
Set bit to OFF
(Initialize the REQ global
variable and set the DI area
Set all to 0)
0/1
Accessible state
on the Robot controller side
Set bit to OFF
and return to 0/0
0/0

Fig. 2.5.1(a) Bit state transition

(a) In the case of DI/DO: 1/0 or 0/1

Do not copy the variable for REQ into the DI area. (Wait for the Robot controller to change DI/DO: 1/1 or 0/0.)

(b) In the case of DI/DO: 1/1
Set all variables for REQ except the counter to 0. (The bit status becomes DI/DO: 0/1, since the bit on the PLC side also becomes FALSE.)
The COUNT_HEAD of REQ is incremented by 1 and the same value is copied to the COUNT_TAIL.
Copy the variable for REQ to the DI area and clear REQ other than the counter.

(c) In the case of DI/DO: 0/0
Set the handshake bit in the PLC to TRUE.
The COUNT_HEAD of REQ is incremented by 1 and the same value is copied to the COUNT_TAIL.
Copy the variables for REQ into the DI area.

The values to be input to the REQ global variables in FRC_WriteGroupData for communication are as follows.

Data Name	Parameter Name	Data Type	Description
REQ data	COUNT_HEAD	USINT	Meaning: REQ counter at the beginning of data. Add +1 before copying to DI area. Input range: 1~255
	COUNT_TAIL	USINT	Meaning: REQ counter at the end of data. Copy the value of COUNT_HEAD. Input range: 1~255

2.5.2 FRC_ReadGroupData

This FB reads the response sent from the Robot controller to the PLC as DO. Use this FB at the beginning of the PLC program to copy the latest data from the DO to the structure in the PLC.

This FB is placed at the beginning of the PLC program to confirm the handshake bits from the Robot controller in DO area. If the PLC is in the state to available to read DO, copy the ACK from the Robot controller to global variables.

(a) In the case of DI/DO: 1/1
Copy data in the DO area to the variable for ACK.

(b) In the case of DI/DO: 0/1
If the handshake bit in the DO area is FALSE, set the handshake bit in the global variable to FALSE to change bit status to DI/DO: 0/0.

(c) In the case of DI/DO: 0/0
Set the exclusion bit to FALSE so that the PLC can send the REQ. This bit is used just before the processing part of each FB.

(d) In the case of DI/DO: 1/0
Set the exclusion bit to TRUE, since the PLC cannot send REQ.

The following output is outputted during the communication. It can also be used to confirm data tearing.

Data Name	Parameter Name	Data Type	Description
ACK data	COUNT_HEAD	USINT	Meaning: ACK counter at the beginning of data. This value is incremented by 1 when the bit state changes from the Robot controller. Input range: 1~255
	COUNT_TAIL	USINT	Meaning: ACK counter at the end of data. Copy the value of COUNT_HEAD. Input range: 1~255

2.6 ADMINISTRATIVE FB

These are the FBs that does not create a TP instruction. These FBs give the command to take immediate action on the Robot controller. This section describes the value to be entered in the REQ data to execute the command and the value of the response from the Robot controller in the ACK data.

2.6.1 FRC_Initialize

This FB initializes the Robot controller parameters for PLC Motion I/O Interface Function. The TP program that executes the command sent from the PLC is started. This FB should be executed first to communicate with the Robot controller. If this FB is not executed, application processing in the Robot controller will not be executed.

NOTE

When the robot controller starts the PLC Motion I/O Interface Function, you cannot execute a file backup or a system backup on the robot controller since the TP program is running and cannot be backed up. Use the FRC_Abort command to terminate the communication before executing a file backup of the robot controller.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 1
	POS_UTOOLNUM	USINT	Meaning: FB Major Version Input Value: 3
	POS_UFRAMENUM	USINT	Meaning: FB Minor Version Input Value: 0
	POS[0]	REAL	Meaning: Magic Number 1 Input Value: 70834.50
	POS[1]	REAL	Meaning: Magic Number 2 Input Value: 8192.25
	POS[2]	REAL	Meaning: Magic Number 3 Input Value: 1.25
	POS[3]	REAL	Meaning: Magic Number 4 Input Value: 99437.00
	POS[4]	REAL	Meaning: Magic Number 5 Input Value: 95892.50
	POS[5]	REAL	Meaning: Magic Number 6 Input Value: 19656.50
	POS[6]	REAL	Meaning: Magic Number 7 Input Value: 84301.50
	POS[7]	REAL	Meaning: Magic Number 8 Input Value: 2048.25
	POS[8]	REAL	Meaning: Magic Number 9 Input Value: 11688.50

Data Name	Parameter Name	Data Type	Description
ACK data	MOSTRM	BOOL	Meaning: PLC Motion I/O Interface Status Output Value: TRUE (OK)
	SRVRDY	BOOL	Meaning: Servo ready Output Value: TRUE (OK)
	REMOTE	BOOL	Meaning: Remote Setting Output Value: TRUE (In the case of the Remote setting is enabled on the Robot controller)
	SINGLESTEP	BOOL	Meaning: Single step mode Output Value: TRUE(In the case of the Single step mode is enabled on the Robot controller)
	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 1
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.2 FRC_Abort

This FB aborts the TP program created by the FRC_Initialize. This FB also deletes all commands data buffered in the Robot controller. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 25
ACK data	MOSTRM	BOOL	Meaning: PLC Motion I/O Interface Status Output Value: FALSE
	TPMODE	USINT	Meaning: TP program status Output Value: 1 (Aborted)
	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 25
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.3 FRC_Pause

This FB pauses the TP program in execution. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE. However, this command will not work when the TP program is already paused or stopped.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 3

Data Name	Parameter Name	Data Type	Description
ACK data	TPMODE	USINT	Meaning: TP program status Output Value: 2 (Paused)
	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 3
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.4 FRC_Continue

This FB continues the TP program in paused. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE. However, this FB should be called only after the controller has been reset after a fault. This FB will not work under the following conditions.

- Program Aborted
- Servo not ready
- TP is enabled

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 4
ACK data	TPMODE	USINT	Meaning: TP program status Output Value: 0 (Running)
	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 4
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.5 FRC_Reset

This FB resets the Robot controller alarms. The alarms that require user action cannot be reset with this FB. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 27
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 27
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.6 FRC_WriteUFrameData

This FB writes the user frame data to the specified User coordinates in the Robot controller. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 9
	POS_UFRAMENUM	USINT	Meaning: User frame number Input range: 1~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS	Array[0-8] of REAL	Meaning: User Coordinates to write
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 9
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.7 FRC_WriteUToolData

This FB writes the tool frame data to the specified Tool coordinates in the Robot controller. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 11
	POS_UTOOLNUM	USINT	Meaning: Tool frame number Input range: 1~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS	Array[0-8] of REAL	Meaning: Tool coordinates to write
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 11
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.8 FRC_ReadUFrameData

This FB reads the specified user frame data from the Robot controller. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 8
	VAL6	USINT	Meaning: User frame number to read Input range: 1~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 8
	ACK_POS	FRC_POS_T	Meaning: User Coordinates read from the Robot controller
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.9 FRC_ReadUToolData

This FB reads the specified tool frame data from the Robot controller. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 10
	VAL6	USINT	Meaning: Tool frame number to read Input range: 1~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 10
	ACK_POS	FRC_POS_T	Meaning: Tool Coordinates read from the Robot controller
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.10 FRC_SelectUFrameData

This FB switches the current user frame number to the selected number. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 52
	VAL6	USINT	Meaning: User frame number to read Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 52
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.11 FRC_SelectUToolData

This FB switches the current tool frame number to the selected number. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 51
	VAL6	USINT	Meaning: User tool number to read Input range: 1~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 51
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.12 FRC_TouchUp

This FB reads the position recorded with TP. After FRC_Initialize is success, this FB can always be executed, even if the MOSTRM in the ACK data is FALSE. See Chapter 3 for usage.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 16
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 16
	POS_IDX	USINT	Meaning: Position Index recorded from the Robot controller. Output range: 1~100
	ACK_POS	FRC_POS_T	Meaning: Position data recorded from the Robot controller
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.13 FRC_GroupSetOverride

This FB sets the program override.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 20
	VAL6	USINT	Meaning: Program Override to be set (%) Input range: 1~100
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 20
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.14 FRC_GroupReadActualPosition

This FB reads the current TCP position of the robot in world frame.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 14
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 14
	ACK_POS	FRC_POS_T	Meaning: TCP Position data
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.6.15 FRC_GroupReadJointPosition

This FB reads the current TCP position of the robot in joint representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 19
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 19
	ACK_POS	FRC_POS_T	Meaning: TCP Position data in joint representation
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7 TP INSTRUCTION FB

These are FBs that add TP instructions to the lines of the TP program. All FANUC motion FBs (FRC_MoveLinearAbsolute, etc.) only support buffer mode. Therefore, the motion commands sent from the PLC are buffered in the Robot controller. FANUC does not support any other mode, such as aborting mode, specified by the PLCopen standard.
This section describes the value to be entered in the REQ data to execute the command and the value of the response from the Robot controller in the ACK data.

NOTE

> If a CNT motion is buffered in the robot controller, the CNT motion will not be executed unless the PLC sends another motion. Therefore, the PLC should always send a FINE termination type motion before it calls FRC_ABORT.

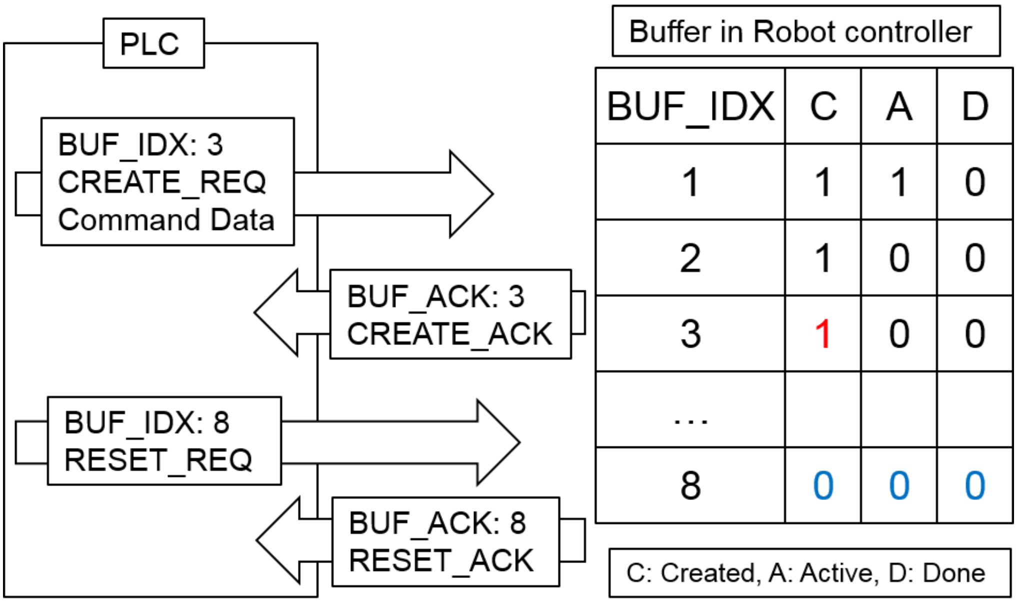
2.7.1 Execution Status of TP Instruction

The execution state of the TP line is managed in the Robot controller buffer. The PLC sends the following REQ to confirm the buffer status.

REQ Type	Description
CREATE_REQ	Request to confirm that TP line is created in the Robot controller.
ACTIVE_REQ	Request to confirm that TP line is executing in the Robot controller.
DONE_REQ	Request to confirm that TP line is completed in the Robot controller.
RESET_REQ	Request to empty the Robot controller’s buffer contents.

The Fig. 2.7.1(a) is the schematic of buffer in the Robot controller. The buffer is size 8 ring buffer containing the index and the status bits (Created, Active, Done) of the TP line.
The following example will be explained.

- The TP line corresponding to the index 8 has been executed
- The TP line corresponding to the index 1 is running
- The TP line corresponding to the index 2 is created



BUF_IDX	C	A	D
1	1	1	0
2	1	0	0
3	1	0	0
…			
8	0	0	0

Fig. 2.7.1(a) Buffer schematic

1. Request the creation of the TP instruction from the PLC. Specify the index 3 of the empty buffer, set the CREATE_REQ to TRUE and input the command data.

NOTE

The command data input is not required except for the CREATE_REQ. For the input data, see the explanation of each FB from section 2.7.2.

2. The Robot controller adds the TP instruction to the TP line, when receiving the CREATE_REQ. If the TP instruction is successfully added, the Robot controller sets the Created status bit at the specified index. Then, CREATE_ACK in the ACK data is set to TRUE and the requested index 3 is returned as the BUF_ACK.
3. Make a request from PLC to empty the buffer in the Robot controller. Specify the index 8 and set the RESET_REQ to TRUE.
4. When the Robot controller received the RESET_REQ while the Done bit is set in the buffer with the specified index, the Robot controller empty the buffer. If the Robot controller succeeds in emptying the buffer, the Robot controller sets the RESET_ACK in the ACK data to TRUE and returns the requested index 8 as the BUF_ACK.

NOTE

1 When the confirmation request of the status before the state of the specified index is TRUE, the Robot controller returns the status of the buffer at the time the request was received. For example, if the specified index is the active status, a new TP line is not created even if CREATE_REQ is set to TRUE. The ACTIVE_ACK is returned instead.

2 If the buffer is full, more line cannot be added. The buffer must be empty if you want to create more than 8 lines.

2.7.2 FRC_MoveLinearAbsolute

This FB executes linear motion to a destination position specified in Cartesian representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 201
	VAL2	UINT	Meaning: Program Speed (mm/sec) Input range: 1~2000 (Maximum value varies depending on the Robot model)
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)

Data Name	Parameter Name	Data Type	Description
REQ data	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in Cartesian representation
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 201
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.3 FRC_MoveLinearAbsoluteJ

This FB executes linear motion to a destination position specified in joint representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 209
	VAL2	UINT	Meaning: Program Speed (mm/sec) Input range: 1~2000 (Maximum value varies depending on the Robot model)
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)

Data Name	Parameter Name	Data Type	Description
REQ data	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in joint representation
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 209
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.4 FRC_MoveLinearRelative

This FB executes an incremental linear motion to a destination position specified in Cartesian representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 202
	VAL2	UINT	Meaning: Program Speed (mm/sec) Input range: 1~2000 (Maximum value varies depending on the Robot model)
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in Cartesian representation (Incremental)

Data Name	Parameter Name	Data Type	Description
REQ data	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 202
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.5 FRC_MoveDirectAbsolute

This FB executes a joint motion to a destination position specified in Cartesian representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 203
	VAL2	UINT	Meaning: Program Speed (%) Input range: 1~100
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in Cartesian representation
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 203
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.6 FRC_MoveDirectRelative

This FB executes incremental joint motion to a destination position specified in Cartesian representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 204
	VAL2	UINT	Meaning: Program Speed (%) Input range: 1~100
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in Cartesian representation (Incremental)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 204
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.7 FRC_MoveAxesAbsolute

This FB executes joint motion to a destination position specified in joint representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 207
	VAL2	UINT	Meaning: Program Speed (%) Input range: 1~100
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in joint representation
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 207
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.8 FRC_MoveAxesRelative

This FB executes incremental joint motion to a destination position specified in joint representation.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 208
	VAL2	UINT	Meaning: Program Speed (%) Input range: 1~100
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in joint representation (Incremental)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 208
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.9 FRC_MoveCircularAbsolute

This FB executes a circular motion to a destination position specified in Cartesian representation. This FB inputs two position data, destination position and via position. The 64 bytes of DI cannot be used to send destination position and via position at the same time, since the position data is 46 bytes. The 2 types of parameters are described below, since it is necessary to divide REQ.

The first send/receive parameters are as follows. Set NOT_REQFIN of REQ data to TRUE and send in divided.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 205
	VAL2	UINT	Meaning: Program Speed (%) Input range: 1~100
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in Cartesian representation
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
	NOT_REQFIN	BOOL	Meaning: Request incomplete bit Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: FALSE (Transmission is incomplete)

The second send/receive parameters are as follows. When the REQFIN_ACK of the ACK data is FALSE after sending the first data, send the second data.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 205
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the via position
	POS	Array[0~8] of REAL	Meaning: Via position in Cartesian representation
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 205
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (Transmission is complete)

2.7.10 FRC_MoveCircularRelative

This FB executes an incremental circular motion to a destination position specified in Cartesian representation. The 2 types of parameters are described below, since it is necessary to divide REQ same as FRC_MoveCircularAbsolute.

The first send/receive parameters are as follows. Set NOT_REQFIN of REQ data to TRUE and send in divided.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 206
	VAL2	UINT	Meaning: Program Speed (%) Input range: 1~100
	VAL6	USINT	Meaning: Termination Type Input Value: 1 (Fine), other than 1 (CNT)
	VAL7	USINT	Meaning: CNT value for CNT termination type. Ignored for FINE termination type. Input range: 0~100
	VAL1	USINT	Meaning: Acceleration value (%) Input range: 0~100
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the destination position
	POS	Array[0~8] of REAL	Meaning: Destination position in Cartesian representation (Incremental)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
	NOT_REQFIN	BOOL	Meaning: Request incomplete bit Input Value: TRUE
ACK data	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: FALSE (Transmission is incomplete)

The second send/receive parameters are as follows. When the REQFIN_ACK of the ACK data is FALSE after sending the first data, send the second data.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 206
	POS_UTOOLNUM	USINT	Meaning: Tool frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	POS_UFRAMENUM	USINT	Meaning: User frame number of the destination position Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)

Data Name	Parameter Name	Data Type	Description
REQ data	POS_HEAD	Array[0~7] of SINT	Meaning: Configuration of the via position
	POS	Array[0~8] of REAL	Meaning: Via position in Cartesian representation (Incremental)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 206
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (Transmission is complete)

2.7.11 FRC_TriggerAlongPath

This FB adds a Time Before, Time After, or Distance Before motion option to a Move FB. This FB can be used with FRC_AddMotionOption. It can be used in the order FRC_AddMotionOption, FRC_TriggerAlongPath, Move FB or FRC_TriggerAlongPath, FRC_AddMotionOption, Move FB.

It is necessary to divide REQ same as FRC_MoveCircularAbsolute. The first send/receive parameters are as follows. Set NOT_REQFIN of REQ data to TRUE and send in divided. Add Move FB or FRC_AddMotionOption as the send/receive parameter for the second and subsequent times.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 17
	VAL6	USINT	Meaning: Trigger Type Input Value: 1 (Time Before), 2 (Distance Before), 3 (Time After)
	VAL2	UINT	Meaning: Depends on the value of VAL6 (Trigger Type) 1: Set time of Time Before (ms) 2: Not used in Distance Before (VAL4 is used) 3 :Set time of Time After (ms) Input range: 0~30000 (In the case of the Time Before), 0~500 (In the case of the Time After)
	VAL4	UINT	Meaning: Set distance of Distance Before (0.1mm unit) Only used if VAL6value (Trigger Type) is 2 Input range: 0~9999
	VAL3	UINT	Meaning: Output Port Index of the Robot controller Input range: 1~8192 (The maximum value depends on the number of I/O setting in the Robot controller)
	POS_UTOOLNUM	USINT	Meaning: Output Port Type of the Robot controller Input Value: 1 (Digital Output), 2 (Robot Output)
	POS_UFRAMENUM	USINT	Meaning: Output port ON/OFF Input Value: other than 0(ON), 0(OFF)

Data Name	Parameter Name	Data Type	Description
REQ data	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
	NOT_REQFIN	BOOL	Meaning: Request incomplete bit Input Value: TRUE
ACK data	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: FALSE (Transmission is incomplete)

2.7.12 FRC_AddMotionOption

This FB adds Wrist Joint, Minimum Rotation, Offset Position Register or Offset Vision Register motion option to a Move FB. This FB can be used with FRC_TriggerAlongPath. It can be used in the order FRC_AddMotionOption, FRC_TriggerAlongPath, Move FB or FRC_TriggerAlongPath, FRC_AddMotionOption, Move FB.

It is necessary to divide REQ same as FRC_MoveCircularAbsolute. The first send/receive parameters are as follows. Set NOT_REQFIN of REQ data to TRUE and send in divided. Add Move FB or FRC_TriggerAlongPath as the send/receive parameter for the second and subsequent times.

NOTE

1 The Wrist Joint motion only applies to Linear and Circular motion and it is ignored when used in a joint motion instruction.

2 The MROT is available only when the R640 option is loaded on the Robot controller. Otherwise, the Robot controller will return an error for this instruction.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 18
	VAL1	USINT	Meaning: Enable Wrist Joint motion Input Value: 1/0 (Enable/Disable)
	VAL6	USINT	Meaning: Enable Minimum Rotation motion Input Value: 1/0 (Enable/Disable)
	POS_UTOOLNUM	USINT	Meaning: Enable Offset Position Register Input Value: 1/0 (Enable/Disable)
	VAL3	UINT	Meaning: Position Register number to use Input range: 1~200 (Maximum value can be increased by the Robot controller setting or option)
	POS_UFRAMENUM	USINT	Meaning: Enable Offset Vision Register Input Value: 1/0 (Enable/Disable)
	VAL4	UINT	Meaning: Vision Register number to use Input range: 1~10 (Maximum value can be increased by the Robot controller setting)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
	NOT_REQFIN	BOOL	Meaning: Request incomplete bit Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: FALSE (Transmission is incomplete)

2.7.13 FRC_SetUFrame

This FB sets the current active user frame number.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 103
	VAL6	USINT	Meaning: User frame number to set Input range: 0~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 103
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.14 FRC_SetUTool

This FB sets the current active user tool number.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 102
	VAL6	USINT	Meaning: User tool number to set Input range: 1~253 (The maximum value depends on the setting of the number of coordinates of the robot controller)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 102
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.15 FRC_SetPayload

This FB sets the current active payload number.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 105
	VAL6	USINT	Meaning: Payload number to set Input range: 1~10 (Maximum value can be increased by the Robot controller setting)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 105
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.16 FRC_Call

This FB executes call instruction.

NOTE

The DI/DO is big endian in the Robot controller. Use big endian when inputting 4 SINT data in the Real data type in IEEE754 format.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 106
	POS_UTOOLNUM	USINT	Meaning: Length of TP program Name Input range: 1~36
	POS	Array[0~8] of REAL	Meaning: The 36 byte data area for storing the TP program name character string Input Value: Input 4 SINT data in Real data type in IEEE754 format. Input the SINT data by referring to the ASCII code table.
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 106
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.17 FRC_WaitForDIN

This FB causes the TP program execution wait until the Robot controller DI reaches the specified value.

NOTE

If the Wait instruction is the last line of the TP program, the FB will not go into Done state and the program execution is paused. Add another TP instruction FB after the FRC_WaitForDIN to make sure the program can resume execution.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 101
	VAL3	UINT	Meaning: Index value of DI Input range: 1~8192 (The maximum value depends on the number of I/O setting in the Robot controller)
	VAL4	UINT	Meaning: ON/OFF of DI Input Value: Other than 2 (ON), 2 (OFF)
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE

Data Name	Parameter Name	Data Type	Description
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 101
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.7.18 FRC_WaitTime

This FB causes the TP program execution wait until the specified amount of time.

NOTE

If the Wait instruction is the last line of the TP program, the FB will not go into Done state and the TP program execution is paused. Add another TP instruction FB after the FRC_WaitTime to make sure the program can resume execution.

Data Name	Parameter Name	Data Type	Description
REQ data	CMD_ID	USINT	Meaning: Command ID Input Value: 104
	VAL4	UINT	Meaning: Wait time (10ms unit) Input range: 0~32767
	BUF_INDEX	USINT	Meaning: Buffer index to create the TP instruction Input Value: empty buffer index
	CREATE_REQ	BOOL	Meaning: Request to create the TP instruction Input Value: TRUE
ACK data	CMD_ID_ACK	USINT	Meaning: ACK for CMD_ID Output Value: 104
	BUF_ACK	USINT	Meaning: ACK for BUF_INDEX Output Value: The specified buffer index
	CREATE_ACK	BOOL	Meaning: ACK for CREATE_REQ Input Value: TRUE
	ERR_ID	UDINT	Meaning: Error ID in the Robot controller Output Value: 0 (Success), Other than 0 (see section 2.8)
	REQFIN_ACK	BOOL	Meaning: Request completion bit Output Value: TRUE (In the case of the other than FB which requires split transmission)

2.8 ERROR ID IN ROBOT CONTROLLER

The following are Error IDs in the Robot controller for this function. If the Error ID is not listed, operate the TP to check the alarm history.

Error ID	Meaning	Remedy
158000	PLC Motion Interface internal error	Cycle power the Robot controller.

Error ID	Meaning	Remedy
158001	Invalid User Tool Number	Make sure the input user tool number is within range.
158002	Invalid User Frame Number	Make sure the input user frame number is within range.
158003	Invalid Position Register	Make sure the input position register number is within range.
158004	Invalid Speed Override	Make sure the input override range is 1~100.
158005	Cannot execute TP program	Make sure the TP is disable and the Robot controller is in AUTO mode. Clear any error in the Robot controller and set the servo on before executing the TP program.
158006	Controller servo is off	Clear any error in the Robot controller.
158007	Teach Pendant is enabled.	Make sure the TP is disable and the Robot controller is in AUTO mode.
158008	PLC Interface is not running	The TP program created by FRC_Initialize is not running. Clear any error in the Robot controller and execute the FRC_Initialize again.
158009	Invalid AOI version	The FB version is not supported in the Robot controller. Make sure the FB version.
158010	Invalid parameter value	Clear any error in the Robot controller and execute the FRC_Initialize again.
158011	Stream Motion is not paused	The TP program is not paused and cannot be resumed. Clear any error in the Robot controller and execute the FRC_Initialize again.
158012	Cannot Resume TP Program	The TP program cannot be resumed. Clear any error in the Robot controller and execute the FRC_Initialize again.
158013	Cannot Reset Controller	There are Errors that cannot be cleared by FRC_Reset. Make sure the Robot controller status.
158014	Invalid PLC Command	The FB command ID is invalid. Make sure the FB in the PLC program.
158015	Administrative Command Fail	The administrative FB execution is time out. Clear any error in the Robot controller and execute the FRC_Initialize again.
158016	Invalid Controller state	The controller is in error state when PLC sent the administrative command. Make sure the robot controller is in valid state before sending administrative command.
158017	Please Cycle Power	The hot start is enabled. Cycle power the Robot controller since this function changes to COLD start.
158018	Invalid Payload Schedule.	Make sure PLC send the correct payload schedule number to the controller.
158019	Invalid motion option.	Invalid motion option in motion instruction, or the controller is not loaded with the required option. Make sure PLC send the correct motion option.
158020	Invalid Vision Register	Make sure the vision register number is within range.
158021	Invalid Program Name	Make sure the TP program exists.

3 ROBOT CONTROLLER SETUP

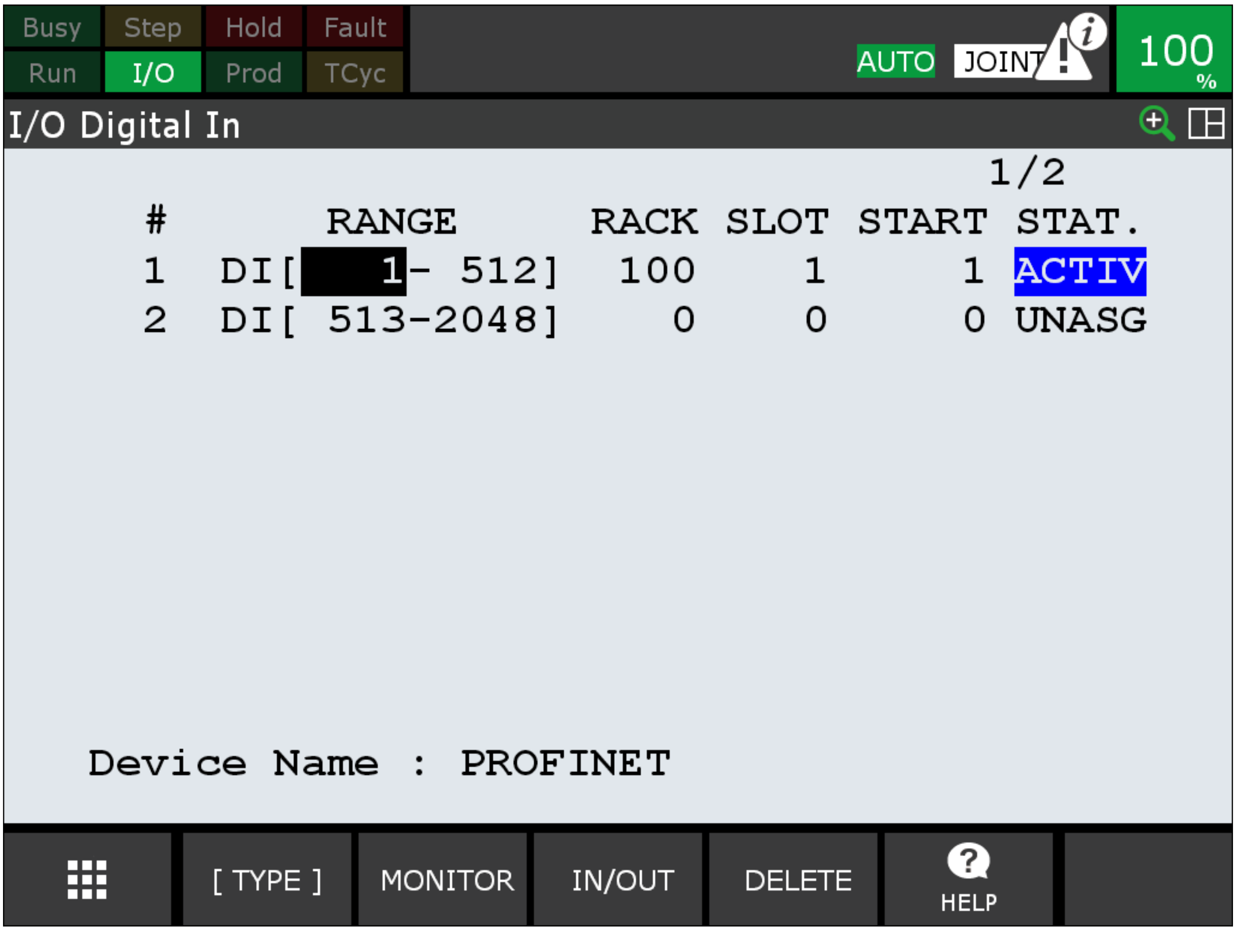
This section shows you how to set up the robot controller to use PLC Motion I/O Interface Function.

NOTE

Each DI/DO between PLC and robot controller requires 512 points for sending and receiving FB's command. In the future, the input/output size may be changed when a new AOI is added.

3.1 DI/DO SETUP

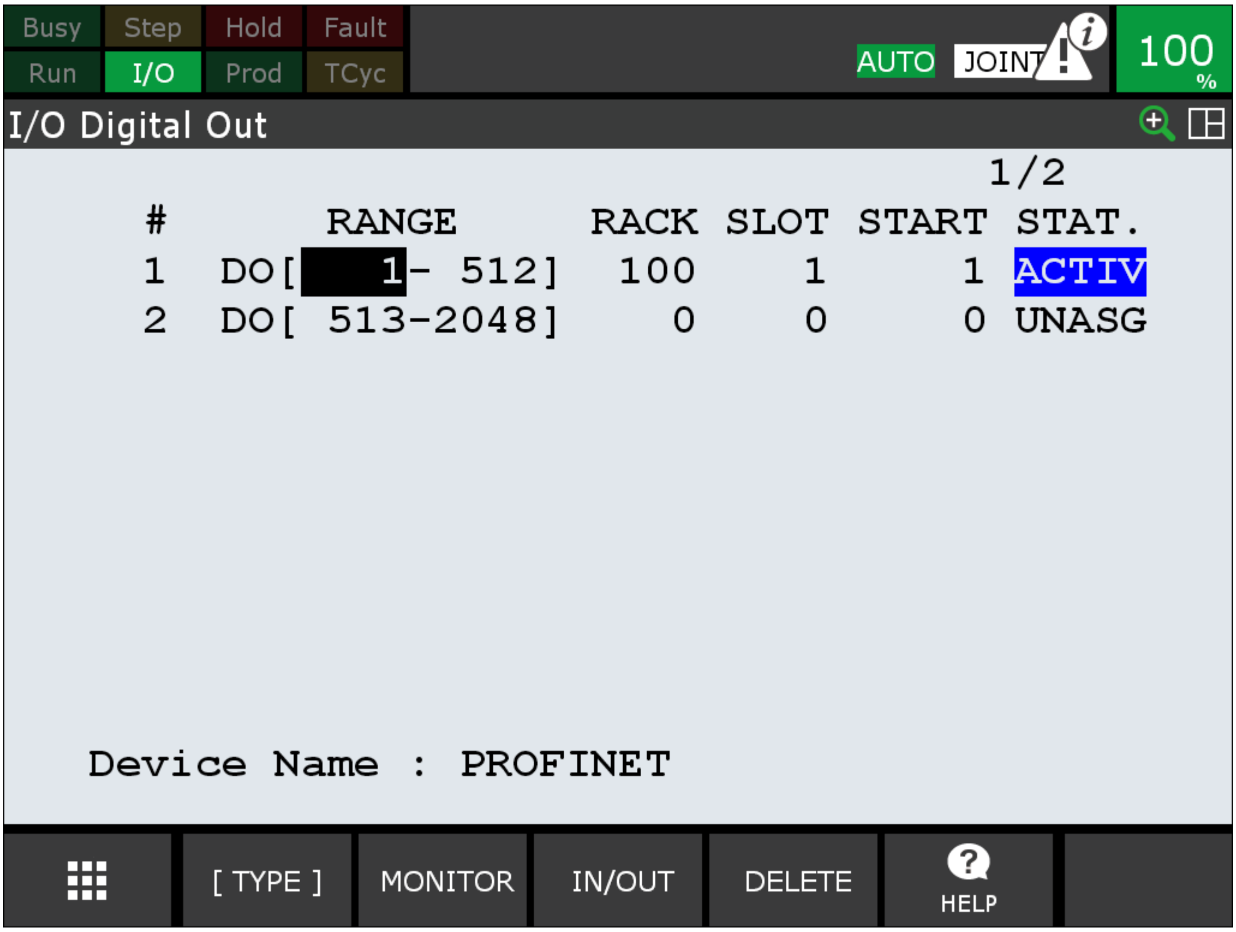
You can setup the DI/DO in the Digital I/O configuration screen. It does not matter to what number the start index is set, however the range needs 512 points. The rack depends on the type of I/O module. The slot and the start also depend on the I/O settings for PLC. Please refer to “OPERATOR’S MANUAL (Basic Function) (B-83284EN)” if you have any question regarding setting up the DI/DO. Cycle power after the DI/DO configuration is completed. The Fig. 3.1(a) shows the example of the DI/DO configuration after setup.



```
                                        1/2
    #       RANGE       RACK  SLOT  START  STAT.
    1   DI[    1-  512]  100     1      1  ACTIV
    2   DI[  513-2048]     0     0      0  UNASG

  Device Name  :  PROFINET
```

Fig. 3.1(a) DI Setup



I/O Digital Out

```
                                            1/2
     #        RANGE        RACK  SLOT  START  STAT.
     1    DO[    1-  512]   100     1      1  ACTIV
     2    DO[  513-2048]      0     0      0  UNASG



   Device Name  :  PROFINET
```

Fig. 3.1(b) DO Setup

3.2 SYSTEM VARIABLES SETUP

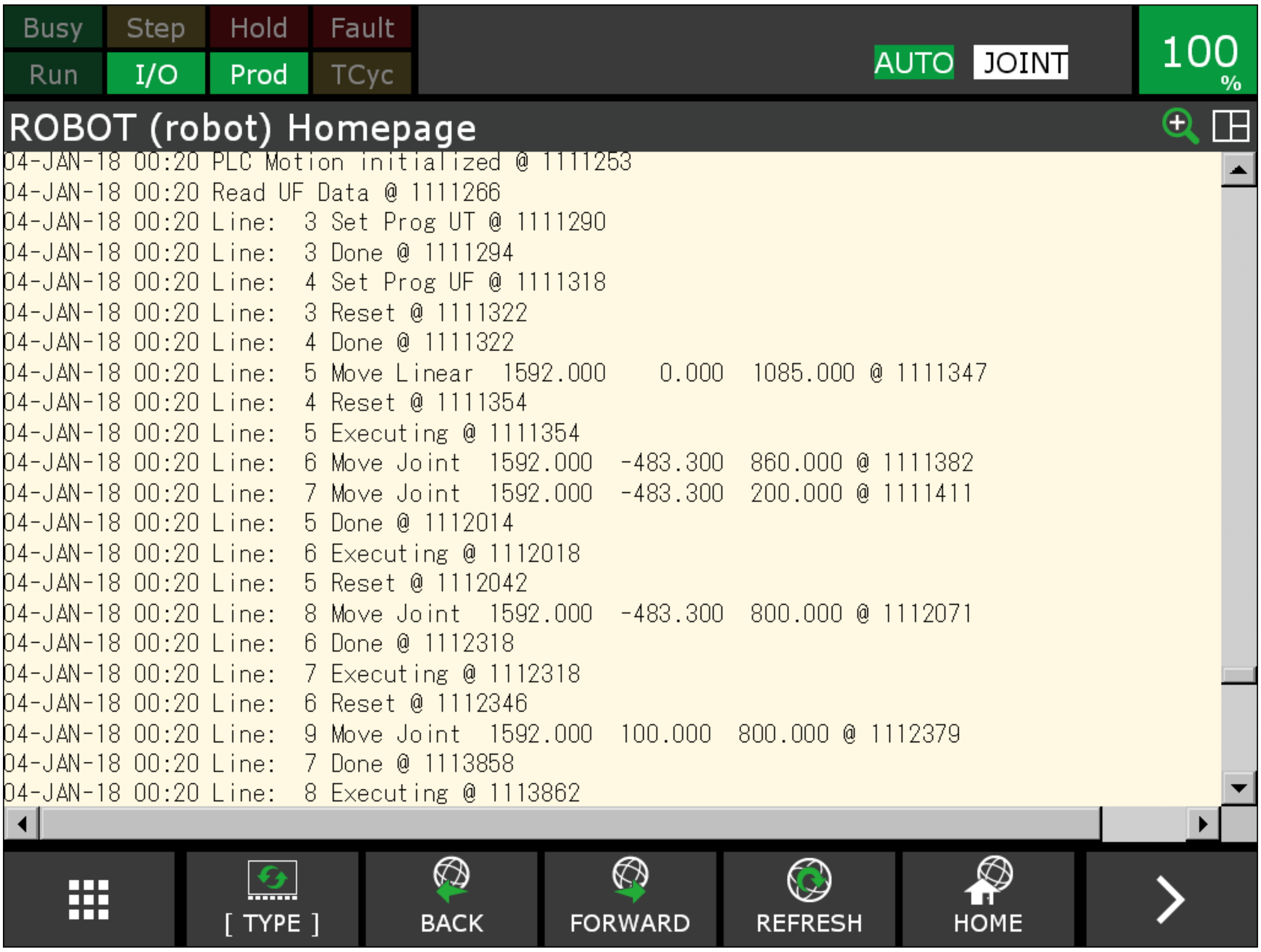
Set the system variable, $PLC_MOTION.$ENB to TRUE to use this function. Set the system variables, $PLC_MOTION.$ DI_STRT_PT and $PLC_MOTION.$ DI_STRT_PT. Input the DI/DO start index set in section 3.1 Cycle power after the configuration is completed.

NOTE

If either the start index value or the DI/DO assignment is incorrect, an alarm is posted and the PLC Motion Interface will not work. In this case, make the necessary change and cycle power.

4 PLC DATA LOG

In order to facilitate debugging and verification interactions between the robot controller and PLC, the PLC log is created to record all the interactions between the robot controller and the PLC. You can find the PLC_Log under the Browser menu.



```
04-JAN-18 00:20 PLC Motion initialized @ 1111253
04-JAN-18 00:20 Read UF Data @ 1111266
04-JAN-18 00:20 Line:  3 Set Prog UT @ 1111290
04-JAN-18 00:20 Line:  3 Done @ 1111294
04-JAN-18 00:20 Line:  4 Set Prog UF @ 1111318
04-JAN-18 00:20 Line:  3 Reset @ 1111322
04-JAN-18 00:20 Line:  4 Done @ 1111322
04-JAN-18 00:20 Line:  5 Move Linear   1592.000     0.000   1085.000 @ 1111347
04-JAN-18 00:20 Line:  4 Reset @ 1111354
04-JAN-18 00:20 Line:  5 Executing @ 1111354
04-JAN-18 00:20 Line:  6 Move Joint  1592.000  -483.300  860.000 @ 1111382
04-JAN-18 00:20 Line:  7 Move Joint  1592.000  -483.300  200.000 @ 1111411
04-JAN-18 00:20 Line:  5 Done @ 1112014
04-JAN-18 00:20 Line:  6 Executing @ 1112018
04-JAN-18 00:20 Line:  5 Reset @ 1112042
04-JAN-18 00:20 Line:  8 Move Joint  1592.000  -483.300  800.000 @ 1112071
04-JAN-18 00:20 Line:  6 Done @ 1112318
04-JAN-18 00:20 Line:  7 Executing @ 1112318
04-JAN-18 00:20 Line:  6 Reset @ 1112346
04-JAN-18 00:20 Line:  9 Move Joint  1592.000  100.000  800.000 @ 1112379
04-JAN-18 00:20 Line:  7 Done @ 1113858
04-JAN-18 00:20 Line:  8 Executing @ 1113862
```

Fig. 4(a) PLC log

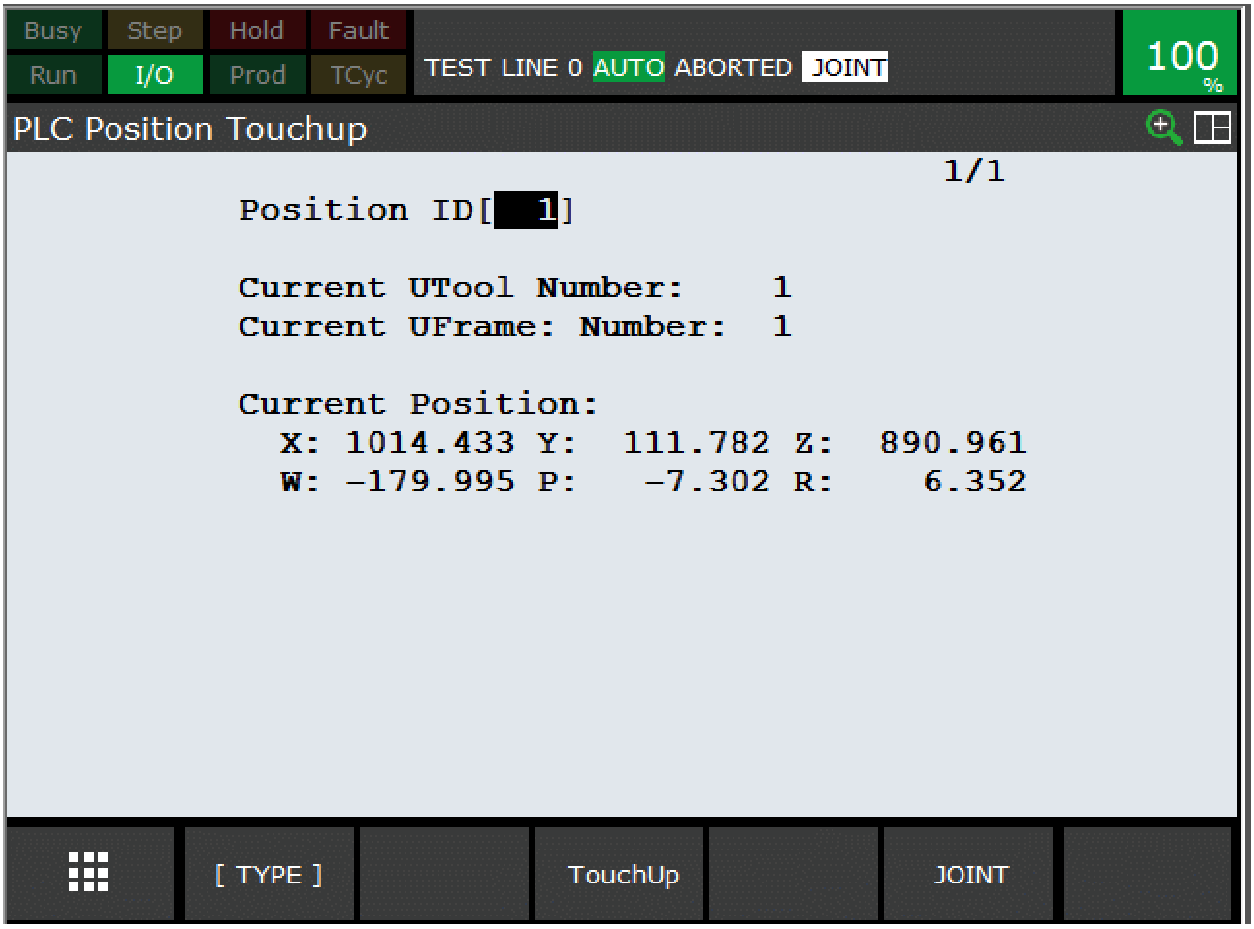
Each line in the PLC log has three sections. The first section is the calendar date and time when the robot controller received the PLC command. The second section is the PLC command or the motion execution status. The third section is the controller internal tick count. It starts with a '@' sign and gives a finer resolution in timing. If the second section starts without a line number, it indicates it is an administrative command. If the section starts with a line number, it can be either a PLC motion command or a TP execution status.

In the Fig. 4(a), the first line indicates that an FRC_Initialize administrative command was received, and the third line shows that an FRC_SetUTool command was received and appended to the TP program as Line 3. The fourth line indicates that the FRC_SetUtool command completed its execution.
The 8 line shows that an FRC_MoveLinearAbsolute command was received and writes the motion instruction as line 5 in the TP program at tick count 1111347. At 10 line, the motion line was executed by the robot controller. In the 11 and 12 line, while the Line 5 motion is being executed, the robot controller receives another two motion commands and add to the TP program as Line 6 and 7. The 13 line shows that the Line 5 motion is completed. The 14 line indicates that the following Line 6 motion was executed, and the 15 line indicates that the Line 5 motion data was cleared from the buffer.

This log does not support dynamic updates. That is, if the PLC log page is opened for viewing, the log is not going to be updated even though this function is active and PLC is sending commands to the robot controller. You have to leave this page and come back again to see new entries to the log.

5 PLC POSITION TOUCH UP

The PLC Position Touchup user interface is available to allow a PLC user to teach a robot position using the TP. The taught position is automatically transferred to the PLC so that the PLC can use this position in the PLC program. This screen is found under the Utility menu.



```
                                          1/1
         Position ID[   1]

         Current UTool Number:     1
         Current UFrame: Number:   1

         Current Position:
           X: 1014.433 Y:   111.782 Z:   890.961
           W: -179.995 P:    -7.302 R:     6.352
```

Fig. 5(a) PLC Position Touchup screen

You can input the position ID number (1~100) and then press F3, TouchUp, to transfer the current position to the PLC in Cartesian position data format. You will have to use the FRC_TouchUp in your PLC program to capture the touchup position data. Please follow the steps outlined below.

1 For an example, create the following PLC program.
2 Execute FRC_Initialize to start communication.
3 Enable your TP and perform a jog operation.
4 You can input the position ID number (1~100) and then press F3, TouchUp.
5 If FRC_TouchUp is executed, the position data is copied to ACK_POS in ACK data and the position ID number is copied to POS_IDX in ACK data. Prepare a FRC_POS_T array on the PLC side so that the data will be copied to the specified array index.
6 Repeat steps 3 to 5 to copy the position data.
7 After you finish position touchup, execute FRC_Abort to finish communication.

APPENDIX

A SAMPLE PLC LOGIC

This chapter describes the sample PLC logic.

A.1 INTERNAL STATE OF FB

There are as following states in each FB excluding the FB that require no Execute input. These states change according to the input to FB or ACK.

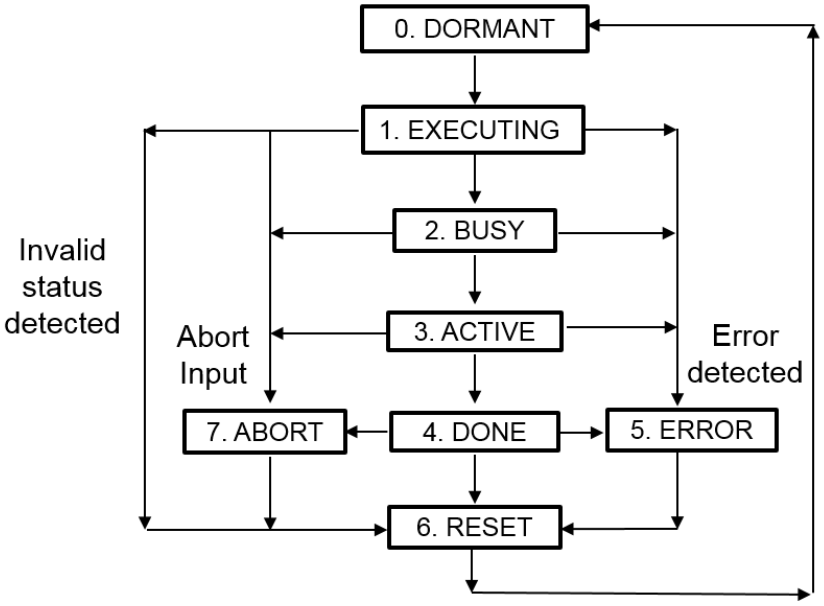


Fig. A.1(a) Internal state of FB

0 DORMANT: Wait until Execute input.
1 EXECUTING: The state that TP line is being created. Only the TP Move instruction FB exists. When Execute input is entered, the state is changed from DORMANT. If the invalid status is detected such as TP enabled, the state is changed to RESET.
2 BUSY: The state that command is being executed or the TP line has been created. In the case of other than the Move instruction FB, the state is changed from DORMANT when Execute is inputted. In the case of the Move instruction FB, the state is changed from EXECUTING when the CREATE_ACK is received.
3 ACTIVE: The state that TP line is being executed. Only the FB of the Move instruction exists. The state is changed from BUSY when the ACTIVE_ACK is received.
4 DONE: The state that the command or the TP line is completed. In the case of the Move instruction FB, the state is changed from ACTIVE when the DONE_ACK is received. In the case of other than the Move instruction FB, the state is changed from BUSY when the CMDID_ACK is received.
5 ERROR: The state is change from other than DORMANT or RESET when the ERR_ID is received.
6 RESET: Initialize the local variables and output in the FB.
7 ABORT: The state is change from other than DORMANT or RESET when the FRC_Abort output is received. Only the TP Move instruction FB exists.

A.2 VARIABLES AND DATA TYPE FOR PLC LOGIC

This section describes the variables and data type defined in the sample PLC logic.

A.2.1 Variables for TP Line Status Buffer

These variables are used in the TP Move instruction FB to manage the execution state of the TP line.

Member Name	Data Type	Description
BufSize	USINT	Buffer size (8)
BufHead	USINT	Head of ring buffer. Increase when TP Move instruction is created.
BufTail	USINT	Tail of ring buffer. Increase when TP Move instruction is done.
BufNumUsed	USINT	The number of used buffer. Increase when TP Move instruction is created. Decrease when buffer in TP is cleared.

A.2.2 Variables for Exclusive Processing

These variables are used to manage PLC logic.

Member Name	Data Type	Description
connErr	BOOL	Communication error flag with the Robot controller. It becomes TRUE in FRC_AOI_Status when a communication error with the Robot controller occurs. It prohibits to FRC_ReadGroupData etc. to execute.
connSuccess	BOOL	Communication success flag. It becomes TRUE when FRC_Initialize is done. It prohibits executing commands without executing FRC_Initialize first.
cmdAbort	BOOL	Abort flag. It becomes TRUE when FRC_Abort is done. It changes the state of TP Move instruction FB to ABORT.
cmdADMLock cmdADM2Lock	BOOL	Exclusion flag for ADM (Administrative) 1 and ADM2 command. It prevent other FBs from executing ADM1 and ADM2 command. ADM1: Commands related to execution such as FRC_Abort. ADM2: Commands that end in one execution, such as FRC_GroupSetOverride.
ONSADM3	BOOL	Occupancy flag for ADM3 command. It becomes TRUE while Execute input in ADM3 command FB. It is used to allow the ADM2 or ADM3 command to be executed even while the ADM3 command is being executed. ADM3: Command that keeps executing while Execute input (FRC_GroupReadActualPosition, FRC_GroupReadJointPosition)
cmdREQLock	BOOL	Exclusion flag for FBs. When the state of the handshake bit is write-protected, it prevent from executing FBs.
cmdCreateLock cmdActiveLock cmdDoneLock	BOOL	Exclusion flag for REQ of the TP move instruction. It prevents other FBs from executing each REQ.

A.2.3 FRC_BUF_T

This data type is used in variables for command buffer.

Member Name	Data Type	Description
cmd	USINT	ADM2 Command ID
data	USINT	Command data used in ADM2
POS	FRC_POS_T	Position data used in ADM2

A.2.4 Variables for command buffer

These variables are used to allow the ADM2 or ADM3 command to be executed even while the ADM3 command is being executed. The command data is stored in this buffer during ADM3 command execution.

Member Name	Data Type	Description
cmd_buf_siz	USINT	Command buffer size (4)
cmd_buf_idx	USINT	Command buffer index (0~3)
ReqBuf[4]	Array[0~3] of FRC_BUF_T	Structure for command buffer

A.3 COMMUNICATION FB

These FBs communicate the data between the Robot controller and the PLC or get the FB status. These FBs have no internal status.

A.3.1 FRC_WriteGroupData

This AOI writes to the PLC's digital outputs and sends messages to the robot controller. Use this AOI at the end of the program.

Priority of REQ

The REQ is sent for each execution cycle in the PLC program. Here, the transmission priority is determined according to REQ. This priority is managed by FRC_WriteGroupData before copying REQ to the DI area.
The data to be copied to the DI area is determined according to the following command priority.

1. ADM1 (Commands related to execution)
2. ADM2 (Commands that end in one execution)
3. Create REQ of the TP Instruction
4. ADM3 (Command that keeps executing while Execute input)
5. REQ other than Create of the TP Instruction

Process

Check the bit state for handshake, and if the state is writable, copy the REQ with higher priority to the DI area. See section 2.5.1 for handshake processing.

Output Error if the connection is lost.

(a) In the case of DI/DO: 1/0 or 0/1
Do not copy the variable for REQ into the DI area (Wait for the Robot controller to change DI/DO: 1/1 or 0/0)

(b) In the case of DI/DO: 1/1

Set all variables for REQ except the counter to 0 (The bit status becomes DI/DO: 0/1, since the bit on the PLC side also becomes FALSE)
The COUNT_HEAD of REQ is incremented by 1 and the same value is copied to the COUNT_TAIL
Copy the variable for REQ to the DI area and clear REQ other than the counter

(c) In the case of DI/DO: 0/0
(i) In the case of the executing of ADM1 or ADM2
Clear REQ (Create, Active, Done, Reset)
Set the handshake bit in the PLC to TRUE
The COUNT_HEAD of REQ is incremented by 1 and the same value is copied to the COUNT_TAIL
Copy the variables for REQ into the DI area
(ii) In the case of the command buffer exist
If there is REQ other than Create of the TP instruction, send the TP instruction REQ and the ADM2 or ADM3 REQ alternately every cycle.

NOTE
ADM3 keeps sending the REQ, so PLC cannot send the other command. Therefore, a buffer for the command is prepared. The command is inputted this buffer when the other of ADM2 or ADM3 is executed while ADM3 is being executed.

(iii) In the case of the ADM3
Copy the command buffer data to the global variables in the PLC.
Set the handshake bit in the PLC to TRUE
Clear REQ (Create, Active, Done, Reset)
The COUNT_HEAD of REQ is incremented by 1 and the same value is copied to the COUNT_TAIL
Copy the variables for REQ into the DI area
Clear the command buffer data.
(iv) In the case of the REQ of the TP instruction
Set the Command ID to 0 if not Create REQ
The COUNT_HEAD of REQ is incremented by 1 and the same value is copied to the COUNT_TAIL
Copy the variables for REQ into the DI area

Parameter

Parameter Name	Data Type	Usage	Description
Error	BOOL	Output	Connection to the Robot controller is lost
ErrorID	UDINT	Output	1: Connection to the Robot controller has been lost.

A.3.2 FRC_ReadGroupData

This FB reads the response sent from the Robot controller to the PLC as DO. Use this FB at the beginning of the PLC program to copy the latest data from the DO to the structure in the PLC.

Process

Output Error if the connection is lost.

(a) In the case of DI/DO: 1/1
Copy data in the DO area to the variable for ACK.

(b) In the case of DI/DO: 0/1

If the handshake bit in the DO area is FALSE, set the handshake bit in the global variable to FALSE to change bit status to DI/DO: 0/0.

(c) In the case of DI/DO: 0/0
Set the exclusion bit to FALSE so that the PLC can send the REQ. This bit is used just before the processing part of each FB.

(d) In the case of DI/DO: 1/0
Set the exclusion bit to TRUE, since the PLC cannot send REQ.

Parameter

Parameter Name	Data Type	Usage	Description
Error	BOOL	Output	Connection to the Robot controller is lost
ErrorID	UDINT	Output	1: Connection to the Robot controller has been lost.

A.3.3 FRC_AOI_Status

This FB checks the connection status and reports internal errors that will cause malfunctioning of the FBs. It is necessary to write it in each vendor's own function to get the status of the PLC.

Process

If the variables for TP line status buffer have invalid value, an internal parameter error is output.
If the communication status of the PLC is checked and there is an error, the module error and the communication error are output.

Parameter

Parameter Name	Data Type	Usage	Description
RobotConnection	HW_DEVICE	InOut	Connection module to the robot controller
COMM_ERROR	BOOL	Output	Error in the Connection module
AOI_INTERNAL_ERROR	BOOL	Output	Internal parameter error used by the FBs
CONNECTION_FAULT	BOOL	Output	PLC is not communicating with robot controller

A.4 ADMINISTRATIVE FBS

These are the FBs that does not create a TP instruction. There are three types defined: ADM1, ADM2, and ADM3.

A.4.1 Common Input/Output Parameters

The common input/output parameters in the ADM FB are shown below. These are omitted in the parameter description of each ADM FB, since these are common in each ADM FB.

Parameter Name	Data Type	Usage	Description
AxesGroup	USINT	Input	Robot group number. Not used in this version.
Execute	BOOL	Input	The FB is executed when a rising edge of the signal is detected
Reset	BOOL	Input	Input to reset parameters of this FB
Done	BOOL	Output	Command execution has been completed
Busy	BOOL	Output	Command is sending to the robot controller
Error	BOOL	Output	Error occurred during execution of this FB

Parameter Name	Data Type	Usage	Description
ErrorID	UDINT	Output	Other than 0, 1,2: Error ID reported by the robot controller 1: Connection to the Robot controller has been lost. 2: The Index of the position recorded exceeds the size of the position array.

NOTE

1 Only one of the outputs; Busy, Done, or Error can be TRUE at one time.
2 Done and Error outputs are set to TRUE for a minimum of one scan time.
3 Done and Error can be reset only after a falling edge is detected in Execute input.

A.4.2 ADM1

These FBs are classified as commands related to execution.

FRC_Initialize

This FB initializes the PLC Motion Interface in the robot controller. It starts the TP program that executes commands sent from PLC.
The following processing is performed depending on the state.

0 DORMANT state
Definition of constants
If Reset is inputted in a state other than DORMANT/RESET, the state changes to RESET state.
If Execute is inputted, the state changes to BUSY state.

1 BUSY state
Output resetAOIs. (The resetAOIs is treated as a Reset input of the other FB. The state changes to the RESET state to initialize local variables and outputs, when the other FB is in the state other than DORMANT.)
Wait three execution cycles in the PLC program. (Wait until the initialization process of the other FB is completed.)
Initialization of the global variables
Input REQ data to global variables
Input the occupancy flag and the exclusion flag, and output Busy.
The state changes to DONE state.

2 DONE state
If there is Command ID ACK, set the exclusion flag of ADM1 to FALSE.
(It allows the other ADM1 commands to be executed.)
(a) If there is no Error, input the communication success flag and output Done. If there is no Execute, the state changes to the RESET state.
(b) If there is Error, output Error and ErrorID, and the state changes to ERROR state.

3 ERROR state
If there is no Execute input, the state changes to the RESET state.

4 RESET state
Initialize the local variables and the output in this FB.

In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
resetAOIs	BOOL	InOut	This is a parameter passed to the Reset input of all TP instruction FBs and ADM FBs used in the PLC program. The other FB uses this reset input to reset any active FB.

FRC_Abort

This FB aborts the TP program created by the FRC_Initialize. This FB also outputs cmdAbort that changes the active TP Move instruction FB to ABORT state.
The process in DORMANT/ERROR/RESET state is the same as the process described in FRC_Initialize.

1. BUSY state
   If there is no other ADM1 command and there is a communication success flag of FRC_Initialize, input the Command ID to the global variables for REQ. Set the occupation flag and exclusion flag, and output Busy, and the state changes to DONE state.

2. DONE state
   (a) If Command ID ACK is received, set the exclusion flag of ADM1 to FALSE, and output cmdAbort. If there is no Error, Output Done. If there is no Execute, the state changes to the RESET state.
   (b) If Command ID ACK is not received or there is Error, Output Error and ErrorID, and the state changes to ERROR state.

The parameters are the same as the parameters described in section A.4.1.

FRC_Pause

This FB pauses the TP program in execution.
The process in DORMANT/BUSY/ERROR/RESET state is the same as the process described in FRC_Abort.

2. DONE state
   (a) If Command ID ACK is received, set the exclusion flag of ADM1 to FALSE.
   If there is no Error, Output Done. If there is no Execute, the state changes to the RESET state.
   (b) If Command ID ACK is not received or there is Error, Output Error and ErrorID, and the state changes to ERROR state.

The parameters are the same as the parameters described in section A.4.1.

FRC_Continue

This FB continues the TP program in paused.
The process in state is the same as the process described in FRC_Pause.
The parameters are the same as the parameters described in section A.4.1.

FRC_Reset

This FB resets the Robot controller alarms.
The process in state is the same as the process described in FRC_Pause.
The parameters are the same as the parameters described in section A.4.1.

A.4.3 ADM2

These FBs are commands that end in one execution. These FBs use the command buffer for using with ADM3 at the same time.
The process in DORMANT/ERROR/RESET state is the same as the process described in FRC_Initialize.

1. BUSY state
   In the case of the no exclusion flag for ADM1/ADM2 of other FB
   (a) If the occupancy flag for ADM3 is TRUE, input the Command ID and Input parameters to the global variables for command buffer. Increase the command buffer index. Output Busy, and the state changes to BUSY state.
   (b) If the occupancy flag for ADM3 is FALSE, input the Command ID and Input parameters to the global variables for REQ. Set the exclusion flag for ADM2 to TRUE, and output Busy, and the state changes to DONE state.

2. DONE state
   (a) If Command ID ACK is received, set the exclusion flag of ADM2 to FALSE, and output.
      (i) If there is no Error, output Done. If there is no Execute, the state changes to the RESET state.
      (ii) If there is Error, output Done. Output Error and ErrorID, and the state changes to ERROR state.
   (b) In the case of the no Command ID ACK
   In the case of the no exclusion flag for ADM1/ADM2 of other FB
      (i) If the occupancy flag for ADM3 is TRUE, input the Command ID and Input parameters to the global variables for command buffer. Increase the command buffer index.
      (ii) If the occupancy flag for ADM3 is FALSE, input the Command ID and Input parameters to the global variables for REQ. Set the exclusion flag for ADM2 to TRUE.

FRC_WriteUFrameData

This FB writes the user frame data to the specified User coordinates in the Robot controller.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
UserFrameNum	USINT	Input	User frame number to be written
UserFrameData	FRC_POS_T	InOut	User frame value to be set

FRC_WriteUToolData

This FB writes the tool frame data to the specified Tool coordinates in the Robot controller.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
ToolFrameNum	USINT	Input	Tool frame number to be written
ToolFrameData	FRC_POS_T	InOut	Tool frame value to be set

FRC_ReadUFrameData

This FB reads the specified user frame data from the Robot controller.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
UserFrameNum	USINT	Input	User frame number to be read
UserFrameData	FRC_POS_T	InOut	User frame data read from the robot controller

FRC_ReadUToolData

This FB reads the specified tool frame data from the Robot controller.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
ToolFrameNum	USINT	Input	Tool frame number to be read

Parameter Name	Data Type	Usage	Description
ToolFrameData	FRC_POS_T	InOut	Tool frame data read from the robot controller

FRC_SelectUFrameData

This FB switches the current user frame number to the selected number.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
UserFrameNum	USINT	Input	User frame number to be written

FRC_SelectUToolData

This FB switches the current tool frame number to the selected number.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
ToolFrameNum	USINT	Input	Tool frame number to be written

FRC_TouchUp

This FB reads the position recorded with TP. See Chapter 5 for usage.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
PR	Array[0~9] of FRC_POS_T	InOut	An array of position data. In the sample, the size is 10.
PositionIndex	USINT	Output	Index of the recorded position (1~100)

FRC_GroupSetOverride

This FB sets the program override.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
Override	USINT	Input	Program override to be set (%)

A.4.4 ADM3

These FBs are commands that keeps executing while Execute input. These FBs use the command buffer for using with other FBs at the same time.
The process in DORMANT/ERROR/RESET state is the same as the process described in FRC_Initialize.

1. BUSY state
   (a) In the case of the Execute Input
   If there is no Create REQ or exclusion flag for ADM1/ADM2, and there is communication success flag of FRC_Initialize, input the Command ID to the global variables for command buffer. Increase the command buffer index. Set the occupancy flag for ADM3 to TRUE, and output Busy, and the state changes to BUSY state.
   (b) In the case of the no Execute Input
   Set the occupancy flag for ADM3 to FALSE, and the state changes to RESET state.

2. DONE state
   (a) In the case of the Execute Input

(i) In the case of the no Command ID ACK
If there is no Error, output contents of ACK_POS and Valid.
If there is Error, output Done. Output Error and ErrorID, and the state changes to ERROR state.

(ii) In the case of the no Command ID ACK
If there is no Create REQ or exclusion flag for ADM1/ADM2, and there is communication success flag of FRC_Initialize, input the Command ID to the global variables for command buffer. Increase the command buffer index. Set the occupancy flag for ADM3 to TRUE.

(b) In the case of the no Execute Input
Set the occupancy flag for ADM3 to FALSE, and the state changes to RESET state.

FRC_GroupReadActualPosition

This FB reads the current TCP position of the robot in world frame.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
RobotPos	FRC_POS_T	InOut	Current TCP position of the Robot
ToolFrame	USINT	Output	Current tool frame that was used when the TCP position was recorded
X	REAL	Output	X of TCP position
Y	REAL	Output	Y of TCP position
Z	REAL	Output	Z of TCP position
W	REAL	Output	W of TCP position
P	REAL	Output	P of TCP position
R	REAL	Output	R of TCP position
E1	REAL	Output	Ext axis 1 position
E2	REAL	Output	Ext axis 2 position
E3	REAL	Output	Ext axis 3 position
Turn4	SINT	Output	Axis 4 turn no.
Turn5	SINT	Output	Axis 5 turn no.
Turn6	SINT	Output	Axis 6 turn no.
Front	SINT	Output	Front/back
Up	SINT	Output	Up/down
Left	SINT	Output	Left/right
Flip	SINT	Output	Flip/non-flip
Valid	BOOL	Output	All outputs of this FB are valid

FRC_GroupReadJointPosition

This FB reads the current TCP position of the robot in joint representation.
In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
RobotPos	FRC_POS_T	InOut	Current TCP position in Joint representation
J1	REAL	Output	Axis 1 position
J2	REAL	Output	Axis 2 position
J3	REAL	Output	Axis 3 position
J4	REAL	Output	Axis 4 position
J5	REAL	Output	Axis 5 position
J6	REAL	Output	Axis 6 position
E1	REAL	Output	Ext axis 1 position
E2	REAL	Output	Ext axis 2 position

Parameter Name	Data Type	Usage	Description
E3	REAL	Output	Ext axis 3 position
Valid	BOOL	Output	All outputs of this FB are valid

A.4.5 Other Instruction

These FBs have no Command ID and internal state. These FBs output the cyclic output of ACK data in section 2.5.2.

FRC_GroupReadActualSpeed

This FB reads the current TCP speed of the Robot.

If the communication error is output from FRC_Status, output Error.
(a) In the case of the Execute Input, output TCP_SPEED of ACK data and Valid.
(b) In the case of the no Execute Input, set the all output to 0.

In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
TCP_Speed	REAL	Output	Current TCP speed
Valid	BOOL	Output	Output of this FB is valid

FRC_Status

This FB reads the status in the Robot controller.

If the communication error is output from FRC_Status, output Error.
Read the corresponding ACK data as follows and output them as parameters.
(a) If MOSTRM of ACK data is TRUE, output STRM_RDY.
(b) If TPMODE of ACK data is 1, output PX_ABORTED. If value is 2, output PX_PAUSED.
(c) If SRVRDY of ACK data is TRUE, output SERVO_RDY.
(d) If TPENB of ACK data is TRUE, output TP_ENABLED.
(e) If REMOTE of ACK data is TRUE, output REMOTE_CFG.
(f) If SINGLESTEP of ACK data is TRUE, output SINGLE_STEP.
(g) If any of the following is not true, output OK_to_run.
- MOSTRM of ACK data is FALSE
- SRVRDY of ACK data is FALSE
- TPENB of ACK data is TRUE
- TPMODE of ACK data is not 0

In addition to the parameters described in section A.4.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
OK_to_run	BOOL	Output	OK to run commands
PX_PAUSED	BOOL	Output	Program execution has been paused
PX_ABORTED	BOOL	Output	Program execution has been aborted
STREAM_RDY	BOOL	Output	Ready to stream commands from PLC
TP_ENABLED	BOOL	Output	TP is enabled
SERVO_RDY	BOOL	Output	Servo is ready
REMOTE_CFG	BOOL	Output	Remote configuration is set
SINGLE_STEP	BOOL	Output	Single step is enabled
Valid	BOOL	Output	All outputs of this FB are valid

A.5 TP INSTRUCTION FB

These are FBs that add TP instructions to the lines of the TP program.

A.5.1 Common input/output parameters

The common input/output parameters in the TP instruction FB are shown below. These are omitted in the parameter description of each TP instruction FB, since these are common in each TP instruction FB.

Parameter Name	Data Type	Usage	Description
AxesGroup	USINT	Input	Robot group number. Not used in this version.
Execute	BOOL	Input	The FB is executed when a rising edge of the signal is detected
Reset	BOOL	Input	Input to reset parameters of this FB
Done	BOOL	Output	The motion has been executed successfully
Busy	BOOL	Output	TP line was created corresponding to this FB
CommandAborted	BOOL	Output	Execution of this FB is aborted
Error	BOOL	Output	Error occurred during execution of this FB.
ErrorID	UDINT	Output	Error ID reported by the Robot controller

NOTE

1 Only one of the outputs; Busy, Active, Done, Error, or CommandAborted can be TRUE at one time.
2 Done, Error, and CommandAborted outputs are set to TRUE for a minimum of one scan time.
3 Done, Error, and CommandAborted can be reset only after a falling edge is detected in Execute input.

A.5.2 Move Instruction (Linear/Joint)

These are the FBs that execute linear or joint motion. The state changes according to the execution status of the TP line.

About REQ

The execution state of the TP line is managed in the Robot controller buffer. The PLC sends the REQ to confirm the status of the buffer in the Robot controller. For more the buffer, see section 2.7.1.
REQ transmission priority is as follows in the sample program.

1. Create (Unless instructions are created, processing will not proceed)
2. Reset (Since the move instruction cannot be created without releasing the buffer)
3. Done
4. Active

Fig. A.5.2(a) is an example of REQ transmission. If you create 3 move instructions with the buffer size of 2, the REQ will be in the following order

1. If CREATE_REQ of FB1 is set to TRUE, the FB1 instruction is created in the TP line1.
2. If CREATE_REQ of FB2 is set to TRUE, the FB2 instruction is created in the TP line2.
3. If ACTIVE_REQ of FB1 is set to TRUE, ACTIVE_ACK is returned since the TP line1 is executing.
4. If DONE_REQ of FB1 is set to TRUE, ACTIVE_ACK is returned since the TP line1 is still executing. Next, if DONE_REQ of FB1 is set to TRUE, DONE_ACK is returned since the TP line1 is done.
5. If RESET_REQ of FB1 is set to TRUE, the buffer of TP line1 is reset, and RESET_ACK is returned.

6. If CREATE_REQ of FB3 is set to TRUE, the FB3 instruction is created in the TP line3.
7. If ACTIVE_REQ of FB2 is set to TRUE, DONE_ACK is returned instead of ACTIVE_ACK since the TP line2 is already done.
8. Skip DONE_REQ since DONE_ACK of the FB2 is returned. If RESET_REQ of FB2 is set to TRUE, the buffer of TP line2 is reset, and RESET_ACK is returned.
9. If ACTIVE_REQ of FB3 is set to TRUE, ACTIVE_ACK is returned since the TP line3 is executing.
10. If DONE_REQ of FB3 is set to TRUE, DONE_ACK is returned since the TP line3 is done.
11. If RESET_REQ of FB3 is set to TRUE, the buffer of TP line3 is reset, and RESET_ACK is returned.

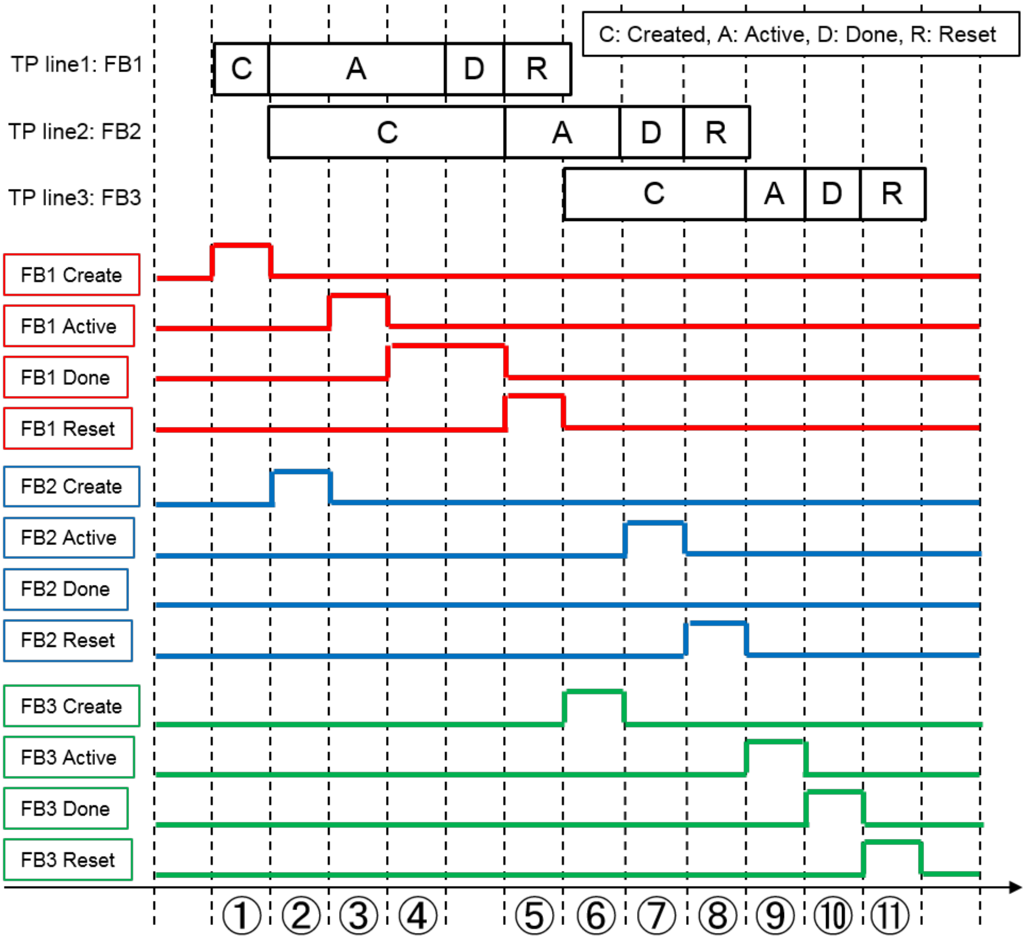


Fig. A.5.2(a) Example of REQ transmission

State process

0. DORMANT state

   Definition of constants

   In the case of the state other than DORMANT/RESET/ABORT

   (a) The state changes to RESET state, when Reset is inputted by resetAOIs of FRC_Initialize.

   (b) The state changes to ABORT state when cmdAbort is inputted from FRC_Abort.

   (c) If Execute is inputted in the DORMANT state, the input parameters are inputted to the local variables and the state changes to EXECUTING state

   If the state of the handshake bit is not writable state, state processing cannot be executed.

1. EXECUTING state

   (a) If the Robot controller status is invalid such as TP enabled, the state changes to RESET state.

(b) If the other FB have not been sent Create REQ, set the variables for Buffer head and input Create REQ (Command ID, Input parameters, CREATE_REQ, and buffer Index) to the global variables of the REQ.

(c) If Create/Active/Done ACK and corresponding buffer index to ACK are received, output Busy and the state changes to BUSY state. If Done ACK is received, the processing in BUSY state is further skipped.

(d) If Create/Active/Done ACK and corresponding buffer index to ACK are not received, send again by inputting Create REQ to global variables for REQ again.

(e) If Error ID ACK and corresponding buffer index to ACK are received, output Error and ErrorID and the state changes to ERROR state.

NOTE

When the PLC sends the Create REQ of FB that immediately finishes execution such as FRC_SetUframe, the TP line may already be Active or Done. Therefore, even if the FB is in the EXECUTING state, the FB assumes that Active or Done ACK is returned after skipping Create ACK. Even if the movement is short, the execution will be completed immediately.

2. BUSY state

If Create/Done REQ of the other FB has been sent, the process is not executed in BUSY state.

(a) If Active REQ has not been sent, input ACTIVE_REQ and buffer Index to the global variables of the REQ.

(b) If Active/Done ACK and corresponding buffer index to ACK are received, output Active and the state changes to ACTIVE state.

(c) If Active/Done ACK and corresponding buffer index to ACK are not received, retry to send Active REQ.

(d) If Error ID ACK and corresponding buffer index to ACK are received, output Error and ErrorID and the state changes to ERROR state.

3. ACTIVE state

If Create REQ of the other FB has been sent, the process is not executed in ACTIVE state.

(a) If Done ACK and corresponding buffer index to ACK are received, input RESET_REQ and buffer Index to the global variables of the REQ.

(b) If Done ACK and corresponding buffer index to ACK are not received, input DONE_REQ and buffer Index to the global variables of the REQ.

(c) If Reset REQ has been sent and Reset ACK and corresponding buffer index to ACK are received, the state changes to DONE state.

(d) If Reset REQ has been sent and Reset ACK and corresponding buffer index to ACK are not received, retry to send Reset REQ.

(e) If Error ID ACK and corresponding buffer index to ACK are received, output Error and ErrorID and the state changes to ERROR state.

4. DONE state

Output Done. If there is no Execute input, set the variables for Buffer tail and the state changes to the RESET state.

5. ERROR state

If Reset ACK and corresponding buffer index to ACK are received, set the variables for Buffer tail and the state changes to the RESET state.

6. RESET state

Initialize the local variables in this FB.

7. ABORT state
   If there is no Execute input, the state changes to the RESET state.

FRC_MoveLinearAbsolute

This FB executes linear motion to a destination position specified in Cartesian representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position in Cartesian representation
Velocity	UINT	Input	Program speed (mm/sec)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveLinearAbsoluteJ

This FB executes linear motion to a destination position specified in joint representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position in Cartesian representation
Velocity	UINT	Input	Program speed (mm/sec)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveLinearRelative

This FB executes an incremental linear motion to a destination position specified in Cartesian representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position (Incremental) in Cartesian representation
Velocity	UINT	Input	Program speed (mm/sec)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveDirectAbsolute

This FB executes a joint motion to a destination position specified in Cartesian representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position in Cartesian representation
Velocity	UINT	Input	Program speed (%)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveDirectRelative

This FB executes incremental joint motion to a destination position specified in Cartesian representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position (incremental) in Cartesian representation
Velocity	UINT	Input	Program speed (%)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveAxesAbsolute

This FB executes joint motion to a destination position specified in joint representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position in Joint representation
Velocity	UINT	Input	Program speed (%)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveAxesRelative

This FB executes incremental joint motion to a destination position specified in joint representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position (incremental) in Joint representation
Velocity	UINT	Input	Program speed (%)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type

Parameter Name	Data Type	Usage	Description
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

A.5.3 Move Instruction (Circular)

These are the FBs that execute circular motion. These FBs input two position data, destination position and via position. The 64 bytes of DI cannot be used to send destination position and via position at the same time, since the position data is 46 bytes. This section describes the process of divided transmission since it is necessary to divide REQ.

The process in state other than EXECUTING state is the same as the process described in section A.5.2.

1. EXECUTING state
   (a) If the Robot controller status is invalid such as TP enabled, the state changes to RESET state.
   (b) If the other FB have not been sent Create REQ, set the variables for Buffer head and input Create REQ (Command ID, Input parameters, CREATE_REQ, and buffer index) to the global variables of the REQ. Do not include via position in the input parameter that you put in the global variables. Set the NOT_REQFIN to TRUE.
   (c) After sending the data of (b), if the REQFIN_ACK and the buffer index corresponding to ACK is received, input Create REQ (Command ID, via position, CREATE_REQ, and buffer index) to the global variables of the REQ.
   (d) If Create/Active/Done ACK and corresponding buffer index to ACK are received, output Busy and the state changes to BUSY state. If Done ACK is received, the processing in BUSY state is further skipped.
   (e) If Create/Active/Done ACK and corresponding buffer index to ACK are not received, send again by inputting Create REQ to global variables for REQ again.
   (f) If Error ID ACK and corresponding buffer index to ACK are received, output Error and ErrorID and the state changes to ERROR state.

FRC_MoveCircularAbsolute

This FB executes a circular motion to a destination position specified in Cartesian representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position in Cartesian representation
ViaPos	FRC_POS_T	InOut	Via position in Cartesian representation
Velocity	UINT	Input	Program speed (mm/sec)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_MoveCircularRelative

This FB executes an incremental circular motion to a destination position specified in Cartesian representation.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
DestPos	FRC_POS_T	InOut	Destination position(incremental) in Cartesian representation
ViaPos	FRC_POS_T	InOut	Via position (incremental) in Cartesian representation
Velocity	UINT	Input	Program speed (mm/sec)
Acceleration	USINT	Input	Acceleration value (%)
Buffer mode	DINT	Input	Not used in this version
Termination Type	DINT	Input	Termination Type
CNT Value	DINT	Input	CNT value for CNT termination type. Ignored for FINE termination type.
Active	BOOL	Output	The motion corresponding to this FB has been executed.

A.5.4 Move Add Instruction

These are the FBs that add motion option to a TP Move Instruction. It is necessary to divide REQ like Circular Move Instruction FBs. The outputs of these FBs match the outputs of the Move Instruction FB since it corresponds to the buffer index of Move Instruction FB.
The process in DORMANT/RESET/ABORT state is the same as the process described in A.5.2.

1. EXECUTING state
   (a) If the Robot controller status is invalid such as TP enabled, the state changes to RESET state.
   (b) If the other FB have not been sent Create REQ, set the variables for Buffer head and input Create REQ (Command ID, Input parameters, CREATE_REQ, and buffer index) to the global variables of the REQ. Set the NOT_REQFIN to TRUE.
   (c) After sending the data of (b), if the REQFIN_ACK and the buffer index corresponding to ACK is received, output Busy and the state changes to BUSY state.
   (d) If Error ID ACK and corresponding buffer index to ACK are received, output Error and ErrorID and the state changes to ERROR state.

2. BUSY state
   If Active/Done ACK and corresponding buffer index to ACK are received, output Active and the state changes to ACTIVE state.

3. ACTIVE state
   If Reset REQ has been sent and Reset ACK and corresponding buffer index to ACK are received, the state changes to DONE state.

4. DONE state
   Output Done. If there is no Execute input, the state changes to the RESET state. (Set the variables for Buffer tail in the Move Instruction FB.)

5. ERROR state
   If Reset ACK and corresponding buffer index to ACK are received, set the state changes to the RESET state. (Set the variables for Buffer tail in the Move Instruction FB.)

FRC_TriggerAlongPath

This FB adds a Time Before, Time After, or Distance Before motion option to a Move Instruction FB.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
TriggerType	USINT	Input	Trigger Type
Distance	UINT	Input	Distance before value (0.01 mm unit)
Delay	UINT	Input	Time before value (ms)

Parameter Name	Data Type	Usage	Description
PortType	USINT	Input	Port Type
PortNum	UINT	Input	Digital Port Number
Value	BOOL	Input	Digital Port value
Active	BOOL	Output	The motion corresponding to this FB has been executed.

FRC_AddMotionOption

This FB adds Wrist Joint, Minimum Rotation, Offset Position Register or Offset Vision Register motion option to a Move Instruction FB.

In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
WristJoint	BOOL	Input	Enable Wrist Joint motion
MROT	BOOL	Input	Enable Minimum Rotation motion
OffsetPR	BOOL	Input	Enable Offset Position Register
PRNumber	UINT	Input	Position Register number
OffsetVR	BOOL	Input	Enable Offset Vision Register
VRNumber	UINT	Input	Vision Register number
Active	BOOL	Output	The motion corresponding to this FB has been executed.

A.5.5 Other Instruction

These FBs add the non-motion TP instruction to the TP line. The FBs other than Wait instruction have no ACTIVE state.

The process in the state other than ACTIVE is the same as the process described in section A.5.2.

2. BUSY state

   If Create REQ of the other FB has been sent, the process is not executed in BUSY state.

   (a) If Done REQ has not been sent, input DONE_REQ and buffer Index to the global variables of the REQ.

   (b) If Done REQ has been sent and Done ACK and corresponding buffer index to ACK are received, input RESET_REQ and buffer Index to the global variables of the REQ.

   (c) If Done REQ has been sent and Done ACK and corresponding buffer index to ACK are not received, retry to send Done REQ.

   (d) If Reset REQ has been sent and Reset ACK and corresponding buffer index to ACK are received, the state changes to DONE state.

   (e) If Reset REQ has been sent and Reset ACK and corresponding buffer index to ACK are not received, retry to send Reset REQ.

FRC_SetUFrame

This FB sets the current active user frame number.

In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
UframeNum	USINT	Input	User frame number to set

FRC_SetUTool

This FB sets the current active user tool number.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
UtoolNum	USINT	Input	User tool number to set

FRC_SetPayload

This FB sets the current active payload number.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
PayloadNum	USINT	Input	Payload number to set

FRC_Call

This FB executes call instruction.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
Program_name	STRING	Input	TP program Name (Up to 36 characters)
Name_len	USINT	Input	Length of TP program Name (Up to 36)

FRC_WaitForDIN

This FB causes the TP program execution wait until the Robot controller DI reaches the specified value.
The process in state is the same as the process described in section A.5.2.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
PortNum	DINT	Input	Index value of DI
Value	BOOL	Input	ON/OFF of DI
Active	BOOL	Output	The Robot controller is waiting for the DIN to be set to the value specified. Program execution is paused

FRC_WaitTime

This FB causes the TP program execution wait until the specified amount of time.
The process in state is the same as the process described in section A.5.2.
In addition to the parameters described in section A.5.1, the following parameters are defined.

Parameter Name	Data Type	Usage	Description
WaitTime	UINT	Input	Wait time (10ms unit)
Active	BOOL	Output	Program execution is paused.

INDEX

REVISION RECORD

Edition	Date	Contents
01	Sep., 2020	